Cooperative and Reliable Packet-Forwarding On Top of AODV

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science

by

Bracha Hod

Supervised by

Prof. Danny Dolev

School of Engineering and Computer Science
The Hebrew University of Jerusalem
Israel

December 8, 2005

Acknowledgments

I wish to express my deepest gratitude to Prof. Danny Dolev. I am grateful to him for his excellent guidance, inspiration, support and everlasting encouragement throughout the years of the study.
I would also like to express my sincere thanks to Dr. Tal Anker for his great guidance, ideas and continuous support during the work.
I am grateful to the members of DANSS Lab for their encouragement.
Finally, I wish to thank my parents for their unmeasurable support and love.

Abstract

Cooperative and reliable packet forwarding presents a formidable challenge in mobile ad hoc networks (MANETs), due to special network characteristics; e.g., mobility, dynamic topology and absence of centralized management. Lack of cooperation, due to misbehavior caused by selfishness or malice, may severely degrade the performance of the network.

Previous studies, relying on a reputation system, have developed solutions designed for Dynamic Source Routing (DSR) protocol. The difference between Ad hoc On-Demand Distance Vector (AODV) and DSR requires examination and modification of these schemes to apply them to AODV.

This thesis highlights various aspects of cooperation enforcement and reliability, when AODV is the underlying protocol. Furthermore, it presents a scalable protocol that combines a reputation system with AODV that addresses reputation fading, second-chance, robustness against liars and load balancing.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The self-organization, which characterizes MANET, combined with bandwidth constraints of the links and limited battery power, make the network vulnerable to many attacks, primarily on the link and the network layers. The assumption made by most ad hoc routing protocols - that every node is reliable and cooperative - does not exist anymore.

Various research studies have focused on increasing network trustworthiness. Most solutions use cryptographic primitives to address security attributes including availability, integrity, authentication, confidentiality, non-repudiation and authorization [1], [2], [3]. These solutions are not always suited to spontaneous networks; those networks that lack a priori relations. Furthermore, they do not enforce cooperation and cannot prevent selfish or malicious attacks in the packet-forwarding phase.

Recent approaches toward cooperation in MANET [4], [5], can be classified into two different categories: (a) schemes based on reputation system [6], [7], and (b) techniques derived from games theory [8], [9], [10]. This thesis deals with the first category, which contains three basic elements: misbehavior detection, misbehavior reaction and a reputation system that integrates between the parts. Our work addresses the several challenges in each of the elements, with a final goal of improving the network availability, reliability and robustness. Our solution does not assume any a priori relations between the nodes or any cryptographic usage.

1.1 Motivation

AODV [11], [12], [13] is one of the leading routing protocols adopted by IETF for MANET. It is an on-demand algorithm that builds routes between nodes, but only

as desired by source nodes, and maintains these routes as long as they are needed. AODV uses sequence numbers to ensure the freshness of routes. It is loop-free, self-starting, and scales to large numbers of mobile nodes.

Most of the research thus far has addressed selfishness and cooperation, assuming DSR [14] as the underlying protocol. In DSR, nodes access a significantly greater amount of routing information than AODV nodes, which enable their more rapid recovery from misbehavior. However, AODV surpasses DSR, in terms of storage and memory overhead [15], [16]. For this reason, it is more scalable, and suited for large networks. Thus, handling misbehavior with AODV is a more challenging task.

1.2 Thesis Contribution

Several solutions have been designed for AODV, most of which rely on explicit acknowledgment, rather than on observation [17], [18].

To our knowledge, this work is the first to combine a reputation system based on passive acknowledgments with AODV and to examine the scalability issues of such a solution. In addition, we analyze a situation of partial dropping, which was not widely handled in other schemes. Furthermore, we present the usefulness of a reputation system with an advanced liars model.

Fundamentally, our solution adapts and integrates several existing cooperation and reputation models from previous works, to a complete system with its own distinct qualities. Our scheme supports many features that were demonstrated in these areas: reputation fading, second-chance, robustness against liars and load-balancing.

1.3 Thesis Outline

The thesis is organized as the following. A short background is given in chapter 2. Chapter 3 introduces the related work that was done in this area. A review of the problem is described in chapter 4. Chapter 5 discusses several issues and challenges in a solution based on a reputation system. The main properties of our scheme are presented in chapter 6. The complete protocol flow is described in chapter 7. Chapter 8 deals with the simulation model. The simulation results and analysis are provided in chapter 9. Chapter 10 outlines the conclusions and future work.

Chapter 2

Background

This chapter provides some background information relating to the core of the thesis.

2.1 Mobile Ad Hoc Network

Mobile Ad hoc Network [19], [20], [21] is an autonomous, self-configuring system of mobile devices (laptops, smart phones, sensors, etc.) connected by wireless links. Each node operates not only as an end-system, but also as a router to forward packets. MANET does not require any fixed infrastructure, such as base stations. Therefore, it is an attractive networking option for connecting mobile devices quickly and spontaneously. MANET has several salient characteristics [22]:

- *Dynamic Topologies* - Nodes are free to move arbitrarily; thus, the network topology may change randomly and rapidly at unpredictable times.
- *Bandwidth-constrained* - Wireless links have significantly lower capacity than their hardwired counterparts. In addition, the realized throughput of wireless communication is often much less than a radio's maximum transmission rate, due to fading, noise, interference conditions, etc.
- *Energy-constrained* - Some or all of the nodes in a MANET may rely on batteries or other exhaustible means for their energy. For these nodes, the most important system design criteria for optimization may be energy conservation.
- *Limited Physical Security* - Mobile wireless networks are generally more prone to physical security threats than wired networks. The increased possibility of eavesdropping, spoofing, and denial-of-service attacks should be carefully considered.

- *Scalability* - Some envisioned networks may be relatively large (e.g. tens or hundreds of nodes per routing area). The need for scalability is not unique to MANETs, but in the light of the preceding characteristics, it is much harder to achieve scalability.

Mobile ad hoc networks were initially designed for military applications, but with the increase of portable devices as well as progress in wireless communication, MANET is gaining importance with increasing number of applications. It can be used for emergency and rescue operation, conferences and campus settings, airport and car networks and other more.

2.2 MANET Routing Protocols

Routing is one of the core issues in MANET. There have been many recent proposals of routing protocols for mobile ad hoc networks. MANET routing protocols can be classified into two main categories: (1) proactive or table-driven, and (2) reactive or on-demand. Proactive protocols periodically broadcast information across the network in a controlled flood. The information is used at each node to build a routing table. Reactive protocols were designed to cope with the low bandwidth in wireless networks. They decrease the amount of control overhead by initiating a route request only when it is required. This advantage, however, comes up with a price of delay when building new routes.

AODV and DSR are the most popular on-demand routing protocols for MANET. Both protocols share various properties, but many of their routing mechanisms are different.

2.2.1 AODV

AODV routing protocol [12] offers a quick adaptation to dynamic link conditions, low processing and memory overhead and low network utilization. It avoids problems (such as "counting to infinity") associated with classical distance vector protocols. Its functionality is divided into two phases: route discovery and route maintenance.

- *Route Discovery* - Route discovery is initiated by a source node broadcasts a route request (RREQ), when it desires a route to a destination for which it does not already have. Every node that receives a RREQ creates a short-lived

reverse route to the source with the next-hop being the node from whom the RREQ was just received. When a RREQ reaches the destination or a node with a valid route, that node responds with a Route Reply (RREP) which travels to the source along the reverse path. Each RREP contains a destination sequence number which is used to prevent routing loops and helps nodes determine the freshness of the information. All nodes that route the RREP to the source also make corresponding forward entries in their routing tables. On receiving the RREP, the source node starts sending data.

- *Route Maintenance* - HELLO messages may be used to detect and monitor links to neighbors. In such case, each node broadcasts periodic HELLO messages to all its neighbors. When a broken link is detected, either by a MAC layer acknowledgment or by not receiving HELLO messages, the detecting node sends Route Error (RERR) message to all predecessor nodes that use the broken link to reach their respective destinations. The RERR packet is propagated towards the source and the route is deleted from the routing table.

2.2.2 DSR

DSR [14] is a source-routing protocol with a similar route discovery process to AODV. In DSR, however, RREQ and RREP packets contain all the intermediate nodes' addresses, so once a RREP is received, the sender node knows the entire route to the destination. Each packet to be routed carries in its header the complete, ordered list of nodes through which the packet must pass. Nodes promiscuously listen to packets and use the information in the packets to learn about routes in the network. Since there can be many routes from a source to a destination, a source may receive multiple route replies from a destination. DSR nodes cache all these routes in their route cache for future use.

The route maintenance of DSR handles link breaks. If an intermediate node detects a link break, it reports an error back to the source, and leaves it to the source to establish a new route. Alternatively, the node may try a different path, if it has an alternate route cached.

2.2.3 Differences Between AODV and DSR

The primary differences between AODV and DSR are: (1) DSR sources determine the whole path to the destinations, while in AODV the routing decision is made hop by

hop; and (2) unlike DSR nodes, which can keep multiple paths in the routing cache, AODV nodes record the information of only a single route in the routing table.

These two features of DSR are useful for increasing path reliability and overcoming misbehaving nodes.

2.3 Trust and Reputation

Trust and reputation [23], [24], [25] play an important role in many disciplines, such as sociology, economics and computer science. They have been extensively studied and discussed, and many definitions have been proposed. In this work, we adopt the following definitions, based on [26]:

- *Trust* is a subjective expectation a node has about another nodes future behavior, based on the history of their encounters.
- *Reputation* is a perception that a node creates through past actions about its intentions and norms.

Trust is viewed from a local perspective, it is based only on direct experience. Reputation, on the other hand, is derived from: (1) direct encounters or observations, and (2) inferences based on information (rating) gathered indirectly.

A reputation system is a system in which the nodes who participate in it compute rating values and then advertise these values among the other nodes. There are three basic properties that a reputation system must have in order to operate properly [24]:

- Nodes must be long-lived, so that with every interaction there is always an expectation of future interactions.
- Ratings about current interactions are captured and distributed.
- Ratings about past interactions must guide decisions about current interactions.

An effective reputation system fulfills the following requirements: (1) accurate rating, using multiple evidences in its calculation. (2) rating correctness, in terms of reflecting the performance over time. (3) fast reaction to recent changes, by correct weighting of the past and current behavior. (4) robustness against several attacks, as liars and rating manipulations.

Chapter 3

Related Work

Recently, a lot of research has focused on the cooperation issue in MANET. Several related issues are briefly presented here.

3.1 Watchdog and Pathrater

Misbehavior detection and reaction are described in [27], by Marti, Giuli, Lai and Baker. The paper presents two extensions to the DSR algorithm: the *watchdog* and the *pathrater*. The watchdog identifies misbehaving nodes by listening promiscuously to the next node transmission. This technique is imperfect due to collisions, limited transmit power and partial dropping. However, according to simulations [28], it is highly effective in source routing protocols, such as DSR. The pathrater uses the knowledge from the watchdog to choose a path that is most likely to deliver packets. The path rating is calculated by averaging the rating of the nodes in the path, where each node maintains a rating for all the nodes it knows in the network.

Watchdog is used intensively in many solutions for the cooperation problem. The main drawback of this idea is that it enables selfishness and misbehaving nodes to transmit packets without punishing them, and thus encourages misbehavior.

3.2 CONFIDANT

Buchegger and Le Boudec present the *CONFIDANT* protocol and various enhancements in [29], [6], [30], [31] and [32]. The protocol defines a collection of components:

a monitor, a reputation system, a path manager and a trust manager. Each node monitors the behavior of its next hop neighbors in a similar manner to watchdog. The information is given to the reputation system that updates the rate of the nodes. Based on the rating, the trust manager makes decisions about providing or accepting route information, accepting a node as part of a route and so on. When a neighbor is suspicious in misbehaving, a node informs its friends by sending them an ALARM message. If a node's rating turns out to be intolerable, the information is relayed to the path manager, which proceeds to delete all routes containing the intolerable node from the path cache. Enhancement of the basic protocol is presented in [31] and provides a strong reputation system that deals well with false reputation. The model uses a modified Bayesian approach and introduces two new mechanisms: re-evaluation and reputation fading for letting a node redeem itself and for preventing a sudden exploitation of good reputation accordingly.

Our solution adapts some basic mechanisms from this work, but constructs different reputation properties and misbehaving reaction for better suiting to AODV. We also deal with partial dropping and advanced liars, which are not extensively addressed in CONFIDANT.

3.3 CORE

Michiardi and Molva propose the *CORE* scheme and various related issues in [7], [33] and [34]. In this scheme, every node computes a reputation value for every neighbor, based on observations that are collected in the same way as watchdog. The reputation mechanism differs between subjective reputation, indirect reputation, and functional reputation. Subjective reputation is calculated directly from neighbors past and present observations, giving more relevance to past observations in order to minimize false detection influence. Indirect reputation is the information collected through interaction and information exchange with other nodes using positive values only. Functional reputation is the global reputation value associated with every node. By avoiding the spread of negative rating, the mechanism resists attacks, such as denial of service. When a neighbor reputation falls below a predefined value, the service provided to the misbehaving node is suspended.

There is a formal proof of CORE properties, based on game theory, but no simulations were done to prove the usefulness of such a system.

3.4 OCEAN

Banal and Baker propose *OCEAN* [8], a scheme for robust packet-forwarding. OCEAN, similarly to previous schemes, is based on nodes' observations. In contrast to previous mechanisms, no rating is exchanged and every node relies on its own information, so the trust management is avoided. The rating is based on a counter that counts the positive and the negative steps a node performs and based on a faulty threshold, the node is added to a faulty list. In the method for route selection, a DSR node appends an avoid list to every generated RREQ and a RREP based on this list. A second-chance mechanism is provided to give nodes that were previously considered misbehaving another opportunity to operate.

OCEAN simulations concludes that a scheme which relays only on first-hand observation performs almost as well and sometimes even better than a scheme that also relies on second-hand information. Our simulation, which uses a low faulty threshold together with full rating exchange, comes into a different conclusion when AODV is the routing protocol.

Chapter 4

Problem Statement

The special properties of MANET drive many misbehavior types; AODV is vulnerable to various kinds of attacks, as described in [35]. When dealing with packet-forwarding, there are several kinds of availability and integrity attacks we consider [36]: dropping (complete or partial), misrouting, modification and fabrication. In this work, we focus on the first attack, which is the most common attack in MANET. Though, detection and reaction of the other attacks are also possible with our scheme.

There are two main motivations which encourage nodes to misbehave: selfishness and malice.

4.1 Selfishness

The limited battery-power, one of MANET characteristics, encourages nodes to use the network for their own communication only, and not for the benefit of other nodes. Refer to AODV, the following selfish behaviors are considered [34]:

- *Node type A* participates in the routing protocol, but may drop part or all the data packets that do not belong to it. This node is interested in saving its battery power, as well as having the capability to receive and transmit its own packets. By forwarding only control packets, it has full information about the available paths, without the cost of data packets transmission. This behavior pattern is a subtype of a gray or black hole, where a node responds positively with a RREP message, even if it does not intend to forward data packets.

- *Node type B* participates in the route maintenance phase, but does not adequately take part in the route discovery phase; The node does not transmit RREQ or RREP messages that are not originated by it, so packets do not pass through it. This may happen, for example, when a node has all the necessary routes, and further information is worthless for it. In this case, the node maintains the existing routes, but it avoids from originating routes for the benefit of other nodes. This type of node is better than a node of type A, as in existence of multiple paths from a source to a destination, an alternative path will be discovered. This behavior does not cause severe damage, but it increases the unfairness in the network and may cause unbalanced load.

- *Node type C* enters to idle status most of the time and does not even send HELLO messages to its neighbors, so they are not aware to its existence. Only when it wishes to communicate with other nodes, it starts the routing protocol. This behavior, called "sleep period operation" [22], is a legitimate behavior, but such a node may not adequately contribute to the network, and nodes that give more should get a better service. According to [37], it is not possible to enforce a node to forward more packets than it sends on average. The case that has to be prevented is a situation in which a node of type C transmits a large amount of packets in a short period, without proportion to the amount of packets it forwards.

- *Node type D* usually performs the routing and the forwarding properly, but when its energy falls under some threshold or in case of temporary overload, it may act as nodes of type A, B or C.

It is important to note that selfish nodes do not intend to damage other nodes. Moreover, they usually do not maliciously cooperate with other nodes because such cooperation requires additional resource usages that they wish to minimize.

4.2 Malice

Malicious nodes aim to damage other nodes without considering their own gain or their battery life as a main concern.

- *Black Hole* is a node that uses the routing protocol to advertise itself as the shortest path to nodes whose packets it wants to intercept. The node can

explicitly send a RREP, or avoids RERR transmission when a link is broken. All the data packets that a black hole gets are dropped.

- *Gray Hole* adversary selectively drops some kinds of data packets but not other. Naturally, a detection of gray hole is more difficult than a black hole detection, because of its ambiguous behavior.

The misbehavior patterns described above contain different specific behaviors inside them. For example, misrouting attack or control packet modification, can be considered as a black hole operation.

In contrary to selfish nodes that do not seek to cooperate, cooperation between malicious nodes is a widespread scenario. One common attack using joint effort is a *wormhole attack* [38]. In such an attack, a node tunnels packets to another node through a private network. For instance, it can send a RREQ packet that will arrive faster than other RREQ packets and thus prevents other routes from being discovered. When the route is constructed, the node drops the data packets to damage other nodes.

Cooperation between malicious node is a very hard problem that cannot be solved without security primitives, so we do not face this problem in this framework and assume no malicious cooperation.

Chapter 5

Issues and Challenges

There are several issues and challenges when designing a cooperative and reliable packet-forwarding scheme on top of AODV. First, it is essential to decide about a detection method: whether to use a passive or an active acknowledgment mechanism. Second, a reputation system must be designed carefully to represent the rating values accurately and to exchange them appropriately. Its robustness against false rating, rating misuse and other attacks is also important. At last, the reaction part includes some issues: whether to use unipath or multipath, how to enforce a desire behavior and how to provide a proper service based on activity. This chapter discusses these issues.

5.1 Misbehavior Detection

Intrusion detection is a wide research area that has been dealt with in various papers, such as: [39] and [36]. Packet dropping can be detected with either passive or active acknowledgment.

5.1.1 Passive Acknowledgment

Watchdog mechanism, a passive acknowledgment technique, is a successful method, but it was designed mainly for DSR. The mechanism assumes that a promiscuous mode is supported by the wireless interfaces, but this assumption is not always true in AODV. For example, nodes in multi wireless networks cannot hear their neighbors forwarding, due to different modulations.

Additionally, AODV nodes are not aware to the further hops after their 1-hop neighbors, so they do not know if their packets were forwarded through the right path. Some essential modifications in the mechanism, like using additional next_hop field in the route entries, are required to applied it for AODV, as described in [40].

Besides the partial incompatibility to AODV, the main drawback of watchdog is that it does not robust against adversary collaboration.

5.1.2 Active Acknowledgment

Active acknowledgment method does not assume any special network characteristics and it is more effective when malicious nodes cooperate. This method uses explicit acknowledgments in order to discover packet dropping. The acknowledgment can be either from a source to a destination, or only through part of the path.

End-to-End Acknowledgments

Network layer protocols usually rely only on MAC layer acknowledgments and do not promise a reliable forwarding from end-to-end. A scheme presented in [17] uses end-to-end acknowledgments in the network layer to notify about packets delivery between peer nodes. When a destination node receives a packet, it sends back to the source node an acknowledgment message that can be encrypted to avoid forgery. If the source node does not receive any acknowledgment before a specific timeout, it assumes that the packet did not reach to the destination.

This scheme identifies unreliable paths better than watchdog, but has several disadvantages. First, end-to-end acknowledgment is not suitable for traffic over UDP, because of the high overhead and delay associated with it. The encryption to avoid forgery also increases the overhead. At last, this method finds out untrustworthy paths, but does not detect the misbehaving nodes along them.

Probing Packets

In the probing technique [41], every node proactively monitors the forwarding behavior of other nodes. For example, if some node A wants to check if node B performs its forwarding correctly, it sends node C, a neighbor of B, a probe message. If node C gets the message, it sends an acknowledgment to node A. Getting the acknowledgment confirms node A that node B forwards the other packets properly.

This scheme also suffers from some drawbacks. First, the detection here is costly, because of the explicit acknowledgments transmission. Second, it is less effective than using end-to-end acknowledgments, e.g. in case of partial dropping. Last and the most important, this method does not suit to AODV environment, since it requires information of the nodes along the path, information that do not exist in the protocol.

Detection of cooperative black holes is addressed in [33] with a similar idea to the probing packets. Every node holds Data Routing Information (DRI) table with information of whether a node succeeded in forwarding data packets through some other node and if he succeeded in getting data packets from that node. Answer 'no' to both entries makes the corresponding node a suspect for being a black hole. By cross checking of the DRI tables along the path to the destination, the black holes can be detected. The protocol overcomes the lack of information in AODV by adding additional headers.

On-Demand Secure Byzantine Routing (ODSBR) [42], [43], combines the two ideas and uses end-to-end acknowledgment from the destination to detect the presence of black holes. When an attack is detected, ODSBR enters to a probing mode in order to discover the attack location. This combination is very effective, but still has the disadvantage of high overhead and latency.

As one can see, the lack of knowledge of AODV nodes about the nodes along the paths makes the detection harder, especially when using explicit acknowledgments. AODV-PA [44] extends AODV with source accumulation feature of DSR, to improve the protocol performance. The implementation of this extension is during the route discovery phase. Each node appends its own address on the RREQ and RREP messages and updates its routing information with the information it obtains from the messages. Simulations show that AODV-PA improves the performance of AODV under some conditions, but not to all the networks.

5.2 Reputation System

Reputation system is one approach to deal with misbehaving nodes. Such a system assumes that better detection and reaction are achieved using nodes collaboration, than when they are solely performed by one node.

5.2.1 Rating Values

Rating values represent the trust level of a node, based on its behavior along time, and can be applied to various actions. Often, the rating is derived from both direct and indirect knowledge, known also as first-hand observation and second-hand observation accordingly. Both the original CONFIDANT protocol and CORE scheme use some weights on the reputation values of different actions to obtain a single combined value per node. This is wrong, because nodes may forward packets well, but act faultily in the rating exchange, and in the opposite direction, so a combined value may not reflect an accurate rating.

A comprehensive analysis [45] on the rating values of CONFIDANT and CORE, comes into a conclusion that both positive and negative rating should be used and advertised, in order to obtain effective results for both well-behaving and misbehaving nodes. The rating is not calculated upon a single observation, but only after some threshold, when the node's behavior can be determined in high precision. Emphasis on past behavior should be limited to avoid misbehaving nodes taking advantage of it. On the other hand, if a node is unable temporarily to perform the forwarding, it should not be penalized severely. The function for rating calculation has a significant effect on the robustness and the efficiency. However, it must not be too complex because of the limited resources.

5.2.2 Rating Exchange

Rating exchange in MANET is derived from its unique characteristics. The transmission cost affects on the frequency and the range of dissemination, towards a local and limited scheme. Nodes' mobility, on the other hand, encourages a global model for better performance.

Frequency

AODV is a reactive protocol with a limited proactive part in the route maintenance (HELLO messages). Rating exchange may use either proactive or on-demand approach. Proactive distribution shares the information continuously, even when there are no extraordinary events. On-demand manner uses information exchange only when a misbehaving node is discovered, or when some suspicions arise.

In a situation of rare or low probability that misbehaving nodes exist, the on-demand approach has an advantage over the proactive method. Unfortunately, selfish

behavior is a widespread phenomenon in spontaneous networks, so the proactive procedure is perhaps better. Yet, a proactive method increases the control messages' volume and the transmission cost. It also may damage the reactive property of AODV, which gives AODV its scalability.

Range

The communication volume of a reputation system depends not only on its distribution frequency, but also on its range. A global model spreads the information through the entire network, while in a local model the information exchange is performed within a restricted region.

Maintaining a global knowledge of the dynamic network memberships can give the ability to effectively punish misbehaving nodes. However, the communication and storage overhead is too high for mobile nodes and does not scale to large networks [46]. A global model is also inadequate because it might enhance the attack possibilities and increase the model problems, such as false information.

A local model, conversely, is more feasible and scalable. Still, the exact range (e.g. 1-hop, 2-hops etc) of dissemination should be determined to balance between the transmission cost and the information benefit.

Mobility and Long-living

The mobility of nodes in MANET has a great influence on the effectiveness of a reputation system . In a dynamic network or a large area with local rating exchange, the long-living property of a reputation system may not be applied.

Two scenarios that may happen in such networks are: (1) a node might not have enough time to discover misbehaving nodes or to punish them. (2) a misbehaving node may act faultily in a region, and while detected by its neighbors, it can move to a new area, where nobody knows it. Therefore, the decision about the rating exchange manner has to take into account nodes' mobility.

5.2.3 Weaknesses and Vulnerabilities

Besides the difficulty to set an efficient and scalable schema in MANET, reputation systems are generally vulnerable to a wide range of attacks.

- *Bad Participation*, due to selfishness or malice drives, is a problem that may occur in the reputation protocol as in the routing protocol.
- *False Rating*, if it is either false accusation or false praise, has the ability to prevent service from nodes. False accusation may cause innocent nodes' repudiation, while false praise has a negative effect by overloading a node because it seemed excessively good.

  The main idea of robust reputation model, as presented in [31] and analyzed in [47], is to consider rating messages only when they come from trusted nodes or when they are close enough to the node's own rating. In addition, it is necessary to limit the indirect rating influence in the total rating.
- *Positive Rating Misuse* is an attack when a node builds up a good reputation and then it behaves maliciously for a period that its reputation is still positive. This attack is common mainly in CORE scheme because it gives relative highly emphasis on past experience.
- *Identity Issue* is a weak point in reputation systems. Ability to change identities easily, impersonating and other identity misuse may significantly degrade the system effectiveness. Unfortunately, this issue is still not addressed properly and most of the solutions so far assume that every node has a single unique identifier that cannot be used by other nodes or be changed easily.

5.3 Reaction

Misbehavior reaction is a set of actions that a node performs to overcome the problems caused by misbehaving nodes. A node wishes to improve its throughput by selecting reliable paths. Moreover, it intends to enforce cooperation by providing service according to the nodes' behavior.

5.3.1 Path Selection

The hop-by-hop routing in AODV gives it the scalability, but at the same time it hardens the construction of reliable paths. Nodes can estimate the path reliability only according to the next hop, while DSR nodes can choose a path based on multiple nodes' rating along the path. Furthermore, the single path property of AODV makes it more difficult to overcome unreliable paths.

When unreliable path is discovered, it is necessary to repeat the route discovery phase, in a similar way to link-failure detection or to RERR message receiving. The re-discovery phase should now ignore any RREP received from the unreliable node. A Local repair technique [12], [48], can be done to improve the network performance. However, when the route re-discovery is too frequent, AODV presents a poor behavior. The repeated re-discovery causes huge routing overhead and data transfer interruption, resulting in serious performance degradation.

Increasing path's reliability while maintaining only a single path is possible by selecting the next hop of a route from several intermediated nodes who transmit a request, instead of selecting the first node among them. Such a solution involves high latency, ignores network utilization and does not keep AODV property of selecting the shortest and the most unloaded path.

Multipath extension to AODV [49] is discussed and implemented in various papers, with a goal of minimizing single path problems, improving the throughput and increasing the load balancing, reliability and fault-tolerance. Part of the extensions are for backup route and other for load balancing.

AODV-BR (for Backup Routing) [50] extends AODV by establishing alternate paths during the route reply phase. When a node that is not a part of a route overhears promiscuously RREP packets, it records the transmitter neighbor as the next hop to a destination in its alternate route table. In case of multiple RREPs, it chooses the best route (the shortest one). When a node detects link failure, it broadcasts a route update message, so its neighbors activate a backup route. The protocol does not perform well under heavy traffic networks and the route selection is limited within one hop distance.

Ad-hoc On-demand Multipath Distance Vector (AOMDV) [51] computes multiple paths during route discovery. It enables node-joint paths, by an additional field of first hop in the RREQ. In addition, each node keeps a track of the list of the source's neighbors. The link-disjoint paths are constructed by replying to various unique neighbors, where the intermediate nodes take different reverse paths. AOMDV is a very sophisticate protocol that have some drawbacks. First, some good routes can be missed because strong constraints of the rules. Second, backup routes are expired due to lack of backup routes maintenance.

Ad-hoc On-demand Distance Vector Multipath (AODVM) [52] is a protocol that enables computation of multiple node-disjoint paths. Intermediates nodes records duplicate RREQ packets in a RREQ table and they are precluded from sending

RREP message directly to the source. The destination sends multiple RREP packets with an additional field to indicate the neighbor from which the particular RREQ arrived. When an intermediate node receives a RREP packet, it deletes the entry corresponding to this neighbor, adds a route entry to its routing table and sends the RREP to the neighbor with the shortest path to the source, taken from the RREQ table. Then, it deletes that neighbor entry from the RREQ table and removes every node that it overhears broadcast a RREP message. The number of paths is very limited when the network density is not high and when the distance between a source and a destination increases, the number of disjoint paths decreases.

5.3.2 Punishment and Reward

There are two ways to enforce a desire behavior in the network [45]: punishing misbehaving nodes or encouraging well-behaving nodes. Commonly, the nodes are more sensitive to punishment than to rewards, so we focus on effective punishment more than on reward.

Punishment of the misbehaving nodes (which do not forward packet properly) is done by dropping all their packets - both control and data packets. The more nodes that identify a misbehaving node and punish it, the more useful the punishment is. A question that arises is whether to accept rating information from such nodes, or just ignore it.

Traffic of misbehaving nodes, which pass through intermediated good nodes who are not aware to the misbehavior, is also an open issue that should be decided. An appropriate punishment would drop the misbehaving node's traffic, whether it is obtained directly or indirectly. Such a policy, however, may cause suspects in well-behaving nodes.

Punishing liars is another issue. It is reasonable to penalize nodes that do not report honestly, to encourage proper information distribution. However, it may discourage nodes from reporting on misbehaving nodes that have not been detected yet. Punishment of liars is commonly implemented by ignoring their reports. It may also be enhanced to packet dropping, but then the problem of incorrect suspicions arises again.

Chapter 6

Properties of the Scheme

This chapter specifies the main features and properties of our scheme, based on the issues previously described.

6.1 Observation Technique

Our scheme detects anomalous behavior using neighbors observations by the passive acknowledgment mechanism, as in [27]. This is less costly and more appropriate to AODV. A transmitting node verifies successful unicast forwarding upon receipt of link-layer acknowledgement from the receiver. Then, it observes its neighbors' behavior by overhearing, either in direct mode (getting packet explicitly) or via promiscuous mode. By examination of the overheard packets, the node is able to confirm its neighbors' good behavior.

Packet examination differs between control and data packets. Control messages are checked thoroughly to detect malicious modification since vulnerable fields, such as hop count and sequence number, may cause AODV malfunction. Data packets are compared by their IP header only, since we mainly focus on correct forwarding.

6.1.1 Observation Weaknesses in AODV

Besides the known problems of the passive acknowledgment technique, it may also cause mistakes in nodes' evaluations in several situations. For instance, a node that receives a RREQ packet with a time-to-live equals to zero processes this packet but does not transmit it. Subsequent packets with a larger time-to-live are not processed

and transmitted. As a result, the node might be suspected as misbehaving in such a situation. Another case that may happen frequently is that during a local repair a node drops buffered packets because of timeout, so it is considered mistakenly as misbehaving.

6.1.2 Discussion

DSR has a great advantage over AODV when it uses overheard packets in a promiscuous mode to determine connectivity and to discover routes, therefore saving bandwidth and reducing power consumption. Since the monitoring method in our scheme examines every overheard packet, further processing of the control packets may be very useful with no significant addition of CPU consumption. Implementation and analysis of this issue remains for future work.

In highly reliable or very loaded system, the observations can be performed once for multiple packets, in order to save resources.

6.2 Reputation System

Rating representation and exchange are the main properties of a rating scheme, since they characterize the system's flexibility, robustness, and effectiveness.

The rating in our scheme is represented by a 32-float value in the continuous range [-1,1]. Use of a positive to negative range enables both reward and punishment. A continuous range is used in order to get maximal precise, but it comes with the cost of float value calculation, which is higher than integer values.

6.2.1 Neighbors Rating

Calculation and management of neighbor rating is done using the Beta distribution function [53], [54]. The Beta function is commonly used to represent probability distributions of binary events. It is defined as:

$$P(x) = \frac{(1-x)^{\beta-1}x^{\alpha-1}}{B(\alpha,\beta)} = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}(1-x)^{\beta-1}x^{\alpha-1}$$

$$\text{where } 0 \leq x \leq 1 \ , \ \alpha > 0 \ , \ \beta > 0$$

Given a process with two possibilities $\{x, \bar{x}\}$, the Beta function estimates the probability of x, based on past observations of x and $\bar{x}$, and by setting:

$$\alpha = 1 + observed\ number\ of\ x$$
$$\beta = 1 + observed\ number\ of\ \bar{x}$$

A node's behavior resembles a binary process. The amount of positive events over a given period are related to x, while negative events are related to $\bar{x}$ accordingly. It is possible to assign variable weights to various events; e.g., greater weight to data packet dropping than to control packet dropping.

Using the derived reputation function and its scaling, given in [53], we denote the direct rating of a node j by its 1-hop neighbor i, based on observations as:

$$DR_{i,j} = \frac{p_{i,j} - n_{i,j}}{p_{i,j} + n_{i,j} + 2} \tag{6.1}$$

$$\text{where } p_{i,j} = recent\ positive\ actions\ of\ j\ observed\ by\ i$$
$$n_{i,j} = recent\ negative\ actions\ of\ j\ observed\ by\ i$$

Past behavior is an integral part of the rating. The rating can be defined accordingly, as:

$$DR_{i,j}(t) = \frac{p_{i,j}(t) - n_{i,j}(t)}{p_{i,j}(t) + n_{i,j}(t) + 2} \tag{6.2}$$

$$\text{where } p_{i,j}(t) = \gamma p_{i,j}(t-1) + p_{i,j}(\Delta t)$$
$$n_{i,j}(t) = \gamma n_{i,j}(t-1) + n_{i,j}(\Delta t)$$
$$\gamma = weight\ of\ past\ behavior\ \ ,\ 0 \leq \gamma \leq 1$$

Attacks of positive ratings misuse can be limited by giving more weight to the recent behavior than the past behavior, expressed by a small γ. Our computation uses the entire history, but as time progresses the impact of old history is diminished. This technique of fading allows effective rating in high mobility network.

$DR_{i,j}(t)$, as defined in equation (6.2), is the rating value published in the reputation protocol.

The total rating, expressed by $TR_{i,j}$, combines the direct rating $DR_{i,j}$ with reputation information from a set of 1-hop neighbors K, denoted by $DR_{k,j}$ for every $k \in K$. K is defined as a set of neighbors that are either evaluated as trusted, or their rating report passes the deviation test, as proposed in [31]. These conditions make the system robust against some types of liars, but do not perfectly prevent smart

liars. The deviation test of a node i checks that the difference of a given rating value $DR_{k,j}$ from the expected rating value $TR_{i,j}$ is not too large. The test is formulated as:

$$\begin{array}{ll} if\ |TR_{i,j} - DR_{k,j}| \leq \Delta & accept\ DR_{k,j} \\ otherwise & reject\ DR_{k,j} \end{array} \tag{6.3}$$

There is no synchronization between the nodes, so we do not define those values with time dependency. A node i may calculate the total rating at time t, either with $DR_{k,j}(t-1)$ or with $DR_{k,j}(t)$.

Combination of direct and indirect rating can be done easily by accumulation of the direct and indirect positive and negative actions, as described in [53]. However, the rating distribution includes one float value, since distribution of two values that represents $p_{k,j}$ and $n_{k,j}$ is much too expensive in terms of storage and bandwidth. Thus, it is possible to define a weight, denoted by w, such that $p_{k,j} + n_{k,j} = w$. Using the given w, a node can estimate $p_{k,j}$ and $n_{k,j}$ as the following:

$$p'_{k,j} = \frac{w(1+DR_{k,j})}{2}, \quad n'_{k,j} = \frac{w(1-DR_{k,j})}{2}$$

so that the total rating is defined as:

$$TR_{i,j}(t) = \frac{p'_{i,j}(t) - n'_{i,j}(t)}{p'_{i,j}(t) + n'_{i,j}(t) + 2} \tag{6.4}$$

$$\begin{array}{c} \text{where } p'_{i,j}(t) = \delta p'_{i,j}(t-1) + p_{i,j}(\Delta t) + \sum_{k \in K} p'_{k,j} \\ n'_{i,j}(t) = \delta n'_{i,j}(t-1) + n_{i,j}(\Delta t) + \sum_{k \in K} n'_{k,j} \end{array}$$

w represents the weight that the node scores, which is a tradeoff between robustness and second-hand information usage. The bigger w is, the more influence surrounding neighbors have, as well as the vulnerability of the system due to false information. On the other hand, very small w might make the whole reputation system irrelevant, since the effect of distributed information is negligible.

Different weights may be assigned to nodes' reports, based on trustworthiness. In the current simulations we decided to give equal weight value to all nodes. This value depends on the number of neighbors, in order to bound the effect of the indirect information over the direct rating, when there are many neighbors.

Since the rating combination is both commutative and associative and we give the

same weight to all the nodes, $p'_{i,j}(t)$ and $n'_{i,j}(t)$ can be defined alternatively as:

$$p'_{i,j}(t) = \delta p'_{i,j}(t-1) + p_{i,j}(\Delta t) + \frac{w}{2}(|K| + \sum_{k \in K} DR_{k,j})$$
$$n'_{i,j}(t) = \delta n'_{i,j}(t-1) + n_{i,j}(\Delta t) + \frac{w}{2}(|K| - \sum_{k \in K} DR_{k,j})$$

While most reputation systems use only a total rating to decide about nodes' behavior, our scheme uses two values: the total rating $TR_{i,j}(t)$ and the total number of negative actions $n'_{i,j}(t)$. Using both values helps in reflecting the performance over time more accurately. For example, behavior of two nodes with a neutral rating, one because it is new and one because it is inconsistent, can be discovered by their recent negative actions.

6.2.2 Remote Nodes Rating

Holding full information about the nodes along the path is neither feasible and nor scalable in AODV. Our simulations show that managing rating even for 2-hop nodes is not worthwhile. The mobility of the nodes makes this information relevant, but as the information tables grow, more overhead and latency are involved. This decreases significantly the scalability, which is an essential property of our scheme.

6.2.3 Trust

Misbehaving nodes might spread false rating information to obtain their own benefit. There is no direct correlation between behavior in the routing protocol and in the rating protocol. Therefore, it is essential to maintain information about the trustworthiness of the nodes and the estimation of the rating reports reliability. The amount of recent belief on a node j by a node i can be expressed as:

$$T_{i,j} = \frac{t_{i,j} - f_{i,j}}{t_{i,j} + f_{i,j} + 2} \tag{6.5}$$

where $t_{i,j} = recent\ true\ reports\ of\ j\ received\ by\ i$
$f_{i,j} = recent\ false\ reports\ of\ j\ received\ by\ i$

The trust as a variable of time is defined as:

$$T_{i,j}(t) = \frac{t_{i,j}(t) - f_{i,j}(t)}{t_{i,j}(t) + f_{i,j}(t) + 2} \tag{6.6}$$

$$\begin{gathered}\text{where } \; t_{i,j}(t) = \mu t_{i,j}(t-1) + t_{i,j}(\Delta t) \\ f_{i,j}(t) = \mu f_{i,j}(t-1) + f_{i,j}(\Delta t) \\ \mu = weight\ of\ past\ belief \;\; , \; 0 \leq \mu \leq 1\end{gathered}$$

If the reported rating is close enough to the estimated rating, then the number of true observed reports is incremented, otherwise the number of false reports is incremented.

A fading mechanism as time progresses is performed in the same way as in the direct rating, defined in equation (6.2). Each node maintains its own trust map, so trust values are not exchanged between the nodes.

6.2.4 Rating Exchange

The reputation distribution is performed continuously, when both good and bad ratings of 1-hop active neighbors and the misbehaving nodes who are on the black list, are broadcast.

Broadcasting is a very useful and frequent mechanism in MANET. The flooding is a fundamental tool to propagate control messages in order to discover and maintain connectivity. In a dense network, where the connectivity can be maintained by only a subset of nodes, the flooding is inefficient and redundant. It could cause many collisions, heavy load and congestion, as well as bandwidth and power consumption. These problems are termed the broadcast storm problem [55]. AODV optimizations include an expending ring search technique [48] to reduce the area flooded by the RREQ. Other mechanisms, such as clustering [56] and multipoint relays (MPR) [57] reduce the dissemination overhead, but are out of the scope of this research.

When there are no other practical alternatives available for rating exchange, it is essential to minimize the cost and the overhead of broadcast transmission. Improving the volume of the rating traffic is achieved by minimizing the bandwidth usage (packet size), the frequency and the range of the dissemination as much as possible.

Simulations as in [58] have shown that the ad hoc network performance is optimal when the number of neighbors is between six and eight. Moreover, an ad hoc network simulation of a realistic movement model incorporation with obstacles [17] has shown that the average number of neighbors per node is considerably lower than in the comparable model without obstacles. Because of the mobility, a node may consider in a given time more nodes as its active neighbors, than in reality. Therefore, this optimal number of six to eight may significantly increase when it is expressed in the

rating packet.

The optimal frequency of the rating distribution depends heavily on the network topology and on the dynamism of the nodes. It should be a compromise between the need to know the true information in a live time (in order to avoid packet loss) and the transmission/processing overhead.

Black list distribution for a larger area may be a useful technique to enable wider punishment of malicious nodes. It is effective mainly when the incidence of malicious nodes is rare. However, misuse of this technique by malicious nodes, for denial of service, might cause substantial damage, more than benefit, so it is not used.

6.3 Reaction

Every node utilizes the rating information in its forwarding decision, both for path selection for its own data packets, and to decide which node to punish or reward, by dropping or forwarding this node's traffic.

6.3.1 Path Selection

Several solutions may be applied to increase paths' reliability, using the 1-hop neighbors rating that each node maintains. Using multipath algorithms to enable selection from various potential routes, is accompanied by high overhead, latency and poor performance in low-density networks. Solutions that involve multiple RREP from the destination hold problems of loops and require costly maintenance [59].

Our solution is a simpler variation of the original protocol, using a greedy strategy. Every node selects the most reliable next hop that it knows on the path. This strategy maximizes the reliability of the path in terms of probability that a packet will be forwarded correctly, if no cooperation exists between malicious nodes. Still, it does not promise any property related to the path's length or load. When a reliable node is favored by many nodes, it may become congested, thus there is another metric that considers also load balancing. In this metric, every node estimates the load of its neighbors by their recent positive actions, $p'_{i,j}(t)$, and selects the less congested node among a group of nodes with a tolerable rating.

The concept of reliable paths is based on differentiation between three reliability levels of active nodes. These levels are based on both the total rating and the total number of positive actions, as the following: (1) *an unreliable node* is a node with

low rating, but with no enough evidence to identify it as misbehaving. Such a node is never being chosen as part of a path. (2) *a reliable node* is a node with average good rating. (3) *a very reliable node* is a node with high rating. Such a node is prefered by multiple nodes, so we wish to balance the load among such nodes.

Load Balancing

When a reliable node is favored by many nodes, it may become congested, thus there is another metric that considers also load balancing. In this metric, every node estimates the load of its neighbors by their recent positive actions, $p'_{i,j}(t)$, and selects the less congested node among a group of nodes with a tolerable rating.

The modified AODV protocol to increase path's reliability is presented below.

Processing and Forwarding Route Requests

a) *Constructing a full path* - When a node has a reply to the request, and this is the first request that was received, it sets a reverse route and generates a reply only if the previous hop is the request originator or a very reliable node.

   Otherwise, it sets a timer and processes every identical request that it receives from other nodes.

   On subsequent requests, if it has not previously transmitted a reply, it checks the node's reliability and if it finds a very reliable node, it sets a reverse route and generates a reply.

   On a timeout, if no reply was sent, the node chooses the most reliable node from the reliable request transmitters, sets a reverse route to it and transmits the reply. In case of no reliable transmitters, the node does not reply at all.

   Using this method, a node prefers transmitting through a very reliable node than through other reliable nodes, if it is not too far or congested. Thus, increasing the path's reliability.

b) *Constructing a reverse path* - If the node does not have a reply to the request, it examines every request with less or the same hop count compared with the first request it got from any node.

If the request was received from an unreliable node, than the node drops it. Upon a first request to be received from a reliable node, the node processes it as the original protocol: sets a reverse route, relays the request and transmits buffered packets.

If a request was previously processed, but the later request comes from more reliable node (with load-balancing consideration), then the node sets a new reverse route and transmits buffered packets, if existing.

In this way, the node ensures that the reverse path it maintains is reliable.

Receiving and Forwarding Route Replies

a) If the reply was received from the destination itself or from a node that seems as a reliable, then the node processes the reply and sets a route to the destination. Otherwise, the node ignores the reply.

b) If the receiving node is an intermediate node, it forwards the reply only if the next hop in the path is reliable.

This new path selection utilizes the information about 1-hop neighbors only, in contrast to the DSR solutions which use rating on several nodes along the path. It involves drawbacks as additional processing overhead and latency, and includes other significant weak points, relating to the protocol properties:

(1) A basic characteristic of AODV is that the most available (and shortest) route is chosen in each route discovery. This property is not saved in our modified AODV protocol and there are many situations in which a node chooses a longer path that is more vulnerable to misbehaving nodes and route breaks, so in the overall view it does not provide the highest reliability. Naturally, because of the short delay we offer (80ms - which is based on the simulation definition that node traversal time is 40ms), the path length is bounded. Additionally, the selection of a longer path can be done only once - by the reply originator, so practically the length of new paths is not much longer than the original paths.

(2) The reliability requirements may cause more dropping because there are less routes. This dropping may affect the rating protocol, when well-behaving nodes are considered as misbehaving.

Destpite that, the results show that even the limited information helps to improve the throughput considerably.

6.3.2 Punishment and Reward

Routing Protocol

In optimal systems with full fairness, nodes get service according to their network contribution. This is achieved by various Quality of Service (QoS) mechanisms. QoS is an important issue in networking and in MANET and has been discussed in many papers. We leave it for a future work.

We offer a simpler approach which differentiates between well-behaving nodes and misbehaving nodes, with an emphasis on punishment. A misbehaving node is isolated from a well-behaving node when its rating decreases below a predefined threshold. The isolation is done by performing a link-break operation (sending RERR packet) and by ignoring further packets from this isolated node (as if the link to this node is down). If a node receives packets of a misbehaving node through a reliable node, it transmits them in order to avoid erroneous suspicions of misbehavior. In the absence of discrimination, when the node behaves badly in a consistent manner, most of its neighbors isolate it and thus it does not get a proper service. Over time, the rating of the misbehaving node fades and increases to zero, so it is afforded a second chance to return back to the network. In this second chance, the node is considered as a disaster-prone. This means that further identification of it as misbehaving requires less observations, and if it is found out as misbehaving again, it is rejected for much longer period. Well-behaving nodes receive service, and temporary problems do not harm their operation.

Rating protocol

Nodes that distribute correct rating information have the chance to modify rating of misbehaving nodes and thus to speed up their detection and isolation. There are many situations where two nodes report honesty, but due to inconsistency of the node or missing evidence, their rating reports are considered false. Since there are many fragile situations, rating of nodes is not affected by their trustworthiness, so liar nodes are not punished.

Chapter 7

Protocol Steps

The combination of the scheme's components provides a complete protocol on top of AODV. This chapter highlights the protocol phases.

7.1 Initialization

On booting, every node initializes all the variables, data structures and timers. The observations and rating calculation start only when the node is a part of an active route, i.e. either it initiates a RREQ packet or receives one, according to AODV protocol. The reaction component starts after some predefined period, once a node collects enough rating information about its neighbors.

7.2 Observations

Every packet that is successfully transmitted or overheard by the MAC layer is examined. If a neighbor should relay a transmitted packet, the packet is inserted to a monitoring buffer and the neighbor's rating is decreased until it sends the packet. Broadcast packets (which do not get link-layer acknowledgments) are inserted to the buffer as well, except HELLO messages. Received packets are compared with the buffered packets, and if a match is found, the neighbor's rating is increased accordingly.

The overheard packets are also used to update the activeness of the neighbors. Together with the HELLO packets, a node has more accurate information on its neighborhood. Periodically, the node flushes its buffer to remove obsolete information.

7.3 Direct Rating Calculation

Neighbor j, detected by a node i, is inserted to a rating list with a neutral rating of zero. The rating factors, recent positive actions $p_{i,j}$ and recent negative actions $n_{i,j}$, are updated permanently according to the observations and are used for period calculation of the direct rating $DR_{i,j}(t)$.

7.4 Rating Exchange

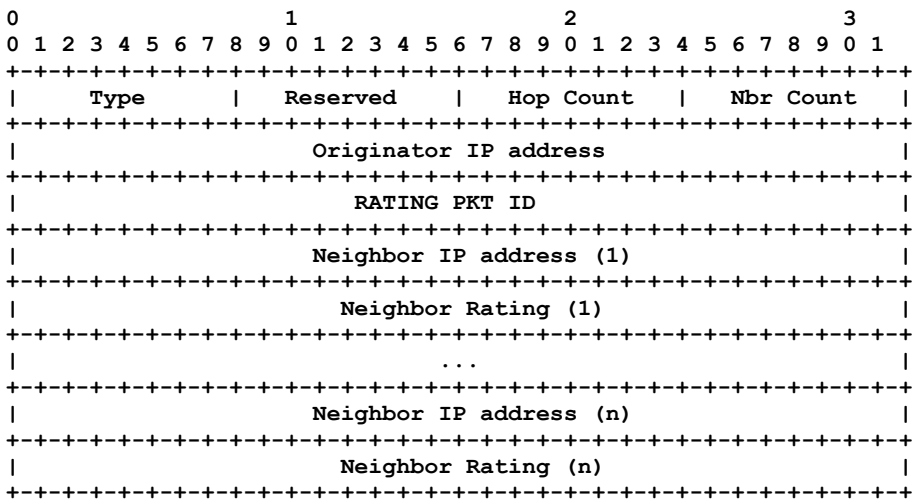
The rating exchange is performed continuously. Every RATING packet (see Fig. 7.1) contains the direct rating of the node's active 1-hop neighbors ($DR_{i,j}$) and the direct rating of all the nodes in the black list. The packets are exchanged only between direct neighbors. Once a node i receives a RATING packet from a node k, it saves the indirect information of all their shared neighbors.

7.5 Total Rating and Trust Calculation

Periodically, the total rating and the trustworthy of active neighbors, as a function of time, are calculated. A node i updates the total rating $TR_{i,j}(t)$, based on the direct and indirect information it has, using the trust information and the deviation test. The trust variables $t_{i,k}$ and $f_{i,k}$ are updated accordingly, and the trustworthy value $T_{i,k}(t)$ is calculated based on these values. From time to time, the node flushes its lists and tables and removes all the irrelevant information.

7.6 Path Selection and Maintenance

The path selection method is taken during the route discovery phase and when a misbehaving node is detected. In the route discovery phase, the node chooses the best route it knows, based on the rating of its neighbors. The path reliability is verified at each data packet transmission. When an unreliable path is discovered, either local repair or RERR transmission are performed. When a misbehaving node is detected, all the routes through it become invalid.



The format of the RATING message is illustrated above, and contains the following fields:

Type	5
Reserved	Sent as 0; ignored on reception.
Hop Count	The number of hops from the originator. 1 in our simulation.
Nbr Count	The number of neighbors included in the message.
Originator IP address	The IP address of the node which originated the RATING message.
RATING PKT ID	A sequence number uniquely identifying the particular RATING packet when taken in conjunction with the originating node's IP address.
Neighbor IP address	The IP address of the neighbor that has been rated.
Neighbor Rating	The direct rating of the neighbor.

Figure 7.1: Rating Message Format

7.7 Misbehaving Nodes' Isolation

The behavior of a node is determined based on its total rating and the number of observations. Basically, less observations are required to consider a node with a very low rating as misbehaving, than a node with a higher rating. A misbehaving node is

inserted to a black list and no further interaction is done with it during that time. This isolation period is the punishment for its behavior. When its rating is faded, it is deleted from the black list and has a second chance to operate. This time, however, a second detection of it as misbehaving is quicker (requires less observations) and for a considerably longer period.

Chapter 8

Simulation Environment

This chapter provides a short overview on the simulation model.

8.1 Introduction

Simulation is a fundamental tool in the development of MANET protocols, because the difficulty to deploy and debug them in real networks. The simulation eases the analyzing and the verification of the protocols, mainly in large-scale systems. It offers flexible testing with different topologies, mobility patterns, and several physical and link-layer protocols. However, a simulation cannot provide evidence in real-world scenarios, due to assumptions and simplifications that it makes. Various examinations, such as [60], show significant divergences between different simulators that demonstrate an identical protocol. Therefore, the results obtained from the simulations should be evaluated appropriately.

Three well-known simulators are used for MANET simulations: NS-2, GloMoSim and OPNET. We chose GloMoSim [61], because it is a scalable simulator that was designed especially to large wireless networks. It supports thousands of nodes, using parallel and distributed environment.

8.2 GloMoSim Overview

GloMoSim [62], [63], [64], was designed as a set of library modules, each of which simulates a communication protocol in the protocol stack. The library uses the OSI layer approach and supports multiple protocols in each layer:

Layer	Model
Physical	**Free-space**, Two-ray
Data Link	CSMA, MACA, **802.11**, TSMA
Network	Bellman-Ford, FISHEYE, WRP, **AODV** ,DSR ,LAR1 ,ODMPR
Transport	TCP, **UDP**
Application	**CBR**, HTTP, TELNET, FTP

Table 8.1: GloMoSim OSI Library

The layers are separated and each layer has its own API. The layers interact with each other using message-passing approach. A combination of different protocols at various layers into a complete protocol suite, as well as extension with alternative protocols, can be done simply. The simulator is built above *PARSEC* [65], a C-based language that was developed for discrete-event simulations. The simulator enables various scenarios, using configuration files, and allows analysis by a trace file with statistics. The visualization tool of GloMoSim, written in Java, shows the network look, nodes' mobility and packet transmissions.

8.3 Simulation Parameters

Various network scenarios were analyzed to prove the model's correctness and characteristics. Every plot was taken as an average of ten different runs. In the simulation experiment, we tested networks from 10 up to 500 mobile hosts.

The area, in which the nodes were placed randomly, was chosen based on the metrics presented in [48] and [66] to maintain the network density and connectivity as constant and balanced.

In all the simulations, we used standard parameters of the channel and radio model: channel capacity of 2MB/s, free space propagation model and radio propagation range of 250 meters. The IEEE 802.11 protocol was used as the medium access control protocol.

The mobile nodes use the random waypoint as the movement model. The range of the speed is from 5 to 20 m/s. Simulations in [67] have shown that minimum speed of zero in the random waypoint model cannot reach a steady state because the speed is continuously decreasing as the simulation progresses. The solution is to set a positive minimum speed and, thus, we give our simulation a minimum speed of 5. The pause time is varied randomly between 0 and 500.

The traffic was produced using a traffic generator, which made randomly constant bit rate (CBR) sessions. The data packet size was 64 Bytes and no fragmentation was used. We avoided data packet transmissions between neighbors, and all the results refer to packets on routes that are above 1-hop length, so more accurate results are achieved.

Default values for some of the protocol parameters are given in Table 8.2. These values are not attempted to be the optimal ones for any network, but we found them as reasonable and effective in the simulation. The original parameters of AODV, as described in RFC 3561 [12] section 10, remain unchanged.

Parameter	**Value**
rating interval for rating calculation and distribution	8.5s
γ, δ, weight of past behavior for direct and total rating	0.8
μ, weight of past belief	0.8
Δ, the deviation test window size	0.5
w, maximum weight of indirect rating (depends on the number of neighbors)	5
rating threshold for misbehaving nodes (together with some minimal observations. As much as the rating is smaller, the smaller number of observations that are required)	-0.2
reliability threshold for path selection (together with some minimal observations)	0.25
trustworthy threshold for accepting reports	0.75
aunreliable node's rating	(-0.2 - 0.25)
reliable node's rating	[0.25 - 0.75)
very reliable node's rating	[0.75 - 1]
reply delay	80ms

Table 8.2: Configuration Parameters

Chapter 9

Simulation Results and Analysis

This chapter examine the performance of our scheme and provides a comprehensive analysis of the results.

9.1 Advantages over Alternative Solutions

The assumption behind the usage of reputation systems is that additional information helps nodes to detect and react better. This assumption should not be taken for granted, though. There are many scenarios in which the additional information hardens the detection. For example, a black hole may seem reliable to those nodes which do not forward data through it, so their good rating advertising slows down its detection. Moreover, since the reputation acceptance is strict, in order to limit the liars effect, the further information may not be used appropriately.

The reputation exchange is found valuable mainly for the following reasons:

1. In contrast to other existing reputation systems, our scheme uses the indirect information for two parameters: the rating value ($TR_{i,j}$) and the number of observations (total positive and negative actions, $p'_{i,j}$ and $n'_{i,j}$). Generally, there is a minimal number of observations that are required before suspecting a node as a misbehaving one. By sharing the experience of other nodes, the number of self-observations is decreased and the detection is quicker, even when the minimal number of observations is low.

2. The number of false positives is usually lower with reputation exchanges, because other nodes' observations moderate a temporary mistaken rating.

3. In a high mobility network, when a node does not have enough information about its surrounding, the information it receives may be useful during its first steps.

However, a system with a rating exchange may not always have a significant advantage, and may even perform worse compared to a scheme without the information distribution. This happens when:

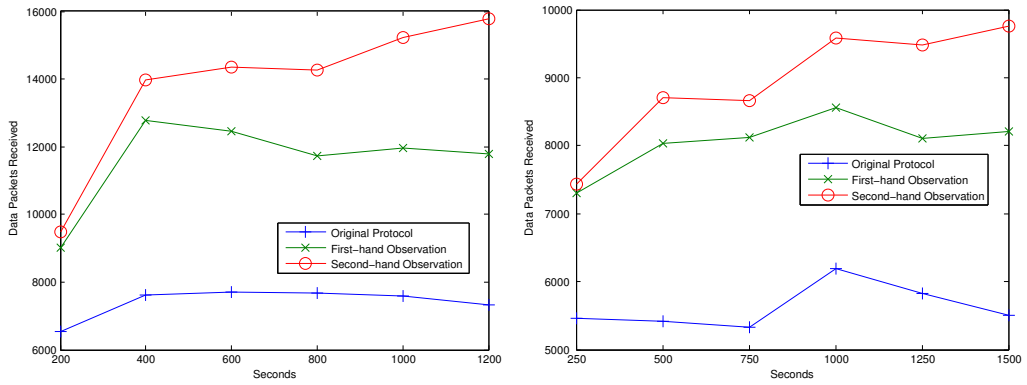
1. Significant amounts of nodes do not have a correct map of their neighbors or there is no sufficient trust relations between the nodes, e.g. too high mobility in a large area, bad connectivity, bad participation, etc.
   The information acceptance is very low in such cases so its effect is negligible.

2. A relatively static network, where only few arrivals and departures occur or the number of shared neighbors between two neighboring nodes is very low.
   The exchanged ratings do not contribute worthwhile information in such conditions.

3. Frequent packet dropping because of load, collisions, long paths and other network factors that make the system unstable.
   In such circumstances, there are many false positives and the overhead of the rating exchange is bigger than the information contribution.

By examination of the throughput (Fig. 9.1), we can see significant improvements by both first-hand observation method and a reputation system, compared to the original AODV protocol. However, the first-hand observations improve the throughput only locally (the changes in the throughput as the time advances are minor), while the second-hand information gradually affects all the network and causes consistent improvement as time progresses. Note, however, that it takes time for the network to become stable because there is a second chance for every misbehaving node. Similar tendencies can be shown for larger networks with 100 nodes. The advantages of the reputation system when the network is larger are applied less obviously than smaller networks since the system converges more slowly.

We can see the same trend even more prominently when we look at the punishment of misbehaving nodes (Fig. 9.2). The differences between the schemes are clearer in the punishment graph, since its components are not effected by collisions, malicious dropping and other external causes to packet dropping, as in the throughput graph.

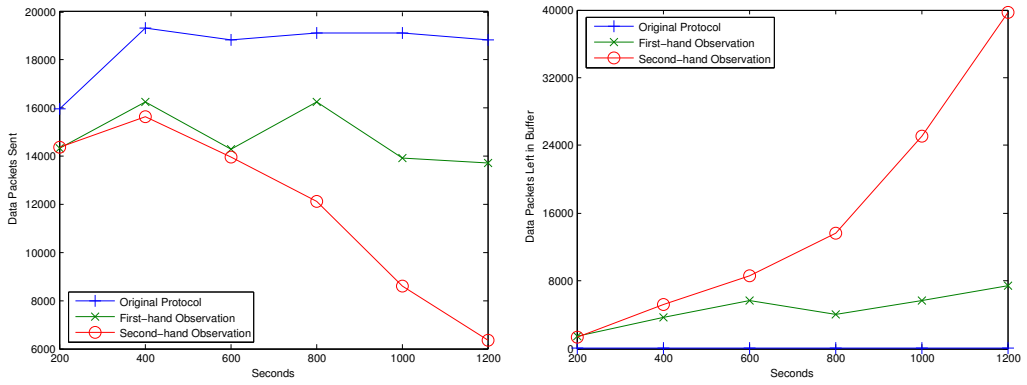


(a) Throughput of Well-behaving Nodes - 50 Nodes, 15 Sources, 15 Black Holes.

(b) Throughput of Well-behaving Nodes - 100 Nodes, 20 Sources, 30 Black Holes.

The network is characterized by full mobility and load. Every node runs 4 different sessions in each period (200 seconds) at a rate of 10 packet/second. The sources, destinations and start time are selected randomly. Different sessions provide various possibilities for path selection. The relatively high rate of packets (usually 4 packet/second is the normal rate) is to decrease the cost of the path, by using it massively for a short time when it is constructed. Our solution works also for slower rates, but involves more control packets.

Figure 9.1: First-hand and Second-hand Observation Effects on Nodes Reward.



(a) Data Packets That Misbehaving Nodes Succeed in Transmitting to Each Period.

(b) Data Packets That Were Left in The Buffer Because No Route Was Found to The Destination.

50 nodes, 15 sources and 15 misbehaving nodes with the same simulation parameters as previous simulation. The sources in 9.2(a) are the misbehaving nodes which transmit packets to the other good nodes. In 9.2(b), the sources are well-behaving and the destinations are misbehaving. The punishment of a node can be reflected either by the number of packets it does not succeed in transmitting or by the number of packets that it does not receive because of its isolation.

Figure 9.2: Punishment of Misbehaving Nodes.

9.2 Partial Data Packets Dropping

Detection and punishment of gray hole nodes are difficult for several reasons.

First, the monitoring is limited because of collisions and mobility. Therefore, a strict treatment to nodes with a relatively low rating will probably cause a large amount of undesirable false positives. On the other hand, soft handling of such cases gives the gray holes opportunities to continue with their misbehavior.

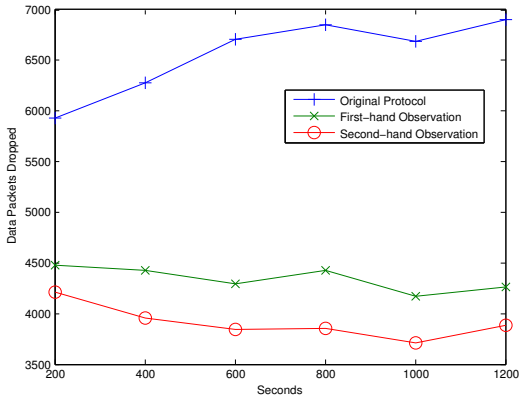
In addition, the reputation system's usefulness is limited in case of node discrimination, because there are many contradictions between the exchanged ratings.

Lastly, inconsistent behavior requires costly path maintenance to ensure that selected paths remain reliable. The maintenance is necessary, since a node can misuse its good rating to be chosen as part of a route. Then, it can slightly misbehave, in a manner that does not cause its isolation.

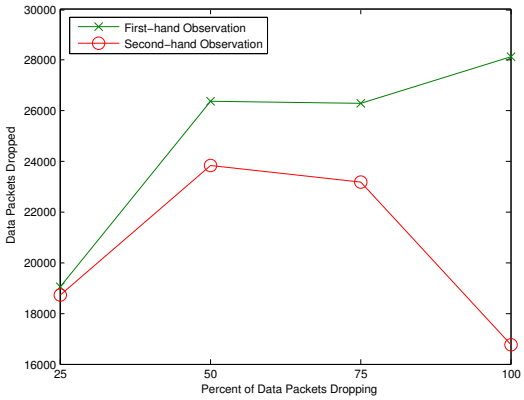
The path maintenance involves further issues. For example, when a node detects a neighbor that does not seem a reliable, but it does not have enough evidence for that, there is a greater doubt whether to continue sending through it or to disconnect it. The first option does not promise a reliable path, while the second option involves overhead of local repairs and deletion of buffered packets because no alternative route exists. In situations of too many disconnections, a good node may be suspected as malicious because it does not find alternative routes.

According to the simulation results (Fig. 9.3(a)), the monitoring is as effective in partial dropping as in total dropping. However, in contrast to the throughput improvements along the time, as was shown in Fig. 9.1, there are almost no changes in both systems as the time advances. Figures 9.3(b) and 9.3(c) provide some explanation for this. Generally, the forwarding reliability is the major concern of a node. It prefers avoiding misbehaving nodes, rather than waiting for total verification of malicious nodes in order to punish them. Full identification of a misbehaving node requires that its rating be under the faulty threshold of zero with enough evidence (observations). This means that a node is detected and punished only after it drops about 50% or more of the total packets[1]. When the dropping percentage is less than

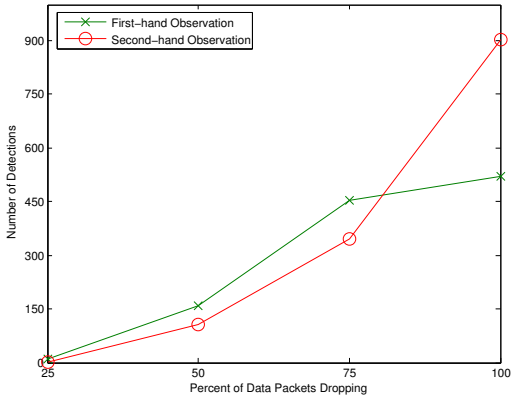
[1]Our rating system considers both control and data packets with weighting the data packets more than control packets. This does not completely solve the problem since the control packets take a significant part of the packets that are forwarded in the network. An advance solution will change our policy to consider only data packets when it seems that control packets are forwarded well. We leave it for future work.



(a) Number of Data Packets Dropped Along the Time. Dropping probability of 50%.



(b) Number of Data Packets Dropped.



(c) Number of Detections.

Each misbehaving node transmits all control packets properly. Thus, the 25%, 50%, 75% and 100% dropping probability of data packets result in approximately 18%, 32%, 45% and 78% dropping from the total transmitted packets accordingly. Fig. 9.3(a) shows the packets dropping along the time. Fig. 9.3(b) and Fig. 9.3(c) present the differences between First-hand observation scheme vs. the full reputation system in the various cases of packet dropping.

Figure 9.3: Partial Data Packets Dropping.

half, there is only avoidance of misbehaving nodes. The avoidance consists of a permanent verification that an active route stays reliable over time and disconnection from the next hop, when its rating decreases beyond some threshold. This disconnection does not involve punishment and isolation, since there is insufficient evidence that the next hop is malicious. However, the node itself prefers not to forward packets through it. The avoidance is effective in increasing path reliability, but because of no punishment, it performs only locally. Despite this, the reputation system is still better

than relying on the first-observations due to the additional information contributed for evaluation nodes in the reliability scale. The better avoidance is expressed by a lower rate of data packets that are dropped and by less misbehavior detections. The lower number of detections indicates that the extra information does indeed help it to identify and disconnect unstable nodes before they reach to the faulty threshold.

Due to the combination of uncertain ratings, contradictions between nodes and the lack of punishments, the contribution is limited but does still exist. The effectiveness of the reputation system is expressed in its entirety when the behavior is more consistent.

9.3 Liars

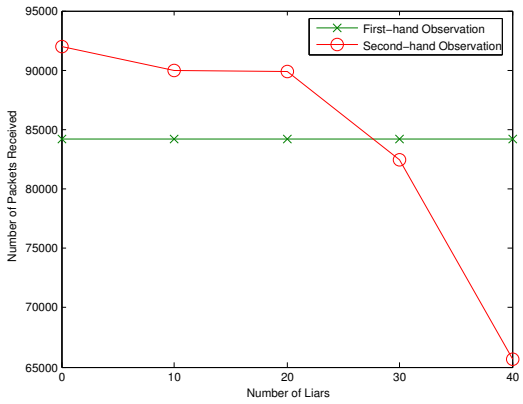
All previous work about robust reputation systems assumed a relatively weak adversary model in which a node either reports extremely negative/positive ratings, random values, inverted values and so on. Our implementation assumes a stronger adversary model in which the liar publishes strategic lies. Those lies are adapted to the ratings that the neighbors hold, in order to be evaluated as trusted and have the ability to adversely affect the other.

For each neighbor, the published rating is constructed as follows:

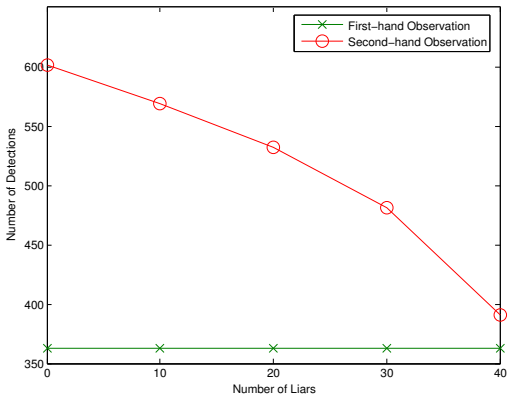
- In case that the average rating received from the neighbors is being either extremely good or extremely bad ($\pm 0.5 - 1$), a wrong rating does not significantly effect it, so the liar prefers publishing the average rating to increase its trustworthiness.

- When the rating is not absolute, its change can affect the status of the node and a lie could harm one node or more. The liar wishes to stand in either the trustworthy test or the deviation test, but since it does not know its trustworthiness by the other nodes, it tries to stand in the deviation test. Therefore, it takes the average rating and compares it to its own information (the direct observations), then increases or decreases the average rating by half of the deviation test window, in contrast to its own information.

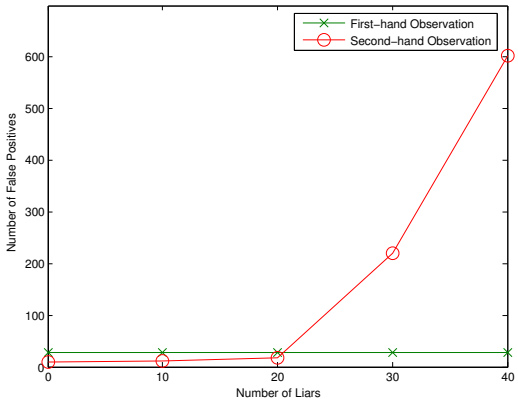
- If no rating is provided by the other nodes, the liar prefers spreading false information more than doing nothing, so it modifies its own information by half of the deviation test. The rating is increased when it is negative and is decreased otherwise.



(a) Throughput.



(b) Number of Node Detections as Misbehave.



(c) Number of False Positives.

Simulation for 1200 seconds, 50 nodes, 10 nodes as black holes and similar traffic parameters as before. Until they are isolated from the network, the black holes distribute correct information, then their reports are ignored and the liars have a larger effect. Note, that since there is a second chance for each node, the number of identifications of nodes as black can be quite large. In addition, because of the avoidance, not all the nodes must be identified as bad. In some cases, however, more than 20 liars are considered as the majority of the running nodes.

Figure 9.4: Liars Effect.

While the deviation test and the trustworthiness prerequisites are enough for simple lies, our scheme requires a consistent majority of good reporters in order to be robust. As it is shown in Fig. 9.4, the system is very robust and performs well until there is a consistent majority of liars. Too many false positives result in poor performance.

9.4 Scalability

Generally, the performance of the original AODV protocol without any misbehaving nodes is poor in larger networks. A reasonable assumption is that with large networks there will be some access points and a central management. However, since the scalability property is one of the desired characteristics, networks with 500 nodes were simulated to examine our scheme.

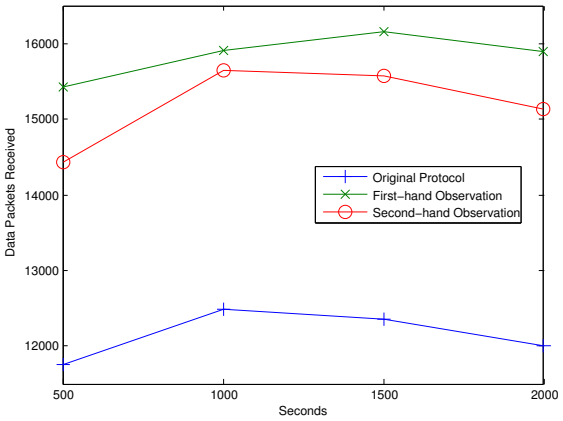
The reputation system was designed from the outset to be scalable and feasible both in large and small networks. Practically speaking, though, it seems that other external factors have greater effects in larger networks.

The main difference between small and large networks is the average path lengths (in our simulation, 3-4 hops in small network vs. 8-13 hops in large network). A long path is more vulnerable to link breaks and requires relatively high control overhead for maintenance. These two conditions, frequent packet dropping, and cost maintenance are major factors in the surprising results we had.

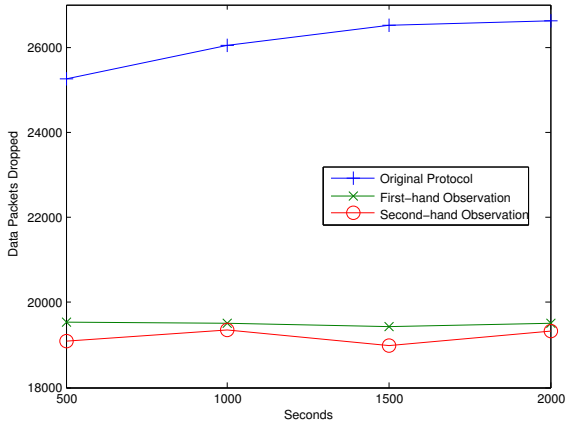
The frequent packet dropping, due to undiscovered routes, unsuccessful local repair and sometimes unreachable destinations, resulted in poor performance when we used the original rating system because of an excessive amount of suspicious and false positives. Consequently, we doubled the number of observations required to detect misbehaving nodes. This, of course, increases the number of dropped packets, but makes the system more stable when the number of false positives is low. The massive control packets that were forwarded in large networks reached 60% to 70% of the total packets transmitted. This means that black hole detections are very difficult to discover and the system, most of the time, is in state of avoidance. As shown previously, the advantage of the reputation system in such cases, compared to First-hand observation method, is limited.

In contrast to the previous simulation results, when we had a correlation between the number of packets that are dropped by malicious nodes and the throughput, the results in a large network, shown in Fig. 9.5, differ.

The reputation system cost, which does not significantly effect small networks, is expressed widely within large networks, in terms of transmission price. This means more bandwidth contention and additional collisions. (The extra overhead in terms of CPU processing and memory storage is minor). As one can see in Fig. 9.5(b), the reputation system manages to suffer less dropped data packets caused by misbehaving nodes. However, the overall number of dropped packets is larger than the



(a) Throughput.



(b) Number of Data Packets Dropped by Misbehaving Nodes.

250 nodes are static and the remainder walk on speed of 5-10 m/s. Other parameters are the same as before. The reputation system with second-hand observations has a tiny advantage over the First-hand observation scheme in the number of data packets that are dropped by misbehaving nodes. Conversely, the throughput of the First-hand observation is better over time than the reputation system.

Figure 9.5: Simulation of 500 Nodes.

corresponding number of the dropped packet when First-hand observation is used (because of network conditions). In such situations, relying on self-observations is better than using the rating exchange.

Chapter 10

Conclusions and Future Work

In this thesis we show that reputation system on top of AODV has an advantage over schemes that rely only on first-hand observations despite the limited amount of information and the additional problems of AODV versus DSR. This advantage includes both profit and punishment according to the behavior, and works for both partial and complete dropping. The reputation system remained robust against advanced liars as well, when a majority of the nodes are trustworthy. In some circumstances, however, the network conditions have greater effect than the reputation system benefits, as in the case of large networks. In such situations, it is better to rely on self-observations.

Our scheme focuses mainly on black and gray holes but can handle also other misbehavior patterns. It can be improved to dynamically change the rating policy, in order to handle the different patterns better (like considering only data packets when control packets are forwarded well).

Additional mechanisms to support QoS and to increase the fairness in the network are possible areas for future research. Our work is dedicated to AODV but can be adopted to other routing algorithms as well as to sensor networks.

Bibliography

[1] L. Buttyfin and J. Hubaux. Report on a working session on security in wireless ad hoc networks, November 2002. Available on: citeseer.ist.psu.edu/buttyfin02report.html.

[2] J. Hubaux, L. Buttyan, and S. Capkun. The quest for security in mobile ad hoc networks. In *Proceedings of the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, CA, USA, October 2001. Available on: citeseer.ist.psu.edu/hubaux01quest.html.

[3] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang. Security in mobile ad hoc networks: Challenges and solutions. *IEEE Wireless Communications*, 11:2–11, February 2004. Available on: http://www.cs.ucla.edu/wing/publication/papers/Yang.WC04.pdf.

[4] P. Obreiter, B. König-Ries, and M. Klein. Stimulating cooperative behavior of autonomous devices - an analysis of requirements and existing approaches. In *Second International Workshop on Wireless Information Systems (WIS2003)*, Angers, France, April 2003. Available on: http://citeseer.ist.psu.edu/558275.html.

[5] M. Conti, E. Gregori, and G. Maselli. Cooperation issues in mobile ad hoc networks. In *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW'04)*, Tokyo, Japan, March 2004. Available on: www.di.unipi.it/~maselli/WWAN_Maselli_G.ps.

[6] S. Buchegger and J. Le Boudec. Performance analysis of the CONFIDANT protocol: Cooperation of nodes - fairness in dynamic ad-hoc networks. In *Proceedings*

of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC), Switzerland, June 2002. Available on: http://citeseer.ist.psu.edu/510340.html.

[7] P. Michiardi and R. Molva. Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Proceedings of The 6th IFIP Communications and Multimedia Security Conference*, pages 107–121, Portorosz, Slovenia, September 2002.

[8] S. Bansal and M. Baker. Observation-based cooperation enforcement in ad hoc networks, July 2003. Available on: http://arxiv.org/pdf/cs.NI/0307012.

[9] L. Buttyán and J. P. Hubaux. Stimulating cooperation in self-organizing mobile ad hoc networks. *MONET*, 8(5):579–592, October 2003. Available on: http://www.hit.bme.hu/~buttyan/publications/ButtyanH03monet.pdf.

[10] P. Michiardi and R. Molva. A game theoretical approach to evaluate cooperation enforcement mechanisms in mobile ad-hoc networks. In *Proceedings of Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt'03)*, SophiaAntipolis, France, March 2003. Available on: citeseer.ist.psu.edu/michiardi03game.html.

[11] Aodv homepage. Available on: http://moment.cs.ucsb.edu/AODV/aodv.html.

[12] C. E. Perkins, E. M. Belding-Royer, and S. R. Das. Ad hoc on-demand distance vector (AODV) routing. RFC 3561, IETF, July 2003. Available on: http://www.ietf.org/rfc/rfc3561.txt.

[13] E. M. Belding-Royer and C. E. Perkins. Evolution and future directions of the ad hoc on-demand distance vector routing protocol. *Ad hoc Networks Journal*, 1(1):125–150, July 2003. Available on: http://www.cs.ucsb.edu/~ebelding/txt/aodv_evol.ps.gz.

[14] J. Broch, D. B. Johnson, and D. A. Maltz. The dynamic source routing protocol for mobile ad hoc networks. Internet draft, IETF, July 2004. Available on: http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt.

[15] E. Royer and C. Toh. A review of current routing protocols for ad-hoc mobile wireless networks. *Mobile Wireless Networks. IEEE Personal Communications*, pages 46–55, April 1999. Available on: citeseer.ist.psu.edu/royer99review.html.

[16] S. R. Das, C. E. Perkins, and E. E. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *INFOCOM (1)*, pages 3–12, 2000. Available on: citeseer.ist.psu.edu/das00performance.html.

[17] M. Conti, E. Gregori, and G. Maselli. Towards reliable forwarding for ad hoc networks. In *Proceeding of Personal Wireless Communications (PWC 2003)*, pages 790–804, Venice, Italy, September 2003. Available on: www.di.unipi.it/~maselli/paper%23119.pdf.

[18] P. Dewan, P. Dasgupta, and A. Bhattacharya. On using reputations in ad hoc networks to counter malicious nodes. In *Proceedings of the 10th International Conference on Parallel and Distributed Systems (ICPADS 2004)*, Newport Beach, CA, USA, July 2004. Available on: http://cactus.eas.asu.edu/partha/Papers-PDF/2004/Dewan-AdHocRouting.pdf.

[19] IETF Working Group MANET. Available on: http://www.ietf.org/html.charters/manet-charter.html.

[20] J. Hoebeke, I. Moerman, B. Dhoedt, and P. Demeester. An overview of mobile ad hoc networks: applications and challenges. In *Proceedings of the 43rd European Telecommunications Congress, FITCE2004*, Ghent, Belgium, November 2004. Available on: http://www.ist-magnet.org/private/files/Dissemination/WP2/1%20An%20Overview%20of%20Mobile%20Ad%20Hoc%20Networks%20-%20Applications%20and%20Challenges.pdf.

[21] I. Chlamtacand M. Conti and J. Liu. Mobile ad hoc networking: Imperatives and challenges. *Ad Hoc Networks*, 1:13–64, 2003. Available on: http://www.ece.ncsu.edu/wireless/Resources/Papers/adhocSurvey.pdf.

[22] S. Corson and J. Macker. Mobile ad hoc networking (MANET): routing protocol performance issues and evaluation considerations. RFC 2501, IETF, January 2001. Available on: http://www.ietf.org/rfc/rfc2501.txt.

[23] D. Gambetta. Can we trust trust? In *Trust: Making and Breaking Cooperative Relations*, chapter 13, pages 213–237. Published Online, 2000. Available on: http://www.sociology.ox.ac.uk/papers/gambetta213-237.pdf.

[24] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman. Reputation systems. *Commun. ACM*, 43(12):45–48, 2000. Available on: http://www.si.umich.edu/~presnick/papers/cacm00/reputations.pdf.

[25] A. Jósang, Roslan Ismail, and Colin Boyd. A survey of trust and reputation systems for online service provision. In *Decision Support Systems*, 2005. Available on: http://security.dstc.edu.au/papers/JIB2005-DSS.pdf.

[26] L. Mui, M. Mohtsahemi, and A. Halberstadt. A computational model of trust and reputation. In *Proceedings of the Thirty-Fifth Hawaii International Conference on SystemSciences*, 2002. Available on: csdl.computer.org/comp/proceedings/\hicss/2002/1435/07/14350188.pdf.

[27] S. Marti, T. J. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In *Mobile Computing and Networking (MOBICOM)*, pages 255–265, 2000. Available on: citeseer.ist.psu.edu/marti00mitigating.html.

[28] S. Buchegger, C. Tissieres, and J. Y. Le Boudec. A test-bed for misbehavior detection in mobile ad-hoc networks - how much can watchdogs really do. Technical Report IC/2003/72, EPFL-DI-ICA, November 2003. Available on: citeseer.ist.psu.edu/645200.html.

[29] S. Buchegger and J. Y. Le Boudec. Nodes bearing grudges: Towards routing security, fairness, and robustness in mobile ad hoc networks. In *Proceedings of the Tenth Euromicro Workshop on Parallel, Distributed and Network-based Processing*, pages 403 – 410. IEEE Computer Society, January 2002. Available on: http://citeseer.ist.psu.edu/535553.html.

[30] S. Buchegger and J. Y. Le Boudec. The effect of rumor spreading in reputation systems for mobile ad-hoc networks. In *Proceedings of WiOpt '03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Sophia-Antipolis, France, March 2003. Available on: http://icapeople.epfl.ch/sbuchegg/bucheggerL03A.pdf.

[31] S. Buchegger and J. Y. Le Boudec. A robust reputation system for mobile ad hoc networks. Technical Report IC/2003/50, EPFL-DI-ICA, July 2003. Available on: icwww.epfl.ch/publications/documents/IC_TECH_REPORT_200350.pdf.

[32] S. Buchegger. *Coping With Misbehavior in Mobile Ad-hoc Networks*. PhD thesis, Swiss Federal Institute of Technology (EPFL), April 2004. Available on: http://icapeople.epfl.ch/sbuchegg/sonjaThesis.pdf.

[33] P. Michiardi and R. Molva. Preventing denial of service and selfishness in ad hoc networks. In *Working Session on Security in Ad Hoc Networks*, Lausanne, Switzerland, June 2002.

[34] P. Michiardi and R. Molva. Simulation-based analysis of security exposures in mobile ad hoc networks. In *European Wireless Conference*, 2002. Available on: citeseer.ist.psu.edu/michiardi02simulationbased.html.

[35] P. Ning and K. Sun. How to misuse aodv: A case study of insider attacks against mobile adhoc routing protocols. In *Proceedings of the 4th Annual IEEE Information Assurance Workshop*, West Point, June 2003. Available on: http://discovery.csc.ncsu.edu/~pning/pubs/Submission2AdHocNet.pdf.

[36] C.Y. Tseng, P. Balasubramanyam, C. Ko, R. Limprasittiporn, J. Rowe, and K. Levitt. A specification-based intrusion detection system for aodv. In *Proceedings of the ACM workshop on Security of ad hoc and sensor networks (SASN '03)*, Fairfax, Virginia, October 2003. Available on: http://seclab.cs.ucdavis.edu/papers/new_aodvids-v1.pdf.

[37] S. Giordano Urpi A., M.A. Bonuccelli. Modelling cooperation in mobile ad hoc networks: a formal description of selfishness. In *Proceedings of WiOpt '03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Sophia-Antipolis, France, March 2003. Available on: www.cs.ucsb.edu/~ebelding/courses/595/s04_gametheory/papers/Giordano_Selfishness.pdf.

[38] Y.C. Hu, A. Perrig, and D. B. Johnson. Packet leashes: A defense against wormhole attacks in wirelessnetworks. In *INFOCOM*, 2003. Available on: http://www.ieee-infocom.org/2003/papers/48_03.PDF.

[39] Y. Huang and W. Lee. A cooperative intrusion detection system for ad hoc netwroks. In *Proceedings of the ACM Workshop on Security of Ad Hoc and*

Sensor Networks (SASN '03), Fairfax Virginia, October 2003. Available on: http://www.cc.gatech.edu/~wenke/papers/sasn.pdf.

[40] Hao Yang, Xiaoqiao Meng, and Songwu Lu. Self-organized network-layer security in mobile ad hoc networks. In *Proceedings of the ACM workshop on Wireless security (WiSE '02)*, Atlanta, GA, USA, September 2002. Available on: citeseer.ist.psu.edu/659429.html.

[41] M. Just, E. Kranakis, and T. Wan. Resisting malicious packet dropping in wireless ad hoc networks. In *Proceedings of ADHOCNOW'03*, Montreal, Canada, October 2003. Available on: http://www.scs.carleton.ca/~kranakis/Papers/adhocnow03.pdf.

[42] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens. An on-demand secure routing protocol resilient to byzantine failures. In *Proceedings of the ACM workshop on Wireless security (WiSE '02)*, September 2002. Available on: www.cs.colorado.edu/~rhan/CSCI_7143_001_Fall_2002/Papers/Awerbuch2002_SecureByzantine.pdf.

[43] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens. Mitigating byzantine attacks in ad hoc wireless networks. Technical report, March 2004. Available on: http://www.cnds.jhu.edu/research/networks/archipelago/publications/Awerbuch-MitigatingByzantine-TechReport1-March2004.pdf.

[44] E. M. Belding-Royer S. Gwalani and C. E. Perkins. Aodv-pa: Aodv with path accumulation. *ICC 2003 - IEEE International Conference on Communications*, 26(1):527–531, May 2003. Available on: www.cs.ucsb.edu/~ebelding/txt/aodvpa.pdf.

[45] P. Yau and C. J. Mitchell. Reputation methods for routing security for mobile ad hoc networks. In *Proceedings of SympoTIC '03, Joint IST Workshop on Mobile Future and Symposium on Trends in Communications*, pages 130–137, Bratislava, Slovakia, October 2003. IEEE press. Available on: www.isg.rhul.ac.uk/~cjm/rmfrsf.pdf.

[46] H. Yang, G. Zhong, and S. Lu. Network performance centric security design in manet. In *Security Workship at ACM MobiHoc 2002*, October 2002. Available on: http://www.cs.ucla.edu/wing/pdfdocs/MC2R02.pdf.

[47] J. Mundinger and J. Y. Le Boudec. Analysis of a reputation system for mobile ad-hoc networks with liars. In *(WiOpt '05)*, Riva del Garda, April 2005. Available on: http://www.mics.org/getDoc.php?docid=1070&docnum=1.

[48] S. J. Lee, E. M. Belding-Royer, and C. E. Perkins. Scalability study of the ad hoc on-demand distance vector routing protocol. *International Journal on Network Management*, 13(2):97–114, March-April 2003. Available on: http://www.cs.ucsb.edu/~ebelding/txt/scalability.pdf.

[49] S. Mueller, R. P. Tsang, and D. Ghosal. Multipath routing in mobile ad hoc networks: Issues and challenges. In *MASCOTS Tutorials*, pages 209–234, 2003. Available on: http://networks.cs.ucdavis.edu/~ghosal/Research/publications/stephen-lncs-multipath-survey-paper-2004.pdf.

[50] S.J. Lee and M. Gerla. AODV-BR: Backup routing in ad hoc networks. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2000)*, Chicago, IL, September 2000. Available on: http://www.hpl.hp.com/personal/Sung-Ju_Lee/abstracts/papers/wcnc2000a.pdf.

[51] M. K. Marina and S. R. Das. On-demand multipath distance vector routing in ad hoc networks. In *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, November 2001. Available on: citeseer.ist.psu.edu/marina01demand.html.

[52] Z. Ye, S. V. Krishnamurthy, and Satish K. Tripathi. A framework for reliable routing in mobile ad hoc networks. In *Proceedings IEEE INFOCOM 2003*, San Franciso, CA, USA, 2003. Available on: http://www.ieee-infocom.org/2003/papers/07_03.PDF.

[53] A. Jósang and R. Ismail. The beta reputation system. In *15th Bled Conference on Electronic Commerce*, Bled, Slovenia, June 2002. Available on: security.dstc.edu.au/papers/JI2002-Bled.pdf.

[54] A. Jósang, S. Hird, and E. Faccer. Simulating the effect of reputation systems on e-markets. In *Proceedings of the First International Conference on Trust Management*, Crete, May 2003. Available on: security.dstc.edu.au/papers/JHF2003-ICTM.pdf.

[55] S. Y. Ni, Y. C. Tseng, Y. S. Chen, and J. P. Sheu. The broadcast storm problem in a mobile ad hoc network. In *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 151–162, Washington, USA, August 1999. Available on: http://www.csie.nctu.edu.tw/~yctseng/papers.pub/mobile6-storm-mobicom99.pdf.

[56] T-J.Kwon, Y. Yi, and M. Gerla. Efficient flooding in ad hoc networks using on-demand (passive) cluster formation. In *Proceedings of Mobihoc*, Annapolis, USA, June 2003. Available on: http://www.cs.ucla.edu/NRL/wireless/PAPER/yjyi_medhoc_final2003.pdf.

[57] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying: An efficient technique for flooding in mobile wireless networks. Technical Report Research Report RR-3898, INRIA, February 2000. Available on: citeseer.ist.psu.edu/qayyum00multipoint.html.

[58] E. Royer, P. Melliar-Smith, and L. Moser. An analysis of the optimum node density for ad hoc mobile networks. In *Proceedings of the ICC 2001 - IEEE International Conference on Communications*, pages 857–861, June 2001. Available on: http://citeseer.ist.psu.edu/475080.html.

[59] A. Murthy P. Sambasivam and E. M. Belding-Royer. Dynamically adaptive multipath routing based on aodv. In *Proceedings of the 3rd Annual Mediterranean Ad hoc Networking Workshop (MedHocNet)*, Bodrum, Turkey, June 2004. Available on: http://www.cs.ucsb.edu/~ebelding/txt/mednocnet_mpaodv.pdf.

[60] Y. Sasson D. Cavin and A. Schiper. On the accuracy of manet simulators. In *ACM Principles of Mobile Computing (POMC 2002)*, Toulouse, France, October 2002. Available on: lsewww.epfl.ch/Documents/acrobat/CSA02b.pdf.

[61] GloMoSim. Available on: http://pcl.cs.ucla.edu/projects/glomosim.

[62] L. Bajaj, M. Takai, R. Ahuja, R. Bagrodia, and M. Gerla. Glomosim: A scalable network simulation environment. Technical Report 990027, UCLA Computer Science Department, 1999. Available on: citeseer.ist.psu.edu/225197.html.

[63] X. Zeng, R. Bagrodia, and M. Gerla. Glomosim: A library for parallel simulation of large-scale wireless networks. In *Workshop on Parallel and Distributed*

Simulation, pages 154–161, Canada, May 1998. Available on: citeseer.ist.psu.edu/zeng98glomosim.html.

[64] J. Nuevo. A comprehensible glomosim tutorial, March 2003. Available on: http://www.cs.virginia.edu/~jx9n/courses/cs656/glomoman.pdf.

[65] PARSEC. Available on: http://pcl.cs.ucla.edu/projects/PARSEC.

[66] Horst Hellbrück and Stefan Fischer. Towards analysis and simulation of ad-hoc networks. In *Proceedings of the 2002 International Conference on Wireless Networks (ICWN02)*, pages 69–75, USA, June 2002. Available on: http://citeseer.ist.psu.edu/531702.html.

[67] J. Yoon, M. Liu, and B. Noble. Random waypoint considered harmful. In *Proceedings IEEE INFOCOM 2003*, San Franciso, CA, USA, 2003. Available on: http://www.ieee-infocom.org/2003/papers/32_03.PDF.