An Improved AODV Routing Security Algorithm Based on Blockchain in Ad Hoc Network

Conglin Ran\textsuperscript{1,4}, Shuailing Yan\textsuperscript{2,4,*}, Liang Huang\textsuperscript{3,4}, Lei Zhang\textsuperscript{1}

\textsuperscript{1} Department of Information Technology Center, Jiujiang University, Jiujiang, 332005, P.R.China
\textsuperscript{2} Department of Mathematics and Computer Science, Hengshui University, Hengshui, 053000, P. R. China
\textsuperscript{3} School of Mathematics & Computer Science, Shangrao Normal University, Shangrao, 334001, P.R. China
\textsuperscript{4} Department of Computer and Software Engineering, Wonkwang University, Iksan, 54538, Republic of Korea
\textsuperscript{*} Correspondence: yanshuailing@163.com

Abstract: Ad Hoc network is a special network with no center and dynamic topology changes. Due to the free mobility of nodes, routing security has always been a bottleneck problem that plagues network development. Therefore, this paper proposes a multi-path QoS (Quality of Service) routing security algorithm based on blockchain by improving the traditional AODV (Ad hoc On-Demand Distance Vector) protocol (AODV-MQS). The algorithm first establishes a blockchain network and saves the state of all nodes by making the intermediate nodes on the chain. Then the smart contract in the blockchain is set to filter out the nodes that meet the QoS constraints. Finally, two largest unrelated communication paths are found in the blockchain network through smart contract, one of which is the main path and the other is the standby path. Simulation experiments show that this algorithm has better performance than other algorithms in an unsafe environment.

Keywords: Ad Hoc network, AODV protocol, blockchain, routing security, QoS; smart contract

1 Introduction

In Ad Hoc networks, due to the exposure of the network, the nodes are faced with some problems not only physical security threats but also a variety of attacks [1-4]. At the same time, because of the characteristics of the network, such as inherent defects of the wireless channel itself, the dynamic changes of the topology, the lack of a centralized control center to protect node information and so on, the routing security issues of mobile Ad Hoc networks need to be resolved urgently. At present, there have been many attack methods [5] [6] and defense schemes [7] [8] for the protocol vulnerabilities and routing security of Ad Hoc networks. A node of the network as an individual may be subjected to other malicious attacks, the attack points are not limited to a certain protocol layer, and even the attack can harm some layers of the protocol stack at the same time. From the perspective of attack mechanism, the attacks can generally be divided into two categories: routing mechanism damage and routing resource consumption [9]. From the perspective of attack methods, the attacks can be classified into four categories: interception, tampering, interruption, and forgery [10].

At present, many scholars focus their research on the handling of attacks ignoring QoS (Quality
of Service), and the actual Ad Hoc network environment often take notice of service quality seriously. Simulation experiments show that this algorithm has obvious advantages over other algorithms in terms of end-to-end time delay, packet delivery rate and control overhead, it can avoid malicious attacks and improve the routing security of Ad Hoc networks.

The rest of this paper is organized as follows. Section 2 discusses the related work of different types of attacks in Ad Hoc network security. Section 3 designs an improved AODV Multi-path QoS routing security algorithm. Section 4 shows the simulation and analysis of the data. Summary and future research directions are concluded in section 5.

2 Related work

At present, many scholars have proposed many solutions to solve black hole attacks in Ad Hoc networks. For example, S. Gupta and others in literature [1] proposed a method to detect black hole nodes by modifying routing requests and routing reply control packets based on the AODV protocol. In this method, the data hop count, destination node, sequence number and other information in the routing table are first counted, and then the trust threshold is set, and the scheme of selecting the optimal route according to the path length in the AODV protocol is converted into relying on the trust threshold. The strategy is carried out to ensure that the packet transmission node is the most likely legal node. The most obvious shortcoming of this strategy lies in the choice of trust threshold. Once the trust value is set unreasonably, it will cause great misjudgment and form a false warning, thus affecting the performance of the entire network. In literature [12] Hongwei Deng et al. proposed a method for detecting the next hop node of the intermediate node, that is, when the source node receives the RREP sent by the intermediate node, it sends a verification packet to the next hop node of the node to authenticate. After receiving the verification information, its next-hop node replies to the source node. After the source node confirms that the intermediate node is a normal node, it sends data through the route. Otherwise, the intermediate node is regarded as a black hole node.

In literature [13] Huang et al. proposed to detect packet loss nodes based on one-way hash chain and one-time hash tag commitment. This detection method mainly relies on the form of forwarding redundant data packets and shared secret keys between nodes. In addition, the source node also needs to predict the sending status of the next data packet. Therefore, the main disadvantage of this method is that the control overhead is large, especially when the network scale is large, and the smooth implementation of the shared key cannot be guaranteed. In literature [14] Papadimitriou et al. used path redundancy and threshold secret sharing technology to achieve the secure transmission of data. The scheme uses an end-to-end authentication method. The packet loss path discovery process does not require intermediate nodes to participate, but it cannot detect packet loss nodes.

In Literature [15] Sunkara Narayanan S et al. proposed a high-level mechanism against wormhole attacks in the MANET network. This mechanism mainly uses the service quality of the network to detect the attacking node, and at the same time judges whether the attack is active or passive according to the round-trip time of the data packet at the node. This type of method can identify wormhole attacks better, but it does not consider the risk of data being intercepted. In literature [16] Aswale A.B. et al. introduced advanced encryption algorithms to the detection of nodes, and adopted channel security detection to avoid the traditional secure communication at the cost of energy and extend the life of the network. However, the algorithm complexity of this method
is too high, which leads to an excessively high control overhead and is not suitable for high-speed mobile Ad Hoc networks with limited network bandwidth.

3 Methods

3.1 Definition of QoS related parameters

1) Path bandwidth

Path bandwidth refers to the minimum bandwidth of all adjacent nodes in the whole routing path, which is represented by $\text{Bandwidth}$.

$$\text{Bandwidth}(s, d) = \min \{ B(i, j), i = 1,2,3L : j = 1,2,3L \}$$

where $\text{Bandwidth}(s, d)$ is the routing path bandwidth with $s$ as the source node and $d$ as the destination node, and $B(i, j)$ is the communication link bandwidth between the intermediate node $i$ and node $j$ that can communicate with each other.

2) Time delay

The time delay of path refers to the time required for packets to arrive at the destination node from the source node, which is represented by $T$. Assuming that all nodes in the network have the same processing capacity and channel bandwidth, and the wireless channel used is symmetric, the size of the route request probe packet, the response packets of the nodes, and the data packets are equal. This chapter divides the data packets transmission time in the network into two parts, namely the processing time $Pr_o$ of the data packet and the transmission time $Tra$ of the data packets in the communication. The processing time $Pr_o$ of the data packet is divided into the waiting processing time $T_w$ of the data packet in the queue and the real actual processing time $T_e$ of the data packet.

$$T = \sum \text{Pr}_o + \sum \text{Tra}_{m-1}, k \geq 2, k \in N, m = k - 1$$

$$\text{Pr}_o = T_w + T_e$$

where $\text{Pr}_o$ represents the time when the node $k$ forwards the information packets in the data transmission, and $\text{Tra}_{m-1}$ represents the time taken by the data packet to pass between the intermediate node $m$ and node $m-1$. Assume that the number of nodes in the network is bigger than 1, so $k \geq 2$. The maximum value of $k$ is the total number of nodes in the network. Combining Eq. 2 and 3, it can see that the time delay $T$ of data transmission is,

$$T = \sum (T_w + T_e) + \sum \text{Tra}_{m-1}, k \geq 2, k \in N, m = k - 1$$

Compared with the transmission time of the data packet in the path and the actual execution time of the data packet, the waiting time of the data packet in the queue is very small, and the waiting time of the data packet is ignored here. So, Eq. 5.4 can be transformed into,

$$T = \sum (T_e) + \sum \text{Tra}_{m-1}, k \geq 2, k \in N, m = k - 1$$

(3) Path survival vitality
The viability of the node is the continuous working time of the node under normal conditions, which is represented by $E_{id}$. It is obtained by the calculation through the smallest degree of the node’s connectivity $\omega$ and the remaining battery consumption $\theta$, which can be expressed by Eq. 6.

$$E_{id} = \frac{\theta}{\omega + 1} + \alpha$$

where $\alpha$ is a balance factor, which can be set freely according to the required path survival expectation, and adding 1 to the denominator is to prevent the invalidity of the Eq. caused by $\omega$ being 0. The value of $\omega$ is the number of nodes under the energy coverage of the node. As shown in Fig. 1, the connectivity of node C is 2, because the two nodes A and B are under the energy coverage of node C. $\theta$ is the original full energy of the node. As time goes by, its energy value continues to decrease, the covered communication radius is also shrinking, and the number of covered nodes decreases, so the value of $\frac{\theta}{\omega + 1}$ does not change much. The value can reflect the sustainable working time of the node to a certain extent, and then the viability of the entire path can be inferred.

The viability of the path is the reference value $E$ for the longest time the path can exist. $E$ is the minimum node viability in the path.

$$E = \min \{E_{id}\}, id = 1, 2, 3, L, n$$ (7)

(4) Comprehensive measurement of QoS parameters

The survivability of the path reflects the usable performance of the path, and the path with greater survivability has superior usability. It can mainly be determined by time delay, available bandwidth and path viability. Set the survivability of the path as $M$, then $M$ can be expressed as the Eq. 8.

$$M = \lambda \cdot \text{Bandwidth} + \chi \cdot \frac{T}{T_B} + \delta \cdot \frac{E}{E_B} + \zeta$$

$$\lambda + \chi + \delta = 1$$

$$0 \leq \lambda \leq 1$$

$$0 \leq \chi \leq 1$$

$$0 \leq \delta \leq 1$$

where $\lambda$, $\chi$, $\delta$ are the coordination coefficients, and $\zeta$ is the balance factor of routing survivability, which can be set according to the detection of data packets in the path. Bandwidth, $T$ and $E$ represent the bandwidth, time delay and path viability of the path respectively, and Bandwidth, $T_B$, $E_B$ respectively represent the estimated standard values of the path bandwidth, time delay and path viability, which can be estimated according to the actual situation.

3.2 Route establishment

The process of route establishment is a process initiated by the source node to establish a blockchain in which the available nodes in the network are continuously connected to the chain by means of request/response. The ultimate goal of the process is to find two short and most irrelevant chains ending with the destination node. When the source node needs to send data to the destination node and the source node does not have a route to the destination node, the source node creates a genesis block to find the lists to the destination node. As shown in Fig. 2. Then the source node
sends the detection packets (EERQ) to its neighbor nodes according to its Merkle tree, the process of finding the destination node is started now.

where Merkle tree is the neighbor node of the node, and Pre-point is the address of the previous node. Since this node is a creator node, there is no previous node, the value is null. ID represents the address of this node.

(1) The source node sends EERQ packets to its neighbor nodes, and the time timer is started and valid time domain $\tau$ is set.

(2) When the delay field $\zeta$ in the EERQ is bigger than valid time domain $\tau$ during transmission, the EERQ packet becomes invalid due to the excessively long time delay in finding path.

(3) According to the storage capacity flag (isfull) in the node memory routing table, when its value is 0, go to (10). When the value is 1, the timer for processing packets of the node is started.

(4) After the neighbor node receives the EERQ packet, it first checks whether it is the first time to receive the EERQ packet. If so, it replies to a normal confirmation packet CEERP and asks for linking to the blockchain. The CEERP package mainly includes the time of receiving the EERQ packet and the maximum output bandwidth of the node. Otherwise, the EERQ packet is discarded and transferred to (5).

(5) After receiving the normal acknowledgement packet CEERP from the neighbor node, at first the source node checks whether its time delay is within a reasonable time and whether the maximum output bandwidth of the node is in an appropriate range, and then it is identified whether the neighbor node is a black hole node or a wormhole node. If the conditions are satisfied, the node is agreed to connect to the blockchain. Otherwise, it will be rejected and detected regularly. Node 1 and node 3 that meet the requirements will be connected to the source node, as show in Fig. 3.

(6) The survivability of this node is calculated according to Eq. 6, and the calculation results are compared with the survivability domain $E$ of the path in the detection packet EERQ. If it is less, the domain value will be updated, otherwise the survivability domain value $E$ of the path in EERQ will remain unchanged.

(7) Check whether the processing packet timer of the node is off, and if so, go to (10). Otherwise, according to the viability of the node, the RREQ forwarding data counter is started and the counter value is set. The value of the path viability $e$ is calculated by Eq. 7. The specific method to determine the node timer is as follows,

①The counter is set to a maximum value, that is, regardless of the viability of the node, the counter value cannot be set to exceed the maximum value. The maximum value is determined according to the specific communication time delay. Set the maximum value as $\text{MAX}$ and the communication time delay as $T$ then,

$$\text{MAX} = \frac{1}{2}T$$

(9)

②The corresponding counter value $D_t$ is set according to the node's vitality $E$, its connectivity $\omega$, and the number of hops $H_d$ in EERQ.

$$W = \omega - \left\lfloor \frac{\omega}{H_d + 1} \right\rfloor$$

(10)

$$D_t = (1 - \frac{1}{W + 1})\text{MAX}$$

(11)
When the connectivity $\omega$ of the node's vitality $E$ is 0, then $D_i$ is 0. Because when the connectivity $\omega$ of node vitality $E$ is 0, the node has no forwarding ability and conditions. From the above Eq., this chapter concludes that only the greater the connectivity of the node and the smaller the hop number in EERQ, the closer the counter of the node is set to the max value, that is, the larger value the node timer could be got.

(8) Judge whether the node is the destination node, if so, go to (10). Otherwise, go to (9).

(9) The node forwards the EERQ request packet, go to (4), and repeats the forwarding of the request packet.

(10) If the destination node is found, the public key information of the destination node is transmitted to the source node according to the blockchain path. Otherwise end this search.

The Ad Hoc network is shown in Fig. 4, where node 2 and node 6 are malicious nodes, and node 4, node 8 and node 11 are nodes that do not meet the constraints. The source node is S and the destination node is D.

After the algorithm, the block of the source node saves all the retrieval results from the source node to the destination node, as shown in Fig. 5.

It can be seen from Fig. 9 that path detection is no longer performed at malicious nodes and nodes that do not meet the constraint conditions. In the figure, the connecting lines at node 2, node 4, node 6 and node 8 lose the arrow, indicating that the blockchain connection has been lost in this part. It can be seen that the network finally formed three paths, R1: S-1-7-11-D, R2: S-1-7-12-13-D and R3: S-3-9-14-D. The three paths are recorded in the memory routing table of the source node in the order of arrival time, and the intermediate nodes on each path enter the table in order of increasing repetition rate, and the initial value is 0. The three available paths in Figure 5-10 enter the memory table as shown in Tab. 1.

| Path name | The values (SHU) of intermediate nodes |
|-----------|----------------------------------------|
| S1        | 0 0 0                                  |
| S2        | 1 1 0 0                                |
| S3        | 0 0 0                                  |

In Tab. 3, the minimum sum of SHU values of each path is taken as the first criterion, because the larger the SHU value, the greater the correlation between the path and the previously reached path. In order to find the largest irrelevant path, it can be obtained by simple accumulation of the SHU value. When the Shu value of the path is the same, the length of the path is taken as the second selection criterion, because in the same correlation, the shorter the path is, the more stable the link is, and the smaller the control cost of the network is. When the length is the same, the order in which the source node receives the public key returned by the destination node according to the corresponding path is taken as the selection standard, because the faster the data packets of the first arrival path are transmitted, the less the end-to-end delay is, and the network data transmission efficiency can be greatly improved.

### 3.3 Route Maintenance

In the Ad Hoc network, although there are multiple paths to ensure the transmission of information, it is still possible that the link connection fails due to node movement, and the occurrence of congestion leads to insufficient bandwidth, transmission timeout, and other path
damage. Therefore, the repair of the path cannot be ignored. This chapter mainly uses three methods to resolve these problems.

1. By selecting two paths with the greatest correlation as the main path for information transmission, once one path is damaged, the other path may not be affected by it.

2. There are backup paths in the source node. When the two selected main paths are all destroyed, one of them can be selected for information transmission without having to search for paths again.

3. Use the link interruption mechanism mentioned for processing.

4 Results and Discussion

4.1 Simulation environment and parameter setting

In this paper, NS3.29 software is used as the simulation platform [10]. The network topology is a network model with nodes randomly distributed in a plane rectangular area of 1000 m × 1000 m. The velocity of mobile nodes is 5m/s to 50m/s. The MAC layer adopts IEEE 802.11 and adopts constant bit rate (CBR) data stream. The simulation time is 900s, and the maximum residence time of nodes is 0s, 5s, 10s, 20s, 30s. The abnormal nodes added during the experiment include a random number of energy-constrained nodes, black hole nodes, wormhole nodes, and sybil nodes. In the simulation evaluation of AODV-MQS, reference [8] algorithm and AODV, the following performance parameters are mainly considered: data packet delivery rate, data end to end delay and control overhead. The main performance parameters are shown in Tab. 4.

Table 4: Experimental parameters and their configuration

| Parameters     | value          | Explanation               |
|----------------|----------------|---------------------------|
| N              | 10-80          | Number of network nodes   |
| DOM            | 1000 m×1000 m  | Area size                 |
| B_W            | 10Mps          | bandwidth                 |
| Vmax/Vmin      | 50/5(m/s)      | Maximum and minimum node speed |
| Tmax/Tmin      | 40/0           | Time interval for sending packets |

4.2 Data Analysis

Fig. 6 shows how the end-to-end average delay varies with node speed when all nodes in the network are normal nodes. It can be seen from the figure that the AODV-MQS algorithm proposed in this paper has a lower time delay than the other two algorithms when the node moving speed is low. Because the maximum irrelevant multipath ensures that when a node in one path moves and cannot communicate, the other alternative path can be used directly without rerouting and searching. For the other two algorithms, there is no alternative path for the destruction of a path, so it has to be rerouted and searched, thereby increasing the end-to-end delay. However, when the nodes move faster, the multi-path proposed in this paper may also break at the same time, which will inevitably lead to time delay similar to the other two algorithms.

Fig. 7 shows how the end-to-end average delay varies with node speed when there are abnormal nodes in the network. It can be seen from the figure that the algorithm proposed in this paper has obvious advantages in end-to-end delay. Although the delay deteriorates with the increase of node speed, the change is slow and tends to be peaceful, which greatly guarantees the end-to-end delay time. And the other two algorithms, compared with the algorithm in this paper, is not good in end-to-end time delay performance, and with the growth of node speed, the time delay becomes more
worse.

Fig. 8 shows how the data packet drop rate varies with node speed when all nodes in the network are normal nodes. It can be seen from the figure that the algorithm proposed in this paper performs worse than the algorithm in the literature [8] within a certain range of node moving speed. The reason may be that the two paths of AODV-MQS algorithm simultaneously broken in this interval, causing more data packets lost. After that, it can be recovered quickly with the acceleration of the node movement speed, because two paths to the destination node are restored in the blockchain. As a whole, the algorithm proposed in this paper can control the packet rate at about 90%, which effectively ensures the smooth operation of the network.

Fig. 9 shows how the delay of data packet delivery rate varies with node speed when there are abnormal nodes in the network. It can be seen from the figure that the algorithm proposed in this paper is obviously better than the other two types of algorithms. With the increase of node's moving speed, the packet delivery rate decreases to a certain extent, but when the speed increases again, the packet delivery rate rises, the main reason is that this paper adopts the blockchain technology to screen the nodes entering the blockchain network, which ensures the reliability of the nodes on the data transmission path. In addition, the algorithm proposed in this paper adopts alternative paths and constraints to ensure the safety of path transmission as much as possible.

5 Conclusion

This paper researches on multipath QoS routing security algorithm (AODV-MQS) based on blockchain. AODV-MQS is an on-demand QoS routing security algorithm based on the improved AODV protocol of the blockchain. The abnormal nodes are avoided effectively by using path survivability constraints and blockchain technology, then two optimized paths are chosen. Finally, routing security of Ad Hoc network is increased greatly. The application of blockchain technology in Ad Hoc networks routing security is a new method, there are many aspects that need further study, such as data transmission encryption, consensus algorithm, energy distribution, etc.

Acknowledgements

The authors acknowledged the anonymous reviewers and editors for their efforts in valuable comments and suggestions. And thanked the Hengshui University and Wonkwang University for the use of their equipment.

Funding

This work was supported by the Project of Hebei Province for Department of Education Youth Fund (Grant No. QN2020520), and Hebei Province for Science and Technology Department Science Popularization (Grant No. 20550301K), and the Teaching Reform Project of Jiangxi Provincial Department of Education (Grant No. JXJG-17-17-15), and the National Social Science Foundation of China (Grant No. 18XXW011).

Availability of data and materials

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Authors’ contributions

Shuailing Yan proposes the innovation ideas, and Conglin Ran does the theoretical analysis, Liang Huang carries out experiments and data analysis. Lei Zhang participated in its design and coordination and helped to draft the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher’s Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.
Author details

1 Department of Information Technology Center, Jiujiang University, Jiujiang, 332005, P.R.China
2 Department of Mathematics and Computer Science, Hengshui University, Hengshui, 053000, P. R. China
3 School of Mathematics & Computer Science, Shangrao Normal University, Shangrao, 334001, P.R. China
4 Department of Computer and Software Engineering, Wonkwang University, Iksan, 54538, Republic of Korea

Reference

[1] Pal, A., Dutta, P., Chakrabarti, A. et al. An efficient load balanced stable multi-path routing for mobile ad-hoc network. Microsyst Technol 2020(1), 101-112(2020).
[2] A. M. Shantaf, S. Kurnaz and A. H. Mohammed. Performance Evaluation of Three Mobile Ad-hoc Network Routing Protocols in Different Environments. 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA). 1-6(2020).
[3] K. Lei, Q. Zhang, J. Lou, B. Bai and K. Xu. Securing ICN-Based UAV Ad Hoc Networks with Blockchain. IEEE Communications Magazine 57(6), 26-32(2019).
[4] A. S. Al Hasan, M. S. Hossain and M. Atiquzzaman. Security threats in vehicular ad hoc networks. 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI). 404-411(2016).
[5] N. Nishanth and A. Mujeeb. Modeling and Detection of Flooding-Based Denial-of-Service Attack in Wireless Ad Hoc Network Using Bayesian Inference. IEEE Systems Journal 2020(3), 1-10(2020).
[6] F. Abdel-Fattah, K. A. Farhan, F. H. Al-Tarawneh and F. AlTamimi. Security Challenges and Attacks in Dynamic Mobile Ad Hoc Networks MANETs. 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT). 28-33(2019).
[7] W. Wei, H. Song, H. Wang and X. Fan. Research and Simulation of Queue Management Algorithms in Ad Hoc Networks Under DDoS Attack. IEEE Access 5(1), 27810-27817(2017).
[8] T. Poongodi, M. S. Khan, R. Patan, A. H. Gandomi and B. Balusamy. Robust Defense Scheme Against Selective Drop Attack in Wireless Ad Hoc Networks. IEEE Access 7(1), 18409-18419(2019).
[9] H. Moudni, M. Er-Rouidi, H. Mouncif and B. El Hadadi. Attacks against AODV Routing Protocol in Mobile Ad Hoc Networks. 2016 13th International Conference on Computer Graphics, Imaging and Visualization (CGiV). 385-389(2016).
[10] F. Abdel-Fattah, K. A. Farhan, F. H. Al-Tarawneh and F. AlTamimi. Security Challenges and Attacks in Dynamic Mobile Ad Hoc Networks MANETs. 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT). 28-33(2019).
[11] S. Gupta, S. Kar and S. Dharmaraja. BAAP: Blackhole attack avoidance protocol for wireless Ad Hoc networks. 2011 2nd International Conference on Computer and Communication Technology (ICCCT-2011). 468-473(2011).
[12] Hongmei Deng, Wei Li and D. P. Agrawal. Routing security in wireless ad hoc networks. IEEE Communications Magazine 40(10), 70-75(2002).
[13] Qiang Huan, I. C. Avramopoulos, H. Kobayashi and Bede Liu. Secure data forwarding in wireless ad hoc networks. IEEE International Conference on Communications 5(1), 3525-3531(2005).
[14] P Papadimitratos, Z Haas. Secure message transmission in mobile ad hoc networks. Elsevier Ad Hoc Networks Journal 1(1), 193-209(2003).
[15] Sankara Narayanan S, Murugabooopathi G. Modified secure AODV protocol to prevent
wormhole attack in MANET. Concurrency and Computation: Practice and Experience 32(4), 2019. [16] Aswale A.B., Joshi R.D. Security Enhancement by Preventing Wormhole Attack in MANET. Innovations in Electronics and Communication Engineering 107(1), 225-237(2020).
[17] M. Karthigha, L. Latha and K. Sripriyan. A Comprehensive Survey of Routing Attacks in Wireless Mobile Ad hoc Networks. 2020 International Conference on Inventive Computation Technologies (ICICT), 396-402(2020).
[18] M. Karthigha, L. Latha and K. Sripriyan. A Comprehensive Survey of Routing Attacks in Wireless Mobile Ad hoc Networks. 2020 International Conference on Inventive Computation Technologies (ICICT), 396-402(2020).

Figure 1. Connectivity of the node C

Figure 2. Genesis block

Figure 3. Nodes connection
**Figure 4.** Ad Hoc network

**Figure 5.** Source node result block
Figure 6. End-to-end average delay comparison with node speed changes in safe environment

Figure 7. End-to-end delay comparison with node speed changes in unsafe environment
Figure 8. Data packet rate comparison with node speed changes in safe environment.

Figure 9. Data packet rate comparison with node speed changes in unsafe environment.