Preventing Selfish Node Behaviors in MANETs Through Hierarchical ARM

†Dr. R. Saminathan and ‡C. Jenithamary

‡ Assistant Professor, Annamalai University, Chidambaram, Tamil Nadu, India.
† PG Student, Annamalai University, Chidambaram, Tamil Nadu, India.

ARTICLE INFO
Article history:
Received 12 October 2014
Received in revised form 26 December 2014
Accepted 1 January 2015
Available online 25 February 2015

Keywords:
Mobile ad hoc networks
Price systems
Reputation systems

ABSTRACT
A hierarchical Account aided Reputation Management system (ARM) to efficiently and effectively provide cooperation incentives. ARM builds a hierarchical locality-aware distributed hash table (DHT) infrastructure for efficient and integrated operation of both reputation and price systems. The infrastructure helps to globally collect all node reputation information in the system, which can be used to calculate more accurate reputation and detect abnormal reputation information. Also, ARM integrates reputation and price systems by enabling higher-reputed nodes to pay less for their received services. ARM proposes an infrastructure called, DHT that helps to marshal all reputation and transaction information in the system of a given node to a specific manager. The managers perform two functions: reputation management and account management. Each manager calculates the reputations and increases/decreases the credits in the accounts of the mobile nodes for which it is responsible. Nodes with reputations below the threshold or deficit accounts are regarded as uncooperative nodes. Managers notify mobile nodes about uncooperative nodes, which are then put into blacklists. The blacklisted nodes’ forwarding requests are ignored by others. Like price systems, ARM also requires that the source node pays the relay nodes for packet forwarding, but it eliminates the need for credit circulation in the network. In the proposed system while selection of reputation manager and energy and degree of the node to considered along with parameter low mobility.

INTRODUCTION
A Mobile Ad hoc Network (MANET) is a continuously self-configuring, infrastructure-less network of mobile devices connected without wires. Ad hoc is Latin and means “for this purpose”. Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Each must forward traffic unrelated to its own use, and therefore be a router.

The primary challenge in building a MANET is equipping each device to continuously maintain the information required to properly route traffic. Such networks may operate by themselves or may be connected to the larger Internet. They may contain one or multiple and different transcivers between nodes. This results in a highly dynamic, autonomous topology. MANETs are a kind of Wireless ad hoc network that usually has a routable networking environment on top of a Link Layer ad hoc network. MANETs consist of a peer-to-peer, self-forming, self-healing network in contrast to a mesh network has a central controller (to determine, optimize, and distribute the routing table). MANETs circa 2000–2015 typically communicate at radio frequencies (30 MHz - 5 GHz).

Multi-hop relays date back to at least 500 BC (Jakobsson, M J. Hubaux, and L. Buttyan, 2003). The growths of laptops and 802.11/Wi-Fi wireless networking have made MANETs a popular research topic since the mid-1990s. Many academic papers evaluate protocols and their abilities, assuming varying degrees of mobility within a bounded space, usually with all nodes within a few hops of each other. Different protocols are then evaluated based on measures such as the packet drop rate, the overhead introduced by the routing protocol, end-to-end packet delays, network throughput, ability to scale, etc.

1. Related Works:
S. Marti, T. J. Giuli, K. Lai, and M. Baker et al, proposed this paper describes two techniques that improve throughput in an ad hoc network in the presence of nodes that agree to forward packets but fail to do so. To mitigate this problem, we propose categorizing nodes based upon their dynamically measured behavior. We use

Corresponding Author: Dr. R. Saminathan, Assistant Professor, Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar – 608 002, Tamil Nadu, India, Tel: 9443435653; E-mail: saimiaucse@yahoo.com,
a watchdog that identifies misbehaving nodes and a path rater that helps routing protocols avoid these nodes (Marti, S., T. J. Giuli, K. Lai, and M. Baker, 2000).

P. Michiardi and R. Molva et al, proposed countermeasures for node misbehavior and selfishness are mandatory requirements in MANET. Selfishness that causes lack of node activity cannot be solved by classical security means that aim at verifying the correctness and integrity of an operation. We suggest a generic mechanism based on reputation to enforce cooperation among the nodes of a MANET to prevent selfish behavior (Michiardi, P and R. Molva, 2002). Each network entity keeps track of other entities’ collaboration using a technique called reputation.

S. Buchegger and J.-Y. L. Boudec et al, proposed Mobile ad-hoc networking works properly only if the participating nodes cooperate in routing and forwarding. How-ever, it may be advantageous for individual nodes not to cooperate. We propose a protocol, called CONFIDANT, for making misbehavior unattractive (Buchegger, S and J.-Y. L. Boudec, 2003); it is based on selective altruism and utilitarianism. It aims at detecting and isolating misbehaving nodes, thus making it unattractive to deny cooperation.

Tiranuch Anantvalee and Jie Wu et al, proposed in a mobile ad hoc network, node cooperation in packet forwarding is required for the network to function properly (Anantvalee, T and J. Wu, 2007). However, since nodes in this network usually have limited resources, some selfish nodes might intend not to forward packets to save resources for their own use. To discourage such behavior, we propose a reputation-based system to detect selfish nodes and respond to them by showing that being cooperative will benefit them more than being selfish.

B. Buchegger and J. Y. LeBoudec et al, proposed in our approach, everyone maintains a reputation rating and a trust rating about everyone else that they care about. From time to time first-hand reputation information is exchanged with others; using a modified Bayesian approach we designed and present in this paper, only second-hand reputation information that is not incompatible with the current reputation rating is accepted (Buchegger, S and J. Y. LeBoudec, 2004) (Buchegger, S and J. Y. LeBoudec, 2003). Thus, reputation ratings are slightly modified by accepted information. Trust ratings (Wang, X. L. Liu, and J. Su, 2012) are updated based on the compatibility of second-hand reputation information with prior reputation ratings.

Prashant Dewan, Partha Dasgupta, and Amiya Bhattacharya et al, proposed the reputations of the nodes, based on their past history of relaying packets, can be used by their neighbors to ensure that the packet will be relayed by the node. This paper introduces a reputation scheme for ad hoc networks (Dewan, P, P. Dasgupta, and A. Bhattacharyas, 2004). Instead of choosing the shortest path to the destination, the source node chooses a path whose next hop node has the highest reputation.

This paper presents a protocol for routing in ad hoc networks that uses dynamic source routing (Johnson, D.B., and D. A. Maltz., 1996). The protocol adapts quickly to routing changes when host movement is frequent, yet requires little or no overhead during periods in which hosts move less frequently.

2. Methodology:

In this paper, we present the proposed system of Account-aided Reputation Management system (ARM). (Refaei, M.T.L., A. DaSilva., M. Eltoweissy., and T. Nadeem., 2010). We examine the abnormal reputation information to improve the effectiveness and efficiency of cooperation incentives.

The proposed system is divided into five modules.

2.1 Manager selection
2.2 Distributed Hash Table Maintenance
2.3 Reputation Management
2.3.1 Misreport Avoidance and False Accusation Avoidance
2.3.2 Collusion Avoidance
2.4 Distributor Reputation Manager Auditing
2.5 Performance Evaluation

Fig. 1 shows that the manager selection selects the reputation manager who maintains the individual nodes reputation value and account value. Manager selection takes the network as an input and produces the output as manager election. Distributed Hash Table (DHT) maintenance stores the maintained data of reputation manager, i.e., node id, reputation value and account value. DHT maintenance takes the node behavior as an input and produces the accounts of network node as output. Reputation manager checks the stored data of DHT to detect the falsified reports. Distributor reputation manager modifies the reputation value, account value and node id during auditing and dismiss the cooperative manager. In performance evaluation less system overhead and less energy consumption is obtained.

2.1 Manager Selection:

Aided reputation manager selects a number of trustworthy and low-mobility nodes as reputation managers. The reputation manager maintains the all nodes reputation value and finding the uncooperative node. Each manager contains number of node. Each node’s reputation value and account is maintained by the manager.
2.2 Distributed Hash Table Maintenance:

The DHT table contains each node reputation value and account information as well as node id. Each object or node is assigned an ID that is the hashed value of the object or node IP address using a consistent hash function. An object is stored in a node whose ID equals or immediately succeeds the object’s ID. The DHT provides two main functions, Insert (ID, object) and Lookup (ID), to store an object to a node responsible for the ID and to retrieve the object. Manager calculates the reputations of all nodes and increases/decreases the credits in the accounts of the mobile nodes for which it is responsible. Nodes with reputations below the threshold or deficit accounts are regarded as uncooperative nodes. Credit of each node referred as account. These three values maintained in the distributed hash table.

2.3 Reputation Management:

2.3.1 Misreport Avoidance and false Accusation Avoidance:

Particular node drop the packet in network. The node neighbors’ are reported to the manager. The nodes are cooperative but they are unable to transmit the requested data. Because that the node is affected in the traffic or thermal noise. In this situation ARM can easily solve this problem. Reputations of each node in a region are reported to one manager. A manager notices that all nodes in an area report low observed reputations, it temporarily ignores the reports to reduce the uncertainty of the reported information in order to avoid punishing nodes for failing to forward packets due to adverse network conditions.

Some misbehaving nodes may report a high reputation for an uncooperative node, and a low reputation for a cooperative node. Since all observed reputations of a node in a region are collected into a manager. Majority of the nodes responses will be accepted to the manager.

Thus, in order to reduce the effect of falsified reports, a manager filters the Ri’s that dramatically deviate from the average of Ri. The deviation of Ri n0 reported by node n0 about node ni is calculated as

\[ \Delta R_{n0} = \left| \frac{\sum R_{n0} - \sum R_{nj}}{|n|} \right| \]

Where n denotes the group of observer’s report Ri to the manager during time T. and |n| denotes number of nodes in the group. ARM sets a threshold δ for the deviation and ignores Ri n0 satisfying \( \Delta R_{n0} > \delta_{1} \). δ1 is determined based on practical rating values. If the values differ greatly, \( \delta_{1} \) should be set to a larger value. The manager m0 then calculates the local reputation value of ni in T denoted by \( R_{ni}^{m0} \)

\[ R_{ni}^{m0} = \frac{\sum R_{n0}}{|n|} \]

Then, the manager reports \( R_{ni}^{m0} \) to ni’s owner manager using \( \text{Insert}(i, R_{ni}^{m0}) \). The expected value of δ is where \( R_{1h} \) and \( R_{1f} \) denote the expected values of honest reports and false reports, and a, b respectively denote the number of honest reports and the number of false reports in interval T.

Where \( R_{1h} \) and \( R_{1f} \) denote the expected values of honest reports and false reports, and a and b respectively denote the number of honest reports and the number of false reports in interval T.

2.3.2 Collusion avoidance:

The nodes in group S and group D are the nodes in the transmission range of mk. The number of nodes in group D overwhelms group S. If the nodes in group D collude with each other to report low Ri for ni , then the
justified reports from group A are ignored by \( m_k \). This problem can be resolved by another filtering process at the Owner manager \( m_i \) that collects all \( R_i^{m_0} \) from different managers \( m_0 \). Again, \( m_i \) computes the variance of \( R_i^{m_0} \) based on \( \Delta R_i^{m_0} \) and ignores \( R_i^{m_0} \) with \( \Delta R_i^{m_0} > \delta g \). \( \delta g \) can be determined in the same way as \( \delta l \).

\[
\Delta R_i^{m_0} = \left| R_i^{m_0} - \frac{\sum_{m_1 \in m} R_i^{m_1}}{|m|} \right|
\]

Where \( m \) is the number of managers that report \( R_i \). After that, the global reputation of node is \( n_i \) calculated as:

\[
R_{gi} = \frac{\sum_{m \in m_0} R_i^{m_0}}{|m|}
\]

The colluders may be in the transmission range of different managers. In this case, if the colluders in the transmission range of the manager do not constitute the majority of the reporting nodes, the reported information from colluders is filtered out by the manager according to \( \Delta R_i n_0 \). Otherwise, the falsified information from colluders is filtered out according to \( R_i^{m_0} \) by the owner manager.

### 2.4 Distributor Reputation Manager Auditing:

A compromised reputation manager may modify nodes’ reputation values and (or) account values in two situations. First, a reputation manager misreports the reputation of a node to its owner manager in the local reputation calculation. Second, the owner manager of a node modifies the reputation value and account value of the node in the global reputation calculation.

In the first situation, since the nodes in the transmission range of a reputation manager always change, the local reputation values of a node can be collected by several reputation managers in an interval. After these managers report the collected reputation values of a node to its owner manager, the owner manager can detect the misbehaviors of the malevolent reputation managers using the collusion avoidance method.

In the second situation, as the owner reputation manager of a node calculates its final reputation value and manages its account value, if the manager modifies the reputation value, no other nodes can detect it. To handle this problem, we use redundant reputation managers for each node.

### 2.5 Performance Evaluation:

**Average system throughput:**

Throughput is the measurement of the number of packets passing through the network over a unit of time.

**System overhead:**

The ARM yields much less overhead than Price, which produces less overhead than Reputation. In ARM, since nodes only communicate with managers, the overhead is proportional to the network size. Though ARM needs to construct and maintain DHT infrastructure in node mobility, its total overhead is still lower than others. In Reputation, each node periodically exchanges reputation information between its neighbors, and then the reputation information of each node is flooded throughout the network, resulting in higher overhead. In Price, credit circulation in the network generates transmission overhead. Defenseless has the smallest amount of overhead as it does not have any cooperation incentive mechanism.

**Energy consumption:**

ARM still consumes much less energy than Reputation and Price because it reduces message exchanges among nodes. The amount of energy consumed for DHT maintenance. A manager consumes more energy since the number of neighbors of each manager increases.

### RESULTS AND DISCUSSION

NS2 is an open-source event-driven simulator designed specifically for research in computer communication networks. Since its inception in 1989, NS2 has continuously gained tremendous interest from industry, academia, and government. Now NS2 (Teerawat Issariyakul., Ekram Hossain., 2009) contains modules for numerous network components such as routing, transport layer protocol, application, etc.

In this simulation, we propose a Hierarchical Account-aided Reputation Management system (ARM) to efficiently and effectively deter selfish node behaviors and provide cooperation incentives.

Aided reputation manager selects a number of trustworthy and low-mobility nodes as reputation managers. Manager calculates the reputations of all nodes. Nodes with reputations below the threshold or deficit accounts are regarded as uncooperative nodes. Nodes with reputations above the threshold are referred as account. Node
id, reputation value and account value are maintained in the distributed hash table. In each region, the node with higher values is selected as manager head.

**Network Architecture:**

![Network Architecture](image)

**Fig. 2:** Network architecture.

The network architecture separate on 4 regions. Dark cyan color nodes indicated by region 1 and region 4. Dark magenta color nodes indicated by region 2 is shown in fig. 2. Dodgerblue color nodes indicated by region 3.

**Manager Selection:**

![Manager Selection](image)

**Fig. 3:** Manager Selection.

Cyan color is indicated by manger head is shown in fig. 3. Manager selection based on low mobility and trustworthy.

**Simulations And Result Analysis:**

**System Overhead vs Time:**

![System Overhead](image)

**Fig. 4:** System overhead.

Fig. 4 shows that x-axis represents the time and y-axis represents the system overhead. Red line plot denotes the system overhead which is the packets are the network other than the data packets.

**Throughput vs time:**

![Throughput](image)

**Fig. 5:** Throughput vs time.
Fig. 5 shows that x-axis represents the time and y-axis represents the throughput. Red line plot denotes the throughput which is the measurement of the number of packets passing through the network over a unit of time. Due to the attacker, the throughput will be low.

Conclusion:

Previous reputation systems and price systems in MANETs cannot effectively prevent selfish behaviors, and they also generate high overhead. In this paper, we propose a hierarchical Account-aided Reputation Management system (ARM) to efficiently and effectively deter selfish node behaviors and provide cooperation incentives. ARM intelligently combines a reputation system and a price system. It builds upon an underlying locality-aware DHT infrastructure to efficiently collect global reputation information in the entire system for node reputation evaluation, which avoids periodical message exchanges, reduces information redundancy, and more accurately reflects a node’s trust. ARM has functions for reputation management and account management, the integration of which fosters cooperation incentives and uncooperation deterrence. ARM can detect uncooperative nodes that gain fraudulent benefits while still being considered trustworthy in previous reputation systems and price systems. Also, it can effectively identify falsified, conspiratorial, and misreported information so as to provide more accurate node reputations. The complementary effects between the reputation system and price system effectively prevent nodes from manipulating policies in individual systems for benefits. In our future work, we will study distributed methods for choosing a manager.

REFERENCES


