Expert Systems with Applications 41 (2014) 3703–3711

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Probabilistic fault detector for Wireless Sensor Network

Bill C.P. Lau [a], Eden W.M. Ma [a], Tommy W.S. Chow [a,b,*]

[a] Centre for Prognostics and System Health Management, City University of Hong Kong, Hong Kong, China
[b] Department of Electronic Engineering, City University of Hong Kong, Hong Kong, China

ARTICLE INFO

Keywords:
Fault detection
Wireless Sensor Network
Naïve Bayes
Maximum Likelihood Estimation

ABSTRACT

This paper proposed a novel centralized hardware fault detection approach for a structured Wireless Sensor Network (WSN) based on Naïve Bayes framework. For most WSNs, power supply is the main constraint of the network because most applications are in severe situation and the sensors are equipped with battery only. In other words, the battery's life is the network's life. To maximize the network's life, the proposed method, Centralized Naïve Bayes Detector (CNBD) analyzes the end-to-end transmission time collected at the sink. Thus all the computation will not be performed in individual sensor node that poses no additional power burden to the battery of each sensor node. We have conducted thorough performance evaluation. The obtained results showed better performance can be obtained under a network size of 100-node WSN simulations at various network traffic conditions and different number of faulty nodes.

1. Introduction

The proliferation in Micro-Electro-Mechanical Systems (MEMS) technology, which makes deploying low-cost, large-scale, and dense small sensor nodes to collect information from harsh environment feasible, has resulted in the emerging of WSNs. A WSN may consists of hundreds or thousands distributed autonomous sensors, which equipped with sensing, computation, and wireless communication devices to monitor or collect information from various environments including battle fields, remote geographical regions, industrial plants, and office buildings (Erdelj, Mitton, & Natalizio, 2013; Geeta, Nalini, & Biradar, 2013; Taneja, Krioukov, Dawson-Haggerty, & Culler, 2013). Nowadays, WSNs has been widely applied in many different applications like railway security (Daliri, Shamshirband, & Besheli, 2011), transportation system (Ray, Goel, & Chandra, 2011), environmental monitoring (Othman & Shazali, 2012), forest fire detection (Aslan, Korpeoglu, & Ulusoy, 2012), and healthcare (Alemdar & Ersoy, 2010).

Sink/sensor pair is a common architecture of WSNs. The sensors are in charge of measuring the environmental status, which may vary with time and space; collaborating with each other; and forwarding the measured data to the sink. The sink is responsible for integrating, analyzing data received from sensors and responding users and applications accordingly (Hsieh, Leu, & Shih, 2010). There are two types of sensor deployment: structured and unstructured. In an unstructured WSN, a dense collection of sensor nodes is deployed in an ad hoc manner into the field. Once deployed, the network is left unattended to perform monitoring and reporting functions. The numerous nodes and ad hoc topology make the network maintenance such as managing connectivity and detecting failures very difficult. In a structured WSN, all or some of the sensor nodes are deployed in a planned manner; hence, fewer nodes are required for the same coverage of the unstructured WSN (Yick, Mukherjee, & Ghosal, 2008). This lowers the network maintenance and management cost in a structure WSN.

The design and resource constraints of a wired network and that of a WSN are quite different. Resource constraints of a WSN include limited amount of energy, short communication range, low bandwidth, and limited processing power and storage in each sensor node. Design constraints are application dependent and are based on the monitored environment (Yick et al., 2008). Due to these constraints, the sensor nodes may fail to perform correct operations. Moreover, the connection between sensor nodes is prone to temporary or permanent failure under severe environments. A successful packet transmission from sensor node to sink is relying on correct propagation among sensor nodes; hence, node failure can severely influence the network performance. A diagnosis mechanism becomes necessary to ensure the operations are correct and the data collected are meaningful to the user (You et al., 2011).

As mentioned before, the network life time depends on the sensors' life. As the sensors are often deployed in an uncontrolled or even harsh environment, they are prone to having faults (Lee & Choi, 2008). Compared to traditional integrated semiconductor chips, sensors and actuators boarded on a MEMS node have higher chance to be faulty (Khan, Daachi, & Djouani, 2012). These

* Corresponding author at: Department of Electronic Engineering, City University of Hong Kong, Hong Kong. Tel.: +852 3442 7756.
E-mail address: eetchow@cityu.edu.hk (T.W.S. Chow).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.11.034

properties pose significant challenges to maintaining high quality of service of WSNs. Therefore, efficient and effective fault management deems to be essential for maintaining a robust WSNs service. Yu, Mokhtar, and Merabti (2007) discussed three phases of fault management process. Fault management aims to identify the faulty sensor nodes, and to exclude them from the network. Fault detection is a basic fault management task in WSNs.

There is a trade-off between prolonging the network lifetime by conserving the energy of individual nodes and maintaining the high quality of network services by implementing complex fault management schemes in the network (Yu et al., 2007). In order to minimize the resources consumption and to preserve the energy of nodes, our proposed method is designed to detect and analyze faulty sensor node(s) using data collected at the sink rather than implementing a complex faulty management scheme. The presented results show the proposed method is effective and reliable. Also, the proposed Naïve Bayes framework is the first of its kind to be deployed for performing WSN faulty node(s) detection. Because of the structure of Naïve Bayes classifier, the proposed method is computational efficient. A simulation environment using Zigbee protocol has been set up for the verification of the proposed method.

In this paper, a novel approach, CNBD was proposed to identify the possible faulty sensor node using Naïve Bayes framework. A new attribute, the end-to-end transmission time of each packet arrived at the sink is analyzed for determining the network status. CNBD does not involve any additional protocol and extra resource consumption of sensor nodes while it suggests a list of suspicious faulty nodes to the user. The rest of the paper is organized as follows. Section 2 discusses the related work. In Section 3, the procedures of CNBD is discussed. Section 4 discusses the simulation environment, results and the possible future development. Section 5 concludes the paper.

2. Related works

2.1. Mechanism of Wireless Sensor Network

WSN is a network consists of sensor devices, called *nodes,* and controller, called *sink*. The nodes, measure the environmental parameters and forward these measurements to the sink, which has no constraint on power, through wireless communication. Fig. 1 shows a simple WSN topology.

There are different communication protocols for WSNs; and each protocol has its own characteristics for different applications. The popular communication protocols include Zigbee/802.15.4, IEEE 1451, WirelessHART, ZigBee IP, and 6LoWPAN. In this paper, the simulator is built using Zigbee/802.15.4 protocol because Zigbee aims at constructing a WSN with low cost, low power consumption, low complexity, and low data transmission rate.

There are two common congestion scenarios: node-level and link-level. Node-level congestion is caused by a buffer overflow in the node when link-level congestion is caused by too many nodes requesting the same node for data transmission simultaneously. Under Zigbee standard, signal from node to sink will travel through the shortest path in normal situation. If any packet losses due to hardware failures or congestions, the signal path will be changed (Fig. 2). It results in higher energy consumption and longer end-to-end packet transmission time.

2.2. Fault detection in Wireless Sensor Networks

Different from wired networks, fault management for WSNs concerns a given region rather than a given link between two nodes. Yu et al. (2007) stated the fault management schemes vary in form of architecture, protocols, and detection algorithms. Generally, the fault management for WSNs can be divided into three phases: fault detection, fault diagnosis, and recovery. In this paper, only fault detection will be discussed.

The fault detection technology can be generally classified as centralized approaches and distributed approaches. Briefly, the sink in the centralized approach usually has uninterrupted power supply and makes the diagnostic decisions by periodically injecting requests or queries to other nodes and waits for replies. In distributed approaches, the updated network status and individual node performance was assessed according to the status reporting messages from nodes or data comparison with the neighbors advancing from the concept in Sengupta and Dahbura (1992).

The recent works on network data fault detection include the use of Takagi–Sugeno–Kang fuzzy inference system (Khan et al., 2012), statistical based Auto-regression (Volosencu, 2012), and Bayesian network (De Paola, Lo Re, Milazzo, & Ortolani, 2013).

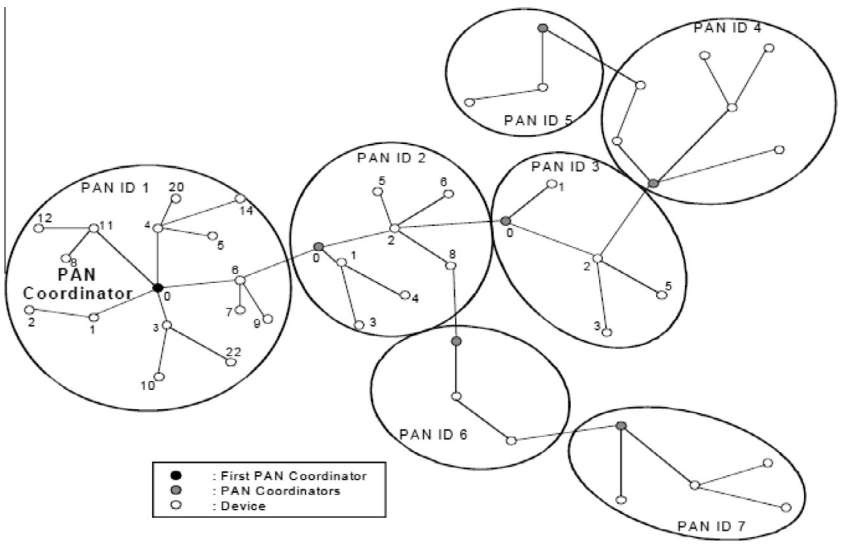

Fig. 1. A simple WSN topology (Fig. 2 in IEEE standard for information technology – telecommunications and information exchange between systems – local and metropolitan area networks – specific requirements – Part 15.4: Wireless MAC and PHY specifications for low-rate WPANs, 2006).

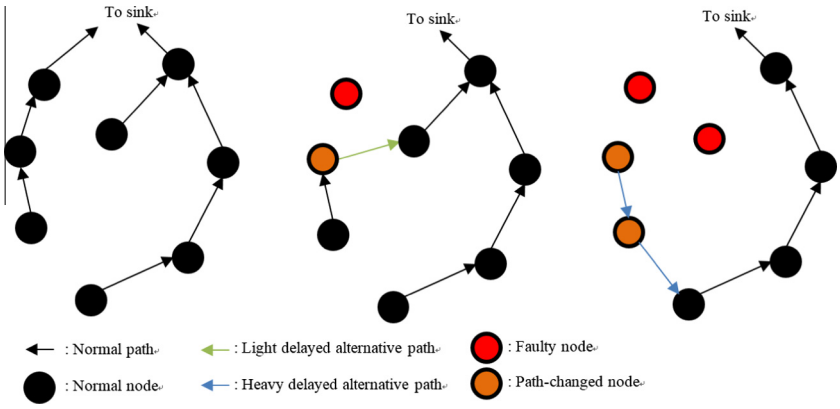

Fig. 2. Different cases of path change.

The Takagi–Sugeno–Kang fuzzy framework was used to model individual sensor node through training with input of neighboring sensor's measurements and an output of its actual measurement. Thus, a node can be identified as faulty when the system output exceeds a predetermined threshold. But the setting of the threshold may not be straightforward as the threshold value may vary with different network structures and applications. A statistical based Auto-regression was used to estimate the forthcoming sensor data based on the previous readings and compare with the real value. If the deviation exceeds the acceptable tolerance, the sensor is reported as failure. Probabilistic approach was also employed to solve the problem. They first separated the network into clusters and used Bayesian network to model all the collected sensor nodes readings. Thus, the trained Bayesian network is able to identify whether the newly acquired reading was generated from a faulty node or not. But it is noticed that sensor nodes are often placed in harsh environment or outdoors. The environmental conditions may fluctuate which caused false alarms. Moreover, hardware faults like physical damages and battery depletion were overlooked. Geeta et al. (2013) tackled the battery failure and other environmental damages by battery power model and interference model. When a node is in low battery power, all services of the node will transfer to neighboring nodes with the highest battery power. However, the interchange of battery capacity information between nodes consumes extra energy from the usual node operations. It is also noted that most algorithms were designed in a distributed approach which results in draining more battery power for performing assessment computations.

The contribution of our proposed probabilistic CNBD is threefolded. First, all detection computations are handled by the sink node which uses no battery power of each sensor node. This approach will have solved many intrinsic sensor node battery problems. Second, the end-to-end transmission time of data packets is always provided by the protocol, which means no additional power is needed for reporting the of each sensor node. Third, Naïve Bayes classifier, a computational efficient and robust probabilistic mechanism, is newly introduced to WSN fault detection. Our obtained results indicate that the proposed framework is capable of handling reliable fault detection task even under a large sensor network of 100 nodes.

2.3. Naïve Bayes modeling

Naïve Bayes is a simple, fast and accurate classifier based on Bayes' theorem with independent assumption. Naïve Bayes was used in many classification applications such as text data-mining (Youn & Jeong, 2009), medical data-mining (Soria, Garibaldi, Ambrogi, Biganzoli, & Ellis, 2011), network intrusion detection (Koc, Mazzuchi, & Sarkani, 2012), and Cheminformatics (Mussa, Mitchell, & Glen, 2013). When we have m classes denoted as $C1, C2, ..., Cm$ and n-dimensional vector for a class t is $D_{Ct} = \{d_{Ct1}, d_{Ct2}, \cdots, d_{Ctn}\}$, where $\sum_i d_{Cti} = 1$ and d_{Cti} is the probability that data i occurs in class t. $S = \{s_1, s_2, ..., s_k\}$ is the total k senses of network operation. The likelihood of scene s_1 is a product of the data that appear in the scene,

$$P(s_1|D_{Ct}) = \frac{(\sum_i N_i)!}{\prod_i N_i!} \prod_i (d_{Cti})^{N_i} \tag{1}$$

where N_i is the number of data i in scene s_1.

The largest posterior probability L provides the most suitable decision of the classification task with prior distributions of all classes $P(D_{Ct})$. It is presented as the following,

$$L = \arg\ \max_c \left[\log P(D_{Ct}) + \sum_i N_i \log d_{Cti} \right] \tag{2}$$

The prior distributions are found during training phase by Maximum Likelihood Estimation (MLE). When the testing attribute values were collected, the classification can be done by equation (2).

2.4. Maximum Likelihood Estimation

There are many sensor nodes involved even in a typical WSN. In other words, there are so many fault conditions and it is not feasible to have enough training sample to determine the conditional probabilities for all cases. Thus, MLE is used to ease the training sample requirement for estimating the conditional Probability Density Function (PDF). Assume the training attribute values $S = \{s_1, s_2, ..., s_n\}$ have a joint density denoted,

$$f_\theta(s_1, s_2, \ldots, s_n) = f(s_1, s_2, \ldots, s_n|\theta) \tag{3}$$

The MLE of θ is to maximize the likelihood function but that would be quite tedious. The log likelihood is usually maximized instead:

$$l(\theta) = \sum_{i=1}^{n} \ln(f(s_i|\theta)) \tag{4}$$

where S is independent and identically distributed. The maximum likelihood estimator $\hat{\theta}$ can be estimated by finding the derivative.

In general, the MLE estimates the PDFs in both normal scenes and faulty scenes according to the training data from ideal and congested networks. There are two advantages to use MLE. First, manageable amount of training data is used to build the PDFs for Naïve Bayes estimation. Second, the estimated PDFs can still provide probabilities for extreme attribute value inputs.

3. Centralized Naïve Bayes detector

Based on operation characteristics of a WSN, the packet transmission time was assumed that behaving like Exponential PDF. MLE is used to estimate conditional probability during training phase.

Fig. 3 showed the general process flow of CNBD and the process details were described as follows:

(1) For both training and testing phase, only the information of the packets such as end-to-end packet transmission time, and source node IDs received by sink were analyzed. The network status could be normal or faulty. If the class label is normal, the network consists of no faulty sensor. If the class label is faulty, the network consists of at least one faulty sensor.

In the training phase,

(2a) The training process was started from the data obtained from the normal class first. When the normal class data were processed, the minimum time value of each node was extracted as an anomaly detection threshold. In typical WSN topology, packets are sent from a node, which picks

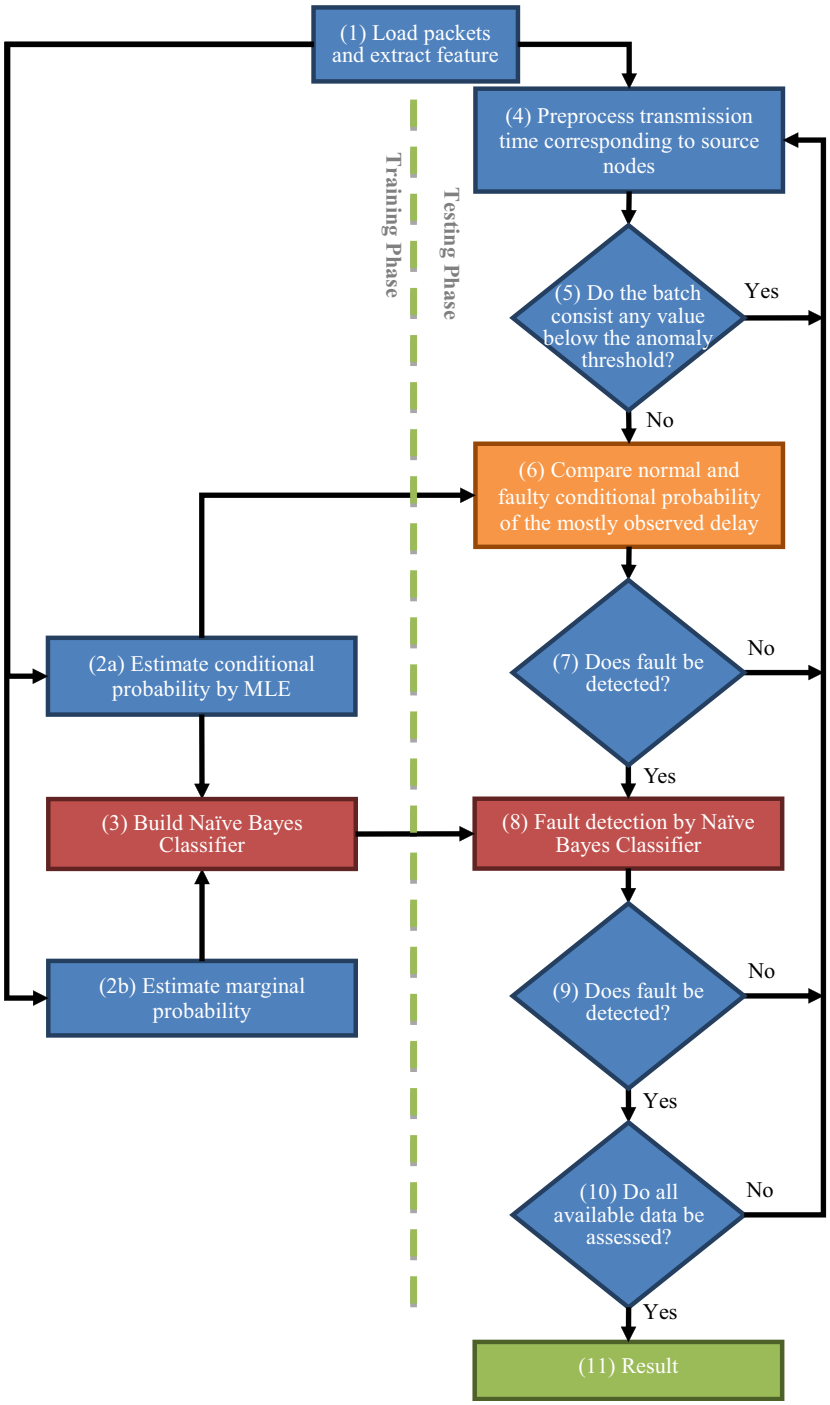

Fig. 3. Implementation of the CNBD.

up the parameter value, to the sink in a many-to-one mechanism. In other words, the effect of faulty nodes to the transmission is depending on the position of faulty node within the topology. If the faulty node is a leave node, the signal cannot be picked up anymore. If the faulty node is at the only path to the sink, signals of the whole branch cannot reach the sink. Only the transmission time deviated from the anomaly detection threshold will be investigated. Both normal and faulty conditional PDF parameter(s) were estimated by MLE for each node.

(2b) The marginal probability of both classes (i.e. normal and faulty) was estimated according to the class labels of the training dataset.

(3) The conditional probability and marginal probability obtained from step (2a) and step (2b), respectively, were used to build the Naïve Bayes classifier, which will be used in step (8) to determine the network status during testing phase.

In the testing phase,

(3) Packets received at sink will be analyzed in a batch (every 1000 received packets as a batch in this study). The end-to-end transmission time of all packets in a batch were grouped according to their source node. Each group was passed to step (5) to check whether the transmission path consists of any faulty node.

(4) As congestion is a common phenomenon during packet transmissions, the end-to-end transmission time may be longer than that of the congestion-free network even there is no faulty node within the path. To ease the confusion between the congestion and real faulty node, every packet group was compared with its corresponding anomaly threshold. If all end-to-end transmission time of a group is lower than the anomaly threshold, the path having at least one faulty node is assumed. In other words, if there is one transmission time value lower than the anomaly threshold, we assume that the longer transmission time is caused by the congestion instead of faulty node.

(5) There may have different end-to-end transmission time value within the packet group due to different traffic situations. The mode value of the transmission time will be used for further analysis. The normal and faulty conditional probability of the mode value within each packet group were estimated and compared with the trained PDFs.

(6) If the faulty conditional probability of the mode value is higher than the normal conditional probability of the mode value, this transmission time will be suspected from a faulty network. Otherwise, it will be assumed from a normal network.

(7) As the mode value may due to the congestion not the faulty node, further investigation is needed. To confirm the network status, the last five transmission times are analyzed by a Naïve Bayes classifier. If there were less than five consecutive values, all the data were used for the estimation. The result obtained from the classifier may override the result obtained from step (7).

(8) If the packet group was defined as from a faulty network, the source node will be defined as a suspicious faulty node. The suspicious faulty node list was updated.

(9) Step (5–9) will be repeated until all packet group are analyzed.

(10) The network status and the suspicious fault node list are reported according to the testing scenes.

4. Results and discussions

4.1. Simulation and implementation

A simulator modeling typical WSNs using Zigbee with hardware fault is developed. A 100-node topology was random generated by adjacency matrix with transmission cost ranging from 1–200. There is always one sink in the network and it operates with unlimited power supply. Data were randomly picked up by nodes and were forwarded to the path heading the sink. Then, the nodes will pack the data to be packets and transmit towards the sink node. Different situations are simulated. Two parameters, traffic congestion condition and number of faulty nodes within the network, are set. Three level of traffic congestion condition are considered. They are congestion-free, light congested, and heavy congested. Under each traffic congestion condition, no fault network and different numbers of faulty nodes within the network are generated. The number of faulty nodes is varied from 1 to 5. For a 100-node topology, there are 100, 4950, 161,700, 3,921,225, and 75,287,520 combinations of faulty nodes for the number of faulty nodes varying from 1 to 5, respectively. It is tedious to go through all combinations; we cap the number of combinations at 5000. Hence, the total scenarios for each traffic congestion level are 20,050. In other words, total 60,150 faulty scenes are generated. Under each scenario, 2000 data packets are generated by sensor nodes randomly.

In this study, CNBD was compared with two performance evaluation methods, Marginal Fault Detector (MFD) and Historical Fault Detector (HFD). MFD made use of the normal data to train the decision making thresholds for testing data from each node. When the transmission time varied due to congestion, the minimum value was selected to be the threshold. If there were any new data larger than the threshold, the end-to-end packet transmission path will be classified as an alternative path from a faulty network and the source node will be marked as a suspicious node.

When MFD uses all normal data to train its threshold, HFD, and CNBD used 60% of faulty data and the similar amount of normal data for training. The rest 40% of faulty data were used for method verification. HFD recorded the transmission time of both normal and faulty scenes for each source node at the same traffic condition. If the same transmission time was found in both normal and faulty scenes record, the value in the faulty scenes record was erased. The network and the source node were classified as faulty and suspicious faulty node respectively if there were testing data equaled to the trained faulty scenes record. CNBD was implemented according to Section 2.3. Fig. 4 shows the Probability Density Functions (PDFs) obtained by MLE of exponential distribution for both normal and faulty end-to-end transmission time. In Fig. 4(a), the faulty and normal PDF from one node are compared. In the Fig. 4(b), it shows the overall comparison between faulty and normal PDFs from the whole network. It can be seen that faulty transmissions would have caused longer transmission time compared with normal transmission. To evaluate the performance of these methods, Scenes Hit Rate, Hit Ratio, and False Alarm Rate are also compared.

4.2. Scenes Hit Rate comparison

Totally 60,153 scenarios for three different traffic conditions were generated in this study. Scenes Hit Rate is the ratio of suspicious nodes matched the theoretical faulty nodes to the total scenes. Fig. 5 shows the Scenes Hit Rate of MFD, HFD, and CNBD for different traffic conditions. CNBD achieved the best result of Scenes Hit Rate in the cases with congestions. The two fault detector did not work well in light traffic congestion and heavy traffic

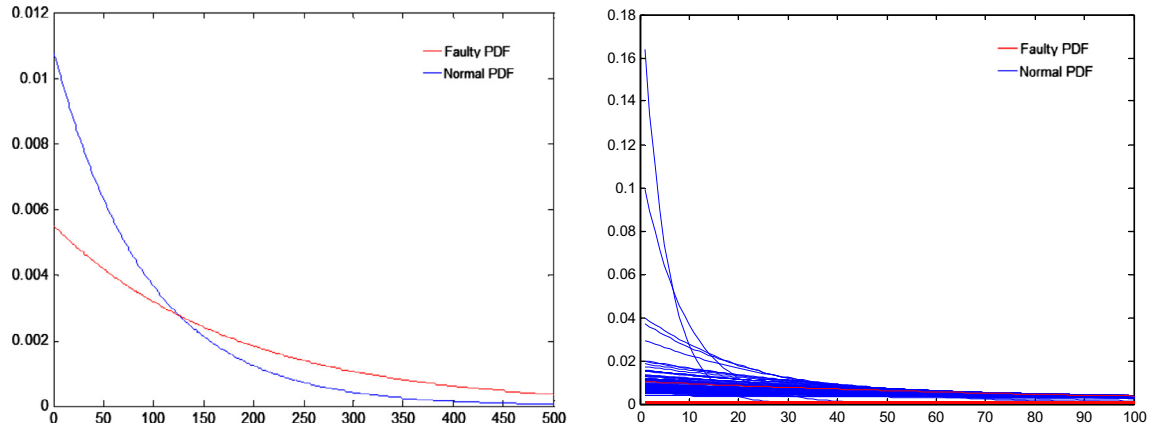

Fig. 4. Exponential distribution of transmission time in normal and faulty scenes.

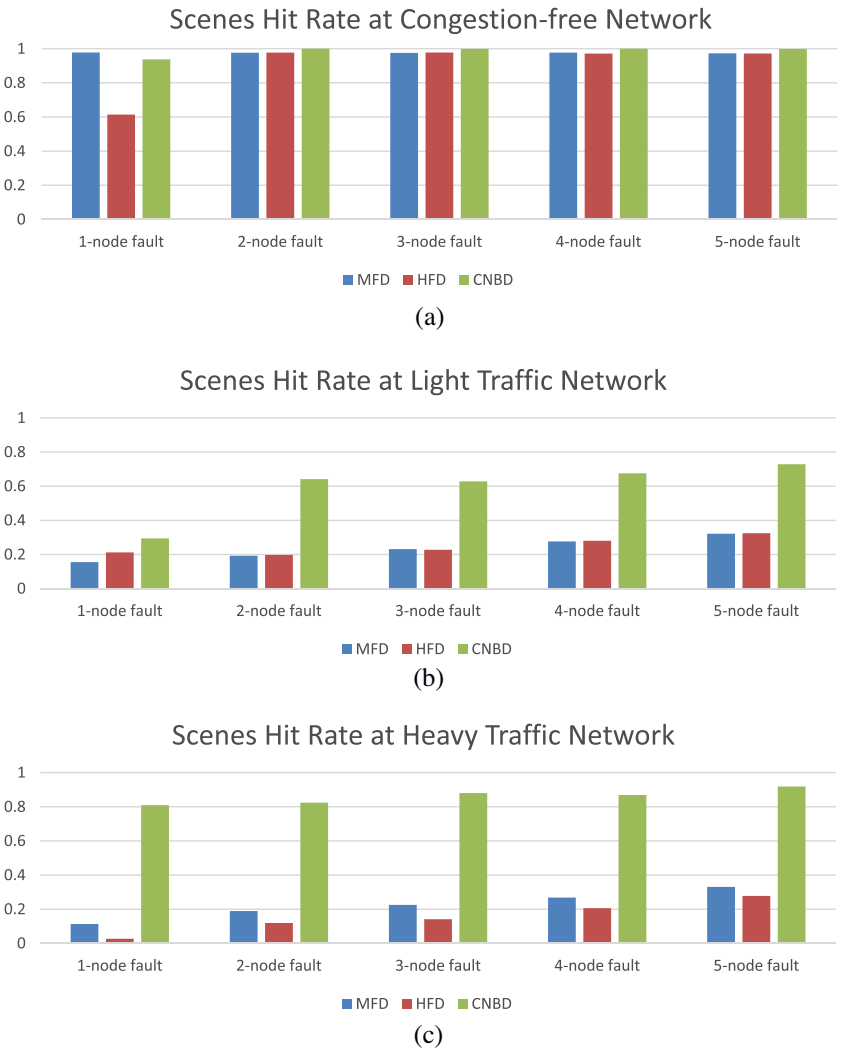

Fig. 5. Scenes Hit Rate of MFD, HFD, and CNBD at (a) congestion-free network, (b) light traffic network, and (c) heavy traffic network.

congestion cases because there are too many possible values in both normal and faulty scenes. Their database or thresholds cannot provide enough information to judge the cause of the longer transmission time was from congestions or from a faulty network. CNBD could achieve 60% or higher rate except the "one-fault with light traffic" case. It was because the faulty scenes were limited in one-fault cases and the training data were not enough for Naïve Bayes classifier to train the conditional probabilities. More faulty

data were used to establish the conditional probability, higher the Naïve Bayes classification accuracy.

4.3. Hit Ratio comparison

Hit Ratio is the ratio of correctly detected faulty transmission time against the total faulty transmission time. CNBD and HFD performed the best and the worst in the comparison respectively. The Hit Ratio decreased when the faulty nodes increased in the congested scenes. It was because the longer transmission time was mainly caused by the network congestion and the prolonged transmission time of alternative path became a small fraction of the reason of the longer transmission time. The conditional probability of faulty transmission time was similar to the conditional probability of normal transmission time. There was a higher probability that the data from faulty network was classified to be data from normal network. Thus, the Hit Ratio decreased when the faulty nodes increased (see Fig. 6).

4.4. False Alarm Rate comparison

False Alarm Rate is the rate of misclassified suspicious nodes against the total suspicious nodes. The congested scenes suffered in high False Alarm Rate in all methods because both faulty nodes and congestions caused the similar longer transmission time. It was not easy to have an accurate classification. False Alarm Rate decreased when the faulty node number increased in congested scenes. It was because more the faulty nodes, higher the possibility to be detected.

Refer to the results, end-to-end packet transmission time is possible to be used for fault detection of the WSN. CNBD provided a higher Hit Ratio for congestion-free traffic condition. It means more nodes using alterative transmission path were detected. That helps the rangers to identify the specific location/region of the faulty node(s). Although the False Alarm Rate of CNBD is slightly higher than that of other methods, it is an acceptable range, say within 5%. Moreover, the disadvantage of slightly higher False Alarm Rate was overcome by the advantage of the far higher Hit Ratio that a more comprehensive suspicious node list helps the rangers to identify the faulty nodes more easily in a large network. All methods suffered in a high False Alarm Rate and around 60% of alterative transmission paths were detected if traffic congestion exists. In other words, only analyzing end-to-end packet transmission time may not be adequate for fault detection for congestion situation because the end-to-end packet transmission time cannot show the reason behind the long transmission time i.e. caused by the faulty node or by congestion (see Fig. 7).

In summary, CNBD newly introduced Naïve Bayes Classifier to large WSN fault detection. The detection attribute, end-to-end transmission time is never mentioned in the literature and it minimized the extra battery power used for health assessment of sensor nodes. CNBD is highly recommended for large WSNs with

Fig. 6. Hit Ratio of MFD, HFD, and CNBD at (a) congestion-free network, (b) light traffic network, and (c) heavy traffic network.

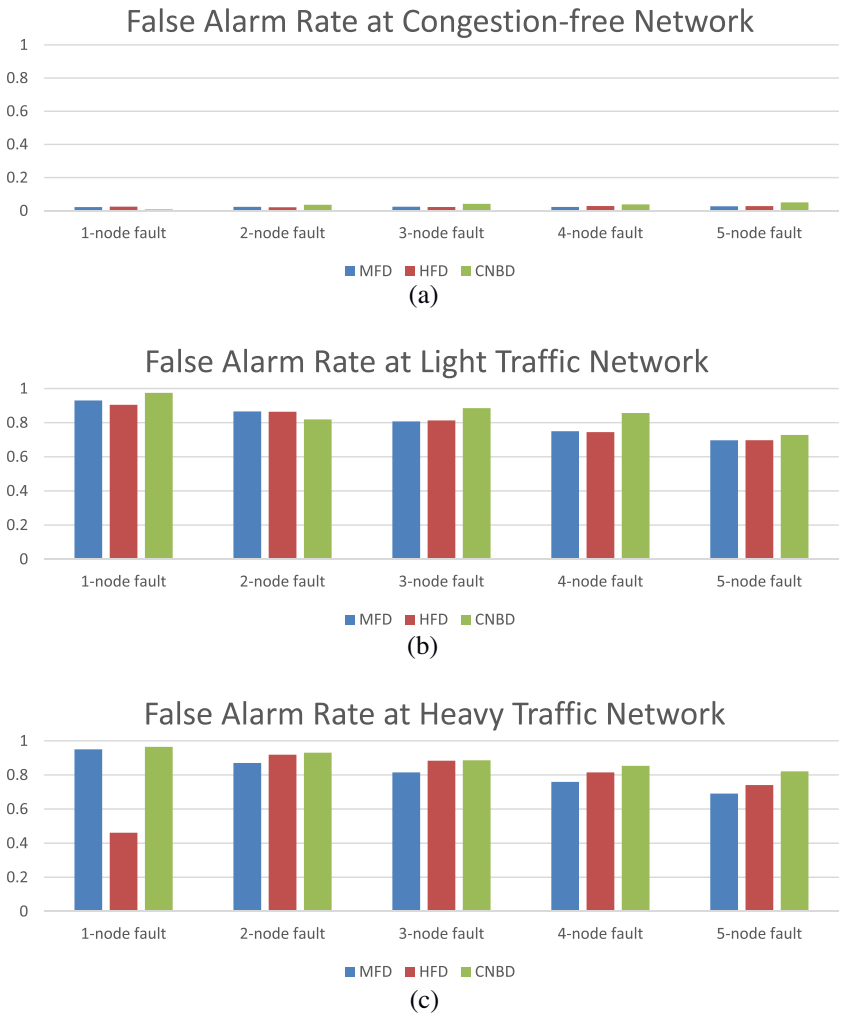

Fig. 7. False Alarm Rate of MFD, HFD, and CNBD at (a) congestion-free network, (b) light traffic network, and (c) heavy traffic network.

low frequency of data acquisition such as the monitoring of indoor environment, agricultural environment, and transportation.

In future, further improvement is needed in three aspects. First, more attributes are needed to improve the performance in congested network. Second, the end-to-end transmission time depends on the deployment of sensor nodes. Mobility of sensor nodes changes the topology of the network and end-to-end transmission time may have large variations. An online updating learning mechanism for the Naïve Bayes framework will be useful for a high mobility WSN. Last, failure of edge nodes does not affect the transmission time of other nodes. More efficient method is needed to detect the failure of those nodes.

5. Conclusions

A novel centralized fault detection method for WSN based on Naïve Bayes framework was introduced. The recent researches mostly focused on the detecting data faults but the battery depletion problem of sensor nodes was overlooked. To raise the energy efficiency of sensor nodes, a new attribute, the end-to-end packet transmission time from source node to the sink was extracted from the communication protocol for determining the network status. If it is a faulty network, a suspicious faulty node list is provided for further investigation. Simulations with three network traffic conditions and faulty node numbers ranging from one to five were used to evaluate the performance of different methods. The fault detectors were only effective in congestion-free scenes while CNBD provided a better Scene Hit Rate and Hit Ratio with slightly worse False Alarm Rate. Further improvement was suggested in detection attributes, mobility and method efficiency aspects.

Acknowledgment

The work described in this paper was partially supported by a collaborative project associated with China Electronic Product Reliability and Environmental Testing and Research Institute (CEPREI) from Guangdong Provincial Department of Science and Technology (Project number: 2011A011302002).

References

Alemdar, H., & Ersoy, C. (2010). Wireless sensor networks for healthcare: A survey. *Computer Networks, 54*(15), 2688–2710.

Aslan, Y. E., Korpeoglu, I., & Ulusoy, Ö. (2012). A framework for use of wireless sensor networks in forest fire detection and monitoring. *Computers, Environment and Urban Systems, 36*(6), 614–625.

Daliri, Z. S., Shamshirband, S., & Besheli, M. A. (2011). Railway security through the use of wireless sensor networks based on fuzzy logic. *International Journal of the Physical Sciences, 6*(3), 448–458.

De Paola, A., Lo Re, G., Milazzo, F., & Ortolani, M. (2013). QoS-aware fault detection in wireless sensor networks. *International Journal of Distributed Sensor Networks, 2013*, 1–12. http://dx.doi.org/10.1155/2013/165732.

Erdelj, M., Mitton, N., & Natalizio, E. (2013). Applications of Industrial Wireless Sensor Networks. In: V. Ç. Güngör, G. P. Hancke (Eds.), *Industrial Wireless Sensor*

Networks: *Applications Protocols and Standards*, CRC Press, (pp. 3–27). Retrieved from <http://hal.archives-ouvertes.fr/docs/00/78/86/29/PDF/CRC2012.pdf>.

Geeta, D. D., Nalini, N., & Biradar, R. C. (2013). Fault tolerance in wireless sensor network using hand-off and dynamic power adjustment approach. *Journal of Network and Computer Applications, 1–12*. http://dx.doi.org/10.1016/j.jnca.2013.02.005.

Hsieh, H.-C., Leu, J.-S., & Shih, W.-K. (2010). A fault-tolerant scheme for an autonomous local wireless sensor network. *Computer Standards & Interfaces, 32*(4), 215–221. http://dx.doi.org/10.1016/j.csi.2009.11.012.

IEEE standard for information technology – telecommunications and information exchange between systems – local and metropolitan area networks – specific requirements – Part 15.4: Wireless MAC and PHY specifications for low-rate WPANs. (2006). In: *IEEE Computer Society* (Vol. 2006, p. 305).

Khan, S. A., Daachi, B., & Djouani, K. (2012). Application of fuzzy inference systems to detection of faults in wireless sensor networks. *Neurocomputing, 94*, 111–120. http://dx.doi.org/10.1016/j.neucom.2012.04.002.

Koc, L., Mazzuchi, T. A., & Sarkani, S. (2012). A network intrusion detection system based on a hidden Naïve Bayes multiclass classifier. *Expert Systems with Applications, 39*(18), 13492–13500. http://dx.doi.org/10.1016/j.eswa.2012.07.009.

Lee, M.-H., & Choi, Y.-H. (2008). Fault detection of wireless sensor networks. *Computer Communications, 31*(14), 3469–3475. http://dx.doi.org/10.1016/j.comcom.2008.06.014.

Mussa, H. Y., Mitchell, J. B., & Glen, R. C. (2013). Full "Laplacianised" posterior Naïve Bayesian algorithm. *Journal of Cheminformatics, 5*(1), 37. http://dx.doi.org/10.1186/1758-2946-5-37.

Othman, M. F., & Shazali, K. (2012). Wireless sensor network applications: A study in environment monitoring system. *Procedia Engineering, 41*, 1204–1210.

Ray, S., Goel, A., & Chandra, N. (2011). Accident detection by wireless sensor network and sending rescue message with GPS. *Journal of Computing, 3*(11), 69–73.

Sengupta, A., & Dahbura, A. T. (1992). On self-diagnosable multiprocessor systems : Diagnosis by the comparison approach. *IEEE Transactions on Computers, 41*(11), 1386–1396.

Soria, D., Garibaldi, J. M., Ambrogi, F., Biganzoli, E. M., & Ellis, I. O. (2011). A "non-parametric" version of the Naïve Bayes classifier. *Knowledge-Based Systems, 24*(6), 775–784. http://dx.doi.org/10.1016/j.knosys.2011.02.014.

Taneja, J., Krioukov, A., Dawson-Haggerty, S., & Culler, D. (2013). Enabling advanced environmental conditioning with a building application stack. In: *2013 International Green Computing Conference Proceedings* (pp. 1–10). California 94720:Berkeley, IEEE. doi:10.1109/IGCC.2013.6604519.

Volosencu, C. (2012). Applying the technology of wireless sensor network in environment monitoring. In: C. Volosencu (Ed.), *Cutting Edge Research in New Technologies* (pp. 97–116). InTech. Retrieved from <http://cdn.intechopen.com/pdfs/35076/InTech-Applying_the_technology_of_wireless_sensor_network_ in_environment_monitoring.pdf>.

Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer Networks, 52*(12), 2292–2330. http://dx.doi.org/10.1016/j.comnet.2008.04.002.

You, Z., Zhao, X., Wan, H., Hung, W. N. N., Wang, Y., & Gu, M. (2011). A novel fault diagnosis mechanism for wireless sensor networks. *Mathematical and Computer Modelling, 54*(1–2), 330–343. http://dx.doi.org/10.1016/j.mcm.2011.02.018.

Youn, E., & Jeong, M. K. (2009). Class dependent feature scaling method using Naïve Bayes classifier for text data mining. *Pattern Recognition Letters, 30*(5), 477–485. http://dx.doi.org/10.1016/j.patrec.2008.11.013.

Yu, M., Mokhtar, H., & Merabti, M. (2007). Fault management in wireless sensor networks. *IEEE Wireless Communications*, 13–19.