Concurrent Diagnosis of Clustered Sensor Networks

Chin-Woo Cho and Yoon-Hwa Choi

Department of Computer Engineering
Hongik University 121-791 Seoul, Korea

Abstract. In this paper, we present an energy-efficient on-line diagnosis algorithm for cluster-based wireless sensor networks. It employs local comparisons of sensed data and dissemination of the decision made by the comparison results. Cluster-heads act as checkers for their associated cluster members. Redundant sensor nodes, as far as sensing coverage is concerned, are partially utilized to tolerate misbehavior of cluster-heads. Final decision on the fault status of sensor nodes is made at the base station. Computer simulation shows that high fault coverage can be achieved for a wide range of fault rates.

1 Introduction

Recently wireless sensor networks are emerging as computing platforms for various applications such as environmental monitoring, security surveillance, and target tracking [1]. Low-cost, low-power, tiny sensor nodes, which consist of sensing, processing, and communication units, are deployed to gather information from the environment and to deliver messages to a remote base station. Faults in sensor networks, however, may jeopardize the integrity of the sensor networks. Fault detection and diagnosis of wireless sensor networks for some particular applications has recently been investigated in [2][3][4]. In [3] each sensor node broadcasts its sensed data to its neighbors and compares its own data with the neighbors to recognize faults in an event region detection application. In [4] a cross-validation-based technique for detecting sensor faults has been developed.

In this paper, we present a diagnosis algorithm for clustered sensor networks [5]. It does not assume any fault-free diagnostic units in sensor nodes. Spare nodes are utilized as checker nodes for their associated clusters. Fault status of sensor nodes is determined during normal operation by combined efforts of cluster-heads (including the associated checker nodes) and the base station. Both sensing and computing faults are identified. Some communication faults may be covered by treating them as computing or sensing faults.

2 Clustered Structure for Diagnosis

In the diagnosis, we use comparisons of sensed data. A cluster-based sensor network, shown in Fig. 1, is used as our network model. Originally each cluster-head at level 1 is expected to receive sensed data from its member nodes, performs data aggregation or fusion, and then sends the aggregated data to the base station. Each cluster-head checks to see if the sensing nodes are working correctly. A fault in the cluster-head, however, may render the diagnosis useless. To cope with this problem, some spare nodes, as far as sensing coverage is concerned, are temporarily used to do the same data aggregation and diagnosis. Nodes at level 1 are connected in Fig. 1 to indicate that they communicate with each other for diagnosis. Final decision on the fault status of sensor nodes will be made at the base station by collecting the diagnosis results and aggregated data from the cluster-heads and from the checker nodes, if necessary. The two-level hierarchy can be extended to even a higher-level structure, as long as the cluster size is not too small, without modifying the diagnosis algorithm to be presented shortly.

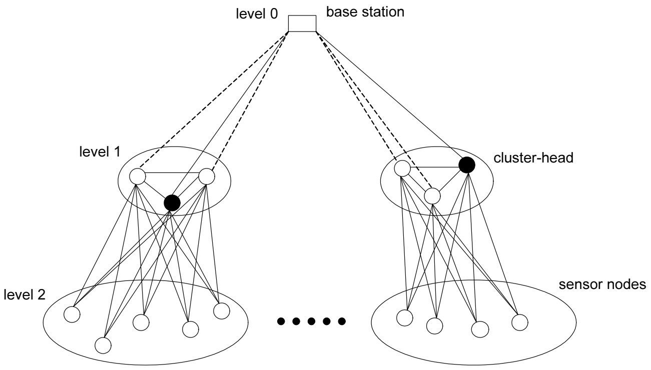

Fig. 1. Two-level hierarchy for fault diagnosis of sensor networks

3 Fault Model

The following fault model is used in the diagnosis. Faults may occur in any nodes in sensor networks, regardless of their locations (i.e., sensing, computing, communication units). Multiple faults may occur, although we assume that in a given cluster there is a single faulty level-1 node. This assumption can be easily removed if sufficient checker nodes (at least k+2 nodes for k faulty nodes) are employed. Also MTBF (mean-time-between-failure) is expected to be much longer than the diagnosis interval.

We also define the data model of the sensor networks, where a sensor node v is called a neighbor of a sensor node u if the distance between them is less than the sensing range. Let u and v be neighbors of each other and $s(u)$ denote the sensed data at node u. Then the condition to be satisfied by u and v is $|s(u) - s(v)| \leq \delta$, where δ may vary depending on the applications. In addition, an event may always be detected by more than $k_{event}(\geq 1)$ sensor nodes.

4 Fault Diagnosis Algorithm

A sensor network is represented here as a graph $G(U,E)$, where U represents the set of sensor nodes in the network and E represents the set of edges connecting sensor nodes. Two nodes u and v are said to be connected for diagnosis if the distance $d(u,v)$ is less than r (radius, sensing range, etc). $G(U,E)$ can also be called a test graph since u and v are compared only if $(u,v)\in E$. The comparison output is 0 if they satisfy the condition provided.
Definition 1: For the graph $G(U,E)$ and $u\in U$, the neighbors of u, $R(u)$ is defined to be $R(u) = \{v \in U : (u, v) \in E\}$.
Definition 2: For the graph $G(U,E)$, a label associated with $(u,v)\in E$ is represented as $S_u[v]$ and is a 0 if they satisfy the required condition addressed in the previous section. Otherwise, $S_u[v]$=1.
Definition 3: For the graph $G(U,E)$ and $u\in U$, the number of 0's in S_u is represented as $|S_u^0|$. Thus, $0 \leq |S_u^0| \leq |R(u)|$.

The diagnosis consists of three phases. It begins with a neighbor table NT, sorted in non-increasing order of node degrees, in each cluster-head-level node (cluster-head v_h and checker nodes v_c's). Each row of the table has received data, fault status indicator FSI (intialized to 1 (faulty)), a list of neighbors, and a flag V indicating whether the node has been visited or not.
Phase 1 (Diagnosing sensing nodes): In this phase, v_h (also v_c's) determines fault-free nodes based on the following two decision criteria:

- For each member node u_i in the cluster, if $|S_i^0|\geq q$(threshold), then FSI_i is set to 0 (fault-free)
- For each member node u_i in the cluster, if it has a neighbor u_k such that FSI$_k$=0 and $|s(u_i) - s(u_k)| \leq \delta$, then u_i is determined to be fault-free.

The proposed diagnosis algorithm at phase 1 is depicted in Fig. 2. Depth-first search is used to visit the sensor nodes in NT until there are no neighbor nodes to visit.

```
Fault Diagnosis()
  Create a neighbor-table $NT$ (sorted)
  Initialize all $FSI$'s to 1(faulty) and $V$'s to 0 (not visited)
  While (until the last node of $NT$ with $V_i$=0)
  If $|S_i^0| \geq q$ then $FSI_i \leftarrow 0$; $V_i \leftarrow 1$ (fault-free); Propagation($i$);
            else if $|S_i^0|$=0, $V_i \leftarrow 1$ (faulty)
 Propagation ($i$)
   For each neighbor node $u_k$ with $S_i[k]$=0 and $V_k$=0
     $FSI_k \leftarrow 0$; $V_k \leftarrow 0$; Propagation($k$);
```

Fig. 2. Fault diagnosis for phase 1 in level-1 nodes

If v_h is faulty, however, the decision made by v_h might be incorrect. Moreover, reliable fusion or aggregation of sensed data, coming from the member nodes, cannot be guaranteed.

Phase 2 (Diagnosing cluster-heads): The cluster head v_h and w checker nodes exchange their aggregated data and the results of phase 1 (FSI's) among themselves. During this process, the base station also receives the same aggregated data and the results of phase 1 from the cluster-head v_h. Each of them performs w comparisons (its own data with those of w other nodes) to see if v_h is fault-free. If the number of matches is greater than or equal to t (threshold), it determines itself fault-free. Each checker node, determined to be fault-free, sends its own FSI's and aggregated data it has generated to the base station only if it finds that v_h is faulty.

Under the given fault model, we need to consider the following two cases.
1) The cluster-head v_h is faulty. In this case (left side of Fig. 3), each of the checker nodes, $v_c' s$, will determine that v_h is faulty and send its own diagnosis results FSI's and and fused data to the base station. The dotted arrow means that the faulty cluster-head might send its own data to the base station.
2) One of the checker nodes is faulty. In this case (right side of Fig. 3), v_h finds itself fault-free and identifies the faulty checker node. Fault-free checker nodes, in the figure only one node v_{c1}, find that v_h is fault-free and will not send the diagnosis results to the base station. The faulty checker node, however, might send its own results to the base station, as indicated by a dotted arrow. The base station simply ignores the information if no other checker node reports that the cluster-head is faulty.

Phase 3 (Final decision at the base station): The base station will receive FSI's and fused data from either the cluster-head (case 2) or more than one checker node (case 1). Based on the information, the fault status of each sensor node will be determined. In this process, an event has to be distinguished from a fault since both may assume sensing values outside of allowed range. In the case of an event, the diagnosis results are very similar to those of common mode failures. Hence the base station, which receives information from all the clusters, can distinguish events from faults, if common mode failures are unlikely to occur or can be controlled.

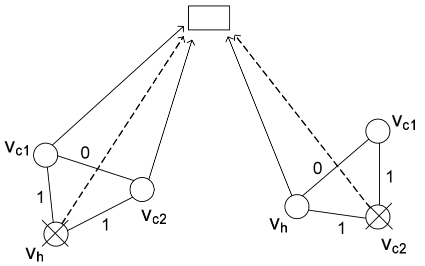

Fig. 3. Two possible fault patterns in level-1 nodes

5 Performance Evaluation

The performance of the proposed diagnosis algorithm is evaluated by computer simulation. Faults are assumed to be independent of each other. Also a sensor node is assumed to be faulty with probability p regardless of the location of the fault.

Let n_g and n_f be the numbers of fault-free nodes and faulty nodes in a sensor network, respectively. Also let n_{gg} be the number of fault-free nodes identified correctly and n_{fg} be the number of faulty nodes diagnosed as fault-free. The following two measures, $\alpha = \frac{\sum_{i=1}^{n}(n_{gg}/n_g)_i}{n}$ and $\beta = \frac{\sum_{i=1}^{n}(n_{fg}/n_f)_i}{n}$ are used to evaluate the performance, where n is the sample size. Apparently α and β lie in between 0 and 1 and our goal is to make α and β very close to one and zero, respectively.

In the simulation, nodes are placed randomly in an $l \times l$ (l=50m) rectangular region and 1000 sample clusters, randomly partitioned, are used to derive statistical data. For a given region with m sensor nodes, the average node degree ($\tilde{d}$) increases almost linearly with k (up to 8), where $k = \frac{m(\pi r^2)}{l^2}$ and r is the sensing range. The sensing coverage for k=3 is 0.906 and the corresponding $\tilde{d}$ is 2.26. Considering the fact that sensor networks need to maintain high coverage, the desired value of k would be greater than 3.

Fig. 4 shows α and β for various values of p, k and q. As expected, α becomes very close to 1 as k (or $\tilde{d}$) increases even for a high p. At the same time, β can be almost perfectly controlled by properly choosing the value of q (threshold). As q increases, however, more fault-free nodes are likely to be misdiagnosed. Removing common mode failures by using higher threshold ends up with losing an increased number of fault-free nodes. Especially sensor nodes at the corners (or borderline) are at risk. Inter-cluster checking at the base station could enhance the performance since fault-free nodes isolated by faulty node(s) due to lower connectivity can be identified. The same simulation has been performed only for the test graphs connected. The results are shown in Fig. 4(d). As expected, a notable difference has been observed.

The proposed diagnosis algorithm is well suited with cluster-based communication. The overhead required is the energy consumed for exchanging diagnosis results and fused data among the cluster-head and checker nodes. Since w (number of checker nodes) is expected to be small, we can claim that the energy used for these communications is manageably small. Periodic checking can further reduce the energy consumed for diagnosis.

6 Conclusions

In this paper, we have presented a technique for locating faulty sensor nodes during normal operation. Faulty nodes have been identified without any fault-free diagnostic units. The diagnosis algorithm is well suited with the cluster-based communication protocols since most of the required communications occur

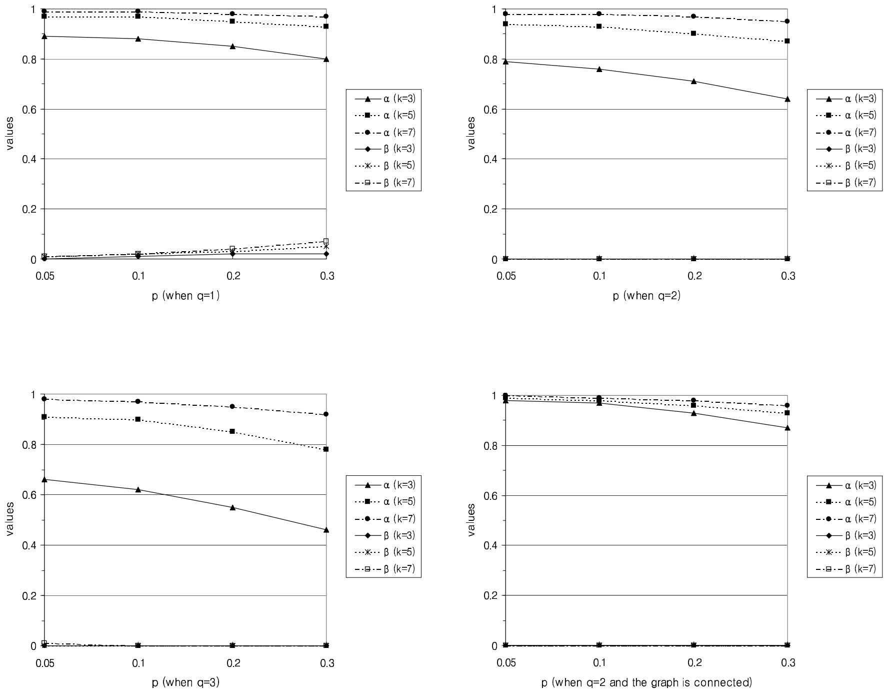

Fig. 4. α and β for various values of p, k, and q.

through the paths originally established. As a result, the energy consumed for diagnosis has been minimized.

References

1. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cyirci, "Wireless Sensor Networks: A Survey," Computer Networks, vol. 38, no.4, pp. 393-422, 2002.
2. C. Jaikaeo, C.Srisathapornphat, C-C. Shen, "Diagnosis of sensor networks," Int. Conf. Communications, vol 5, pp. 1627-1632, June 2001.
3. B. Krishnamachari, and S. Iyengar, "Bayesian Algorithms for Fault-tolerant Event Region Detection in Wireless Sensor Networks," IEEE Transactions on Computers, Vol. 53, No. 3, March 2004.
4. F. Koushanfar, M. Potkonjak, A. Sangiovanni-Vincentelli, "On-line fault detection of sensor measurements," IEEE Sensors, vol. 2, pp. 974-980, Oct. 2003.
5. W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," IEEE Transactions on Wireless Communications, Vol. 1, No. 4, pp. 660-670, Oct. 2002.