The Strategy of Probe Station Selection of Active Probing in WSNs

Hang Zhou, Yang Yang, Xuesong Qiu, Zhipeng Gao
State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing 100876, China
{cesc, yyang, xsqiu, gaozhipeng}@bupt.edu.cn

***Abstract*—In the management of WSNs, the mechanism of fault detection and location based on active probing has been widely applied. The main optimal direction of active probing is to maximize the coverage of nodes in the network by sending minimal set of probes from probe stations. Therefore, before probing, selecting optimizational station set to improve reachable rates of probed nodes has significant influence on detection effect of active probing. In this paper, we propose an optimizational strategy of probe station selection (PSS) by Genetic Algorithm (GA) and achieve the improvement of confirmed achievable rates of probed nodes. Meanwhile, lower runtime cost and more reasonable usage of energy are reached.**

Keywords—active probing; station selection; GA; shadow node;

I. INTRODUCTION

Node faults of WSNs are classified into communication fault and data fault. Communication fault is caused by failures in communication module of nodes or in link between nodes (data package loss or delay due to link congestion) and data fault, which makes the gathered inspecting data except, is caused by failures in sensing module of nodes. The main methods to detect faults in WSNs have two types: passive and active. Passive method, such as distributed fault detection based on Neighbors collaboration, is usually used in detection of data fault. On the contrary, active method is widely applied in detection of communication fault, i.e. detect data loss, delay or routing. Active probing is to send specific data package from probe stations to the network and determine the status of specific region in the network by monitoring the feedback data. The probe package can be as simple as PING command or as complicated as test transaction.

The selection of probe station has decisive effect on assuring the detectability of nodes, especially improving the coverage of nodes and making the usage of remaining energy more reasonable. Therefore, it is of great importance to devise optimized strategy of station selection and provide assistance to further detection and location. In our paper, through analysis on implementation process and functional demands, we put up a strategy of station selection based on GA. Its innovation points can be concluded as below:

In the remaining part of this paper, in section 3 specific demands of station selection are discussed and elimination on shadow nodes according to independent paths is researched. In section 4, we optimize the selection on probe stations by GA in order to eliminate shadow nodes rapidly. In section 4, we run simulation to verify the effect of PSS.

This work was partly supported by Funds for Creative Research Groups of China (61121061), NSFC (61272515, 61372108), Ph.D Programs Foundation of Ministry of Education of China (No. 20110005110011), Fundamental Research Funds for the Central Universities (No. 2014RC1102), and Beijing Higher Education Young Elite Teacher Project (YETP0474) .

II. RELATED WORK

Detection on network faults based on active probing has been widely studied and applied. Probe station selection is a focus during this research. [1] uses a two-pronged approach to compute an efficient beacon set. It formulates the need and design algorithms for computing the set of edges that can be monitored by a beacon under all possible routing states and minimizes the number of beacons used to monitor all network edges. J. D. Horton et al [2] research the optimal and systematic placement of beacons and the properties of a beacon set mapping the network both under theoretical and empirical analysis. [3-4] propose intelligent distribution of probe stations at various traffic points in the network. [5] reduces the number of monitoring stations by routing probing. However, what it concerns about are bandwidth and latency in IP networks. Effects caused by probable faults are not involved in the schemes of station selection and placement proposed above. Moreover, limited to special routing protocols and network environment, most of these algorithms are suitable to traditional IP networks. So they can't be applied to active probing for fault detection in WSNs directly. [6] proposes an algorithm which localizes minimal probe set to assure the coverage of links. It is devised to detect delay or failure in links. In this paper, we focus on station selection in order to improve the coverage of probed nodes and make a contribution to subsequent fault detection and fault location.

III. SELECTION OF PROBE STATIONS BASED ON GA

In this section, we firstly analyze the demands of station selection and build application scene. After this, we involve GA and devise our algorithm (PSS) which adapts well to our proposed scene.

A. Demands of station selection

Definition 1: Two probe paths which have the same destination are independent if and only if no joints (end nodes are not included) exists. It is depicted in Figure 1.

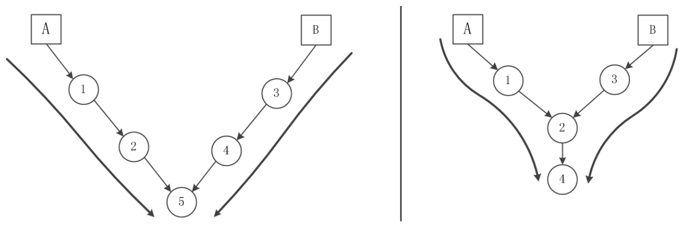

Figure 1. (a) independent paths. (b) joint paths.

In our model, we limit the number of fault nodes and assume that selected stations are not fault. Thus, we have the conclusion that a probe set can cover at most k fault nodes when and only when each node which is not the neighbor of stations is reachable from k independent probe paths. The worst situation is shown in Figure 2.

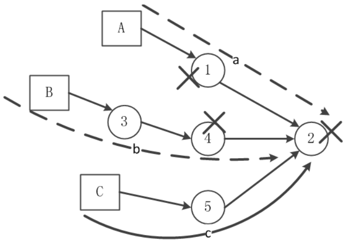

Figure 2. Fault node is reachable in the worst situation

Suppose that three faults exist. In Figure 2, node 2 that is to be probed is fault. Since probe a, b from station A, B are failed due to the faults occur in node 1 and 4, we have to get another independent probe (here is c) to node 2 in order to assure that 2 can be probed. Hence, in this worst situation, three independent probe paths are in need. Especially, when the probed node is one of the neighbors of a station, that is to say, one hop from a station, it is certain to be probed in that no fault nodes could exist between the station and it.

Definition 2: A node which is not a neighbor of stations and has less than k independent paths is called shadow node.

Shadow nodes cannot be probed before enough probe stations are selected. In Figure 2, if probe c doesn't exist, node 2 is a shadow node. Therefore, to eliminate shadow nodes are the main objective of station selection. As probe stations are selected constantly, the number of shadow nodes is decreasing relatively.

B. Strategy of probe station selection

The main idea of our strategy (PSS) is to select stations individually by a greedy way. As the shadow node set is narrowing, PSS ends when shadow node set is empty or the number of stations reaches the upper limit. In the single step of selecting a probe station, we use GA to generate the first population, process the individual selection, crossover and mutation and pass the gene with high fitness to next generation. Finally, we choose node which has highest fitness in the final generation as the station.

For simplification, it is assumed that k fault nodes exist in the network. Thus, a node which is not a neighbor of stations and has less than k independent paths is a shadow node.

Process of PSS is described in the follows:

1) Initialize that shadow node set and candidate station set are both the set of all nodes in the network and selected station set is empty.
2) Iterate candidate station set, select the node which has highest degree as first station and add it to selected station set. After this, remove first station together with its neighbors from shadow node set.
3) Compute the probe paths of newly selected station to all shadow nodes. The purpose of this step is to provide basis to further judgment on shadow nodes.
4) Use GA to select the next station optimally. When shadow node set is empty or the number of stations reaches the upper limit, the process ends; otherwise, turn to step 3.

Pseudo code of this process is shown below. Here, AN represents the set of all nodes in the network; SN is the shadow node set; SS is the selected station set; CS is the candidate station set; PT is the path set from selected stations to all shadow nodes.

```
Algorithm of probe station selection based on GA
Input: MAXSSSIZE, MAXFAULTSIZE
Output: Selected Probe Station Set
Initialization: AN as set of all nodes; SN=AN; SS= □, CS=AN, PT= □;

Select Node f (f □ CS) with highest degree as first station;
Update (f);
While |SN|>0 or |SS|< MAXSSSIZE
      Node n = CalNextStationByGA ();
      Update (n);
End
Return SS;
Function Update (node s)
      Add s to SS;
      Remove S from CS;
      Remove nodes that s releases from SN;
      PS= □;
      Foreach node n, n □ SN
            Compute path P from s to n;
            Add P to PS;
      End
      Add PS to PT;
```

Function CalNextStationByGA randomly generates a population (a node set) and tries to remain good genes as much as possible through encoding, selection, crossover and mutation. The specific designing idea is described as below:

1. Randomly generate initialized population with fixed number of individuals. Each individual is marked by a gene sequence. Encoding rule: transfer coordinate vector (x, y) of a node to binary sequence with 16 bits. For example, coordinate (35, 84) can be encoded as 0010001101010100. Nodes with similar geographic positions have similar routing characteristics.

```
ComputeFitness (node s)
PS= □;
Foreach node n, n □ SN
      Calculate path p from s to;
      Add p to PS;
End
releasedNum=neighbor number of s;
Foreach path p, p □ PS
      flag=false;
      Foreach Set PS1, PS1 □ PT
            Foreach path p1, p1 □ PS1
                  If p and p1 has the same destination
                        If p and p1 are independent
                              flag=true;
                        End
                  End
            End
      End
      If flag==false
            releasedNum++;
      End
End
Return releasedNum*EXP(E); //E is the residual energy of node s
```

2. Compute fitness of each individual and judge whether the algorithm meet the optimization criterion. If not, turn to step 3. Otherwise, output the best individual together with optimum result it brings. The main purpose of fitness function which is indispensable here is to compute the number of shadow nodes that current candidate station can eliminate, weight it by the remaining energy and then return it. It demonstrates that influence from energy is involved in our consideration. Pseudo code of fitness function is as above.

3. Generate new individuals according to certain cross probabilities and cross method. Genes from two father individuals are recombined into new genes so that new generation can be generated. Crossover is irreplaceable in evolution and contributes a lot to search ability.

Adaptive crossover probability is defined as:

$$p_{cross} = \begin{cases} p_c - \dfrac{(p_c - 0.6)(f_{max} - f)}{f_{max} - f}, & f \geq \bar{f} \\ p_c, & f < \bar{f} \end{cases}$$

Here, p_c is a preset value as 0.8; f is the fitness of current individual; $\bar{f}$ is the average fitness among this population; f_{max} is the maximum fitness. After p_{cross} is obtained, determine whether an individual has the chance to cross randomly. Thus, individuals with higher fitness are more likely retained to next generation and superior genes can be passed. Make those with lower fitness cross with each other in order to evolve superior genes.

Irrelevance between two individuals is defined as:

$$r(X,Y) = \sum_{i=1}^{16} x_i \oplus y_i$$

Here, $\oplus$ is XOR (Exclusive OR) operator. For instance, X =1001, Y =1100, then $r(X,Y)$=2. Irrelevance reflects the difference degree between individuals.

After individuals to cross have been decided, run uniform crossover operator as below to generate new individuals. In comparison with traditional single point crossover and multiple point crossover, uniform crossover is more generalized. In it, each gene point is potential to cross. 0-1 mask sequence with the same length to individual gene sequence is randomly generated, in which, the fragments indicate which father genes should be provided to offspring.

4. The mutation process of traditional GA and various improved GA is randomly choosing one bit to reverse, which cannot improve the search performance and stability of binary sequence. We put up that determine the number of bits for mutation according to individual quality.

Adaptive mutation probability is defined as below:

$$P_{mutation} = \begin{cases} P_m \dfrac{f_{max} - f_i}{f_{max} - f_{avg}} & f \geq f_{avg} \\ P_m & f < f_{avg} \end{cases}$$

Here, f_{max} is the maximum fitness in this population; f_{avg} is the average fitness; f is the highest fitness of the two father individuals; f_i is the fitness of current individual; P_m is a preset value within (0, 1). Through self-adaption, premature convergence can be avoided. Moreover, to avoid being destroyed, superior individuals will be copied to next generation.

Next, number of mutation bits can be computed by the formula below:

$$MB = [\alpha \frac{f_{max} - f_i}{f_{max} - f_{min}}]$$

α is a constant within (L/4, L/3) and L is the length of gene sequence; $f_{max} - f_{min}$ indicates the range of fitness in current population; $f_{max} - f_i$ is the difference between current fitness and maximum fitness; $\frac{f_{max}-f_i}{f_{max}-f_{min}}$ demonstrates the quality of current individual in population. The higher it is, the worse the individual is. Thus, number of mutation bits is to be decided according to individual quality.

After mutation probability and mutation bits are decided, mutation operator can be taken through random selection.

1) Generate a random value s and s ? (0, 1). If s<$P_{mutation}$, turn to step 2); Otherwise, return gene sequence X of current individual and mutation ends.

2) Generate a sequence (each bit is 1) with the length MB. Insert 0 into the interval between existing bits. After L-MB rounds, we gain a mutation mask sequence M. An instance is shown in Table 2. Suppose MB=5, L=16.

Table 1. Generating process of mask sequence

Round	Current sequence	Insert position
0	11111	11↓111
1	110111	110111↓
2	1101110	11011↓10
…	…	…
11	0100101000100010	\

3) Run XOR operator between M and X and the mutated gene sequence is obtained.

IV. SIMULATION

In simulation, we use Java to realize PSS and generate data, and then we use Matlab to draw figures. After these, we compare PSS with some others according to these figures. In our simulation model, sensor nodes are randomly distributed in a square with the size 255mx255m. The neighbor nodes of some node are decided according to its communication radius. By this way, all nodes are connected to each other and the WSN is fully connected. Moreover, the remaining energy of each node is randomly set to a percentage. Communication radius is set to 40m. Suppose that there exists three fault nodes in the network, the termination condition of PSS is that the number of selected stations reaches five. An algorithm named SNR, which roughly iterate all candidate nodes by a greedy way and the node that can eliminate maximum shadow nodes is selected as a station, is proposed in [7]. Here, we compare PSS with SNR and a scheme of random selection.

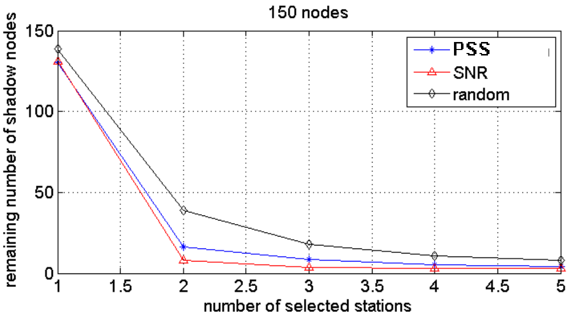

Figure 3. effects of eliminating shadow nodes in condition of 150 nodes.

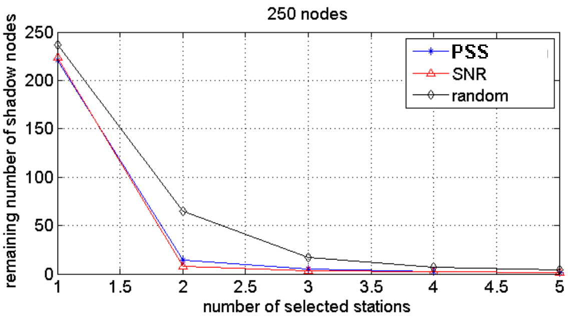

Figure 4. effects of eliminating shadow nodes in condition of 250 nodes.

Figure 4 and 5 show the effects of eliminating shadow nodes of these three algorithms. With the number of selected stations increasing, the number of shadow nodes demonstrates a downward trend. For the second station, since many paths in the network have not been repeatedly covered by probes, shadow nodes can obtain more independent probe paths. As a result, the best effect of eliminating shadow nodes is reached and descending slop is the largest. The effect of SNR must be the most superior because it searches the optimal node by roughly iterating all candidate nodes. When the sum of nodes increases, search space of PSS expands. So the effect gets better and gradually approaches SNR.

Runtime cost of SNR and PSS is compared in Figure 6 and 7. Figure 6 shows the runtime cost of these two algorithms when five rounds are undertaken to select stations in condition of 150, 200, 250, 300 and 350 respectively. As sum of nodes increases, the runtime cost of SNR rises rapidly and the average rising slope is 14.07. This is because the scale of candidate nodes it needs to iterate is rising at the same time. The curve of runtime cost of PSS raises slowly with the rising slope 1.38 owing to its operation size changes little even when density of nodes increases. Figure 7 shows the accumulative runtime cost in the process of station selection. It can be obviously seen in this figure that PSS establishes great advantage on the selection of the second station. The reason, similar to that above, is that many paths in the network have not been repeatedly covered by probes so that shadow nodes can obtain more independent probe paths.

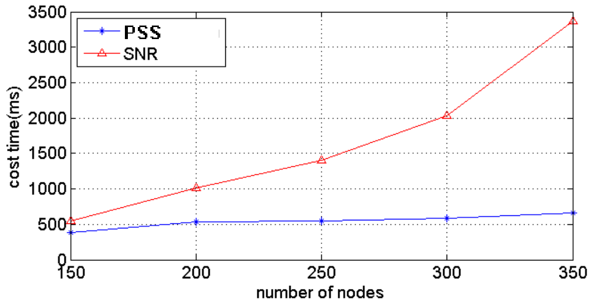

Figure 5. runtime cost in condition of different node density.

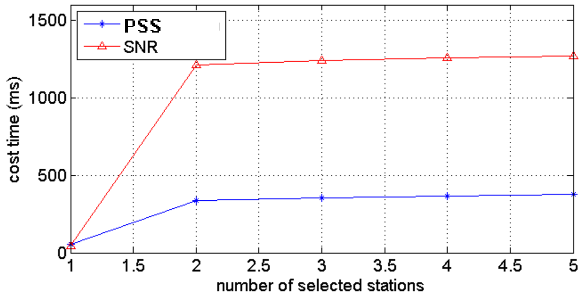

Figure 6. accumulative runtime cost with increasing rounds.

CONCLUSION

A strategy of GA based probe station selection is proposed in order to eliminate shadow nodes in the network. Improved selection, crossover and mutation operator are involved in searching superior nodes which can eliminate maximum shadow nodes. Meanwhile, the usage of energy is considered. Through simulation, PSS achieves excellent performance in eliminating shadow nodes and great advantage on the levels of energy and runtime cost.

REFERENCES

[1] R. Kumar and J. Kaur. Efficient beacon placement for network tomography. In Internet Measurement Conference, IMC, 2004.

[2] J. D. Horton and A. Lopez-Ortiz. On the number of distributed measurement points for network tomography. In Internet Measurement Conference, IMC, 2003.

[3] S. Jamin, C. Jin, Y. Jin, Y. Raz, Y. Shavitt, and L. Zhang. On the placement of Internet instrumentation. In IEEE INFOCOM, Tel Aviv, Israel, Mar 2000.

[4] Allen Downey. Using pathchar to estimate internet link characteristics [A]. ACM SIGCOMM '99[C]. Boston, USA, 241-250.

[5] Breitbart, Y., Chan, C.Y., Garofalakis, M., Rastogi, R., Siberschatz, A.: Efficiently Monitoring Bandwidth and Latency in IP Networks. In: Proc. IEEE INFOCOM 2001 (2001)

[6] Y. Bejerano and Rajeev Rastogi. Robust monitoring of link delays and faults in IP networks. In IEEE INFOCOM, San Francisco, CA, Mar 2003.

[7] Maitreya Natu and Adarshpal S. Sethi. Probe Station Placement for Fault Diagnosis. In Global Telecommunications Conference, GLOBECOM, 2007.