ABSTRACT

Title of dissertation: INTRUSION DETECTION FOR DEFENSE AT THE MAC AND ROUTING LAYERS OF WIRELESS NETWORKS

Svetlana Radosavac
Doctor of Philosophy, 2007

Dissertation directed by: Professor John S. Baras
Department of Electrical and Computer Engineering

The pervasiveness of wireless devices and the architectural organization of wireless networks in distributed communities, where no notion of trust can be assumed, are the main reasons for the growing interest in the issue of compliance to protocol rules. Nevertheless, the random nature of protocol operation together with the inherent difficulty of monitoring in the open and highly volatile wireless medium poses significant challenges. In this thesis, the problem of detection of node misbehavior at the MAC layer and impact of such behavior on two different routing protocols in the Network Layer is considered. Starting from a model where the behavior of a node is observable, we cast the problem within a min-max robust detection framework, with the objective to provide a detection rule of optimum performance for the worst-case attack in the MAC layer. With this framework we capture the uncertainty of attacks launched by intelligent adaptive attackers and concentrate on the class of attacks that are most significant in terms of incurred performance losses. Furthermore, we show that our ideas can be extended to the case where observations are hindered by interference due to concurrent transmissions and derive performance bounds of both the attacker and detection system in such scenarios. We extend the proposed framework to model collaborative attacks and quantify the impact

of such attacks on optimal detection systems by mathematical analysis and simulation. Finally, by using the principle of cross-entropy minimization, we present a general procedure for constructing an optimal attack scenario in the MAC layer under a general set of constraints that can be adapted based on specific requirements of an Intrusion Detection System (IDS).

INTRUSION DETECTION FOR DEFENSE AT THE MAC AND ROUTING LAYERS OF WIRELESS NETWORKS

by

Svetlana Radosavac

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2007

Advisory Committee:
Professor John S. Baras, Chair/Advisor
Professor Gang Qu
Professor Manoj Franklin
Professor Virgil D. Gligor
Professor V. S. Subrahmanian

Dedication

To my parents, for their unconditional love and support throughout my whole life.

Acknowledgements

I would like to thank my advisor, Professor John S. Baras for his continuous guidance and support throughout my PhD studies. I would also like to thank Dr Gang Qu, Dr Manoj Franklin, Dr Virgil Gligor and Dr V. S. Subrahmanian for agreeing to serve on my committee. In particular, I am thankful to Professor Virgil Gligor for constructive comments on my M.S. thesis that lead me to think about several problems in more practical manner and formulate the problems in my PhD thesis in a more clear way. I am very grateful to Professor V. S. Subrahmanian for agreeing to serve on my committee as the Dean's representative without any prior notice.

Many thanks to Professor George V. Moustakides who greatly helped me in understanding of sequential detection principles. His patience and generous feedback on many problems we worked on helped me greatly in writing my thesis and shed light on many unsolved problems I was working on.

I am in particular indebted to my colleagues and friends, with whom I spent many years in College Park. I would like to thank Nassir BenAmmar for his generous help with implementation of the IEEE 802.11 MAC misbehavior models in OPNET and many useful discussions on several open problems. Angela Huang helped me greatly in deeper understanding of routing protocols and their implementation. Special thanks goes to Alvaro Cardenas for many years of friendship and cooperation. I learned a lot from our lengthy discussions on many problems in the area of Intrusion Detection.

I would also like to thank Aleksandar Simić and Katarina Stojadinović for many years of friendship. Both of them, each in their own way, helped me greatly during my studies.

Finally, I would like to thank with all my heart to my family for their infinite love

and support throughout my life. Their love gave me strength to go on during the most difficult moments of my life and the least I can do is to dedicate this thesis to them.

I am grateful for the support of my research work and graduate studies through the following contracts and grants: the U.S. Army Research Office under CIP URI grant No. DAAD19-01-1-0494 and by the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011.

Table of Contents

List of Figures vii

List of Abbreviations ix

1 Introduction 1
1.1 Our contributions . . . 3
1.2 Thesis Organization . . . 5

2 Literature overview 7
2.1 MAC layer misbehavior detection . . . 7
2.2 Cross-layer misbehavior detection . . . 10

3 IEEE 802.11 MAC DCF 12
3.1 Overview of the protocol . . . 12
3.2 IEEE 802.11 MAC Misbehavior . . . 13
3.3 Impact of interference on misbehavior detection schemes . . . 16
3.3.1 Interference due to concurrent transmissions . . . 17
3.3.2 Interference due to simultaneous channel access . . . 18

4 Min-max robust misbehavior detection 20
4.1 Introduction . . . 20
4.2 Problem motivation and sequential detection . . . 21
4.3 Min-max robust detection: definition of uncertainty class . . . 26
4.3.1 Problem description and assumptions . . . 27
4.3.2 Adversary model . . . 28
4.4 Min-max robust detection: derivation of the worst-case attack . . . 32
4.5 Experimental evaluation of optimal attack strategies . . . 37
4.5.1 Impact of multiple competing nodes on the performance of the optimal attacker . . . 43
4.5.2 Performance comparison of MAC layer misbehavior detection schemes 43

5 Collaborative attacks 49
5.1 Definition of the Uncertainty Class . . . 50
5.2 Derivation of the worst-case attack for n=2 adversaries . . . 52
5.3 Derivation of the worst-case attack for $n > 2$ adversaries . . . 55
5.4 Experimental Results . . . 57

6 Impact of interference on the performance of optimal detection schemes 62
6.1 Overview . . . 62
6.2 Problem setup . . . 64
6.2.1 Derivation of the worst-case attack in the presence of interference . . . 66
6.3 FSM for SINR variation . . . 70
6.3.1 System model . . . 70
6.3.2 Performance analysis . . . 71

7 Cross-entropy minimization and its applications in intrusion detection 74
7.1 Analysis of single and multi-stage attacks 74
7.2 Derivation of the worst-case attack using the principle of minimum cross-entropy 78
7.3 Optimal Attack Scenario in the MAC Layer Using the Cross-entropy Method 81

8 Cross-layer impact of optimal attacks 83
8.1 Impact of MAC Layer Misbehavior on the Network Layer: Time to Buffer Overflow 85
8.2 Numerical Results 91
8.2.1 Cross-layer effects of the optimal MAC layer attacks 91
8.2.2 Implementation of an optimal MAC layer-based IDS 95

Bibliography 99

List of Figures

3.1 Nodes A and C contend for accessing node B. In the first attempt A reserves the channel followed by successful access by node C. 13

3.2 Observer nodes and effect of interference due to concurrent transmissions. . 15

4.1 Form of least favorable pdf $f_1^\star(x)$: a) number of legitimate nodes $n = 2$, 1 malicious node and gain factor $\eta = 1, 1.5, 2, 2.5$; b) gain factor $\eta = 1.5$ and number of legitimate nodes $n = 1, 2, 5, \infty$; c) absolute gain $\frac{\eta}{n+1} = \frac{1}{2}$ and number of legitimate nodes $n = 1, 2, 5, 10, 20$. 38

4.2 Average Detection Delay $\mathbb{E}[N]$ as a function of (a) gain factor η; (b) absolute gain $\frac{\eta}{n+1}$ for $\alpha = \beta = 0.01$ 40

4.3 Tradeoff curve for $\frac{\eta}{n+1} = 0.5$, 0.6, 0.8 and $n = 2$. 42

4.4 Tradeoff curve for $\frac{\eta}{n+1} = 0.5$ and $n = 2, 3$. 42

4.5 Tradeoff curve for $\frac{\eta}{n+1} = 0.6$ and $n = 2, 3, 4, 5$. 43

4.6 Tradeoff curve for $\frac{\eta}{n+1} = 0.5$ and $n = 2, 3, 4$. 44

4.7 Tradeoff curves for DOMINO algorithm. One curve shows its performance when detecting an adversary that chooses f_1^D and the other is the performance when detecting an adversary that chooses f_1^* 45

4.8 Tradeoff curves for SPRT algorithm. One curve shows its performance when detecting an adversary that chooses f_1^D and the other is the performance when detecting an adversary that chooses f_1^*. 46

4.9 Tradeoff curves for SPRT and DOMINO algorithms. 47

5.1 The optimal pdf of colluding adversaries. 54

5.2 Tradeoff curves for 2 colluding nodes and $\eta = 0.3$, 0.6 and 0.9. 59

5.3 Tradeoff curves for $\eta = 0.6$: detection times for colluding nodes are up to 2 times longer than for a single node with identical strategy. 59

5.4 Tradeoff curves for $\eta = 0.9$: detection times for colluding nodes are up to 3 times longer than for a single node with identical strategy. 60

5.5 Tradeoff curves for $\eta = 0.9$ (single attacker) and $\eta = 0.4$ (colluding attackers). 61

6.1 Average detection delay for different values of SINR and n=1, 3, 10 62

6.2 PER[%] as a function of SINR for RTS and CTS messages 64

6.3 Noise diagram. 66

6.4 Markov Chain representation of the system. Each state corresponds to a different SINR level. 70

6.5 Performance comparison of the detection scheme with and without interference for $\frac{\eta}{n+1} = 0.8$. 72

8.1 $Node2$ is silenced by the transmission of the selfish node. Consequently, $Node1$ drops large number of packets. 85

8.2 An ongoing attack in the MAC layer breaks the original route, re-routing the traffic through $Node3$. 85

8.3 Arrival and departure times in the queue of length δ 86

8.4 Average Time to buffer overflow for $\rho = \beta/\alpha = 3/2$ (stability) and $\rho = \beta/\alpha = 2/3$ (instability), as a function of the buffer size ν. 89

8.5 Average time to buffer overflow as a function of the traffic rate ratio $\rho = \beta/\alpha$ and buffer size $\nu = 100$. 89

8.6 The amount of lost traffic as a function of detection delay for fixed buffer size ν=100. 91

8.7 Increase in dropped traffic at $Node1$. 93

8.8 Percentage increase in traffic through alternate route as a consequence of an ongoing MAC layer attack. 94

8.9 Proposed cross-layer collaboration 96

List of Abbreviations

AODV	Ad hoc On-Demand Distance Vector
BER	Bit Error Rate
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear To Send
CUSUM	Cumulative Sum
CW	Contention Window
DCF	Distributed Coordination Function
DIFS	Distributed Inter-Frame Space
DoS	Denial of Service
DSR	Dynamic Source Routing
IDS	Intrusion Detection System
LAR	Location-Aided Routing
MAC	Media Access Control
MACA	Multiple Access Collision Avoidance
NAV	Network Allocation Vector
PER	Packet Error Rate
RTS	Request To Send
SIFS	Short Inter-frame Space
SINR	Signal to Interference and Noise Ratio
SPRT	Sequential Probability Ratio Test

Chapter 1

Introduction

Deviation from legitimate protocol operation in wireless networks has received considerable attention from the research community in recent years. The pervasive nature of wireless networks with devices that are gradually becoming essential components in our everyday life justifies the rising interest on that issue. In addition, the architectural organization of wireless networks in distributed secluded user communities raises issues of compliance with protocol rules. More often than not, users are clustered in communities that are defined on the basis of proximity, common service or some other common interest. Since such communities are bound to operate without a central supervising entity, no notion of trust can be presupposed.

Furthermore, the increased level of sophistication in the design of protocol components, together with the requirement for flexible and readily reconfigurable protocols has led to the extreme where wireless network adapters and devices have become easily programmable. As a result, it is feasible for a network peer to tamper with software and firmware, modify its wireless interface and network parameters and ultimately abuse the protocol. This situation is referred to as protocol misbehavior. The goals of a misbehaving peer range from exploitation of available network resources for its own benefit up to network disruption. The solution to the problem is the timely and reliable detection of such misbehavior instances, which would eventually lead to network defense and response mechanisms and isolation of the misbehaving peer. However, two difficulties arise: the random nature of some protocols (such as the IEEE 802.11 medium access control one)

and the nature of the wireless medium with its inherent volatility. Therefore, it is not easy to distinguish between a peer misbehavior and an occasional protocol malfunction due to a wireless link impairment. An additional difficulty specific for the wireless environment arises when observations of protocol participants are hindered by interference due to concurrent transmissions. As a consequence, a detector may miss one or more control messages sent by the attacker, which delays the detection process due to the fact that a detector registers erroneous observation sequence. In the less severe case, when the perceived and actual interference levels are similar, the detector is aware of existence of discrepancies between the measured and actual behavior of monitored peers and either adjusts its detection strategy or notifies the rest of the network that it is unable to reach a reliable decision. In the more severe case when the perceived interference level is significantly lower than the actual one, an increase in false negatives is observed, i.e. the number of missed detections increases.

Further challenges arise in the presence of multiple collaborating adversaries. We assume that colluding participants collaborate by exchanging information and by taking actions that amplify each other's effects on network functionality. Furthermore, such collaborative attacks employ "intelligence", that is, observe actions of detectors and defenders and adjust the timing or the stages or the actions of the adversaries. Understanding the performance of the collaborating adversaries versus the collaborating detectors and defenders is a key issue that involves several fundamental challenges that include modeling of optimal adversarial strategies, optimal detection, timely localization etc.

It is reasonable to assume that an intelligent adversary does not focus his activities at the origin of the attack only, but attempts to disrupt the network functionality on a larger scale by employing strategies that result in both horizontal and vertical propagation

of misbehavior. As a consequence, a detection system that resides in a single network layer may not be sufficient for detection of more sophisticated attacks strategies.

It is important to mention that due to the unpredictability of wireless protocols and the medium itself, it is impossible to completely predict adversarial behavior. More specifically, as it will be demonstrated in this thesis, such approach is undesirable and leads to construction of an IDS that is capable of detecting only a narrow class of attacks. For that specific class of attacks the given IDS exhibits superior detection rate, but when the adversarial strategy slightly deviates from the original one, the detection rate quickly falls below an acceptable threshold. In this thesis we aim to provide general performance bounds for the worst-case attack scenarios in wireless networks for the case of a single intelligent adversary in the environment with and without interference and colluding adversaries. We adopt the game-theoretic approach for modeling such behaviors and extend our analysis by introducing the notion of minimum cross-entropy. The provided scenarios represent the worst-case performance bounds of the detection system.

1.1 Our contributions

In the first part of the thesis, we address the problem of MAC protocol misbehavior detection at a fundamental level and cast it as a min-max robust detection problem. We perform our analysis by assuming the presence of an intelligent adaptive adversary. Our work contributes to the current literature by: (i) formulating the misbehavior problem as a min-max robust sequential detection problem that encompasses the case of an intelligent attacker, (ii) quantifying performance losses incurred by an attack and defining an uncertainty class such that the focus is only on attacks that incur "large enough" performance losses, therefore avoiding the trap of wasting system resources on detection and

notification of minor short-term disruptions in the network that may or may not be of adversarial nature, (iii) obtaining an analytical expression for the worst-case attack and the number of observations required for detection, (iv) establishing an upper bound on number of required samples for detection of any of the attacks of interest, therefore providing the worst-case performance evaluation of the given detection system, (v) extending the basic model to scenarios with interference due to concurrent transmissions and obtaining performance bounds of both the adversary and the detection systems in such settings. We implement the derived optimal class of attacks in the network simulator OPNET [1] and compare the performance of such attacks against optimal and sub-optimal detection schemes. Furthermore, we extend the proposed framework by formulating the problem of optimal detection against misbehavior of intelligent colluding attackers in the IEEE 802.11 MAC and obtain an upper bound on number of required samples for detection of such attacks. In addition to that, we perform detailed evaluation of collaborative attacks and quantify their performance by comparing their effects on the system with the effects of a single attacker of identical strength and emphasize the importance of localization in timely detection of such attacks.

The different layers in the network stack communicate with each other, enabling the propagation of misbehavior instances between layers. Thus, misbehavior that takes place at the MAC layer can significantly affect the routing process as well. The current literature only considers brute force attacks, such as Denial of Service (DoS) attacks in the MAC layer and their impact on the Network Layer. In this thesis we investigate the effects of the worst-case attacks that originate in the MAC layer on two routing protocols. We show by analysis and simulation that vertical propagation of misbehavior gives rise to new threats, such as false accusation of legitimate nodes by the IDS located in the

network layer. Additionally, the distributed nature of the wireless ad hoc networks as well as the randomness of the employed protocols, makes the task of detection and localization of malicious participants extremely challenging.

Finally, we apply the principle of minimum cross-entropy and derive a general framework for construction of optimal attacks in the IEEE 802.11 MAC.

1.2 Thesis Organization

The thesis is organized as follows. Chapter 2 discusses existing work in the areas of the IEEE 802.11 MAC misbehavior detection and cross-layer propagation and detection of such attacks. Chapter 3 presents a brief overview of the IEEE 802.11 MAC DCF and analyzes its potential vulnerabilities (i) in regular settings and (ii) in the presence of interference. In Chapter 4 we formally define our problem of misbehavior detection and place it into a min-max robust framework. We define performance bounds of an intelligent adaptive attacker and the quickest IDS using game-theoretic approach and perform both analytical and experimental evaluation in various settings. In Chapter 5, we extend the proposed framework to the case of colluding adversaries and obtain the expression for the worst-case attack for the case of $n \geq 2$ collaborating adversaries. We analyze the impact of collaborating adversaries on the performance of the system and compare the effects to the one obtained by a single attacker of the same strength in terms of detection delay. In Chapter 6 we continue the analysis from Chapter 4 by providing a detailed analysis of impact of interference on the performance of quickest detection schemes. In Chapter 7 we apply the method of cross-entropy minimization to the problem of worst-case attacks in the IEEE 802.11 MAC. Finally, in Chapter 8 we analyze the impact of the worst-case MAC layer attacks on the performance of two Network Layer protocols and propose an efficient

cross-layer detection scheme that provides timely prevention of vertical propagation of such attacks.

Chapter 2

Literature overview

Protocol misbehavior has been studied in various scenarios in different communication layers and under several mathematical frameworks. To our knowledge, there exists no unique adversarial model that can be used for evaluation of existing IDSs. The lack of such models that capture a wide class of misbehavior strategies (with brute force strategy being the extreme instance of misbehavior) represents a major problem for evaluation and performance comparison of existing detection schemes. In addition to that, the absence of such models makes a fair performance comparison of existing schemes almost impossible due to the fact that each detection scheme is constructed for detection of a specific class of adversarial strategies. As an illustration of this point we observe two detection systems IDS_1 with detection strategy $\mathcal{D}_1$ and IDS_2 with detection strategy $\mathcal{D}_2$ which were constructed for detection of adversarial strategies $\mathcal{A}_1$ and $\mathcal{A}_2$ respectively. We claim that due to the fact that each detection system was constructed for detection of a specific class of attacks, IDS_1 will exhibit superior performance in detecting adversarial strategy $\mathcal{A}_1$. On the other hand, it will exhibit sub-optimal performance for detection of an attack that belongs to a class $\mathcal{A}_2$. The same will hold for IDS_2. This claim will be illustrated by detailed experimental analysis in Chapter 4.

2.1 MAC layer misbehavior detection

The authors in [2] focus on MAC layer misbehavior in wireless hot-spot communities. They propose a sequence of conditions on available observations for testing the extent to

which MAC protocol parameters have been manipulated. The advantage of the scheme is its simplicity and easiness of implementation, although in some cases the method can be deceived by cheating peers, as the authors point out. A different line of thought is followed by the authors in [3], where a modification to the IEEE 802.11 MAC protocol is proposed to facilitate the detection of selfish and misbehaving nodes. The approach presupposes a trustworthy receiver, since the latter assigns to the sender the back-off value to be used. The receiver can readily detect potential misbehavior of the sender and accordingly penalize it by providing less favorable access conditions through higher back-off values for subsequent transmissions. A decision about protocol deviation is reached if the observed number of idle slots of the sender is smaller than a pre-specified fraction of the allocated back-off. The sender is labeled as misbehaving if it turns out to deviate continuously based on a cumulative metric over a sliding window. This work also presents techniques for handling potential false positives due to the hidden terminal problem and the different channel quality perceived by the sender and the receiver. The work in [4] attempts to prevent scenarios of colluding sender-receiver pairs by ensuring randomness in the course of MAC protocol.

A game-theoretic framework for the same problem at the MAC layer is provided in [5]. Using a dynamic game model, the authors derive the strategy that each node should follow in terms of controlling channel access probability by adjustment of contention window, so that the network reaches its equilibrium. They also provide conditions under which the Nash equilibrium of the network with several misbehaving nodes is Pareto optimal for each node as well. The underlying assumption is that all nodes are within wireless range of each other so as to avoid the hidden terminal problem.

Node misbehavior can be viewed as a special case of denial-of-service (DoS) attack or

equivalently a DoS attack can be considered as an extreme instance of misbehavior. DoS attacks at the MAC layer are a significant threat to availability of network services. This threat is intensified in the presence of the open wireless medium. In [6], the authors study simple DoS attacks at the MAC layer, show their dependence on attacker traffic patterns and deduce that the use of MAC layer fairness can mitigate the effect of such attacks. In [7] the authors focus on DoS attacks against the IEEE 802.11 MAC protocol. They describe vulnerabilities of the protocol and show ways of exploiting them by tampering with normal operation of device firmware.

As it can be seen from the above analysis, mostly brute force and DoS attacks are considered in current literature. Such approaches exclude existence of *intelligent adaptive* adversary that has the ability to change his behavior depending on the type of the deployed IDS and the current environment (i.e. number of competing nodes, interference levels, etc.). In this work we adopt the notion of an intelligent adaptive adversary and evaluate his impact on optimal IDS. By adopting a general adversarial model we (i) derive performance bounds of the adversary, (ii) derive performance bounds of the IDS (i.e. evaluate the best and worst-case scenarios with respect to the detection delay) and (iii) enable comparison of several existing adversarial strategies and detection systems by placing them in our framework.

Misbehavior detection has been studied at the network layer for routing protocols as well. The work in [8] presents the watchdog mechanism, which detects nodes that do not forward packets destined for other nodes. The pathrater mechanism evaluates the paths in terms of trustworthiness and helps in avoiding paths with untrusted nodes. The technique presented in [9] aims at detecting malicious nodes by means of neighborhood behavior monitoring and reporting from other nodes. A trust manager, a reputation

manager and a path manager aid in information circulation throughput the network, evaluation of appropriateness of paths and establishment of routes that avoid misbehaving nodes. Detection, isolation and penalization of misbehaving nodes are also attained by the technique above.

2.2 Cross-layer misbehavior detection

Various IDS techniques, mostly based on misuse and anomaly detection principles, have been proposed for attack detection and prevention. Most of the existing intrusion detection approaches focus on attack detection and response at a particular layer of the protocol stack, mostly the network layer. The effects of the various attacks launched in one layer on the performance of another layer have not been widely investigated. The authors in [10] present a cautionary perspective on cross-layer design. They emphasize the importance of the approach and discuss the architectural problems that cross-layer design, if done without care, can create. In [11], the authors define the notion of cross-layer design and state three main reasons for using it in the wireless environment: (i) the unique problems created by the wireless links; (ii) the possibility of opportunistic communication on wireless links and (iii) the new modalities of communication offered by the wireless medium. In addition to that, they classify cross-layer design proposals and present proposals for implementing cross-layer interactions. The field of intrusion detection has not appropriately addressed the importance of cross-layer design and its benefits in attack detection and prevention. In [12] the authors use a cross-layer based IDS system to analyze the anomalies in the network. They introduce the concept of integrating multiple layers of the protocol stack for more efficient intrusion detection. In [13] the authors study the interaction of the routing and MAC layer protocols under different mobility parameters.

They simulate interaction between three MAC protocols (MACA, 802.11 and CSMA) and three routing protocols (AODV, DSR and LAR scheme) and perform statistical analysis in order to characterize the interaction between layers in terms of latency, throughput, number of packets received and long term fairness. In [14] the authors quantify the impact of link-layer misbehavior on the performance of two routing protocols, DSR and AODV. They investigate two brute force attacks in the link layer: constant RTS/CTS packet dropping and back-off manipulation and prove by simulation that each of the above attacks propagates to the network layer, affecting the overall network performance. In [15], the authors aim to develop a cross-layer detection framework that detects and localizes malicious participants in various layers of the network. They consider only brute force attacks, such as DoS attack in the MAC layer and packet dropping in the network layer.

Chapter 3

IEEE 802.11 MAC DCF

3.1 Overview of the protocol

The most frequently used MAC protocol for wireless networks is the IEEE 802.11 MAC protocol, which uses a distributed contention resolution mechanism for sharing the wireless channel. Its design attempts to ensure a relatively fair access to the medium for all participants of the protocol. In order to avoid collisions, the nodes follow a binary exponential back-off scheme that favors the last winner amongst the contending nodes.

In Distributed Coordinating Function (DCF) of the IEEE 802.11 MAC protocol, coordination of channel access for contending nodes is achieved with Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) [16]. A node with a packet to transmit selects a random back-off value b uniformly from the set $\{0, 1, \ldots, W-1\}$, where W is the (fixed) size of the contention window. The back-off counter decreases by one at each time slot that is sensed to be idle and the node transmits after b idle slots. In case the channel is perceived to be busy in one slot, the back-off counter stops momentarily. After the back-off counter is decreased to zero, the transmitter can reserve the channel for the duration of data transfer. First, it sends a request-to-send (RTS) packet to the receiver, which responds with a clear-to-send (CTS) packet. Thus, the channel is reserved for the transmission. Both RTS and CTS messages contain the intended duration of data transmission in the duration field. Other hosts overhearing either the RTS or the CTS are required to adjust their Network Allocation Vector (NAV) that indicates the duration for which they will defer transmission. This duration includes the SIFS intervals, data

packets and acknowledgment frame following the transmitted data frame. An unsuccessful transmission instance due to collision or interference is denoted by lack of CTS or ACK for the data sent and causes the value of contention window to double. If the transmission is successful, the host resets its contention window to the minimum value W.

Fig. 3.1 illustrates the scenario of contending nodes using the protocol.

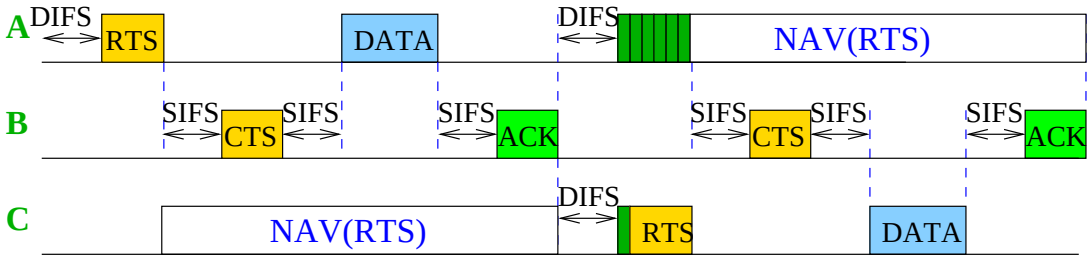

Figure 3.1: Nodes A and C contend for accessing node B. In the first attempt A reserves the channel followed by successful access by node C.

Typical parameter values for the MAC protocol depend on the physical layer that IEEE 802.11 uses. Table 3.1 shows the parameters used when the physical layer is using direct sequence spread spectrum (DSSS).

3.2 IEEE 802.11 MAC Misbehavior

As it has been seen in Sect. 3.1, the IEEE 802.11 DCF favors the node that selects the smallest back-off value among a set of contending nodes. Therefore, a malicious or selfish node may choose not to comply to protocol rules by occasionally or constantly selecting small back-off values, thereby gaining significant advantage in channel sharing over regularly behaving, honest nodes. Moreover, due to the exponential increase of the contention window after each unsuccessful transmission, non-malicious nodes are forced to select their future back-offs from larger intervals after every access failure. Consequently,

DIFS	$50\mu s$
SIFS	$10\mu s$
SlotTime	$20\mu s$
ACK	112bits+PHY_header=$203\mu s$
RTS	160bits+PHY_header=$207\mu s$
CTS	112bits+PHY_header=$203\mu s$
DATA	MAC_header (30b)+DATA(0-2312b)+FCS(4b)
Timeouts	300-$350\mu s$
CW_{min}	32 time slots
CW_{max}	1024 time slots

Table 3.1: Parameters for DSSS

their chances of accessing the channel decrease even further. Apart from intentional selection of small back-off values, a node can deviate from the MAC protocol in other ways as well. He can (i) choose a smaller size of contention window; (ii) wait for shorter interval than DIFS or (iii) reserve the channel for larger interval than the maximum allowed NAV duration. In this work, we adhere to protocol deviations that occur due to manipulation of the back-off values.

The nodes that are instructed by the protocol to defer transmission are able to overhear transmissions from nodes whose transmission range they reside in. Therefore, silenced nodes can observe the behavior of transmitting nodes. The question that arises is whether there exists a way to take advantage of this observation capability and use it to identify potential misbehavior instances. If observations indicate a misbehavior event, the observer nodes should notify the rest of the network about this situation or could launch

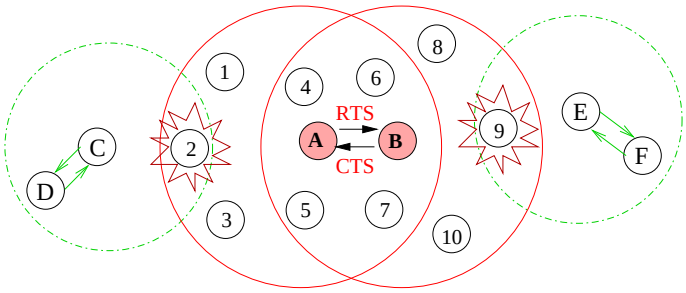

Figure 3.2: Observer nodes and effect of interference due to concurrent transmissions.

a response action in order to isolate the misbehaving nodes. Detecting misbehavior is not straightforward even in the simplest case, namely that of unobstructed observations. The difficulty stems primarily from the non-deterministic nature of the access protocol that does not lead to a straightforward way of distinguishing between a legitimate sender, that happens to select small back-offs, and a misbehaving node that maliciously selects small back-offs. The open wireless medium and the different perceived channel conditions at different locations add to the difficulty of the problem. Additional challenges arise in the presence of interference due to ongoing concurrent transmissions. Fig. 3.2 depicts a scenario where node A or B is malicious. At this stage, we assume that A is the only misbehaving node and that no other node in its vicinity transmits. We assume that nodes have clocks that are synchronized through the use of GPS devices. Additional issues arising from errors in clock synchronization are not investigated in this work. Node A accesses the channel by using a randomly selected back-off value within its contention window. When the back-off counter decreases to zero, A sends an RTS to B, which replies with a CTS. Node A's RTS message silences nodes 1 to 7, which are in A's transmission radius. Similarly, node B's CTS silences nodes 4 to 10. Following the RTS-CTS handshake, A sends a data segment to B. After the transmission is over, A attempts to access the channel anew by selecting a back-off value again and the procedure repeats. Nodes 1-10 can hear

the transmissions of nodes A or B, or of both, depending on whose transmission radius they reside in. Consider the i-th transmission of node A. A node in its transmission range finds time point t_i of RTS packet reception from

$$t_i = T_{i-1} + T_{\text{DIFS}} + b_i, \ i > 1, \tag{3.1}$$

where T_{i-1} denotes the end time point of reception of the previous data segment and b_i is the random back-off value. Thus, the back-off values can be easily derived. Note that the back-off value before transmission of the first data segment cannot be found since there does not exist any previous reference point to compare it to. A node within transmission range of B can also compute the back-off used by A by using as a reference the time point of reception of the overheard ACK from node B for the previous data segment. Then, a node can measure time point t'_i of CTS packet reception and compute the back-off of node A by using

$$t'_i = T_{\text{ACK},i-1} + T_{\text{DIFS}} + b_i + T_{\text{RTS}} + T_{\text{SIFS}}, \ i > 1. \tag{3.2}$$

Similarly with the RTS, the first back-off value cannot be found. Clearly, the entire sequence of back-offs of node A is observable in this fashion. It should also be noted that the identity of the node who uses those back-offs (which could be potentially a misbehaving one) is revealed in the corresponding fields of RTS or CTS messages.

3.3 Impact of interference on misbehavior detection schemes

Up to this point, we have assumed that both the attacker and the detector observe each back-off value and that no errors are present. However, the main characteristic of the wireless medium is its unpredictability and instability. Namely, it is not realistic to assume that both the attacker and the detector will always obtain a perfect sequence of

back-off values. It is reasonable to assume that due to interference both the adversary and the IDS will obtain a mixture of correct and erroneous observations at certain points of time. A detailed analysis of such scenarios and their impact on the performance of optimal attackers and detection schemes will be provided in Chapter 6.

In order to provide an insight into impact of interference on the performance of the IEEE 802.11 MAC participants, we now describe two scenarios in which observations of nodes 1-3 and 8-9 from Fig. 3.2 are hindered by interference and hence correctness of observations is influenced.

3.3.1 Interference due to concurrent transmissions

Assume that node C has obtained access to the channel and therefore node 2 is silenced. Node C is in the process of transmitting data packets to node D. If observer node 2 is within transmission range of C, C's transmission is overheard by node 2. Clearly, the ongoing transmission of C is experienced as interference at node 2 and obstructs node 2's observations. In case of significant interference level, node 2 may not be able to obtain the timing of received RTS of node A and find the back-off value. Additional ongoing transmissions increase the perceived interference level. Evidently, obstructed measurements due to interference create additional problems in detecting misbehavior, as will be seen in the sequel. The extent to which observations of node 2 are influenced by interference depends on the relative proximity of 2 to node A and to the interfering nodes, since the received signal strength of the RTS packet and the interference is a function of signal strength decay with distance.

3.3.2 Interference due to simultaneous channel access

Node 2 that is silenced by A's RTS observes the sequence of back-offs of node A. If node 2 is in the interference range of node C and C is out of the interference range of A, C may attempt to access the channel at the same time. If the RTS packets from nodes A and C overlap in time when received at node 2, node 2 receives a garbled packet and cannot distinguish neither the transmitter identity nor the packet reception time.

Interference from concurrent data transmissions and simultaneous channel access also affects measurements of nodes within the transmission range of node B. Both types of impairments lead to difficulties in misbehavior detection because they cause corruption of measurements. The probability of the second type of impairment is admittedly much lower than that of the first type, since it requires that nodes A and C access the channel almost at the same time. Although this problem is different from the first one, we will elaborate on obstruction of observations owing only to the first scenario.

A comment about the effect of misbehavior in a network-wide scale is in place here. Each node within transmission range of a malicious node increases its contention window exponentially after each unsuccessful transmission attempt. The same holds for nodes which are located out of the transmitter's range but are able to transmit to nodes that are silenced by the transmitter (in our case, nodes C and E). They may constantly attempt to communicate with silenced nodes and consequently increase their contention windows. In that respect, the effect of a malicious node spreads in an area much larger than their transmission range and may affect channel access of nodes throughout that area.

Another arising issue is the notification of the rest of the network about the misbehavior. Although all nodes within transmission range of nodes A and B above can deduce potential misbehavior, the nature of IEEE 802.11 MAC protocol prohibits them

from obtaining access to the channel and transmitting notification information.

Chapter 4

Min-max robust misbehavior detection

4.1 Introduction

As it has been seen in Chapter 3, a malicious or selfish node may choose not to comply to protocol rules by occasionally or constantly selecting small back-off values. As a consequence of this modified access policy, such node may gain significant advantage in channel sharing over honest nodes that comply to the protocol rules. An additional obstacle in such settings arises due to the exponential increase of the contention window after each unsuccessful transmission, which decreases the chances of channel access by legitimate protocol participants.

Several frameworks for attack detection and preventions have been proposed in recent years. However, as it has been pointed out in Chapter 2, none of the proposed approaches considers intelligent adaptive attackers. More specifically, all known detection schemes are constructed for detection and prevention of either brute force or sub-optimal attacks that are focused against a specific detection scheme in a specific adversarial setting. If we assume that a specific attack strategy $\mathcal{A}_{S_1}$ was constructed against a detection algorithm $\mathcal{D}_1$ deployed by an intrusion detection system IDS_1, then the same attack strategy becomes sub-optimal once a new detection algorithm $\mathcal{D}_2$ is deployed. This results in quicker and in most cases instantaneous detection of attacks.

In this work we present a general framework for detection and prevention of intelligent adaptive adversaries. More specifically, we address the problem of MAC protocol misbehavior detection at a fundamental level and cast it as a min-max robust detection

problem, therefore capturing both the goal of the detection system (minimize detection delay) and the goal of the attacker (maximize gain). The main contributions of this work are: (i) formulation of the misbehavior problem as a min-max robust sequential detection problem that encompasses the case of a sophisticated attacker, (ii) quantification of performance losses incurred by an attack and definition of an uncertainty class that focuses only on attacks that incur "large enough" performance losses, (iii) derivation of an analytical expression for the worst-case attack and the number of observations required for attack detection, (iv) establishment of an upper bound on number of required samples needed for detection of any of the attacks of interest.

4.2 Problem motivation and sequential detection

At this point we revisit the setup presented in Fig. 3.2 and focus on monitoring the behavior of node A for the single-hop communication with node B. We assume that any node within the transmission range of A or B observes the same sequence of measurements of back-off values used by A. Since the sequence of observations is the same, the procedure that will be described in the sequel can take place in any of the observer nodes. Since the back-off measurements are enhanced by an additional sample each time A attempts to access the channel, an on-line sequential scheme is suitable for the nature of the problem. The basis of such a scheme is a sequential detection test that is implemented at an observer node. The objective of the detection test is to derive a decision as to whether or not a misbehavior occurs as fast as possible, namely with the least possible number of observation samples. Since the observation samples are random variables, the number of required samples for taking a decision is a random variable as well.

A sequential detection test is a procedure which with every new information that arrives asks the question whether it should *stop* receiving more samples or continue sampling. If the answer to the first question is to stop (because sufficient information has been accumulated) then it proceeds to the phase of making a *decision* on the nature of the data. It is therefore clear that there are two quantities involved: a stopping time (s.t.) N which is a random variable taking positive integer values and denoting the time we decide to stop getting more data; and a decision rule d_N which at the time of stopping N decides between the two hypotheses $\mathbf{H}_0, \mathbf{H}_1$ and therefore assumes the values 0,1. For simplicity, let us denote with $\mathcal{D}$ the combination $\mathcal{D} = (N, d_N)$ of the s.t. N and the decision rule d_N.

The probability of false alarm and the probability of missed detection constitute inherent tradeoffs in a detection scheme. Clearly, we can obtain small values for both of these two decision error probabilities by accumulating more information, that is, at the expense of larger detection delay. A logical compromise would therefore be to prescribe some maximal allowable values for the two error probabilities, and attempt to *minimize* the expected detection delay. Expressing this problem under a more formal setting, we are interested in finding a sequential test $\mathcal{D} = (N, d_N)$ that solves the following constraint optimization problem

$$\inf_{N, d_N} \mathbb{E}_1[N], \quad \text{under the constraints } \mathbb{P}_0[d_N = 1] \leq \alpha; \ \mathbb{P}_1[d_N = 0] \leq \beta; \tag{4.1}$$

where $\mathbb{P}_i$, $\mathbb{E}_i$ denote probability and expectation under hypothesis $\mathbf{H}_i$, $i = 0, 1$, and $0 < \alpha, \beta < 1$ are the prescribed values for the probability of false alarm and miss respectively.

This mathematical setup was first proposed by Wald in [17], where he also introduced the Sequential Probability Ratio Test (SPRT) for its solution. The SPRT test is defined

in terms of the log-likelihood ratio S_n

$$S_n = \ln \frac{f_1(x_1, \ldots, x_n)}{f_0(x_1, \ldots, x_n)}, \tag{4.2}$$

of the two joint probability density functions $f_i(x_1, \ldots, x_n)$ of the data $\{x_1, \ldots, x_n\}$ under hypothesis $\mathbf{H}_i$, $i = 0, 1$. The corresponding s.t. N and decision rule d_N are then given by

$$N = \inf_n \{n : \ S_n \notin [A, B] \tag{4.3}$$

$$d_N = \begin{cases} 1 & \text{if } S_N \geq B \\ 0 & \text{if } S_N \leq A, \end{cases} \tag{4.4}$$

where thresholds $A < 0 < B$ depend on the specified values of P_{FA} and P_M. From Wald's identity [17]

$$\mathbb{E}[S_N] = \mathbb{E}[N] \times \mathbb{E}[\Lambda] \tag{4.5}$$

where $\mathbb{E}[\Lambda]$ is the expected value of the logarithm of the likelihood ratio. By using a similar approach as the one in [18, pp.339-340], we can derive the following inequalities

$$1 - P_M \geq e^a P_{FA} \ \text{ and } \ P_M \leq e^b (1 - P_{FA}), \tag{4.6}$$

where a and b are the thresholds of SPRT. When the average number of required observations is very large, the increments Λ_j in the logarithm of the likelihood ratio are also small. Therefore, when the test terminates with selection of hypothesis $\mathbf{H}_1$, S_N will be slightly larger than a, while when it terminates with selection of $\mathbf{H}_0$, S_N will be very close to b. Therefore, the above inequalities hold to a good approximation as equalities. Under this assumption, the decision levels a and b that are required for attaining performance (P_{FA}, P_M) are given by,

$$a = \ln \frac{1 - P_M}{P_{FA}} \ \text{ and } \ b = \ln \frac{P_M}{1 - P_{FA}}. \tag{4.7}$$

Following the derivations of [17, 18],

$$\mathbb{E}[S_N] = aP_D + b(1 - P_D) \tag{4.8}$$

where $P_D = 1 - P_M$ is the probability of detection of SPRT. By substituting the above equation into Eq. (4.5) and utilizing the fact that $\mathbb{E}[S_N] = const = C$ for a given IDS with fixed P_D and P_M, the following expression for detection delay is derived:

$$\mathbb{E}[N] = \frac{\mathbb{E}[S_N]}{\mathbb{E}[\Lambda]} = \frac{C}{\mathbb{E}\Big[\ln \frac{f_1}{f_0}\Big]} \tag{4.9}$$

We can see that the SPRT test continues sampling as long as the log-likelihood ratio takes values within the interval (A, B) and stops taking more samples the first time it exceeds it. Once stopped, the decision function d_N decides in favor of hypothesis $\mathbf{H}_1$ when S_N exceeds the largest threshold and in favor of $\mathbf{H}_0$ when S_N is below the smallest threshold. If in particular the data are independent and identically distributed (i.i.d.) under both hypotheses then the log-likelihood ratio S_n takes the following simple form

$$S_n = \sum_{k=1}^{n} \ln \frac{f_1(x_k)}{f_0(x_k)} = S_{n-1} + \ln \frac{f_1(x_n)}{f_0(x_n)},\ S_0 = 0. \tag{4.10}$$

Here $f_i(x)$ is the common probability density function (pdf) of the samples under hypothesis $\mathbf{H}_i,\ i = 0, 1$. Notice that the recurrent relation on the right hand side of Eq.(4.10) allows for an efficient computation of the statistics S_n which requires only constant number of operations per time step and finite memory (we only need to store S_n as opposed to the whole sequence $\{x_n, \ldots, x_1\}$).

Optimality of SPRT in the sense described in (4.1) is assured *only* when the data are i.i.d. under both hypotheses [19]. For other data models there exists a very rich literature referring to asymptotic optimality results (see for example [20]). Concluding, we should also mention that the actual optimality of SPRT is significantly stronger than the one

mentioned in (4.1). The SPRT not only minimizes the average delay under $\mathbf{H}_1$ but also *simultaneously* minimizes the alternative average delay $\mathbb{E}_0[N]$. This double optimality property is rather remarkable and not encountered in any other detection scheme.

It is clear from the previous discussion that our intention is to follow a sequential approach for the detection of attacks. It is important to notice that in order to be able to use the SPRT it is necessary to specify both probability density functions $f_i(x),\ i = 0, 1$ under the two hypotheses. Although the pdf $f_0(x)$ of a legitimate node is known, this is not the case for an attacker. Furthermore, specifying a candidate density $f_1(x)$ for an attacker without some proper analysis may result in serious performance degradation if the attacker's strategy diverges from our selection.

In order to be able to propose a specific detection rule we need to clarify and mathematically formulate the notion of an "attack". We should however place our main emphasis to attacks that incur large gains for the attacker (result in higher chances of channel access). An attack will then have devastating effects for the network, in the sense that it would deny channel access to the other nodes and would lead to unfair sharing of the channel. Besides, if we assume that the detection of an attack is followed by communication of the attack event further in the network so as to launch a network response, it would be rather inefficient for the algorithm to consider less significant (and potentially more frequent) attacks and initiate responses for them. Instead, it is meaningful for the detection system to focus on encountering the most significant attacks and at the same time not to consume resources of any kind (processor power, energy, time or bandwidth) for dealing with attacks whose effect on performance is rather marginal.

4.3 Min-max robust detection: definition of uncertainty class

Previously, we stressed the sequential nature of our approach and the implicit need to consider most significant attacks. The approach should also cope with the encountered (statistically) uncertain operational environment of a wireless network, namely the random nature of protocols and the unpredictable misbehavior or attack instances. Hence, it is desirable to rely on robust detection rules that would perform well regardless of uncertain conditions. In this work, we adopt the min-max robust detection approach where the goal is to optimize performance for the worst-case instance of uncertainty. More specifically, the goal is to identify the least favorable operating point of a system in the presence of uncertainty and subsequently find the strategy the optimizes system performance when operating in that point. In our case, the least favorable operating point corresponds to the worst-case instance of an attack and the optimal strategy amounts to the optimal detection rule. System performance is measured in terms of number of required observation samples to derive a decision.

A basic notion in min-max approaches is that of a *saddle point*. A strategy (detection rule) $\mathcal{D}^\star = (N^\star, d_N^\star)$ and an operating point (attack) $f_1^\star$ in the uncertainty class form a saddle point if:

1. For the attack $f_1^\star$, any detection rule $\mathcal{D}$ other than $\mathcal{D}^\star$ has worse performance. Namely $\mathcal{D}^\star$ is the optimal detection rule for attack $f_1^\star$ in terms of minimum (average) number of required observations.

2. For the detection rule $\mathcal{D}^\star$, any attack f_1 from the uncertainty class, other than $f_1^\star$ gives better performance. Namely, detection rule $\mathcal{D}^\star$ has its worst performance for attack $f_1^\star$.

Implicit in the min-max approach is the assumption that the attacker has full knowledge of the employed detection rule. Thus, it can create a misbehavior strategy that maximizes the number of required samples for misbehavior detection delaying the detection as much as possible. Therefore, our approach refers to the case of an intelligent attacker that can adapt its misbehavior policy so as to avoid detection. One issue that needs to be clarified is the structure of this attack strategy. Subsequently, by deriving the detection rule and the performance for that case, we can obtain an (attainable) upper bound on performance over all possible attacks.

4.3.1 Problem description and assumptions

According to the IEEE 802.11 MAC standard, the back-off for each legitimate node is selected from a set of values in a contention window interval based on uniform distribution. The length of contention window is $2^i W$ for the ith retransmission attempt, where W is the minimum length of the contention window. In general, some back-off values will be selected uniformly from $[0, W]$ and others will be selected uniformly from intervals $[0, 2^i W]$, for $i = 1, \ldots, I_{\max}$ where $I_{\max}$ is the maximum number of re-transmission attempts. Without loss of generality, we can scale down a back-off value that is selected uniformly in $[0, 2^i W]$ by a factor of 2^i, so that all back-offs can be considered to be uniformly selected from $[0, W]$. We now present the problem and justify the above assumptions.

Assume each station generates a sequence of random back-offs $X_1, X_2, \ldots, X_i$ in order to access the channel. The back-off values of each legitimate protocol participant are then distributed according to the pdf $f_0(x)$, which is specified by the MAC layer protocol. Furthermore, the pdf of the misbehaving participants is unknown to the system and is denoted with $f_1(x)$.

We assume that a detection agent (e.g., the access point) monitors and collects the back-off values of a given station. It is important to note that observations are not perfect and can be hindered by concurrent transmissions or external sources of noise. It is impossible for a passive monitoring agent to know the internal exponential back-off stage of a given monitored station due to collisions, or to the fact that a station might not have anything to transmit. Furthermore, in practical applications the number of false alarms in anomaly detection schemes is very high. Consequently, instead of building a "normal" profile of network operation with anomaly detection schemes, we utilize specification based detection. In our setup we identify "normal" (i.e., a behavior consistent with the IEEE 802.11 specification) profile of a backlogged station in the IEEE 802.11 without any competing nodes, and notice that its back-off process $X_1, X_2, \ldots, X_i$ can be characterized with pdf $f_0(x_i) = 1/(W+1)$ for $x_i \in \{0, 1, \ldots, W\}$ and zero otherwise. We claim that this assumption minimizes the probability of false alarms due to imperfect observations. At the same time, a safe upper bound on the amount of damaging effects a misbehaving station can cause to the network is maintained.

Although our theoretical results utilize the above expression for f_0, the experimental setting utilizes the original implementation of the IEEE 802.11 MAC. In this case, the detection agent needs to deal with observed values of x_i larger than W, which can be due to collisions or due to the exponential back-off specification in the IEEE 802.11.

4.3.2 Adversary model

We assume that the adversary has full control over the pdf $f_1(x)$ and the back-off values it generates. In addition to that, we assume that the adversary is intelligent, i.e. the adversary knows everything the detection agent knows and can infer the same

conclusions as the detection agent. As it has already been mentioned, this assumption enables the detector to obtain the upper bound on the detection delay. In this work we consider continuously back-logged nodes that always have packets to send. Thus, the gain of the adversary $\mathcal{G}$ is signified by the percentage of time in which it obtains access to the medium. This in turn depends directly on the relative values of back-offs used by the attacker and by the legitimate nodes. In particular, the attacker competes with the node that has selected the smallest back-off value out of all nodes.

In order to derive an expression for the gain of the adversary, we first need to compute the probability P_1 of the adversary to access the channel as a function of the pdfs $f_1(.)$ and $f_0(.)$. Following the IEEE 802.11 protocol, the back-off counter of any node freezes during the transmissions and reactivates during free periods. Therefore, let us observe the back-off times during a fixed period T that *does not include* transmission intervals. Consider first the case of one misbehaving and one legitimate node and assume that within the time period T, we observe $X_1, \ldots, X_N$, N samples of the attacker's back-off and $Y_1, \ldots, Y_M$, M samples of the legitimate node's back-offs. It is then clear that the attacker's percentage of accessing the channel during the period T is $N/(N+M)$. In order to obtain the desired probability we simply need to compute the limit of this ratio as $T \to \infty$. Notice that

$$X_1 + \cdots + X_N \leq T < X_1 + \cdots + X_{N+1}$$

$$Y_1 + \cdots + Y_M \leq T < Y_1 + \cdots + Y_{M+1},$$

which yields

$$\frac{\frac{N}{X_1+\cdots+X_N}}{\frac{N}{N+1}\frac{N+1}{X_1+\cdots+X_{N+1}} + \frac{M}{M+1}\frac{M+1}{Y_1+\cdots+Y_{M+1}}} \geq \frac{\frac{N}{T}}{\frac{N}{T}+\frac{M}{T}} \geq \frac{\frac{N}{N+1}\frac{N+1}{X_1+\cdots+X_{N+1}}}{\frac{N}{X_1+\cdots+X_N} + \frac{M}{Y_1+\cdots+Y_M}}. \tag{4.11}$$

Letting $T \to \infty$ results in $N, M \to \infty$ and from the previous double inequality, by applying

the Law of Large Numbers, we conclude that

$$\mathrm{P}_1 = \lim_{N,M\to\infty} \frac{N}{N+M} = \frac{\frac{1}{\mathbb{E}_1[X]}}{\frac{1}{\mathbb{E}_1[X]} + \frac{1}{\mathbb{E}_0[Y]}}. \tag{4.12}$$

Using exactly similar reasoning the probability P_1, for the case of one misbehaving node against n legitimate ones, takes the form

$$\mathrm{P}_1 = \frac{\frac{1}{\mathbb{E}_1[X]}}{\frac{1}{\mathbb{E}_1[X]} + \frac{n}{\mathbb{E}_0[Y]}} = \frac{1}{1 + n\frac{\mathbb{E}_1[X]}{\mathbb{E}_0[Y]}} = \frac{1}{1 + n\frac{2\mathbb{E}_1[X]}{W}}, \tag{4.13}$$

where in the last equality we have used the fact that the average back-off of a legitimate node is $W/2$ (because f_0 is uniform in $[0, W]$).

If the attacker were legitimate then $\mathbb{E}_1[X] = \mathbb{E}_0[Y]$ and his probability of accessing the channel, from Eq. (4.13), would have been $1/(n+1)$. It is therefore clear that whenever

$$\mathbb{E}_1[X] = \epsilon\mathbb{E}_0[Y], \quad \text{with } \epsilon \in (0, 1) \tag{4.14}$$

the attacker enjoys a gain as compared to any legitimate node since then

$$\mathrm{P}_1 = \eta\frac{1}{n+1} > \frac{1}{n+1}, \quad \text{where } \eta = \frac{1+n}{1+\epsilon n} \in (1, n+1). \tag{4.15}$$

In other words his probability of accessing the channel is greater than the corresponding probability of any legitimate node by a factor $\eta > 1$.

Using the simple modelling introduced in the previous paragraph we are now able to quantify the notion of an "attack". Let η be a quantity that satisfies $1 < \eta < n+1$ and consider the class $\mathcal{F}_\eta$ of all pdfs that induce a probability P_1 of accessing the channel that is no less than $\eta/(n+1)$. Using Eq. (4.14) and Eq. (4.15), the class $\mathcal{F}_\eta$ can be explicitly defined as

$$\mathcal{F}_\eta = \left\{ f_1(x) : \int_0^W x f_1(x)\, dx \le \frac{1 - \frac{\eta}{n+1}}{n\frac{\eta}{n+1}} \frac{W}{2} \right\}, \quad 1 < \eta < n+1. \tag{4.16}$$

This class includes all possible attacks for which the incurred relative gain exceeds the legitimate one by $(\eta - 1) \times 100\%$. The class $\mathcal{F}_\eta$ is the uncertainty class of the robust approach and η is a tunable parameter. Notice from Eq. (4.15) that since P_1 is a probability the *gain factor* η must not exceed $n + 1$ in order for the previous inequality to produce a nonempty class $\mathcal{F}_\eta$.

By defining the class $\mathcal{F}_\eta$, we imply that the detection scheme should focus on attacks with larger impact to system performance and not on small-scale or short-term attacks. We define the severity of the attack by changing the gain factor η. Values of η larger but close to 1 are equivalent to low-impact attacks whereas values significantly larger than 1 are equivalent to DoS attacks.

We note that each system will have different tolerance levels for different behaviors and consequently the class $\mathcal{F}_\eta$ cannot be universally defined. We say that a system $\mathcal{S}$ is *robust* against a class of attacks $\mathcal{F}_\eta$ if its IDS can detect an adversary $\mathcal{A} \in \mathcal{F}_\eta$ with detection delay T_d (or N if the delay is measured in observed number of samples), while maintaining the performance level of the system above the pre-defined threshold $\mathcal{P}_T$. The parameters T and $\mathcal{P}_T$ are not fixed and vary depending on how strict security is required in a given system. A system $\mathcal{S}$ is *optimal* if its IDS is capable of constructing a universal detection strategy that minimizes the detection delay for the worst-case attack scenario. We now formally define a robust IDS.

Definition 4.3.1. *An IDS is robust against a class of attacks $\mathcal{F}_\eta$, if it can detect any adversary $\mathcal{A} \in \mathcal{F}_\eta$ with detection delay $T_d < T_{dc}$, where T_{dc} is the detection delay for which the performance level of legitimate protocol participants falls below the pre-defined threshold $\mathcal{P}_T$, while maintaining the pre-defined probability of false alarms P_{FA} and probability of miss P_M.*

In the light of the previously defined $\mathcal{F}_\eta$, it is now possible to formally define capabilities of the adversary. We assume the adversary has full control over his actions. In order to describe the capabilities of the attacker we define a feasible class of attacks $\mathcal{F}$ that describes his probable set of actions. In addition to that, we assume that for each attack strategy $\mathcal{A}_\mathcal{S} \in \mathcal{F}$ there exists an associated gain of the adversary $\mathcal{G}$. If there exist k possible attack strategies within the given class $\mathcal{F}$, then the strategy $\mathcal{A}_{S_1}$ corresponds to legitimate behavior and the strategy $\mathcal{A}_{S_k}$ corresponds to the DoS attack. Consequently, each of the strategies results in gains $\mathcal{G}_1$ and $\mathcal{G}_k$ respectively.

We assume the objective of the adversary is to design an access policy which maximizes his gain $\mathcal{G}$ over the defined period of time, while minimizing the probability of detection, P_D. If the adversary is malicious, his goal is to minimize the gain of the other participants. On the other hand, a greedy adversary attempts to maximize his own gain, which may or may not result in minimizing the gain of the other participants. We now formally define the notion of an intelligent adversary.

Definition 4.3.2. *An adversary $\mathcal{A}$ is* intelligent *if, given a set of attack strategies $\mathcal{A}_\mathcal{S} \in \mathcal{F}$, it is always capable to choose an appropriate strategy $\mathcal{A}_{S_i},\ i = 1, \ldots, k$ that minimizes the probability of detection P_D for a given gain $\mathcal{G}_i,\ i = 1, \ldots, k$.*

4.4 Min-max robust detection: derivation of the worst-case attack

Hypothesis $\mathbf{H}_0$ concerns legitimate operation and thus the corresponding pdf $f_0(x)$, as was mentioned before, is the uniform one. Hypothesis $\mathbf{H}_1$ corresponds to misbehavior with unknown pdf $f_1(x) \in \mathcal{F}_\eta$.

The objective of a detection rule is to minimize the number of the required observation samples N so as to derive a decision regarding the existence or not of misbehavior.

The performance of a detection scheme is quantified by the average number of samples $\mathbb{E}_1[N]$ needed until a decision is reached, where the average is taken with respect to the distribution $f_1(x)$ employed by the attacker. This expectation is clearly a function of the adopted detection rule $\mathcal{D}$ and the pdf $f_1(x)$, that is,

$$\mathbb{E}_1[N] = \phi(\mathcal{D}, f_1). \tag{4.17}$$

Let $\mathcal{T}_{\alpha,\beta}$ denote the class of all sequential tests for which the false alarm and missed detection probabilities do not exceed some specified levels α and β respectively. Consider also the class $\mathcal{F}_\eta$ of densities $f_1(x)$ as in (4.16) for some prescribed gain factor $\eta > 1$. In the context of the min-max robust detection framework, the goal is to optimize performance in the presence of worst-case attack, that is, solve the following min-max problem

$$\inf_{\mathcal{D}\in\mathcal{T}_{\alpha,\beta}} \sup_{f_1\in\mathcal{F}_\eta} \phi(\mathcal{D}, f_1). \tag{4.18}$$

Solving a min-max problem is usually complicated, unless one can obtain a *saddle point* solution.

Definition 4.4.1. *A pair $(\mathcal{D}^\star, f_1^\star)$ is called a saddle point of the function ϕ if*

$$\phi(\mathcal{D}^\star, f_1) \le \phi(\mathcal{D}^\star, f_1^\star) \le \phi(\mathcal{D}, f_1^\star);\ \forall \mathcal{D} \in \mathcal{T}_{\alpha,\beta},\ \ \forall f_1 \in \mathcal{F}_\eta. \tag{4.19}$$

As we can see a saddle point $(\mathcal{D}^\star, f_1^\star)$ of ϕ consists of a detection scheme $\mathcal{D}^\star$ and an attack distribution $f_1^\star$. Equation (4.19) is a formal statement of properties 1 and 2 that were mentioned in Section 4.3. The property that is important here in connection to the min-max problem (4.18) is the fact that the saddle point pair $(\mathcal{D}^\star, f_1^\star)$ also solves the min-max problem, since one can prove that [21]

$$\inf_{\mathcal{D}\in\mathcal{T}_{\alpha,\beta}} \sup_{f_1\in\mathcal{F}_\eta} \phi(\mathcal{D}, f_1) \ge \sup_{f_1\in\mathcal{F}_\eta} \phi(\mathcal{D}^\star, f_1) = \phi(\mathcal{D}^\star, f_1^\star). \tag{4.20}$$

Saddle point solutions are much easier to obtain than their min-max counterparts. Unfortunately saddle point solutions do not always exist. In view of Eq. (4.20), instead of solving Eq. (4.18) it is sufficient to find the saddle point that solves Eq. (4.19). The saddle point pair $(\mathcal{D}^\star, f_1^\star)$ is specified in the next theorem.

Theorem 4.4.2. *Let the gain factor $\eta \in (1, n+1)$ and the maximal allowable decision error probabilities α, β be given. Then the pair $(\mathcal{D}^\star, f_1^\star)$ which **asymptotically** (for small values of α, β) solves the saddle point problem defined in (4.19) is the following*

$$f_1^\star(x) = \frac{\mu}{W} \frac{e^{\mu(1-\frac{x}{W})}}{e^\mu - 1}, \tag{4.21}$$

where $\mu > 0$ is the solution to the following equation in μ

$$2\left(\frac{1}{\mu} - \frac{1}{e^\mu - 1}\right) = \frac{1 - \frac{\eta}{n+1}}{n\frac{\eta}{n+1}}. \tag{4.22}$$

The corresponding decision rule $\mathcal{D}^\star = (N^\star, d_{N^\star})$ is the SPRT test that discriminates between $f_1^\star(x)$ and $f_0(x)$(the uniform density) and is given by

$$\begin{aligned} S_n^\star &= S_{n-1}^\star + \ln \frac{f_1^\star(x_n)}{f_0(x_n)} \\ &= S_{n-1}^\star + \mu\left(1 - \frac{x_n}{W}\right) + \ln\left(\frac{\mu}{e^\mu - 1}\right); \quad S_0^\star = 0. \end{aligned} \tag{4.23}$$

$$N^\star = \inf_n \{n : S_n^\star \notin [A, B]\} \tag{4.24}$$

$$d_{N^\star} = \begin{cases} 1 & \text{if } S_N^\star \geq B \\ 0 & \text{if } S_N^\star \leq A. \end{cases} \tag{4.25}$$

Proof. We first note that (4.22) is equivalent to

$$\int_0^W x f_1^\star(x)\, dx = \frac{1 - \frac{\eta}{n+1}}{n\frac{\eta}{n+1}} \frac{W}{2} \tag{4.26}$$

which assures that $f_1^\star(x)$ defined in (4.21) is a member of the uncertainty class $\mathcal{F}_\eta$. Let us now demonstrate that for any gain factor $\eta \in (1, n+1)$ there always exists $\mu \in (0, \infty)$ so

that (4.22) is true. Notice that for $\eta \in (1, n+1)$ we have that $1/(n+1) < \eta/(n+1) < 1$. If we now call $\Phi(\mu) = 2\left(\frac{1}{\mu} - \frac{1}{e^{\mu}-1}\right)$ then $\Phi(\mu)$ is a continuous function of μ. Furthermore we observe that $\Phi(0) = 1 > \eta/(n+1)$; while one can show that $\lim_{\mu\to\infty} \Phi(\mu) = 0 < \eta/(n+1)$. Since we can find two values of μ one yielding a smaller and another a larger value than $\eta/(n+1)$, due to continuity, we can argue that there exists $\mu > 0$ such that the equality in (4.22) is assured. In fact this μ is unique since it is also possible to show that $\Phi(\mu)$ is strictly decreasing.

Let us now proceed to the saddle point problem given by Eq. (4.19). We observe that the right hand side of the inequality suggests that $\mathcal{D}^\star$ must be the optimum detection structure for $f_1^\star(x)$. Indeed, this is how $\mathcal{D}^\star$ is defined, since it is selected as the SPRT test that optimally discriminates between $f_1^\star(x)$ and the uniform $f_0(x)$.

In order to show that the left hand side is also true, we adopt an *asymptotic* approach. By considering that the two maximal error probabilities α, β are small, it is possible to use efficient approximations for the two thresholds A, B and the average detection delay function $\phi(\mathcal{D}^\star, f_1)$. Specifically, from [17] we have that A and B can be approximated as

$$A = \ln \frac{\beta}{1-\alpha}, \quad B = \ln \frac{1-\beta}{\alpha}, \tag{4.27}$$

and the expected delay by the expression

$$\phi(\mathcal{D}^\star, f_1) = \frac{A\beta + B(1-\beta)}{\int_0^W \ln \frac{f_1^\star(x)}{f_0(x)} f_1(x)\, dx}. \tag{4.28}$$

In fact these formulas become exact if the SPRT statistics $S_n^\star$ hits exactly (does not overshoot) the two thresholds A, B at the time of stopping. This for example happens in continuous-time and continuous-path processes.

Since the numerator in the previous formula is constant, the left hand side inequality in (4.19) is true if the denominator in Eq. (4.28) is minimized for $f_1(x) = f_1^\star(x)$. Because

we consider $f_1(x) \in \mathcal{F}_\eta$, inequality (4.16) applies, therefore we can write

$$\begin{aligned}
\int_0^W \ln \frac{f_1^\star(x)}{f_0(x)} f_1(x)\, dx &= \mu \int_0^W \left(1 - \frac{x}{W}\right) f_1(x)\, dx + \ln\left(\frac{\mu}{e^\mu - 1}\right) \\
&\geq \mu \left(1 - \frac{1+n-\eta}{2n\eta}\right) + \ln\left(\frac{\mu}{e^\mu - 1}\right) \\
&= \mu \int_0^W \left(1 - \frac{x}{W}\right) f_1^\star(x)\, dx + \ln\left(\frac{\mu}{e^\mu - 1}\right) \\
&= \int_0^W \ln \frac{f_1^\star(x)}{f_0(x)} f_1^\star(x)\, dx, \qquad (4.29)
\end{aligned}$$

where for the first inequality we used (4.16) and for the last two equalities we used (4.21),(4.26). This concludes the proof. □

Regarding Theorem 4.4.2 we would like to point out that our selection of $f_1^\star(x)$ was in fact the outcome of a rigorous analysis. We basically used the additional property enjoyed by the saddle point solution to solve not only the min-max problem in (4.18) but also its max-min version

$$\sup_{f_1 \in \mathcal{F}_\eta} \inf_{\mathcal{D} \in \mathcal{I}_{\alpha,\beta}} \phi(\mathcal{D}, f_1). \qquad (4.30)$$

It turns out that this latter problem can be solved directly (using standard variational techniques), thus providing us with a suitable candidate pdf $f_1^\star(x)$ for the saddle point problem (4.20). Of course we then need to go through the preceding proof in order to establish that $f_1^\star(x)$ is indeed the correct pdf.

As it was mentioned above, the min-max robust detection approach captures the case of an intelligent adaptive attacker. The SPRT algorithm is part of the intrusion detection system module that resides at an observer node. With the method outlined in this chapter, an observer node monitors the behavior of another node with the objective to derive a decision as fast as possible. In other words, the observer (and hence the system) attempts to minimize the number of required samples so as to improve its payoff in terms of improved chances for channel access. On the other hand, an intelligent attacker that

knows the detection algorithm attempts to delay this decision as much as possible so as to increase his own benefit in terms of chances for channel access. The attacker aims at a strategy that causes performance degradation for other nodes by remaining undetected.

At this point, an additional comment regarding the adversary assumptions is in place. In this specific setting we assume that the adversary is aware that an IDS is using the SPRT as a detection strategy and will stop misbehaving before it is detected. Although this may seem as a disadvantage, it is actually not. The optimal IDS forces and adversary to either (i) occasionally follow the protocol rules and shift below the threshold B; (ii) apply a mild misbehavior strategy that is below the threshold B at all times or (iii) relocate as soon as the threshold B is approached. In (i) and (ii) the attacker has to stop misbehaving or compromise with achieving a very mild advantage over other participants. In case (iii) the deployment of an optimal IDS forces an adversary to relocate frequently, therefore increasing the cost of launching an attack. It is important to note that the relocation space of an adversary is not infinite, i.e. a greedy user has to send packets to another node. Unless there is a set of collaborating adversaries, an adversary that chooses to employ aggressive misbehavior policy will be quickly detected.

4.5 Experimental evaluation of optimal attack strategies

In this section we perform experimental evaluation of optimal attack strategies derived in the previous section. The goal of the evaluation is to assess the performance of our approach and identify the relative impact of different system parameters on it. In order to evaluate the detection delay of our detection scheme against a specific class of attacks, the performance is measured in terms of the average required number of observation samples, $\mathbb{E}[N]$ in order to derive a decision, which essentially denotes the delay in

detecting a misbehavior instance. In addition to that, we investigate the influence of the number of regular participants on the form of the least favorable distribution $f_1^*(x)$.

Parameter η defines the class of attacks of interest since it specifies the incurred relative gain of the attacker in terms of the probability of channel access. In that sense, η can be interpreted as a sensitivity parameter of the detection scheme with respect to attacks, which is determined according to the IDS requirements. IEEE 802.11 MAC is implemented and MATLAB is used to evaluate the performance of our scheme, taking into account the sequence of observed back-offs.

In Fig.4.1 we present the form of the least favorable attack pdf $f_1^\star(x)$ as a function of the gain factor η and the number of legitimate nodes n.

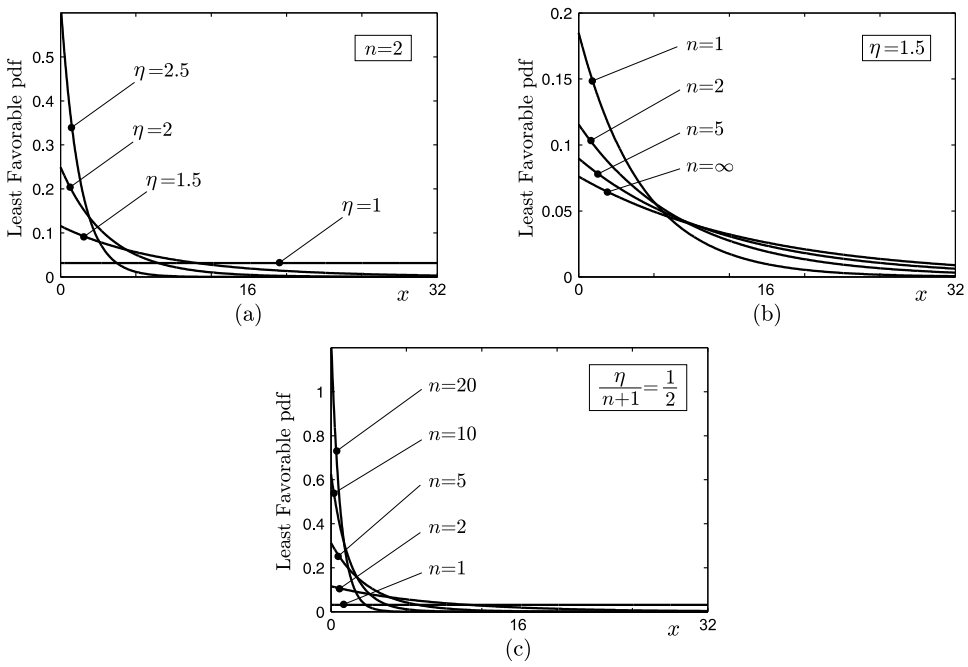

Figure 4.1: Form of least favorable pdf $f_1^\star(x)$: a) number of legitimate nodes $n = 2$, 1 malicious node and gain factor $\eta = 1, 1.5, 2, 2.5$; b) gain factor $\eta = 1.5$ and number of legitimate nodes $n = 1, 2, 5, \infty$; c) absolute gain $\frac{\eta}{n+1} = \frac{1}{2}$ and number of legitimate nodes $n = 1, 2, 5, 10, 20$.

Fig. 4.1a depicts the form of the density for $n = 2$ legitimate nodes competing with one attacker for values of the gain factor $\eta = 1, 1.5, 2, 2.5$. We observe that as $\eta \to 3$ (the maximum possible gain for $n = 2$) the density tends to a Dirac delta function at $x = 0$ which corresponds to DoS attack, representing the extreme case of misbehavior where the attacker consumes all the available resources.

In Fig. 4.1b we fix the gain factor to $\eta = 1.5$ (the attacker enjoys 50% more access to the channel than a legitimate node) and plot $f_1^\star(x)$ for number of legitimate nodes $n = 1, 2, 5, \infty$. We observe that as the number n of legitimate nodes increases, the attacker converges towards a less aggressive strategy. The interesting point is that the least favorable pdf converges very quickly to a limiting function as the number of legitimate nodes increases. This example confirms that it is possible to detect an attacker even if there is a large number of legitimate nodes present, since the attacker in order to maintain his relative gain must use a pdf which differs from the nominal uniform.

Instead of fixing the attacker's gain relatively to the gain of a legitimate node, we now examine what happens when the attacker follows a more aggressive policy and demands channel access for a *constant* percentage of time, regardless of the number of existing nodes. To achieve this goal, the gain factor η must be selected so that $\eta\frac{1}{n+1}$ is a constant. Fig. 4.1c depicts this specific scenario for $\frac{\eta}{n+1} = \frac{1}{2}$. In other words, the attacker has access to the channel 50% of the time, regardless of the number of competing nodes. We can see that when $n = 1$ the attacker behaves legitimately, but as the number n of legitimate nodes increases, the attacker quickly resorts to the strategies that are of DoS type in order to maintain this fixed access percentage. This is evident from the fact that the least favorable pdf tends to accumulate all its probability mass at small back-off values.

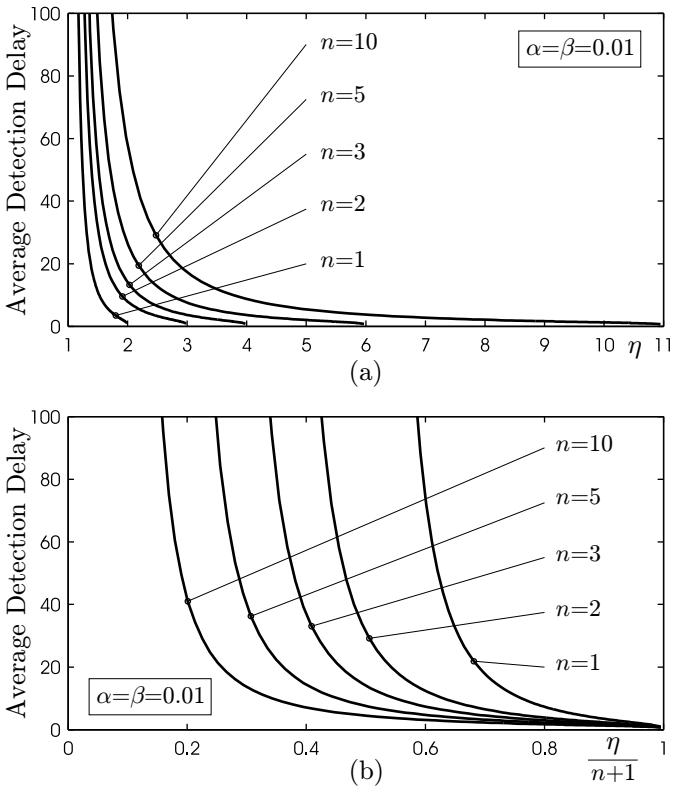

Figure 4.2: Average Detection Delay $\mathbb{E}[N]$ as a function of (a) gain factor η; (b) absolute gain $\frac{\eta}{n+1}$ for $\alpha = \beta = 0.01$

In order to obtain some intuition from our results, we consider the case of one attacker competing with $n \geq 1$ legitimate nodes. In Fig. 4.2a we depict the average required number of observation samples as a function of the parameter η. We fix the probability of detection and the probability of false alarm to 0.99 and 0.01 respectively and measure the Average Detection Delay $\mathbb{E}[N]$ for $1 < \eta < n + 1$. The graph shows that low values of η prolong the detection procedure, since in that case the attacker does not deviate significantly from the protocol. On the other hand, a large η signifies a class of increasingly aggressive attacks for which the detection is achieved with very small delay. Due to the fact that the value of η is limited with the number of legitimate nodes, we cannot compare the performance of the system for different values of n unless the absolute

gain $\frac{\eta}{n+1}$ is used. In Fig. 4.2b we depict $\mathbb{E}[N]$ as a function of the absolute gain. It can be seen that detection becomes more efficient as the number of participating legitimate nodes increases. For example, for an absolute gain of 0.6, the IDS will require 10 times less samples to detect misbehavior for $n = 5$, than for the case of $n = 1$.

Finally, we implement the worst-case attack pdf characterized by Eq. 4.21 in the network simulator OPNET. We take advantage of the experimental setup and perform evaluation as a tradeoff between the average time to detection, T_d, and the average time to false alarm,T_{fa}, a quantity that is more meaningful and intuitive in practice. It is important to emphasize that the realistic false alarm rate used by actual intrusion detection systems is much lower than $\alpha = 0.01$ used in the theoretical analysis. We claim that this false alarm rate leads to an accurate estimate of the false alarm rates that need to be satisfied in actual anomaly detection systems [22, 23]. Due to that fact we set $\beta = 0.01$ and vary α from 10^{-2} up to 10^{-10} (where $\alpha = 10^{-10}$ corresponds to one false alarm during the whole simulation period). The back-off distribution of an optimal attacker was implemented in the network simulator OPNET and tests were performed for various levels of false alarms. The backlogged environment in OPNET was created by employing a relatively high packet arrival rate per unit of time: the results were collected for the exponential(0.01) packet arrival rate and the packet size was 2048 bytes. The results for both legitimate and malicious behavior were collected over a fixed period of 1.5min. We note that the simulations were performed with nodes that followed the standard IEEE 802.11 access protocol (with exponential back-off). The system's performance was evaluated for three values of absolute gain: 0.5, 0.6 and 0.8 and the results are presented in Fig. 4.3. By observing the tradeoff curves in Fig. 4.3 we conclude that the system's detection delay decreases significantly as the attacker's absolute gain increases. To illustrate this claim,

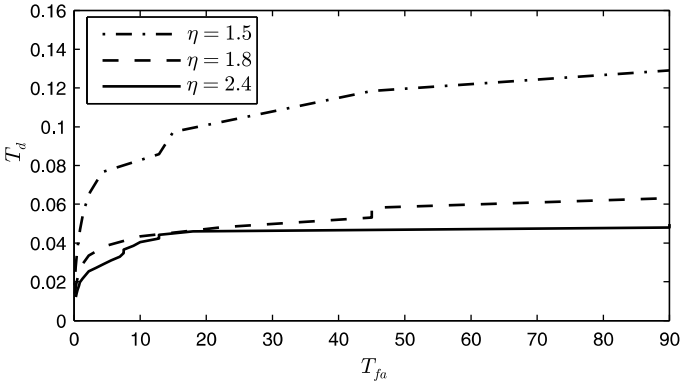

Figure 4.3: Tradeoff curve for $\frac{\eta}{n+1} = 0.5$, 0.6, 0.8 and $n = 2$.

we observe the best case system performance, i. e. one false alarm over the whole simulation period of 1.5min, and note that the detection delay for the absolute gain of 80% is approximately 3.5 times shorter than in the case when the absolute gain is 50%. This again confirms the efficiency of our proposed detection system against most aggressive worst-case optimal attacks. In order to illustrate the influence of the number of legitimate

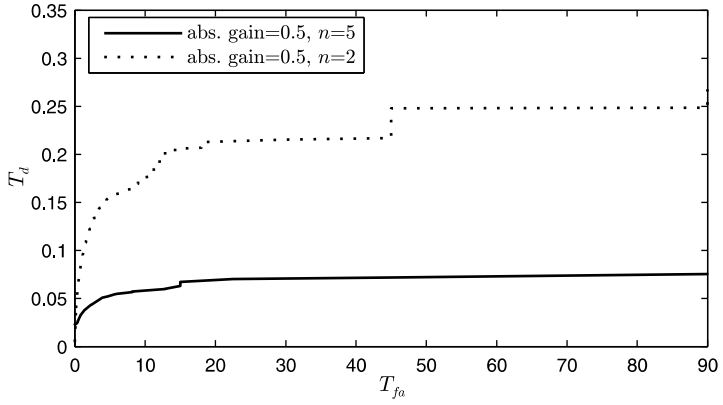

Figure 4.4: Tradeoff curve for $\frac{\eta}{n+1} = 0.5$ and $n = 2, 3$.

competing nodes on the detection time, we compare the performance of the detection system for the case when $n = 2$ and $n = 5$. In order to obtain fair comparison, we use the same value of absolute gain, $\frac{\eta}{n+1} = 0.5$. The results are presented in Fig. 4.4. As expected, all nodes experience higher number of collisions in the congested environment,

resulting in delayed detection. It is important to note that the traffic generation rate used in Fig. 4.4 is lower than the one used in Fig. 4.3. By observing the curves for $\frac{\eta}{n+1} = 0.5$ in both figures, we note that the detection system experiences larger delay when lower traffic rates are used, which is logical since all nodes access channel less frequently, generating smaller number of back-off samples within the same time interval.

4.5.1 Impact of multiple competing nodes on the performance of the optimal attacker

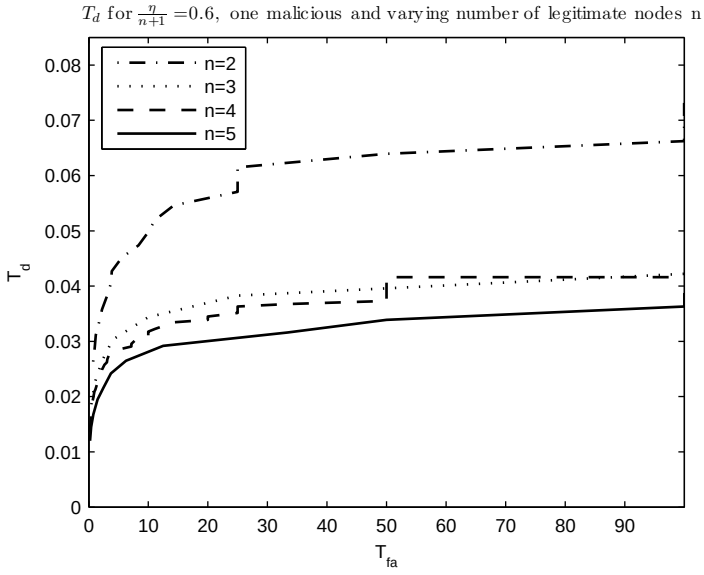

Figure 4.5: Tradeoff curve for $\frac{\eta}{n+1} = 0.6$ and $n = 2, 3, 4, 5$.

4.5.2 Performance comparison of MAC layer misbehavior detection schemes

In Sect. 4.1 we argued that (i) the performance of a sub-optimal detection scheme will be degraded in the presence of an optimal attack and (ii) an attacker that deploys a sub-optimal strategy (i.e. strategy that is constructed against a specific detection system) will be detected with substantially smaller detection delay than the optimal one when a

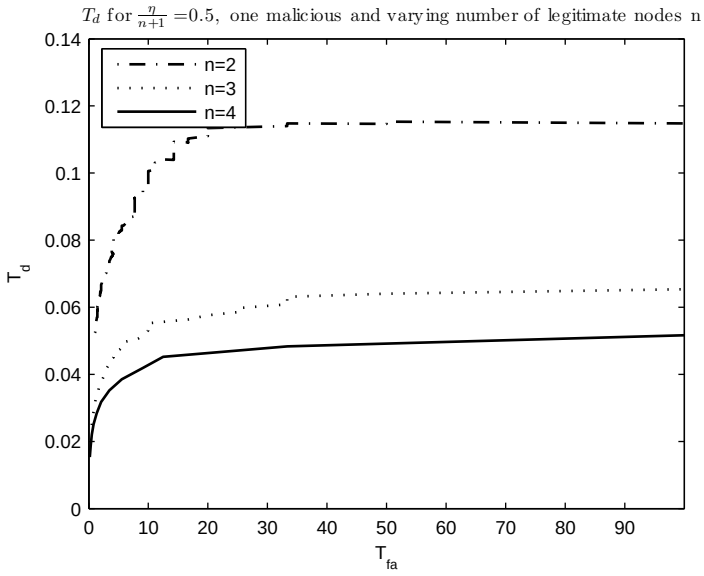

Figure 4.6: Tradeoff curve for $\frac{\eta}{n+1} = 0.5$ and $n = 2, 3, 4$.

quickest detection scheme (i.e. optimal detection scheme) is deployed. We now confirm the above statement by experimental evaluation. In particular, as an example of a sub-optimal detection scheme we analyze the performance of DOMINO [2] and compare its performance against the optimal, SPRT-based detection scheme, in the presence of optimal and sub-optimal attacks.

The back-off distribution of the optimal attacker was implemented in the network simulator OPNET and tests were performed for various levels of false alarms. The results presented in this section correspond to the scenario consisting of two legitimate and one selfish node competing for channel access. It is important to mention that the resulting performance comparison of DOMINO and SPRT does not change for any number of competing nodes. SPRT always exhibits the best performance.

In order to demonstrate the performance of all detection schemes, we choose to present the results for the scenario where the attacker attempts to access channel for 60% of the time (as opposed to 33% if it was behaving legitimately). The backlogged

environment in OPNET was created by employing a relatively high packet arrival rate per unit of time: the results were collected for the exponential(0.01) packet arrival rate and the packet size was 2048 bytes. The results for both legitimate and malicious behavior were collected over a fixed period of 100s.

The evaluation was performed as a tradeoff between the average time to detection and the average time to false alarm. It is important to mention that the theoretical performance evaluation of both DOMINO and SPRT was measured in number of samples. Here, however, we take advantage of the experimental setup and measure time in number of seconds, a quantity that is more meaningful and intuitive in practice.

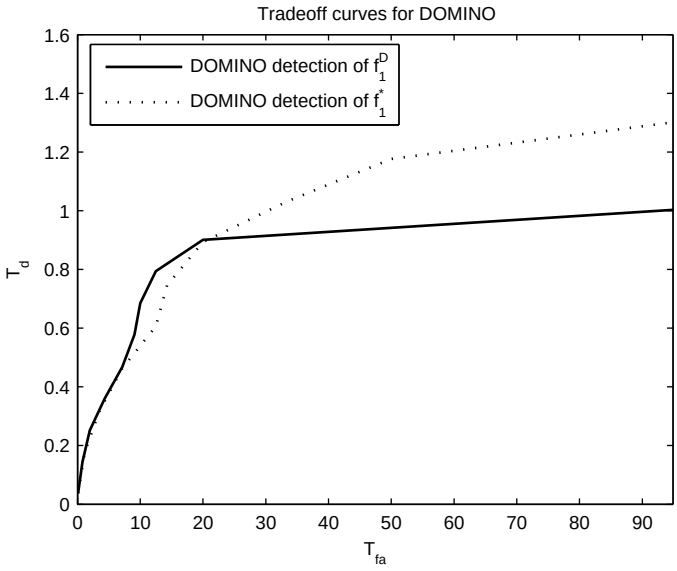

Figure 4.7: Tradeoff curves for DOMINO algorithm. One curve shows its performance when detecting an adversary that chooses f_1^D and the other is the performance when detecting an adversary that chooses f_1^*

The first step in our experimental evaluation is to show that the performance of a sub-optimal detection scheme (DOMINO) is degraded in the presence of an optimal

attack f_1^*. Fig. 4.7 provides experimental evidence confirming our predictions. Namely, DOMINO detection scheme was constructed for detection of a specific class of attacks described in [2, 24]. We denote that class of attacks with f_1^D. As it can be seen from Fig. 4.7, the detection delay of DOMINO algorithm increases up to 40% when an optimal attack strategy f_1^* is deployed. More specifically, the results presented in Fig. 4.7 illustrate the fact that an adversary using f_1^* against DOMINO can misbehave for longer periods of time without being detected than by using p_1^D. We now evaluate the performance of an

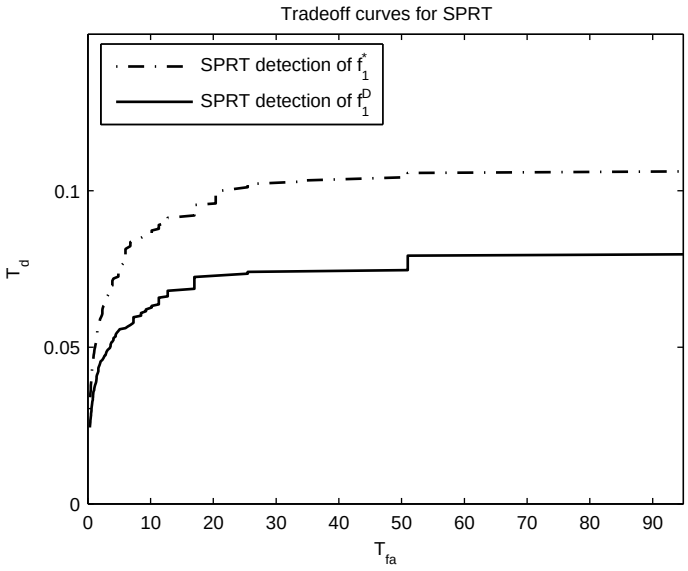

Figure 4.8: Tradeoff curves for SPRT algorithm. One curve shows its performance when detecting an adversary that chooses f_1^D and the other is the performance when detecting an adversary that chooses f_1^*.

attacker that deploys a sub-optimal strategy f_1^D (which was constructed against DOMINO detection scheme) against the quickest detection (SPRT) scheme and compare it with the performance of an attacker that deploys optimal strategy f_1^*. The results are presented in Fig. 4.8. As expected, a sub-optimal attack f_1^D is detected with a substantially smaller

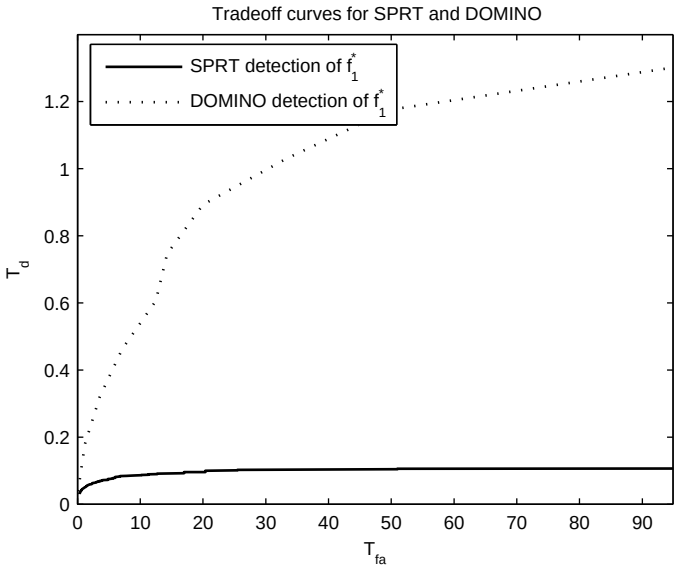

Figure 4.9: Tradeoff curves for SPRT and DOMINO algorithms.

detection delay than the optimal one when the SPRT-based detection scheme (i.e. optimal detection scheme) is deployed. More specifically, we observe that the detection delay for a sub-optimal strategy is approximately 50% smaller than the one for the optimal strategy.

We now test how the optimal (SPRT) and sub-optimal (DOMINO) detection algorithms compare to each other. Fig. 4.9 shows that SPRT significantly outperforms DOMINO in the presence of an optimal attacker. We have therefore confirmed by experimental evaluation that SPRT is the best test when the adversary selects f_1^*. Nevertheless, f_1^* can be argued to be a good adversarial strategy against any detector in the asymptotic observation case, since f_1^* is in fact minimizing the Kullback-Leibler divergence from the specified pdf f_0. The result is that the probability of detection of any algorithm (when the false alarm rate goes to zero) is upper bounded by $2^{D(f_1||f_0)}$, where $D(p||q)$ denotes the Kullback-Leibler divergence between two pdf's [25]. On the other hand, it was not possible to find any theoretical motivation for the definition of f_1^D and, hence, we claim

it is sub-optimal strategy for the given settings.

Chapter 5

Collaborative attacks

The problem treatment in Chapter 4 assumed the existence of a single intelligent adversary and the scenario where two or more protocol participants collaborate in order to degrade the performance of legitimate participants was not considered. In this chapter we extend the proposed framework to the case of $n \geq 2$ collaborating adversaries and evaluate the performance of quickest detection scheme under this setting. We show that, although extremely efficient in terms of increased detection delay and performance losses of the system, the collaborative strategies are difficult to implement due to synchronization issues that arise from random nature of the protocol and the unpredictability of wireless medium.

As we have already pointed out, we consider detection strategies in the presence of intelligent misbehaving nodes that are aware of the existence of monitoring neighboring nodes and adapt their access policies in order to avoid detection. Due to the fact that we now deal with multiple adversaries that collaborate with the common goal of disrupting network functionality, additional assumptions need to be adopted. First of all, we assume that colluding nodes collaborate by exchanging information and by taking actions that amplify each other's effects on network functionality. More specifically, we assume that each individual action can produce a desired effect only if properly coordinated with other actions. The rest of the assumptions about the adversary model are identical as in the case of a single adversary. We assume that the adversaries are *knowledgeable*, i.e. they know everything a monitoring node knows about the detection scheme and *intelligent*,

i.e. they can make inferences about the situation in the same way the monitoring nodes can. We assume that the goal of the misbehaving hosts is to choose an optimal attack strategy that minimizes the probability of detection P_D (or equivalently a strategy that maximizes the probability of avoiding detection P_M), while maximizing their gain (access to the channel).

It is now clear that two additional difficulties arise in this new setting, one at the side of the detector and one at the side of collaborating adversaries. As it has been pointed out, the adversaries need to be synchronized and consequently need to be able to communicate (exchange information) at all times in order to launch an efficient attack. On the other hand, the detector needs to be able to efficiently correlate individual actions across users in order to identify a single attack. Hence, a robust detector needs to be able to both localize and detect an ongoing collaborative attack with minimum delay.

5.1 Definition of the Uncertainty Class

Following the approach proposed in Sect. 4.3 we again adopt a min-max robust approach for defining the uncertainty class. In this setting we assume the detection system adopts the optimal detection rule $\mathcal{D}^{\star}_{12} = (N^{\star}_{12}, d^{\star}_{N_{12}})$ and the collaborating adversaries adopt the optimal access policy $f^{\star}_{12}$. The goal of the adversaries is to create a misbehavior strategy that maximizes the number of required samples for misbehavior detection delaying the detection as much as possible. On the other hand, the adversaries aim to disrupt the functionality of the network and minimize the probability of access to the legitimate protocol participants.

In order to quantify the performance of the detection scheme and the attacker, we introduce the parameter η, which defines the class of attacks of interest and specifies the

incurred relative gain of the attacker in terms of the probability of channel access. In that sense, η can be interpreted as a sensitivity parameter of the detection scheme with respect to attacks, which is determined according to the IDS requirements.

In this section we follow the same set of assumptions about the IEEE 802.11 MAC protocol as in Chapter 4. We assume that one of misbehaving collaborating nodes and a legitimate node intend to access the channel at the same time instance. In order to have a fair basis for comparison, assume that they start their back-off timers at the same time. We let the random variable X_0 stand for the back-off value of a legitimate user, hence it is uniformly distributed in $[0, W]$. Also, let the random variables X_1 and X_2 stand for the misbehaving nodes (attackers), with unknown pdf $f_{12}(x_1, x_2)$ with support $[0, W]$. The relative advantage of the attacker is quantified as the probability of accessing the channel, or equivalently the probability that its back-off is smaller than that of the legitimate node, $\Pr(X_0 < min(X_1, X_2))$.

Suppose that all nodes were legitimate. If p is the access probability of each node, then the probability of successful channel access achieves fairness for $p^* = 1/3$ for each node. Now, if two nodes collaborate, they receive gain from their attack if $\Pr(X_0 < min(X_1, X_2)) \leq \frac{\eta}{3}$. In order to quantify this, let $\eta \in [0, 1]$ and define the class of attacks

$$\mathcal{F}_\eta = \left\{ f_{12}(x_1, x_2) : \int_0^W \int_0^W \frac{\min(x_1, x_2)}{W} f_{12}(x_1, x_2)\, dx_1\, dx_2 \leq \frac{\eta}{3} \right\}. \tag{5.1}$$

where we used the fact that $f_0(x) = \frac{1}{W}$. The class defined by expression 5.1 includes attacks for which the incurred relative loss of the legitimate participants exceeds a certain amount (or equivalently, incurred relative gain exceeds a certain amount). The class $\mathcal{F}_\eta$ is the uncertainty class of the robust approach and the parameter η is a tunable parameter. By defining the class $\mathcal{F}_\eta$, we imply that the detection scheme should focus on attacks with larger impact to system performance and not on small-scale or short-term attacks.

5.2 Derivation of the worst-case attack for n=2 adversaries

By following the approach from Chap. 4, we assume that hypothesis $\mathbf{H}_0$ concerns legitimate operation and thus the corresponding pdf $f_0(x)$, is the uniform one. Hypothesis $\mathbf{H}_1$ corresponds to misbehavior with unknown pdf $f_{12}(x_1, x_2) \in \mathcal{F}_\eta$. Since the objective of a detection rule is to minimize the number of observation samples N_{12} needed for deriving a decision regarding the existence or not of misbehavior, we adopt the SPRT as our optimal detection rule $\mathcal{D}_c^\star$ for detection of the worst-case attack $f_{12}^\star$. The performance of the optimal detection scheme is again quantified by the average number of samples $\mathbb{E}_{12}[N]$ needed until a decision is reached, where the average is taken with respect to the distribution $f_{12}(x_1, x_2)$ employed by the attacker. This expectation is a function of the adopted detection rule $\mathcal{D}_{12}$ and the pdf $f_{12}(x_1, x_2)$

$$\mathbb{E}_{12}[N] = \phi(\mathcal{D}_{12}, f_{12}(x_1, x_2)). \tag{5.2}$$

From Eq.(4.9) the average number of samples is

$$\mathbb{E}_{12}[N] = \frac{\mathbb{E}[S_N]}{\mathbb{E}[\Lambda]} = \frac{C}{\mathbb{E}_{12}\left[\ln \frac{f_{12}(X_1,X_2)}{f_0(X_1)f_0(X_2)}\right]} \tag{5.3}$$

where $f_0(x_i) = \frac{1}{W}$ (denotes the uniform distribution of normal operation), $C = aP_D + b(1 - P_D)$, and the expectation in the denominator is with respect to the unknown attack distribution f_{12}. In the context of the minmax robust detection framework, the goal is to optimize the performance of the detection scheme in the presence of the worst-case attack, that is, solve the following min-max problem

$$\inf_{\mathcal{D}_{12} \in \mathcal{I}_{\alpha,\beta}} \sup_{f_{12} \in \mathcal{F}_\eta} \phi(\mathcal{D}_{12}, f_{12}). \tag{5.4}$$

Since C from Eq. (5.3) is a constant, the solution of the above min-max problem reduces to:

$$\min_{f_{12}} \int_0^W \int_0^W f_{12}(x_1 x_2) \ln f_{12}(x_1 x_2) dx_1 dx_2 \tag{5.5}$$

subject to the constraints,

$$\int_0^W \int_0^W f_{12}(x_1 x_2) dx_1 dx_2 = 1 \tag{5.6}$$

and

$$\int_0^W \int_0^W \frac{\min(x_1 x_2)}{W} f_{12}(x_1 x_2) dx_1 dx_2 \leq \frac{\eta}{3} \tag{5.7}$$

The first constraint enforces the fact that f_{12} is a pdf and the second one holds due to the fact that $f_{12} \in \mathcal{F}_\eta$. By applying the Karush-Kuhn-Tucker (KKT) conditions, we find that the function $f_{12}^*(x_1, x_2)$ has the following form:

$$f_{12}^*(x_1, x_2) = e^{-1-\lambda} e^{-\mu \min(x_1, x_2)/W} \tag{5.8}$$

where λ and μ are the Lagrange multipliers that correspond to the constraints and are functions of W and η only. These can be obtained by the system of equations:

$$\begin{aligned} \frac{2W^2(e^{-\mu} + \mu - 1)}{\mu^2} &= e^{1+\lambda} \\ \frac{2W^2}{\mu^3}(2e^{-\mu} + \mu e^{-\mu} - 2 + \mu) &= \frac{\eta}{3} e^{1+\lambda} \end{aligned} \tag{5.9}$$

For the purpose of illustrating the actual effects of collaborating adversaries on the performance of the system we now observe two collaborating adversaries under the assumption that they act as a single adversary. Fig. 5.1 depicts the form of the density f_{12} of two collaborating attackers for various values of the parameter η. Again, as in Chap. 4, we observe that as $\eta \to 1$, the density tends to a Dirac delta function at $x = 0$, which corresponds to DoS attack. However, unlike in the case of a single attacker, the detection

system does not observe the pdf from the Fig. 5.1 until the stage of localization. The IDS, or more specifically the observers, see each adversary as a separate entity, therefore observing significantly milder strategy than the one that is actually being used against the system, as we will see in Sect. 5.4.

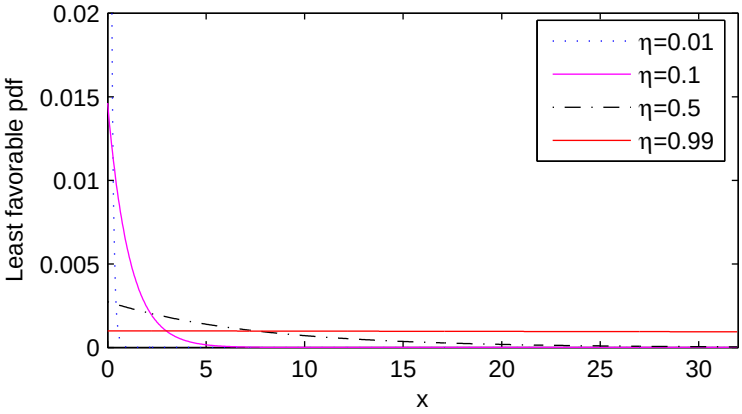

Figure 5.1: The optimal pdf of colluding adversaries.

Interestingly, Eq. (5.8) shows that the worst-case attack distribution f_{12}^{*} again takes exponential form, just like in the case of a single adversary. We now need to prove that the pair $\mathcal{D}_{12}^{\star}, f_{12}^{\star}$ is a saddle point of the function ϕ, where the saddle point was defined by Def. 4.19. The right hand side of the inequality suggests that $\mathcal{D}_{12}^{\star}$ must be the optimum detection structure for $f_{12}^{\star}(x_1, x_2)$. Indeed, this is how $\mathcal{D}_{12}^{\star}$ is defined, since it is selected as the SPRT test that optimally discriminates between $f_{12}^{\star}$ and the uniform pdf f_0. This proves the right hand side of the saddle point inequality. Following the identical approach as in the case of Theorem 4.4.2, we prove that $\phi(\mathcal{D}_{12}^{\star}, f_{12}^{\star}) \geq \phi(\mathcal{D}_{12}^{\star}, f_{12})$ for all $f_{12} \in \mathcal{F}_{\eta}$, therefore proving the left inequality in (4.19). We have now shown that the pair $(\mathcal{D}_{12}^{\star}, f_{12}^{\star})$, where $\mathcal{D}_{12}^{\star}$ is SPRT and $f_{12}^{\star}(x_1, x_2)$ is the exponential density constitute a saddle point of ϕ. This means that the min-max equality holds and we can interchange the order of min

and sup in the optimization problem above [21]. Then, the problem

$$\max_{f_{12}\in\mathcal{F}_\eta} \min_{d_{12}\in\mathcal{D}_{12}} \phi(d_{12}, f_{12}) \tag{5.10}$$

has the same solution with (4.18).

As was mentioned above, the min-max robust detection approach captures the case of an intelligent adaptive attacker. The SPRT algorithm is part of the intrusion detection system module that resides at an observer node. In other words, the observer (and hence the system) attempts to minimize the number of required samples so as to improve its payoff in terms of improved chances for channel access. On the other hand, an intelligent attacker that knows the detection algorithm attempts to delay this decision as much as possible so as to increase his own benefit in terms of chances for channel access. The attacker aims at a strategy that causes performance degradation for other nodes by remaining undetected.

Naturally, if the attacker is intelligent and is aware of the optimal detection strategy of the given system, he can choose to misbehave until the estimated detection point and after that he can either obey the protocol rules for certain time or choose to relocate. The quickest detection framework employed in our analysis forces the adversary to follow the protocol rules or relocate as often as possible, thereby increasing the cost of launching an attack.

5.3 Derivation of the worst-case attack for $n > 2$ adversaries

In order to proceed towards derivation of the worst-case attack for the case of $n > 2$ adversaries we first redefine the uncertainty class described by Eq. 5.1. In the setup with more than 2 collaborating adversaries, the relative advantage of the adversaries is again quantified as the probability of accessing the channel, or equivalently the probability that

their back-off is smaller than that of the legitimate node.

Suppose that we observe the behavior of $n+1$ legitimate nodes, where $n > 1$. If p is the access probability of each node, then the probability of successful channel access achieves fairness for $p^* = \frac{1}{n+1}$ for each node. Now, if n nodes collaborate, they receive gain from their attack if $\Pr(X_0 < min(X_1, \ldots, X_n)) \leq \frac{\eta}{n+1}$. In order to quantify this, let $\eta \in [0, 1]$ and define the class of attacks for $f_{1\ldots n}(x_1, \ldots, x_n)$

$$\mathcal{F}_\eta = \left\{ \int_0^W \ldots \int_0^W \frac{\min(x_1, \ldots, x_n)}{W} f_{1\ldots n}(x_1, \ldots, x_n)\, dx_1 \ldots\, dx_n \leq \frac{\eta}{n+1} \right\}. \tag{5.11}$$

Assuming that the SPRT is used, we again seek an attack distribution f^* such that $\phi(d^*, f^*) \geq \phi(d^*, f)$ for all other attacks $f \in \mathcal{F}_\eta$.

From Eq.(4.9) the average number of samples is

$$\mathbb{E}[N] = \frac{\mathbb{E}[S_N]}{\mathbb{E}[\Lambda]} = \frac{C}{\mathbb{E}_{1\ldots n}\left[\ln \frac{f_{1\ldots n}(X_1,\ldots,X_n)}{f_0(X_1)\ldots f_0(X_n)}\right]} \tag{5.12}$$

where $f_0(x_i) = 1/W$ (denotes the uniform distribution of normal operation), $C = aP_D + b(1 - P_D)$, and the expectation in the denominator is with respect to the unknown attack distribution f. Since C is a constant, the problem of finding the attack that maximizes the required number of observations reduces to the problem:

$$\min_{f_{1\ldots n}} \int_0^W \ldots \int_0^W f_{1\ldots n}(x_1 \ldots x_n) \ln f_{1\ldots n}(x_1 \ldots x_n) dx_1 \ldots dx_n \tag{5.13}$$

subject to the constraints,

$$\int_0^W \ldots \int_0^W f_{1\ldots n}(x_1 \ldots x_n) dx_1 \ldots dx_n = 1 \tag{5.14}$$

$$\int_0^W \ldots \int_0^W \frac{\min(x_1 \ldots x_n)}{W} f_{1\ldots n}(x_1 \ldots x_n) dx_1 \ldots dx_n \leq \frac{\eta}{n+1} \tag{5.15}$$

The first constraint enforces the fact that f is a pdf and the second one holds due to the fact that $f \in \mathcal{F}_\eta$. By applying the Karush-Kuhn-Tucker (KKT) conditions, we find that the function $f^*_{1\ldots n}(x_1, \ldots, x_n)$ has the following form:

$$f^*_{1\ldots n}(x_1, \ldots, x_n) = e^{-1-\lambda} e^{-\mu \min(x_1,\ldots,x_n)/W} \tag{5.16}$$

where λ and μ are the Lagrange multipliers that correspond to the constraints and are functions of W and η only. These can be obtained by numerically solving the above constraints.

Again, Eq. (5.16) shows that the worst-case attack distribution $f^*_{1\ldots n}$ again takes exponential form, just like in the case of a single adversary. Following the identical approach as in the case of Theorem 4.4.2, we prove that $\phi(d^*, f^*) \geq \phi(d^*, f)$ for all $f \in \mathcal{F}_\eta$, therefore proving the left inequality in (4.19). We have now shown that the pair (d^*, f^*), where d^* is SPRT and $f^*(x_1, \ldots, x_n)$ is the exponential density constitute a saddle point of ϕ. This means that the so-called min-max equality holds and we can interchange the order of min and sup in the optimization problem above [21]. Then, the problem

$$\max_{f \in \mathcal{F}_\eta} \min_{d \in \mathcal{D}} \phi(d, f) \tag{5.17}$$

has the same solution with (4.18).

5.4 Experimental Results

We now proceed to experimental evaluation of the analyzed scenario. In order to correctly capture the behavior of colluding attackers and evaluate the advantage over the non-colluding strategies, we compare the performance of a *single optimal attacker* from [26] with the performance of colluding attackers who generate the optimal back-off sequence according to the pdf f^*_{12}. The detection schemes employed in [2, 26] use different metrics

to evaluate the performance of attackers and the detection algorithms. We believe that the performance of the detection algorithms is better captured by employing the expected time before detection $\mathbb{E}[T_D]$ and the average time between false alarms $\mathbb{E}[T_{FA}]$ instead of detection delay $\mathbb{E}[N]$, used in [26], or throughput, used in [2], as the evaluation parameters.

It is important to note that the chosen values of the parameter a in all the experiments are small and vary from 10^{-2} to 10^{-10}. We claim that this represents an accurate estimate of the false alarm rates that need to be satisfied in actual anomaly detection systems [22, 23], a fact that was not taken into account in the evaluation of previously proposed systems.

The back-off distribution of both optimal single attacker from [26] and optimal colluding attackers from Eq. (5.8) was implemented in the network simulator Opnet and tests were performed for various levels of false alarms and various values of the parameter η. The sequence of optimal back-off values was then exported to Matlab and the quickest detection tests were performed on the given sets of data.

We first analyze the effectiveness of the quickest detection scheme against colluding attackers with different levels of aggressiveness (different values of η). We chose 3 different values of η: 0.3, 0.6 and 0.9, where η=1 represents the scenario where all nodes follow the rules of the protocol. The results of the above strategies are presented in Fig. 5.2. As expected, the detection delay increases with η and is almost identical for higher values of η. This re-confirms the effectiveness of the optimal SPRT-based detection scheme for detection of nodes that significantly deviate from the protocol rules. However, it is important to quantify the advantage of the colluding scheme over a single attacker in order to justify employment of an additional attacker. It is to be expected that the colluding nodes will experience larger detection delays, depending on the η they choose for their

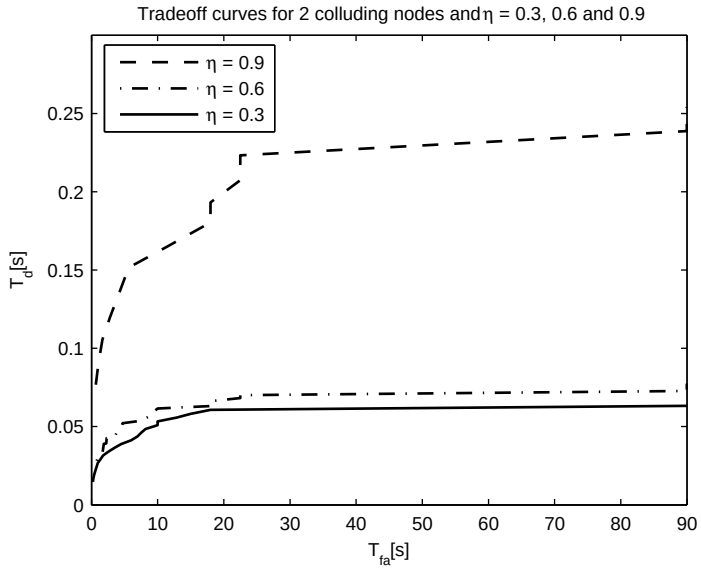

Figure 5.2: Tradeoff curves for 2 colluding nodes and $\eta = 0.3$, 0.6 and 0.9.

access strategy. Fig. 5.3 compares the performance of colluding and single attackers for η=0.6. It is important to mention that the crucial advantage of colluding nodes is that

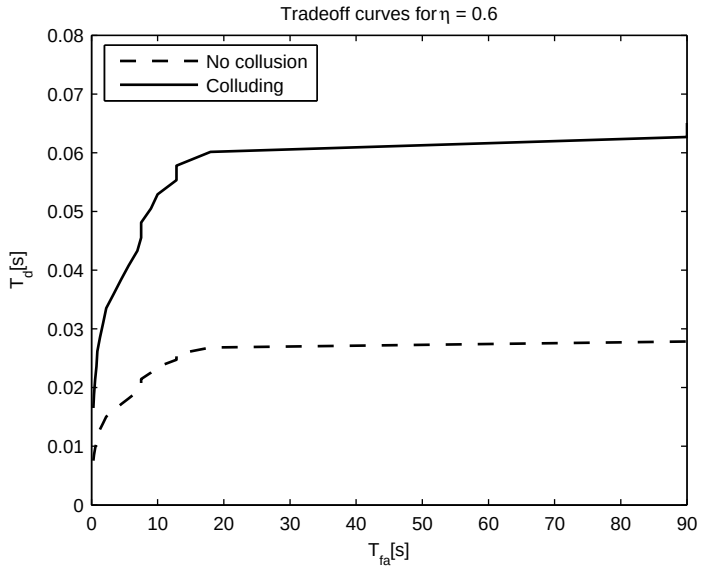

Figure 5.3: Tradeoff curves for $\eta = 0.6$: detection times for colluding nodes are up to 2 times longer than for a single node with identical strategy.

the detection system is not aware of collaboration among the attackers and performs detection on a *single* malicious node. As expected, the detection delay for colluding nodes is approximately 2 times higher than for a single attacker. In order to illustrate the effect of η on the detection delay, we now perform the same test with η=0.9. As it can be seen from Fig. 5.4, the detection delay for colluding nodes increases even further as the aggressiveness of the attackers decreases. Finally, we fix η=0.9 for the case of a single attacker and

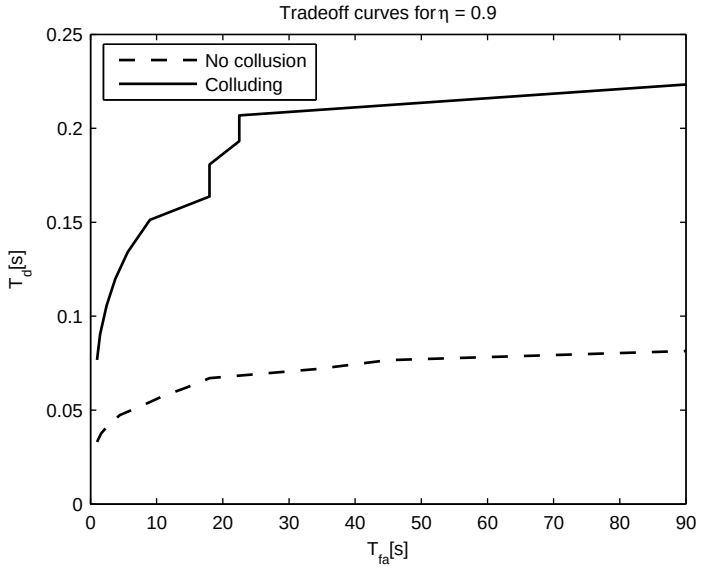

Figure 5.4: Tradeoff curves for $\eta = 0.9$: detection times for colluding nodes are up to 3 times longer than for a single node with identical strategy.

attempt to find the corresponding value of η for the case of colluding nodes that will have the same detection delay. As it can be seen from Fig. 5.5, the corresponding value of η is approximately 0.4, which represents a significant gain (recall that η=0 represents the DoS attack) and enables colluding attackers to significantly deviate from the protocol rules with the detection delay equivalent to the one when there is almost no misbehavior. Finally, it is important to address the issue of overhead of the proposed detection algorithm. The

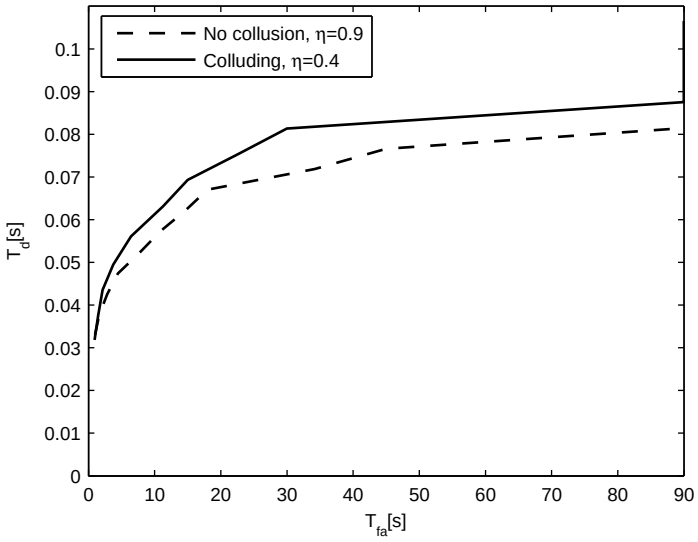

Figure 5.5: Tradeoff curves for $\eta = 0.9$ (single attacker) and $\eta = 0.4$ (colluding attackers).

SPRT is highly efficient since no observation vectors need to be stored. The only storage complexity is the one needed for the pdfs f_1 and f_0, the thresholds "a" and "b" and the current statistic S_n. In addition to that, the SPRT algorithm is also time-efficient, since in order to compute the log-likelihood we only need to compute the ratio of two functions (f_0 and f_1, which are very simple to evaluate) and add this value to the current statistic S_n. Therefore, the overhead of the proposed algorithm is low and can be obtained by adding the two previously mentioned values.

Chapter 6

Impact of interference on the performance of optimal detection schemes

6.1 Overview

In Chap. 3, Sect. 3.3 we briefly introduced the importance of considering impact of interference on the performance of detection schemes. Before proceeding to analytical evaluation, we analyze the behavior of optimal detection scheme presented in Chap. 4 in the presence of interference. We assume that (i) the main source of interference are concurrent transmissions of neighboring nodes, (ii) the effects of interference are observed in terms of reduced Signal-to-Interference and Noise Ratio (SINR) and (iii) reduction in SINR results in missed observations (RTS or CTS packets) at the observers side. Depending

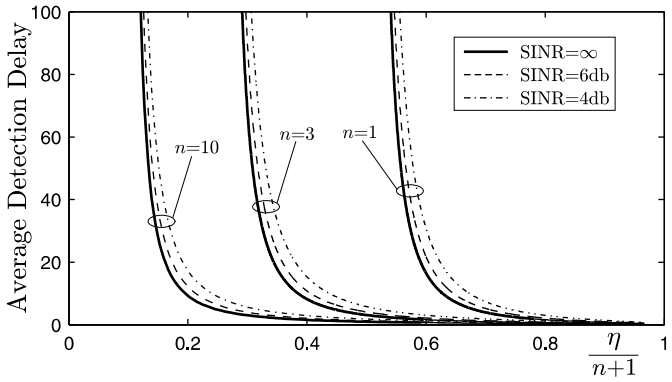

Figure 6.1: Average detection delay for different values of SINR and n=1, 3, 10

on interference conditions, a percentage of the back-off samples collected by the observer nodes are corrupted (not measured correctly). In that case, the most convenient measure of performance is the Packet Error Rate (PER) of RTS/CTS messages. In this case, PER indicates the amount of additional measurements required for reaching a decision,

depending on whether the observer node resides within range of the attacker (RTS observations) or the receiver (CTS observations) of the attack. Fig. 6.1 shows the average required number of samples needed for detection of an optimal attacker for different intensity of interference, with respect to the absolute gain $\frac{\eta}{n+1}$. System performance is evaluated for $n = 1, 3$ and 10. For large values of P_d it can be observed that intense interference conditions (reflected in the SINR values of 3-4 dB) can increase the number of required samples by $85\% - 120\%$ compared to the case when no interference is present. It is also worth mentioning that as the aggressiveness of an attacker increases, the number of samples needed for detection decreases, regardless of the SINR values. However, in real IDSs, the P_{FA} needs to be much lower than the one used in most theoretical analysis in current literature [23, 22]. As a consequence, the detection delay in the presence of intense interference is still significantly higher than in conditions without interference, even for more aggressive attacks. This will be demonstrated in the remainder of this chapter. Finally, we observe that for SINR$> 8dB$, the performance of the detection scheme is not affected significantly by interference due to the fact that most RTS/CTS messages are received correctly. Hence, interference can be viewed as an aid to the adversary in the sense that it provides him additional benefit by prolonging detection. Consequently, this leads to raising the cost of detection. Due to different lengths of RTS and CTS messages, the number of samples needed to detect misbehavior is lower when CTS messages are used in measurements. For example, for SINR values of 3-4 dB, $\alpha = \beta = 0.01$, we observe an increase of $85 - 100\%$ in the number of required samples compared to that with no interference. Therefore, when assigning observer roles to nodes, emphasis should be given to those nodes that are located within range of the receiver. The amount of additional measurements needed for detection expressed in the form of PER for different values of

SINR is presented in Fig. 6.2.

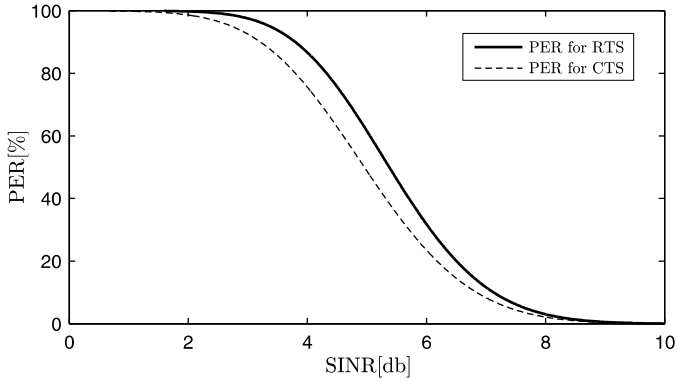

Figure 6.2: PER[%] as a function of SINR for RTS and CTS messages

It can be observed from Fig. 6.1 and Fig. 6.2 that as a result of interference the observer may not hear RTS or CTS messages, which results in a corrupted observation sequence and detection delay. Given the fact that timely detection of attacks is of crucial importance in wireless environments, this represents a significant obstacle. In the remainder of this section we will perform detailed analysis of possible interference scenarios and their impact on the performance of detection schemes. We will analyze the worst-case performance of the detection scheme and establish performance bounds.

6.2 Problem setup

Before proceeding towards a formal analysis of the interference problem at the observers side, we first address the issue at the attackers side. In this work we assume that the goal of the attacker is to deny medium access to legitimate protocol participants. The attacker achieves this by adopting strategies that give him higher access probability and consequently increase his own gain. In the presence of interference we assume the attacker attempts to access the medium with the same strategy that was presented in Chap. 4. However, due to low SINR, it may miss CTS message from the receiver and not send any

data. We now note that, although the adversary does not gain access to the medium, in this case the main goal is achieved: (i) the adversary transmits RTS message and silences his neighborhood for the duration of the potential data transmission and (ii) the receiver sends CTS message which silences his own neighborhood, just as if the whole exchange of data were successful. Hence, the adversary, whose goal is to deny access to legitimate participants, still achieves his goal in the presence of interference and need not change his own strategy. On the other hand, the presence of errors at the detector's side will result in delayed detection and needs to be considered. In this scenario, we assume that the detector experiences interference and fails to detect a new control message sent by an attacker with probability p_2. As a consequence, the detector will no longer observe the original attacker's strategy $f_1^\star(x)$. Instead, it will observe the new back-off distribution, $\tilde{f}_1^\star(x)$ which is generated according to the following set of rules:

1. The real back-off x_1 is observed with probability $1 - p_2$;
2. back-off $x_1 + x_2$ is observed with probability $p_2(1 - p_2)$ (one transmission of the attacker is not observed);
3. back-off $x_1 + x_2 + x_3$ is observed with probability $p_2^2(1 - p_2)$ (2 transmissions of the attacker are not observed);
4. ...
5. back-off $x_1 + \ldots + x_i$ is observed with probability $p_2^{i-1}(1 - p_2)$ (i-1 transmissions of the attacker are not observed);

where each back-off x_i is generated according to the original pdf $f_1^\star(x)$ given by the Eq. (4.21). For example, the new pdf generated by missing one transmission, can be calculated as $P(X_1+X_2 \leq Y)$, which is nothing else but convolution of $f_1^\star(x) * f_1^\star(x)$. In order

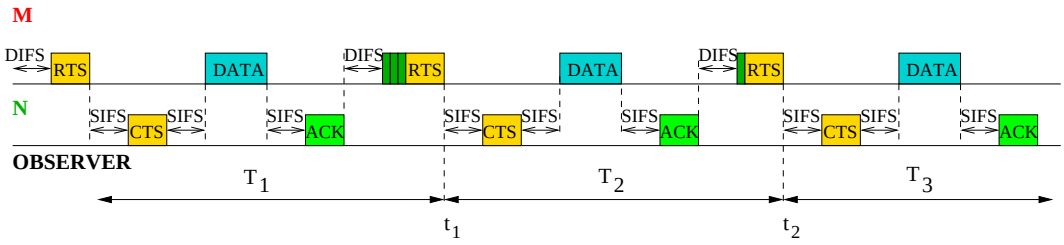

Figure 6.3: Noise diagram.

to illustrate this, we present a simple scenario in Fig. 6.3. We assume the malicious node M attempts to access the channel using the optimal pdf $f_1^\star(x)$, generating corresponding back-off values b_i. When no interference is present, an observer (detector) that is measuring back-off values of neighboring stations measures time periods between successive RTS messages, T_i and calculates the corresponding back-off values b_i (an example of such calculation is provided in Chap. 3 or in [27]). However, if the observer misses the second control message, it measures back-off $b_1 + b_2$ a time instance t_2 instead of registering two successive back-off values b_1 and b_2 at time instances t_1 and t_2 respectively. Depending on the duration of interference, the observer retrieves a corrupted back-off sequence, which results in detection delay.

6.2.1 Derivation of the worst-case attack in the presence of interference

In this section we derive an expression for the worst-case attack in the presence of interference following the framework from Chap. 4 and evaluate the performance loss of the detector in such scenarios. We assume that the adversary generates the back-off sequence using an optimal pdf $f_1^\star(x)$. As a consequence of interference, the detector observes a different back-off sequence and a different pdf of both the adversary and legitimate participant: $\tilde{f}_1^\star(x)$ and $\tilde{f}_0(x)$ respectively. Following the approach from Chap. 4, the detection delay is inversely proportional to $\int \tilde{f}_1^\star(x) \log \frac{\tilde{f}_1^\star(x)}{\tilde{f}_0(x)} dx$. However, $\tilde{f}_0(x)$ is no longer

uniform and now the problem of finding the attack that maximizes the required number of observations needed for detection reduces to the problem:

$$\min_{\tilde{f}_1^\star} \int \tilde{f}_1^\star(x) \log \frac{\tilde{f}_1^\star(x)}{\tilde{f}_0(x)} dx \tag{6.1}$$

subject to the constraints,

$$\int x f_1^\star(x) dx \leq \eta \text{ and } \int x f_1^\star(x)\, dx = 1 \tag{6.2}$$

where η has the same meaning as in Chap. 4. We now observe that the constraints from Eq. (6.2) are with respect to $f_1^\star(x)$ and the original expression in Eq. (6.1) that needs to be minimized is with respect to $\tilde{f}_1^\star(x)$. In order to derive an expression for the optimal pdf we first prove the following claim:

Claim 6.2.1. *Imposing constraints on $f_1^\star(x)$ is equivalent to imposing constraints on $\tilde{f}_1^\star(x)$, i.e. there exists a linear relation between the constraints with a known factor.*

Proof. Assuming that the probability of missing a control message sent by an attacker is p_2, the expression for $\tilde{f}_1^\star(x)$ can be expressed as:

$$\tilde{f}_1^\star(x) = (1-p_2) f_1^\star(x) + p_2(1-p_2) f_1^\star * f_1^\star(x) + p_2^2(1-p_2) f_1^\star * f_1^\star * f_1^\star(x) + \ldots \tag{6.3}$$

where "*" denotes convolution. Applying the Laplace transform to the Eq.(6.3) yields:

$$\tilde{F}_1^\star(s) = (1-p_2) F_1^\star(s) + p_2(1-p_2)(F_1^\star)^2(s) + p_2^2(1-p_2)(F_1^\star)^3(s) + \ldots \tag{6.4}$$

After applying the well known properties of the Laplace transform: F(0)=1 and $\frac{\partial F(s)}{\partial s}_{|s=0} = -\int x f(x) dx$ to the Eq. (6.4), the following expression is obtained:

$$\frac{\partial \tilde{F}_1^\star(s)}{\partial s}_{|s=0} = [(1-p_2) + 2p_2(1-p_2) + 3p_2^2(1-p_2) + \ldots] \frac{\partial F_1^\star(s)}{\partial s}_{|s=0} \tag{6.5}$$

By using $\frac{\partial F(s)}{\partial s}_{|s=0} = -\int x f(x)\, dx$ it is now easy to derive from Eq.(6.5) that

$$\int x \tilde{f}_1^\star(x)\, dx = \frac{1}{1-p_2} \int x f_1^\star(x)\, dx$$

which concludes the proof. □

We now transfer the constraints from $f_1^\star(x)$ to $\tilde{f}_1^\star(x)$ and form the following Lagrangian:

$$L(\lambda,\mu) = \int \tilde{f}_1^\star(x) \log \frac{\tilde{f}_1^\star(x)}{\tilde{f}_0(x)}\, dx + \lambda \int x\tilde{f}_1^\star(x)\, dx + \mu \int \tilde{f}_1^\star(x)\, dx \tag{6.6}$$

where μ is the Lagrange multiplier corresponding to equality constraints and λ is the Karush-Kuhn-Tucker (KKT) multiplier corresponding to the inequality constraint. The KKT conditions can be expressed as follows:

1. $\frac{\partial L}{\partial \tilde{f}_1^\star(x)} = 0$
2. $\lambda \geq 0$
3. $\lambda(\int x\tilde{f}_1^\star(x)\, dx - \eta) = 0$
4. $\int \tilde{f}_1^\star(x)\, dx = 1$
5. $\int x\tilde{f}_1^\star(x)\, dx \leq \eta$

In order to derive a result using the condition (1), we apply the method of variations to Eq.(6.6). In order to proceed further, we assume that

$$\tilde{f}_\epsilon^\star(x) = (1-\epsilon)\tilde{f}_1^\star(x) + \epsilon\delta(x)$$

which corresponds to perturbation around $\tilde{f}_1^\star(x)$. By replacing $\tilde{f}_1^\star(x)$ with $\tilde{f}_{1\epsilon}^\star(x)$ in Eq. (6.6), the criterion becomes a function of ϵ. Consequently, if $\tilde{f}_1^\star(x)$ is optimum, then the derivative with respect to ϵ at $\epsilon = 0$ must be 0. If we take the derivative and set $\epsilon = 0$, we obtain

$$\int (\delta(x) \log \frac{\tilde{f}_1^\star(x)}{\tilde{f}_0(x)} + \delta(x) + \lambda x\delta(x) + \mu\delta(x))dx = \int \delta(x)(\log \frac{\tilde{f}_1^\star(x)}{\tilde{f}_0(x)} + 1 + \lambda x + \mu)dx = 0 \tag{6.7}$$

Since the Eq.(6.7) must be valid for any density $\delta(x)$, the following expression for $\tilde{f}_1^{\star}(x)$ is obtained:

$$\log \frac{\tilde{f}_1^{\star}(x)}{\tilde{f}_0(x)} + 1 + \lambda x + \mu = 0$$

and consequently

$$\tilde{f}_1^{\star}(x) = \tilde{f}_0(x) e^{-1-\mu} e^{-\lambda x} \tag{6.8}$$

By analyzing the second KKT condition, $\lambda \geq 0$, for (i) $\lambda = 0$ and (ii) $\lambda > 0$, we conclude that $\lambda > 0$ at all times, i.e. all constraints are active. We now observe that $\tilde{f}_1^{\star}(x)$ from Eq. (6.8) is of exponential nature only if $\tilde{f}_0(x)$ is either exponential nature or constant (as in Chap. 4). Due to the fact that $f_0(x) \sim Unif[0, W]$

$$F_0(s) = \frac{1 - e^{-Ws}}{Ws}$$

It is now easy to derive the relation between $\tilde{F}_0(s)$ and $F_0(s)$ from Eq.(6.3):

$$\tilde{F}_0(s) = (1 - p_2) F_0(s)(1 + p_2 F_0(s) + p_2^2 F_0^2(s) + \ldots) = \frac{(1 - p_2) F_0(s)}{1 - p_2 F_0(s)} \tag{6.9}$$

Obviously, $\tilde{f}_0(x)$ is neither constant nor exponential, which results in $\tilde{f_1}^{\star}(x)$ not being of exponential nature any more. Consequently, the analysis from the previous chapters is no longer valid. Although the adversary still accesses the channel using the pdf $f_1^{\star}(x)$ (and denies channel access to the legitimate participants for the same amount of time)and the legitimate participants access the channel using the uniform pdf $f_0(x)$, the detector observes different access distributions for both the adversary and legitimate participants, which results in different detection delay. We now propose a framework for establishing performance bounds of the adversary and the IDS in the presence of interference.

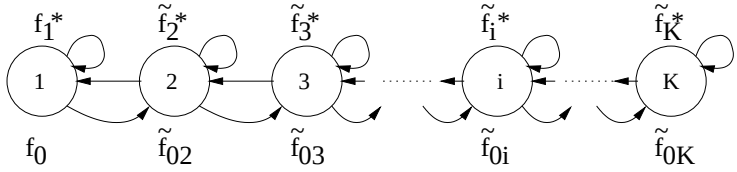

Figure 6.4: Markov Chain representation of the system. Each state corresponds to a different SINR level.

6.3 FSM for SINR variation

As it has previously been pointed out, the detector will miss an observation with certain probability, which consequently results in erroneous back-off observations. In this analysis we adopt the approach from [28] and apply it to the case of the IEEE 802.11 noisy environment.

6.3.1 System model

Let $\mathcal{S} = s_1, s_2, \ldots, s_K$ denote the state space of a Markov chain with K states. Each of the observed K states corresponds to a certain SINR level. We assume that each SINR level results in a corresponding observation error at the detector's side. More specifically, we assume that $SINR_i$ results in observing back-off $\tilde{x_i} = x_1 + \ldots + x_i$ instead of observing separate back-off values $x_1, x_2, \ldots, x_i$. Consequently, we assume that the detector observes an erroneous back-off generation pdf in each state $i \neq 1$, equal to $\tilde{f_i^\star}(x) = \underbrace{f_1^\star(x) * \ldots * f_1^\star(x)}_{i}$, where "*" denotes convolution. A system is said to be in the state s_i if the corresponding SINR values are in the range $[\Gamma_k, \Gamma_{k+1})$. Consequently, the system can be characterized with the following set of thresholds: $\vec{\Gamma} = [\Gamma_1, \ldots, \Gamma_{K+1}]$. Furthermore, let P_{ij} and π_i represent the state transition probability and the steady state

probability respectively. We assume the transitions happen between the adjacent states, resulting in $P_{k,i} = 0$ for $|k - i| > 1$. The actual values of the thresholds and transition probabilities can be obtained by simulation (i.e. in [28]) and the analysis of methods used for such performance evaluation is beyond scope of this thesis.

6.3.2 Performance analysis

In order to evaluate the performance of the IDS in the presence of interference we first return to Fig. 6.4. It has already been mentioned that in each state of the Markov chain the detector observes a different back-off sequence, i.e. in state i, the observed back-off will be $x_1 + \ldots + x_i$ and the detector will register a single (large) back-off value instead of registering i separate (small) back-off values. We now observe the worst-case scenario, when $i \to \infty$. Since $x_1, x_2, \ldots$ is a sequence of random variables which are defined on the same probability space, they share the same probability distribution and are independent, the distribution of their sum $S_i = x_1 + \ldots + x_i$ approaches the normal distribution $\mathcal{N}(i\mu, \sigma^2 i)$. Hence, for K (from Fig. 6.4) sufficiently large, the distance between the observed distributions becomes the distance between $\mathcal{N}(K\mu_1, \sigma_1^2 K)$ and $\mathcal{N}(K\mu_0, \sigma_0^2 K)$, where $\mu_i,\ \sigma_i,\ i = 0, 1$ represent the mean and variance of legitimate and adversarial distributions.

Due to the fact that the detection delay $\mathbb{E}[N]$ is inversely proportional to the KL-distance between the original and adversarial distributions, the only fact we are interested in at this point is how this distance changes as the interference level increases. For this analysis we again return to the Markov chain in Fig. 6.4. We now observe states i and $i + 1$ of the Markov chain. We observe that the corresponding distributions in states i and $i + 1$ are $\tilde{f}_i^{\star}$, $\tilde{f}_{0i}$ and $\tilde{f}_{i+1}^{\star}$, $\tilde{f}_{0(i+1)}$ respectively. Using the proof from [29] we form the following Lemma:

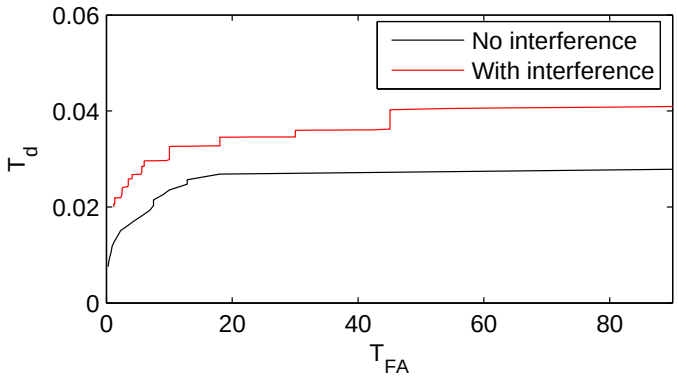

Figure 6.5: Performance comparison of the detection scheme with and without interference for $\frac{\eta}{n+1} = 0.8$.

Lemma 6.3.1. *If the distributions at states i and $i+1$ of the Markov chain are $\tilde{f}_i^{\star}$, $\tilde{f}_{0i}$ and $\tilde{f}_{i+1}^{\star}$, $\tilde{f}_{0(i+1)}$ respectively, then $D(\tilde{f}_i^{\star}||\tilde{f}_{0i}) > D(\tilde{f}_{i+1}^{\star}||\tilde{f}_{0(i+1)})$ for all $i \geq 1$.*

The above lemma states that the KL-distance between the original and the adversarial distributions *decreases* as i increases. Knowing that i increases with the increase of interference level (or decrease in the SINR level), we conclude that the KL-distance between the observed distributions decreases with the increase of interference. Since the detection delay $\mathbb{E}[N]$ is inversely proportional to the KL-distance, it is easy to see that the detection delay increases with the increase of interference level in the system. This result was expected even by intuitive analysis, since the detector observes larger back-off sequences than the actual ones, which logically leads to delay in detection (i.e. the detector believes that the adversary is accessing the channel using legitimate back-off function). In order to illustrate the impact of interference on the performance of a detection scheme, we simulate the interference scenario where the detector observes back-off $x_1 + x_2$ instead of two separate back-off values for the value of absolute gain $\frac{\eta}{n+1} = 0.8$. The results are presented in Fig. 6.5. We can see that even low interference level has significant impact

on the performance of the detector and the detection delay increases up to 50%.

We now quantify the impact of interference at the performance of the IDS in terms of P_D and P_{FA}. It is known from [25] that P_D decreases as the distance between the observed distribution decreases.As a consequence of this change, the operating point of the detection system shifts from (P_{FA_1}, P_{D_1}) to (P_{FA_2}, P_{D_2}), where $P_{D_1} > P_{D_2}$ and $P_{FA_1} > P_{FA_2}$. Consequently, with the increase in interference levels will force the IDS towards the operating point (P_{FA_k}, P_{D_k})=(0,0). The interpretation of this result is that the features of the deployed IDS are not good enough for the environment and that either more IDSs need to be deployed or another, more robust, IDS needs to be deployed.

We now observe that the presence of interference can severely affect the detector's performance. The solution to this problem is to have multiple detectors with different sensitivity levels available and depending on the requirements of the IDS and environment conditions, decide which ones to use. For example, in systems where timely decision making is of crucial importance, the deployed IDSs need to be more robust to interference (and thus more expensive [22]) and it is also advisable to deploy multiple detectors in order to minimize the probability of error in decision making. Finally, as we have seen, it is important not only to detect a quickest detection system, but the crucial step in designing a precise and robust IDS is to evaluate the environment in which it will be operating and be able to provide certain performance guarantees, such as that in environments with SINR<SINR$_c$, the system will be able to to guarantee detection delay T_{D_i} with P_{FA_i}, P_{D_i}. If the guarantees do not satisfy the needs of the system, either a more expensive detection system needs to be purchased or alternative detection methods need to be deployed.

Chapter 7

Cross-entropy minimization and its applications in intrusion detection

In [26] the problem of quickest detection of an optimal attacker was considered and the performance was evaluated based on the average detection delay. A specific class of exponential functions was found to represent the worst case attack scenario. In this work we present the first step towards building a general procedure for constructing an optimal attack scenario in the MAC layer under general set of constraints that can be adapted based on specific requirements of an IDS. To achieve this, we use the principle of minimum cross-entropy [30] which represents a general method of inference about an unknown probability density and given new information in the form of constraints on expected values. More specifically, we use the fact from [31] that given a continuous prior density and new constraints, there is only one posterior density satisfying these constraints and can be obtained by minimizing cross-entropy. Using the before mentioned facts, we show that the general expression for the worst-case optimal attack in the IEEE 802.11 MAC is of exponential nature.

7.1 Analysis of single and multi-stage attacks

The principle of minimum cross-entropy provides a general method of inference about an unknown probability density $q_f(x)$ when there exists a prior estimate and new information I about $q_f(x)$ in the form of constraints on expected values. In this notation x represents a state of a system that has $\mathbf{B}$ possible states, corresponding to possible back-off choices. In addition to that we introduce the set $\mathcal{D}$ of all probability densities

$q(x)$ on $\mathbf{B}$ such that $q(x) \geq 0$ for $x \in \mathbf{B}$. The principle states that, of all densities that satisfy the constraints, one should choose the *posterior* $q_f(x)$ with the least cross-entropy

$$H[q_f, p] = \int q(x) \log \frac{q(x)}{p(x)} dx, \tag{7.1}$$

where $p(x)$ is a *prior* estimate of $q_f(x)$ [30]. Furthermore, in [31], the authors show that the principle of minimal cross-entropy is the uniquely correct method for inductive inference when new information is given in the form of expected values. More specifically, given information in the form of constraints on expected values, there is only one distribution satisfying the constraints that can be chosen by a procedure that satisfies the consistency axioms. To apply this principle to the problem of MAC layer misbehavior detection we need to note that the goal of the attacker is to achieve maximal gain over a certain period of time, while minimizing the probability of detection P_D. We assume the existence of the set of constraints $\mathfrak{I}$ that describe the effects of the desired attack. Additionally, we assume that $\mathfrak{I}$ consists of several overlapping constraint subsets $\mathfrak{I}_1 \subset \mathfrak{I}_2 \ldots \subset \mathfrak{I}_i \ldots \subset \mathfrak{I}_K$, where $\mathfrak{I}_1$ corresponds to the DoS attack and $\mathfrak{I}_K$ corresponds to the normal behavior. More specifically, we assume that the decrease in the index i corresponds to the increase in the aggressiveness of the attackers strategy (i.e. by decreasing i we decrease the state space from which the possible back-off values can be chosen, restricting the attacker to choose from the set consisting of low back-off values). As the coefficient i increases, the constraints on the attackers pdf are relaxed and the behavior converges towards normal. Finally, we revisit the definition of constraint I representing it using the constraint set notation as $I = (q_f \in \mathfrak{I})$.

It has already been mentioned that q_f denotes the attacker's desired probability density function. The prior pdf p is an estimate of q_f prior to learning the constraints imposed upon the pdf. In our scenario, p is uniform due to the fact that every legitimate

participant in the IEEE 802.11 protocol chooses his back-off according to the uniform pdf. Given the uniform prior p and the new information in the form of constraints on the expected value,

$$\int x f_1(x) dx \leq \Im_f \tag{7.2}$$

where $\Im_f$ is the final constraint, the posterior density q_f is chosen by minimizing the cross-entropy $H[q_f, p]$ in the constraint set $\Im_f$:

$$H[q_f, p] = \min_{q' \in \Im_f} H[q', p] \tag{7.3}$$

The above equation describes the behavior of a non-adaptive intelligent attacker. Namely, the attacker chooses to diverge from the original uniform pdf to the new pdf f_1 in one step. This strategy leads to sudden changes in the wireless medium and sudden decrease in throughput. It has been shown in [27] that the above set of constraints leads to the attack strategy that is detected after observing N back-off samples, assuming that the IDS relies solely on the detection based on the number of back-off samples counted in the given time interval. However, if this detection strategy is combined with any change detection mechanism that aims to detect sudden changes in the number of dropped packets (such as *watchdog* [8]) or throughput, the existence of the attacker can be detected much earlier. We instead propose an adaptive intelligent strategy that converges from the original uniform pdf towards the desired q_f in k steps, where k is chosen according to the attacker's strategy.

The first one involves aggressive approach, where the attacker diverges from the uniform pdf by choosing a subclass of pdf's with small back-off values, resulting in the final pdf $q_f(x)$. Alternatively, the attacker may choose to converge towards the desired

pdf in 2 (or more steps). We now prove that the attacker converges towards the same final pdf, regardless of the number of steps involved if certain conditions regarding the constraints are fulfilled.

Proposition 7.1.1. *Assume the constraints I_1 and I_2 are given by $I_1 = (q_f \in \mathfrak{I}_1)$ and $I_2 = (q_f \in \mathfrak{I}_2)$ for constraint sets $\mathfrak{I}_1, \mathfrak{I}_2 \in \mathfrak{D}$. If $(p \circ I_1) \in \mathfrak{I}_2$ holds, then $q_f = p \circ I_1 = p \circ (I_1 \wedge I_2)$.*

In other words, the above proposition states that if the result of taking information I_1 into account already satisfies the constraints imposed by additional information I_2, then taking I_2 into account doesn't change the final outcome. The proof follows the same lines as the one in [31].

Proof. It is known by the definition that $(p \circ I_1) \in \mathfrak{I}_1$ holds. Additionally, by the assumption $(p \circ I_1) \in (\mathfrak{I}_1 \cap \mathfrak{I}_2)$ holds as well. By using the properties of $\circ$ operator defined in [31], the following set of equations can be derived:

$$p \circ I_1 = (p \circ I_1) \circ (I_1 \wedge I_2) = (p \circ I_1) \circ I_2 \tag{7.4}$$

Finally, using the fact that $q_f = p \circ I$ has the smallest cross-entropy of all densities in $\mathfrak{I}_1$ and consequently in $\mathfrak{I}_1 \cap \mathfrak{I}_2$. □

The correspondence to the strategy of an adaptive intelligent attacker is now obvious. The constraint I_1 corresponds to the more aggressive attack strategies that incur larger gain within a short period of time by choosing small back-off values. This strategy results in the final pdf q_f after taking into consideration the constraint I_1. If the attacker first chooses a milder strategy by choosing constraint I_2 that picks back-offs from a larger set of

values, the final pdf differs from q_f and is denoted as p_M. By knowing that constraint set I_1 already satisfies the constraints imposed by I_2 and applying the previous proposition, we arrive to the conclusion that regardless of the number of steps applied, the final pdf of the attacker will always be q_f if the constraints applied in the adaptive strategy are already included by the most aggressive strategy.

7.2 Derivation of the worst-case attack using the principle of minimum cross-entropy

We now proceed with the description of the attacker. We assume that the attacker is *intelligent*: he is aware of the existence of monitoring neighboring nodes and adapts its access policy in order to avoid detection. In addition to that, the attacker has full information about the properties of the employed IDS and its optimal detection strategy. Unlike [26], we assume that the attacker does not choose a single strategy belonging to a specified class of pdf's for the whole length of the attack. We assume that the attacker's goal is to obtain a long term gain by gradually changing his access policy. The attacker adapts to the new conditions in the system after the expiration of period Δt and updates its pdf given the new set of constraints. Therefore, the goal of the attacker is twofold:

- to diverge from the original pdf step by step by minimizing the distance between the original and new distribution
- to constantly update his access policy by relaxing the initial constraints

It has been pointed out in [26] that the derived exponential pdf had the minimal differential entropy (which is equivalent to the case of the maximum entropy when uniform priors are used) over all pdf's in the class of functions of interest. We now use the cross-

entropy principle to show that *all* optimal attacks have pdf's that belong to the class of exponential functions. Depending on the specific environmental parameters, such as P_D, P_{FA}, the aggressiveness of the attack, the attack speed etc. a specific subclass (which is again of exponential nature) that satisfies the defined constraints is derived.

We now derive the general solution for cross-entropy minimization given arbitrary constraints and illustrate the result with the specific IEEE 802.11 MAC attack defined in Chap. 4. The cross-entropy method can be outlined as follows. Given a positive prior density p and a finite set of constraints:

$$\int q(x)dx = 1, \tag{7.5}$$

$$\int f_k(x)q(x)dx = \bar{f}_k,\ k = 1, \ldots, m \tag{7.6}$$

we wish to find a density q that minimizes

$$H(q, p) = \int q(x) \log \frac{q(x)}{p(x)} dx \tag{7.7}$$

subject to the given set of constraints. By introducing Lagrange multipliers β and λ_k ($k = 1, \ldots, m$) corresponding to the constraints, the following expression for the Lagrangian is obtained:

$$\begin{aligned} L(q, \beta, \lambda_k, k = 1, \ldots, m) &= \int q(x) \log \frac{q(x)}{p(x)} dx \\ &+ \beta(\int q(x)dx - 1) \\ &+ \sum_{k=1}^{m} \lambda_k (\int f_k(x)q(x)dx - \bar{f}_k) \end{aligned}$$

Thus the condition for optimality is:

$$\log \frac{q(x)}{p(x)} + 1 + \beta + \sum_{k=1}^{m} \lambda_k f_k(x) = 0. \tag{7.8}$$

Solving for q leads to

$$q(x) = p(x) exp\left(-\lambda_0 - \sum_{k=1}^{m} \lambda_k f_k(x)\right) \tag{7.9}$$

with $\lambda_0 = \beta + 1$. The cross-entropy at the minimum can be expressed in terms of the λ_k and f_k as

$$H(q, p) = -\lambda_0 - \sum_{k=1}^{m} \lambda_k \bar{f}_k \tag{7.10}$$

It is necessary to choose λ_0 and λ_k so that all the constraints are satisfied. In the presence of the constraint (7.5) we can rewrite the remaining constraints in the form

$$\int (f_k(x) - \bar{f}_k) q(x) dx = 0 \tag{7.11}$$

If we find values for the λ_k such that

$$\int (f_i(x) - \bar{f}_i) p(x) exp(-\sum_{k=1}^{m} \lambda_k f_k(x)) dx = 0 \tag{7.12}$$

the constraint (7.11) is satisfied and (7.5) is satisfied by setting

$$\lambda_0 = \log \int p(x) exp\left(-\sum_{k=1}^{m} \lambda_k f_k(x)\right) dx. \tag{7.13}$$

If the solution of Eqn. (7.13) can be found, the values of λ_k can be found from the following relation:

$$-\frac{\partial}{\partial \lambda_k} \lambda_0 = \bar{f}_k \tag{7.14}$$

By finding all the parameters from the given set of constraints, the attacker derives the new optimal pdf, $q(x)$, that minimizes cross-entropy. Due to the fact that the attacker aims to achieve a certain gain over a long period of time, we assume that the attacker will modify his access policy by using $q(x)$ until new information about the system is collected. At that point the attacker again applies the procedure outlined in Eq. (7.5)-(7.14) and calculates the new pdf, $q_1(x)$, diverging from the original uniform distribution even further.

7.3 Optimal Attack Scenario in the MAC Layer Using the Cross-entropy Method

We now apply the results from Sect. 7.2 to the specific case of an attack in the IEEE 802.11 MAC. Due to the fact that every node in the IEEE 802.11 MAC protocol is assumed to back-off uniformly, the attacker's initial pdf $p(x)$ is assumed to be uniform over the interval $[0, W]$. The attacker wants to adapt to the conditions of the wireless environment by diverging from $p(x)$ and choosing the new pdf $q(x)$. In general, we claim that the posterior distribution q can be expressed as a function of the prior distribution and the newly obtained information $q = p \circ I$, where I stands for the known constraints on expected values and $\circ$ is an "information operator" [31].

Using the results of the attack analysis from [27] the following set of constraints is obtained for the attacker's posterior pdf $q(x)$:

$$\int_0^W q(x)\,dx = 1 \tag{7.15}$$

and

$$\mathcal{F}_\eta = \left\{ q(x) : \int_0^W xq(x)\,dx \leq C_1 \right\}, \tag{7.16}$$

where $C_1 = f(\eta, n)$. Constraint (7.15) is due to the properties of a pdf and the constraint (7.16) was obtained in [27] by bounding the long-term probability of channel access in

the scenario with one malicious node and n legal nodes. The above class $\mathcal{F}_\eta$ includes all possible attacks for which the incurred relative gain exceeds the legitimate one by $(\eta - 1) \times 100\%$. The class $\mathcal{F}_\eta$ is the uncertainty class of the robust approach and η is a tunable parameter. Using the derivations from Sect. 7.2 and a uniform prior, the following expression for the optimal pdf $q(x)$ is derived:

$$q(x) = \frac{\lambda_1}{W(e^{\lambda_1} - 1)} e^{\lambda_1 (1 - \frac{x}{W})}, \tag{7.17}$$

where the parameter λ_0 has been expressed as a function of λ_1. The parameter λ_1 is a solution to the following equation:

$$2 \left(\frac{1}{\lambda_1} - \frac{1}{e^{\lambda_1} - 1} \right) = \frac{n + 1 - \eta}{n\eta} \tag{7.18}$$

After the period of Δt the attacker takes into account new conditions in the form of the newly imposed constraints I and using $q(x)$ as a prior calculates the new optimal pdf $q_1(x)$.

Chapter 8

Cross-layer impact of optimal attacks

Under regular conditions the MAC layer has to go through multiple transmissions before detecting a link failure. The detection delay induced by additional congestion due to the presence of one or more attackers causes the feedback delay to the routing layer. We now prove that an intelligent attacker acting under the optimal strategy described with the pdf $f_1^\star(x)$ derived in Chap. 4 can cause devastating effects in the network layer if no MAC layer-based IDS is employed. Furthermore, we show that by employing a quickest detection scheme proposed in Chap. 4, the effects of such attacks can be easily prevented by isolating the detected attacker at the origin of the attack. Finally, we propose a cross-layer based cooperation scheme that is mainly oriented towards preventing propagation of local effects of MAC layer attacks.

We start our analysis by observing the scenario presented in Fig. 8.1 where selfish node accesses the channel by using an optimal attack strategy. When the back-off counter decreases to zero, the selfish node sends an RTS to node $Int2$, which replies with CTS. The RTS message silences $Node2$ which is in the wireless range of the selfish node. $Source1$ and $Node1$ are out of the range of both sender and receiver. Under the assumption that $Source1$ establishes a route to $Destination1$ through $Node1$ and $Node2$, it is reasonable to assume that $Node1$ will attempt to transmit to $Node2$ during the transmission period of selfish node (we assume that all nodes are backlogged and always have traffic to send). $Node2$ is silenced by selfish node's RTS and is not able to reply with a CTS. After a time period equal to CTS timeout, $Node1$ increases its contention window exponentially and

attempts to retransmit upon its expiration. We assume that $Node1$ constantly attempts to communicate with silenced nodes and consequently increases its contention window until it reaches its maximal value. At the same time, $Source1$ sends its regular traffic to $Node1$, increasing its backlog over time. As the misbehavior coefficient of the selfish node increases (or equivalently its back-off decreases), the selfish node gains larger percentage of channel access. Consequently, $Node2$ is silenced more frequently, increasing the backlog at $Node1$.

Assuming that each node has a finite buffer of size ν, we now derive a general expression for expected time to buffer overflow at $Node1$. Furthermore, by analyzing the scenario in Fig. 8.1 we simplify the general expression, deriving an expression applicable for analysis of effects of an optimal attack. We show by analysis an simulation that if no ID mechanism is employed in the MAC layer, the optimal MAC attack forces legitimate nodes to drop significant number of packets due to buffer overflow. If a watchdog-based or a more sophisticated reputation-based detection scheme is employed in the network layer, one or more legitimate nodes can easily be flagged as malicious due to the large number of dropped packets.

Finally, we analyze the scenario presented in Fig. 8.2 and present the effects of an optimal MAC layer attack on routes that are out of the wireless range of the attacker. We show that an intelligent attacker can easily cause route failure by attacking nodes that belong to the routes with the highest capacity. The results are presented for two routing protocols: Dynamic Source Routing Protocol (DSR) [32] and Ad hoc On Demand Distance Vector (AODV) [33].

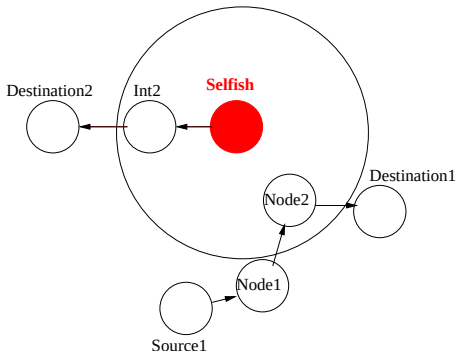

Figure 8.1: $Node2$ is silenced by the transmission of the selfish node. Consequently, $Node1$ drops large number of packets.

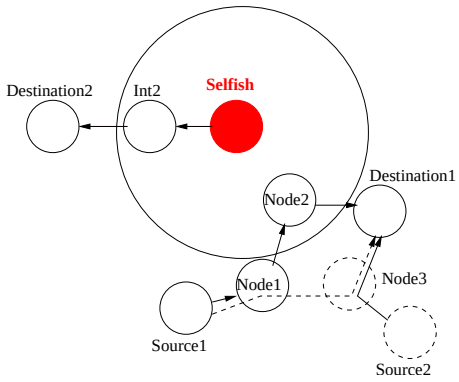

Figure 8.2: An ongoing attack in the MAC layer breaks the original route, re-routing the traffic through $Node3$.

8.1 Impact of MAC Layer Misbehavior on the Network Layer: Time to Buffer Overflow

As it has been mentioned, the secondary effect of an optimal MAC layer attack can be as devastating as the primary ones with respect to the network connectivity. If no alternative route can be found, a non-DoS optimal MAC layer attack can produce a DoS-like effects in the network layer due to the exponential nature of the IEEE 802.11 DCF back-off algorithm (such as causing buffer overflow in $Node1$ from Fig. 8.1). This section provides a comprehensive analysis of the scenario presented in Fig. 8.1, followed

by analysis of the scenario presented in Fig. 8.2 and simulation results.

We denote the incoming traffic as α_t and the outgoing traffic as β_t and assume that both processes are Poisson with parameters α and β respectively. Consequently, δ_t represents the difference between the incoming and outgoing traffic: $\delta_t = (\alpha_t - \beta_t)^+$ at time t. Equivalently, δ_t represents the increase rate of packets in the buffer over time or *backlog*. In this setup we are interested in finding the time of buffer overflow

$$T = \inf_t \{\delta_t \geq \nu\} \tag{8.1}$$

where ν denotes the buffer size. Clearly T is random, in fact it is a *stopping time*. Next we are going to develop closed form expressions for the *average-time-to-overflow*, that is, $\mathbb{E}[T]$.

If $U_1 < U_2 < U_3 < \dots$ represent the arrival times and $V_1 < V_2 < V_3 < \dots$ the departure times, a typical form of the paths of δ_t is depicted in Fig. 8.3. We observe that

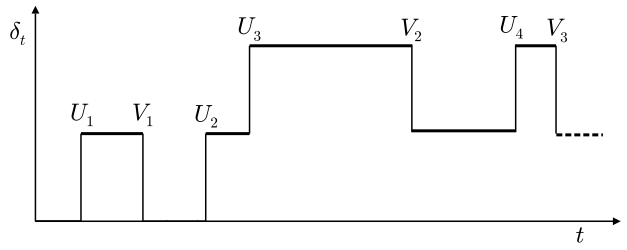

Figure 8.3: Arrival and departure times in the queue of length δ

δ_t exhibits piecewise constant paths with discontinuities of size equal to ± 1. Without loss of generality we are going to assume that these paths are *right continuous*. In order to be able to compute $\mathbb{E}[T]$ we need to study the paths of the process $g(\delta_t)$ where $g(\cdot)$ denotes a continuous nonlinear function. If $t \leq T$ is any time instant before overflow, using the right continuity of δ_t, we can write

$$g(\delta_t) - g(\delta_0) = \sum_{n=1}^{\alpha_t} g(\delta_{U_n}) - g(\delta_{U_n}\text{-}) + \sum_{n=1}^{\beta_t} g(\delta_{V_n}) - g(\delta_{V_n}\text{-}) \tag{8.2}$$

where U_n-, V_n- denote the time instant right before the n-th arrival and departure respectively. Since the discontinuities of δ_t are equal to ± 1 (depending on whether we have arrival or departure), we can write

$$g(\delta_{U_n}) = g(\delta_{U_n}\text{-} + 1), \quad \text{and} \quad g(\delta_{V_n}) = g\left((\delta_{V_n}\text{-} - 1)^+\right)$$

with the latter positive part needed because we have a departure only when the buffer is not empty. Substituting both equalities in (8.2) the following expression is obtained

$$\begin{aligned} g(\delta_t) - g(\delta_0) &= \int_0^t [g(\delta_s\text{-} + 1) - g(\delta_s\text{-})] d\alpha_s \\ &+ \int_0^t [g\left((\delta_s\text{-} - 1)^+\right) - g(\delta_s\text{-})] d\beta_s. \end{aligned}$$

Replacing in the latter expression $t = T$ and applying expectation we have

$$\begin{aligned} \mathbb{E}[g(\delta_T)] - g(\delta_0) &= \mathbb{E}\left[\int_0^T [g(\delta_s\text{-} + 1) - g(\delta_s\text{-})] d\alpha_s\right] \\ &+ \mathbb{E}\left[\int_0^T [g\left((\delta_s\text{-} - 1)^+\right) - g(\delta_s\text{-})] d\beta_s\right]. \end{aligned}$$

Because T is a stopping time and δ_s- is in the past of the time instant s, according to [34], in the previous two expectations we can replace $d\alpha_t$ with αdt and $d\beta_t$ with βdt where α, β, recall, are the corresponding rates of the two Poisson processes α_t, β_t. This leads to the following equation

$$\begin{aligned} &\mathbb{E}[g(\delta_T)] - g(\delta_0) = \\ &\qquad \mathbb{E}\Big[\int_0^T \Big\{\alpha[g(\delta_s\text{-} + 1) - g(\delta_s\text{-})] + \\ &\qquad\qquad \beta[g\left((\delta_s\text{-} - 1)^+\right) - g(\delta_s\text{-})]\Big\} ds\Big]. \end{aligned} \tag{8.3}$$

Notice now that if we select $g(\cdot)$ to satisfy the difference equation

$$\alpha[g(\delta + 1) - g(\delta)] + \beta[g\left((\delta - 1)^+\right) - g(\delta)] = -1 \tag{8.4}$$

then Eqn. (8.3) simplifies to

$$g(\delta_0) - \mathbb{E}[g(\delta_T)] = \mathbb{E}[T]. \tag{8.5}$$

Since $\delta_t \geq 0$ the function $g(\cdot)$ needs to be defined only for non-negative arguments. However, in order to avoid using the positive part in (8.4), we can extend $g(\cdot)$ to negative arguments as follows

$$g(\delta) = g(0), \text{ for } -1 \leq \delta \leq 0, \tag{8.6}$$

and this simplifies (8.4) to

$$\alpha[g(\delta+1) - g(\delta)] + \beta[g(\delta-1) - g(\delta)] = -1. \tag{8.7}$$

Furthermore, since at the time of stopping T we have a full buffer, that is, $\delta_T = \nu$ (with ν denoting the buffer size), if we impose the additional constraint

$$g(\nu) = 0, \tag{8.8}$$

and recall that $\delta_0 = 0$, from (8.5) we obtain $\mathbb{E}[T] = g(0)$.

Summarizing, we have $\mathbb{E}[T] = g(0)$ where $g(\cdot)$ is a function that satisfies the difference equation (8.7) and the two boundary conditions (8.6), (8.8). Since ν is an integer it suffices to solve (8.7) for integer values of δ meaning that (8.7) can be seen as a recurrence relation of second order. The solution to our problem can thus be easily obtained and we have

$$\mathbb{E}[T] = \begin{cases} \frac{1}{\alpha}\left\{\frac{\rho}{(1-\rho)^2}\left[\rho^\nu - 1\right] + \frac{\nu}{1-\rho}\right\} & \text{for } \alpha \neq \beta, \\ \frac{1}{\alpha}\left\{\frac{\nu+\nu^2}{2}\right\} & \text{for } \alpha = \beta, \end{cases} \tag{8.9}$$

where $\rho = \beta/\alpha$ denotes the ratio between the outgoing and incoming traffic rates. In order to examine the effects of various levels of traffic on the network stability needs to be examined. By definition, stability of the network means bounded backlogs over time, i.e. $\sup E[\delta_i(t)] < \infty$ for all nodes i in the network. We observe that whenever $\alpha > \beta$

(or $\rho < 1$) the exponential term (for large buffer size ν) is negligible as compared to the linear term and the queue needs, in the average, linear time to overflow (instability). In the opposite case $\alpha < \beta$ (or $\rho > 1$), the exponential term prevails and the average-time-to-overflow becomes exponential (stability). These observations can also be seen in Fig. 8.4 for $\rho = \beta/\alpha = 3/2$ and $\rho = \beta/\alpha = 2/3$ where we plot the average time as a function of the buffer size ν. Equivalently, $\alpha > \beta$ implies increase of backlog in the given node over a period of time and vice versa.

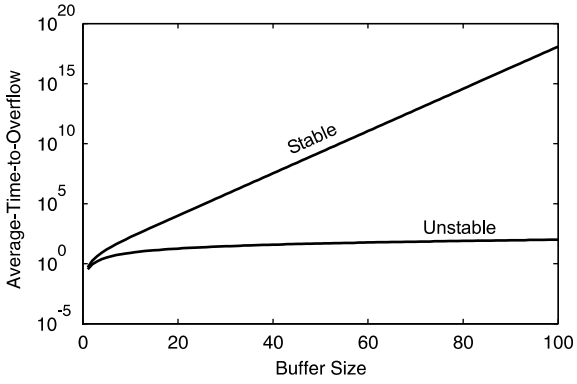

Figure 8.4: Average Time to buffer overflow for $\rho = \beta/\alpha = 3/2$ (stability) and $\rho = \beta/\alpha = 2/3$ (instability), as a function of the buffer size ν.

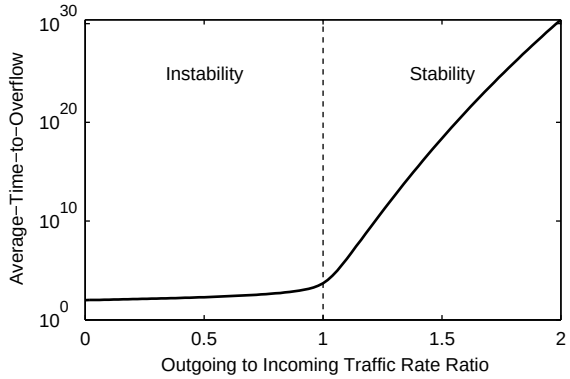

Figure 8.5: Average time to buffer overflow as a function of the traffic rate ratio $\rho = \beta/\alpha$ and buffer size $\nu = 100$.

In the stable case, we observe the extremely large average time required to overflow even for small values of the buffer size. In Fig. 8.5 we plot the average time as a function of the traffic rate ratio $\rho = \beta/\alpha$, assuming normalized incoming rate $\alpha = 1$ and buffer size $\nu = 100$. For any other value of α, according to (8.9), we simply need to divide by α.

We now return to the analysis of the scenario presented in Fig. 8.1. It has already been mentioned that with the increase of the aggressiveness of the attacker (i. e. parameter η in Eq. 4.15), the percentage of channel access for $Node2$ will accordingly decrease. Meanwhile, $Source1$ keeps generating traffic at the same rate, sending packets to $Node1$. With $Node2$ being silenced, $Node1$ has the parameter β equal to zero. Eq. 8.9 also suggests that whenever $\alpha \gg \beta$ (or $\rho \ll 1$) then $\mathbb{E}[T] \approx \frac{\nu}{\alpha}$. In order to proceed further with the discussion we need to note that finding the average time to buffer overflow $\mathbb{E}[T]$ is equivalent to finding the average time until the observed node starts losing traffic due to buffer overflow. We need to note that the scenario in which $\alpha \gg \beta$ represents the secondary effects of an optimal attack. We assume that the network has an Intrusion Detection System (IDS) implemented and that it detects a network layer attack with an average delay of Δt. Assuming that the buffer overflow happens at time t, the attack is detected at time $t_1 = t + \Delta t$. Consequently, the amount of traffic lost (TL) due to buffer overflow in node i in a network of k nodes at time t_1 can be defined as:

$$TL = \sum_{i=1}^{k} \alpha_i \left(t_1 - \frac{\nu}{\alpha_i} \right).$$

It can be easily observed from this expression that even small detection delays of the order of a couple of seconds have relatively large traffic loss as a consequence.

To illustrate the amount of lost traffic due to detection delay in the network layer we present the results of the above analysis for a single node in Fig. 8.6 for various rates of incoming traffic. As expected, the amount of lost traffic increases as the incoming traffic

rate increases. It can be easily observed that even small detection delays of the order of a couple of seconds have relatively large traffic loss as a consequence.

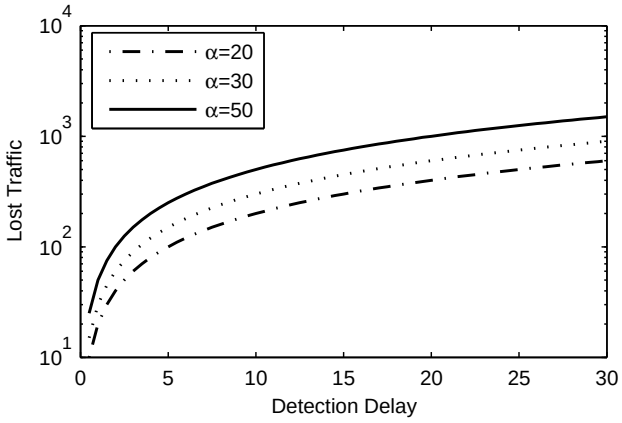

Figure 8.6: The amount of lost traffic as a function of detection delay for fixed buffer size ν=100.

8.2 Numerical Results

8.2.1 Cross-layer effects of the optimal MAC layer attacks

In order to illustrate the effects of an optimal MAC layer attack on the network layer we analyze the two scenarios presented in Fig. 8.1 and Fig. 8.2 with DSR and AODV as routing protocols. Before proceeding with the analysis, a short description of the routing protocols used in the experiments is provided.

DSR is a source routing protocol: the source knows the complete hop-by-hop route to the destination and routes are stored in node caches. It consists of two basic mechanisms: Route Discovery and Route Maintenance. When a node attempts to send a data packet to a new destination, the source node initiates a route discovery process to dynamically determine the route. Route Discovery works by flooding Route Request (RREQ) packets. RREQ packets propagate throughout the network until they are received by a node with a

route to the destination in its cache or by the destination itself. Such a node replies to the RREQ with a route reply (RREP) that is routed back to the original source. The RREQ builds up the path traversed until that moment by recording the intermediate nodes and the RREP routes itself back to the source by traversing the path backwards. If any link along a path breaks, Route Maintenance mechanism is invoked by using a Route Error (RERR) packet, resulting in removal of any route that contains that link. If the route is still needed by the source, a new route discovery process is issued.

AODV uses table-driven hop-by-hop routing. It applies a similar Route Discovery process as DSR. However, instead of using route caches, it uses routing tables to store routing information, one entry per destination. AODV relies on routing table entries to propagate a RREP back to the source and to route data packets to the destination. Furthermore, AODV uses sequence numbers (carried by all packets) to determine freshness of routing information and to prevent routing loops. One notable feature of AODV is the use of timers regarding utilization of routing table entries. Namely, a routing entry in the table may expire if it is not used recently. Moreover, a set of neighboring nodes that use this entry is also maintained; these nodes are notified through RERR packets when the next hop link breaks. This process is recursively repeated by each node, thereby effectively deleting all routes using the broken link. Upon that, a new Route Discovery process is initialized.

We now evaluate the cross-layer impact of the optimal attacker in the MAC layer. The results of the simulations are presented in Fig. 8.7 and Fig. 8.8. Fig. 8.7 analyzes the performance of $Node1$ from Fig. 8.1 as a function of ϵ with DSR and AODV as the routing protocols for two cases (i) without MAC layer-based IDS and (ii) with the MAC layer-based IDS. It is reasonable to expect that $Node2$ is denied channel access more frequently

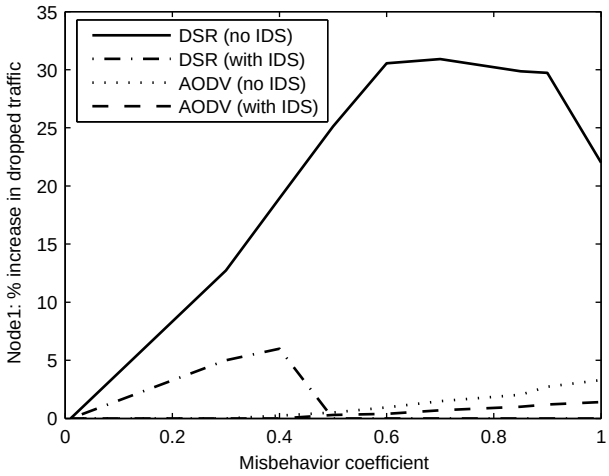

Figure 8.7: Increase in dropped traffic at $Node1$.

as the aggressiveness of the selfish node increases in the absence of a MAC layer-based IDS. Consequently, $Node1$ is disabled from forwarding packets towards the destination. After evaluating the scenario from Fig. 8.1, we note that the percentage of dropped packets at $Node1$ increases with with the aggressiveness of the attacker, since $Node2$ is denied access to the channel due to transmissions of the selfish node. We observe that the percentage increase in dropped traffic is almost linear until ϵ=0.6. However, further increase in aggressiveness of the attacker does not bring any significant benefit in terms of increase of dropped traffic at legitimate nodes. This effect is due to the operating mechanism of the DSR protocol. Namely, if the neighboring node (in this case $Node2$) does not respond to the requests of the sender for a certain period of time, the route maintenance mechanism of DSR protocol sends a RERR and a new RREQ is issued. Consequently, the contents of the buffer are flushed after the issue of RERR. Therefore, the maximum value of percentage increase in dropped traffic due to the malicious behavior in the MAC layer is bounded by (i) size of the maintenance buffer in the observed node and (ii) the route maintenance timeout value (which in this case corresponds to 40% increase in dropped traffic, even

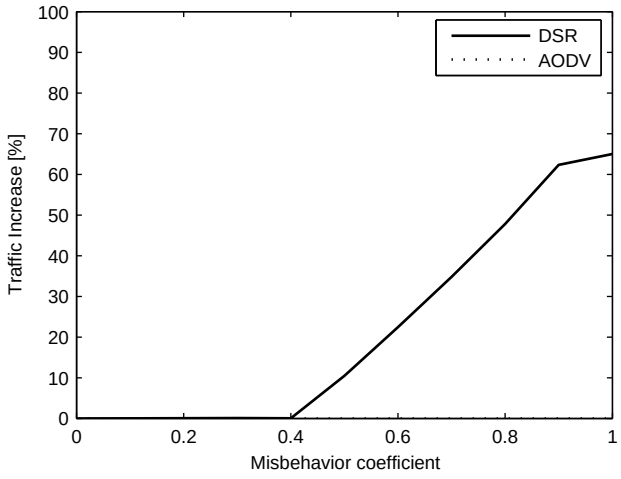

Figure 8.8: Percentage increase in traffic through alternate route as a consequence of an ongoing MAC layer attack.

in the case of the DoS attack). Another interesting observation is that the number of dropped packet decreases for the maximal value of the misbehavior coefficient. This can be easily explained by the fact that $Source1$ cannot establish a route to $Destination1$ when a DoS attack is launched. Consequently, very few packets are sent to $Node1$, most of which are dropped due to unavailability of the neighboring node. AODV, on the other hand, exhibited high resistance to misbehavior with the percentage of dropped packets being close to zero and almost independent of the degree of misbehavior. The difference in performance of two protocols can be explained as follows. If a node that belongs to a DSR route detects a broken link, it tries to salvage packets waiting in send buffer by trying to search for an alternative route in the route cache. Once this process fails, the packets in the buffer are dropped and a RERR is sent to the source. AODV, on the other hand, has no route cache, but instead uses local repair when a broken link is detected. Namely, if a node detects a broken link, it sends RREQ directly to the destination. This implies that misuses that are targeted at disrupting services can generate only temporary

impact, forcing the attacker to repeat misuses at higher frequency in order to disrupt the service. Observing the results in Fig. 8.7, we conclude that the local repair mechanism of AODV protocol can handle failures due to MAC layer attacks with much higher success rate than DSR.

To further illustrate the effects of an optimal MAC layer attack on the network layer we now proceed to the analysis of the scenario presented in Fig. 8.2. An additional traffic generating source ($Source2$) and an additional node ($Node3$) that resides in the wireless range of $Node1$ are added. These additional nodes enable creation of an alternative route to $Destination1$ in case of failure of $Node2$. We repeat the same misbehavior pattern of the selfish node as in the previous scenario and record the traffic increase through an alternative route. Due to the failure of $Node2$ and the exponential nature of back-off mechanism of $Node1$, $Node2$ becomes unreachable after the certain threshold (that corresponds to $\epsilon = 0.4$) and traffic is re-routed to the final destination through $Node3$. This topology ensures better throughput for legitimate nodes and decreases the total number of dropped packets for the DSR protocol due to the fact that after the initial route is broken, an alternative route from its cache is used to send packets. AODV, due to the identical reasons as in the previous example, is again superior to DSR with respect to the number of packets dropped and does not use the alternative route.

8.2.2 Implementation of an optimal MAC layer-based IDS

The experimental results of the scenario that employs an optimal MAC layer attack were presented in Sect. 8.2.1 and illustrated its effects in terms of lost traffic. In order to prevent (i) vertical propagation of attacks and (ii) false accusations of legitimate nodes we present the detection scheme presented in Fig. 8.9. The proposed scheme consists of two

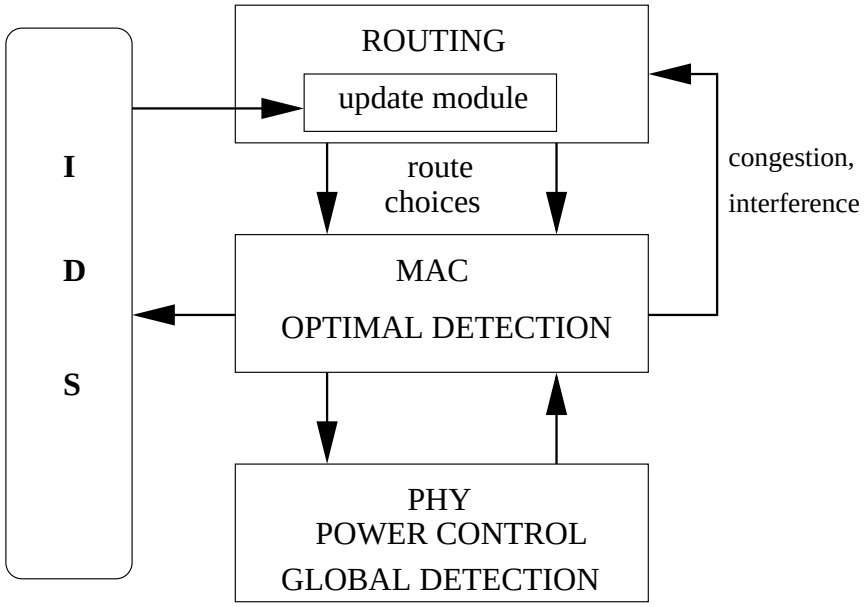

Figure 8.9: Proposed cross-layer collaboration

modules. *Module* 1, residing in the MAC layer employs the SPRT-based detection strategy described in Chap. 4. The advantage of having the MAC layer module is two-fold. First of all, we avoid the trap of false accusations in the MAC layer due to collisions and constant retransmissions. Secondly, as we will see in the remainder of the section, it reduces the probability of false alarms in the Network Layer as well. *Module* 2 resides in the Network Layer and employs already existing detection mechanisms, such as watchdog or any other suitable algorithm for detection of malicious activities. The major problem with Network Layer-based detection algorithms is that they rely on observing the number of dropped packets as the main source of information and base their decisions on misbehavior on that information. However, a node may drop significant amount of packets due to either poor channel conditions (i.e. interference) or network congestion, which may lead to false accusations. In order to prevent this scenario, we establish vertical communication among the detection modules. Both layers send their information to the IDS module. *Module* 1 sends the list of misbehaving nodes in the MAC layer and *Module* 2 sends

the list of nodes with suspicious behavior (i.e. nodes which are accused by watchdog or some other mechanism). In addition to that, we assume the IDS has the information about the network topology, such as interference graphs, existing paths etc. Using the information obtained from *Module* 1 and topology information it makes a decision about misbehavior and broadcasts the decision information throughout the network. In addition to that, using the topology information it makes a temporary decision about the best route choices in order to avoid congested areas that were created due to misbehavior.

We now implement the optimal MAC layer-based detection scheme presented in [26] and investigate the effects on the dropped traffic in the network layer with DSR and AODV as routing protocols. We assume that all nodes that take part in the detection process are legitimate and do not falsely accuse their peers of misbehavior. The results are presented in Fig. 8.7. Observing the results for the DSR protocol performance we note that the IDS achieves maximum performance for misbehavior coefficients that are larger than 0.5 (i.e. more aggressive attacks). This can be easily explained by noting that the MAC layer IDS was constructed to detect a class of more aggressive attacks that have higher impact on the system performance. On the other hand, the low impact attacks take longer to be detected and influence the performance of the routing protocol. Namely, low-impact attacks achieve certain gain in channel access time when compared to legitimate nodes. This causes temporary congestion in the MAC layer, where legitimate nodes back-off for larger periods of time due to the exponential nature of back-off mechanism in IEEE 802.11 DCF. Even after the selfish node is isolated, the legitimate nodes compete among themselves for channel access, which causes a small increase in dropped traffic. When the performance of low impact attacks is analyzed, it can be observed that the congestion effects last for additional 5-10s after the isolation of the attacker. However,

the IDS detects and isolates aggressive selfish nodes almost instantly, causing no effects in the network layer. Consequently, the percentage increase in dropped traffic at legitimate nodes for aggressive strategies of an optimal attacker is equal to zero. We also note that AODV is more robust to MAC layer attacks from the reasons mentioned in Sect. 8.2.1 and consequently implementation of a MAC layer-based IDS has no significant influence on its performance.

We conclude that the effect of MAC layer misbehavior on the network layer is twofold: (i) legitimate nodes are forced to drop significant number of packets due to unavailability of their neighbors that are blocked by the selfish node; (ii) consequently, it causes significant decrease in throughput due to unavailability of one or more nodes belonging to the initial route. This gives rise to a larger number of false positives generated by an ID mechanism that resides in the network layer since most of the network-based ID mechanisms are threshold-based and react only after a certain number of dropped packets per second is exceeded. Consequently, if no MAC layer ID mechanism is employed, legitimate nodes can be accused of misbehaving. This proves the necessity of existence of ID mechanisms in both MAC and network layers.

Bibliography

[1] OPNET modeler. [Online]. Available: http://www.opnet.com

[2] M. Raya, J.-P. Hubaux, and I. Aad, "DOMINO: A system to detect greedy behavior in IEEE 802.11 Hotspots," in *Proceedings of MobiSys '04*, 2004, pp. 84–97.

[3] P. Kyasanur and N. Vaidya, "Detection and handling of MAC layer misbehavior in wireless networks," in *Proc. of International Conference on Dependable Systems and Networks*, June 2003, pp. 173–182.

[4] A. A. Cárdenas, S. R. Radosavac, and J. S. Baras, "Detection and prevention of MAC layer misbehavior in ad hoc networks," in *Proceedings of SASN '04*, 2004, pp. 17–22.

[5] M. Čagalj, S. Ganeriwal, I. Aad, and J.-P. Hubaux, "On Cheating in CSMA/CA Ad Hoc Networks," EPFL-DI-ICA, Tech. Rep. IC/2004/27, March 2004.

[6] V. Gupta, S. Krishnamurthy, and M. Faloutsos, "Denial of service attacks at the MAC layer in wireless ad hoc networks," in *Proc. of IEEE MILCOM*, Anaheim, CA, October 2002, pp. 1118–1123.

[7] J. Bellardo and S. Savage, "802.11 denial-of-service attacks: real vulnerabilities and practical solutions," in *Proc. of USENIX Security Symposium*, Washington, DC, August 2003, pp. 15–28.

[8] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Proc. of ACM MOBICOM*, 2000, pp. 255–265.

[9] S. Buchegger and J.-Y. L. Boudec, "Performance Analysis of the CONFIDANT Protocol," in *Proceedings of MobiHoc*, Lausanne, June 2002, pp. 226–236.

[10] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross-layer design," *IEEE Wireless Communications*, vol. 12, no. 1, pp. 3–11, February 2005.

[11] V. Srivastava and M. Motani, "Cross-Layer Design: A Survey and the Road Ahead," in *IEEE Communications Magazine*, vol. 43, no. 12, December 2005, pp. 12–19.

[12] Y. Zhang, W. Lee, and Y.-A. Huang, "Intrusion detection techniques for mobile wireless networks," *Wireless Networks*, vol. 9, no. 5, pp. 545–556, 2003.

[13] C. Barrett, M. Drozda, A. Marathe, and M. V. Marathe, "Analyzing interaction between network protocols, topology and traffic in wireless radio networks," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC'03)*, vol. 3, New Orleans, 2003, pp. 1760–1766.

[14] L. Guang and C. Assi, "Vulnerabilities of ad hoc network routing protocols to MAC misbehavior," in *Proc. IEEE International Conference on Wireless And Mobile Computing, Networking And Communications*, vol. 3, 2005, pp. 146 – 153.

[15] G. Thamilarasu, A. Balasubramanian, S. Mishra, and R. Sridhar, "A cross-layer based intrusion detection approach for wireless ad hoc networks," in *Proc. IEEE International Conference on Mobile Adhoc and Sensor Systems*, 2005.

[16] IEEE, "IEEE wireless LAN medium access control (MAC) and physical layer (PHY) specifications," 1999. [Online]. Available: http://standards.ieee.org/getieee802/

[17] A. Wald, *Sequential Analysis.* New York: John Wiley and Sons, 1947.

[18] C. W. Helstrom, *Elements of signal detection and estimation.* Prentice-Hall, 1995.

[19] A. Wald and J. Wolfowitz, "Optimum character of the sequential probability ratio test," *Ann. Math. Statist.*, vol. 19, pp. 326 – 339, 1948.

[20] V. Dragalin, A. Tartakovsky, and V. Veeravalli, "Multihypothesis Sequential Probability Ratio Tests - Part I: Asymptotic optimality," *IEEE Trans. on Information Theory*, vol. 45, no. 7, pp. 2448 – 2461, November 1999.

[21] D. Bertsekas, *Convex analysis and optimization.* Athena Scientific, 2003.

[22] A. A. Cárdenas, J. S. Baras, and K. Seamon, "A framework for the evaluation of intrusion detection systems," in *Proceedings of the 2006 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2006.

[23] S. Axelsson, "The base-rate fallacy and its implications for the difficulty of intrusion detection," in *Proc. of the 6th ACM Conference on Computer and Communications Security (CCS '99)*, November 1999, pp. 1–7.

[24] A. A. Cárdenas, S. Radosavac, and J. S. Baras, "Performance Comparison of Detection Schemes for MAC Layer Misbehavior," in *Proceedings of INFOCOM '07*, 2007.

[25] R. E. Blahut, *Principles and Practice of Information Theory.* Addison-Wesley, 1987.

[26] S. Radosavac, J. S. Baras, and I. Koutsopoulos, "A Framework for MAC Protocol Misbehavior Detection in Wireless Networks," in *Proceedings of the 4th ACM workshop on Wireless security*, Cologne, Germany, September 2005, pp. 33–42.

[27] S. Radosavac, G. V. Moustakides, J. S. Baras, and I. Koutsopoulos, "An analytic framework for modeling and detecting access layer misbehavior in wireless networks," *submitted to ACM Transactions on Information and System Security (TISSEC)*, 2006.

[28] Q. Zhang and S. A. Kassam, "Finite-state Markov model for Rayleigh fading channels," *IEEE Transactions on Communications*, vol. 47, no. 11, pp. 1688–1692, November 1999.

[29] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* N. Y.: John Wiley & Sons, Inc., 1991.

[30] J. E. Shore and R. W. Johnson, "Properties of Cross-Entropy Minimization," *IEEE Transactions on Information Theory*, vol. 27, no. 4, pp. 472–482, July 1981.

[31] ——, "Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy," *IEEE Transacions on Information Theory*, vol. 26, no. 1, pp. 26–37, January 1980.

[32] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *Mobile Computing*, pp. 153–181, 1996.

[33] C. E. Perkins and E. M. Royer, "Ad hoc On-Demand Distance Vector Routing," in *Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, February 1999, pp. 90–100.

[34] P. E. Protter, *Stochastic Integration and Differential Equations*, 2nd ed. Springer, 2004.