Dissertation

Feasibility and Applications of a Wireless Firewall

Vom Fachbereich Informatik
der Technischen Universität Kaiserslautern
zur Verleihung des akademischen Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
genehmigte Dissertation
von

Matthias Stephan Wilhelm

Dekan:	Prof. Dr. Klaus Schneider
Datum der Aussprache:	13. April 2015
Promotionskommission:	
Vorsitzender:	Prof. Dr. Roland Meyer
Berichterstatter:	Prof. Dr. Jens B. Schmitt
	Prof. Dr. Reinhard Gotzhein
	Prof. Dr. Matthias Hollick

D 386

Abstract

Most of today's wireless communication devices operate on unlicensed bands with uncoordinated spectrum access, with the consequence that RF interference and collisions are impairing the overall performance of wireless networks. In the classical design of network protocols, both packets in a collision are considered lost. However, with the current proliferation of wireless applications, e.g., WLANs, car-to-car networks, or the Internet of Things, this conservative approach is increasingly limiting the achievable network performance in practice. Instead of shunning interference, this thesis questions the notion of „harmful" interference and argues that interference can, when generated in a controlled manner, be used to increase the performance and security of wireless systems. Using results from information theory and communications engineering, we identify the causes for reception or loss of packets and apply these insights to design system architectures that benefit from interference. Because the effect of signal propagation and channel fading, receiver design and implementation, and higher layer interactions on reception performance is complex and hard to reproduce by simulations, we design and implement an experimental platform for controlled interference generation to strengthen our theoretical findings with experimental results. Following this philosophy, we introduce and evaluate a system architecture that leverage interference. First, we identify the conditions for successful reception of concurrent transmissions in wireless networks. We focus on the inherent ability of angular modulation receivers to reject interference when the power difference of the colliding signals is sufficiently large, the so-called *capture effect*. Because signal power fades over distance, the capture effect enables two or more sender–receiver pairs to transmit concurrently if they are positioned appropriately, in turn boosting network performance. Second, we show how to increase the security of wireless networks with a centralized network access control system (called WiFire) that selectively interferes with packets that violate a local security policy, thus effectively protecting legitimate devices from receiving such packets. WiFire's working principle is as follows: a small number of specialized infrastructure devices, the guardians, are distributed alongside a network and continuously monitor all packet transmissions in the proximity, demodulating them iteratively. This enables the guardians to access the packet's content before the packet fully arrives at the receiver. Using this knowledge the guardians classify the packet according to a programmable security policy. If a packet is deemed malicious, e.g., because its header fields indicate an unknown client, one or more guardians emit a limited burst of interference targeting the end of the packet, with the objective to introduce bit errors into it. Established communication standards use frame check sequences to ensure that packets are received correctly; WiFire leverages this built-in behavior to prevent a receiver from processing a harmful packet at all. This paradigm of „over-the-air" protection without requiring any prior modification of client devices enables novel security services such as the protection of devices that cannot defend themselves because their performance limitations prohibit the use of complex cryptographic protocols, or of devices that cannot be altered after deployment.

Acknowledgements

I would like to thank my advisor Prof. Jens Schmitt for his continuous support and never-ending supply of good ideas. His style of research had a large influence on me and strongly shaped the research presented in this thesis. I would also like to thank all the members of my Ph.D. committee, namely Prof. Mayer, Prof. Gotzhein, and Prof. Hollick for their support and valuable suggestions.

The team at DISCO was also a valuable source of inspiration and amusement. I would especially like to thank Ivan Martinovic for the opportunity to research together with him. I also extend my gratitude to the other (current and previous) members, Matthias Schäfer, Steffen Bondorf, Hao Wang, Wint Yi Poe, Daniel Berger, Michael Beck, Adam Bachorek, Nicos Gollan, and the extended DISCO family, with Steffen Reithermann, Barbara Erlewein, Edith Hofbauer, and Markus Fuchs, and many more. I wish you all the best for your life during and after DISCO.

The WiFire concept was largely shaped in a fruitful collaboration with armasuisse W+T, and Vincent Lenders had a large part in the successful design and implementation of the wireless firewall. Additionally, this thesis would not have been possible without the efforts of numerous students that completed their theses or worked on projects with me. I hope that they learned something useful in the process.

Finally, I would like to thank my parents and family for their support and faith in me.

Previously Published Material

This thesis revises and extends on the following previous publications:

Wilhelm, Matthias, Vincent Lenders, and Jens B. Schmitt (2013a). *An Analytical Model of Packet Collisions in IEEE 802.15.4 Wireless Networks.* Tech. rep. Kaiserslautern, Germany: Dept. of Computer Science, TU Kaiserslautern. arXiv: 1309.4978 [cs.NI].

– (2014). “On the Reception of Concurrent Transmissions in Wireless Sensor Networks”. In: *IEEE Transactions on Wireless Communications.* doi: 10.1109/TWC.2014.2349896.

Wilhelm, Matthias, Ivan Martinovic, Jens B. Schmitt, and Vincent Lenders (2011a). “Short Paper: Reactive Jamming in Wireless Networks—How Realistic is the Threat?” In: *Proceedings of the 4th ACM Conference on Wireless Network Security.* WiSec 2011. (Hamburg, June 15–17, 2011). New York, NY, USA: ACM, pp. 47–52. doi: 10.1145/1998412.1998422.

– (2011b). “WiFire: A Firewall for Wireless Networks”. In: *Proceedings of the ACM SIGCOMM 2011 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* SIGCOMM 2011. (Toronto, Aug. 15–19, 2011). New York, NY, USA: ACM, pp. 456–457. doi: 10.1145/2018436.2018518.

– (2011c). “WiSec 2011 Demo: RFReact—A Real-time Capable and Channel-aware Jamming Platform”. In: *SIGMOBILE Mobile Computing and Communications Review* 15, pp. 41–42. doi: 10.1145/2073290.2073300.

– (2013b). *Air Dominance in Sensor Networks: Guarding Sensor Motes using Selective Interference.* Tech. rep. Kaiserslautern, Germany: Dept. of Computer Science, TU Kaiserslautern. arXiv: 1305.4038 [cs.NI].

Wilhelm, Matthias and Jens B. Schmitt (2012). “Interference Scripting: Protocol-aware Interference Generation for Repeatable Wireless Testbed Experiments”. In: *Proceedings of the 4th Annual Wireless of the Student, by the Student, and for the Student Workshop.* S3 2012. (Istanbul, Aug. 22–26, 2012). New York, NY, USA: ACM, pp. 21–23.

WILHELM, MATTHIAS, JENS B. SCHMITT, and VINCENT LENDERS (2012). “Practical Message Manipulation Attacks in IEEE 802.15.4 Wireless Networks”. In: *Workshop Proceedings of the 16th International GI/ITG Conference on Measurement, Modelling, and Evaluation of Computing Systems* and *Dependability and Fault Tolerance*. MMB & DFT 2012. (Kaiserslautern, Mar. 21, 2012). Kaiserslautern, Germany, pp. 29–31.

Contents

Acknowledgements iii

Previously Published Material v

1. Introduction 1
- 1.1. Motivation 1
- 1.2. The Concept of Firewalls 3
- 1.3. A Firewall for Wireless Networks 4
- 1.4. Contributions and Outline 7

2. State of the Art in Physical Layer Security 9
- 2.1. Jamming for Confidentiality 9
- 2.2. Jamming for Access Control 11
  - 2.2.1. Wireless LAN 11
  - 2.2.2. Implanted Medical Devices 13
  - 2.2.3. Radio Frequency Identification Tags 14
  - 2.2.4. Wireless Sensor Networks 15
  - 2.2.5. Other Applications 16
- 2.3. Summary 16

I. Conception and Analysis of Wireless Firewalls 17

3. Protection Concept 19
- 3.1. Protection Scenario 19
- 3.2. Using Interference for Access Control 20
- 3.3. Research Challenges 22
  - 3.3.1. Understanding Packet Reception 22
  - 3.3.2. Optimal Jamming Techniques 23
  - 3.3.3. Nature of Protection 23
- 3.4. Technical Challenges 24
  - 3.4.1. Reactive and Selective Operation 25
  - 3.4.2. Efficient Operation 26

3.4.3. Effective Operation 27

4. Understanding the Reception of Colliding Packets 29
4.1. Introduction 29
4.1.1. The Case of Concurrent Transmissions 29
4.1.2. Modeling Approach 31
4.2. Parameters of the Analysis 32
4.3. Background on IEEE 802.15.4 35
4.3.1. Reception Process 35
4.3.2. Causes of Loss on the IEEE 802.15.4 PHY 37
4.4. System Model 39
4.4.1. Sender Model 39
4.4.2. Channel Model 43
4.4.3. Receiver Model 43
4.5. Mathematical Analysis 45
4.5.1. Synchronized Signal 46
4.5.2. Carrier Phase Offset 46
4.5.3. Time Offset 47
4.5.4. Both Offsets 48
4.5.5. Analysis Summary 48
4.6. Mathematical Evaluation 49
4.6.1. Methodology 49
4.6.2. Reception of the Synchronized Signal of Interest 53
4.6.3. Reception of Interfering Signals with Independent Payload 60
4.7. Discussion 64
4.7.1. Signal to Interference Ratio SIR 64
4.7.2. Time Offset τ 64
4.7.3. Carrier Phase Offset φ_c 65
4.7.4. Number of Concurrent Interferers 65
4.8. Summary 66

5. Protection Analysis 67
5.1. Protection Constraints 67
5.1.1. Primer on Guardian Detection Sensitivity 67
5.1.2. Protection Conditions 68
5.2. Attacker Models 69
5.3. Attacker–Guardian Asymmetry 69
5.4. Summary 73

II. Design and Implementation of WiFire 75

6. System Architecture 77
6.1. Design Space of Wireless Firewalls 77
6.1.1. Store-and-Forward Wireless Firewalls 77
6.1.2. Operational Requirements 79
6.1.3. Architectures for Wireless Firewalls 80
6.2. The Architecture of the WiFire System 82
6.2.1. System Architecture 82
6.2.2. An Overview of the System Components 82
6.3. Summary 85

7. System Implementation 87
7.1. System Implementation Overview 87
7.1.1. Implementation Guidelines 88
7.1.2. USRP2 Platform Details 88
7.2. Resource Mapping 89
7.3. Subsystem: Detection 90
7.3.1. IEEE 802.15.4 Receiver 90
7.3.2. Packet Framing 90
7.3.3. Detector Implementation 90
7.4. Subsystem: Decision 91
7.5. Detection Evaluation 91
7.6. Decision Subsystem Evaluation 93
7.7. Overall System Response Time 95
7.8. Summary 96

III. Experimental Evaluation of WiFire 99

8. Jamming Evaluation 101
8.1. Validation of the Collision Model 101
8.1.1. Experimental Setup 101
8.1.2. Experimental Results 102
8.1.3. Concurrent Transmissions Summary 106
8.2. Experimental Evaluation of Jamming 106
8.2.1. Effectiveness of Jamming Waveforms 106
8.2.2. Effect of the Center Frequency on Jamming Performance 110
8.2.3. Minimum Interference Duration 110
8.2.4. Jamming Summary 112

9. WiFire System Evaluation 113

9.1. Evaluation of the Protection Coefficient 113
    9.1.1. Experimental Setup and Methodology 114
    9.1.2. Experimental Results 116
    9.1.3. Summary 120
9.2. Selectivity Experiments 120

10. Discussion 123

10.1. Additional Options for Rules 123
    10.1.1. Positioning 123
    10.1.2. Link Signatures 124
    10.1.3. Device Identification 124
10.2. Technical Considerations 125
    10.2.1. Additional Communication Technologies 125
    10.2.2. Optimal Guardian Deployment 125
10.3. Non-technical Considerations 126
    10.3.1. Economic Aspects 126
    10.3.2. Legal Aspects 126
10.4. Using WiFire for Interference Research 127
    10.4.1. Interference in Experiments 128
    10.4.2. Interference Scripting 129
    10.4.3. Application Examples 130
10.5. Summary 131

11. Conclusion 133

11.1. Summary 133
11.2. Future Work 133

A. Pulse Integration 135

B. Colliding Packets Demodulator Output 141

Bibliography 145

Curriculum Vitae 161

Publication List 163

List of Figures

1.1. The firewall concept in wired networks. 4
1.2. Wireless network without protection 5
1.3. Network with wireless firewall protection 6

3.1. Example of a protection scenario. 19
3.2. WiFire jamming time constraints 25

4.1. Structure of IEEE 802.15.4 frames 35
4.2. System model for collision analysis 39
4.3. Minimum shift keying (MSK) modulation example. 41
4.4. Effect of carrier phase offsets in a collision 47
4.5. Effect of time offsets in a collision 47
4.6. Effect of time and phase offsets in a collision 48
4.7. Capture threshold for colliding packets with independent payload 51
4.8. Effect of signal to interference ratio SIR on the PRR for independent payload 52
4.9. Capture threshold for colliding packets with independent payload with extended time offsets. 54
4.10. Capture threshold for colliding packets with identical payload 56
4.11. Effect of signal to interference ratio SIR on the PRR for identical payload 57
4.12. Capture threshold for colliding packets with identical payload with extended time offsets. 59
4.13. Effects of several interferers in a collision 60
4.14. Reception performance of interfering signals 61
4.15. Error rates of all parameter constellations 63

6.1. The operation of WiFire in detail. 79
6.2. Implementation component diagram 83

7.1. Implementation resource allocation 89
7.2. Receiver sensitivity 92
7.3. Rule delay analysis 93
7.4. Jammer reaction time 95

8.1. Experimental results of PRR performance 103
8.2. Experimental results of SER performance 105
8.3. Performance of IEEE 802.15.4 jamming waveforms. 108
8.4. Comparison of jamming waveforms. 109
8.5. Frequency-dependency of jamming. 110
8.6. Minimum jamming duration for IEEE 802.15.4. 111

9.1. Results from the corridor experiments. 117
9.2. Results of the gymnasium experiment. 119
9.3. Results of the sports field experiment. 120
9.4. Underground tunnel results 121
9.5. Experimental results with concurrent traffic 122

A.1. Examples for rectangle integration 136

List of Tables

2.1. Comparison of related work. 12

4.1. Chipping sequences of IEEE 802.15.4 36
4.2. Correction factors used in the derivations. 45

7.1. Jamming speed summary 97
7.2. Summary of time parameters 97

10.1. Additional rule definition criteria 124

List of Symbols

$2T$	Time duration of a bit in the MSK modulation
$\mathcal{A}_{\text{force}}$	Attackers using a brute-force strategy
α_k^I	Information bit at position k of the synchronized sender
$a_I(t)$	Information sequences for the in- and quadrature phase transmitted in $s(t)$
α	Path loss coefficient
A_s	RF signal amplitudes of $s(t)$ at the receiver
$\mathcal{A}_{\text{stealth}}$	Attackers using a stealthy strategy
$b_{I,i}(t)$	Information sequences for the in- and quadrature phase transmitted in $u_i(t)$
$c_{\xi,k}$	Chipping sequence of symbol ξ at bit position k
d_{ag}	Distance between the attacker and the guardian
d_{av}	Distance between the attacker and the victim
d_{gv}	Distance between victim and the guardian
γ_{SIR}	Signal-to-interference threshold for reception
$L(d)$	Path loss and fading function of distance d
$\Lambda_u^I(k)$	Contribution of signal $u(t)$ to the bit decision in bit interval k
ω_c	Angular velocity of the carrier wave with frequency f_c
$\varphi_{p,i}$	Baseband pulse phase offset of the RF signal of interferer i
$\hat{o}_k^I$	Decision variable of the detector for bit k of I component

o_k^I	Detected bit of an uncoded transmission.
ω_p	Angular velocity of baseband pulses (periodic by $4T$)
P_a	Radiated power by the attacker
$\varphi_{c,i}$	Carrier phase offset of the RF signal of interferer i
P_g	Radiated (interference) power by the guardian
$\phi_I(t)$	Basis function of the MSK modulation to demodulate bits
$\Pi(t)$	Unit pulse (step) function.
$\mathcal{R}$	Resulting attack range, performance metric for attackers
$r(t)$	Resulting superposition of RF signals at the receiver
σ_j^{HDD}	Detected symbol after DSSS decoding, the index j using hard decision decoding (HDD)
σ_j^{SDD}	Detected symbol after DSSS decoding, the index j using hard decision decoding (SDD)
$s(t)$	MSK modulated RF signal emitted by a synchronized sender
τ_i	Time offset of the RF signal of interferer i
$u_i(t)$	MSK modulated RF signal by interferer i with possible time offset τ_i and phase offset $\varphi_{c,i}$
ξ	Input DSSS symbol used for transmission

1. Introduction

Contents

3.1. Protection Scenario **19**
3.2. Using Interference for Access Control **20**
3.3. Research Challenges **22**
3.4. Technical Challenges **24**

1.1. Motivation

Wireless networks are growing in importance at a rapid pace, and wireless technologies are becoming pervasive in our daily lifes. This process leads to the situation that wireless connectivity is increasingly used in security-critical contexts. However, providing security in wireless networks is generally harder to achieve compared to wired networks. Due to the broadcast nature of radio frequency (RF) propagation, network access cannot be regulated physically; anyone in transmission range can eavesdrop or inject arbitrary messages.[1] The current response to this threat is to use strong cryptographic protection to ensure that messages remain confidential and that only authorized parties can participate in a network. This approach, however, is not always easily applicable because wireless devices have several unique characteristics: *(i)* the devices often have limited computational resources and are optimized for a particular application, *(ii)* they run on batteries and thus have the primary goal to maximize lifetime, *(iii)* they may have limited programmability or cannot be modified at all, *(iv)* they may be mobile and travel across different security domains, and *(v)* they are often personal belongings that are operated and configured by (possibly security-oblivious) end users. It is hard to imagine that devices such as sensor motes, RFID chips or implanted medical devices implement a full range of security measures despite these challenges. Their

---

[1] For example, researchers were able to eavesdrop on Bluetooth phone calls from more than a mile away: `http://www.wired.com/politics/security/news/2004/08/64463`.

protection task is highly asymmetric because all security protocols must be implemented on each resource-limited device while the adversary can use high-performance systems. This thesis sets out to address these issues with a remote protection system that takes over the burden of protection from the low-power devices, especially for devices in the context of wireless sensor networks (WSNs).

Security in Wireless Sensor Networks

WSNs are extending their application scope from industrial monitoring and location tracking to more personal and assistant technologies, such as in health care (CHIPARA et al., 2010), assisted living (HNAT et al., 2011; WOOD et al., 2008), and home energy saving applications (JIANG et al., 2009; LU, SOOKOOR, et al., 2010). ZigBee-enabled devices such as door locks, occupancy sensors, panic buttons and electrical sockets are already available as low-cost consumer electronics ready to be deployed in users' residences. Imagine an emergency scenario where a gas leakage detector rises an alarm or a panic button is pressed, and since an occupancy sensor reports an occupied room, the door lock system decides to unlock and provide emergency exits. While such a scenario is a perfect motivation for using WSN technologies, it also provides an attractive playground for an attacker.

In contrast to wired networks where physical control of traffic is inherently given, wireless networks are open by nature. For this reason, both IEEE 802.15.4 (at the link-layer) and ZigBee (at the upper layers) define conventional security services for frame protection, device authorization, key distribution, and key establishment. However, they also take into account restrictions of battery-powered, performance-limited and low-cost devices and offer tradeoffs between resource requirements and security objectives, depending on the particular application scenario. For example, according to the IEEE 802.15.4 standard, there are three security modes: *(i)* no security, *(ii)* access control lists (ACLs) based on a source address, and *(iii)* secured mode, offering a choice of strong security suites such as 128 bit AES-CCM. From a security perspective, only the latter option offers protection in an adversarial setting. Similarly, the ZigBee-2007 specification describes key management and key exchange methods. It specifies three types of keys: *(i)* master key, used as an initial shared secret to generate link keys, *(ii)* link key, dynamically generated secret keys shared only between two devices, and *(iii)* network key, a global secret key shared among all WSN devices. Yet, the master and link keys are optional. Hence, it is realistic to assume that the security of standard ZigBee networks may reside in only a single shared key with the obvious risk that the capture of a single device and extraction of the secret key could jeopardize the security of the whole network. Along these lines, a recently available security analysis toolbox called KillerBee (Wright, 2010) offers a set of attack vectors, such as Over-the-Air (OTA) key sniffing, MAC address manipulation, key extraction from memory, and denial-of-service attacks based on flooding WSNs with memory-consuming association requests.

1.2. The Concept of Firewalls

Given this plethora of security issues, a new security paradigm is needed that offers protection even in the face of severe resource limitations. Ideally, what you want is an external guardian system that supports devices in their task of protecting themselves. A wish list of its features may be:

- remote protection for several devices in parallel,
- support for generic and programmable security policies, and
- transparent operation; no changes to the existing devices should be necessary.

Remote protection helps to off-load security costs in terms of energy, storage, and computation time to an external security infrastructure, programmable security policies enable an easy adaptation to new technologies or threats, and transparent operation ensures that any wireless device, even a legacy device or a device with fixed programming, can be protected this way. In wired networks, many of these properties are found within the concept of *network firewalls*.

Firewalls

Firewalls are located at the edges of networks, controlling the access to the networks they are to protect, effectively defining a trust boundary between inside and outside world. To that end, firewalls analyze in- and outbound packet streams and classify into trusted and untrusted traffic, blocking the latter. A comprehensive definition of firewall is given by Bellovin and Cheswick (1994). They define a firewall as a collection of components placed between two networks that collectively have the following properties:

- All traffic from inside to outside, and vice versa, must pass through the firewall.

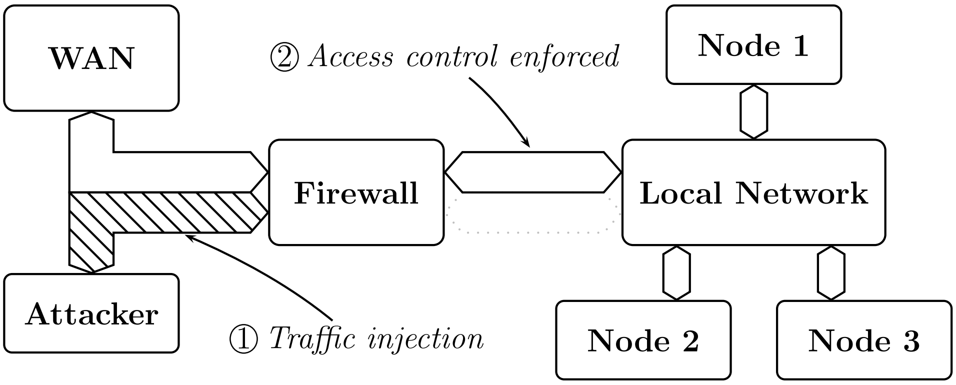


Figure 1.1.: The firewall concept in wired networks.

- Only authorized traffic, as defined by the local security policy, will be allowed to pass.
- The firewall itself is immune to penetration.

This principle is depicted in Figure 1.1. The local network on the right is protected by a central firewall that bridges all access from the untrusted outside network and controls all incoming traffic. Attacks from the untrusted network can be detected and filtered before malicious traffic can arrive to any of the protected nodes. Network administrators are generally very much in favor of firewalls as they enable a central control of policy enforcement. This is, in principle, achieved in a *transparent* way, without necessitating changes to existing protocols or host configurations. A firewall's actions are usually specified by generic policies defined by filtering rules.

1.3. A Firewall for Wireless Networks

From the previous discussion, it is hard to deny that the concept of enforcing security policies by blocking unwanted traffic before it reaches the clients could also be attractive in supporting a practical approach to wireless network security. A noteworthy property of the firewall concept is the basic assumption that one cannot rely on the end systems to run correctly implemented and configured security software. This assumption is even more satisfied in wireless networks than in the traditional Internet context:

- Wireless devices are often personal belongings of the respective user and thus out of control for central system administration. This creates a

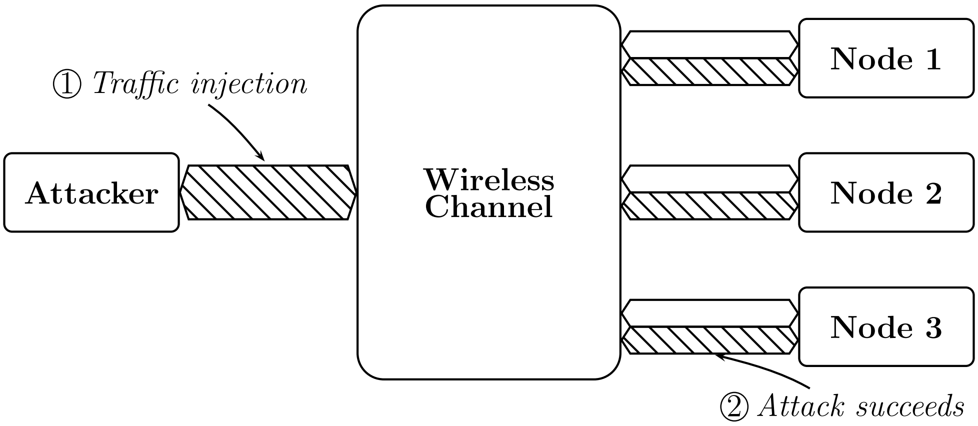


Figure 1.2.: Wireless network without protection; an attacker has full access to the wireless channel and can communicate with any device in the vicinity.

large potential for misconfigurations, lazy security updates and infected clients.

- Wireless clients are mobile, often dynamically changing back and forth between trust boundaries. This usually results in more complex security protocols.
- The fundamental broadcast characteristics of wireless networks make them a perfect playground for all kinds of attacks. In contrast to wired networks where physical control of traffic is inherently given, wireless networks are open by nature.

Perhaps surprisingly, firewalls are entirely nonexistent in the wireless domain. Yet, the need to selectively control and block radio communication is particularly high in a broadcast environment since any node may receive and send packets.

The problem is, however, that in contrast to wired networks, the broadcast nature of the wireless channel does not provide any physical separation of the traffic and thus setting up a boundary between inside and outside world is much harder to achieve, especially when considering mobile nodes. Consequently, preventing a packet from being received cannot be based on simple and silent dropping at a store-and-forward device but requires a different mechanism. This is depicted in Figure 1.2.

This thesis sets out to explore a way to offer a security service similar to that of firewalls in wired networks. We propose a distributed guardian system

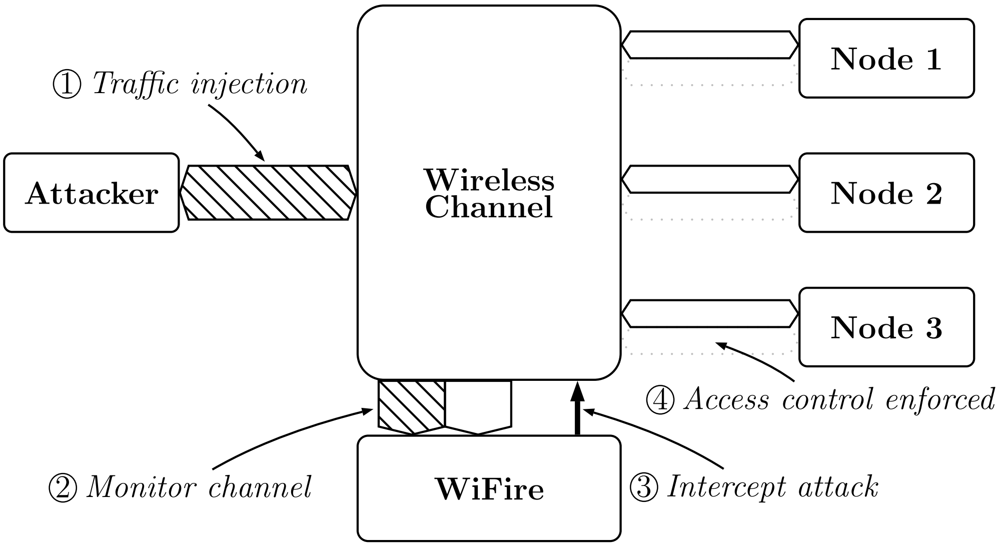


Figure 1.3.: Network with a remote protection. An additional system, the wireless firewall, is added to control the communication on the wireless channel. It monitors the channels and intercepts all packets that do not adhere to the policy, implementing access control for the protected network.

to protect wireless networks based on physically regulating channel access by means of selective interference. The guardians are deployed alongside a sensor network, inspecting all local traffic, classifying packets based on their content, and destroying any malicious packet while still on the air.

Bringing Firewalls to Wireless Networks

In our approach, we aim to protect wireless devices from attacks over-the-air and on a per-packet basis, lifting the security burden from them. Our approach combines selective interference and rule-based security policies to a generic protection mechanism for wireless networks.

To that end, we explore the concept of wireless firewalls. The core idea is simple and yet effective. The working principle is simple—if we cannot prevent the transmission of a malicious packet, we may still prevent its reception. The protection system listens to the wireless medium and attempts to demodulate and decode ongoing transmissions. When a received signal matches any predefined blocking rule, a jamming signal is sent by the system to interfere with the ongoing frame transmission at the receiver.

The wireless firewall achieves this by content-based classification and selective interference that is just long enough to induce checksum errors in malicious packets. This way, the protection is fully transparent to the network: everything received without errors is trustworthy.

1.4. Contributions and Outline

The main goal of this thesis is to show that such remote protection is theoretically sound and technically feasible. We aim to explore the protection properties of such systems, and the possibilities and limitations of the approach.

Because the protection system operates on the physical layer, it is essential to perform an experimental evaluation together with system modeling and analysis. The interaction of different systems on the physical layer is very hard to capture correctly in communication models. For this reason, we first set out to understand the effects of jamming and interference analytically to design an optimal protection system. Subsequently, to validate the analytical results and to show the technical feasibility, we also design and implement such a remote protection system, called WiFire, and perform a series of experiments that facilitates the understanding of remote protection using RF interference.

This thesis makes several contributions:

- We introduce the concept of remote protection using RF interference. In contrast to existing solutions, it supports a selective filtering of packet transmissions using a ruleset that is checked during the transmission of the packet. Additionally, we focus to provide access control from a distance, in the order of several meters, instead of approaches that require a close proximity between all devices.

- To maximize the effects of RF interference and to provide reliable protection, it is essential to understand the effects of collisions at the physical layer to identify the reasons why packets are either lost or received. In this thesis, we provide a comprehensive analysis of packet collisions that uses the mathematical representation of RF signals and a realistic receiver model to derive detailed reception conditions. These insights help to ensure that the wireless firewall operates successfully, but the results also offer new insights in the area of network performance modeling and network protocols.

- Based on the realistic system model in the previous item, we analyze the protection properties of wireless firewalls. Because the system operates on the physical layer, the nature of protection is different from cryptographic protection, and this analysis yields the insight that the relation of signal strengths at the receiver plays a decisive role. Our results show that a remote protection is indeed feasible, but new factors must be considered like the positions of all network participants.

- After we have shown that the protection concept is sound, we present the design and implementation of WiFire, a prototype system for a wireless firewall. We overcome several technical challenges in the design of such systems, and are the first who provide a reactive and selective jamming system for IEEE 802.15.4 networks. Using system benchmarks, we show that all our design goals are met.
- In order to validate our experimental results for reception in collisions and to understand the real-world effects of jamming, we provide a comprehensive experimental evaluation of jamming, which considers factors like interference duration, optimal jamming waveforms, and difference receiver designs.
- Finally, we provide a real-world performance evaluation of WiFire, validating that the wireless firewall concept is indeed feasible in practice.

The thesis outline is as follows. We first provide a discussion of related work. While physical layer security is a very active area, we show that the use of selective interference for access control is a novel concept. In the first part of this thesis, we outline the concept and provide an analytical treatment of the interference effects that we use for our system, such as collisions on the wireless channel, and the protection properties of RF interference. In the second part, we offer insights into the system design and implementation, discussing the pros and cons of different architectural choices. In the last part, we employ the system implementation to perform an experimental evaluation of our concept. We show that our models that we developed in the analysis are a good prediction for real-world behavior, and that the protection concept of wireless firewalls is a suitable approach for wireless networks. Finally, we provide a detailed discussion of further technical and non-technical aspects, and conclusions on this work.

2. State of the Art in Physical Layer Security

Contents

4.1. Introduction 29
4.2. Parameters of the Analysis 32
4.3. Background on IEEE 802.15.4 35
4.4. System Model 39
4.5. Mathematical Analysis 45
4.6. Mathematical Evaluation 49
4.7. Discussion 64
4.8. Summary 66

To understand the research context of this work, we first review other research contributions in the literature that consider security mechanisms on the physical layer. Physical layer security is a new paradigm that is being explored to protect wireless devices from attacks. The general idea is to shift the protection from the upper layers to the physical layer of communications, which opens up physical features like angle of arrival, the shape of a waveform, the unpredictability of channel fading, or properties of noise and interference, which all can be employed to protect devices. Two of these physical layer approaches are closely related to the topic of this thesis, namely jamming for confidentiality (i.e., adding interference on the wireless channel to prevent the reception of outside eavesdroppers) and jamming for access control (i.e., interference is generated to prevent the reception of inside receivers).

2.1. Jamming for Confidentiality

In information theory the concept of *secrecy capacity* of broadcast channels with noise was studied extensively, starting with Wyner's work on wiretap channels (1975). The goal is to enable confidential communications without secrets over a public broadcast channel. Recently, several authors augmented the wiretap channel by considering the creation of intentional noise to boost the secrecy capacity of the channel. This approach is known under various designation, such as *artificial noise* (GOEL and NEGI, 2008; R. LIU et al., 2008), *cooperative jamming* (X. HE and YENER, 2009; TEKIN and YENER, 2008), *friendly jamming* (VILELA et al., 2011), the *relay-eavesdropper channel* (LAI and EL GAMAL, 2008), or the *wiretap channel with helping interferer* (TANG et al., 2011). These works show that confidentiality in the sense of information theory can be achieved even if the signal-to-noise-and-interference ratio (SINR) of the adversary is higher. Our work is orthogonal to these approaches because we exploit jamming to *control* medium access instead of ensuring message confidentiality at the physical layer, i.e., while the jamming in the related work is targeting attackers to prevent them to receive messages, our approach here is to jam the nodes in the protected network, preventing them from receiving unauthorized messages.

Work on wireless network security applies result on the wiretap channel to prevent information leakage from a protected geographical area by hindering the reception process of eavesdroppers using intentional RF interference in more realistic scenarios. KIM et al. (2012) propose *defensive jamming,* a method to prevents eavesdroppers from detecting messages correctly by using jamming directed towards the border of the protected area. The basic assumption of this approach is that the network deployment area itself is physically secured, so that the eavesdropper must remain outside a surrounding security perimeter. The authors perform an analysis of jammer placement strategies based on the SINR model and show that eavesdropping can be prevented successfully by an appropriate spatial jammer placement. The work of SANKARARAMAN et al. (2012) analyzes similar attack scenarios and presents algorithms for optimal jamming power assignment and jammer placement, both for a fixed number of jammers and a near-optimal number of jammers. While the methods used on these works can be adapted to analyze the performance and deployment of our wireless firewall system, this work is orthogonal because they consider passive (eavesdropping) attackers, while we focus on active (injecting) attackers.

Cooperative jamming from a third node is used to make decoding at eavesdropping nodes impossible (LAI and EL GAMAL, 2008). iJam (GOLLAKOTA and KATABI, 2011) achieves the same purpose without a third node and for OFDM signals.

Confidentiality of transmissions in these approaches relies on the fact that an eavesdropper cannot overcome the effects of interference. However, if the interference pattern is known, it is possible to eliminate the interfering waveform and recover the original signal (Halperin, Anderson, et al., 2008). And even if the interference patterns is not known, it is possible for the attacker to increase the degrees of freedom that the receiver has. For example, Tippenhauer et al. (2013) showed that an attacker always has the option to increase its number of antennas, allowing to isolate the defensive jamming signal and to recover the confidential information. Our approach does not suffer from these drawbacks, because we interfere with the reception of an attacked device, whose capabilities are known to us, instead of interfering with the reception of an eavesdropper, which can have arbitrarily advanced capabilities.

2.2. Jamming for Access Control

The concept of using selective interference for access control has recently been proposed in several application areas: to protect implanted medical devices (IMDs) from malicious readers, to increase the privacy of RFID tags, and to ensure authentic communication in WSNs. In contrast to these works, we provide a system that allows configurable security policies based on packet content and aims to provide a central protection over larger distances in a *networked* setting, in contrast to a reader–single device setting. A summary of the following comparison is provided in Table 2.1.

2.2.1. Wireless LAN

Wireless intrusion detection and prevention systems (WIPS) are also closely related to the wireless firewall (Scarfone and Mell, 2007, §5) in the context of wireless LANs. They are primarily focus on detecting attacks and policy violations in IEEE 802.11 networks. However, commercial products for WLAN protection such as AirMagnet by Fluke Corporation (2011), AirDefense by Motorola Solutions (2011) or SpectraGuard by AirTight Networks (2011) do not prevent the reception of packets; rather, they exploit the fact that communication is only possible after reaching an associated state with an access point, and repeatedly break this association to the adversary. These systems have only limited capability to deny access to unauthorized nodes by sending spoofed disassociation and de-authentication frames to rogue devices. This approach is not applicable to protect low-power wireless networks because their protocols do not use such association mechanisms.

System	Application area	Maximum reaction time	Guard distance	Blocking criteria	Prototype evaluation
IMD shield	Implanted medical devices (IMDs)	10 ms	20 cm	Each packet is blocked and selectively forwarded	√ (USRP2)
IMDGuard	IMDs	Tens of ms	20 cm	Guard notices a spoofing attack	∼ (MICAz)
Warlock Duke	Improvised explosives (IEDs)	n/a	n/a (100 m)	Any signal in guarded freq. bands	√ (custom)
Blocker Tags	RFID	300 µs	20 cm	Tag query to protected prefix	× (tag)
RFID Guardian	RFID	300 µs	1 m	Tag query to tag in ACL	× (handheld)
Jamming for Good	Sensor networks	5 ms	2–3 m	Address+RSSI of registration packet	√ (MICAz)
WiFire	**Sensor networks**	**64 µs**	**10–20 m**	**Per-packet decision (header+payload)**	**√ (USRP2)**

Table 2.1.: Comparison of related protection systems using physical layer responses.

Thus, in contrast to these higher-layer approaches, the wireless firewall must operate on the physical layer to achieve its goal.

Our approach supports more sophisticated access blocking rules and does not rely on valid frame semantics to deny access but uses physical-layer jamming. Furthermore. our approach is immune to malicious nodes which discard disassociation and de-authentication frames to remain connected.

2.2.2. Implanted Medical Devices

There are several works that consider securing IMDs to ensure the safety and privacy of patients with IMDs. This is because the devices may otherwise send out confidential information or even be reconfigured wirelessly against the patients wishes (Fu, 2009; Halperin, Heydt-Benjamin, et al., 2008).

Heartbeats. IMDs face challenges similar to WSNs, namely low computational resources and limited energy. Gollakota, Hassanieh, et al. (2011) describe an external IMD protection system, or IMD “shield,” that allows to regulate access to the device using selective interference, protecting it from malicious readers. The shield is a battery-powered device that is worn close the implanted device (with a distance of less than 20 cm), e.g., in the form of a pendant. It acts as a proxy that simultaneously receives and destroys any packet related to the protected IMD. If the packet is going to the IMD, the shield checks whether the reader is trusted and forwards the packet. If the packet originated from the IMD, the packet is forwarded in encrypted form to the querying reader to protect the patients’s privacy. A USRP2-based prototype system is presented, showing that an attacker can only succeed if it uses high transmit power and close proximity.

IMDGuard. Xu et al. (2011) also describe an external guardian system that protects IMDs from untrusted readers. It supports an IMD during cryptographic operations, and uses selective interference when an attacker tries to disturb this operation. This system also works transparently, i.e., there is no need to modify the IMD after it is implanted. However, in contrast to the concept of the wireless firewall, the external guardian must be placed in close proximity to the protected device, which constitutes a reader–device setting instead of a networked setting with several devices that we are aiming for. Additionally, they do not offer programmable security policies: IMDGuard uses packet timing to detect malicious packets. However, the concept relies on cryptographic protocols and uses the physical layer response only as a counter-measure to spoofing attacks. In contrast to these works, we offer configurable policies for several devices in a distributed sensor network setting.

2.2.3. Radio Frequency Identification Tags

Radio frequency identification (RFID) tags are well known for their extreme resource limitations and lack of reprogrammability. Several contributions show that remote protection can be used in this context as well.

Blocker Tags. To protect the privacy of RFID tags from malicious readers, Juels, Rivest, et al. (2003) introduce the "blocker tag" to prevent the tag discovery by confusing readers with artificial collisions. The attacker queries a prefix of node IDs (e.g., the first two bits), and on collisions the reader refines the prefix, such that the blocker tag can force the reader to traverse the full address space by generating intentional interference. In this case, the reader assumes that several tags are present and that more specific queries are required; but as the collisions are triggered for all queries, the reader is forced to search the complete address space (e.g., 2^{64} addresses). Juels and Brainard (2004) extend this concept to signal privacy policies to benign readers.

RFID Guardian. Rieback et al. (2005; 2007) offer a similar solution but support the protection of configurable sets of RFIDs (blocker tags only support address blocks). A battery-powered handheld device, the "RFID Guardian," monitors all queries and interferes with a tag's response to hide its presence from malicious readers. The system enables the use of configurable access control lists (ACLs) for arbitrary sets of RFID tags that specify which readers are allowed to interact with which tags. Again, this is a reader–device setting with small distances (up to 1 m for RFID Guardian). However, the main difference to our work is that these schemes do not operate on a per-packet basis: malicious queries are actually received by the tag, and only the tag's response is blocked. This is not problematic because RFID tags commonly do not keep state information. With our implementation, we can prevent sensor motes from receiving any malicious packet, also protecting their internal state. With the wireless firewall, we want to protect devices starting with the very first packet.

2.2.4. Wireless Sensor Networks

Jamming for Good. A closely related work protects sensor motes from spoofed packets (based on RSSI information) using selective interference (Martinovic, Gollan, et al., 2008; Martinovic, Pichota, and Schmitt, 2009). This work showed that injection attacks against WSNs can be mitigated in a cooperative manner by jamming packets with suspicious signal fingerprints.

In these works, sensor devices support each other to prevent impersonation attacks, i.e., an attacker pretending to be part of the network by spoofing its source address to look like an existing network device. When a sensor device wants to send data, it first sends a "reservation packet" to notify others of its transmission wish before it starts sending the data packet. Sensor devices in the vicinity compare the claimed source address and the physical signal fingerprint of the reservation (which depends on the device's location and is hard to spoof) with previously observed measurements. In case of a mismatch (i.e., a potential attack), the device schedules a concurrent transmission with the data packet, intercepting the spoofed packet. This approach explores the concept of over-the-air support to increase security, but also forces an expensive mode of operation on the network.

Because standard sensor motes are considered in their experiments, a special admission frame prior to the actual data frames is necessary to relax the timing constraints in order to decouple the jamming decision from the actual jamming process The protection is performed by the network motes themselves, analyzing if the RSSI signature of the registration packet matches with the claimed source address, and scheduling the transmission of an interfering packet concurrent with the data packet in case of a mismatch. However, the requirement that motes must receive packets that are not addressed to them and send packets for interference is expensive in terms of energy. So, while the goals are similar, the approach is different. We explore the use of specially designed wireless firewall devices that provide a per-packet central enforcement of access policies, without requiring a custom MAC protocol. The wireless firewall explores the use of a dedicated security infrastructure to protect wireless devices in a fully transparent way: no modifications are required, the devices do not even need to know that WiFire is protecting them.

Home jamming. Brown et al. (2013) present a jamming system that protects a home network from injection attacks in the spirit of Jamming for Good introduced above. The authors present a dedicated device that focuses on the MAC protocol of the ZigBee standard, which is prevalent in the context of home automation.

2.2.5. Other Applications

In the military context, the U.S. military employs mobile jamming systems to protect convoys in Iraq from improvised roadside bombs, stopping a bomb trigger signal from arriving at the bomb's receiver (LaShomb, 2006).[1] However, not much is known about the system's implementation and operation.

2.3. Summary

The application of physical layer security has seen a steady increase of research contributions that explore the feasibility and fitness of this approach. We have seen that while the wireless firewall concept builds on a strong foundation of recent results, there are a great number of research questions that are introduced by the fact that the wireless firewall protects from a distance, rather than from a close proximity.

---

[1] A news article on IED jammers used in the Iraq campaign is available at `http://edition.cnn.com/2007/TECH/08/13/cied.jamming.tech`

Part I.

Conception and Analysis of Wireless Firewalls

3. Protection Concept

Contents

5.1. Protection Constraints 67
5.2. Attacker Models 69
5.3. Attacker–Guardian Asymmetry 69
5.4. Summary 73

The purpose of this chapter is to provide a bird's eye view on the operation of wireless firewalls. At the same time, this is the right place to discuss some basic architectural alternatives to such protection systems and provide the rationale for top-level design decisions we have taken.

3.1. Protection Scenario

We consider a wireless sensor network scenario with three types of devices:

1. Wireless sensor nodes that perform a distributed sensing application; these nodes are the target of attacks and hence are referred to as *victim* nodes v.
2. *Attacker* nodes a that attempt to send malicious packets to victim nodes v to disrupt their intended network operation.

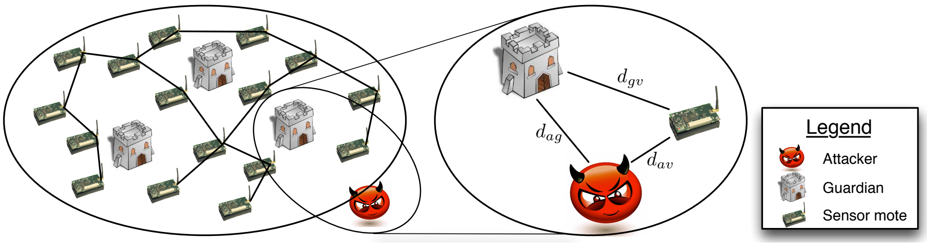


Figure 3.1.: Sensor network deployment with co-located guardian nodes.

3. *Guardian* nodes g that want to prevent victim nodes v from receiving malicious packets from attackers a.

The guardian uses a limited pulse of interference to protect the devices in the vicinity. Of special interest in this problem setting is the spatial aspect. We assume that the attacker cannot move arbitrarily close to the protected devices.

3.2. Using Interference for Access Control

To contrast the novel approach required for a wireless firewall to the standard (packet filter) firewall, we discuss the mode of operation of these conventional devices. The standard firewall basically has to execute the following three steps:

Step 1: Receive and *store* a packet.

Step 2: Classify the packet into trusted or untrusted.

Step 3: If trusted *forward* the packet, else *block* it.

A fundamental assumption thus is that firewalls are store-and-forward devices. In our approach we deviate from this assumption and essentially implement the wireless firewall as a cut-through device altering the three basic steps into:

Step 1': Analyze the *signal stream.*
We need to operate on the physical layer instead of the network layer since otherwise a transmission is over before we even had the chance to detect a policy violation.

Step 2': *Detect* untrusted signal features.
The detection of policy violations literally has to be done on the fly.

Step 3': If untrusted feature detected, *interfere.*
We are not in the forwarding path of the wireless traffic, so in order to virtually block certain packets we interfere with the reception.

Packets are destroyed in Step 3' by triggering the packet checksum mechanism of IEEE 802.15.4. Since the CRC of packets at victim nodes is erroneous when at least one symbol is wrong, packets that are hit strong enough by interference from a guardian node are discarded by the victim receivers.

Thus, the wireless firewall proceeds in three steps in its operation: *(i)* spotting packets, *(ii)* distinguishing friends from foes, and *(iii)* destroying malicious packets. We will discuss each step in more detail.

Detecting packets

The system continuously monitors the wireless medium, spotting packets for analysis. It exploits the fact that an attacker must be constructive to inject packets, it must comply to the correct frame format to ensure that its packets have a chance to be received. In this sense, the system scans the medium for the physical layer header of IEEE 802.15.4, consisting of a preamble and start-of-frame delimiter (SFD).

Jamming decision

In order to form a correct decision, it is necessary to have as much access to the packet content as possible. Based on this information, a set of rules defines packets that are deemed malicious. Similar to standard firewalls, it is also conceivable to block all traffic that does not match any of the rules.

Preventing packet reception

We propose to prevent packet receptions by controlled interference. The intended receiver then either misses the packet completely or detects a corrupted frame with a failed integrity check. This approach to prevent receptions gives the wireless firewall a *transparency* property: protected devices do not need to know about the wireless firewall presence, no protocol adaptations or control messages from the firewall are necessary. Therefore, the system can be added to existing legacy networks to patch their security problems, filtering out malicious packets. During normal operation, the system monitors the channel passively and reacts to immediate threats only.

Benefits of the Jamming Approach

In a standard firewall blocking comes for free: just discard untrusted packets silently. The wireless firewall has to become active when we discover a policy violation: we jam an untrusted packet such that it cannot be received properly at a protected node. In a certain sense jamming can be seen as a physical layer warning from the firewall to a node within its network. This may motivate the idea that we could provide for such a warning also on higher layers, effectively relieving us from the burden to jam a “passing by” packet in real-time. In fact, some of the commercial wireless intrusion detection systems (e.g., AirDefense) use IEEE 802.11 DEAUTH messages in an attempt to set stations associated to a rogue access point free again. Nevertheless, we opted for jamming as a blocking primitive for the following reasons:

- Jamming does not carry any semantics; higher layer warnings require interpretation by the receiving node, as such they are prone to new attacks exploiting this interpretation.
- Jamming is idempotent, attacks against jamming equal jamming itself; this results in the fact that any persistent attack is reduced to a denial of communication in the protected wireless network, effectively locking it up, which is the typical response of a firewall if its protected network is under heavy attack.
- Jamming protects many receivers at the same time; this is owing to the fact that the jamming signal is inherently broadcast.
- Jamming allows for a transparent protection; what is received correctly can be trusted immediately without requiring to wait for a possible warning, which could complicate protocol processing at end systems and would result in a loss of firewall transparency.

3.3. Research Challenges

In this thesis, we set out to answer the following research questions to develop a wireless firewall:

- **Understanding packet receptions:** We want to identify the factors that govern packet reception.
- **Optimal jamming:** What methods can be used to ensure that a packet is indeed dropped?
- **Nature of protection:** Under what circumstances does such a protection on the physical layer work? In particular, what kind of attacks can be prevented?

3.3.1. Understanding Packet Reception

To build an effective wireless firewall, it is important to understand the reception process at the receiver, and especially to identify conditions that ensure that a malicious packet is destroyed. For this reason, we first set out to identify these conditions analytically. This challenge is addressed in Chapter 4.

3.3.2. Optimal Jamming Techniques

Using the insights from the reception process analysis, another important aspect relates to the choice of the interference waveform and interference patterns. It is well known that certain waveforms are more effective at causing bit errors than others (Poisel, 2011). The effectiveness of the interference waveform depends on the particular modulation scheme and the design of the receiver. Advantageously, wireless sensor nodes are generally not designed to suppress in-band interference well. They only low-pass filter the baseband signal and send the output directly to the demodulator (DeBruhl and Tague, 2011). We see in our experimental evaluation in Section 8.2.3 that a continuous sine wave, a waveform that is generally known as being ineffective, happens to destroy packets even more effectively in IEEE 802.15.4 with the same power budget than band-limited white noise or modulated signals. Using a very narrowband waveform like this provides the advantage that other technologies like WLAN are able to efficiently suppress this kind of interference (Karhima et al., 2004) and hence remain unaffected by the guardians.

3.3.3. Nature of Protection

When considering the security protocols that employ cryptographic primitives, the necessary conditions for provable security are readily stated. For example, given that a secret key is not leaked to an adversary, the confidentiality of an encrypted link can be established in a straightforward way. This modularity enables the definition of more complex protocols from cryptographic building blocks, which can still be analyzed for their security.

In contrast, a protection at the physical layer often does not support a statement that a protocol is secure under any conceivable circumstance. Rather than fully preventing an attack, the difficulty to perform an attack is increased by a physical layer security scheme, and the goal is to raise the costs of an attack to regions that negates any benefit the attacker might have from its attack. For example, when applying spread spectrum to a protected radio signal in order to make it more robust against interference, the attack power that is necessary to destroy the communication is increased by a certain factor (e.g., by a factor of 100 or 1,000), but a strong or close attacker can still succeed. In this sense, the goal of a protection on the physical layer is to reduce the risks of using wireless connectivity, but not to guarantee that no attack is possible. To better understand wireless firewalls, we make the effort to find a suitable metric to analyze the protection that a wireless firewall offers. We evaluate this protection against two generic attackers a stealthy attacker that hides from detection of the wireless firewall, and a brute force attacker that negates the protection by overcoming the effects of RF interference. For these attackers, we analyze how the risk of a successful attack is reduced.

3.4. Technical Challenges

Clearly, jamming must be used with care, which brings about a list of desirable, but challenging attributes that must be solved technically:

- **Reactive/Selective**: the wireless firewall needs to reactively jam depending on whether policy violations are detected in the signal stream. Consequently, it only jams a subset of the overall packet stream selectively.
- **Efficient**: The jamming duration shall be kept minimal in order to keep interference with concurrent legitimate transmissions low.
- **Effective**: We need to jam successfully with high probability (i.e., detect all and jam all packets).

On the other hand, this mode of “friendly jamming” can also support us to make the task of successful jamming easier. In fact, the devices protected by the wireless firewall *want* to be jammed. This brings us into a convenient position compared to an adversarial jammer, because we can assume to have access to knowledge, e.g., secret hopping frequencies. Thus anti-jamming measures like spread-spectrum modulation or even recent interference cancellation techniques (HALPERIN, ANDERSON, et al., 2008) become no issue. Furthermore, we can control the placement, the sending power, and the antenna configurations of the wireless firewall to plan for an effective jamming. And last but not least, we can employ several instances to make it very hard for an attacker to circumvent its transmission to be jammed by the protection system.

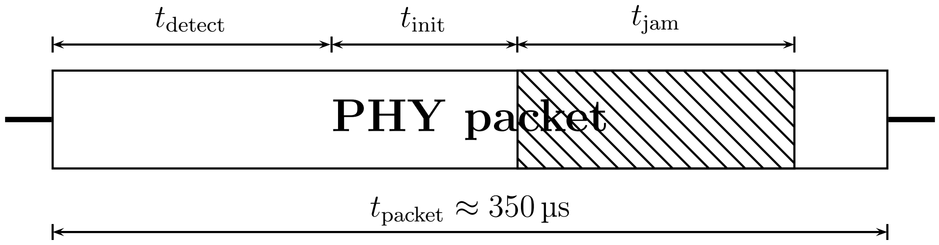


Figure 3.2.: Time constraints of IEEE 802.15.4: the reactive jammer must detect a transmission, initiate the jamming process and interfere with the transmission to prevent a packet reception.

3.4.1. Reactive and Selective Operation

We set ambitious goals for our wireless firewall system: an accurate detection of RF transmissions as well as reliable and precise jamming, all while a packet is still on the air.

Policy enforcement

The system must enable the definition of central policies and must be able to enforce them. The wireless firewall must take responsibility from the end-points and manage it centrally. We want a central point of control and configuration to prevent the reception of transmissions that do not comply with the specified policy. The system must provide a complete enforcement of the defined policy, in particular, it must detect and prevent all policy violations. Transmissions that comply with the policy must be received by the destinations. Due to the broadcast nature of the wireless channel, this shows the major departure that the system must take from the wired firewall analogy: all transmissions that are not actively blocked reach their destination.

Timing challenges

In order to achieve all three steps of operation that the wireless firewall performs per packet, the system must be fast. To get an impression of the timing requirements, refer to Figure 3.2. The system must detect a transmission and decide whether it must be jammed or not (with the required time t_{detect}), schedule and initialize the sending of a jamming signal (with delay t_{init}), and send a short, yet sufficient jamming burst to destroy the packet (t_{jam}), all while it is being transmitted.[1] The concurrent jamming must exceed the shortest interference time t_{jam}^{min} to cause a packet loss. Therefore, we require

$$t_{\text{detect}} + t_{\text{init}} + t_{\text{jam}}^{min} \leq t_{\text{packet}},$$

i.e., to react quickly enough to hit the packet for the minimal required jamming duration. In the case of IEEE 802.15.4, the shortest packets are ACKs, with a duration of $t_{\text{packet}} = 352\,\mu\text{s}$. This strict deadline means that the system design must support real-time operation to perform all three steps for each packet individually, before it can arrive at the receiver.

3.4.2. Efficient Operation

The drawback using controlled interference is that we introduce more interference than necessary, which could be problematic for the operation of the sensor network or other wireless networks in the same frequency band. One system design goal for the wireless firewall is therefore to make this interference as "friendly" as possible by using very short periods of interference and using waveforms that are effective at the target sensor nodes but negligibly disturbing for other wireless networking technologies operating in the vicinity. We expect the wireless firewall to be friendly to co-existing networks. This is important because an unintentional derogation of co-existing networks could prevent a certification of guardian devices or reduce the acceptance of their deployments; we discuss legal issues at the end of this thesis. We show that, by using jamming waveforms that are efficient against one technology but have only limited effect of other technologies, that this can be achieved.

Reflections on Guardian Interference

To achieve "friendly" interference, the duration of the blocking signals should be as short as possible. In IEEE 802.15.4, one erroneous symbol already leads to a wrong CRC checksum, and the packet at the victim node is discarded. Hence, a guardian who detects an unauthorized packet ideally needs to interfere with that packet for the duration of a single symbol (16 µs). We investigate in Section 8.2.3 that, under real-world settings, it requires a slightly higher duration of selective guardian interference to effectively block all undesired packets. However, this short interference period is still limited and negligible compared to the signal duration of the packet emitted by the

---

[1] We do not consider the propagation delay in our analysis, we assume short distances between all devices.

attacker. A wireless sensor network will hence not be significantly impacted since IEEE 802.15.4 nodes perform carrier sensing and should not transmit concurrently with the attacker anyway.

3.4.3. Effective Operation

To be effective the guardians should reliably detect all packets on the air and interfere with all unauthorized ones. At the same time, the interference should be limited in order to minimize possible effects on concurrent communications on the same frequency band. This poses a quite different problem compared to proactive jamming performance evaluations in the literature BAYRAKTAROGLU et al. (2008) and LAW et al. (2009). The reactive jammer must be able to destroy transmissions at the receiver even if a sender has already started a transmission. We analyze the causes of loss on the physical layer and identify the jamming signal that causes the minimal packet reception ratio (PRR). We evaluate the minimal jam duration and show that the required jamming burst can be as short as $t_{\text{jam}}^{min} = 26\,\mu\text{s}$ to ensure a PRR of $0\,\%$.

4. Understanding the Reception of Colliding Packets

Contents

6.1. Design Space of Wireless Firewalls 77
6.2. The Architecture of the WiFire System 82
6.3. Summary 85

4.1. Introduction

In this chapter, we aim identify the reasons for packet loss at the receiver caused by collisions on the physical layer. For this purpose we develop a detailed model of the physical layer of IEEE 802.15.4, in line with our goal to develop a system for WSN protection. The motivation for this approach is twofold. First, with this knowledge, we can identify the conditions for loss and use this knowledge to accurately model the protection offered by wireless firewalls, as well as finding the most efficient approach for jamming. For this reason, we start out with a model that is as general as possible, with a large number of parameters. The goal is to identify parameters that matter most, and use this knowledge to develop a simplified model to analyze the protection properties of wireless firewalls in the next chapter. Second, identifying the reasons for packet loss in collisions is also of general interest in wireless network research because of a novel avenue of research where interference is controlled instead of avoided. This perception shift led to the exploitation of concurrent transmissions.

4.1.1. The Case of Concurrent Transmissions

Conventional wireless communication systems consider packet collisions as problematic and try to avoid them by using techniques like carrier sense, channel reservations (virtual carrier sense, RTS/CTS handshakes), or arbitrated medium access (TDMA, polling). The intuition is that concurrent transmissions cause irreparable bit errors at the receiver and render packet transmissions undecodable. However, researchers have found that this notion is too conservative. If the power of the signal of interest exceeds the sum of interference from colliding packets by a certain threshold, packets can in general still be received successfully despite collisions at the receiver. This effect, referred to as the *capture effect* (LEENTVAAR and FLINT, 1976), has been explored extensively and validated in many independent practical studies on various communication systems such as IEEE 802.11 (FOO and HUANG, 2008; GUMMADI et al., 2007; KOCHUT et al., 2004; J. LEE et al., 2007) and IEEE 802.15.4 (GEZER et al., 2010; MAHESHWARI et al., 2008; SON et al., 2006).

Over the past years, the view on packet collisions has therefore changed considerably. Since it is possible for some or even all packets in a collision to survive, there are opportunities to increase the overall channel utilization and to improve the network throughput by designing protocols that carefully select terminals for transmitting at the same time (SHA et al., 2009; VUTUKURU et al., 2008). The benefits and potential performance improvements of concurrent transmission are not just of theoretical interest but have been demonstrated practically and adopted in application areas such as any-cast (DUTTA, DAWSON-HAGGERTY, et al., 2010; DUTTA, MUSĂLOIU-E., et al., 2008), neighbor counting D. WU et al. (2014), or rapid network flooding (DODDAVENKATAPPA et al., 2013; FERRARI et al., 2011; LU and WHITEHOUSE, 2009; WANG, Y. HE, et al., 2012; WANG, Y. LIU, et al., 2014), especially in the context of wireless sensor networks (WSNs).

Although protocols that exploit concurrent transmissions have shown the potential to boost the overall performance of existing wireless communication systems, their success cannot be explained with capture threshold models based on the Signal to Interference and Noise Ratio (SINR) alone. Recent studies have shown that, while the relative signal powers of colliding packets indeed play an important role in the reception probability, other factors are also of major importance. For example, several experimental studies report that the relative timing between colliding packets has a significant influence on the reception probability (J. LEE et al., 2007; SANTHAPURI et al., 2008). Others report that the coding (DAVIS and GRONEMEYER, 1980) or packet content (DUTTA, DAWSON-HAGGERTY, et al., 2010) may also greatly influence the reception performance in the presence of collisions. Further factors such as the carrier phase offset between a packet of interest and colliding packets also need to be considered (PÖPPER et al., 2011).

4.1.2. Modeling Approach

In this chapter, we strive to provide a comprehensive model accounting for all these factors, focusing on packet collisions in IEEE 802.15.4 based WSNs. Such a model will allow protocol designers to better understand the root causes of packet reception and exact conditions under which concurrent transmissions actually work, and thus to design optimal protocols based on these factors. While previous studies (GUMMADI et al., 2007; MAHESHWARI et al., 2008; WHITEHOUSE et al., 2005; YUAN and HOLLICK, 2013; ZIMMERLING et al., 2013) also looked at factors that determine the success of concurrent packet reception, these works are either based on practical experiments and have therefore led to empirical models that cannot be generalized easily, or derived simplified models that do not account for all impact factors. This work advances the field by providing a unified analytical model accounting for the major factors identified above (see also Section 4.2). Our model (→ Section 4.4) is based on a mathematical representation of the physical layer using continuous-time expressions of the IQ signals entering the receiver's radio interface. This fundamental and comprehensive model allows to represent an arbitrary number of colliding packets as a linear superposition of the incoming signals.

A major contribution of this chapter is a closed-form analytical representation of the bit decision variable at an optimal receiver's demodulator output based on these IQ signals (→ Section 4.5). This result enables the deterministic computation of the bit demodulation decision and hence to compute the actual performance of concurrent transmissions for any colliding parameter constellations. Having a bit-level model of reception is not only beneficial for the comprehension of the collision process, it also contributes to application areas where a precise bit-level analysis is needed, such as partial packet reception (JAMIESON and BALAKRISHNAN, 2007), understanding bit error patterns in low-power wireless networks (HERMANS et al., 2014; SCHMIDT et al., 2013; K. WU et al., 2012), understanding the network performance in heavily used spectrum conditions ZHAO and GOVINDAN (2003), or signal manipulation attacks at the physical layer (PÖPPER et al., 2011).

Using our model, we explore the parameter space of the reception of MSK-modulated colliding packets considering both uncoded and Direct Sequence Spread Spectrum (DSSS) based systems (→ Section 4.6), analyzing the influence of the parameters on the resulting packet reception ratio (PRR) for concurrent transmissions. While the analysis shows that our model agrees with experimental results in the literature, it also provides much more detailed insights into the performance characteristics of protocols that exploit collisions (Doddavenkatappa et al., 2013; Dutta, Dawson-Haggerty, et al., 2010; Ferrari et al., 2011; Lu and Whitehouse, 2009; Wang, Y. He, et al., 2012; Wang, Y. Liu, et al., 2014). In particular, we show that the good performance of these protocols should be attributed equally to coding (e.g., DSSS) and power capture. In addition, based on our analysis we identified parameter constellations where concurrent transmissions work reliably. We therefore propose a generalization of the traditional capture threshold model based on the power ratios towards a *capture zone*. Capture zones result from the model insight that reception success does not depend on the power ratio between interfering signals alone, but on the time and phase offsets of sender and receiver as well. We discuss parameter settings for an optimal protocol design (→ Section 4.7). Finally, we perform an experimental evaluation (Section 8.1) using the system we develop in the next chapters.

4.2. Parameters of the Analysis

Different factors influence the probability of a successful reception under collisions. This section discusses the main factors that have been identified in the literature. Subsequently, we consider them jointly in our mathematical model to predict the outcome of concurrent transmissions.

Signal Power Ratio

The signal power is a crucial factor for successful reception in general, and it plays a major role in the reception under collisions as well. SINR-based models are widely used to model the packet reception in a shared medium, for example in the Physical Model (Gupta and Kumar, 2000) and its variants (Cardieri, 2010; Maheshwari et al., 2008). The classical SINR model states that a stronger signal is received if its signal power P_s exceeds the channel noise P_n and the sum of interfering signal powers $\sum_i P_i$ by a given threshold, i.e.,

$$\frac{P_s}{P_n + \sum_i P_i} > \delta_{\text{SINR}}.$$

This simple model is accurate for uncorrelated interfering signals such as additive white Gaussian noise (AWGN). However, when the interference is correlated (such as colliding packets), this model is not always accurate and further factors must be considered (Gummadi et al., 2007; J. Lee et al., 2007; Santhapuri et al., 2008).

Signal Timing Offsets

The relative timing of colliding packets greatly influences the reception process. This is because the receiver locks onto a packet during the synchronization phase at the start of the transmission. If a stronger signal arrives later, it captures the receiver and disturbs the first packet reception, and both packets in the collision are lost. Thus, in packet radios, power capture alone is not sufficient for successful reception, rather the receiver must be synchronized and locked onto the captured signal as well. Several research contributions analyze possible collision constellations and their effect on packet reception (J. Lee et al., 2007; Santhapuri et al., 2008), and propose a new receiver design that releases the lock when a stronger packet arrives, discards the first and receives the second packet, the so-called *message-in-message (MIM) capture* (J. Lee et al., 2007; Whitehouse et al., 2005). Subsequent works apply these insights to improve network throughput. For example, Manweiler et al. (2012) propose collision scheduling to ensure that MIM is leveraged, thus increasing spatial reuse.

Channel Coding

A further factor that influences packet reception success is bit-level coding. For example, in DSSS systems a group of b bits is encoded into a longer sequence of B chips (Proakis and Salehi, 2007). The benefit of this approach is that resilience to interference is increased because the chipping sequences can be cross-correlated at the receiver, which effectively filters out uncoded noise. However, DSSS systems require interfering signals to be uncorrelated, e.g., signals without coding or with orthogonal chipping sequences (as in CDMA), to achieve their theoretical coding gain. Another possibility is a sufficient time offset between interfering packets with the same coding; this phenomenon is known as *delay capture* (Davis and Gronemeyer, 1980). As networking standards such as IEEE 802.11 and IEEE 802.15.4 generally use DSSS with identical codes for all participants, existing experimental works on collisions and capture observe the effects of DSSS implicitly.

Packet Contents

Experimental results show that packets with identical payload and aligned starting times result in good reception performance and reduced latency in broadcast scenarios. For example, Dutta, Dawson-Haggerty, et al. (2010) show that short packets can be received in such collisions with a

PRR over 90 %, thus enabling the design of an efficient receiver-initiated link layer. Similarly, the latency of flooding protocols widely used in WSNs can be greatly reduced (Ferrari et al., 2011; Wang, Y. He, et al., 2012). In these works, experiments in IEEE 802.15.4 networks reveal that the tolerable time offset between concurrent messages is small (approx. 500 ns), which adds challenges to protocol design and implementation. These insights also show that capture and packet synchronization alone are not sufficient to explain the performance of these protocols, and bit-level modeling that also includes signal timing and content is necessary.

Carrier Phase Offsets

Considering the reception of bits at the physical layer, knowledge of the carrier phase at the receiver is crucial for successful reception of phase modulated signals because the information is carried in the phase variations of the signal, such that these offsets should be minimized (Proakis and Salehi, 2007). Typically this is achieved during the synchronization phase of packet reception, and thus existing capture models have omitted phase offsets. However, there are two reasons why this is not sufficient. First, in novel protocols exploiting packet collisions, the synchronization during the preamble is not always able to succeed. Second, there are other new applications of concurrent transmissions that try to abandon the synchronization procedure. For example, Pöpper et al. (2011) investigate the possibility of manipulating individual message bits on the physical layer, and conclude that carrier phase offsets are the major hindrance to do so reliably. The sensor nodes communicate using IEEE 802.15.4 on the 2.4 GHz band and we assume that all nodes send and receive on the same channel. We further assume that the victim sensor nodes v act in compliance to the IEEE 802.15.4 standard and that they discard received packets for which the CRC checksum is incorrect. The CRC checksum is a 16 bit field calculated over a packet's payload and headers. The CRC checksum is erroneous when a packet has at least one symbol error (we do not consider error coding mechanisms such as the concept of Liang et al. (2010)). We further assume that the victim nodes communicate without header encryption, the wireless firewall can therefore eavesdrop and decode any transmitted packets. This is no limitation of our concept for two reasons: First, header encryption is generally not used in wireless networks because all messages must be received, decrypted, and checked for integrity using all available link keys, which is extremely inefficient; the IEEE 802.15.4 standard (2006, §5.5.6) only considers *data* confidentiality as a security service. Second, even if header encryption is used the wireless firewall can be given access to

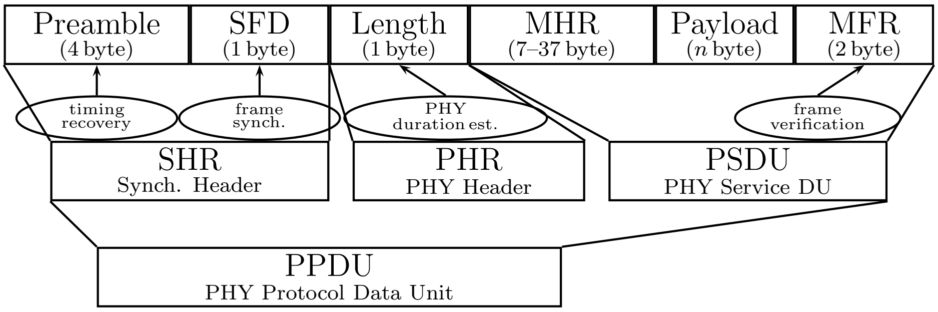


Figure 4.1.: The structure of IEEE 802.15.4 frames.

the network's cryptographic material because they are part of the network infrastructure. Then a guardian can decrypt packets during transmission (there are efficient FPGA implementations of AES available (Hodjat and Verbauwhede, 2004)).

4.3. Background on IEEE 802.15.4

Before going into details, we briefly cover aspects of the IEEE 802.15.4 physical layer that are necessary for the later discussion of jamming against such networks. Although IEEE 802.15.4 (2006) defines four different physical layers for the wireless interconnection of devices in wireless personal area networks (WPANs), we limit ourselves here to the 2.4 GHz PHY because of its widespread use. The standard defines 16 channels labeled Channel 11–26, with a bandwidth of 2 MHz each and a 5 MHz interspacing. Bytes in the PHY protocol data unit (PPDU) are transmitted at a rate of 250 kbps. They are divided into groups of 4 bit, which are then mapped to a set of 16 symbols. These symbols are spread with the corresponding 32 bit pseudo-noise (PN) chipping sequence, i.e., IEEE 802.15.4 uses direct sequence spread spectrum (DSSS) with a spreading factor of eight. This stream of chips is then modulated onto the carrier using O-QPSK with half-sine pulse-shaping, and transmitted over the wireless medium to the receiver.

4.3.1. Reception Process

The reception process can be explained in terms of the PPDU headers (SHR and PHR), shown in Figure 4.1. The essential components are shown in more detail, and ellipses show the required reception steps for these components. When a carrier is detected, the receiver synchronizes with the predefined

Table 4.1.: Chipping sequences used in the 2.4 GHz PHY of IEEE 802.15.4.

Symbol ξ	Bits	Chipping sequence bits[a][b] $(c_{\xi,0},\ldots,c_{\xi,31})$																													
0	0000	**1**	**1**	**0**	**1**	1	0	0	1	1	1	0	0	0	0	1	1	0	1	0	1	0	0	1	0	0	0	1	0	1	1	1	0
1	0001	1	1	1	0	**1**	**1**	**0**	**1**	1	0	0	1	1	1	0	0	0	0	1	1	0	1	0	1	0	0	1	0	0	0	1	0
2	0010	0	0	1	0	1	1	1	0	**1**	**1**	**0**	**1**	1	0	0	1	1	1	0	0	0	0	1	1	0	1	0	1	0	0	1	0
3	0011	0	0	1	0	0	0	1	0	1	1	1	0	**1**	**1**	**0**	**1**	1	0	0	1	1	1	0	0	0	0	1	1	0	1	0	1
4	0100	0	1	0	1	0	0	1	0	0	0	1	0	1	1	1	0	**1**	**1**	**0**	**1**	1	0	0	1	1	1	0	0	0	0	1	1
5	0101	0	0	1	1	0	1	0	1	0	0	1	0	0	0	1	0	1	1	1	0	**1**	**1**	**0**	**1**	1	0	0	1	1	1	0	0
6	0110	1	1	0	0	0	0	1	1	0	1	0	1	0	0	1	0	0	0	1	0	1	1	1	0	**1**	**1**	**0**	**1**	1	0	0	1
7	0111	1	0	0	1	1	1	0	0	0	0	1	1	0	1	0	1	0	0	1	0	0	0	1	0	1	1	1	0	**1**	**1**	**0**	**1**
8[c]	1000	**1**	**0**	**0**	**0**	1	1	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	1	1	0	1	1	1	1	0	1	1
9	1001	1	0	1	1	**1**	**0**	**0**	**0**	1	1	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	1	1	0	1	1	1
10	1010	0	1	1	1	1	0	1	1	**1**	**0**	**0**	**0**	1	1	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	1	1
11	1011	0	1	1	1	0	1	1	1	1	0	1	1	**1**	**0**	**0**	**0**	1	1	0	0	1	0	0	1	0	1	1	0	0	0	0	0
12	1100	0	0	0	0	0	1	1	1	0	1	1	1	1	0	1	1	**1**	**0**	**0**	**0**	1	1	0	0	1	0	0	1	0	1	1	0
13	1101	0	1	1	0	0	0	0	0	0	1	1	1	0	1	1	1	1	0	1	1	**1**	**0**	**0**	**0**	1	1	0	0	1	0	0	1
14	1110	1	0	0	1	0	1	1	0	0	0	0	0	0	1	1	1	0	1	1	1	1	0	1	1	**1**	**0**	**0**	**0**	1	1	0	0
15	1111	1	1	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	1	1	0	1	1	1	1	0	1	1	**1**	**0**	**0**	**0**

---

[a] The IQ chips are shown interleaved, dark background denotes in-phase chips.

[b] The sequences are shifted cyclically by four chips, **bold chips** are the first chips of symbol 0 (or 8) for reference.

[c] The second half of the chipping sequences are equal to the first except that quadrature bits are inverted.

preamble sequence (eight "`0`" symbols in the standard) to compensate the phase and frequency offset of the incoming transmission. This is necessary as the sender and receiver are not synchronized; with this step, the receiver recovers the timing of both chips and symbols, and the symbol clock adjusts to the symbol boundaries. The receiver then expects a specific two symbol sequence, the Start-of-Frame Delimiter (SFD) that marks the beginning of the PHY header (PHR) and the following MAC layer frame. This process is called frame synchronization. The PHR consists of 7 bit containing the length of the following PHY service data unit (PSDU), which allows for duration estimation of the transmission. At each symbol clock tick, a decision is made as to which of the 16 possible symbols was the one most likely transmitted during the last period. At the end of the PSDU, the MAC footer (MFR) contains a 16 bit integrity checksum (frame check sequence, FCS) using CRC16 that verifies whether the frame is received without errors. If this is the case, the received frame is passed to the higher layers. We refer the reader to (OH and S.-G. LEE, 2005; SCHMID, 2006) for a more detailed treatment of the IEEE 802.15.4 PHY and the properties of different transceiver designs.

4.3.2. Causes of Loss on the IEEE 802.15.4 PHY

To understand the underlying reasons for the effectiveness of different jamming approaches, we need to identify the causes of packet loss on the IEEE 802.15.4 physical layer. For a study focused on the IEEE 802.11 physical layer, refer to the work of GUMMADI et al. (2007).

Symbol Misdetection and Integrity Errors

Once a frame is detected, the most likely symbol that has recently been transmitted is chosen on each symbol clock tick. Strong jamming transmissions concurrently with a symbol cause a symbol misdetection if a sufficient number of chips are flipped, consequently generating bit errors on higher layers. Integrity checks such as the CRC16 check of IEEE 802.15.4 detect these errors, resulting in a packet drop as no forward error correction (FEC) is used in IEEE 802.15.4. Thus, a single symbol error is sufficient to destroy a complete packet. Similarly, the MHR contains addressing information and the frame type, which can trigger packet drops if damaged even before the integrity of the frame is checked.

Failed Timing Recovery

If a jammer interferes with the preamble at the beginning of the transmission, it can cause the timing recovery to fail. A corrected phase and frequency offset are crucial for a successful packet reception, as otherwise symbol decisions are based on sub-optimal (non-peak) sampling times that decrease the SNR dramatically. This makes the symbol decisions more prone to errors, even if the jammer interfered during the preamble only. Additionally, a failure to lock onto a transmission can also cause the frame synchronization (discovering the SFD) to fail, such that a packet is overheard completely even if the incoming signal is strong.

Frame Synchronization and Damaged PHY Headers

With this strategy, a jammer interferes with the SFD or PSDU length field. After the SFD is detected, the receiver knows that a frame is arriving and starts to interpret a number of incoming symbols determined by the frame length. A proactive jammer can insert SFDs on the channel to trigger frame detection events at a receiver. The receiver then fails to detect any further transmission for a period of time as it is already occupied with decoding channel noise. In addition, a reactive jammer is able to selectively block the SFD symbols such that a receiver does not detect a frame, or to introduce an error in the frame length field that also results in a misinterpretation of the frame's fields.

Limited Dynamic Range

Common commercial receiver designs use mechanisms that make receivers more robust in regular situations, but have a jamming amplification effect, such as Automatic Gain Control (AGC). AGC is a control loop that adjusts the amplification of incoming baseband signals to fill the complete dynamic range of the analog-to-digital converter (ADC). This enables transceiver designers to use cheap ADCs with low resolution, such as 4–6 bit (Oh and S.-G. Lee, 2005). However, on the downside, an adversary can exploit the AGC mechanism in two ways: either through a pre-emptive locking of the receiver to low amplification, which makes other signals too weak to receive (causing failed timing recovery or frame synchronization), or by reactively sending a strong jamming signal to the receiver that uses a high gain setting (causing clipping in the ADC and therefore symbol misdetection). Interestingly, both of these strategies affect following symbols after the jamming has ceased, as the control loop does not react instantaneously.

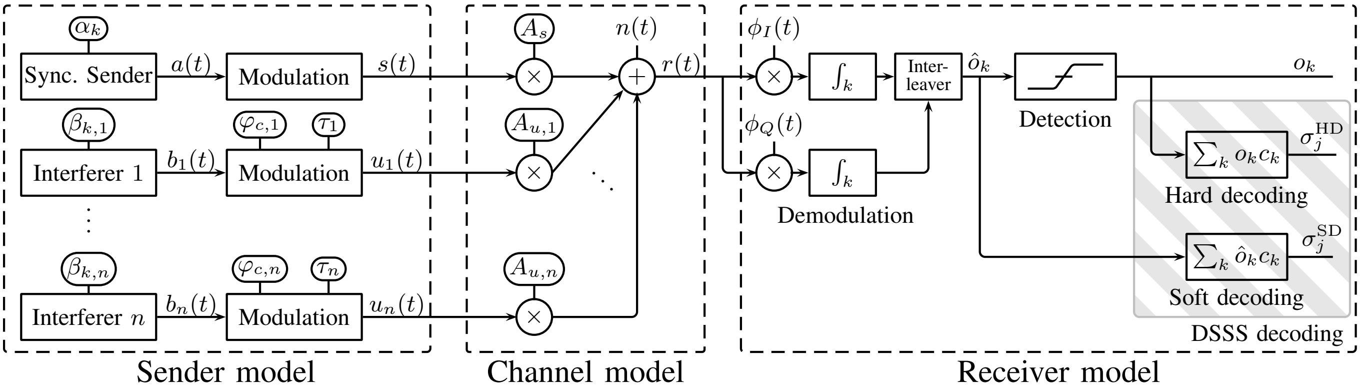


Figure 4.2.: System model, its parameters are shown in ovals (information bits α_k, β_k, carrier phase offset φ_c, time offset τ, signal amplitudes A_s, A_u). We consider one synchronized sender and n interferers on a collision channel that is the input to a receiver. Here, three channel coding schemes are considered, *(i)* uncoded, *(ii)* DSSS with hard decision decoding (HDD), and *(iii)* DSSS with soft decision decoding (SDD); resulting in different receiver paths.

4.4. System Model

In this section, we discuss the system model underlying our analysis, as shown in Figure 4.2. It considers all factors from the previous section. From a bird's eye view, the model consists of three components: *(i)* the sender model that modulates the physical layer signals of $n + 1$ transmitters, one fully synchronized signal of interest (SoI) and n interferers with possibly differing transmission starting times and payloads; *(ii)* the channel model with all senders sharing a single collision channel that outputs a scaled superposition of all signals (according to their corresponding power at the receiver), and *(iii)* the receiver model with three detection methods: uncoded, DSSS with hard decision decoding (HDD), and DSSS with soft decision decoding (SDD). In the following, we discuss each component in detail.

4.4.1. Sender Model

In the first component, we modulate the physical signals of $n+1$ senders. We instantiate our model with the Minimum Shift Keying (MSK) modulation, a widely used digital modulation with desirable properties, and of special interest because of its use in the 2.4 GHz PHY of IEEE 802.15.4 (2006, Sec. 6.5), but we also discuss other modulation schemes including O-QPSK,

QPSK, and BPSK. For the signal representations, we follow the notation of Proakis and Salehi (2007, Sec. 4.3).

Synchronized Sender

We assume that the receiver is fully synchronized to the SoI, i.e., the synchronization process has successfully acquired this signal and all interferers have relative offsets to it. The signal is then given by

$$s\left(t\right) = a_I\left(t\right)\cos\left(\frac{\pi t}{2T}\right)\cos\omega_c t + a_Q\left(t\right)\sin\left(\frac{\pi t}{2T}\right)\sin\omega_c t. \tag{4.1}$$

The signal consists of two components, the in- (I) and the quadrature-phase (Q) components. Modulated onto each component are the information signals (carrying the bits represented by $\alpha_k^I, \alpha_k^Q \in \{\pm 1\}$) given by

$$a_I\left(t\right) = \sum_{k=-\infty}^{\infty} \alpha_k^I \Pi\left(\frac{t-2kT}{2T}\right) \tag{4.2}$$

$$a_Q\left(t\right) = \sum_{k=-\infty}^{\infty} \alpha_k^Q \Pi\left(\frac{t-\left(2k+1\right)T}{2T}\right), \tag{4.3}$$

which represents a train of unit pulses Π with duration $2T$, the bit duration of the modulation (e.g., $2T = 1\,\mu\text{s}$ in IEEE 802.15.4). The unit pulses are defined by

$$\Pi\left(t\right) = \begin{cases} 0 & \text{if } |t| > \frac{1}{2} \\ \frac{1}{2} & \text{if } |t| = \frac{1}{2} \\ 1 & \text{if } |t| < \frac{1}{2} \end{cases} \tag{4.4}$$

The information signals are staggered, i.e., the Q-phase information signal is delayed by T in $a_Q\left(t\right)$. These signals are then shaped with half-sine pulses of duration $2T$, and modulated onto a carrier with frequency $\omega_c/2\pi$ (e.g., 2.4–2.48 GHz in IEEE 802.15.4). In the following, we use the angular frequency of baseband pulses $\omega_p = \pi/2T$, such that the first cosine term in (4.1) may be represented by $\cos\omega_p t$.

A graphical illustration of such an MSK-modulated signal is shown in Figure 4.3. The modulated bit sequence is $\mathtt{1\underline{1}1\underline{0}0\underline{1}0\underline{0}1\underline{1}}$; quadrature-phase bits are underlined, the bits are thus distributed to both phases alternately. This multiplexing to the IQ components is shown in the figure as blue, rectangular pulse trains, where each pulse has a duration of $2T$. In the first modulation step, these rectangular pulses are shaped with half-sines, resulting in the the

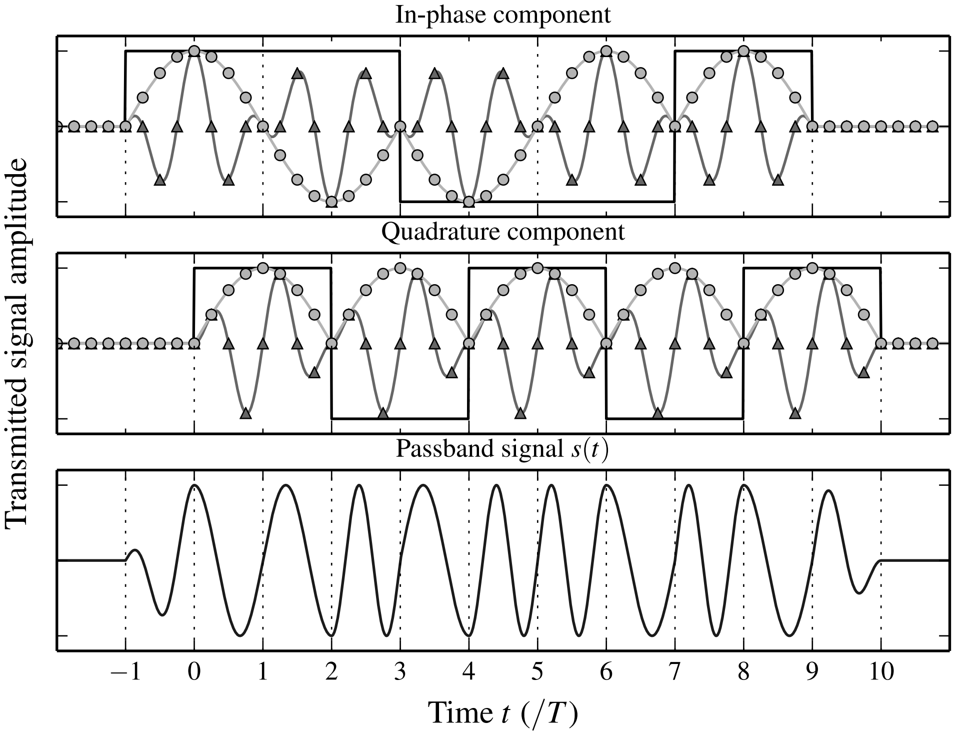


Figure 4.3.: Minimum shift keying (MSK) modulation example.

red sinusoids with ◦ markers in Figure 4.3. In the final step, the waveform is modulated on a carrier, resulting in the green waveform (with △ markers). Both modulated IQ signals are added to result in the (real-valued) passband time signal in the bottom figure. The figure also shows the properties that the the quadrature component is delayed; we observe an additional staggering of T.

(Unsynchronized) Interferers

In addition to the synchronized sender, we consider n interferers transmitting concurrently, using the same modulation. These signals may not be synchronized to the receiver and each may carry its own payload. This introduces three additional parameters that influence the signal, the time offset τ_i, the carrier phase offset $\varphi_{c,i}$, and the information bits $\beta_{k,i}$. With a positive τ_i, an interfering signal arrives later at the receiver than the synchronized signal.

The signal at the receiver for interferer i is given by

$$\begin{aligned} u_i\left(t;\tau_i,\varphi_{c,i}\right) = & \, b_{I,i}\left(t-\tau_i\right)\cos\omega_p\left(t-\tau_i\right)\cos\left(\omega_c t+\varphi_{c,i}\right) \\ & + b_{Q,i}\left(t-\tau_i\right)\sin\omega_p\left(t-\tau_i\right)\sin\left(\omega_c t+\varphi_{c,i}\right). \end{aligned} \tag{4.5}$$

We assume that the phase offsets $\varphi_{c,i}$ are constant for the duration of a packet, i.e., there is no carrier frequency offset during a transmission. In our experiments in Section 8.1, we show that this assumption is reasonable because receiver implementations are compensating for possible drifts. For convenience, we express the pulse phase offset caused by τ_i as $\varphi_{p,i} = \omega_p \tau_i$.

Other Modulation Schemes

While our results are derived for the MSK modulation, it is possible to adapt them to other variants of the phase shift keying (PSK) modulation. We briefly describe the differences to major variants and highlight how these affect the analysis. Further details on the relationship between PSK modulation schemes can be found in Proakis and Salehi (2007) and Pasupathy (1979).

Offset QPSK. O-QPSK with a half-sine pulse shape is identical to MSK (Schmid, 2006) and the results therefore also apply for this modulation. If O-QPSK is used in combination with rectangular pulse shaping instead, the signal is then given by

$$s_{\mathrm{O-QPSK}}\left(t\right) = \frac{1}{\sqrt{2}}\left(a_I\left(t\right)\cos\omega_c t + a_Q\left(t\right)\sin\omega_c t\right).$$

The altered pulse shape leads to the omission of the factor $\cos\omega_p t$ present in (4.1), because the rectangular shaping is already included in the information signal $a\left(t\right)$. This leads to a simplification of our MSK results because pulse phase offsets φ_p that are caused by the time offset τ are not present.

Quadrature PSK. Considering *QPSK*, the change from O-QPSK is the missing time shift T in the quadrature phase. This leads to a different information signal for the Q phase,

$$a'_Q\left(t\right) = \sum_{k=-\infty}^{\infty} \alpha_k^Q \Pi\left(\frac{t-2kT}{2T}\right).$$

When adapting our results to QPSK, this affects the indices k of the colliding bits.

Binary PSK. This scheme considers only the in-phase components of QPSK, its signal is given by

$$s_{\mathrm{BPSK}}(t) = \frac{1}{\sqrt{2}} a_I(t) \cos \omega_c t.$$

This simplifies the derivations and results further, because there is no contribution from the Q phase signal in collisions.

4.4.2. Channel Model

In our model, we use an additive collision channel. The relation for the output signal is

$$r(t) = A_s\, s(t) + \sum_{i=1}^{n} A_{u,i}\, u_i(t; \tau_i, \varphi_{c,i}) + n(t). \tag{4.6}$$

Each signal is scaled by a positive, real-valued factor A, which contains both, possible signal amplifications by the sender and path loss effects that reduce the power at the receiver. In our evaluation, we use the Signal to Interference Ratio (SIR) at the receiver, given by $\mathrm{SIR} = A_s^2 / \left(\sum_{i=1}^{n} A_{u,i}^2\right)$, to characterize the power relationship of the interfering signals. The contribution of all noise effects is accumulated in the linear noise term $n(t)$; possible instantiations are a noiseless channel or a white Gaussian noise channel.

4.4.3. Receiver Model

In the final component of the model, we feed the signals' superposition $r(t)$ into an optimal receiver to discern the detected bits. The signal is demodulated and fed into one of three detector implementations: one for uncoded bits, and two variants of DSSS decoding.

Demodulation

Demodulation is performed for I and Q individually and the bits are then interleaved. We limit our discussion to the I component for brevity.

We use the matched filter function $\phi_I(t) = (2/T) \cos \omega_p t \cos \omega_c t$ and low-pass filtering for downconversion and demodulation, which is the optimal receiver for noiseless and Gaussian channels in the sense that it minimizes the bit error probability Proakis and Salehi (2007, Sec. 4.3). The received signal $r(t)$ is multiplied by $\phi_I(t)$ and integrated for each bit period k to form the decision variable

$$\hat{o}_k^I = \Lambda_r^I(k) = \int_{(2k-1)T}^{(2k+1)T} r(t)\,\phi_I(t)\,dt. \tag{4.7}$$

The resulting (real) value is called *soft bit.* Because the combination of the interferers in the received signal is linear, the individual contributions can be divided into integrals for each signal:

$$\hat{o}_k^I = \Lambda_s^I(k) + \sum_{i=1}^{n} \Lambda_{u_i}^I(k) + \Lambda_n^I(k)\,.$$

In our analytical evaluation in the following section, we derive closed-form expressions for $\Lambda_{u_i}^I$ and $\Lambda_{u_i}^Q$ to analyze the receiver output after a signal collision.

We point out that this simplified model does not include receiver-side techniques such as Automatic Gain Control (AGC) or phase tracking; however, we conjecture that the reception performance is still comparable. In fact, as our experiments in Section 8.1 show, this assumption is justified and the simplified model is able to predict the reception behavior of real-world receiver implementations with good accuracy. We leave the investigation on the effects of these advanced techniques to future work.

Uncoded Bit Detection

The detection operation for uncoded transmissions is slicing, essentially a sign operation on the demodulation output, which results in binary output $o_k \in \{\pm 1\}$. Thus, a bit of the SoI is flipped if the contribution of the interferers changes the bit's sign.

DSSS Decoding

For coded transmissions, the number of chips exceeds the bits in a symbol, i.e., even if several chips are flipped it is still possible to decode a symbol correctly. We consider 2^b symbols ξ with chipping sequence c_ξ, each with a block length of B bit (i.e., the number of chips). For example, we have $b = 4$, $B = 32$ in IEEE 802.15.4 (see also Table 4.1).

We differentiate two modes of operation for the DSSS decoder, namely hard decision decoding (HDD) and soft decision decoding (SDD) (Proakis and Salehi, 2007).

Symbol	Definition
$k' = k - \lfloor \tau / 2T \rfloor$	Correction factor for the bits active in a decision interval
$k^{Q'} = k - \lfloor (\tau + T) / 2T \rfloor$	Correction factor for Q bits during I detection
$k^{I'} = k - \lfloor (\tau - T) / 2T \rfloor$	Correction factor for I bits during Q detection
$\underline{\tau} = \tau - 2k'T$	Relative shift in a bit of interest k
$\underline{\tau}^{Q} = \tau + T - 2k^{Q'}T$	Relative shift in a bit of interest k for the leaking Q-phase
$\underline{\tau}^{I} = \tau - T - 2k^{I'}T$	Relative shift in a bit of interest k for the leaking I-phase

Table 4.2.: Correction factors used in the derivations.

Hard Decision Decoding

In *HDD*, the decoder uses sliced (binary) values o_k as its input, and then chooses the symbol with the highest bit-wise cross-correlation of all chipping sequences. In this way, HDD can be viewed as an additional step that takes a group of uncoded bits with B elements (from the uncoded bit detection described above) to determine a symbol σ_j^{HD}, i.e., a group of b bits. For HDD, the decoder is given by

$$\sigma_j^{\mathrm{HD}} = \arg \max_{0 \leq \xi < 2^b} \left| \sum_{k=0}^{B-1} o_{jB+k} \, c_{\xi,k} \right| . \tag{4.8}$$

Soft Decision Decoding

In *SDD*, the real-valued, unquantized demodulator output $\hat{o}_k$ (*soft bits*) is used as decoder input directly, in contrast to the binary values o_k used in HDD. This is beneficial because soft bits provide a measure of detection confidence and demodulation quality, and thus adds weighting to the bits used in the cross-correlation. The determined symbol is denoted as σ_j^{SDD}.

4.5. Mathematical Analysis

Based on the system model in Figure 4.2, we analyze the contributions of each interfering signal to the overall demodulator output; the sum of these contributions is the decision variable of bit detection. We first present the general case considering all system parameters in theorem 4.1. Subsequently, we illustrate its interpretation using selected parameter combinations. The new notation that is used to represent index or time offset conversions are collected in Table 4.2.

Theorem 4.1. *For an interfering MSK signal $u\left(t\right)$ with offset parameters τ and φ_c, the contribution to the demodulation output $\Lambda_u^I\left(k\right)$ is given by*[1]

$$\begin{aligned}\Lambda_u^I\left(k\right) = \frac{1}{2T}A_u \Bigg\{ & \cos\varphi_c \left(\cos\varphi_p \left(\underline{\tau}\beta_{k-1}^I + \left(2T - \underline{\tau}\right)\beta_k^I\right) - \frac{2T}{\pi}\sin\varphi_p\left(\beta_{k-1}^I - \beta_k^I\right)\right) \\ & - \sin\varphi_c \left(\sin\varphi_p \left(\underline{\tau^Q}\beta_{kQ'-1}^Q + \left(2T - \underline{\tau^Q}\right)\beta_{kQ'}^Q\right) + \frac{2T}{\pi}\cos\varphi_p\left(\beta_{kQ'-1}^Q - \beta_{kQ'}^Q\right)\right)\Bigg\}.\end{aligned}$$

The proof of this theorem can be found in Appendix B. To provide a better understanding of the effects of the parameters, we focus on selected parameter constellations and discuss the resulting equations. Then we revisit theorem 4.1 and discuss the combination of effects.

4.5.1. Synchronized Signal

In the simplest case both offsets, time and phase, are zero, i.e., the interfering signal is also fully synchronized to the receiver. The result is given by

$$\Lambda_u^I\left(k\right) = A_u\beta_k^I.$$

The signal's contribution to the k-th bit is $\Lambda_u^I\left(k\right) = A_u\beta_k^I$. The bit decision of bit k, i.e., the sign of the equation, is governed by β_k^I. The magnitude of the contribution is controlled by the amplitude of the signal A_u, and thus stronger signals lead to a greater contribution to the decision variable $\hat{o}_k$. As an example, consider two signals $s\left(t\right)$ and $u\left(t\right)$ that are both fully synchronized to the receiver. The detector output of bit k is then $A_s\alpha_k^I + A_u\beta_k^I$. If both senders transmit the same bit ($\alpha_k^I = \beta_k^I$), then the signals interfere constructively and push the decision variable further away from zero. If, on the other hand, the bits are different, then the decision variable has the sign of the stronger signal; this is the well-known power capture effect for a single bit.

4.5.2. Carrier Phase Offset

Next, we analyze the effect of carrier phase offsets when the signals are fully time-synchronized ($\tau = 0$), as shown in Figure 4.4. The result is given by

$$\Lambda_u^I\left(k\right) = A_u\left(\cos\varphi_c\beta_k^I - \frac{1}{\pi}\sin\varphi_c\left(\beta_{k-1}^Q - \beta_k^Q\right)\right).$$

---

[1] We omit the subscript i for clarity in the equations. The results for the quadrature phase are given by the same equations when the roles of I and Q are exchanged.

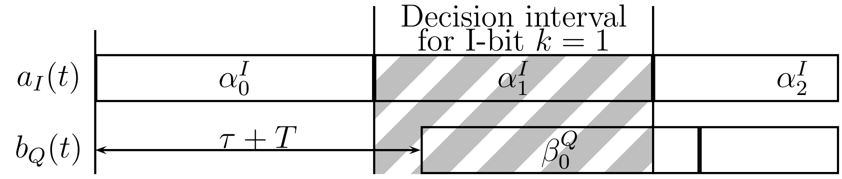


Figure 4.4.: Effect of carrier phase offset φ_c in a collision: several bits influence the bit decision on bit k in a collision between two signals. The carrier phase offsets lead to a leakage of the quadrature phase, and because the Q-bits are staggered, there is an additional shift of T in the bit indices. The active bits in the decision interval are highlighted.

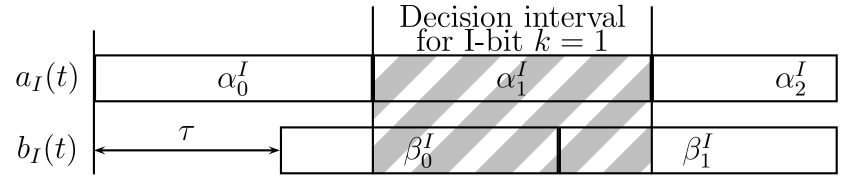


Figure 4.5.: Example of a time offset τ during a detection: three bits influence the bit decision on the second bit in a collision between two signals. The active bits in the decision interval are highlighted (the synchronized sender's bit α_1^I and interferer's in-phase bits β_0^I and β_1^I).

We observe two effects of the carrier phase offset. First, the bit contribution of β_k^I is scaled by $\cos\varphi_c \leq 1$, which leads to reduced absolute values (and thus a smaller contribution to the decision variable) and potentially causes the bit β_k^I to flip for $\varphi_c \in ((\pi/2,)\,(3/2)\,\pi)$. Second, the quadrature phase starts to leak into the decision variable and thus two additional bits β_{k-1}^Q, β_k^Q influence the outcome. This contribution, however, is scaled by $\pi^{-1}\sin\varphi_c$, and only appears when the two Q bits are alternating during the integration interval. In essence, uncontrolled carrier phase offsets may lead to unpredictable bits in the detector output because of carrier phase offset induced bit flips.

4.5.3. Time Offset

If the signals are phase-matched but shifted in time, the detector output is given by

$$\Lambda_u^I(k) = \frac{1}{2T} A_u \left(\cos\varphi_p \left(\underline{\tau}\beta_{k-1}^I + (2T - \underline{\tau})\,\beta_k^I \right) - \frac{2T}{\pi} \sin\varphi_p \left(\beta_{k-1}^I - \beta_k^I \right) \right).$$

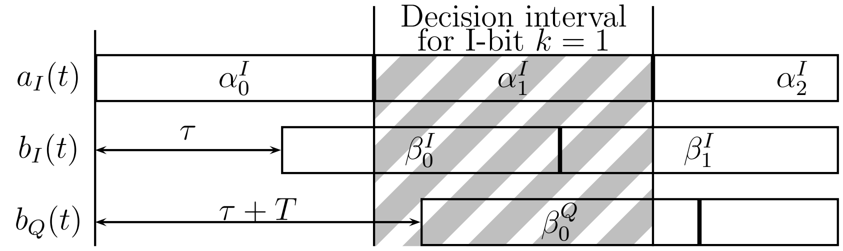


Figure 4.6.: Example of time and phase offsets combined: the decision on the second bit ($k = 1$) is influenced by four bits in this example (the synchronized sender's bit α_1^I, the interferer's in-phase bits β_0^I and β_1^I due to time shifts, and quadrature bit β_0^Q from carrier phase offsets).

We make three observations here. The bit index k needs to be adjusted because bits may be time-shifted into the integration interval, see Figure 4.5; the new index is given by $k' = k - \lfloor \tau/2T \rfloor$, with $\lfloor \cdot \rfloor$ denoting the floor function. We call these *active bits* because they contribute to the bit decision. These bits overlap partially or fully, and their active time duration is $\underline{\tau} = \tau - 2 \lfloor \tau/2T \rfloor T$, the underscore signifies that its value is confined to the interval $[0; 2T)$. However, these bits do not contribute to the decision directly but are scaled by $\cos \varphi_p$, which is caused by the half-sine pulse shaping of MSK. This scaling means that bit contributions are diminished and may be flipped by certain time offsets. Finally, a term scaled by π^{-1} is introduced that is only present when bits are alternating. However, these bits are the same in-phase bits β_{k-1}^I, β_k^I, the Q phase does not leak in this setting.

4.5.4. Both Offsets

Finally, when both offsets are present as in theorem 4.1, we can interpret the result as a combination of the above effects. A graphical illustration of the active bits is shown in Figure 4.6. Due to the staggering of bits (the Q bits are delayed by T), the indices of leaking bits of the Q phase also need to be adjusted, the new index is $k^{Q\prime} = k - \lfloor (\tau + T)/2T \rfloor$, and the active time interval $\underline{\tau^Q}$ is derived similarly to above.

4.5.5. Analysis Summary

In summary, we observe that the contribution of the interfering signal is complex and that φ_c and φ_p can potentially flip the original bits β_k^I. This should be bad news for collision-aware protocols that use identical payload to achieve constructive interference (e.g., SCIF (Wang, Y. He, et al., 2012)): these bits can flip easily and then generate destructive interference. However, coding helps to alleviate these negative effects as we will see in the next section.

4.6. Mathematical Evaluation

Equipped with the closed-form analytical model of the bit-wise receiver outputs, we systematically explore the parameter space of the reception of concurrent transmissions in detail.

4.6.1. Methodology

In order to numerically study the transmission reception success under interference we perform so-called Monte Carlo simulations (see Jain (1991)); that means we do time-static simulations of independent packet transmissions in which we randomly vary the analytical model's parameters to investigate their influence on performance parameters such as packet reception ratio, bit and symbol error rate. Conceptually, the simulator is just a software version of the mathematical model (written in Python) applied to a whole packet; it is not meant to validate the model but to experiment with randomly chosen values for the model parameters and to provide more insights on the success probability of concurrent transmissions. The simulation code is available for download at `http://disco.cs.uni-kl.de/content/collisions`; there, the interested reader can also find an interactive visualization of the model.

For most experiments, the time offset between sender and interferer is fixed and is our primary factor in the numerical analysis, i.e., in the plots we show the reception performance depending on the time offset. The other parameters of the model are treated as secondary factors and are randomly varied. Generating 1,000 independent packet transmissions for each data point in the presented graphs thus represents the secondary factors' average contribution to the reception success. We provide more details on the choices for the model's parameters for sender and channel in the following.

Instantiation of the Sender Model

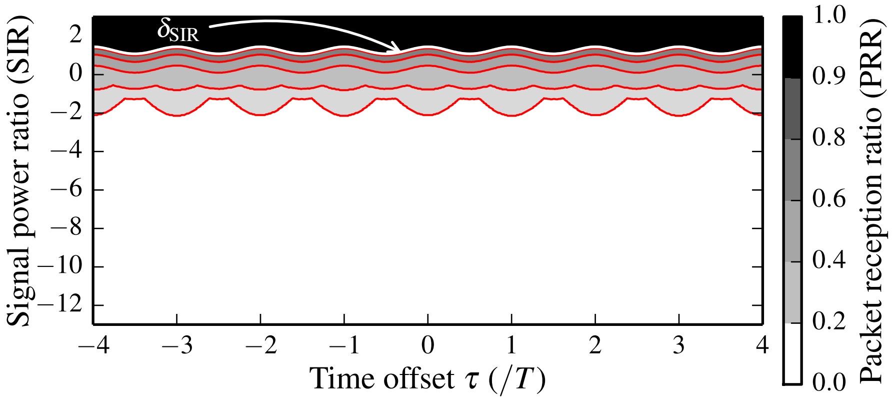
For ease of presentation, we mainly consider the presence of one synchronized sender and one interferer; we denote these parties as $\mathcal{S}$ and $\mathcal{I}$ with signals

$s(t)$ and $u(t)$, respectively. In Section 4.6.2, we consider the n interferer case separately. We analyze the reception performance of groups of associated bits, or packets; in this case, a single bit error leads to a packet drop. The packet reception ratio (PRR) is the fraction of packets that arrive without errors divided by the total number of packets. We use packets with a length of 64 bit. We consider two categories of colliding packets, either with independent ($\mathcal{S}$ and $\mathcal{I}$ trying to exploit spatial reuse) or identical content ($\alpha_k = \beta_k$, as it is the case for collision-aware flooding protocols). The bits to send are chosen in the following manner: for uncoded transmissions, α_k is drawn bitwise i.i.d. from a Bernoulli distribution over $\{-1, 1\}$, and either the same procedure is performed for β_k (independent packets) or simply copied over from α_k (identical packets). For coded packets, we draw symbols i.i.d. uniform random from $\{0, \ldots, 15\}$ and spread these symbols according to the chipping sequences defined by the IEEE 802.15.4 standard (2006, Sec. 6.5). This means that 4 bit groups are first spread to 32 bit chipping sequences before they are transmitted in α_k, β_k. The chipping sequences are given in Table 4.1. Note that for symbols `1`–`7`, the chipping sequences are shifted versions of the symbol `0`, while for the other half (symbols `8`–`15`), the quadrature-phase bits are inverted.

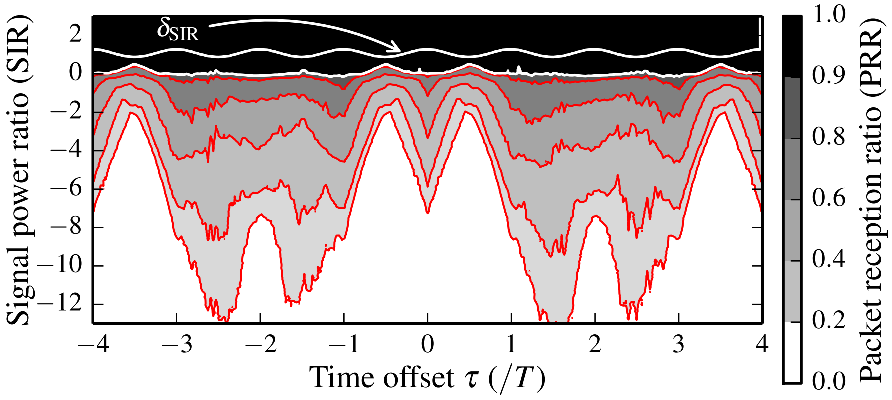
In accordance to the literature (RAPPAPORT, 2001), as the carrier phase offset is hard to control because of oscillators drifts and other phase changes during transmission, we draw φ_c i.i.d. uniform randomly from $[0; 2\pi)$ for each packet unless stated otherwise. On the other hand, we use the same time offset τ for all packets because experimental work shows that this timing can be precisely controlled. For example, Glossy (FERRARI et al., 2011) achieves a timing precision of 500 ns over 8 hops with 96 % probability, and WANG, Y. LIU, et al. (2014) report a 95 % percentile time synchronization error of at most 250 ns. For our simulations, we used 1,000 packets for each value of τ.

Instantiation of the Channel Model

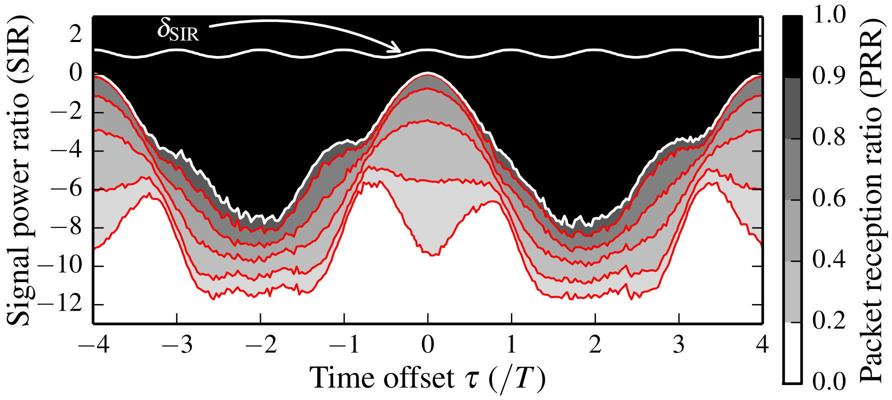
To concentrate on the impact of signal interference, we consider a noiseless channel. This is a well-accepted assumption when both signals are significantly above noise floor level (POISEL, 2011, Sec. 8). We set $A_s = 1$ and $A_u = \mathrm{SIR}^{-\frac{1}{2}}$.



(a) Uncoded transmissions.

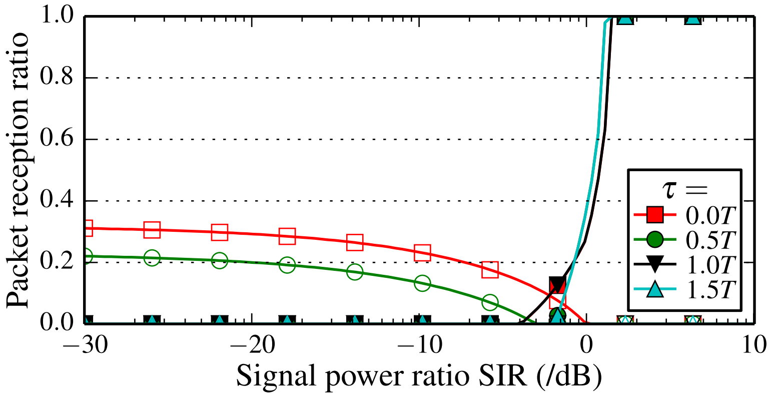


(b) DSSS with hard decision decoding.

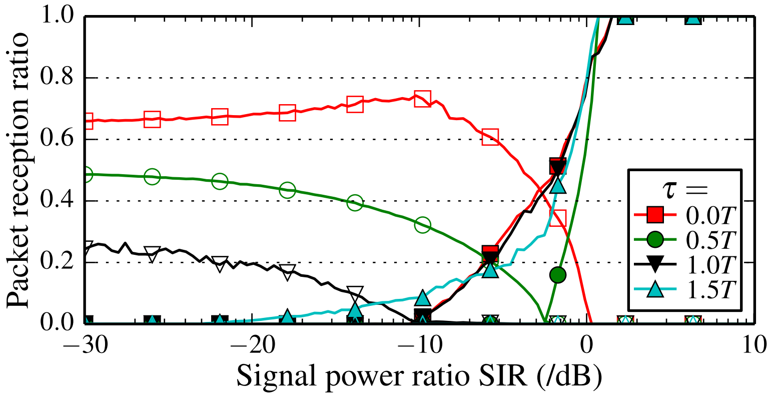


(c) DSSS with soft decision decoding.

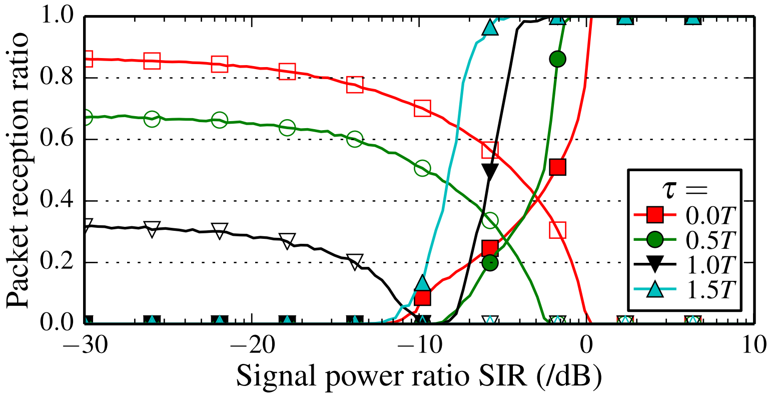
Figure 4.7.: The capture threshold for two colliding packets with *independent* payload, varying with the signals' power ratio SIR and time offset ($\tau = 0$ indicates that the signals overlap fully).



(a) Uncoded transmissions.



(b) DSSS with hard decision decoding.



(c) DSSS with soft decision decoding.

Figure 4.8.: Effect of signal to interference ratio SIR on the PRR for *independent* payload. Filled markers represent the reception of the synchronized sender's packets, empty markers represent the reception of the interferer's packets.

4.6.2. Reception of the Synchronized Signal of Interest

Capture Threshold Under Independent Payload

In our first case study, we consider the transmission of independent payload. This situation occurs, e.g., when two uncoordinated senders detect a clear channel, transmit, and the packets collide at the receiver. Our metric of interest is the PRR of the SoI, i.e., we observe the probability to overcome the collision. The results for three classes of receivers are shown in Figure 4.7 and Figure 4.8.

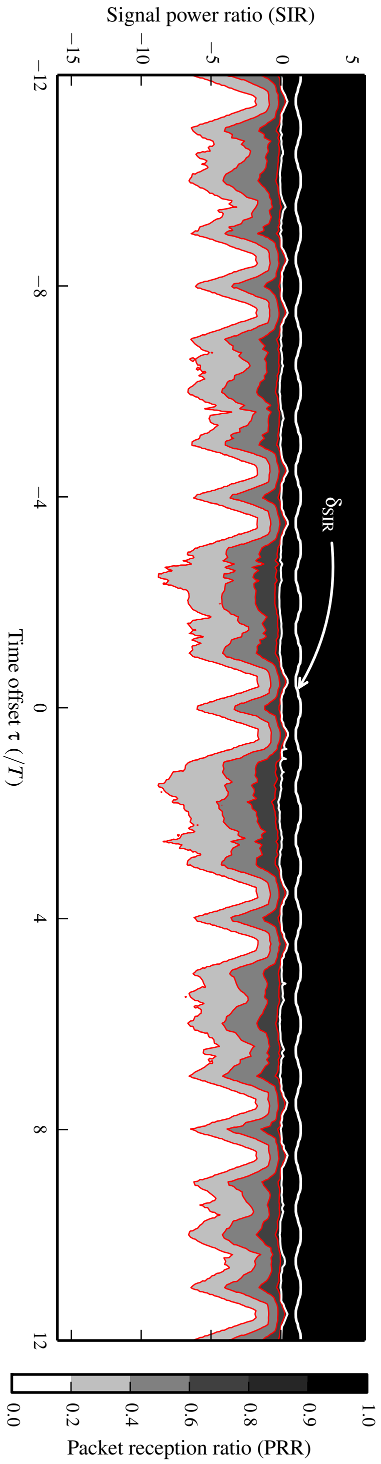
Uncoded transmissions. From Figure 4.7a, we observe that the capture threshold is a good model to describe the PRR of interfering, uncoded transmissions. If the SoI is stronger by a threshold δ_{SIR} of 2 dB, all its packets are received.[2] This behavior persists for all choices of τ, i.e., packet reception is independent from the properties of the interfering signal (we only see a minor periodic effect). Below the threshold, there is a narrow transitional region with non-zero PRR. Under uncoded transmissions, our model is able to recover the classical capture threshold for MSK and is in accordance to experimental results in the literature (Gezer et al., 2010; Son et al., 2006). In Figure 4.8a, we observe for uncoded transmissions and in the negative SIR regime that the reception of the interferer's packets is poor (max. 30 % PRR) even with perfect time synchronization ($\tau = 0$). The synchronized sender requires a positive SIR for a high PRR independent of the interferers timing, and the transitional region is narrow.

Hard decision decoding. When considering HDD (Figure 4.7b), we note that the threshold abstraction is still valid and the performance improvement of coding is only 1 dB (the coding gain is canceled when the same chipping sequences are used). In the transitional region, there is a wider parameter range that results in non-zero PRRs, e.g., when τ is close to integer values (and thus $\cos \varphi_p \approx 0$), we observe a better PRR for $\mathcal{S}$. These results show that coding with HDD yields only limited benefits if all senders use identical chipping sequences. In Figure 4.8b, we observe that the interferer's packets have an increasing chance of reception, strongly depending on the timing. For the synchronized sender, the transitional region is widened significantly.

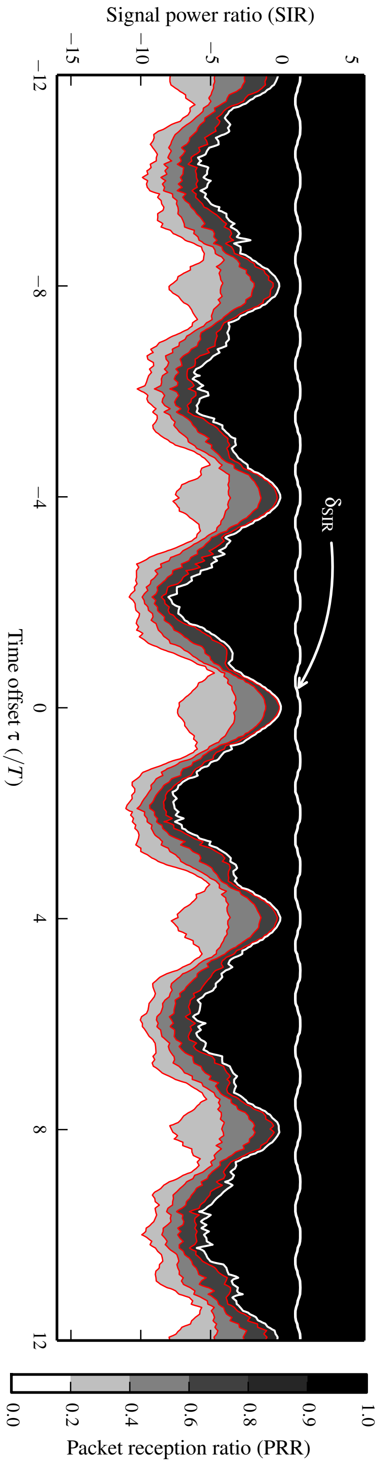
Soft decision decoding. Finally, for SDD we observe a strong dependence between PRR and time offset (Figure 4.7c). Only for positions without chip-

---

[2] For the numerical values of δ_{SIR} shown in the figures, we used a PRR threshold of 90 %.



(a) DSSS with hard decision decoding.



(b) DSSS with soft decision decoding.

Figure 4.9.: Capture threshold for colliding packets with independent payload with extended time offsets.

ping sequence shifts ($\tau = 0$, and because of the way IEEE 802.15.4 sequences are chosen[3], $\tau = 4kT$, $k \in \mathbb{Z}$) the performance is comparable to the HDD case. For different time shifts, we can achieve a 6–8 dB coding gain despite the use of identical chipping sequences; especially for offsets $\tau = 4kT + 2T$, we can achieve a clear coding gain. The reason is that soft bits contain additional information on the detection confidence, which helps to improve the detection performance in the cross-correlation. Again, Figure 4.8c shows this behavior for selected values of τ from a different point of view.

This insight suggests that two senders may benefit from coding even when using independent payloads, provided that they time their collisions precisely. This may help to increase the number of opportunities for concurrent transmissions, i.e., interfering nodes can be much closer to a receiver and still achieve the same PRR performance. In other words, a *constant* capture threshold is too conservative when collision timing can be precisely controlled, because the performance of SDD is very sensitive to time offsets.

Capture Threshold under Identical Payload

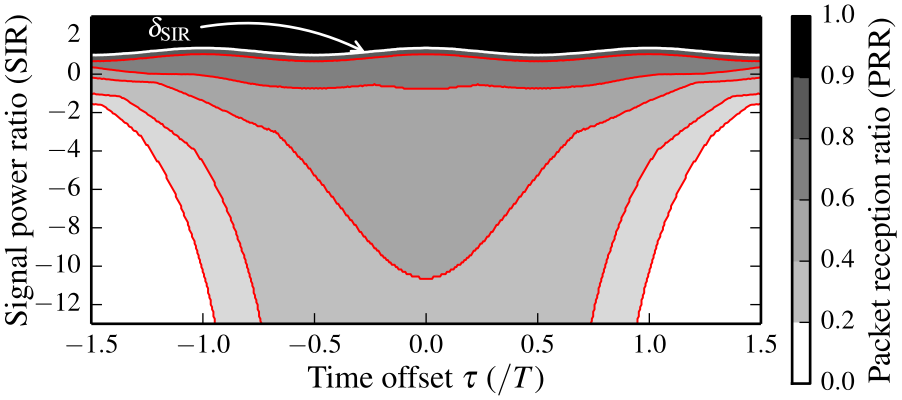
When considering the collisions of identical packets, we observe very different results (Figure 4.10 and Figure 4.11): a good reception performance is possible despite even a negative SIR.[4]

Uncoded transmissions. For uncoded transmissions, the PRR performance is shown in Figure 4.10a. While in this case the threshold for a PRR of 100 % is still equal to the independent payload case, substantially more packets are received in the transitional region with time shifts less than $\pm 0.75T$. However, PRRs around 30 % are usually not sufficient to boost the performance of network protocols. The reason for this limited performance is the carrier phase offset φ_c: with negative SIR, the interfering signal dominates the bit decision at the receiver, and with larger offsets $\varphi_c \in ((\pi/2)\,; (3/2)\,\pi$, the term $\cos\varphi_c$ changes its sign and flips all subsequent bits. In this sense, the literature conjecture that constructive interference is the reason for the good performance of flooding protocols (Wang, Y. He, et al., 2012; Wang, Y. Liu, et al., 2014) is only valid if the receiver is synchronized to the strongest signal and if the phase offset φ_c can be neglected. However, because the collisions start during the preamble when using such protocols, successful

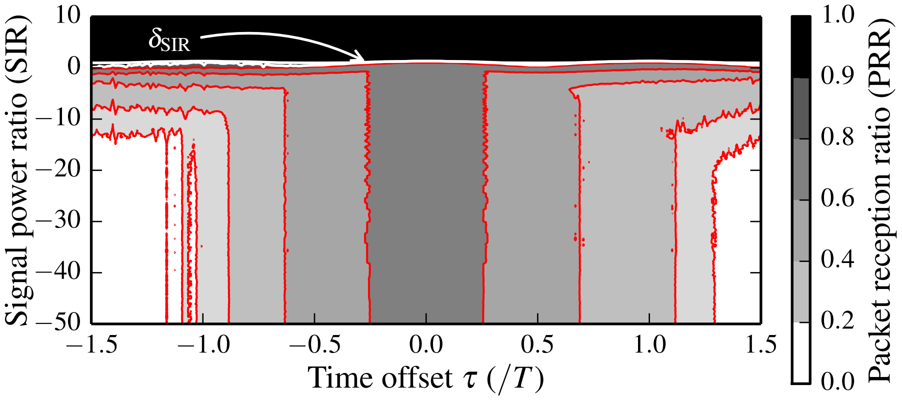
---

[3] See Table 4.1. The chipping sequences are not independently chosen, they constitute shifted versions of a single generator sequence with shifts of 4 IQ bits.

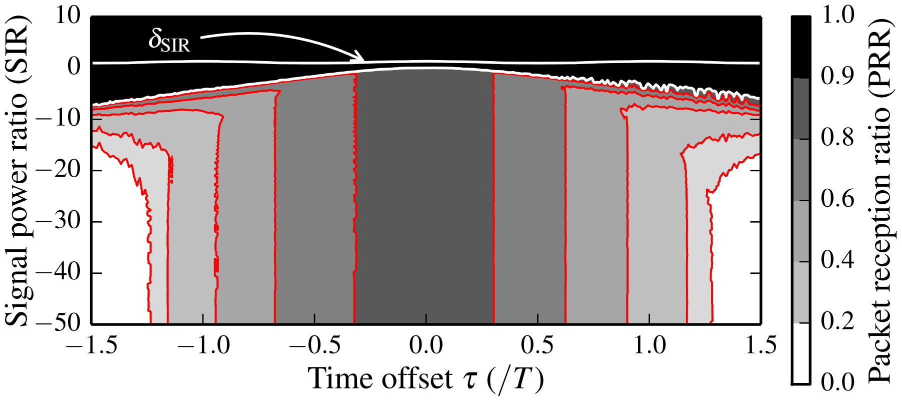
[4] We note that with increasing time offsets τ the PRR performance approaches the results for independent payloads, see Figure 4.12.



(a) Uncoded transmissions.

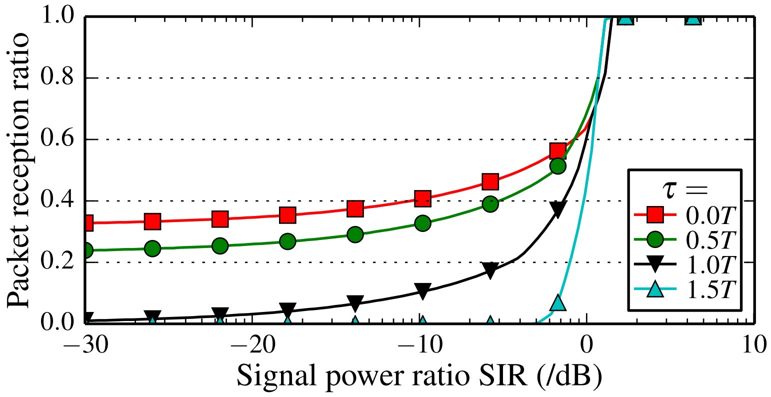


(b) DSSS with hard decision decoding.

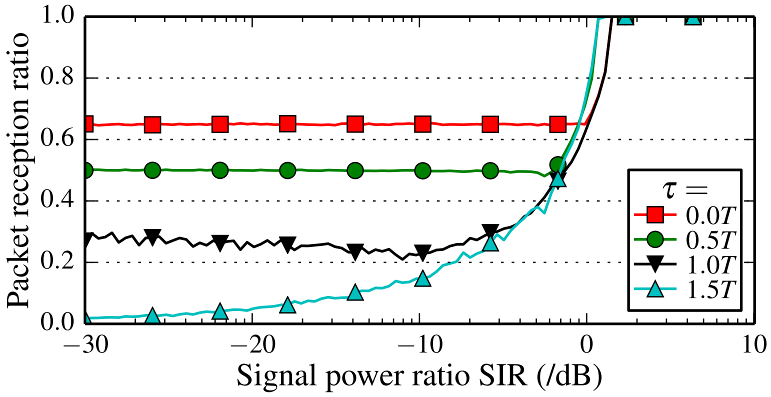


(c) DSSS with soft decision decoding.

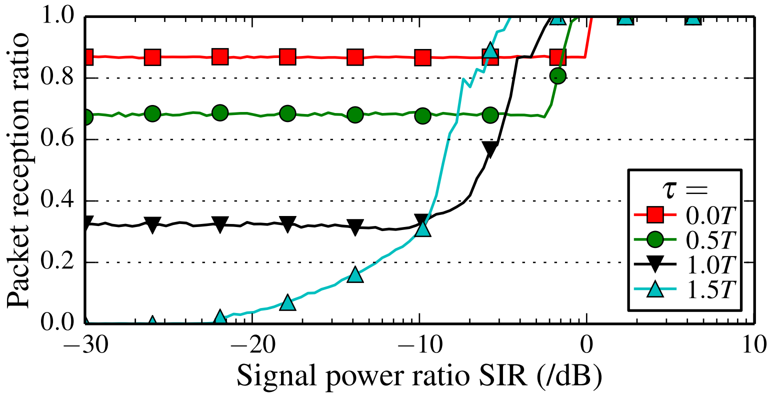
Figure 4.10.: The capture threshold for colliding packets with *identical* content depending on the power ratio SIR and the time offset τ. In all three figures, we show the threshold δ_{SIR} for identical and uncoded payload as reference.



(a) Uncoded transmissions.



(b) DSSS with hard decision decoding.



(c) DSSS with soft decision decoding.

Figure 4.11.: Effect of signal to interference ratio SIR on the PRR for *identical* payload. Because both use the same payload, only filled markers are present in contrast to Figure 4.8.

synchronization cannot be ensured. Therefore, there must be another mechanism that recovers flipped bits. In Figure 4.11a^, we also observe that the PRR varies from approx. 30 % to 100 % and is highly dependent on the SIR.

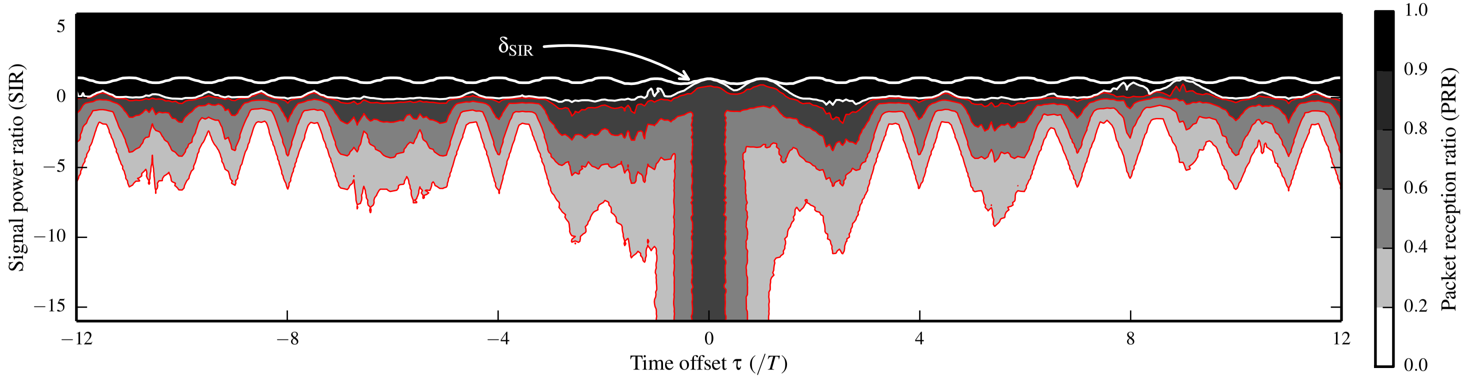
Hard decision decoding. The reception performance of coded messages provides a hint in this direction (Figure 4.10b). We observe a corridor of τ values ($\tau = \pm 0.2T$ or 100 ns in IEEE 802.15.4) that has a PRR of 60–80 % in the center (note the larger SIR scale on the y-axis). Figure 4.11b shows that the PRR takes two values that are more stable across the SIR range: in the negative SIR regime the PRR is around 65 %, and 100 % in the positive SIR regime. The reason is that when two signals with identical payload collide with a small time offset, a reception is still possible even if the interfering signal is far stronger. This suggests that the interfering signal is received instead of the SoI, and that coding helps to overcome bit flips of β_k induced by the carrier phase. The explanation is a property of (4.8): even if all bits are flipped by $\cos \varphi_c$, the (absolute) correlation is still maximal for the correct chipping sequence. This shows that DSSS used in IEEE 802.15.4 is a key factor to make the collision-aware protocols work.

Soft decision decoding. The experimentally observed performance in the literature is even superior to Figure 4.10b (Dutta, Dawson-Haggerty, et al., 2010; Ferrari et al., 2011; Wang, Y. He, et al., 2012). Taking SDD into account, this gap is closed (Figure 4.10c). There is a strong center region for $\tau \leq \pm 0.3T$, or 150 ns in 802.15.4, with a PRR of approximately 90 %. This behavior is also followed in Figure 4.11c. Now, this matches well with existing experimental results. This means that the reception performance is very good in this center region *independent* of the SIR, i.e., no power control is required and perfect time synchronization is unnecessary for successful reception.

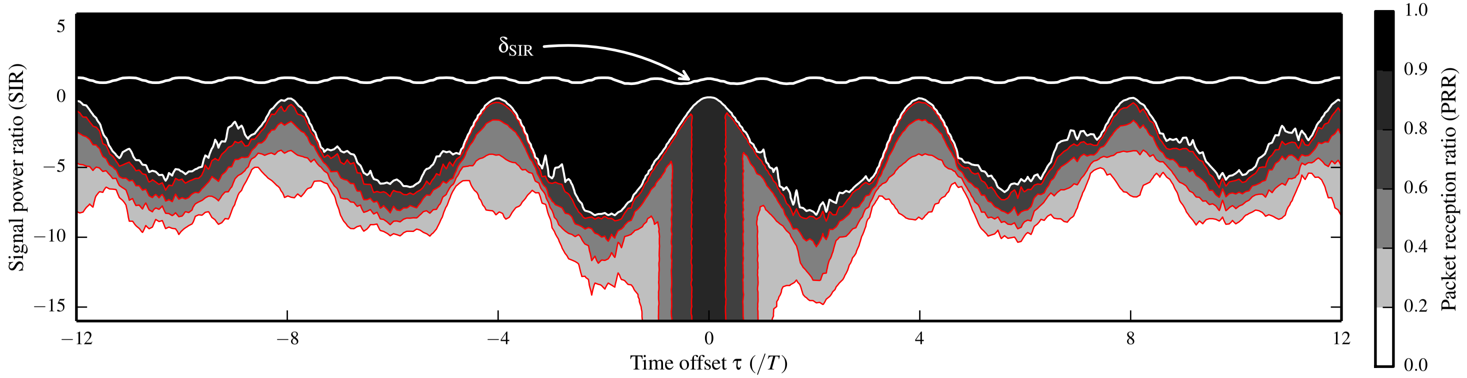
Effect of Several Interferers

In this subsection, we consider the effect of one strong interferer compared to several interferers with the same power when combined, but evenly distributed across the interferers. We consider the following scenario: all interferers are time-synchronized ($\tau_i = 0$), but each has an i.i.d. uniform random phase offset $\varphi_{c,i}$ (and independent payload bits $\beta_{k,i}$ if different content is assumed). The interference power varies with $\frac{n}{2} P_{\text{SoI}}$ for a number of interferers $n \in \{1, \dots, 8\}$, with each interferer having a signal power at the receiver of $\frac{1}{2} P_{\text{SoI}}$.

Under the classical capture threshold model both interference types share the same SIR and thus lead to the same PRR at the receiver. However, as we



(a) DSSS with hard decision decoding.



(b) DSSS with soft decision decoding.

Figure 4.12.: Capture threshold for colliding packets with identical payload with extended time offsets.

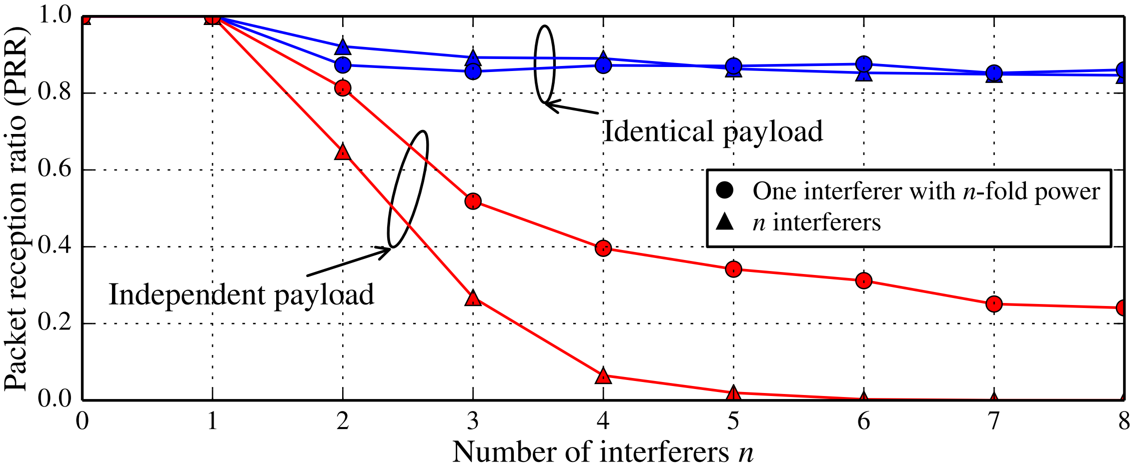
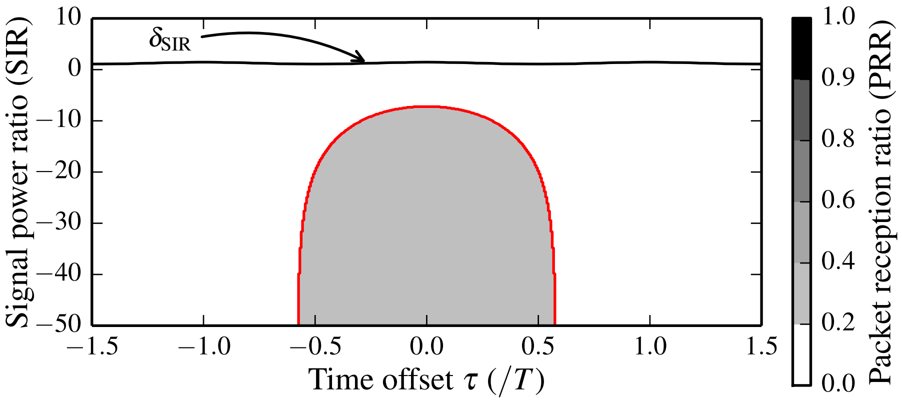


Figure 4.13.: Reception ratio for SDD under one strong interferer or n weaker interferers, but all with equal received power. For identical payload the difference is small, for independent payload several interferers are more destructive than one.

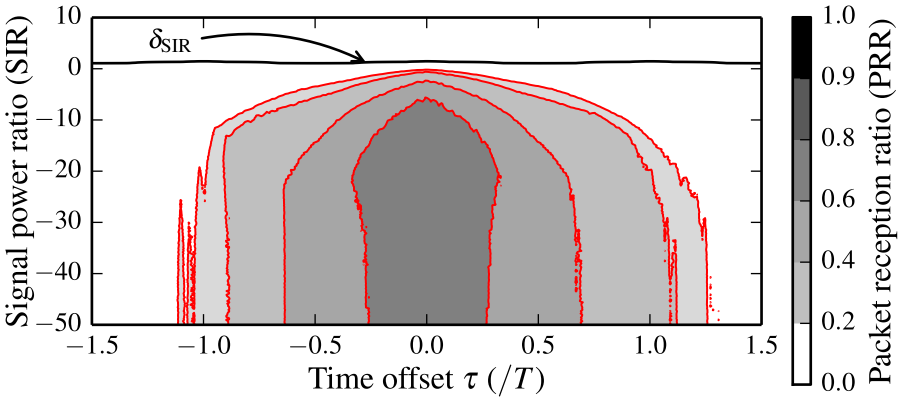
observe in Figure 4.13, this is only the case for identical payload, for independent payload n interferers prove to be more destructive despite having the same signal power. While experimental results by Ferrari et al. suggested this result (2011, Fig. 12) for identical payload, the root cause is now explained by our model: a single interferer is more likely affected by high attenuation ($\cos \varphi_c \approx 0$) than n independent interferers, resulting in a higher likelihood of destructive interference. However, in case of identical payload, even an effective interferer is still received correctly in 90 % of the cases. The observation for independent payload reveals another problem of SINR models: relying on the signal power ratio alone discards the crucial effects of each interferer's offsets.

4.6.3. Reception of Interfering Signals with Independent Payload

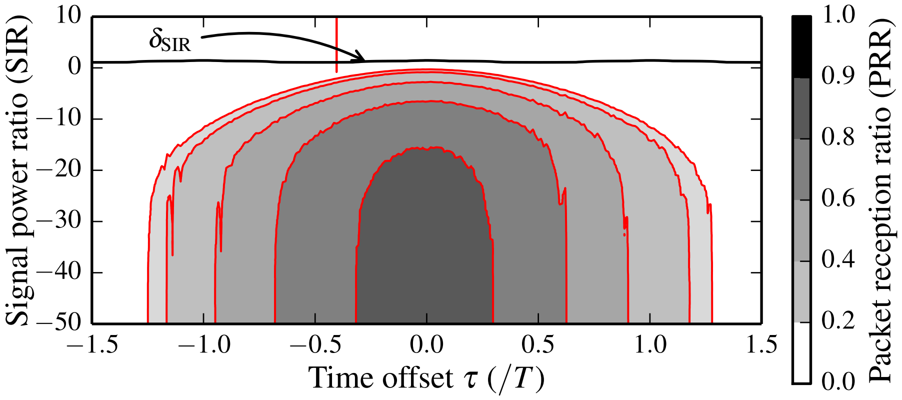
Our results explain why and when collision-aware protocols work even without power control: coding enables the reception of interfering signals despite carrier phase and time offsets. In this section, we revisit the case of independent payload but focus our interest now on the reception of the *interfering* signal, i.e., we treat the interfering signal $u(t)$ as the SoI and observe the reception of β_k instead of α_k. Related work by Pöpper et al. (2011) shows



(a) Uncoded transmissions.



(b) DSSS with hard decision decoding (HDD).



(c) DSSS with soft decision decoding (SDD).

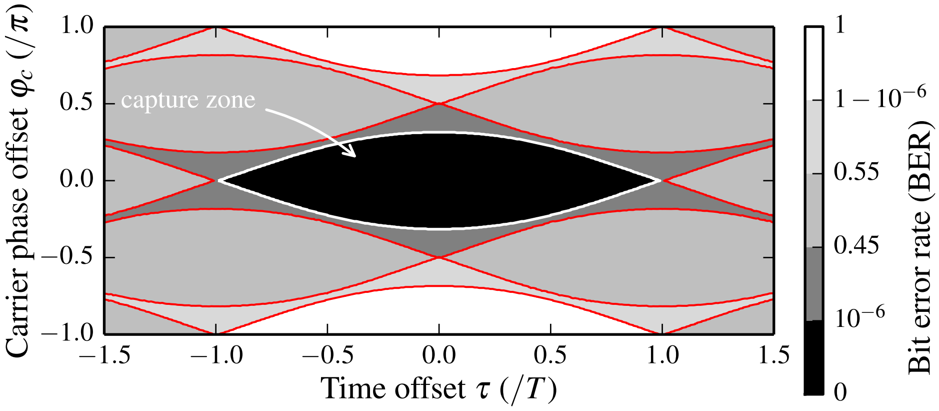
Figure 4.14.: Reception regions of an *interfering* signal with *independent* payload. For reference the reception threshold for a synchronized signal δ_{SIR} (from Figure 4.7a) is also shown.

that for uncoded systems the reception of interfering signals is indeterministic; in contrast, we show analytically and experimentally (Section 8.1) that real systems can receive unsynchronized, interfering packets reliably when using *coded* messages.

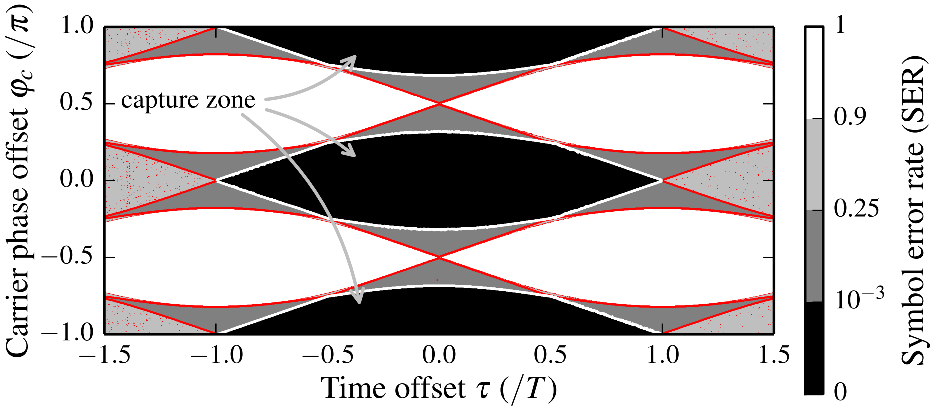
Uncoded transmissions. This case is shown in Figure 4.14a. In this case, a reception is only successful if bits are not flipped by either φ_c or φ_p, and we observe a PRR of 20–30 % in the center region (SIR < -10 dB and $|\tau| < 0.5T$) in our evaluation. The reason for the poor reception performance is visible in Figure 4.15a; the acceptable parameter values of τ and φ_c that lead to an error-free packet reception have tight constraints. The interfering signal must hit into a *capture zone* defined by the signal parameters, which permits the signal to have only small time and carrier phase offsets.

Hard decision decoding. In this setting, the PRR in the central area increases to approx. 60 % (Figure 4.14b). In Figure 4.15b, we see the reason for the increase: while the general shape is the same, we see a second capture zone around $\varphi_c = \pm\pi$. There are two explanations for this. First, we use the same sliced bits from the uncoded case as input for DSSS correlation, which thus possess the same error characteristics. Second, because of the use of absolute correlation values in the correlation (see (4.8)), the adverse effect of large phase offsets can be repaired. Specifically, this means that even if all bits are flipped, the correlation value is still maximal for the correct chipping sequence. This use of DSSS thus doubles the PRR of an interfering signal.

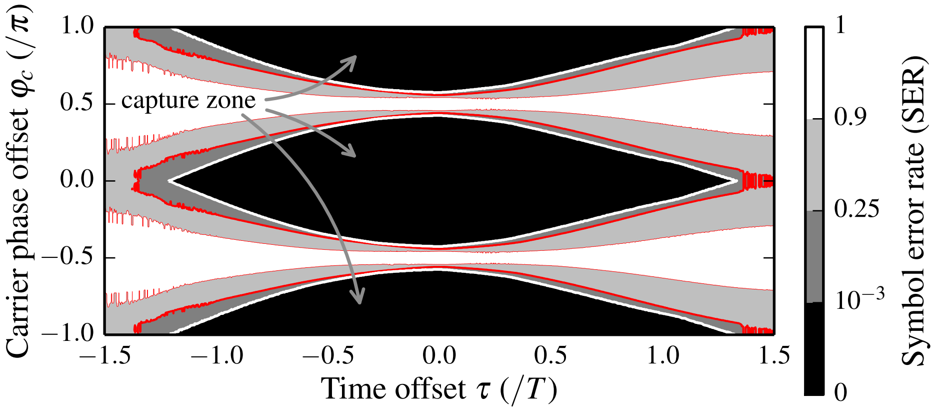
Soft decision decoding. Finally, in Figure 4.14c, we see a central area below SIR $= -23$ dB and a width of $0.25T$ that has a PRR for the interfering signal of approx. 90 %. This means that, if the power difference is large enough, a receiver can ignore a synchronized signal and recover the interfering one despite its offsets. Figure 4.15c shows this in terms of the capture zone. The eye-shaped regions are much wider compared to the other receiver designs, and especially for the central region with minor deviations of τ, the SER is negligible. Problems in the reception only occur for carrier phase offsets such that $\cos\varphi_c \approx 0$. These results show that interfering signals can indeed be received, which helps in collision-aware protocols or other intentional collisions, e.g., in message manipulation attacks on the physical layer. To validate this new result, we present an experimental study of such receptions with real receiver implementations next.



(a) Bit error rate for uncoded transmissions.



(b) Symbol error rate for DSSS/HDD.



(c) Symbol error rate for DSSS/SDD.

Figure 4.15.: Relation between error rates and signal parameters, time offset τ and carrier phase offset φ_c (with SIR $= -40\,\mathrm{dB}$). An unsynchronized packet is successfully received if the parameter combinations fall inside the dark *capture zones*.

4.7. Discussion

Here, we provide a summary of our main findings and highlight key conclusions for the design of protocols that leverage concurrent transmissions. In particular, we summarize how the notion of capture zones enables engineers and protocol designers to choose optimal parameter ranges for signal power ratio, time offset, and carrier phase offset to ensure a successful reception despite collisions.

4.7.1. Signal to Interference Ratio SIR

Our model confirms that when the SoI is above the SIR threshold $\delta_{\mathrm{SIR}} \approx 2\,\mathrm{dB}$, then a successful reception is guaranteed (the capture effect). This is consistent with existing results; for the CC2420 transceiver, GEZER et al. (2010), MAHESHWARI et al. (2008), DUTTA, DAWSON-HAGGERTY, et al. (2010), and SON et al. (2006) report an experimentally observed threshold of about 3 dB. Considering that their channels were not noise-free and that SINR measurements were collected by the radio transceivers themselves, rather than calibrated measurement equipment such that inaccuracies may arise, this is consistent with our results. If it can be ensured that the stronger signal arrives first and the synchronization process succeeds, the SIR-based capture threshold is a valid model for receiver behavior.

A different matter is the case when the SoI is located in the negative SIR regime, i.e., the interfering signal is stronger than the SoI. This situation occurs if an interferer is closer to the receiver or the synchronization process fails because of a collision during the preamble (which is the case, for example, for the collision-aware flooding protocols). Our model gives better insights in this situation and shows that a reception may still be possible *no matter what SIR*, given that the interfering signal parameters are in the capture zone as defined by the time offset τ and carrier phase offset φ_c. Valid settings for these parameters are discussed below.

4.7.2. Time Offset τ

As a guideline derived from the capture zone, the time offset τ should be below $T/2$ for successful concurrent transmissions with identical content, which translates to 250 ns for IEEE 802.15.4. Thus network flooding protocols, for example Glossy, should aim to keep the transmission start time error below this value to ensure a desired PRR above 75 %. If $\tau < 200\,\mathrm{ns}$ can be ensured, the achievable PRR is approximately 90 %. We note that this ensures worst-case performance (i.e., the SoI is always in the negative SIR regime). The actual performance may be higher in situations with positive SIR or successful synchronization.

4.7.3. Carrier Phase Offset φ_c

If the carrier frequency offset at the receiver can be precisely controlled by the senders, there are several options. Interferers can choose $\varphi_c \approx \pm\frac{\pi}{2}$ to minimize their effect on the SoI, reducing their influence to signal demodulation. On the other hand, interferers could aim for the capture zone (e.g., $|\varphi_c| < 0.4\pi$ or $|\varphi_c| > 0.6\pi$ for $\tau = 0$ and SDD) to ensure that their signal is received without errors. There are, however, few approaches in the literature that aim to exploit this. The reason is that the carrier phase at another physical location is hard to predict except in static and free space scenarios because of fading and multipath effects. PÖPPER et al. (2011) show for uncoded QPSK that carrier phase offsets are the major hindrance for a (malicious) interferer to control the bit decisions. In contrast, the results based on our model suggest that such precise phase control is not necessary when DSSS is used, and that intentional message manipulations by deliberate interference are indeed a real threat (WILHELM, SCHMITT, and LENDERS, 2012).

4.7.4. Number of Concurrent Interferers

Our results in Section 4.6.2 explain why the number of interferers only has a small impact on reception performance for concurrent transmissions using identical payload. FERRARI et al. (2011) observed this behavior in their experiments, achieving a stable PRR above 98 % for 2–10 concurrent transmissions. MAHESHWARI et al. (2008) observed that the SIR threshold is not varying with an increasing number of interferers. On the other hand, LU and WHITEHOUSE (2009) reported a decreasing PRR when the number of interferers is increased. However, the Flash Flooding protocol relies on capture, such that increased time offsets may also influence the results. Some related work claims that a greater number of concurrent transmitters cause problems (DODDAVENKATAPPA et al., 2013; WANG, Y. HE, et al., 2012) because “the probability of the maximum time displacement across different transmitters exceeding the required threshold for constructive interference” may increase. Our model shows that these protocol-related issues should be addressed with more precise timing synchronization across the network. For independent payload, we show that 2–3 interferers are sufficient to reduce the

PRR significantly. This confirms the effect reported by GEZER et al. (2010) that the PRR decreases with an increasing number of interferers.

4.8. Summary

In this chapter, we developed a comprehensive analytical model for concurrent transmissions of IEEE 802.15.4 over a wireless channel. As shown in an extensive parameter space exploration, the model recovers insights from experimental results found in the literature and going beyond that, explains the root causes for successful concurrent transmissions exploited in a new generation of sensor network protocols that intentionally generate collisions to increase network throughput or to reduce latency. Our results reveal that power capture alone is not sufficient to explain the performance of such protocols. Rather, coding is an essential factor in the success of these protocols because it crucially widens the capture zone of acceptable signal offsets, increasing the probability of successful reception.

Considering effective jamming and its modeling, we observed that the use of the SIR threshold results in acceptable precision, especially when the signals in the collision are not correlated. For this reason, we mainly consider the power ratio of colliding signals in the protection analysis, but we note that there is additional potential to increase the jamming effectiveness with precisely timed jamming.

5. Protection Analysis

Contents

7.1. System Implementation Overview 87
7.2. Resource Mapping 89
7.3. Subsystem: Detection 90
7.4. Subsystem: Decision 91
7.5. Detection Evaluation 91
7.6. Decision Subsystem Evaluation 93
7.7. Overall System Response Time 95
7.8. Summary 96

5.1. Protection Constraints

In this chapter, we analyze the nature of protection offered by our system of distributed guardians. For ease of exposition, we deliberately use simple propagation and receiver models to deliver the key insights on the concept (we use log-distance path loss and an SINR-based receiver model). The applicability of these models is established in the previous chapter; especially the SINR was shown to be a good and conservative for interfering signals with independent payload, or interfering signals with different waveforms.

5.1.1. Primer on Guardian Detection Sensitivity

In order to correctly detect and demodulate a packet at a receiver, the incoming signal power at the antenna must be greater than the receiver sensitivity. This sensitivity represents the minimum signal power at the antenna that results in a specified packet error performance. The receiver sensitivity is given by $S = N_T N_F \mathrm{SNR}_{\mathrm{min}}$ (Adamy, 2001), where N_T is the thermal noise, N_F is the noise figure of the particular receiver and $\mathrm{SNR}_{\mathrm{min}}$ a modulation-specific threshold for the minimum required signal-to-noise ratio. For IEEE 802.15.4-compliant receivers with O-QPSK modulation on the 2.4 GHz band, optimal coherent detection, and a maximum packet error rate of 1 %, the theoretical sensitivity limit is $S_{\text{min}} = -112.2\,\text{dBm}$. However the standard only demands a sensitivity of at least $-85\,\text{dBm}$, and commercial radio modules exhibit sensitivities ranging between $-92\,\text{dBm}$ to $-110\,\text{dBm}$.[1]

5.1.2. Protection Conditions

For a guardian g to protect a victim node v from an attacker a, the guardian must be able to detect the signals from the attacker and interfere with the attacker's signal at the victim node such that the packet is discarded. This leads to the following two necessary conditions for guardian protection.

Condition 5.1 (Detection)**.** Let P_a be the emitted signal power of the attacker and $L\left(d_{ag}\right)$ be the path loss and fading between the attacker and the guardian. The guardian is able to detect the malicious signals iff

$$P_a L\left(d_{ag}\right) \geq S_g,$$

where S_g is the sensitivity of the guardian.

This condition ensures that the malicious signal can be classified by the guardian and enables it to initiate defensive countermeasures.

Condition 5.2 (Destruction)**.** Let P_a be the emitted signal power of the attacker, P_g be the emitted power of the guardian's interference signal, and $L\left(d_{av}\right)$, $L\left(d_{gv}\right)$ be the path loss between attacker–victim and guardian–victim, respectively. The guardian is able to destroy the packet at the victim if the signal-to-interference ratio is below a certain threshold

$$\frac{P_a L\left(d_{av}\right)}{P_g L\left(d_{gv}\right)} < \gamma_{\text{SIR}},$$

where the threshold γ_{SIR} is determined by the modulation scheme, the interference waveform and how well the victim node is able to suppress such interference. If the interference waveform is zero-mean white gaussian noise, the threshold γ_{SIR} is between 0 dB and 3 dB according to the IEEE 802.15.4 standard (2006). More effective waveforms by the guardians leads to higher

---

[1] An online comparison of various commercial receiver implementations with links to the corresponding data sheets is available at `http://en.wikipedia.org/wiki/Comparison_of_802.15.4_radio_modules`

values of γ_{SIR}. We evaluated this value for the MICAz platform with several waveforms in Section 8.2.3 and find an additional gain of 3–5 dB in jamming effectiveness.

A sensor network remains protected if there exists at least one guardian that fulfills these two conditions for each attacker and victim node pair. In the remaining of this section, we discuss that these conditions are generally easy to fulfill as there exists a large asymmetry between the capability requirements for attackers and guardians.

5.2. Attacker Models

The goal of the attacker is to inject packets of its choice into the protected network. We consider two attacker strategies against the guardians that operate as a wireless firewall:

Definition 5.3 (Brute-force attacker $\mathcal{A}_{\mathrm{force}}$)**.** The attacker tries to overcome the guardian's interference by using a large transmission power P_a:

$$\frac{P_a L(d_{av})}{P_g L(d_{gv})} > \gamma_{\mathrm{SIR}}.$$

The goal of this attacker is to mitigate the interference effects at the receiver by increasing the strength of its transmissions.

Definition 5.4 (Stealthy attacker $\mathcal{A}_{\mathrm{stealth}}$)**.** The attacker tries to choose a small transmission power P_a such that the injected packet is received only by the victim but is not detected by the guardian. The attack is successful iff

$$P_a L(d_{av}) \geq S_v \quad \text{and} \quad P_a L(d_{ag}) < S_g.$$

While the first strategy is easy to implement, it also leads to a steeply increased energy cost of the attacker (an analysis for IEEE 802.15.4 is given in Proposition 5.13). The second strategy is more challenging to implement in practice as very fine-grained power control and a clever positioning of the attacker is required to perform the attack successfully.

5.3. Attacker–Guardian Asymmetry

To illustrate the asymmetry between an attacker attempting to inject malicious packets and the guardian protecting the victim nodes, we consider a log-normal path loss model $L(d) = L_0 d^{-\alpha}$, with reference path loss L_0, distance d, and path loss coefficient α. We use the following metric to capture the asymmetry between an attacker and a guardian:

Definition 5.5 (Attack Range)**.** The (worst-case[2]) attack range $\mathcal{R}$ is the maximum distance d_{av} between attacker a and victim node v such that an attack still succeeds.

An attacker wants the attack range to be as large as possible such that it may launch the attack from arbitrary locations and still remain undetected in the physical world. A large attack range further makes an attacker powerful as it can attack more nodes from a single location. From the perspective of the network, a small attack range is desired as it forces an attacker to expose itself in the physical world and minimizes the number of victim nodes that an attacker may attack simultaneously. If the attacker is constrained in terms of reachable locations, e.g., when the attacker can only attack from outside a building, the number of sensor motes in its attack range may also be zero for all reachable attack positions, effectively thwarting the attack.

Example 5.6. Without a guardian, the attack range $\mathcal{R}_{\mathcal{A}_{\text{force}}}$ of an attacker $\mathcal{A}_{\text{force}}$ using a powerful COTS transmitter ($P_a = 20\,\text{dBm}$) and MICAz victims ($S_v = -94\,\text{dBm}$) under the log-normal model with path loss parameters specified in the IEEE 802.15.4 standard (2006, Annex E.5.3)[3] is $\mathcal{R} = 384.51\,\text{m}$.

The effect of an active guardian is to considerably reduce the attack range. We derive bounds on the attack range for both attacker models $\mathcal{A}_{\text{stealth}}$ and $\mathcal{A}_{\text{force}}$ in the following.

Proposition 5.7. *The attack range $\mathcal{R}_{\mathcal{A}_{stealth}}$ of attacker $\mathcal{A}_{stealth}$ is bounded by $d_{vg} / \left(\sqrt[\alpha]{S_v/S_g} - 1 \right)$ if a guardian is present.*

Proof. To prevent detection, the attacker must satisfy two conditions simultaneously: its received signal power at the victim must be above the sensitivity level of the victim and below the sensitivity of the guardian:

$$P_a d_{av}^{-\alpha} \geq S_v \tag{5.1}$$

$$P_a d_{ag}^{-\alpha} < S_g \tag{5.2}$$

---

[2] This notion of attack range is under worst-case assumptions where attacker, victim, and guardian are on one line with the victim in the middle, maximizing the distance between attacker and guardian d_{ag} for a given attacker–victim distance d_{av}.

[3] We converted the parameters given in the standard to our non-logarithmic model: reference distance $d_0 = 8\,\text{m}$, $\alpha = 3.3$, path loss at d_0 is $L_0 = d_0^{\alpha} 10^{58.5/10}$. Later derivations use distance ratios and are thus independent of the choice of L_0.

Following (5.1) and (5.2), the attacker must choose a power P_a that satisfies $d_{av}^{\alpha} S_v \leq P_a < d_{ag}^{\alpha} S_g$. Thus, such a P_a only exists iff $d_{av}^{\alpha} S_v < d_{ag}^{\alpha} S_g$. Using this condition and the relation $d_{ag} \leq d_{av} + d_{gv}$ (by triangle inequality), we find that the attacker $\mathcal{A}_{\text{stealth}}$ can only remain undetected if its distance to v satisfies

$$d_{av} < \frac{d_{vg}}{\sqrt[\alpha]{S_v/S_g} - 1}. \tag{5.3}$$

The bound is independent of the attacker's transmit power. □

The asymmetry with the attacker model $\mathcal{A}_{\text{stealth}}$ is provided by the sensitivity ratio S_v/S_g (as discussed in Section 5.1.1): the higher this ratio, the smaller is the attack range $\mathcal{R}_{\mathcal{A}_{\text{stealth}}}$. Additionally, this range only applies if the attacker maximizes its distance to the guardian, such that it has to find a position where the attack is feasible first. The following example applies typical values to illustrate how the attack range is considerably reduced compared to 5.6 where no guardian is present.

Example 5.8. In Section 9.2, we present an indoor application scenario with MICAz sensor motes and a KillerBee attacker. Assuming a near-optimal receiver at the guardian with a sensitivity of −110 dBm (−112.2 dBm is the theoretical sensitivity limit of an ideal IEEE 802.15.4 receiver), tolerating a limited number of bit errors and hence a non-zero false positive rate with a conservative improvement of 6 dB to keep the false positive rates low (say less than 0.001 %), the sensitivity of the guardian becomes $S_g = -116\,\text{dBm}$, resulting in a ratio S_v/S_g of 22 dB. With a maximal protection distance $d_{gv} = 10\,\text{m}$ and $\alpha = 3.3$, the attack range decreases to $\mathcal{R}_{\mathcal{A}_{\text{stealth}}} = 2.75\,\text{m}$, which is a reduction by a factor of 140 compared to the case where no guardian is present.

Proposition 5.9. *The attack range $R_{\mathcal{A}_{force}}$ for attacker model $\mathcal{A}_{force}$ is bounded by $\sqrt[\alpha]{P_a/(\gamma_{SIR} P_g)}\, d_{gv}$.*

Proof. If the attack signal is detected, a brute-force attacker has still the opportunity to overcome the interference signal if

$$\frac{P_a d_{av}^{-\alpha}}{P_g d_{gv}^{-\alpha}} \geq \gamma_{\text{SIR}}.$$

Consequently, the attacker must choose its distance such that

$$d_{av} \leq \sqrt[\alpha]{\frac{P_a}{\gamma_{\text{SIR}} P_g}}\, d_{gv}. \tag{5.4}$$

The attacker is assumed to be power-limited, such that this bound exists. □

Example 5.10. In Section 9.2, a sensor network with several MICAz motes captured by an attacker is considered, which our guardian system effectively disconnects from the network. MICAz motes (like many other COTS transmitters) have a maximum output power of $P_a = 0\,\mathrm{dBm}$, the transmit power of a guardian node is $P_g = 20\,\mathrm{dBm}$, and the interference waveform being used is assumed as $\gamma_{\mathrm{SIR}} \approx 3\,\mathrm{dB}$. With a maximal protection distance $d_{gv} = 10\,\mathrm{m}$, the attack range $\mathcal{R}_{\mathcal{A}_{\mathrm{force}}}$ of the brute-force attacker is brought down to $0.70\,\mathrm{m}$.

Definition 5.11 (Protection coefficient)**.** The factor ρ that couples the maximum attack range of the brute force attacker to the distance between guardian and victim, given in $d_{av} \leq \rho d_{gv}$, with $\rho = \sqrt[\alpha]{\frac{P_a}{\gamma_{\mathrm{SIR}} P_g}}$, is called the protection coefficient.

The protection coefficient is a metric for the relation between attack and protection distance. A small value indicates that the attacker must move in much closer for the same guardian distance d_{gv}, which implies a larger protection radius and thus a better protection. In Section 9.1, we evaluate this metric in several real-world scenarios.

Last but not least, a brute-force attacker also has to pay a price in terms of energy investment to successfully mount its attack.

Definition 5.12 (Energy cost)**.** The energy cost of an attacker $\mathcal{A}_{\mathrm{force}}$ is the ratio of the energy required by the attacker to send a packet over the energy required by the guardian to block the packet.

Proposition 5.13. *The energy cost is*

$$\frac{P_a d_{av}^{-\alpha} \ell \cdot 32\,\mu\mathrm{s}}{P_g d_{gv}^{-\alpha} t_{interfere}}, \tag{5.5}$$

where ℓ *denotes the packet length in bytes and* $t_{interfere}$ *the duration of the interference signal from the guardian.*

Example 5.14. If, for ease of exposition, we assume that the attacker and the guardian emit at the same power and are at the same distance from the victim, the (received) power ratio $P_a d_{av}^{-\alpha} / P_g d_{gv}^{-\alpha}$ is equal to one. Considering a typical IEEE 802.15.4 packet size of 32 bytes and a necessary interference duration of $t_{\mathrm{interfere}} = 16\,\mu\mathrm{s}$ to destroy one symbol, the energy cost is 64. In other words, the guardian needs to invest 64 times less transmit energy to destroy a packet than the attacker needs to invest to transmit the packet to the victim.

This observation shows again an asymmetry in the required energy between an attacker and a guardian. While an attacker might in some cases still be able to slip through the protection of the guardians, the effort required to do so is considerably higher than the effort required by the guardians. Therefore, the guardians are able to effectively and efficiently control and block undesired transmissions from attackers before they can reach the network.

5.4. Summary

We observed that the nature of protection of such a remote access control system depends highly on the spatial distribution of all involved parties. Because the physical layer effect of RF interference is used, the power level of all involved signals plays a major role in the the outcome of reception, as we have seen also in the previous chapter. However, based on the assumption that the capture threshold yields a good approximation of receiver performance, we have found a parameter that can be used to characterize the protection of such a system, the protection coefficient ρ. It captures how much closer the attacker has to approach devices in the network to attack them, in comparison to the distance to the wireless firewall. Based on our model, we observe that the attacker has to approach the network by less than a tenth of the guardian distance.

Part II.

Design and Implementation of WiFire

6. System Architecture

Contents

8.1. Validation of the Collision Model 101

8.2. Experimental Evaluation of Jamming 106

To meet the technical challenges that we defined, it is important to understand the design options that are available, as well as the performance capabilities of the underlying systems that are used in the implementation. This chapter gives a closer view of the design space of wireless firewalls, and the architectural choices that influence the overall system performance and implementation complexity, and puts forward a system architecture, called WiFire, that realized the concept of wireless firewalls.

6.1. Design Space of Wireless Firewalls

First, we discuss alternatives for designing transparent wireless firewall systems. Due to the conflict between flexibility and real-time capability, we are mainly concerned with the optimal mapping of the three operational steps *RF analysis*, *violation detection*, and *jamming* to the computational resources found in modern radio platforms.

6.1.1. Store-and-Forward Wireless Firewalls

Taking a step back, we start by reviewing the existing firewall design concept of store and forward, and conceive two basic alternatives to design store-and-forward wireless firewalls.

Enforce Store-and-Forward A wireless firewall could be inserted into the data path as a further hop, such that standard firewall processing can be applied since store-and-forward operation is enforced.

While being straightforward, this approach represents a severe transparency violation, as essentially all protocols would have to be changed and configured to account for this explicit additional hop. The firewall has to rely on the end devices to make these configurations (in particular, insert it into their data path), hence no completely central policy enforcement is achieved. Furthermore, it would fundamentally halve the throughput of the wireless channel since each transmission would be doubled and the firewall could easily become a bottleneck unless it is given some priority when accessing the wireless medium (which again may mean modifications to existing protocols). Last but not least, such an "enforce store-and-forward" wireless firewall faces analog threats to a regular access point with respect to impersonation; attackers are likely to soon come up with rogue wireless firewalls. Our approach avoids all of these problems by being transparent. In particular, this rules out impersonation attacks.

Emulate Store-and-Forward A second approach could be to emulate a store-and-forward behavior for the wireless firewall. To that end, the firewall could jam every packet it sees, yet at the same time receive and store it, classify and forward it if trusted. The concurrent sending (jamming) and receiving of packets over a single channel is technically very challenging but may, in principle, be feasible (see Choi et al. (2010) and Gollakota, Hassanieh, et al. (2011) for first steps along these lines).

Hypothesizing its feasibility, this approach avoids the necessity of protocol changes and special configurations of end systems. However, trusted traffic is negatively affected by the wireless firewall with respect to delay and also losses, as the firewall has a high jamming activity and thus induces a high interference for concurrent transmissions. In particular, this is very bad for "polite" medium access protocols like CSMA/CA. In fact, an "emulate store-and-forward" wireless firewall almost behaves as a constant jammer, only becoming silent if there is no activity on the wireless channel. Again, throughput is halved because a single transmission is effectively doubled.

So, in comparison to our approach, an "emulate store-and-forward" wireless firewall exhibits a number of disadvantages while the system challenges can probably be considered at least as hard. To be fair, it offers the potential to enable more complex rule sets since the classification can be done without strict timing requirements.

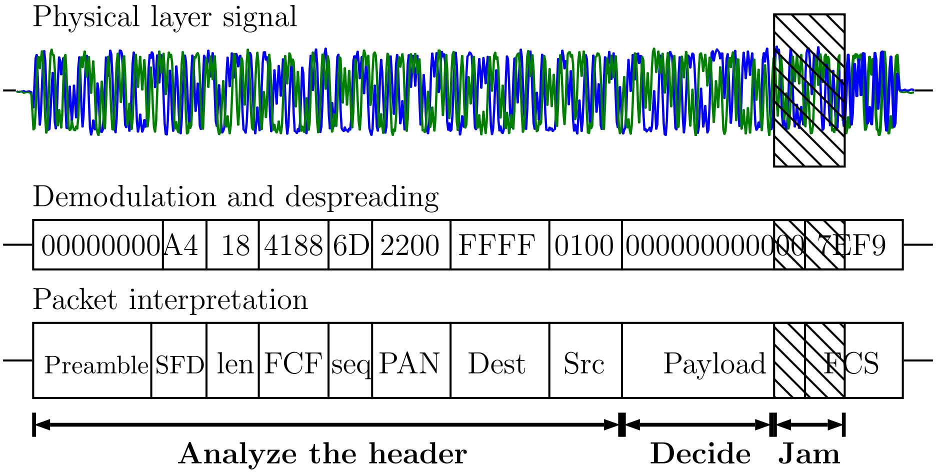


Figure 6.1.: Operation of WiFire: first, the packet's signal is demodulated to access its content for classification. WiFire must wait for the payload bytes to arrive (t_{listen}) before the rule checker can start. When the packet is declared malicious (after t_{decide}), the transmission of interference is prepared (t_{init}), and the offending packet destroyed ($t_{\text{interfere}}$) before it may be received.

6.1.2. Operational Requirements

As a result from this discussion, we follow a cut-through approach for WiFire mainly for its transparency, rising to the following challenge: there is a firm deadline for all the actions of WiFire, namely when a packet is completely received at the destination node. Hence, we face stringent *real-time* requirements for WiFire. Thus, we are interested in the real-time performance characteristics of WiFire, especially with regard to the system challenges of low-latency and reliable packet dropping at the receiver. Starting with this chapter, we only consider the aspects of the IEEE 802.15.4 standard in terms of physical layer details.

To illustrate the operation of WiFire and to identify the technical challenges that arise, we discuss the interception of a single malicious packet. Figure 6.1 shows our packet, a 26 byte IEEE 802.15.4 data packet going to the broadcast address `0xFFFF` of sub-network `0x22`. The transmission duration of this packet is 832 µs, and it starts with a physical layer header, link layer header, and payload; it ends with a 16 bit checksum (CRC). WiFire operates as follows:

- It first detects the packet using the preamble and start-of-frame delimiter (SFD), which signals the beginning of the packet. Then, it proceeds to demodulate the content of the packet, gaining access to header fields and payload, which is subsequently used to decide whether the packet is malicious. The longer the reception period, denoted by t_{listen}, the more content is available to classify the packet; on the other hand, if WiFire listens for too long, it may be unable to still destroy the packet. If the decision is based on the header fields, t_{listen} is 480 µs, permitting a maximum system response time of 352 µs; if the full payload is considered, t_{listen} is 768 µs and only 64 µs remain. This illustrates that WiFire must fulfill very strict timing requirements to both classify and destroy the packet.

- As soon as the necessary content is available, the rule checker is started to compare the detected packet to the stored security policy. The execution time of the rule checker is denoted by t_{decide}.
- If the rule checker concludes that the packet is violating the security policy, it initiates the transmission of a burst of interference. The time required to set up this operation is denoted by t_{init}.
- Finally, the transmission of interference must reliably destroy the packet. In the IEEE 802.15.4 standard, a single bit error is sufficient to destroy the packet, even if the bit error is in the last CRC byte. The duration of interference ($t_{\text{interfere}}$) must be long enough to force at least one bit error, but not necessarily longer.
- The overall system response time is denoted by t_{response}, which is defined as the time from the start of classification to the end of the interference, i.e., $t_{\text{response}} = t_{\text{decide}} + t_{\text{init}} + t_{\text{interfere}}$. We aim for a response time below 64 µs.

In summary, the system must fulfill tight timing requirements; the overall time to listen to a packet and respond must be much smaller than the packet's duration.

6.1.3. Architectures for Wireless Firewalls

While the strict real-time requirements mandate to stay as close to the hardware as possible, this severely reduces the flexibility of the system. Thus, we aim to build a *software-defined* wireless firewall, putting as much functionality into programmable hardware and software as possible. Current software-defined radio (SDR) platforms can be divided into four logical tiers: hardware, reprogrammable hardware logic, firmware, and the host. Each tier can be used to implement functionality of WiFire, and has benefits and drawbacks. We first describe these tiers and then explore different approaches to allocating the three building blocks of WiFire to these tiers, discussing the tradeoffs that arise.

Remote host

The prevalent architecture of SDRs is to use an RF front-end that collects sampling data, performs signal processing steps (filtering, digital down-conversion), and sends the data to a host computer. The main hindrance for real-time applications in this architecture are the delay and bandwidth limitations introduced by host communication. NYCHIS et al. (2009) show that it is necessary to move time-critical components closer to the radio to alleviate this problem. Recent research results show that this issue can also be overcome with clever design (e.g., the Sora platform (TAN et al., 2009) meets the timing requirements of 802.11g), with the benefit of keeping computations on a general purpose processor (GPP) that is easy to program and debug. On the other hand, analyzing signals along different dimensions puts a high computational burden on the system, e.g., results with Sora show that two dedicated cores are necessary to receive 802.11g communications. This limitation may interfere with WiFire's goal to support multiple wireless technologies concurrently.

Reprogrammable hardware

Another resource readily available on current SDR platforms (such as WARP, Sora, USRP1+2) are FPGAs. They provide a convenient way to interconnect different hardware components on the register transfer level to form complex systems. We can tap this resource and implement parts of WiFire in reprogrammable hardware. Using massively parallel computations means allows to cut the latency of the sequential GPP execution down. Further, exact timing with an explicit clock is crucial to achieve a high precision and effectiveness in jamming. NGUYEN et al. (2014) present an FPGA-based system that achieves a 80 ns reaction time for IEEE 802.11.

Firmware

However, an FPGA-only approach has practical disadvantages. Such systems have long development cycles because of the complex place-and-route process. Furthermore, WiFire has many volatile components: most importantly, the policies must be flexible and easily reconfigurable. Frequently, these issues are reduced by the use of GPPs directly connected or even implemented on the FPGA. For example, the USRP2 platform has a 50 MHz aeMB softcore on the FPGA that runs its firmware. Such processors share the practical benefits of host-based GPPs, though their capabilities are generally limited, prohibiting extensive computations. Yet, this hybrid approach offers a flexible way to implement parts of a wireless firewall with real-time but without high performance requirements.

Hardware

Parts of the functionality can also be implemented in hardware, e.g., as an ASIC design. This results in the best achievable performance and fidelity, but also leads to systems tailored for a specific application that cannot easily be adapted or upgraded to different requirements. Functionality that is basic to the operation and rarely changes is a good implementation candidate for this tier.

6.2. The Architecture of the WiFire System

6.2.1. System Architecture

The system architecture is depicted in Figure 6.2. The workflow of the system is (from left to right): *(i)* detecting and demodulating relevant transmissions, *(ii)* forming decisions based on the packet content, and *(iii)* interfering with malicious packets.

6.2.2. An Overview of the System Components

Next, we provide a brief description of each functional component. The details of the implementation are given in the following chapter.

Detection Subsystem

This subsystem continuously scans the RF medium to detect any packet that might be received by the network and receives and delivers the packet's content to the subsequent decision subsystem. It consists of a receiver optimized for speed that synchronizes to packets and demodulates the contained bytes, and a framer that interprets the incoming data, providing access to header fields and payload bytes.

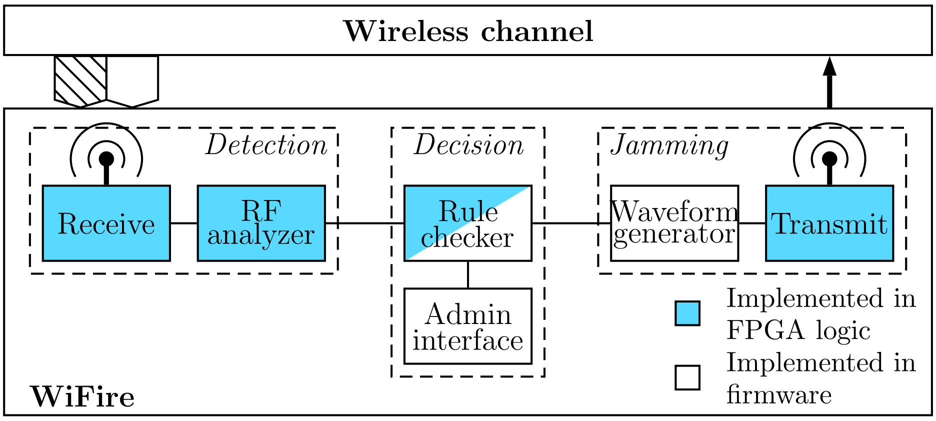


Figure 6.2.: Component diagram of the guardian implementation. We consider two different rule checker implementations (firmware/FPGA).

Decision Subsystem

The decision system is triggered when a pre-defined point in the packet is reached (e.g., when the full link layer header is available), in order to trigger the decision process on whether the packet should be blocked.

Rule checker: The rule checker classifies incoming packets according to a pre-defined policy. It is the critical component for real-time operation because the overall reaction time mainly depends on its execution time. Therefore we have implemented and evaluated two different versions: *(i)* firmware code written in C and running on the USRP2's (soft-) micro-controller, which offers runtime reconfigurability but is comparatively slow, and *(ii)* an implementation in FPGA logic that reduces the reaction time, but the security policy must be specified at compile time. In both implementations, the rule checker notifies the interference subsystem with an interrupt that a short burst of interference must presently be generated to destroy the malicious packet.

Definition of rules: The firmware-based rule checker allows to define content-based rules in the style of `iptables`, defining rule chains that consist of one or more rules, each with zero or more matches (such as source or destination address) via an administration interface. We implemented a command line tool (`wftables`), which generates a data structure that can be directly interpreted by the firmware rule checker. An example is the following rule definition with two matches (preventing the reception of all control packets going to the broadcast address in PAN `0x22`):

```
wftables -A -m dst --pan 0x22 --addr 0xFFFF
            -m type --ctrl
         -j DROP
```

This mechanism allows to define complex access policies and deploy them on the distributed guardians. An example chain that can directly be interpreted by WiFire's rule checker is

```
wftables -A -m ftype --data       # Rule 1
            -m dst --mode 2 --pan 0x11 --addr 10
            -j JAM
wftables -A -m ftype --ctrl       # Rule 2
            -m ! src --mode 2 --pan 0x11 --addr 1
            -j JAM
wftables -A -j ACCEPT             # Default policy
```

This rule set specifies that all transmissions are allowed (the default policy), except for:

1. data frames coming from node `10` in the current networks (with the identifying PAN ID), and
2. control frames that are sent from other source addresses than `1`.

Rule 1 is used to block all data traffic from a node of choice, virtually separating it from the network. This can be used to enforce a fast *node revocation*, e.g., when the keying material of this node is leaked. Rule 2 enforces that only node 1 (e.g., the PAN coordinator) can send control messages on the current channel to protect other nodes from being hijacked. The FPGA rule checker uses hardware gates to compare detected packet bytes to a table of predefined values in parallel, such that the execution time is considerably reduced.

Selective Interference Subsystem

When a packet is classified as malicious, the guardian takes action and prevents the reception of the packet by its protected sensor motes.

Waveform generator: With a software-defined radio, arbitrary waveforms can be specified using a sequence of I/Q samples, which completely defines a transmitted signal. Our guardian implementation supports continuous wave (CW), noise, and arbitrary symbols modulated using the IEEE 802.15.4 2.4 GHz PHY as interference waveforms.

Transmission: The I/Q samples are finally sent out on the physical layer by modulating a jamming signal onto a carrier in the 2.4 GHz band, and amplified up to a maximum output power of 20 dBm (100 mW). External antennas and amplifiers can be used to boost the effectively radiated power further.

6.3. Summary

When considering the optimal architecture of wireless firewalls, we discussed that there are several feasible avenues to take. A major design goal of WiFire is to have an extensible research platform that supports other applications than the wireless firewall, therefore more dedicated wireless firewall systems may take the approach to place all components closer to the hardware to achieve better performance, and to reduce hardware costs and energy consumption.

7. System Implementation

Contents

9.1. Evaluation of the Protection Coefficient 113
9.2. Selectivity Experiments 120

7.1. System Implementation Overview

We implement the guardian nodes on the USRP2 software-defined radio (SDR) platform. The SDR paradigm allows full physical layer access by the guardian; in contrast, off-the-shelf receivers do not allow low-latency access to detected symbols, nor do they offer freedom in choosing the transmitted interference waveform. However, these factors are crucial to enable a deep look into packets and to use the most effective interference waveform with a given power budget. For example, related reactive jamming systems implemented on sensor motes (Z. He and Voigt, 2011; Martinovic, Pichota, and Schmitt, 2009; O'Flynn, 2011) do not allow to classify based on content and to destroy packets during transmission. Still, an implementation on the USRP2 is not straightforward because the time to classify and intercept a packet is below 500 µs, which forces tight timing constraints on the system. In the common host-based architecture for software radio using GNU Radio, where digital samples are forwarded to a host via Ethernet and signal processing is performed on the host CPU, large and non-deterministic latencies prevent to meet real-time requirements (Nychis et al., 2009). In our implementation, we fulfill the requirements of real-time detection and subsequent destruction of a packet on the air; we design the guardian as a stand-alone system, completely implemented on the USRP2's FPGA and on-board microcontroller. The system is optimized for speed to reach reaction times of tens of microseconds to allow access to the full payload and still reliably destroy the packet.

7.1.1. Implementation Guidelines

Our choice is to reuse as much as possible of the existing USRP2 design to provide patches to the mainline code. This also requires a system design for WiFire that is oriented on the tiers presented above. We present the rationale of the design here.

The components with strict timing or high bandwidth requirements must be implemented on the FPGA level. This is mainly the case for the RF analyzers, and generally speaking the rule checker can achieve a minimal reaction time when it is implemented on FPGA as well. The drawback of this approach is that it lacks flexibility in the definition and modification of rule sets. We chose to implement the rule checker in the firmware, where the rules can directly be changed with command frames via Ethernet. The tasks that are not time-critical are moved to the host. Here, the tasks are the configuration of the system, and the logging.

Design tools

The firmware uses an adapted version of the `gcc` toolchain to target the MicroBlaze processor and is preserved. For the RF analysis parts, we wanted to stay close to the standard workflow of RF engineers with model-based design tools. We designed our RF analysis components with Simulink and generated the required Verilog code using the Xilinx code generator. This way, we can model and simulate our sub-systems before integration into the system, and can integrate existing work of analyzers and receiver implementations in our system.

7.1.2. USRP2 Platform Details

The system requirements we have derived from our goals, especially the need for real-time operation makes it necessary for us to depart from this split architecture where RF data forwarding add additional delay; our system must be implemented on the USRP2 itself.

USRP2 hardware description

The USRP2 platform is equipped with a Xilinx Spartan-3 FPGA running with a clock speed of 100 MHz, which provides sufficient performance and a fine-grained timing resolution of 10 ns per cycle. Additionally, the USRP2 has enough free resources (only 40 % of the FPGA is occupied) to add our prototype while reusing the functionality of the original system. We modified

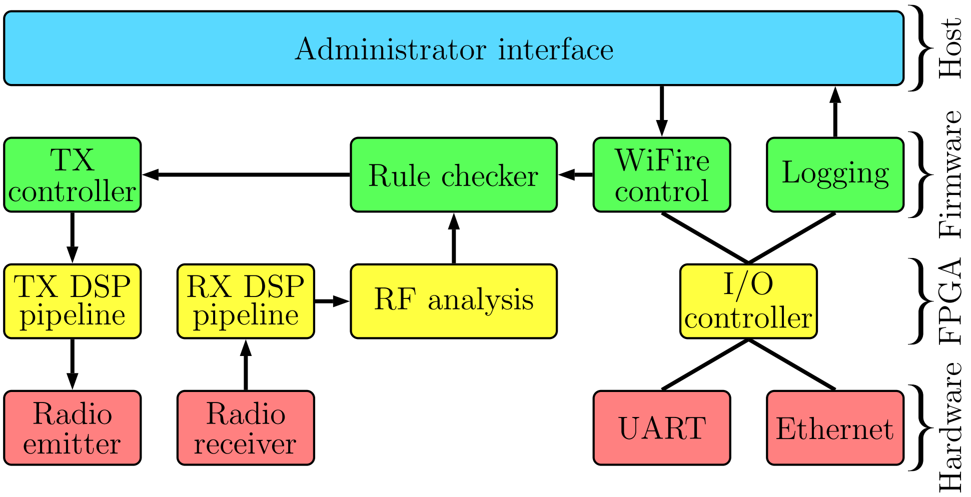


Figure 7.1.: WiFire's design, showing the functional components and their allocation to the USRP2's resources.

the UHD FPGA code and firmware from Ettus Research. The operation of the USRP2 is controlled by a softcore processor in FPGA logic that executes firmware code written in C. This offers an easy integration path for our system and a maximum of reuse. However, the sequential program execution of the firmware may introduce larger time deviations into the system. The magnitude of these effects are evaluated in the next section. We added a detection module in FPGA logic that receives complex samples from the RX DSP pipeline and interrupts the firmware on a detection event. We altered the firmware to await such interrupts and to initiate the jamming process, which causes the USRP2 to start sending a ready-made jamming waveform on the channel.

7.2. Resource Mapping

In WiFire, we aim for flexibility, so we must place the components to different tiers according to their timing and computational requirements, but with a maximum of configurability. In this following, we identify the critical paths in the operation of the wireless firewall, and find an appropriate mapping to the hardware that achieves our goals. A more detailed view on the components is given in Figure 7.1.

7.3. Subsystem: Detection

This subsystem continuously scans the RF medium to detect any packet that might be received by the network and receives and delivers the packet's content to the subsequent decision subsystem. It consists of a receiver optimized for speed that synchronizes to packets and demodulates the contained bytes, and a framer that interprets the bytes according to the IEEE 802.15.4 standard, providing access to header fields and payload bytes.

7.3.1. IEEE 802.15.4 Receiver

We implemented an IEEE 802.15.4 receiver using coherent O-QPSK demodulation and a correlating direct sequence de-spreader to recover symbols in FPGA logic. The de-spreader removes the direct sequence spreading code used in the 2.4 GHz PHY. The receiver directly operates on the stream of complex (I/Q) samples coming from the USRP2's analog-to-digital converter: it first synchronizes with the incoming preamble, then detects each detected symbol (4 bit of information), and finally delivers it to the framer. The FPGA implementation ensures that the detection latency is limited, we measured a delay in the receiver below 4 µs from the presence of the signal on the channel to the time it is available for interpretation by the rule checker.

7.3.2. Packet Framing

This component interprets the received symbols according to the IEEE 802.15.4 packet definition, granting access to header fields and payload. IEEE 802.15.4 supports several address modes that use varying header layouts that must be supported by the framer. This component also notifies the rule checker that a new packet was detected via interrupts and provides a memory mapping that can be queried to gain access to header fields and payload bytes.

7.3.3. Detector Implementation

For every clock cycle, a new complex RF sample is available as input to the detector module. Considering the symbol duration of 16 µs in 802.15.4, we have 1600 clock cycles per symbol available, which enables complex detector designs. We implemented a PHY header (preamble+SFD) detector in our prototype. First, we perform an MSK demodulation on the signal (as explained in (Schmid, 2006)), and feed the resulting stream of chips into a correlating receiver that detects a SHR on the channel accurately. Once a SFD is detected, an interrupt is triggered at the programmable interrupt controller. Our detector adds a 4 µs delay after the SFD because of the time needed for correlation.

7.4. Subsystem: Decision

The decision subsystem is pivotal to support a selective operation, and ensuring that devices in the vicinity are protected by WiFire.

Rule decision

The decision system is triggered via interrupts by the packet framer when a pre-defined point in the packet is reached (e.g., when the full link layer header is available) to trigger the decision process on whether the packet should be blocked; it classifies incoming packets according to a pre-defined policy. It is the critical component for real-time operation because the overall response time mainly depends on its execution time. Therefore, we implemented two different versions: *(i)* firmware code written in C and running on the USRP2's (soft-) micro-controller, which offers runtime reconfigurability but is comparatively slow, and *(ii)* an implementation in FPGA logic that reduces the response time, but the security policy must be specified at compile time. In both implementations, the rule checker notifies the interference subsystem via interrupts that a short burst of interference must now be generated to destroy a malicious packet. The firmware-based rule checker allows to define content-based rules in the style of `iptables`, defining rule chains that consist of one or more rules, each with zero or more matches (such as source or destination address). We implemented a command line tool (`wftables`), which generates a data structure that can be directly interpreted by the firmware rule checker. This mechanism allows to define complex access policies and to deploy them on the distributed WiFire guardians. The FPGA rule checker uses hardware gates to compare detected packet bytes to a table of predefined values in parallel, such that the execution time is considerably reduced.

7.5. Detection Evaluation

This part of the evaluation is concerned with the speed and precision of the packet detection subsystem. A correct operation is crucial because only detected packets are classified (and destroyed if necessary). We evaluate the

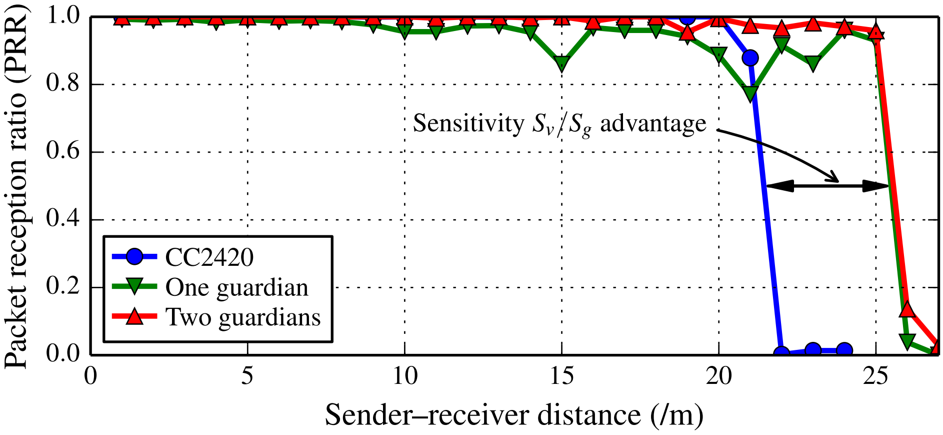


Figure 7.2.: Receiver sensitivity, detection performance: packet reception ratio with increasing distance. The guardian shows a better sensitivity, allowing to protect sensor motes from a distance.

guardian's ability to detect packets correctly while varying the distance between attacker and guardian (d_{ag}). Due to path loss, the signal strength gradually reduces, making it harder to distinguish signal from noise. As reference, we compare the performance of a standard-compliant MICAz receiver in this experiment to observe if the sensitivity of the guardian is better than that of the motes in the protected network.

Experimental methodology

The receiver under test (either guardian or MICAz mote) is placed at a fixed position at one end of the experimentation area, a 3 m wide hallway. This indoor scenario is a challenging test for the receiver because multipath effects and inter-symbol-interference (ISI) increase the difficulty to detect signals correctly. The guardian runs on a USRP2 with an XCVR2450 transceiver and omni-directional antennas (3 dBi gain). The attacker is a MICAz mote with its default antenna, using its maximum output power (0 dBm) to transmit 100 packets/s (which contain 48 symbols and have a duration of 768 µs) for 10 s.

Starting from a 1 m attacker–receiver distance, we then move the attacker to a set of measurement positions with distances $d_{ag} = 1, \ldots, 30$ m. For each position, the attacker transmits 1,000 broadcast packets. The receiver (guardian or MICAz) counts the successfully detected packets (confirmed by a CRC check) for the packet reception ratio.

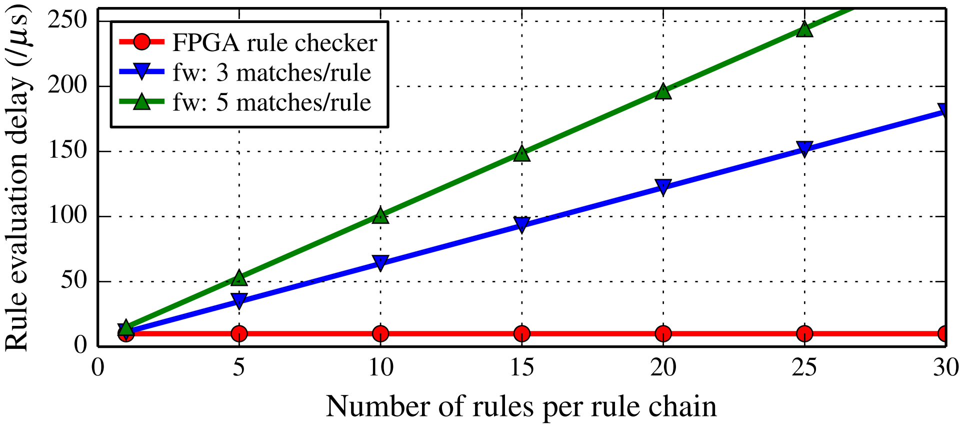


Figure 7.3.: Reaction delay for two rule checker implementations: firmware with different rule configurations (flexible but slow) and FPGA (static rules but fast).

Results

The detection results are shown in Figure 7.2. The COTS receiver performs well in this experiment, with a detection radius of approximately 20 meters. With 12 out of 20,000 packets in this range, only a limited number of packets was not received successfully. Our FPGA-based receiver implementation has an increased reception radius (up to 25 m), indicating a higher sensitivity compared to the MICAz mote. However, the performance for longer distances deteriorates slightly because the receiver is sensitive to multipath fading effects, causing a fraction of packets to be missed in some locations. We mitigate this effect by using a second guardian. In this setup, the number of missed packets is again reduced. Overall, the system fulfills the goal of a better receiver sensitivity discussed in the WiFire evaluation, such that the guardians are suitable to protect sensor nodes from a distance.

7.6. Decision Subsystem Evaluation

By reaction delay t_{react}, we refer to the time from receiving the content bytes of interest until the interference is finished. This delay affects how deep the guardian can look into a packet, because the interference must overlap with it. It is mainly dependent on the execution time t_{decide} of the guardian rule checker to decide whether the packet should be destroyed. An overly slow decision process may shift the selective interference behind the end of the packet. We put both firmware and FPGA-based rule checker to the test.

Experimental methodology

The measurements are taken using the FPGA's 100 MHz internal clock to record timestamps, which allows us to reach a timing precision of 10 ns. The rule evaluation is timed from the instant a hardware interrupt signals the detection of an IEEE 802.15.4 header by the framer to the instant the rule system returns a verdict on how to treat the packet. In the firmware-based rule system, this is the return of the C function call that activates the rule checker. For the FPGA-based rule checker, the timestamp is taken when the rule checker interrupt to start the interference arrives in the firmware.

We vary the number of rules in the chain, and the number of matches in each rule (each match is a C function that reads data from the framer and compares these values to constants stored in the chain). The rules are chosen such that none of them matches the packet, hence we measure the worst-case run time where the chain is traversed completely. As the rules are evaluated in parallel in the FPGA-based implementation, increasing the rule set size affects the FPGA resource usage and not the timing.

Firmware-based rule checker results

The compound execution times depicted in Figure 7.3 show that the reaction delay is depending on the rule set used. The reason is that our implementation follows the general design of `iptables` using linked lists with variable size, such that a small overhead occurs for the evaluation of each rule chain, rule, and the matches contained in a rule to traverse the list. As the used micro-controller only supports a single thread, the required execution time is deterministic but increasing with each rule.

To break the delays into components, we analyzed the rule checker implementation in depth. First, a constant time of 4.03 µs is needed to enter the interrupt handler, jump into the rule checker and back, and trigger the interference process. This cost is independent of the chain's contents and is paid for each detected packet. To evaluate one rule in the chain, the guardian needs 0.26 µs for evaluation (mainly the time to traverse the chain). Evaluating the time needed for individual rule, each match needs 0.34 µs to start the execution of the associated test function. The overall running time of a match function depends on the logic of the match itself. Considering the address match, a match with representative complexity, the execution takes 1.86 µs before the function returns. All matches in a rule are checked sequentially,

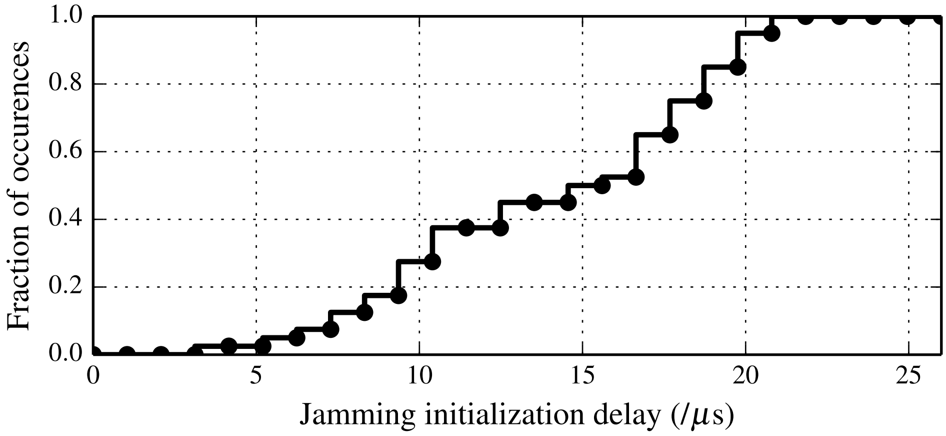


Figure 7.4.: Reaction time t_{init} of the jammer.

such that the execution times add up. For example, the execution of one rule with 3 matches accounts to 5.58 µs. So, when considering a chain with 20 rules of 3 matches each, the guardian requires 116 µs or approximately 4 payload bytes to react in the worst case (when all rules are traversed).

FPGA-based rule checker results

To allow more deterministic decision delays and to support a deeper look into the packet, we also implemented a rule checker implemented in FPGA logic that is less flexible but provides faster reaction times. Using this approach we are able to cut the latency down to a limited number of FPGA clock cycles, i.e., below one microsecond up to 10 µs. This enables us to achieve an overall reaction time of $t_{\text{response}} = 39\,\mu\text{s}$, even with complex rule sets. This means that the guardian can base its decision on the complete payload and still hit the CRC bytes at the end of the packet to cause a packet drop.

7.7. Overall System Response Time

First, we determine the jamming initialization time after a packet is detected (t_{init}). More precision enables a "surgical" jamming where we can operate on a (sub-) symbol level. Further, we evaluate whether the firmware-based approach with its indeterministic timing is sufficient for our strict timing requirements. For this experiment, we place a MICAz mote close to the RX antenna of the jammer and start detecting its transmissions. The jammer schedules a jamming request as soon as an SFD is detected and initiates the transmission of the jamming signal. Using a second USRP2, we monitor and collect samples from the channel and measure the time from the end of the SFD and the beginning of the jamming signal. We use power envelope detection to identify the start of the packet and the start of the jamming signal; the resulting t_{init} is the elapsed time between these two events, minus 4 µs from the detector.

The empirical CDF (ECDF) of the experimental results is shown in Figure 7.4. We observe a delay of $t_{\text{init}} = 14.4\,\mu\text{s}$ on average, which is mainly caused by the firmware latency. For the summary of delay components, the RX/TX turnaround from the daughterboard accounts for 1 µs, a small number of FPGA cycles is spent in the TX DSP pipeline, the rest (and the deviations) is caused by the interrupt handling and the additional processing in the firmware.

We proceed by evaluating the system response time t_{response} achieved by WiFire. As a reference value, if we want to read the complete payload and perform the classification and selective interference during the checksum at the end of the packet, this time must be less than 64 µs. The results of our evaluation are shown in Figure 7.3. For the firmware-based rule checker, the delay depends on the number and complexity of rules to be checked. The response time for a representative rule set is 160 µs, or 5 byte before the end of the packet. For the FPGA implementation, the execution time is much shorter: even with a complex rule set, the response time does not exceed 39 µs. This enables rules using the complete packet payload and still ensures the destruction of the packet.

7.8. Summary

The guardian implementation allows real-time detection of malicious packets with a high accuracy during their transmission and a reliable destruction before the packet may arrive at a receiver. The results are summarized in Tables 7.1 and 7.2. The system is able to classify and destroy 99.9 % of the packets, even if the classification depends on the last byte in the payload, because the reaction time of the system (39 µs) is shorter than the duration of the CRC field (64 µs).

Blocking rule	Offset (bytes)	Maximum t_{response} (µs)	Firmware	FPGA
Start-of-Frame Delim.	5	864	$\surd$	$\surd$
Frame Control Field	7–8	768	$\surd$	$\surd$
Source Address	14–15	544	$\surd$	$\surd$
Payload byte #16	27	160	$\surd$	$\surd$
Last payload byte	30	64	×	$\surd$

Table 7.1.: Impact of overall reaction delay on feasible blocking rules. The symbol $\surd$ indicates that the guardian can use the respective blocking rule and still destroy the packet.

		Delay (µs)	
Parameter	Description	Firmware	FPGA
t_{decide}	Rule checker execution time (var.)	116	10
t_{init}	Duration decision–start transmitting	3	3
$t_{\text{interfere}}$	Interference duration for packet drop	26	26
$\boldsymbol{t_{\text{response}}}$	**Overall reaction delay**	**145**	**39**

Table 7.2.: Time parameters of our guardian implementation, indicating that packets can still be destroyed after observing a large part of their contents.

Part III.

Experimental Evaluation of WiFire

8. Jamming Evaluation

Contents

10.1. Additional Options for Rules 123
10.2. Technical Considerations 125
10.3. Non-technical Considerations 126
10.4. Using WiFire for Interference Research 127
10.5. Summary 131

8.1. Validation of the Collision Model

To show the validity and accuracy of our model presented in Chapter 4, we implemented and experimented with an application that is strongly dependent on physical layer characteristics, the reception of unsynchronized signals. We performed this experiment with two widely used commercial IEEE 802.15.4 receiver implementations (TI CC2420 and Atmel AT86RF230) to demonstrate that our results are receiver-independent. The results validate our claim that our model accurately captures the behavior of realistic receivers in the face of concurrent transmissions.

Here, we provide experimental evidence that our model accurately captures the behavior of existing receiver implementations. We focus our efforts on the reception of interfering signals because this topic is not well covered experimentally in the literature. We note that we also validated our analytical results with a simulation model based on the numerical integration of time-discrete signals, which confirmed the correctness of our model at the symbol and chip levels. The purpose of this section is to show that our simplifying assumptions, especially for the receiver model, are justified.

8.1.1. Experimental Setup

To perform this experiment, the requirements for the interferer differ from the scope of operation of Commercial Off-The-Shelf (COTS) devices. We need to *(i)* transmit arbitrary symbols on the physical layer, without restrictions like PHY headers, *(ii)* synchronize to ongoing transmissions with high accuracy, and *(iii)* schedule transmissions at a fine time granularity. To meet these requirements, we implemented a custom software-defined radio based experimental system.

Interferer Implementation

To this end, we modified WiFire to recover the timing of the other signal and send arbitrary IEEE 802.15.4 symbols at controlled time offsets. Because of its implementation in the USRP2's FPGA, the system is able to tune the start of transmission with a granularity of 10 ns and send arbitrary waveforms.

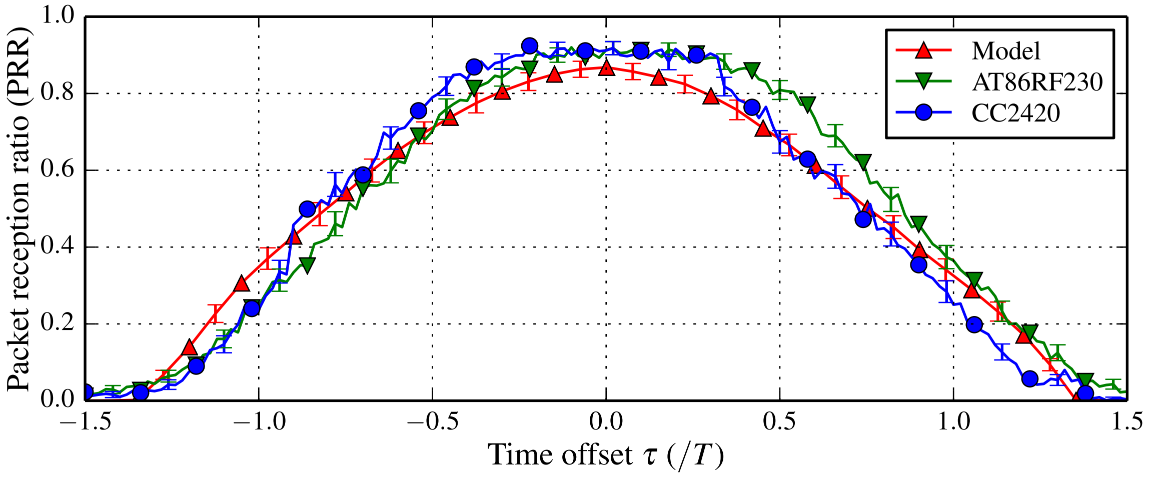
Experimental Methodology

In our experiments, we consider three parties in the network: a standard-compliant receiver (we monitor the behavior of two implementations to test for hardware dependencies, Atmel AT86RF230 and TI CC2420), a synchronized sender $\mathcal{S}$ (a COTS RZ Raven USB), and the interferer $\mathcal{I}$ described above. The procedure is as follows: $\mathcal{S}$ sends a packet with PHY headers, MAC header, and 8 byte payload. $\mathcal{I}$ time-synchronizes with this signal and schedules the transmission of 8 different bytes at the beginning of the payload of $\mathcal{S}$. The receiver first synchronizes on $\mathcal{S}$ and receives its header, but experiences a collision in the payload bits. We note that the receivers do not attempt to correct bit errors, retransmissions are used for error recovery during normal operation. Damaged packets are simply detected using the checksum at the end and discarded in case of failure. For the experiments we reconfigured the devices so that all packets are recorded, even if the checksums did not match.

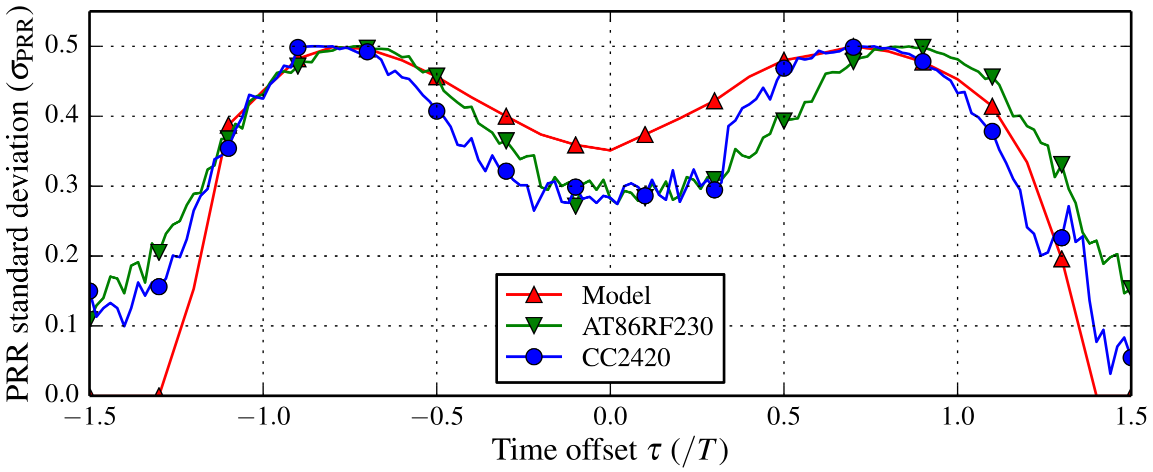
We chose values of τ in $(-1.5T; 1.5T)$ or ± 750 ns in steps of 10 ns; for each time offset τ, we sent 1,000 packets and analyzed the payload detected by the receiver. We derived the value of τ empirically, i.e., we chose the point with maximum PRR in the center as $\tau = 0$. We adjusted the transmit power of $\mathcal{I}$ to result in a SIR of -40 dB to be in the region of interest.

8.1.2. Experimental Results

We analyze our measurements using two metrics, packet reception ratio and symbol error rate.



(a) Comparison of packet reception ratios.



(b) Comparison of PRR standard deviations.

Figure 8.1.: Experimental results for two receivers in terms of packet reception (PRR) performance and PRR standard deviation compared to our model. Both receivers display a behavior that is well-described by the model.

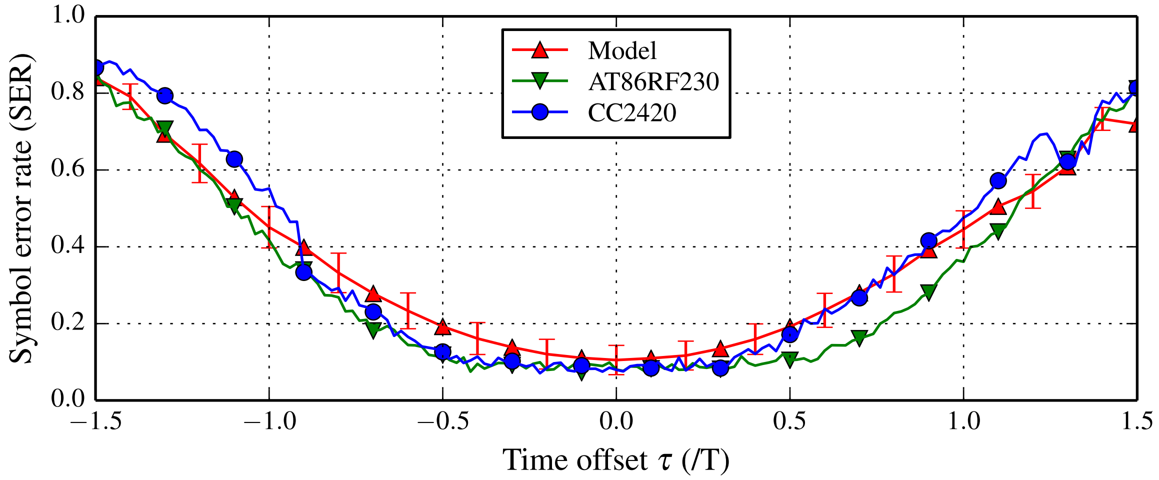
Packet reception ratio (PRR)

Based on the received packet data from the experiments, we derive the PRR as the number of packets with correct payload (of the interferer) divided by the total number of packets. In other words, we measure the empirical success probability for a message manipulation attack. The experimental results for the mean PRR of the two receivers are shown in Figure 8.1a. We observe a good fit with the predictions of our model to both receivers, Atmel AT86RF230 and TI CC2420. In the central region, the receivers show a slightly better ability to receive the interfering signal than predicted by our analytical model. The reason is that our model makes the assumption that no frequency offset is present and that the receiver does not try to resynchronize with a stronger signal. However, receivers must be able to tolerate frequency offsets of up to 100 kHz (IEEE Std 802.15.4 2006, Sec. 6.9.4) and thus track and possibly correct the phase during the packet reception process. Yet, as the results show, our assumptions still yield a good approximation of the real receiver behavior.

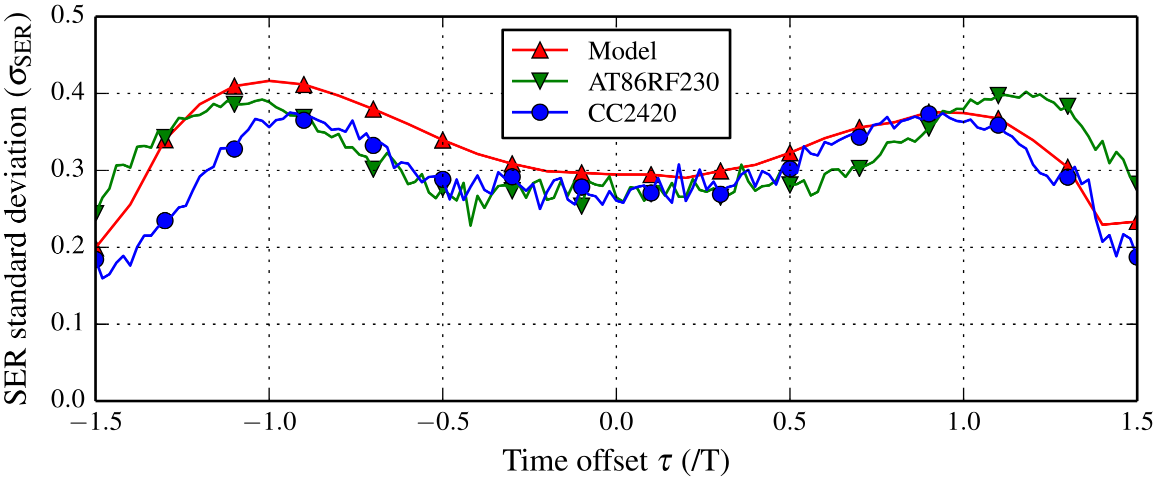
To further validate our model, we perform an analysis of the standard deviation of the measured PRR values (Figure 8.1b). In general, the second order statistics follow the non-trivial shape well. On closer inspection, we observe three regions in the graph. For $|\tau| < 0.5$, our model slightly overestimates the standard deviation; the reason is that the PRR performance of the COTS receivers is better than our model, leading to less variance. For $0.5 < |\tau| < 1.1$, the curves are close to each other. Finally, in the zone with $|\tau| > 1.1$, the model slightly underestimates the standard deviation, again because the real receivers perform better than the model predicts. Still, the model provides a good approximation of the behavior of widely used receivers for interfering signals under the assumption of random carrier phase offsets.

Symbol error rate (SER)

We derive the SER by summation of the number of symbol errors across the payload of all received packets for a given time offset τ, and divide this sum by the total number of payload symbols. This metric gives better insights into the causes for packet errors, and provides another validation for the capture zone. In Figure 8.2a, we observe that the fit is good for the symbol error rate as well, with a slightly better SER performance for the COTS receivers as expected. Considering the SER standard deviation (Figure 8.2b), we observe a similar behavior as in the PRR case, the predictions of the model and the measured results provide a good fit in both, curve shape and absolute values.



(a) Comparison of symbol error rates.



(b) Comparison of SER standard deviations.

Figure 8.2.: Comparison of experimentally measured symbol error rates and standard deviations, and the SER values predicted by our model.

8.1.3. Concurrent Transmissions Summary

Our results show that our analysis of packet collisions in IEEE 802.15.4 networks shows a good fit to the experimental results of several widely used receivers.

8.2. Experimental Evaluation of Jamming

With WiFire, we can deliberately produce collisions between the packet in transmission and a jamming signal. Depending on the error correction capabilities of the standard, only a small fraction of a packet must actually be destroyed to make the packet unrecoverable, resulting in a drop at the receiver. We are interested in the optimal waveform that offers a high chance to intercept a packet, and the minimum interference duration that introduces bit errors into the packet with high probability. We identify the factors that influence the jamming performance, and select the optimal jamming signal. In this section, we identify the causes of packet loss on the physical layer of IEEE 802.15.4, as well as which jamming signals and timings are consequently the most effective ones against such transmissions. The results are verified through systematic experiments in a WSN testbed with MICAz motes.

8.2.1. Effectiveness of Jamming Waveforms

Based on our analyzes in Chapters 4 and 5, we want to identify jamming waveforms that are the most effective against IEEE 802.15.4. By waveform, we refer to the shape of the RF signal transmitted on the channel, specified by a sequence of I/Q samples. We check the susceptibility to three different jamming waveforms that trigger the causes presented in the previous section: symbol, timing, and frame sync errors. The signals we consider are *(i)* wideband noise, *(ii)* a narrowband continuous wave (single-tone jamming), and *(iii)* IEEE 802.15.4 modulated signals with different content, such as random symbols, preambles or SFDs to interfere with the PHY packet reception process.

Experimental Setting

We conduct the experiments in a room with a surface area of 4 m × 3 m, with two MICAz motes programmed as sender and receiver placed at 2 m apart, and a USRP2 as the jammer in the same room. The USRP2 is equipped with an XCRV2450 board with a maximum transmit power of 100 mW (20 dBm), and 3 dBi omnidirectional antennas. The jamming waveforms are generated on a host PC using GNU Radio. We use constant jamming and deactivate the clear channel assessment functionality of the sender such that it transmits irrespective of the channel state to ensure that we only observe physical layer effects. We do not use reactive jamming at this point because this would introduce new uncertainties into the experiment, however, the results also apply to reactive jamming. We vary the transmission power of the jammer (denoted as *jammer gain*) and measure the resulting PRR at the receiver, i.e., packets that successfully passed the CRC check despite jamming. The main factor of successful jamming is a sufficiently high power level at the receiver, e.g., by using external amplification. Alternatively, we can say that the jammer has an operational range that is defined by the selected output power, and influenced by the propagation environment and transmission delay.

Jamming Waveforms

We concentrate on physical layer attacks against 802.15.4 instead of jamming approaches against MAC mechanisms (LAW et al., 2009; LIN and NOUBIR, 2005) such as attacking the clear channel assessment (CCA). We evaluated three candidate waveforms for their ability to prevent a packet reception: a continuous wave (CW) positioned at the center frequency of the channel, white noise with a bandwidth of 500 kHz around the center frequency, and random symbols spread and modulated as specified in IEEE 802.15.4. We evaluated their performance according to the required transmit power to achieve a packet reception ratio of 0 %. We generated the modulated signals using the UCLA ZigBee implementation (SCHMID, 2006).

Results for IEEE 802.15.4 Modulated Waveforms

We evaluated five patterns: random symbols, preamble (`0x00`), SFD (`0xA7`), synchronization header SHR (preamble+SFD), and SHR+PHR headers (preamble+SFD+length). Each of the sequences has a different effect on the receiver. Random symbols interfere with the symbol recognition and can therefore flip symbols (Figure 8.4). We expected the preamble or SFD symbols to interfere with the timing recovery, but these two waveforms are comparable to random symbols in their jamming efficiency. The reason is that the receiver locks onto stronger preambles (the *capture* effect), and that SFDs without preambles are not detected by the receiver because of lacking timing recovery.

Network degradations with weaker jamming transmissions are observed for the SHR and SHR+PHR waveforms. The receiver can lock onto such

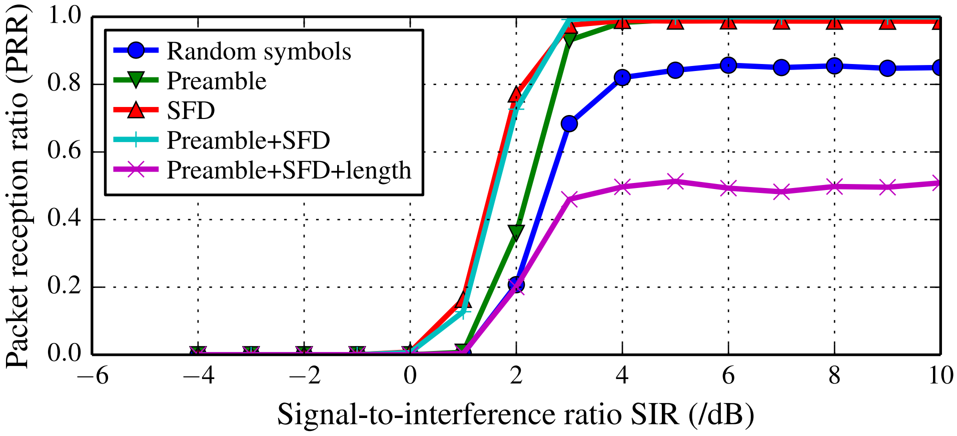


Figure 8.3.: Waveforms interfering with timing recovery and frame synchronization.

jamming signals even if they are weaker than the legitimate signal. Thus, even with a smaller jammer power a severe reduction in the PRR is possible as the receiver is busy decoding noise (see the comparison in Figure 8.3). This effect can be amplified further through the use of a valid length field after the SFD, forcing the receiver to stay longer in the reception state. For a proactive jammer this attack is attractive, because even weak signals at the receiver can still cause severe reductions in the PRR.

Results for Simple Jamming Waveforms

We used two different uncorrelated waveforms for jamming in addition to the modulated jamming, continuous (sine) waves and noise, see Figure 8.4.

Single-tone jamming. We used a constant signal that is modulated on the carrier, resulting in a continuous wave in passband. This very narrowband signal may be expected to perform badly as only a small portion of the IEEE 802.15.4 channel bandwidth is affected (see also Poisel (2011) for in-depth results on CW jamming). However, several effects cause a superior jamming efficiency in our experiments: in Figure 8.4, this waveform possesses a clear γ_{SIR} advantage over modulated jamming. First, this waveform interferes with timing recovery, the receiver detects the jamming signal as a second carrier signal, and the frequency mismatch makes a phase correction impossible. The second effect is that it has the largest signal amplitude of the tested waveforms; it provides more power per Hertz with a limited power

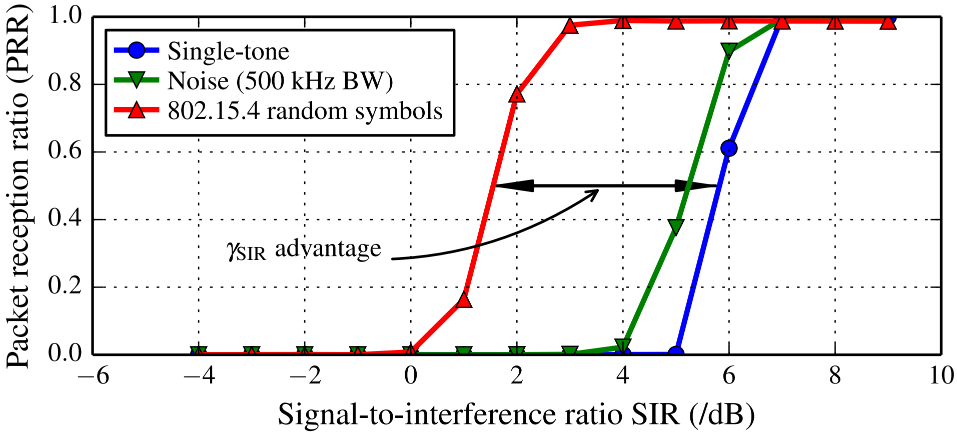


Figure 8.4.: Impact of waveforms that interfere with the symbol decision.

budget as the signal is more concentrated on the channel. This causes AGC to react faster, which results in chip misdetection on smaller power levels in comparison to other jamming waveforms. So despite its simplicity and ineffectiveness in theory, this waveform offers the best performance against COTS devices that are not hardened against jamming attacks.

Noise jamming. Wideband interference is always present in wireless communications, such that the receivers are specifically designed to withstand its influence. Its main effect is chip flipping that increases the likelihood of symbol misdetection. However, for a limited power budget (e.g., 20 dBm for the USRP2) the jamming signal's power is spread over a wider spectrum, depending on the bandwidth of the signal. This is the main factor why noise jamming has a limited efficiency in our tests; we achieved the best results with a BW of 500 kHz, yet it was always a few dB less efficient in comparison to single-tone jamming to achieve a PRR of 0 %.

Jamming Waveform Summary

The continuous wave offers the most energy-efficient way to interfere with reception against the sensor platform we use, with the additional benefits that it is easy to generate on the SDR and that co-existing technologies such as WLAN filter out this type of interference relatively well, hence their operation is not disturbed by the wireless firewall.

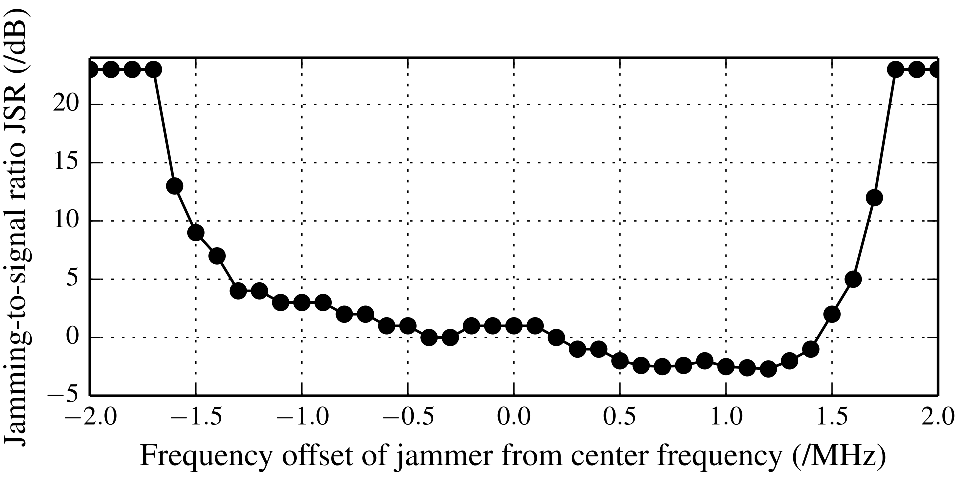


Figure 8.5.: Impact of relative position of the tone jamming signals in the channel. Relative jamming power required to ensure a PRR of 0 % for the receiver.

8.2.2. Effect of the Center Frequency on Jamming Performance

The relative position of the tone in the channel is also an important factor. We experimented with different offset values from the channel's center frequency, and the results are shown in Figure 8.5.[1] We observe that the channel filter of the MICAz transceiver has a width of 3 MHz, which cancels out-of-band interference. Additionally, a jamming signal directly on the center frequency is less effective in comparison to a 1 MHz frequency offset (on the corner frequency of the modulation), which complies with results in the literature (Poisel, 2011). Surprisingly, this effect is not symmetric. Negative frequency shifts have a 3 dB higher tolerance to CW jamming than positive shifts. We can only speculate why this is the case, but an artifact from either the USRP2's behavior (nonlinearities in the transmitter chain) or the receiver chip are potential explanations.

8.2.3. Minimum Interference Duration

We first evaluate how long WiFire must hit a packet to successfully destroy it. Using the CW waveform, we evaluated the minimum interference duration

---

[1] Note that the measurements result from a different experimental setup and the jammer gain values are not directly comparable to the other results.

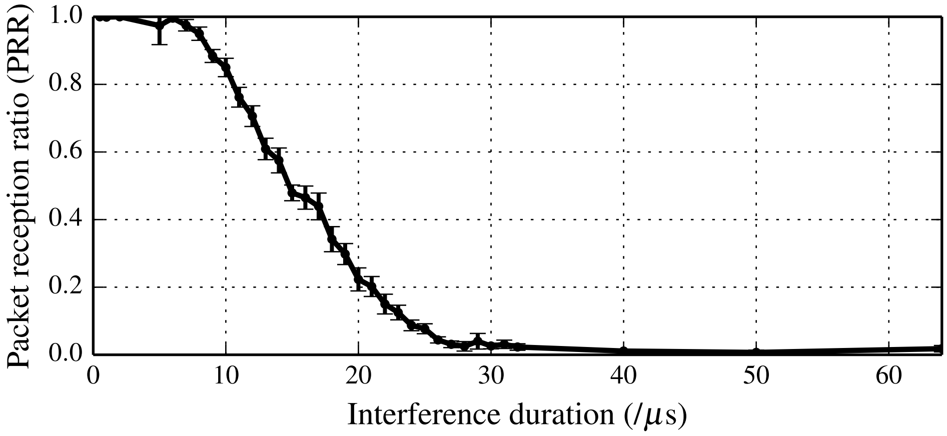


Figure 8.6.: Minimum interference duration for IEEE 802.15.4 radios, interfering with 26 µs of a packet transmission is sufficient to trigger packet drops at the receiver reliably.

t_{jam}^{min} on the channel to reliably destroy a packet with jamming.

Setup

Two MICAz motes are programmed as sender and receiver. To ensure that the jamming duration is the only factor in this performance measurement, the receiver is placed close to the jammer's TX antenna. Thus, WiFire and victim are placed in close vicinity such that the WiFire's interference is stronger than the signal of each packet. For each jam duration we consider, we transmit 100 packets and measure the PRR at the receiver, with 10 repetitions each. We use a single-tone, continuous wave as the jamming waveform.

Results

The results in Figure 8.6 show (with 95 % confidence intervals for the PRR means) that an interference duration of 26 µs is sufficient to destroy a packet. In theory, the destruction of a single symbol (16 µs) should be enough to cause a dropping probability of 93.75 % (there is still a 1 in 16 chance that the correct symbol is chosen), but due to symbol misalignments we require a slightly longer jamming duration to ensure interference with a complete symbol. This result has three implications: first, the energy cost of the attacker is high; while the attacker must transmit a complete packet to be successful (e.g., 1024 µs for a 32 byte packet), the wireless firewall invests 40 times less energy to successfully prevent the reception. Second, the interference duty cycle of the wireless firewall is very low, minimizing the effect on co-existing networks. This is a critical point for real-world deployments of WiFire. From the view of a single channel, WiFire's behavior is comparable to frequency hopping systems such as Bluetooth. In fact, Bluetooth Power Class 1 devices (IEEE 802.15.1(2005, Sec. 7.2)) use the same transmit power (100 mW) as WiFire and occupy a 2 MHz IEEE 802.15.4 channel for approximately 25 ms per second, which is comparable to the emissions of WiFire reacting to an attacker with maximum rate (1,000 packets/s). This shows that WiFire can effectively control the wireless channel while using very limited emissions, comparable to licensed devices. Third, we are able to observe large parts of a packet because t_{response} is small enough to observe the complete payload and still reliably destroy the packet.

8.2.4. Jamming Summary

Considering reactive jamming, IEEE 802.15.4 modulated symbols are not as effective, since the receiver is already locked on the transmission. Due to the design choices of the transceiver in the MICAz sensor motes, single-tone jamming proves more efficient for reactive jamming than actual IEEE 802.15.4 waveforms with a limited power budget. This waveform reliably jams transmissions of the sensor motes in our experiments, and it is easily generated in software. The most efficient placement of the tone is at 1 MHz above the center frequency of the channel. Our results show that a jamming duration of 26 µs is sufficient to cause a packet loss in IEEE 802.15.4. This very selective and precise jamming minimizes the impact on co-existing networks nearby.

9. WiFire System Evaluation

Contents

11.1. Summary **133**
11.2. Future Work **133**

9.1. Evaluation of the Protection Coefficient

In this chapter, we want to evaluate the spatial protection performance using controlled experiments in various environments. We focus on the brute force attacker. The main factors are the two key distances:

- the distance from attacker to a victim node d_{av}, which governs the attack signal power at the victims antenna, and
- the distance from guardian to victim d_{gv}, which governs the protection signal power at the victims antenna.

We derived the results in our protection analysis in Chapter 5.1. We have established, for the brute force attacker

$$d_{av} \leq \sqrt[\alpha]{\frac{P_a}{\gamma_{\text{SIR}} P_g}} d_{gv}.$$

This means that for a successful attack, the attacker has to position itself closer than ρd_{gv} to the victim to succeed, with the factor $\rho = \sqrt[\alpha]{\frac{P_a}{\gamma_{\text{SIR}} P_g}}$ being the protection coefficient. From the point of view of the protection system, we have to increase the value of ρ such that the attack condition is not met. In a numerical example in 5.10, we calculated a value of $\rho \approx 0.07$ for a realistic setting (i.e., the attacker must be 14.2 times closer than WiFire to succeed), however this applied under worst case assumptions and with a simplified fading model. The goal of this section is to evaluate the protection factor in realistic settings.

9.1.1. Experimental Setup and Methodology

This section describes the setup of this extended measurement campaign that was carried out over several days. We first describe the methodology that puts the distances as the primary parameters of the experiment, and enables us to analyze the effect of various distance combinations. Additionally, we are interested in the effects of realistic fading, which leads to the choice of several indoor and outdoor scenarios.

Setup

There are three parties that take part in the experiments: a group of ten sensor motes under attack, an attacker mote, and two guardian devices. The attacker and victim devices are placed in a straight line with a distance of 10 cm between devices, the attacker being in the middle of the line. This setup leads to pairs of victim devices with a distance of 10 cm, 20 cm, ..., 50 cm from the attacker. Stated differently, we vary d_{av} from 10 cm to 50 cm, which implies that the power of the attacker is affected by channel fading only by a very limited amount. The attack device is a MICAz sensor mote with a power output of 1 mW.

Methodology

The experiment is performed as follows: the attacker broadcasts 1,000 packets to all victims simultaneously for each position, which are logged by the victims individually on reception. This means that, in the unprotected case, all devices observe the same packet count, and the attack is a complete success. To provide protection, a group of two WiFire guardian devices is placed in the vicinity of the setup, with varying distances from 1 m to 30 m, and programmed to intercept all incoming attack packets with a power of 100 mW.

In terms of the protection factor ρ, this leads to a dynamic range of this parameter in the experiment from $\rho = 0.003$ for the combination $d_{av} = 10\,\text{cm}$ and $d_{gv} = 30\,\text{m}$, up to a value of $\rho = 0.5$ for the combination of $d_{av} = 0.5\,\text{m}$ and $d_{gv} = 1\,\text{m}$.

Locations used

The variation of distances used in the experiments captures the effects of path loss, which is also covered in our protection analysis. However, in realistic environments the effects of short-term fading effects like multipath fading plays in important role as well, as we have seen in the experiments in the previous chapters. To evaluate the effects that these important transmission properties play, we repeat the same experiment in several realistic settings, namely (i) corridor, (ii), gymnasium, (iii) sports field, and (iv) underground tunnel; all on the premises of the TU Kaiserlautern. We briefly describe the environments and the predicted effects of fading to system performance.

Corridor: This setting constitutes a standard office environment with concrete walls, office doors, windows, etc. While we ensure that line of sight connections exist between all devices, this indoor scenario includes non-negligible multipath fading components with a large number of reflecting surfaces. Because multipath fading has an adverse effect to the performance of WiFire, this scenario is considered to be challenging both to the reception and destruction of the attack packets. This is also the scenario that was used for the experimental results in the other parts of this work.

Gymnasium: This large sports hall is another representative indoor scenario. It is a single large room with a high ceiling and large area. We chose this setup because the connections are predominantly line of sight, with only a limited number of reflecting surfaces and thus only a few multipath components. This scenario should lead to protection results that are more favorable to WiFire, and protection in this scenario should also behave closer to the analytical model.

Outdoor sports field: We perform measurements on a large outdoor area with very few reflecting surfaces in the vicinity. Consequently, this scenario should lead to the most favorable protection results, with predominant line of sight connections (and possibly two-ray ground reflections). The results should offer the most straightforward comparison to the analytical results.

Maintenance tunnels: Finally, we perform our experiments in a several meters deep underground tunnel. Tunnels are known to have very adverse effects to wireless transmissions because of multipath effects, and we use this scenario to evaluate protection performance in the worst case. We stress that a protection from WiFire is generally not necessary in these settings because the tunnels offer an extremely good physical protection from attacks; we are aiming to establish a better understanding of the adverse effects of multipath fading, and the limitations of the analytical model.

9.1.2. Experimental Results

We performed several large measurement campaigns to evaluate the spatial protection performance of WiFire.

Corridor Scenario

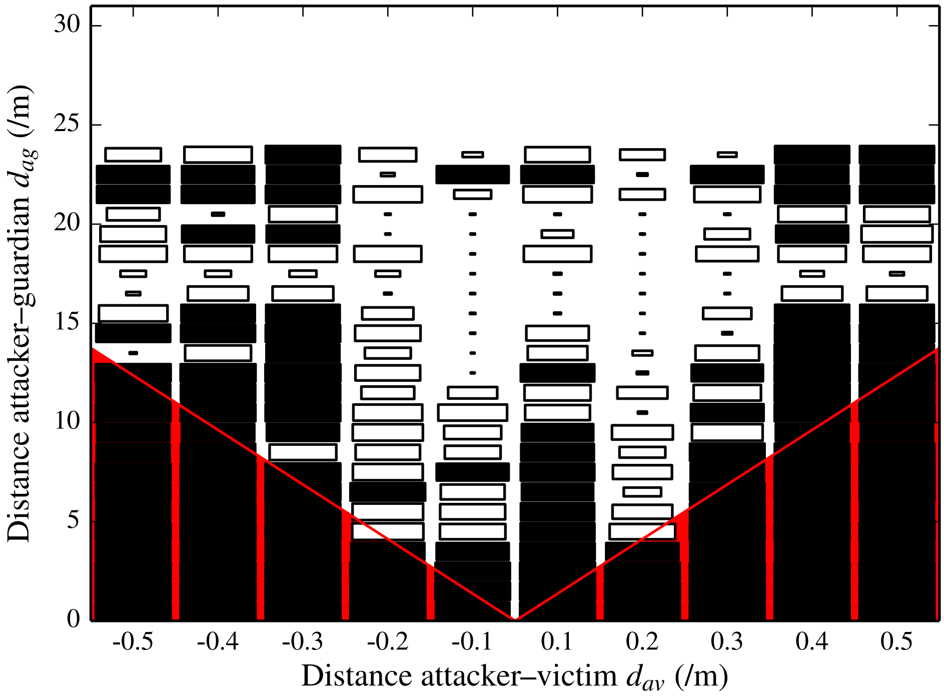
In the indoor corridor environment, we performed the experiment twice to get a feeling for stability of the experiments. The results from these two runs are shown in Figure 9.1. We observe that both runs yield very similar results, thus we only performed a single measurement campaign for the other scenarios.

Interpretation of the Hinton plots: The interpretation of these plots is as follows: on the x-axis, we show the distance of the victim nodes from the attacker in the center. For example, the attacker has two neighboring devices with a distance of 10 cm, both to the left and right of it. Overall, there are ten columns for each victim node, with pairs of nodes that have the same distance from the center. This also implies that the graphs should be symmetrical at the center position under a simple path loss model, because the distances in each direction are the same. Additionally, the graph should have a V-shape, because the attacker is affected by more severe path loss for larger d_{av} distances, resulting in a better protection performance of WiFire.

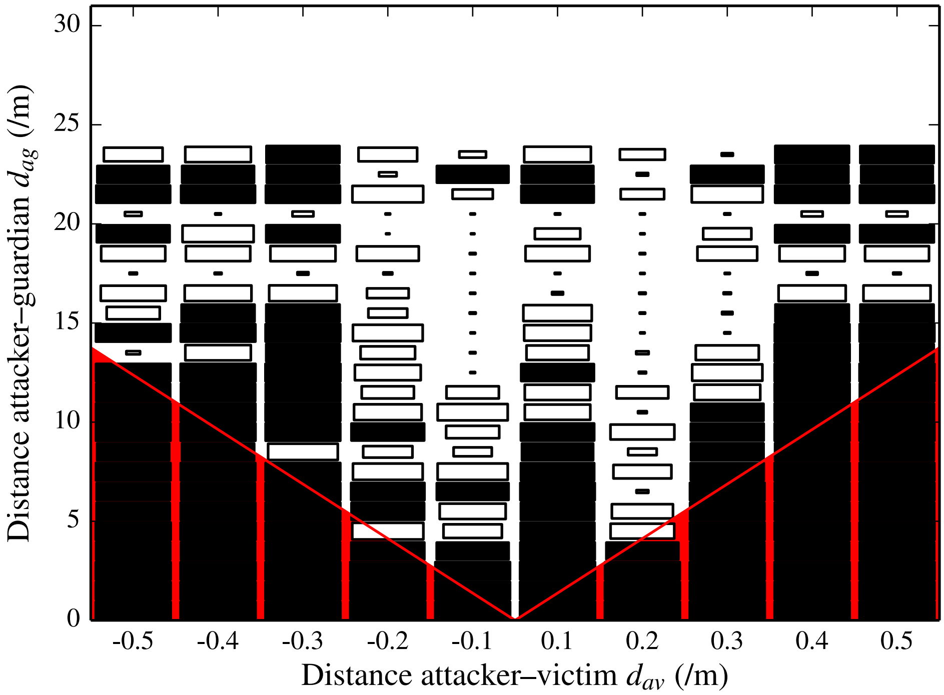
In each column, a number of boxes is shown. Each box shows the packet reception performance for a certain WiFire–victim distance d_{gv}. Figure 9.1 shows that d_{gv} was varied from 1 m to 23 m in this set of experiments. Finally, the surface area of each box indicates the packet reception rate for packets that were sent from the attacker. A large surface area indicates a small PRR of the victim, i.e., an effective protection from WiFire. For PRR values below 5 %, the surface area of the boxes is shown in black to support the visual impression of the areas of effective protection.

To analyze the protection coefficient ρ, we find the steepest line from the center that only has black boxes below. This line and the area below is shown in red in the figures. The intuition that we want to identify a set of d_{av}, d_{gv} pairs that always ensure a successful protection. Because of the linear relationship $d_{av} = \rho d_{gv}$ for successful protection, we can infer the minimal value of ρ that is supported by WiFire in this fashion by analyzing the slope.

Results: In the corridor scenario, we observe that a slope for ρ can indeed be found. The results follow the V-shape, and an area of good protection



(a) Run 1. The experimental protection coefficient is $\rho = 0.042$.



(b) Run 2. The experimental protection coefficient is $\rho = 0.042$.

Figure 9.1.: Disco corridor results.

can be identified when the WiFire distance d_{gv} is below 12 m. In this case, we derive for an attacker distance of 50 cm the protection coefficient of $\rho = 0.5\text{m}/12\text{m} = 0.0417$. When we compare this value to the predictions from the analysis ($\rho = 0.07$), we observe that the protection characteristics of WiFire are more favorable in realistic experiments. The reason is that the analysis only considers the worst case layout, with the attacker being as far away from WiFire as possible. In realistic environments, this is not always the case. However, with these experiments we established that the protection coefficient exists, raising the hope that this factor can be used to help in the operation of remote protection systems such as WiFire.

Looking closer at the results, we note that the protection coefficient could be even more favorable, with good protection results even up to 23 m for the larger values of d_{av}. However, there are a limited number of distance combinations that lead to imperfect protection, e.g., for $d_{av} = -0.5$ m and $d_{gv} = 13$ m, with a considerable PRR for the attacker. Surprisingly, the protection is again very favorable for $d_{gv} = 14$ m, an effect that is not captured in the analytical model. The explanation for this observation is that multipath fading has an adverse effect on the performance of WiFire for certain spatial constellations that are not only governed by the distances between devices. An illustrative example is the bad protection performance at the distances of 17 m and 20 m. While the protection is still favorable for larger WiFire distances, these gaps show that an attacker can find certain positions that allow an attack. Therefore the deployment of WiFire should be conservative in scenarios with multipath fading.

Gymnasium Scenario

We repeated the experiment in the large sports hall scenario. The results of this experiments are shown in Figure 9.2. We observe that the V-shape is even more pronounced, the reduced amount of multipath fading reduces the spots with weak protection results. However, a small number of weak constellations at 11 m to 15 m imply that a maximum distance of 13 m should be used in this scenario as well, resulting in the comparable of ρ as given in the corridor scenario.

Outdoor Sports Field

As expected, the outdoor field scenario results in the most regular results and consistent protection results for most constellations (the results are shown in Figure 9.3). Positions with reduced protection for the distances of 40 cm

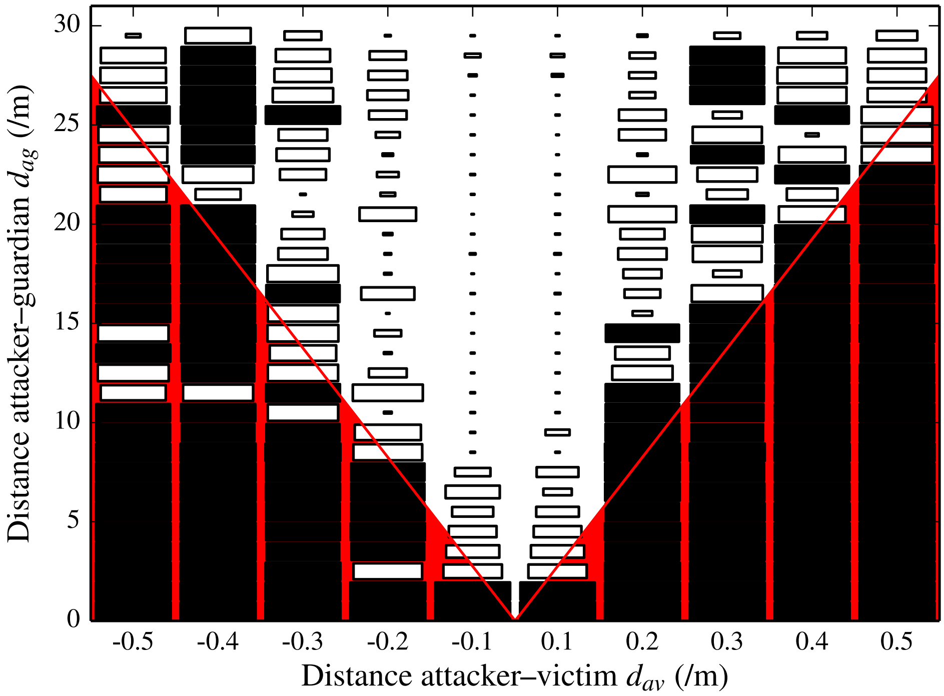


Figure 9.2.: Sports hall indoor. $\rho = 0.02$

and 50 cm are still the the PRR area of 90–95 %. These results validate that the analytical model is suitable to analyze the protection offered by WiFire, and that the protection coefficient can be used for both scenarios with and without small-scale fading effects.

Service Tunnels

In the final experiment, we evaluate the system performance in the face of the most severe multipath fading effects in the underground tunnel experiment. The results are shown in Figure 9.4. As expected, the results are comparable to the results in the corridor, but the number of constellations with insufficient protection is strongly increased. In consequence, WiFire must be deployed very closely to the protected devices, in a distance of approximately 4 m in this experiment.

The conclusion from this extreme example is that a protection from WiFire is still possible with the right deployment, but multipath fading has an adverse effect on the protection performance, i.e., increasing the distance an attacker can have from the victim devices.

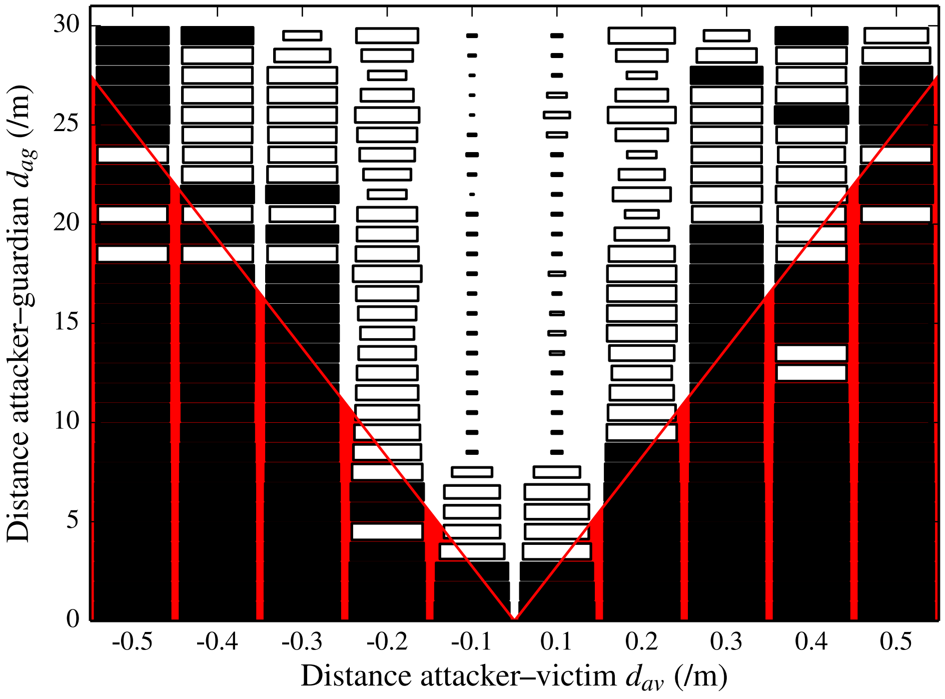


Figure 9.3.: Sports field outdoor. $\rho = 0.02$

9.1.3. Summary

In this section, we showed experimentally that a protection coefficient can indeed be found, that it is a good measure to gauge the increase of protection that WiFire offers. If a minimum possible distance of an attacker can be guaranteed, the concept of remote access control can be used to ensure the security of devices in the vicinity of guardian devices.

9.2. Selectivity Experiments

The experiments in this section show that WiFire operates selectively, i.e., communication from co-existing networks is not affected.

Scenario

As a scenario, we consider node capture and replication attacks in WSNs (Parno et al., 2005). While this problem is mainly treated as a key management issue in the literature (Chan et al., 2005; Parno et al., 2005), we show that compromised sensor motes can also be removed on the physical

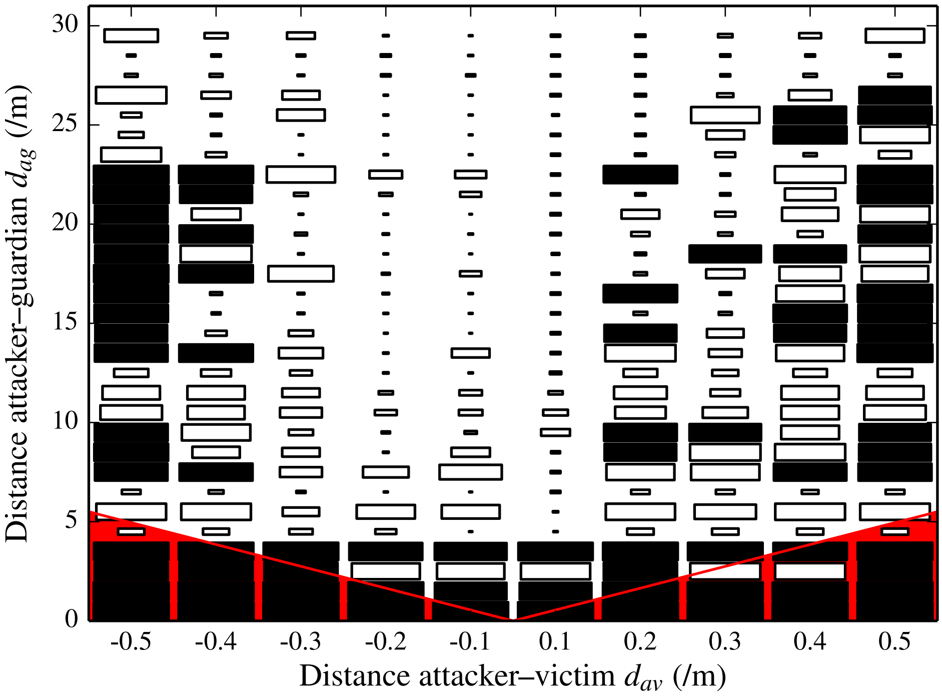


Figure 9.4.: Underground tunnels. $\rho = 0.1$

layer once they are identified. We refer to this as *instant* revocation because once the blocking rules are committed to the guardian, the motes' channel access is instantaneously blocked and they are thus disconnected from the network; it is not necessary to reliably distribute a revocation command in the (possibly Byzantine) network. From the guardian's perspective, the channel control policy is to detect packets from revoked nodes by their source addresses (sensors `0x1111` (I_1), `0x1112` (I_3), `0x1115` (I_5)) and network ID (`0xACAC`) and to destroy those packets:

```
wftables -A -m src --addr 0x1111 --pan 0xACAC -j DROP
wftables -A -m src --addr 0x1112 --pan 0xACAC -j DROP
wftables -A -m src --addr 0x1115 --pan 0xACAC -j DROP
```

In this experiment, six MICAz motes consecutively start transmitting with 10 packets/s. After 70 seconds, three nodes are revoked for 90 seconds, then allowed again for 20 seconds, and finally revoked for the rest of the experiment. We are interested in packets from revoked nodes able to reach the network (false negatives) and the impact of the guardian on the legitimate traffic (false positives).

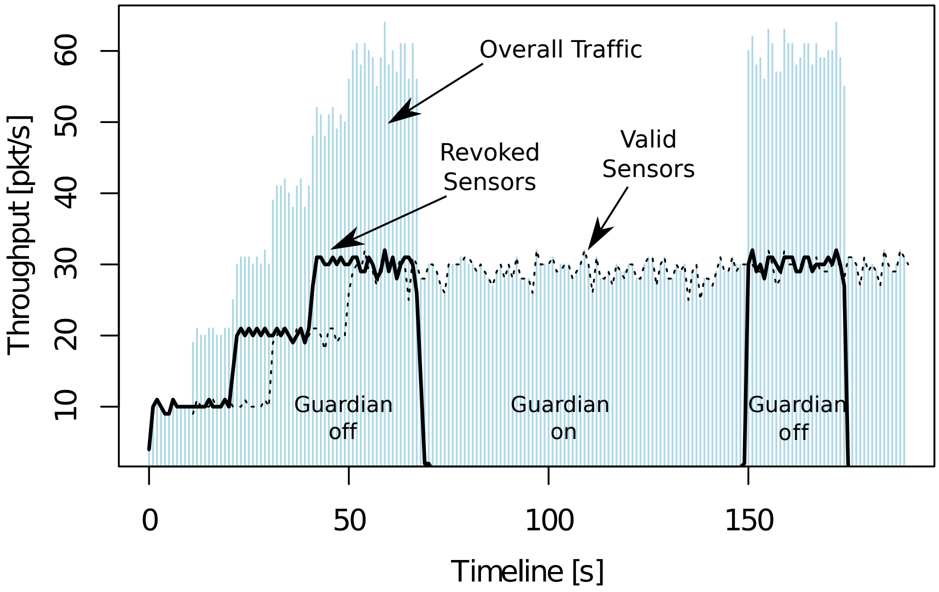


Figure 9.5.: Central node revocation of three MICAz motes enforced by WiFire in a network consisting of six motes. After 70 seconds, WiFire is configured to selectively block traffic transmitted from revoked devices (3 motes).

Results

The results are shown in Figure 9.5. The stepwise traffic increase is due to the consecutive start of the transmissions. The black solid line is the cumulative traffic of the nodes to be revoked, the dashed line shows the traffic of legitimate nodes, and the overall traffic is depicted by the bars in the background. As can be seen, the guardian immediately reacts by completely blocking the traffic from the revoked nodes. During the revocation phases, the amount of legitimate traffic equals the overall traffic, so there are no false positives. The number of false negatives is one packet at the beginning and at the end of revocation phases (due to the transition of the guardian's rule reconfiguration).

10. Discussion

This chapter discusses additional analyzers, and technical and non-technical considerations when applying a selective jamming system for wireless access control.

10.1. Additional Options for Rules

The current system implementation supports classification based on the content of the packet, such that certain types of attacks (spoofing and replay attacks) are hard to single out. In these cases, an attacker uses the same packet contents that a legitimate devices would use, and all content-based rules would block legitimate transmissions as well. Thus, the limitation to content-based policies limits the application scenarios that can be solved with the wireless firewall concept.

However, packet contents are not the only feature extractable from incoming packets: the physical characteristics of a signal are influenced by RF propagation phenomena and transmitter characteristics. There is a wide range of additional physical features that can increase the attack detection options of our guardian system. The physical characteristics of wireless communication could actually be exploited to provide interesting new opportunities for security policies, e.g., restricting communication based on the received signal strength, the angle of arrival of a transmission, the particular radio technology, or the physical device fingerprint. We briefly introduce a selection of features considered in the literature.

10.1.1. Positioning

Energy-based positioning. Several methods to infer a sender's position are proposed in the literature. The use of power level information such as RSS is used to position devices precisely (Bahl and Padmanabhan, 2000), even to localize them in large scenarios (Haeberlen et al., 2004). This enables location-aware applications such as *geo-fencing* (Sheth et al., 2009). The

Layers	Filtering criteria
application	content
transport	source/destination port, flags, protocol
network	source IP, destination IP, protocol
link	source/destination MAC, ESSID, frame type, security
physical	modulation, channel/frequency, RSS, angle of arrival, location, device fingerprints

Table 10.1.: Example criteria for additional wireless firewall rules.

guardians could then benefit from rules that use the sender location in their decision and detect spoofing attacks (Sheng et al., 2008).

Direction-based positioning. With antenna configurations such as directional antennas or antenna arrays, guardians can also gain capabilities of position-based classification. These methods use the direction of arrival to infer the position of a transmitter. Angle of arrival information has already been shown to be valuable for securing WLANs (Xiong and Jamieson, 2010).

10.1.2. Link Signatures

A more complex method using physical layer information to position devices is the use of link signatures (Patwari and Kasera, 2007). This method is able to discern the position of two devices with a large probability using spectral information and may be used to prevent spoofing and replay attacks.

10.1.3. Device Identification

This analysis method enables to uniquely identify transmitting devices (Danev, Zanetti, et al., 2012). For example, the concept of device fingerprints (Brik et al., 2008) uses imperfections in the TX chain of transmitters to associate packets to the transceiver hardware used. These features are stable and unique enough to identify devices even from the same production line. This technique can be used to whitelist trusted devices, blocking adversaries that cannot mimic the physical layer behavior of the devices.[1]

---

[1] While spoofing can still be achieved (Danev, Luecken, et al., 2010), the attacker must increase its effort significantly.

10.2. Technical Considerations

10.2.1. Additional Communication Technologies

Our experimental results show that effective remote protection using the wireless firewall is feasible. While our implementation presented here is specifically designed for 802.15.4, adaptations for different technologies are mainly a matter of exchanging the detector for different standards, and choosing an effective jamming waveform. Probably, the most crucial factor remains the reaction time. Nevertheless, when considering other technologies, the duration of an ACK frame for 802.11g (without legacy devices) is $t_{\text{packet}} \approx 30\,\mu\text{s}$, while our current prototype implementation reacts in $20\,\mu\text{s}$. This shows that even high-speed communication standards such as WLAN can be targeted with the system described here. First steps in this direction are made by Berger et al. (2014). The authors present a programmable jamming system based on modified off-the-shelf WLAN access points that supports a limited rule set for 802.11g and 802.11n technologies.

Still, it is true that other standards may contain anti-jamming measures that would impede the operation of a wireless firewall. However, in contrast to adversarial jamming attacks, the network nodes want the protection of WiFire. As a part of the administrated infrastructure we have access to network knowledge, i.e., we can assume to know secret sequences (e.g., spreading or hopping) and can easily counter spread spectrum techniques. With knowledge of the algorithms used, this can also overcome interference cancellation techniques recently explored (Halperin, Anderson, et al., 2008), so that such future networks can still benefit from the service of wireless firewalls. Overcoming more sophisticated anti-jamming techniques may be harder to implement but the basic argument still holds. This knowledge allows us to design tailored jamming signals that are the most effective against the devices in the network.

10.2.2. Optimal Guardian Deployment

Operating on the physical layer has benefits but also generates new challenges: we must aim to detect any packet that might arrive at a network node, and ensure that all violating packets are destroyed. These issues make the position and number of guardians important factors during deployment of the guardian system, and an optimization based on analytical models along the lines of Chapter 5.1 would be desirable. Methods that may be applied for this purpose are presented in information theory literature in the context of physical layer confidentiality in wireless networks Kim et al. (2012) and Sankararaman et al. (2012). These results suggest that by using a security perimeter around the network attacks can be mitigated completely because no network devices are then located in the attack range from all reachable attacker locations. Alternatively, a training phase or site survey can be helpful to support the optimization of the guardian deployment. As several WiFire instances can co-exist, we can effectively increase the protected area by replication, and to jam in a cooperative manner to increase the jamming effectiveness.

10.3. Non-technical Considerations

In this section, we consider non-technical aspects that can affect the operation of wireless firewall devices. We discuss the economic and legal aspects that must be considered when using controlled interference.

10.3.1. Economic Aspects

The WiFire devices are additional infrastructure that is deployed alongside the wireless network. While this offers several benefits (central control, on-demand security, or the possibility to "patch" legacy networks), it may raise the question of cost. The number of additional devices depends on the WSN deployment area and on the desired level of protection. Thus, the cost per device should be small; one option is to implement the system with COTS transceiver and micro-controller chips, possibly sacrificing flexibility. On the other hand, recent results also show that it is possible to build cheaper (approximately US$ 100) and more energy-efficient SDR platforms (Dutta, Kuo, et al., 2010). However, even in the current implementation with USRP2 software-defined radios, the price of approximately US$ 2,000 is still acceptable for a development platform.

10.3.2. Legal Aspects

The intentional generation of interference may raise the concern whether our guardians can be operated legally. In general, this question is not simple to answer because the rules governing spectrum access vary across countries and frequency bands. The U.S. Code of Federal Regulations (2010) mandates in §15.5 that "harmful interference," an emission that "obstructs or repeatedly interrupts a radiocommunications service operating in accordance with

[Chapter 15]," is forbidden. However, as we limit the interference to adversarial packets by analyzing and deciding on a per-packet basis, we might argue that no "service" is interrupted.

Regarding unintentional interference with co-existing networks, we point out that the guardian accesses the channel scarcely. The guardian emits a 26 µs signal per packet and is silent for the rest of the time (e.g., 998 µs for 32 byte packets). From the view of a single channel, such a behavior is also observed for frequency hopping systems such as Bluetooth. In fact, Bluetooth Power Class 1 devices (IEEE 802.15.1 (2005, Sec. 7.2)) use the same transmit power (100 mW) as our guardians and occupy a 2 MHz IEEE 802.15.4 channel for approximately 25 ms per second, which is comparable to the emissions of a guardian reacting to an attacker with maximum rate (1000 packets/s). This also shows that attacking the guardian infrastructure itself (by deliberately triggering interference) leads only to a limited channel occupancy. In addition, as we consider the operation in the 2.4 GHz band, co-existing devices such as IEEE 802.11 receivers commonly filter out the simple sinusoidal waveform we chose. Thus, while a comprehensive discussion of the legal aspects of RF interference is out of scope for this paper (as well as out of our expertise), we observe that the selective and efficient operation of the guardians effectively limits interference, and finally remark that sometimes legislation follows technical innovation.

10.4. Using WiFire for Interference Research

While the concept of protecting sensor networks with the guardian system is our main focus, we note that the generation of selective interference can also be useful for research on the effects of interference on network performance, allowing to perform repeatable experiments with real hardware. Related work in this area uses sensor mote hardware for interference generation, which limits the capabilities of such interference generation systems (BOANO, VOIGT, et al., 2011; Z. HE and VOIGT, 2011). Using the selective and protocol-aware interference generation capabilities of WiFire, a more fine-grained control over interference can be achieved. For example,WiFire can be deployed in a wireless testbed to generate arbitrary interference patterns based on packet content. In this spirit, the guardians may enforce that all ACK packets of a chosen device are lost, or 10 % of all network traffic is affected by microwave/WLAN/Bluetooth-like interference following a bursty pattern.

10.4.1. Interference in Experiments

The performance of a wireless network under interference is hard to predict for several reasons. First, a packet may still survive interference depending on the circumstances: the positions of sender, receiver, and interferer; the propagation environment; the resistance of the receiver against the particular type of interference; or the use of coding to recover damaged packets (e.g., (LIANG et al., 2010)). Second, the interference may exhibit patterns that harm network performance disproportionately, e.g., by destroying a packet and all corresponding retransmissions. Analytical treatment or simulations of network protocols often require simplifying assumptions to keep the evaluation treatable. For example, instead of complex channel models that are required to capture physical wave interactions in realistic environments, only simple fading models such as path distance or log-normal shadowing are often used. Another example is that the concurrent use of different medium access protocols may break the assumption of independence of interference events, which is often assumed to keep the analysis simple.

Because of these issues real-world experiments are widely used to explore the performance of protocols in realistic settings. The goal is to show that the protocol performs as intended in a representative environment. Still it is very challenging to reproduce experimental results in other environments (and oftentimes even in the same environment). To increase comparability and repeatability, researchers started to build and share testbeds where experiments can be performed under stable conditions. Yet, even in these controlled environments, interference is still problematic in experimentation, especially its repeatable and reliable generation. A common approach to interference generation is to deploy COTS devices in the testbed (e.g., using WLAN access points or sensor motes) and program them to send packets at random times or with a fixed rate. While this approach yields insights in the performance of a protocol in crowded settings, it is neither realistic nor repeatable (BOANO, Z. HE, et al., 2009). Another approach is to record interference patterns and replay them during the experiment. For example, BOANO, VOIGT, et al. (2011) show that sensor motes may be programmed to precisely capture and generate interference in a sensor network testbed.

Despite these efforts, the effects of interference are still a “black box” to experimenters. Even if exactly the same interference pattern is reproduced, the response of the network may deteriorate from its previous behavior because of small timing differences or internal system effects, leading to differing execution traces and a different performance. Oftentimes the response of the network is the parameter of interest (e.g., the packet reception ratio), not the duty cycle of the interferers. What we need is protocol-aware interference that also takes the state of the network into consideration, allowing direct control of these parameters.

10.4.2. Interference Scripting

The idea is to deploy dedicated interferers in the testbed that are programmed and controlled individually from a central point by adapting WiFire's architecture. Each device monitors the channel, detects packets, and generates interference in response to packet and timing events according to a supplied script. We briefly describe the necessary changes to support scripting for a wide range of application scenarios.

Scripting

In the existing system, we use a classification system similar to firewall rules. While this concept is already sufficient for some realistic interference scenarios, there are more components required for a fully featured experimental system. The vision is that these components can be combined to define arbitrary interference patterns in a script, using control structures like loops and conditions. The script ties together the following conditions to select packets for interference:

Packet content. Access to the packet content is an important step to protocol-awareness; it allows to restrict interference to packets with a chosen source/destination, to packet types, or bits in the payload. The real-time demodulation of WiFire allows this access to the content 4 µs after the corresponding physical layer symbol was on the air.

Timing. To support pre-recorded interference patterns or to occupy the channel for measuring the performance of carrier-sense based protocols, the system must also support scheduled interference. The USRP2 supports a timing precision of 10 ns and transmission timestamps that allow this mode of operation.

Randomness. Because blocking all packets of a kind is not a realistic interference scenario (when no adversarial setting is considered), we need a way to define interference patterns that follow random distributions of choice, e.g., 30 % of the packets with bursty interference. This feature requires the implementation of random sources with distributions of interest.

Protocol state. State information enables interference decisions with memory. Taking a trace of packets into consideration, this enables *white box* testing of wireless protocols, triggering interference only when a sequence of packets was previously detected. For example, a node may loose its connection to surrounding nodes each time it successfully associates with the network.

Interference waveforms. To generate interference patterns using multiple communication standards, or to mimic unintentional interferers such as microwave ovens, we need a way to choose interfering waveforms on a per-packet basis. The USRP2 supports arbitrary sequences of samples as digital representation of waveforms, allowing to store or compute them on-the-fly.

10.4.3. Application Examples

Next, we describe some possible applications of interference scripting.

Protocol-aware interference. With our approach, it is possible to define fine-grained interference patterns, e.g., targeting specific MAC layer packets such as ACKs to debug the interaction between application and OS layer software (this approach was used in related work to discover race conditions in the Contiki operating system (Z. He and Voigt, 2011)). However, in contrast to that work, our system does not rely on timing information, such that we can decide if interference is required and start it before the packet is over. An application of this is to target neighborhood discovery messages (used in routing protocols) to observe the behavior of the network in such adverse conditions.

Virtual topologies. By selectively interfering with packets based on their header addresses, we can define and enforce virtual topologies by blocking a subset of neighbors, or making links between nodes directional. This increases the control of the experimenter over the topology, and adds flexibility to an existing testbed. For example, it can alleviate the need of physically rearranging devices in the network, or help to evaluate the behavior of protocols in changing environments rapidly.

Arbitrary loss processes. Targeting packets directly allows to choose which random distribution the packet loss process should follow. This allows to compare the performance of protocols under the same conditions, e.g., using 70 % packet loss with interference bursts. Even if the protocols are operating differently, e.g., using different medium access strategies, we can control the interference patterns precisely because we are not relying on timing but targeting packets directly. This is also important when comparing results to simulations because the same random distributions can be used. This enables direct comparison between simulated results and observations from the real system.

Arbitrary waveforms. Emulating different devices in the vicinity (microwave ovens, baby phones, WLAN or ZigBee devices) is also an important application, which can be combined with the previous strategies to choose the interfering shape for each packet individually.

10.5. Summary

We noted that the scope of wireless firewalls can be extended in several dimensions. By applying new types of rules that codify the physical layer characteristics of trusted and untrusted communications, the decision process can integrate these new features and thus allow the definition of attacks that cannot be prevented when looking at the packet content alone, such as replay attacks. The consideration of IEEE 802.15.4 in this thesis followed the insight that the physical layer can readily be analyzed and implemented, and a similar approach to MAC layer protection would have been much more difficult. Still, there exists a large number of interesting applications for the wireless firewall concept in other areas, such as local and wide area data networks or cellular networks. Finally, the system that is described in this thesis can also be applied to other areas in the context of wireless research, e.g., to support experiments that explore the effects of interference to network performance.

11. Conclusion

11.1. Summary

In this thesis, we presented the concept of a firewall for wireless networks, a system that provides centralized channel access control to protect devices that cannot fend for themselves. By operating on the physical layer, it is possible to offer this protection transparently, i.e., the guarded devices can continue to operate without any notion that they are being protected. This means that no additional, energy-consuming security protocols are necessary, and that the wireless firewall is a passive device that only monitors the wireless channel when no attack is taking place. This is in contrast to conventional security mechanisms, which put an additional burden to a network constantly. Due to the selective policy enforcement of the wireless firewall, the security policy can be adapted easily to protect against new threats, or to accommodate for changes in the network, e.g., in the network topology. We analyzed the effects of RF interference on the reception process and showed that only a limited amount of interference is required to effectively block wireless transmissions, and that even simple waveforms achieve a good jamming success rate. Finally, the design, implementation, and system evaluation of WiFire substantiated the claim that a remote protection using RF interference is feasible, even in challenging indoor scenarios.

11.2. Future Work

The research on this novel concept is only at its beginning, there are many issues that must be addressed before we will see wireless firewalls in the wild. These issues range from a more detailed understanding of the protection properties of wireless firewalls, and a subsequent deployment strategy of guardian devices in challenging environments, e.g., indoor offices with co-existing WLAN networks, moving people and other variable RF attenuators, and severe multipath fading effects. Another important issue is to identify scenarios where access control on the physical layer is really necessary and cannot be performed equally well or even better. While there are applications where the wireless firewall is a good contender, e.g., restricting WLAN communication of prison inmates while keeping up your own connectivity or a mobile jamming system that prevents the detonation of roadside bombs, the applications of wireless firewalls are certainly much narrower than for wired firewalls. This thesis only provided the first steps on the way to access control on the physical layer.

A. Pulse Integration

A central equation for deriving the influence of individual bits on the demodulator output is the integration of the superposition of time shifted unit pulses $\Pi(t)$ (defined in equation (4.4)). This is especially important because of signal time offsets τ that shift the pulses relative to the integration interval. Situations that arise are shown in Figure A.1.

To this end, we first derive the general result to the integration over one bit interval k for arbitrary, integrable functions $f(t)$. We consider two variants, the integration of in-phase bits, and the special case of integrating quadrature-phase bits in the bounds of I-bits (which happens when Q-bits leak into the I-phase), i.e.,

$$
\begin{aligned}
S_k^I(f) &= \int_{(2k-1)T}^{(2k+1)T} b_I(t-\tau)\, f(t)\, dt \\
S_k^Q(f) &= \int_{(2k-1)T}^{(2k+1)T} b_Q(t-\tau)\, f(t)\, dt.
\end{aligned}
$$

Our approach is to split each equation into two parts where the unit pulse is the constant 1 function to simplify the equations. Since only one pulse is active at any point in time, such splitting is possible.

Integrating Bit Pulses During the I Integration Interval

Integration of I-bits

To perform the integration, we first derive the two indices that have active pulses during the integration interval. The shift introduced by τ lead to the two new bits with indices $k' = k - \left\lfloor \frac{\tau}{2T} \right\rfloor$ and $k' - 1$. The remaining time offset inside the selected bits is $\underline{\tau} = \tau - 2k_\tau T$, i.e., each of the two bits is active for the time interval $\underline{\tau}$ and $2T - \underline{\tau}$, respectively. Because of this definition, the values of $\underline{\tau}$ are restricted to the interval $[0, 2T)$—negative values would activate previous bits, which is prevented by the floor operation.

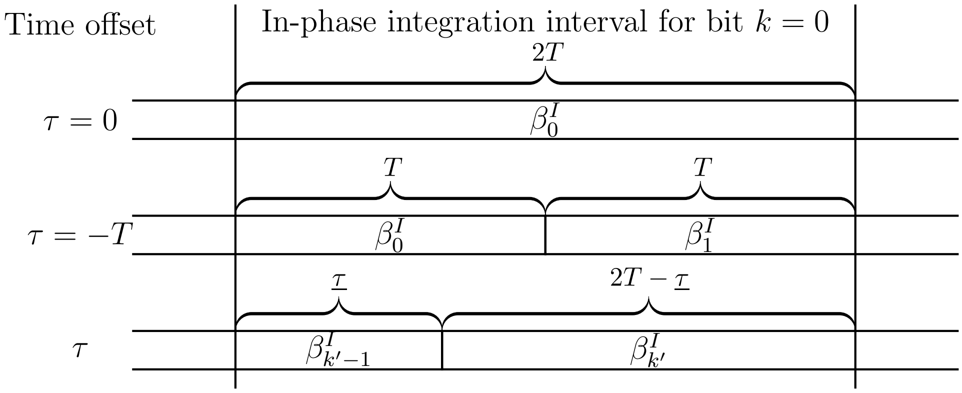


Figure A.1.: Examples of active bits in the integration interval for the I-bit k. For $\tau = 0$, the only active bit in the integration interval is β_0^I. When the signal starts half a bit-length too early ($\tau = -T$), there are two bits β_0^I and β_1^I that contribute equally to the bit decision, both are active for a duration of T. In the general case of a time offset τ, there are two active bits with indices $\beta_{k'-1}^I$ and $\beta_{k'}^I$, with an active time duration of $\underline{\tau}$ and $2T - \underline{\tau}$, respectively.

For the in-phase component, we derive

$$\begin{aligned}
&S_k^I\left(f\right)\\
&= \int_{(2k-1)T}^{(2k+1)T} b_I\left(t - \tau\right) f\left(t\right) dt\\
&= \int_{(2k-1)T}^{(2k+1)T} b_I\left(t - 2k_\tau T - \underline{\tau}\right) f\left(t\right) dt\\
&= \int_{(2k-1)T}^{(2k+1)T} \sum_{k=-\infty}^{\infty} \beta_k^I \Pi\left(\frac{t - \underline{\tau} - \left(k + k_\tau\right) 2T}{2T}\right) f\left(t\right) dt
\end{aligned}$$

Re-labeling the bit indices k to k' (note: positive time shifts lead to negative index shifts)

$$
\begin{aligned}
&= \int_{(2k-1)T}^{(2k+1)T} \left(\beta^I_{k-1} \Pi \left(\frac{t - \underline{\tau} - (k-1)\,2T}{2T} \right) + \beta^I_k \Pi \left(\frac{t - \underline{\tau} - 2kT}{2T} \right) \right) f(t)\, dt \\
&= \beta^I_{k-1} \int_{(2k-1)T}^{(2k+1)T} \Pi \left(\frac{t - \underline{\tau} - (k-1)\,2T}{2T} \right) f(t)\, dt + \beta^I_k \int_{(2k-1)T}^{(2k+1)T} \Pi \left(\frac{t - \underline{\tau} - 2kT}{2T} \right) f(t)\, dt
\end{aligned}
$$

Use the fact that the shifted pulses are zero during parts of the integration interval

$$
= \beta^I_{k-1} \int_{(2k-1)T}^{(2k-1)T+\underline{\tau}} \Pi \left(\frac{t - \underline{\tau} - (k-1)\,2T}{2T} \right) f(t)\, dt + \beta^I_k \int_{(2k-1)T+\underline{\tau}}^{(2k+1)T} \Pi \left(\frac{t - \underline{\tau} - 2kT}{2T} \right) f(t)\, dt
$$

The Π pulses are constant 1 in the new integration intervals

$$
\begin{aligned}
&= \beta^I_{k-1} \int_{(2k-1)T}^{(2k-1)T+\underline{\tau}} f(t)\, dt + \beta^I_k \int_{(2k-1)T+\underline{\tau}}^{(2k+1)T} f(t)\, dt \\
&= \beta^I_{k-1} \Big[F(t) \Big]_{2kT-T}^{2kT-T+\underline{\tau}} + \beta^I_k \Big[F(t) \Big]_{2kT-T+\underline{\tau}}^{2kT+T}
\end{aligned}
$$

If the function to integrate is the constant 1 function ($f(t) = 1$), then we derive

$$
S^I_k(1) = \underline{\tau}\beta^I_{k-1} + (2T - \underline{\tau})\,\beta^I_k \qquad \text{(A.1)}
$$

Integration of Q-bits

When Q bits leak into the in-phase, we have the consider the additional shift of T due to the staggering of bits in the MSK modulation. We provide the derivation of this special case here. First, we substitute the timing offset τ with $\tau^Q = \tau + T$ to accommodate of the staggering. Second, the bit indices must be re-adjusted because of the shift; the new index is denoted by $k^{Q\prime} = k - \lfloor (\tau + T)/2T \rfloor$. For the case of the constant 1 function, we derive then

$$
S^Q_k(1) = \underline{\tau^Q}\beta^Q_{k^{Q\prime}-1} + \left(2T - \underline{\tau^Q}\right)\beta^Q_{k^{Q\prime}}. \qquad \text{(A.2)}
$$

Deriving Special Cases: $S_k^I\left(\cos 2\omega_p t\right)$ and $S_k^Q\left(\cos 2\omega_p t\right)$

Integration of I-bits

We derive the result of bit pulse integration for this special case.

$$S_k^I\left(\cos 2\omega_p t\right)$$

$$= \int_{(2k-1)T}^{(2k+1)T} b_I\left(t-\tau\right)\cos 2\omega_p t \, dt$$

$$= \beta_{k-1}^I \left[\frac{1}{2\omega_p}\sin 2\omega_p t\right]_{(2k-1)T}^{(2k-1)T+\underline{\tau}} + \beta_k^I \left[\frac{1}{2\omega_p}\sin 2\omega_p t\right]_{(2k-1)T+\underline{\tau}}^{(2k+1)T}$$

$$= \frac{\beta_{k-1}^I}{2\omega_p}\left[\sin 2\omega_p t\right]_{(2k-1)T}^{(2k-1)T+\underline{\tau}} + \frac{\beta_k^I}{2\omega_p}\left[\sin 2\omega_p t\right]_{(2k-1)T+\underline{\tau}}^{(2k+1)T}$$

Performing the integration results in (we denote $\omega_p\underline{\tau} = \underline{\varphi_p}$):

$$= \frac{\beta_{k-1}^I}{2\omega_p}\left[\sin\left(\left(2k-1\right)\pi + 2\underline{\varphi_p}\right) - \sin\left(\left(2k-1\right)\pi\right)\right]$$

$$+ \frac{\beta_k^I}{2\omega_p}\left[\sin\left(\left(2k+1\right)\pi\right) - \sin\left(\left(2k-1\right)\pi + 2\underline{\varphi_p}\right)\right]$$

$$= \frac{\beta_{k-1}^I}{2\omega_p}\left[\sin\left(-\pi + 2\underline{\varphi_p}\right) - \sin\left(-\pi\right)\right] + \frac{\beta_k^I}{2\omega_p}\left(\sin\pi + \sin 2\underline{\varphi_p}\right)$$

$$= -\frac{\beta_{k-1}^I}{2\omega_p}\sin 2\underline{\varphi_p} + \frac{\beta_k^I}{2\omega_p}\sin 2\underline{\varphi_p}$$

$$= \sin 2\underline{\varphi_p}\left(-\frac{\beta_{k-1}^I}{2\omega_p} + \frac{\beta_k^I}{2\omega_p}\right)$$

Using $\sin 2\underline{\varphi_p} = \sin\left(2\omega_p\left(\tau - 2k_\tau T\right)\right) = \sin\left(2\varphi_p - 2k_\tau\pi\right) = \sin 2\varphi_p$

$$= -\frac{1}{2\omega_p}\sin 2\varphi_p\left(\beta_{k-1}^I - \beta_k^I\right)$$

The overall result is

$$S_k^I\left(\cos 2\omega_p t\right) = -\frac{1}{2\omega_p}\sin 2\varphi_p\left(\beta_{k-1}^I - \beta_k^I\right) \tag{A.3}$$

Integration of Q-bits

In this case, the use of τ^Q leads to a different phase shift $\varphi_p^Q = \omega_p \left(\tau + T\right) = \omega_p \tau + \frac{\pi T}{2T} = \varphi_p + \frac{\pi}{2}$ that leads to changes in the integration. Using the following two simplifications the derivation can be performed analogously to the previous subsection.

$$\sin 2\underline{\varphi_p^Q} = \sin 2\omega_p \underline{\tau^Q} = \sin\left(2\frac{\pi}{2T}\left(\tau^Q - 2k_\tau^Q T\right)\right) = \sin\left(\frac{\tau^Q \pi}{T} - 2k_\tau^Q \pi\right) = \sin 2\varphi_p^Q$$

and

$$\sin 2\varphi_p^Q = \sin\left(2\varphi_p + \pi\right) = -\sin 2\varphi_p$$

The overall result is

$$S_k^Q\left(\cos 2\omega_p t\right) = \frac{1}{2\omega_p}\sin 2\varphi_p\left(\beta_{kQ'-1}^Q - \beta_{kQ'}^Q\right) \tag{A.4}$$

Deriving Special Cases: $S_k^I\left(\sin 2\omega_p t\right)$ and $S_k^Q\left(\sin 2\omega_p t\right)$

Integration of I-bits

We derive the result of bit pulse integration for this special case.

$$\begin{aligned}
&S_k^I\left(\sin 2\omega_p t\right) \\
&= \int_{(2k-1)T}^{(2k+1)T} b_I\left(t-\tau\right)\sin 2\omega_p t \, dt \\
&= \beta_{k-1}^I\left[-\frac{1}{2\omega_p}\cos 2\omega_p t\right]_{(2k-1)T}^{(2k-1)T+\underline{\tau}} + \beta_k^I\left[-\frac{1}{2\omega_p}\cos 2\omega_p t\right]_{(2k-1)T+\underline{\tau}}^{(2k+1)T} \\
&= -\frac{\beta_{k-1}^I}{2\omega_p}\left[\cos 2\omega_p t\right]_{(2k-1)T}^{(2k-1)T+\underline{\tau}} - \frac{\beta_k^I}{2\omega_p}\left[\cos 2\omega_p t\right]_{(2k-1)T+\underline{\tau}}^{(2k+1)T}
\end{aligned}$$

Performing the integration results in (we denote $\omega_p\underline{\tau} = \underline{\varphi_p}$):

$$
\begin{aligned}
&= -\frac{\beta^I_{k-1}}{2\omega_p}\left[\cos\left((2k-1)\pi + 2\underline{\varphi_p}\right) - \cos\left((2k-1)\pi\right)\right] \\
&\quad - \frac{\beta^I_k}{2\omega_p}\left[\cos\left((2k+1)\pi\right) - \cos\left((2k-1)\pi + 2\underline{\varphi_p}\right)\right] \\
&= -\frac{\beta^I_{k-1}}{2\omega_p}\left(\cos\left(-\pi + 2\underline{\varphi_p}\right) - \cos\left(-\pi\right)\right) - \frac{\beta^I_k}{2\omega_p}\left(\cos\pi - \cos\left(-\pi + 2\underline{\varphi_p}\right)\right) \\
&= -\frac{\beta^I_{k-1}}{2\omega_p}\left(1 - \cos 2\underline{\varphi_p}\right) + \frac{\beta^I_k}{2\omega_p}\left(1 - \cos 2\underline{\varphi_p}\right) \\
&= -\frac{1}{2\omega_p}\left(1 - \cos 2\underline{\varphi_p}\right)\left(\beta^I_{k-1} - \beta^I_k\right)
\end{aligned}
$$

Using $\cos 2\underline{\varphi_p} = \cos\left(2\omega_p\left(\tau - 2k_\tau T\right)\right) = \cos\left(2\varphi_p - 2k_\tau\pi\right) = \cos 2\varphi_p$

$$= -\frac{1}{2\omega_p}\left(1 - \cos 2\varphi_p\right)\left(\beta^I_{k-1} - \beta^I_k\right)$$

The overall result is

$$S^I_k\left(\sin 2\omega_p t\right) = -\frac{1}{2\omega_p}\left(1 - \cos 2\varphi_p\right)\left(\beta^I_{k-1} - \beta^I_k\right) \tag{A.5}$$

Integration of Q-bits

This case can be performed analogously to Appendix A, with the following two simplifications:

$$\cos 2\underline{\varphi^Q_p} = \cos 2\omega_p\underline{\tau^Q} = \cos\left(2\frac{\pi}{2T}\left(\tau^Q - 2k^Q_\tau T\right)\right) = \cos\left(\frac{\tau^Q\pi}{T} - 2k^Q_\tau\pi\right) = \cos 2\varphi^Q_p$$

and

$$\cos 2\varphi^Q_p = \cos\left(2\varphi_p + \pi\right) = -\cos 2\varphi_p$$

The overall result is

$$S^Q_k\left(\sin 2\omega_p\left(t\right)\right) = -\frac{1}{2\omega_p}\left(1 + \cos 2\varphi_p\right)\left(\beta^Q_{k_{Q'}-1} - \beta^Q_{k_{Q'}}\right) \tag{A.6}$$

B. Colliding Packets Demodulator Output

With the tools presented in Appendix A, we can now proceed to prove theorem 4.1.

Theorem B.1. *For an interfering MSK signal $u\left(t\right)$ with parameters τ and φ_c, the contribution to the demodulation output $\Lambda_u^I\left(k\right)$ is given by*

$$\begin{aligned}\Lambda_u^I\left(k\right) = \frac{1}{4}A_u \Bigg\{ &\cos\varphi_c \left[\cos\varphi_p \left(\underline{\tau}\beta_{k-1}^I + \left(2T - \underline{\tau}\right)\beta_k^I\right) - \frac{2T}{\pi}\sin\varphi_p \left(\beta_{k-1}^I - \beta_k^I\right)\right] \\ &- \sin\varphi_c \left[\sin\varphi_p \left(\underline{\tau^Q}\beta_{kQ'-1}^Q + \left(2T - \underline{\tau^Q}\right)\beta_{kQ'}^Q\right) + \frac{2T}{\pi}\cos\varphi_p \left(\beta_{kQ'-1}^Q - \beta_{kQ'}^Q\right)\right]\Bigg\}.\end{aligned}$$

Proof. We first derive the resulting signal after demodulation (equation (4.7)).

$$\begin{aligned}&u\left(t\right)\phi_I\left(t\right) \\ &= A_u\left[b_I\left(t-\tau\right)\cos\left(\omega_p t - \varphi_p\right)\cos\left(\omega_c t + \varphi_c\right)\right. \\ &\qquad \left. + b_Q\left(t-\tau\right)\sin\left(\omega_p t - \varphi_p\right)\sin\left(\omega_c t + \varphi_c\right)\right]\left[\cos\omega_p t\cos\omega_c t\right] \\ &= A_u\left[\left(b_I\left(t-\tau\right)\cos\left(\omega_p t - \varphi_p\right)\cos\omega_p t\cos\left(\omega_c t + \varphi_c\right)\cos\omega_c t\right)\right. \\ &\qquad \left. + \left(b_Q\left(t-\tau\right)\sin\left(\omega_p t - \varphi_p\right)\cos\omega_p t\sin\left(\omega_c t + \varphi_c\right)\cos\omega_c t\right)\right] \\ &= \frac{A_u}{4}\left[\left(b_I\left(t-\tau\right)\left(\cos\varphi_p + \cos\left(2\omega_p t - \varphi_p\right)\right)\left(\cos\varphi_c + \cos\left(2\omega_c t + \varphi_c\right)\right)\right)\right. \\ &\qquad \left. + \left(b_Q\left(t-\tau\right)\left(\sin\left(-\varphi_p\right) + \sin\left(2\omega_p t - \varphi_p\right)\right)\left(\sin\varphi_c + \sin\left(2\omega_c t + \varphi_c\right)\right)\right)\right]\end{aligned}$$

We apply perfect lowpass filtering ($\star$) to filter out high-frequency components ($2\omega_c t$)

$$\begin{aligned}&\overset{\star}{=} \frac{A_u}{4}\left[\left(b_I\left(t-\tau\right)\cos\varphi_c\left(\cos\varphi_p + \cos\left(2\omega_p t - \varphi_p\right)\right)\right)\right. \\ &\qquad \left. + \left(b_Q\left(t-\tau\right)\sin\varphi_c\left(\sin\left(2\omega_p t - \varphi_p\right) - \sin\varphi_p\right)\right)\right] \\ &= \frac{A_u}{4}\left[\left(b_I\left(t-\tau\right)\cos\varphi_c\left(\cos\varphi_p + \cos 2\omega_p t\cos\varphi_p + \sin 2\varphi_p t\sin\varphi_p\right)\right)\right. \\ &\qquad \left. + \left(b_Q\left(t-\tau\right)\sin\varphi_c\left(-\sin\varphi_p + \sin 2\omega_p t\cos\varphi_p - \cos 2\omega_p t\sin\varphi_p\right)\right)\right]\end{aligned}$$

The bit decision is performed by integration over the bit interval k.

$$
\begin{aligned}
&\int_{(2k-1)T}^{(2k+1)T} u\left(t\right)\phi_I\left(t\right)dt \\
&= \frac{A_u}{4}\left[\cos\varphi_c \int_{(2k-1)T}^{(2k+1)T} b_I\left(t-\tau\right)\left(\cos\varphi_p + \cos 2\omega_p t\cos\varphi_p + \sin 2\varphi_p t\sin\varphi_p\right)dt\right. \\
&\qquad\left. + \sin\varphi_c \int_{(2k-1)T}^{(2k+1)T} b_Q\left(t-\tau\right)\left(-\sin\varphi_p + \sin 2\omega_p t\cos\varphi_p - \cos 2\omega_p t\sin\varphi_p\right)dt\right] \\
&= \frac{A_u}{4}\left[\cos\varphi_c \mathcal{X}_1 + \sin\varphi_c \mathcal{X}_2\right]
\end{aligned}
$$

We derive the results for both terms $\mathcal{X}_1$ and $\mathcal{X}_2$ individually in the following two sections.

Putting the two results in equation (B.1) and equation (B.2) together, the overall result is

$$
\begin{aligned}
&\int_{(2k-1)T}^{(2k+1)T} u\left(t\right)\phi_I\left(t\right)dt \\
&= \frac{A_u}{4}\left\{\cos\varphi_c\left[\cos\varphi_p\left(\underline{\tau}\beta^I_{k-1} + \left(2T-\underline{\tau}\right)\beta^I_k\right) - \frac{2T}{\pi}\sin\varphi_p\left(\beta^I_{k-1}-\beta^I_k\right)\right]\right. \\
&\qquad\left. - \sin\varphi_c\left[\sin\varphi_p\left(\underline{\tau^Q}\beta^Q_{kQ'-1} + \left(2T-\underline{\tau^Q}\right)\beta^Q_{kQ'}\right) + \frac{2T}{\pi}\cos\varphi_p\left(\beta^Q_{kQ'-1}-\beta^Q_{kQ'}\right)\right]\right\}
\end{aligned}
$$

□

Integrating the Term $\mathcal{X}_1$

$$
\begin{aligned}
&\int_{(2k-1)T}^{(2k+1)T} b_I\left(t-\tau\right)\left(\cos\varphi_p + \cos 2\omega_p t\cos\varphi_p + \sin 2\varphi_p t\sin\varphi_p\right)dt \\
&= \cos\varphi_p \int_{(2k-1)T}^{(2k+1)T} b_I\left(t-\tau\right)dt + \cos\varphi_p \int_{(2k-1)T}^{(2k+1)T} b_I\left(t-\tau\right)\cos 2\omega_p t\,dt \\
&\quad + \sin\varphi_p \int_{(2k-1)T}^{(2k+1)T} b_I\left(t-\tau\right)\sin 2\omega_p t\,dt \\
&= \cos\varphi_p S^I_k\left(1\right) + \cos\varphi_p S^I_k\left(\cos 2\omega_p t\right) + \sin\varphi_p S^I_k\left(\sin 2\omega_p t\right)
\end{aligned}
$$

By using the results in Appendix A (equations (A.1), (A.3) and (A.5)), we can reformulate this equation to

$$
\begin{aligned}
&= \cos\varphi_p \left(\underline{\tau}\beta^I_{k-1} + (2T - \underline{\tau})\,\beta^I_k\right) \\
&\quad - \frac{\beta^I_{k-1}}{2\omega_p}\left(\cos\varphi_p \sin 2\varphi_p + \sin\varphi_p\,(1 - \cos 2\varphi_p)\right) + \frac{\beta^I_k}{2\omega_p}\left(\cos\varphi_p \sin 2\varphi_p + \sin\varphi_p\,(1 - \cos 2\varphi_p)\right)
\end{aligned}
$$

Simplifying this equation yields the desired result.

$$
\begin{aligned}
&= \cos\varphi_p \left(\underline{\tau}\beta^I_{k-1} + (2T - \underline{\tau})\,\beta^I_k\right) - \left(\frac{\beta^I_{k-1} - \beta^I_k}{2\omega_p}\right)\left(\sin 2\varphi_p \cos\varphi_p - \cos 2\varphi_p \sin\varphi_p + \sin\varphi_p\right) \\
&= \cos\varphi_p \left(\underline{\tau}\beta^I_{k-1} + (2T - \underline{\tau})\,\beta^I_k\right) - \frac{\sin\varphi_p}{\omega_p}\left(\beta^I_{k-1} - \beta^I_k\right) \\
&= \cos\varphi_p \left(\underline{\tau}\beta^I_{k-1} + (2T - \underline{\tau})\,\beta^I_k\right) - \frac{2T}{\pi}\sin\varphi_p\left(\beta^I_{k-1} - \beta^I_k\right)
\end{aligned}
$$

In the second step in the previous derivation, we used the following simplification:

$$
\begin{aligned}
&\sin 2\varphi_p \cos\varphi_p - \cos 2\varphi_p \sin\varphi_p + \sin\varphi_p \\
&= 2\cos^2\varphi_p \sin\varphi_p - \left(2\cos^2\varphi_p - 1\right)\sin\varphi_p + \sin\varphi_p \\
&= \left(2\cos^2\varphi_p - 2\cos^2\varphi_p + 1 + 1\right)\sin\varphi_p \\
&= 2\sin\varphi_p
\end{aligned}
$$

Overall, the result is

$$
\mathcal{X}_1 = \cos\varphi_p \left(\underline{\tau}\beta^I_{k-1} + (2T - \underline{\tau})\,\beta^I_k\right) - \frac{2T}{\pi}\sin\varphi_p\left(\beta^I_{k-1} - \beta^I_k\right) \tag{B.1}
$$

Integrating the Term $\mathcal{X}_2$

We will now derive the second integral. We must use the rules for Q pulse integration with I intervals (Appendix A).

$$
\begin{aligned}
&\int_{(2k-1)T}^{(2k+1)T} b_Q\,(t-\tau)\left(-\sin\varphi_p - \cos 2\omega_p t \sin\varphi_p + \sin 2\omega_p t \cos\varphi_p\right) dt \\
&= -\Bigg[\int_{(2k-1)T}^{(2k+1)T} b_Q\,(t-\tau)\sin\varphi_p dt + \int_{(2k-1)T}^{(2k+1)T} b_Q\,(t-\tau)\cos 2\omega_p t \sin\varphi_p dt \\
&\qquad - \int_{(2k-1)T}^{(2k+1)T} b_Q\,(t-\tau)\sin 2\omega_p t \cos\varphi_p dt\Bigg] \\
&= -\left[\sin\varphi_p S^Q_k\,(1) + \sin\varphi_p S^Q_k\,(\cos 2\omega_p t) - \cos\varphi_p S^Q_k\,(\sin 2\omega_p t)\right]
\end{aligned}
$$

By using the results in Appendix A (equations (A.2), (A.4) and (A.6)), we can reformulate this equation to

$$\begin{aligned}
&= -\sin\varphi_p \left(\underline{\tau}^{Q}\beta^{Q}_{kQ'-1} + \left(2T - \underline{\tau}^{Q}\right)\beta^{Q}_{kQ'}\right) \\
&\quad - \frac{1}{2\omega_p}\left(\sin\varphi_p \sin 2\varphi_p + \cos\varphi_p\left(1 + \cos 2\varphi_p\right)\right)\left(\beta^{Q}_{kQ'-1} - \beta^{Q}_{kQ'}\right)
\end{aligned}$$

Simplifying yield the desired result

$$\begin{aligned}
&= -\sin\varphi_p \left(\underline{\tau}^{Q}\beta^{Q}_{kQ'-1} + \left(2T - \underline{\tau}^{Q}\right)\beta^{Q}_{kQ'}\right) \\
&\quad - \frac{1}{2\omega_p}\left(\sin 2\varphi_p \sin\varphi_p + \cos 2\varphi_p \cos\varphi_p + \cos\varphi_p\right)\left(\beta^{Q}_{kQ'-1} - \beta^{Q}_{kQ'}\right) \\
&= -\left[\sin\varphi_p \left(\underline{\tau}^{Q}\beta^{Q}_{kQ'-1} + \left(2T - \underline{\tau}^{Q}\right)\beta^{Q}_{kQ'}\right) + \frac{2T}{\pi}\cos\varphi_p\left(\beta^{Q}_{kQ'-1} - \beta^{Q}_{kQ'}\right)\right]
\end{aligned}$$

In the last step, we used the following simplification

$$\begin{aligned}
&\sin 2\varphi_p \sin\varphi_p + \cos\varphi_p + \cos 2\varphi_p \cos\varphi_p \\
&= 2\sin^2\varphi_p \cos\varphi_p + \cos\varphi_p + \left(1 - 2\sin^2\varphi_p\right)\cos\varphi_p \\
&= \left(2\sin^2\varphi_p + 1 + 1 - 2\sin^2\varphi_p\right)\cos\varphi_p \\
&= 2\cos\varphi_p
\end{aligned}$$

Overall, the result is

$$\mathcal{X}_2 = -\left[\sin\varphi_p \left(\underline{\tau}^{Q}\beta^{Q}_{kQ'-1} + \left(2T - \underline{\tau}^{Q}\right)\beta^{Q}_{kQ'}\right) + \frac{2T}{\pi}\cos\varphi_p\left(\beta^{Q}_{kQ'-1} - \beta^{Q}_{kQ'}\right)\right] \tag{B.2}$$

Bibliography

IEEE Standard 802.15.1 (2005). *802.15.1-2005: IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs).* IEEE Computer Society. DOI: `10.1109/IEEESTD.2005.96290` (Cited on pp. 112, 127).

IEEE Standard 802.15.4 (2006). *802.15.4-2006: IEEE Standard for Information technology—Telecommunications and information exchange between systems—Local and metropolitan area networks—Specific requirements—Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs).* IEEE Computer Society. DOI: `10.1109/IEEESTD.2006.232110` (Cited on pp. 34, 35, 39, 50, 68, 70, 104).

ADAMY, DAVID (2001). *EW 101: A First Course in Elecronic Warfare.* Boston, MA, USA: Artech House Publishers. ISBN: 9781580531696 (Cited on p. 67).

AIRTIGHT NETWORKS (2011). *Complete wireless security for your network: SpectraGuard Enterprise.* Retrieved from `www.airtightnetworks.com` (Cited on p. 11).

BAHL, PARAMVIR and VENKATA N. PADMANABHAN (2000). "RADAR: An In-Building RF-Based User Location and Tracking System". In: *Proceedings of the 19th IEEE International Conference on Computer Communications.* INFOCOM 2000. (Tel-Aviv, Mar. 26–30, 2000). Vol. 2. New York, NY, USA: IEEE, pp. 775–784. DOI: `10.1109/INFCOM.2000.832252` (Cited on p. 123).

BAYRAKTAROGLU, EMRAH, CHRISTOPHER KING, XIN LIU, GUEVARA NOUBIR, RAJMOHAN RAJARAMAN, and BISHAL THAPA (2008). "On the Performance of IEEE 802.11 under Jamming". In: *Proceedings of the 27th IEEE International Conference on Computer Communications.* INFOCOM 2008. (Phoenix, AZ, Apr. 13–18, 2008). New York, NY, USA: IEEE, pp. 1265–1273. DOI: `10.1109/INFOCOM.2008.183` (Cited on p. 27).

BELLOVIN, STEVEN M. and WILLIAM R. CHESWICK (1994). "Network Firewalls". In: *IEEE Communications Magazine* 32(9), pp. 50–57. DOI: `10.1109/35.312843` (Cited on p. 3).

Berger, Daniel S., Francesco Gringoli, Nicolò Facchi, Ivan Martinovic, and Jens B. Schmitt (2014). “Gaining Insight on Friendly Jamming in a Real-World IEEE 802.11 Network”. In: *Proceedings of the 7th ACM Conference on Security and Privacy in Wireless and Mobile Networks.* WiSec 2014. (Oxford, July 23–25, 2014). New York, NY, USA: ACM. doi: `10.1145/2627393.2627403` (Cited on p. 125).

Boano, Carlo Alberto, Zhitao He, Yafei Li, Thiemo Voigt, Marco Zuniga, and Andreas Willig (2009). “Controllable Radio Interference for Experimental and Testing Purposes in Wireless Sensor Networks”. In: *Proceedings of the 34th IEEE Conference on Local Computer Networks.* LCN 2009. (Zürich, Oct. 20–23, 2009). New York, NY, USA: IEEE, pp. 865–872. doi: `10.1109/LCN.2009.5355013` (Cited on p. 128).

Boano, Carlo Alberto, Thiemo Voigt, Claro Noda, Kai Romer, and Marco Zúñiga (2011). “JamLab: Augmenting Sensornet Testbeds with Realistic and Controlled Interference Generation”. In: *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks.* IPSN 2011. (Chicago, IL, Apr. 12–14, 2011). New York, NY, USA: IEEE, pp. 175–186. url: `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5779101` (Cited on pp. 127, 128).

Brik, Vladimir, Suman Banerjee, Marco Gruteser, and Sangho Oh (2008). “Wireless Device Identification with Radiometric Signatures”. In: *Proceedings of the 14th Annual International Conference on Mobile Computing and Networking.* MobiCom 2008. (San Francisco, CA, Sept. 14–19, 2008). New York, NY, USA: ACM, pp. 116–127. doi: `10.1145/1409944.1409959` (Cited on p. 124).

Brown, James, Ibrahim Ethem Bagci, Alex King, and Utz Rödig (2013). “Defend Your Home! Jamming Unsolicited Messages in the Smart Home”. In: *Proceedings of the 2nd ACM Workshop on Hot Topics on Wireless Network Security and Privacy.* HotWiSec 2013. (Budapest, Apr. 19, 2013). New York, NY, USA: ACM, pp. 1–6. doi: `10.1145/2463183.2463185` (Cited on p. 15).

Cardieri, Paulo (2010). “Modeling Interference in Wireless Ad Hoc Networks”. In: *IEEE Communications Surveys & Tutorials* 12(4), pp. 551–572. doi: `10.1109/SURV.2010.032710.00096` (Cited on p. 32).

Chan, Haowen, Virgil D. Gligor, Adrian Perrig, and Gautam Muralidharan (2005). “On the Distribution and Revocation of Cryptographic Keys in Sensor Networks”. In: *IEEE Transactions on Dependable and Secure Computing* 2(3), pp. 233–247. doi: `10.1109/TDSC.2005.37` (Cited on p. 120).

Chipara, Octav, Chenyang Lu, Thomas C. Bailey, and Gruia-Catalin Roman (2010). “Reliable Clinical Monitoring using Wireless Sensor Networks: Experiences in a Step-Down Hospital Unit". In: *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems.* SenSys 2010. (Zürich, Nov. 3–5, 2010). New York, NY, USA: ACM, pp. 155–168. doi: `10.1145/1869983.1869999` (Cited on p. 2).

Choi, Jung Il, Mayank Jain, Kannan Srinivasan, Phil Levis, and Sachin Katti (2010). "Achieving Single Channel, Full Duplex Wireless Communication". In: *Proceedings of the 16th Annual International Conference on Mobile Computing and Networking.* MobiCom 2010. (Chicago, IL, Sept. 20–24, 2010). New York, NY, USA: ACM, pp. 1–12. doi: `10.1145/1859995.1859997` (Cited on p. 78).

Code of Federal Regulations (2010). *Code of Federal Regulations, Title 47, Chapter I FCC Part 15—Radio Frequency Devices.* Washington DC, USA: Federal Communications Commission (Cited on p. 126).

Danev, Boris, Heinrich Luecken, Srđan Čapkun, and Karim El Defrawy (2010). "Attacks on Physical-Layer Identification". In: *Proceedings of the 3rd ACM Conference on Wireless Network Security.* WiSec 2010. (Hoboken, NJ, Mar. 22–24, 2010). New York, NY, USA: ACM, pp. 89–98. doi: `10.1145/1741866.1741882` (Cited on p. 124).

Danev, Boris, Davide Zanetti, and Srđan Čapkun (2012). "On Physical-Layer Identification of Wireless Devices". In: *ACM Computing Surveys* 45(1), 6:1–6:29. doi: `10.1145/2379776.2379782` (Cited on p. 124).

Davis, Donald and Stephen Gronemeyer (1980). "Performance of Slotted ALOHA Random Access with Delay Capture and Randomized Time of Arrival". In: *IEEE Transaction on Communications* 28(5), pp. 703–710. doi: `10.1109/TCOM.1980.1094718` (Cited on pp. 30, 33).

DeBruhl, Bruce and Patrick Tague (2011). "Digital Filter Design for Jamming Mitigation in 802.15.4 Communication". In: *Proceedings of the 20th International Conference on Computer Communications and Networks.* ICCCN 2011. (Maui, HI, July 31–Aug. 4, 2011). New York, NY, USA: IEEE, pp. 1–6. doi: `10.1109/ICCCN.2011.6006020` (Cited on p. 23).

Doddavenkatappa, Manjunath, Mun Choon Chan, and Ben Leong (2013). "Splash: Fast Data Dissemination with Constructive Interference in Wireless Sensor Networks". In: *Proceedings of the 10th USENIX Symposium on Networked Systems Design and Implementation.* NSDI 2013. (Lombard, IL, Apr. 2–5, 2013). Berkeley, CA, USA: USENIX Association, pp. 269–282. url: `http://dl.acm.org/citation.cfm?id=2482626.2482653` (Cited on pp. 30, 32, 65).

Dutta, Prabal, Stephen Dawson-Haggerty, Yin Chen, Chieh-Jan Mike Liang, and Andreas Terzis (2010). "Design and Evaluation of a Versatile and Efficient Receiver-Initiated Link Layer for Low-Power Wireless". In: *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems.* SenSys

2010. (Zürich, Nov. 3–5, 2010). New York, NY, USA: ACM, pp. 1–14. DOI: `10.1145/1869983.1869985` (Cited on pp. 30, 32, 33, 58, 64).

Dutta, Prabal, Ye-Sheng Kuo, Akos Ledeczi, Thomas Schmid, and Peter Volgyesi (2010). "Putting the Software Radio on a Low-Calorie Diet". In: *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks.* HotNets-IX. (Monterey, CA, Oct. 20–21, 2010). New York, NY, USA: ACM, 20:1–20:6. DOI: `10.1145/1868447.1868467` (Cited on p. 126).

Dutta, Prabal, Răzvan Musăloiu-E., Ion Stoica, and Andreas Terzis (2008). "Wireless ACK Collisions Not Considered Harmful". In: *Proceedings of the 7th ACM SIGCOMM Workshop on Hot Topics in Networks.* HotNets-VII. (Calgary, Oct. 6–7, 2008). New York, NY, USA: ACM, 4:1–4:6. URL: `http://conferences.sigcomm.org/hotnets/2008/papers/4.pdf` (Cited on p. 30).

Ferrari, Federico, Marco Zimmerling, Lothar Thiele, and Olga Saukh (2011). "Efficient Network Flooding and Time Synchronization with Glossy". In: *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks.* IPSN 2011. (Chicago, IL, Apr. 12–14, 2011). New York, NY, USA: IEEE, pp. 73–84. URL: `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5779066` (Cited on pp. 30, 32, 34, 50, 58, 60, 65).

Fluke Corporation (2011). *AirMagnet solutions: Wi-Fi done right.* Retrieved from `http://www.airmagnet.com` (Cited on p. 11).

Foo, Justin and Defeng Huang (2008). "Multiuser Diversity with Capture for Wireless Networks: Protocol and Performance Analysis". In: *IEEE Journal on Selected Areas in Communications* 26(8), pp. 1386–1396. DOI: `10.1109/JSAC.2008.081005` (Cited on p. 30).

Fu, Kevin (2009). "Inside Risks: Reducing Risks of Implantable Medical Devices". In: *Communications of the ACM* 52(6), pp. 25–27. DOI: `10.1145/1516046.1516055` (Cited on p. 13).

Gezer, Cengiz, Chiara Buratti, and Roberto Verdone (2010). "Capture Effect in IEEE 802.15.4 Networks: Modelling and Experimentation". In: *Proceedings of the 2010 5th IEEE International Symposium on Wireless Pervasive Computing.* ISWPC 2010. (Modena, Italy, May 5–7, 2010). New York, NY, USA: IEEE, pp. 204–209. DOI: `10.1109/ISWPC.2010.5483727` (Cited on pp. 30, 53, 64, 66).

Giustiniano, Domenico, Vincent Lenders, Jens B. Schmitt, Michael Spuhler, and Matthias Wilhelm (2013). "Detection of Reactive Jamming in DSSS-based Wireless Networks". In: *Proceedings of the 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks.* WiSec 2013. (Budapest, Apr. 17–19, 2013). New York, NY, USA: ACM, pp. 43–48. DOI: `10.1145/2462096.2462104`.

Goel, Satashu and Rohit Negi (2008). “Guaranteeing Secrecy using Artificial Noise”. In: *IEEE Transactions on Wireless Communications* 7(6), pp. 2180–2189. DOI: `10.1109/TWC.2008.060848` (Cited on p. 10).

Gollakota, Shyamnath, Haitham Hassanieh, Benjamin Ransford, Dina Katabi, and Kevin Fu (2011). “They Can Hear Your Heartbeats: Non-Invasive Security for Implantable Medical Devices”. In: *Proceedings of the ACM SIGCOMM 2011 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* SIGCOMM 2011. (Toronto, Aug. 15–19, 2011). New York, NY, USA: ACM, pp. 2–13. DOI: `10.1145/2018436.2018438` (Cited on pp. 13, 78).

Gollakota, Shyamnath and Dina Katabi (2011). “Physical Layer Wireless Security Made Fast and Channel Independent”. In: *Proceedings of the 30th IEEE International Conference on Computer Communications.* INFOCOM 2011. (Shanghai, Apr. 10–15, 2011). New York, NY, USA: IEEE, pp. 1125–1133. DOI: `10.1109/INFCOM.2011.5934889` (Cited on p. 10).

Gummadi, Ramakrishna, David Wetherall, Ben Greenstein, and Srinivasan Seshan (2007). “Understanding and Mitigating the Impact of RF Interference on 802.11 Networks”. In: *Proceedings of the ACM SIGCOMM 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* SIGCOMM 2007. (Kyoto, Aug. 27–31, 2007). New York, NY, USA: ACM, pp. 385–396. DOI: `10.1145/1282380.1282424` (Cited on pp. 30–32, 37).

Gupta, Piyush and P. R. Kumar (2000). “The Capacity of Wireless Networks”. In: *IEEE Transactions on Information Theory* 46(2), pp. 388–404. DOI: `10.1109/18.825799` (Cited on p. 32).

Haeberlen, Andreas, Eliot Flannery, Andrew M. Ladd, Algis Rudys, Dan S. Wallach, and Lydia E. Kavraki (2004). “Practical Robust Localization over Large-Scale 802.11 Wireless Networks”. In: *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking.* MobiCom 2004. (Philadelphia, PA, Sept. 26–Oct. 1, 2004). New York, NY, USA: ACM, pp. 70–84. DOI: `10.1145/1023720.1023728` (Cited on p. 123).

Halperin, Daniel, Thomas E. Anderson, and David Wetherall (2008). “Taking the Sting out of Carrier Sense: Interference Cancellation for Wireless LANs”. In: *Proceedings of the 14th Annual International Conference on Mobile Computing and Networking.* MobiCom 2008. (San Francisco, CA, Sept. 14–19, 2008). New York, NY, USA: ACM, pp. 339–350. DOI: `10.1145/1409944.1409983` (Cited on pp. 11, 24, 125).

HALPERIN, DANIEL, THOMAS S. HEYDT-BENJAMIN, BENJAMIN RANSFORD, SHANE S. CLARK, BENESSA DEFEND, WILL MORGAN, KEVIN FU, TADAYOSHI KOHNO, and WILLIAM H. MAISEL (2008). "Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses". In: *Proceedings of the 2008 IEEE Symposium on Security and Privacy.* S&P 2008. (Oakland, CA, May 18–21, 2008). Washington DC, USA: IEEE Computer Society, pp. 129–142. DOI: `10.1109/SP.2008.31` (Cited on p. 13).

HE, XIANG and AYLIN YENER (2009). "Cooperative Jamming: The Tale of Friendly Interference for Secrecy". In: *Securing Wireless Communications at the Physical Layer.* Ed. by RUOHENG LIU and WADE TRAPPE. New York, NY, USA: Springer US, pp. 65–88. DOI: `10.1007/978-1-4419-1385-2_4` (Cited on p. 10).

HE, ZHITAO and THIEMO VOIGT (2011). "Precise Packet Loss Pattern Generation by Intentional Interference". In: *Proceedings of the 7^{th} IEEE International Conference on Distributed Computing in Sensor Systems.* DCOSS 2011. (Barcelona, June 27–29, 2011). New York, NY, USA: IEEE, pp. 1–6. DOI: `10.1109/DCOSS.2011.5982225` (Cited on pp. 87, 127, 130).

HERMANS, FREDERIK, HJALMAR WENNERSTRÖM, LIAM MCNAMARA, CHRISTIAN ROHNER, and PER GUNNINGBERG (2014). "All is not Lost: Understanding and Exploiting Packet Corruption in Outdoor Sensor Networks". In: *Wireless Sensor Networks: Proceedings of the 11^{th} European Conference, EWSN 2014.* Ed. by BHASKAR KRISHNAMACHARI, AMY L. MURPHY, and NIKI TRIGONI. Lecture Notes in Computer Science 8354. Berlin, Germany: Springer Berlin Heidelberg, pp. 116–132. DOI: `10.1007/978-3-319-04651-8_8` (Cited on p. 31).

HNAT, TIMOTHY W., VIJAY SRINIVASAN, JIAKANG LU, TAMIM I. SOOKOOR, RAYMOND DAWSON, JOHN STANKOVIC, and KAMIN WHITEHOUSE (2011). "The Hitchhiker's Guide to Successful Residential Sensing Deployments". In: *Proceedings of the 9^{th} ACM Conference on Embedded Networked Sensor Systems.* SenSys 2011. (Seattle, WA, Nov. 1–4, 2011). New York, NY, USA: ACM, pp. 232–245. DOI: `10.1145/2070942.2070966` (Cited on p. 2).

HODJAT, ALIREZA and INGRID VERBAUWHEDE (2004). "A 21.54 Gbits/s Fully Pipelined AES Processor on FPGA". In: *Proceedings of the 12^{th} Annual IEEE Symposium on Field-Programmable Custom Computing Machines.* FCCM 2004. (Napa Valley, CA, Apr. 20–23, 2004). New York, NY, USA: IEEE, pp. 308–309. DOI: `10.1109/FCCM.2004.1` (Cited on p. 35).

JAIN, RAJ K. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling.* Hoboken, NJ: John Wiley & Sons (Cited on p. 49).

Jamieson, Kyle and Hari Balakrishnan (2007). “PPR: Partial Packet Recovery for Wireless Networks”. In: *Proceedings of the ACM SIGCOMM 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* SIGCOMM 2007. (Kyoto, Aug. 27–31, 2007). New York, NY, USA: ACM, pp. 409–420. doi: `10.1145/1282427.1282426` (Cited on p. 31).

Jiang, Xiaofan, Minh Van Ly, Jay Taneja, Prabal Dutta, and David Culler (2009). “Experiences with a High-Fidelity Wireless Building Energy Auditing Network”. In: *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems.* SenSys 2009. (Berkeley, CA, Nov. 4–6, 2009). New York, NY, USA: ACM, pp. 113–126. doi: `10.1145/1644038.1644050` (Cited on p. 2).

Juels, Ari and John Brainard (2004). “Soft Blocking: Flexible Blocker Tags on the Cheap”. In: *Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society.* WPES 2004. (Washington DC, Oct. 28, 2004). New York, NY, USA: ACM, pp. 1–7. doi: `10.1145/1029179.1029181` (Cited on p. 14).

Juels, Ari, Ronald L. Rivest, and Michael Szydlo (2003). “The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy”. In: *Proceedings of the 10th ACM Conference on Computer and Communications Security.* CCS 2003. (Washington DC, Oct. 27–30, 2003). New York, NY, USA: ACM, pp. 103–111. doi: `10.1145/948109.948126` (Cited on p. 14).

Karhima, Teemu, Aki Silvennoinen, Michael Hall, and Sven-Gustav Haggman (2004). “IEEE 802.11b/g WLAN Tolerance to Jamming”. In: *Proceedings of the 2004 IEEE Military Communications Conference.* MILCOM 2004. (Monterey, CA, Oct. 31–Nov. 3, 2004). Vol. 3. New York, NY, USA: IEEE, pp. 1364–1370. doi: `10.1109/MILCOM.2004.1495141` (Cited on p. 23).

Kim, Yu Seung, Patrick Tague, Heejo Lee, and Hyogon Kim (2012). “Carving Secure Wi-Fi Zones with Defensive Jamming”. In: *Proceedings of the 7th ACM Symposium on Information, Computer, and Communication Security.* AsiaCCS 2012. (Seoul, May 2–4, 2012). New York, NY, USA: ACM, pp. 53–58. doi: `10.1145/2414456.2414487` (Cited on pp. 10, 126).

Kochut, Andrzej, Arunchandar Vasan, A. Udaya Shankar, and Ashok Agrawala (2004). “Sniffing Out the Correct Physical Layer Capture Model in 802.11b”. In: *Proceedings of the 12th IEEE International Conference on Network Protocols.* ICNP 2004. (Berlin, Oct. 5–8, 2004). New York, NY, USA: IEEE, pp. 252–261. doi: `10.1109/ICNP.2004.1348115` (Cited on p. 30).

Lai, Lifeng and Hesham El Gamal (2008). “The Relay-Eavesdropper Channel: Cooperation for Secrecy”. In: *IEEE Transactions on Information Theory* 54(9), pp. 4005–4019. doi: `10.1109/TIT.2008.928272` (Cited on p. 10).

LaShomb, Lisandra (2006). "Making the Impossible a Reality". In: *Defense Contract Management Agency Communicator* 6(3), pp. 50–53. url: `http://www.dcma.mil/communicator/summer06/DCMA_Comm_v06n03_full.pdf` (Cited on p. 16).

Law, Yee Wei, Marimuthu Palaniswami, Lodewijk Van Hoesel, Jeroen Doumen, Pieter Hartel, and Paul Havinga (2009). "Energy-Efficient Link-Layer Jamming Attacks against Wireless Sensor Network MAC Protocols". In: *ACM Transactions on Sensor Networks* 5(1), 6:1–6:38. doi: `10.1145/1464420.1464426` (Cited on pp. 27, 107).

Lee, Jeongkeun, Wonho Kim, Sung-Ju Lee, Daehyung Jo, Jiho Ryu, Taekyoung Kwon, and Yanghee Choi (2007). "An Experimental Study on the Capture Effect in 802.11a Networks". In: *Proceedings of the 2nd ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization.* WiNTECH 2007. (Montréal, Sept. 10, 2007). New York, NY, USA: ACM, pp. 19–26. doi: `10.1145/1287767.1287772` (Cited on pp. 30, 32, 33).

Leentvaar, Krijn and Jan H. Flint (1976). "The Capture Effect in FM Receivers". In: *IEEE Transactions on Communications* 24(5), pp. 531–539. doi: `10.1109/TCOM.1976.1093327` (Cited on p. 30).

Liang, Chieh-Jan Mike, Nissanka Bodhi Priyantha, Jie Liu, and Andreas Terzis (2010). "Surviving Wi-Fi Interference in Low Power ZigBee Networks". In: *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems.* SenSys 2010. (Zürich, Nov. 3–5, 2010). New York, NY, USA: ACM, pp. 309–322. doi: `10.1145/1869983.1870014` (Cited on pp. 34, 128).

Lin, Guolong and Guevara Noubir (2005). "On Link Layer Denial of Service in Data Wireless LANs". In: *Wireless Communications and Mobile Computing* 5(3), pp. 273–284. doi: `10.1002/wcm.221` (Cited on p. 107).

Liu, Ruoheng, I. Marić, P. Spasojević, and Roy D. Yates (2008). "Discrete Memoryless Interference and Broadcast Channels With Confidential Messages: Secrecy Rate Regions". In: *IEEE Transactions on Information Theory* 54(6), pp. 2493–2507. doi: `10.1109/TIT.2008.921879` (Cited on p. 10).

Lu, Jiakang, Tamim Sookoor, Vijay Srinivasan, Ge Gao, Brian Holben, John Stankovic, Eric Field, and Kamin Whitehouse (2010). "The Smart Thermostat: Using Occupancy Sensors to Save Energy in Homes". In: *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems.* SenSys 2010. (Zürich, Nov. 3–5, 2010). New York, NY, USA: ACM, pp. 211–224. doi: `10.1145/1869983.1870005` (Cited on p. 2).

Lu, Jiakang and Kamin Whitehouse (2009). "Flash Flooding: Exploiting the Capture Effect for Rapid Flooding in Wireless Sensor Networks". In: *Proceedings* *of the 28th IEEE International Conference on Computer Communications.* INFOCOM 2009. (Rio de Janeiro, Apr. 19–25, 2009). New York, NY, USA: IEEE, pp. 2491–2499. DOI: `10.1109/INFCOM.2009.5062177` (Cited on pp. 30, 32, 65).

Maheshwari, Ritesh, Shweta Jain, and Samir R. Das (2008). "A Measurement Study of Interference Modeling and Scheduling in Low-Power Wireless Networks". In: *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems.* SenSys 2008. (Raleigh, NC, Nov. 5–7, 2008). New York, NY, USA: ACM, pp. 141–154. DOI: `10.1145/1460412.1460427` (Cited on pp. 30–32, 64, 65).

Manweiler, Justin, Naveen Santhapuri, Souvik Sen, Romit Roy Choudhury, Srihari Nelakuditi, and Kamesh Munagala (2012). "Order Matters: Transmission Reordering in Wireless Networks". In: *IEEE/ACM Transactions on Networking* 20(2), pp. 353–366. DOI: `10.1109/TNET.2011.2164264` (Cited on p. 33).

Martinovic, Ivan, Nicos Gollan, and Jens B. Schmitt (2008). "Firewalling Wireless Sensor Networks: Security by Wireless". In: *Prococeedings of the 3rd Workshop on Practical Issues in Building Sensor Network Applications.* SenseApp 2008. (Montréal, Oct. 17, 2008). New York, NY, USA: IEEE, pp. 770–777. DOI: `10.1109/LCN.2008.4664279` (Cited on p. 15).

Martinovic, Ivan, Paul Pichota, and Jens B. Schmitt (2009). "Jamming for Good: A Fresh Approach to Authentic Communication in WSNs". In: *Proceedings of the 2nd ACM Conference on Wireless Network Security.* WiSec 2009. (Zürich, Mar. 16–18, 2009). New York, NY, USA: ACM, pp. 161–168. DOI: `10.1145/1514274.1514298` (Cited on pp. 15, 87).

Motorola Solutions (2011). *Motorola AirDefense—Security & Compliance Solutions.* Retrieved from `http://www.airdefense.net` (Cited on p. 11).

Nguyen, Danh, Cem Sahin, Boris Shishkin, Nagarajan Kandasamy, and Kapil R. Dandekar (2014). "A Real-time and Protocol-aware Reactive Jamming Framework Built on Software-defined Radios". In: *Proceedings of the 2014 ACM Workshop on Software Radio Implementation Forum.* SRIF 2014. (Chicago, IL, Aug. 18, 2014). New York, NY, USA: ACM, pp. 15–22. DOI: `10.1145/2627788.2627798` (Cited on p. 81).

Nychis, George, Thibaud Hottelier, Zhuocheng Yang, Srinivasan Seshan, and Peter Steenkiste (2009). "Enabling MAC Protocol Implementations on Software-Defined Radios". In: *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation.* NSDI 2009. (Boston, MA, Apr. 22–24, 2009). Berkeley, CA, USA: USENIX Association, pp. 91–105. URL: `http://portal.acm.org/citation.cfm?id=1558977.1558984` (Cited on pp. 81, 87).

O'Flynn, Colin P. (2011). "Message Denial and Alteration on IEEE 802.15.4 Low-Power Radio Networks". In: *Proceedings of the 4th IFIP International Conference on New Technologies, Mobility and Security.* NTMS 2011. (Paris, Feb. 7–10, 2011). New York, NY, USA: IEEE, pp. 1–5. doi: `10.1109/NTMS.2011.5720580` (Cited on p. 87).

Oh, Nam-Jin and Sang-Gug Lee (2005). "Building a 2.4-GHz Radio Transceiver using IEEE 802.15.4". In: *IEEE Circuits and Devices Magazine* 21(6), pp. 43–51. doi: `10.1109/MCD.2005.1578587` (Cited on pp. 37, 38).

Parno, Brian, Adrian Perrig, and Virgil D. Gligor (2005). "Distributed Detection of Node Replication Attacks in Sensor Networks". In: *Proceedings of the 2005 IEEE Symposium on Security and Privacy.* S&P 2005. (Oakland, CA, May 8–11, 2005). Washington DC, USA: IEEE Computer Society, pp. 49–63. doi: `10.1109/SP.2005.8` (Cited on p. 120).

Pasupathy, Subbarayan (1979). "Minimum Shift Keying: A Spectrally Efficient Modulation". In: *IEEE Communications Magazine* 17(4), pp. 14–22. doi: `10.1109/MCOM.1979.1089999` (Cited on p. 42).

Patwari, Neal and Sneha K. Kasera (2007). "Robust Location Distinction using Temporal Link Signatures". In: *Proceedings of the 13th Annual International Conference on Mobile Computing and Networking.* MobiCom 2007. (Montréal, Sept. 9–14, 2007). New York, NY, USA: ACM, pp. 111–122. doi: `10.1145/1287853.1287867` (Cited on p. 124).

Poisel, Richard A. (2011). *Modern Communications Jamming: Principles and Techniques.* 2nd edition. Boston, MA, USA: Artech House Publishers. isbn: 9781608071654 (Cited on pp. 23, 50, 108, 110).

Pöpper, Christina, Nils Ole Tippenhauer, Boris Danev, and Srđan Čapkun (2011). "Investigation of Signal and Message Manipulations on the Wireless Channel". In: *Computer Security – ESORICS 2011: Proceedings of the 16th European Symposium on Research in Computer Security.* Lecture Notes in Computer Science 6879. Berlin, Germany: Springer Berlin Heidelberg, pp. 40–59. doi: `10.1007/978-3-642-23822-2_3` (Cited on pp. 30, 31, 34, 60, 65).

Proakis, John and Masoud Salehi (2007). *Digital Communications.* 5th edition. New York, NY, USA: McGraw-Hill. isbn: 9780071263788 (Cited on pp. 33, 34, 40, 42–44).

Rappaport, Theodore S. (2001). *Wireless Communications: Principles and Practice.* 2nd edition. Upper Saddle River, NJ, USA: Prentice-Hall. isbn: 9780130422323 (Cited on p. 50).

Rieback, Melanie R., Bruno Crispo, and Andrew S. Tanenbaum (2005). "RFID Guardian: A Battery-Powered Mobile Device for RFID Privacy Management". In: *Information Security and Privacy: Proceedings of the 10th Australasian Conference, ACISP 2005.* Ed. by Colin Boyd and Juan Manuel González Nieto. Lecture Notes in Computer Science 3574. Berlin, Germany: Springer Berlin Heidelberg, pp. 259–273. doi: `10.1007/11506157_16` (Cited on p. 14).

– (2007). "Keep on Blockin' in the Free World: Personal Access Control for Low-Cost RFID Tags". In: *Security Protocols: 13th International Workshop – Revised Selected Papers.* Ed. by Bruce Christianson, Bruno Crispo, James A. Malcolm, and Michael Roe. Lecture Notes in Computer Science 4631. Berlin, Germany: Springer Berlin Heidelberg, pp. 51–59. doi: `10.1007/978-3-540-77156-2_6` (Cited on p. 14).

Sankararaman, Swaminathan, Karim Abu-Affash, Alon Efrat, Sylvester David Eriksson-Bique, Valentin Polishchuk, Srinivasan Ramasubramanian, and Michael Segal (2012). "Optimization schemes for protective jamming". In: *Proceedings of the 13th ACM International Symposium on Mobile Ad Hoc Networking and Computing.* MobiHoc 2012. (Hilton Head Island, SC, June 12–14, 2012). New York, NY, USA: ACM, pp. 65–74. doi: `10.1145/2248371.2248383` (Cited on pp. 10, 126).

Santhapuri, Naveen, Srihari Nelakuditi, and Romit Roy Choudhury (2008). "On Spatial Reuse and Capture in Ad Hoc Networks". In: *Proceedings of the 2008 IEEE Wireless Communications & Networking Conference.* WCNC 2008. (Las Vegas, NV, Mar. 31–Apr. 3, 2008). New York, NY, USA: IEEE, pp. 1628–1633. doi: `10.1109/WCNC.2008.291` (Cited on pp. 30, 32, 33).

Scarfone, Karen and Peter Mell (2007). *Guide to Intrusion Detection and Prevention Systems (IDPS).* NIST Special Publication 800-94. National Institute of Standards and Technology. url: `http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf` (Cited on p. 11).

Schmid, Thomas (2006). *GNU Radio 802.15.4 En- and Decoding.* Tech. rep. TR-UCLA-NESL-200609-06. Networked & Embedded Systems Laboratory, University of California at Los Angeles. url: `http://nesl.ee.ucla.edu/document/show/282` (Cited on pp. 37, 42, 90, 107).

Schmidt, Florian, Matteo Ceriotti, and Klaus Wehrle (2013). "Bit Error Distribution and Mutation Patterns of Corrupted Packets in Low-Power Wireless Networks". In: *Proceedings of the 8th ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization.* WiNTECH 2013. (Miami, FL, Sept. 30, 2013). New York, NY, USA: ACM, pp. 49–56. doi: `10.1145/2505469.2505475` (Cited on p. 31).

Sha, Mo, Guoliang Xing, Gang Zhou, Shucheng Liu, and Xiaorui Wang (2009). "C-MAC: Model-Driven Concurrent Medium Access Control for Wireless Sensor Networks". In: *Proceedings of the 28th IEEE International Conference* *on Computer Communications.* INFOCOM 2009. (Rio de Janeiro, Apr. 19–25, 2009). New York, NY, USA: IEEE, pp. 1845–1853. DOI: `10.1109/INFCOM.2009.5062105` (Cited on p. 30).

Sheng, Yong, Keren Tan, Guanling Chen, David Kotz, and Andrew Campbell (2008). “Detecting 802.11 MAC Layer Spoofing Using Received Signal Strength”. In: *Proceedings of the 27th IEEE International Conference on Computer Communications.* INFOCOM 2008. (Phoenix, AZ, Apr. 13–18, 2008). New York, NY, USA: IEEE, pp. 1768–1776. DOI: `10.1109/INFOCOM.2008.239` (Cited on p. 124).

Sheth, Anmol, Srinivasan Seshan, and David Wetherall (2009). “Geo-Fencing: Confining Wi-Fi Coverage to Physical Boundaries”. In: *Pervasive Computing: Proceedings of the 7th International Conference, Pervasive 2009.* Ed. by Hideyuki Tokuda, Michael Beigl, Adrian Friday, A. J. Bernheim Brush, and Yoshito Tobe. Lecture Notes in Computer Science 5538. Berlin, Germany: Springer Berlin Heidelberg, pp. 274–290. DOI: `10.1007/978-3-642-01516-8_19` (Cited on p. 123).

Son, Dongjin, Bhaskar Krishnamachari, and John Heidemann (2006). “Experimental Study of Concurrent Transmission in Wireless Sensor Networks”. In: *Proceedings of the 4th ACM Conference on Embedded Networked Sensor Systems.* SenSys 2006. (Boulder, CO, Oct. 31–Nov. 3, 2006). New York, NY, USA: ACM, pp. 237–250. DOI: `10.1145/1182807.1182831` (Cited on pp. 30, 53, 64).

Spuhler, Michael, Domenico Giustiniano, Vincent Lenders, Matthias Wilhelm, and Jens B. Schmitt (2014). “Detection of Reactive Jamming in DSSS-based Wireless Communications”. In: *IEEE Transactions on Wireless Communications* 13(3), pp. 1593–1603. DOI: `10.1109/TWC.2013.013014.131037`.

Tan, Kun, Jiansong Zhang, Ji Fang, He Liu, Yusheng Ye, Shen Wang, Yongguang Zhang, Haitao Wu, Wei Wang, and Geoffrey M. Voelker (2009). “Sora: High Performance Software Radio using General Purpose Multi-Core Processors”. In: *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation.* NSDI 2009. (Boston, MA, Apr. 22–24, 2009). Berkeley, CA, USA: USENIX Association, pp. 75–90. URL: `http://portal.acm.org/citation.cfm?id=1558977.1558983` (Cited on p. 81).

Tang, Xiaojun, Ruoheng Liu, Predrag Spasojević, and H. Vincent Poor (2011). “Interference Assisted Secret Communication”. In: *IEEE Transactions on Information Theory* 57(5), pp. 3153–3167. DOI: `10.1109/TIT.2011.2121450` (Cited on p. 10).

TEKIN, ENDER and AYLIN YENER (2008). “The General Gaussian Multiple-Access and Two-Way Wiretap Channels: Achievable Rates and Cooperative Jamming”. In: *IEEE Transactions on Information Theory* 54(6), pp. 2735–2751. DOI: `10.1109/TIT.2008.921680` (Cited on p. 10).

TIPPENHAUER, NILS OLE, LUKA MALISA, AANJHAN RANGANATHAN, and SRĐAN ČAPKUN (2013). “On Limitations of Friendly Jamming for Confidentiality”. In: *Proceedings of the 2013 IEEE Symposium on Security and Privacy.* S&P 2013. (San Francisco, CA, May 19–22, 2013). Washington DC, USA: IEEE Computer Society, pp. 160–173. DOI: `10.1109/SP.2013.21` (Cited on p. 11).

VILELA, JOÃO P., MATTHIEU BLOCH, JOÃO BARROS, and STEVEN W. MCLAUGHLIN (2011). “Wireless Secrecy Regions With Friendly Jamming”. In: *IEEE Transactions on Information Forensics and Security* 6(2), pp. 256–266. DOI: `10.1109/TIFS.2011.2111370` (Cited on p. 10).

VUTUKURU, MYTHILI, KYLE JAMIESON, and HARI BALAKRISHNAN (2008). “Harnessing Exposed Terminals in Wireless Networks”. In: *Proceedings of the 5^{th} USENIX Symposium on Networked Systems Design and Implementation.* NSDI 2008. (San Francisco, CA, Apr. 16–18, 2008). Berkeley, CA, USA: USENIX Association, pp. 59–72. URL: `http://dl.acm.org/citation.cfm?id=1387594` (Cited on p. 30).

WANG, YIN, YUAN HE, XUFEI MAO, YUNHAO LIU, ZHIYU HUANG, and XIANGYANG LI (2012). “Exploiting Constructive Interference for Scalable Flooding in Wireless Networks”. In: *Proceedings of the 31^{st} IEEE International Conference on Computer Communications.* INFOCOM 2012. (Orlando, FL, Mar. 25–30, 2012). New York, NY, USA: IEEE, pp. 2104–2112. DOI: `10.1109/INFCOM.2012.6195593` (Cited on pp. 30, 32, 34, 49, 55, 58, 65).

WANG, YIN, YUNHAO LIU, YUAN HE, XIANG-YANG LI, and DAPENG CHENG (2014). “Disco: Improving Packet Delivery via Deliberate Synchronized Contructive Interference”. In: *IEEE Transactions on Parallel and Distributed Systems* PP(99). To appear., p. 1 (Cited on pp. 30, 32, 50, 55).

WHITEHOUSE, KAMIN, ALEC WOO, FRED JIANG, JOSEPH POLASTRE, and DAVID CULLER (2005). “Exploiting the Capture Effect for Collision Detection and Recovery”. In: *Proceedings of the 2^{nd} IEEE Workshop on Embedded Networked Sensors.* EmNetS-II. (Sydney, May 30–31, 2005). New York, NY, USA: IEEE, pp. 45–52. DOI: `10.1109/EMNETS.2005.1469098` (Cited on pp. 31, 33).

WILHELM, MATTHIAS, VINCENT LENDERS, and JENS B. SCHMITT (2013a). *An Analytical Model of Packet Collisions in IEEE 802.15.4 Wireless Networks.* Tech. rep. Kaiserslautern, Germany: Dept. of Computer Science, TU Kaiserslautern. arXiv: `1309.4978 [cs.NI]`.

WILHELM, MATTHIAS, VINCENT LENDERS, and JENS B. SCHMITT (2014). “On the Reception of Concurrent Transmissions in Wireless Sensor Networks”. In: *IEEE Transactions on Wireless Communications.* DOI: `10.1109/TWC.2014.2349896`.

WILHELM, MATTHIAS, IVAN MARTINOVIC, JENS B. SCHMITT, and VINCENT LENDERS (2011a). “Short Paper: Reactive Jamming in Wireless Networks—How Realistic is the Threat?” In: *Proceedings of the 4th ACM Conference on Wireless Network Security.* WiSec 2011. (Hamburg, June 15–17, 2011). New York, NY, USA: ACM, pp. 47–52. DOI: `10.1145/1998412.1998422`.

– (2011b). “WiFire: A Firewall for Wireless Networks”. In: *Proceedings of the ACM SIGCOMM 2011 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* SIGCOMM 2011. (Toronto, Aug. 15–19, 2011). New York, NY, USA: ACM, pp. 456–457. DOI: `10.1145/2018436.2018518`.

– (2011c). “WiSec 2011 Demo: RFReact—A Real-time Capable and Channel-aware Jamming Platform”. In: *SIGMOBILE Mobile Computing and Communications Review* 15, pp. 41–42. DOI: `10.1145/2073290.2073300`.

– (2013b). *Air Dominance in Sensor Networks: Guarding Sensor Motes using Selective Interference.* Tech. rep. Kaiserslautern, Germany: Dept. of Computer Science, TU Kaiserslautern. arXiv: `1305.4038 [cs.NI]`.

WILHELM, MATTHIAS and JENS B. SCHMITT (2012). “Interference Scripting: Protocol-aware Interference Generation for Repeatable Wireless Testbed Experiments”. In: *Proceedings of the 4th Annual Wireless of the Student, by the Student, and for the Student Workshop.* S3 2012. (Istanbul, Aug. 22–26, 2012). New York, NY, USA: ACM, pp. 21–23.

WILHELM, MATTHIAS, JENS B. SCHMITT, and VINCENT LENDERS (2012). “Practical Message Manipulation Attacks in IEEE 802.15.4 Wireless Networks”. In: *Workshop Proceedings of the 16th International GI/ITG Conference on Measurement, Modelling, and Evaluation of Computing Systems* and *Dependability and Fault Tolerance.* MMB & DFT 2012. (Kaiserslautern, Mar. 21, 2012). Kaiserslautern, Germany, pp. 29–31 (Cited on p. 65).

WOOD, ANTHONY D., JOHN A. STANKOVIC, GILLES VIRONE, LEO SELAVO, ZHIMIN HE, QUIHUA CAO, THAO DOAN, YAFENG WU, LEI FANG, and RADU STOLERU (2008). “Context-Aware Wireless Sensor Networks for Assisted Living and Residential Monitoring”. In: *IEEE Network* 22(4), pp. 26–33. DOI: `10.1109/MNET.2008.4579768` (Cited on p. 2).

WRIGHT, JOSHUA (2010). *KillerBee—Framework and tools for exploiting ZigBee and IEEE 802.15.4 networks.* Retrieved from `http://code.google.com/p/killerbee`, March 2011. (Cited on p. 3).

WU, DINGMING, CHAO DONG, SHAOJIE TANG, HAIPENG DAI, and GUIHAI CHEN (2014). “Fast and Fine-grained Counting and Identification via Constructive Interference in WSNs”. In: *Proceedings of the 13th ACM/IEEE International Conference on Information Processing in Sensor Networks.* IPSN 2014. (Berlin, Apr. 15–17, 2014). New York, NY, USA: ACM, pp. 191–202. DOI: `10.1109/IPSN.2014.6846752` (Cited on p. 30).

WU, KAISHUN, HAOYU TAN, HOI-LUN NGAN, YUNHUAI LIU, and LIONEL M. NI (2012). “Chip Error Pattern Analysis in IEEE 802.15.4”. In: *IEEE Transactions on Mobile Computing* 11(4), pp. 543–552. DOI: `10.1109/TMC.2011.44` (Cited on p. 31).

WYNER, AARON D. (1975). “The Wire-Tap Channel”. In: *Bell System Technical Journal* 54(8), pp. 1355–1387. DOI: `10.1002/j.1538-7305.1975.tb02040.x` (Cited on p. 10).

XIONG, JIE and KYLE JAMIESON (2010). “SecureAngle: Improving Wireless Security using Angle-of-Arrival Information”. In: *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks.* HotNets-IX. (Monterey, CA, Oct. 20–21, 2010). New York, NY, USA: ACM, 11:1–11:6. DOI: `10.1145/1868447.1868458` (Cited on p. 124).

XU, FENGYUAN, ZHENGRUI QIN, CHIU C. TAN, BAOSHENG WANG, and QUN LI (2011). “IMDGuard: Securing Implantable Medical Devices with the External Wearable Guardian”. In: *Proceedings of the 30th IEEE International Conference on Computer Communications.* INFOCOM 2011. (Shanghai, Apr. 10–15, 2011). New York, NY, USA: IEEE, pp. 1862–1870. DOI: `10.1109/INFCOM.2011.5934987` (Cited on p. 13).

YUAN, DINGWEN and MATTHIAS HOLLICK (2013). “Let’s Talk Together: Understanding Concurrent Transmissions in Wireless Sensor Networks”. In: *Proceedings of the 38th IEEE Conference on Local Computer Networks.* LCN 2013. (Sydney, Oct. 21–24, 2013). New York, NY, USA: IEEE, pp. 219–227. DOI: `10.1109/LCN.2013.6761237` (Cited on p. 31).

ZHAO, JERRY and RAMESH GOVINDAN (2003). “Understanding Packet Delivery Performance in Dense Wireless Sensor Networks”. In: *Proceedings of the 1st ACM Conference on Embedded Networked Sensor Systems.* SenSys 2003. (Los Angeles, CA, Nov. 5–7, 2003). New York, NY, USA: ACM, pp. 1–13. DOI: `10.1145/958491.958493` (Cited on p. 31).

ZIMMERLING, MARCO, FEDERICO FERRARI, LUCA MOTTOLA, and LOTHAR THIELE (2013). “On Modeling Low-Power Wireless Protocols Based on Synchronous Packet Transmissions”. In: *Proceedings of the IEEE 21st International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems.* MASCOTS 2013. (San Francisco, CA, Aug. 14–16, 2013). New

York, NY, USA: IEEE, pp. 546–555. DOI: `10.1109/MASCOTS.2013.76` (Cited on p. 31).

Curriculum Vitae

Personal Information

Name Matthias Stephan Wilhelm

Nationality German

Professional Experience

2015–2016 Researcher at TOYOTA InfoTechnology Center Co., Ltd. in Tokyo, Japan

Education

2009–2015 Ph.D. student at the Department of Computer Science, Distributed Computer Systems Group, Technical University of Kaiserslautern Germany

2002–2009 Studies at the Technical University Kaiserslautern
Degree: Diplom Technoinformatik (Dipl. Technoinform.)
Thesis: "Design and Implementation of a Key Generation Protocol for Wireless Sensor Networks"

1992–2001 Gauß-Gymnasium, Worms, Germany

1988–1992 Otto-Hahn-Grundschule, Westhofen, Germany

Publication List

Journal Articles

Martinovic, Ivan, Paul Pichota, Matthias Wilhelm, Frank A. Zdarsky, and Jens B. Schmitt (2009). “Bringing Law and Order to IEEE 802.11 Networks—A Case for DiscoSec”. In: *Pervasive and Mobile Computing* 5(5), pp. 510–525. doi: 10.1016/j.pmcj.2009.03.002.

Spuhler, Michael, Domenico Giustiniano, Vincent Lenders, Matthias Wilhelm, and Jens B. Schmitt (2014). “Detection of Reactive Jamming in DSSS-based Wireless Communications”. In: *IEEE Transactions on Wireless Communications* 13(3), pp. 1593–1603. doi: 10.1109/TWC.2013.013014.131037.

Wilhelm, Matthias, Vincent Lenders, and Jens B. Schmitt (2014). “On the Reception of Concurrent Transmissions in Wireless Sensor Networks”. In: *IEEE Transactions on Wireless Communications.* doi: 10.1109/TWC.2014.2349896.

Wilhelm, Matthias, Ivan Martinovic, and Jens B. Schmitt (2013). “Secure Key Generation in Sensor Networks Based on Frequency-selective Channels”. In: *IEEE Journal on Selected Areas in Communications* 31(8), pp. 1779–1790. doi: 10.1109/JSAC.2013.130911.

Wilhelm, Matthias, Ivan Martinovic, Jens B. Schmitt, and Vincent Lenders (2011c). “WiSec 2011 Demo: RFReact—A Real-time Capable and Channel-aware Jamming Platform”. In: *SIGMOBILE Mobile Computing and Communications Review* 15, pp. 41–42. doi: 10.1145/2073290.2073300.

Refereed Papers in Conference Proceedings

Eberz, Simon, Martin Strohmeier, Matthias Wilhelm, and Ivan Martinovic (2012). “A Practical Man-In-The-Middle Attack on Signal-Based Key Generation Protocols”. In: *Computer Security – ESORICS 2012: Proceedings of the 17th European Symposium on Research in Computer Security.* Ed. by Sara Foresti, Moti Yung, and Fabio Martinelli. Lecture Notes in Computer

Science 7459. Berlin, Germany: Springer Berlin Heidelberg, pp. 235–252. doi: `10.1007/978-3-642-33167-1_14`.

Giustiniano, Domenico, Vincent Lenders, Jens B. Schmitt, Michael Spuhler, and Matthias Wilhelm (2013). "Detection of Reactive Jamming in DSSS-based Wireless Networks". In: *Proceedings of the 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks.* WiSec 2013. (Budapest, Apr. 17–19, 2013). New York, NY, USA: ACM, pp. 43–48. doi: `10.1145/2462096.2462104`.

Martinovic, Ivan, Paul Pichota, Matthias Wilhelm, Frank A. Zdarsky, and Jens B. Schmitt (2008). "Design, Implementation, and Performance Analysis of DiscoSec—Service Pack for Securing WLANs". In: *Proceedings of the 9th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks.* WoWMoM 2008. (Newport Beach, CA, June 23–26, 2008). New York, NY, USA: IEEE, pp. 1–10. doi: `10.1109/WOWMOM.2008.4594831`.

Martinovic, Ivan, Frank A. Zdarsky, Matthias Wilhelm, Christian Wegmann, and Jens B. Schmitt (2008). "Wireless Client Puzzles in IEEE 802.11 Networks: Security by Wireless". In: *Proceedings of the 1st ACM Conference on Wireless Network Security.* WiSec 2008. (Alexandria, VA, Mar. 31–Apr. 2, 2008). New York, NY, USA: ACM, pp. 36–45. doi: `10.1145/1352533.1352541`.

Schäfer, Matthias, Martin Strohmeier, Vincent Lenders, Ivan Martinovic, and Matthias Wilhelm (2014a). "Bringing up OpenSky: A Large-scale ADS-B Sensor Network for Research". In: *Proceedings of the 13th ACM/IEEE International Conference on Information Processing in Sensor Networks.* IPSN 2014. (Berlin, Apr. 15–17, 2014). New York, NY, USA: ACM, pp. 83–94. doi: `10.1109/IPSN.2014.6846743`.

– (2014b). "Demo Abstract: OpenSky – A Large-scale ADS-B Sensor Network for Research". In: *Proceedings of the 13th ACM/IEEE International Conference on Information Processing in Sensor Networks.* IPSN 2014. (Berlin, Apr. 15–17, 2014). New York, NY, USA: ACM, pp. 313–314. doi: `10.1109/IPSN.2014.6846779`.

Wilhelm, Matthias, Ivan Martinovic, and Jens B. Schmitt (2009a). "Lightweight Key Generation based on Physical Properties of Wireless Channels". In: *Proceedings of the 11. Kryptotag der Gesellschaft für Informatik e. V.* (Trier, Nov. 30, 2009). Ed. by Ralf Küsters. Trier, Germany: University of Trier, p. 4.

– (2009b). "On Key Agreement in Wireless Sensor Networks based on Radio Transmission Properties". In: *Proceedings of the 5th Annual Workshop on Secure Network Protocols.* NPSec 2009. (Princeton, NJ, Oct. 13, 2009). New York, NY, USA: IEEE, pp. 37–42. doi: `10.1109/NPSEC.2009.5342245`.

– (2010b). “Secret Keys from Entangled Sensor Motes: Implementation and Analysis”. In: *Proceedings of the 3rd ACM Conference on Wireless Network Security.* WiSec 2010. (Hoboken, NJ, Mar. 22–24, 2010). New York, NY, USA: ACM, pp. 139–144. DOI: `10.1145/1741866.1741889`.

WILHELM, MATTHIAS, IVAN MARTINOVIC, JENS B. SCHMITT, and VINCENT LENDERS (2011a). “Short Paper: Reactive Jamming in Wireless Networks—How Realistic is the Threat?” In: *Proceedings of the 4th ACM Conference on Wireless Network Security.* WiSec 2011. (Hamburg, June 15–17, 2011). New York, NY, USA: ACM, pp. 47–52. DOI: `10.1145/1998412.1998422`.

– (2011b). “WiFire: A Firewall for Wireless Networks”. In: *Proceedings of the ACM SIGCOMM 2011 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* SIGCOMM 2011. (Toronto, Aug. 15–19, 2011). New York, NY, USA: ACM, pp. 456–457. DOI: `10.1145/2018436.2018518`.

WILHELM, MATTHIAS, IVAN MARTINOVIC, ERSIN UZUN, and JENS B. SCHMITT (2010). “SUDOKU: Secure and Usable Deployment of Keys on Wireless Sensors”. In: *Proceedings of the 6th Annual Workshop on Secure Network Protocols.* NPSec 2010. (Tokyo, Oct. 5, 2010). New York, NY, USA: IEEE, pp. 1–6. DOI: `10.1109/NPSEC.2010.5634458`.

WILHELM, MATTHIAS and JENS B. SCHMITT (2012). “Interference Scripting: Protocol-aware Interference Generation for Repeatable Wireless Testbed Experiments”. In: *Proceedings of the 4th Annual Wireless of the Student, by the Student, and for the Student Workshop.* S3 2012. (Istanbul, Aug. 22–26, 2012). New York, NY, USA: ACM, pp. 21–23.

WILHELM, MATTHIAS, JENS B. SCHMITT, and VINCENT LENDERS (2012). “Practical Message Manipulation Attacks in IEEE 802.15.4 Wireless Networks”. In: *Workshop Proceedings of the 16th International GI/ITG Conference on Measurement, Modelling, and Evaluation of Computing Systems* and *Dependability and Fault Tolerance.* MMB & DFT 2012. (Kaiserslautern, Mar. 21, 2012). Kaiserslautern, Germany, pp. 29–31.

Technical Reports

WILHELM, MATTHIAS, VINCENT LENDERS, and JENS B. SCHMITT (2013a). *An Analytical Model of Packet Collisions in IEEE 802.15.4 Wireless Networks.* Tech. rep. Kaiserslautern, Germany: Dept. of Computer Science, TU Kaiserslautern. arXiv: `1309.4978 [cs.NI]`.

WILHELM, MATTHIAS, IVAN MARTINOVIC, and JENS B. SCHMITT (2010a). *Key Generation in Wireless Sensor Networks Based on Frequency-selective Channels*

– *Design, Implementation, and Analysis.* Tech. rep. Kaiserslautern, Germany: Dept. of Computer Science, TU Kaiserslautern. arXiv: `1005.0712 [cs.CR]`.

Wilhelm, Matthias, Ivan Martinovic, Jens B. Schmitt, and Vincent Lenders (2013b). *Air Dominance in Sensor Networks: Guarding Sensor Motes using Selective Interference.* Tech. rep. Kaiserslautern, Germany: Dept. of Computer Science, TU Kaiserslautern. arXiv: `1305.4038 [cs.NI]`.