Chen M, Yu G, Gu Y *et al.* An efficient method for cleaning dirty-events over uncertain data in WSNs. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 26(6): 942–953 Nov. 2011. DOI 10.1007/s11390-011-1191-y

An Efficient Method for Cleaning Dirty-Events over Uncertain Data in WSNs

Mo Chen[1,2] (陈 默), *Student Member*, *CCF*, *ACM*
Ge Yu[2,*] (于 戈), *Senior Member*, *CCF*, *Member*, *ACM*, *IEEE*, Yu Gu[2] (谷 峪), *Member*, *CCF*, *ACM*
Zi-Xi Jia[2] (贾子熙), and Yan-Qiu Wang[2] (王艳秋), *Student Member*, *CCF*, *ACM*

[1] *Software College, Northeastern University, Shenyang 110004, China*

[2] *College of Information Science and Engineering, Northeastern University, Shenyang 110004, China*

E-mail: {chenmo, yuge, guyu, jiazixi, wangyanqiu}@ise.neu.edu.cn

Received April 16, 2010; revised July 19, 2011.

Abstract Event detection in wireless sensor networks (WSNs) has attracted much attention due to its importance in many applications. The erroneous abnormal data generated during event detection are prone to lead to false detection results. Therefore, in order to improve the reliability of event detection, we propose a dirty-event cleaning method based on spatio-temporal correlations among sensor data. Unlike traditional fault-tolerant approaches, our method takes into account the inherent uncertainty of sensor measurements and focuses on the type of directional events. A probability-based mapping scheme is introduced, which maps uncertain sensor data into binary data. Moreover, we give formulated definitions of transient dirty-event (TDE) and permanent dirty-event (PDE), which are cleaned by a novel fuzzy method and a collaborative cleaning scheme, respectively. Extensive experimental results show the effectiveness of our dirty-event cleaning method.

Keywords wireless sensor networks, event detection, dirty-event, uncertainty, clean

1 Introduction

Wireless sensor networks (WSNs) are constituted by a large number of tiny autonomous nodes each with sensing, computing and wireless communication capabilities[1]. These unreliable, low-cost sensor nodes are deployed in the physical surroundings to gather and process information of the physical world[2-5]. One of the most important applications of WSNs is to monitor, detect, and report the occurrences of interesting events based on the presence of abnormal measurements. For example, pollutants in the air are monitored by chemical sensors to announce the presence of unusual high chemical concentration. Another example comes from the fire alarm system consisting of smoke detectors, which will give warnings when detects an unexpected fire event. These event-based applications require sensor nodes to report events to a sink node in a timely manner once an event is detected. Event detection techniques are different from data-driven and query-driven techniques, where nodes regularly report sensor readings to the sink node or respond to the queries periodically issued by a sink node.

There are two primary challenges for event detection. First, sensor nodes are often deployed in harsh or hostile environments, thus they are likely to have faults or subject to measurement errors. Faulty sensor nodes are prone to generate erroneous abnormal data, which possibly lead to false event reports. Second, due to the imperfection of physical devices and communication delay, it is often infeasible for sensors to obtain accurate readings. In other words, an important property of sensor data is uncertainty, which is an inherent aspect of the data. To solve the first problem, several fault-tolerant techniques have been studied [6-11]. All of the existing solutions, however, focus on event detection over certain sensor data. In this paper, we focus on effective event detection over uncertain data, which has not been addressed before to our best knowledge.

The primary contributions of this paper are as follows. 1) A novel cleaning model based on binary data mapping is proposed. Unlike traditional approaches,

Regular Paper

This research was supported by the National Basic Research 973 Program of China under Grant No. 2012CB316201, the National Natural Science Foundation of China under Grant Nos. 61003058, 60933001 and the Fundamental Research Funds for the Central Universities under Grant No. N090104001.

*Corresponding Author

the proposed mapping method considers the inherent uncertainty of each measurement. 2) Formulated definitions of dirty events and monitored events are described. Based on a binary data sequence, the dirty events are classified into two types, namely transient dirty-event (TDE) and permanent dirty-event (PDE). 3) A fuzzy cleaning method for TDE is proposed, which associates data cleaning with fuzziness. Moreover, a collaborative PDE cleaning scheme for the type of directional monitored events is proposed, which is based on ring-structured clusters. 4) Extensive experiments show the effectiveness of the proposed cleaning methods.

The rest of this paper is organized as follows. Section 2 introduces the related work. The cleaning model is described in Section 3. We discuss our cleaning methods in Section 4 and show our experimental results in Section 5. Section 6 concludes this paper.

2 Related Work

Our work is related with the existing work in two major categories: fault-tolerant event detection and probabilistic data managing.

2.1 Fault-Tolerant Event Detection

Event detection in WSNs has attracted much attention due to its importance in many applications[12-14]. In order to detect events accurately, several fault-tolerant event detection schemes are proposed to solve the fault-event elimination problem in WSNs[6-11]. To guarantee the detecting accuracy, a distributed probabilistic Bayesian fault-tolerant algorithm is proposed in [6] to eliminate and correct the faulty sensor readings by considering the spatial correlation of the readings from nearby sensors. Luo *et al.*[7] proposed a fault-tolerant energy-efficient event detection paradigm. Ding *et al.*[8] proposed localized fault-tolerant event boundary detection algorithms for the identification of faulty sensors and the detection of the events, which are purely localized and scale well to large sensor networks. However, these studies focus on the spatial information to detect faulty nodes, whereas we consider both the temporal and spatial information in our work. Elmoustapha and Riley[11] presented preliminary steps leading to a geometric based approach to fault-tolerance in distributed detection using sensor networks. In [9], a fault-tolerant event boundary detection algorithm using the clustering technique based on a maximum spanning trees is presented. Sensor nodes are classified into two clusters by the distances, based on which the event boundary nodes are determined. Sorabh *et al.*[10] proposed a scalable and efficient scheme for detecting large-scale physically-correlated events in sensor networks. The scheme in this work estimates the size of an event by a small subset of the nodes in WSNs, and infers the presence or absence of a significant event just from the signals received from this subset. However, this paper does not take into account the uncertainty of signals from sparse samples of the nodes, which is significantly different from our works. In addition, the scheme in [10] is only suitable for large-scale physically-correlated events, and this constraint is not required in our work. The algorithms explored in [15-16] are designed for 0/1 decision predicate computation, in which no collaboration among neighboring sensors is considered. Niu *et al.*[17] proposed a maximum likelihood estimator, which uses binary readings that are communicated to a central processing unit to estimate the event position. The algorithm proposed in [18] is designed to address a new research area of collaborative signal information processing.

However, all these traditional approaches are not directly applicable to uncertain sensor data since they are proposed for certain data. Moreover, our work focuses on "fuzzy cleaning" for successive uncertain sensor data and develops a "collaborative cleaning" scheme for the type of the events with directional diffusion, which has not been studied before.

2.2 Probabilistic Data Managing

The probabilistic uncertainty model was first introduced in [19] for continuous sensor data. A flexible model of uncertainty, which is defined by a lower and upper bound and a probability density function of the values inside the bounds, is proposed. In [20], the problem of indexing one-dimensional uncertain data for answering "probabilistic threshold range query" is studied. Tao *et al.*[21] extended the indexing solution to support uncertain data in high-dimensional space. In [22], Cheng *et al.* investigated a cleaning problem for uncertain and probabilistic databases, with the goal of optimizing the expected quality improvement under a limited budget. The PWS-quality metric introduced in the paper is a measure that quantifies the ambiguity of query answers under the possible world semantics, whereas our work focuses on cleaning the data whose uncertainty is given by a continuous distribution. Khoussainova *et al.*[23] presented a system for correcting input data errors probabilistically using user-defined integrity constraints. However, the cleaning system was designed for missed and duplicated RFID (Radio Frequency Identification) data, which is unsuitable for event detection in WSNs.

Tan *et al.*[24] proposed a probabilistic disc model by extending the existing analytical results based on a classical disc model to the context of stochastic detection.

This model is used for real-time target detection. By considering the probability of detecting targets within a sensing range, the model captures the stochastic characteristics of real-world intrusion detection, such as probabilistic detecting ability and false alarms. A series of studies are based on the proposed probabilistic disc model[25-27]. The authors studied the impact of data fusion on the delay of detecting mobile targets[25] and extended the study to the general cases of signal decay and target speed[26]. In [27], Tan *et al.* adaptively calibrated the fusion parameters to increase system sensing performance in the presence of dynamics of environment and monitored phenomenon. In summary, our probabilistic model focuses on the inherent uncertain property of sensor measurement, whereas the probabilistic disc model mentioned above is derived from sensing range.

3 Cleaning Model

In this section, the cleaning model adopted in this paper is introduced. We propose a binary mapping method which is based on a novel uncertain sensor data model. According to the mapped binary data sequence, we address the formulated definitions of two types of dirty-events.

3.1 Uncertain Data Model

Usually, the uncertainty of sensor data is represented by probabilistic distributions of attributes[28-29]. Under this model, the value of a given attribute is represented as a collection of alternative values, each with an associated probability, or a range of values with an associated probability density function (*pdf*). Specifically, each sensor stores a *pdf* of the "original" attribute value, where "original" means the true reflection of surroundings. This uncertainty model of original data can be obtained by Bayesian cleaning approach in [30]. The distribution is denoted by $f_o(x)$, which is considered as a Normal distribution, i.e., $x \sim N(\mu_o, \sigma_o^2)$, where μ_o and σ_o^2 are the mean and variance of x, respectively. Generally, about 99.7% of values drawn from $N(\mu_o, \sigma_o^2)$ lie within three standard deviations (3-*sigma* rule). Therefore, we give $[\mu_o - 3\sigma_o, \mu_o + 3\sigma_o]$ as the interval of normal values for $f_o(x)$. To be more precise, we specify the interval by different values of $\omega_o (\omega_o \in \mathbb{R})$, that is

$$\begin{aligned} &F(\mu_o + \omega_o\sigma_o; \mu_o, \sigma_o^2) - F(\mu_o - \omega_o\sigma_o; \mu_o, \sigma_o^2) \\ &= \Phi(\omega_o) - \Phi(-\omega_o) = 2\Phi(\omega_o) - 1 = \mathrm{erf}\Big(\frac{\omega_o}{\sqrt{2}}\Big), \end{aligned} \tag{1}$$

where $F_o(x)$ is the cumulative distribution function (*cdf*) of $f_o(x)$ and $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} \mathrm{d}t$ is the error function.

Now we discuss the problem of checking whether a sensor measurement is a normal data or not according to the above uncertain data model. In the previous work of certain sensor data, an empirical normal interval is given for the abnormity checking. However, this approach cannot be used in our work since the uncertainty of the measurement needs to be considered. Thus, we introduce a novel checking method as follows.

As we know, the measurements of sensors are usually contaminated by additive random noises from sensing devices. We assume that the random noise n_s at each sensor follows a normal distribution $f_r(n_s)$, i.e., $n_s \sim N(\mu_r, \sigma_r^2)$. Given a sensor measurement v_k, let $f_m(z_k) = v_k - f_r(n_s)$, where z also follows a normal distribution, i.e., $z_k \sim N(v_k - \mu_r, \sigma_r^2)$. Hence, we obtain the probability of v_k being a normal data by

$$p_k = \int_{-\omega_o\sigma_o}^{+\omega_o\sigma_o} f_m(z_k)\mathrm{d}(z_k). \tag{2}$$

3.2 Binary Data Mapping

As is well known, sensor nodes have limited resources (processing capabilities, memory and power). Therefore, it makes sense to use binary data since a binary decision is an "easier" problem to solve. In addition, binary data require "lower" communication cost, which benefits the energy cost of sensor nodes. Therefore, we give the following mapping rule which maps sensor measurements into binary data.

$$b_k = \begin{cases} 0, & \text{if } p_k \geqslant p_{th}, \\ 1, & \text{if } p_k < p_{th}, \end{cases} \tag{3}$$

where p_{th} is an empirical probability threshold. For example, a sensor has a *pdf* $f_o(x) \sim (26, 6^2)$ of true temperature values. Device noise $n_s \sim N(0, 5^2)$ and $\omega_o = 1.96$. Therefore, we obtain the normal value interval $[21.2, 30.8]$. If successive measurements $v_1 = 24.5$, $v_2 = 23.6$, $v_3 = 33.9$, then we obtain $p_1 = 0.92$, $p_2 =$

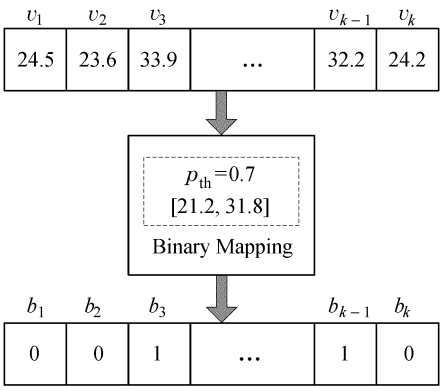


Fig.1. Example of binary data mapping.

0.85, $p_3 = 0.09$, which are mapped into $b_1 = 0$, $b_2 = 0$, $b_3 = 1$, according to $p_{th} = 0.7$ as shown in Fig.1. We focus on a single attribute for simplicity in this paper, and the extension to multiple attributes can be achieved easily by associated *pdf*.

3.3 Dirty-Event Definitions

Intuitively, when a remarkable change in a sensor data is detected, i.e., measurements of the sensor are mapped into abnormal binary data, something must have happened to the sensor. The cases in which sensor measurements are especially prone to be abnormal are summarized as follows.

Case 1 (Device Noise). As mentioned above, the data are collected from the real world by imperfect sensing devices with random noises.

Case 2 (External Noise). Sensors are sensitive to the noise from external source, due to which the changes of sensor measurements appear randomly. For instance, the electromagnetic signal from a mobile phone can cause the abnormal measurements of nearby sensors.

Case 3 (Hardware Failure). The performance of sensors tends to deteriorate when the sensors are suffering a hardware failure, such as exhausted battery power.

Case 4 (Monitored Event Occurrence). A monitored event is defined as a particular phenomenon that changes the real-world state, e.g., forest fire, chemical spill, air pollution, etc. In the usage for monitoring these events, sensors will generate abnormal data when a monitored event happens in a geographic region where the sensors are deployed.

In particular, abnormal measurements appear differently in the above cases. Device noise has minor effects on the sensor measurements, which is removed in Subsection 3.1. The errors brought by external interference (Case 2) are considered as random errors[30]. The external interference usually comes from transitory environment change. In another word, the variation of environmental conditions in Case 2 lasts for a short time, thus the resultant abnormal data are generated with low frequency. In contrast to Case 2, a sensor generates successive abnormal data which last for a relatively long time when it is in Cases 3 and 4. In particular, it can be considered that a monitored event has occurred if numerous abnormal data are caused. According to the definition of Case 4, environment variations can be considered as monitored events if they caused abnormal data uninterruptedly, i.e., monitored events are represented by abnormal readings which last for a longer time. For example, in fire alarming applications, the rising of the temperature and the smoke density in the monitored area are the monitored events. Furthermore, if most neighboring sensors generate abnormal readings simultaneously, a monitored event occurs. On the other hand, the abnormal data are caused by hardware failures if the changes appear in geographically independent sensors. In a word, abnormal measurements due to faulty devices are likely to be uncorrelated, while the sensors which have detected monitored events are spatially correlated. In this paper, we consider the abnormal data caused by Cases 1, 2 and 3 as "erroneous data". Therefore, these erroneous data should be cleaned in order to eliminate their negative influence on the reliability of event detection. We use "dirty-event" to refer to an erroneous data as well as the case it comes from, that is, the occurrence of an erroneous measurement is considered as an atomic dirty-event.

The dirty-events caused by Cases 1 and 2 break the temporal correlation of the readings. Specifically, the sensor nodes are required to periodically perform observation as they are monitoring the events. The nature of the physical phenomenon caused by a monitored event constitutes of the temporal correlation between each consecutive reading of a sensor node. However, the degree of the correlation between consecutive sensor measurements may vary irregularly and disorderly due to Cases 1 and 2. Therefore, the dirty-events caused by these two cases are considered as a type of dirty-event, namely TDE, which can be cleaned by temporal information of individual sensors. On the other hand, the dirty-events caused by Case 3 cannot affect the temporal correlation significantly since the abnormal data appear consecutively and incessantly. But this kind of dirty-events breaks the spatial correlations of the measurements of neighbor sensor nodes, based on the fact that the sensor failures (Case 3) are likely to be stochastically independent, while the monitored event measurements (Case 4) are likely to be spatial correlated due to the dense deployment. Therefore, the dirty-events caused by Case 3 are considered as a type of dirty-event, namely PDE, which can be cleaned by the spatial information of neighbor sensor nodes.

We assume that sensor s_i processes streaming data within a sliding window w, and the size of the window is denoted by l_w. Such an assumption is consistent with existing wireless sensor systems. Let $(\langle b_p, t_p\rangle, \ \langle b_{p+1}, t_{p+1}\rangle, \cdots, \langle b_{q-1}, t_{q-1}\rangle, \langle b_q, t_q\rangle)$ denote the mapped binary data stream, where t_j is the timestamp of b_j. Assume the sliding window is tuple-based, such that $l_w = q - p$. According to the different "behavior" of erroneous data mentioned above, dirty-events are classified into two types as follows.

Definition 1. *For all the dirty-events in m-sliding windows of sensor s_i, if $0 < \sum_{k=1}^{m} N_{A\_b}(w_k)/\sum_{k=1}^{m} l_k < \Theta$, the dirty-event here is considered as a transient dirty-event (TDE), otherwise, the dirty-event*

is considered as a permanent dirty-event (PDE), where $N_{A\_b}(w_k)$ denotes the total number of the abnormal binary data in k-th sliding window, Θ is a user-specific threshold.

The intuition of setting threshold Θ is that, when a PDE or a monitored event occurs, there will be changes in the readings of the sensors that are affected by the event during the event period. According to the definition, the value of threshold Θ is an important factor in events cleaning. Therefore, a threshold selection scheme is given here to determine the value of Θ efficiently, that is, Θ is set to $\Theta^{\triangle}(1+\frac{t_q-t_p}{T_{pe}})$, where T_{pe} is the expected period of PDEs in a real application and $\Theta^{\triangle}$ is a user-specific threshold which can be set according to the historic data.

4 Cleaning Dirty-Events

In this section we discuss two novel dirty-events cleaning methods, which are respectively proposed for TDE and PDE.

4.1 TDE Cleaning

Generally, external interferences take place randomly and the details are unknown to the users in practical applications. Therefore, in this part, we propose a fuzzy cleaning method for TDE based on fuzzy sets theory so as to make TDE cleaning procedure more effective and scalable for various kinds of interferences. We adopt a fuzzy set to include TDEs of a sensor, namely TDE fuzzy set. Given a binary data sequence $B_w=(b_p,b_{p+1},\ldots,b_{q-1},b_q)$ in a sliding window w, the definition of TDE fuzzy set is addressed as follows.

Definition 2. *Let $O_B=\{B_{w_1},B_{w_2},\ldots,B_{w_m}\}$ be the domain of the fuzzy set, element B_{w_k} is a binary data sequence of the k-th sliding window, TDE fuzzy set is defined as*

$$T\_F: O_B \to [0,1]$$
$$B_{w_k} \to T\_F(B_{w_k}),$$

where $T\_F(B_{w_k})$ is the membership function.

Note that given two binary data sequences with the same number of abnormal data, $B_{w_k}=(0,1,0,1,0,1,1,0,1,1,0,0)$ and $B_{w_{k'}}=(0,0,0,0,0,1,1,1,1,1,1,1)$, the abnormal binary data in B_{w_k} are more likely to belong to TDE fuzzy set. Therefore, we extract three types of subsequences from B_w to obtain more effective fuzzy membership function,namely 3-subsequence extracting method.

We exemplify that the 3-subsequence extracting method is optimal for dirty-event cleaning in the following. According to the temporal correlation of the sensor measurements, the precondition of the extracting method is that we merely focus on the subsequences of which the middle element is abnormal. If we extract 2-subsequences in the binary data sequence, namely 2-subsequence extracting method, only three types of 2-subsequences can be extracted, i.e., subsequences (0, 1), (1, 0) and (1, 1). Hence, the extracted 2-subsequences of a binary data sequence may lose exact representations of the temporal consecutiveness. If we extract 4-subsequences in the binary data sequence, namely 4-subsequence extracting method, twelve types of 4-subsequences can be extracted since there are two middle elements, i.e., subsequences (0, 1, 0, 0), (1, 1, 0, 0), (1, 1, 0, 1), (0, 1, 0, 1), (0, 0, 1, 0), (1, 0, 1, 0), (0, 0, 1, 1), (1, 0, 1, 1), (0, 1, 1, 0), (1, 1, 1, 0), (0, 1, 1, 1), (1, 1, 1, 1). Obviously, these 4-subsequences cause more complex computations, which will lead to much more time-consuming of dirty-event cleaning. To balance the cleaning accuracy and computation time, we employ 3-subsequence extracting method, of which the details are presented in Table 1. In order to reduce computational complexity, we only extract the 3-subsequence in which the mid-element is an abnormal binary data, that is, we ignore the subsequences (1, 0, 1), (1, 0, 0) and (0, 0, 1).

Table 1. Three Types of Subsequences

Denotation	Extracting Rules
s_B_1	$s\_B_1=(b_p,b_{p+1},b_{p+2})$ when $b_{p+1}=1$ and $b_p=b_{p+2}=0$, i.e., $s\_B_1=(0,1,0)$
s_B_2	$s\_B_2=(b_p,b_{p+1},b_{p+2})$ when $b_p=b_{p+1}=1$ and $b_{p+2}=0$, or $b_{p+1}=b_{p+2}=1$ and $b_p=0$, i.e., $s\_B_2=(1,1,0)$ or $(0,1,1)$
s_B_3	$s\_B_3=(b_p,b_{p+1},b_{p+2})$ when $b_p=b_{p+1}=b_{p+2}=1$, i.e., $s\_B_3=(1,1,1)$

Moreover, we adopt a one-by-one extraction method in order that the subsequences can indicate the succession of abnormal binary data maximally. The extraction procedure is illustrated in Fig.2.

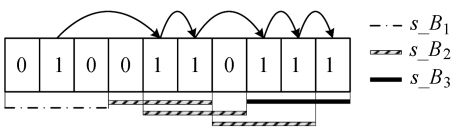


Fig.2. One-by-one extraction.

As mentioned above, the succession of abnormal readings caused by external interference is much worse than those caused by Cases 3 and 4. Therefore, the abnormal binary data of s_B_1 should be cleaned directly, by the way of being converted to normal binary value "0". According to Definition 1 (consider $m=1$ for simplicity), two fuzzy membership functions of subsequences s_B_2 and s_B_3 are given by Lemma 1 and

Lemma 2, respectively.

Lemma 1. *Let $N_{s\_B_2}(B_w)$ denote the number of s_B_2 on B_w, where $\lambda_{s\_B_2} \in [2, \Theta l_w]$, such that*

$$T\_F_{s\_B_2}(B_w) = \begin{cases} 1, & \text{if } 1 \leqslant N_{s\_B_2}(B_w) < \lambda_{s\_B_2}, \\ \dfrac{l_w - 2 - N_{s\_B_2}(B_w)}{l_w - 2 - \gamma_{s\_B_2}}, & \\ \text{if } \gamma_{s\_B_2} \leqslant N_{s\_B_2}(B_w) \leqslant l_w - 2, & \end{cases} \tag{4}$$

where $T\_F_{s\_B_2}(B_w)$ is the TDE trapezoidal membership function of s_B_2.

Proof. In the case that all the extracted s_B_2 from B_w are connected with each other, such as (0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0). Ideally, we have $N_{A\_b}(w) = N_{s\_B_2}(B_w)$. On the contrary, if the extracted subsequences are definitely disconnected, such as (0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0), we have $N_{A\_b}(w) = l_w - 2$ ($N_{s\_B_2}(B_w) = 2$). Therefore, $N_{A\_b}(w)$ satisfies $N_{s\_B_2}(B_w) \leqslant N_{A\_b}(w) \leqslant l_w - 2$. By Definition 1, critical value Θl and $\lambda_{s\_B_2}$ are substituted to the inequation, we thus have $2 \leqslant \lambda_{s\_B_2} \leqslant \Theta l_w$. □

Lemma 2. *Let $N_{s\_B_3}(B_w)$ denote the number of s_B_3 on B_w, where $\lambda_{s\_B_3} \in [\frac{\Theta l_w}{3}, \Theta l_w - 2]$, such that*

$$T\_F_{s\_B_3}(B_w) = \begin{cases} 1, & \text{if } 1 \leqslant N_{s\_B_3}(B_w) < \lambda_{s\_B_3}, \\ \dfrac{l_w - 2 - N_{s\_B_3}(B_w)}{l_w - 2 - \lambda_{s\_B_3}}, & \\ \text{if } \lambda_{s\_B_3} \leqslant N_{s\_B_3}(B_w) \leqslant l_w - 2, & \end{cases} \tag{5}$$

where $T\_F_{s\_B_3}(B_w)$ is the TDE trapezoidal membership function of s_B_3.

Proof. Similarly, if all the extracted s_B_3 from B_w are connected with each other, such as (0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0). Ideally, we have $N_{A\_b}(w) = N_{s\_B_3}(B_w) - 2$. Moreover, there exists $N_{A\_b}(w) = 3N_{s\_B_3}(B_w)$ when the extracted subsequences are definitely disconnected, such as (0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0). Therefore, $N_{A\_b}(w)$ satisfies $N_{s\_B_3}(B_w) - 2 \leqslant N_{A\_b}(w) \leqslant 3N_{s\_E_3}(B_w)$. We have $\frac{\Theta l_w}{3} \leqslant \lambda_{s\_B_3} \leqslant \Theta l_w - 2$ by substituting Θl and $\lambda_{s\_B_3}$ to the inequation. □

Based on Lemma 1 and Lemma 2, the fuzzy membership functions of subsequence s_B_2 and s_B_3 are obtained, which are employed to clean TDEs in a sliding window. Fig.3 gives an example of the TDE trapezoidal membership function of s_B_2. In Fig.3, we observe that if the number of s_B_2 is less than or equal to $\lambda_{s\_B_2}$, the value of membership function $T\_F_{s\_B_2}(B_w)$ is 1, that is, $\lambda_{s\_B_2}$ corresponds the inflexion point of TDE membership function, which indicates that the abnormal data in B_w are more likely to be considered as a PDE if $\gamma_{s\_B_2} \leqslant N_{s\_B_2}(B_w) \leqslant l_w - 2$. Therefore, $\lambda_{s\_B_2}$ is a key point that belongs to interval $[2, \Theta l_w]$ according to

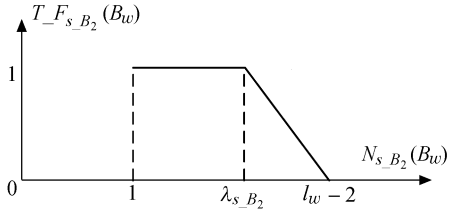


Fig.3. Example of TDE membership function.

Lemma 1. Lemma 2 has similar meaning as Lemma 1, which is omitted here due to the space limitation.

In the following, we address our fuzzy TDE cleaning procedure. Firstly, we clean the sudden abnormal data by extracting subsequence s_B_1 and converting the mid-element to normal binary data. Secondly, we calculate the overall membership grade of s_B_2 and s_B_3, which is obtained by $P_{TDE} = h_{s\_B_2} \cdot T\_F_{s\_B_2}(B_w) + h_{s\_B_3} \cdot T\_F_{s\_B_3}(B_w)$, where $h_{s\_B_2}$ and $h_{s\_B_3}$ are the contributive weights of s_B_2 and s_B_3 in B_w, respectively. Thus $h_{s\_B_2} = \frac{N_{s\_B_2}(B_w)}{N_{s\_B_2}(B_w)+N_{s\_B_3}(B_w)}$ and $h_{s\_B_3} = \frac{N_{s\_B_3}(B_w)}{N_{s\_B_2}(B_w)+N_{s\_B_3}(B_w)}$. We use an overall TDE cleaning threshold Θ^* (Θ^* is considered as an application-based threshold) to check the existence of TDEs. The cleaning procedure is illustrated in Fig.4. For each sensor s_i with $P_{TDE} \geqslant \Theta^*$, it sends message $Mes\_P_{[i]}$ to cluster head s_c. $Mes\_P_{[i]}$ consists of a node id and its P_{TDE} value. Then, based on Mes_P from all the nearby sensors, s_c decides whether or not to clean the abnormal data in B_w, i.e., s_c needs to find out the abnormal data are caused by which case. If sensor s_i receives informing message $Mes\_F_{[i]}$ from s_c, it needs to forward binary sequence B_w to be PDE-cleaned, that

```
Input: binary data sequence B_w
Output: TDE-cleaned B_w
1   foreach abnormal element b_i in B_w do
2       if b_{i-1} = b_{i+1} = 0 then
3           convert to b_i = 0
4       end
5   end
6   calculate the overall membership grade P_TDE
7   if P_TDE < Θ* then
8       foreach abnormal element b_i in B_w do
9           convert to b_i = 0
10      end
11  else
12      sensor s_i sends message Mes_P_[i]
13      if s_i receives message Mes_F_[i] then
14          forward binary sequence B_w to be PDE-cleaned
15      end
16  end
```

Fig.4. TDE cleaning algorithm.

is, the abnormal data in s_i are caused by Case 3 or 4.

TDE cleaning method does not need numerous data exchange among nearby sensor nodes. Therefore, less neighboring information exchange is required during the TDE cleaning and the energy consumption is much lower. However, if abnormal data appear frequently, i.e., PDEs occur, there is not redundant of temporal correlated information that can be used. As a result, PDEs cannot be efficiently cleaned by TDE cleaning method.

4.2 PDE Cleaning

In this subsection, we propose a collaborative dirty-event cleaning scheme for PDEs based on spatial correlations among neighboring sensor data. First, we make the following assumptions.

Assumption 1. *The nodes are uniformly deployed in a field of detecting area and they are static. Their positions are known by localization devices, e.g., a small fraction of the sensor nodes use GPS, while the rest of them estimate their locations using localization algorithms.*

Assumption 2. *The source of a monitored event causes a continuous affect that diffuses toward some direction and there are no environmental changes outside the affected region.*

Assumptions 1 is quite common and reasonable for WSNs. Assumption 2 defines an event diffusion model that is appropriate for the situation where some substance is released in the environment. For example, wind pushes the substance toward some direction. However, Assumption 2 may not be appropriate for the source that emits a continuous signal, which diffuses uniformly in all directions, such as sound or electromagnetic waves.

To perform PDE cleaning effectively, we adopt a ring-structured clustering method as shown in Fig.5,

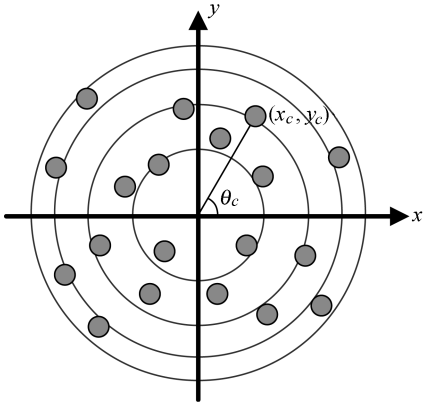


Fig.5. Ring-structured clusters.

which is suitable for our monitored events. Specially, the nodes are clustered into several ring-structured clusters, which support energy-efficient cleaning by working in turns. Since the clustering method is not the focus of this paper, we omit the details of the clustering procedure, as well as the cluster head selection mechanism. Now we discuss the collaborative cleaning procedure for PDE.

Coordinate Conversion Phase. According to the above ring-structured clusters, we convert the positions of each sensor with Cartesian coordinate (x_c, y_c) to polar coordinate (θ_c, r_c), clearly, such conversion is

$$r_c = \sqrt{x_c^2 + y_c^2}, \tag{6}$$

$$\theta_c = \begin{cases} 0, & \text{if } x_c = 0 \text{ and } y_c = 0, \\ \arcsin\left(\dfrac{y_c}{r_c}\right), & \text{if } x_c \geqslant 0, \\ -\arcsin\left(\dfrac{y_c}{r_c}\right) + \pi, & \text{if } x_c < 0. \end{cases} \tag{7}$$

Since the ring-structured clustering method clusters the sensors with similar radial coordinate, we assume that sensors $s_1, s_2, \ldots, s_k$ within one cluster have the same radial coordinate, i.e., $r_1 = \cdots = r_k = r^*$, where r^* denotes the radius of the corresponding outer ring of the cluster.

Collaborative Cleaning Phase. We consider the sensor with abnormal readings as alarmed sensor. Each alarmed sensor s_i sends an alarming report message $Mes\_A_{[i]}$ to the cluster head s_c, which consists of a node *id*, a polar coordinate and the membership grade P_{TDE_i}. Once having received $Mes\_A_{[i]}$ from all the alarmed sensors, s_c computes their correlations on the abnormal binary sequence and geographic position. Hence, the following correlation coefficient is adopted in this paper.

$$J_c = \frac{1}{N_{al} - 1} \sum_{i=1}^{N_{al}} \chi_i (\theta_i - \overline{\theta})^2, \tag{8}$$

where N_{al} is the number of all the alarmed sensors that have sent $Mes\_A_{[i]}$, and χ_i is similarity weight of the binary data sequence, which is obtained by P_{TDE_i} as follows.

$$\chi_i = \frac{P_{TDE_i}}{\sum_{i=1}^{N_{al}} P_{TDE_i}}. \tag{9}$$

Upon calculating J_c, s_c cleans the PDEs of the alarmed sensors by following rules: if correlation coefficient J_c exceeds a correlation threshold J_η, abnormal readings of the alarmed sensors indicate the occurrence of a monitored event. The abnormal information Mes_E is sent to sink node by s_c. Else, the abnormal readings are caused by hardware faults since

the alarmed sensors are spatially uncorrelated. Sequentially, s_c sends a cleaning message $Mes\_C_{[i]}$ to each alarmed sensor s_i, which informs the sensors to clean the PDEs. The above cleaning procedure is illustrated in Fig.6.

Input: B_w of alarmed sensors in working cluster
Output: PDE-cleaned B_w
foreach alarmed sensor s_i **do**
  s_i sends message $Mes\_A_{[i]}$ to s_c
  s_c calculates J_c
  if $J_c < J_\eta$ **then**
    s_c sends message $Mes\_C_{[i]}$ to s_i
    foreach abnormal element b_i in B_w **do**
      convert $b_i = 1$ to $b_i = 0$
    end
  else
    send Mes_E to the sink node
  end
end

Fig.6. PDE cleaning algorithm.

PDE cleaning method needs frequent exchanges of data among neighboring nodes, which will cause high energy consumption. However, PDE cleaning method leads to the efficient cleaning of PDEs, which TDE cleaning method cannot complete. In the traditional methods, e.g., majority voting scheme, abnormal data cannot be considered as dirty-events if they are sensed by most of the nodes in a cluster. However, these methods are based on the assumption that the monitored events diffuse all around. Therefore, traditional methods are less efficient when the monitored event diffuses within a fan shaped zone with small angel, e.g., the event shown in Fig.7. In particular, majority voting scheme is prone to take the abnormal data caused by monitored events as the results of hardware failure. On the other hand, our PDE cleaning method is superior to the traditional methods when the monitored events have above behaviors since our method adopts polar coordinate instead of Cartesian coordinate, which is verified in the experiments in Section 5.

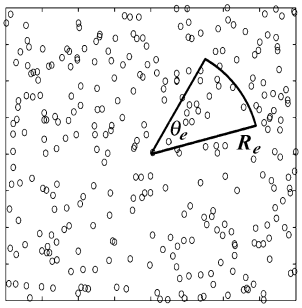


Fig.7. Example of the simulated events.

5 Experimental Evaluation

In this section, we present a set of simulation results to analyze the performance of our dirty-event cleaning method. We first describe our experimental settings, and then discuss the experimental results and evaluate the effectiveness of our approach.

5.1 Experimental Setup

Our experiments are conducted on a 1.86 GHz Intel Core 2 6300 CPU and 2 GB RAM. In the simulation, 400 sensors are randomly deployed with a uniform distribution in a square detection area of size 800×800. Real-world sensor data observing the environment (NEU lab[31]) temperature are collected in our lab by distributed MTS101 sensor boards. However, due to the constraint on resource and environment, the original data are collected with coarse granularity on time scale and incomplete samples on the space scale. Therefore, we generate a more detailed synthetic dataset for the experiments based on the raw data shown in Fig.8.

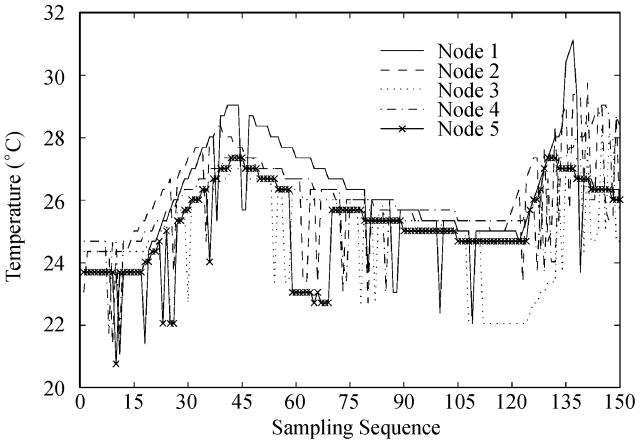


Fig.8. Sampling sequences of temperature.

According to Assumption 2, the behaviors of the monitored events are modeled as a diffusion toward some direction. The diffusing range is a fan shaped zone with angel θ_e and radius R_e as shown in Fig.7.

Moreover, the event source point is selected randomly for each generated event. The uncertain data model of each sensors are obtained from the above processed datasets. To test the cleaning efficiency of our method, we use the following parameters: device noises $n_s \sim N(0, 1^2)$, sampling period $T_s = 5\,\text{s}$ and the size of sliding window $l_w = 30T_s$. In addition, external interference and monitored events are generated at a period of $T_{int} = \delta_{int}T_s$ and $T_e = \delta_e T_s$, respectively. The monitoring duration is fixed at $5000T_s$.

5.2 Performance Study

We use two metrics E-recall and E-precision to evaluate the performance of our cleaning scheme. E-recall is defined as the fraction of monitored events that are reported correctly. E-precision denotes the fraction of reported events that are actually monitored events. In the simulation, we did a comparative study on three possible cleaning approaches: 1) DEC: our proposed dirty-event cleaning method; 2) T_DEC: DEC method which is modified by abandoning the fuzzy theorem-based strategy; 3) P_DEC: DEC method which adopts the well-known majority voting scheme in PDE cleaning phase.

First, we test the performance of DEC against the other two methods in terms of E-recall and E-precision. In order to make the comparison of the two metrics more convincing, we employ another dataset from the Intel Lab Data[①], which is collected from 54 sensors. The Intel Lab data cannot be directly used for dirty-event cleaning, hence, we preprocess the dataset (namely Intel Lab data) in the same way as the above dataset (namely NEU data). Also, the parameters of monitoring events are set differently for the purpose of effective comparison. Fig.9 shows the E-recall results for various threshold Θ ($\Theta = 0.1, 0.3, 0.5, 0.7, 0.9$), while Fig.10 shows the E-precision results. It can be seen in Fig.9 that with the increase of Θ, E-recall of the three methods drops rapidly. This is because the abnormal data caused by monitored events are prone to be cleaned as TDEs when Θ increases.

Note that even a few of abnormal data in a sliding window are prone to be considered as the results of PDE or monitored events when Θ decreases, so that smaller E-precision of the three methods are achieved as shown in Fig.10. From the above sets of experiments, we observe that $\Theta = 52\% \sim 54\%$ leads to an optimal balance of E-recall and E-precision. Therefore, the default value of Θ is set to 53% in the following experiments.

Moreover, in all cases, DEC approach achieves best performance, specially for E-recall. As expected, DEC outperforms T_DEC since the fuzzy scheme gives more effective TDE estimation on abnormal data by extracting different kinds of subsequences. DEC also outperforms P_DEC since the majority voting scheme is prone to take the abnormal data caused by monitored events as the results of hardware failure ($\theta_e = 30°$). In addition, we give a comparative study for DEC and P_DEC by changing the diffusion angle θ_e of the monitored events. As shown in Fig.11, our scheme significantly outperforms the majority voting scheme until $\theta_e \approx 190°$.

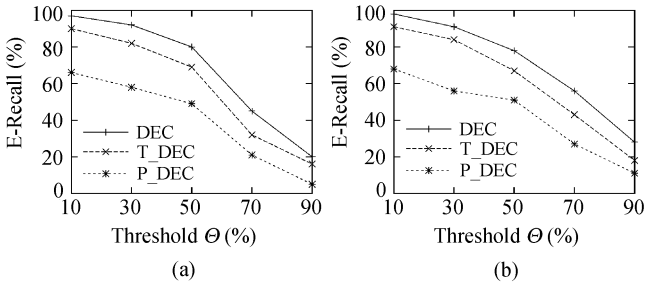


Fig.9. E-recall comparison. (a) NEU data. (b) Intel Lab data.

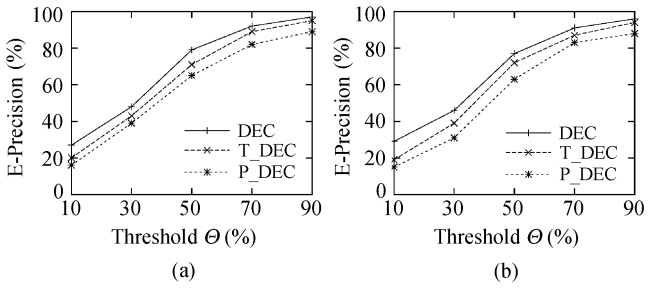


Fig.10. E-precision comparison. (a) NEU data. (b) Intel Lab data.

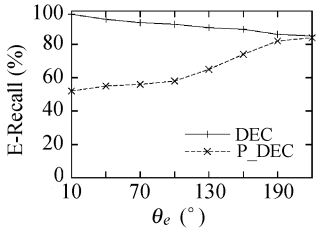


Fig.11. E-recall comparison.

Our algorithm with different time duration of monitored events is simulated to show the values of E-recall in Fig.12, where the E-recall of the conventional spatial correlation majority voting scheme (namely SFT) is included as well for comparison. From Fig.12, we observe that the E-recall of both methods decrease as the time duration of event increases since dirty-events caused by Cases 1 and 2 may be taken for abnormal data caused by Case 4. Furthermore, it is observed that SFT leads

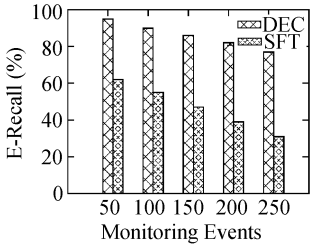


Fig.12. E-recall comparison.

[①] http://db.csail.mit.edu/labdata/labdata.html

to a worse E-recall when the time duration of event increases. This is because DEC considers the uncertainty of sensor data as it performs cleaning, while SFT only processes the detection over certain data. In addition, our algorithm cleans both TDEs and PDEs using spatial and temporal correlation, while SFT only considers the spatial correlation.

In order to investigate the applicability and robustness of our cleaning scheme, several cases are simulated. As illustrated in Fig.13, our scheme consistently performs well with various percentages of faulty sensors ($R_{fa} = 0.1 \sim 0.4$). Specially, even for a high hardware failure probability at 40%, DEC scheme can achieve about 85% in E-recall. While DEC also bears frequent external interference with various percentages of interfering range (R_{in}), which is shown in Fig.14. By cleaning the abnormal data which appear suddenly in subsequence s_B_1, most of the abnormal data caused by external interference are ignored during the detection. Therefore, the dirty-events caused by more frequent external interference can also be cleaned effectively, as we note that, DEC still exhibits good performance when $\delta_{int} = 5$.

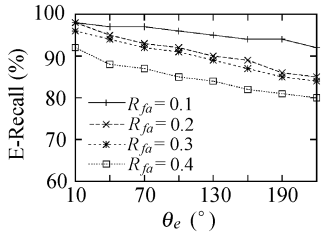


Fig.13. E-recall vs R_{fa} vs θ_e.

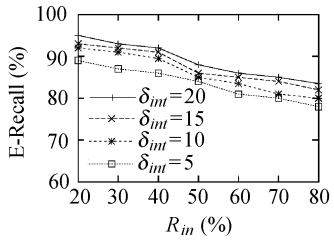


Fig.14. E-recall vs δ_{int} vs R_{in}.

The E-recall and E-precision curves for different threshold values in Fig.15 and Fig.16 show that higher E-recall can be achieved by sacrificing E-precision. Another observation obtained from Fig.15 and Fig.16 is that the values of the two metrics increase as the generation period of the monitored events becomes longer, which indicates that our cleaning method works more effectively for less frequent events.

In the following, we compared the total energy consumption of TDE cleaning method, PDE cleaning method and TPDE cleaning method. TPDE cleaning

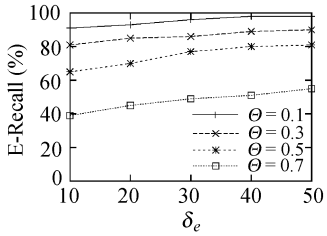


Fig.15. E-recall vs Θ vs δ_e.

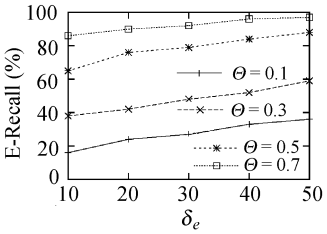


Fig.16. E-precision vs Θ vs δ_e.

method is the method that uses both temporal and spatial information to clean both types of dirty-events. From Fig.17, we observe that the consumed energy of TDE cleaning method is the least among these three methods since it needs little information exchanges. As we know, the energy consumption caused by computing is much cheaper than communicating, and information exchanging frequency is an important part of energy consumption during the dirty-event cleaning procedure. It is also observed that TPDE cleaning method consumes much more energy than TDE cleaning method and PDE cleaning method. This is because TPDE needs more information exchanges when it is utilized to clean both types of dirty-events. Thus we conclude that, the proposed separated cleaning scheme is much more energy-efficient than non-separated cleaning method.

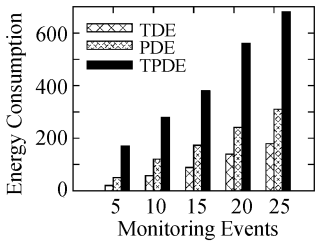


Fig.17. Energy consumption comparison.

6 Conclusions

Processing over uncertain data in WSNs has become increasingly important due to the inherent uncertainty in sensor data. Previous studies on fault-tolerant event detection are addressed in the context of certain sensor data. In this paper, we focus on the dirty-event

cleaning over uncertain sensor data, which, to the best of our knowledge, no other work has studied before. A probability-based mapping scheme is introduced, which maps the uncertain data into binary data so as to reduce the computation and communication cost. Furthermore, we propose two novel cleaning approaches, which effectively clean the two types of dirty-events (i.e., TDE and PDE) during the detection of directional monitored events. Extensive experiments demonstrate the effectiveness of our methods, under various settings. Based on the proposed uncertainty model of sensor data, exploring dirty-event cleaning methods for other types of monitored events (e.g., non-directional diffusing events) are our future focus.

References

[1] Akyildiz I F, Su W, Sankarasubramaniam Y *et al.* Wireless sensor networks: A survey. *Journal of Computer Networks*, 2002, 38(4): 393-422.

[2] Lu K J, Qian Y, Rodriguez D *et al.* Wireless sensor networks for environmental monitoring applications: A design framework. In *Proc. Global Communications Conference*, Washington, USA, Nov. 26-30, 2007, pp.1108-1112.

[3] Mainwaring A M, Culler D E, Polaste J *et al.* Wireless sensor networks for habitat monitoring. In *Proc. the 1st Int. Workshop on Wireless Sensor Networks and Applications*, Atlanta, USA, Sep. 28, 2002, pp.88-97.

[4] Wang M M, Cao J N, Li J *et al.* Middleware for wireless sensor networks: A survey. *Journal of Computer Science and Technology*, 2008, 23(3): 305-326.

[5] Garetto M, Gribaudo M, Chiasserini C F *et al.* Sensor deployment and relocation: A unified scheme. *Journal of Computer Science and Technology*, 2008, 23(3): 400-412.

[6] Krishnamachari B, Iyengar S S. Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks. *IEEE Transactions on Computers*, 2004, 53(3): 241-250.

[7] Luo X W, Dong M, Huang Y L. On distributed fault-tolerant detection in wireless sensor networks. *IEEE Transactions on Computers*, 2006, 55(1): 58-70.

[8] Ding M, Chen D, Xing K *et al.* Localized fault-tolerant event boundary detection in sensor networks. In *Proc. IEEE INFOCOM 2005*, Miami, USA, Mar. 13-17, 2005, pp.902-913.

[9] Li C R, Liang C K. A fault-tolerant event boundary detection algorithm in sensor networks. In *Proc. ICOIN 2007*, Estoril, Portugal, Jan. 23-25, 2007, pp.406-414.

[10] Gandhi S, Suri S, Welzl E. Catching elephants with mice: Sparse sampling for monitoring sensor networks. In *Proc. SenSys 2007*, Sydney, Australia, Nov. 4-9, 2007, pp.261-274.

[11] Ould-ahmed-vall E, Riley G F, Heck B S. A geometric-based approach to fault-tolerance in distributed detection using wireless sensor networks. In *Proc. IPSN 2006*, Nashville, USA, Apr. 19-21, 2006, pp.203-215.

[12] Bahrepour M, Meratnia N, Havinga P J M. Use of AI techniques for residential fire detection in wireless sensor networks. In *Proc. Workshops of the 5th IFIP Conference on Artificial Intelligence Applications and Innovations*, Thessaloniki, Greece, Apr. 23-25, 2009, pp.311-321.

[13] Wilson D, Shepherd L. Chemical and biological sensors for environmental monitoring. In *Proc. 2008 Int. Symposium on Circuits and Systems*, Seattle, USA, May 18-21, 2008, pp.1990-1993.

[14] Vu C T, Beyah R A, Li Y S. Composite event detection in wireless sensor networks. In *Proc. 2007 Int. Performance Computing and Communications Conference*, New Orleans, USA, Apr. 11-13, 2007, pp.264-271.

[15] Li D, Wong K D, Hu H Y *et al.* Detection, classification and tracking of targets. *IEEE Signal Processing Magazine*, 2002, 19(2): 17-29.

[16] Palpanas T, Papadopoulos D, Kalogeraki V *et al.* Distributed deviation detection in sensor networks. *SIGMOD Record*, 2003, 32(4): 77-82.

[17] Niu R, Varshney P. Target location estimation in wireless sensor networks using binary data. In *Proc. of 38th Ann. Conf. Information Sciences and Systems*, New Jersey, USA, Mar. 17-19, 2004.

[18] Michaelides M P, Panoyiotou C G. SNAP: Fault tolerant event location estimation in sensor networks using binary data. *IEEE Transactions on Computers*, 2009, 58(9): 1185-1197.

[19] Cheng R, Kalashnikov D, Prabhakar S. Evaluating probabilistic queries over imprecise data. In *Proc. 2003 ACM SIGMOD Int. Conf. Management of Data*, San Diego, USA, Jun. 9-12, 2003, pp.551-562.

[20] Cheng R, Xia Y, Prabhakar S *et al.* Efficient indexing methods for probabilistic threshold queries over uncertain data. In *Proc. the 30th Int. Conf. Very Large Data Bases*, Toronto, Canada, Aug. 29-Sept. 3, 2004, pp.876-887.

[21] Tao Y F, Cheng R, Xiao X K *et al.* Indexing multi-dimensional uncertain data with arbitrary probability density functions. In *Proc. the 31st Int. Conf. Very Large Data Bases*, Trondheim, Norway, Aug. 30-Sept. 2, 2005, pp.922-933.

[22] Cheng R, Chen J C, Xie X K. Cleaning uncertain data with quality guarantees. In *Proc. PVLDB 2008*, Auckland, New Zealand, Aug. 23-28, 2008, pp.722-735.

[23] Khoussainova N, Balazinska M, Suciu D. Towards correcting input data errors probabilistically using integrity constraints. In *Proc. MobiDE 2006*, Chicago, USA, Jun. 25, 2006, pp.43-50.

[24] Xing G L, Tan R, Liu B Y *et al.* Data fusion improves the coverage of wireless sensor networks. In *Proc. the 15th Int. Conf. Mobile Computing and Networking*, Beijing, China, Sep. 20-25, 2009, pp.157-168.

[25] Tan R, Xing G L, Liu B Y *et al.* Impact of data fusion on real-time detection in sensor networks. In *Proc. the 30th IEEE Real-Time Systems Symposium*, Washington, USA, Dec. 1-4, 2009, pp.323-332.

[26] Tan R, Xing G L, Xu X T *et al.* Analysis of quality of surveillance in fusion-based sensor networks. In *Proc. the 8th Int. Conf. Pervasive Computing and Communications*, Mannheim, Germany, Mar. 29-Apr. 2, 2009, pp.37-42.

[27] Tan R, Xing G L, Liu X *et al.* Adaptive calibration for fusion-based wireless sensor networks. In *Proc. the 29th Int. Conf. Computer Communication*, San Diego, USA, Mar. 15-19, 2009, pp.2124-2132.

[28] Cheng R, Prabhakar S. Managing uncertainty in sensor databases. *SIGMOD Record*, 2003, 32(4): 41-46.

[29] Cheng R, Kalashnikov D, Prabhakar S. Evaluating probabilistic queries over imprecise data. In *Proc. 2003 SIGMOD Int. Conf. Management*, San Diego, USA, Jun. 9-12, 2003, pp.551-562.

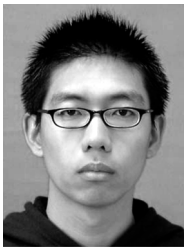
[30] Elnahrawy E, Nath B. Cleaning and querying noisy sensors. In *Proc. the 2nd ACM Int. Conf. Wireless Sensor Networks and Applications*, San Dirgo, USA, Sept. 19, 2003, pp.78-87.

[31] Chen M. Study on in-network data cleaning techniques for event detection in wireless sensor network. [Master Thesis] Northeastern University, 2008.

Mo Chen received her B.S. and M.S. degrees in computer science from Northeastern University, China, in 2005 and 2007 respectively. She is currently a Ph.D. candidate in the Department of Computer Software and Theory, Northeastern University. She is now an assistant lecturer in Software College of Northeastern University. She is a student member of the CCF and member of the ACM. Her research interests include wireless sensor network and uncertain database.

Ge Yu received his B.E. degree and M.E. degree in computer science from Northeastern University of China in 1982 and 1986, respectively, Ph.D. degree in computer science from Kyushu University of Japan in 1996. He has been a professor at Northeastern University since 1996. He is a member of the IEEE, ACM, and a senior member of the CCF. His research interests include database theory and technology, distributed and parallel systems, embedded software, and network information security.

Yu Gu received his Ph.D. degree from the College of Information Science and Engineering, Northeastern University of China in 2010. He is an associate professor of Northeastern University, China. He is a member of the CCF and ACM. His major research interests include spatiotemporal data management, RFID data management and data stream.



Zi-Xi Jia received his Ph.D. degree in pattern recognition and intelligent system from Northeastern University of China, in 2009. He is a lecturer in the Department of Computer Science and Technology of Northeastern University. His area of research is wireless sensor network.

Yan-Qiu Wang received her B.S. and M.S. degrees in computer science from Northeastern University, China, in 2007 and 2009 respectively. She is currently a Ph.D. candidate at the Department of Computer Software and Theory, Northeastern University. She is a student member of the CCF and member of the ACM. Her research interests include spatial temporal query and internet of things.