Anomaly IoT Node Detection Based on Local Outlier Factor and Time Series

Fang Wang[1, *], Zhe Wei[1] and Xu Zuo[2]

Abstract: The heterogeneous nodes in the Internet of Things (IoT) are relatively weak in the computing power and storage capacity. Therefore, traditional algorithms of network security are not suitable for the IoT. Once these nodes alternate between normal behavior and anomaly behavior, it is difficult to identify and isolate them by the network system in a short time, thus the data transmission accuracy and the integrity of the network function will be affected negatively. Based on the characteristics of IoT, a lightweight local outlier factor detection method is used for node detection. In order to further determine whether the nodes are an anomaly or not, the varying behavior of those nodes in terms of time is considered in this research, and a time series method is used to make the system respond to the randomness and selectiveness of anomaly behavior nodes effectively in a short period of time. Simulation results show that the proposed method can improve the accuracy of the data transmitted by the network and achieve better performance.

Keywords: Local outlier factor, time series, Internet of Things, anomaly node detection.

1 Introduction

The IoT is usually composed of many heterogeneous nodes and these nodes are capable of perception, calculation, communication and control [Chelloug and Ei-Zawawy (2018); Kim, Min and Kim (2019); Badshah, Ghani and Qureshi (2019); Ren, Liu, Ji et al. (2018); Wang, Gao, Liu et al. (2019); Wang, Gao, Yin et al. (2018); Yin, Zhou, Zhang et al. (2017)]. Through these nodes, a complex, dynamic and distributed IoT can be formed. The intelligent network based on the IoT has also been continuously researched and applied, and many practical application fields have emerged such as intelligent medical, intelligent city, and intelligent agriculture [Wang, Kong, Li et al. (2019); Su, Sheng, Leung et al. (2019); He, Guo, Liang et al. (2019); Su, Sheng, Xie et al. (2019); Sun, Ma and Wang (2018)].

However, most of the nodes in the IoT are resource constrained devices, which are relatively weak in aspects of computing power, storage capacity and communication capacity. Therefore, in the face of network security problems, it is not appropriate to

[1] School of Computer Science, Civil Aviation Flight University of China, Sichuan, 618307, China.

[2] Anzina PTY Ltd., Sydney, NSW 2118, Australia.

[*] Corresponding Authors: Fang Wang. Email: cafuc_wf@foxmail.com.

Received: 18 January 2020; Accepted: 16 April 2020.

CMC. doi:10.32604/cmc.2020.09774 www.techscience.com/journal/cmc

apply complex encryption, decryption, digital signature and other traditional algorithms on these nodes. In addition, due to the heterogeneity and diversity of nodes in the IoT, some nodes will show anomaly behavior in a period of time. For example, in order to protect themselves, some nodes will alternate between normal behavior and anomaly behavior so that they can get away with some traditional anomaly node detection algorithms and achieve the purpose of destroying the network. At the same time, it is difficult for the network system to find anomaly nodes in a short period of time. Thus, in the IoT, the traditional network security methods are not suitable enough to deal with these bad or anomaly nodes, which has become the main obstacle to the development of the IoT [Farooq, Waseem and Khairi (2015)].

Motivated by the anomaly behavior nodes in the IoT, this study proposes an anomaly IoT node detection based on local outlier factor and time series (LOFT). The main contributions of this study are: 1) using local outlier detection algorithm to detect nodes in the IoT, which only needs one parameter, has relatively low complexity, and is suitable for the IoT nodes with limited computing ability; 2) in order to further verify whether the node is an anomaly one, unlike many traditional methods, the variation of node behavior in terms of time is considered. By using the time series method, the network system can effectively respond to the randomness and selectivity of anomaly node behavior so as to distinguish and deal with such nodes in a short time, reduce their adverse effects on the network, and improve the integrity of the network.

The organization of this study is as follows: in Section 2, some representative node detection methods in the IoT are analyzed and summarized; in Section 3, the basic principle and main characteristics of the local outlier detection algorithm used in this study are introduced; in Section 4, the time series analysis is studied and the relevant algorithm used is presented; simulation tests and conclusions are shown respectively in Sections 5 and 6.

2 Related works

For the IoT with limited resources, long-term application of intrusion detection algorithm will lead to excessive energy consumption of network nodes. To address this issue, Sedjelmac et al. [Sedjelmaci, Senouci and Al-Bahri (2016)] proposed a game theory-based anomaly detection method for resource constrained IoT. In this method, only when a new attack is about to occur, can the anomaly detection mechanism be activated to balance the accuracy of detection and the energy consumption of network system. Kumar et al. [Kumar and Kulothungan (2017)] proposed a method to detect and identify node anomaly behavior for DoS attack in the context of IoT. This method checks whether the communication behavior of network nodes is anomaly by monitoring the protocol conversion sequence of MAC layer for a period of time. When anomaly nodes are found, a topology management method is used to prevent anomaly nodes from launching network attacks.

Shafi et al. [Shafi, Basit and Qaisar (2018)] proposed an anomaly detection and prevention system IDPs based on the fog computing framework. IDPs is deployed on the edge of the network and by adding a management layer to the fog to deal with the change of network scale as well as allocate necessary computing resources for anomaly detection.

In Lyu et al. [Lyu and Jin (2017)], an anomaly recognition method for IoT based on hyper ellipsoid clustering and fog empowered is proposed. In this method, data processing and clustering tasks are completed by the fog layer and cloud layer nodes, and the anomaly detection is also performed by these two layers. Not only the detection time is short, but also the energy consumption of the terminal node can be reduced.

Amouri et al. [Amouri, Alaparthy and Morgera (2018)] proposed an intrusion detection mechanism based on cross layer and network behavior. This mechanism uses a hybrid learning method to divide intrusion detections into local detection and global detection. In the local detection stage, it uses supervised learning and decision tree to classify data, which is used to imitate network behavior and infer node state. In the global stage, it passes the classified correct instances to super nodes, and uses iterative linear regression to generate the cumulative fluctuation measurement of malicious nodes and normal nodes to establish the records of these nodes and identify malicious ones. In Thanigaivelan et al. [Thanigaivelan, Nigussie and Virtanen (2018)], a hybrid IoTs internal anomaly detection method based on reactive node and cross layer is proposed. In this method, the node only monitors the neighbor nodes in its single hop area and all the observed information is analyzed and managed in the local node, which enables the node to judge the threat independently and responds accordingly.

Bostani et al. [Bostani and Sheikhan (2017)] proposed a real-time anomaly detection method based on MapReduce. This method consists of two parts: the anomaly intrusion detection module and the designated intrusion detection module, the designated intrusion detection agent deployed in the routing node analyzes the behavior of its host node and sends the analysis results to the root node through the normal data packets; the anomaly detection agent deployed in the root node applies the unsupervised path optimization forest algorithm to build the node cluster, and identifies the anomaly nodes by means of the distributed and MapReduce framework as well as the application of voting mechanism. Li et al. [Li, Sun and Wu (2018)], an anomaly data detection method based on parallel distributed computing in the framework of the IoT is proposed. This method is based on the rough set logic inference method and the non-uniform distributed cluster routing method to identify the anomaly nodes. It also considers the residual energy of nodes and the distance between nodes and the base station so as to reduce the energy consumption of the whole network.

3 Local outlier factor

The detection method used in this study is called local outlier factor, or LOF [Gan and Zhou (2018)]. LOF is an algorithm from data mining. The input of the algorithm is a group of data readings, and the output is the evaluation of each input data reading. From this evaluation, we can infer the degree of confidence or anomaly associated with a certain data object reading. It is called “local” because the algorithm only considers the data object in a small neighborhood and the calculation of outlier only depends on the extent to which a data object is isolated from other objects in the same neighborhood, that is, the outlier describes the deviation degree of a data object. If the outlier value of a data object is larger, it means that the data object is more likely to be anomaly. In practical applications, the corresponding generating node of the data is considered to be a malicious node. Compared with other outlier algorithms, the advantage of LOF is that the output of the algorithm is not only binary, but also can be classified into more characteristics such as good, bad, excellent and so on.

The LOF algorithm only needs one parameter *k*. On one hand, it is used to represent the minimum number of data objects that constitute the nearest neighbor. On the other hand, it can also be used to define the size of the local neighborhood of a data object. The algorithm is described as follows.

1) Let $d(p,x)$ represent the distance between object p and object $x \in D$, where D is a collection of data objects.

2) For any positive integer k, let $k-dist(p)$ represent the k distance of the object p, which is defined as the distance between the object p and an object $o \in D$ and meets the following requirements: there is at least k objects of $o' \in D \setminus \{p\}$ so $d(p,o') \le d(p,o)$; there is at most $k-1$ objects of $o' \in D \setminus \{p\}$ so $d(p,o') < d(p,o)$;

For the k distance of p, its neighborhood contains all objects with a distance of no more than $k-dist(p)$ with p, i.e.,

$$N_{k-dist(p)}(p) = \{q \in D \setminus \{p\} \mid d(p,q) \le k-dist(p)\} \tag{1}$$

where the object q is called the nearest k neighbor of p.

Eq. (1) can be understood as follows: there is a virtual circle around each data object so that each virtual circle includes at least k data objects, which can be regarded as k neighbors of p.

3) According to the distance between the object p and each point in the k neighbors the reachable distance is defined by

$$reach-dist_k(p,o) = max\{k-dist(o), d(p,o)\} \tag{2}$$

4) According to the average distance of each data object in the neighborhood, a density parameter can be obtained and is called local reachable density, which is defined by

$$Lrd_k(p) = \frac{1}{\frac{\sum_{o \in N_k(p)} reach-dist_k(p,o)}{|N_k(p)|}} \tag{3}$$

5) Through the average value of the ratio of the local reachable density of p and its nearest k neighbors' local reachable density, the local outlier factor of p is defined by

$$LOF_k(p) = \frac{\sum_{o \in N_k(p)} \frac{Lrd_k(o)}{Lrd_k(p)}}{|N_k(p)|} \tag{4}$$

After detecting the anomaly factors, the internal nodes of the IoT can be classified as the normal nodes and the anomaly nodes. The detailed process of the algorithm is recommended in Gan et al. [Gan and Zhou (2018)].

4 Time series

Time series analysis is a statistical method used for dynamic data processing. Its purpose is to study and try to find the rules followed by a random data series, and then analyzes and obtains useful feature information in the series. In practice, data sequence is usually composed of a series of time point data. For example, suppose $\mathbf{Z}$ is a random variable whose time series can be expressed as $\mathbf{Z} = \{z_1, z_2, ...z_n\}$, where z_n is the value of $\mathbf{Z}$ at time n.

Suppose that the behavior of network nodes is directly related to the data they transmit or send, that is, good behavior nodes generate normal data and bad or anomaly behavior nodes generate anomaly data. For an anomaly behavior node, the data generated will fluctuate with time, and the changed data can be regarded as the data sequence in the time series. Therefore, the time series can be used to analyze and process the anomaly data sequence of the node and judge whether or not the node belongs to the anomaly behavior node.

In this study, time series analysis consists of three parts: node data series, benchmark test series, and a certain similarity test module. In application, the similarity test module can be treated as a function and the two data series are two input parameters. After the function is executed, the return value is the similarity result. The data sequence of a node is the data values that the node changes with time. In this study, it is expressed by $\mathbf{R}_i$

$$\mathbf{R}_i = \{r_i(t_1), ..., r_i(t_n)\} \tag{5}$$

where $r_i(t_n)$ is the data value of the node i at time t_n. The benchmark test sequence is composed of a series of comparative test data, which is expressed as $\mathbf{\Phi}$

$$\mathbf{\Phi} = \{\phi(t_1), ..., \phi(t_n)\} \tag{6}$$

And each test data should be compared with the data at its corresponding time, for example, $\phi(t_j)$ *vs.* $r_i(t_j)$.

In addition, considering the resource constraints of most nodes in the IoT, the similarity test module should adopt a lightweight and feasible computing method. In this research, the above two sequences are regarded as two vectors and the cosine value of their included angle is used to calculate the similarity. The similarity Ψ is defined by

$$\Psi = \frac{\mathbf{R}_i \cdot \mathbf{\Phi}}{\|\mathbf{R}_i\| \|\mathbf{\Phi}\|} = \frac{\sum_{j=1}^{n} r_i(t_j) \times \phi(t_j)}{\sqrt{\sum_{j=1}^{n} (r_i(t_j))^2} \times \sqrt{\sum_{j=1}^{n} (\phi(t_j))^2}} \tag{7}$$

Furthermore, in order to map the above results to $[0,1]$, the above equation is normalized, and the result is expressed as follows:

$$\Psi' = 1 - \frac{\arccos(\Psi)}{\pi} \tag{8}$$

Eq. (8) shows that the closer the result is to 1, the more similar the two vectors are. In the practical application, the value of similarity can be defined according to the actual situation. For example, in the application with high urgency, the similarity can be set to be closer to 1 or equal to 1, which completely eliminates the anomaly behavior of nodes. Through the computing of similarity, the anomaly behavior node can be defined as: if a node is an anomaly behavior node, its similarity should meet $\Psi^{'} > \Theta$, where $\Theta \in [0,1]$ is the similarity threshold.

5 Simulation tests

The data sent out by normal behavior nodes in the network generally does not appear in the anomaly range, assuming that 30% of normal nodes send out data in the anomaly range with a probability of 10%; the data sent out by anomaly behavior nodes usually presents anomaly behavior in the network, such as selective data transmission or data modification. In addition, it is also assumed that:

1) The network nodes are deployed in a certain area for temperature data sensing. For the convenience of the study, node scale is reduced to a cluster, and 100 nodes are deployed in a circular area, among which 25 anomaly nodes are evenly distributed, and the cluster head is located in the center of the circle;

2) The temperature data range from normal nodes is (25-30), and the temperature reading range from anomaly nodes randomly alternates between (25-30) and outside of (25-30);

3) Each node has its own unique ID, and the cluster head knows the ID information of all other nodes, each node can communicate with the cluster head directly;

4) The cluster head sends the temperature request to the nodes in the cluster regularly, and judges the corresponding nodes according to the received temperature readings. Once the node is identified as the anomaly node, the cluster head will no longer request data from the node, even if the data from the node is received, it will be discarded.

5.1 Number of inactive nodes

In this part, the number of inactive nodes is tested, and the comparison method is LOFT proposed in this study and the one used in Gan et al. [Gan and Zhou (2018)]. In this part, the number of inactive nodes refers to the number of nodes identified and isolated by the cluster head. Suppose that the cluster head performs 500 data queries to all nodes in the network at a fixed time interval. In this test, the length of the time series is set to be 10 queries and 20 queries respectively, the anomaly benchmark test sequence is the temperature reading outside the interval of (25-30), and the similarity thresholds are 0.8 and 0.9 respectively. The test results are shown in Figs. 1 and 2.

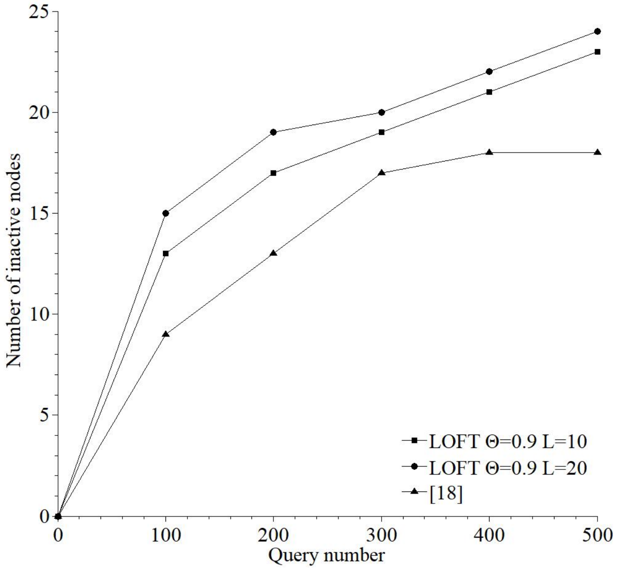

Figure 1: Number of inactive nodes with Θ=0.9

It can be seen from Figs. 1 and 2 that with the increasing number of cluster head queries, the number of inactive nodes in Gan et al. [Gan and Zhou (2018)] is also increasing, from the initial about 9 nodes to around 16 at the 500th query, but still far less than the number of anomaly nodes 25 in the system. This is because in the method of Gan et al. [Gan and Zhou (2018)], anomaly nodes can send data selectively, which makes it difficult to find and isolate such nodes in a short time. In the proposed method, anomaly nodes can be detected and isolated quickly because anomaly nodes are identified under a certain a period of time.

In addition, in Fig. 1, it can be found that with the appropriate increase of the length of time series, the number of identifying inactive nodes also increases. For example, when the cluster head queries for the 300th time, the number of inactive nodes in the proposed method is about 18 and 20 respectively. It can be seen that the growth of time series is helpful for the system to identify the anomaly nodes in a larger range. It can also be found in Figs. 1 and 2 that under the same conditions, properly reducing the similar threshold value will also help speed up the identification of anomaly nodes. This is because the reduction of similar threshold value correspondingly increases the severity of similar judgment, making the anomaly nodes easier to be identified and isolated.

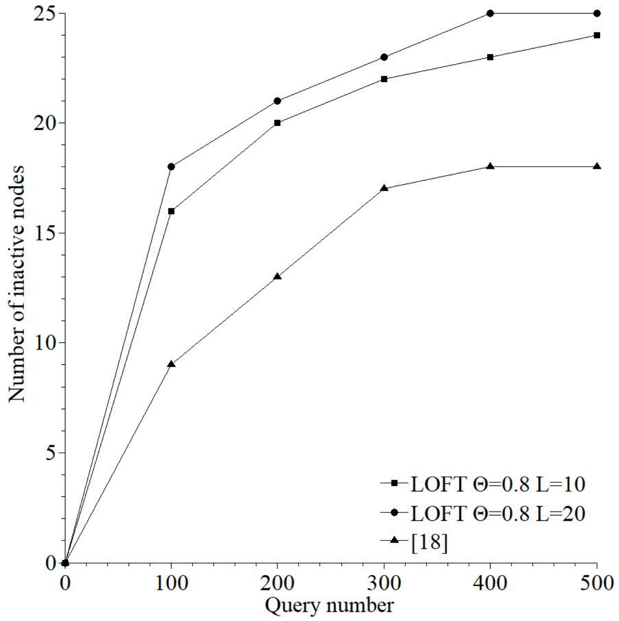

Figure 2: Number of inactive nodes with Θ=0.8

5.2 Anomaly data rate

In this part, the rate of anomaly data received by cluster head from anomaly nodes is tested. The anomaly data rate is defined as the ratio of the number of received anomaly data and the number of all data from the network that the cluster head should receive. It can be seen that the smaller the anomaly data rate is, the better the performance of the corresponding method will be. The comparison method is the method used in LOFT and that in Gan et al. [Gan and Zhou (2018)]. Test results are shown in Figs. 3 and 4.

It can be seen from Figs. 3 and 4 that with the increasing number of queries from cluster head, the anomaly data rate of both methods decreases. This is because in both methods, the number of inactive nodes is increasing, that is, more and more inactive nodes are identified and isolated by the system. However, in the proposed method, the anomaly data rate declines faster. This is also for the reason that anomaly nodes are identified under a fixed period of time and they become inactive in a shorter time, making the cluster head receive less anomaly data than the compared method. For example, in the 400th query in Fig. 3, the anomaly data rate of the proposed method is respectively about 8% and 6%, while that of the comparative method is about 11%. In addition, similar to the case in test 1, increasing the length of time series or reducing the value of similarity threshold in the proposed method will help to reduce the anomaly data rate and improve the correct data rate in the network system.

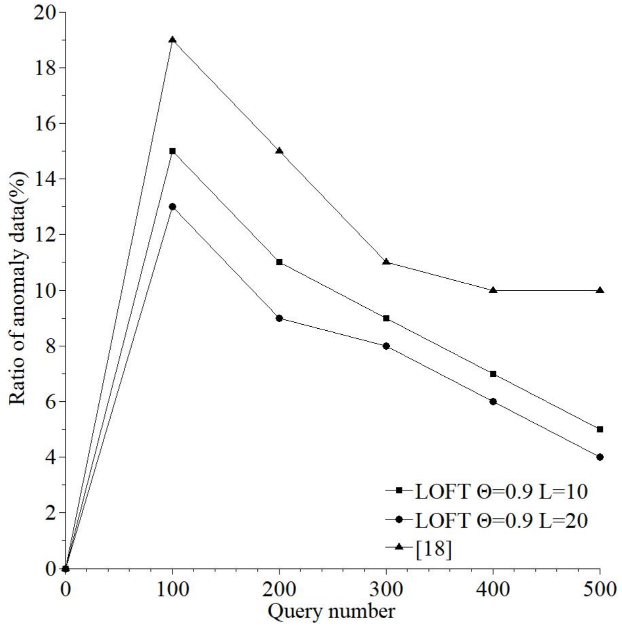

Figure 3: Ratio of anomaly data with Θ=0.9

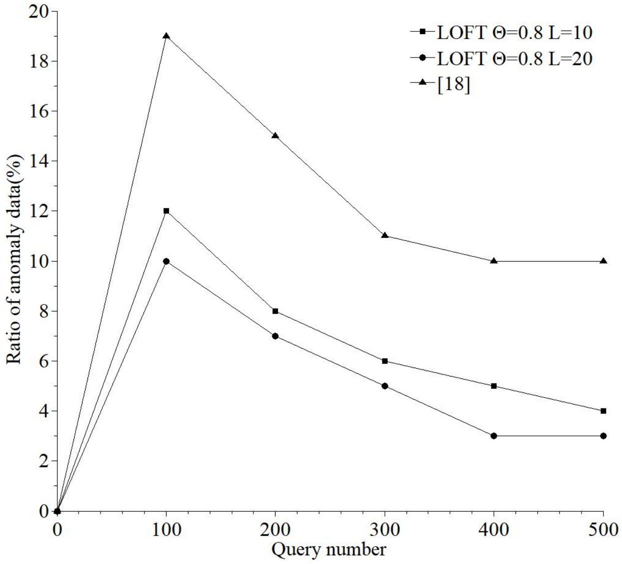

Figure 4: Ratio of anomaly data with Θ=0.8

6 Conclusion

Traditional security solutions are not suitable for the identification of anomaly nodes in the IoT. According to the characteristics of the nodes in the IoT, this study uses a lightweight local outlier detection and time series method to detect the anomaly behavior nodes in the

IoT. The main contributions of this study are: 1) use a low complexity local outlier detection algorithm to detect anomaly nodes; 2) apply a time series method to further verify whether the node is anomaly or not and make the network effectively respond to the randomness and selectivity of anomaly node behavior in a short time. The simulation results show that the proposed method has a good effect in improving the data accuracy of network transmission. In addition, the length and similarity threshold of time series cannot be selected at will. We will further analyze and study them in the future work.

Acknowledgement: The authors would like to thank the anonymous reviewers and the associate editor, whose constructive comments help improve the presentation of this work.

Funding Statement: This work is partially supported by the Ministry of Education of China (www.moe.gov.cn) under grant Nos. 201802123091 (received by F. W.) and 201802123068 (received by Z. W.), Scientific Project of CAFUC (www.cafuc.edu.cn) under grant Nos. F2017KF02 and J2018-3 (both received by Z. W.), and Teaching Reform Project of CAFUC (www.cafuc.edu.cn) under grant No. E2020044 (received by Z. W.) .

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Amouri, A.; Alaparthy, V.; Morgera, S. (2018): Cross layer-based intrusion detection based on network behavior for IoT. *IEEE 19th Wireless and Microwave Technology Conference*, pp. 1-4.

Badshah, A.; Ghani, A.; Qureshi M. A.; Shamshirband, S. (2019): Smart security framework for educational institutions using Internet of Things (IoT). *Computers, Materials & Continua*, vol. 61, no. 1, pp. 81-101.

Bostani, H.; Sheikhan, M. (2017): Hybrid of anomaly-based and specification-based IDS for Internet of Things using unsupervised OPF based on MapReduce approach. *Computer Communications*, no. 98, pp. 52-71.

Chelloug, S. A.; Ei-Zawawy, M. A. (2018): Middleware for Internet of things: survey and challenges. *Intelligent Automation and Soft Computing*, vol. 24, no. 2, pp. 309-317.

Farooq, M.; Waseem, M.; Khairi. A. (2015): A critical analysis on the security concerns of Internet of Things (IoT). *International Journal of Computer Applications*, vol. 111, no. 7, pp. 1-6.

Gan, Z.; Zhou, X. (2018): Abnormal network traffic detection based on improved LOF algorithm. *10th International Conference on Intelligent Human-Machine Systems and Cybernetics*, no. 1, pp. 142-145.

Kumar, S.; Kulothungan, K. (2017): An anomaly behavior-based detection and prevention of DoS attack in IoT environment. *Ninth International Conference on Advanced Computing*, pp. 287-292.

Li, Q.; Sun, R.; Wu, H. (2018): Parallel distributed computing based wireless sensor network anomaly data detection in IoT framework. *Cogitative Systems Research*, no. 52, pp. 342-450

Lyu, L.; Jin, L. (2017): Fog-empowered anomaly detection in IoT using hyper ellipsoidal clustering. *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1174-1184.

Ren, Y. J.; Liu, Y. P.; Ji, S.; Sangaiah, A. K.; Wang, J. (2018): Incentive mechanism of data storage based on blockchain for wireless sensor networks. *Mobile Information Systems*, vol. 2018, pp. 1-10.

Sedjelmaci, H.; Senouci, S. M.; Al-Bahri, M. (2016): A lightweight anomaly detection technique for low-resource IoT devices: a game-theoretic methodology. *IEEE International Conference on Communications*, pp. 1-6.

Shafi, Q.; Basit, A.; Qaisar, S. (2018): Fog-assisted SDN controlled framework for enduring anomaly detection in an IoT Network. *IEEE Access*, vol. 6, no. 1, pp. 73713-73723.

Su, J.; Sheng, Z.; Leung, V. C. M.; Chen, Y. (2019): Energy efficient tag identification algorithms for RFID: survey, motivation and new design. *IEEE Wireless Communications*, vol. 26, no. 3, pp. 118-124.

Su, J.; Sheng, Z.; Xie, L.; Li, G.; Liu, A. (2019): Fast splitting-based tag identification algorithm for anti-collision in UHF RFID system. *IEEE Transactions on Communications*, vol. 67, no. 3, pp. 2527-2538.

Sun, L.; Ma, J.; Wang, H. (2018): Cloud service description model: an extension of USDL for cloud services. *IEEE Transactions on Services Computing*, vol. 11, no. 2, pp. 354-368.

Thanigaivelan, N. K.; Nigussie, E.; Virtanen, S. (2018): Hybrid internal anomaly detection system for IoT: reactive nodes with cross-layer operation. *Security and Communication Networks*, pp. 1-15.

Wang, J.; Gao, Y.; Liu, W.; Sangaiah, A. K.; Kim, H. J. (2019): An intelligent data gathering schema with data fusion supported for mobile sink in wireless sensor networks. *International Journal of Distributed Sensor Networks*, vol. 15, no. 3, pp. 1-9.

Wang, J.; Gao, Y.; Yin, X.; Li, F.; Kim, H. J. (2018): An enhanced PEGASIS algorithm with mobile sink support for wireless sensor networks. *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1-9.

Yin, B.; Zhou, S. W.; Zhang, S. W.; Gu, K.; Yu, F. (2017): On efficient processing of continuous reverse skyline queries in wireless sensor networks. *KSII Transactions on Internet and Information Systems*, vol. 11, no. 4, pp. 1931-1953.