Vol.2010-MBL-56 No.9
Vol.2010-ITS-43 No.9
2010/11/11

Practical Byzantine Fault Tolerance Strategy in Wireless Networks

趙　蘊龍[†, ††]　　瀬崎　薫[††]

In order to improve the reliability of the wireless networks, a new concept of Byzantine Cell is proposed, and a practical Byzantine fault tolerance wireless network framework is also constructed. And then a Byzantine fault tolerance algorithm is given to enhance the existing wireless routing protocols, and a case study is also given to show the Byzantine fault tolerance algorithm applying method, which is the BAODV protocol developed from AODV. The simulation results are given to show that the enhanced routing protocol based on the proposed Byzantine fault tolerance strategy can significantly improve reliability of the wireless networks.

Practical Byzantine Fault Tolerance Strategy in Wireless Networks

YUNLONG ZHAO[†, ††]　KAORU SEZAKI[††]

In order to improve the reliability of the wireless networks, a new concept of Byzantine cell is proposed, and a practical Byzantine fault tolerance wireless network framework is also constructed. And then a Byzantine fault tolerance algorithm is given to enhance the existing wireless routing protocols, and a case study is also given to show the Byzantine fault tolerance algorithm applying method, which is the BAODV protocol developed from AODV. The simulation results are given to show that the enhanced routing protocol based on the proposed Byzantine fault tolerance strategy can significantly improve reliability of the wireless networks.

1. Introduction

Wireless networks are becoming more and more popular in our society as a necessary complement of Internet backbone, which have already been utilized by the customers in some particular situation, such as WLAN by use of WiFi technology. While in order to expand the broad application of wireless networks, the high reliability and security of it must be guaranteed, so it is definitely necessary to do such research work as the efficient strategy to improve the reliability of wireless networks.

The topology of wireless networks has the inherence of high instability and weak reliability, so one of the efficient methods is just to try to find out all the stable and reliable sub-parts of the whole wireless networks and to fully utilize them to improve the network reliability. Byzantine fault tolerance technology is a sub-field of error tolerance research inspired by the Byzantine Generals' Problem. The object of Byzantine fault tolerance is to be able to defend against Byzantine failures, in which components of a system fail in arbitrary with the use of Byzantine fault tolerance to improve wireless networks reliability[1]. Byzantine fault tolerance had been successfully applied in the computer architecture design to enhance the computer reliability, so this paper will propose a strategy to tolerate Byzantine failures in the wireless networks.

2. Basic Idea

Because of the dynamic topology change, instable wireless link connectivity and mesh topology in wireless networks, wireless networks have the inherence of high instability and weak reliability, so wireless networks reliability has always been a key issue and hot topic[2]. According to the Byzantine General Theory, if there are 3M+1 generals, solution allows up to M traitors, which means that a sub-system with 3M+1 components can cope with at most M components failure. For the special simple situation of making M=1 in wireless networks, if there exist a sub-net with totally 4 nodes connecting with each other, such sub-net can tolerate 1 node failure, such as sending the incorrect package or holding the wrong routing tables[3]. So this

* [†]ハルビン工程大学
Harbin Engineering University
[††]東京大学生
The University of Tokyo

paper proposes a practical Byzantine fault tolerance strategy in wireless networks, which will explore all the sub-nets with the ability to tolerate Byzantine fault in the whole networks.

2.1 Some Definitions and Notations

- Byzantine Cell (U_C).

If there are 4 nodes among all nodes in wireless networks, which can communicate with each other, the sub-net with the 4 nodes is called Byzantine Cell, shown in Figure 1.

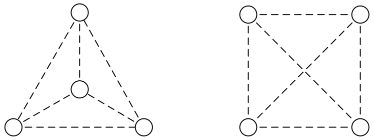


Figure 1 Byzantine Cell.

- Neighbor Node (N1Node).

If node A connects with node B, node A is called Neighbor Node of node B, on the contrary node B is the Neighbor Node of node A.

- Two Hops Neighbor Node (N2Node).

If node A connects with node B, and node B connects with node C, which means that node A is 2 hops away from node C, node A is called Two Hops Neighbor Node of node C, on the contrary node C is Two Hops Neighbor Node of node A.

- Two Hops Neighbor Form (N2form).

The data structure of N2form is shown in Table 1, which is used to store the address of N1Node and N2Node of the current node.

Table 1 Two Hops Neighbor Form.

Address of N1Node	Address of N2Node
IP address	IP address

- Byzantine Cell Form (Bform).

Byzantine Cell Form is shown in Table 2, which is used to store the Byzantine Cell name and each node address in the Cell.

Table 2 Byzantine Cell Form.

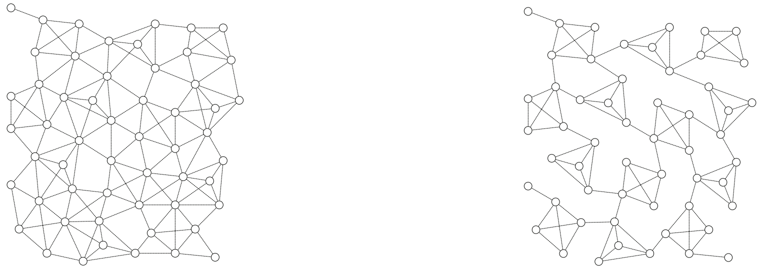
Node1 Address	Node2 Address	Node3 Address	Node4 Address	Byzantine Cell Name
IP address1	IP address2	IP address3	IP address4	Cell Name

- Byzantine Time (T_B).

Byzantine Time is the maximum time period when the Byzantine fault tolerance algorithm is carried out.

2.2 Simple Example

It is well illustrated by a simple example, shown in figure 2.



(a) Wireless Networks Topology

(b) Reconstructing Topology with Byzantine Cells

Figure 2 Byzantine Cells Construction.

The original wireless network is randomly constructed like figure 2.a, which is a mesh topology. In order to fully exploring its Byzantine fault tolerance ability, firstly all available Byzantine Cells should be found, shown in figure 2.b. Of course, only the Byzantine Cell has the ability to tolerate Byzantine fault, and the rest parts do not. Then some key information, such as routing table or some others you considering, should be selected and stored in each node belonging to the same Byzantine Cell, so that the key information has a backup in each node. In case of one node of the Byzantine Cell sending the wrong information due to its failure, the other 3 nodes can correct it with their own backup information.

3. Byzantine Fault Tolerance Strategy

In order to find out all the available Byzantine Cell in wireless networks and fully utilize them to tolerate the Byzantine failure, a practical Byzantine Fault Tolerance strategy is proposed.

Because the procedure of detecting and constructing Byzantine Cells accompanies the route discovery process, some modification of the RREQ format should be made to record the N2Node information. So the neighbor node address is added into the RREQ. The Byzantine Cell detection and construction process is a distributed process and should be implemented by each node. The detailed algorithm is described as follows.

a) Start a timer T.

b) The current node should retrieve the sending node IP address as its N1Node and the sending node's neighbor node IP address as its N2Node from the receiving RREQ package.

c) The current node should look up its N2form to find if the receiving information of N1Node and N2Node have already been stored in the N2form. If not, it will add them into N2form, else discard them.

d) If the timer T is less than $T_{B,}$ go back to step b), else continue to step e).

e) Till now, the N2form has been constructed completely.

f) The current node can check N2form to find out its 3 different neighbor nodes which are also connected with each other. So one Byzantine Cell is established including the 3 different neighbor nodes and itself.

g) The current node look up Bform to check if the Byzantine Cell is already stored in Bform. If so, go back to step f), else continue to step h).

h) The current node can name the Byzantine Cell with some naming strategy, such as jointing the 4 nodes' IP address by ascending order to be the Cell name, which can guarantee the name's uniqueness. Then, it stores the Byzantine Cell into Bform.

i) The current node forwards the new Byzantine Cell information to the other 3 nodes.

j) If the N2form has not been checked entirely, go back to step f), else continue to step k).

k) When the current node receives the Byzantine Cell information from its neighbor node, it should check if such information has been stored in Bform. If so, it just discard it, else add it into Bform. End.

4. Simulation and Result

The Byzantine Fault Tolerance Strategy can be used by any routing protocol to improve the networks reliability. To evaluate its performance, we select the classic AODV[4] protocol for a case study to illustrate how to use the proposed strategy. The modified protocol is named as BAODV (Byzantine-based AODV).

4.1 Simulation Setup

The simulation experiments were carried out in the NS2 network simulator. Simulation scenario was built up of 100 nodes, MAC layer used IEEE802.11b DCF protocol, the radius of node coverage area is 150m, and the channel capacity holds at 2Mb/s. We placed nodes in a $1000 \times 1000\text{m}^2$ square. In addition, every gateway can cover a router at least and routers connected by wireless channel. Nodes send UDP packet with a constant bit rate (CBR), the size of packet is 512 byte, and a communication node select the other node randomly. The backup information is set as the node IP, and the backup time is set as the moment when building Byzantine Cell is finished. Each node selects a random time to send packet in the first 10 seconds of simulation. We set each simulated scene time as 1000s, and we record after the first 50s so that the network can reach to a steady state.

4.2 Result

Firstly, we simulate the AODV protocol and BAODV protocol. To examine the fault tolerance ability when node failure, we select the number of abnormal nodes were 5, 10, 15, 20, 25. Nodes selected randomly when the simulation started 100s, and changed the selected node IP randomly. We simulated each scene 10 times, and the experiment data gain from the mean of 10 simulation. The metrics of packet delivery ratio, routing overhead and delay of routing acquisition were selected to compare performance between AODV and BAODV.

1) Packet Delivery Ratio

According to the figure 3, as the number of abnormal nodes increasing in network, the packet delivery ratio keeps falling. But BAODV protocol is always better than AODV protocol in the packet delivery ratio.

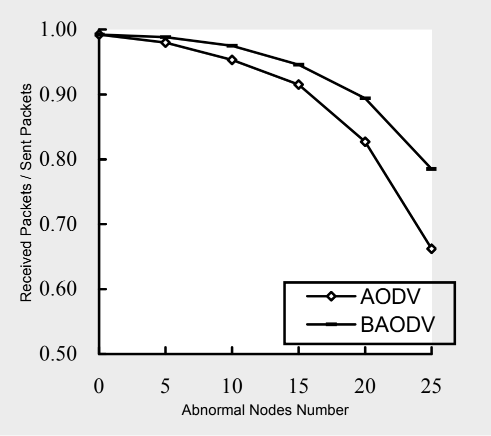


Figure 3 Impact of Packet Delivery Ratio by Nodes Abnormal Comparison

2) Routing Overhead

As figure 4 shows, as the number of abnormal nodes increasing in network, the routing overhead showed an upward trend. But BAODV protocol has the fault-tolerant ability of Byzantine, its capability of route maintenance capability is better than AODV protocol, So as a result of the emergence of abnormal nodes in the routing overhead is always less than the AODV protocol.

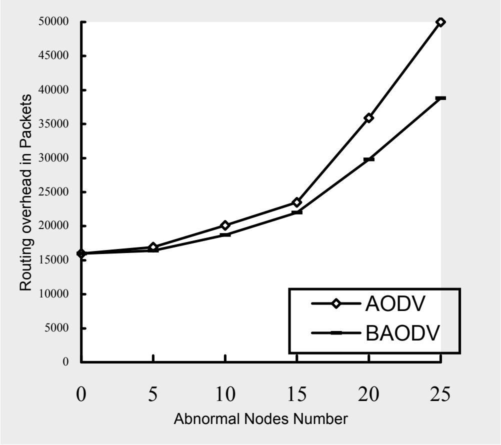


Figure 4 Impact of Routing Overhead by Nodes Abnormal Comparison

3) Delay of Routing Acquisition

While the increasing of invalid nodes in the network, the routing delay is increasing too (see figure 5). But the routing delay of BAODV protocol is lower than AODV protocol because of node invalid.

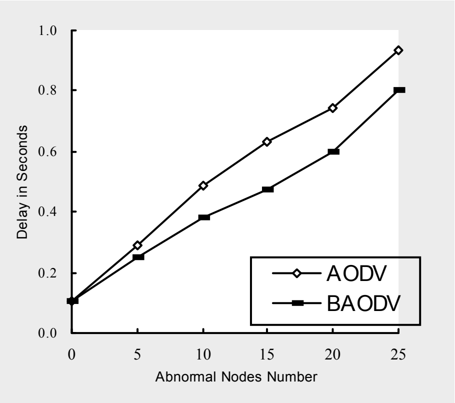


Figure 5 Impact of Routing Delay by Nodes Abnormal Comparison

Through the comparison of simulation data, it can show the conclusions:

At the normal link condition, the improved BAODV protocol is worse than AODV protocol at packet delivery ratio, routing overhead and routing delay. Executing Byzantine algorithm can impact index of routing protocol, but the impact is weak.

When the part nodes become invalid, BAODV protocol is better than AODV protocol at packet delivery ratio, routing overhead and routing delay. The improved BAODV protocol is better than AODV protocol at reliability of data transmission.

5. Conclusion

Wireless networks reliability is one of the most important preconditions to be widely applied by the customers. This paper firstly proposes a new concept of Byzantine Cell, and then gives a practical Byzantine fault tolerance strategy based on Byzantine Cell. At last, AODV protocol is modified by the proposed strategy, named BAODV, and the simulation results show that BAODV protocol is better than AODV protocol at reliability of data transmission, so the efficiency of the proposed Byzantine fault tolerance strategy is proved

Acknowledgment

This research was supported by the National Natural Science Foundation of China under Grant No. 61003235, the Natural Science Foundation of Heilongjiang Province of China under Grant No. F200902, and the Ministry of Education, Culture, Sports, Science and Technology (MEXT).

References

1) Lamport L, Shostak R, Pease M. The byzantine generals problem. ACM Transactions on Programming Languages and Systems, Vol. 4, No. 3, pp.382-401 (1982)
2) Perlman R. Routing with byzantine robustness. United States: Sun Microsystems, Inc. Technical Reports, SERIES13103, (2005)
3) Fedotova N, Veltri L. Byzantine generals problem in the light of P2P computing//International Workshop on Ubiquitous Access Control, San Jose, Italy: University of Parma, (2006)
4) S. J. Lee, E. M. Belding Royer, C. E. Perkins. Scalability Study of the Ad Hoc On-Demand Distance-Vector Routing Protocol. International Journal of Network Management, Vol. 13, No. 2, (2003)