Global Journals LaTeX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

An Efficient Misbehaving Node Detection Algorithm in Manet

Shaheen Bohra[1] and Naveen Choudhary[2]

[1] College of Technology and Engineering

Received: 8 December 2014 Accepted: 31 December 2014 Published: 15 January 2015

Abstract

Manet is a collection of self-organizing mobile nodes participating in the network to forward packets for each other. However, some nodes in the network do not forward packets in order to save their energy. But these nodes make use of other nodes to forward their packets. Such unfair use of the network leads to degradation of its performance. So it is very important to detect such misbehaving nodes in the network. So in order to improve network performance we propose a scheme that is a combination of overhearing and acknowledgement based method to detect misbehaving nodes. The scheme is proposed to be built on top of DSR routing protocol.

Index terms— manet; misbehaving nodes; node cooperation; DSR protocol.

1 Introduction

obile ad hoc networks are of immense importance in many scenarios where infrastructure setup is not feasible. It is of great importance in disaster management scenarios. Ad hoc networks can be setup when a group of nodes communicate with each other by forwarding each other's packet or data. Nodes communicate with each other if they are in radio range of each other. So if two nodes are not within each other's range their data is transmitted with the help of intermediate nodes. So the intermediate nodes play a key role in efficient data transmission in ad hoc networks. In ad hoc networks each node has its own resources and the most crucial resource is power. Each node tries to save its energy so that it can use it for its own transmission. And maximum amount of energy is consumed during packet transmission. So when a packet arrives at intermediate nodes, some nodes drop the packet as they don't want to waste their energy in transmission of other nodes packet. So network becomes disconnected and packet doesn't reach their destination. Such nodes in the network are called misbehaving nodes. These node reply to route request and become part of route but when the packet actually arrives, they drop the packet. Because of this the sender node again sends route request to establish another route. It may happen that the other route also contains misbehaving nodes. If such process repeats the sender assumes that it is not possible to route the packet to the destination and it drops the packet. Such nodes decrease the network efficiency. Further, if an alternative path is found which does not contain misbehaving nodes, it leads to increased delay of packet transmission.

So it is very important to detect such misbehaving nodes in the network as these nodes cause unnecessary burden on cooperative nodes. These nodes use other nodes resource and transmit their packets but don't forward other nodes packet. So their detection becomes even more important to induce fairness in the network.

2 II.

3 Related Work

Buttayan and Hubaux [1][2] introduced a virtual currency method called Nuglets. In this technique a node has to pay other node for forwarding its packet. This requirement makes all the nodes interested in forwarding other nodes packet as they also need nuglets to forward their data packets. Payment of nuglets is either done by source node or destination node. The problem with this technique is that it is difficult to estimate the number of nuglets required by source node. Further the absence of central monitoring mechanism makes it even more difficult to induce fairness in the network.

Zhong and Yang **??**1][8] proposed an incentive based mechanism called Sprite. In this a node collects receipt for each forwarded packet. The receipt is nothing but the hash of the packet. To provide fairness in the network it has a central monitoring mechanism called credit clearance service. All the nodes send their receipt to the CCS. The CCS is responsible for providing credit to the nodes. The main disadvantage with this method is that the CCS can become a source of bottleneck.

Marti [7] proposed watchdog/Pathrater model in which overhearing technique is used to identify misbehaving nodes. When a node forwards a packet, it observes the next node to find whether it forwards the packet or not. A node is considered as misbehaving if it does not forward the packet. The misbehaving counter is incremented each time misbehavior is detected. If the counter exceeds a threshold value, that node is considered as misbehaving and is avoided by pathrater in future routes. Michiardi and Molva [1] [10] proposed Core, which uses a different reputation mechanism. It calculates a combined reputation rating. This rating is formed by direct observation, indirect observation and task specific behavior.

He and Dapeng Wu [12] proposed Sori, which also rely on watchdog mechanism. It also relies on both direct observation and second hand information. Each node maintains a neighborhood list which contains the number of packets received and forwarded by each neighbor. It also punishes the nodes which are considered misbehaving.

Soltanali [13] proposed a reputation-based scheme consisting of four modules. The Monitor module is based on the watchdog model. The opinion manager is responsible for formulating opinion regarding nodes behavior and advertises the opinion to neighboring nodes periodically. The Reputation Manager processes these opinions and derives a trust metric for a specific node. The Routing/Forwarding Manager use the trust metrics to select a routing path.

Bansal and Baker [5] proposed a reputation based method called OCEAN. This method relies on overhearing technique to find out misbehaving nodes. When a nodes rating falls below a faulty threshold it is added to faulty list. It also use second chance mechanism which allows previously considered misbehaving nodes to become active in the network again.

Balakrishnan, Deng and Varshney [4] proposed an acknowledgement based scheme. The overhearing technique monitors the sender of the next hop link which leads to false detections. So to confirm the successful packet reception this scheme makes use of a special type of acknowledgement packet called TWOACK which is send by the two hop neighbor. When a node does not receive an acknowledgement it considers the entire link to be misbehaving. So this leads to most of the nodes being unavailable for routing packets.

Roubaiey, Sheltami, Mahmoud, Shakshuki and Mouftah [14] proposed an adaptive acknowledgement AACK method which is a modified TWOACK method. It is a combination of end to end acknowledgement and TWOACK method. It uses a function to calculate the number of hops and depending on the result it either uses end to end acknowledgement or Twoack. The use of end to end acknowledgement results in reduced overhead of acknowledgement packets. And instead of detecting misbehaving links, it detects misbehaving nodes.

4 III. Proposed Methodology for Effecient Misbehaving Node Detection

We proposed a combination of overhearing and acknowledgement based scheme which is designed to be built on top of DSR routing protocol. Reputation based schemes such as OCEAN [5] relies on overhearing technique which face problems like ambiguous collisions, limited transmission power, limited overhearing range and false detections. The TWOACK [4] scheme solves the overhearing problems by the use of special type of acknowledgement packet termed as TWOACK. The receiver of the next hop link is responsible for sending acknowledgement to confirm successful packet receipt. But the disadvantage of TWOACK scheme is the additional overhead of acknowledgement packets. So to improve the efficiency of overhearing mechanism and reduce the overhead caused by acknowledgement scheme, we proposed a combination of overhearing and acknowledgement based method.

In this section we first describe OCEAN [5] and TWOACK [4] scheme and then we describe our proposed technique.

5 a) OCEAN (Observation based cooperation enforcement in ad hoc networks)

OCEAN [5] is a reputation based mechanism which relies on direct observation of nodes behavior. It is composed of five components to discover misbehaving nodes: 1. Neighbor Watch: It is responsible for monitoring the neighboring nodes and is based on watchdog model. It uses overhearing technique to detect node misbehavior. 2. Route Ranker: Every node maintains ratings for each of its neighboring nodes. Initially the rating is zero and then it is incremented or decremented according to positive or negative event observed by neighbor watch module. When the rating of node exceeds the Faulty Threshold, the node is added to the faulty list. 3. Rank-based Routing: It uses the information derived from the Neighbor Watch in route selection. To avoid the routes containing nodes in the faulty list, we add a variable-length field to the DSR Route-Request Packet (RREQ) called the avoid-list. The avoid list is a list of nodes that the RREQ transmitter wants to avoid in its future routes. A node appends its faulty list to the avoid list on re-broadcasting a RREQ. Any node which receives the RREQ checks the RREQ avoid list. The avoid list and RREQ route is compared to check if there is a common

node. This common node is a misbehaving node. The node drops the RREQ when such misbehaving node is detected else it either sends DSR Route reply or rebroadcast the RREQ. Similarly, a DSR Route-Reply (RREP) is accepted only if the route in the RREP does not contain a node in the locally-

6 Global Journal of C omp uter S cience and T echnology

Volume XV Issue II Version I Year ( ) H maintained faulty list. Otherwise, the RREP is simply dropped. 4. Malicious Traffic Rejection: It rejects traffic from nodes that are considered misbehaving. 5. Second Chance Mechanism: It is intended to consider the nodes that were previously considered misleading to become useful again. This is useful when a well behaved node is marked as a misbehaving node. Since OCEAN uses overhearing mechanism it is prone to problems like ambiguous collision, limited transmission power, limited overhearing range and false detections.

The monitoring mechanism of OCEAN relies on overhearing mechanism which can face problems like:

7 Ambiguous Collisions

When node A forwards packet to node B, node A start overhearing node B to check whether it forwards the packet to C. While overhearing if another node sends data to node A, node A fails to overhear node B‘s transmission. This is called ambiguous collision and leads to false detection TWOACK [4] technique solves the problem of ambiguous collision as node A will receive TWOACK [5] from node C which confirms successful packet reception.

8 Limited Overhearing Range

A cooperative node B may use low transmission power to send data towards C. Since node A‘s overhearing range is limited, it fails to overhear node B‘s transmission and detects B as a misbehaving node. This again leads to false detection. The TWOACK [4] scheme is capable of solving limited overhearing range problem.

9 Limited Transmission Power

A node can limit its transmission power such that the signal is strong enough to be overheard by the previous node but too weak to be received by the recipient node .This would also cause false detection. This problem can also be solved by TWOACK [4] method.

10 b) Twoack Scheme

TWOACK [4] scheme solves the overhearing problems described above by the use of an explicit acknowledgement packet termed as TWOACK. When a node forwards a packet, it verifies that the packet is received successfully by the node that is two hops away on the source route. This is done through the use of a special type of acknowledgment packet, called TWOACK. Suppose that the source S wants to send the packet to destination D. Source S will perform route discovery process and find a route to reach the destination D.

Now suppose A forwards a data packet to B, which is to be forwarded to C, A cannot detect if the packet has reached C successfully or not. Overhearing the node B would only tell A whether B is sending out the packet or not. However, A cannot tell that C has received the packet or not. The possibility of collisions at both A and C makes the overhearing technique vulnerable to false detections. The TWOACK packet sent by node C tells node A that the data packet has successfully reached node C. Each node mantains a list of data packet ID that are yet to receive TWOACK from next to next hop. When an acknowledgement is received it removes the ID from the list. When the node does not receive acknowledgement after a timeout, it increments the misbehaving counter C mis tiil the counter reaches the threshold value after which the link is marked as misbehaving. When the node detects misbehavior, it sends RERR message informing the source about detected misbehavior.

The main disadvantage of TWOACK scheme is he high routing overhead caused by TWOACK and RERR packets. High routing overhead also affects the increase in average latency of data packets. Another drawback of TWOACK technique is that it detects misbehaving links which gives misbehaving nodes more chance to drop the packets as it might be connected to other links [14].

11 c) Proposed Methodology

OCEAN [5] uses overhearing technique which suffers from problems like ambiguous collision, limited transmission power, and limited overhearing range. This results in false detections. The TWOACK [4] scheme is capable in solving overhearing problems and results in better detection technique.

But the TWOACK technique increases the routing overhead in the network due to broadcast of alarm message as well as acknowledgement packets.

To reduce the overhead of acknowledgement packets and improve the performance of overhearing technique we proposed a combination of overhearing and acknowledgement scheme. We modified the monitoring mechanism of OCEAN [5] method by introducing a positive threshold. The rating of node is initialized to zero in the beginning. When a node forwards the packet we increment its rating and similarly

12 Global Journal of C omp uter S cience and T echnology

Volume XV Issue II Version I Year ( )H

we decrement its rating if it drops the packet. First we start with the reliable TWOACK [4] scheme to detect misbehaving nodes. Each node while sending a packet checks the next node rating. If the rating is less than defined positive threshold, it continues with the TWOACK scheme so that we get sufficient positive evidence of nodes cooperative behavior. When the rating of node becomes equal to positive threshold it switch to overhearing technique to reduce the overhead of TWOACK scheme [4]. Each node maintains list of neighboring nodes and their rating. When node rating reaches faulty threshold we add the node to faulty list.

Each node on receiving a RREQ checks the faulty list. If the node sending the RREQ falls in the faulty list, its RREQ is simply dropped. When a node receives RREP it checks its faulty list to find if the path contains misbehaving node. If the path contains misbehaving nodes then the node does not use this path. Otherwise it sends the packet using this path.

We also use the second chance mechanism so that the nodes which were previously considered misbehaving can become active in the network again. This is useful in the case if a cooperative node is detected as a misbehaving node. The second chance rating of node is not initialized to 0 in order to prevent misbehaving nodes to further exploit the network [5]. The second chance timeout [5] should also be neither too high nor too low as high timeout will give less chance to the well behaved nodes and low timeout will allow misbehaving nodes to quickly enter the network again. So the second chance threshold is set to -30 and second chance timeout is 30. The faulty threshold should also be neither be too low nor too high. Low faulty threshold will quickly add nodes in the faulty list whereas high faulty threshold will give misbehaving node more chance to exploit the network. So the faulty Threshold is set to -40 and results are evaluated at different positive threshold.

Figure **??**

13 Performance Evaluation a) Simulation Methodology

The simulation is performed on network simulator ns2 with 50 mobile nodes moving in a 750×750 m 2 flat area. The transmission range of each node is 250 m. The IEEE 802.11 MAC layer and a random waypoint mobility model was assumed with pause time of 0 second is used. We used CBR traffic between pairs of nodes. The source and destination for each CBR pair are randomly chosen and there is no limit on the number of sources or destinations that a node can host. The scheme is analyzed by running simulations for networks with 10 CBR pairs. Each CBR source generates packets of size 512 Bytes, and transmits 8 packets per second. Each simulation lasts 100 seconds. 5 simulation runs (using different seeds) were used to obtain each data point. Table 1 shows the configuration parameters used by us for the simulation.

14 Year ( )

H We used the following performance metrics to evaluate our method at different percentage of misbehaving nodes. 1. Packet Delivery Ratio: It is defined as number of packets that successfully reached the destination node to the number of packets sent by the source node. 2. Average Latency: It is defined as the time taken by a data packet to travel from source node to the destination node. 3. Throughput: It is defined as the number of packets successfully received by the destination node. It is measured in Kbps. 4. Routing Overhead: It is defined as the number of routing related transmission to the total number of transmissions.

15 b) Simulation Results and Discussions

Figure **??** shows the average latency experienced by packets in our proposed scheme at varying positive Threshold. In a network of 50 nodes, 10 nodes were misbehaving. We observed that with the increase of positive threshold the delay experienced by packets to reach the destination increases. It is due to the increase of TWOACK packets in the network as well as the switching overhead which accounts to computation time and power. After positive threshold 80 the delay increases sharply.

In Figure 5, we show the packet delivery ratio of our proposed scheme under the same conditions as mentioned above. It is observed that maximum packet delivery ratio is obtained at positive threshold 80 and after that packet delivery ratio varies and we do not observe much increase in it. It is due to increase in latency at high positive threshold and because of this less number of packets are able to reach the destination. So there is not much gain in increasing the positive threshold. In Figure **??**, we show the throughput of our proposed scheme at varying positive threshold. We observe that initially the throughput increases with increase in positive threshold but after a limit at positive threshold 80, we do not observe much increase in through put. It is due to the delay experienced by packets which results in less no of packets to reach the destination. detected misbehavior, overhead experienced by our proposed scheme will be less than TWOACK schme. Now we evaluate the performance metrics at positive threshold 80 and varying percentage of misbehaving nodes. Figure 8 compares the packet delivery ratio of the TWOACK, OCEAN and our proposed scheme as a function of different percentage misbehaving nodes. The percentage of misbehaving nodes in the network was varied from 0 (all nodes are well-behaved) to 40%. From the figure, we can observe that the packet delivery ratio of our scheme is more than the OCEAN method. The packet delivery ratio decreases as the number of misbehaving node increase. This is due to the problem of missing routes and the overhead of searching for alternative routes. Compared with the OCEAN

scheme, our proposed scheme maintains a relatively high packet delivery ratio. For example, when there are 40% nodes that are misbehaving, the proposed scheme delivers about 81-89% of data traffic. In Figure 9, we show the throughput of the TWOACK scheme, OCEAN scheme and our proposed scheme. The network parameters are the same as those used to obtain figure 8. It is evident from the curves that the throughput of our proposed scheme is better than OCEAN scheme. In Figure 10, we show the routing overhead of the TWOACK scheme and our proposed scheme. The network parameters are the same as those used to obtain figure 8. It is evident from the curves that the routing overhead of our proposed scheme is much less than TWOACK scheme. Routing overhead of TWOACK scheme is mainly due to the transmissions of the TWOACK packet for each data packet processed by each of the triplets and the transmissions of RERR packets to report misbehaving nodes.

16 Conclusion

Mobile Ad Hoc Networks (MANETs) have been an active area of research over the past few years, due to their potentially widespread application in military and civilian communications. Such a network is highly dependent on the cooperation of all its members participating in the network. This makes it highly vulnerable to selfish nodes. In this paper, we have proposed and evaluated a scheme which is a combination of acknowledgement and overhearing based scheme, which can be easily added-on to source routing protocols such as the DSR protocol. The schemes detect misbehaving nodes so that other nodes may avoid them in future route selections, with the aim of overall improvement in performance metrics such as throughput, average latency, routing overhead and packet delivery ratio. Simulations have showed that, in a network where up to 40% of the nodes are misbehaving, the proposed scheme improves the throughput and packet delivery ratio compared to OCEAN method and reduced overhead and latency as compared to the TWOACK method. By introducing acknowledgements in OCEAN method the overhead is increased, but it is still less than original TWOACK scheme.

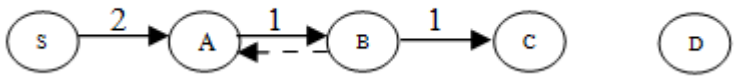
Therefore the proposed scheme can prove quite fruitful especially if less than half of network nodes are misbehaving. The scheme solves the overhearing problems unlike OCEAN and also keeps the routing overhead manageable under low to moderate traffic load unlike TWOACK.

17 Global Journal of C omp uter S cience and T echnology

Volume XV Issue II Version I Year ( ) [1] [2]

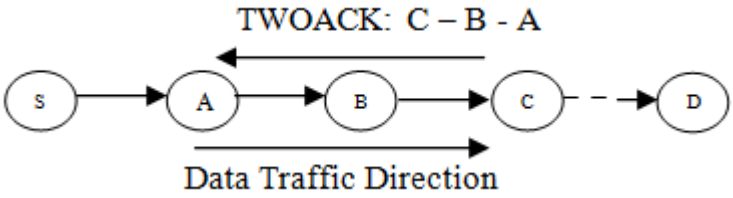
Figure 1:

[1] © 2015 Global Journals Inc. (US)
[2] © 2015 Global Journals Inc. (US) 1



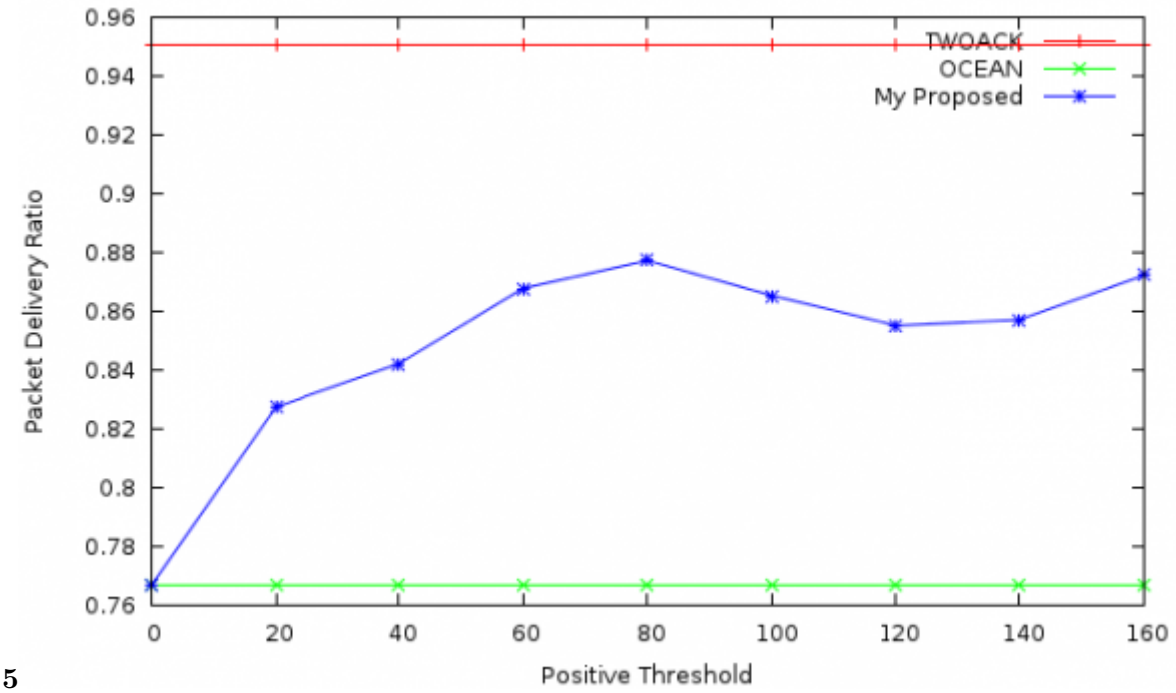
1

Figure 2: Figure 1 :



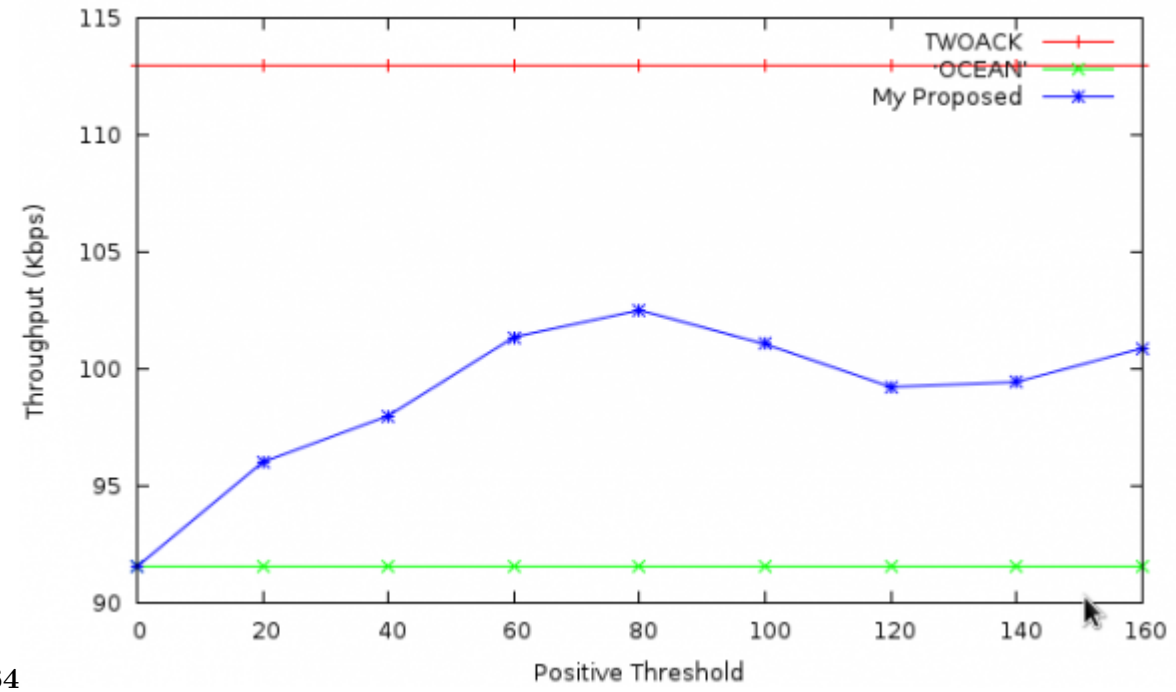
2

Figure 3: Figure 2 :



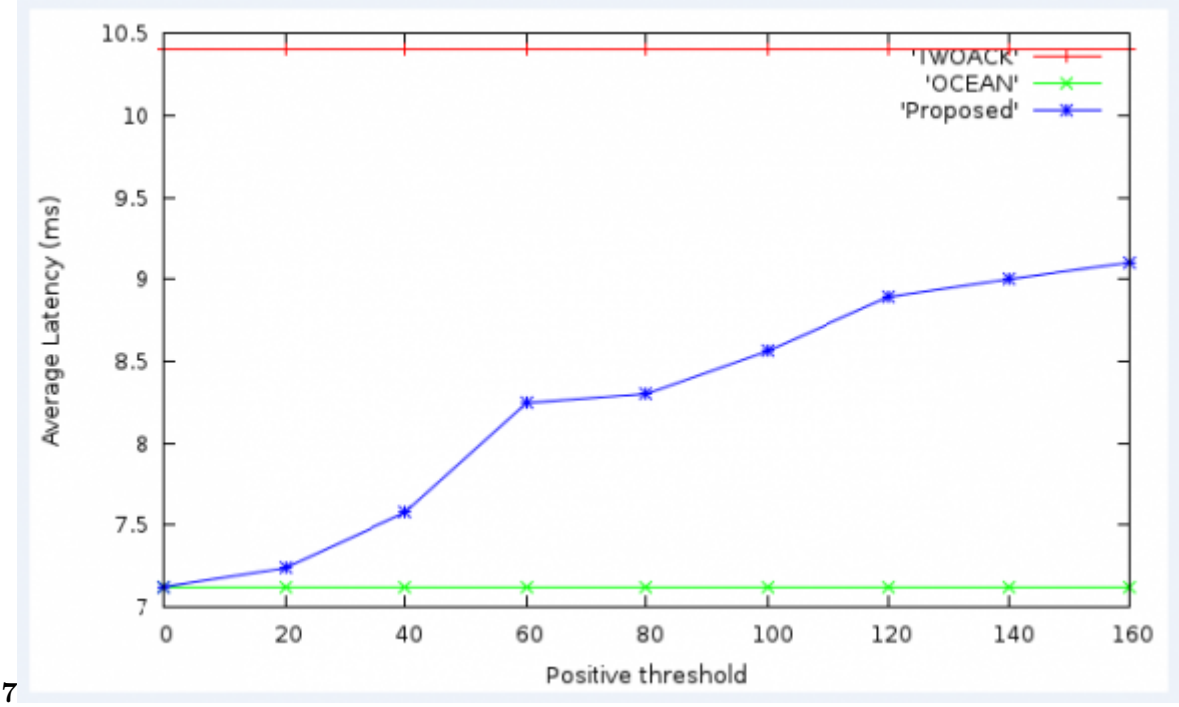
5

Figure 4: Figure 5 :



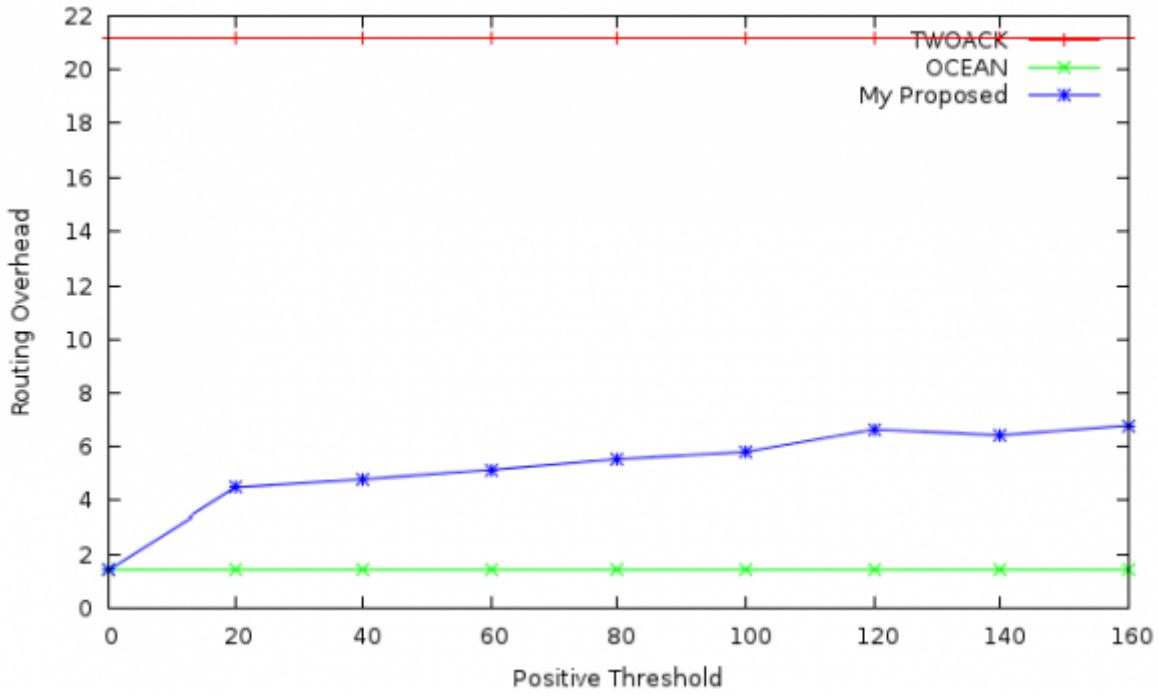
64

Figure 5: Figure 6 :HFigure 4 :



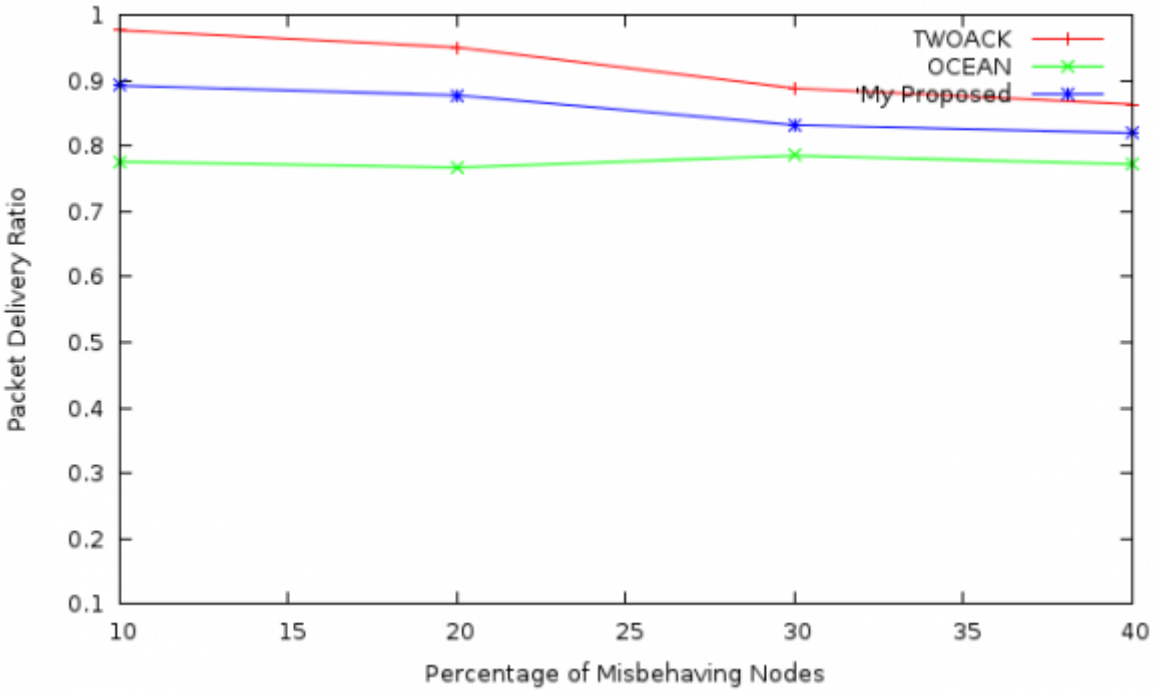
7

Figure 6: Figure 7 :



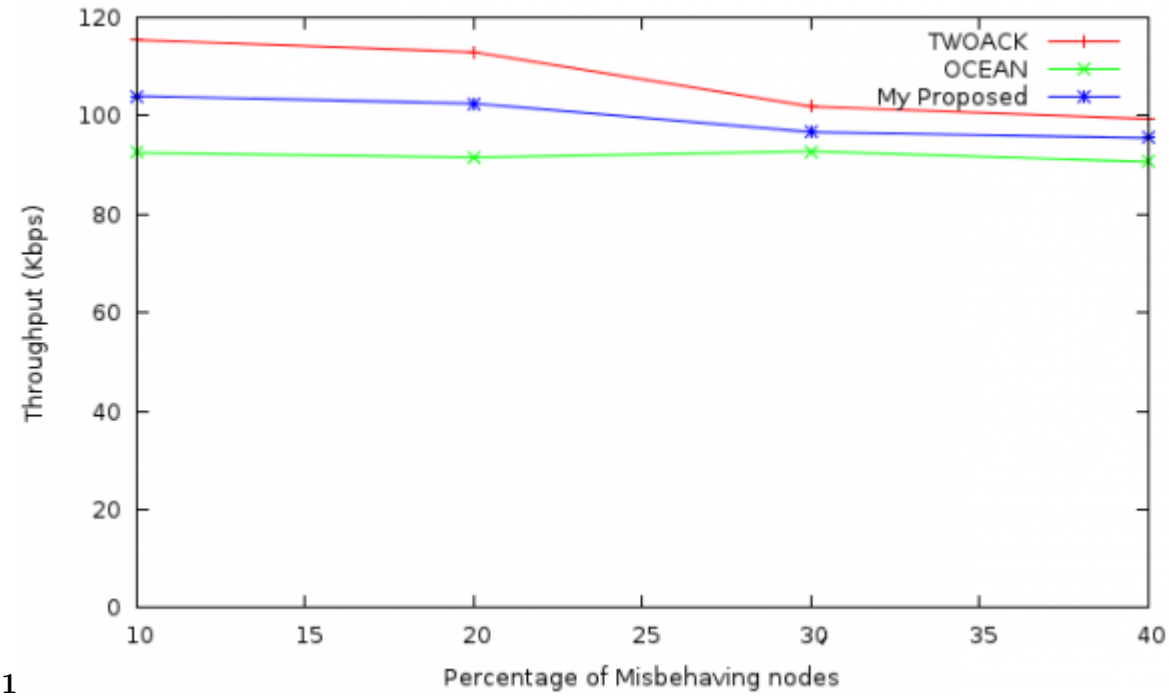
8

Figure 7: Figure 8 :



9

Figure 8: Figure 9 :



1011

Figure 9: Figure 10 :HFigure 11 :

1

Parameters	Value
Number of nodes	50
Simulation Area	750 x 750 m
Mobility Model	Random waypoint model with pause time 0
Traffic Type	CBR
Packet size	512 bytes
Packet Rate	8 /sec
Maximum connections	10

Figure 10: Table 1 :

.1 Global Journals Inc. (US) Guidelines Handbook 2015

www.GlobalJournals.org

[Padiya et al.] 'A System for MANET to Detect Selfish Nodes Using NS2'. S D Padiya , R Pandit , S Patel . *International Journal of Engineering Science and Innovative Technology* 1.

[Al-Roubaiey et al. ()] 'AACK: Adaptive Acknowledgment Intrusion Detection for MANET with Node Detection Enhancement'. A Al-Roubaiey , T Sheltami , A Mahmoud , E Shakshuki , H . *24th IEEE International Conference on Advanced Information Networking and Applications*, Mouftah 2010.

[Soltanali et al. ()] 'An Efficient Scheme to Motivate Cooperation in Mobile Ad hoc Networks'. S Soltanali , S Pirahesh , S Niksefat , M Sabaei . *Proceedings of the 2007 Third International Conference on Networking and Services*, (the 2007 Third International Conference on Networking and Services) 2007.

[Saxena and Rana ()] 'Analysis of Selfish and Malicious Nodes on DSR Based Ocean Protocol in MANET'. A Saxena , J L Rana . *International Journal of Computer Science and Communication Technologies* 2010. 3.

[Michiardi and Molva ()] 'CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks'. P Michiardi , R Molva . *Communication and Multimedia Security Conference*, 2002. 2002.

[Misra et al. ()] *Guide to wireless ad hoc networks*, S Misra , I Woungang , S C Misra . 2009. Springer.

[Marti et al. ()] 'Mitigating Routing Misbehavior in Mobile Ad Hoc Networks'. S Marti , T J Giuli , K Lai , M Baker . *Proc. International conference on mobile computing and networking (MobiCom)*, (International conference on mobile computing and networking (MobiCom)) 2000.

[Bansal and Baker ()] 'Observation-based cooperation enforcement in ad hoc network'. S Bansal , M Baker . *IEEE arXiv* 2003. 2.

[Buchegger and Boudec ()] 'Performance Analysis of the CONFIDANT Protocol : Cooperation of Nodes - Fairness in Dynamic Ad Hoc NeTworks'. S Buchegger , J Y Boudec . *Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, (IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)) 2002.

[Vijaya ()] 'Secure 2ACK Routing Protocol in mobile ad hoc networks'. K Vijaya . *Proceedings of TENCON '08*, (TENCON '08) 2008. IEEE p. .

[Qi et al. ()] 'SORI: A Secure and Objective Reputation-based Incentive Scheme for Ad-hoc Networks'. H Qi , O D Wu , P Khosla . *IEEE Wireless Communications and Networking Conference*, 2004. 2 p. .

[Zhong et al. ()] 'Sprite: a simple, cheat-proof, credit-based system for mobile ad-hoc networks'. S Zhong , J Chen , Y Yang . *INFOCOM 2003.Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*, 2003. 3 p. .

[Buttyan and Hubaux ()] 'Stimulating cooperation in self-organizing mobile ad hoc networks'. L Buttyan , J Hubaux . *Mobile Networks and Applications* 2002. 8 p. .

[Balakrishnan et al. ()] 'TWOACK: Preventing Selfishness in Mobile Ad Hoc Networks'. K Balakrishnan , J Deng , P K Varshney . *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC '05)*, (IEEE Wireless Comm. and Networking Conf. (WCNC '05)) 2005. 4 p. .