A Distributed Adaptive Scheme for Detecting Faults in Wireless Sensor Networks

JAE-YOUNG CHOI, SUNG-JIB YIM, YOON JAE HUH and YOON-HWA CHOI
Department of Computer Engineering
Hongik University
Seoul, KOREA
yhchoi@cs.hongik.ac.kr

Abstract: This paper presents a distributed adaptive scheme for detecting faults in wireless sensor networks. Each sensor nodes makes a local decision based on the comparisons of its own readings with those of neighbors, along with the dissemination of the decision to them, if necessary. At the end of each fault detection cycle, each node dynamically adjusts critical parameters in the distributed fault detection algorithm, such as node degree and thresholds, resulting in high performance for a wide range of fault probabilities. By extensive computer simulation the scheme is shown to be scalable with the number of faulty sensor nodes except for sparse networks where the average node degree is extremely low.

Key–Words: sensor networks, distributed, adaptive, fault detection, fault diagnosis

1 Introduction

Wireless sensor networks are expected to be increasingly used in various industrial, health care, environmental monitoring, and military surveillance applications [1],[2],[3]. They are often composed of hundreds or even thousands of low-cost, tiny sensor nodes each with sensing, data processing, and communicating capabilities. When a large number of sensor nodes are distributed to monitor a vast field where the operational conditions are harsh or hostile, they are likely to have faults and measurement errors, and thus become unreliable. Hence it is crucial that the networks remain operational all the time even in the presence of faults in the networks. Moreover, faulty sensor nodes need to be identified and isolated unless they can act as communication nodes.

Fault tolerance problems in wireless sensor networks have recently been investigated and the results are presented in the literature [8],[18]. Among others fault detection in sensor readings, malicious node detection, and fault-tolerant event detection have been of the main concern and they are closely related.

Jaikaeo et al. [4] have pointed out the response implosion problem which occurs when a high volume of incoming replies triggered by diagnosis queries cause the central diagnosing node to become a bottleneck. Three operations, sampling, self-orchestrated, and diffused computation schemes, to overcome the problem were presented. On-line detection of sensor faults using a cross-validation-based technique was proposed in [5]. Statistical techniques are employed to locate sensor nodes that have the highest probability to be faulty. In [8] an external manager was introduced to perform fault detection in an event-driven wireless sensor network. A network management architecture, named MANNA, was proposed. System-level diagnosis in ad-hoc network based on the PMC model was presented in [6]. A new diagnostic model based on one-to-many communication paradigm was introduced. Both hard and soft faults are detected. In addition, the change in network topology during diagnosis was also considered. Performance analysis of a distributed comparison-based self-diagnosis protocol for wireless ad-hoc networks has been presented in [7].

Malicious node detection in wireless sensor networks has been investigated in [11],[12],[13]. In [12] a mechanism based on signal strength and geographical information for detecting malicious nodes was proposed. Curiac et al. [13] employed an autoregression technique in detecting malicious nodes. Sensor's output is compared with its estimated value computed by an autoregression predictor to identify suspicious nodes. A neural network based approach has been proposed in [11], where malicious nodes in sensor networks are discovered by using an on-line neural network predictor based on past and present values obtained from neighboring nodes.

Fault tolerance has also been studied in event detection of wireless sensor networks. In [9] Bayesian fault recognition algorithm was presented to solve the fault-event disambiguation problem in sensor networks. Ding et al. [14] have proposed a localized fault identification algorithm, where each sensor node compares its own sensed data with the median of those of neighbors to determine its own status. The performance of the localized diagnosis, however, is limited due to the non-uniform nature of node degrees in sensor networks with random deployment. Luo et al. [10] have proposed a fault-tolerant energy-efficient event detection paradigm for wireless sensor networks. For a given detection error bound, minimum neighbors are selected to minimize the communication volume. Both Bayesian and Neyman-Pearson detection methods are presented.

A distributed fault detection scheme for sensor networks has been proposed in [15]. It uses local comparisons with a modified majority voting, where each sensor node makes a decision based on comparisons between its own sensing data and those of neighbors, while considering the confidence level of its neighbors. The scheme, however, is a little complex in the sense that information exchange between neighboring nodes has to occur twice to reach a local decision based on a threshold. Li et al. [16] have presented a distributed fault detection algorithm base on the idea of classifying two clusters with different readings from a set of sensor readings. Maximum spanning trees are employed in the clustering of sensor nodes. Transient faults in sensing and communication have been investigated in [17]. A simple distributed algorithm has been proposed to tolerate transient faults in the fault detection process. Some other fault management schemes can be found in the survey written by Yu et al. [18].

Most of the algorithms presented so far achieve extremely high performance for a relatively low fault probability. They, however, need to be scalable since their performance degrades significantly as the number of faulty nodes increases. This is due to the tradeoff between detection accuracy (DA) and false alarm rate (FAR) in most fault detection algorithms for wireless sensor networks. When the fault probability p is low, both DA and FAR can be kept extremely high and low, respectively. As p increases, however, the algorithms can hardly meet the requirements regardless of the values of the important parameters such as thresholds.

In this paper, we present a distributed adaptive fault detection scheme for wireless sensor networks to overcome the low scalability problem of existing fault detection algorithms. The scheme can achieve consistent performance for a wide range of fault probabilities by dynamically adjusting the network topology and algorithm parameters to adapt to the increase in the number of faulty nodes. Computer simulation shows a notable difference in performance except for sparse networks.

The remainder of the paper is organized as follows. Section 2 describes the network and fault models for fault detection in wireless sensor networks. Our distributed adaptive fault detection algorithm is presented in Section 3. Experimental results are shown in Section 4. Conclusion is made in Section 5.

2 Network and Fault Models

We consider fault detection problem in wireless sensor networks where n sensor nodes are distributed randomly or regularly in a 512×512 square grid of unit area. All sensor nodes in the network are assumed the same communication range. In other words, each node can only communicate with its neighbors within the distance in any direction. Sensor networks with three different average node degrees, 7, 15, and 20, are considered for random deployment. In the case of regularly deployed sensor networks, degrees of 4, 8, 12, and 20 are chosen for simplicity of configuration.

In this paper, a sensor's reading is said to be erroneous if it is significantly different from those of its neighbors. A sensor node that generates an erroneous measurement is not always treated as a faulty sensor node. When a sensor node exhibits consistent faulty behavior, it will be determined to be faulty and isolated from the network, if necessary. Sensor nodes with some transient faults or some measurement errors will be treated as normal nodes.

Faults may occur in any nodes in a sensor network with the same probability p, regardless of their locations. In addition, p is assumed to increase with time to reflect the increase in the number of faulty sensor nodes in the network as time goes on. Noise-related measurement errors and incorrect sensor readings due to some transient faults are also assumed to occur independently in time and space with probability q. Sensor nodes with malfunctioning sensors will be isolated only for fault detection purposes. They, however, are allowed to participate in the network operation as communication nodes since they are still capable of routing information.

3 Adaptive Fault Detection

Existing fault detection algorithms can be classified into two types: $centralized$ and $distributed$. In centralized approach, a centralized senor node performs fault management of the entire network. It, however, can hardly be applied to large-scale networks due to the single point of failure, high volume of message traffic, and rapid energy depletion in the nodes closer to the central node. Hence most research on fault detection of sensor networks is focused on developing distributed and computationally efficient fault detection algorithms [18].

In distributed fault detection, fault status of each sensor node is determined by local decision (such as majority voting, comparison of sensed data with the median of the received data from neighbors, etc.) [14] or by local decision with dissemination of the decision to neighbors [15],[17]. The following two metrics, detection accuracy (DA) and false alarm rate (FAR), are used in this paper to evaluate the fault detection performance, where DA is defined as the ratio of the number of faulty sensor nodes detected to the total number of faulty nodes and FAR is the ratio of the number of faulty sensor nodes diagnosed as faulty to the total number of good nodes.

Based on our analysis and subsequent extensive simulation we have observed the following two facts that can be used in developing fault detection algorithms.

- As the threshold for local decision θ increases, both DA and FAR increase, resulting in unsatisfactory performance for relatively high p.
- Dissemination of the local decision to neighbors decreases both DA and FAR, resulting in unsatisfactory performance for relatively high p.

Hence DA and FAR cannot be kept high and low, respectively, as the number faulty sensor nodes increases, showing poor scalability with respect to p. Especially in a harsh environment faults are likely to occur more frequently and unexpectedly. The resulting increase in p requires a scalable fault detection algorithm to almost guarantee high DA and low FAR at the same time.

Our distributed adaptive fault detection algorithm is designed to maintain high DA and low FAR even with a significant increase in the number of faulty sensor nodes in wireless sensor networks. To do that each sensor node constructs a neighbor table NT including the comparison results of its own sensor readings and those of neighbors. The table will be updated each time fault detection is performed or periodically at a given time interval. Since the mean time between failures (MTBF) is expected to be much longer than the fault detection cycle time, it is not necessary to update the table very often. Moreover, our fault detection algorithm to be presented is quite insensitive to a small increase in p. The information stored in the table will be used to redefine the neighboring nodes for fault detection and to dynamically adjust the algorithm parameters such as threshold θ and the effective node degree d_t (at time t) for fault detection, etc. Here d_t is determined by subtracting the number of faulty neighboring nodes identified from the initial node degree d. Based on the results of fault detection each node will be in one of two states: good (0) and faulty (1).

Several other terms are also defined and included in the table below for convenience.

Summary of Notation

d : node degree of a sensor node
d_t : effective node degree at time t
$\tilde{d}$: average node degree of a sensor network
x_i : sensor reading at node i, v_i
p : fault probability
q : transient fault probability for good nodes
r : transient fault probability for faulty nodes
θ : decision threshold
k_i : no. of matching neighbors
m : window size for tolerating transient faults
α : threshold for screening out transient faults
F : fault status of a sensor node

The proposed fault detection scheme, consisting of five steps, can be depicted as follows, where two sensor nodes are neighbors of each other if the distance between them is less than the transmission range. Although sensed data are exchanged in the description of the algorithm, any local decision (normal/abnormal) at each node can also be used instead to identify faulty sensor nodes with simple modifications.

Adaptive Fault Detection Scheme
For a sensor network with average node degree $\tilde{d}$
0. Create NT (neighbor table) and set F to 1 (faulty)
1. Obtain the sensor readings $x_j's$ of neighbors
2. Determine k_i, the number of $matching$ neighbors
3. If $k_i \geq \theta$, set F to 0 (good)
4. Repeat l times for the undetermined nodes
   If one of its matching neighbors is determined
   to be good, then set F to 0 (good)
5. Update NT and adjust θ by taking d_t into account

In step 0, the fault status of each sensor node is initialized to 1 (i.e., faulty). That is, a sensor node can change its status to 0 (i.e., good) only if it determines itself to be good. In step 1, each sensor node receives sensor readings from its neighboring nodes. It then determines the number of matching neighbors in step

2. In step 3, if the number of matching neighbors is greater than or equal to the given threshold θ, the node will determine itself to be good. Step 4 is not necessary (i.e., l=0 is possible) although it can improve the performance depending on the network topology, fault pattern, and fault probability. Apparently the communication overhead involved will increase with l. Hence in practice l=1 would be desirable, if necessary.

In step 5, various adjustments can be made to adapt to the increase in faulty sensor nodes in the network, and hence to improve or maintain fault detection performance. The reason for the adjustment will be explained shortly. Among others we consider the following three cases in this paper.

- $\theta = \tilde{d}$ (average node degree)
- $\theta = \frac{d_t}{2}$
- $\theta = \max(\delta, \frac{d_t}{2})$, where δ is a small positive integer.

The first one can be applied even in the case where p is relatively high, although the communication overhead involved in disseminating the local decision to neighbors might cause some problem if required more than once. The second one is the well-known majority voting, which is adequate for a relatively small value of p. The third one, our proposed threshold, is a practical compromise between them. In other words, it applied majority voting until the node degree is relatively high. At the time it reaches a small predetermined number, the threshold freezes at the point, resulting in θ higher than that of the majority voting. Here δ= 3 (or 2) is chosen for comparison and illustration.

As an illustration of the adaptive fault detection scheme consider the sensor nodes in a randomly deployed sensor network, where five nodes are faulty (crossed out) and node v_i is chosen to describe how the fault detection proceeds. Without loss of generality δ=2 is chosen. In step 1, v_i obtains the sensor readings of its five neighbors v_{1-5}. In step 2, it compares its own reading with those of neighbors to determine k_i, the number of matching neighbors. Since d_i=5 and k_i=3 in the case, k_i is greater than or equal to d_i/2. Hence v_i will determine itself to be good in step 3. In the meantime, v_2 performs the same threshold test (but with θ=4) and finds that it cannot pass the threshold since it has only three matching neighbors out of seven. At the end of step 3, v_i is determined to be good, while v_2 remains in the same initial state.

In step 4, however, the decision made by v_i will be disseminated and v_2 will then be determined to be good with the help of v_i, as indicated by an arrow.

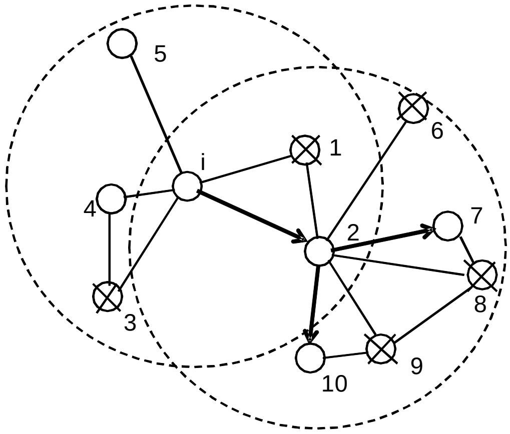

Figure 1: An illustration of the proposed fault detection algorithm

If v_7 and v_{10} are still undetermined, they will correctly determine themselves to be good in the next round with the aid of v_2. Dissemination of the decision made at a sensor node requires communication overhead, limiting the number of times it can be applied. Although the scheme is depicted with a variable l for generality, we only consider the cases of l=0 and 1, to make the algorithm practically useful.

Step 5 updates NT (neighbor table) and adjusts θ and d_t periodically. Assuming that the current fault detection cycle requires the adjustment of parameters, we change θ and d_t accordingly by taking the faulty nodes identified so far into account. In this example, v_i removes v_1 and v_3 from its neighbor list for fault detection purposes and changes its node degree d_i to 3. This dynamic adjustment will allow the fault detection scheme to be applied effectively even with increasing number of faulty nodes in the network with time. Suppose that v_5 turns into a faulty node at a certain time and generates unusual sensor measurements. Without step 5, v_i cannot pass the threshold test since k_i at this time is 2 while θ is 3. Step 5 will allow v_i to determine itself to be good, since it has two matching neighbors and θ at this time is 2 as well.

To justify the effectiveness of the proposed adaptive fault detection scheme, we use Fig. 2, where all the n sensor nodes are assumed to be good in the beginning and the number of faulty sensor nodes increases linearly with time. Transient faults may occur during the entire period of time, although sensor nodes with transient faults or noise-related measurement errors are treated as good nodes, and thus should not be isolated from the rest of the network. In the figure, the dotted and solid vertical lines represent the good and faulty nodes, respectively. That is, n sensor nodes are divided into two groups: good (dotted) and faulty (solid) nodes.

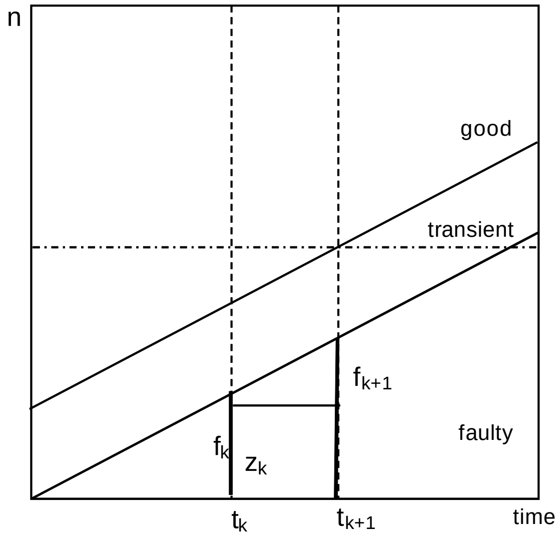

Figure 2: The usefulness of adjusting parameter values

Consider the number of active nodes at two adjacent fault detection cycles, t_k and t_{k+1}. The number of faulty nodes at time t_k is denoted by f_k. Also the number of faulty nodes identified up to time t_k is represented by z_k. Hence f_k-z_k is the number of faulty nodes determined to be good due to the incompleteness of the fault detection scheme.

Suppose that at the end of fault detection cycle t_k the node degree and threshold are adjusted, depending on the results of fault detection. That is, z_k nodes are diagnosed as faulty so far, and thus removed from the neighbor list for the purpose of fault detection. Apparently each node updates its table and set d_t and θ accordingly. At time t_{k+1} the number of active nodes is $n - z_k$, as opposed to n. Hence the ratio of the number of abnormal readings to the total number of active nodes, R_{adapt}, can be written as

$$R_{adapt} = \frac{f_{k+1} + n_{tran} - z_k}{n - z_k}, \tag{1}$$

where n_{tran} represents the number of transient faults or noise-related measurement errors at the time. If our adaptive scheme is not applied, the ratio R_{org} is given by

$$R_{org} = \frac{f_{k+1} + n_{tran}}{n} \tag{2}$$

The ratios are important since the majority voting and our proposed adaptive scheme will use $\frac{n}{2}$ and max($\delta, \frac{n-z_k}{2}$) as θ, respectively. For both threshold tests, the number of nodes with normal readings is the same, while the number of abnormal readings is reduced in our adaptive scheme. In Fig. 2, for example, f_{k+1}+n_{tran} is n/2 at time t_{k+1}. For the majority voting, in that case, each sensor node will receive on average the same number of normal and abnormal readings from its neighbors, resulting in poor fault detection performance. On the other hand, at time t_{k+1} our adaptive scheme removes z_k faulty nodes from consideration, equivalent to reduction in the fault probability p, and thus can maintain high performance even with increasing number of faulty sensor nodes.

Since transient faults are likely to occur in sensor readings, decision based on a single sensing data might be incorrect. To tolerate transient faults in sensor readings, a matching neighbor in step 2 of the fault detection algorithm can be determined based on the comparison results of m consecutive fault-detection rounds with a threshold α. If the window size m=10 and α=0.7, for example, at least seven (out of ten) sensor readings must pass the equality test to become a matching neighbor. In other words, three transient faults in sensor readings can be tolerated.

Our adaptive scheme with filtering some transient faults will further enhance the fault detection performance. Screening out some transient faults using m consecutive measurements will effectively reduce the number of transient faults n_{tran} in Fig. 2. If all of them are filtered, the ratio R_{adapt} will change to $\frac{f_{k+1}-z_k}{n-z_k}$. Lowering the ratio will improve DA, and this in turn will make z_{k+1} very close to f_{k+1}, and so on. As long as the ratio is kept below a certain limit, our adaptive scheme will maintain consistent performance for a wide range of p. Hence the performance strongly depends on how often fault detections are performed and how often θ and d_t are adjusted.

4 Experimental Results

Computer simulation is carried out in a sensor network, where 1024 sensor nodes are deployed randomly in a rectangular region of size 512×512 units. All the nodes are assumed to have a common transmission range. The fault probability p is defined to be a function of time, initialized to a certain value and increased with time. The increase in p between two adjacent update cycles is assumed to be 0.05 to estimate the performance in highly fault-prone sensor networks. More frequent updates will lower p, and thus can achieve better performance.

We first perform simulation for some existing schemes: majority voting (MV), majority voting with dissemination (MVD), local detection with threshold $\theta \approx \tilde{d}$ (LT), and local detection with threshold $\theta \approx \tilde{d}$ and dissemination (LTD) [17]. The results for various values of p when the average node degree $\tilde{d}$ is approximately 20 are shown in Fig. 3 and Fig. 4. LT achieves the best performance among them as far as DA is concerned, while MVD outperforms others with respect to FAR. Due to the tradeoff between DA and FAR, satisfactory results cannot be obtained when $p \geq 0.25$. Even for $p < 0.25$ further improvement in performance is desirable.

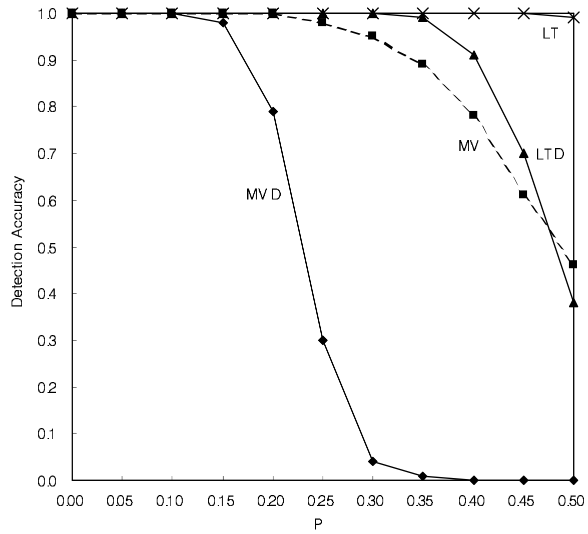

Figure 3: Detection accuracy (DA) for MV, MVD, LT, and LTD when $\tilde{d} \approx 20$

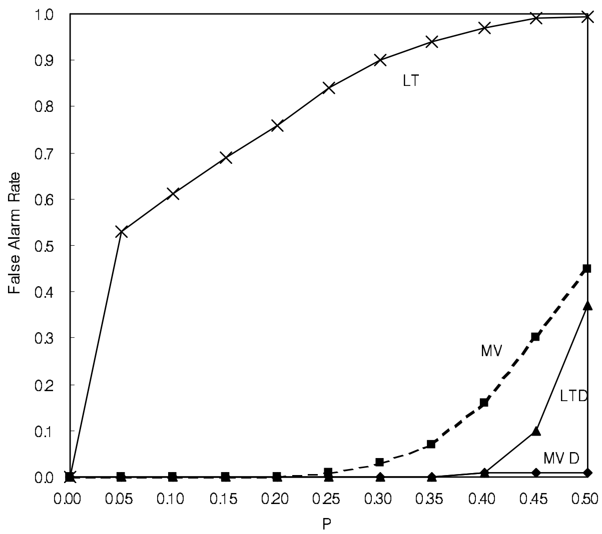

Figure 4: False alarm rate (FAR) for MV, MVD, LT, and LTD when $\tilde{d} \approx 20$

We then perform simulation for our adaptive fault detection scheme where the threshold θ and the effective node degree at time t, d_t, are adjusted in accordance with the increase in the identified faulty sensor nodes in the network. They are named AMVD and ALTD to emphasize adaptivity. The number of faulty nodes is increased as the simulation proceeds. To take the changes into account, θ is adjusted accordingly. The results for $\tilde{d}\approx 20$ and $\tilde{d}\approx 7$ are shown in Fig. 5 and Fig. 6, respectively, for various values of p.

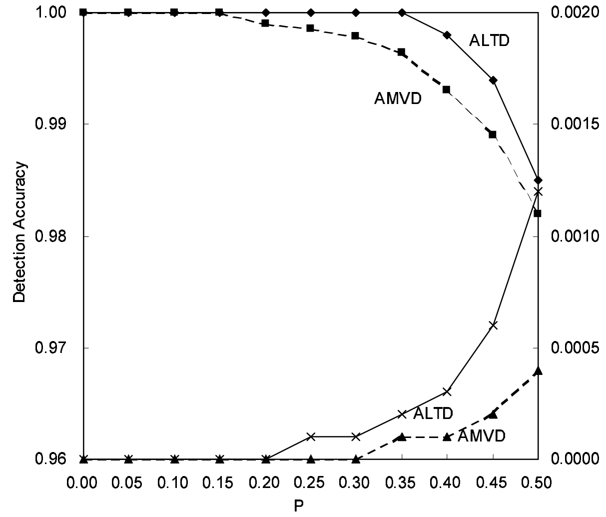

Figure 5: DA and FAR for AMVD and ALTD when $\tilde{d} \approx 20$

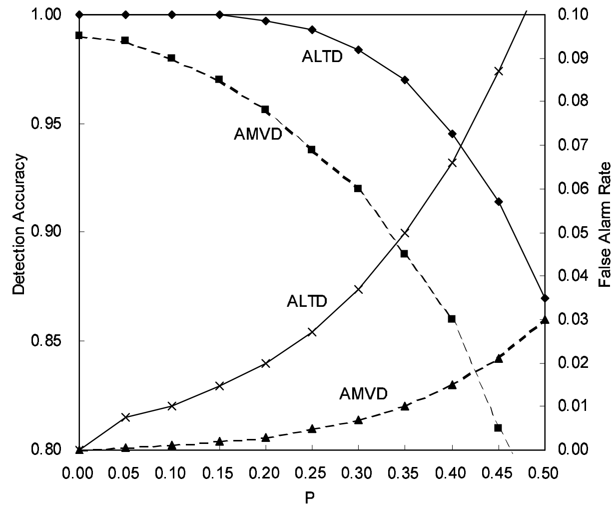

Figure 6: DA and FAR for AMVD and ALTD when $\tilde{d} \approx 7$

Both DA and FAR are kept extremely high and low, respectively, for a wide range of p values. For $\tilde{d}\approx 7$, however, some degradation in performance is observed as p increases due to the low network connectivity. In wireless sensor networks, however, the average node degree is expected to be high to maintain sensing coverage and achieve robustness to faults. Moreover, as the number of faulty nodes increases, the

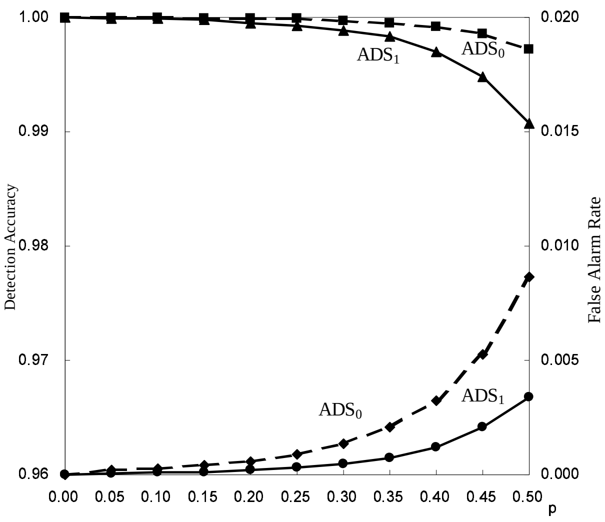

Figure 7: DA and FAR for ADS_0 and ADS_1 when $\tilde{d}\approx$20

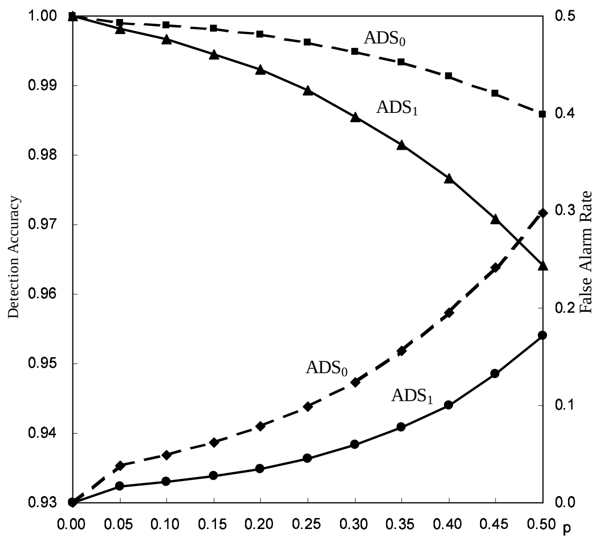

Figure 8: DA and FAR for ADS_0 and ADS_1 when $\tilde{d}\approx$7

effective node degree decreases, requiring additional sensor nodes to be deployed to establish proper network connectivity.

Due to the dissemination in step 4 of the fault detection scheme, the communication overhead for $\theta\approx\tilde{d}$ could be problematic when p is relatively high. To overcome this problem, we make a practical compromise, and then perform simulation for $\theta = \max(\delta, \frac{d_t}{2})$, where δ=3 is chosen. The scheme is named ADS_l where l is the number of iterations in step 4. The results for $\tilde{d}\approx$20 are shown in Fig. 7, where ADS_0 and ADS_1, the adaptive scheme with l=0 and 1, respectively, achieve extremely high performance. ADS_0 and ADS_1 for $\tilde{d}\approx$ 7 need improvements in FAR as shown in Fig. 8, as p increases. As the number of faulty nodes in a sparse sensor network increases, performance degradation is likely to be unavoidable without adding more sensor nodes. Since the average node degree is expected to be relatively high in sensor networks, we can claim that ADS_0 and ADS_1 can achieve acceptable performance for a wide range of p without increasing the communication overhead. Similar performance has also been observed for δ=2.

In order to estimate how much the proposed scheme tolerates transient faults, we then perform similar simulation with some transient faults in sensor readings. Fault detection is applied periodically. The period is assumed to be much less than the time for an increase of 0.05 in the fault probability p. In other words, a number of fault detection rounds are completed in the mean time between failures. Each good node may now have a transient fault in sensor reading with probability q. Hence in reality the sensed data at a node may be erroneous with probability p+(1-p)q. Also all the faults are assumed to be independent. Decision is made based on the current comparison result along with m-1 previous results. That is, a node makes decision on its fault status using preassigned values of m and α. Without loss of generality m=10 and α=0.7 are chosen in the simulation. The results for $\tilde{d}\approx$20 and q=0.15 are shown in Fig. 9, where the initial fault probability and θ are set to 0.1 and max(3, $\frac{d_t}{2}$), respectively.

Both DA and FAR for ADS_0 and ADS_1 are acceptably high and low, respectively, although ADS_1 achieves much improvement in FAR with some degradation in DA. Since there is a negligible tradeoff between DA and FAR in selecting the value of l, it is not a concern to find the optimum value of l. If the communication overhead involved in dissemination of the local decision at each node is a major concern, l=0 will be the best choice.

We additionally consider transient faults that cause a faulty sensor node to assume a normal value. To reflect this in the simulation, we define r to be the probability that a faulty sensor node reports a normal reading due to a fault. In practice, r is likely to be much smaller than q since the range of normal values is much narrower than that of abnormal values. The results for $\tilde{d}\approx$20, q=0.1, and r=0.1 are shown in Fig. 10, where only l=0 and 1 are chosen to reduce the communication overhead in fault detection. As expected, both DA and FAR are still acceptably high and low, respectively. If the majority voting is applied instead, the performance will degrade considerably due

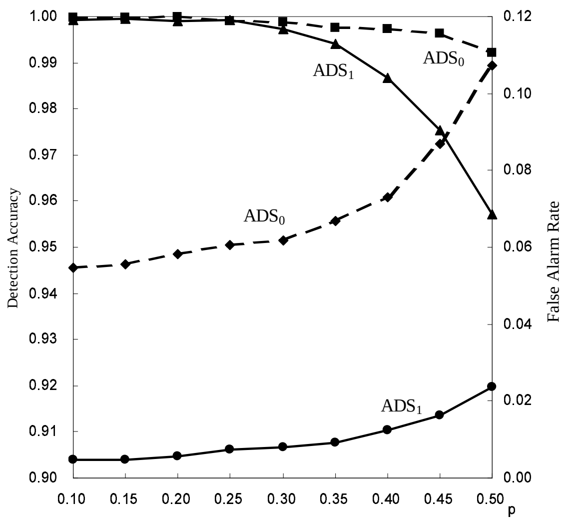

Figure 9: DA and FAR for ADS$_0$ and ADS$_1$ when $\tilde{d}\approx$20 and q=0.15

to negative impact of q and r, on top of the existing faulty sensor nodes.

Simulation results for $\tilde{d}\approx$15 are shown in Fig. 11. Some degradations in performance both in DA and FAR are observed, although the degradations are negligibly small.

We also perform simulation after changing the value of δ to 2. The results for $\tilde{d}$=15 and q=r=0.1 are compared in Table 1. A slight improved performance has been observed when d=20. As expected, FAR can be improved by choosing δ=3, while DA can be enhanced by lowering δ to 2. Overall an extremely high performance has been observed for both values of δ.

Table 1: DA for ADS$_0$ and ADS$_1$ when $\tilde{d}\approx$15, q=0.1, and r=0.1

	ADS$_0$		ADS$_1$	
p	δ=3	δ=2	δ=3	δ=2
0.05	0.9997	0.9999	0.9981	0.9973
0.10	0.9998	0.9998	0.9982	0.9986
0.15	0.9999	0.9996	0.9982	0.9982
0.20	0.9998	0.9995	0.9983	0.9978
0.25	0.9996	0.9995	0.9970	0.9963
0.30	0.9994	0.9988	0.9955	0.9942
0.35	0.9988	0.9979	0.9935	0.9895
0.40	0.9980	0.9960	0.9899	0.9830
0.45	0.9972	0.9930	0.9825	0.9713
0.50	0.9957	0.9866	0.9709	0.9529

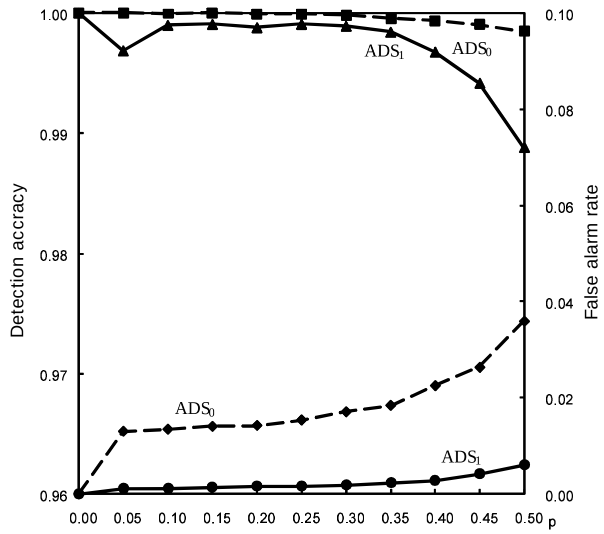

Figure 10: DA and FAR for ADS$_0$ and ADS$_1$ when $\tilde{d}\approx$20 and q=r=0.1

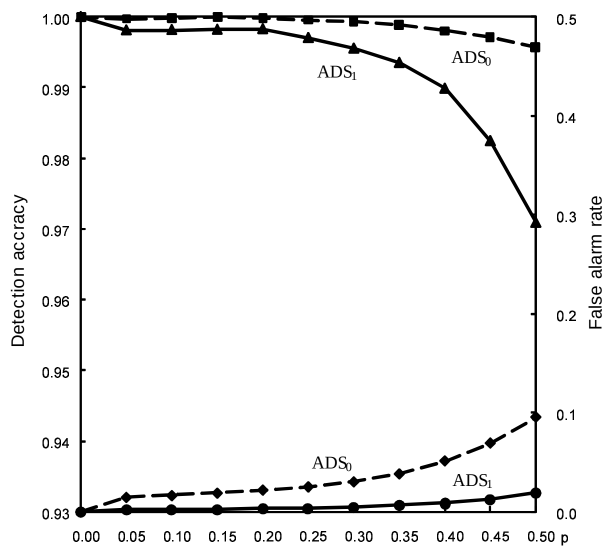

Figure 11: DA and FAR for ADS$_0$ and ADS$_1$ when $\tilde{d}\approx$15 and q=r=0.1

Table 2: FAR for ADS_0 and ADS_1 when $\tilde{d}\approx15$, q=0.1, and r=0.1

p	ADS_0		ADS_1	
	δ=3	δ=2	δ=3	δ=2
0.05	0.0147	0.0138	0.0019	0.0017
0.10	0.0165	0.0140	0.0023	0.0020
0.15	0.0190	0.0144	0.0024	0.0023
0.20	0.0221	0.0163	0.0030	0.0027
0.25	0.0253	0.0169	0.0037	0.0028
0.30	0.0303	0.0182	0.0046	0.0038
0.35	0.0382	0.0205	0.0064	0.0043
0.40	0.0513	0.0249	0.0083	0.0058
0.45	0.0693	0.0304	0.0117	0.0085
0.50	0.0954	0.0403	0.0187	0.0116

The simulation is conducted in highly fault-prone sensor networks to obtain a worst case performance and to reduce the simulation time involved. Hence the actual performance might be much better than that shown above. Much improved performance can be expected if we run the fault detection algorithm and adjust parameters more often. As the update interval decreases, the number of newly added faulty nodes also decreases, resulting in a relatively small increase in p between two adjacent update cycles. If the increase is 0.01 instead of 0.05, for example, both DA and FAR become almost perfect for a wide range of p.

In the case of a regularly deployed sensor network, the performance of the fault detection scheme is more predictable. From the simulation of four regularly deployed sensor networks with the node degree of 4, 8, 12, and 20, we have observed very similar performance trends with high DA and low FAR.

5 Conclusion

In this paper, we presented an adaptive scheme for detecting faulty sensor nodes in wireless sensor networks. Faulty nodes are identified locally in a distributed manner. Each sensor node makes a decision based on the information obtained from its neighboring nodes. To cope with the low scalability problem of existing fault detection algorithms parameters for fault detection are dynamically adjusted to adapt to the changing network status. The resulting improvements in performance were observed by computer simulation. Both detection accuracy and false alarm rate are controlled to be acceptably high and low, respectively, even with the increasing number of faulty nodes. Simplicity of the fault detection scheme is emphasized to make it practically useful. Transient faults are treated properly to utilize the sensor nodes as much as possible by screening out only the faulty nodes repeatedly generating erroneous measurements. The scheme was also applied to regularly deployed sensor networks. Similar performance was observed, as expected. The relation between the fault detection cycle and the mean-time-between-failures (MTBF) needs to be further investigated to determine the time to adjust the parameter values.

Acknowledgements: This work was supported by Grant No. R01-2006-000-10073-0 from the Basic Research Program of the Korea Science and Engineering Foundation.

References:

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cyirci, ”Wireless Sensor Networks: A Survey,” Computer Networks, vol. 38, no.4, 2002, pp. 393-422.

[2] Z. Bojkovic and B. Bakmaz, ”A survey on wireless sensor networks deployment,” WSEAS Transactions on Communications, vol.7 ,no. 12, Dec. 2008, pp. 1172-1181.

[3] V. Gupta and R. Pandey, ”Data fusion and topology control in wireless sensor networks,” WSEAS Int. Conf. Applied Electromagnetics, Wireless and Optical Communications, Tenerife, Spain, Dec. 2007, pp. 135-140.

[4] C. Jaikaeo, C.Srisathapornphat, C-C. Shen, ”Diagnosis of sensor networks,” Int. Conf. Communications, vol 5, pp. 1627-1632, June 2001.

[5] F. Koushanfar, M. Potkonjak, A. Sangiovanni-Vincentelli, On-line fault detection of sensor measurements, IEEE Sensors 2 (2003) 974.980.

[6] S. Chessa, and P. Santi, ”Comparison-based system-level fault diagnosis in ad hoc networks,” 20th Symp. Reliable Dist. Syst., 2001, pp. 257-266.

[7] M. Elhadef, A. Boukerche, and H. Elkadiki, ”Peformance analysis of a distributed comparison-based self-diagnosis protocol for wireless ad hoc networks,” Int. Workshop Modeling Analysis and Simulation of Wireless and Mobile Systems, 2006, pp. 165-172.

[8] L.B. Ruiz, I.G. Siqueira, L.B. Oliveira, H.C. Wong, J.M.S. Nogueira and A.A.F. Liureiro, ”Fault management in event-driven wireless sensor networks,” MSWIM’04, 2004.

[9] B. Krishnamachari, and S. Iyengar, ”Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks,” IEEE Transactions on Computers, Vol. 53, No. 3, March 2004.

[10] X. Luo, M. Dong, and Y. Huang, ”On distributed fault-tolerant detection in wireless sensor networks,” IEEE Trans. Computers, Vol. 55, No. 1, Jan 2006, pp. 58-70.

[11] D. Curiac, C.Volosencu, A. Doboli, O. Dranga, and T. Bednarz, ”Neural network based approach for malicious node detection in wireless sensor networks,” Proc. WSEAS Int. Conf. on Circuits, Systems, Signal and Telecommunications, Jan. 2007, pp. 17-19.

[12] W.R. Pires Jr., T.H. Figueiredo, H.C. Wong, and A.A. Loureiro, ”Malicious node detection in wireless sensor networks,” IEEE 18th Int. Parallel and Distributed Processing Symp, 2004.

[13] D. Curiac, O. Banias, and O. Dranga, ”Malicious node detection in wireless sensor networks using an autoregression technique,” Third Int. Conf. Networking and Services, 2007.

[14] M. Ding, D. Chen, K. Xing and X. Cheng, ”Localized fault-tolerant event boundary detection in sensor networks,” IEEE INFOCOM, Mar 2005, pp. 902-913.

[15] J. Chen, S. Kher and A. Somani, ”Distributed fault detection of wireless sensor networks,” Proc 2006 Workshop DIWANS, pp. 65-72.

[16] C. Li and C.K. Liang, ”A localized fault detection algorithm in sensor networks,” Wireless and Optical Communications, 2006.

[17] M.H. Lee and Y.-H. Choi, ”Fault detection of wireless sensot networks,” Computer Communications 31, 2008, pp. 3469-3475.

[18] M. Yu, H. Mokhtar, and M. Merabti, ”Fault management in wireless sensor networks,” IEEE Wireless Communications, Dec. 2007, pp. 13-19.