International Journal of Innovative
Computing, Information and Control
Volume **15**, Number **4**, August **2019**

ICIC International ©2019 ISSN 1349-4198
pp. **1351–1364**

EARLY STOPPING FAULT DIAGNOSIS AGREEMENT ON WIRELESS SENSOR NETWORK OF IOT

Shu-Ching Wang[1], Wei-Shu Hsiung[1], Mao-Lun Chiang[1,*] and Yao-Te Tsai[2,*]

[1]Department of Information Management
Chaoyang University of Technology
168 Gifeng E. Rd., Wufeng, Taichung 41349, Taiwan
{ scwang; s10714902 }@cyut.edu.tw; *Corresponding author: mlchiang@cyut.edu.tw

[2]Department of International Business
Feng Chia University
100 Wenhwa Rd., Seatwen, Taichung 40724, Taiwan
*Corresponding author: yaottsai@fcu.edu.tw

Received December 2018; revised April 2019

Abstract. *The urban Internet of Things (IoT) is designed to support the vision of smart city, which aims at exploiting the most advanced communication technologies to support added-value services for the administration of the city and for the citizens. Conventionally, Wireless Sensor Network (WSN) is often used to construct an IoT environment. However, when an ultra large disaster occurs, communication networks would be severely disconnected by the damages of network nodes. In order to achieve fault-tolerance of WSN in IoT, one must deal with the Byzantine Agreement (BA) problem. The BA problem occurs when the fault-free nodes in a distributed system can reach a common agreement before performing specified in instances where faults may exist. Solving most BA problems requires that all fault-free nodes reach agreement during the same round. This kind of agreement is called Immediate Byzantine Agreement (IBA). When the number of actual faulty nodes is less than the number of tolerated faulty nodes, the participants should be stopped the protocol and reach agreement earlier. This kind of protocol is called as an Early Stopping Protocol (ESP). Our study also solves the Fault Diagnosis Agreement (FDA) problem. The goal of FDA is to enable each fault-free node to detect/locate a common set of faulty nodes. However, a lot of messages need to be collected to solve the BA and FDA problem in the previous works. In this study, a new protocol ESFDA (Early Stopping Fault Diagnosis Agreement) is proposed to allow the participants to reach agreement earlier as if the participants found the number of faulty nodes less than the upper bound of allowable faulty nodes when the protocol is executed. Meanwhile, ESFDA can detect/locate the maximum number of faulty nodes with dual failure modes (dormant and malicious exist simultaneous) efficiently as early stop as possible. As a result, ESFDA can be proposed to solve the BA, ESP, and FDA problems simultaneously to provide greater computing abilities by enhancing the reliability of WSN.*

Keywords: IoT, WSN, Fault tolerant, Fault diagnosis agreement, Early stopping

1. **Introduction.** Due to the growing developments in advanced metering and digital technologies, smart cities have been equipped with different electronic devices on the basis of Internet of Things (IoT), therefore becoming smarter than before [1]. The IoT is a recent communication paradigm that envisions a near future, in which the objects of everyday life will be equipped with microcontrollers, transceivers for digital communication, and suitable protocol stacks that will make them able to communicate with one another and with the users, becoming an integral part of the Internet [1].

DOI: 10.24507/ijicic.15.04.1351

The concept of IoT aims at making the Internet even more immersive and pervasive [2]. By enabling easy access and interaction with a wide variety of devices, the IoT will foster the development of a number of applications that make use of the potentially enormous amount and variety of data generated by such objects to provide new services. The application of the IoT paradigm to an urban context is of particular interest, as it responds to the strong push of many national governments to adopt ICT (Information and Communications Technologies) solutions in the management of public affairs, thus realizing the so-called smart city concept [3].

Wireless Sensor Networks (WSNs) are playing a more and more key role in several application scenarios such as smart city. Furthermore, WSNs are characterized by high heterogeneity because there are many different proprietary and non-proprietary solutions. This wide range of technologies has delayed new deployments and integration with existing sensor networks [2,4]. Fueled by the adaptation of a variety of enabling devices such as embedded sensor and actuator nodes, the IoT has stepped out of its infancy and is on the verge of revolutionizing current fixed and mobile networking infrastructures into a fully integrated future Internet [5]. The WSN can be used as a sensing drive arm for the IoT, and is seamlessly integrated into the infrastructure to form a smart city.

Since WSNs of IoT are deployed in open areas without protection from disaster, they are vulnerable to various types of attacks. Sensor nodes of WSN are essential for detecting various kinds of data of the serious disasters in residential areas [6]. Figure 1 is a structure of a typical WSN [7]. Moreover, the network configuration underlying the emergency must be considered in setting up a new network [8]. Therefore, the stability and reliability of WSNs are important issues to keep environment good for data transmission [9]. In other words, to propose a mechanism to allow all well-perform nodes to reach an agreement is necessary to ensure WSN stable and reliable.

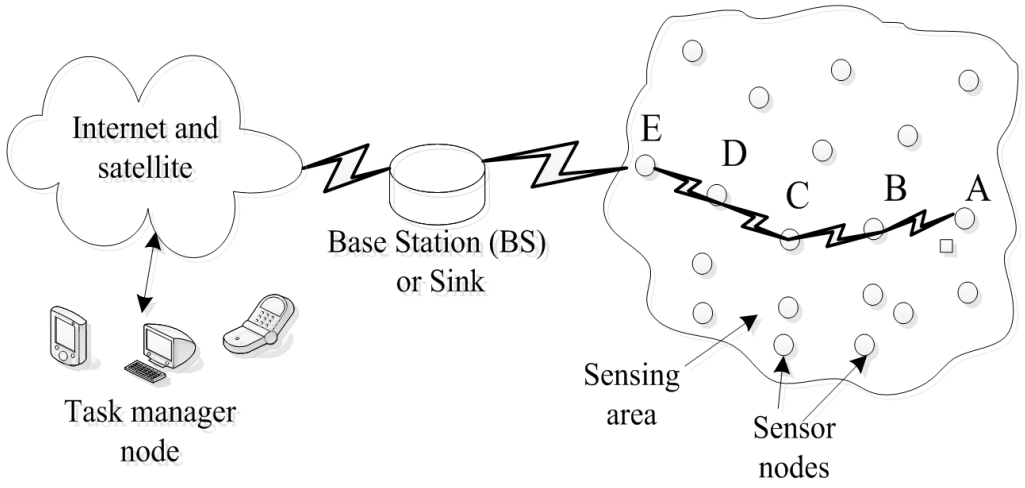

FIGURE 1. The structure of a typical WSN

The WSN consists of a set of nodes that can communicate with each other by exchanging messages. A mechanism to allow a given set of nodes to agree on a common value is necessary for reliable smart city [5,9]. Such a unanimity problem was called Byzantine Agreement (BA) [10]. It requires a number of independent nodes to reach agreement in cases where some of those nodes might be faulty. Namely, the goal of BA is making the fault-free nodes reach a common value.

Traditionally, BA problem is defined as follows. (1) There are n (where $n > 3$) nodes, of which at most one-third of the total number of nodes can fail without completely

disrupting a workable network. (2) The nodes communicate with each other by message exchange in a well-defined network. (3) The message's sender is always identifiable by the receiver. (4) A node is chosen as a source randomly, and its initial value is broadcast to other nodes as well as to itself [10]. Agreement is reached if all fault-free nodes agree on a common value. Based on these assumptions, various protocols for the BA problem have been developed in order to meet the following requirements [10].

- **Agreement**: All fault-free nodes agree on a common value v.
- **Validity**: If the initial value of the source is v_s and the source is fault-free, then all fault-free nodes shall agree on the value v_s; i.e., $v = v_s$.

Some protocols require that each fault-free node stops the message exchange in the same round $(f_m + 1)$ and achieves agreement immediately when the number of tolerable malicious node faults is (f_m) in system [11]. This kind of agreement is called Immediate Byzantine Agreement (IBA) [12]. However, if no faulty node exists in the system or the number of faulty nodes is less than f_m, then $f_m + 1$ rounds of message exchanges are still needed to reach a common agreement. Therefore, an improved algorithm is invoked, the Eventual Byzantine Agreement (EBA) [12], that allows individual nodes to stop during rounds in which the received messages are sufficient to reach a common agreement.

After the common agreement is achieved, the collected messages can be used to help each fault-free node to detect/locate faulty components and reconfigure the network subsequently. For this reason, a closely related and important issue, the Fault Diagnosis Agreement (FDA) problem [13] also must be reviewed. After achieving FDA, the performance and integrity of a distributed network can be guaranteed. The protocol designed for FDA must satisfy the following conditions.

- ***Agreement**: All fault-free nodes identify the common set of faulty nodes.
- ***Fairness**: No fault-free component is incorrectly detected as faulty by any fault-free node.

The protocol MFDA proposed by Wang et al. [13] can be used to solve the FDA problem in cluster-based WSNs. However, MFDA is an IBA that all fault-free nodes must be in the same round to reach an agreement, and MFDA can only detect/locate malicious faulty nodes. Therefore, the results of MFDA cannot be applied to WSNs with dual failure modes (including the malicious and dormant faults). Hsiao et al. proposed an *evidence-based* protocol, FDAMIX, with dual failure modes to solve the FDA problem [14]. The FDAMIX protocol first collects the messages received in the BA protocol GPBA designed for dual faults as evidence, and then uses this evidence to detect/locate faulty nodes [3]. Since a faulty node may alter the messages they transmit, these symptoms can be used to detect/locate faulty nodes. In the FDAMIX protocol, all fault-free nodes can identify the maximum number of faulty nodes and tolerate dual faults in nodes. However, their protocol requires $f_m + 2$ (where $f_m \leq \lfloor (n-1)/3 \rfloor$) rounds of message exchanges due to the GPBA requiring $f_m + 1$ rounds of message exchanges to achieve agreement even if the number of faulty nodes is less than f_m. Since message passing is a time-consuming phase, the number of messages increases protocol overhead resulting in the protocol being unable to solve the BA problem efficiently and quickly. Namely, if the number of actual faulty nodes is found less than the upper bound of allowable faulty nodes, the agreement protocol can stop earlier. The early stopping protocol is more practical and efficient than traditional researches. Therefore, an efficient protocol ESFDA (Early Stopping Fault Diagnosis Agreement) is proposed in this study to solve the EBA problem and help all fault-free nodes on WSN of IoT to detect/locate maximum number of faulty nodes having dual failure modes using the minimum number of message exchanges.

The rest of this paper is organized as follows. Section 2 illustrates the previous work and underlying assumptions. The details of the proposed protocol are shown in Section 3. Section 4 illustrates the examples we have devised in detail. Subsequently, the correctness and complexity of the proposed protocol are illustrated in Section 5. Finally, the conclusion is presented in Section 6.

2. **Previous Work and Underlying Assumptions.** The components in the network may not always work well. A node is said to be fault-free if it follows protocol specifications during the execution of a protocol; otherwise, the node is said to be faulty. The symptoms of node failure can be classified into dormant fault and malicious fault [13]. A dormant faulty node always can be identified by the receiver if the transmitted message was encoded appropriately, i.e., by NRZ-code, Manchester code before transmission [15]. The behavior of a malicious faulty node is unpredictable and arbitrary. The message transmitted by a malicious faulty node is random or arbitrary. It is the most damaging failure type and causes the worst problem. When a node in the network is faulty or damaged, the services provision of the system will be affected. And the dormant faulty node and the malicious faulty node in the network may exist simultaneously. Most previous studies treat dormant faults as malicious faults; it is not very fair and realistic. Therefore, the node may have both dormant fault and malicious fault (dual failure modes) in the WSN as the assumption of this study. In this study, the agreement problem is revisited to enlarge the fault tolerant capability by allowing dual failure modes to exist in the WSN.

In general, IBA protocols require all fault-free nodes to reach a common agreement during the same round. Dolev and Reischuk [11] stated that IBA cannot be achieved for n (where $n > f_m + 1$) nodes with at most f_m (where $f_m \leq \lfloor (n-1)/3 \rfloor$) faults within f_m or fewer rounds. However, the EBA protocols allow fault-free nodes to achieve agreement during different rounds once they receive enough information [12]. In EBA protocols, the nodes may stop the protocol in round r ($< f_m$) due to the number of actual faulty components (f_{act}) being less than the maximum number of fault tolerant f_m; thus a lower bound of the number of message exchanges is $\min\{f_{act} + 2, f_m + 1\}$ for ESFDA.

Based on the *evidence-based* FDA protocol [16], the proposed protocol ESFDA can detect/locate faulty components by message exchange. The proposed protocol can solve the BA problem if the following constraints are satisfied:

[Constraint 1]: $n > \lfloor (n-1)/3 \rfloor + 2f_m + f_d$,

[Constraint 2]: $c > 2f_m + f_d$,

[Constraint 3]: $\sigma > f_m + 1$.

[Constraint 1] represents the number of nodes (n) required, it must be greater than $\lfloor (n-1)/3 \rfloor + 2f_m + f_d$, including both malicious faulty nodes (f_m) and dormant faulty nodes (f_d). The malicious faulty nodes must be fewer than $(2n + 3 - 3f_d)/6$; and fault detection can be achieved when the influence of the dormant faults is clear. [Constraint 2] specifies the connectivity requirement (c). Based on Graph theory [6], at least c disjoint paths must exist between any pairs of nodes n_a and n_b when the connectivity of the network is c. Therefore, the connectivity needed to satisfy the total number of received messages should be greater than the number of fake messages from the faulty node(s) when the influences of dormant faults are removed. In [Constraint 3], the required rounds of message exchange (σ) are assumed more than $f_m + 1$. Based on Abraham and Dolev [16], no fault diagnosis protocol for malicious faults is ever complete. The fault diagnosis protocol cannot detect/locate the faulty components if the symptoms of faulty behavior of nodes are not obvious. Therefore, the symptoms of faulty behavior on nodes appear during the message exchange procedure are assumed. Fisher and Lynch [17] proposed $f_m + 1$ rounds are the lower boundary for message exchanges if the transmission medium

is reliable. Therefore, the required number of rounds for solving the generalized BA problem should not be less than $f_m + 1$. According to the constraints above, the number of detectable/located faulty nodes is $f_m + f_d$ in our protocol.

Based on the above, this study proposes a novel protocol, ESFDA, to solve the BA, ESP, and FDA problems simultaneously. In the proposed ESFDA, the number of message exchanges is reduced and the reliability of WSN can be improved. In other words, compared to the previous methods [13,14], ESFDA allows each fault-free node can tolerate the maximum number of faulty nodes and detect/locate the maximum number of faulty nodes under dual failure modes using the minimum number of message exchanges in a WSN.

3. **The Proposed Protocol ESFDA.** As demonstrated by the WSN shown in Figure 1, each sensor node in a WSN is allocated to different sensing areas, and must be able to cooperate with other sensor nodes to provide greater computing abilities, even if some sensor nodes are faulty. As a result, a corresponding WSN is proposed in Figure 2. In this WSN environment, each sensor node works with other nodes to meet various IoT applications. Therefore, the fault tolerance issue needs to be modified to ensure the reliability of WSNs, and the faulty sensor nodes must be detected and eliminated. Thus, the concept of ESP is applied in this study that allows its participants to obtain common values early, during different rounds. As a result, a greater computing ability by enhancing the reliability of WSN of IoT can be provided by the early diagnosis WSN protocol.

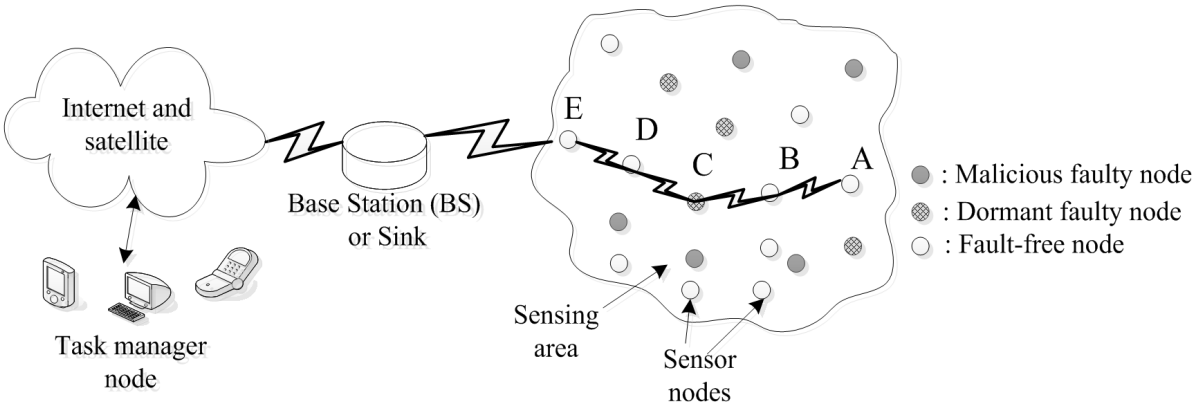

FIGURE 2. A WSN with dual faulty sensor nodes

In this section, the proposed ESFDA to solve the BA, ESP, and FDA problem in a WSN of IoT is introduced. There are three parts of an ESFDA: *message exchange phase*, *decision making phase*, and *fault diagnosis phase*. Additionally, the *early stopping function* is invoked during the *message exchange phase* to determine whether the number of received messages is sufficient to reach a common value.

At the beginning of ESFDA, the required rounds (σ) of message exchanges are $\min\{f_{act} + 2, f_m + 1\}$ based on the results of [11,12] where f_{act} is the actual number of faulty nodes that occurred in the system. Dolev et al. [12] had proved that the $\min\{f_{act} + 2, f_m + 1\}$ is the lower bound of the number of rounds required for EBA. In general, the sensing data is obtained by sensor node. The corresponding sensors must be able to cooperate with each other to provide greater computing abilities in the *message exchange phase*. As a result, each sensor node needs to broadcast its initial values to the other sensor nodes until round σ. During this phase, an *ic-tree* (information collecting tree – T_i) is constructed to store the received messages, as shown in Figure 3. The vertices of the *ic-tree* are marked

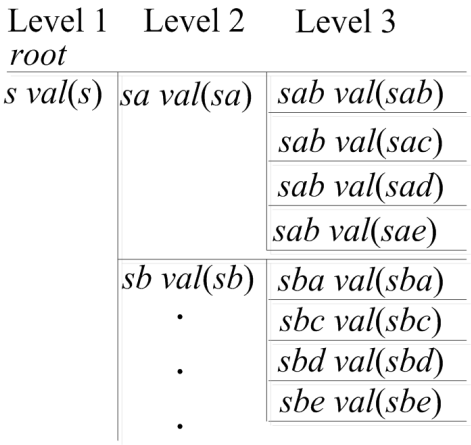

FIGURE 3. The *ic-tree* (T_i)

with a list of sensor node names. The list of sensor node names contains the name of the sensor nodes from which stored messages are transmitted. The vertices with repeated sensor node names must be removed in the *ic-tree* in order to avoid cyclical influence from the faulty nodes. The cyclical influence is due to the fact that the messages from a faulty sensor node may be stored repeatedly in an *ic-tree* and result in an incorrect common value from the taking of a simple majority. Furthermore, each node can execute the *early stopping function* to check whether the protocol can be stopped when $\sigma > 2$ during *message exchange phase*. This function uses the concept found in [10] to solve dual failures with nodes.

In the *early stopping function*, the MAJ function is applied to the *ic-tree* of each node n_i (where $1 \leq i \leq n$) to obtain the MAJv_i. Subsequently, the Δv_i^k (where $1 \leq k \leq f_m+1$) is accumulated when the vertices of the *ic-tree* of each node are equal to the MAJv_i. The protocol can be stopped early if the following constraint can be satisfied:

$\Delta v_i^k > (f_m-(r-1))+(n-r-1-|\lambda|)/2$, where f_m is the number of allowable malicious faulty nodes, r is the current round of message exchanges, n is the total number of nodes in a WSN of IoT, and $|\lambda|$ is the number of λs that received from dormant faulty nodes.

In the *decision making phase*, each fault-free node can compute a common value by applying the voting function VOTE to the root of an *ic-tree*. In the last phase, the *fault diagnosis phase*, the dormant faulty components can be detected/located by rule R_1 since the transmitted message is encoded by using the Manchester code [15] before transmission. Therefore, the messages sent from the dormant faulty components can be replaced by λ and can be removed in each round of message exchange. In addition, the rules R_2, R_3, and R_4 can be used to detect/locate malicious faulty nodes in the *fault diagnosis phase*. The primary function of R_2 and R_3 is to use the BA constraint to identify the faulty components. Initially, node n_k can be marked as a malicious fault-like node ($^*f_k$) when $val(\alpha k) \neq$ MAJv_i of each *ic-tree* in node n_k. Subsequently, node n_k can be detected as a malicious faulty node when the $^*f_k > \lfloor (n-1-f_d)/3 \rfloor$. Finally, rule R_4 is used to identify whether the source is fault-free by taking majority value on T_i^k. Due to the $n > \lfloor (n-1)/3 \rfloor + 2f_m + f_d$ constraint, the source node can be identified as a malicious faulty node when $val(T_i^{k-1}) \neq val(T_j^{k-1})$. The details of ESFDA are shown as Figure 4 and the parameters used in this study are set as follows.

Preprocessing:
To determine the total number of rounds required $\sigma = \min\{f_{act} + 2, f_m + 1\}$;

Message Exchange Phase:
$\sigma = 1$, do:

1) The source node broadcasts its initial value v_i to other nodes and itself.
2) Each node n_j stores its received message to construct a vector V_j; if the source node has a dormant fault, then the value λ replaces the initial value received from the source node.

$\sigma = 2$, do:

1) Each node broadcasts V_i, and then receives a column vector broadcasted by other nodes, and constructs an *ic-tree* for each round. If the sending node has a dormant fault, then the value λ replaces the value received from the sending node.

For $\sigma > 2$, do:

1) Each node broadcasts its *ic-tree* to others, and stores the value to level σ of its *ic-tree*. If the sending node has a dormant fault, then the value λ replaces the value received from the sending node.
2) Run **Early Stopping Function**.

Decision Making Phase:

1) Applying the **VOTE Function** to the root of each node's *ic-tree* T_i, the common value VOTE(s) is obtained.

Fault Diagnosis Phase: Let n_h $(1 \le h \le n)$ be a fault-free node. n_h can detect node n_k $(1 \le k \le n)$ as faulty if:

R1: n_h receives the λ from n_k (or no message is received from n_k) and the number of copies of n_k is greater than $\lfloor (n-1-f_d)/3 \rfloor$; then node n_k is a dormant faulty node.

R2: For each T_i
If $val(\alpha k) \neq \mathrm{MAJ}v_i$, then ${}^*f_k = {}^*f_k + 1$.
If ${}^*f_k > \lfloor (n-1-f_d)/3 \rfloor$, then node n_k is a malicious faulty node.

R3: If ${}^*f_k < \lfloor (n-1-f_d)/3 \rfloor$ and $f_m < \lfloor (n-1-f_d)/3 \rfloor$, then
For each T_i, $1 \le i \le k$
${}^*f_\alpha = {}^*f_\alpha + 1$;
If ${}^*f_\alpha > \lfloor (n-1-f_d)/3 \rfloor$, then node n_α is a malicious faulty node.

R4: Compute the T_i^{k-1} by taking the majority value on each T_i^k.
If $val\left(T_i^{k-1}\right) \neq val\left(T_j^{k-1}\right)$ $(i \neq j)$ and $f_m < \lfloor (n-1-f_d)/3 \rfloor$, then the source s is a malicious faulty node.

Early Stopping Function:

Step 1: Use the **MAJ Function** on the *ic-tree* of each node and obtain the majority value $\mathrm{MAJ}v_i$;
Step 2: Compute the number of values equal to $\mathrm{MAJ}v_i$ as Δv_i^k $(1 \le k \le f_m + 1)$;
Step 3: If the value $\Delta v_i^k > (f_m - (r-1)) + (n - (r-1) - \lambda)/2$ then the message exchange is stopped; else go to next round of **Message Exchange Phase**.

MAJ Function:

1) The majority value in the set of $\{val(\alpha j) | 1 \le j \le n\}$, if the majority exist.
2) Otherwise, the complement value of $val(\alpha)$, denoted as $\neg val(\alpha)$, is chosen.

VOTE(α) Function:

1) $val(\alpha)$, if α is a leaf.
2) The majority value in the set of $\{\mathrm{VOTE}(\alpha i) | 1 \le i \le n$, and vertex αi is a child of vertex $\alpha\}$, if such majority values exist.
3) A default value Φ is chosen.

FIGURE 4. The proposed protocol ESFDA

- n: The total number of nodes in a WSN of IoT.
- $val(s)$: The value of the source node.
- V_k: The vector stored in node n_k.
- MAJ_i: The majority value of node n_i.
- λ: The value substituted for the received value from a dormant faulty node.

- c: The connectivity of WSN. Based on the Graph theory [18], at least c disjoint paths must exist between any pairs of node n_a and n_b when the connectivity of the network is c.
- Φ: The default value, and $\Phi \in \{0, 1\}$.
- σ: The required rounds of message exchanges, $\sigma = \min\{f_{act} + 2, f_m + 1\}$.
- f_m: The number of allowable malicious faulty nodes.
- f_d: The number of allowable dormant faulty nodes.
- f_{act}: The actual number of nodes with malicious fault.
- T_i: An information collecting tree of node n_i.
- ${}^*f_k$: The number of malicious fault-like nodes n_k.

In short, the concept is used to detect/locate the influences of faulty nodes in this study. In the worst case, the received value in the first round of message exchange of each node is different due to the fact that the malicious faulty source node exists. Similarly, the values stored of Level 2 of *ic-tree* in the second round of message exchanges received from the first round of message exchange are influenced by the malicious faulty nodes. However, the influences from second round of message exchanges can be removed in third round of message exchanges by taking the majority value on Level 3. This is because there are more than $\lfloor (n-1-f_d)/3 \rfloor$ fault-free nodes in a network; thus, the stored values of Level 2 of *ic-tree* in third round of message exchanges can be obtained without the faulty influence of second round of message exchanges. Namely, the stored values of Level 2 of *ic-tree* in third round of message exchanges are the actual values sent from the second round of message exchanges without influence from the malicious faulty nodes. Therefore, node n_k can be marked as ${}^*f_k$ when its value is not equal to the majority value of its received values on all nodes. Subsequently, all ${}^*f_k$s are accumulated from Level 3 of *ic-tree* and compared with BA constraint $\lfloor (n-1-f_d)/3 \rfloor$. Node n_k will be identified as a malicious faulty node when number of ${}^*f_k$ is more than $\lfloor (n-1-f_d)/3 \rfloor$, because there are at least f_m+1 fault-free nodes in the network. In next section, the details of the proposed protocol ESFDA are introduced. The influence situation of each round is shown in Figure 5.

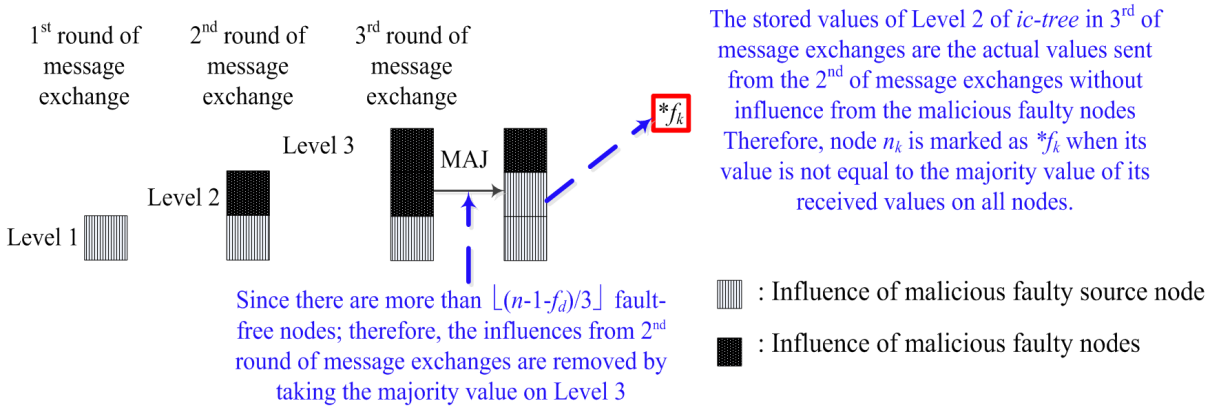

FIGURE 5. The influence situation of each round

4. **An Example of Executing ESFDA.** In this section, an example is used to illustrate the protocol ESFDA executed. There are 14 nodes in the original WSN of IoT in Figure 6. In this example, nodes n_a and n_b are assumed to be malicious nodes; furthermore, node n_a is a malicious source node. Node n_c is assumed to be a dormant faulty node.

The protocol begins with the source node n_a sending the values 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1 and 1 to nodes n_a, n_b, n_c, n_d, n_e, n_f, n_g, n_h, n_i, n_j, n_k, n_l, n_m, and n_n respectively.

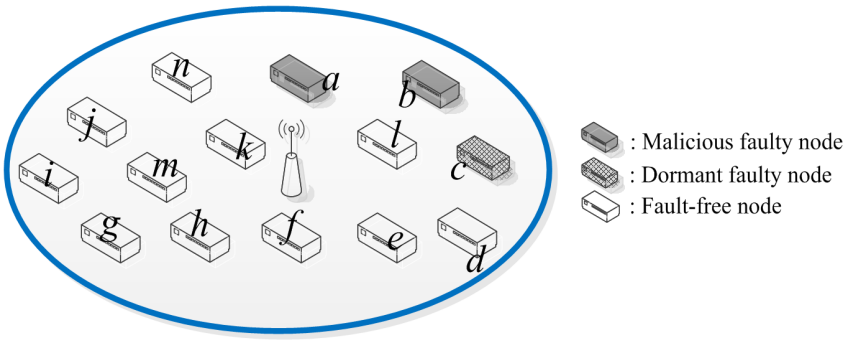

FIGURE 6. The original WSN

TABLE 1. The transfer values of faulty nodes

Sending node \ Receiving node	n_a	n_b	n_c	n_d	n_e	n_f	n_g	n_h	n_i	n_j	n_k	n_l	n_m	n_n
n_a (malicious node)	0	0	0	0	0	0	0	1	1	1	1	1	1	1
n_b (malicious node)	0	1	0	1	1	1	1	0	1	0	0	0	0	0
n_c (dormant node)	λ	λ	0	λ	λ	λ	λ	λ	λ	λ	λ	λ	λ	λ

Subsequently, each node exchanges its received value with other nodes. Unfortunately, the malicious nodes n_a and n_b may send differing values to other nodes, and the message(s) through dormant faulty node n_c can be detected and the value λ is replaced as the message received, as shown in Table 1. In this example, the behaviors of malicious faulty nodes are assumed to be smart and consistent with their transfer procedure. For example, the malicious node n_b sends the value 1, which is different from the received value from the source node n_a, to node n_e. Subsequently, node n_b always sends 1 to node n_e to avoid detection by node n_e. After three rounds of message exchanges, each fault-free node can construct its *ic-tree* T_i level by level, as in Figure 7.

In general, our ESFDA protocol requires $\sigma = \lfloor (n-1-f_d)/3 \rfloor = \lfloor (14-1-1)/3 \rfloor + 1 = 5$ rounds to exchange messages. Based on the *early stopping function* of the ESFDA protocol, the number of messages is insufficient to achieve agreement, each node cannot stop early during the first round and second round. In the third round, the message exchange can be stopped because the early stopping condition $\Delta v_i^k > (f_m - (r-1)) + (n - (r-1) - \lambda)/2$ is satisfied. The computations of Δv_i^k in the third step of Early Stopping Function are as follows:

Round 1: $\Delta v_{d\sim n}^1 = 1 < (f_m - (r-1)) + (n - (r-1) - |\lambda|)/2 = (4 - (1-1)) + (14 - (1-1) - 1)/2 = 10.5$ (cannot stop)

Round 2: $\Delta v_{d\sim n}^2 = 8 < (f_m - (r-1)) + (n - (r-1) - |\lambda|)/2 = (4 - (2-1)) + (14 - (2-1) - 1)/2 = 9$ (cannot stop)

Round 3: $\Delta v_{d\sim n}^3 = 8 > (f_m - (r-1)) + (n - (r-1) - |\lambda|)/2 = (4 - (3-1)) + (14 - (3-1) - 1)/2 = 7.5$ (stop)

For example: Δv_d^3 of $val(ab)$'s sub-tree in Level $3 = 6 > 7.5$, then the protocol ESFDA can be stopped in Round 3. Based on the early stopping function, the collected messages of each node are sufficient to achieve agreement and move into next phase, the *decision making phase*. The function VOTE is used to obtain a common value in this phase, and the common value ("1") of fault-free nodes can be reached if $n > \lfloor (n-1)/3 \rfloor + 2f_m + f_d$, where the connectivity (c) of the network is $c > 2f_m + f_d$.

Subsequently, the *fault diagnosis phase* is invoked to detect/locate the faulty nodes. For example, node n_d can use the rule R_1 to detect node n_c as a dormant faulty node. Then,

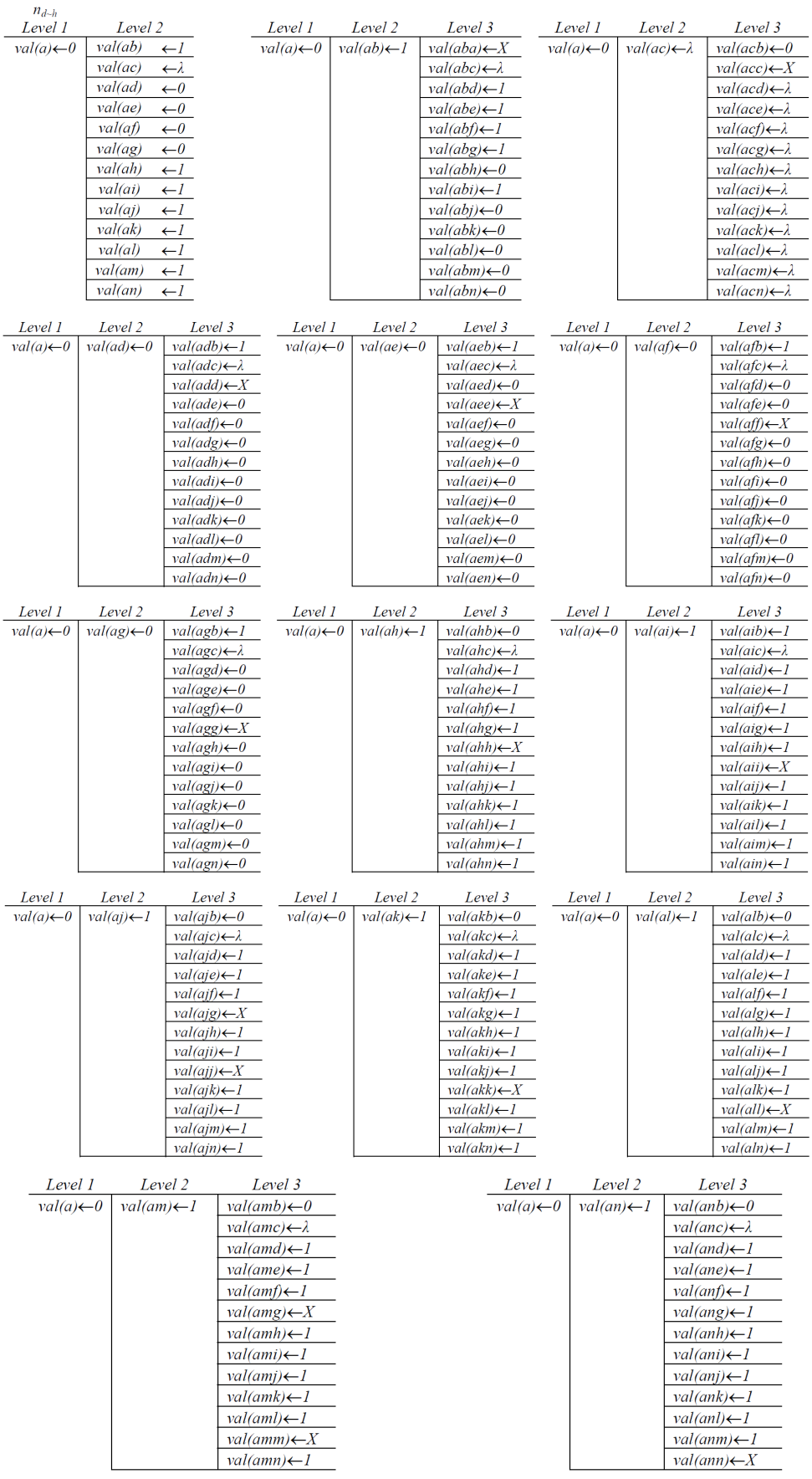

FIGURE 7. The *ic-tree* of each fault-free node

node n_d marks some nodes as fault-like nodes ($^*f_k$) by rule R_2 when $val(\alpha k) \neq \text{MAJ}v_i$. Therefore, node n_b can be detected as a malicious faulty node due to $^*f_b > \lfloor (n-1-f_d)/3 \rfloor = 4$. In rule R_3, the precursor nodes (the sender in last round) of fault-like nodes are marked when $^*f_k < \lfloor (n-1-f_d)/3 \rfloor$, such as with n_d, n_e, n_f and n_g. Therefore, node n_b can be detected as a malicious faulty node due to $^*f_b > \lfloor (n-1-f_d)/3 \rfloor$ ($^*f_b = {}^*f_b + 4$) after $^*f_b$ is accumulated in rule R_3. Furthermore, the source node n_a can be detected as a malicious faulty node by R_4. The results of diagnosis of nodes are shown as follows.

R_1: n_c as a dormant faulty node
R_2: $^*f_b$: (n_b as a malicious faulty node) $^*f_d \sim {}^*f_n$: 1
R_3: $val(abd, abe, abf, abg) = {>}^*f_b + 4$ (n_b as a malicious faulty node)
R_4: n_a as a faulty source node

5. **The Correctness and Complexity of ESFDA.** In this section, the correctness and complexity of ESFDA are proved if $n > (\lfloor (n-1)/3 \rfloor) + 2f_m + f_d$ and $c > 2f_m + f_d$, where f_m is the number of allowable malicious faulty nodes and f_d is the number of allowable dormant faulty nodes existing simultaneously. Only $\min\{f_{act} + 2, f_m + 1\}$ rounds of message exchanges are required to make all fault-free nodes reach a common agreement.

Lemma 5.1. *Messages sent through dormant faulty components can be detected by a fault-free destination node.*

Proof: A fault-free destination node can detect message(s) from dormant faulty components if the protocol appropriately encodes a transmitted message by using the Manchester code [15] before transmission.

Lemma 5.2. *Fault-free nodes can receive messages from fault-free nodes, if* $c > 2f_m + f_d$.

Proof: A fault-free sender node broadcasts a message to others and to itself. In the worst case, a fault-free node can receive $c - f_d$ messages transmitted in each round of the message exchanges because dormant faulty components can be detected. If $c - f_d > 2f_m$, a fault-free node can determine the correct messages from sender nodes by utilizing the majority value from the values received in each message exchange round.

Theorem 5.1. *A fault-free node can remove the faulty influences from dormant faulty nodes, if* $c > 2f_m + f_d$.

Proof: By Lemmas 5.1 and 5.2, the theorem is proved.

Lemma 5.3. *A fault-free destination node can detect a dormant faulty sender node by the forwarding technique.*

Proof: If the number of value λ is greater than or equal to $c - \lfloor (n-1)/3 \rfloor$, then the sending node has a dormant fault. This is because there are at most $\lfloor (n-1)/3 \rfloor$ malicious faulty nodes in the network; hence, there are at most $\lfloor (n-1)/3 \rfloor$ non-λ values in the vector V_i.

Theorem 5.2. *A fault-free node can detect a dormant faulty node in the network.*

Proof: In the protocol ESFDA, there are σ rounds of message exchanges during BA problem, where $f_m \leq \lfloor (n-1)/3 \rfloor$ and $n > 3$, so there are at least two rounds of message exchanges during the *message exchange phase*. Each fault-free node can receive the message from the source node during the first round of message exchange and receives other nodes' messages during the second round of message exchanges. Each node can receive all other nodes' messages that exist in the network after two rounds of message exchanges.

According to Lemma 5.3, each fault-free node can detect the dormant faulty nodes in the network.

Lemma 5.4. *Each value v of a fault-free node n_j received from the fault-free source node n_h is fixed. The value is fixed in round* $\min\{f_{act}+2, f_m+1\}$ *when there are f_{act} faulty nodes extant.*

Proof: Case 1: Source and receivers are fault-free nodes.

Generally, the fault-free source node n_j sends the initial value to the fault-free receiver n_h in the next round. Then the receiver n_h sends the same value to its descendant nodes. Subsequently, the value in the descendant path should be the same. This is because the fault-free descendant receiver does not change the original value that it transfers to its descendant receivers.

Case 2: Source and receivers are faulty nodes.

The faulty source node sends different values to faulty receiving nodes; the value might alternate from round to round. Due to only f_{act} faulty nodes being extant in the system, the longest faulty path only has length f_{act}. Therefore, the influences of vertices of descendant path ($\geq f_{act}+1$) can be eliminated by the majority function; subsequently the values of the descendant path are fixed again based on Case 1. In other situations, these influences can be eliminated such as the faulty source node and the fault-free receiver nodes or the fault-free source node versus the faulty receiver nodes situations.

Theorem 5.3. *Protocol ESFDA can achieve agreement in* $\min\{f_{act}+2, f_m+1\}$ *rounds.*

Proof: By Lemma 5.4, the values of a descendant path are fixed to a common value in round $\min\{f_{act}+2, f_m+1\}$ and early stopping. Therefore, the constraints (Agreement) and (Validity) are also achieved in round $\min\{f_{act}+2, f_m+1\}$.

Lemma 5.5. *Protocol ESFDA satisfies the (Agreement) constraint if* $n > (\lfloor (n-1)/3 \rfloor) + 2f_m + f_d$, $c > 2f_m + f_d$, *and* $\sigma > f_m + 1$.

Proof: The influence of a faulty malicious node can be detected/located due to the symptoms of faulty node behavior appearing in the *message exchange phase*. Each fault-free node can mark and accumulate the malicious fault-like nodes when $val(\alpha k) \neq$ majority value in each *ic-tree*. This is because the correct vertex α has at least $2f_m+1$ children, of which at least f_m+1 children to exist in system. Thus, these f_m+1 correct vertices can detect/locate the faulty behavior of nodes, and the majority value of vertex α is common. As a result, all correct vertices of each *ic-tree* are common. The (Agreement) constraint is satisfied.

Lemma 5.6. *Protocol ESFDA satisfies the (Fairness) constraint if* $n > (\lfloor (n-1)/3 \rfloor) + 2f_m + f_d$, $c > 2f_m + f_d$, *and* $\sigma > f_m + 1$.

Proof: Each fault-free node in this study does not treat other fault-free nodes as faulty because there are at least f_m+1 correct vertices that can detect/locate the faulty behavior of nodes; thus, the (Fairness) constraint is satisfied.

Theorem 5.4. *Protocol ESFDA can solve the FDA problem if* $n > \lfloor (n-1)/3 \rfloor + 2f_m + f_d$, $c > 2f_m + f_d$, *and* $\sigma > f_m + 1$.

Proof: By Lemma 5.5 and Lemma 5.6, the theorem is proved.

Theorem 5.5. *Protocol ESFDA requires* σ $(\min\{f_{act}+2, f_m+1\})$ *rounds to solve the FDA problem in dual failure modes (containing malicious and dormant faults) in the network if* $n > \lfloor (n-1)/3 \rfloor + 2f_m + f_d$ *and* $c > 2f_m + f_d$, *and σ rounds are the minimum number of rounds of message exchanges.*

Proof: By Theorem 5.4, the theorem is proved.

Theorem 5.6. *ESFDA requires* $\min\{f_{act}+2, f_m+1\}$ *rounds to solve the EBA problem.*

Proof: Based on the work of Fischer and Lynch [17], for a fail-safe network, if the transmission medium is reliable, then f_m+1 rounds are the lower boundary for message exchanges. Therefore, the required number of rounds for solving the BA problem should not be less than f_m+1. In the EBA problem, the protocol of Abraham and Dolev [16] also proves that the actual number of rounds of message exchanges is $\min\{f_{act}+2, f_m+1\}$ when the fallible components are only nodes in well-defined connected networks. Therefore, the ESFDA protocol is more efficient than traditional protocols [3, 13] in cases where the ESFDA protocol stops early.

6. **Conclusions.** Smart city is the one that uses ICTs to make the city services and monitoring more aware, interactive, and efficient [5]. The utilization of ICT can achieve this objective presents an opportunity or the development of smart cities. Therefore, smart cities have been equipped with different electronic devices on the basis of IoT [1]. The IoT provides a virtual view, via the Internet Protocol, to a huge variety of real life objects. Its appeal is the ubiquitous generalized access to the status and location of any "thing". WSNs are well suited for long-term environmental data acquisition for IoT [19]. Nevertheless, when an ultra large disaster occurs, communication networks would be severely disconnected by the damages of network nodes. To achieve high reliability in a WSN of IoT, a mechanism that allows a set of nodes to reach a common agreement, even in the presence of faulty nodes, is needed.

In this study, our protocol ESFDA solves the BA problem in cases of nodes in dual failure modes. Our protocol can stop the procedure earlier than alternative protocols so long as there are sufficient messages. For example, if there are only two ($f_{act}=2$) malicious nodes in a 16 nodes WSN, the maximum rounds of message exchanges are decreased from 6 to 4 ($r=\min\{f_{act}+2, f_m+1\}$). After the procedure of message exchanges is stopped, the ESFDA can make each fault-free node detect/locate the maximum number of faulty nodes but needs f_m+1 rounds even if the $f_{act}+2$ is less than f_m+1. Therefore, the

Table 2. The comparisons among MFDA, FDAMIX and ESFDA

	MFDA [13]	**FDAMIX [14]**	**ESFDA**
Agreement category	IBA	IBA	EBA
Time to stop the agreement	Reach agreement during the ***same rounds***	Reach agreement during the ***same rounds***	Reach agreement during the ***different rounds***
Network topology	Cluster-based WSN	c-connectivity network	WSN
Faulty type	Malicious faulty nodes	Dormant and malicious faulty nodes existed simultaneously	Dormant and malicious faulty nodes existed simultaneously
Number of rounds	f_m+1	f_m+2	$\min\{f_{act}+2, f_m+1\}$
Fault tolerant	$f_c > 3f_{mc}$	$n > 3f_m+f_d$ $c > 2f_m+f_d$	$n > (\lfloor (n-1)/3 \rfloor) + 2f_m + f_d$ $c > 2f_m+f_d$

· f_{mc}: the number of allowable malicious faulty clusters; f_c: the number of clusters

proposed evidence-based protocol ESFDA can detect/locate maximal numbers of faulty nodes with dual failure modes using the minimum number of message exchanges. Our protocol is more practical and efficient than previous protocols [13,14]. A comparison of the studies among MFDA [13], FDAMIX [14] and the new proposed ESFDA is given in Table 2.

Acknowledgment. This work was supported in part by the Ministry of Science and Technology MOST 107-2221-E-324-005-MY3.

REFERENCES

[1] H. Arasteh, V. Hosseinnezhad, V. Loia, A. Tommasetti, O. Troisi, M. Shafie-khah and P. Siano, IoT-based smart cities: A survey, *Proc. of the IEEE 16th International Conference on Environment and Electrical Engineering*, pp.1-6, 2016.

[2] C. Perera, A. Zaslavsky, P. Christen and D. Georgakopoulos, Sensing as a service model for smart cities supported by Internet of Things, *Trans. Emerging Telecommunications Technologies*, vol.25, no.10, pp.81-93, 2014.

[3] H. S. Siu, Y. H. Chin and W. P. Yang, Byzantine agreement in the presence of mixed faults on processors and links, *IEEE Trans. Parallel and Distributed System*, vol.9, no.4, pp.335-345, 1998.

[4] L. Sun and X. Liu, Algorithm of signal source localization via sensor networks, *ICIC Express Letters*, vol.12, no.5, pp.441-447, 2018.

[5] J. Jin, J. Gubbi, S. Marusic and M. Palaniswami, An information framework for creating a smart city through Internet of Things, *IEEE Internet of Things Journal*, vol.1, no.2, pp.112-121, 2014.

[6] A. E. Kouche, Towards a wireless sensor network platform for the Internet of Things: Sprouts WSN platform, *Proc. of the IEEE International Conference on Communications*, pp.632-636, 2012.

[7] I. F. Akyildiz, Wireless sensor networks in challenged environments such as underwater and underground, *Proc. of the 17th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2014.

[8] M. Takahashi, A self-repairing function exploiting resonance for high-gain adaptive control with faulty sensors, *International Journal of Innovative Computing, Information and Control*, vol.14, no.6, pp.2141-2150, 2018.

[9] P. K. Sangdeh, M. Mirmohseni and F. Poursabzi, Applying the Byzantine agreement in wireless sensor networks based on clustering, *Proc. of the 23rd Iranian Conference on Electrical Engineering*, pp.619-624, 2015.

[10] L. Lamport, R. Shostak and M. Pease, The Byzantine generals problem, *ACM Trans. Programming Languages and Systems*, vol.4, no.3, pp.382-401, 1982.

[11] D. Dolev and R. Reischuk, Bounds on information exchange for Byzantine agreement, *Journal of the ACM*, vol.32, pp.191-204, 1985.

[12] D. Dolev, R. Reischuk and H. R. Strong, Early stopping in Byzantine agreement, *Journal of the ACM*, vol.37, no.4, pp.720-741, 1990.

[13] S. C. Wang, K. Q. Yan and S. S. Wang, Fault diagnosis agreement in wireless sensor network, *Sensors and Materials*, vol.29, no.7, pp.1043-1048, 2017.

[14] H. S. Hsiao, Y. H. Chin and W. P. Yang, Reaching fault diagnosis agreement under a hybrid fault model, *IEEE Trans. Computers*, vol.49, no.9, pp.980-986, 2000.

[15] A. M. Căilean, B. Cagneau, L. Chassagne, M. Dimian and V. Popa, Novel receiver sensor for visible light communications in automotive applications, *IEEE Sensors Journal*, vol.15, no.8, pp.4632-4639, 2015.

[16] I. Abraham and D. Dolev, Byzantine agreement with optimal early stopping, optimal resilience and polynomial complexity, *Proc. of the 47th Annual ACM Symposium on Theory of Computing*, pp.605-614, 2015.

[17] M. Fisher and N. Lynch, A lower bound for the assure interactive consistency, *Information Processing Letters*, vol.14, no.3, pp.183-186, 1982.

[18] N. Deo, *Graph Theory with Applications to Engineering and Computer Science*, Courier Dover Publications, 2017.

[19] M. T. Lazarescu, Design of a WSN platform for long-term environmental monitoring for IoT applications, *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol.3, no.1, pp.45-54, 2013.