Reaching Byzantine Agreement underlying VANET

Shu-Ching Wang[1], Ya-Jung Lin[1] and Kuo-Qin Yan[2*]
[1] Department of Information Management, Chaoyang University of Technology
Taichung, 41349 Taiwan, R.O.C.
[e-mail: scwang@cyut.edu.tw]
[2] Department of Business Administration, Chaoyang University of Technology
Taichung, 41349 Taiwan, R.O.C.
[e-mail: kqyan@cyut.edu.tw]
Corresponding author: Kuo-Qin Yan

Received February 6, 2018; revised April 22, 2018; revised December 24, 2018; accepted February 6, 2019; published July 31, 2019

Abstract

The Internet of Things (IoT) enables machines and devices in a global network to connect and provide applications. The Vehicular Ad-hoc NETwork (VANET) allows vehicles in the network to communicate with each other as an application of the IoT. The safety and comfort of passengers can be improved through VANET related applications. In order to be able to provide related applications, there must be a reliable VANET topology. As a result of the Byzantine agreement (BA), fault tolerance can be solved in VANET. In order to improve the reliability of the system, even if some components in the system are damaged, a protocol is needed to assist the system to perform normally. Therefore, the BA problem in VANET with multiple impairments is revisited in this research. The proposed protocol allows all normal processing elements (PEs) to reach agreement using the least amount of information exchange. Moreover, the proposed protocol can tolerate the largest number of damaged PEs in VANET.

Keywords: Internet of Things, VANET, fault tolerant, Byzantine agreement

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

http://doi.org/10.3837/tiis.2019.07.002 **ISSN : 1976-7277**

1. Introduction

Recently, VANET has become more and more popular, it is the most important part of Intelligent Transportation Systems (ITSs) [1]. Through the VANET application, any vehicle can issue safety information to other vehicles and nearby stationary roadside units (RSUs) [2]. Based on the information received, the user can adjust their traffic path. The information received by the RSU will be provided to the Traffic Control Center as a reference for adjusting the traffic signal.

VANET is formed by vehicles and RSUs, and communication with each other uses multi-hop wireless communication. However, VANET is a network that is susceptible to signal interference, therefore may be subject to different attacks, causing the service of ITS to be affected [3]. VANET is characterized by highly mobile PEs, resulting in rapid network topology changes [4]. In order to provide a highly reliable VANET, even if there is a damaged PE in VANET, a protocol is needed to allow a group of PEs to reach an agreement.

Crucially, VANET requires a reliable data transfer mechanism to provide a reliable environment [5]. Therefore, the achievement of reliable and trustworthy agreement in the VANET is one of the most important issues to consider when designing a highly reliable system. In previous work, it was concluded that the ability to reach an agreement between normal PEs is a key component of any fault-tolerant system in order to cope with the effects of damaged components.

In 1980, the issue of agreement was first proposed by Pease et al. [6], and the problem was named the Byzantine Agreement (BA) problem [7]. Classic BA problems are considered for synchronous fixed networks where the boundaries of processing and transmission delays for normal components are proper [8].In the research of Dolev and Reischuk, they proved that there is only one damaged PE, and the protocol in the asynchronous network is impossible [9]. Since VANET is an asynchronous network, the results of previous research on BA issues [10-16] cannot be directly used to solve BA problems in asynchronous VANET. Chandra and Toueg have proposed a fault detector [17] that can be used to detect asynchronous PEs in VANET

To ensure fault tolerance and reliability in VANET, the BA problem in asynchronous VANET will be revisited in this study. The protocol proposed by this study was named Reliable and Trustworthy Agreement Protocol (RTAP). RTAP allows all normal PEs to achieve agreement using the least amount of information exchange. At the same time, RTAP can tolerate the maximun number of damaged PEs have been proven.

The rest of this article is arranged as follows: Section 2 illustrates the related work of our research. The definition of RTAP is given in Section 3. Section 4 describes RTAP in detail. Two examples are explained in Section 5. In Section 6, it is the proof of RTAP's lemmas and theorems. In Section 7, the conculsion and future works are provided.

2. Related Work

The BA problem and the architecture of VANET are introduced in this section.

2.1 BA problem

The BA problem is a useful issue when implementing highly reliable distributed services. In many related distributed applications, even if some PEs in the system fail, the normal PEs in the system can still achieve an agreement. With the agreement, many applications can be implemented, for example: the locating location of the copied files [18], two-part commitment in the distributed database system [19], and the landing task controlled by the flight control system [20]. This agreement problem was known as the Byzantine Protocol (BA) problem and was proposed by Lamport et al. [7]. The definitions of the BA problem are:

1) There are n PEs ($n \geq 4$), of which less than one-third of the PEs may be damaged PEs without damaging the feasible network.
2) In a fully connected network, PEs can exchange information with other PEs.
3) The sender of the information can be recognized by the receiver.
4) When the protocol is executed, a PE is selected as the Commander and its initial data d_c will be sent to other PEs.

According to the definitions, when the following conditions are satisfied, the BA can be solved [7]:

(Condition 1): All normal PEs agree on an agreement information.

(Condition 2): If the Commander is a normal PE, the initial data sent by the Commander will be agreed by all normal PEs.

In a distributed system, the components of system may not always work well. If a PE follows the provisions of the protocol during protocol execution, the PE is said to be normal; otherwise, the PE is a damaged PE. There are two kinds of PE failure: dormant-damaged and Byzantine-damaged [9]. When the transmitted information can be properly encoded by NRZ or Manchester code before transmission, then the receiver can recognize the dormant-damaged PE [21]. Because the behavior of Byzantine-damaged PEs is crazy, the information sent by Byzantine-damaged PEs is unpredictable. Therefore, the Byzantine-damaged is the most destructive type and leads to the most serious problems. In this study, the BA problem was revisited by allowing fault-tolerant blending of hybrid-damage (dormant and Byzantine-damaged) in VANET.

2.2 VANET architecture

VANET can provide large number applications of ITS. There are two main types of devices in VANET: on-board unit (OBU) and RSU. Where the OBU is installed on the vehicle and the RSU is deployed along the roadside of the road [22]. The communication between OBUs, or between an OBU and an RSU is achieved through a wireless medium.

There are a lot of PEs that join VANET and they have high mobility. The PE in the mobile VANET has high mobility and is therefore completely different from a fully connected network or broadcast network. Since PEs in VANET are highly mobile, these PEs can join or leave the network at will. Therefore, the agreement of PE in VANET is very important for network stability. As network technology grows very fast, the application of mobile VANET has become a trend. Therefore, it is very important to provide a stable and highly reliable environment by solving the problem of BA in mobile VANET. A new protocol in mobile VANET will be proposed in the study that allows all normal PEs in the system to obtain an agreement. In this study, the BA problem is revisited with multiple damaged PEs; the assumptions used in mobile VANETs are listed as follows:

1) PE in VANET is mobile.
2) If a PE joins or leaves the network, the PE will be treated as a newly joined PE.
3) Each PE can know the total number of current PEs in VANET at any time.

Many of the challenges of VANET can be solved through cluster networks [23]. Due to the needs of VANET, broadcast information is needed to update the vehicles with location and security information. Congestion generated in the network will lead to “broadcast storm problems" [24], Therefore, cluster topology is recommended to address congestion issues [24, 25]. VANET is a highly mobile environment, so the network topology has fast changing characteristics. Therefore, grouping vehicles into groups with similar mobility through clustering will reduce the mobility between neighbor PEs. Therefore, cluster VANET has become a more practical used VANET. PEs in a VANET cluster can achieve certain specific goals through cooperation with each other [25]. In other words, a specific service will be provided through a cluster-based VANET that consists of a set of loosely or tightly connected PEs. For example, the PEs in a cluster at the same traffic intersection can detect the status of traffic is smooth, with lots of traffic or traffic congestion. **Fig. 1** is an example of cluster-based VANET.

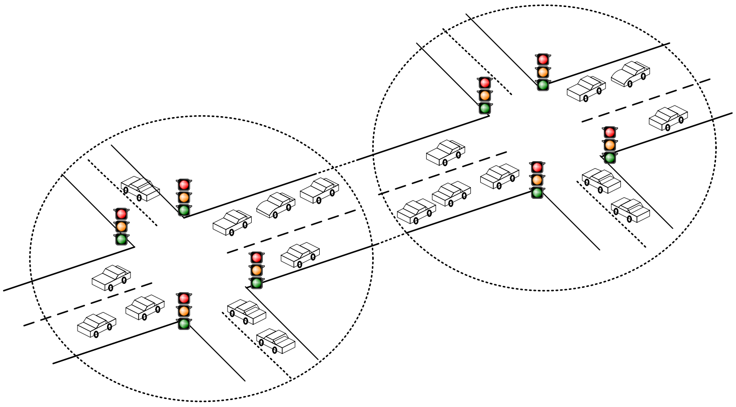


Fig. 1. An example of cluster-based VANET

3. Basic Concept of RTAP

In this study, it is considered that PE is reliable during protocol execution in VANET; PE (vehicle) may cause unpredictable behavior of information exchange due to interference from certain noise or intrusion by hijackers. In VANET, there are many PEs that are interconnected and work together to provide specific services. Implementing an agreement for the same information in VANET requires a protocol even if some PEs in the system damage, so that VANET can still operate normally.

The proposed protocol, Reliable and Trustworthy Agreement Protocol (RTAP), allows all normal PEs to acheve agreement using the least amount of information exchange, and can tolerate the largest number of damaged PEs in the VANET. In RTAP, there are two parts: the *information gathering part (ig-part)*, and the *agreement making part (am-part)*. In order for all normal PEs to achieve agreement, each PE must collect enough exchanged information from all other PEs if they are normal. Therefore, exchanging the received information can help normal PEs to obtain enough exchanged information.

Fischer and Lynch have proved that t+1 times of information exchange are sufficient and necessary conditions for solving BA problem, where $t = \lfloor (n\text{-}1)/3 \rfloor$ and n is the number of PEs in the network [8]. According to the research by Dolev & Reischuk [9] and Bar-Noy et al. [26], t+1 is the least amount of information exchange required to solve the BA problem. However, the total number of PEs may be changed at any time in the mobile VANET, so the number of times required is not fixed with (t+1). Nevertheless, the required times are expectable in the VANET, and it will follow the result of the protocol proposed by Bar-Noy et al. [26]. For instance, there are six PEs in the original VANET, and RTAP must execute $\delta= t+1 = (\lfloor(n-1)/3\rfloor)+1 = (\lfloor(6-1)/3\rfloor)+1 = 2$ times of information exchange. If the total number of PEs in the VANET becomes seven after the first time of information exchange because a new PE joins the network, the RTAP will need 3 times of information exchange (δ= t+1 =$\lfloor$(n-1)/3$\rfloor$)+1 = ($\lfloor$(7-1)/3$\rfloor$)+1). Therefore, RTAP needs one extra time to exchange information.

A hierarchical structure is used by this study to store received information. This structure is named as the message collect tree (mc-tree). Each normal PE maintains such mc-tree during the execution of RTAP. In the first time, Commander *c* transmits its initial data to other PEs. The PE can always identify the sender of the information as an assumption for the study. When an information sent from a Commander *c* is received by a normal PE, the received information (denoted as *inf*(*c*)) will be stored at the first layer or the root of the mc-tree of the receiving PE. In the second time, each PE transmits the *inf*(*c*) of its mc-tree to all other PEs. If the information *inf*(*c*) is sent by PE_1 to PE_2, the received information (denoted as *inf*(*c1*)) from PE_1 will be stored by PE_2 in the vertex *c1* of its mc-tree. Similarly, if *inf*(*c1*) is sent by PE_2 to PE_3, the received value is named *inf*(*c12*) and stored by PE_3 in the vertex *c12* of the mc-tree. The information *inf*(*c12…n*) stored in the vertex *c12…n* of the mc-tree indicates that the received information is transfered from the Commander *c*, PE_1, ..., PE_n, where PE_n is the last PE that passed the information. In summary, the first layer of the mc-tree is named *c*, indicating that the stored information is transferer from the Commander *c* in the first time; and the vertices of the mc-tree are named by the list of PE names. The PE name list contains the names of the PEs through which the stored information was sent. An example of mc-tree is shown in **Fig. 2**.

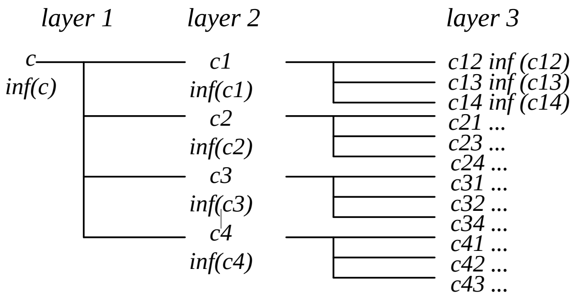


Fig. 2. An example of mc-tree

Basically, all normal PEs in each cluster of VANET execute RTAP in order to make all normal PEs in the same cluster achieve an agreement. During the *ig-part*, all PEs will communicate with other PEs and themselves. Furthermore, each PE has high mobility, and the PEs may join or leave the network at any time. When a PE joins a specific cluster through the *ig-part*, the function *PE-join* of RTAP will be executed to obtain the information from other PEs in the same cluster. Furthermore, if a PE leaves the cluster, then the function *PE-leave* of RTAP will be executed to reconstruct the mc-tree. Since the PEs in VANET are mobile, the times required for information exchange are not inherent in the beginning, and the number of information exchange must be adjusted at any time according to the (*t*+1) times proved by Fischer and Lynch [8] to achieve same value. Each PE must obtain the total number of PEs in the same cluster to determine the number of times of information exchange required. Thus, the protocol will use δ to represent the required times of information exchange. After δ times of information exchange, the collected information are stored in the mc-tree. In the *am-part*, each normal PE in the same cluster computes a same value by applying the majority decision function to the information collected by the *mg-part* and stored on a PE's mc-tree to reach an agreement.

The number of allowed fallible PEs in the network can be determined due to the total number of PEs in the network and the types of failure of the PEs. In the research result of Lamport et al. [7], the assumption of PE fault type is Byzantine in a static network. The constraint of Lamport et al. [7] is $n>3f_m$, where f_m is the number of Byzantine-damaged PEs. The f_m Byzantine-damaged PEs, the f_d dormant-damaged PEs, and the f_a absent PEs can be tolerated by RTAP at any time, where $n>3f_m+f_d+f_a$. Agreement will not be reached by all normal PEs if the total number of damaged PEs exceeds the limit. That is, each normal PE can reach the same protocol value if $n>3f_m+f_d+f_a$. Therefore, there are at least $n-\lfloor(n-1-f_d-f_a)/3\rfloor-f_d-f_a$ normal PEs, and these normal PEs have the same agreement value. That is, in the worst case, a joined PE can receive $n-\lfloor(n-1-f_d-f_a)/3\rfloor-f_d-f_a$ copies of the same information more than $\lfloor(n-1-f_d-f_a)/3\rfloor$, so a joined PE can get the agreement value by the majority function *DEC*.

The BA problem of asynchronous cluster-based VANET with dormant and Byzantine-damaged PEs will be considered in this study. And, the PEs in a cluster at the same traffic intersection can detect congestion or normal traffic status at the same intersection. Therefore, the agreement of each cluster is reached separately. The parameters of the proposed RTAP are listed as follows:

- VANET is an asynchronous network.
- Each PE in the VANET can be uniquely identified.
- N_i is the set of PEs in cluster i of VANET and $|N_i|= n_i$, where n_i is the number of PEs in the underlying cluster i of VANET, and $n_i\geq4$.
- Assume that the PEs of VANET may be damaged.
- There is only one commander in each cluster, and the initial data of the Commander is sent to all PEs for the first time when RTAP is executed.
- f_{im} is the number of Byzantine-damaged PEs in cluster i.
- f_{id} is the number of dormant-damaged PEs in cluster i.
- f_{ia} is the maximum number of absent PEs in cluster i.
- f_{in} is the maximum number of damaged PEs in cluster i, where $f_{in} = f_{im}+f_{id}+f_{ia}$.

4. The Reliable and Trustworthy Agreement Protocol (RTAP)

RTAP to solve the BA problem in a cluster-based VANET is introduced in this section. Generally, each normal PE of the same cluster of VANET executes the same RTAP simultaneously to reach agreement among normal PEs in the same cluster. There are two parts, the *ig-part* and the *am-part*. Because PEs in the VANET have high mobility, PEs may join or leave the VANET at any time. In the proposed protocol, the PE that joins in the VANET before the *am-part* is called the "joined PE", and the PE that leaves the VANET in the *ig-part* is called the "leaved PE". Since the leaved PE leaves the VANET, it cannot transmit and receive the information from other PEs in the VANET. Thus, the failure detector will also be unable to detect the leaved PE by the same concept (no response PE).

When performing RTAP, there are some PEs that can join or leave the network at any time, but each normal PE must determine the times required for information exchange in the *ig-part* firstly, then collect the received information sent by each PE. Finally, a majority decision function *DEC* is applied to the information collected in each normal PEs, and the agreement can be reached. In summary, when the BA protocol is applied, each normal PE can agree on the same information sent by the Commander. In addition, the number of times required for executing RTAP in cluster i is t_i+1 (where $t_i=\lfloor(n_i-1)/3\rfloor$). RTAP can tolerate f_{im}

Byzantine-damaged PEs, f_{id} dormant-damaged PEs, and f_{ia} absent PEs, where $n_i > 3f_{im}+f_{id}+f_{ia}$.

The goal of the *ig-part* is to collect the information. In VANET, PEs in the same cluster have all or part of the topology information of the cluster, and each PE can send information to other PEs in the same cluster. In the *ig-part*, the number δ must first be computed, where $\delta = t_i + 1$, and $t_i = \lfloor (n_i -1)/3 \rfloor$. RTAP let *r* represent the current number of information exchange. Then, in the first time of the *ig-part*, *r*=1, the Commander of cluster *i* transmits its initial data d_c to all other PEs in cluster *i*, and the d_c from the Commander will be stored by each PE in its first layer of mc-tree. After the first time of *ig-part* (*r*>1), each PE of cluster *i* transmits the information at layer *r*-1 in its mc-tree to all other PEs in cluster *i*. However, each PE of cluster *i* stores the information received at layer *r* of its mc-tree, where $1 \le r \le \delta$.

In addition, the PE that received the information can always detect the information through dormant-damaged PEs if the transmitted information is encoded by Manchester code [21]. Therefore, the information through dormant-damaged PEs can be detected and the value λ is replaced as the information received. The value λ is used to represent the absence information.

The goal of the *am-part* is to compute a same agreement value for solving the BA problem. After the *ig-part*, each PE in cluster *i* has its own mc-tree; and in the *am-part*, the repeatable vertices in the mc-tree are deleted to avoid duplication of interference by the damaged PEs. Then, the function *DEC* is used for each PE's mc-tree in cluster *i* from the layer t_i+1 to first layer of the mc-tree, and the agreement value *DEC*(*c*) is obtained. Finally, the agreement value *DEC*(*c*) of cluster *i* is transmitted to the joined PEs. RTAP tolerates f_{im} Byzantine-damaged PEs, f_{id} dormant-damaged PEs and f_{ia} absent PEs at any time, and requires δ information exchanges to reach an agreement in cluster *i*. The RTAP protocol is shown in **Fig. 3**.

```
Reliable and Trustworthy Agreement Protocol (RTAP)
Each PE executes the function preprocessing to get δ.
ig-part:
For r = 1 do:
    The Commander multicastes its initial data d_c to all PEs in the same cluster.
    Each PE stores d_c in the first layer of its mc-tree; if the Commander has a dormant fault, then the
    value λ replaces the d_c received from the Commander.
For r = 2 to δ do:
    If a PE joins the cluster, then PE-join is executed.
    If a PE leaves the network, then PE-leave is executed.
    The preprocessing is executed to check the required times.
    The information at layer (r-1)th of mc-tree for every PE are boradcased to all PEs in the same
    cluster.
    The received information are stored at layer r of PE's mc-tree; if the sending PE has a
    dormant-damaged, then the value λ replaces the information received from the sending PE.
am-part:
If a PE joins in the network, then PE-join is executed.
If a PE has left the network, then PE-leave is executed.
The repeatable vertices in the mc-tree are deleted.
A same agreement value of each PE is determined by using the function DEC.
Function preprocessing:
   δ = ⌊(n_i-1)/3⌋+1.
Function PE-join:
   1) The information received in the (r-1)th time is sent by each PE in cluster i to the new PE newly
      joined to cluster i.
   2) The new PE stores the majority of information received from other PEs in cluster i at layer r-1
```

```
        of its mc-tree.
**Function *PE-leave*:**
  The information received from the left PEs are delteded and the mc-tree is reconstructed.
**Function *DEC*($\alpha$):**
  If the $\alpha$ is in the last layer of mc-tree, then output *inf*($\alpha$); else            /*Rule 1*/
    if the majority value in the set of {$DEC(\alpha i)|1 \le i \le n$ and vertex $\alpha i$ is a child of vertex $\alpha$} exists,
        then output the majority value; else                                                   /*Rule 2*/
      output the default value $\phi$, where $\phi \in \{0, 1\}$ .                             /*Rule 3*/
```

Fig. 3. The RTAP protocol

5. Execution Example of RTAP

Two examples were used to illustrate the protocol proposed in this study. **Fig. 4** is a 2-cluster VANET used to illustrate the mobility of PEs in VANET. The first example is used to illustrate that when a PE joins VANET's cluster A, RTAP can make all normal PEs to get agreement. The second example will show how the protocol operates when a PE leaves Cluster B of VANET.

When a new PE joins the network, RTAP can still make each normal PE reach an agreement in the Cluster A of the mobile VANET as shown in **Fig. 5**. There are five PEs in Cluster A originally. The worst case when discussing the BA problem is that the Commander is a Byzantine-damaged PE [9]. Therefore, in this example it is assumed that the Commander (PE_2) is a Byzantine-damaged PE, and PE_3 be in dormant-damaged, this means that PE_2 can send arbitrary information to different PEs in Cluster A. In order to solve the BA problem among the normal PEs in Cluster A of the example, RTAP requires $\delta = t+1 = \lfloor (5-1)/3 \rfloor + 1 = 2$ times of information exchange in the *ig-part*.

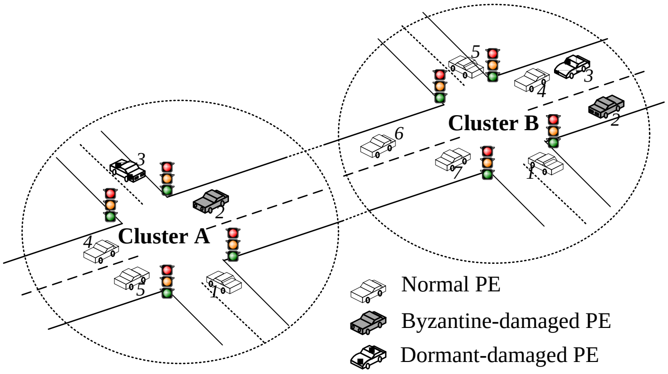


Fig. 4. An example of 2-cluster VANET

After the *ig-part* is executed, a two-layer mc-tree is constructed. During the *am-part*, the majority decision function *DEC* is applied to each normal PE's mc-tree to get the *DEC*(c). The steps of RTAP executed on normal PE_1 in Cluster A when some PEs have joined the network is shown in **Fig. 5**. The steps for other normal PEs in Cluster A are the same as those for PE_1. The information that a damaged PE agrees on is irrelevant.

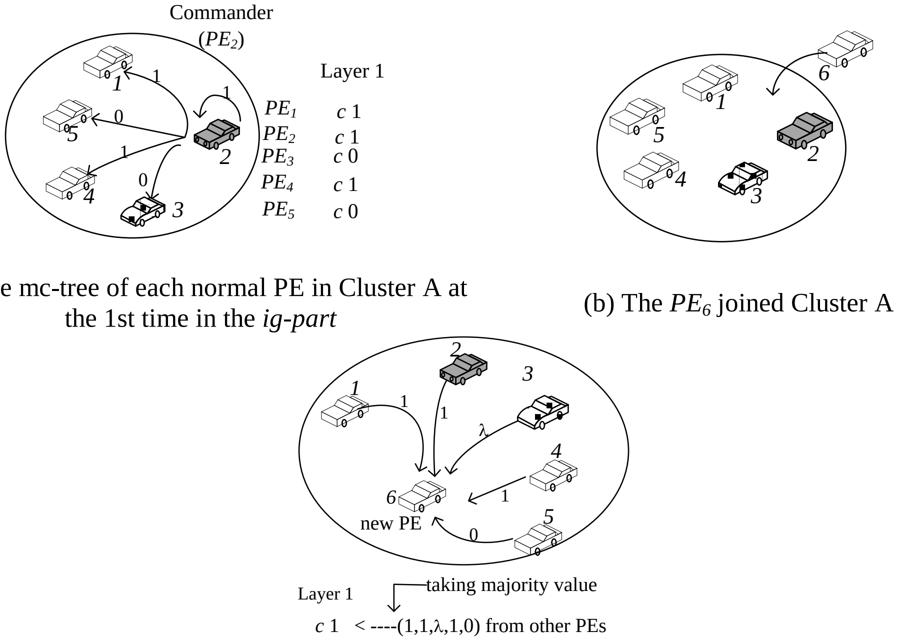
When the *ig-part* begins, the Commander (PE_2) of Cluster A will broadcast its initial data "1" to all PEs of Cluster A in the first time. If the Commander is Byzantine-damaged, it may multicast information to all PEs viciously. If the received information of PE_1, PE_2, PE_3, PE_4, and PE_5 are 1, 1, 0, 1, and 0, respectively, the received information is stored by each PE of

Cluster A into the first layer of its mc-tree. **Fig. 5(a)** only presents the PEs' mc-tree. And, the same processes are performed by each normal PE to get the same agreement data.

When PE_6 has joined Cluster A is shown in **Fig. 5(b)**. Because it does not join the *ig-part* in the first time, it does not receive the information. Now, the information received in the first time by each PE in the original Cluster A must be sent to the new PE, and the majority of the received value will be used as the information received by the new PE_6 in the first time. The progression is shown in **Fig. 5(c)**.

The result of PE_1 in Cluster A to execute the second time of *ig-part* in RTAP is shown in **Fig. 5(d)**. In the second time, all PEs of Cluster A broadcast the first layer information to the other PEs in Cluster A and themselves, and all PEs store the received information to the second layer of their mc-tree. For example, each normal PE of Cluster A broadcasts its *inf(c)* to others in Cluster A and itself. Because, PE_3 in Cluster A is a dormant-damaged PE, then the value λ replaces the information received from PE_3. If PE_1 receives 1 from PE_1, 0 from PE_2, λ from PE_3, 1 from PE_4, 0 from PE_5, and 1 from PE_6, then (1,0,λ,1,0,1) will be stored by PE_1 in the vertices (*c1*, *c2*, *c3*, *c4*, *c5*, *c6*) of its mc-tree. Since the vertices of the mc-tree are not allowed to be repeated, *inf*(*c2*) is deleted. PE_1 stores (1,λ,1,0,1) into the vertices (c*1*, *c3*, *c4*, *c5*, *c6*) of its mc-tree is presented in **Fig. 5(d)**. RTAP requires $\delta = t+1 = \lfloor (n-1)/3 \rfloor + 1 = \lfloor (6-1)/3 \rfloor + 1 = 2$ times, thus, the RTAP is terminated at the end of the second time of *ig-part*.

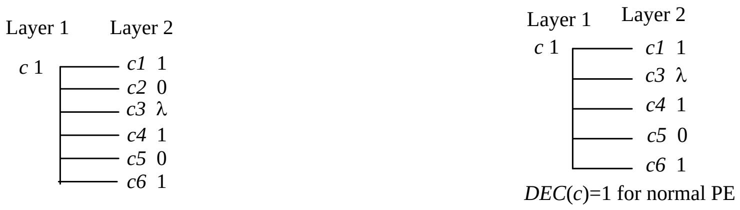
Subsequently, each normal PE of Cluster A executes the *am-part* to the information on its mc-tree in order to get an agreement information. The information are gathered during the *ig-part*, and stored on each normal PE's mc-tree. The mc-tree of PE_1 of Cluster A is presented in **Fig. 5(d)**. In the *am-part*, *DEC* is used to the first layer of the mc-tree. The agreement data (*DEC*(*c*)=1) of each normal PE in Cluster A is shown in **Fig. 5(e)**. Since all normal PEs perform the same procedures, when the number of damaged PEs in Cluster A is less than or equal to $\lfloor (6-1)/3 \rfloor = 1$, the agreement must be achieved.



(a) The mc-tree of each normal PE in Cluster A at the 1st time in the *ig-part*

(b) The PE_6 joined Cluster A

(c) The majority value of the received information obtained by PE_6



(d) The result of PE_1 in Cluster A to execute the 2nd time of *ig-part*

(e) The *am-part*

Fig. 5. An example of five PEs of Cluster A to execute RTAP

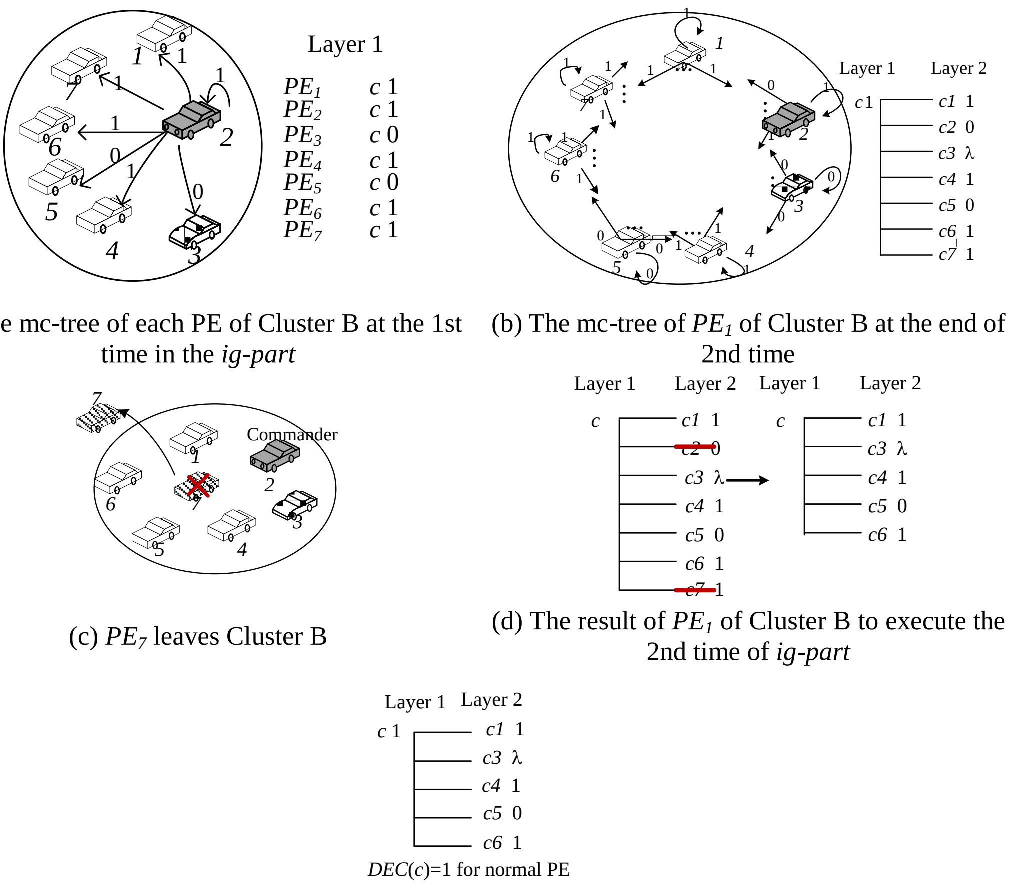
The agreement of normal PEs in the same cluster will be affected when some PEs leave the network. In this case, it is also important to let the normal PEs reach a same information. When some PEs leave Cluster B of **Fig. 4**, RTAP can still make each normal PE reach agreement in cluster B, as shown in **Fig. 6**. There are seven PEs in the original Cluster B, and suppose the Commander (PE_2) is a Byzantine-damaged PE and PE_3 is dormant-damaged. In this case, the required times $\delta = t+1 = \lfloor (7-1)/3 \rfloor + 1 = 3$ in the *ig-part*.

The right side of **Fig. 4** is the original Cluster B. In this case, 3 ($\delta = t+1 = \lfloor (7-1)/3 \rfloor + 1$) times of information exchange are required. Thus, a three-layer mc-tree is constructed by every PE at the *ig-part*. In the *am-part*, *DEC* is used to every normal PE's mc-tree to get an agreement value *DEC*(*c*). When some PEs leave Cluster B, the progress of RTAP performed on normal PE_1 of Cluster B is given in **Fig. 6**.

First, when the *ig-part* is started, the Commander (PE_2) of Cluster B multicasts its initial data "1" to all PEs of Cluster B in the first time. The Commander may arbitrarily broadcast information to all PEs of Cluster B if the Commander is in Byzantine-damaged. **Fig. 6(a)** assumes that the received values of PE_1, PE_2, PE_3, PE_4, PE_5, PE_6, and PE_7 of Cluster B are 1, 1, 0, 1, 0, 1, and 1, respectively. The received information is stored by each PE into the first layer of its mc-tree. **Fig. 6(b)** only presents the normal PEs' mc-tree. In this example, each PE of Cluster B performs the same processes, and all of the normal PEs can reach agreement via the execution of RTAP.

The beginning of the second time of information exchange, every PE of Cluster B broadcasts its first layer data to others and itself, as shown in **Fig. 6(b)**. If some PEs fail (such as PE_2 and PE_3), they will give arbitrary information to other PEs; otherwise, PEs will multicast a same information to others and themselves. Each normal PE stores the received information to the corresponding vertex of the second layer of the mc-tree. When PE_7 leaves Cluster B as presented in **Fig. 6(c)**. Other PEs of Cluster B will not receive any information from PE_7. The information received from PE_7 will be deleted by the PEs that are still in Cluster B, and the mc-tree will also be reconstructed as shown in **Fig. 6(d)**. The PEs of Cluster B will continue to execute the RTAP. Since PE_7 leaves Cluster B, the number of PEs in Cluster B will become 6. At this point, the number of information exchanges required will become $t+1 = \lfloor (6-1)/3 \rfloor + 1 = 2$. Therefore, the *ig-part* needs to perform only two times. There is no duplicate vertex allowed in the mc-tree to avoid the impact of the damaged PE, so *inf*(*c2*) is omitted. The (1,λ,1,0,1) are stored into the vertices (*c1*, *c3*, *c4*, *c5*, *c6*) of the mc-tree by PE_1 of Cluster B, as shown in **Fig. 6(d)**.

After finishing the *ig-part*, each normal PE of Cluster B executes the *am-part* on the information of its mc-tree to get the agreement information. The information are gathered during the *ig-part* and stored on each normal PE's mc-tree as shown in **Fig. 6(d)**. Then, *DEC* is used to apply to the first layer of mc-tree. The agreement value (*DE*C(*c*)=1) of all normal PEs of Cluster B are shown in **Fig. 6(e)**. Then, the agreement is reached. Since all normal PEs of Cluster B perform the same protocol, the agreement will be reached when the number of damaged PEs in Cluster B is less than or equal to $\lfloor (n\text{-}1)/3 \rfloor = \lfloor (6\text{-}1)/3 \rfloor = 1$.



(a) The mc-tree of each PE of Cluster B at the 1st time in the *ig-part*

(b) The mc-tree of PE_1 of Cluster B at the end of 2nd time

(c) PE_7 leaves Cluster B

(d) The result of PE_1 of Cluster B to execute the 2nd time of *ig-part*

(e) The *am-part* of each PE of Cluster B

Fig. 6. An example of seven PEs of Cluster B to execute RTAP

6. The of RTAP

The properness and complexity of RTAP will be proved in this section.

6.1 RTAP Properness Verification

To prove the properness of our protocol [8], vertex α is denoted as collaborative if the information stored in vertex α of the mc-tree of each normal PE is the same. If each normal PE shares the same initial data of the Commander in the root of mc-tree, and if the root of mc-tree in a normal PE is collaborative, then agreement is reached because the root is collaborative. Thus, the (Condition 1) and (Condition 2) can be rewritten as:

(Condition 1’): Root c is collaborative, and
(Condition 2’): $DEC(c) = d_c$ for each normal PE, if the Commander c is normal.

The term *collaborative border* is defined as follows: When every path from root to leaf of mc-tree contains a collaborative vertex, the set of collaborative vertices is formed a collaborative border. In other words, every normal PE has the same information obtained within the collaborative border if it exists within a normal PE's mc-tree. Subsequently, using the majority decision function (*DEC*) to get the majority information of the root in mc-tree, every normal PE can get the same root data because they all use the same input (the same collected information within the collaborative border). The same majority decision results in the same output (the root data).

Since RTAP can solve the BA problem, it is possible to check the properness of RTAP based on "proper vertex" and "real data".

(1) **Proper vertex**: When PE_i is a normal PE, the vertex αi of the mc-tree is a proper vertex. In other words, the proper vertices are used to store information received from the normal PE.

(2) **Real data:** The $inf(\alpha i)$ in the mc-tree of a normal PE is the real data of the vertex αi. That is, the data stored in the proper vertex is named real data.

A PE receives its stored information from a normal PE with a proper vertex, and the normal PE always broadcasts the same information to all PEs. Therefore, the proper vertices of such mc-tree are collaborative. As a result, all the proper vertices of the mc-tree are also collaborative. Again, by the definition of a proper vertex, a collaborative border does exist within the mc-tree. Thus, the root can be proven to be a collaborative vertex (Condition 1' is true) due to the existence of a collaborative border, regardless of the correctness of the Commander. An agreement on the root data can be reached. Next, the Condition 2' needs to be checked. When the Commander is a normal PE, the real data of the root is the initial data of the Commander. That is, each normal PE's root value is the initial data of the Commander if the Commander is normal; therefore, Condition 2' is true when the Commander is normal. Since both Condition 1' and Condition 2' are satisfied, the BA problem can be solved regardless of whether the Commander is normal or damaged.

Lemma 1. The information sent by a dormant-damaged PEs can be detected by the normal receiving PEs.

Proof: If the protocol encodes the transmitting messages by the Manchester code, the dormant-damaged PE can be detected by the receiving PE [21].

Theorem 1. A normal receiving PE can receive information from sending PEs without influence from any damaged PEs between the sending PE and receiving PE in same cluster *i* if $n_i>3f_{im}+f_{id}+f_{ia}$.

Proof: By Lemma 1, we can remove the influence of dormant-damaged PEs between any paired sending PE and receiving PE in each time of information exchange, and we can rule out the influence of Byzantine-damaged PEs between any pairs of PEs in each time of information exchange if $n_i>3f_{im}+f_{id}+f_{ia}$. This is because the normal sending PE sends n_i copies of information to normal receiving PEs. In the worst case, a normal receiving PE receives $n_i-f_{id}+f_{ia}$ information transmitted by the normal sending PE because information from dormant and absent PEs can be detected; in addition, $n_i-f_{id}+f_{ia}>3f_{im}$. Therefore, a normal receiving PE can determine the normal information by taking the majority value.

Lemma 2. A normal receiving PE can detect the dormant-damaged sending PE.

Proof: If the number of λ is greater than or equal to $(n_i-1)-\lfloor(n_i-1)/3\rfloor$ in cluster *i*, then the sending PE has a dormant-damaged. This is because there are at most $\lfloor(n_i-1)/3\rfloor$

Byzantine-damaged PEs in the network, hence there are at most $\lfloor(n_i\text{-}1)/3\rfloor$ non-λ data.

Theorem 2. A normal PE can detect all dormant-damaged PEs in the VANET.

Proof: In the protocol RTAP, there are t_i+1 times of information exchanges in cluster *i*, where $t_i \leq \lfloor(n_i\text{-}1)/3\rfloor$ and $n_i \geq 4$. Thus, there are at least two times of information exchanges during the *ig-part*. Each normal PE can receive the information from the Commander of cluster *i* during the first time of *ig-part*, and receive other PEs' information during the second time of *ig-part*. Therefore, each PE of cluster *i* can receive all other PEs' information in the same cluster after two times of information exchanges. According to Lemma 2, each normal PE can detect all dormant-damaged PEs within the cluster.

Lemma 3. All proper vertices of mc-tree are collaborative.

Proof: There are no repeatable vertices remain in mc-tree. At the level t_i+1 or above, the proper vertex α has at least 2t_i+1 children of which at least t_i+1 children are proper. The real data of these t_i+1 proper vertices is collaborative, and the majority value of vertex α is the same. The proper vertex α is collaborative in the mc-tree if the level of α is less then t_i+1. As a result, all proper vertices of the mc-tree are collaborative.

Lemma 4. A collaborative border exists in the mc-tree of the normal PE.

Proof: There are t_i+1 vertices along each root-to-leaf path of an mc-tree in which the first layer is labeled by the Commander name, and the others are labeled by a sequence of PE names. Since at most t_i PEs in cluster *i* can be in damaged, there is at least one vertex that is proper along each root-to-leaf path of the mc-tree. Using Lemma 3, the proper vertex is collaborative, and the collaborative border exists in each normal PE's mc-tree.

Lemma 5. Let α be a vertex, α is collaborative if there is a collaborative border in the subtree rooted at α.

Proof: If the height of α is 0 and the collaborative border of α exists, then α is collaborative. If the height of α is δ and the children of α are all consistent. By induction, the vertex α is collaborative for the children of height at δ-1.

Corollary 1. The root is collaborative if a collaborative border exists in the mc-tree.

Theorem 3. The root of a normal PE's mc-tree is collaborative.

Proof: By Lemma 3, 4, 5, and Corollary 1, the theorem is proven.

Theorem 4. Protocol RTAP solves the BA problem in a VANET.

Proof: To prove the theorem, it must be shown that the RTAP meets Condition 1' and Condition 2'.

(Condition 1'): First layer is collaborativer. By Theorem 3, Condition 1' is satisfied.

(Condition 2'): *DEC*(*c*) = d_c for each normal PE, if the Commander is normal.

If the Commander is normal, then it broadcasts the same initial data d_c to all PEs. The data of proper vertices for all normal PEs' mc-tree is d_c. Thus, each proper vertex of the mc-tree is collaborative (by Lemma 1), and its data is d_c. Since the Commander is normal, the root of the mc-tree is also a proper vertex by Lemma 5. By Theorem 3, this root is collaborative. The computed value *DEC*(*c*) = d_c is stored in the root for all normal PEs. Thus, Condition 2' is satisfied.

6.2 RTAP Complexity Verification

The complexity of RTAP will be verified by two factors: 1) the number of information exchanges required, and 2) the total number of damaged PEs allowed. Theorems 5, 6, 7, and 8 are used to prove that RTAP has been the optimal solution. The number of information exchanges required for RTAP is proved in Theorem 5. Theorems 6 and 7 have proved that

RTAP solves the BA problem by using a expectable amount of information exchange and allowing the maximum number of damaged PEs, respectively. In Theorem 8, the fault tolerance of RTAP is proved. Therefore, the optimality of RTAP will be obtained

Theorem 5: RTAP requires δ (t_i+1) information exchanges in cluster *i*, and can tolerate f_{im} Byzantine-damaged PEs, f_{id} dormant-damaged PEs, and f_{ia} absent PEs, where $t_i \leq \lfloor(n_i-1)/3\rfloor$.

Proof: 1) Some PEs may join the cluster *i* or leave the cluster *i*, and the total number of PEs may be changed at any time. The time required must be re-evaluated at any time. Although the amount of time required may vary at any time, it is always predictable. δ is used to represent the time required for RTAP. And, according to the result of Fischer and Lynch [8], $\delta = t_i+1 = \lfloor(n_i-1)/3\rfloor+1$.

2) By Theorem 1, RTAP can tolerate f_{im} Byzantine-damaged PEs, f_{id} dormant-damaged PEs, and f_{ia} absent PEs at any time, where $n_i > 3f_{im}+f_{id}+f_{ia}$. When the total number of damaged PEs in cluster *i* exceeds the upper limit that can be tolerated, normal PEs cannot be agreed at this time. Therefore, the theorem can be proved.

Theorem 6: RTAP can solve BA problem in cluster *i* with a minimum expected number of information exchanges.

Proof: *t*+1 is the minimum number of times needed to get enough information to reach the BA has been proven by Dolev and Reischuk [9]. Thus, the number of required times of information exchanges in RTAP is $\delta = t_i+1$ times at any time in cluster *i* and this number is the minimum.

Theorem 7: The number of allowable f_{im} Byzantine-damaged PEs, f_{id} dormant-damaged PEs, and f_{ia} absent PEs is the maximum, where $n_i>3f_{im}+f_{id}+f_{ia}$ in RTAP of cluster *i*.

Proof: When the total number of damaged PEs exceeds the allowable limit, then the normal PE does not have enough information to eliminate the effects of damaged PEs. The $n_i>3f_{im}+f_{id}+f_{ia}$ must be conformed to make each normal PE obtain an agreement result. Thus, at least $n_i-\lfloor(n_i-1-f_{id}-f_{ia})/3\rfloor-f_{id}-f_{ia}$ PEs in cluster *i* are normal and have the same agreement data. In the worst case, if $n_i-\lfloor(n_i-1-f_{id}-f_{ia})/3\rfloor-f_{id}-f_{ia}$ copies of the same information are received by the joined PE, which is larger than $\lfloor(n_i-1-f_{id}-f_{ia})/3\rfloor$, so the joined PE can determine the agreement value by *DEC*.

Theorem 8: Using RTAP, the total number of allowable damaged PEs of VANET is optimal, and the number of information exchanges is minimal.

Proof: In a *C*-cluster based VANET, the PEs in each cluster execute RTAP parallelly, where *C* is the number of clusters in the VANET. By Theorem 7, the number of allowable damaged PEs in cluster *i* is $f_{im}+f_{id}+f_{ia}$. Therefore, in this *C*-cluster based VANET, the total number of allowable damaged PEs of VANET is $\sum_{i=1}^{C}(f_{im}+f_{id}+f_{ia})$, and it is the maximum. By Theorem 6, the total number of information exchanges in cluster *i* is t_i+1. Therefore, the total number of information exchanges in a *C*-cluster based VANET is the largest (t_i+1) for all cluster *i* ($\mathrm{MAX}_{i=1}^{C}(t_i+1)$), and it is necessary.

7. Conclusions

Since PEs in VANET have mobile characteristics, these PEs can join or leave the topology at any time. In addition, the topology may be unstable because some PEs in the network may damage. In recent years, the network topology has moved toward mobility [4]. However, in the past, the relevant agreement protocols on BA [7,8] could not solve the BA problem in VANET, and no BA protocol was designed for the characteristics of VANET [7,8]. Therefore, the BA problem in the VANET with the hybrid damaged PE is re-examined, and the proposed protocol can tolerate the most destructive type of damaged PEs. **Table 1** shows a comparison of the proposed protocols with the previous related studies. These research results achieved BA on different topologies, including Broadcasting Network (BCN), Fully Connected Network (FCN), Multicasting Network (MCN), Wireless Sensor Network (WSN), Virtual Subnet Network (VSN), and Cloud Computing environment (CC). The RTAP proposed in this study can achieve the agreement value of all normal PEs in VANET. The RTAP uses the least amount of information exchange to eliminate the impact of hybrid damaged PE, and can tolerate the maximum number of damaged PEs at any time. That is, RTAP has the following features:

- The BA problem in a cluster-based VANET is solved by RTAP.
- The newly joined PE is allowed to reach the agreement value by RTAP.
- Only using the minimum number of information exchanges, BA problem can be solved by RTAP.
- The reliability of VANET is improved by allowing Byzantine-damaged PEs, dormant-damaged PEs, and absent PEs exist simultaneously.

Table 1. The comparisons of the proposed protocols with the previous related studies

Topology Damage types Protocols	BCN		FCN		MCN		WSN		VSN		CC		VANET	
	B	*H*	*B*	*H*	*B*	*H*	*B*	*H*	*B*	*H*	*B*	*H*	*B*	*H*
RB [27]	*													
OM [7]			*											
PSL [8]			*											
Algorithms using authentication [9]			*											
MS [10]			*	*										
EMAP [11]					*									
OAP [12]							*							
OGBA [13]							*	*						
MVSAP [14]									*					
TMCC [15]											*			
DFP [16]											*	*		
RTAP													*	*

B: Byzantine-damaged, H: Hybrid-damaged

In the future, the proposed protocol will be simulated and the results will be compared to other past agreement protocols. In addition, only considering the PE damaged in the BA problem is insufficient for high reliable VANET. In the practical application of VANET, the transmission medium in the network might also be crashed, omission, or Byzantine damaged. Therefore, the proposed protocol will be further extended to solve the situation where the transmission media and PEs are damaged at the same time in VANET.

References

[1] Kashif Naseer Qureshi and Abdul Hanan Abdullah, “A survey on intelligent transportation systems,” *Middle-East Journal of Scientific Research*, vol. 15, no. 5, pp. 629-642, May, 2013.

[2] Yi-Ling Hsieh and Kuochen Wang, “Dynamic overlay multicast for live multimedia streaming in urban VANETs,” *Computer Networks*, vol. 56, no. 16, pp. 3609-3628, November, 2012. Article (CrossRef Link)

[3] Priyanka Rawat, Kamal Deep Singh, Hakima Chaouchi, and Jean Marie Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *The Journal of supercomputing*, vol. 68, no. 1, pp. 1-48, April, 2014. Article (CrossRef Link)

[4] Ankita Dixit, Shweta Singh, and Kushal Gupta, "Comparative study of P-AODC and improved AODV in VANET," *International Journal of Advanced Research in Computer Science and Management Studies*, vol. 3, no. 1, pp. 270-275, January, 2015.

[5] Mohammad Wazid, Ashok Kumar Das, Vanga Odelu, Neeraj Kumar, Mauro Conti, and Minho Jo, "Design of secure user authenticated key management protocol for generic IoT networks," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 269-282, February, 2018. Article (CrossRef Link)

[6] Marshall Pease, Robert Shostak, and Leslie Lamport, "Reaching agreement in the presence of faults," *Journal of the ACM*, vol. 27, no. 2, pp. 228-234, April, 1980. Article (CrossRef Link)

[7] Leslie Lamport, Robert Shostak, and Marshall Pease, "The Byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 382-401, July, 1982. Article (CrossRef Link)

[8] Fischer, Michael J. and Nancy A. Lynch, "A lower bound for the time to assure interactive consistency," *Information Processing Letters*, vol. 14, no. 4, pp. 183-186, September, 1982. Article (CrossRef Link)

[9] Danny Dolev and Rüdiger Reischuk, "Bounds on information exchange for Byzantine agreement," *Journal of the ACM*, vol. 32, no. 1, pp. 191-204, January, 1985. Article (CrossRef Link)

[10] Hin-Sing Siu, Yeh-Hao Chin, and Wei-Pang Yang, "A note on consensus on dual failure modes," *IEEE Transactions on Parallel and Distributed Systems*, vol. 7, no. 3, pp. 225-230, March, 1996. Article (CrossRef Link)

[11] Shu-Ching Wang, Kuo-Qin Yan, and Chien-Fu Cheng, "Byzantine agreement under unreliable multicasting network," *Information Technology Journal*, vol. 2, no. 2, pp. 104-115, April-June, 2003. Article (CrossRef Link)

[12] Kuo-Qin Yan, Shu-Ching Wang, Chin-Shan Peng, and Shun-Sheng Wang, "Optimal Byzantine agreement protocol for cluster-based wireless sensor networks," *ScienceAsia Journal*, vol. 40S, pp. 8-15, February, 2014. Article (CrossRef Link)

[13] Shu-Ching Wang, Kuo-Qin Yan, Chin-Ling Ho and Shun-Sheng Wang, "The optimal generalized Byzantine agreement in cluster-based wireless sensor networks," *Computer Standards & Interfaces*, vol. 36, no. 5, pp. 821-830, September, 2014. Article (CrossRef Link)

[14] Shu-Ching Wang, Kuo-Qin Yan, and Guang-Yan Zheng, "Efficient Byzantine agreement in a virtual subnet network," in *Proc. of the Second International Conference on Availability, Reliability and Security*, pp. 812-818, April 10-13, 2007**.** Article (CrossRef Link)

[15] Shu-Ching Wang, Shun-Sheng Wang and Kuo-Qin Yan, "New anatomy of trustworthy mobile cloud computing," *Information Technology and Control*, vol. 45, no. 4, pp. 349-357, October-December, 2016. Article (CrossRef Link)

[16] Shun-Sheng Wang and Shu-Ching Wang, "The consensus problem with dual failure PEs in a cloud computing environment," *Information Sciences*, vol. 279, pp. 213-228, September, 2014. Article (CrossRef Link)

[17] Tushar Deepak Chandra and Sam Toueg, "Unreliable failure detectors for reliable distributed systems," *Journal of the ACM*, vol. 43, no. 2, pp. 225-267, March, 1996. Article (CrossRef Link)

[18] Najme Mansouri, Gholam Hosein Dastghaibyfard', and Ehsan Mansouri, "Combination of data replication and scheduling algorithm for improving data availability in Data Grids," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 711-722, March, 2013. Article (CrossRef Link)

[19] Donald Hale Jr and Robert E. Ployhart, "A two-part longitudinal model of a turnover event: Disruption, recovery rates, and moderators of collective performance," *Academy of Management Journal*, vol. 59, no. 3, pp. 906-929, February, 2016. Article (CrossRef Link)

[20] Davide Scaramuzza, Michael C. Achtelik, Lefteris Doitsidis, Fraundorfer Friedrich, Elias Kosmatopoulos, Agostino Martinelli, and Daniel Gurdan, "Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in GPS-denied environments," *IEEE Robotics & Automation Magazine*, vol. 21, no. 3, pp. 26-40, August, 2014. Article (CrossRef Link)

[21] Alin-Mihai Căilean, Barthélemy Cagneau, Luc Chassagne, Mihai Dimian, and Valentin Popa, "Novel receiver sensor for visible light communications in automotive applications," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4632-4639, April, 2015. Article (CrossRef Link)

[22] Mohamed Nidhal Mejri, Jalel Ben-Othman, and Mohamed Hamdi Mejri, "Survey on VANET security challenges and possible cryptographic solutions," *Vehicular Communications*, vol. 1, no. 2, pp. 53-66, April, 2014. Article (CrossRef Link)

[23] Behnam Hassanabadi, Christine Shea, Le Zhang, and Shahrokh Valaee, "Clustering in vehicular ad hoc networks using affinity propagation," *Ad Hoc Networks*, vol. 13, pp. 535-548, February, 2014. Article (CrossRef Link)

[24] Yu-Chee Tseng, Sze-Yao Ni, Yuh-Shyan Chen, and Jang-Ping Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wireless networks*, vol. 8, no. 2-3, pp. 153-167, March, 2002. Article (CrossRef Link)

[25] Wai Chen and Shengwei Cai, "Ad hoc peer-to-peer network architecture for vehicle safety communications," *IEEE Communications Magazine*, vol. 43, no. 4, pp. 100-107, April, 2005. Article (CrossRef Link)

[26] Amotz Bar-Noy, Danny Dolev, Cynthi Dwork, and H. Raymond Strong, "Shifting gears: Changing algorithms on the fly to expedite Byzantine agreement," *Information and Computation*, vol. 97, no. 2, pp. 205-233, April, 1992. Article (CrossRef Link)

[27] Oezalp Babaoglu and Drummond Rogério, "Streets of Byzantium: Network architectures for fast reliable broadcasts," *IEEE Transactions on Software Engineering*, vol. SE-11, no. 6, pp. 546-554, June, 1985. Article (CrossRef Link)

Shu-Ching Wang received her Ph.D. in Information Engineering from National Chiao-Tung University, Taiwan. Currently, she is a Professor at the Department of Information Management, Chaoyang University of Technology, Taiwan. Her current research interests include distributed data processing, Big data, Internet of Things, VANET, Fog computing, and Cloud computing.

Ya-Jung Lin is a Ph.D. student of the Department of Information Management, Chaoyang University of Technology, Taiwan. Her current research interests include distributed data processing, grid computing, Internet of Things, VANET, and Cloud computing.

Kuo-Qin Yan received his Ph.D. in Computer Sciences from National Tsing-Hua University, Taiwan. Currently, he is a Professor at the Department of Business Administration, Chaoyang University of Technology, Taiwan. His current research interests include distributed fault tolerant computing, mobile computing, Internet of Things, VANET, and Cloud computing.