RAAODV: a Reputation-Aware AODV for Mobile Ad hoc Networks

Ahmed Al-hamadani
University of Mosul
Mosul, Iraq
Aalhamadani@uomosul.edu.iq

William H. Allen
Florida Institute of Technology
Melbourne, FL, USA
wallen@fit.edu

ABSTRACT

Mobile Ad hoc Networks (MANETs) are self-configuring autonomous networks consisting of a set of mobile nodes connected by wireless links. The lack of fixed infrastructure and centralized administration makes it necessary for nodes to cooperate with each other to communicate. However, some nodes may choose to stop forwarding packets intended for others to conserve power, bandwidth, and computational resources. These selfish nodes can have a detrimental impact on the performance of the entire network due to packet loss, denial of service, a decrease in network throughput, and network partitioning. The legacy Ad hoc On-demand Distance Vector (AODV) routing protocol suffers from the inability to mitigate this selfish behavior. In this paper, a new variant of AODV called Reputation-Aware AODV (RAAODV) is proposed and evaluated. RAAODV monitors neighboring nodes, manages reputations, and reacts against detected selfish behavior. Experiments show that RAAODV improves network performance by 54% above legacy AODV when as many as 50% of nodes behave selfishly all the time.

1. INTRODUCTION

In a Mobile Ad hoc Network (MANET), all nodes are free to join, to move about as needed, and to leave the network at will. As a result, the network topology can change rapidly and is difficult to administer. MANETs are usually deployed in dynamic and agile environments, such as for emergency or rescue operations, battlefield networks, disaster relief environments, or for spontaneous meetings. The decentralized administration, frequent changes in routing, and strong reliance on collaboration of nodes for routing and forwarding can make MANETs more vulnerable to different kinds of attacks and more susceptible to numerous security issues.

The limited wireless capacity and bounded transmission range causes each individual node in a MANET to rely on neighboring nodes to forward packets towards the node's destinations. Conventional routing protocols like Dynamic Source Routing (DSR) [7] and Ad hoc On-demand Distance Vector (AODV) [11] were developed under the assumption that all nodes in a MANET are cooperative. That is, whenever a particular node is expected to forward a packet for another node, it will do so and will not intentionally drop or corrupt the packet. However, with the existence of resource constraints such as limited battery power and restricted bandwidth, this is not always the case. Some nodes may choose to drop packets which are not intended for them to conserve power, bandwidth, and/or CPU cycles while they will still expect other nodes to forward their own packets [4]. The existence of selfish behavior in the network may degrade the network performance significantly due to packet loss, denial of service, isolation of non-selfish nodes, and reduced throughput [3].

To address this problem, many solutions can be found in the literature that propose the use of reputation values to judge the selfishness of a particular node. The majority employ the DSR routing protocol to take advantage of its source routing feature which can be used to detect and avoid routes that contain one or more selfish nodes. However, DSR-based solutions suffer from poor scalability and increased control overhead. To their advantage, DSR-based solutions can also use two sources of information to evaluate the cooperation of a particular neighboring node. First-hand information is gained from directly observing the cooperation of a specific neighboring node, while second-hand information is collected from the other neighbors of that node and represents their opinions about the reputation of the specific node. Although the use of second-hand information can be valuable in detecting selfish nodes more quickly, it introduces new problems such as false accusations, rumor spreading and collusion.

This paper proposes a solution that incorporates reputation based decisions into the AODV routing protocol in an effort to support scalability while mitigating the effects of selfish behavior in MANETs. This approach uses only direct (i.e., first-hand) observations to avoid the disadvantages of second-hand information described above. This proposed variant of AODV, which we call Reputation-Aware AODV (RAAODV), attaches a Reputation System Unit (RSU) to the original AODV protocol. The RSU consists of three components. The Behavior Monitor (BM) is responsible for directly monitoring neighboring nodes to detect selfish behavior. The Reputation Manager (RM) manages the reputations of neighboring nodes based upon their cooperation history. The Path Manager (PM) updates routing tables to reflect the current reputation of neighboring nodes so that the most trusted neighbors are selected for packet routing.

The remainder of this paper is organized as follows. Section 2 discusses related research. Section 3 describes the

ACM SE '14 March 28-29, 2014 Kennesaw, GA USA

ACM 978-1-4503-2923-1/14/03 ...$15.00
http://dx.doi.org/10.1145/2638404.2638462

structure of RAAODV and how its components detect selfish behavior and use that information to improve routing. Section 3 also describes the simulation environment used to evaluate RAAODV and the results of our experiments. Section 4 discusses the impact of the proposed approach.

2. RELATED WORK

Buchegger and Le Boudec [3] defined reputation as "the opinion a node has of another", while Michiardi and Molva [9] defined reputation as "the amount of trust inspired by a particular member of a community in a specific setting or domain of interest". Members that collaborate with others in a community develop a good reputation among the community members and therefore can use the shared community resources, while those who refuse to be cooperative earn a poor reputation among the community members. Uncooperative nodes are gradually isolated and excluded from the community until they choose to cooperate again [9]. In the context of MANET routing, the reputation of a node is the amount of trust other nodes have for this node regarding its contributions in forwarding activities. Therefore, a reliably forwarding node becomes the best node that other nodes can deal with [6].

2.1 DSR-driven reputation-based solutions

The majority of the reputation-based solutions proposed in the literature are based on the DSR routing protocol, taking advantage of its source routing approach and the availability of second-hand information on the reputation of nodes.

Marti et al. proposed a solution that consists of two components, namely, *watchdog* and *pathrater* [8]. The *watchdog* observes misbehaving nodes by using promiscuous mode to monitor the transmissions of neighboring nodes. Nodes that repeatedly fail to forward packets within a preset interval are considered to be misbehaving and their reputation is adjusted accordingly. The *pathrater* selects the best route depending on the reputation ratings of the monitored nodes. The main weakness of the *watchdog* and *pathrater* approach is that nodes only avoid using misbehaving nodes but do not avoid forwarding packets from the misbehaving nodes, thus the selfish nodes continue to operate normally.

The CONFIDANT protocol [1], presented by Buchegger and Le Boudec, consists of four components: the monitor, the reputation system, the trust manager, and the path manager. The *monitor* resembles the watchdog mechanism in detecting the deviations in behavior of the next node. If misbehavior is detected, the *trust manager* sends ALARM messages to warn other nodes about the misbehaving node. Arriving ALARM messages are passed to the trust manager for evaluation of trustworthiness before triggering a reaction. The *reputation system* only updates a suspect node's reputation after receiving a preset number of ALARMs regarding that node, combined with the evaluating node's own experience from monitoring the suspect node. The *path manager* ranks the available paths by the reputations of their nodes to aid in selecting a reliable path for transmission. The propagation of negative information in the form of ALARM messages raises the issue of false accusation and rumor spreading and the authors addressed that in other work [2].

2.2 AODV-driven reputation-based solutions

In [5], Dewan et al. use direct acknowledgments rather than a watchdog mechanism to detect misbehaving nodes. When a source node sends a packet to a destination node, it waits for an acknowledgment from the destination node. If an acknowledgment is received, the source node will increment the reputation of the first hop node. Similarly, each intermediate node in that route to the destination node will reward its respective next hop node accordingly. If there is no acknowledgment received at the source node within a preset timeout, the source node will penalize the first hop node by decrementing its reputation, as does every node in the route until the selfish node is reached.

Wang et al. [12] presents an approach that distinguishes selfish peers from cooperative ones based exclusively on local observations of the routing protocol behavior by monitoring the activity of control packets instead of the forwarding of data packets. Their approach employs a finite state machine of locally-observed routing protocol actions to maintain a description of the behavior of each neighboring node rather than using watchdog or promiscuous operations.

3. OVERVIEW OF RAAODV

This chapter describes the design of a proposed new variant of AODV called Reputation-Aware AODV (RAAODV) that provides a potential solution to the problem of selfish nodes in MANETs.

3.1 Motivation

DSR nodes can store multiple routes to a given destination and can learn additional routes by monitoring packets intended for other nodes. Consequently by knowing the reputation of each node along all of these routes, it can select the route with the best reputation. Although these features of DSR can be attractive for developing an effective reputation-based system, they introduce new problems. As more nodes begin transmitting on the same route, the route becomes congested and causes a bottleneck; and if the route is broken due to link failure, many nodes will be affected. Additionally, nodes may extract information from stale routes, resulting in unnecessary overhead. This problem is further complicated if the network is highly dynamic with frequent changes since the number of stale routes increases and more route discoveries are needed.

In contrast, AODV is a hop-by-hop routing protocol which relies on routing information that is stored locally in each node to determine the best path to the destination. Thus, data packets only carry the destination address from hop to hop. This considerably reduces the memory and processing overheads for nodes and makes AODV a suitable choice for large or dynamic networks. In addition, AODV utilizes sequence numbers to install the fresher routes in the routing table and uses a hello message mechanism to keep track of neighbor connectivity and rapidly detect link failures. This allows it to adapt more quickly to frequent changes network topology. Building a reputation-based solution based on AODV has clear advantages, especially for large and highly dynamic networks. Therefore, RAAODV is an extension of AODV.

3.2 Assumptions

RAAODV relies on a number of assumptions to work more efficiently. These assumptions are as follows:

- RAAODV focuses on mitigating the effects of selfish behavior in the network. It is not intended to address other types of misbehavior triggered by faulty or malicious nodes at any layer (e.g. eavesdropping, spoofing, modification, fabrication, tunneling, replaying, etc.).
- Only the source and destination nodes in the network are assumed to be reliable and unselfish. They can also be chosen for forwarding packets for other nodes. Any intermediate node could be unreliable and selfish.
- RAAODV expects that some of the intermediate nodes are selfish, but not all of them, as this is a very extreme condition in the network.
- Selfish nodes do not drop control packets, only data packets. From the selfish node's point of view, it is important to forward routing control packets and be fully aware of all routing messages so it can send its own packets correctly.
- The wireless interface of each node has an omnidirectional antenna and supports the promiscuous mode for receiving packets and all wireless links are bidirectional and symmetric.
- Nodes do not maliciously forward packets to a nonexistent node in order to appear as if they are not behaving selfishly, since forwarding packets in this way consumes their own resources.
- Fragmentation is disabled when generating and sending any packet.

3.3 Design and Implementation

RAAODV incorporates a Reputation System Unit (RSU) within the original AODV mechanism to aid in making reputation aware routing and forwarding decisions. The RSU consists of three components, namely, the Behavior Monitor (BM), the Reputation Manager (RM), and the Path Manager (PM). These components collaborate with each other and form a layer that fits in between the MAC layer and the network layer of the protocol stack at each node. Figure 1 illustrates RAAODV architecture in detail.

The *Behavior Monitor* (BM) monitors the one-hop neighboring nodes, detects their selfish or cooperative behaviors and reports that information to the RM. The BM utilizes a mechanism similar to the *watchdog* mechanism [8] taking advantage of the broadcast medium of the wireless channel. When a node A transmits a packet to another node B, all nodes that are in the same wireless coverage area of node A can monitor this transmission activity. Therefore, the BM of any particular node can overhear the transmissions of the next hop for a passive acknowledgment, which can be considered as an implicit indication whether the next hop has actually forwarded the packet or not. After sending a packet to a next hop, if the same packet is received within a specific time, it can be inferred that the next hop has behaved cooperatively by forwarding this packet.

Despite the fact that passive acknowledgments suffer from the impact of several wireless problems such as collisions, interference, and limited transmission power, they cause less overhead and delay than active acknowledgments, and this is beneficial especially in a highly dynamic network such as a MANET. RAAODV requires that a node turn on promiscuous mode in its wireless interface, thus the node can receive

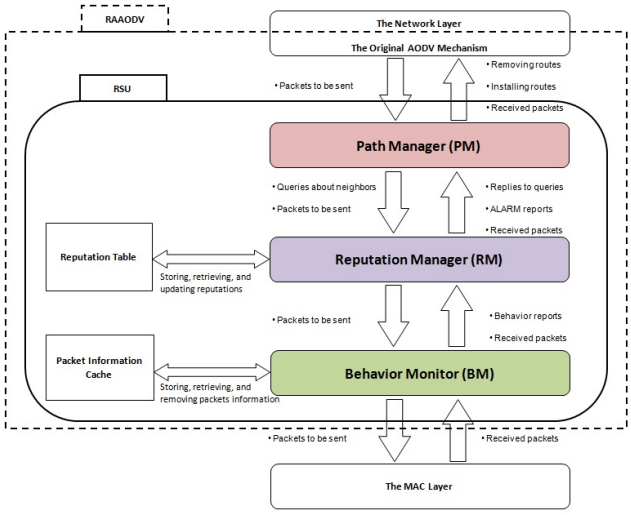


Figure 1: RAAODV Architecture

any packet within its wireless transmission range even if the packet is addressed to another node. The BM keeps track of the forwarding behavior of every node by storing identifying information for each the packet in a cache called the Packet Information Cache (PIC). Before sending any data packet, the BM creates an entry in the PIC for this packet. Each entry contains the packet ID, source address, next hop address, and expiration time. The packet ID is a 16-bit identification number to uniquely identify this packet among the packets of a specific source node. The source address is the IP address of the packet's source node. The combination of source address and packet ID uniquely identify a packet in the network. The next hop address is the MAC address of the next hop node, used to verify that the next hop has forwarded the packet. The expiration time indicates when the entry will be removed and the corresponding next hop will be punished for misbehavior. Depending on the behavior of the next hop node, i.e., whether it forwards the packet within the expiration time, the BM will deliver either a positive or negative behavior report regarding that next hop node to the RM.

The *Reputation Manager* (RM) maintains a reputation rating for each of the neighboring nodes. This reputation rating is used to determine whether the neighboring node should be used to forward packets. To manage the reputation ratings, the RM maintains a reputation table in which each record (i.e. row) includes the reputation information for a single neighboring node and consists of its MAC address, reputation value, and a redemption interval (which will be described below).

When a specific node sends or forwards a packet to a destination through a one-hop neighboring node X in a new installed route, it will create a new record for node X in the reputation table and assign it an initial reputation value $R_{Initial}$. Based on reputation reports received from the BM regarding the behavior of node X, the RM will update the reputation value of node X in the reputation table accordingly. If the report was positive, the reputation value will be incremented by an increment valueR_{Inc}, but it must not exceed a reputation ceiling value R_{Ceil}. If the report was negative, the reputation value will be decremented by a decre-

ment value R_{Dec}, but must not go lower than a reputation floor value R_{Floor}. After decrementing, if the reputation value of node X is lower than a preset reputation threshold value R_{Thr}, node X will be considered as a selfish node and an ALARM report will be sent to the PM indicating the detection of a selfish node.

Recognizing that a node could earn a bad reputation due to network congestion or by temporarily moving out of transmission range, a node that is marked as selfish is given an opportunity to redeem itself by restoring the node's reputation after a preset interval, referred to as the *redemption interval*, which is stored in the node's entry in the reputation table. After a node marked as selfish has been excluded for a time period that exceeds its redemption interval, the RM will redeem the node by setting the reputation of the excluded node back to the initial reputation value $R_{Initial}$. Another benefit of the redemption mechanism is to give those nodes which are no longer misbehaving a second chance to cooperate with other nodes. However, if the redeemed node returns to its former selfish behavior, it will be isolated again and its redemption interval will be doubled. In other words, the redemption interval grows in a binary exponential manner in order to produce a more severe punishment every time a node returns to selfish behavior.

The *Path Manager* (PM) helps in making sure that routing decisions pass through reliable nodes by ensuring that every next-hop in the routing table points only to nodes with good reputations, which leads to routing packets via a reliable path from the source node to the destination node. Based on the reputation information provided by the RM, the PM may discard data packets, reject offers to install a route, ignore requests to reach a destination, or even remove an existing route if it detects that the neighboring node is selfish.

The Path Manager has three main functions which can be summarized as follows:

- When receiving a data or control packet from a neighboring node, the PM consults the RM to learn whether the neighboring node is selfish. If it is, the packet will be discarded by the PM; otherwise it will be forwarded to the AODV mechanism to continue processing it as usual.

- When receiving an ALARM report from the RM that a neighboring node is selfish, the PM will remove every route where the next hop in the route is that selfish node. While removing any affected route containing references to some precursor nodes, the PM will trigger the RERR dissemination process originally included in the route maintenance mechanism of legacy AODV to inform the PM of each of the precursor nodes about the breakage of a link due to the detection of a selfish node, along with the affected destinations from this breakage. The precursor nodes are those neighbors that have at least one route in their routing table pointing to one of the affected destinations where the next hop is the node that sent the RERR message.

- When receiving a RERR message from a neighboring node, the PM will remove every route pointing to the affected destinations referenced by the RERR message, where the next hop in the route is the neighboring node that sent the RERR message. If there are precursor nodes in any removed route, the RERR message will be forwarded to those precursor nodes.

Table 1: Mobility parameters with typical values

Mobility Parameter	Value
Network area size	1250×1250 meters
Total number of nodes	50
Mobility model	Random WayPoint (RWP)
Speed of each node	Uniform distribution (0-20)m/s
Node pause time	0

While it is possible that any node can falsely claim that a neighboring node is selfish by sending a fake RERR message, this kind of malicious behavior exploits a vulnerability that already exists in the route maintenance mechanism included in the original AODV protocol. It is not within the scope of this research to solve that problem.

3.4 Performance Evaluation

To evaluate the performance of RAAODV, a number of simulations were implemented in NS3 [10]. Each experiment ran a total simulation time of 8,000 seconds and was repeated ten times, with the results averaged over the ten runs. The main goals of the experiments were to assess the performance of RAAODV under various network conditions and to determine the optimal combination of parameter values for RAAODV. The parameters can be divided into three major groups, namely, mobility parameters, wireless communication parameters, and RAAODV parameters. Mobility parameters are those parameters that affect the movement of nodes and thus the formation of the routes that connect the source nodes to the destination nodes. Mobility parameters, along with typical values, are shown in TABLE 1. Wireless communication parameters are those parameters that affect the volume of traffic being transferred over the network. TABLE 2 shows the wireless communication parameters, along with typical values. The RAAODV parameters and typical values for those parameters are included in TABLE 3.

The experiments were divided into two main groups, those with a static selfish behavior, i.e. where nodes do not change their behavior through the entire run, and those with dynamic selfish behavior, i.e. where certain nodes can switch their behavior between selfish and cooperative states during the simulation run. The first group of experiments evaluates the accuracy of detection for selfish nodes in a network with a topology that changes due to node mobility. The second group of experiments uses a similar network, but allows nodes to change their behavior to evaluate the impact of the redemption policy.

Experiments with *static selfish behavior* evaluate the performance of RAAODV when a certain percentage of nodes behave selfishly by always dropping any packet that is not addressed to them. All other nodes behave normally. Figures 2(a) and 2(b) show the impact on network Goodput and the Packet Loss Ratio (PLR) respectively when 10%, 30%, and 50% of nodes behave selfishly all of the time and the underlying routing protocol is legacy AODV without any reputation evaluation. It should be noted that the selection and placement of the selfish nodes is random for each of the ten simulation runs. It is clear that the performance of the network is significantly affected by the presence of such nodes. When 10% of nodes are selfish, the performance degrades

Table 2: Wireless communication parameters with typical values

Wireless communication parameter	Value
Number of senders	10
Number of receivers	10
Percentage of selfish nodes	10-30-50%
MAC protocol	IEEE802.11
Radio communication range	250 meters
Bandwidth	2Mbps
Traffic rate	CBR 0.512Kbps
Packet size	64 bytes

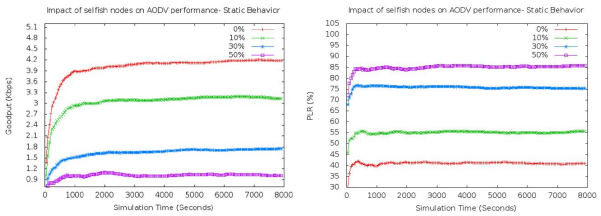
Table 3: RAAODV parameters with typical values

RAAODV parameter	Value
Behavior monitoring timeout	60 milliseconds
Reputation ceiling	1.0
Reputation floor	0.35
Reputation threshold	0.4
Initial reputation	0.5
Reputation increment	0.1
Reputation decrement	0.2

to about 75% of the value without any selfish nodes. The performance degrades to about 41.6% when 30% of nodes behave selfishly and to about 24% when 50% of them do so.

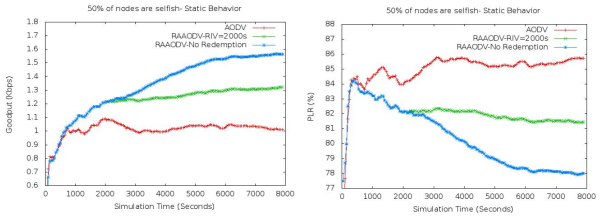
Figures 3(a) and 3(b) show how RAAODV enhances network performance in terms of Goodput and PLR respectively, compared to legacy AODV, when 50% of nodes behave selfishly all the time. As in figures 2(a) and 2(b), legacy AODV reaches a certain level of performance and remains at that level through the simulation run. But, after a âĂIJwarm-upâĂİ period of less than 1000 seconds in which RAAODV detects selfish nodes and excludes them from routes, RAAODV is able to improve on the performance of legacy AODV. The policy that RAAODV with redemption follows regarding selfish nodes does not differ from that of RAAODV with no redemption until the redemption decision is taken at the 2000th second. This explains the difference between the curves for RAAODV with and without redemption that begins around that time. Note that the network performance is worse for RAAODV with redemption than for RAAODV without redemption in a network where all nodes that behave selfishly do so all the time. This occurs because the redemption mechanism is initially giving the statically selfish nodes the opportunity to drop more packets before detecting them again as selfish. Over time, the binary exponential increase in redemption interval should cause RAAODV with redemption to do this less and less often. However, it would require longer simulation runs than those shown here to demonstrate that affect. Although RAAODV with redemption may allow selfish nodes to be included in routes again, leading to additional packet loss, it still results in an improvement in network performance of about 30% over legacy AODV, while this improvement reaches 54% for RAAODV with no redemption.

Experiments with *dynamic selfish behavior* evaluate the performance of RAAODV when randomly selected nodes switch their behavior from cooperative to selfish or vice versa during the simulation run. To implement this dynamic behavior, the simulation time is divided into periods of 500 seconds. At the beginning of the simulation, each of the selected nodes is assigned an interval based on a random multiple of 500 seconds. When this interval expires, the node will switch its behavior (e.g. from cooperative to selfish to conserve its low battery power). Then, it will be assigned another random multiple of 500 seconds, after which another behavior switch takes place (e.g. from selfish to cooperative after charging the battery). This process of switching behavior from selfish to cooperative and vice versa at random multiples of 500 seconds may occur many times during the simulation run. Consequently, this random selection will cause the number of nodes that are behaving selfishly to vary at different times during the simulation, making the task of detecting and isolating the selfish nodes more difficult. This also complicates the selection of optimum RAAODV parameter values, especially those related to the redemption mechanism.



(a) Goodput (b) PLR

Figure 2: Impact of nodes behaving selfishly all the time on AODV's network performance

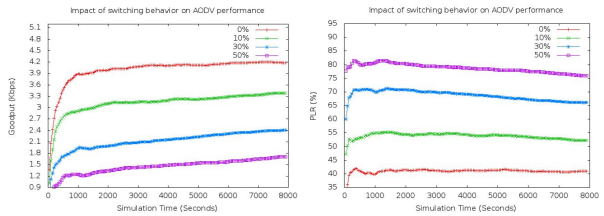


(a) Goodput (b) PLR

Figure 3: Impact of RAAODV on network performance when 50% of nodes behave selfishly all the time

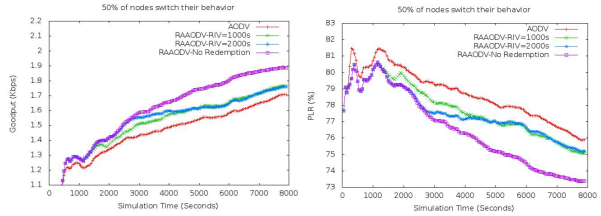
Figures 4(a) and 4(b) show how the network performance, in terms of Goodput and PLR respectively, is degraded when 10%, 30%, or 50% of nodes switch their behavior between selfish and cooperative during the simulation time and where the underlying routing protocol is legacy AODV without any reputation evaluation. It is clear that the performance of the network is significantly affected by the presence of such nodes. When 10% of nodes switch their behavior during the simulation time, the performance degrades to about 78% of its original value. This degradation reaches 56% when 30% of the nodes are selfish and 40% when the percentage of selfish nodes is 50%.

Figures 5(a) and 5(b) show how RAAODV enhances network performance in terms of Goodput and PLR respectively compared to legacy AODV when 50% of nodes switch their behavior between selfish and cooperative during the



(a) Goodput (b) PLR

Figure 4: Impact of nodes that switch their behavior on legacy AODV's network performance



(a) Goodput (b) PLR

Figure 5: Impact of RAAODV with and without redemption on network performance when 50% of nodes switch their behavior

simulation time. It can be noted that the performance of RAAODV with no redemption is more consistent during the simulation. This is because any node showing a selfish behavior at any time during the simulation will be isolated for the duration of the run. On the other hand, the performance of RAAODV using redemption with an initial timer value of either 1000 seconds or 2000 seconds fluctuates due to the difficulty of dealing with those nodes that are not static in their selfish behavior.

4. CONCLUSION

This paper has introduced a new variant of the AODV routing protocol called Reputation-Aware AODV (RAAODV) with the goal of mitigating the effects of selfish behavior in MANETs. Most solutions for this problem use the DSR protocol to take advantage of its source routing and many of them also employ second-hand information from neighboring network nodes as an additional source of reputation information. The design of RAAODV focuses on two main goals, namely, creating a scalable solution for detecting and mitigating selfish behavior in large-sized networks and avoiding the problems of false accusation, rumor spreading, and collusion that can result from propagating second-hand reputation information. RAAODV, with its simple mechanism, shows comparable results to other proposed solutions with the network performance improved by 54% above legacy AODV when as many as 50% of nodes behave selfishly all the time and about 12.5% above legacy AODV when up to 50% of nodes randomly switch their behavior between cooperative and selfish.

5. REFERENCES

[1] S. Buchegger and J.-Y. Le Boudec. Performance analysis of the confidant protocol. In *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, pages 226–236. ACM, 2002.

[2] S. Buchegger and J.-Y. Le Boudec. Coping with false accusations in misbehavior reputation systems for mobile ad-hoc networks. Technical report, KTH Royal Institute of Technology, 2003.

[3] S. Buchegger and J.-Y. Le Boudee. Self-policing mobile ad hoc networks by reputation systems. *Communications Magazine, IEEE*, 43(7):101–107, 2005.

[4] L. Buttyán and J.-P. Hubaux. Report on a working session on security in wireless ad hoc networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(1):74–94, 2003.

[5] P. Dewan, P. Dasgupta, and A. Bhattacharya. On using reputations in ad hoc networks to counter malicious nodes. In *Tenth International Conference on Parallel and Distributed Systems*, pages 665–672. IEEE, 2004.

[6] D. Djenouri, L. Khelladi, and N. Badache. A survey of security issues in mobile ad hoc networks. *IEEE communications surveys*, 7(4), 2005.

[7] D. Johnson, Y. Hu, and D. Maltz. The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. *RFC4728*, pages 2–100, 2007.

[8] S. Marti, T. J. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *International Conference on Mobile Computing and Networking*, volume 6, pages 255–265, 2000.

[9] P. Michiardi and R. Molva. Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Advanced Communications and Multimedia Security*, pages 107–121. Springer, 2002.

[10] NS3Team. NS3 Manual. `http://www.nsnam.org/documentation/`, 2013.

[11] C. Perkins, E. Belding-Royer, and S. Das. Rfc 3561-ad hoc on-demand distance vector (aodv) routing. *Internet RFCs*, pages 1–38, 2003.

[12] B. Wang, S. Soltani, J. SHAPIRO, and P.-N. Tan. Local detection of selfish routing behavior in ad hoc networks. *Journal of Interconnection Networks*, 7(01):133–145, 2006.