ON DETECTION MECHANISMS AND THEIR PERFORMANCE FOR PACKET DROPPING ATTACK IN AD HOC NETWORKS

by

Tanapat Anusas-amornkul

B.Eng. in Electrical Engineering, Kasetsart University,1995

MS. in Telecommunications, University of Colorado at Boulder,1998

Submitted to the Graduate Faculty of

the School of Information Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2008

UNIVERSITY OF PITTSBURGH

SCHOOL OF INFORMATION SCIENCES

This dissertation was presented

by

Tanapat Anusas-amornkul

It was defended on

July 24, 2008

and approved by

Dr.Prashant Krishnamurthy, Telecommunications Program

Dr.David Tipper, Telecommunications Program

Dr.Richard Thompson, Telecommunications Program

Dr.James Joshi, Information Sciences

Dr.Daniel Mossé, Computer Science

Dissertation Director: Dr.Prashant Krishnamurthy, Telecommunications Program

ON DETECTION MECHANISMS AND THEIR PERFORMANCE FOR PACKET DROPPING ATTACK IN AD HOC NETWORKS

Tanapat Anusas-amornkul, PhD

University of Pittsburgh, 2008

Ad hoc networking has received considerable attention in the research community for seamless communications without an existing infrastructure network. However, such networks are not designed with security protection in mind and they are prone to several security attacks. One such simple attack is the packet dropping attack, where a malicious node drops all data packets, while participating normally in routing information exchange. This attack is easy to deploy and can significantly reduce the throughput in ad hoc networks.

In this dissertation, we study this problem through analysis and simulation. The packet dropping attack can be a result of the behavior of a selfish node or pernicious nodes that launch blackhole or a wormhole attacks. We are only interested in detecting this attack but not the causes of the attack. In this dissertation, for simple static ad hoc networks, analysis of the throughput drop due to this attack along with its improvement when mitigated are presented. A watchdog and a newly proposed "cop" detection mechanisms are studied for mitigating the throughput degradation after detection of the attack. The watchdog mechanism is a detection mechanism that has to be typically implemented in every node in the network. The cop detection mechanism is similar to the watchdog mechanism but only a few nodes opportunistically detect malicious nodes instead of all nodes performing this function. For multiple flows in static and mobile ad hoc networks, simulations are used to study and compare both mechanisms. The study shows that the cop mechanism can improve the throughput of the network while reducing the detection load and complexity for other nodes.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

In every path we have walked, we always meet new people and this is true in our lives. For me, the very first people I met were my parents. Without them, I can't be existing today and I would like to dedicate this dissertation to my parents, Mr.Kamol and Mrs.Jongjit Anusas-amornkul for their unconditional love and support. They have always given me the encouragement to succeed in every path in my life and I am very grateful for that. I would like to thank my sisters and brother, who support me in every way they can. Then, I thanks the many friends I met along my path to the University of Pittsburgh.

Here, I met my advisor, Dr.Prashant Krishnamurthy, who has guided me through many years without tiredness. He helped me to go through every step for the Ph.D. process with his kindness and wisdom. I am truly thankful for his advise in both academic and in life. I also would like to thank my committee members, Dr.Richard Thompson, Dr.David Tipper, Dr.Daniel Mossé and Dr.James Joshi ,who kindly helped me shape up my idea and complete this dissertation. I would like to especially thank Dr.Daniel Mossé for his suggestion on the $x\%$ watchdog scheme in this dissertation.

In our lives, we can't survive without friends. I have met so many good friends in my life and I can't express my appreciation to them all. However, I would like to thank all of them for their help and support. I especially thank my friends here who make me feel like home in Pittsburgh, "The City of Bridges", Pennsylvania, USA.

1.0 INTRODUCTION

Ad hoc networks have been used in many applications which mandate a dynamic network set up in the absence of fixed infrastructure. Originally, ad hoc networks were conceived for applications related to battlefields, Mars exploration, and disaster recovery [1]. The design of ad hoc networks has been mainly focused on proper operation. With technology advancements, an ad hoc device is expected to be cheap and affordable such that mass usage is possible for various applications. Thus, it is possible that eventually malicious nodes also become easy to deploy in ad hoc networks and they cause among other impacts, performance degradation. One easy attack that can be deployed by malicious nodes is the "packet dropping attack", which is an attack where a malicious node participates in routing information exchanges but it drops all data packets passing through it. This attack causes a significant performance reduction in ad hoc networks, especially in static networks. There are several other possible attacks as discussed in [2], [3], [4], [5], [6] and [7].

1.1 PACKET DROPPING ATTACKS IN AD HOC WIRELESS NETWORKS

In ad hoc networks, a node performs both terminal and routing functions to form an infrastructureless network. Therefore, a node is assumed to be helpful to other nodes to forward packets toward the correct destination. When a node does not forward packets for others, but silently or intentionally drops them, it is called a "packet dropping attack."

In ad hoc networks, packets may be dropped for several reasons (outside of honest causes such as collisions, channel errors, or buffer overflows). First, packets are dropped in the

situation when a node aims on saving its own energy. This is mainly because, in a wireless environment, the most energy is consumed in the transmit mode. If a node does not forward packets, it does not use its own energy for packet transmission and preserves its energy longer.

Second, when a node is trying to save its bandwidth, the packets are dropped. Bandwidth is also considered as a scarce resource in a wireless environment. If the node does not forward packets for others, it will have more bandwidth to send (or receive) its own packets. Therefore, its transmission capacity will be increased. In both scenarios, the node is regarded as selfish, where it does not purposefully plot to degrade the network throughput but acts to preserve its own resources [8] and [9].

Third, a malicious node can deploy a blackhole or wormhole attack to drop packets. In a blackhole attack, a malicious node exchanges routing information with other nodes but it will drop all data packets intentionally. In a wormhole attack, two or more nodes in the network form a wormhole using a wired connection, other wireless channels, or exploiting protocol vulnerabilities in order to create a route that is either really short or appears to be short such that a source believes that the best way to reach a destination is to go through the malicious nodes. In all cases, all data packets will be dropped without considering energy or bandwidth saving factors. Other malicious attacks can also cause packets to be dropped as well.

As mentioned earlier there are honest reasons why a node may simply drop data packets after correctly participating in route discovery. Packets are dropped if a node malfunctions and cannot perform the regular function of forwarding packets. Such node behavior is unpredictable. When a network is congested, packets cannot be forwarded to other nodes and they are also dropped. Congestion in ad hoc networks could occur depending on ad hoc network applications. Lastly, wireless channels are known to be unreliable. Burst channel errors due to interference, fading, etc. could occur while a node is sending packets over an open air interface. Like interference, when a network is jammed, data packets cannot be sent or received at any node in a jammed area. Packets from a non-jammed area cannot be sent through the jammed area and they are also dropped. However, the nodes in the jammed area may not have intentions to drop packets otherwise.

A summary of packet dropping is shown in Figure 1.1. Throughout the body of this dissertation, we will mainly focus on the characteristics of the attack without considering the causes of packet dropping because the final result is performance degradation in ad hoc networks. A node which drops all data packets is called a malicious node in this work.

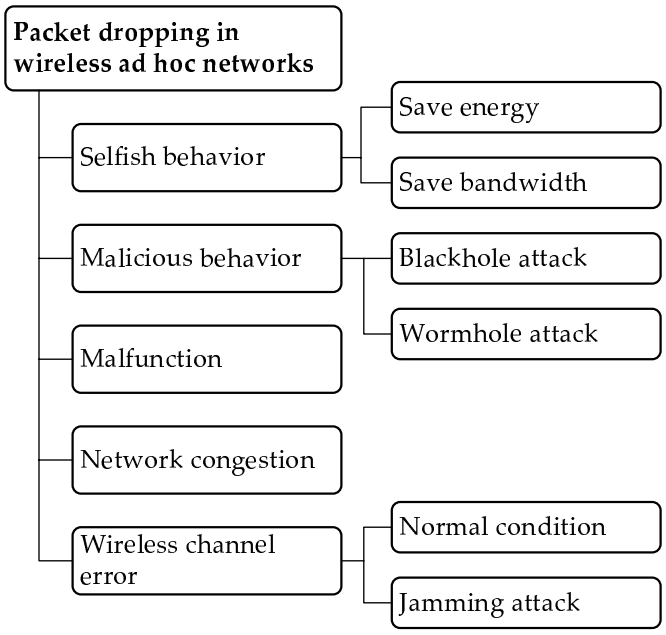


Figure 1.1: Packet dropping summary

1.2 RESEARCH BACKGROUND

Originally, an ad hoc network was conceived for military and disaster recovery applications due to its characteristics for rapid deployment and absence of infrastructure. The design of ad hoc networks mainly focused on the usability and performance of the network. As technology advancements bring about cheaper devices and better energy efficiency, the use of ad hoc networks in wider areas of applications such as wireless Internet services becomes likely. Individuals may use ad hoc devices for group communications in an ad hoc fashion without centralized management. In this case, a node can either be cooperative or uncooperative to its neighbors in order to save its own energy or bandwidth. With slight

modifications, a device could be easily changed to become selfish or malicious (e.g., by not forwarding packets to its neighbors). The packet dropping attack can largely reduce the overall network throughput.

A large body of existing literature in ad hoc networks addresses misbehaving node detection mechanisms. However, most proposed mechanisms require additional computational power from all ad hoc nodes participating in the same network. Such detection mechanisms may not be suitable for practical use due to the extra intensive computational processes that need to be implemented in all ad hoc devices (which are normally resource limited).

In one of the earliest detection approaches, Marti et al. proposed simple mechanisms, called watchdog and pathrater, to detect a misbehaving node in order to choose the most reliable path for packet forwarding [10]. In watchdog mechanism, a node promiscuously listens to its next-hop neighbor and counts the number of non-forwarded packets. If the counter is over a threshold, the source will be notified. Then, a pathrater calculates a new path using the most reliable link and uses it to send subsequent packets. The performance of the proposed detection mechanism has several weaknesses, i.e., ambiguous collision, receiver collisions, limited transmission power, false misbehavior, collusion, and partial dropping, as described in [10]. The COllaborative REputation mechanism (CORE) [11] and Cooperative Of Nodes: Fairness In Dynamic Ad-hoc NeTworks (CONFIDANT) [12] protocols propose to use a reputation system for improvement of the detection performance.

Most of the proposed works require detection functions to be implemented by every node. This not only introduces an extra task in each node, which is generally not computationally efficient for a small device having limited resources but also introduces trust issues. If every node is detecting packet dropping attacks, and communicating the presence of malicious nodes to other nodes, this mandates an assumption of trust between every pair of nodes in the network. Without such a trust relationship assumption and implementation of security measures to verify and enforce trust, a malicious node in the system could easily falsify the reputation value [13].

1.3 MOTIVATION, APPROACH AND RESEARCH CONTRIBUTIONS

In summary, most of the research literature focuses on implementing packet dropping attack detection mechanisms in every node regardless of the node's capability or consideration of trust issues. A simpler detection technique is preferable for resource limited devices. Tradeoffs should be considered between the detection effectiveness and efficiency of the detection mechanism. A question that needs to be answered is whether it is possible to have fewer nodes that detect packet dropping attacks, and what impact it has on the system's performance. This is the focus of this dissertation.

This work is motivated by the above question as well as the following observations. In real world scenarios beyond ad hoc networks, police officers are trusted agents who are authorized and dedicated to catch criminals while civilians only carry out their regular task. In this research, we consider the deployment of special nodes that are dedicated towards the detection of packet dropping attacks as explained next. The use of special nodes is also motivated by the recent research area of opportunistic networking [14] that has increasingly gained more attention in the networking research community. Opportunistic networking introduces heterogeneity in the type of nodes into the network in order to improve the overall network performance. For example, mobile nodes in a sensor network could be used to collect and disseminate data in a sparsely connected quasi-static network.

Inspired by the real world scenario and the ideas from opportunistic networking, a malicious node detection mechanism is proposed in this dissertation, which we call the *Cop* mechanism. In this mechanism, nodes are classified into two categories: a *Cop* node and a regular node. A *Cop* node acts like police officers to opportunistically detect a malicious node or an uncooperative node while regular nodes only perform normal functions without worrying about detecting attackers. This scheme reduces the load for detection function in all regular nodes. The *Cop* detection mechanism aims on mitigation of the impact of packet dropping attacks launched by a malicious node. The idea here is as follows. The *Cop* uses a threshold-based approach with the Dynamic Source Routing protocol. The *Cop* passively operates in a promiscuous mode. When a malicious node is detected, a *Cop* node sends an alarm message to a corresponding source, which will try to route all subsequent packets

through a diverted route that does not pass the malicious node. Therefore, the network throughput is increased. However, this scheme has some tradeoffs, i.e., the use of special nodes in the network (increasing the number of nodes to be deployed) and longer detection times.

The following is a list of contributions of this dissertation:

- Studied the effect of packet dropping attack on performance in wireless ad hoc networks using both analysis and simulations
- Proposed the use of a special node (heterogeneity in the network) to perform detection of packet dropping attacks in wireless ad hoc networks
- Analyzed the throughput for watchdog and cop mechanisms in static ad hoc networks using a probability tree and probability weighted averages (see Chapter 4)
- Compared detection mechanisms in various wireless scenarios
- Identified system parameters for choosing a detection mechanism in the network and parameters to guard against the attack
- Recommended design guidelines to facilitate selection of an appropriate detection mechanism and parameters

1.4 ORGANIZATION

This dissertation is organized as follows. Chapter 2 reviews the background and literature on ad hoc networks and attack mitigation. Chapter 3 presents a system model of detection mechanisms, namely the watchdog and cop mechanisms along with pseudo-codes for node and detection mechanism implementation. Chapter 4 presents a simple throughput analysis for a static ad hoc network with and without the detection mechanisms. Chapter 5 studies the detection mechanisms in various scenarios using simulations. Based on the studies in Chapters 4 and 5, Chapter 6 gives a design guideline for choosing a proper detection mechanism and parameters for selection. Lastly, contributions and future work are described in Chapter 7.

2.0 BACKGROUND AND LITERATURE REVIEWS

This chapter first presents background on ad hoc wireless networks and routing issues. Then, existing literature on malicious node detection mechanisms will be explained in greater detail. However, the focus of this work is detection mechanisms for packet dropping attacks, which include mitigation of the impact of selfish nodes or malicious nodes.

2.1 BACKGROUND

2.1.1 Introduction to Ad Hoc Wireless Network

An ad hoc wireless network is an infrastructure-less network, which can be built on the fly in a dynamic fashion. Unlike a wireless infrastructure network, an ad hoc network has certain characteristics such as self-organization, lack of fixed infrastructure for support, and communication using multi-hop routes [1].

An ad hoc network is a so-called a self-organized network because typically there is no central authority among ad hoc mobile nodes. Therefore for all ad hoc nodes participating in the same ad hoc network to be able to function and communicate, each node must implement common functions such as addressing, routing, power control, etc. Another important characteristic of an ad hoc network is the ability of a mobile node to move freely while still connecting to other mobile nodes within the same network in an ad hoc fashion. Specifically, a mobile node can move in any direction and is still able to participate in any communication. Ad hoc networks may or may not comprise of mobile nodes, or may have a mix of mobile and static nodes.

In term of power consumption, a mobile node usually operates with limited battery power and reduced computational capability. If a mobile device uses high computational power or increased communication, the battery will be dramatically drained out. Therefore, a trade-off between the computational capability and power consumption in an ad hoc device is always of interest. A sender and a receiver can be anywhere in an ad hoc network. However, it may not possible for a sender to communicate with a receiver directly (i.e., over a single hop) due to the limited distance of radio coverage. Thus, a transmitted packet may traverse several hops to reach an intended receiver. Each intermediate node must forward the received packets to its neighbor to complete the communications between a sender and a receiver.

Another example of an infrastructure-free wireless network is a Wireless Mesh Network (WMN), which is used for providing low-cost Internet services. A WMN organizes its own topology to suitably integrate several different types of networks together, i.e., the Internet, and WLAN (Wireless Local Area Network) networks in order to form seamless communications. It consists of mesh routers and clients [15]. Mesh routers usually have limited mobility. They are responsible for performing routing functions, to form a backbone of the network, and interfacing with other networks. Mesh clients can be static or mobile while accessing network resources via mesh routers. Both mesh routers and clients are normally connecting to each other via wireless links to gain access to gateway routers which connect to the Internet or other networks. Typically, mesh clients can communicate among themselves via a mesh router.

There is no wired-infrastructure within the WMN. Therefore, each mesh router is normally assumed to forward packets for others to create a well connected network. However in WMN, mesh routers could be owned by several authorities, companies, or even individuals who add their wireless equipment to the network. Therefore, a cooperative assumption may not be valid since a mesh router may not forward packets to its neighbors in order to save bandwidth for its own communications as well as to preserve its energy resources. Such a non-cooperative router is regarded as a selfish or a malicious router in this dissertation as explained in Chapter 1. As we show later, the throughput performance can be impacted significantly in the presence of selfish or malicious routers depending on their locations in the network.

In this dissertation, both ad hoc networks and WMN backbone networks are studied in order to show how the performance degrades in the presence of malicious nodes. There exist some differences between ad hoc networks and WMN backbones. Ad hoc networks are concerned largely with energy efficiency since an ad hoc node is typically a small and resource limited device. A WMN backbone network is concerned more with efficient usage of bandwidth since mesh routers must form a well-connected backbone network. However such routers are typically at fixed locations, where the power availability is not a problem.

2.1.2 Ad Hoc Routing

In a wireless infrastructure network, a packet is routed to a destination via routers. However, there is no special routers in an ad hoc network and, hence, each ad hoc node must perform routing functions in order to forward a packet to the destination. A routing protocol is, therefore, needed to complete the communications. Based on route creation, routing protocols can be classified into two categories which are (a) proactive and (b) reactive routing protocols. A proactive routing protocol is a routing protocol that creates a route to every node whether or not there is a packet to be sent to every destination. In contrast, a reactive routing protocol is a routing protocol that creates a route to a specified destination whenever there is a packet to be sent to that destination.

2.1.2.1 Proactive Routing Proactive routing is also sometimes called a table-driven routing since each node creates a routing table to its neighbors and other nodes within the network prior to sending a packet even though some routes may not be ever used. The routing updates are periodically sent to its neighbors in order to keep an up-to-date routing information related to the entire network. Such routing protocols are good for a low mobility ad hoc network. Since routes are not continuously changed, routing update packets are not necessarily sent very frequently. Consequently, the traffic load due to route updates can be small. This routing technique has the advantage of low set-up delay since each node already has a route to any destination within the network and packets are ready to be sent out at anytime. However, the route information has to be maintained even though it may

not be used. The routing load in the network is unnecessarily increased if the nodes are mobile. A most commonly known proactive routing protocol for ad hoc networks is the Destination-Sequenced Distance Vector (DSDV) protocol [16].

The DSDV protocol uses a packet sequence number to identify freshness of route information. Each node is responsible to maintain its own routing table. Periodically, routing update packets are sent to all neighbors. For each routing update packet created, the sequence number is typically increased to make sure that the packet is the latest routing update packet.

2.1.2.2 Reactive Routing Reactive routing, also called on-demand routing, creates a route only if there is a packet to be sent to a particular destination. In this case, it is not necessary for an ad hoc node to update its routing table periodically. Thus, reactive routing is suitable for a highly dynamic ad-hoc network (e.g., high mobility) since the network topology could be changed dramatically and the routing protocol must adapt to the changes quickly. Nevertheless, this approach has a certain amount of delay that arises due to the setting up a route before any packet transmission. Examples of reactive routing protocols are Dynamic Source Routing (DSR) protocol [17] and Ad hoc On-Demand Distance Vector (AODV) routing protocol [18].

In the DSR protocol, when a node has a packet to send (and if it does not have a route in its route cache for the destination under consideration), it will broadcast a route request (RREQ) packet to its neighbors. Neighbor nodes must append their own address in the RREQ packet and rebroadcast the RREQ packet to their neighbors. Finally the RREQ packet is received at the destination and the intended destination node will send a unicast packet (a route reply (RREP)) back to the sender over the reverse path of the source route information that is recorded in the RREQ packet that it receives. To enhance the performance of DSR protocol, additional mechanisms are used to learn new routes by operating in a promiscuous mode to overhear any communications within its range as well as snooping route reply (RREP) packets and then adding a new source route to its routecache. This could save time and traffic overhead to send a route request in later communications.

AODV takes advantage of both DSR and DSDV protocols. It operates in an on-demand

fashion similar to DSR and uses a sequence number and a routing table similar to DSDV. When a sender has a packet to send, it will search its routing table for a route record to the intended destination that has not expired. If a route record exists, it will send a packet directly to the designated neighbor which is the next hop to the destination. If there is no route record, it will broadcast a Route Request (RREQ) packet to its neighbors. When an intermediate node receives this packet, it updates its routing table with the sender ID, which is its current neighbor, and then rebroadcasts the packet to its neighbors. When an intended receiver receives the RREQ packet, it checks the sequence number in the RREQ packet. If this packet has a higher sequence number than was previously observed, this RREQ is a fresher packet. The receiver notes the new sequence number. Then, it sends a Route Reply (RREP) packet back to the sender.

Typically, the AODV packet has a smaller packet header than DSR since each intermediate node has its own routing table and there is no source route information in each packet header. In addition, it generates lesser traffic than DSDV since nodes do not have to maintain routing records to all nodes in the network and simply maintain routing records for destinations that are currently of interest.

In this dissertation, we use DSR as the routing protocol since it readily provides information about all nodes that are along a route from a source to the destination. The work is general, in that other routing protocols can also be considered for this work, but will need modification to enable nodes to obtain the necessary information related to nodes on the path to the destination.

2.2 LITERATURE REVIEW OF MITIGATION OF THE IMPACT OF MISBEHAVING NODES IN WIRELESS AD HOC NETWORKS

Misbehaving nodes are nodes that do not perform normal node operations as previously discussed in Chapter 1. They do not conform to normal operation and can cause network performance degradation. In order to mitigate the impact of misbehaving nodes, several mechanisms are proposed to solve the problem. The proposed works to mitigate the impact

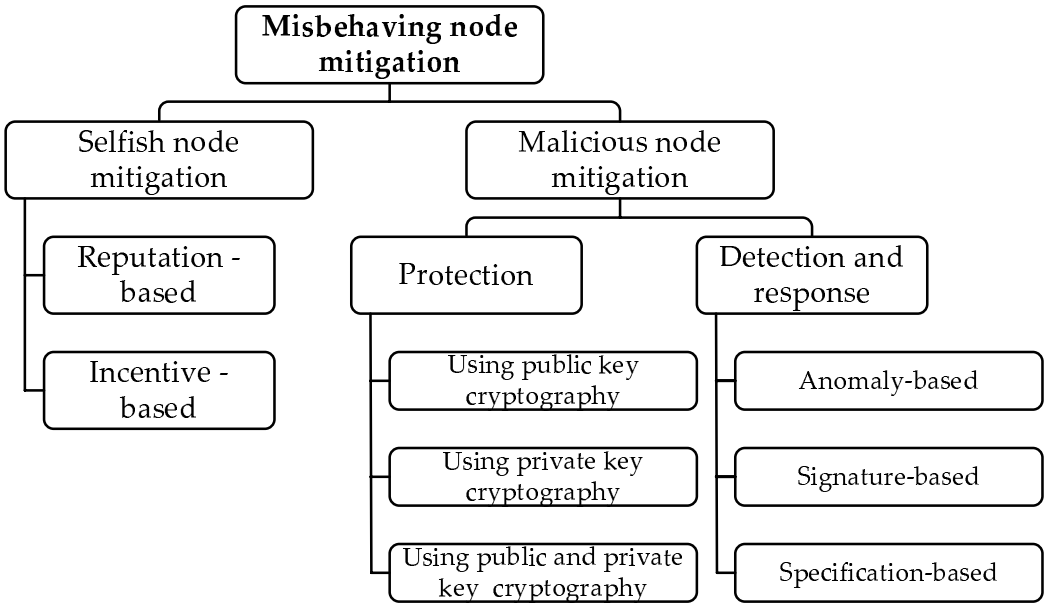


Figure 2.1: Misbehaving Node Mitigation Summary

of misbehaving nodes can be categorized into selfish node and malicious node mitigation as summarized in Figure 2.1. In this chapter, mitigation of packet dropping attacks is the focus. However, this attack behavior is similar to a selfish node behavior and the selfish node mitigation is explained along with the malicious node mitigation.

2.2.1 Selfish Node Mitigation

Many research works contribute to selfish node mitigation in ad hoc networks. The proposed mechanisms can be categorized into two approaches namely (i) incentive-based approaches, and (ii) reputation-based approaches. An incentive-based approach aims on discouraging a node from becoming selfish. A reputation-based approach aims on detecting a selfish node and responding with appropriate action.

2.2.1.1 Incentive-based mechanisms In the incentive-based approach, a currency-based or an incentive mechanism may be applied. Senders or receivers must pay for packets

to be forwarded using *virtual money* or *credits* [19, 20, 21, 22, 23, 24, 25]. This mechanism encourages a node to participate in packet forwarding activities. If a node does not forward packets of others, it may not collect enough credits to send or receive its own packets.

Buttayan and Hubaux proposed the use of virtual currency, called nuglets [19]. When a node forwards a packet for others, a nuglet counter of that node is incremented by one. When a source node wants to send a packet, it must have enough nuglets (e.g., more than the number of intermediate nodes required to reach a destination). If a source node has enough nuglets, it can send packets. Otherwise, it must first collect more nuglets by forwarding packets from other source nodes. A tamper proof device must be used to maintain the nuglet counter. Miranda, et al. proposed an algorithm to discourage selfish behavior by keeping track of each neighbor's state. Three states – friend, foe or selfish are used [20]. Each node continuously monitors its neighbors' behavior and periodically exchanges a control message containing every neighbor's state information with its neighbors. Upon receiving a control message, each node updates its neighbor's state information as specified in the proposed algorithm. Even though, a decentralized algorithm is used to avoid complicated payment schemes, the overhead is still high due to periodic broadcasting of control messages by every node within the network.

Zhong proposed "Sprite", a simple cheat-proof credit-based system, for use in a centralized manner [21]. In the Sprite mechanism, a node will keep the "record of receipt" at the reception of a packet. From time to time, a node must report the recorded information to a Credit Clearance Service (CCS), to determine the amount of charges or credits to every node involved in the packet transmissions.

Raghavan, et al. proposed a priority forwarding mechanism for "self-interested" nodes to participate in packet forwarding [22]. Packet forwarding is classified into priced priority forwarding and unpriced best-effort forwarding. A node that forwards priority packets will receive credits. Each node will be charged, from the credit account, a certain price in order to send priority packets. Nodes may send packets with no charge using a best-effort mode. This scheme tries to motivate each node to forward priority packets to gain more credits for their own packets. A node with no credits can also send packets but other nodes treat its packets with best-effort forwarding only.

Crowcroft proposed to use congestion prices, i.e., bandwidth and energy, for modeling of an incentive scheme [23]. The proposed mechanism deploys directional wireless antennas to send packets through multiple routes to a destination. Power and bandwidth prices are dynamically adapted using a rate control model in [26] and it operates in a distributed fashion. Specifically, each node updates its prices based on current power and bandwidth usage.

Focusing on fairness as well as collaboration, the Fee Arbitrated Incentive Architecture (FAIR) is proposed [24]. The term fairness here means that the amount of benefits that a node will receive is proportional to the amount of contribution that a node makes towards network operation. The node must collaborate to help other nodes route packets to their destinations in order to send its own packets. Feedback schemes are proposed to dynamically adjust the FAIR performance. Three resource constraints, including energy, bandwidth and processor constraints, are used in performance optimization.

Typically, a motivation-based approach uses a per-packet-based approach to model the credit or incentives, but Zhang proposed a Secure Incentive Protocol (SIP) that uses a session-based approach [25]. However, SIP assumed the use of a tamper-proof module, similar to the work in [19]. In the SIP protocol, a session initiator and a session responder (a source-destination pair) will be charged for a service and intermediate nodes are rewarded with credits when they forward packets for a source-destination pair. SIP consists of three phases, i.e., Session initialization phase, Data forwarding phase and Rewarding phase. Each intermediate node is awarded a number of credits based on the number of forwarded packets. Asymmetric key cryptography is used for securing the protocol from credit fraudulence and other attacks.

Normally in an incentive scheme, a source node does not know the exact price to pay for sending a packet but Hauspie and Simplot-Ryl proposed to use a virtual money mechanism over a route discovery protocol for a source node to know the exact price to be charged for each packet sent on a route [27].

The proposed works described above attempt to solve the problem of selfishness by means of algorithms that motivate a selfish node to forward packets. A game-theory model can be applied to this problem assuming that the nodes are rational. Each node tries to

maximize its own benefits or utility (for example, throughput and energy). Formal game theoretical models have been proposed to motivate a selfish node to forward packets in order to optimize the throughput perceived at each node as well as to motivate it to cooperate. Various proposed mechanisms use different game strategies based on different design goals.

Srinivasan proposed to use a Generous TIT-FOR-TAT (GTFT) strategy in an acceptance algorithm which is used to determine whether to accept or reject a relay request at each node in order to optimize the network throughput [28].

Eidenbenz proposed connectivity and reachability games for topology control games in static ad hoc networks [29]. In this work, a node must choose its radius to reach a destination and the choice of radius is a strategy. Urpi modeled a forwarding strategy using a Bayesian game for a frame level cooperation depending on their energy level [30].

Anderegg proposed a new routing protocol, called Ad hoc-VCG, to guarantee the discovery of the most cost-effective route [31]. It is a reactive routing protocol with the design objective of truthfulness. This protocol is robust against a single cheating node but it may fail in a scenario when two or more selfish nodes cooperate in maximizing their total payments because they can collaborate in an overpriced payoff.

Ileri proposed a pricing algorithm to encourage node forwarding by reimbursing forwarding based on user-and-network centric incentive mechanisms [32]. The objective is to maximize network and utility in bits per joule with the prices of channel use, reimbursement forwarding, transmitter power control and forwarding and destination preferences.

Cai proposed a CAP-SV protocol (Contribution rewArd routing Protocol with Sharpley Value) [33]. In CAP-SV protocol, a payment scheme uses Sharpley-Value, a well-known concept in game theory, to allocate pay-off for each node. The same authors proposed another protocol, called Transmission power rEcursion Auction Mechanism (TEAM) routing protocol, to motivate selfish nodes to focus on transmission power efficiency and truthfulness [34]. TEAM is designed to optimize the aggregate transmission power between a source and a destination. It shows a significant message complexity improvement over the Ad hoc-VCG protocol.

Zhong proposed integration of game theory with routing and cryptographic techniques towards a secure and incentive routing protocol [35]. The key idea is that the link cost is

determined by two nodes in order to charge an appropriate price without cheating, and the protocol uses hash chains to deliver payments securely.

Eidenbenz proposed a COMMIT protocol based on VCG payment scheme [36]. In COMMIT protocol, a sender must commit for a maximum affordable payment before the request is processed. If a network can accept the offer, then, a sender can send a packet. Marbach proposed to use game theory for relaying data packets in the scenario where a node is allowed to freely determine the price and bandwidth usage for both its own and others' traffic [37].

Bandyopadhyay proposed to use a repeated game for each packet to model the cooperation problem [38]. In each stage of the game, a node can choose to cooperate or not cooperate but it will be punished if it does not cooperate for several stages, called a "punishment" phase. However, if it decides later on to cooperate, it has to forward packets for others for several stages, called a "parole" stage and after that other nodes will forward its packets in a "rehabilitation" phase.

In summary, these proposed incentive algorithms aim towards solving the cooperation problem in ad hoc networks but they also introduce complexity in ad hoc nodes. Huang, et.al. suggested that a motivation-based approach is complex and should be used only for specific applications rather than general applications [39].

In addition, incentive schemes may not be fair to all nodes, especially nodes located at the edge of a network, since some nodes may not get enough credits to send their own packets when they do not have much chance to forward packets for others being located at the edge. The cooperation problems are modeled as mathematical models using game theory. However, most models are based on several assumptions which may not be practically feasible in a real ad hoc network. Moreover, the complexity, need for tamper-proof chips, etc. of this approach are key issues which makes it impractical for use in a real network.

2.2.1.2 Reputation-based mechanisms A reputation-based mechanism usually uses a reputation system in order to detect and rate a selfish node. Reputation can be defined as the performance of a node participating in the base protocol as seen by others [40]. A node's reputation is used to decide whom to trust and to encourage every node in a network to be trustworthy. Typically, a selfish node (or malicious node) may drop data packets to

preserve its energy (or harm the network performance). A detection mechanism will use the statistics of non-forwarded packets at each node to determine whether a node is a selfish (or malicious) node.

Wireless ad hoc networks use air as the communications medium where signal typically propagates in all directions due to the use of omnidirectional antennas. Therefore, if two nodes are within communications range, they can communicate directly to each other. A passive acknowledgement (PACK) technique uses this property wherein a node overhears its own packet being forwarded by its neighbor since all neighbors must be within range. The "Watchdog" mechanism [10] was proposed to be used over the DSR routing protocol. It monitors all neighbors' behavior by operating in a promiscuous mode. A sender listens to its neighbors and keeps counting the number of non-forwarded packets in order to detect whether its neighbor forwards recently received packets or not. If the counter is over a threshold limit, it will report to a pathrater mechanism. After the detection phase, the "Pathrater" mechanism [10] is used in a response phase to evaluate each path to ensure that packets from a sender to a receiver are forwarded through the most reliable path, or a path with a high-rating.

There is no punishment action in Watchdog/Pathrater directed towards a selfish node. Simulation results show that the detection-based approach gives a throughput of 82% while with a normal DSR protocol the throughput is 68% in a scenario where mobile nodes have a 0 pause time and 40% of nodes are selfish. A major drawback of this mechanism lies in its battery power consumption since every node has to constantly listen to the medium [41].

"CORE", a collaborative reputation mechanism [11], is proposed that uses "Watchdog" as the monitoring mechanism. A reputation table is created at each node to keep track of reputation values of other nodes. Only positive rating factors can be distributed among nodes since a selfish node may send false negative rating factors to other nodes and may cause disruption of the reputation system. This mechanism devises a penalty action towards a selfish node by denying all services given to it. Michiardi proposed the use of cooperative game and non-cooperative game approaches to evaluate the effectiveness of CORE mechanism to detect and rate neighbors for their reputation [42].

Another proposed reputation mechanism is "CONFIDANT", Cooperation Of Nodes:

Fairness In Dynamic Ad-hoc NeTworks [12]. CONFIDANT has four main components namely a monitor, a reputation system, a path manager, and a trust manager. These components are required to be implemented in every node. Each node monitors its neighbors by listening to the transmission of the next node or by watching routing protocol behavior. A trust manager is used to manage ALARM messages, which are sent when a misbehaving node is detected. The reputation system is used to rate every node in a network. A path manager is responsible to rank a path according to a security metric, e.g., reputation of the node in the path and to get rid of any path containing a selfish node. In addition, a path manager will penalize a selfish node by denying all services to it. Through a study the protocol performance, the authors showed that the throughput given by CONFIDANT in a scenario when a third of nodes behave selfishly is very close to the throughput of a normal network condition without selfish nodes.

"CineMA", Cooperation Enhancement in MANETs, is proposed to respond to a selfish node by limiting the number of packets forwarded by it [43]. CineMA uses the same penalty scheme as in CORE and CONFIDANT. Unlike CORE and CONFIDANT, CineMA only needs a group of nodes to perform necessary functions. It consists of three main modules including a Watchdog module, a reputation system module, and an interface queue module. A Watchdog module performs system monitoring to collect information. A reputation system uses collected information to determine the level of cooperation based on the number of received packets and the number of forwarded packets. These values are also used, at the interface queue module, to limit the amount of packets which a selfish node is allowed to transmit. CineMA requires the use of a cryptographic mechanism to ensure secure communications among all nodes implementing CineMA functions. Although, overall throughput and performance of CineMA have not been proven, a major advantage of CineMA is that it can limit the sending rate of a selfish node.

All previous works discussed so far use Watchdog mechanism as a major part of the protocols. The watchdog mechanism itself has several weaknesses, i.e., ambiguous collisions, receiver collisions, inability to detect packet drops using limited transmission power, false misbehavior, collusion and partial dropping [10].

Therefore, a "TWOACK" mechanism [44],[45] is proposed in replacement of the Watch-

dog mechanism and it is a network-layer acknowledgement-based scheme that can be added to any source routing protocol. TWOACK mechanism can guarantee that a data packet has reached a node two hops away. A TWOACK acknowledgement packet is introduced to notify a node that a sent packet is forwarded to a node that is two hops away. If the node does not receive any TWOACK packets within a specific timeout period, a counter is incremented by the number of non-forwarded packets. If a selfish node is detected, a route error (RERR) packet will be sent back to the source node. Other nodes can overhear RERR packets and keep a record of a selfish node such that a path through such a node will be avoided. The study shows that in a scenario when 40 percent of nodes behave selfishly, TWOACK delivers throughput of 85-90% while regular DSR without a selfish node delivers a throughput of only 70-75%. This, however, requires additional transmissions (the TWOACK) for every transmitted packet which incurs additional energy penalty compared to the Watchdog mechanism.

Table 2.1 summarizes all detection based mechanisms proposed in the literature.

Table 2.1: Reputation-based Approach Summary - Selfish Node Mitigation

Literatures	**Based Protocol**	**Highlights**	**Detection Functionality**
Watchdog [10]	DSR	• Using watchdog mechanism and avoid using a selfish node in a path	All nodes
CORE [11]	All	• Using weighted average rating to combine direct and indirect reputations • Only positive reputation is exchanged • Selfish node is isolated upon detection	All nodes
CONFIDANT [12]	DSR	• Using weighted average rating to combine direct and indirect reputations • Alarm is sent upon detection • Selfish node is isolated from network	All nodes
CineMA [43]	DSR	• Use number of received and forward packets as level of cooperation • Limit the network usage upon detection	Some nodes
TWOACK [44]	DSR	• Using TWOACK packets to guarantee forwarding packets in two hops • Avoid using a selfish node path	All nodes

2.2.2 Malicious Node Mitigation

Malicious node mitigation can be classified into two categories: (i) prevention and protection, (ii) detection and response. A prevention mechanism guards against a malicious node's attack by applying cryptographic mechanisms such as encryption and authentication. However, it cannot guard against insider attacks. A detection and response mechanism detects misbehavior activities and responds to an attack.

In this dissertation, the main focus is on addressing detection and response mechanisms as well as their impact on overall performance of an ad hoc network.

2.2.2.1 Protection Mechanisms A protection mechanism applies cryptographic techniques to secure communications over an ad hoc network in order to prevent any malicious activity. Most research works focus on securing a routing protocol which is a key component for a wireless ad hoc network to operate properly. The two most important security services for a secure a routing protocol are authentication and data integrity services. Various types of secure routing protocols can be classified, based on the usage of cryptographic techniques, into three categories including (i) usage of secret key cryptography, (ii) usage of public key cryptography and (iii) usage of both secret key and public key cryptography. A summary of protection mechanisms is shown in Table 2.2.

1. Using Secret Key Cryptography :

A secure routing protocol uses secret key cryptography to protect the routing information through the means of encryption. When secret key cryptography is applied, a secret key has to be pre-shared or distributed off-line before a secure communication can be established. Each node will then use a secret key during a communication session with the other party. Such secret keys can also be used to establish trust relationships, but key distribution will be expensive here.

Typically, in secure routing over an ad hoc network, authentication is more important than confidentiality. For this reason, Perrig, et. al., proposed a broadcast authentication protocol called Time Efficient Stream Loss-tolerant Authentication protocol (TESLA) [46]. In TESLA, each packet is signed using a secret key. Then, the key will be disclosed within

a specific time interval. Therefore, any node can validate all packets received during that time interval. This scheme required that all nodes must maintain time synchronization. In this case, a secret key has to be shared between a sender and a receiver.

TESLA has been used in various ad hoc routing protocols including Ariadne based on the DSR protocol [47] and Secure Efficient Ad hoc Distance Vector (SEAD) based on the DSDV protocol [48]. In addition, TESLA can be applied to guard against a strong attack, called a wormhole attack using Packet Leashes [3], which adds information, e.g. synchronized clock, to each packet to restrict the maximum travel distance of a packet.

Although, TESLA can securely protect against a malicious attack, it is considered not practical because time synchronization among all independent ad hoc nodes is very difficult to achieve.

Another approach, called A Secure Routing Protocol (SRP) [49], proposed to use a Message Authentication Code (MAC) for verification of packet authenticity. Typically, the MAC is added to secure the header field of existing on-demand routing protocols.

2. Using Public Key Cryptography

Public key cryptography is applied for data and identity authentication using a digital signature. Before sending out a packet, a sender node must sign a digital signature using its private key. Then, any node can, later, verify authenticity of a received packet using a sender's public key. It is assumed that public keys of all participating nodes must be known in advance. Therefore, a Certificate Authority (CA) or PGP-like key distribution is required for a public key distribution.

Secure routing protocols that use public key cryptography include Authenticated Routing for Ad hoc Network (ARAN) [50, 51], Secure Link State Routing Protocol (SLSP) [52], Secure AODV [53], AODV-S [54], SecAODV [55], Security-aware Adaptive DSR Protocol (SADSR) [56], and Secure DSR (SDSR) [57].

Generally, public key cryptography is computational intensive [58] and may not suitable for resource-limited devices.

3. Using both Secret Key and Public Key Cryptography :

Another proposed approach applies both secret key and public key cryptography for a secure communications. Secret key cryptography is used for a secure routing information exchange and public key cryptography is used for message and identity authentication as well as secret key distribution.

Both secret key and public key cryptography are suggested to be used in several protocols. Bhargava [59] and Awerbuch [60] proposed to use secret key cryptography for confidentiality and public key cryptography for authentication. This approach combines advantages of both cryptography techniques. However, it still requires a high computational capability and consumes the battery power in all ad hoc nodes.

Table 2.2: Protection Mechanism Summary - Malicious Node Mitigation

Security Mechanism	**Literature**	**Based Routing Protocol**
Secret Key Cryptography	Ariadne [47]	DSR
	SEAD [48]	DSDV
	Packet Leash [3]	-
	SRP [49]	Reactive
Public Key Cryptography	ARAN [50][51]	Reactive
	SLSP [52]	Proactive
	SAODV [53]	AODV
	AODV-S [54]	AODV
	SecAODV [55]	AODV
	SADSR [56]	DSR
	SDSR [57]	DSR
Both	EADM [59]	AODV
	ODSBR [60]	Reactive

2.2.2.2 Detection and Response Mechanisms Detection-and-response mechanisms are used whenever a prevention mechanism fails. The failure could occur due to a security breach by a malicious attacker or a misbehaving insider. Therefore, a detection mechanism is necessary to secure a network against such attacks as well as minimize magnitude and scope of a successful attack.

Detection mechanisms can be classified based on detection analysis into three categories including (i) anomaly-based detection, (ii) specification-based detection and (iii) signature-based detection.

1. Anomaly-based Detection

Much of the literature uses anomaly-based detection for intrusion detection mainly because the mobility characteristic of an ad hoc network makes it difficult for specification-based and signature-based detections to accurately detect an intrusion.

Patcha [61] proposed an extension to the Watchdog scheme to detect colluding nodes or a wormhole attack and improve the performance of a watchdog scheme. The proposed mechanism classifies ad hoc nodes into two types: ordinary nodes and trusted nodes. Ordinary nodes perform all regular activities. Watchdog functions are implemented only in trusted nodes. In actual implementation, an extension to AODV protocol must be done to support new packet types. In addition, three threshold counters must be maintained at nodes implementing Watchdog functions.

In the Watchdog scheme itself, detection of misbehaving nodes relies only on a passive acknowledgement hearing from neighbor nodes. Therefore, information collected may take sometime in order to determine misbehaving nodes. Overall, the Watchdog detection rate can be low.

Overseeing this problem, Kargl [9] proposed a Mobile Intrusion Detection System (MobIDS) scheme to be used over the Secure DSR (SDSR) protocol [57]. The main goal of MobIDS is to improve the detection rate. MobIDS sensors collect information from both an activity-based overhearing mechanism or a Watchdog mechanism and probing packets. Each node has to listen to the wireless channel and send probe packets to detect an attack. Therefore, this mechanism requires more memory and processing power at each node.

Another detection mechanism is proposed by Medidi, et.al. for improvement over previously proposed mechanisms to detect a packet dropping attack by correlated inter-layer information [62]. A detection manager is implemented at a source node. Two main modules are required including a data collection module and a data analysis module. The data collection module will collect local information, e.g. DSR route request, route error messages, ICMP time exceed and TCP time-out. Only useful information is then extracted and fed into a data analysis module. This detection mechanism is shown to effectively detect an attack and the false positive rate is low. However, keeping a record of data required consumes a large amount of memory as well as processing power to analyze all data.

A distributed probing technique is proposed to detect a malicious packet dropping attack [63]. Probing packets are sent regularly to various destinations. At the reception of a probing packet, a node must respond. If a node does not respond to the probing packets, it will be regarded as a malicious node. After detection, a new path must be chosen in order to avoid using the path passing through a malicious node. The author also suggested the use of cryptographic techniques to secure probing packets. The disadvantage of the probing technique is the high overhead due to the distribution of probing packets.

Wang [64] proposed an intrusion detection technique to identify "a false destination sequence number attack", where a malicious node sends a higher sequence number than a normal sequence number for the destination such that it will then use the fake routing information. Typically, DSDV or AODV routing protocols are vulnerable to such attack since both protocols use a sequence number to determine freshness of routing packets received. In this technique, a sender node keeps monitoring a route request packet broadcasted back in order to detect whether a sequence number in the route request packet has been modified.

Patwardhan [65] proposed an intrusion detection mechanism for SecAODV [55], based on threshold-based anomaly detection, to detect a data packet dropping attack, which drops all data packets but forwards routing packets. Operations of the proposed mechanism uses the Watchdog mechanism and counts the number of non-forwarded packets until it is reaches a threshold.

Overall, all of the above proposed mechanisms use simple algorithms to detect a malicious node. More complex algorithms used in detection mechanisms will be presented next.

A Sinkhole Intrusion Detection System (SIDS) [66] is proposed to detect a sinkhole or a blackhole attack in DSR protocol. Three parameters are introduced including (i) sequence number discontinuity or duplication, (ii) previous image ratio, and (iii) route add ratio. A sequence number discontinuity is an average value of differences between the current and the last sequence numbers at every node.

The previous image ratio is a ratio of the number of verified images to the number of total images received. The number of verified images is the number of route-broadcast packets received at a node from a certain neighbor that can be traced and verified to the earlier broadcast packets transmitted by other nodes having the same source-destination, sequence

number and with appropriately inserted route records.

A route add ratio is the ratio of the number of routes passing through a node to the number of routes added to the node's routing table. SIDS uses fuzzy rules to determine thresholds for the above parameters in order to detect a sinkhole attack. Specifically, appropriate thresholds can be assigned through a system training process to learn normal behavior as well as sinkhole behavior of each network topology. SIDS has shown to give high detection accuracy with zero-percent false-positives. Nonetheless, it may not be practical for a rapidly deployed ad hoc network due to the overhead in the training process.

Huang [67] proposed automatically constructing anomaly detection models by using a data mining method that uses cross-feature analysis to capture the inter-feature correlation patterns in normal traffic. Features are constructed from route related features, (e.g., route add count, route remove count, route find count) and traffic related features, (e.g., packet type, flow direction). By collecting the information from normal traffic, the feature patterns are used as normal profiles for detecting an attack. The proposed work aims to efficiently detect a blackhole attack and a selective packet dropping attack.

Zhang [68] proposed to use anomaly detection models by using information from the routing protocols, i.e., percentage of changed routes (PCR) and percentage of changes in sum of hops of all the routes (PCH). In the training process, these values are collected to create a normal profile, which is used for comparing with current activity to detect an attack. The authors suggested to use GPS to provide location and velocity of nodes for local data sources. It is not easy to have GPS in all nodes in practical networks.

Awerbuch et.al. [69] proposed a new versatile protocol called ODSBR protocols which can detect several types of attacks, i.e., a black-hole attack, a flood rushing attack, and a Byzantine wormhole attack. ODSBR protocol is a source routing protocol that uses end-to-end acknowledgements from a destination to detect the presence of a blackhole attack. After the number of lost packets is over a threshold, ODSBR enters a probing mode to detect the location of the attack. The protocol also performs a hop-by-hop authentication and integrity checking of route discovery packets to prevent the flood rushing attack. A wormhole attack appears to the protocol as a failure link between two nodes. If it is detected, a wormhole link weight will be increased in order to avoid the link. The performance is significantly

improved over AODV routing protocol. However, it also introduces a high overhead to the routing protocol.

Gonzalez et.al. [70] proposed a new detection algorithm to detect a misbehaving node, which does not forward data packets by using a flow conservation principle. They claimed to solve the weakness of the watchdog mechanism by using information from the neighbors of an analyzed node, instead of overhearing the transmission. The algorithm generates more overhead to the network for detection but the result does not show comparison with other works.

2. Specification-based Detection

Tseng proposed a specification-based intrusion detection system for AODV [71]. There are two types of nodes, a regular node and a network monitor node. The network monitor (NM) nodes are nodes that detect incorrect RREQ and RREP messages using a finite state machine (FSM). The NM nodes are distributed within the network and will listen and keep the last received RREQ and RREP messages for source and destination nodes in a tree-like structure to ease the tracking of an attack.

AODV Extended Finite State Automaton (AODV EFSA) [5] also applies the same technique using a finite state machine to keep track of events in order to detect misbehavior activities. Specification-based detection is also suggested to be used together with an anomaly-based detection to effectively detect many types of attacks in an ad hoc network. However, it introduces high complexity in the implementation.

3. Signature-based Detection

A signature-based detection called AODVSTAT [72] applies a State Transition Analysis Technique (STAT), which was developed in a wired network, for attack detection in AODV routing protocol. In STAT, an attack signature is described by a sequence of actions that an attacker performs to compromise system security. AODVSTAT uses sensors to perform a stateful analysis of packet streams to detect signs of intrusion. AODVSTAT can detect one-hop and distributed attacks to AODV routing protocol with low false-positive rates.

Table 2.3 summarizes the detection mechanisms that have been currently proposed in

research community.

Table 2.3: Detection Mechanism Summary - Malicious Node Mitigation

Literature	IDS type	Based Protocol	All Nodes?	Detected Attacks
Patcha [61]	Anomaly	AODV	Some	- Blackhole - Warmhole
MobIDS [9]	Anomaly	SDSR	All	- Blackhole
Medidi [62]	Anomaly	DSR	Source	- Blackhole
Just [63]	Anomaly	DSR	Some	- Blackhole
Wang [64]	Anomaly	DSDV, AODV	Senders	- False destination sequence
Patwardhan [65]	Anomaly	SecAODV	All	- Blackhole
SIDS [66]	Anomaly	DSR	All	- Blackhole
Huang [67]	Anomaly	AODV, DSR	All	- Blackhole - Selective packet dropping
Zhang [68]	Anomaly	AODV, DSR	All	- Packet Dropping - Routing Modification
ODSBR [69]	Anomaly	Source routing	all	- Blackhole - Flood rushing - Byzantine wormhole
Tseng [71]	Specification	AODV	Some	- Route modification - Spoofing - Dropping
AODV EFSA [5]	Specification and Anomaly	AODV	All	- Flooding - Route Modification - Rushing attack - Dropping (data+routing)
AODVSTAT [72]	Signature	AODV	Some	- Spoofing - False sequence number - Dropping packets - Resource depletion

2.3 LIMITATION OF CURRENT WORK

To mitigate the selfish or malicious node problem in ad hoc networks, specifically packet dropping attacks, two approaches are proposed – incentive-based approach, and reputation-based approach. Each proposed approach aims to either effectively detect packet dropping nodes or motivate all nodes to cooperate in forwarding data packets for others.

However, Huang, et al. claimed that incentive or motivation-based approaches are still in a developing stage and perhaps cannot be used publicly yet [39]. A reputation-based approach faces problems with how a reputation value is defined and calculated, how to detect disreputable behavior, and how to distribute the reputation information [73]. In addition, a reputation-based approach has been shown to be effective only in a static ad hoc network environment [74]. Moreover, a trust relationship has to be set up between every pair of nodes. This is because, when a node updates reputation values, a message must be sent to inform its neighbors. Without a trust relationship assumption, a malicious node can falsify a reputation value and cause corruption in the reputation system [13].

For the IDS system, most of the proposed works require a detection function to be implemented in every node. However, an ad hoc node may be a small and resource limited device. It is not computationally efficient and not practical to deploy detection functions in each node.

In this dissertation, the Cop mechanism is also proposed to detect malicious or selfish nodes that perform packet dropping attacks using only a few special nodes, called Cop nodes. Each node only trusts Cop nodes, not its neighbors. The Cop mechanism is an adaptation of the watchdog mechanism that reduces the number of detecting nodes and introduces opportunistic detection. The Cop mechanism has several advantages. First, the Cop is simple but yet effective in detecting a malicious node accurately (as we show later). Secondly, only a small modification is required in regular nodes. Trust relationships need be maintained only with Cop nodes and not between all pairs of nodes. However, it is not the best solution for all scenarios. Therefore, trade-offs between the Cop and Watchdog mechanisms are studied in this work.

3.0 SYSTEM MODELS AND EXPERIMENTAL DESIGN

From a review of the research literature, we see that a malicious node can degrade the throughput performance significantly (although systematic analysis or simulation is lacking in the research literature). A simple solution to mitigate the problem of packet dropping attacks is to use a watchdog mechanism as proposed in [10]. However, a new mechanism is called a Cop mechanism is proposed here to help save the energy consumed by deploying the detection function in every node by assigning only a few special nodes for the detection function. The Cop mechanism also reduces the number of trust relationships necessary and is motivated by the general concept of introducing heterogeneous nodes in ad hoc networks. Other nodes in the network that employ the Cop mechanism for detection are regular nodes without any detection function. Watchdog and Cop mechanisms can improve the overall network throughput by detecting malicious nodes and in the case of the Cop, perhaps assist nodes in forwarding packets that were otherwise dropped.

In the watchdog architecture, two types of nodes are considered. The first type is a regular node with a built-in watchdog mechanism to detect its neighbors' activity. The second type is a malicious node, which drops all data packets but responds to all routing information exchange. In order to save energy in the watchdog mechanism, it is possible to allow all nodes to perform the detection function with a duty cycle, that is for $x\%$ of the time. The result from this scheme will be shown in Chapter 5.

In the Cop architecture, there are three types of ad hoc nodes. The first type is a regular node, which is responsible for exchanging routing information and sending or forwarding data packets to a destination. The second type is a Cop node, which is responsible for listening to the wireless channel and detecting deviant behavior of other nodes. If a misbehaving activity is detected, a Cop node will send out alarm messages to its neighbors and also the sender

node. The Cop node optionally helps forwarding packets for a regular node. The third type is a malicious node, which is similar to the malicious node in the watchdog architecture. The numbers of Cops, malicious nodes, and regular nodes can be different, as also the nature of their placement and movement in the network.

3.1 ASSUMPTIONS

In Cop and Watchdog implementation, the following assumptions are applied.

1. Trust Relationship
   - In Watchdog implementation, all nodes trust their neighbors.
   - In Cop implementation, all nodes trust only Cop nodes but need not trust their neighbors.
2. Behavior
   - All Cop nodes always have good behavior and have higher energy than regular nodes.
3. Wireless Channel
   - There is no wireless channel error. All nodes use omni-directional antennas for bi-directional communications.

3.2 REGULAR NODE MODEL

A regular node must be slightly modified to accommodate operations of the node functioning as Cop. When a regular node receives an alarm message from Cop, it checks route-cache information and removes all paths that contain a malicious node. Additionally, it forwards the packet back to a source node, which will respond by finding a new path not passing through the malicious node. The source node has to be alerted in order to update the source route information for the next data packets to be sent. At the receiving of an alarm message, a regular node creates a badnode table to record a malicious node's address. A regular node will ignore any routing information exchange with all malicious nodes listed

in the table. The operation of a regular node is explained in a diagram as shown in Figure 3.1.

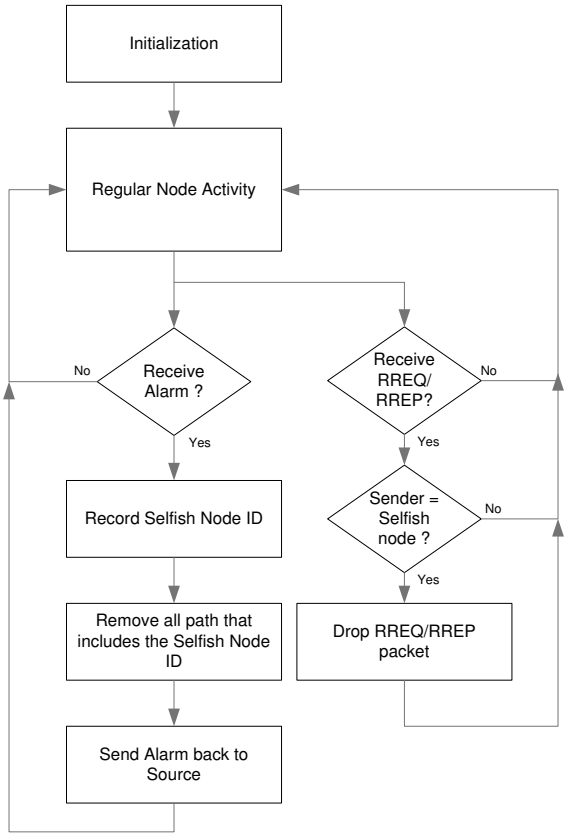


Figure 3.1: Regular Node Diagram

3.3 MALICIOUS NODE MODEL

One of the functions of a malicious node is regular routing information exchange with its neighbors to set up a route to a destination. Another main function is to drop all data packets that pass through it. In addition, it will also drop all alarm messages in order to protect itself from other nodes. The selfish node model is shown in Figure 3.2.

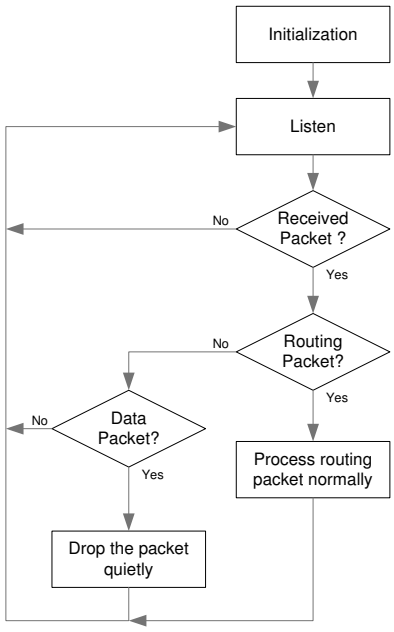


Figure 3.2: Malicious Node Diagram

3.4 WATCHDOG MODEL

A basic Passive ACKnowledgement (PACK) detection mechanism, called watchdog, is studied for comparison with the Cop mechanism. In the watchdog mechanism, all nodes are required to implement watchdog algorithm for malicious node detection. After finished the transmission, each sender or forwarder continues listening to the wireless channel to check whether its next hop neighbor forwards the packets onward to the next node or not. If its neighbor does not forward the received packet within a time-out period and the number of non-forwarding packets is over a maximum threshold, it will send an alarm message to a source node. After receiving an alarm message, a source node will find a new route which does not pass through the malicious node. Note that each node only keeps track of its own packets from its neighbors. The pseudo code of watchdog mechanism is shown in Figure 3.3.

An energy saving scheme for the watchdog mechanism is to perform the detection function (i.e., listening to see if its own packets are forwarded) $x\%$ of the time, (for e.g. 50%).

```
/* Watchdog Pseudo-code */

if sender/forwarder overhears a data packet
{
    if expected packets
    {
        recorded as a forwarded packet
        status(nexthop) = good
    }
    if sent packets Time-Out
    {
        if count(non-forwarded pkt) > threshold
        {
            if status(nexthop) != good
            {
                send alarm packet to source
                status(nexthop) = malicious
            }
        }
    }
}
```

Figure 3.3: Watchdog Pseudo code

The algorithm is changed by adding a timer check for the detection function to work as it is designed. When the detection function is off, all nodes simply perform regular node functions. When the detection function is on, all nodes detect their neighbor activities for a malicious node. This mechanism can save the energy but it could increase the detection time to detect a malicious node.

3.4.1 Watchdog Parameters

Two parameters are important for the watchdog mechanism. The first parameter is the maximum threshold and the second parameter is the time-out. The maximum threshold depends on the data rate in the network. The time-out is used for avoiding a false accusation of a regular node. The relationship between the watchdog parameters and the data rate is shown in Chapter 4.

3.5 COP NODE MODEL

The Cop architecture is implemented over the DSR routing protocol using ns-2 network simulator version 2.30. Initially, one ad hoc node acts as Cop which can detect and respond to a malicious node. Specifically, the Cop keeps moving around within a network area in order to find a malicious node. Note that a Cop can optionally help forwarding packets to its neighbors. While moving, the Cop listens to the wireless channel in a promiscuous mode and keeps records of overheard data packets. The Cop will keep track of all neighbor node behavior. If a node forwards a packet to its neighbor and Cop overhears that communications, the node status is changed from neutral to good status. If a node does not cooperate in data packet forwarding, the number of non-forwarded packets (or the reputation value) is recorded by the Cop. A malicious node is detected if the number of non-forwarded packets exceeds a maximum threshold. To avoid false accusation, a hold-down timer will be activated when the maximum threshold is reached. If the timer is expired and a suspicious node has not forwarded any packets, the Cop will send an alarm message to a source node. After receiving

an alarm message, a source node will find a new route which does not include a malicious node. However, a source node may still use a route containing a malicious node. Therefore, the Cop will resend an alarm message to the source node to ensure that the malicious node is no longer included in a route. The pseudo code of Cop mechanism is illustrated in Figure 3.4.

There are two schemes in the case of the Cop mechanism. In the first scheme, a Cop does the detection function only. The Cop node does not help in forwarding packets. In the second scheme, a Cop detects malicious nodes and helps with forwarding packets. The difference between the two mechanisms is in the implementation. Since a cop node does not forward any packets in the first scheme, it is not able to learn a route to notify a source after a malicious node is detected. The cop node then broadcasts an alarm message to its neighbors, which will find a route and forward it to the source node. This is an efficient scheme but it causes high routing overhead. In the second scheme, a cop node knows the route to the source and it can send an alarm packet directly to the source. This scheme helps reducing the routing overhead but the alarm message is prone to be dropped by any malicious nodes that have escaped detection.

3.5.1 Cop Parameters

From the Cop implementation's perspective, the performance of Cop depends largely on three parameters, (1) the maximum threshold, (2) a hold-down timer, and (3) Cop's traveling speed. The maximum threshold parameter depends on the traffic load in the network. The hold-down timer is used for avoiding a false accusation when a cop is not yet in a coverage area of a regular node but the maximum threshold is reached. Therefore, the hold-down timer parameter depends on the traveling speed of the Cop. These three parameters will affect the malicious node detection time as well as the false positive rate. Therefore, these parameters must be set correctly according to the network load and the speed of the Cop node.

To illustrate the effect of these parameters on the Cop performance, we use an explanation with sample results shown in Figure 3.5. The average numbers of sent packets and received

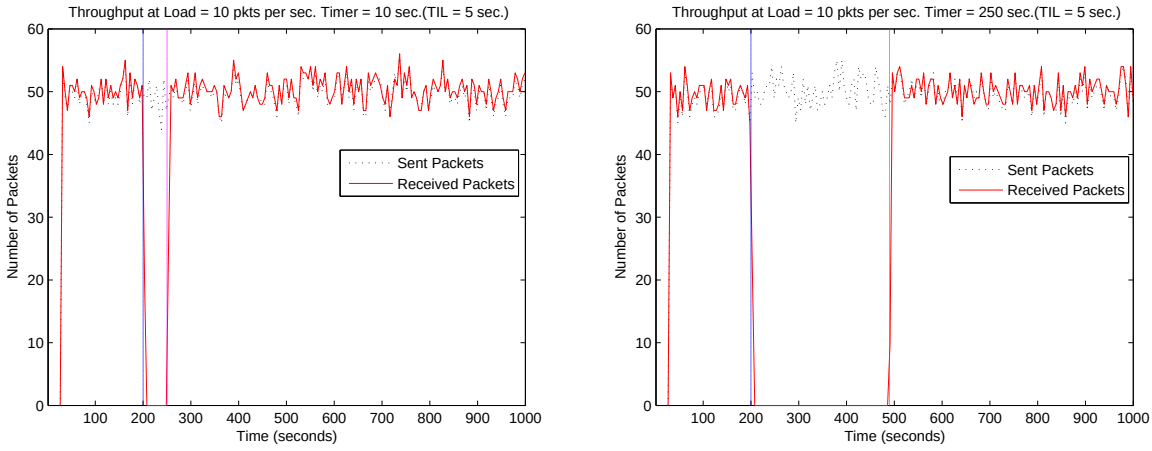
```
/* Cop Node Pseudo-code */

if overhearing a data packet
{
    counter(nexthop)++
    if sender != source
        {
            status(nexthop) = good
        }
    if counter(nexthop) > Threshold
    {
        timer starts
        if timer > maxTimer
        {
            if status(nexthop) != good
            {
                send alarm packet to source
                status(nexthop) = malicious
            }
        }
    }
}
```

Figure 3.4: Cop Pseudo code

packets in a 5-second time interval are plotted against the simulation time. The topology is similar to Example 4 in Chapter 4. In this case, only one active connection is established over an 8-node network carrying a load of 10 packets per second. The Cop travels at a speed of 10 meters per second and the maximum threshold is set to 100 packets. The hold-down timer is set to be 10 and 250 seconds. The experiment is done over a 1000-second simulation time and it is assumed that a malicious node and Cop are active after 200 seconds of simulation time.

It is shown that with a larger hold-down timer, the Cop takes a longer time to detect a malicious node. Before a malicious node is detected, packets will be dropped and, therefore, overall network throughput is decreased. The longer the detection time, the larger the reduction in network throughput. In other words, if the Cop detects a malicious node faster, the network throughput will be improved faster. However, a false alarm is possible when the parameters are set improperly.



(a) Hold-down Timer = 10 sec.

(b) Hold-down Timer = 250 sec.

Figure 3.5: The effect of Hold-down timer in Cop mechanism

3.6 EXPERIMENTAL DESIGN

This dissertation is divided into two major studies, an analytical study and a simulation study. For the analytical study, simple static ad hoc networks are considered for mathematical tractability in order to understand how the packet dropping attack impacts the throughput performance of the network and how detection mechanisms improve the throughput when a malicious node is detected. Moreover, the effect of detection parameters is studied.

In the simulation study, more complex scenarios, i.e., both static ad hoc networks, MANETs, and WMNs, were simulated to compare both detection mechanisms along with their variations. In addition, wireless channel effects are studied (although by parameterizing them) to show the performance impact on not only the attack performance but also the detection performance. The details of each study is explained below.

3.6.1 Analytical study

The objective of this study is to explain the impact of the packet dropping attack in static ad hoc networks and throughput improvements due to the detection mechanisms in term of throughput performance. The analysis is described and compared with simulations for validation. Examples are used to clarify analytical calculations. Each example has different number of nodes but similar node density. However, locations and the numbers of malicious nodes are different depending on the example scenario. The analysis shows the throughput in term of the detection time. Since the detection time depends on detection parameters, traffic loads are varied with different detection parameters in order to observe the throughput variations. Simulations are used to validate the analysis with 30 repetitions.

3.6.2 Simulation study

The main goal of this study is to compare the performance of detection mechanisms, namely, watchdog and cop mechanisms, in terms of throughput, routing overhead and detection efficiency ratio. The detection efficiency is a new performance metric and will be

explained in more detail in Chapter 5. Three different types of networks, static ad hoc networks, mobile ad hoc networks (MANETs), and wireless mesh networks (WMNs) are studied. The experimental design factors and values are shown in Table 3.1. However, the experimental design results in large sets of results and we show only interesting results in this dissertation. The number of repetitions is 30 simulation runs. The responses to the system are throughput, routing overhead, and detection efficiency ratio.

Table 3.1: Experimental Design Factors in Simulation Study

Experimental Design Factors	**Level 1**	**Level 2**
General - Number of nodes	16	49
General - Simulation area	$500 \times 500\ m^2$	$1000 \times 1000\ m^2$
MANETs - pause time	0 sec.	60 sec.
MANETs - speed	1 m/s	10 m/s
WMNs - Number of gateways	1	2
Wireless - Transmission range	250 m.	500 m.
Watchdog	100% detection	50% detection
Cop	1 Mobile cop	4 Static cops

4.0 THROUGHPUT ANALYSIS OF DETECTION MECHANISMS

4.1 INTRODUCTION

In this chapter, we present a simple analysis for understanding and studying how the packet delivery ratio (throughput) in ad hoc networks is impacted by packet dropping attacks and how the detection mechanism improves the packet delivery ratio in the network. To simplify the analysis, static ad hoc networks are considered. More complicated scenarios including mobility are studied in the next chapter using simulations.

4.2 DETECTION MECHANISMS

As explained in the previous chapters, the watchdog mechanism is the one that is primarily used in the research literature for detecting packet dropping attacks. We propose the Cop mechanism to alleviate the burden on all nodes to perform the packet dropping attack detection function as well as to reduce the number of trust relations that have to be established in the network. In this chapter, we consider the analysis of these two detection mechanisms. For completeness, a brief summary of the two mechanisms is provided below.

4.2.1 Watchdog

The watchdog approach uses threshold-based detection to detect malicious nodes. Henceforth such nodes are assumed to simply drop packets after participating in routing functions. Moreover, unless otherwise mentioned they drop all data packets. The source node and the

forwarding node keep monitoring their next hop node's activity (unless otherwise mentioned, all the time). When a packet is forwarded to the next node, the monitoring node will mark the packet as "forwarded" in its buffer. In addition, it keeps increasing a non-forwarded packet counter if a packet is not forwarded. If the counter is over a maximum threshold and a suspicious node still does not forward any data packets, an alarm message is sent to notify the source, which will record the suspicious node as a malicious node and eliminate it in the routecache. The source will drop all other routing packets related to the malicious node and find a new route without the malicious node in the path.

4.2.2 Cop

As described in Chapters 1 and 3, we propose a new approach to help protect ad hoc networks from packet dropping attacks by introducing a special node, called *Cop*. The idea behind this paradigm is that one or more *Cops* are the only nodes which perform monitoring and detection functions to protect the network from the attack. *Cop* nodes are assumed to be trustworthy and detect the attack by either moving within the network and monitoring malicious activity or by static placement in the network so as to cover the area that needs monitoring. The *Cop* detection mechanism is similar to the watchdog mechanism, which uses threshold-based detection but only *Cops* keep monitoring the wireless channel for their neighbors' activities. Other nodes in the network perform routing and forwarding functions but not the detection function. By introducing such *Cop*s, it is possible that the energy in the rest of the network nodes can be prolonged so that the network's objectives are met better. However, the throughput performance with time is equal to or less than the throughput performance over time using the watchdog mechanism since *Cop* nodes may not be within every node's transmission range all the time in order to detect malicious activity.

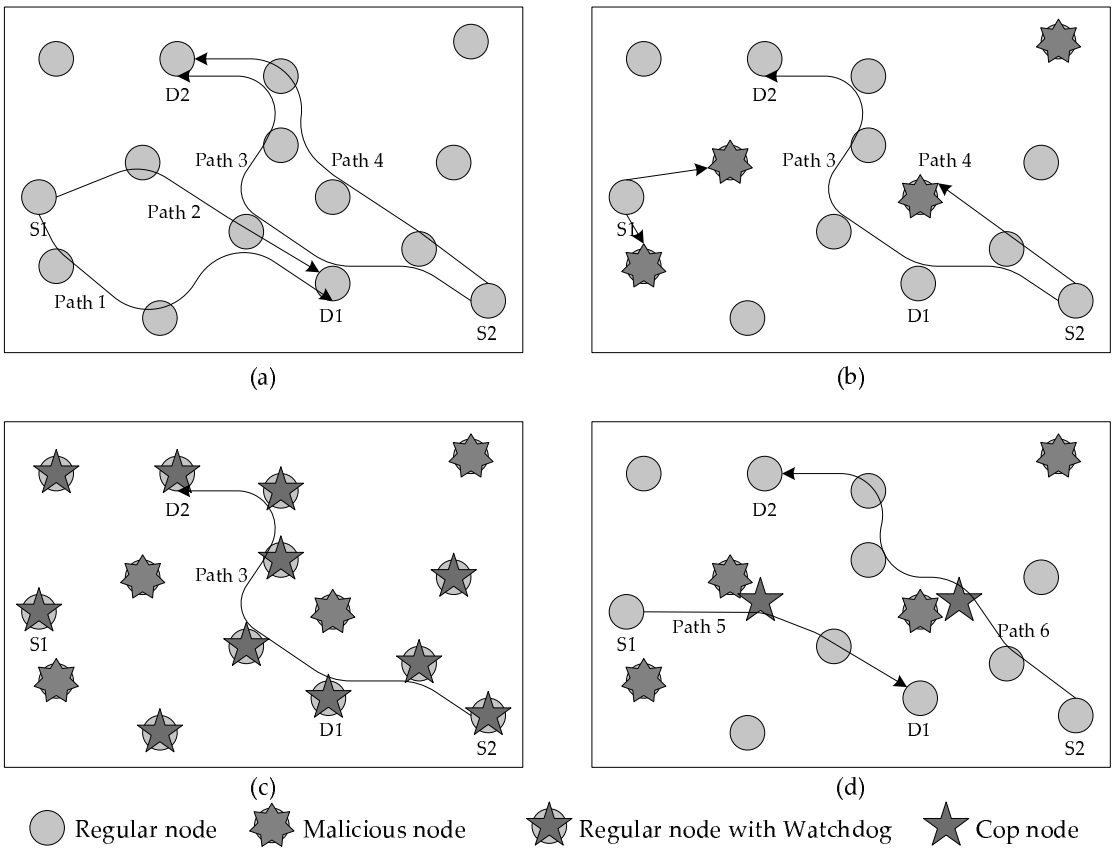


Figure 4.1: Example scenario

4.3 THROUGHPUT ANALYSIS IN A STATIC AD HOC NETWORK

4.3.1 Overview

In this study, the focus is on the packet dropping attack and the corresponding detection mechanisms mentioned above. The assumption here is that malicious nodes participate in routing information exchange or otherwise act so as for it to be possible that they are selected as intermediate nodes along routes from source nodes to destination nodes. However, they will drop all data packets that they are supposed to forward. The watchdog and cop mechanisms are studied in this chapter to mitigate the attack using packet delivery ratio (throughput) as the performance metric.

To illustrate the general effect of the packet dropping attack and its mitigation, consider Figure 4.1 that shows an example of a 15-node static ad hoc network with 2 sources and 2 destinations. Figure 4.1(a) illustrates a normal ad hoc network. The connection 1 (S1-D1) has two paths, i.e., Path 1 and Path 2 and the connection 2 (S2-D2) also has two paths, i.e., Path 3 and Path 4. The connections are not interrupted since all nodes are regular nodes and packets are delivered at the destinations. No packet is dropped intentionally.

Figure 4.1(b) shows a network with four malicious nodes, which drop all data packets. In the case of connection 1, since Path 1 and Path 2 have to pass through malicious nodes, packets from the source S1 cannot reach the destination D1 and this causes the throughput to fall to zero. In the case of connection 2, if the source S2 chooses Path 3, all packets will reach the destination D2 but if the source chooses Path 4, all packets will be dropped. We assume that the selection of a path to the destination is equally likely if everything else is equal (e.g., the number of hops). So on average, the throughput of connection 2 will be 50%. So far we have not assumed any detection mechanism.

When a watchdog mechanism is implemented (where nodes check to see if their packets are being forwarded), a malicious node is detected, and this information gets to source S1, there is no other path that can reach the destination D1 and the performance cannot be improved for connection 1 as shown in Figure 4.1(c). In the case of connection 2, if Path 4 is chosen, when the malicious node is detected, S2 can employ the alternative Path 3. All

subsequent data packets will follow Path 3 to the destination D2.

The cop mechanism is shown in Figure 4.1(d). Two static cop nodes are added and assigned to detect the malicious node and they also help in forwarding packets. Therefore, after malicious nodes are detected, sources have two new paths, Path 5 and Path 6, to destinations and all packets can reach the destinations. Figures 4.1(c) and (d) show how the performance is improved when detection mechanisms are implemented. The important question is how much improvement is possible after a malicious node is detected.

4.3.2 General Analysis

This chapter presents a simple analysis of Packet Delivery Ratio (PDR) in ad hoc networks with and without malicious nodes and with different detection mechanisms. We define PDR as follows.

$$PDR = \frac{Total\ number\ of\ received\ packets\ at\ destination}{Total\ number\ of\ sent\ packets\ by\ source} \tag{4.1}$$

In order to simplify our analysis, we make some assumptions:

- One source and one destination are considered in a grid network (single flow).
- Malicious nodes perform a 100% data packet dropping attack.
- The probability of choosing any path is equally likely for all paths with identical routing metrics from a source to a destination.
- The chosen route is always the shortest path from the source to the destination.
- There is no more than 1 malicious node in any path.*

Note: * - This assumption is relaxed in a future example with some modifications.

Implicit in the assumptions is that a malicious node does not modify routing packets to have higher chances to be chosen as an intermediate node than other regular nodes. However, the analysis can be applicable to such scenarios as well, except that the classification of paths and probabilities of selecting them have to be modified to accommodate the fact that a route which includes malicious nodes is more likely initially. Similarly, the analysis presented here can be enhanced to include the impact of dropping only a fraction of data packets or other

detection mechanisms. We do not consider such enhancements in this dissertation and suggest them as part of future work.

Since the packet dropping attack is studied, the same assumptions as described previously hold. A malicious node participates in routing information exchange but drops all data packets. Therefore, a destination will not receive any data packets if a malicious node is an intermediate node along a selected path and the PDR is zero since the total number of received packets is zero.

When a detection mechanism is implemented (e.g., using a threshold based detection), a malicious node can be detected and excluded from the network (we assume the Watchdog mechanism for the time being). A new route will be created and all subsequent data packets will follow the new path. However, more than one malicious node can be in the network and a second malicious node can also be present in the newly set-up path. Detecting nodes need to keep monitoring their neighbors and detect other malicious nodes and exclude them from the network. Once all malicious nodes are detected, data packets can be received at the destination. The performance depends on the probability of choosing a path to the destination. When a malicious node is selected as an intermediate node, the PDR is zero (as described above). In contrast, when a path contains only regular nodes, all packets will be received at a destination and the PDR is one since total number of received packets equals to total number of sent packets (note that we assume that there is no congestion or other packet losses here).

The total number of paths with a given routing metric from a source to a destination has to be known in order to calculate the probability of choosing a path that contains a malicious node or one that has only regular nodes. When a malicious node is detected and excluded from the network, the number of paths is changed and a new probability has to be recalculated. This process is recursively continued until the last malicious node is detected. A probability tree is used to analyze this problem as shown in Figure 4.2. There are n stages to be considered for n malicious nodes as described next.

The average PDR can be computed using a probability weighted average of the number of received packets at a destination for all possible cases. In our analysis, a constant bit rate traffic is assumed and packets start sending at time 0. Therefore, the number of sent

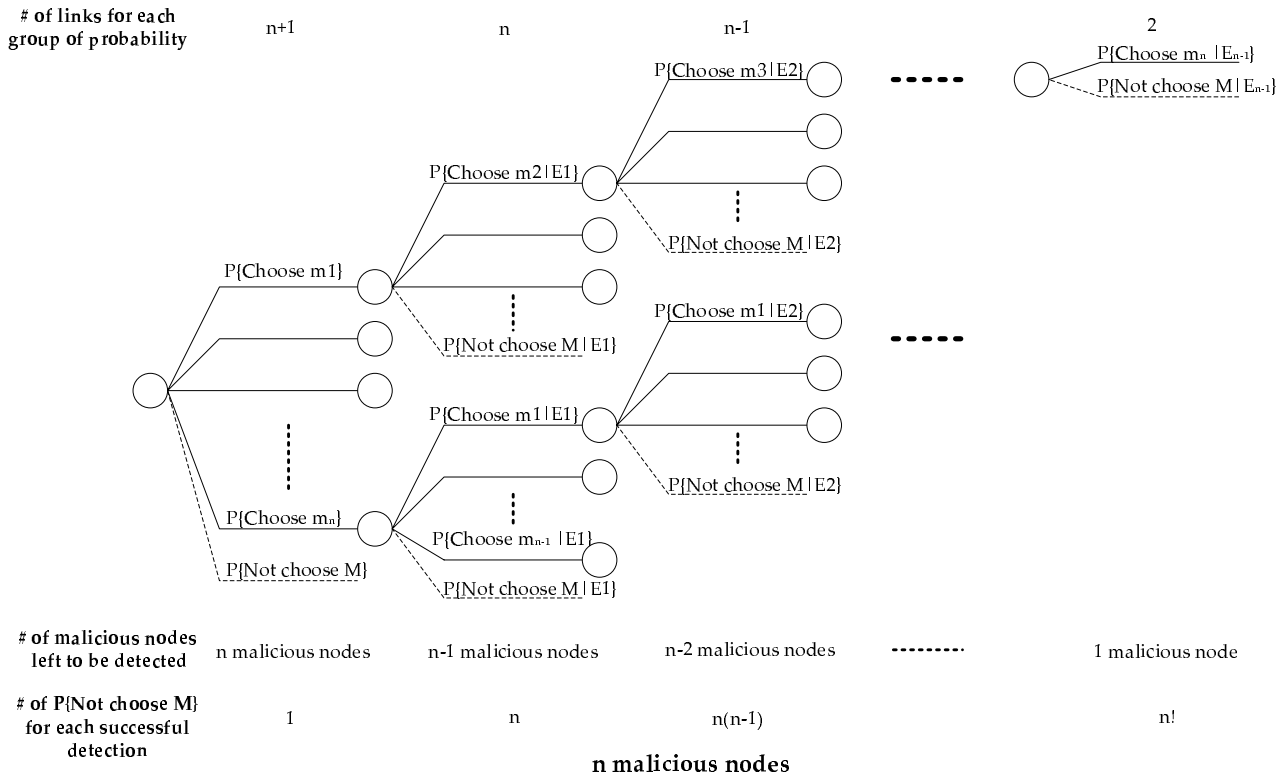


Figure 4.2: Probability Tree for N malicious nodes

packets is constant for the observation duration. To analyze the average PDR performance, parameters are defined as follows.

Let:

N = Set of non-detected malicious nodes

M = Set of all malicious nodes

n = Total number of malicious nodes

m_i = A malicious node i

E_j = Event that j malicious node(s) is(are) already detected

T = Total observation time in seconds

R = Data rate in packets per second

T_{det_i} = Detection time when the i^{th} malicious node is detected

With watchdog mechanism, the average PDR is shown in Equation 4.2:

$$PDR_{avg} = P\{choose\,path\,excluding\,nodes\,in\,M\} \times PDR(choose\,path\,excluding\,nodes\,in\,M) + \sum_{i=1}^{n} P\{choose\,path\,including\,node\,m_i\} \times PDR(choose\,path\,including\,node\,m_i) \quad (4.2)$$

where, $PDR(choose\,path\,including\,node\,m_i)$ is the PDR of choosing the node m_i in the path but there are other paths, that can be chosen after m_i is detected at the time T_{det_i}. In what follows, we use a shorter notation. $P\{choose\,path\,excluding\,nodes\,in\,M\}$ is written as $P\{not\,choose\,M\}$, $P\{choose\,path\,including\,node\,m_i\}$ is written as $P\{choose\,m_i\}$ and so on. Moreover,

$$PDR(choose\,m_i) = P\{not\,choose\,M|E_1\} \times PDR(not\,choose\,M|E_1) + \sum_{i \in N} P\{choose\,m_i|E_1\} \times PDR(choose\,m_i|E_1)$$

After a malicious node is detected, that node is excluded from all the paths from a source to a destination. Therefore, $P\{choose\,m_i|E_k\}$ is the probability of choosing a path containing m_i which excludes the paths containing the first k detected malicious nodes.

The equation is recursively continued until the last malicious node is detected as follows:

$$
\begin{aligned}
PDR(choose\, m_i|E_1) &= P\{not\, choose\, M|E_2\} \times PDR(not\, choose\, M|E_2) \\
&\quad + \sum_{i \in N} P\{choose\, m_i|E_2\} \times PDR(choose\, m_i|E_2) \\
&\vdots \\
PDR(choose\, m_i|E_{n-2}) &= P\{not\, choose\, M|E_{n-1}\} \times PDR(not\, choose\, M|E_{n-1}) \\
&\quad + \sum_{i \in N} P\{choose\, m_i|E_{n-1}\} \times PDR(choose\, m_i|E_{n-1}) \\
PDR(choose\, m_i|E_{n-1}) &= \left(\frac{T - T_{det_n}}{T}\right)
\end{aligned}
$$

Here T_{det_n} is the time at which all n malicious nodes have been detected and excluded from the network. The final result in the last equation arises due to the fact that once the last malicious node has been detected, all of the packets reach the destination. This assumes that there are paths that exist to the destination after all malicious nodes have been detected, there is no congestion in the network, nor are there other reasons for packets being dropped (e.g., channel errors or buffer overflows). It is also important to note that $P\{choose\, m_i|E_{n-1}\}$ is the probability of choosing m_i but excluding the paths containing the previous $(n-1)$ detected malicious nodes. The total number of sent packets is RT and the total number of received packets, after the last malicious node is detected, is $R(T - T_{det_n})$. Hence the PDR is the ratio of these two quantities. This is a simple analysis that assumes that malicious nodes are detected sequentially, one by one, and multiple malicious nodes are not present along the same path. The analysis becomes more complicated otherwise.

When n malicious nodes are in the network, the total number of possible cases to be considered is:

$$
Total\, cases = n! + \sum_{i=0}^{n-1} \frac{n!}{(n-i)!} \tag{4.3}
$$

The intuition behind this equation is as follows. The PDR_{avg} is recursively computed when a malicious node is detected until all malicious nodes are detected. This implies $n!$ because we assume that any one of the n malicious nodes may be detected first. While this is

not really true because the malicious node that is detected first depends on the path selected and detection mechanism, our assumptions limiting traffic to one flow and one malicious node per path allows this approximation. Every time a malicious node is detected, one case, where no malicious node is chosen, does not have more leaves in the tree and this implies the last part of the equation as shown in Figure 4.2. For example in a static network with 3 malicious nodes, the total number of cases is 16, which comes from 2 parts: the probability of choosing any one of the malicious nodes and the probability of not choosing a malicious node. For the first part, there are 3 stages for this example and each stage depends on the number of malicious node left to be detected. The first stage has 3 malicious nodes to be detected and the second stage has 2 malicious nodes to be detected and the last stage has 1 malicious node to be detected. The total number of cases for choosing any malicious nodes is 3×2×1 for all 3 stages. The second part is the cases for not choosing any malicious nodes. The first stage has 1 possible case, the second stage has 3 possible cases and the last stage has 6 possible cases. Total number of cases of not choosing any malicious nodes is 1+3+6. Therefore, the total number of cases for this example is 16 cases.

Since the cop mechanism uses an algorithm similar to that of the watchdog mechanism, the differences between the two mechanisms arises due to the fact that the only detecting nodes are cop nodes. The cop node has to be in the transmission range of a forwarding node and the intended receiver to correctly detect a malicious node. If a cop node is moving in a network (mobile cop), it has a limited time to monitor each node's activity. If it is static, called a static cop, its performance is similar to the watchdog mechanism, but it may have limited coverage. Therefore, the cop mechanism usually takes equal or longer detection time than the watchdog mechanism. The PDR calculation is similar to the watchdog mechanism but the difference is in the detection time.

4.3.3 Detection time calculation

The analysis shows that PDR depends on the detection time, which is the time when a malicious node is detected. Since we assume a threshold-based detection, the detection time can be computed as follows.

Let:

TH = Threshold

TO = Time-out

R = Data rate

The detection time for the watchdog mechanism is calculated as follows:

$$T_{det} = T_{start} + \frac{TH}{R} + TO \tag{4.4}$$

where,

T_{det} = Detection time

T_{start} = Time when a node is activated from the beginning to the time when a node starts detecting a malicious node

The detection time calculation for mobile cop mechanism is different from that for the watchdog mechanism in that a mobile cop node may not overhear the relevant communications all the time when it is mobile. It intermittently detects a malicious node when it is in the range of both a sender and a forwarder in a static ad hoc network. The detection time for mobile cop mechanism is as follows:

$$T_{det} = \begin{cases} T_{start} + \frac{TH}{R} + TO & if \quad T_{DD} \geq \frac{TH}{R} + TO \\ T_{start} + \frac{TH}{R} + TO + kT_{IDD} & if \quad T_{DD} < \frac{TH}{R} + TO \end{cases} \tag{4.5}$$

Here T_{DD} = Detection duration when a mobile cop node is in the range of a sender and a forwarder to detect a malicious node

k = number of rounds that a mobile cop node needs to traverse to detect a malicious node

T_{IDD} = Inter-detection duration when a mobile cop node waits to be in a range of a sender and a forwarder to detect a malicious node

where, k is calculated as follows:

$$k = \left\lfloor \frac{\frac{TH}{R} + TO}{T_{DD}} \right\rfloor \tag{4.6}$$

and T_{IDD} is calculated as follows:

$$T_{IDD} = \frac{Distance}{Cop\,speed} \tag{4.7}$$

Here "Distance" is the total distance where a mobile cop is not in the range of both the sender and forwarder to detect a malicious node. In this study, T_{start} is 0 since a source node starts sending packets at time 0. It is important to note that the detection time calculation assumes knowledge of the location of a malicious node such that the detection duration time and inter-detection duration time are known. This is for analysis purposes only. Since T_{IDD} is not a parameter for the watchdog mechanism, it confirms that the mobile cop mechanism results in an equal or longer detection time compared to the watchdog mechanism. However, T_{IDD} is used when the watchdog mechanism does not perform monitoring 100% of the time for detection and it will be discussed in the next chapter.

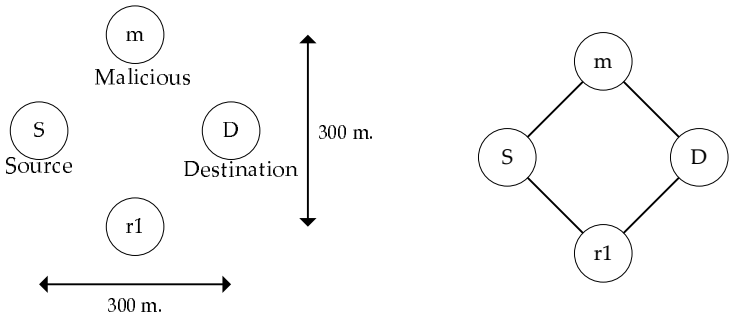
There is a special case when a source does not start sending packets after a mobile cop is already in T_{DD} period. The detection duration for the first round is not a full detection duration. If T_{DD} is greater than $\frac{TH}{R} + TO$, a mobile cop may need an extra round to come back and detect a malicious node. This adds additional detection time to the mobile cop mechanism. In practice, when T_{DD} is equal to $\frac{TH}{R} + TO$, a mobile cop cannot perfectly detect a malicious node within 1 round, it also needs another round to come back and detect the malicious node as well.

4.4 EXAMPLES

In this section, four examples are presented to explain the calculation of PDR described in the previous section with concrete network topologies and malicious nodes when packet dropping attack occurs along with the two detection mechanisms. The first 3 examples are devoted to analyzing the watchdog mechanism and the last example is for both watchdog and cop mechanisms. These examples compare analytical results with simulations.

4.4.1 Example 1: 4-Node Network - Watchdog

A 4-node network is considered here as shown in Figure 4.3. All nodes are static. A source sends packets to a destination through one of two paths. A malicious node ("m") drops all data packets that pass through it. Node "r" is a regular honest node and fully functional to forward data packets.



(a) 4-node network set-up (b) 4-node network topology

Figure 4.3: 4 Node network scenario

From the scenario presented, there are 2 paths between the source and destination, S-r1-D and S-m-D. As per the assumptions, each path has an equal probability to be chosen by the source. Therefore, $P\{choose\, m\} = P\{not\, choose\, m\} = \frac{1}{2}$. Data rate is "R" packets per second and the simulation time is "T" seconds. The total number of sent packets is "RT". There is no congestion and no channel errors are considered in this case. Therefore, the total number of received packets is "RT" when the source chooses the path S-r1-D.

$$PDR(not\, choose\, m) = \frac{RT}{RT} = 1$$

When the source chooses the path S-m-D, there will be no received packets at the destination. Therefore,

$$PDR(choose\, m) = \frac{0}{RT} = 0$$

One of the paths gives a 100% PDR, but, on the other hand, the other path gives 0% PDR.

In order to calculate the average PDR, Equation 4.2 is simply adapted to this scenario as follows:

$$\begin{aligned} PDR_{avg} &= P\{choose\, m\} \times PDR(choose\, m) \\ &\quad + P\{not\, choose\, m\} \times PDR(not\, choose\, m) \end{aligned} \tag{4.8}$$

There are 2 cases to be considered here. The first case is where only a malicious node is in the network without the watchdog mechanism. The second case is where a malicious node is in the network and the watchdog mechanism is implemented.

Case 1: Network with a malicious node but no watchdog implementation

We consider the average PDR in the system when a malicious node is in the network.

$$\begin{aligned} PDR_{avg} &= P\{choose\, m\} \times PDR(choose\, m) \\ &\quad + P\{not\, choose\, m\} \times PDR(not\, choose\, m) \\ &= \frac{1}{2} \times 0 + \frac{1}{2} \times 1 \\ &= \frac{1}{2} \end{aligned}$$

Case 2: Network with a malicious node and watchdog implementation

In this case, the watchdog mechanism is implemented in all nodes, except the malicious node. When a malicious node is an intermediate node and is detected, the total number of received packets is $R(T - T_{det})$.

$$\begin{aligned} PDR_{avg} &= P\{choose\, m\} \times PDR(choose\, m) \\ &\quad + P\{not\, choose\, m\} \times PDR(not\, choose\, m) \\ &= \frac{1}{2} \times \frac{R(T - T_{det})}{RT} + \frac{1}{2} \times 1 \\ &= \frac{1}{2} \times \frac{(T - T_{det})}{T} + \frac{1}{2} \\ &= \frac{2T - T_{det}}{2T} \\ &= 1 - \left(\frac{T_{det}}{2T}\right) \end{aligned} \tag{4.9}$$

From Equation 4.9, it is important to note that PDR_{avg} relies on the detection time, as described in the previous section. However, the detection time depends on both the threshold setting and the traffic rate. When the sending rate is high, the detection time is faster than when the sending rate is low with the same threshold setting.

4.4.1.1 Detection parameter analysis Based on the previous section, the detection time can be calculated from the load in the network with the specified threshold and time out settings. Figure 4.4 plots the detection time against load where one malicious node is in the network as shown in Example 1. When the load is increased, the detection time is decreased. In addition, a higher threshold and time out settings result in a longer detection time.

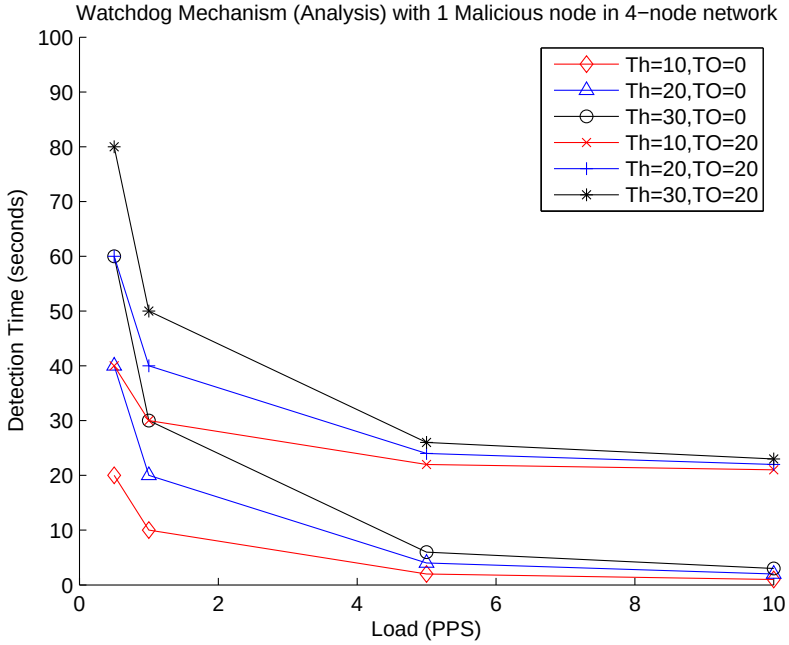


Figure 4.4: Detection time VS. Load for Watchdog mechanism

From Example 1, the throughput is plotted in Figure 4.5. The threshold and time-out settings are varied in order to demonstrate the effect of parameter settings in terms of PDR performance. The results show that when the load is increased with the same detection threshold, the detection time is decreased. If a time-out is added to the mechanism, the

detection time is increased by the time-out value that is additional to the detection time when time-out is zero seconds.

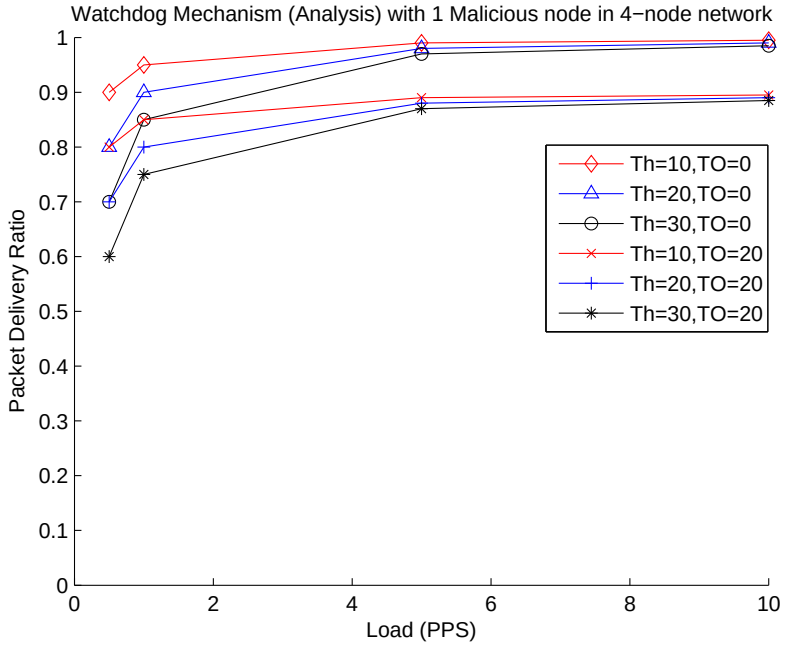


Figure 4.5: PDR VS. Load for Watchdog mechanism

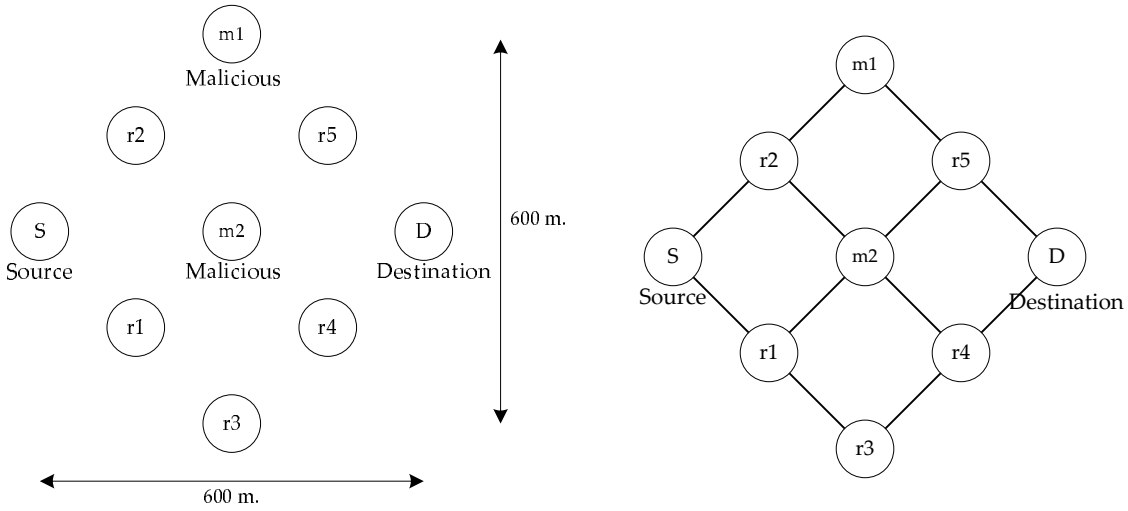
In summary, Figure 4.5 shows that PDR depends on the detection time, which also depends on the threshold, time-out, and load in the network.

The parameter analysis can be applied to other networks as well. The equations as shown earlier can be used to calculate the detection time and PDR. However, the analysis has to be completed such that all the parameters can be fully analyzed and the parameter analysis can be plotted by varying the parameters.

4.4.2 Example 2: 9-node network - Watchdog

In this example, a larger network is considered. One source S, one destination D, and two malicious nodes, m_1 and m_2, are considered in this scenario. The network setup and network topology are shown in Figure 4.6.

From the network scenario, there are 6 shortest paths between the source and destination as follows.



(a) 9-node network set-up (b) 9-node network topology

Figure 4.6: 9 Node network scenario

$$
\begin{aligned}
Path1 &= S - r2 - m_1 - r5 - D \\
Path2 &= S - r2 - m_2 - r5 - D \\
Path3 &= S - r2 - m_2 - r4 - D \\
Path4 &= S - r1 - m_2 - r5 - D \\
Path5 &= S - r1 - m_2 - r4 - D \\
Path6 &= S - r1 - r3 - r5 - D
\end{aligned}
$$

The probability of choosing m_1 in the path (Path1) is $\frac{1}{6}$. However, m_2 is in the middle of the network topology and most of the paths use it as an intermediate node. The probability of choosing m2 (Path2 to Path5) is $\frac{4}{6}$. Lastly, the probability of not choosing any path containing a malicious node (Path6) is $\frac{1}{6}$. Like case 1 in the previous study, when no detection mechanism is implemented and two malicious nodes exist in the network, PDR_{avg} is only $\frac{1}{6}$. When the watchdog mechanism is implemented, the PDR_{avg} can be computed as follows:

$$
\begin{aligned}
PDR_{avg} &= P\{choose\, m_1\} \times PDR(choose\, m_1) \\
&\quad + P\{choose\, m_2\} \times PDR(choose\, m_2) \\
&\quad + P\{not\, choose\, m_1 \& m_2\} \times PDR(not\, choose\, m_1 \& m_2)
\end{aligned}
$$

Since one malicious node is detected but the other malicious node is still in the network, a new path that is selected by the source may have the other malicious node in it. Therefore, the PDR{choose m_1} has also two possible cases as shown in the following equation:

$$
\begin{aligned}
PDR(choose\, m_1) &= P\{choose\, m_2 | E_1\} \times PDR(choose\, m_2 | E_1) \\
&\quad + P\{not\, choose\, m_2 | E_1\} \times PDR(not\, choose\, m_2 | E_1)
\end{aligned}
$$

The probability values in this scenario are:

$$
\begin{aligned}
P\{choose\, m_1\} &= \frac{1}{6} \\
P\{choose\, m_2\} &= \frac{4}{6} \\
P\{not\, choose\, m_1 \& m_2\} &= \frac{1}{6} \\
P\{choose\, m_2 | E_1\} &= \frac{4}{5} \\
P\{not\, choose\, m_2 | E_1\} &= \frac{1}{5}
\end{aligned}
$$

Figure 4.7 shows the complete probability tree for this example.

In order to simplify the analysis, Equation, T_{det_1} and T_{det_2} are used as the detection times when the first and second malicious nodes are detected respectively. The final equation for this analysis is:

$$
\begin{aligned}
PDR_{avg} &= \frac{1}{6} \times \left(\frac{4}{5} \times \frac{T - T_{det_2}}{T} + \frac{1}{5} \times \frac{T - T_{det_1}}{T}\right) \\
&\quad + \frac{4}{6} \times \left(\frac{1}{2} \times \frac{T - T_{det_2}}{T} + \frac{1}{2} \times \frac{T - T_{det_1}}{T}\right) + \frac{1}{6} \\
&= 1 - \left(\frac{11 T_{det_1} - 14 T_{det_2}}{30 T}\right)
\end{aligned} \tag{4.10}
$$

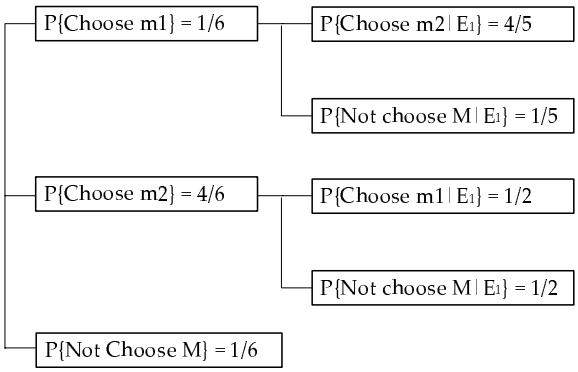


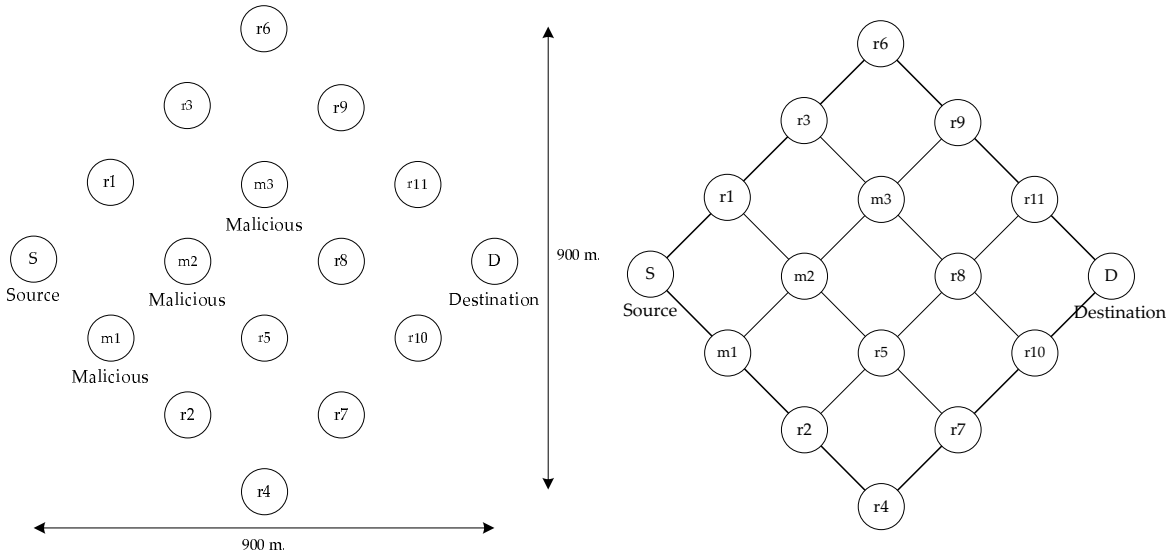
Figure 4.7: Probability Tree for 9 node network scenario

4.4.3 Example 3: 16-node network (joint paths) - Watchdog

In this scenario, we relax one of the assumptions, which is that there is no more than 1 malicious node in a path. We investigate a 16-node network with 3 malicious nodes, which share several paths. The network topology is shown in Figure 4.8.

The Equation 4.2 does hold with this scenario but it needs to be modified when we do the probability calculation. This is a special scenario where the probability calculation is specific to this scenario. From the scenario, multiple malicious nodes can be in the same path between the source and the destination. The probability of choosing paths has to be calculated when a malicious node is detected and the first malicious node in the path is usually detected prior to other malicious nodes. Therefore, the probability is modified to agree with the equation. In this scenario, the total number of shortest paths from the source to the destination is 20 as follows.

$$
\begin{aligned}
Path1 &= S - r1 - r3 - r6 - r9 - r11 - D \\
Path2 &= S - r1 - r3 - m_3 - r8 - r10 - D \\
Path3 &= S - r1 - r3 - m_3 - r9 - r11 - D \\
Path4 &= S - r1 - r3 - m_3 - r8 - r11 - D
\end{aligned}
$$



(a) 16-node network set-up (b) 16-node network topology

Figure 4.8: 16 Node network scenario

$$
\begin{aligned}
Path5 &= S - r1 - m2 - m_3 - r9 - r11 - D \\
Path6 &= S - r1 - m_2 - m_3 - r8 - r11 - D \\
Path7 &= S - r1 - m_2 - m_3 - r8 - r10 - D \\
Path8 &= S - r1 - m_2 - r5 - r8 - r11 - D \\
Path9 &= S - r1 - m_2 - r5 - r8 - r10 - D \\
Path10 &= S - r1 - m_2 - r5 - r7 - r10 - D \\
Path11 &= S - m_1 - m_2 - m_3 - r9 - r11 - D \\
Path12 &= S - m_1 - m_2 - m_3 - r8 - r11 - D \\
Path13 &= S - m_1 - m_2 - m_3 - r8 - r10 - D \\
Path14 &= S - m_1 - m_2 - r5 - r8 - r11 - D \\
Path15 &= S - m_1 - m_2 - r5 - r8 - r10 - D \\
Path16 &= S - m_1 - m_2 - r5 - r7 - r10 - D \\
Path17 &= S - m_1 - r2 - r5 - r8 - r11 - D \\
Path18 &= S - m_1 - r2 - r5 - r8 - r10 - D
\end{aligned}
$$

$$
\begin{aligned}
Path19 &= S - m_1 - r2 - r5 - r7 - r10 - D \\
Path20 &= S - m_1 - r2 - r4 - r7 - r10 - D
\end{aligned}
$$

The probabilities for all cases are shown as follows:

$$
\begin{aligned}
P\{choose\,only\,(m_1)\} &= \frac{4}{20} \\
P\{choose\,only\,(m_2)\} &= \frac{3}{20} \\
P\{choose\,only\,(m_3)\} &= \frac{3}{20} \\
P\{choose\,only\,(m_1\&m_2)\} &= \frac{3}{20} \\
P\{choose\,only\,(m_1\&m_3)\} &= \frac{0}{20} \\
P\{choose\,only\,(m_2\&m_3)\} &= \frac{3}{20} \\
P\{choose\,only\,(m_1\&m_2\&m_3)\} &= \frac{3}{20} \\
P\{not\,choose\,M\} &= \frac{1}{20}
\end{aligned}
$$

From this scenario, when paths with only m_1 and m_2, only m_1 and m_3, or only m_1, m_2 and m_3 are considered, the malicious node m_1 will be detected first and we can add all the probabilities to determine $P\{choose\,m_1\}$. This analysis can repeatedly be used with other probability calculations. Therefore, the initial probability is:

$$
\begin{aligned}
P\{choose\,m_1\} &= \frac{10}{20} \\
P\{choose\,m_2\} &= \frac{6}{20} \\
P\{choose\,m_3\} &= \frac{3}{20} \\
P\{not\,choose\,M\} &= \frac{1}{20}
\end{aligned}
$$

Figure 4.9 shows the probability tree for this scenario, including all the probability values. Therefore, the P_{avg} for this example is:

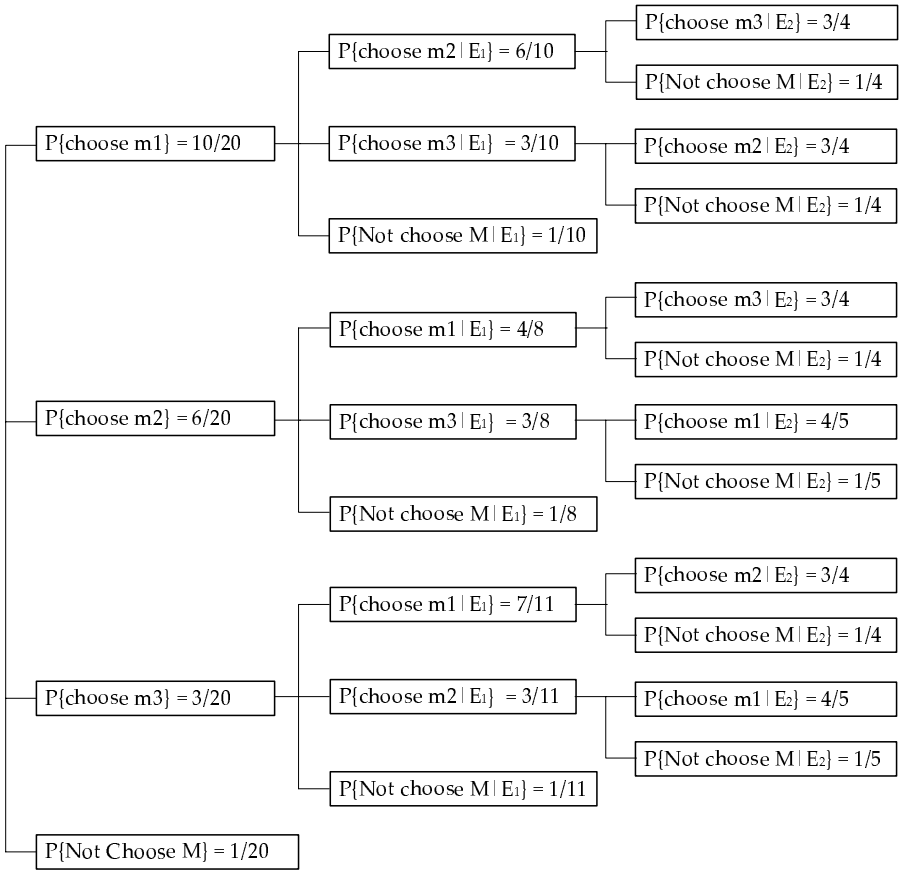


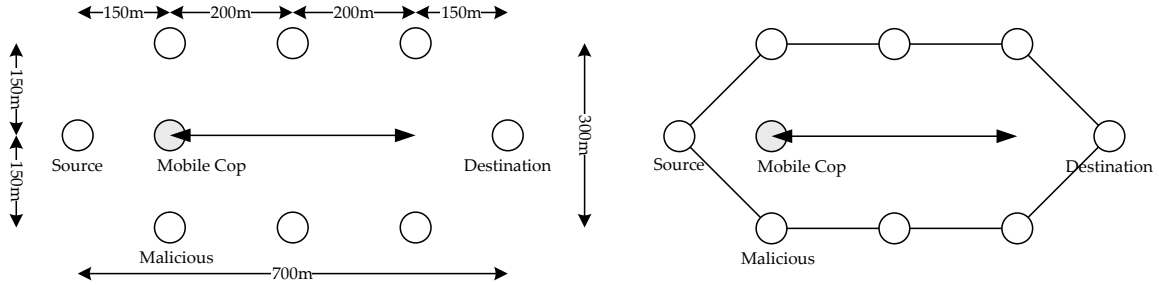
Figure 4.9: Probability tree for 16 node network scenario

$$
\begin{aligned}
PDR_{avg} &= P\{choose\, m_1\} \times PDR(choose\, m_1) \\
&\quad + P\{choose\, m_2\} \times PDR(choose\, m_2) \\
&\quad + P\{choose\, m_3\} \times PDR(choose\, m_3) \\
&\quad + P\{not\, choose\, M\} \times PDR(not\, choose\, M) \\
&= \frac{10}{20} \times \left[\frac{6}{10} \times \left(\frac{3}{4} \times \frac{T - T_{det_3}}{T} + \frac{1}{4} \times \frac{T - T_{det_2}}{T} \right) \right. \\
&\quad + \frac{3}{10} \times \left(\frac{3}{4} \times \frac{T - T_{det_3}}{T} + \frac{1}{4} \times \frac{T - T_{det_2}}{T} \right) \\
&\quad \left. + \frac{1}{10} \times \left(\frac{T - T_{det_1}}{T} \right) \right] \\
&\quad + \frac{6}{20} \times \left[\frac{4}{8} \times \left(\frac{3}{4} \times \frac{T - T_{det_3}}{T} + \frac{1}{4} \times \frac{T - T_{det_2}}{T} \right) \right. \\
&\quad + \frac{3}{8} \times \left(\frac{4}{5} \times \frac{T - T_{det_3}}{T} + \frac{1}{5} \times \frac{T - T_{det_2}}{T} \right) \\
&\quad \left. + \frac{1}{8} \times \left(\frac{T - T_{det_1}}{T} \right) \right] \\
&\quad \frac{3}{20} \times \left[\frac{7}{11} \times \left(\frac{3}{4} \times \frac{T - T_{det_3}}{T} + \frac{1}{4} \times \frac{T - T_{det_2}}{T} \right) \right. \\
&\quad + \frac{3}{11} \left(\frac{4}{5} \times \frac{T - T_{det_3}}{T} + \frac{1}{5} \times \frac{T - T_{det_2}}{T} \right) \\
&\quad \left. + \frac{1}{11} \times \left(\frac{T - T_{det_1}}{T} \right) \right] + \frac{1}{20} \\
&= 1 - \left(\frac{89 T_{det_1} - 180 T_{det_2} - 567 T_{det_3}}{880 T} \right) \qquad (4.11)
\end{aligned}
$$

4.4.4 Example 4: 8+1-node Network - Watchdog and Cop

In order to study the throughput performance with the Cop mechanism, we analyze a scenario as shown in Figure 4.10. In this case, traffic is generated from a source to a destination. The mobile Cop travels at 10 m/s speed along a straight line between the source and destination nodes. Note that the total number of nodes for the watchdog mechanism is 8 nodes and that for the cop mechanism is 9 nodes (the cop node is extra). In this scenario, the transmission range for each node is 250 m. and a malicious node is next to the source. The cop node can detect a malicious node when a source chooses the path that includes

the malicious node. Therefore, the cop node can perform the detection within 100 m from the source, as shown in Figure 4.11. Therefore, the detection duration is 10 seconds. If a threshold is set that is higher than the number of packets sent during the detection duration, the cop has to wait and come back to continue the detection on the next round. The inter-detection duration is 60 seconds. This could cause a delay in detection time. However, if the threshold is set too short, false positives can occur and this causes the PDR to be reduced.



(a) 8+1 node network set-up (b) 8+1 node network topology

Figure 4.10: 8+1 node network scenario with 1 Mobile Cop

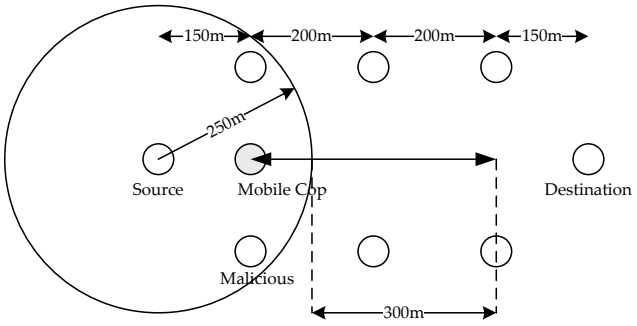


Figure 4.11: Mobile Cop mechanism analysis - 8+1 node network

With the watchdog mechanism, the analysis is similar to the 4-node network in Example 1, explained earlier because there are only 2 paths from the source to destination and 1 malicious node. Therefore, the PDR_{avg} is as follows:

$$PDR_{avg} = P\{choose\, m\} \times PDR(choose\, m)$$

$$
\begin{aligned}
& + P\{not\, choose\, m\} \times PDR(not\, choose\, m) \\
= \; & 1 - \left(\frac{T_{det}}{2T}\right)
\end{aligned}
$$

The Cop mechanism also used the same equation but the detection time is different as explained previously. With the Cop mechanism, the average detection time between the first and second rounds of detection is used in the equation above to get the average PDR in the network.

When the cop speed is changed, the detection time can be either faster or slower depending on the detection duration, which also depends on threshold, time-out setting and the sending rate. The equations 4.6 and 4.7 show the relationship between these parameters.

4.5 SIMULATION

To further investigate the validity of our analysis, the example networks are simulated (with identical network topologies as the networks used for analysis). The tool for the simulation is Network Simulator 2 or ns-2 version 2.30.

To simulate the watchdog mechanism, some assumptions are made as follows:

- Packets are dropped only by malicious nodes. Therefore, no congestion is considered here.
- There is no error in the wireless channel.

4.5.1 Parameter setting

Simulation parameters are shown in Table 4.1. A source starts sending packets at time 0 for all simulations in order to replicate the analysis scenarios.

4.5.2 Node Implementation

In the simulation, there are 4 types of nodes, a regular node, a malicious node, a regular node with watchdog mechanism and a cop node. A regular node is a node that performs

Table 4.1: Parameter Setting for static networks

Parameters	**Setting**			
Number of nodes	4	9	16	8+1
Simulation area	300 x 300 m^2	600 x 600 m^2	900 x 900 m^2	300 x 700 m^2
Node buffer size	50 packets			
Propagation model	Two-ray ground model			
Transmission range	250 meters			
MAC Protocol	802.11 CSMA/CA			
Link Bandwidth	2 Mbps			
Routing Protocols	DSR			
Traffic type	CBR			
Packet size	64 Bytes			
Packet rate	0.5, 1, 5, 10 pkts/second			
Number of connections	1			
Watchdog - Maximum Threshold	10 pkts			
Watchdog - Time-out	0, 20 seconds			0 second
Mobile Cop - Maximum Threshold	-			10 pkts
Mobile Cop - Time-out	-			0 second

regular functions of ad hoc node, which participates in routing messages exchange and forwards data messages. A malicious node participates in routing message exchange but drops all data packets that pass through it and also drops all alarm messages that a detecting node wants to send back to a source (see Chapter 3).

A regular node with built-in watchdog mechanism performs the same functions as a regular node and additional watchdog mechanism. By implementing watchdog mechanism, each node monitors its neighbor to see whether it forwards packets to the next node or not. If it doesn't overhear a forwarded packet, it will keep counting the number of non-forwarded packets and if it reaches the maximum threshold, it will delay for a certain time-out to avoid a false accusation. After the time-out is expired and the suspicious node still does not forward data packets, it will send an alarm to the source. If the source detects a malicious node, it does not send an alarm but it will find another route to send packets without the detected malicious node in the path.

The last node is a cop node, which performs the detection function while other regular

nodes do not. Its mechanism is similar to the regular node with built-in watchdog mechanism but it will not forward any packets for other nodes (optional). The main purpose for this node is to detect a malicious node. Therefore, this node uses a promiscuous mode to listen to the wireless channel in order to overhear the transmission of its neighbors. If its neighbor does not forward data packets and the number of non-forwarding packets is over the threshold setting, it will broadcast an alarm message. Each node will forward the message back to the source. The source will change to another route after receiving it. The cop also listens to routing packets to identify its neighbors at the beginning of a simulation such that the cop does not falsely detect a regular node that is at a far distance as a malicious node when the cop is not within the node transmission range yet.

4.6 COMPARISON BETWEEN ANALYSIS AND SIMULATION

In this work, we compare results from analysis and simulation. There are 3 cases, simulation only, simulation with analysis, and analysis only. For "simulation only" case, the PDR is from our simulation model with a 100 second simulation time and 50 repetitions for each point with a 90% confidence interval. For "simulation with analysis" case, the average detection time from *simulation* is used to calculate the average throughput in the analysis instead of the estimate of detection time using load and time as previously described. For the last case, "analysis only", we will not use any information from simulation and it gives an upper-bound on the average throughput. This is because of the following reason. The threshold-based detection is used and the load is known. The detection time is estimated from the threshold setting. When a malicious node is detected, a new path has to be discovered. But the switching time to another route is not considered in this case.

4.6.1 Example 1: 4-node network - Watchdog

Our simulation models are simulated with two different parameter settings for watchdog mechanism. The first setting is with a 10 packet threshold and 0 second time out. The

second setting is the same as the first setting but the time out is changed to 20 seconds. Table 4.2 and Table 4.3 show the detection time in seconds for simulation and analysis for both settings.

Table 4.2: Detection time: 4 node network (Threshold = 10 pkts,Time out = 0 sec.)

Load (PPS)	0.5	1.0	5.0	10.0
Simulation (sec.)	24.78	12.10	2.38	1.25
Analysis (sec.)	20.0	10.0	2.0	1.0

Table 4.3: Detection time: 4 node network (Threshold = 10 pkts,Time out = 20 sec.)

Load (PPS)	0.5	1.0	5.0	10.0
Simulation (sec.)	43.86	31.56	22.26	21.15
Analysis (sec.)	40.0	30.0	22.0	21.0

In order to compare our analysis with simulation, PDR is used as a metric to plot against the loads in the network. Figure 4.12 and Figure 4.13 show the PDR from both analysis and simulation with 2 different settings. From the figures, it is clear that the analysis can give a close estimation to simulations.

4.6.2 Example 2: 9-node network - Watchdog

Table 4.4 and Table 4.5 show the detection time in seconds from simulation and analysis for both settings. As expected, the detection time from simulation is higher than the time from analysis because a new route has to be discovered before sending packets after detection. This delay is not considered in the analysis.

Figure 4.14 and Figure 4.15 show the result from both analysis and simulation with the two different parameter settings. The analysis results show similar trends as simulation results but the PDR from analysis is higher than the PDR from simulation. This is because the detection time from analysis is the best case scenario, which gives fastest detection time without considering new route recovery delays after a malicious node is detected. From

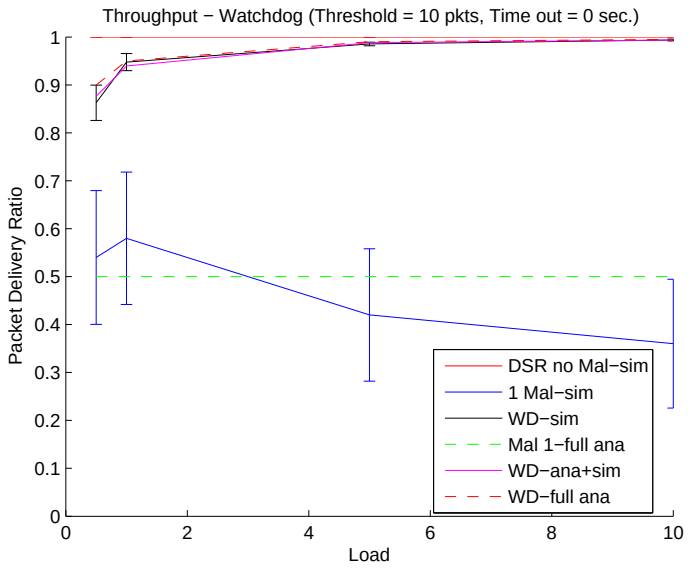


Figure 4.12: PDR: 4 node network (Threshold = 10 pkts,Time out = 0 sec.)

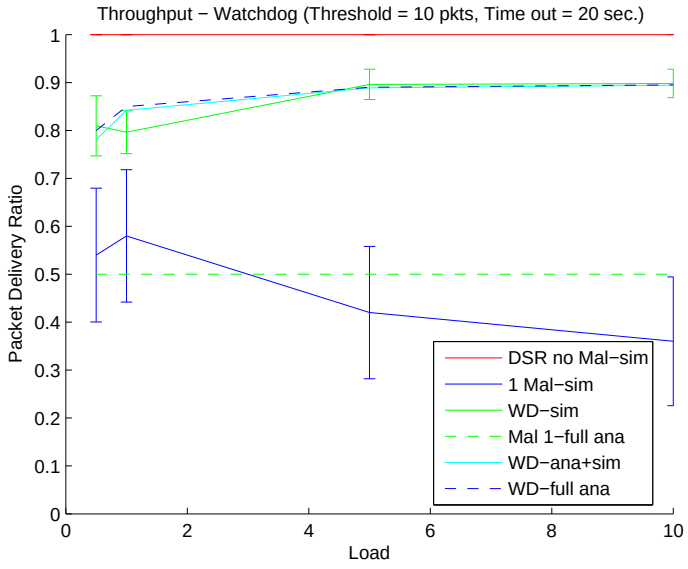


Figure 4.13: PDR: 4 node network (Threshold = 10 pkts,Time out = 20 sec.)

simulation results, false positives are possible at 10 packet per second load if the time out is set to zero. In contrast, the false positive does not happen with the time out is set to 20 seconds.

Table 4.4: Detection time: 9 node network (Threshold = 10 pkts,Time out = 0 sec.)

Load (PPS)	0.5		1.0		5.0		10.0	
	T_{det_1}	T_{det_2}	T_{det_1}	T_{det_2}	T_{det_1}	T_{det_2}	T_{det_1}	T_{det_2}
Simulation (sec.)	23.81	47.77	11.95	24.18	2.44	6.67	1.20	2.58
Analysis (sec.)	20.0	40.0	10.0	20.0	2.0	4.0	1.0	2.0

Table 4.5: Detection time: 9 node network (Threshold = 10 pkts,Time out = 20 sec.)

Load (PPS)	0.5		1.0		5.0		10.0	
	T_{det_1}	T_{det_2}	T_{det_1}	T_{det_2}	T_{det_1}	T_{det_2}	T_{det_1}	T_{det_2}
Simulation	43.20	86.76	31.84	65.26	22.31	47.54	21.16	45.30
Analysis	40.0	80.0	30.0	60.0	22.0	44.0	21.0	42.0

4.6.3 Example 3: 16-node network - Watchdog

For this example, the results are shown in Figure 4.16 and 4.17. The results from the analysis do not look similar to the results from the simulation when the time-out is 0. This is because the network is bigger and there are more paths to be chosen from a source to a destination. In addition, in the simulation, when a malicious node is detected, the source may not find another route, which does not contain a malicious node in the path. When a source does not find a route to the destination, all subsequent packets are dropped. In addition, false alarms can cause the PDR to drop at higher loads since the threshold is set to be quite low. Therefore, the PDR results from simulation are smaller than the results from analysis. When the time-out is 20 seconds, the results from analysis and simulation are similar because the false alarms are fewer when the time-out is increased but also the detection time is increased.

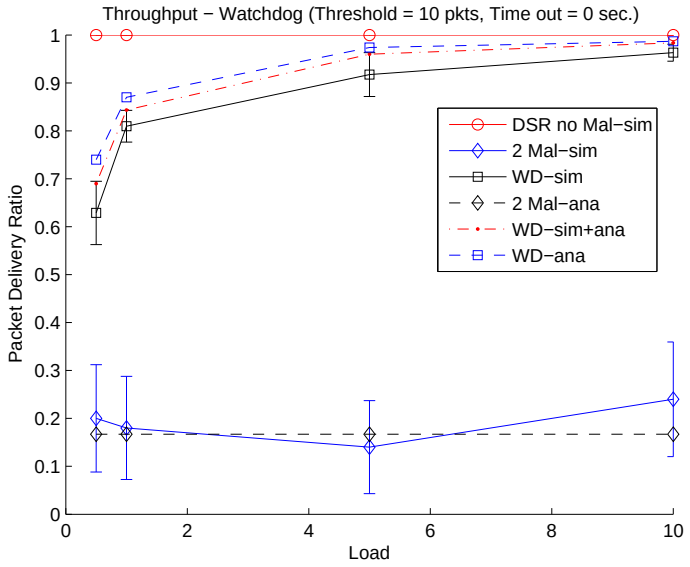


Figure 4.14: PDR: 9 node network(Threshold = 10 pkts,Time out = 0 sec.)

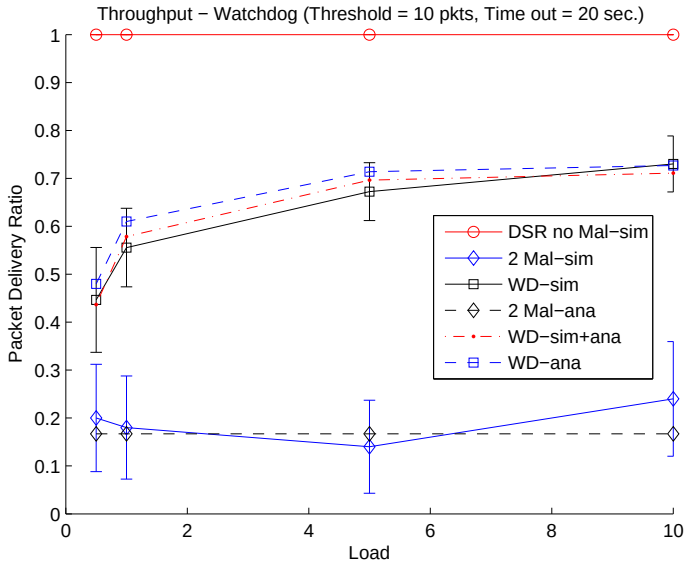


Figure 4.15: PDR: 9 node network (Threshold = 10 pkts,Time out = 20 sec.)

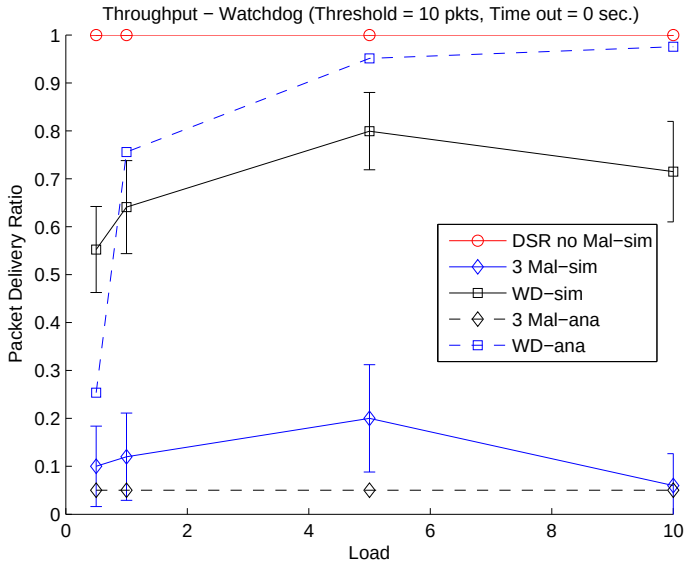


Figure 4.16: PDR: 16 node network (Threshold = 10 pkts,Time out = 0 sec.)

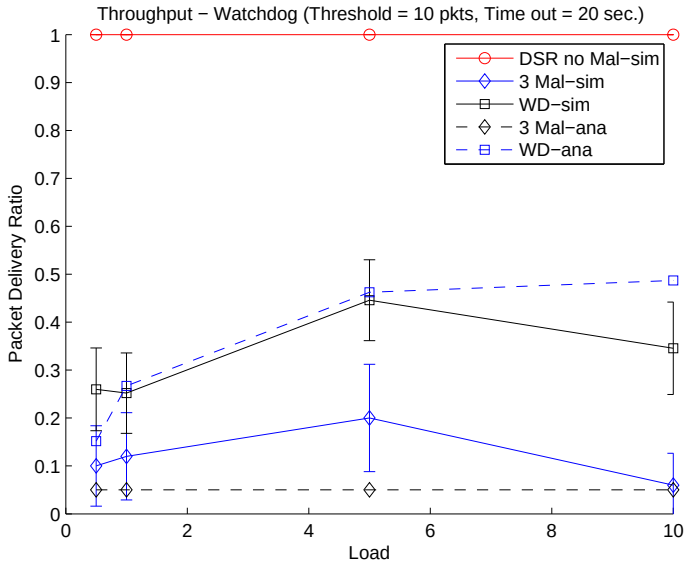


Figure 4.17: PDR: 16 node network (Threshold = 10 pkts,Time out = 20 sec.)

4.6.4 Example 4: 8+1-node network - Watchdog and Cop

The PDR from this study is shown in Figure 4.18. In this scenario, *Mobile Cop* can only detect a malicious activity when it is in the transmission range of the source. Therefore, it has a 100 meter detection duration to detect a malicious node. If it cannot detect a malicious node in the first round, it will come back and continue detecting the node when it is in the source's transmission range. This causes a significant delay for *Mobile Cop* to detect a malicious node but it can perhaps save energy for the other nodes. It is a trade-off between energy and throughput performance.

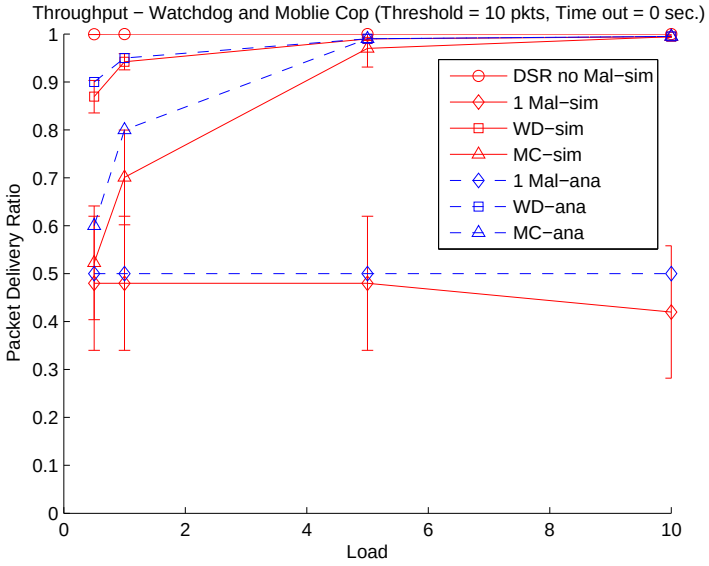


Figure 4.18: PDR: 8+1 node network (Threshold = 10 pkts,Time out = 0 sec.)

The number of monitored packets is shown in Figure 4.19 to investigate the energy savings in terms of detection function. The number of monitored packets is a performance metric when each detecting node has to listen to its neighbors to detect a malicious node. The result shows that watchdog consumes more energy for detection function than *Mobile Cop* does.

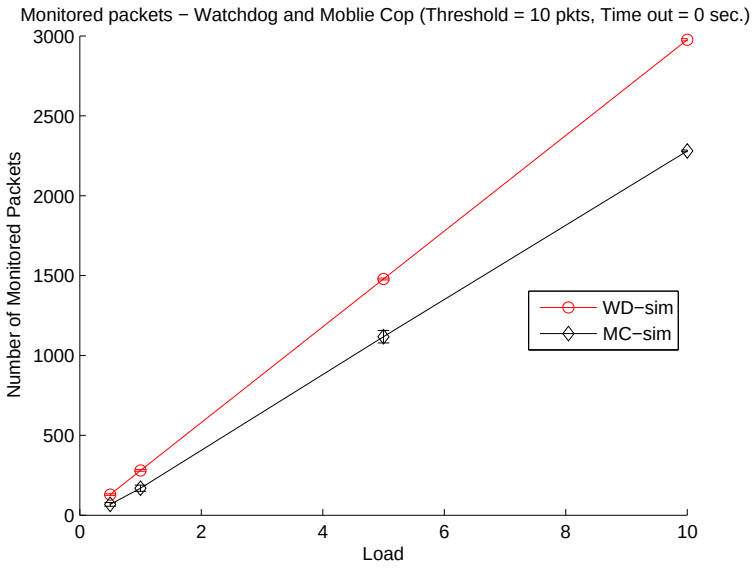


Figure 4.19: Monitored packets: 8+1 node network (Threshold = 10 pkts,Time out = 0 sec.)

4.7 ANALYSIS OF ASYMMETRY

In previous sections, all the communications are assumed to be bi-directional. An interesting issue emerges from the research literature that show the presence of asymmetric link communications in wireless ad hoc networks. Asymmetric links are defined as the wireless links that do not have the same transmission range between nodes but the transmission rate is the same. Asymmetry can exist because of radio propagation and receiver design, different transmit powers, etc. The simulations cannot be used without a lot of modifications.

Instead, we extend the simple analysis for studying this problem. From the previous analysis, a packet dropping attack works best when a malicious node is an intermediate node in the path between a source and a destination. A malicious node has higher chances to be chosen as an intermediate node when it has higher transmission range than other nodes and vice versa. Therefore in this analysis, a malicious node is set to have a higher transmission range than regular nodes. In order to analyze this problem, the following assumptions are introduced:

- One source and one destination are considered in a grid network.
- A malicious node performs a 100% data packet dropping attack.
- The chosen route is the shortest path (i.e., in terms of hop count) from a source to a destination.
- A malicious node has higher transmission power than a regular node.
- Routing protocol supports asymmetric link communication.

By default, the transmission range for regular nodes is 250 meters but, in this study, a transmission range for a malicious node is 500 meters. The reception sensitivity is the same for all nodes. The setting helps a malicious node shortcut some intermediate nodes in order to reach a destination faster than other routes. Therefore, a source always chooses a malicious node. In the case of the watchdog mechanism, Equation 4.2 does still hold but the probability of chosen path is changed as described in the examples in this section.

4.7.1 Example 5: 9-node network

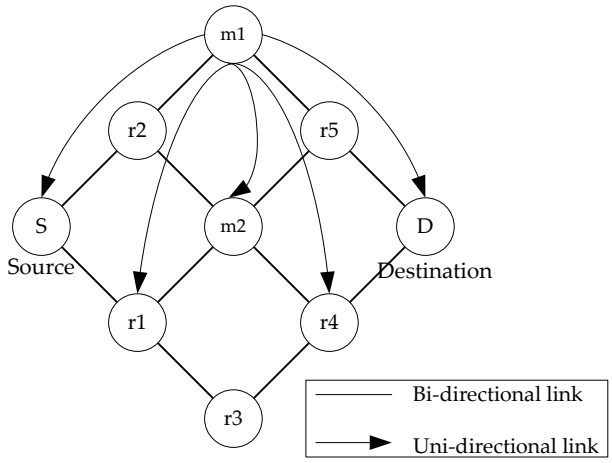


Figure 4.20: 9 node network topology with asymmetric link

This example has similar network set-up as Example 2. Figure 4.20 shows the topology with asymmetry due to the malicious node having a larger range. In this figure, only the range of m_1 is presented but m_2 also has the same range as m_1. In this case, both malicious

nodes, m_1 and m_2, are able to by-pass one node to make a route shorter. If the hop count is used as a routing metric, both malicious nodes are always chosen first as intermediate nodes. At the beginning, the total number of possible paths is 3 as follows:

$$S - r2 - m_1 - D$$
$$S - r2 - m_2 - D$$
$$S - r1 - m_2 - D$$

Therefore, the probability values in this scenario are:

$$\begin{aligned} P\{choose\, m_1\} &= \frac{1}{3} \\ P\{choose\, m_2\} &= \frac{2}{3} \\ P\{not\, choose\, m_1 \& m_2\} &= 0 \\ P\{choose\, m_1 | E_1\} &= 1 \\ P\{not\, choose\, m_1 | E_1\} &= 0 \\ P\{choose\, m_2 | E_1\} &= 1 \\ P\{not\, choose\, m_2 | E_1\} &= 0 \end{aligned}$$

In order to simplify the Equation, T_{det_1} and T_{det_2} are the detection times when the first and second malicious nodes are detected respectively. The final equation for this analysis is:

$$\begin{aligned} PDR_{avg} &= \frac{1}{3} \times \left(1 \times \frac{T - T_{det_2}}{T} + 0 \times \frac{T - T_{det_1}}{T}\right) \\ &\quad + \frac{2}{3} \times \left(1 \times \frac{T - T_{det_2}}{T} + 0 \times \frac{T - T_{det_1}}{T}\right) + 0 \\ &= 1 - \left(\frac{T_{det_2}}{T}\right) \end{aligned} \tag{4.12}$$

From the simplified equation above, it is important to note that both malicious nodes have to be detected and excluded in order to ensure that the PDR_{avg} is above zero.

4.7.2 Example 6: 16-node network

This example demonstrates a 16 node network with asymmetry. Unlike Example 3, malicious nodes can shorten the route to a destination such that they have more chances to be chosen as intermediate nodes. Figure 4.21 shows the topology with m_3 creating asymmetry as a demonstration. It is important to note that all malicious nodes have higher ranges than regular nodes.

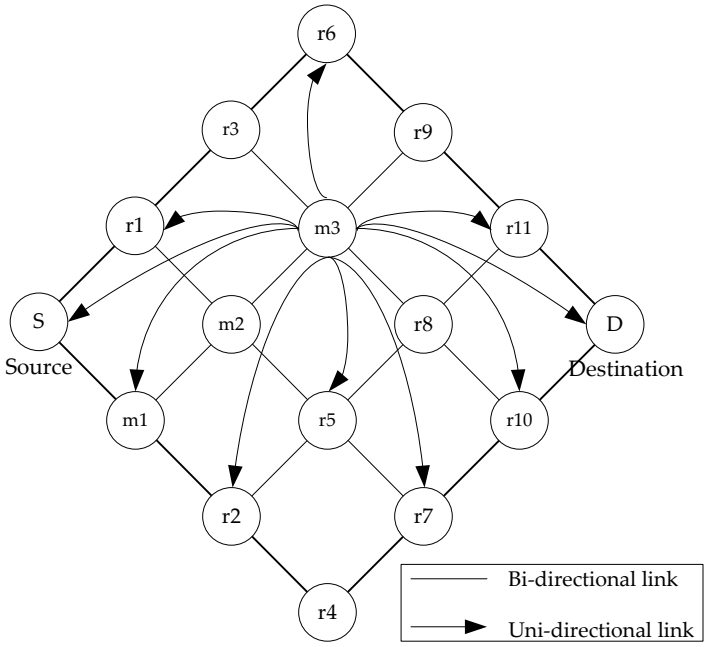


Figure 4.21: 16 node network topology with asymmetric link

There is only 1 possible path from a source to a destination with the shortest path at the beginning and 3 paths are to be chosen when m_1 is detected as follows:

$$S - m_1 - m_3 - D$$

$$S - r1 - m_2 - r10 - D$$

$$S - r1 - m_2 - r11 - D$$

$$S - r1 - r3 - m_3 - D$$

The probabilities for this example are:

$$\begin{aligned} P\{choose\, m_1\} &= 1 \\ P\{choose\, m_2\} &= 0 \\ P\{choose\, m_3\} &= 0 \\ P\{not\, choose\, M\} &= 0 \\ P\{choose\, m_2 | E_1\} &= \frac{2}{3} \\ P\{choose\, m_3 | E_1\} &= \frac{1}{3} \\ P\{not\, choose\, M | E_1\} &= 0 \end{aligned}$$

Therefore, the simplified PDR_{avg} is shown below:

$$\begin{aligned} PDR_{avg} &= P\{choose\, m_1\} \times PDR(choose\, m_1) \\ &\quad + P\{choose\, m_2\} \times PDR(choose\, m_2) \\ &\quad + P\{choose\, m_3\} \times PDR(choose\, m_3) \\ &\quad + P\{not\, choose\, M\} \times PDR(not\, choose\, M) \\ &= 1 \times \left[\frac{2}{3} \times \left(1 \times \frac{T - T_{det_3}}{T} + 0 \times \frac{T - T_{det_2}}{T} \right) \right. \\ &\quad + \frac{1}{3} \times \left(1 \times \frac{T - T_{det_3}}{T} + 0 \times \frac{T - T_{det_2}}{T} \right) \\ &\quad \left. + 0 \times \left(\frac{T - T_{det_1}}{T} \right) \right] \\ &\quad + 0 + 0 + 0 \\ &= 1 - \left(\frac{T_{det_3}}{T} \right) \end{aligned} \tag{4.13}$$

From these two examples, all malicious nodes are to be chosen first as intermediate nodes. Then a regular path is chosen after all malicious nodes are detected. It means that the detection time for each node has to be quick such that the PDR_{avg} can be increased.

4.7.3 Note on transmission range vs. detection mechanisms

When nodes in the network do not have similar transmission power, both watchdog and cop mechanisms could give a false alarm under conditions where the detector is not in the transmission range of both the sender and a forwarder.

For example in Figure 4.22(a), all nodes are regular nodes with watchdog mechanism built-in. When a sender (r1) has higher transmission power than a forwarder (r3), r1 doesn't receive any forwarding packets from r3 and non-forwarding packets in r1 will keep increasing and reach a threshold, thereby causing a false alarm. In contrast, when a source (S) sends packets to r1, S receives all packets that r1 forwards to its next hop node. This is a normal operation of watchdog. For cop mechanism in Figure 4.22(b), a cop node C1 overhears a transmission from S and r1 but not from r3. The cop will send a false alarm to the source S. A cop node C2 also cannot overhear a transmission from r3 and a false alarm message is sent. In contrast, a cop node C3 can overhear transmission from both r1 and r3, this is a normal detection scheme.

In the case of the watchdog mechanism, when a sender has a high transmission power and a forwarder has a low transmission power, false alarms could happen. However, when a sender has a low transmission power and a forwarder has a high transmission power, watchdog mechanism works properly. The rule of thumb for the cop mechanism is that a cop node has to be in both sender's and forwarder's transmission ranges in order to correctly detect a malicious node without a false alarm.

When malicious nodes do not have the same transmission range (high and low transmission ranges), the analysis can be combined with symmetric and asymmetric communications to calculate the PDR_{avg} of the network.

4.8 SUMMARY

In this chapter, an analysis of watchdog and cop mechanisms with a malicious node in a network is presented using a probability weighted average from a probability tree. The

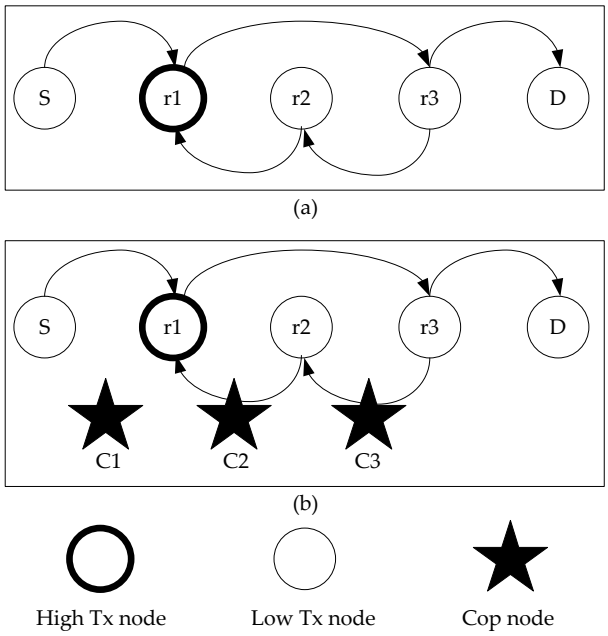


Figure 4.22: Example scenario - False alarm in asymmetric communication

impact of a malicious node and the detection mechanisms are analyzed in terms of the average PDR. From our analysis, the throughput is increased when the detection time is decreased and the detection time depends on the threshold and load in the network. In a static network, the analysis will be more complex if more malicious nodes are in the network and node density is higher. In addition, if the nodes are mobile, the probability of choosing a malicious node is changing over time and the analysis will be a lot more complicated. Therefore, the study of the detection mechanisms for mobile ad-hoc networks will use the simulation tool instead of analysis to study the performance in the next chapter.

For asymmetric communications, when a malicious node has a higher transmission range than other nodes, it will be chosen first as an intermediate node since the path is a shortest path. When a watchdog detection mechanism is implemented, all malicious nodes have to be detected and the PDR can be improved.

The analysis presented here can be extended to other scenarios and possibly attacks where it is possible to predict how routes are chosen from a source to its destination (i.e.,

with what probabilities). As demonstrated here, such an analysis, while more optimistic than simulations, can provide quick ideas of how the performance degrades and how quickly and to what extent it may be possible to restore the PDR in the network.

5.0 PERFORMANCE EVALUATIONS ON DETECTION MECHANISMS

In this chapter, the impact of the packet dropping attack is studied in three types of wireless networks, i.e., a static ad hoc network, a mobile ad hoc network (MANET) and a wireless mesh network (WMN). In addition, the performance enhancements (mitigation) with the Cop mechanism, is compared with the corresponding improvement with the watchdog mechanism. In this chapter, as throughout this dissertation, a malicious node will drop all data packets if it is chosen as an intermediate node along a route and its impact is shown in this chapter.

The simulation tool for this study is ns-2 simulator [75], which has been widely used in research community for studying the performance in both wired and wireless networks.

5.1 PERFORMANCE METRICS

Three performance metrics are presented here as follows:

Packet Delivery Ratio (PDR) is like the goodput performance of the network. PDR is the ratio of the total number of received packets to the total number of sent packets as described in Chapter 4.

$$PDR = \frac{Total\ number\ of\ received\ packets\ at\ destination}{Total\ number\ of\ sent\ packets\ by\ source}$$

Routing Overhead is the total number of routing messages that are sent within the network. For the detection mechanisms, alarm messages are counted as a routing overhead along with other routing messages.

Detection Efficiency Ratio (DER) is the ratio of the total number of received packets from all destinations to the total number of monitored packets. The total number of monitored packets is the number of packets that nodes have to monitor (listen to neighbors) for detecting malicious activity. A monitoring node is a source or a regular nodes in a path between a source and a destination. Since in a detection function, each node needs to process each monitored packet by receiving them, this function consumes energy in the received mode for each packet. The higher the DER, the better is the detection performance. This is due to the fact that the system uses fewer numbers of monitored packets (or the energy) for the detection function in order to detect a malicious node.

$$DER = \frac{Total\,number\,of\,received\,packets\,at\,destination}{Total\,number\,of\,monitored\,packets\,at\,monitoring\,node}$$

The detection time is defined as the time taken to detect a malicious node since the start of the simulation. Generally, if a malicious node is detected within a small detection time, the PDR will be high since the route can be changed sooner such that packets will not be routed through a malicious node and, thus, more packets will reach the destination.

5.2 SIMULATION PARAMETER SETTINGS

ns-2 simulator is used as a tool to study performances of a packet dropping attack and its mitigation. Simulation parameters are presented in Table 5.1. Parameters could be changed within studies and they will be specified appropriately.

5.3 AD-HOC NETWORK

In this study, static ad hoc networks are simulated to evaluate effects of the packet dropping attack on the network's PDR and also how well the detection mechanisms improve the throughput performance when the attack is deployed. To study the effects, the worst case and random case scenarios in a grid-like network are implemented. Simulation areas are

Table 5.1: Parameter Setting - Watchdog and Cop mechanisms

Parameters	**Static Ad-Hoc network**	**MANET**	**WMN**
Number of nodes	16 / 49		
Simulation area	500 × 500 m^2 / 1000 × 1000 m^2		
Node buffer size	50 packets		
Propagation model	Two-ray ground model		
Transmission range	250 meters		
MAC Protocol	802.11 CSMA/CA		
Link Bandwidth	2 Mbps		
Routing Protocols	Dynamic Source Routing (DSR)		
Traffic type	CBR		
Packet size	64 bytes		60 bytes
Packet rate	1 packet/second		
Number of connections	5		
Mobility Model	-	Random waypoint	-
Pause time	-	0 / 60 seconds	-
Node speed (average)	-	1 / 10 m/s	-
Number of gateways	-		1 / 2
Watchdog - Max. threshold	10 packets		
Watchdog - Time-out	0 second		
Watchdog - Det. Interval	50 % in 100 seconds / 100 %		
Cop - Max. threshold	10 packets		
Cop - Hold-down timer	0 second		
Cop - Speed	10 m/s		

500×500 m^2 and 1000×1000 m^2. Number of nodes are 16 and 49 nodes (plus detection nodes in the case of the Cop mechanism). Results are discussed for each network scenario.

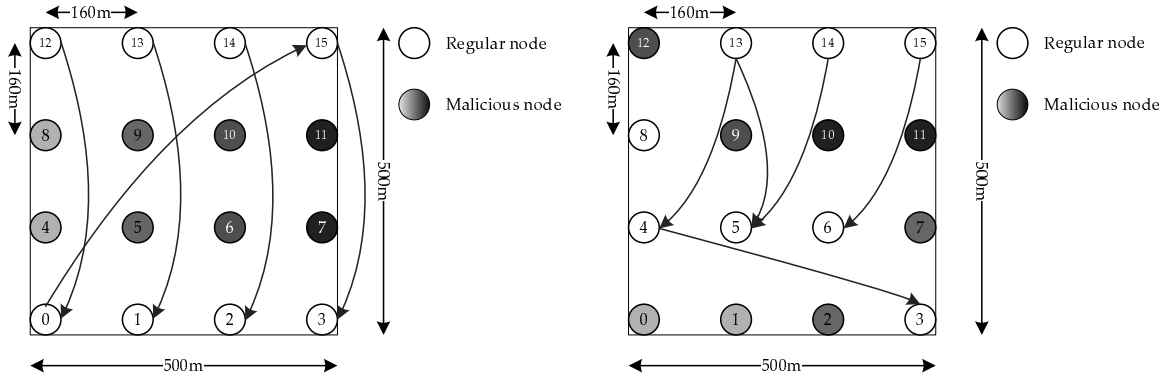
5.3.1 16-node network in 500m × 500m area

16-node grid-like networks are shown in Figure 5.1 for the worst case and random case scenarios. 5 connections are considered. The network connectivity is shown in Figure 5.2. In scenario 1, all malicious nodes are in between sources and destinations. Scenario 2 is a random case where sources and destinations are randomly chosen in the network. The color of malicious nodes in the figures varies from gray color to black color. A lighter-gray node represents a malicious node that is active earlier than a darker-gray node. For example, with 12.5% of the nodes being malicious nodes, only the two lightest gray nodes are active in the network. Other gray nodes are not active during the simulation and they perform as regular nodes.

In the case of Cop mechanism, 1 Mobile Cop and 4 Static Cop schemes are studied. In a mobile cop scheme, a Cop node moves back and forth in a straight line within the middle of the network as shown in Figure 5.3 at 10 m/s speed. The 4 static cop scheme uses 4 static cop nodes to detect malicious nodes in the network as shown in Figure 5.4. Cops have an option to help with forwarding packets or not. When a Cop does not help forwarding packets, the label in the results is "no help" at the end and its function is only the malicious node detection. When it helps in forwarding packets, there is no label in the results and it acts as a malicious node detector and a regular node.

Results

There are two scenarios for this study. Figure 5.5 shows the PDR results from the simulations. The first one is a worst case scenario and the impact from malicious nodes are significant as expected. The PDR is decreased when the number of malicious nodes are increased. When 50% of nodes are malicious nodes, the PDR is dropped to zero. Each scheme and its variation perform with similar trends. The highest PDR is with the 4 static cop scheme since it adds 4 more nodes to the network. An interesting observation is that when 4 cop nodes don't help in forwarding data packets, it gives higher PDR than 4 cop



(a) scenario 1 (worst case) (b) scenario 2 (random case)

Figure 5.1: 16 node network - 500m × 500m

nodes, which help forwarding packets. This is because of the different alarm schemes. When a cop node helps with forwarding packets, it will send an alarm as a unicast packet to a source but this alarm message could be dropped by any malicious node. However, when a cop node does not help with forwarding packets, it will broadcast an alarm message to its neighbors, which will find a route and forward packets to the source. This is because the cop node does not learn any route to any destination and the broadcast scheme is used. In this way, the alarm message has a higher chance to reach the source so that a new route will be established. The PDR is the highest with this scheme.

PDRs in watchdog schemes are better than those in one mobile cop scheme. The PDRs with 100% detection and 50% detection watchdog schemes are similar but 100% detection watchdog gives a little better PDR than 50% detection watchdog. It is important to note that 50% detection watchdog starts working when the first overheard packet is received. Nodes in the same path is almost synchronized. There is an exception if nodes forward packets for multiple connections. The watchdog parameter setting plays an important role in the performance. The threshold should be set such that the detection period is within one detection period in 50% detection for detecting a malicious node faster and improving the PDR. That's why 50% detection scheme performs similar to 100% detection scheme. In the case of the one mobile cop scheme, a mobile cop can still improve the PDR, with both

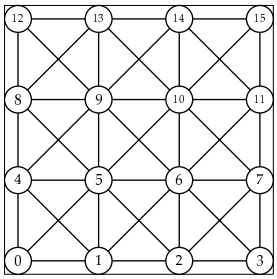


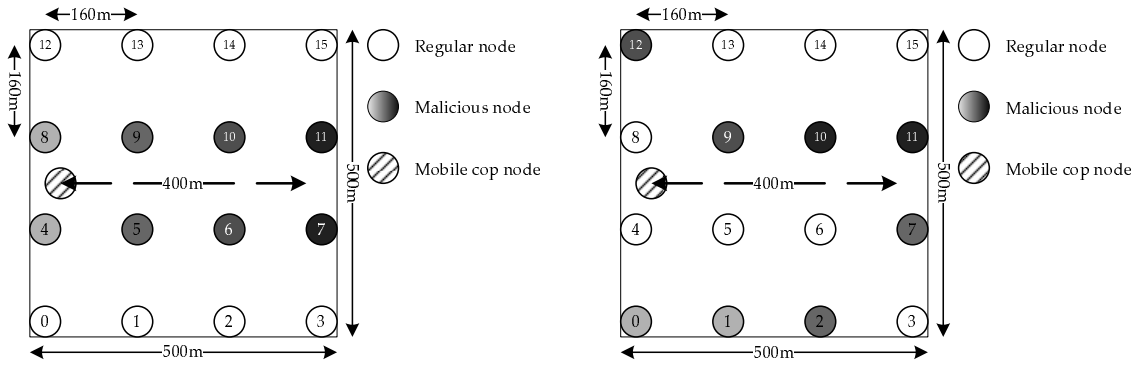
Figure 5.2: 16 node network - Network connectivity - 500m × 500m

"help" and "no help" schemes, but not much since it takes more time to detect malicious nodes in the network.

Another scenario is the random case (scenario 2), where sources, and destinations are randomly chosen. However, malicious nodes are chosen sequentially from a list of nodes that are not sources or destinations. The trend of the results is similar to the previous scenario. In the case of the mobile cop scheme, and a small fraction of malicious nodes, the PDR is less than the PDR for only DSR because the cop concluded too early that a regular node is a malicious node. A detection time-out is needed for the cop to correctly detect a malicious node in this network. PDRs from watchdog and 4 static cop node schemes are close to each other because there is an alternative route to the destination after detection, and also help from static cop nodes is not important in this scenario.

Figure 5.6 shows the routing overhead from each detection scheme for both scenarios. In the worst case scenario, 100% detection and 50 % detection, watchdog mechanisms result in the highest routing overhead in the network. With 50% of nodes being malicious, each source periodically keeps trying to find an alternative route but it is not possible to find the route and that adds more overhead to the network. Cop mechanisms add more routing overhead to the network as well. For the random case, the routing overhead for all schemes are not significantly different since an alternative route to a destination can be found easily.

The last metric is DER, shown in Figure 5.7. The worst performance for both cases is using 4 static cops since they monitor all packets in the network and it causes the highest



(a) scenario 1 (worst case) (b) scenario 2 (random case)

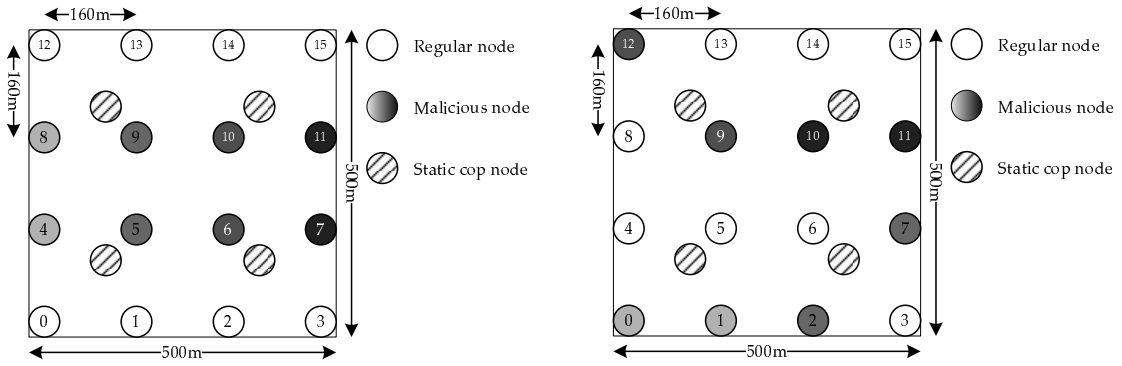
Figure 5.3: 16 node network with 1 MC - 500m × 500m

number of monitored packets. For this scheme, the PDR performance is high but DER performance is low and it is not suitable for an energy efficient scheme if the energy consumption of cops is a significant factor. However, our assumption is that cop nodes have better capability than other nodes. With this assumption, the overall network lifetime itself can be improved with this scheme since all other nodes do not spend energy for the detection mechanism. In contrast, if the assumption is not true, the 4 static cop scheme gives the worst performance in terms of the DER metric.

The best DER performance is with the 50% watchdog mechanism since a node does not monitor its neighbors all the time but it still can detect a malicious node. Therefore, the number of monitored packets is cut in half from the 100% watchdog scheme. A mobile cop scheme gives a little higher DER performance than a full watchdog scheme because it uses less numbers of processed packets than a full watchdog scheme.

5.3.2 Effect of size of area

In this study, the worst case scenario is considered as the base network and the network topology is similar to the previous study but in a larger simulated area as shown in Figure 5.8. The main difference between this study and the previous study is the network connectivity. This network represents a sparse network, instead of a dense network. The network



(a) scenario 1 (worst case) (b) scenario 2 (random case)

Figure 5.4: 16 node network with 4 SCs - 500m × 500m

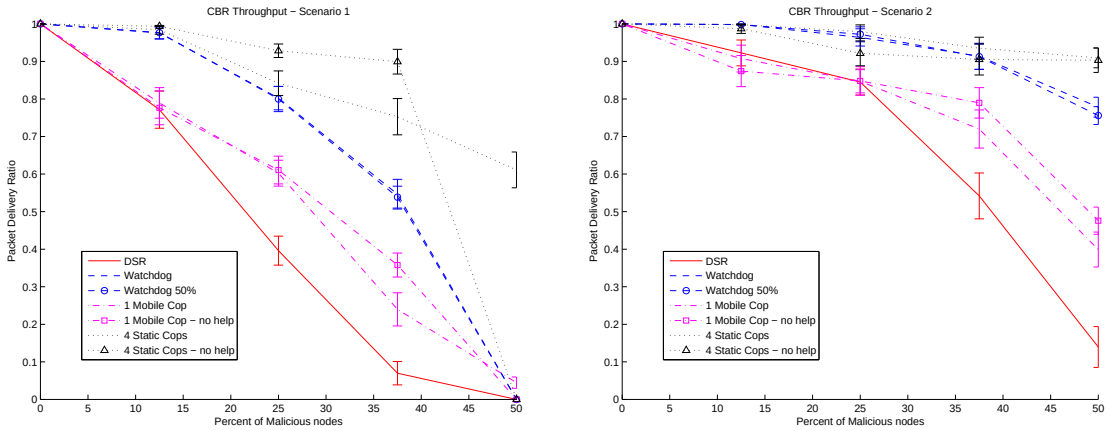
connectivity is shown in Figure 5.9.

Since the network is big and a mobile cop cannot travel in a straight line to monitor all nodes in the network, it travels in a square shape within the network as shown in Figure 5.10. In the 4 static cop scheme, the cop nodes are added to the network as shown in Figure 5.11 to cover all the nodes but they cannot communicate directly with each other.

Results

When a network is larger, the number of possible paths between nodes are reduced. For a large network, paths that lead to destinations are decreased but, in the worst case scenario, the probability to choose a malicious node is also reduced. Watchdog mechanisms in the large network give better PDR performance than those in a small network as shown in Figure 5.12(a). Since a mobile cop's moving path is different in order to cover all of the nodes, the PDR performance for a mobile cop in a small network is improved more than that in a large network as shown in Figure 5.12(b). This is because the cop needs more time to cover its neighbors in a large network and, therefore, the detection time is longer. In the case where 4 static cops are used, the PDR in a small network is a lot higher than that in a large network because they can help in forwarding packets to destinations but this is not possible in a large network. However, it helps improving the PDR performance compared to regular DSR protocol.

Watchdog mechanism in a small network produces more routing overhead than that in



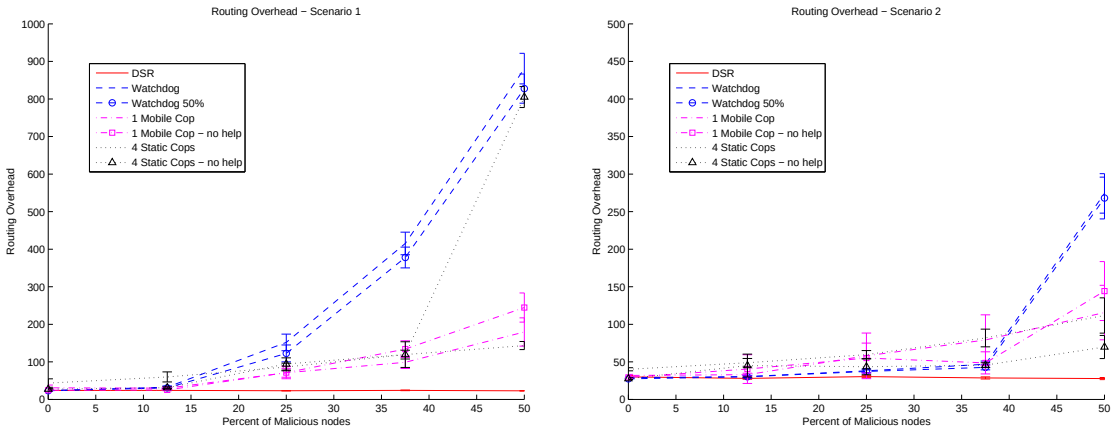
(a) scenario 1 (worst case) (b) scenario 2 (random case)

Figure 5.5: PDR: 16 node static network - 500m × 500m

a large network when the number of malicious nodes is increased because nodes respond to routing packets in a small network more than those in a large network and the number of routing messages is higher in a small network as shown in Figure 5.13. It is important to note that malicious nodes drop all alarm messages, which may not be delivered to a source in a large network. For cop mechanisms, the routing overhead for mobile cop is similar for both small and large networks. An interesting observation is that 4 static cop mechanism in a large network generates higher routing overhead than all others because the cop nodes keep sending alarm messages back to sources if they have not changed the route containing a malicious node and the sources will all do route discovery repeatedly. This causes a high overhead in 4 static cop scheme.

When the DER metric is considered as shown in Figure 5.14, the watchdog mechanism in a small network gives a better DER performance than those in a large network because the number of processed packets in a small network is less than that in a large network. In contrast, DER for cop mechanisms in a large network is higher than those in a small network since cops processed fewer numbers of packets. The cops overhear less number of packets from its neighbors when the area is large.

In a large network, PDR performance depends on node density. When node density is



(a) scenario 1 (worst case) (b) scenario 2 (random case)

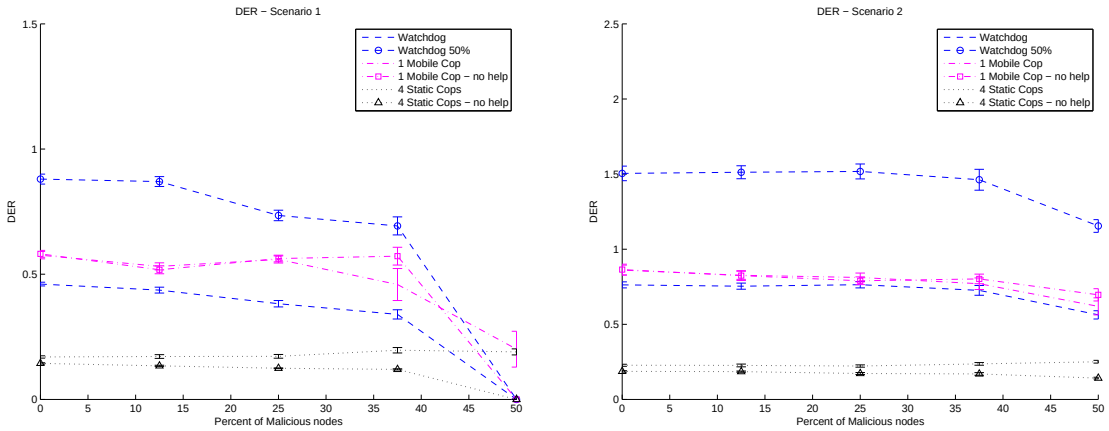
Figure 5.6: Routing overhead: 16 node static network - 500m × 500m

high, a source has more chances to choose a path not containing a malicious node or a new path without a malicious node after the first malicious node is detected. The PDR can be thus increased.

When comparing detection mechanisms, watchdog mechanism gives high PDR than mobile cop mechanism because watchdog can detect malicious nodes faster than mobile cop. When a static cop is implemented, it has some limitations if it is not in the range of both the sender and the forwarder since it cannot detect a malicious node in that case. Therefore, watchdog performs better than cop when the network is large.

5.3.3 Effect of number of nodes

In this study, the number of nodes is increased from 16 nodes to 49 nodes in 1000 × 1000 $meters^2$. A network setup for 49 node network is shown in Figure 5.15 as a worst case scenario and the network connectivity is shown in Figure 5.16. The number of nodes is increased but the percentage of malicious node and number of connections is roughly the same. It is important to note that both networks are not directly comparable in terms of the overall area for all nodes located and the malicious node locations. However, this study shows the effect of detection mechanisms in a certain circumstance.



(a) scenario 1 (worst case) (b) scenario 2 (random case)

Figure 5.7: DER: 16 node static network - 500m × 500m

Results

The PDRs for watchdog mechanisms improve in both 16 node and 49 node networks but the PDR in the 16 node network is higher than the PDR in the 49 node network, as shown in Figure 5.17. Due to the number of malicious nodes, the probability to find a good route to a destination is high in the 16 node network. However, cop mechanisms do not improve the PDR performance. The problem with the cop mechanism performance is that the cop nodes do not provide enough coverage for monitoring all nodes in the network. If the number of cop nodes is increased to cover all the nodes, the PDR performance will be improved.

The routing overhead performance is shown in Figure 5.18. Watchdog mechanisms for both 16 and 49 node networks give similar numbers for the routing overhead. For cop mechanisms, 4 static cop mechanism in the 16 node network results in a higher overhead than others because the source keeps trying to find a new route to a destination after detection. In addition, static cops keep sending alarm messages but it is less likely to reach the destination than in the 49 node network because alarm messages are dropped by malicious nodes.

The DER is shown in Figure 5.19. As expected, watchdog 50% gives better DER than watchdog 100%. The DER for 50% watchdog for 16 node network is higher than 100% watchdog because it has fewer nodes to monitor the malicious activity. For cop mechanisms,

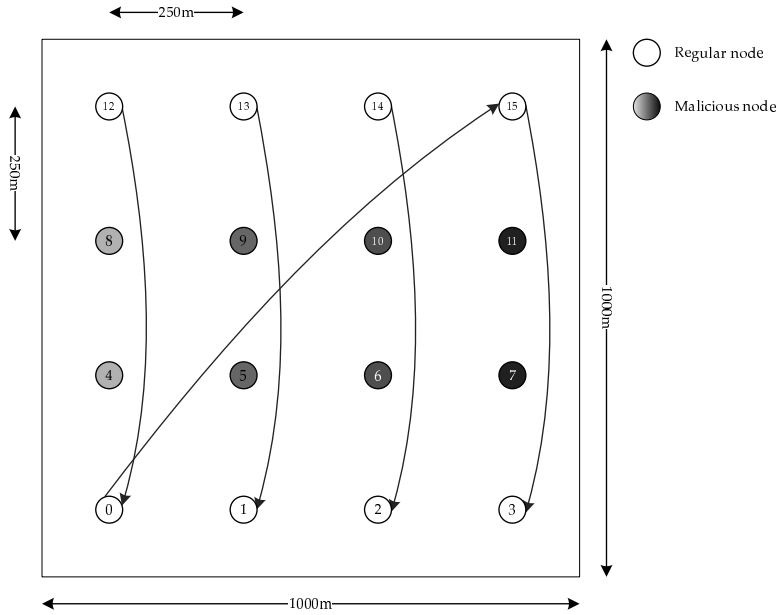


Figure 5.8: 16 node network - scenario 1 (worst case) - 1000m × 1000m

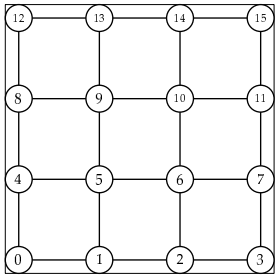


Figure 5.9: 16 node network - Network connectivity - 1000m × 1000m

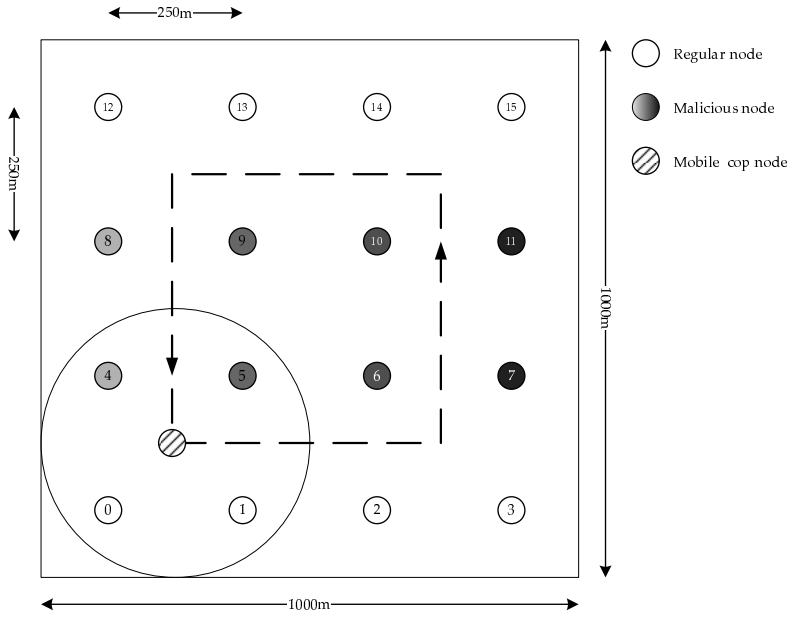


Figure 5.10: 16 node network with 1MC - scenario 1 (worst case) - 1000m × 1000m

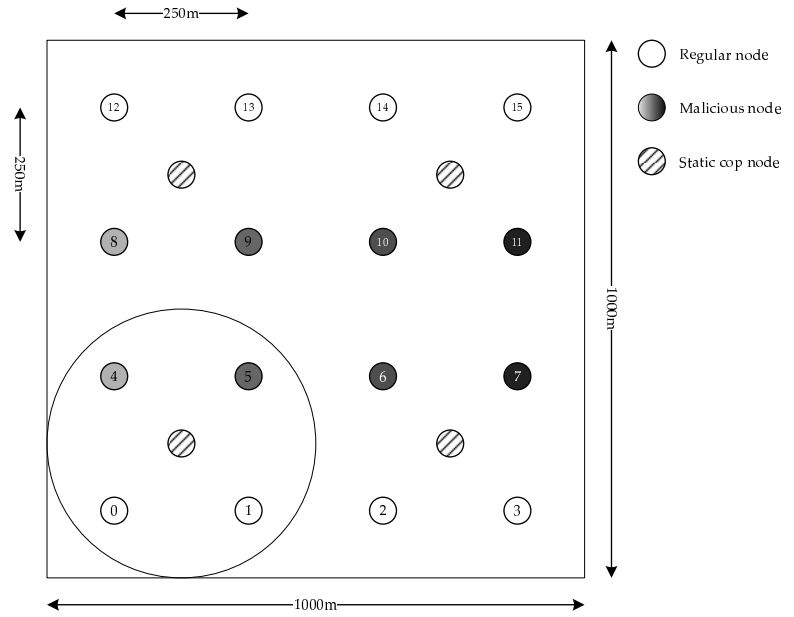
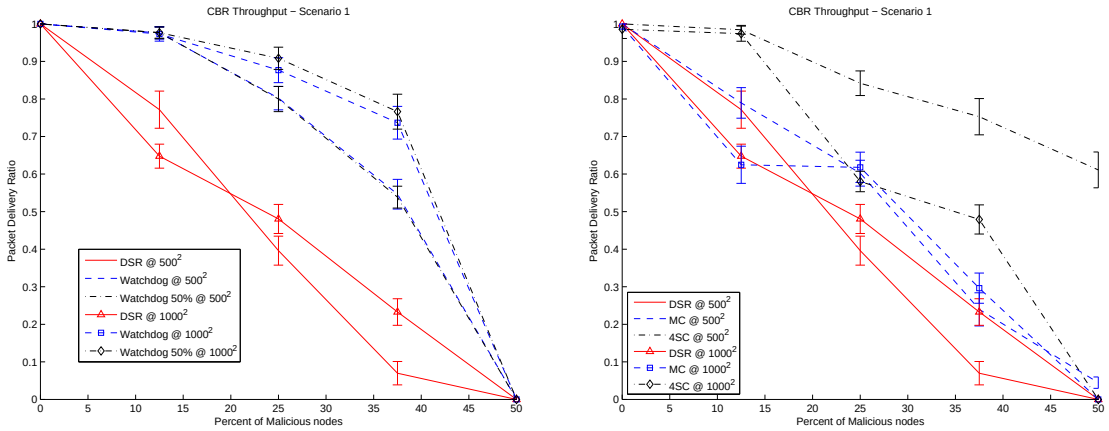


Figure 5.11: 16 node network with 4SCs - scenario 1 (worst case) - 1000m × 1000m



(a) Watchdog mechanisms (b) Cop mechanisms

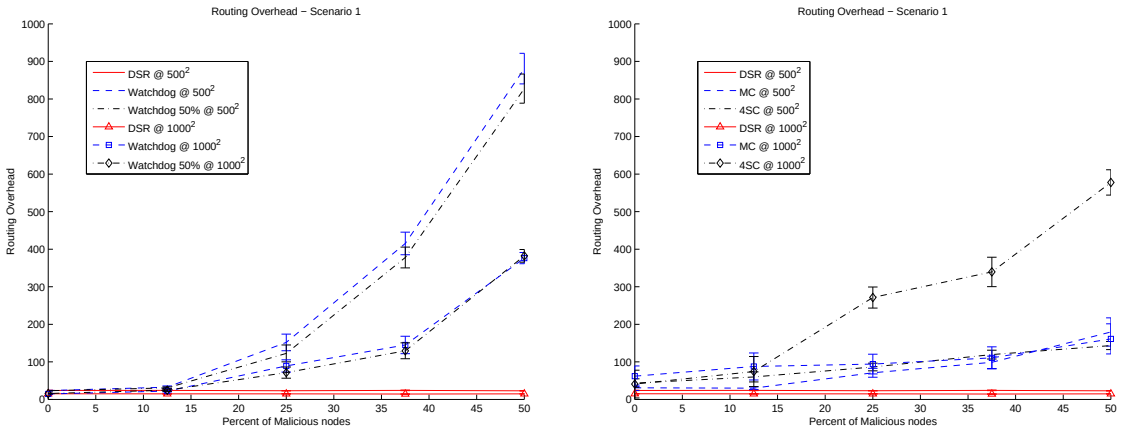
Figure 5.12: PDR: 16 node static network - Effect of area size

the mobile cop mechanism for 16 node network is the best because it monitors fewer nodes in the network.

5.3.4 Effect of Mobility

In this study, nodes are moving inside a 500 × 500 m^2 network with a random way-point mobility model. The model has two parameters, a pause time and an average speed. The pause time is the time that a node stops moving for changing to another direction. The average speed is a speed of a node on average when it moves within a network. Our study chooses 0 and 60 seconds for pause time and 1 and 10 m/s speeds for average speed. 0 second pause time means that a node continues moving inside the network without stopping before changing its direction. A node stops moving for 60 seconds before changing its direction in 60 second pause time. An average of 1 m/s is a walking speed and an average of 10 m/s is a city speed at 22.39 MPH. In the simulation, two mobility patterns and two connection patterns are simulated for each study.

This study is implemented in 500 × 500 m^2 area and all nodes are moving within the area with the pause time of 0 and 60 seconds and a speed of 1 and 10 m/s. A speed of 1 m/s is represented as “lo” and a speed of 10 m/s is represented as “hi” in figures below. At



(a) Watchdog mechanisms

(b) Cop mechanisms

Figure 5.13: Routing overhead: 16 node static network - Effect of area size

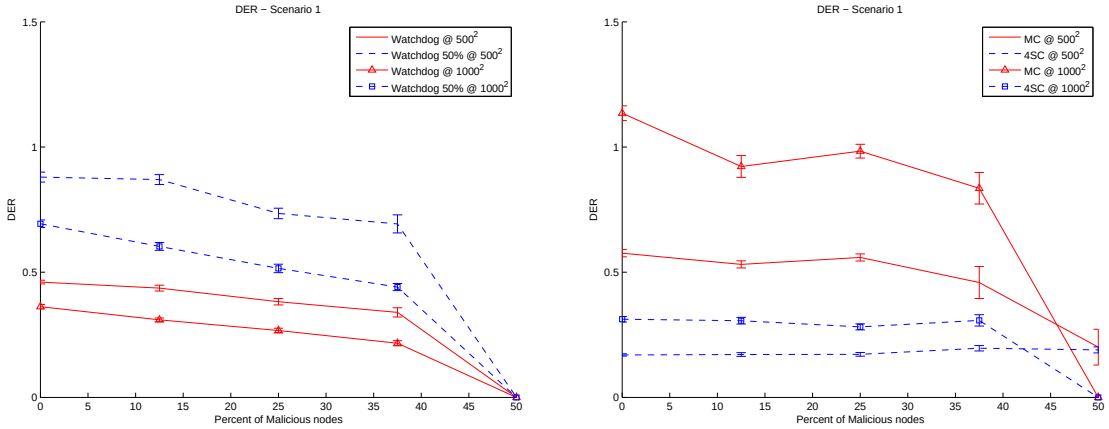
0 second pause time and high mobility speed (p0,hi), the network is highly mobile and the network topology is changed frequently. On the other hand, at 60 second pause time and low mobility speed (p60,lo), the network does not change its topology frequently. The other two settings, (p0,lo) and (p60,hi), are in between. For the cop mechanism, the cop node is fixed at the center of the area and monitors its neighbors that pass through the monitoring area.

Results

In (p0,hi), the PDR for watchdog 50% is worst because nodes cannot keep up with the mobility and then the detection function is deactivated as shown in Figure 5.20. When a node reactivates, the network topology is changed. This causes a false alarm and the PDR is lowest. The static cop mechanism performs as good as regular DSR in term of PDR. Since the network is highly mobile, the detection mechanisms do not perform well.

In (p60,lo), the network is less mobile, all detection mechanisms perform well as shown in Figure 5.21. The static cop and watchdog mechanisms have similar PDR performance. For the other two cases, the 100% watchdog gives highest PDR and static cop and 50% watchdog gives similar PDR performance.

When the routing overhead is considered as shown in Figure 5.22 and 5.23, a static cop



(a) Watchdog mechanisms

(b) Cop mechanisms

Figure 5.14: DER: 16 node static network - Effect of area size

produces a high number of routing overhead packets in all cases since it monitors all nodes activity and it keeps sending an alarm message if a source uses a route containing a malicious node in a path.

As far as the DER performance goes, Figure 5.24 and 5.25 show that a static cop consumes more energy than watchdog and watchdog 50% because the cop node monitors packets at the center of the area and all nodes are moving toward the center of the area.

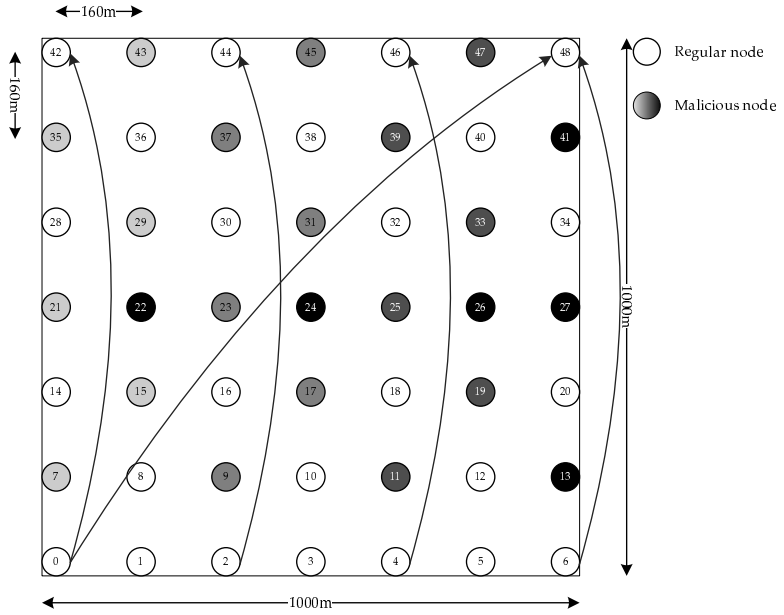


Figure 5.15: 49 node network - scenario 1 (worst case) - 1000m × 1000m

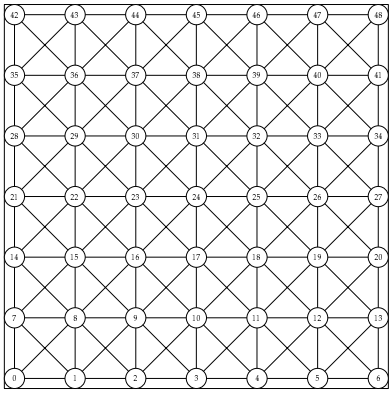
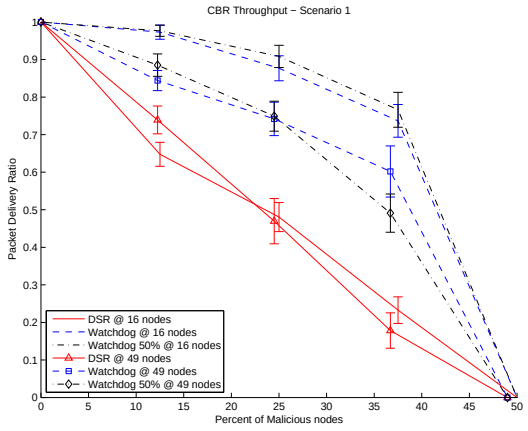
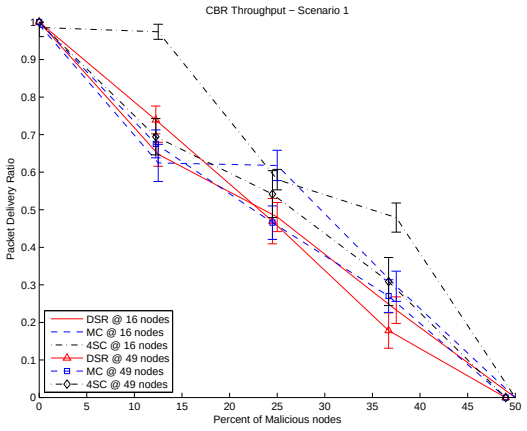


Figure 5.16: 49 node network connectivity - 1000m × 1000m

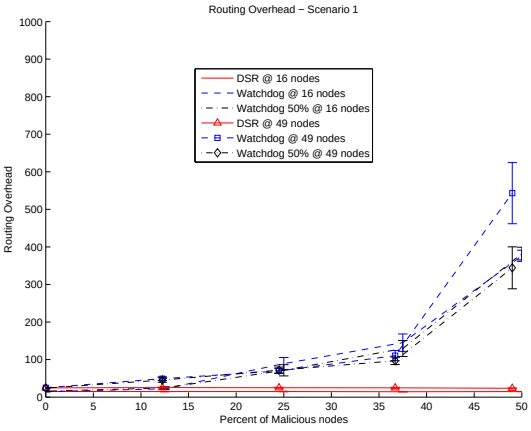


(a) Watchdog mechanisms

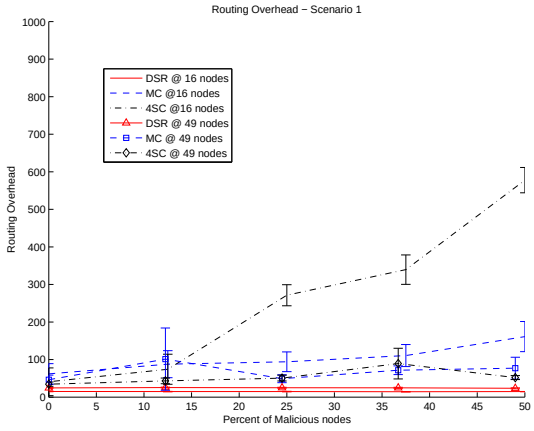


(b) Cop mechanisms

Figure 5.17: PDR: a static network - Effect of number of nodes

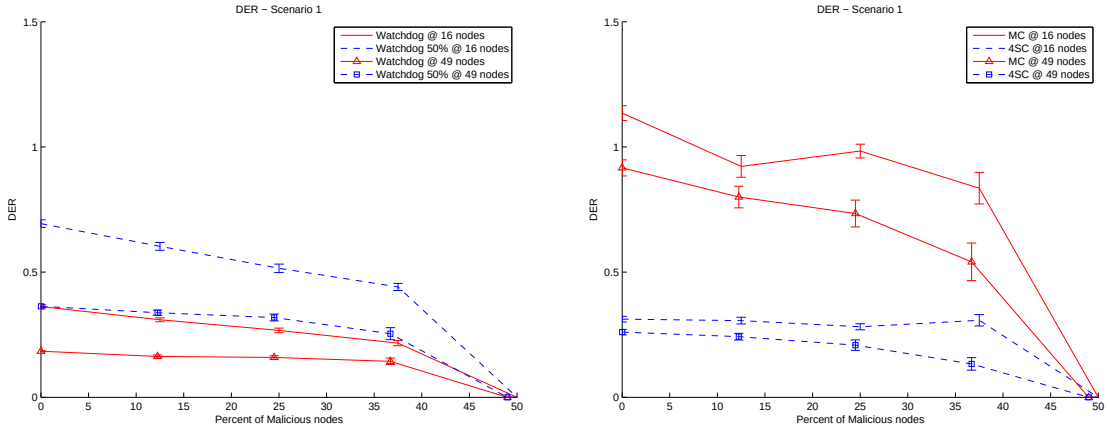


(a) Watchdog mechanisms



(b) Cop mechanisms

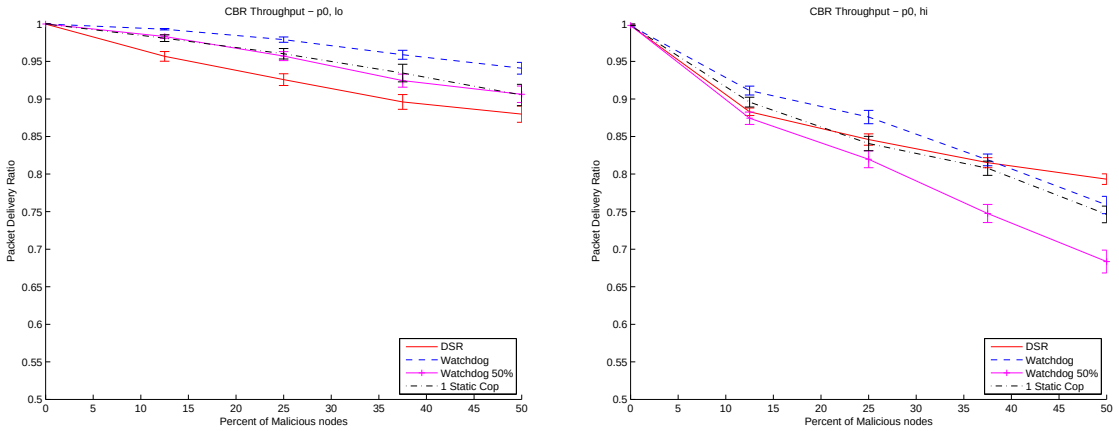
Figure 5.18: Routing overhead: a static network - Effect of number of nodes



(a) Watchdog mechanisms

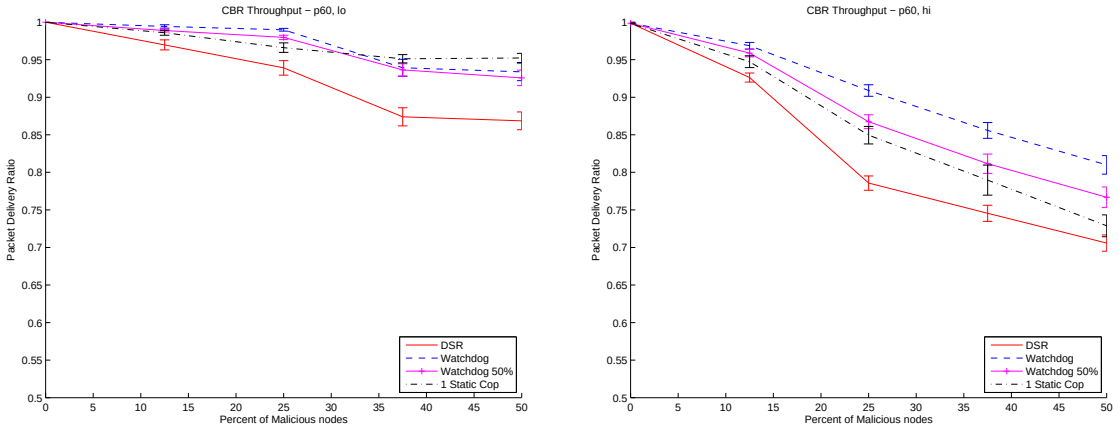
(b) Cop mechanisms

Figure 5.19: DER: a static network - Effect of number of nodes



(a) Low speed (b) High speed

Figure 5.20: PDR: 16 node network - Pause time = 0 sec. - 500m × 500m

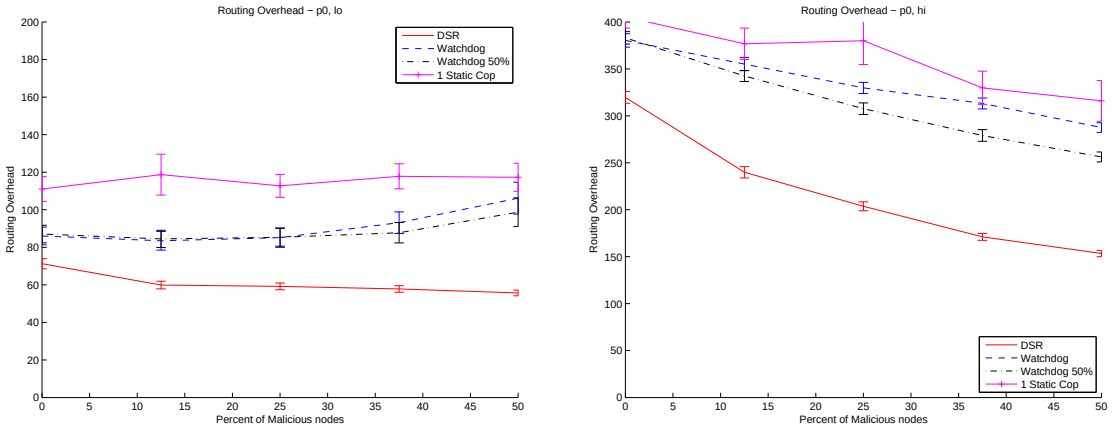


(a) Low speed (b) High speed

Figure 5.21: PDR: 16 node network - Pause time = 60 sec. - 500m × 500m

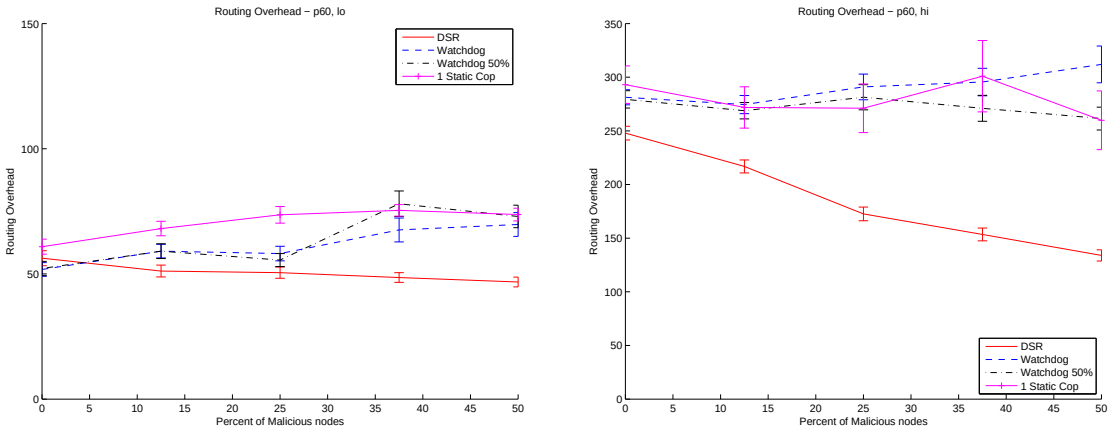
5.3.5 Effect of threshold and time-out setting

In this study, the analysis in Chapter 4 on the detection parameters is simulated to prove its concept for a static ad hoc network. A 500 × 500 m^2 static network is in the worst case scenario topology with 2 different threshold and time-out settings. The former setting is a 15 packet threshold and 15 second time-out, represented as "1515". The latter setting is a



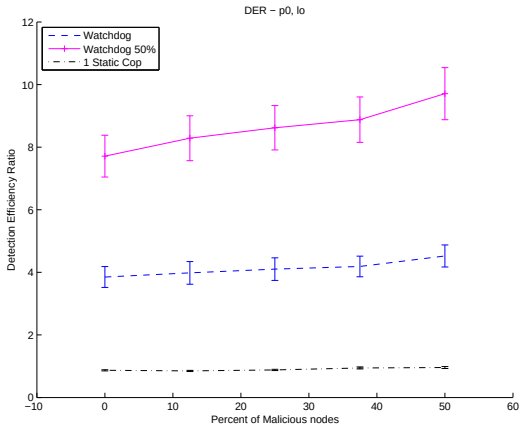
(a) Low speed (b) High speed

Figure 5.22: Routing overhead: 16 node network - Pause time = 0 sec. - 500m × 500m

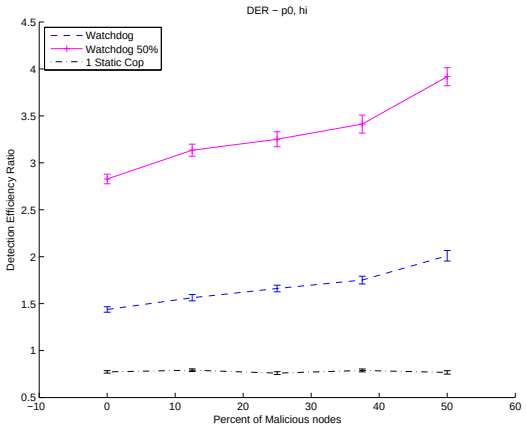


(a) Low speed (b) High speed

Figure 5.23: Routing overhead: 16 node network - Pause time = 60 sec. - 500m × 500m

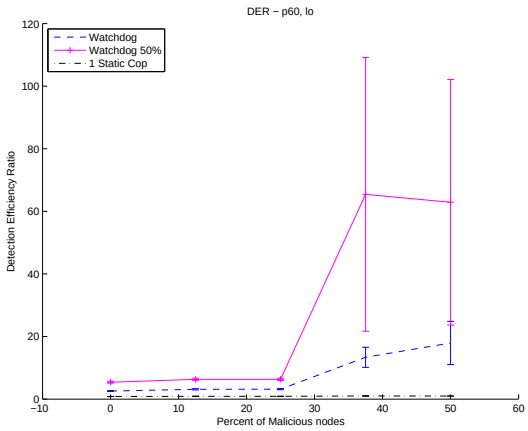


(a) Low speed

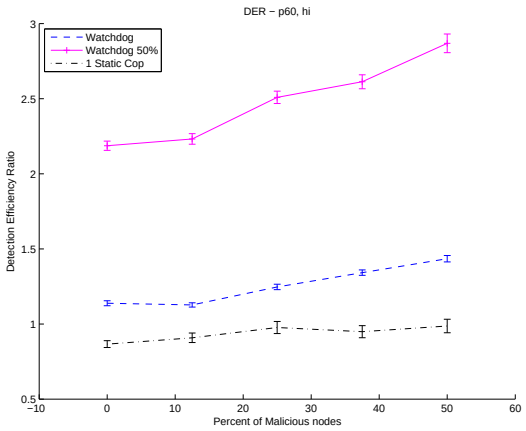


(b) High speed

Figure 5.24: DER: 16 node network - Pause time = 0 sec. - 500m × 500m



(a) Low speed



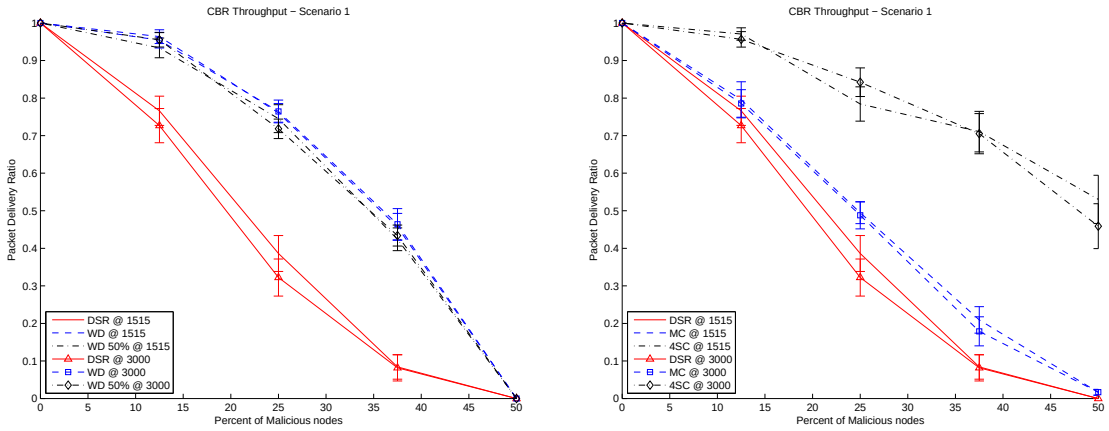
(b) High speed

Figure 5.25: DER: 16 node network - Pause time = 60 sec. - 500m × 500m

30 packet threshold and 0 second time-out, represented as "3000". Since the packet sending rate is 1 packet per second and it takes 15 seconds to send 15 packets, both schemes should have similar detection times. When the first setting is considered, the total detection time is 30 seconds (from the analysis) to detect a malicious node, which is similar to the second setting. Note that this study only focuses on PDR performance to show the effect.

Results

Figure 5.26 shows the PDRs for both settings. It is expected to have a similar PDR performance for both watchdog and cop mechanisms. The results confirm that there is no statistical difference between these 2 settings for a static ad hoc network.



(a) Watchdog mechanisms (b) Cop mechanisms

Figure 5.26: PDR: 16 node static network - Effect of threshold and time-out setting

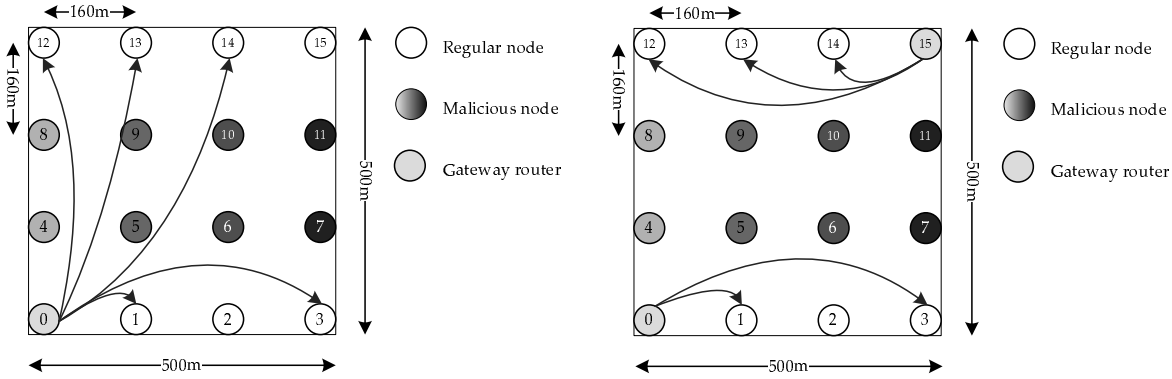
5.4 WIRELESS MESH NETWORK

Another wireless network under study here is a Wireless Mesh Network (WMN), which provides a low-cost Internet service in the area where wired-infrastructure is not available or limited. In this work, a Wireless Mesh Network is simulated with one and two gateway routers to connect to the Internet. All nodes are fixed in a grid-like topology and all connections from the Internet pass through a designated gateway. The performance metric for this study is different from the previous studies because each connection throughput is used

instead of the overall network PDR. The performance for each connection in WMN networks is evaluated (in terms of throughput) with the watchdog and cop mechanisms. This also studies the effect of packet dropping attacks on each connection. A motivation for a node to be selfish is that it tries to save its bandwidth, not its energy, since all routers are static and have potential access to power sources.

5.4.1 16-node network in 500m × 500m area

Two network scenarios are simulated with worst case and a random case scenarios as shown in Figure 5.27 and 5.28 respectively. The number of gateway routers is studied by having the same number of connections but different numbers of gateways. One and two gateway routers are considered here. 5 connections are simulated.

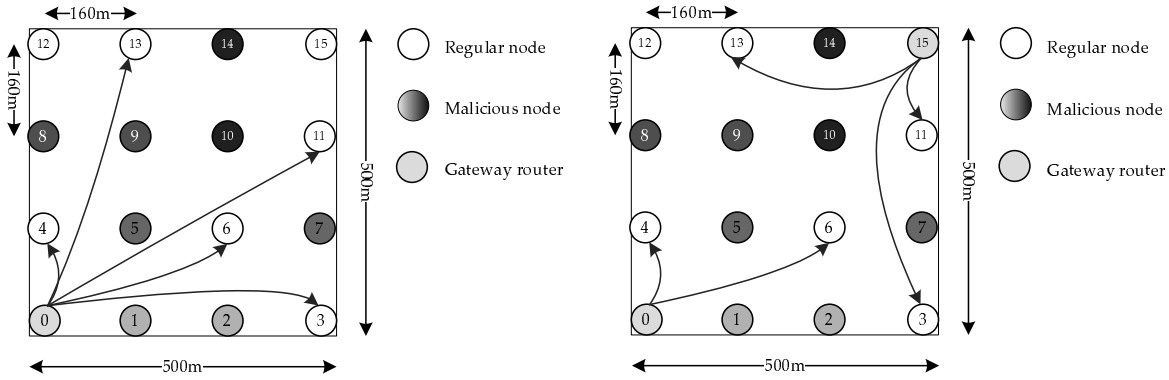


(a) 1 gateway (b) 2 gateways

Figure 5.27: 16 node WMN network - scenario 1 (worst case) - 500m × 500m

Results

Figure 5.29 and 5.30 show the throughput from the worst case and random case scenarios where 4 malicious nodes (25% of nodes) are active. From the results, the throughput is improved when one more gateway router (Node 15) is added because there are more numbers of possible paths from a source to a destination which do not include a malicious node in the path. The results also show that the watchdog mechanisms outperform the mobile cop mechanism because the watchdog mechanisms detect a malicious node faster than the cop mechanism. The throughputs from both watchdog mechanisms are not different. However,



(a) 1 gateway (b) 2 gateways

Figure 5.28: 16 node WMN network - scenario 2 (random case) - 500m × 500m

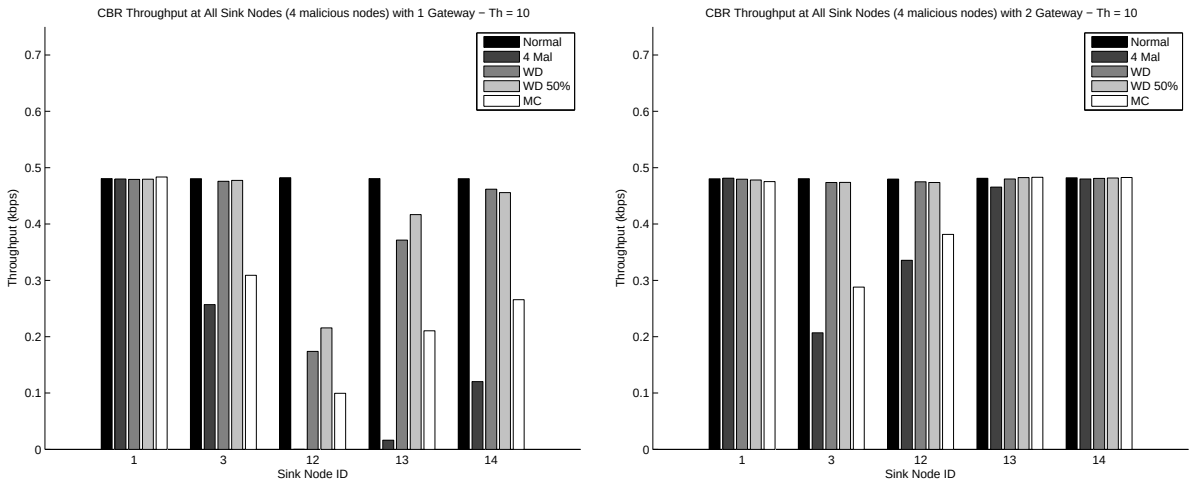
the mobile cop mechanism improves the throughput when malicious nodes are in the network, comparing to DSR itself.

5.4.2 Effect of the size of area

In this study, the parameter settings are similar to the previous study except the simulated area and node locations. The simulated area is changed from 500m × 500m area to 1000m × 1000m area. Figure 5.31 shows the 16 node network in the worst case scenario with 1 gateway. In the case of the a mobile cop mechanism, the cop's mobility path is a square shape, similar to the path in the 16 node network in a 1000m × 1000m area and a static ad hoc network (shown in Figure 5.10).

Results

Figure 5.32 and 5.33 show the throughput from the worst case scenario with 1 and 2 gateway routers. In a larger network, the throughputs with detection mechanisms are higher than the throughputs in a smaller network and the results show similar trends to the results in a static ad hoc network. They confirm that watchdog and cop mechanisms improve the throughput of the WMN network and watchdog mechanisms outperform cop mechanism. However, cop mechanism improves the throughput compare to the situation when no detection mechanism is built in the network. The location of a malicious node



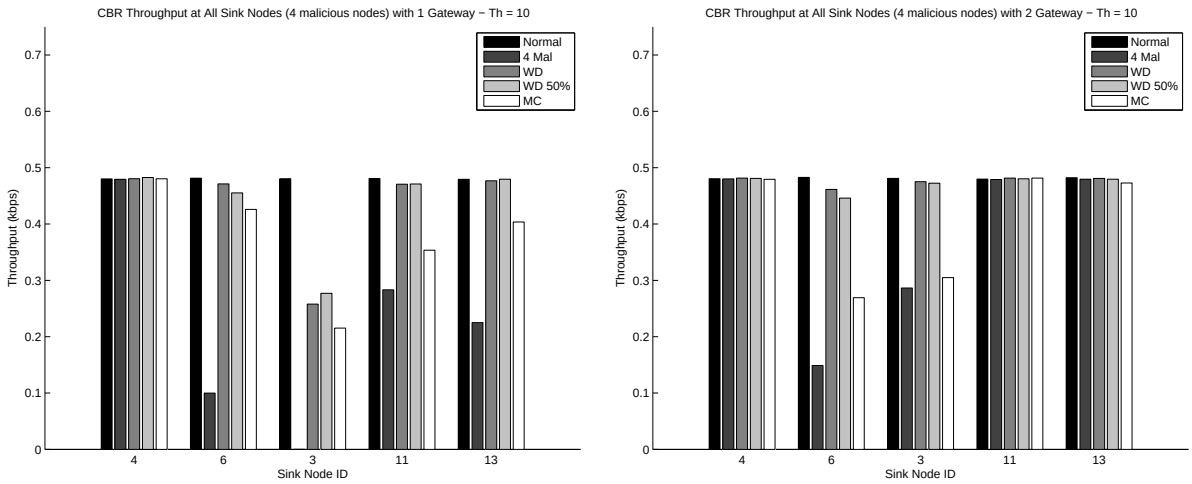
(a) 1 gateway (b) 2 gateways

Figure 5.29: Throughput: 16 node network - (worst case) - 500m × 500m

has an effect on the attack. If it is near a source or destination, the throughput drops significantly. The number of gateway routers helps reducing the packet dropping attack effectiveness by distributing connections to more gateways to avoid malicious nodes.

5.5 STUDY OF BENIGN DROPPED PACKETS

In this study, the problem is to cope with a wireless channel error or light congestion in a network, along with the packet dropping attack. How to distinguish this attack from other errors is a big question. A threshold based detection is not suitable for this type of problem because it only counts the number of non-forwarding packets regardless of current network conditions. In general, packets are dropped because of a malicious node, a wireless channel error or a network congestion. A ratio-based detection may provide a better detection against this problem as discussed below. The detection scheme is to monitor a flow-in and a flow-out of its neighbor nodes, adapted from a flow conservation in graph theory [70]. Flow-in and flow-out means the number of received packets and sent packets of a monitored node. In an



(a) 1 gateway (b) 2 gateways

Figure 5.30: Throughput: 16 node network - (random case) - 500m × 500m

ideal condition, flow-in is equal to flow-out in an error-free and congestion-free network with no malicious nodes.

In an ideal condition,

$$\sum(flow - in) = \sum(flow - out)$$

In practice, wireless channels are not reliable and cause errors at a receiver. In addition, ad hoc networks could have a congestion if the load is higher than a transmission link capacity. Therefore, flow-in is usually higher than flow-out.

In a practical condition,

$$\sum(flow - in) > \sum(flow - out)$$

When a malicious node does not drop all data packets but it helps forwarding some packets, a tolerance percentage (alpha, α) has to be set to tolerate dropped packets caused by channel errors or congestion, but not a malicious node. For this ratio detection scheme to perform properly, a malicious node must drop packets with a higher dropping percentage (delta, δ) than the tolerance percentage. In the ratio-based detection scheme, each node

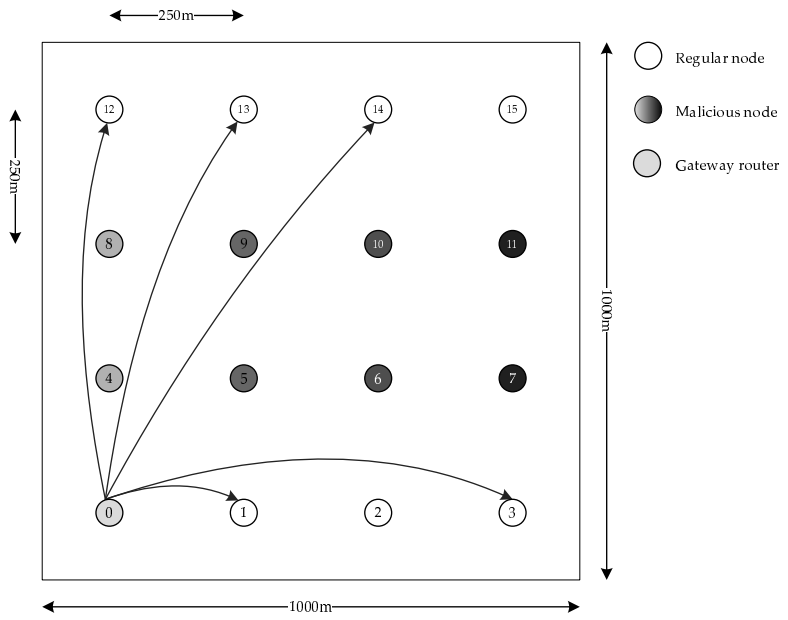


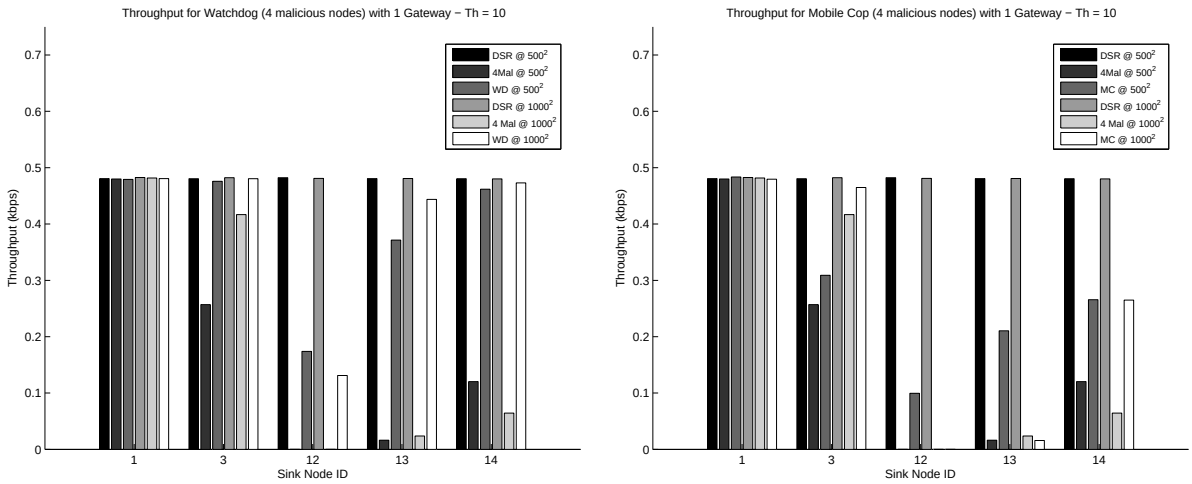
Figure 5.31: 16 node WMN with 1 gateway - scenario 1 (worst case) - 1000m × 1000m

monitors its next hop neighbor in forwarding data packets locally and it does not get information from other nodes. In order to detect a malicious node, the following condition here has to be met:

$$if \left[\frac{flow-out}{flow-in} \;<\; (1-\alpha)\right] \rightarrow A\ malicious\ node\ is\ detected$$

Given that, $\delta > \alpha$ for a detection correctness.

For this detection scheme, the flow-in and flow-out information are observed and the time to collect this information is crucial. At each time period, a detecting node collects information and determines the ratio. If the ratio is less than $(1-\alpha)$, it is assumed that a malicious node is detected. If not, it will wait for another time to check the node again. This detection condition is done periodically. Therefore, two parameters for this ratio-based detection are the tolerance percentage (α) and the periodic check time. The tolerance percentage is a crucial parameter since the dropping percentage is not available in practice. Therefore, this mechanism does not work when delta is less than alpha. In addition, if a



(a) Watchdog (b) Mobile Cop

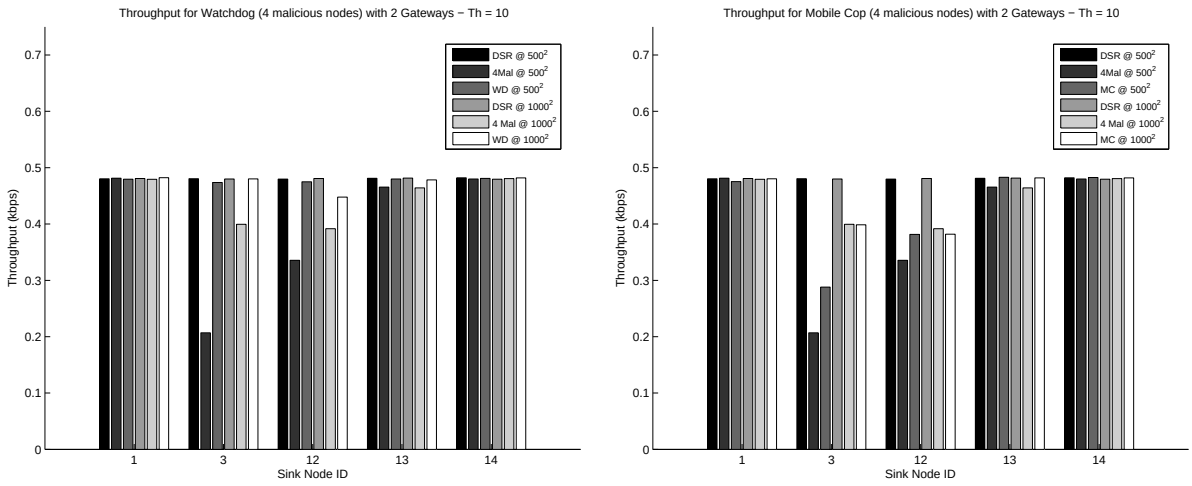
Figure 5.32: Throughput: 16 node network - (worst case) with 1 Gateway

network is heavily congested or the wireless channel is jammed with a jamming attack, this scheme will not able to perform well for detecting a malicious node.

5.5.1 Simulation results

The simulation studies are similar to the ones previously described for both static and mobile ad hoc networks. The first result is from a static ad hoc network in a worst case scenario to compare PDR performance with different dropping percentages. For a mobile ad hoc network, the results give some interesting findings with the ratio based detection mechanism.

5.5.1.1 Static ad hoc network The worst case static ad hoc network scenario is simulated with 16 nodes in a 500m × 500m area. The detection parameters are 20% tolerance percentage and the periodic check-time is 30 seconds. The dropping percentages are 30% and 70%. Figure 5.34 shows the effect of this new detection scheme. The watchdog and 4 static cop mechanisms improve the PDR performance but mobile cop mechanism does not. This is because the mobile cop does not have complete information in order to detect a malicious

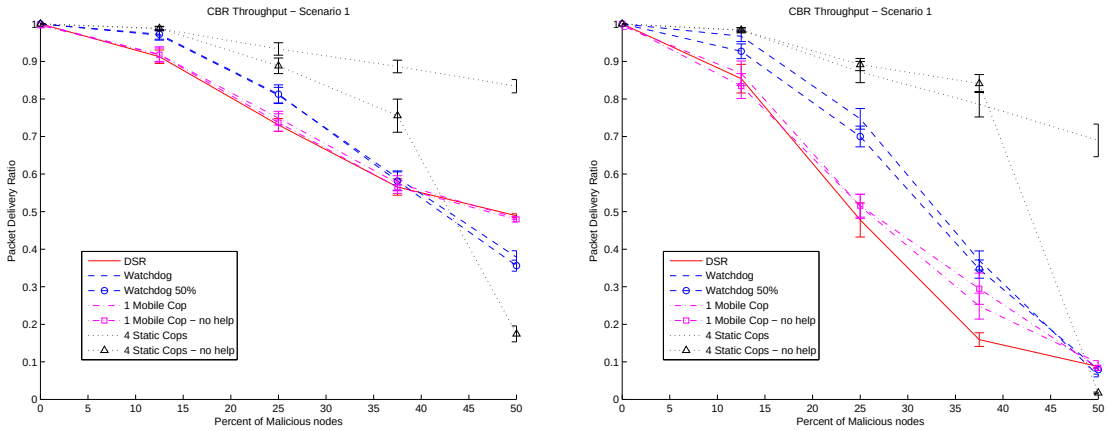


(a) Watchdog (b) Mobile Cop

Figure 5.33: Throughput: 16 node network - (worst case) with 2 Gateways

node correctly.

5.5.1.2 Mobile ad hoc network The simulation settings are similar to the previous study on the mobile ad hoc network. A 16 node network is simulated in 500m × 500m simulation area but the detection approach is changed from a threshold based to a ratio based detection mechanism. The detection parameters are 20% tolerant percentage and 30 second periodic check time. The dropping percentage is set to 70%. The results are shown in Figure 5.35 and 5.36. With 70% dropping percentage, the PDRs with a regular DSR and 50% of the nodes being malicious nodes are more than 80% for all cases. When considering the watchdog mechanism, the PDRs are worse than a regular DSR and a static cop because the topology is frequently changed and when a periodic check time is due, a next hop node is moving away such that the collected information is incomplete and the ratio is less than the tolerance percentage. Therefore, a false alarm is sent from a detecting node. It is important to note that the ratio based approach can generate a false alarm easily if nodes are mobile and local observation information is not enough to decide whether a node is a malicious node or not. A static cop mechanism slightly helps in improving the PDRs for some cases since



(a) $\alpha = 0.2$, $\delta = 0.3$ (b) $\alpha = 0.2$, $\delta = 0.7$

Figure 5.34: PDR: 16 node static network - Study of benign dropped packet

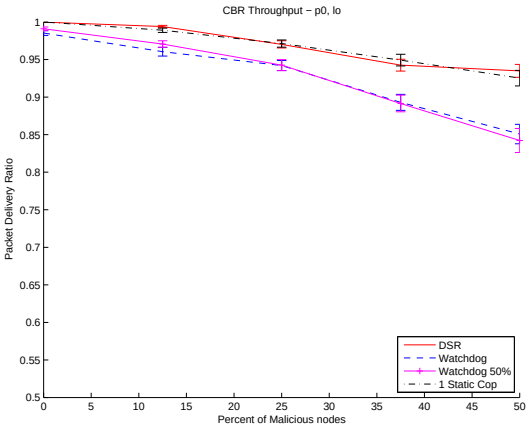
it is fixed and overhears most of the communications within the network.

5.6 STUDY OF DIFFERENT TRANSMISSION RANGES

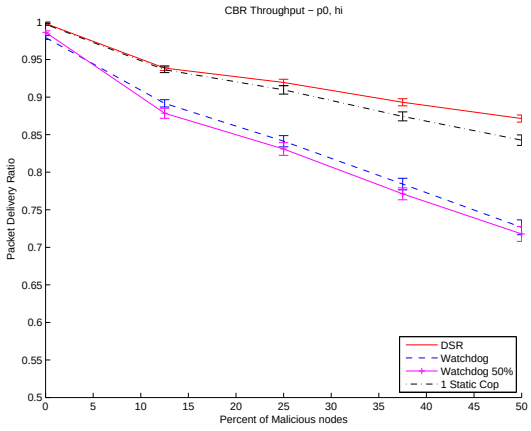
The transmission range of an ad hoc node can be changed by adjusting the transmission power. A higher power is needed in order to increase a transmission range. The question is whether the transmission range has an effect on the attack itself and the detection mechanisms or not. From the previous studies, the transmission range is set to 250 meters by default for all nodes in the network. In this study, the transmission range is increased to 500 meters in order to study its effects to both packet dropping attack and detection mechanism performances. The simulation set-up is similar to the previous descriptions except for the transmission range in a worst case static network scenario in a 500m $\times$ 500m area.

5.6.1 Simulation results

Figure 5.37 shows the throughput from two different transmission ranges, 250 m. and 500 m in a small area. In a 500 m. transmission range scenario, all nodes are able to

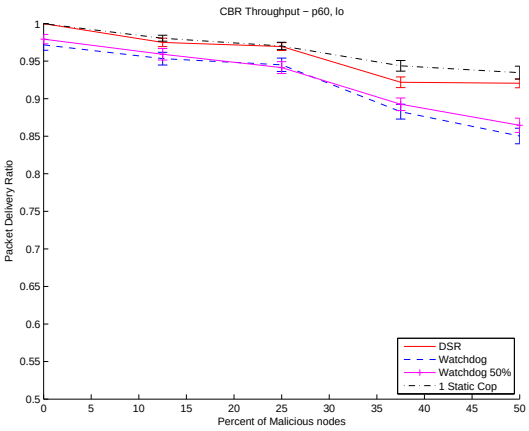


(a) Low speed

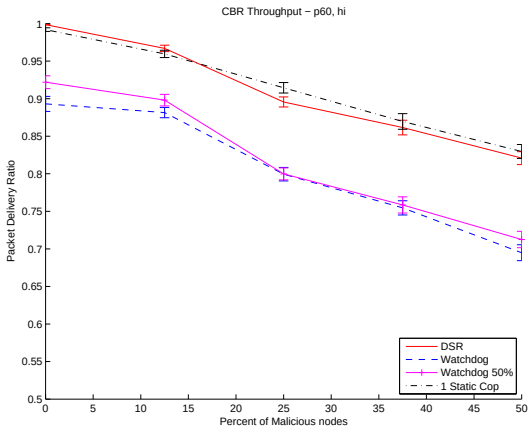


(b) High speed

Figure 5.35: PDR: 16 node network - 0 sec. pause time. - $\alpha = 0.2, \delta = 0.7$ - 500m $\times$ 500m



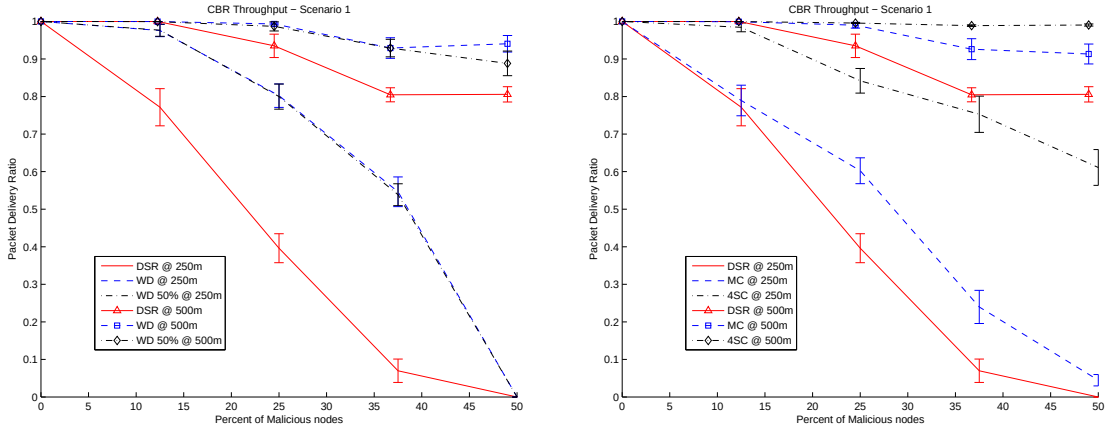
(a) Low speed



(b) High speed

Figure 5.36: PDR: 16 node network - 60 sec. pause time - $\alpha = 0.2, \delta = 0.7$ - 500m $\times$ 500m

send and receive packets within one hop away while the nodes with 250 m. transmission range cannot. As expected, the higher transmission range provides better performance since nodes do not rely on their neighbors to forward packets. Therefore, the effect from the packet dropping attack is lessened. Both watchdog and cop mechanisms improve the PDR significantly because a detecting node is always in a transmission range of a sender and a forwarder. In addition, a detecting node can send an alarm message within one hop away and more alternate paths are available to reach a destination.



(a) Watchdog mechanisms (b) Cop mechanisms

Figure 5.37: PDR: 16 node static network - Study of transmission range

Remarks: The transmission range is an important issue for detection performance. We assume that all nodes have the same transmission range. When the transmission range is small, a mobile cop needs to move for a longer time to detection a malicious node and the number of static cops has to be increased to cover all nodes in the area. This is not an issue for watchdog since all watchdog nodes have to be in the transmission range of each other in order to successfully communicate with each other and the detection can be thus performed.

5.7 SUMMARY

In this chapter, the performance study of both watchdog and cop mechanisms are presented. In most of the cases, watchdog mechanism outperforms cop mechanism in term of

PDR performance because each node detects its own neighbors directly with shorter detection time. The 50% watchdog mechanism gives equal or a little less PDR performance than 100% watchdog mechanism and it saves more energy by reducing the monitoring for the detection function. A mobile cop mechanism needs to monitor its neighbors when it is in the transmission range of a sender and a receiver. The time to detect a malicious node with the mobile cop mechanism is as equal as or more than the time for watchdog mechanism. A static cop can perform as well as watchdog in a small network but not in a large network. In MANETs with high mobility speed, the detection mechanisms do not perform well to detect malicious nodes. However, the mechanisms give good performance when there is low mobility since the topology is gradually changing. In summary, both detection mechanisms help detecting a malicious node for packet dropping attack and improving the throughput performance.

The wireless effects are studied with the packet dropping attack and its mitigation is considered. The detection mechanisms for benign dropped packets have to be changed from a threshold-based to a ratio-based approach such that it can correctly detect a malicious node. However for this detection, composite information has to be collected in order to make a correct decision to detect a malicious node. The energy efficient schemes, namely mobile cop and 50% watchdog mechanisms, cannot be used with ratio-based detection. Finally, a study of the effect of transmission range shows that when the transmission range is increased, the packet dropping attack effectiveness is significantly reduced and detection mechanisms also improve in performance.

6.0 A DESIGN GUIDELINE FOR PACKET DROPPING ATTACK DETECTION

In the previous chapters, analysis and simulations demonstrated the performance of different detection mechanisms. Each detection mechanism was compared with others to determine where advantages and disadvantages for different network scenarios occur. With this information, this chapter will discuss design guidelines for detecting packet dropping attacks for a network designer. Both a recommended detection mechanism and detection parameters are the results that can be used when the network is deployed.

6.1 AVAILABLE DETECTION MECHANISMS AND THEIR VARIATIONS

The main goal for this study is to detect packet dropping attacks, which is easy to deploy under any network conditions. Four detection mechanisms are considered here.

- Watchdog mechanism – all nodes perform monitoring and detection 100% of the time to detect malicious nodes.
- Watchdog with X % detection – nodes perform monitoring and detection x% of the time
- Mobile cop mechanism – a few moving nodes are in the network for detecting malicious nodes
- Static cop mechanism – a few static nodes are in the network to cover the area for detecting a malicious node

For each mechanism, two approaches are possible to be implemented as follows:

- Threshold-based detection - a detecting node only counts the number of non-forwarded packets
- Ratio-based detection - a detecting node counts both forwarded and non-forwarded packets

6.2 SUMMARY OF SIMULATION RESULTS

From our simulation study, we can summarize the results as follows:

- Watchdog performs best in most scenarios.
- Watchdog 50% is as good as watchdog but it uses less energy for detection, especially in static ad hoc networks.
- Cop performs well in a small network. The number of mobile cop is smaller than the number of static cops but it has poor detection time and it doesn't work well in MANETs.

6.3 DESIGNER REQUIREMENTS

A designer should have some requirements for the network to operate in terms of the expected performance. The parameter setting will be tailored to meet the design requirements. Here is the information that needs to be considered.

- Load in the network - sending rate from a source to a destination
- Detection time - time between a detector node starting to detect a malicious node and sending the alarm message
- % drop tolerance for noisy wireless channel or congested network (optional)

The requirements may not be fulfilled if a designer wants the fastest detection time and low false alarm rates. This is a trade-off to be considered.

6.4 RECOMMENDED DETECTION MECHANISM AND PARAMETER SETTING

The outputs from the system conditions and designer requirements are as follows:

- Recommended detection mechanism
- Parameter settings

6.4.1 Recommended detection mechanism

We propose the use of incremental deployment of detection schemes. The detection mechanism has to be changed depending on the malicious node density.

Table 6.1: Recommended detection approaches

Malicious node density	Recommend detection mechanism(s)
Low	Mobile Cop
Medium	Static cop or 50% watchdog
High	100% Watchdog

When a network is set up without any expectation of any attack, a mobile cop can be used to guard against unexpected attacks on the network. However, the number of malicious nodes may be increasing such that a mobile cop cannot effectively detect malicious nodes. Therefore, static cops or watchdog 50% can be deployed into the network. Static cops should be placed to cover all nodes in the network and that increases the number of cop nodes significantly for a large area network. However, watchdog 50% can be implemented or activated even if the network is very large such that static cops are not suitable. When the number of malicious nodes is high, both mechanisms may not work well and watchdog should be implemented or activated in all nodes.

With the incremental detection scheme, energy for each node can be saved when the number of malicious nodes is small to medium. Energy cannot be saved when the number of malicious nodes is high because all nodes must actively detect malicious nodes in the network. This is an intuitive result - extra security forces are needed only when the situation demands it, and then the cost will inevitably be high.

6.4.2 Parameter settings

From the designer requirements, the detection time for threshold-based watchdog and static cop mechanisms is calculated from the equation below:

$$T_{det} = \frac{TH}{R} + TO \tag{6.1}$$

This equation is an estimated detection time for a static ad hoc network. For MANETs, this value is the minimum detection time.

The detection time for mobile cops or x% watchdog detection mechanisms is calculated as follows:

$$T_{det} = \begin{cases} \frac{TH}{R} + TO & if \quad T_{DD} \geq \frac{TH}{R} + TO \\ \frac{TH}{R} + TO + nT_{IDD} & if \quad T_{DD} < \frac{TH}{R} + TO \end{cases} \tag{6.2}$$

This equation is for a static ad hoc network with a mobile cop and the location of a malicious node is known. Otherwise, it is difficult to estimate the detection time since it depends on detection duration and inter-detection duration.

The detection time (T_{det})(in seconds) and the data rate (R) (in packets per second) are specified from the requirements. Therefore, the threshold and time-out settings are easily calculated. It is important to note that the detection time in this chapter is different from detection time in Chapter 4 because in the analysis, we knew exactly when to start the detection and when to stop the detection such that the throughput can be calculated. In practice, we don't know this information and the only thing we know is the general parameter settings from the requirements. Therefore, the detection time here is the time to detect a malicious node after a suspicious activity is noted.

In the case of the ratio based mechanism, parameters of note are the drop tolerance percentage and the detection time interval. The specified detection time can be used as a guideline for the detection time interval.

However, the requirements are used as a guideline to set appropriate values. If the required detection time is too short, false alarms can easily be triggered and this causes throughput reduction. If it is too high, a malicious node is detected slowly and the PDR is

also dropped. Thus these are only guidelines towards a better design of a packet dropping attack mitigation system. More specific parameter settings depend on the designer who should evaluate the appropriate values.

6.5 MISCELLANEOUS REMARKS

6.5.1 Node collaboration

In this dissertation, we have not assumed any information exchange between cops or nodes that are not part of the route being considered by Watchdog. Cops independently acquire knowledge of suspicious nodes. However, when cops collaborate with each other, the detection information can be shared to improve detection times. This causes increased overhead for cop mechanism, especially if cops are sparsely deployed and have to communicate using other nodes in the network. It is possible to use regular nodes to relay the communication between cops but if regular nodes cannot be trusted, the information can be modified or dropped. Another method is to use a high transmission range or use directional antenna for cops to communicate with each other but it consumes more energy. This method is possible since cops could have more energy than regular nodes as per our assumption of heterogeneity.

In contrast, watchdog can have distributed collaboration but if watchdog nodes are not trustable, false information can cause the false accusation of a good node and PDR can drop. Trust is an important issue for watchdog mechanism with collaboration.

When we analyzed the network, we did not consider the collaboration between cops or watchdog nodes. However, if the collaboration is considered in static ad hoc networks, it does not increase the throughput for watchdog because the topology of the networks is not changed. In the case of static cops in a large static network, collaboration between them can increase the throughput since all static cops cannot be in the ranges of all senders and forwarders. After collaboration, cops have more information to detect malicious nodes in the network (e.g., static cop 1 hears a sender, but static cop 2 does not hear the forwarder

– here static cop 1 can only hear the sender and static cop 2 can only hear the forwarder).

When we consider collaboration in MANETs, throughput of the networks can be decreased because the topology is changing most of the time such that a false alarm can easily occur if nodes do not collaborate. In a high speed mobile network, collaboration might not be helpful since the network topology is changing rapidly such that the information exchanges between watchdog nodes are not updated rapidly enough to detect a malicious node. For low speed mobile networks, collaboration can be useful for watchdog nodes in detecting a malicious node.

With the cop mechanism, collaboration between cops is crucial for correct detection in MANETs because cops may not be able to cover all the nodes and the topology is changing. Therefore, if the correct detection information is shared and updated, watchdog and cop mechanisms can improve the throughput of the network.

6.5.2 Network lifetime

When we consider Watchdog and Cop mechanisms in term of energy, the watchdog function has to be implemented in all nodes and each node needs to monitor its neighbors if it is chosen as an intermediate node. Energy will be consumed in the receiving mode for detection function. In addition, the nodes need to read and write their memory in buffers for detection functions and these consume energy for each detecting node.

If we consider the network lifetime as the time at which the first node dies [76], watchdog nodes obviously have shorter life time than regular nodes in cop mechanism since watchdog nodes need to overhear neighbors' communications and buffer packets for detection function such that they consume more energy than regular nodes in cop mechanism. However, we have not explicitly used models for determining lifetime here.

6.5.3 Node modification

When a detection algorithm is changed, a detecting node has to be modified to update the algorithm. In the watchdog mechanism, the update has to be done one by one for each node and if more nodes are in the network, the modification time will be longer. However, when

the algorithm is changed for the cop mechanism, only cop nodes have to be modified. The modification time for cop mechanism is considerably less than that for watchdog mechanism. Here, we are referring to mechanisms that may adaptively morph from simple thresholds or ratios to more advanced pattern recognition schemes based on response to intelligent or sophisticated attacks.

6.6 LIMITATIONS OF THIS WORK

Here is a list of limitations of this work.

1. Watchdog mechanism suffers from several drawbacks, i.e., ambiguous collision, receiver collisions, limited transmission power, false misbehavior, collusion, and partial dropping. From these drawbacks, watchdog can allow the false alarm to occur, which will cause network degradation. We don't take these drawbacks into account for this study.
2. Cop mechanism is similar to watchdog mechanism and therefore it has the same drawbacks as watchdog mechanism.
3. We are interested in the performance of the detection mechanisms, but not the causes of misbehavior.
4. We do not consider the dynamic behavior of a malicious node, which tries to fake as a good node and later on attacks the network and vice versa.
5. We do not fully study the detection duration and inter-detection duration for cop mechanism since it depends on each network scenario.

7.0 CONCLUSIONS AND FUTURE WORKS

7.1 CONCLUSIONS

This dissertation studied detection performance of mechanisms that aim to mitigate the impact of packet dropping attacks. The contribution of this work is listed as follows.

- A new detection mechanism is proposed and studied for addressing the detection of packet dropping attacks by using a few special nodes (cops) to opportunistically detect a malicious node in a network. Even though, this paradigm is not as good as the original watchdog mechanism in terms of detection time, it can help improve the performance of the network with fewer nodes performing monitoring and detection functions. The cop nodes can be either static or mobile depending on the network designer.
- An analysis for a static ad hoc network that uses a probability tree to find the weighted average PDR of the network with different detection mechanisms and numbers of malicious nodes in the network is presented. The analysis shows the real challenge of this study in that the throughput performance depends on the number of paths in the network and the locations of malicious nodes. The detection time for threshold-based detection depends on the data rate, the threshold setting and an optional time-out setting. With cop nodes, the detection time also depends on the detection range (that a cop is in the range of a sender and a suspicious node).
- Simulation results show the effects of network sizes, numbers of nodes and mobility speeds to help understand the impact of the packet dropping attack and its mitigation.
- Wireless effects impact the performance of not only the packet dropping attack but also the detection mechanisms. When the overall transmission range is increased, the effect

of the attack is reduced and the detection mechanisms can detect a malicious node easier since it can overhear most of the communications in the network.

- This study shows that a ratio-based detection mechanism can help improving the PDR performance when all information from its neighbors is collected completely. Otherwise, false alarms can easily occur. A ratio-based detection does not work well especially in mobile ad hoc networks when the detection information is collected locally and is incomplete. More nodes need to collect the information and send this to a collector to make a better decision on whether a suspicious node is actually malicious.

7.2 FUTURE WORK

In this dissertation, a simple cop mechanism was studied but it can be improved in several ways. Here are the lists of possibilities to enhance this work further.

- Study the mobile cop path to better detect a malicious node as in opportunistic networking by implementing and locating a cop node in an area that is likely to have a high malicious node density.
- Study an on-call cop scheme where a cop node is at a station and gets a message from a regular node to investigate suspicious activity. After the detection, a cop node returns to the station. This works well in a static ad hoc network because the network topology is not changed much and the cop can move between the source and the destination pairs.
- Apply the throughput analysis for other types of attack, such as a wormhole attack and include other detection mechanisms. Extend the analysis for incentive based schemes as well, to see what kind of incentives make the best sense.
- Study the cop paradigm with other types of attacks, such as wormhole attacks, by implementing new algorithms for detecting such attacks in cop nodes. Note that characteristics of attacks have to be known in order to correctly implement the detection algorithm. However, this paradigm cannot be applied directly to all types of attacks since it only works on an attack that does not require collaboration for detection. In this work cop

nodes act independently. It may be possible to include cooperation between multiple cops as well.

BIBLIOGRAPHY

[1] M. Gerla, *Ad Hoc Networks : Technologies and Protocols*. Springer, 2005, ch. 1, pp. 1–22.

[2] J. Douceur, "The sybil attack," 2002. [Online]. Available: citeseer.ist.psu.edu/douceur02sybil.html

[3] Y.-C. Hu, A. Perrig, and D. Johnson, "Packet leashes: A defense against wormhole attacks in wireless networks," in *the 22nd IEEE Computer and Communications Societies (INFOCOM'03)*, 2003.

[4] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Rushing attacks and defense in wireless ad hoc network routing protocols," in *Proceedings of the 2003 ACM Workshop on Wireless Security (WiSe 2003)*. ACM, September 2003, pp. 30–40.

[5] Y. an Huang and W. Lee, "Attack analysis and detection for ad hoc routing protocols," in *Proceedings of The 7th International Symposium on Recent Advances in Intrusion Detection (RAID 2004)*, September 2004.

[6] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens, "Mitigating byzantine attacks in ad hoc wireless networks," Center for Networking and Distributed Systems , Johns Hopkins University," Technical Report, 2004.

[7] D. Spiewak, T. Engel, and V. Fusenig, "Towards a threat model for mobile ad-hoc networks," in *ISP'06: Proceedings of the 5th WSEAS International Conference on Information Security and Privacy*. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2006, pp. 35–40.

[8] P. Michiardi and R. Molva, "Simulation-based analysis of security exposures in mobile ad hoc networks," in *EW'2002, European wireless 2002 - February 25-28, 2002, Firenze, Italy*, Feb 2002.

[9] F. Kargl, A. Klenk, S. Schlott, and M. Weber, "Advanced detection of selfish or malicious nodes in ad hoc networks," August 2004.

[10] S. Marti, T. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *The 6th ACM International Conference on Mobile Computing and Networking*, 2000.

[11] P. Michiardi and R. Molva, "CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *CMS'2002, Communication and Multimedia Security 2002 Conference, September 26-27, 2002, Portoroz, Slovenia / Also published in the book : Advanced Communications and Multimedia Security /Borka Jerman-Blazic & Tomaz Klobucar, editors, Kluwer Academic Publishers, ISBN 1-4020-7206-6, August 2002 , 320 pp*, August 2002.

[12] S. Buchegger and J.-Y. L. Boudec, "Performance analysis of the confidant protocol (cooperation of nodes: Fairness in dynamic ad-hoc networks)," in *MOBIHOC'02*, 2002.

[13] J. Mundinger and J.-Y. L. Boudec, "Analysis of a reputation system for mobile ad-hoc networks with liars," *wiopt*, vol. 00, pp. 41–46, 2005.

[14] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks," in *IEEE Communications Magazine*, ser. 11, vol. 44, November 2006.

[15] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," *Computer Networks Journal (Elsevier)*, vol. 47, no. 6, pp. 445–487, March 2005.

[16] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers," in *SIGCOMM 94 Conference on Communications Architectures, Protocols and Applications*, August 1994, pp. 234–244.

[17] D. B. Johnson, D. A. Maltz, and Y.-C. Hu, "The dynamic source routing protocol for mobile ad hoc networks (dsr)," Published Online, IETF MANET Working Group, INTERNET-DRAFT, July 2004, expiration: January 2005. [Online]. Available: http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-10.txt

[18] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, "Ad hoc on-demand distance vector (aodv) routing," Internet-Drafts, February 2003.

[19] L. Buttyan and J.-P. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," in *Mobile Networks and Applications*, 2003, pp. 579–592.

[20] H. Miranda and L. Rodrigues, "Friends and foes: Preventing selfishness in open mobile ad hoc networks," in *ICDCSW'03*, 2003.

[21] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks," in *Proceedings of IEEE INFOCOM '03*, San Francesco, CA, April 2003.

[22] B. Raghavan and A. C. Snoeren, "Priority forwarding in ad hoc networks with self-interested parties," in *Workshop on Economics of Peer to Peer*, June 2003.

[23] F. K. J. Crowcroft, R. Gibbens and S. Ostring, "Modelling incentives for collaboration in mobile ad hoc networks," in *Proceedings of Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks , Ad Hoc, and Wireless Networks (WiOpt'03)*, France, March 2003.

[24] A. Mok, E. C. Bina Mistry, and B. Li, "Fair: Fee arbitrated incentive architecture in wireless ad hoc networks," in *10th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS'04)*, 2004, p. 38.

[25] Y. Zhang, W. Lou, and Y. Fang, "Sip: a secure incentive protocol against selfishness in mobile ad hoc networks," in *IEEE Wireless Communications and Networking Conference (WCNC'04)*, March 2004.

[26] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," in *Journal of the Operational Research Society*, vol. 49, 1998. [Online]. Available: citeseer.ist.psu.edu/kelly98rate.html

[27] M. Hauspie and I. Simplot-Ryl, "Cooperation in ad hoc networks: Enhancing the virtual currency based models," in *Proceedings of the 1st ACM International Conference on Integrated Internet Ad hoc and Sensor Networks (InterSense 2006)*, Nice, France, May 2006.

[28] P. N. V. Srinivasan and R. R. Rao, "Cooperation in ad hoc networks," in *Proceedings of IEEE INFOCOM '03*, San Francesco, CA, April 2003.

[29] S. Eidenbenz, V. S. A. Kumar, and S. Zust, "Equilibria in topology control games for ad hoc networks," in *DialM-POMC 2003*, September 2003.

[30] A. Urpi, M. A. Bonuccelli, and S. Giordano, "Modelling cooperation in mobile ad hoc networks: a formal description of selfishness," in *Proceedings of Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks , Ad Hoc, and Wireless Networks (WiOpt'03)*, France, March 2003.

[31] L. Anderegg and S. Eidenbenz, "Ad hoc-vcg: A truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents," in *MobiCom'03*, 2003.

[32] O. Ileri, S.-C. Mau, and N. Mandayam, "Pricing for enabling forwarding in self-configuring ad hoc networks," in *IEEE Wireless Communications and Networking Conference (WCNC'04)*, March 2004.

[33] J. Cai and U. Pooch, "Allocate fair payoff for cooperation in wireless ad hoc networks using shapley value," in *Preceedings of the 1st IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS04)*, 2004.

[34] ——, "Play alone or together - thruthful and efficient routing in wireless ad hoc networks with selfish nodes," in *Preceedings of the 18th IEEE International Parallel and Distributed Processing Symposium (IPDPS'04)*, Fort Lauderdale, FL, 2004.

[35] S. Zhong, L. E. Li, Y. G. Liu, and Y. R. Yang, "On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks—an integrated approach using game theoretical and cryptographic techniques," in *Proceedings of the Eleventh ACM Annual International Conference on Mobile Computing and Networking (Mobicom)*, Cologne, Germany, August 2005.

[36] S. Eidenbenz and G. Resta, "Commit: A sender-centric trouthful and energy-efficient routing protocol for ad hoc networks with selfish nodes," in *Preceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05)*, 2005.

[37] P. Marbach and Y. Qiu, "Cooperation in wireless ad hoc networks: a market-based approach," *IEEE/ACM Trans. Netw.*, vol. 13, no. 6, pp. 1325–1338, 2005.

[38] S. Bandyopadhyay and S. Bandyopadhyay, "A game-theoretic analysis on the conditions of cooperation in a wireless ad hoc network," in *Proceedings of Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'05)*, 2005, pp. 54–58.

[39] E. Huang, J. Crowcroft, and I. Wassell, "Rethinking incentives for mobile ad hoc networks," in *Proceedings of the ACM SIGCOMM workshop on Practice and theory of incentives in networked systems*, Portland, USA, 2004, pp. 191–196.

[40] S. Buchegger and J.-Y. L. Boudec, "Self-policing mobile ad hoc networks by reputation systems," in *IEEE Communications Magazine*, ser. 7, vol. 43, July 2005, pp. 101–107.

[41] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. C. Chen, "A survey of energy efficient network protocols for wireless networks," *Wirel. Netw.*, vol. 7, no. 4, pp. 343–358, 2001.

[42] P. Michiardi and R. Molva, "A game theoretical approach to evaluate cooperation enforcement mechanisms in mobile ad hoc networks," in *WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, Sophia-Antipolis, France, March 2003.

[43] M. Frank, P. Martini, and M. Plaggemeier, "Cinema: Cooperation enhancement in manets," in *Proceedings of the 29th Annual IEEE International Conference on Local Computers Networks (LCN'04)*, 2004.

[44] K. Balakrishnan, J. Deng, and P. K. Varshney, "Twoack: Preventing selfishness in mobile ad hoc networks," in *IEEE Wireless Communications and Networking Conference (WCNC'05)*, 2005.

[45] K. Liu, J. Deng, P. K. Varshney, and K. Balakrishnan, "An acknowledgment-based approach for the detection of routing misbehavior in manets," *IEEE Transactions on Mobile Computing*, vol. 6, no. 5, pp. 536–550, 2007.

[46] A. Perrig, R. Canetti, D. Song, and J. Tygar, "The tesla broadcast authentication protocol," in *CryptoBytes*, 2002, pp. 2–13.

[47] Y.-H. Hu, A. Perrig, and D. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," in *The 8th Annual International Conference on Mobile Computing and Networking*, 2002.

[48] Y.-C. Hu, D. Johnson, and A. Perrig, "Secure efficient distance vector routing in mobile wireless ad hoc networks," in *the 4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, 2002.

[49] P. Papadimitratos and Z. Haas, "Secure routing for mobile ad hoc networks," in *SCS Communication Networks and Distributed Systems Modeling and Simulation Conference*, 2002.

[50] K. Sanzgiri, B. Dahill, B. Levine, C. Shields, and E. Belding-Royer, "A secure routing protocol for ad hoc networks," in *the 10th IEEE International Conference on Network Protocols (ICNP)*, November 2002.

[51] K. Sanzgiri, D. LaFlamme, , B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, "Authenticate routing for ad hoc networks," in *IEEE Journal on Selected Area in Communications*, ser. 3, vol. 23, March 2005.

[52] P. Papadimitratos and Z. Haas, "Secure link state routing for mobile ad hoc networks," in *Applications and the Internet Workshops*, 2003.

[53] M. G. Zapata and N. Asokan, "Securing ad-hoc routing protocols," in *the 2002 ACM Workshop on Wireless Security (WiSe 2002)*, September 2002, pp. 1–10.

[54] H. Yang, X. Meng, and S. Lu, "Self-organized network-layer security in mobile ad hoc networks," in *ACM Workshop on Wireless Security*, 2002.

[55] A. Patwardhan, J. Parker, A. Joshi, M. Iorga, and T. Karygiannis, "Secure routing and intrusion detection in ad hoc networks," in *3rd International Conference on Pervasive Computing and Communications*, March 2005.

[56] S. Ghazizadeh, O. Ilghami, E. Sirin, and F. Yaman, "Security-aware adaptive dynamic source routing protocol," 2002. [Online]. Available: http://www.cs.umd.edu/~shayan/

[57] F. Kargl, A. Geis, S. Schlott, and M. Weber, "Secure dynamic source routing," in *the 38th Annual Hawaii International Conference on System Sciences, 2005. HICSS'05*, January 2005.

[58] P. Prasithsangaree and P. Krishnamurthy, "On a framework for energy-efficient security protocols in wireless networks," in *Computer Communications*, ser. 17, vol. 27, November 2004, pp. 1716–1729.

[59] S. Barbara and D. Agraffe, "Scalable security schemes for ad hoc networks," in *IEEE Military Communications Conference*, 2002.

[60] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens, "An on-demand secure routing protocol resilient to byzantine failures," in *ACM Workshop on Wireless Security*, 2002.

[61] A. Patcha and A. Mishra, "Collaborative security architecture for black hole attack prevention in mobile ad hoc networks," in *IEEE Radio and Wireless Conference, 2003. RAWCON '03*, August 2003, pp. 75–78.

[62] S. R. Medidi, M. Medidi, and S. Gavini, "Detecting packet-dropping faults in mobile ad-hoc networks," in *the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 2, November 2003.

[63] M. Just, E. Kranakis, and T. Wan, "Resisting malicious packet dropping in wireless ad hoc networks using distributed probing," in *ADHOC-NOW '03*, 2003.

[64] W. Wang, Y. Lu, and B. Bhargava, "On security study of two distance vector routing protocols for mobile ad hoc networks," in *IEEE International Conference on Pervasive Computing and Communications (PerCom'03)*, March 2003, pp. 35–46.

[65] A. Patwardhan, J. Parker, A. Joshi, T. Karygiannis, and M. Iorga., "Secure routing and intrusion detection in ad hoc networks," in *Third IEEE International Conference on Pervasive Computing and Communications*, March 2005.

[66] B. J. Culpepper and H. C. Tseng, "Sinkhole intrusion indicators in dsr manets," in *Proceedings of First International Conference on Broadband Networks (BroadNets)*, October 2004, pp. 681–688.

[67] Y.-A. Huang, W. Fan, W. Lee, and P. Yu, "Cross-feature analysis for detecting ad-hoc routing anomalies," in *23rd International Conference on Distributed Computing Systems*, 2002.

[68] Y. Zhang, W. Lee, and Y.-A. Huang, "Intrusion detection techniques for mobile wireless networks," *Mobile Networks and Applications*, 2003.

[69] B. Awerbuch, R. Curtmola, D. Holmer, H. Rubens, and C. Nita-Rotaru, "On the survivability of routing protocols in ad hoc wireless networks," in *First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM)*. IEEE/CreateNet, September 2005, pp. 327–338.

[70] O. F. Gonzalez, M. Howarth, and G. Pavlou, "Detection of packet forwarding misbehavior in mobile ad hoc networks," in *Proceedings of the International Conference on Wired/Wireless Internet Communications (WWIC 2007)*, Coimbra, Portugal, May 2007.

[71] C.-Y. Tseng, P. Balasubramanyam, C. Ko, R. Limprasittiporn, J. Rowe, and K. Levitt, "A specification-based intrusion detection system for aodv," in *the 1st ACM workshop on*

Security of ad hoc and sensor networks, Conference on Computer and Communications Security, 2003, pp. 125–134.

[72] G. Vigna, S. Gwalani, K. Srinivasan, E. Belding-Royer, and R. Kemmerer, "An intrusion detection tool for aodv-based ad hoc wireless networks," in *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, Tucson, AZ, December 2004, pp. 16–27.

[73] P.-W. Yau and C. Mitchell, "Reputation methods for routing security for mobile ad hoc networks," in *SympoTIC'03*, 2003.

[74] R. Carruthers and I. Nikolaidis, "Certain limitations of reputation–based schemes in mobile environments," in *MSWiM '05: Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems.* New York, NY, USA: ACM Press, 2005, pp. 2–11.

[75] K. Fall and K. Varadhan, "The ns manual," December 2003. [Online]. Available: http://www.isi.edu/nsnam/ns/

[76] J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1, pp. 22–31 vol.1, 2000.