Research Article

An Energy-Efficient Outlier Detection Based on Data Clustering in WSNs

Hongyeon Kim and Jun-Ki Min

School of Computer Science and Engineering, Korea University of Technology and Education, Byeongcheon-myeon, Cheonan, Chungnam 330-708, Republic of Korea

Correspondence should be addressed to Jun-Ki Min; jkmin@koreatech.ac.kr

Received 30 November 2013; Accepted 6 March 2014; Published 10 April 2014

Academic Editor: Hwa-Young Jeong

Copyright © 2014 H. Kim and J.-K. Min. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Sensor nodes in wireless sensor networks are prone to malfunction because they are exposed to the nearby environment directly. Consequently, wrong sensor readings occurred from sensor nodes and these readings are called an outlier. Commonly, since an outlier deviates from normal sensor readings and it can bring about some problems, various techniques to detect the outliers have been proposed. In this paper, we propose an efficient outlier detection technique based on data clustering. In order to decide the width of the cluster that consists of the sensor readings, we applied the Pigeonhole Principle and then detected the outliers based on clusters. In experiments, we demonstrate the efficiency of our proposed technique compared to other outlier detection techniques.

1. Introduction

Recently, since development of the integrated circuit, the size of the sensor is gradually reduced and various sensors are built in the sensor node in a wireless sensor network (WSN). Sensor nodes detect the various and huge information (e.g., temperature, humidity, and light) around their environments and they communicate to the base station and others using radio transmission. Accordingly, WSNs are used in various applications such as environment and habitat monitoring [1, 2], combat field surveillance [3], security [4], or health care [5] applications.

Commonly, sensor nodes are severely constrained in terms of the computation power, communication bandwidth, and battery power. Among these limitations, the power is of utmost importance, since replacing the battery of sensor nodes is too either expensive or impossible [6, 7]. Thus, the energy preservation is a major research issue since it directly impacts the life time of the network. Recently, much research has shown that the radio communication is more expensive than the computation or the sensing. Thus, many techniques [8–13] have been proposed in order to reduce the communication overhead.

Particularly, since sensor nodes are placed outdoor for the applications such as the disaster monitoring [14] and habitat monitoring, the sensor nodes can be malfunctioned or the sensor readings may be incorrect due to external impact such as the severe external environments [15]. In addition, due to sudden changes in external environments, some sensor readings may deviate significantly from the normal sensor readings. These abnormal sensor readings are called the outliers. For example, assume that several sensor nodes are deployed in a mountain to monitor forest fires. When an outlier value is detected and sent to a forest guard, the forest guard can identify the actual forest fires or he/she can initialize the sensor node if the outlier is generated by the malfunctioned sensor. Thus, the outlier detection is a quite important task to detect an event or to maintain sensor networks harmoniously.

In this paper, we focus on an energy-efficient outlier detection technique in WSNs based on data clustering. To construct data clusters, we use the Pigeonhole Principle. By applying cluster width, called the permission range, obtained by the Pigeonhole Principle, we partition the data space into several clusters. However, if we partition the data space evenly, some sensor readings are identified as outliers.
although similar sensor readings of them are detected by sensor nodes. Thus, we partition the domain of data unevenly based on the location of sensor readings. Then, we identify the outliers according to the user-defined threshold $\theta$ which is related to the number of sensor readings in a cluster.

The remainder of the paper is organized as follows. Section 2 discusses related work. In Section 3, we present the background of our work. Section 4 introduces the proposed outlier detection technique in WSNs. Section 5 presents an empirical evaluation. Section 6 summarizes the paper.

2. Related Work

In WSNs, a lot of outlier detection techniques have been proposed. In [16], an outlier detection technique was proposed to collect the outliers with respect to the neighbor sensor nodes. Each sensor node calculates the median of the sensor readings received from the neighbors and its readings. Each sensor node computes the mean $\mu$ and the standard deviation $\sigma$ of the differences between its sensor readings and the calculated median. If the standardized value $(v - \mu)/\sigma$ of a sensor reading $v$ is greater than or equal to the user-defined threshold, it is regarded as an outlier.

Palpanas et al. proposed an outlier detection technique based on the the Epanechnikov kernel function [17]. Given a value $v$, each sensor node estimates the number of values around $v$ using the kernel function. If the number of values around $v$ is less than a user-defined threshold $p$, a value $v$ is regarded as an outlier. However, to make the kernel function, each sensor node transmits the required information to the base station along the routing path. Thus, it wastes a lot of energy.

In [18], an outlier detection technique based on data clustering, called DC, was proposed. Given a user-defined threshold $\epsilon$, if the distance between any two sensor readings is less than $\epsilon$, they become a cluster. When the distance between a pair of cluster centers is less than $\epsilon$, they are merged. The intercluster distance (ICD) of a cluster to $k$-nearest clusters is computed to detect the outlier clusters. When ICD of a cluster is quite different from the mean of ICDs, it is regarded as an outlier cluster. Then, the information of outlier clusters is broadcasted into WSNs. Thus, it consumes much energy.

In [19], an outlier detection technique based on an ellipsoid was proposed. Each sensor node constructs an ellipsoid boundary of sensor readings using the mean and the covariance of sensor readings and transmits its ellipsoid boundary to the base station. The base station merges all received ellipsoids, computes the global ellipsoid boundary, and broadcasts the global ellipsoid to all sensor nodes. Then, with respect to the global ellipsoid boundary, each sensor node identifies the outliers among its sensor readings.

Recently, an outlier detection technique based on the distance between sensor readings and the estimation deviation was proposed [20]. Each sensor node computes the expected deviation and the average distance between pairs of sensor readings detected within recent time interval. If the average distance is greater than the expected deviation, the sensor reading at the current time becomes an outlier. However, when the expectation model is frequently updated, the communication overhead increases.

3. Preliminary

In this section, we present the basic model of sensor networks briefly.

3.1. Sensor Networks. We consider a sensor network consisting of $n$ stationary sensor nodes $\{S_1, S_2, ..., S_n\}$ deployed in a field of interest and the powered base station serving as an access point for users to pose ad hoc queries. We use a routing tree [9] which is frequently used as a primitive to collect sensor nodes. Two nodes capable of bidirectional wireless communication directly are referred to as the neighbors for each other. Each sensor node can broadcast a message to all of its neighbors (or from a parent to its child nodes) at a time. A simple sensor network using a tree routing is shown in Figure 1. In Figure 1, $S_1$ to $S_3$ are the intermediate sensor nodes that have the child sensor nodes, and $S_4$ to $S_9$ are the leaf sensor nodes that have no child node.

Each sensor node generates its readings periodically. A sampling period is known as an epoch [21]. To agree on a global time base that allows sensor nodes to start and finish each epoch simultaneously, each sensor node executes the SMACS protocol [22] or a global time synchronization protocol [23]. Based on global time synchronized, nodes sleep for a certain period of time in each epoch to minimize energy consumption and each sensor node awakes to sample and receive results when its neighbors try to propagate a message.

4. Outlier Detection Based on Clustering

In this section, we present our proposed outlier detection technique which is based on data clustering in WSNs. In addition, we introduce an efficient data transmission scheme for our outlier detection technique.

4.1. Clustering Technique. Generally, in the outlier detection techniques based on data clustering, the width of clusters is the most important, since the number of sensor readings in each cluster is affected by the width of clusters. If the width of clusters is too large, all sensor readings may belong to a single cluster. Otherwise, each cluster may have only a single sensor reading. Thus, the outliers cannot be identified in these cases.
To solve the above problem, in this paper, we applied the Pigeonhole Principle to determine the width of clusters. We regard that pigeonholes and pigeons are the domain of sensor readings and the sensor readings, respectively. Thus, when we partition this domain into $x$ number of subdomains, where $x$ is less than the number of sensor readings, at least one subdomain contains more than two sensor readings.

Given a set of sensor readings $R = \{r_1, r_2, \ldots, r_n\}$, we can acquire the domain of $R$ as $[\text{MAX}(R), \text{MIN}(R)]$. Then, we obtain the permission range (PR) as follows:

$$PR = \left(\frac{\text{MAX}(R) - \text{MIN}(R)}{n-1}\right).$$

We use PR as the width of clusters and identify the outliers. For instance, given a set of sensor readings $R$ shown in Figure 2, PR of $R$ is obtained as $1.7 = (\text{MAX}(R) - \text{MIN}(R))/(n-1) = (19 - 2)/(11 - 1)$. If the number of sensor readings in a cluster is less than or equal to the user-defined threshold $\theta$, the cluster is the outlier cluster and the sensor readings in the outlier clusters are considered as the outliers.

In our proposed technique, if $r_i$ and $r_j$ are close to each other, a single cluster contains $r_i$ and $r_j$. For instance, as shown in Figure 3, since the difference between sensor readings $r_i$ and $r_j$ is close, a cluster $C_i$ for them is constructed. And then, a sensor reading $r_k$ is inserted into $C_i$, since the difference between $r_k$ and $r_j$ is less than PR. But $r_k$ is not inserted into $C_i$. The result of nonequipartitioning is presented in Figure 3. Each cluster consists of $C_1 = \{r_1, r_2, r_3\}$, $C_2 = \{r_4\}$, $C_3 = \{r_5, r_6\}$, and $C_4 = \{r_7, r_8, r_9, r_{10}, r_{11}\}$, respectively. Consequently, when a user-defined threshold $\theta$ is 1, a cluster $C_2$ is an outlier cluster and a sensor reading $r_4$ in $C_2$ is an outlier.

4.2. Clustering Scheme for WSNs. If each sensor node transmits its readings to the base station blindly at each epoch and the base station computes the outliers, each sensor node consumes much energy. In this section, we present an efficient data transmission scheme for our outlier detection algorithm. We assume that all sensor nodes take sensor readings periodically and keep these readings into their local storage for a time window $w$. Note that when $w$ is 1, each sensor node transmits data to the base station at each epoch.

At first, according to the Definition 1, each sensor node constructs clusters using its sensor readings detected within $w$. If the number of sensor readings in a cluster is greater than a user-defined threshold $\theta$, all sensor readings in this cluster cannot be the outliers (i.e., nonoutlier cluster (NOC)). Otherwise, all sensor readings in a cluster may be the outliers, and then we call such clusters the outlier candidate clusters (OCCs).

Along the routing path to the base station, each sensor node transmits NOCs and OCCs. For NOCs, cluster ranges (CRs) are transmitted only where CR consists of minimum and maximum values of sensor readings in a NOC. In contrast, for OCCs, the sensor readings in OCC are transmitted to the parent node.

When a parent node $p$ received CRs of NOCs and OCCs from its child nodes, $p$ attempts to merge them with its clusters. To merge the clusters, we use the following definition.

**Definition 2.** Given two cluster ranges $CR_i = [\text{min}_i, \text{max}_i]$ and $CR_j = [\text{min}_j, \text{max}_j]$, where $\text{min}_i < \text{min}_j$, if $\text{min}_j - \text{max}_i \leq PR$, we say $CR_i$ and $CR_j$ overlap within PR.

If a pair of cluster ranges overlap within PR, there are at least two sensor readings which are close and contained in different clusters. Thus, if two cluster ranges $CR_i (= [\text{min}_i, \text{max}_i])$ and $CR_j (= [\text{min}_j, \text{max}_j])$ of clusters $C_i$ and $C_j$ overlap within the permission range PR, two clusters $C_i$ and $C_j$ are merged into a new cluster whose CR is $[\text{MIN}(\text{min}_i, \text{min}_j), \text{MAX}(\text{max}_i, \text{max}_j)]$.

Note that, when two clusters are merged into a new cluster where at least one of them is a NOC, the merged cluster cannot be an OCC, since the number of sensor readings in a NOC is already greater than $\theta$. Thus, when a sensor node transmits a NOC, we do not need to transmit all sensor readings in the NOC and it only needs the cluster range CR of the NOC. Thus, each sensor node reduces the energy.
consumption when it transmits the NOCs since the volume of the transmitted data from each sensor decreases.

In contrast, when two OCCs are merged into a new cluster, we check whether the number of sensor reading in the new cluster is greater than θ or not. Since each sensor node sends all sensor readings in each OCC, we can easily count the number of sensor readings in the new cluster.

Along the routing path from each sensor node to the base station, CRs of NOCs and OCCs are merged and transmitted gradually. Finally, the base station can determine the outliers among the received OCCs. Recall that the cluster ranges (CRs) rather than all sensor readings in NOCs are transmitted along the routing paths. Thus, we can reduce the energy consumption of each sensor node.

For example, given a WSN with 11 sensor nodes in Figure 4, where the user-defined threshold θ = 1 and the domain of a set of sensor readings is [2, 19], each sensor node constructs the clusters using its sensor readings. Then, a cluster of each sensor node becomes the outlier candidate cluster (OCC) based on Definition 3.

In Figure 5, S4 merges its OCCs and the received OCCs coming from S3 and S6 based on Definition 2, and then S4 obtains a NOC1: [15, 17]. Similarly, S6 obtains the OCCs (i.e., OCC1: [3], OCC2: [8], and OCC3: [18]).

A sensor node S3 receives the clusters from its child nodes (S4, S5, and S8), and then S3 merges the received clusters. Since the cluster range of OCC1 coming from S6 overlaps the sensor readings of S3 and S5 within PR, S3 generates a new cluster NOC2. Similarly, S2 merges NOC1 and NOC2 into NOC3. But OCC2 is not merged since it does not overlap with other clusters. And another sensor node S7 generates the clusters OCC1 and OCC3.

A sensor node S1 receives the clusters from its child nodes as shown in Figure 7. S1 merges NOC3 and OCC4 into NOC4, and it merges its sensor reading and OCC3 into NOC5. Finally, S1 transmits the clusters to the base station, and then the base station detects the outlier cluster using the outlier candidate cluster OCC8.

4.3. Multidimensional Clustering Scheme. Generally, since the sensor nodes in WSNs are equipped with several sensor devices, sensor readings are multidimensional. For instance, when a sensor node obtains the weather information, it detects temperature as well as other parameters such as humidity and light. Thus, we propose the clustering scheme for a set of multidimensional data.

Let a set of d-dimensional sensor readings $R^d$ be $\{r_1^d, r_2^d, \ldots, r_n^d\}$ (|$R^d$| = n). A d-dimensional sensor reading $r_i^d$ is represented as $\langle r_i^1, \ldots, r_i^d \rangle$, where $r_i^k$ denotes the kth dimensional value of $r_i^d$. On each dimension k such that $1 \leq k \leq d$, we obtain the permission range $PR_k = \langle MAX(r_i^k), \ldots, r_i^d \rangle / (n - 1)$ derived from (1).

Based on Definition 3, for every dimension k with $1 \leq k \leq d$, $CR_i^d$ and $CR_j^d$ overlap within $PR_k$; we merge the clusters $C_i$ and $C_j$ into a new cluster.
On the first dimension, since \( \min_1^d[1] < \min_2^d[1] \) and \( \max_1^d[1] - \max_2^d[1] (= 5 - 4) \) are less than or equal to \( PR_1 (= 1) \), \( CR_1^d \) and \( CR_2^d \) overlap within \( PR_1 \). \( CR_1^d \) and \( CR_2^d \) also overlap within \( PR_2 \) on the second dimension. Thus, \( CR_1^d \) and \( CR_2^d \) are merged into a new cluster. However, \( CR_3^d \) is not merged into other clusters since \( CR_3^d \) is not overlapped with any clusters on the second dimension such as \( \min_3^d[2] - \max_3^d[2] (= 18 - 13) > PR_2 (= 1.5) \).

5. Experiments

5.1. Experimental Environments. To evaluate the performance of our proposed algorithm compared with the state-of-the-art algorithm, we used a set of real-life data which is provided by Intel Berkeley Research Lab [24]. A sensor network consists of 54 sensor nodes, and each sensor node is deployed in 40.5\( \times \)31 (m\(^2\)) area as shown in Figure 8. The base station is located at the center of the area. We set the default communication distance 7 m. The maximum depth and the maximum width of a routing tree in sensor network are 5 and 3, respectively. Sensor readings consist of temperature (Celsius), humidity (%), and illumination (Lux).

As competitors, we implemented the Brute-Force (BF) and an outlier technique based on data clustering (DC) [18]. In BF, each sensor node transmits its readings to the base station at each epoch. In DC, each sensor node in a network transmits a set of intercluster distances (ICDs) in respect of sensor readings along the routing paths to the base station. The intercluster distance is the distance of centers in any two data clusters in each sensor node. If the intercluster distance is less than the user-defined threshold \( \epsilon \), two data clusters are merged. When the base station receives ICDs from all sensor nodes, it computes the ICDs of the \( k \)-nearest clusters. And then, the base station broadcasts the means of ICDs in order to identify the outliers in each sensor node. We set \( \epsilon \) and \( k \) for DC are 0.26 and 4, respectively. We called our proposed algorithm PC.

To compute the energy consumption, we use the free space channel model [25]. Under this model, to transmit an \( l \)-bits message and a distance \( c \), a sensor expends

\[
E_T (l, c) = E_{T\text{-elec}} (l) + E_{T\text{-amp}} (l, c) = l \times E_{\text{elec}} + \xi_{\text{amp}} \times l \times c^2.
\]  

And, to receive this message, a sensor expends

\[
E_R (l) = E_{R\text{-elec}} (l) = l \times E_{\text{elec}}.
\]

In this experiment, we set 50 nJ/bit to the electronic circuit constant \( E_{\text{elec}} \) and 100 pJ/bit/m\(^2\) to the transmit amplifier constant \( \xi_{\text{amp}} \). We set the size of packet as 40 bytes. The parameters used in our experiment are summarized in Table 2.

5.2. Experimental Result. To evaluate the energy consumption of each outlier detection algorithm, we run our own simulator for 1000 epoches and plot the total energy consumption. Figure 9 shows the energy consumption varying the window size \( w \). When the size of a window is small (i.e., \( w = 2 \)), BF shows the best performance since BF transmits sensor readings, but DC and PC transmit the information of clusters.

However, as \( w \) increases, the performance of PC and DC is improved since the cluster information rather than sensor readings is transmitted. Furthermore, the performance gap between PC and DC increases with increasing \( w \). As \( w \) increases, the number of nonoutlier clusters (NOCs) increases in PC. Thus, the size of data to be transmitted decreases. In DC, the energy consumption is reduced since DC is also based on data clustering as PC. However, our proposed PC is better than DC because PC used the permission

![Figure 7: Clustering in S1.](image1)

![Figure 8: Placement of sensor nodes [24].](image2)

**Table 1:** Example of CRs on 2 dimensions.

| Cluster ranges | \( k = 1 \) | \( k = 2 \) |
|----------------|-------------|-------------|
| \( CR_1^d \)  | \( CR_1^d[k] = [3, 4] \) | \( CR_1^d[k] = [12, 13] \) |
| \( CR_2^d \)  | \( CR_2^d[k] = [5, 6] \) | \( CR_2^d[k] = [10, 11] \) |
| \( CR_3^d \)  | \( CR_3^d[k] = [7, 9] \) | \( CR_3^d[k] = [18, 19] \) |

**Table 2:** Parameters.

| Parameter          | Default value | Range |
|--------------------|---------------|-------|
| Packet size \( p \) | 40            | 40    |
| Window size \( w \) | 8             | 2, 5, 8, 11, 14 |
| Threshold \( \theta \) | 2             | 1, 2, 3, 4, 5 |
| Dimension \( d \) | 1             | 1, 2, 3 |
range to determine the cluster width. Additionally, in DC, the mean of ICDs needs to be broadcasted to obtain the outliers in each sensor node. Our proposed algorithm PC is better than the DC about 58% on the average.

Figure 10 shows the energy consumption varying \( \theta \). As shown in Figure 10, the performances of all techniques are stable in spite of varying \( \theta \). The energy consumption of DC is constant, since the intercluster information has a fixed size. But the energy consumption of PC increases slightly, since the number of sensor readings in an outlier candidate cluster increases slightly. Nonetheless, our proposed technique PC shows the best performance in terms of energy efficiency. In this experiment, our proposed technique is better than the DC about 51%.

Figure 11 shows the energy consumption varying the dimension \( d \). The energy consumption of all techniques is increased according to varying \( d \), because the packet of the sensor nodes on multidimension contains more information as compared with that on 1 dimension. The energy consumption of PC is less than other techniques BF and DC. In other words, our proposed technique is better than the DC about 48%.

6. Conclusion

In this paper, we present an efficient outlier detection technique in WSNs. To obtain the appropriate width of clusters, we applied the Pigeonhole Principle. In our proposed technique, each sensor node in WSNs constructs and merges the clusters based on the permission range PR. Then, our proposed technique uses two kinds of clusters (NOC and OCC) in order to detect the outliers and reduce the energy consumption of each sensor. In our experiments with a set of real-life data, we show that our proposed technique outperforms existing techniques significantly.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1B3003060).

References

[1] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin, “Habitat monitoring with sensor networks,” Communications of the ACM, vol. 47, no. 6, pp. 34–40, 2004.

[2] X. Zhou, Y. Ge, X. Chen, Y. Jing, and W. Sun, “A distributed cache based reliable service execution and recovery approach in manets,” Journal of Convergence, vol. 3, no. 1, pp. 5–12, 2012.
[3] W. M. Merrill, F. Newberg, K. Sohrabi, W. Kaiser, and G. Pottie, "Collaborative networking requirements for unattended ground sensor systems," in Proceedings of the IEEE Aerospace Conference, vol. 5, pp. 2153–2165, 2003.

[4] M. S. Obaidat and F. Zarai, "Novel algorithm for secured mobility and ip traceability for wlan networks," Journal of Convergence, vol. 3, no. 2, pp. 1–8, 2012.

[5] J. K. Y. Ng, "Ubiquitous healthcare: healthcare systems and applications enabled by mobile and wireless technologies," Journal of Convergence, vol. 3, no. 2, pp. 15–20, 2012.

[6] D. Gupta, A. De, and K. Chatterjee, "Performance study of genus 3 hyperelliptic curve cryptosystem: a survey of qos based routing protocols for wireless sensor networks," Journal of Information Processing Systems, vol. 8, no. 1, pp. 145–158, 2012.

[7] R. M. Itjihadie, B. C. Hidayanto, A. Affandi, Y. Chisaki, and T. Usagawa, "Dynamic content synchronization between learning management systems over limited bandwidth network," Human-Centric Computing and Information Sciences, vol. 2, no. 17, pp. 1–16, 2012.

[8] D. J. Abadi, S. Madden, and W. Lindner, "REED: robust, efficient filtering and event detection in sensor networks," in Proceedings of the 31st International Conference on Very Large Data Bases (VLDB’05), pp. 769–780, September 2005.

[9] S. Madden, M. J. Franklin, and J. M. H. W. Hong, "Tag: a tiny aggregation service for ad-hoc sensor networks," in Proceedings of the symposium on Operating Systems design and implementation, vol. 36, pp. 131–146, 2002.

[10] B. Singh and D. K. Lobiyal, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," Human-Centric Computing and Information Sciences, vol. 2, no. 13, pp. 1–18, 2012.

[11] R. Sumathi and M. Srinivas, "A survey of qos based routing protocols for wireless sensor networks," Journal of Information Processing Systems, vol. 8, no. 4, pp. 589–602, 2012.

[12] N. Trigoni, A. Guitton, and A. Skordylis, "Poster abstract: Routing and processing multiple aggregate queries in sensor networks," in Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys ’06), pp. 391–392, November 2006.

[13] X. Yang, H. B. Lim, T. M. Zsu, and K. L. Tan, "In-network execution of monitoring queries in sensor networks," in Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 521–532, June 2007.

[14] Q. L. Liu and D. H. Oh, "Performance evaluation of multihop communication based on a mobile multi-robot system in a subterranean laneway," Journal of Information Processing Systems, vol. 8, no. 3, pp. 471–482, 2012.

[15] S. K. Dhurandher, M. S. Obaidat, and M. Gupta, "A novel energy-aware cluster head selection based on particle swarm optimization for wireless sensor networks," Human-Centric Computing and Information Sciences, vol. 2, no. 3, pp. 1–14, 2012.

[16] W. Wu, X. Cheng, M. Ding, K. Xing, F. Liu, and P. Deng, "Localized outlying and boundary data detection in sensor networks," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 8, pp. 1145–1156, 2007.

[17] T. Palpanas, D. Papadopoulos, V. Kalogeraki, and D. Gunopulos, "Distributed deviation detection in sensor networks," SIGMOD Record, vol. 32, no. 4, pp. 77–82, 2003.

[18] S. Rajasegarar, C. Leckie, M. Palaniswami, and J. C. Bezdek, "Distributed anomaly detection in wireless sensor networks," in Proceedings of the 10th IEEE Singapore International Conference on Communications Systems (ICCS ’06), November 2006.
