

Routing Optimization for Wireless Sensor Network based on Cloud Adaptive Particle Swarm Optimization Algorithm

Xu Bao

School of Traffic Engineering, Huaiyin Institute of Technology, Huai'an 223003, Jiangsu, China
e-mail: baoxubaoxu@126.com

Abstract

One of the most important targets of routing algorithm for Wireless Sensor Network (WSN) is to prolong the network lifetime. Aimed at the features of WSN, a new routing optimization approach based on cloud adaptive particle swarm optimization algorithm is put forward in this paper. All paths appear at the same time in one round are fused in one particle, and the coding rule of particle is set down. The particle itself is defined as its positions, the number of replaceable relay nodes in paths is defined as the velocity of particle. Cloud algorithm is used to optimize the inertia weight of particle. Optimize rules are laid out, and both residual energy of nodes and variance of all paths' length are considered in objective function. Simulations find out the best value of balance factor in objective function, also prove that this approach can control the energy consumption of network, enhance the viability of nodes, and prolong the lifetime of network.

Keywords: wireless sensor network (WSN), routing, cloud adaptive particle swarm optimization algorithm

Copyright © 2013 Universitas Ahmad Dahlan. All rights reserved.

1. Introduction

Detect traffic incident promptly and solve it correctly are the essential of improving traffic safety and raising transportation efficiency. As a new generation of adhoc network, WSN is applied in traffic incident detection [1-3], its structure is showed in Figure 1.

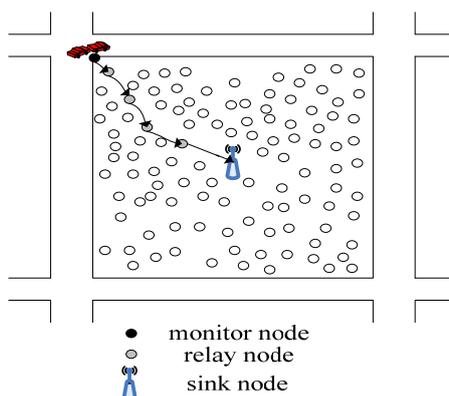


Figure 1. Structure of WSN in Traffic Incident Detection

Once traffic incident occurs, monitor nodes deployed on the wayside send incident information to sink node with the help of several relay nodes deployed in particular area. But the energy of relay nodes is limited, and the dead of single node will accelerate the dead of whole network [4-6], so how to construct efficient and energy-saving routing algorithm to prolong the lifetime of nodes and network is the key to traffic incident detection.

Cloud adaptive particle swarm optimization algorithm(CAPSO) [7] combines cloud theory [8, 9] and particle swarm algorithm [10], divides particles in one swarm into three classes based on the size of particle fitness value, and takes cloud theory adjust the inertia weight of middling particles, so it can both raise the velocity of convergence speed and secure the diversity of swarm. Reference [11] optimizes the path between cluster heads utilizing CAPSO in order to cut down the energy consumption of cluster heads and prolong the lifetime of network, but its objective function only considers residual energy of nodes. As the balance of energy consumption of every path has important influence to network lifetime, the algorithm in reference [11] has room for improvement.

This paper puts forward a new routing algorithm for WSN based on CAPSO, constructs the objective function considering both nodes' residual energy and balance of paths' energy consumption, fuses all paths appear at the same time in one round as a particle, defines the position and velocity of particle, and optimizes particle position by cloud theory. This algorithm can cut down energy consumption of WSN and prolong the lifetime of network in traffic incident detection and information transmission.

2. Cloud Adaptive Particle Swarm Optimization Algorithm

Supposing that there are D paths in WSN at the same time, that is the particle swarm exists in D dimensional searching space. There are N particles in a swarm, in the k th iteration, the i th particle's position is $X_i^k = (X_{i1}^k, X_{i2}^k, \dots, X_{iD}^k)$, its best position in history is $pbest_i^k = (P_{i1}^k, P_{i2}^k, \dots, P_{iD}^k)$, its velocity is $V_i^k = (V_{i1}^k, V_{i2}^k, \dots, V_{iD}^k)$, and the best particle position as a whole is $gbest = (p_{g1}^k, p_{g2}^k, \dots, p_{gD}^k)$. Every particle adjusts its position as follows:

$$V_{id}^{k+1} = \omega V_{id}^k + c_1 r_1 (P_{id}^k - X_{id}^k) + c_2 r_2 (P_{gd}^k - X_{id}^k) \quad (1)$$

$$X_{id}^{k+1} = X_{id}^k + V_{id}^{k+1} \quad (2)$$

Where ω is the inertia weight; c_1 and c_2 are nonnegative accelerate constant; r_1 and r_2 are random number between 0 and 1.

Adjust rule of particle velocity is as follows:

$$\begin{cases} V_{id} = V_{\max}, & V_{id} > V_{\max} \\ V_{id} = -V_{\max}, & V_{id} \leq -V_{\max} \end{cases} \quad (3)$$

Supposing in the k th iteration, fitness of X_i is f_i^k , average fitness of all particles in particle swarm is $f_{avg}^k = \frac{1}{N} \sum_{i=1}^N f_i^k$, average fitness of particles whose fitness are superior to f_{avg}^k is f'_{avg} , average fitness of particles whose fitness are inferior to f_{avg}^k is f''_{avg} , the best fitness in particle swarm is f_{best}^k , CAPSO determines the value of ω by the size of particle fitness:

1. If $f_i^k > f'_{avg}$, the fitness of X_i is close to f_{best}^k , so in order to accelerate local convergence, ω is assigned as 0.4;
2. If $f''_{avg} < f_i^k < f'_{avg}$, the fitness of X_i is not good, needs to optimize the value of ω .

Firstly, determines the expected value:

$$E_x = f_{best}^k \quad (4)$$

Secondly, determines the entropy of particle:

$$E_n = (f'_{avg} - f_{best}^k) / c_1 \quad (5)$$

Thirdly, determines the hyper entropy of particle:

$$H_e = E_n / c_2 \quad (6)$$

Fourthly, determines the value of ω :

$$\omega = 0.9 - 0.5e^{-\frac{(f_i^k - E_n)^2}{2(E_n)^2}} \quad (7)$$

$$E'_n = \text{normrnd}(E_n, H_e) \quad (8)$$

normrnd is the generator of normal random number.

3. If $f_i^k < f_{avg}^n$, the fitness of X_i is poor, in order to accelerate global search, ω is assigned as 0.9.

3. Network Model

Suppose that there is a square monitor area, D monitor nodes are deployed on the boundary of the square, N relay nodes are deployed in the square, sink node is deployed in the center of the square, and its ID is $D+N+1$, monitor nodes send information to sink node by prompt relay nodes. Communicate radius of all nodes are adjustable.

The distance between sending node and receiving node d has a threshold of d_0 . Sending node's energy consumption is shown in (9) [12]:

$$E_{tx} = \begin{cases} E_{elec} \cdot l + E_{amp} \cdot l \cdot d^2, & d \leq d_0 \\ E_{elec} \cdot l + E_{amp} \cdot l \cdot d^4, & d > d_0 \end{cases} \quad (9)$$

Receiving node's energy consumption is shown in (10):

$$E_{rx} = E_{elec} \cdot l \quad (10)$$

$$d_0 = \sqrt{\frac{E_{fs}}{E_{amp}}} \quad (11)$$

Where l is the *bits* of message; E_{elec} is the energy consumption of sending/receiving 1 *bit* message; E_{fs} and E_{amp} are amplifier's energy consumption of transmitting 1 *bit* message in unit area.

4. Routing Optimization for WSN Based on CAPSO

4.1. Routing Constraint Condition

In order to balance energy consumption of nodes and simplify the amount of calculation, this paper supposes that path constructing satisfies two conditions:

1. All monitor nodes can't act as relay nodes;
2. Every relay node alive can only exit in one path or in sleep state.

Supposing that there are M living nodes in paths one round, modeling above two constraint conditions, we can get constraint model as follows:

$$x_{kij} = \begin{cases} 1 & \text{the } k\text{th path contains arc } (i, j); i = 1, 2, 3 \dots M; j = 1, 2, 3 \dots M \\ 0 & \text{else} \end{cases} \quad (12)$$

$$y_{ki} = \begin{cases} 1 & \text{the } k\text{th path contains node } i; i = 1, 2, 3 \dots M \\ 0 & \text{else} \end{cases} \quad (13)$$

$$\sum_{k=1}^D y_{ki} = 1 \quad i = 1, 2, 3 \dots M \tag{14}$$

$$\sum_{i=1}^M x_{kij} = \begin{cases} 0 \\ y_{kj} \end{cases} \tag{15}$$

$$\sum_{j=1}^M x_{kij} = \begin{cases} 0 & i \in T; k = 1, 2 \dots D \\ y_{ki} & else \end{cases} \tag{16}$$

Where, $arc(i, j)$ represents the arc from starting node i to end node j . Equation (15) represents there is only one starting node connecting to end node in every arc, and all starting node of every path can't be the end nodes of any arc. Equation (16) represents that only one end node connecting to starting node in every arc, and all end nodes of every path can't be the starting nodes of any arc. S is the set of starting nodes of paths, T is the set of end node of paths, just sink node.

4.2. Particle and its Position

In order to optimize routing for WSN utilizing CAPSO, particle and its position must be defined firstly. Because of the particularity of routing, traditional definition of particle position can't satisfy the optimization needs. This paper fuses all path occur at the same time into one particle, and takes itself as its position.

Supposing that a particle is a round number S with the length of L , which can be divided into D parts. The first number of every part is the ID of different monitor node, the last number of every part is the ID of sink node, and the middle numbers of every part are the ID of relay nodes in every path, so every part of one particle represents one path from any monitor node to sink node. For example, Figure 2 is a particle and its position, there are 3 paths appear at the same time, so the particle is divided into 3 parts, the first part of the particle represents the path from monitor node with the ID 1 to sink node with the ID 21 containing relay nodes with ID of 5, 7, and 10.

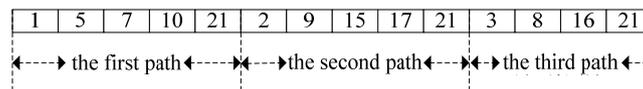


Figure 2. Particle and its Position

4.3. Particle Velocity

Velocity is used to improve the position of particle, so particle velocity V contains D dimensional variables. Every path in X_i adjusts itself in the k th iteration, if the fitness of new X_i is the best one in its history, X_i^k is taken as $pbest_i^k$. The best particle in the particle swarm is taken as $gbest^k$. V_{id}^k represents the number of replaceable relay nodes in X_{id}^k (the d th path), which is influenced by both P_{id}^k and P_{gd}^k , let q_{id}^k represents the numbers of different relay nodes between X_{id}^k and P_{id}^k in corresponding place in the d th path, let g_{id}^k represents the numbers of different relay nodes between X_{id}^k and P_{gd}^k in corresponding place in the d th path, so in the $(k + 1)$ th iteration, V_{id}^{k+1} can be computed as follows:

$$V_{id}^{k+1} = \omega V_{id}^k + c_1 r_1 q_{id}^k + c_2 r_2 g_{id}^k \tag{17}$$

Figure 3 represents the best position of X_i in the k th iteration.

Figure 4 represents the best position in particle swarm in the k th iteration.

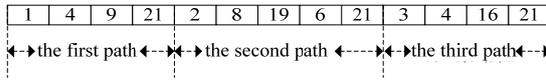


Figure 3. The Best Position of X_i in History

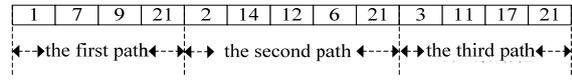


Figure 4. Particle and its Position

In the first path, except for monitor node and sink node, there are two different relay nodes between X_{i1} and P_{i1} , and also two different relay nodes between X_{i1} and P_{g1} . So the number of replaceable relay nodes in the $(k + 1)$ th iteration of X_{i1} is $V_{i1}^{k+1} = \omega V_{i1}^k + 2c_1r_1 + 2c_2r_2$.

4.4. Particle Optimization

Because of the process of optimization of X_i is the process of X_i be close to g_{best} continuously, in the $(k + 1)$ th iteration, this paper uses V_{id}^{k+1} different relay nodes from P_{gd}^k to replace corresponding nodes in X_{id}^k , and form a new particle X_i^{k+1} :

$$X_{id}^{k+1} = X_{id}^k \oplus (V_{id}^{k+1} \otimes P_{gd}^k) \tag{18}$$

$V_{id}^{k+1} \otimes P_{gd}^k$ represents V_{id}^{k+1} different relay nodes between X_{id}^k and P_{gd}^k in corresponding place. $X_{id}^k \oplus$ represents replace corresponding nodes in turn.

4.5. Objective Function

In X_i , if one path's length is longer or shorter than other's, the nodes in longer paths will die quicker. So in order to balance energy consumption of nodes and prolong the lifetime of network, the objective function must consider both nodes' residual energy and variance of all paths' length. In addition, because energy consumption of node in sleep state is less than it in work state, every living node is set in only one path or in sleep state. The objective function is as follows:

$$Z = \min f \tag{19}$$

$$f = \varepsilon f_1 + (1 - \varepsilon) f_2 \tag{20}$$

$$f_1 = E_0 - E_c \tag{21}$$

$$f_2 = \frac{\sum_{i=1}^D (S_i - \bar{S})^2}{M} \tag{22}$$

$$S_i = \sum_{j=1}^{D+N+1} \sum_{k=1}^{D+N+1} C_{ij} \cdot x_{kij} \tag{23}$$

$$\bar{S} = \frac{\sum_{i=1}^D S_i}{D} \tag{24}$$

E_0 is the initial energy of relay node, E_c is residual energy of relay node, ε is the balance factor, C_{ij} is square of the distance between node with ID i and node with ID j , S_i is the sum of path length square.

4.6. Steps of Optimization

The lifetime of network is divided into several rounds, in every round, the algorithm in this paper optimizes initial paths after several iterations based on CAPSO, and the steps of the algorithm are as follows:

Step 1: Initializes network, determines initial value of parameters, such as the initial energy, balance factor, and initial velocity. Lets round=1.

Step 2: Generates D paths, each path contains different starting node (monitor node) and sink node, on the basis, fuses D paths into one particle. Repeats this process to generate several particles and form a particle swarm.

Step 3: Starts iterations.

1. Lets iteration=1.

2. Computes the fitness of every particle in k th iteration by (20), determines q_i^k and g_i^k , uses cloud theory optimizing the inertia weight of every particle, computes V_i^{k+1} by (17), and adjusts X_i by (18). Lets iteration=iteration+1.

3. Judges the end condition of iteration is satisfied or not. If yes, saves g_{best} and transmitter information through paths in g_{best} . If no, moves to 2.

Step 4: Computes energy consumption of every node by (9) and (10), judges the end condition of stop of network is satisfied or not. If yes, stops working. If no, lets round=round+1, and moves to Step 2.

5. Simulations and Analysis

150 relay nodes are deployed in 100m×100m square area randomly, 4 monitor nodes are deployed on the apexes of the square, sink node is in the center of the square. Initial energy of every monitor node is 5J, initial energy of every relay node is 0.5J, and the energy of sink node is not limited. In every round, network generates 4 different paths to send information from 4 monitor nodes to sink node. The parameters' value in (9) and (10) are taken from paper [7], $c_1 = c_2 = 1.4926$, the variation range of velocity is [0, 3], there are 20 particles in one particle swarm, the number of iteration times in every round is 50, and simulation carries out 3600 rounds.

Figure 5 shows the comparison of residual energy of the network in RO (the algorithm put forward in this paper) and OA (the algorithm put forward in [6]) separately. In Figure 5, the gradient of RO with $\varepsilon = 0.7$ is the smallest one, and the gradient of OA is almost the same as gradient of RO with $\varepsilon = 0.3$ and $\varepsilon = 0.5$. This represents that energy consumption of RO with $\varepsilon = 0.7$ is the lowest one, so as it is more energy-saving than OA.

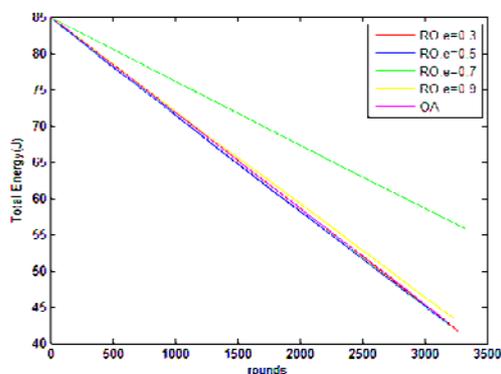


Figure 5. Comparison of Residual Energy in RO and OA

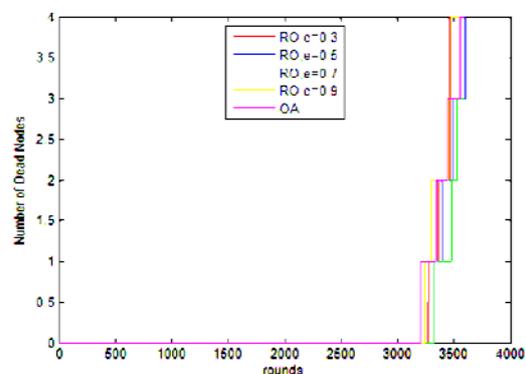


Figure 6. Comparison of Number of Dead Nodes in RO and OA

Figure 6 shows the number of dead nodes in RO and OA separately. In Figure 6, the network appears the first dead node in round 3326, and only has 3 dead nodes until round 3600 in RO with $\varepsilon = 0.7$; the network appears the first dead node in round 3206, and appears the fourth dead node in round 3551 in OA. This represents that RO with $\varepsilon = 0.7$ can balance nodes' residual energy and paths' length well.

6. Conclusion

This paper optimizes routing for WSN based on CAPSO, defines particle, its position and its velocity, designs the optimize rule, adjusts the inertia weight by cloud theory, and constructs object function fusing nodes' residual energy and variance of paths' length, thus cutting down the energy consumption and prolonging the lifetime of network. On the basis of theory model, this paper finds the best balance factor in objective function, and proofs the effectiveness of RO by simulations.

Acknowledgement

This work was Supported by JiangSu province college natural science foundation (No.12KJD510004), and Huai'an science and technology project (No. HAG2011047).

References

- [1] Hu QW. *A Framework for Traffic Accident Scene Investigation with GPS VRS, Road Database and Stereo Vision Integration*. Proceedings of 2011 International Workshop on Multi-Platform/Multi-Sensor Remote Sensing and Mapping. Xiamen. 2011; 1-5.
- [2] Li AZ. *Traffic Accident Characteristics Analysis Based on Fuzzy Clustering*. Proceedings of 2012 IEEE Symposium on Electrical and Electronics Engineering. Kuala Lumpur. 2012; 468-470.
- [3] De Sousa LD, Frery AC, Nakamura EF, Loureiro AAF. *Event Detection Framework for Wireless Sensor Networks Considering Data Anomaly*. Proceedings of 2012 IEEE Symposium on Computers and Communications. Cappadocia. 2012; 500-507.
- [4] Wu RH, He J, Li TJ, Shi HC. *Energy-efficient Coverage Hole Self-repair in Mobile Sensor Networks*. Proceedings of 2009 International Conference on New Trends in Information and Service Science. Beijing. 2009; 264-268.
- [5] Bao X, Ju YF. Distributed Coverage-Hole Repair Algorithm towards Nodes Failure in Wireless Sensor Networks. *Applied Mechanics and Materials*. 2012; 135-136: 464-469.
- [6] Hsieh KY, Sheu JP. *Hole Detection and Boundary Recognition in Wireless Sensor Networks*. Proceedings of IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications. Tokey. 2009; 72-76.
- [7] Wei XQ, Zhou YQ, Huang HJ, Luo DX. Cloud Adaptive Particle Swarm Optimization Algorithm. *Computer Engineering and Applications*. 2009; 45(76): 48-50.
- [8] Li DY, Meng HJ, Shi XM. Membership Clouds and Membership Cloud Generators. *Journal of Computer Research and Development*. 1995; 32(6): 15-20.
- [9] Zhang FZ, Fan YZ, Shen CZ, Li DY. Intelligent Control Based on Membership Cloud Generators. *Journal of Aeronautics*. 1998; 20(1): 89-92.
- [10] Kennedy J, Eberhart RC. *Particle Swarm Optimization*. Proceedings of IEEE Conference on Neural Networks IV. Piscataway. Perth. 1995; 1942-1948.
- [11] Xia KW, Gao F, Wu R, Liu NP, Zheng F. Optimal Wireless Sensor Network using Cloud Adaptive Particle-Swarm-Optimization Algorithm. *Control Theory & Applications*. 2011; 28(9): 1175-1178.
- [12] Heinzelman WR, Chandrakasan A, Balakrishnan H. *Energy-Efficient Communication Protocol for Wireless Microsensor Networks*. Proceedings of the 33rd Annual Hawaii International Conference on System Sciences. Hawaii. 2000; 1-10.