Sensor Distribution Optimization Based on Extending-tree in Sensor Network[1)]

WEN Ying-You[1] FENG Yong-Xin[2] WANG Guang-Xing[1]

[1](*Northeastern University, Shenyang* 110004)

[2](*Shneyang Ligong University, Shenyang* 110004)

(E-mail: remcall@gmail.com)

Abstract In a sensor network, reasonable distribution of sensor nodes will do much good to the improvement of sensorial ability. In a sensor network constructed by randomly scattering, a better sensing coverage can be achieved by topology adjustment utilizing mobility of some sensor nodes. To solve this problem, we define an extending-tree in the sensor network using Voronoi diagrams and Delaunay network. On this base, a new optimization algorithm of sensor node distribution based on genetic algorithm is designed, which provides a sound effective means to improve the sensorial ability of network. Simulation output shows that this algorithm can achieve an optimizing node distribution in the object area, by which sensorial ability of the whole sensor network can be improved at a relatively low cost.

Key words Sensor network, ad hoc, distribution optimization, genetic algorithm

1 Introduction

A wireless sensor network is a large-scale distributed network comprised of many small sensing devices equipped with memory, processor and short-range wireless communication[1,2]. Sensor node distribution plays a very important role in the improvement of network sensing ability. In a randomly scattering network, many redundant nodes are used to acquire better connectivity of network. In this case, sensing shadow and blind area are inevitable due to the uneven distribution of nodes. So it becomes a key point in network management to conduct a node distribution optimization at a lower cost. To solve this problem, we define an extending-tree in the sensor network by utilizing Voronoi diagrams and Delaunay network. On this base, a new optimization algorithm of sensor node distribution based on genetic algorithm is designed.

2 Extending-tree generation based on Delaunay triangulation

In a sensor network, there are two kinds of nodes, namely sensor node and sink node. Data collected by sensor nodes are usually sent to sink nodes and transmitted to observers finally. Therefore, better sensorial ability and coverage will be achieved at a low cost if an extending-tree at the root of the sink node is generated and used to perform an optimization operation.

2.1 Topology discovery and node localization in sensor network

Topology discovery is always the focus in the study of sensor networks[3~5]. Node localization is the basis of most applications, especially military applications. Locating with GPS (Global Positioning System) is usually adopted in sensor networks[6,7].

By utilizing topology discovery and sensor localization, information about node distribution and distance between sensors can be acquired easily, which provides a basis for node distribution optimization later. As this paper focuses mainly on distribution optimization, we will not discuss related processes of node localization and topology discovery below.

2.2 Extending-tree of sink node in sensor network

In the process of sensor networking by randomly scattering, many reluctant nodes are used to acquire better connectivity. Although these randomly distributed reluctant nodes provide the sensor network with flexibility and stability, they do degrade the sensibility because uneven distribution of nodes results in overlapped sensing fields, shadows and blind sensing area in the network. We adopte

1) Supported by National High-Tech Research and Development Plan (2003AA712032)
Received May 11, 2004; in revised form December 18, 2004

the concept of Voronoi diagram and Delaunay triangulation to describe topology of the sensor network, and define the extending-tree of the sink node on the basis of Delaunay triangulation.

It is usual to say that Voronoi diagram is the union of the boundaries between the Voronoi regions defined as the set of points at least as close to one site as to any other[8]. Let e be a point belonging to field E, then the Voronoi region of e can be defined as follows

$$VR(e) = \{p \in R^n | d(p,e) \leqslant d(p,e'),\ \forall e' \neq e,\ e' \in E\} \tag{1}$$

Then the Voronoi diagram of E, denoted as $VD(E)$, can be defined as the union of boundaries of the Voronoi regions:

$$VD(E) = \bigcup_i VR(e_i) \tag{2}$$

By drawing a line between any two points whose Voronoi domains touch, a set of triangles is obtained, known as the Delaunay triangulation.

As to the sensor network, by utilizing topology discovery and node localization, it is easy to construct Delaunay triangulation, from which we can get neighbor information about each node; also we can find two nearest nodes in network. Based on Delaunay triangulation of sensor network, we define extending-tree of a sensor node in network. Let S be a node set in field A, and then S can be described as follows

$$S = \{S_i(x_i, y_i) | S_i \in A\} \tag{3}$$

where $(x_, y_i)$ is the position of node S_i in network. Then we can get weighted undirected graph $G(v, e)$ from Delaunay triangulation corresponding to the nodes in S. Each edge in graph G is given a weight equal to the distance between two end nodes. Let $K = \{K_i(x_i, y_i) | K_i \notin A\}$ be a node set out of area A, and then we can define extending-tree of arbitrary node in S relative to K as follows:

$$T(S_i - > K) = \bigcup_i Path(S_i - > K_i), \quad K_i \in K \tag{4}$$

where $Path(S_i - > K_i)$ is the maximum span route in undirected graph G with end nodes S_i and K_i. In this path, distance between two nodes is not less than minimum distance between two nodes in any other path in graph G with end points S_i and K_i, and the number of nodes in this path is less than any other path that can be found in G. The maximum span route reflects a best-extended connection between two nodes. One thing to be mentioned is that maximum span route is not unique in a specific undirected graph.

To get a maximum span route in a sensor network with N nodes, we can use algorithm given below[9].

Algorithm 1.

Step 1. Topology discovery and node localization.

Step 2. Initializing, Voronoi diagram generation and Delaunay triangulation constructed.

Step 3. Generate a weighted undirected graph $G(v, e)$; each edge is assigned a weight value with distance between two end points.

Step 4. Delete all edges in G related with node S_i and generate a new graph G'.

Step 5. Minlength=the minimum length of edge in G', MaxLength=the maximum length of edge in G', step=(MaxLength-MinLength), value=Minlength+step.

Step 6. If step>threshold then

```
For each e ∈ e′
  if (weight>value)
    Add e to new weighted undirected graph G″
Step=step/2
If Findpath (from Si to Ki) in G″ then
  value=value+step
Else
  Value=value−step
Repeat Step 6
```

Step 7. Maximum span route export.

To generate extending-tree of a sink node, we should clip Delaunay triangulation to four parts according to Cartesian coordinate system with sink node as an origin. By importing node out of target area in each quadrant, we establish the maximum span route from sink node to these imported nodes. All nodes in these routes are selected as backbone, and the remnant nodes in each quadrant connect to the nearest node of backbone. After this, extending-tree of sink node is generated.

In fact, different imported node set out of target area will result in different extending-tree of sink node. Various node sets can be chosen according to the distribution of sensor nodes in a specific target area. The process of extending-tree generation is illustrated in Fig. 1.

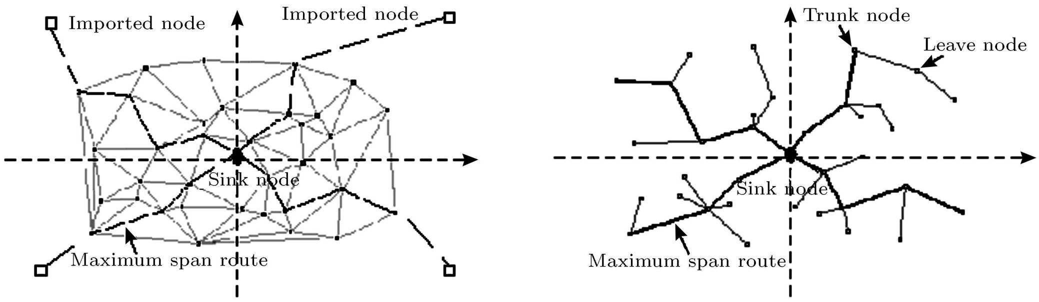


Fig. 1 The process of extending-tree generation based on delaunay triangulation

In a sensor network, nodes in backbone of extending-tree have a better dispensability than other leave nodes. They diminish the impact on sensing ability of network caused by overlapped coverage, so these nodes should be kept unchanged. While to those leave nodes in extending-tree, most of them are agminate and not far enough to avoid overlapped coverage of sensing domain, so these nodes should be given priority to adjust in the process of distribution optimization. To perform a satisfying node distribution optimization at a lower cost, we suppose a new algorithm based on genetic algorithm utilizing extending-tree generated above.

3 Sensor distribution optimization based on genetic algorithm

Acquiring a good distribution in specific target area belongs to optimization problem. Genetic algorithm gained much attention for its metrics to solve this kind of problem[10,11].

To simplify discuss of this problem, we import Sensing-Parameter as a description of sensing ability of node in different position. Providing the object area is digitalized as pixels and each pixel size is $\Delta x \times \Delta y$. Each pixel can be viewed as a sensor location, where a sensing-parameter is assigned according to the propagation model and actual surroundings. By doing so, the optimization problem of node distribution is converted to pursuing largest sensing range in target area by adjustment of sensor nodes′ positions.

3.1 Genetic representation of ristribution optimization problem in genetic algorithm

Genetic algorithm simulates biology evolution process[11]. As to sensor node distribution optimization problem, we adopted a new gene representation based on node coordinate. Each gene represents a node′s coordinate, which stands for its position in target area. Coordinates of node can be easily converted to longitude and latitude of node site by means of GPS. The process of code mapping is illustrated in Fig. 2.

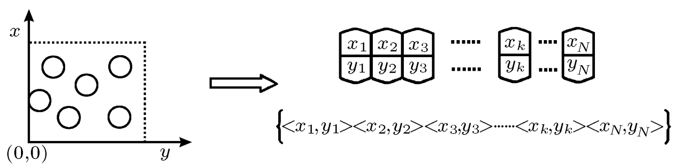


Fig. 2 Gene representation of node distribution optimization

By using representation described above, we set up the initial population of genetic algorithm to solve node distribution optimization. Given target area:

$$\{A(x,y): \quad 0 \leqslant x \leqslant Length \quad 0 \leqslant Width\} \tag{5}$$

The population is composed of T individuals. Each individual represents a distribution of sensor nodes. Let Te be the extending-tree, in which backbone node set is $L=\{L_i(xi,yi)|Li \notin Te,\ L_i \in A\}$, the number of trunk nodes is K, then the initial population is generated as follows:

$$\begin{cases} Pop_1=\{g_{11},g_{12},g_{13},\cdots g_{1j},\cdots g_{1N}\} \\ Pop_2=\{g_{21},g_{22},g_{23},\cdots g_{2j},\cdots g_{2N}\} \\ Pop_3=\{g_{31},g_{32},g_{33},\cdots g_{3j},\cdots g_{3N}\} \\ \qquad \vdots \\ Pop_i=\{g_{i1},g_{i2},g_{i3},\cdots g_{ij},\cdots g_{iN}\} \\ \qquad \vdots \\ Pop_T=\{g_{T1},g_{T2},g_{T3},\cdots g_{Tj},\cdots g_{TN}\} \end{cases} \tag{6}$$

where $g_{ij}=(x_{ij},y_{ij})(1 \leqslant i \leqslant T,\ 1 \leqslant j \leqslant N)$ is coordinate of node represented by gene j of individual i in population. When $j <= (N-K)$, g_{ij} represents backbone node of extending-tree; While $j > (N-K)$, g_{ij} represents leave node of extending-tree, and changed as follows:

$$\begin{cases} x_{ij}=x_{L_{N-j}}+\varepsilon, & \varepsilon \sim N(0,r) \\ y_{ij}=y_{L_{N-j}}+\varepsilon, & \varepsilon \sim N(0,r) \end{cases} \tag{7}$$

where ε obey Gauss Distribution with zero average; r is sensing radius of node in target area.

3.2 Crossover, mutation and selection in genetic algorithm

K individuals $(Pop_1, Pop_2, Pop_3, \cdots Pop_K)$ selected randomly from parent population to conduct crossover operation. The first operation is done with (Pop_1, Pop_2) as parents, and second time, take (Pop_2, Pop_3) as parents and so on. By doing this, $(K-1)$ offspring individuals are generated. Each parents$'$ gene are inherited by two offspring individuals. In our algorithm, we conducted crossover operation only to those leave nodes in extending-tree. To leave nodes, the operation can be described as follows.

Select k individuals randomly from parent population with probability P_c. Let (Pop_i, Pop_{i+1}) be the locus of g_{ik} and $g_{(i+1)k}$, then the coordinates represented by locus of offspring Chd_{ik} can be decided as follows:

$$\begin{cases} x_{Chd_{ik}}=(x_{Pop_{ik}}+x_{Pop_{(i+1)k}})/2+\psi_x, & \psi_x \sim N(0,x_{Pop_{(i+1)k}}-x_{Pop_{ik}}) \\ y_{Chd_{ik}}=(y_{Pop_{ik}}+y_{Pop_{(i+1)k}})/2+\psi_y, & \psi_y \sim N(0,y_{Pop_{(i+1)k}}-y_{Pop_{ik}}) \end{cases} \tag{8}$$

where $(x_{Pop_{ik}}, y_{Pop_{ik}})$, $(x_{Pop_{(i+1)k}}, y_{Pop_{(i+1)k}})$ are node coordinates of parents. ψ_x and ψ_y obey Gauss distribution with average value of zero.

Mutation will add new gene to population. In order to reduce change to population brought by mutation, we apply mutation operation only to individual generated by crossover. Gene is selected randomly from offspring individuals with probability of P_m. If original gene represents leave node whose coordinate is (v_1, v_2), then new gene will be $(v_1+\psi_1, v_2+\psi_2)$, where ψ_1, ψ_2 obey Gauss distribution of $N(0, r^2)$. As we should keep trunk nodes of extending-tree unchanged, so mutation doesn$'$t apply to these nodes too.

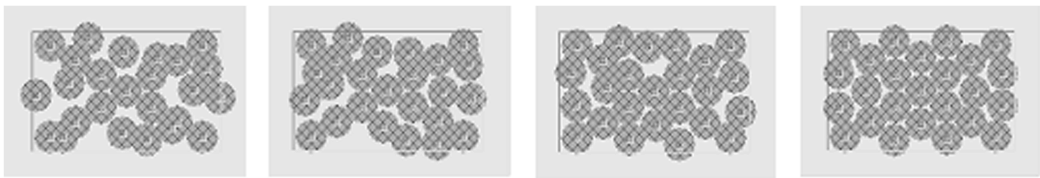
Selection is done in an extended space composed of all parent and offspring individuals. We conduct fitness changing before the process of selection so that the fitness between individuals will keep a reasonable difference, which will avoid algorithm converge too fast. At the same time, to avoid getting local optimal solution, we changed sampling space from which composed of parent and offspring individuals to offspring individuals only every few generation. This change is similar to artificial mutation in biology technology. By doing this, research space is redefined, which provides chances for algorithm to jump off local optimal solutions. Adjustment of step length can be done as follows.

$$Length(n)=\begin{cases} ([n/phase]+1)*step_1, & n \leqslant n_{phase} \\ step_2, & n > n_{phase} \end{cases} \tag{9}$$

4 Simulation and performance analysis

To demonstrate the validity of our algorithm, we conduct simulations of two kinds. In the first simulation, we conduct a node distribution optimization of 25 sensor nodes in a 2.3×1.4Km field. Target area is digitalized as $m \times n$ pixels and the pixel size is $10m \times 10m$, to make our simulation more convenient, we assign same sensing ability parameter to each pixel, which means the sensor node will have the same sensing ability of 200 meters wherever it will be in the target area.

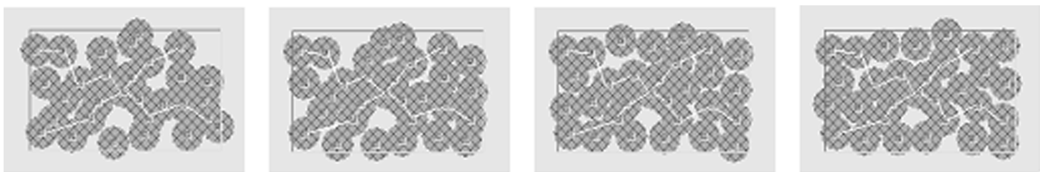
As we want to demonstrate the validity of optimization based on genetic algorithm, we don′t conduct extending-tree generation before the algorithm runs. Instead, we specify an extending-tree composed of nodes evenly distributed on the diagonal of the target area. By doing this, we can test if the algorithm can perform an effective distribution optimization. Algorithm runs 600 generations, in which P_c is 0.6, P_m is 0.315 and the population number is 45. Simulation output is given in Fig. 3.



(a) Initial distribution coverage 75.6% (b) Distribution after 100G coverage 80.32% (c) Distribution after 400G coverage 85.52% (d) Distribution after 600G coverage 89.03%

Fig. 3 Distribution optimization based on a specified extending-tree

In the second simulation, we conduct an integrate optimization on 30 sensor nodes in 2.3Km×1.4Km target area. The initial distribution of 30 sensors in target area is illustrated in Fig. 4 (a), where extending-tree of sink node is drawn using white lines. Other parameters of algorithm keep unchanged, output are shown in Fig. 4.

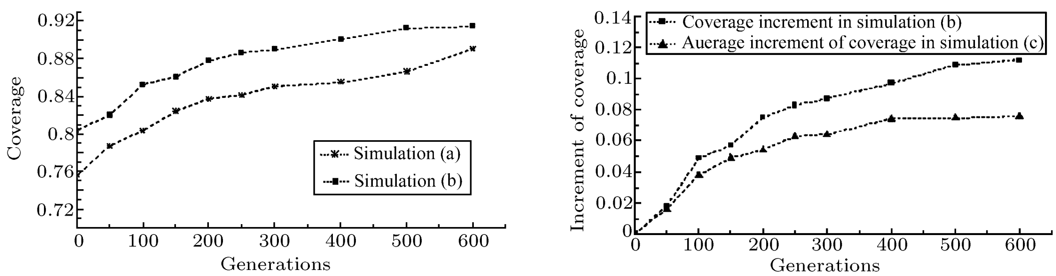


(a) Initial distribution coverage 80.35% (b) Distribution after 100G coverage 85.27% (c) Distribution after 400G coverage 90.07% (d) Distribution after 600C coverage 91.47%

Fig. 4 Distribution optimization based on extending-tree of sink node

During the last two simulations, sensing coverage over target area in different phases of algorithm is recorded, as shown in Fig. 5 (a).

We also conduct another simulation without using the extending-tree generated in last simulation. Instead, we choose some nodes randomly as stable nodes during optimization process. To compare with last simulation, the number of nodes (include sink node) selected is the same with the number of trunk nodes in last simulation, and the initial distribution is the same as last simulation too. We conduct 10 tests, each of which uses a group of nodes randomly selected as stable node set. We record the average increment of coverage during our simulation. The output is shown in Fig. 5 (b)



(a) (b)

Fig. 5 Simulation results and performance comparison

From the simulation given above, we can see that adjustment of node position based on genetic algorithm can acquire a perfect coverage over target area. Our optimization algorithm is able to generate extending-tree of sink node according to node distribution in target area, and improve the sensorial ability of whole network with lower cost by utilizing mobility of leave nodes on extending-tree. Performance comparison shows that optimization based on extending-tree can achieve a far better improvement than any other adjustments of node position can do, which provides a sound and accurate reference to topology management of sensor network.

5 Conclusion

In wireless sensor network, uneven distributions of node will causes sensing shadow and blind area. With the information acquired from topology discovery and node localization, sensing shadow and blind area can be eliminated at large by utilizing mobility of sensor node, which will do much help to improve the sensorial ability of whole network. To achieve this improvement, extending-tree of sink node is defined on the basis of Voronoi Diagram and Delaunay triangulation, and a node distribution optimization based on genetic algorithm is supposed. Simulation output shows that node distribution optimization based on extending-tree can achieve a satisfying coverage improvement at a relatively lower cost. Our algorithm can improve the sensorial ability of whole network, and it will provide a sound and accurate reference to topology adjustment in later management of network. Later research will focus mainly on the generation of extending-tree and low-cost distribution optimization.

References

1 Hill J, Szewczyk R, Woo A, Hollar S. System architecture directions for networked sensors. In: Proceedings of the 9th ACM International Conference on Architectural Support for Programming Languages and Operating Systems New York, USA: ACM Press ASPLOS-IX, 2000. 93~104

2 Pottie G, Kaiser W. Wireless sensor networks. *Communications of the ACM*, 2000, **43**(5): 51~58

3 Deb B, Bhatangar S, Nath B. A topology discovery algorithm for sensor networks with applications to network management. DCS Technical Report DCS-TR-441, Rutgers University, 2001

4 Roychoudhury R, Bandyopadhyay S, Paul K. A distributed mechanism for topology discovery in ad hoc wireless networks using mobile agents. In: Proceedings of IEEE First Annual Workshop on Mobile and Ad hoc Networking and Computing (MobiHoc), Piscataway, USA: IEEE Press, 2000. 145~146

5 Fan C, Liu M Y. A distributed monitoring mechanism for wireless sensor networks. In: Proceedings of the ACM Workshop on Wireless Security, ASPLOS-IX, 2002. 57~66

6 Joe Albowicz, Alvin Chen, Zhang Li-xia. Recursive position estimation in sensor networks. In: Proceedings of the International Conference on Network protocols (ICNP′01). California: IEEE Computer Society, 2001. 35~41

7 Bulusu N, Estrin D, Girod L. Scalable coordination for wireless sensor networks: self-configuration localization systems. In: Proceedings of the 6th IEEE International Symposium on Communication Theory and Application. Ambleside: University of California, 2001. 103~112

8 Aurenhammer F. Voronoi diagrams—a survey of a fundamental geometric data structure. *ACM Computing Surveys*, 1991,**23**(3): 345~405

9 Meguerdichian S, Koushanfar F, Potkonjak M, Srivastava M. Coverage problems in wireless ad-hoc sensor networks. In: Proceedings of the IEEE Infocom, Alaska, USA: IEEE Press, 2001. 1380~1387

10 Calegari P, Cuidec F. Genetic approach to radio network optimization for mobile systems. In: Proceedings of the 47th IEEE VTC. Phoenix, USA: IEEE Press, 1997. 755~759

11 Gen M, Cheng R. Genetic Algorithms and Engineering Design. John Wiley & Sons Press, USA, 1996

WEN Ying-You Received his master degree from Northeastern University, Shenyang, P. R. China in 2003. He is currently a Ph. D. candidate of Northeastern University. His research interests include network management, mobile ad hoc network, and network planning.

FENG Yong-Xin Received her Ph. D. degree from Northeastern University, Shenyang, P. R. China in 2003. Her research interests include wireless communication.

WANG Guang-Xing Professor of Northeastern University, Shenyang, P. R. China. His research interests include wireless communication and network management.