

# Underwater Wireless Sensor Network Deployment Based on Chaotic Particle Swarm Optimization Algorithm

<http://dx.doi.org/10.3991/ijoe.v11i1.4144>

Shaojuan Su, Tianlin Wang  
Dalian Maritime University, Dalian, China

**Abstract**—The application of particle swarm optimization algorithm in underwater wireless sensor node deployment strategy was studied. The chaotic particle swarm optimization algorithm was proposed. Set up a function variance to determine whether the particles enter the premature state. Then chaos searched and took the chaotic search optimal solution vector as the global optimization solution to guide the particles quickly jump out of local optimum, accelerate the convergence speed and to overcome the premature problem of standard particle swarm algorithm brought. The network coverage was the fitness function to optimize the deployment of nodes question. The chaotic particle swarm optimization algorithm was used to wireless sensor network node deployment strategy. The experimental results showed that this method could effectively improve the convergence speed and increase network coverage.

**Index Terms**—underwater wireless sensor network (WWSN), chaotic, particle swarm optimization, network coverage

## I. INTRODUCTION

Because the random deployment has the advantages of low cost and easy to realize, application of wireless sensors are mostly the random deployment. For wireless sensor networks with random deployment optimization goal was to reduce the blind spot coverage, to reduce the number of sensor nodes, to improve the coverage network etc. These problems belong to NP (Nondeterministic Polynomial) problem, it is difficult to find a better solution by using traditional optimization methods [1,2].

Although it was proved that the particle swarm optimization algorithm could effectively achieve the wireless sensor network deployment optimization, but the standard particle swarm algorithm is easy to fall into the "premature" to limit the search range of particles. The method was improved so that it can quickly realize the global optimization and then effectively avoid the local optimal solution.

## II. THE UNDERWATER WIRELESS SENSOR NETWORK LAYOUT MODEL

### A. The Node Model

Boolean sensing model is one of the most simple and most commonly model. In the two-dimensional plane, coverage of sensor nodes were seen as a node as the center,  $R_S$  as the circular radius shown in Figure 1. In three-dimensional space, the 0-1 model is a sphere which takes a node as the center and  $R_S$  as the radius. In the theoretic

cal analysis, all points in the disc or sphere range are regarded as that can be sensed by the node [3]. Such as two-dimensional plane, a node S coordinates  $(x, y)$ , the target P coordinates  $(x_p, y_p)$ ,  $d(s, p)$  are the euclidean distance between the node S to the point P. That is  $d(s, p) = \sqrt{(x_s - x_p)^2 + (y_s - y_p)^2}$ . In the 0-1 model, the occurrence probability of node S detection to P for:

$$c(s, p) = \begin{cases} 1, & d(s, p) \leq R_S \\ 0, & d(s, p) > R_S \end{cases}$$

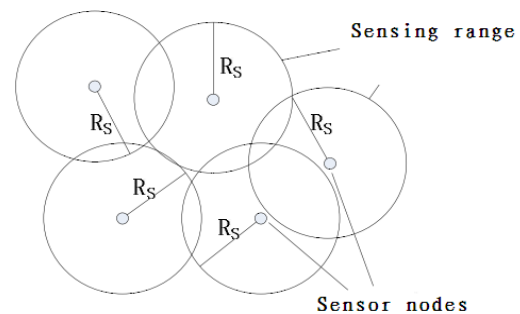


Figure 1. Sensing range

Covering area of the nodes was a key content of this paper. Network coverage often appeared the phenomenon of sensor nodes coverage overlap shown in Figure 2. It is difficult to calculate by mathematical method. The grid method for approximate estimation coverage area was used in the paper. In order to improve the calculation accuracy, the grid size is 0.1 m\*0.1 m. When the point was in the sensor perception range, namely all point in the grid would be perceived shown in figure 3. Such as point (22, 47) in the sensor range, then all the points in the area was considered to be perceived. While the point (43, 38) is not in its sensing range, then all the points in the area was considered not to be perceived. For the whole nodes, only when the target point was not in the sensor range of all nodes, the point was not covered [4]. Therefore the probability of coverage of all sensors on the point was shown in formula 1.

$$c(x_p, y_p) = 1 - \prod_{i=1}^N (1 - c_i(x_p, y_p)). \quad (1)$$

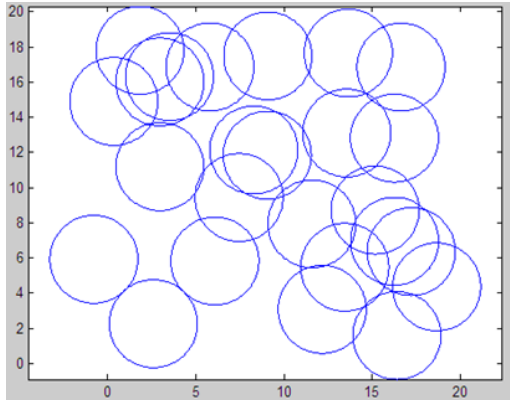


Figure 2. Coverage overlap phenomenon

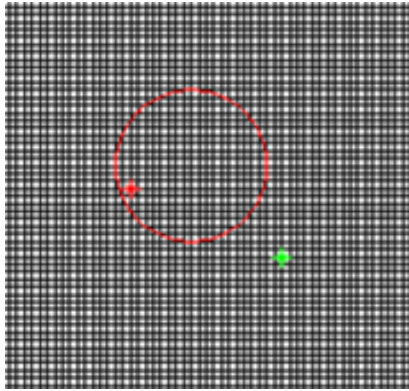


Figure 3. The sensor perception range

#### A. The Target Function

The wireless sensor network nodes deployment was converted to an optimization problem. The goal of optimization was to make the network coverage largest. The fitness function  $f(x)$  is the coverage rate of the corresponding strategies shown in formula (2). How to effectively deploy sensors to make the coverage largest and calculate the coverage efficiency in the fixed sensor nodes number cases were focused on in the paper [5].

$$\text{Coverage} = \frac{\text{Thenode of the networkcoverage area}}{\text{The ratio of the target surveillane area}}$$

$$\text{efficiency} = \frac{\text{The network covers}}{\text{A singlenode sensingarea} \times \text{The total number of nodes}}$$

$$f(x, y) = \sigma = \frac{S_{\text{cover}}}{S_A} = \frac{\sum_{x=0}^m \sum_{y=0}^n (x_p, y_p)}{m \times n} \quad (2)$$

### III. COVERAGE OPTIMIZATION CALCULATION BASED ON CHAOS PARTICLE SWARM

#### A. Particle Swarm Optimization Algorithm

Particle swarm optimization (PSO) mathematical expression is followed. Assuming a  $D$  dimensional search space,  $m$  particles form a group. The position of the particle  $i$  in the search space is  $x_i$  and vector is  $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})^T$  and the flying speed is  $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})^T$ . The individual extremum of particle  $i$  is  $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})^T$  and the global extremum is  $g_g = (p_{g1}, p_{g2}, \dots, p_{gD})^T$ . In the formula  $g$  is the number of fitness optimum in the group [6,7].

The particles are iterated operation according to the following formulae (3) and (4) and schematic diagram is shown in Figure 4.

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 \cdot r_1 \cdot (p_{id}^k - x_{id}^k) + c_2 \cdot r_2 \cdot (g_{gd}^k - x_{id}^k) \quad (3)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (4)$$

$\omega$  is non negative number, called the inertia weight (inertia weight), Its role is to adjust the algorithm of global and local search ability of balance.  $i = 1, 2, \dots, m$   $d = 1, 2, \dots, D$ . Acceleration constant  $C_1$  and  $C_2$  (acceleration constants) is a non negative;  $r_1$  And  $r_2$  is a random number between  $[0, 1]$ .  $v_{id} \in [-v_{\max}, v_{\max}]$ ;  $v_{\max}$  is a constant, setting by the user;  $k = 1, 2, \dots$  is the number of iterations.

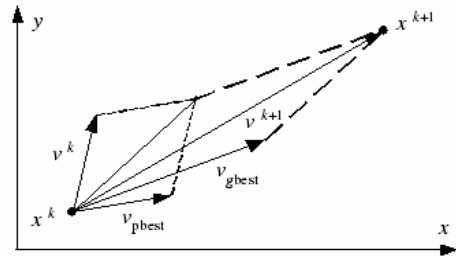


Figure 4. A schematic diagram of the search in the PSO algorithm

#### B. The Improvement of Pso

Although the improvement of PSO can solve the particle swarm algorithm optimization problem effectively, because of it's easy to fall into premature convergence and local minimum. There are the phenomena in the optimization process:

(1) When the particles reach the local optimal solution position of the discrete space premature convergence, the rest quickly close to the position. After a period of time a large number of particles will stay in this position. The particle swarm is aggregation.

(2) When a large number of particles is in the position of the local optimal solution, switching operation sequence of these particles do not occur exchange operation,

so that these particles stop at the local optimal solution and lost the ability to continue the search.

According to the above phenomenon, the chaos search and particle swarm algorithm were combined with to handle the premature convergence problem. First particle swarm optimization operation run until the particle swarm algorithm was in premature convergence, and then run the chaos search. After rearch N (a constant) time, the optimal chaotic search solution vector was took as the global optimal particle swarm optimization solution vector to guide the particles quickly jump out of local optimum and accelerate the convergence speed. This method can effectively avoid the blindness of adaptive particle swarm optimization algorithm. Under the guidance of the new optimal position, particle swarm optimization will do more refined local search to improve the accuracy of the algorithm [8].

In order to determine whether particles enter the premature state, a function variance  $\sigma^2$  was set up in formula (5).

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N \left( \frac{f_i - f_{avg}}{f} \right)^2 \quad (5)$$

$f_i$  is the fitness of the  $i$  particles,  $f_{avg}$  is the average fitness of particle swarm,  $\sigma^2$  is the particle swarm colony fitness variance, N is the number of particles.  $f$  is normalized scaling factor, Its role is to limit the size of the  $\sigma^2$ . Values are as follows:

$$f = \max \{1, \max \{|f_i - f_{avg}|\}\}$$

The variance  $\sigma^2$  of the population's fitness reflects all the particles "aggregation" degree shown as formula (5). The  $\sigma^2$  is smaller the particle swarm "gather" extent is greater. If the algorithm does not meet the end condition, the "aggregation" will enable the group to lose diversity in premature state. So when  $\sigma^2 < C$  (C is a given constant), premature treatment should be done.

If the premature phenomenon appeared, chaos optimize in the local area is done which take  $p_g = (p_{g1}, p_{g2}, \dots, p_{gD})$  as the two carrier optimal solution to guide the particles quickly jump out of local optimum and accelerate the convergence speed.

#### IV. THE EXAMPLE

In order to verify the validity of the algorithm, two kinds of models were adopted to simulate experiment using Matlab7.0. The standard PSO algorithm and chaotic particle swarm optimization algorithm were used to optimize the deployment of wireless sensor network node under water.

Assumptions reserve on the 30 sensor nodes with sensing radius  $r=10$  m in the rectangular area of the 100 m\*100 m.

Where, m and n are the length and the width of the target area.

The number of population is 20. Maximum number of iterations was proposed for 500 times. The inertia weight was used linear descent method. Other parameters were as follows

$$v_{max} = 20, \omega_{max} = 1.2, \omega_{min} = 0.3, c_1 = c_2 = 2$$

If the random deployment of sensor nodes, there will be a large number of nodes repeat coverage problem, as shown in Figure 5, the coverage rate is only 59.24%.

Figure 6 and figure 7 are the graph simulation. Circles represented the node sensing area. The optimize result by standard PSO algorithm was shown in Figure 6 and the result by the swarm algorithm improved shown in Figure 7.

The Coverage rate is 73.71% using standard PSO algorithm and 85.74% using Chaotic particle swarm optimization deployment.

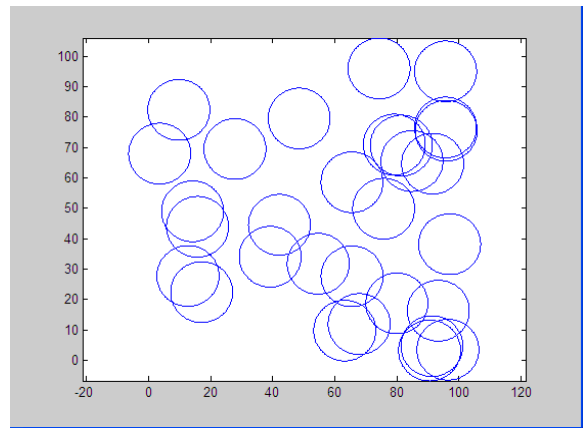


Figure 5. Random deployment of sensor nodes

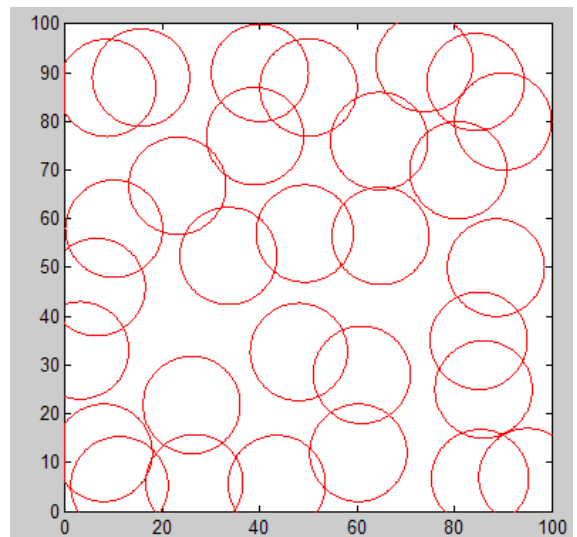


Figure 6. The optimize result by standard PSO algorithm

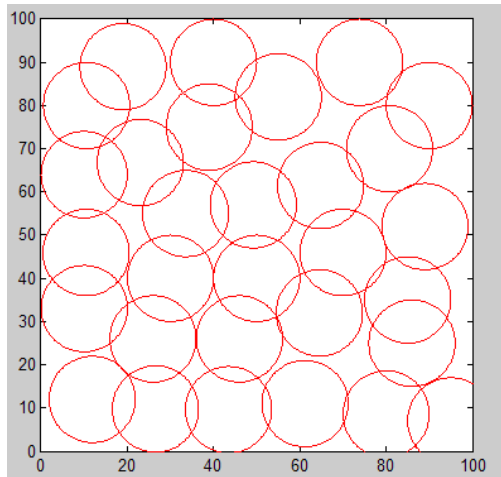


Figure 7. The result by the chaotic swarm algorithm

The improved algorithm coverage compared to the standard PSO algorithm improves 12.03%. Network coverage rate increases by 26.5% comparing with the initial coverage rate. Thus the use of chaotic particle swarm algorithm was more advantage to improve the coverage rate.

Because the chaotic particle swarm algorithm had good ability to avoid the local extremum and obtain the global extremum. Better coverage can be obtained by using the hybrid algorithm and the results were more uniform deployment and blind spot less.

#### V. CONCLUSION

The uncertainty node deployment method for underwater wireless sensor networks is easy to cause great redundancy and uneven distribution. According to characteristics of randomly deployed wireless sensor networks under water, the chaotic particle swarm algorithm is adopted for wireless sensor network node deployment strategy optimization. The experimental results show that the method

can be effective to improve the convergence speed and the network coverage rate.

#### REFERENCES

- [1] J.-P. Watson, H. Greenberg, W.E. Hart. "A multiple-objective analysis of sensor placement optimization in water networks," In: *Proc. World Water and Environment Resources Congress*, Salt Lake City, June 2004.
- [2] H. Gonzalez-Banos, J.C. Latombe. "A randomized artgallery algorithm for sensor placement," In: *Proc. 17th ACM Symposium on Computational Geometry (SoCG'01)*, Boston, MA, June 2001.
- [3] W. Xun. "Deployment optimization strategy of wireless sensor network nodes based on artificial intelligence hybrid algorithm," Nanjing Forestry University, 2013, pp. 31-32.
- [4] A.P. Liu, Z. Liu, Y.S. Luo. "A research on the coverage and connectivity problem in underwater wireless sensor networks," *Chinese J. Sensor & Actuators*, 2009, vol. 22, no. 1, pp.116-120.
- [5] C.F. Huang, Y.C. Tseng. "A survey of solutions to the coverage problems in wireless sensor networks," *J. Internet Tech.*, 2005, vol. 6, no. 1, pp. 1-8.
- [6] R. Eberhart, J. Kennedy. "A new optimizer using particle swarm theory," *Proc. 16th Int. Symposium on Micro Machine and Human Science*. Nagoya, Japan: IEEE Press, 1995, pp. 39-43. <http://dx.doi.org/10.1109/MHS.1995.494215>
- [7] Y.H. Shi, R.C. Eberhart "A modified particle swarm optimizer," *Proc. IEEE Int. Conf. Evolutionary Computation*. Piscataway, USA: IEEE Press, 1998, pp. 69-73.
- [8] S.J. Su. "Uncertain analytical method research and application on coast dry bulk cargo ships transportation," Wuhan University of Technology, 2007, pp. 68-99.

#### AUTHORS

**Shaojuan Su** is with Transportation Equipments and Ocean Engineering College, Dalian Maritime University (e-mail: katie306@163.com).

**Tianlin Wang** is with Transportation Equipments and Ocean Engineering College, Dalian Maritime University (e-mail: Second\_engineering@163.com).

This work is supported by the research fund of NFSC with the grant number: 51209027, and the Fundamental Research Funds for the Central Universities (3132014068). Submitted 14 October 2014. Published as submitted by the authors 25 January 2015.