NORTHEASTERN UNIVERSITY
GRADUATE SCHOOL OF COMPUTER SCIENCE
PH.D. THESIS APPROVAL FORM

THESIS TITLE: Robust Wireless Communication for
Multi-Antenna, Multi-Rate, Multi-Carrier Systems

AUTHOR: Triet Dang Vo-Huu

Ph.D. Thesis Approved to complete all degree requirements for the Ph.D. Degree in Computer Science

Professor Guevara Noubir, Thesis Advisor — 10/8/2015 Date

Professor Erik-Oliver Blass, Airbus Group Innovations/Northeastern University — 8/25/15 Date

Professor Rajmohan Rajaraman, Northeastern University — 10/8/15 Date

Professor Srdjan Capkun, ETH Zurich — 8/27/15 Date

D. Starobinski

Professor David Starobinski, Boston University — 09/18/15 Date

GRADUATE SCHOOL APPROVAL:

Director, Graduate School — 10/8/2015 Date

COPY RECEIVED IN GRADUATE SCHOOL OFFICE:

Recipient's Signature — 10/23/2015 Date

Distribution: Once completed, this form should be scanned and attached to the front of the electronic dissertation document (page 1). An electronic version of the document can then be uploaded to the Northeastern University-UMI website.

Ph.D. Dissertation

Robust Wireless Communication for Multi-Antenna, Multi-Rate, Multi-Carrier Systems

Triet Dang Vo-Huu

College of Computer and Information Science

Northeastern University

Ph.D. Committee

Professor Guevara Noubir	Advisor, Northeastern University
Professor Erik-Oliver Blass	Airbus Group Innovations / Northeastern University
Professor Rajmohan Rajaraman	Northeastern University
Professor Srdjan Capkun	External member, ETH Zurich
Professor David Starobinski	External member, Boston University

June 2015

Abstract

Today's trend of migrating radio devices from hardware to software provides potentials to create flexible applications for both commercial and military use. However, this raises security concerns, as malicious attacks can also be alleviated to break legitimate communications. In this research work, we study and build practical systems to mitigate the impact of some serious jamming threats: high-power jamming, rate-based attacks, and jamming in multicarrier systems.

First, we develop SAIM – a hybrid system of mixed mechanical steerable antenna and software jamming cancellation – to counter high-power jamming adversary. The system robustness relies on a new antenna structure we specially design for anti-jamming purpose, and a set of new algorithms that can adaptively and effectively mitigate the jamming signal up to 100,000 times higher than legitimate signals. Our solution is appropriate for malicious environments with powerful jammers, where mechanical steering is feasible, e.g., military applications.

In residential wireless deployments, even limited-power rate adaptation attacks can dramatically affect the system performance. To tackle this problem, we develop CBM for hiding rate information, and – at the same time – increasing resiliency against jammers up to seven times higher than regular rate-exposing systems. The resiliency boost is achieved by our new generalized non-uniform Trellis Coded Modulation codes, while the modulation and code information is concealed by a new efficient method for cryptographically interleaving baseband symbols.

As a third part of this work, we investigate the jamming impact on multicarrier systems with focus on Wi-Fi communications. Toward this goal, we develop the first open source software defined radio Wi-Fi stack that can operate at high rates and allow detailed analysis from Physical Layer to Medium Access Control and Link Layers. Using our platform, we analyze the interleaving process specified by IEEE 802.11 and exploit it to devise an efficient Interleaving Jamming strategy that can totally block the Wi-Fi communications with jamming power less than 1% of the regular transmitted signal power. Our jamming strategy is at least 5dB and up to 15dB more power-efficient than those that are unaware of the Wi-Fi interleaving structure.

Acknowledgements

First of all, I would like to express my deep gratitude to my Ph.D. advisor, Professor Guevara Noubir, who has continuously been supporting me during my Ph.D. work and inspiring me to tackle challenging research problems. He has not only brought to me great high-level ideas but also walked me through deep technical details. Besides work, he is also the one I always trust and feel comfortable to discuss and share my thoughts on non-technical things. To be advised by Guevara is the best luck in my academic life.

I am especially grateful to Bishal Thapa, Tao Jin and Professor Erik-Oliver Blass who have been giving invaluable experiences and advices, spending days and nights with me on discussing new ideas and solutions. I would like to sincerely thank Professor Rajmohan Rajaraman, Professor Srdjan Capkun, and Professor David Starobinski for being in my Ph.D. committee, taking their invaluable time for evaluating my work and providing thoughtful and helpful comments.

My warm thanks go to my friends Abhishek Samanta, Aldo Cassola, Amirali Sanatinia, Sevtap Duman and others in the Secure Systems Lab for everything they did for me, and for moments we shared and spent together.

Finally, I want to dedicate this dissertation to my parents Võ Hữu Thảo and Nguyễn Ngọc Hạnh, my wife Nguyễn Ngọc Minh Châu, and my younger brother Võ Hữu Đăng Tiến who unconditionally love me and have been constantly supporting me during my Ph.D. work.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Over the last decades, wireless communication has proved to be an enabling technology for an increasingly large number of applications. The convenience of wireless and its support of mobility has revolutionized the way we access data, information services, and interact with the physical world. Beyond enabling mobile devices to access information and data services ubiquitously, it is today widely used in cyber-physical systems such as air-traffic control [126], power plants synchronization, transportation systems, and human body implantable devices [44]. For example, the United States Congress recently passed an FAA bill that speeds up the switching to GPS-based air traffic control [75]. The trend of wireless communication utilization in the electricity grid is already visible with over 20 millions smart meters already installed in the US and over 70 million worldwide [87]. Wireless Remote Terminal Units (W-RTU) with long-range wireless communication capabilities have been used for many years and several companies are increasingly switching to Wireless RTUs, e.g., vMonitor [115], Industrial Control Links [52], Synetcom [109], and Semaphore [101]. This pervasiveness elevated wireless communication systems to the level of critical infrastructure. The broadcast nature of wireless makes it vulnerable to jamming attacks and information leakage.

Jamming is a prominent security threat, as it cannot only lead to denial of service attacks, but can also be the prelude to sophisticated spoofing attacks against cellular, WiFi, and GPS system [23, 63, 100]. For example, an adversary can make a cellular network disappear and spoof it by a rogue network (usually a downgraded 2G network that does not have the appropriate mutual-authentication mechanisms but is still accepted by today's deployed devices). Another example consists of jamming the GPS signals and replacing them with a stronger replayed version. Beyond corrupting the location information, which can have severe impact on air traffic control, this can also stealthily corrupt time synchronization, which is critical for controlling electricity flow in power grids [103, 124, 127]. This jamming/replay attack applies not only to commercial grade GPS but also military ones. With the fast growth of both hardware and software-defined radios along with the spectrum becoming a scarce resource, jamming recently not only regained interest in the wireless security community, but also caused sufficient concerns to trigger an FCC campaign to enforce anti-jamming laws as stated by the chief of the FCC's Enforcement Bureau on February 2011: "Jamming devices create serious safety risks. In the coming weeks and months, we'll be intensifying our efforts through partnerships with law enforcement agencies to crack down on those who continue to violate the law" [31, 70]. Evidence of such attacks in the real world started emerging in the last few months, e.g., personal Wi-Fi hotspots are jammed [32]; or many rogue base stations are discovered around the United States which can potentially force user phone devices to associate to themselves, turn down the encryption mechanism and eventually download malicious applications to the devices [100].

In this research work, we are studying the impact of some serious jamming threats: high-power jamming, rate-based jamming, and jamming in multicarrier systems. To counter high-power jamming, we design a new antenna system with adaptive controlling capability combined with a novel digital cancellation technique that can harden the communication system by reducing the jamming power by up to 50dB ($100,000$ times). To mitigate rate-based jamming attacks, we develop a new algorithm to discover a new class of non-uniform Trellis Coded Modulation codes, and by using these codes together with our efficient cryptographic interleaving technique, we can add up to 7dB of robustness to existing systems. Finally, we show that one can devise a very efficient jamming attack based on the knowledge of Wi-Fi interleaving structure. Without appropriate protection mechanisms, our jamming strategy can dramatically degrade the whole network performance at very low cost, at least 5dB and up to 15dB more power-efficient than those unaware of the Wi-Fi interleaving structure. We verify our solutions in practice by building the prototypes on software-defined radio framework.

1.1 High-Power Jamming

1.1.1 Motivation

Jamming with high power is today realistic with jamming hardware against GPS, Cellular Systems, and WiFi already available on the Internet for few tens of dollars. More powerful jammers can also easily be made given that they do not necessitate to generate precise, clean RF signals. For example, various website (e.g., YouTube) have online tutorials to build High Energy RF (HREF) guns from a $50 microwave oven's magnetron that can generate a 1KWatt interfering signal (covering hundreds of meters) and can be tuned to a wide range of frequencies by slightly modifying its resonant cavity [19, 83]. In contrast, the communication nodes' regulated power is usually limited to 100mW. This type of jamming can completely block the ongoing transmission between communication nodes.

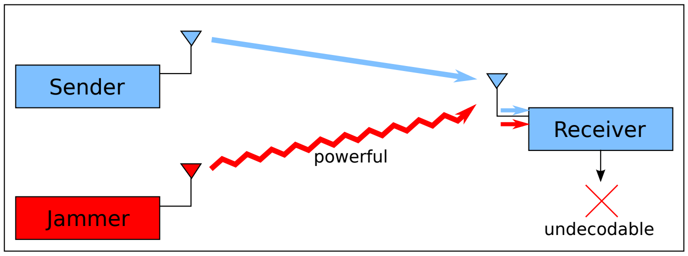


Figure 1.1: Traditional system under jamming attack: Data is not decoded correctly at the receiver due to low SNR.

An illustration of high-power jamming attack is shown in Figure 1.1, where the receiver cannot decode the received signal due to powerful interference generated by the jammer, which results in low received SNR (Signal-to-Noise Ratio) preventing data to be recovered.

1.1.2 Proposed solution

We develop SAIM (Steerable-separable Antenna for Interference Mitigation) system for countering high-power jamming attacks. SAIM is a hybrid framework consists of two stages of operation:

First stage – Antenna Auto-Configuration. (Figure 1.2) We introduce a novel two-element antenna that dynamically reconfigures to track the jammer and to weaken its signal by up to 28 dB (nearly 640 times). Our new design with two moving elements is *simple*, *low-cost*, and has unique characteristics unexplored in mechanically steerable antennas. Our configuration algorithm allows to converge on element separation/rotation that maximizes the SINR within 20 seconds.

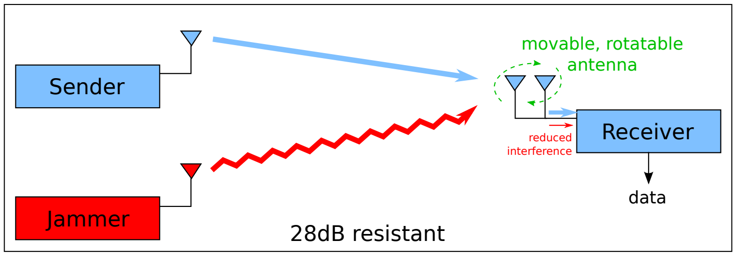


Figure 1.2: SAIM's 1st stage: Two-element Antenna Control.

Second stage – Digital Jamming Cancellation. (Figure 1.3) To further mitigate the interference, we also use a single-element antenna to get an additional copy of the jamming signal and develop a MIMO-like interference cancellation technique tailored for anti-jamming. Unlike traditional MIMO and beam-forming techniques we do not rely on training sequences. We demonstrate a reliable communication equivalent to reducing the jamming impact by 48 dB (nearly 64,000 times).

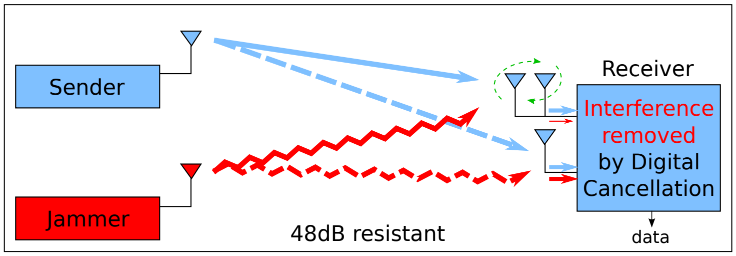


Figure 1.3: SAIM's 2nd stage: Digital Cancellation.

Hybrid system. To summarize, our hybrid SAIM system is capable of:

- *High-power anti-jamming:* We are able to efficiently remove unknown jamming signals up to 48 dB (almost five orders) of power higher than legitimate user's signals and recover the user data with an acceptable bit error rate.
- *Zero-knowledge anti-jamming:* We neither require knowledge about the legitimate signals (no additional preamble, no training sequence), nor knowledge about the jammer (unknown location, variable jamming power).
- *Environment adaptiveness:* The system works efficiently in both outdoor as well as indoor environments and can handle multipath jamming.

1.2 Jamming on Link Rate Adaptivity

1.2.1 Motivation

Although high-power jamming is extremely effective against wireless communications, the limitation of this type of jamming is that it requires powerful interference generating sources, which are not always feasible in practice. Moreover, powerful jammers can be detected with high probability and eventually physically removed from the network. This motivates a more complicated jamming technique, which exploits the exposing of control information in wireless protocols. In particular, knowing the rate being used by the communication link can lead to a very efficient attack for the adversary. Most of wireless systems can only operate reliably at a bit error rate of 10^{-6} or below (for instance, a TCP packet of typical size 1440 bytes can only be transmitted at a success probability of 99% if the bit error rate of the channel is under 10^{-6}), at which, a communication using 64-QAM modulation will require the transmitter to transmit at a power 18dB (60 times) higher than the noise level (Figure 1.4). This implies that an adversary only needs to use a jamming power of about 60 times lower than the transmitter's power to make the communication unreliable. In contrast, a BPSK communication requires a stronger adversary to achieve the same jamming impact.

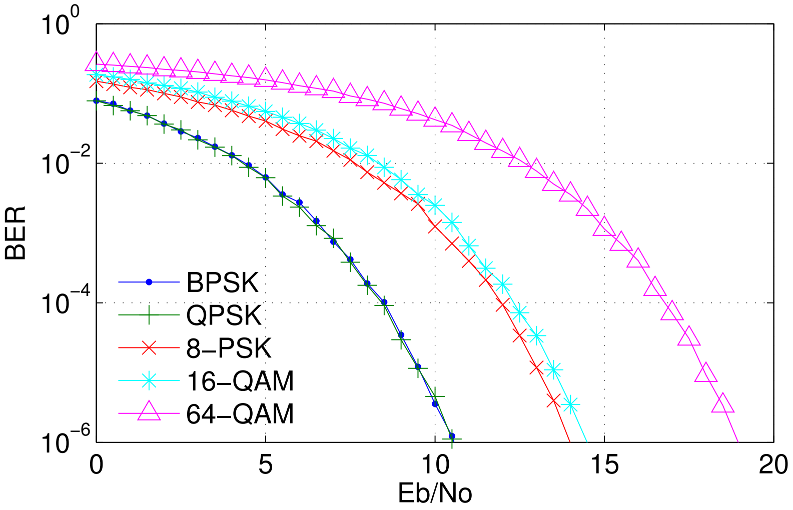


Figure 1.4: Simulation results show bit error rates of communications with various modulations versus normalized SNR (E_b/N_0). Lower data rates (lower-order modulation) are more resilient against jamming.

Recent work [74] showed that knowledge of the transmission rate enables selective jamming of packets resulting in very efficient attacks on all the Wi-Fi rate adaptation protocols investigated. Rate Adaptation Algorithms (RAA) adjust the physical layer transmission rate to the channel characteristics; ideally selecting a low rate MCS for a low SNR channel and a high rates for high SNR [11, 51, 58, 61, 68, 71, 79, 117, 120]. To illustrate the criticality of hiding the communication rate, we first provide some insights into rate adaptation attacks. The key idea in the highly efficient RAA-attack is to force a device to use a low rate (e.g., 1Mbps), by jamming all packets of higher rates. This allows: (1) a reflection attack where the victim is occupying the channel for very long time preventing other transmissions (i.e., 54 times channel occupancy for 802.11g to over 300 times for 802.11n), (2) a saturation of the network that induces a higher collision probability resulting in some RAA maintaining a network-wide low rate even when the adversary stops jamming. This self-sustained interference is called a congestion collapse. The RAA-attack is aggravated by the fact that jamming high rate packets is easier than jamming low rate packets and only requires interfering with few symbols. Furthermore, the comeback to higher rates is slow. Finally, the equiprobability of transmission among devices enables the adversary to focus on a single link to degrade the whole network [21]. Previous work [74, 81, 86] demonstrated both analytically and experimentally that knowledge of the rate used in a transmission enables selective jamming of packets resulting in link degradation (e.g., from 54 Mbps to 1 Mbps), and it also blocks other devices and causes high collisions provoking a long-lasting network-wide congestion collapse. The adversary only needs to jam a small carefully selected fraction (less than 5%) of the packets to achieve orders of magnitude more efficiency than blind jamming [74]. In summary, rate-based attacks can dramatically degrade the performance of the whole wireless communication network. This motivates us to investigate the countermeasure proposed in the next section.

1.2.2 Proposed solution

We introduce CBM (Conceal and Boost Modulation), an integrated solution to conceal the rate information of wireless transmissions while simultaneously boosting the communication resiliency against interference. The main components of our CBM system are depicted in Figure 1.5.

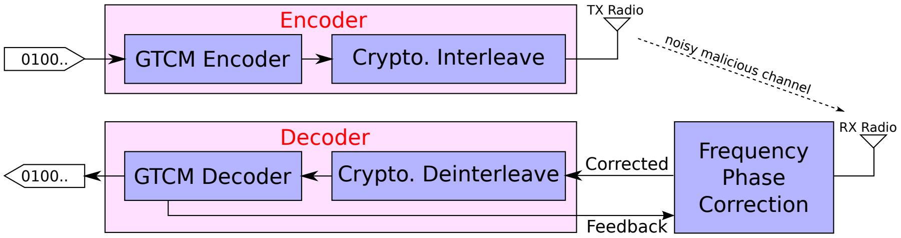


Figure 1.5: CBM's main components.

The proposed solution is based on a generalization of Trellis Coded Modulation combined with Cryptographic Interleaving. We developed efficient algorithms for discovering and validating new trellis codes capable of upgrading any modulation constellation in {BPSK, QPSK, 8-PSK, 16-QAM, 64-QAM} to any higher order modulation. We relax the uniformity and devise explicit codes with higher coding gain than traditional codes conforming to uniformity. By Cryptographic Interleaving, CBM also helps mitigating passive attacks against users traffic analysis [3]. Our simulation results show that in most cases this modulation hiding scheme has the side effect of boosting resiliency by up to seven times in comparison to regular rate-exposing systems.

However, in real experiments using a testbed of USRP N210 [30], we show that with a standard frequency offset correction and phase tracking, the performance of the proposed schemes is severely degraded. To preserve the robustness of the system, we develop a 2-pass frequency offset correction and phase tracking mechanisms that integrates within the receiver. Our final system achieves an experimentally measured performance boost of up to 4dB in addition to concealing the communication rate, and therefore mitigating rate attacks.

1.3 Jamming in Multicarrier System

1.3.1 Motivation

With the success of today's multicarrier systems such as 3G, LTE, Wi-Fi, the Orthogonal Frequency Division Multiplexing (OFDM) and Multiple Input Multiple Output (MIMO) become the most promising techniques for very high throughput future wireless systems. However, this also raises new challenges to network operators as the wireless medium exposes more to the adversaries.

In this work, we focus on the robustness of Wi-Fi (IEEE 802.11) against jamming attacks. Wi-Fi is emerging as the primary medium for wireless Internet access. Cellular carriers are increasingly offloading their traffic to Wi-Fi Access Points (APs) to overcome capacity challenges, limited RF spectrum availability, cost of deployment, and keep up with the traffic demands driven by user generated content. Wi-Fi Offloading is facilitated by 3GPP standards for Non-3GPP Access Networks Discovery and Roaming [1], IETF seamless USIM-based strong authentication and secure communication protocols such as EAP-SIM/AKA [40, 50], and IEEE seamless handover and authentication protocols across networks IEEE802.11u [53]. Studies forecast a sustained 50% yearly growth in Wi-Fi offloading for many years to come [89, 91, 97]. This trend is paved by the increasing deployments of Hotspot 2.0 (HS2) Access Points enabled by seamless handover across networks implementing the IEEE802.11u amendment [53]. Moreover, manufacturers of laptops and streaming devices, such as the Apple MacBook Pro and the Roku streaming player, are removing Ethernet ports and entirely relying on Wi-Fi, and several new variants of Wi-Fi are being developed to suit different environments (e.g., IEEE 802.11p for vehicular networking and IEEE 802.11af for TV white spaces).

Due to quick adoption of Wi-Fi, the repetition of Wi-Fi regains interests in the research community. However, despite the central role of Wi-Fi in today's wireless communications systems, there is still no platform available to the research community to analyze Wi-Fi networks from the physical layer to the network layer with the precision that it deserves. Previous work's results on the robustness of Wi-Fi communications, especially on low-level details in the Physical Layer, are mostly achieved via simulation tools rather than practical experiments. The impact of jamming on control and synchronization mechanisms of practical Wi-Fi systems, therefore, remains unclear.

Over the last few years researchers made great progress in characterizing Wi-Fi networks by extrapolating information provided by the drivers of commodity Wi-Fi cards. These systems supported by clever algorithms were able to infer surprising properties, sometimes in relatively larger scale setups [84, 93, 94]. However, they remain intrinsically limited by the information provided by commodity cards drivers (e.g., RSSI per carrier with a ms granularity). Precise information about the timing of packets, backoff, collisions, packet capture effects, hidden terminals, fine grain state of the channel, interference from smart misbehaving devices and stealthy jamming remain mostly invisible through the driver's API eye.

While focusing on commodity Wi-Fi cards for analyzing the RF spectrum was partially a deliberate choice because of their ubiquity, low-cost, ease of programming, and large scale deployments, other researchers encouraged by the availability of software defined radio (SDR) platforms, endeavored in the task of developing a Wi-Fi protocol stack from the physical to the link layer. Since the open source implementation of IEEE802.11 by BBN (operating at 1 & 2Mbps with downsampled baseband signals to 4MHz) [6], several efforts have been made to enable Wi-Fi communications on low-cost SDR platforms such as the Utah University FPGA-assisted implementation [34], or the more recent attempts [14, 15]. However, to the best of our knowledge none of these platforms exceeds QPSK modulation in terms of design and have very limited performance even for these low rates. A unique characteristic of higher order modulations 16-QAM and 64-QAM is that they not only require phase but also amplitude correction, which significantly more challenging as we will discuss in the next sections. Commercial platforms despite their high cost do not provide a complete Wi-Fi solution. An illustration of this is National Instruments' (USRP's mother company) 802.11 Application Framework priced at $5K [72].

1.3.2 Contributions

Motivated by the potential of a Wi-Fi SDR in enabling wireless research, and the expanding offerings of hardware SDR platforms, we developed SWiFi. To the best of our knowledge, it is the first implementation that can transmit & receive all IEEE802.11abg packets (reaching 54Mbps rates with QAM64 modulation and with 3/4 coding rate). We note that SWiFi is limited by the hardware SDR delay, and does not run within the SDR FPGA. Therefore it cannot send ACK packets within the SIFS interval. It is however designed with the goal to enable the analysis of Wi-Fi networks in ways not possible before. Furthermore, despite the timing constraints that prevents it from engaging in a unicast communication with other devices, it can still send broadcast packets, spoof, and decode any IEEE802.11abg packets.

Beyond the substantial amount of work spent in carefully implementing and testing all the mechanisms of the abg physical layer, the most challenging component was to devise novel techniques for Frequency Offset correction, and Frequency Domain Equalization that overcome the limited quality wideband RF Front End of SDR platforms. We successfully tested our implementation on two types of SDR platforms the mid-range USRP 2 & N210, and the low cost $275 HackRF [46]. SWiFi on these platforms compares favorably with commercial cards. However, as we will discuss in the evaluation section, the HackRF requires a firmware reprogramming to sustain the Wi-Fi necessary sampling rate (20Msps).

We believe that SWiFi will enable research at a new level. To support our claims, we developed some preliminary companion tools that enable the analysis of 802.11 MAC timing (SIFS), and 802.11 links (rate adaptation). Using our SWiFi platform, we also devise an efficient jamming strategy targeting to the interleaving process employed by IEEE 802.11 that is much more efficient than a blind jamming strategy.

Our main contribution can be summarized as follows.

- The first Wi-Fi SDR implementation capable of decoding high order modulation packets with new algorithms for frequency domain equalization.
- A rigorous and comprehensive evaluation demonstrating performance at least similar to commercial Wi-Fi cards.
- Preliminary tools for analyzing Wi-Fi networks using SWiFi.
- An efficient Interleaving Jamming strategy that is at least 5dB and up to 15dB more power-efficient than jamming unaware of interleaving structure. Our jamming strategy can also corrupt all Wi-Fi packets by using a low jamming power equal to only 1% of Wi-Fi communications' signal power. Moreover, the jamming power can be further reduced 10 times (0.1% of regular transmit power) while still causing 99% of packets incorrectly decoded.

1.4 Outline

The rest of this dissertation consists of five chapters. In Chapter 2, I describe our SAIM solution for countering high-power jamming attacks. Chapter 3 discusses the techniques used by our CBM system for mitigating the rate-based attacks. Prior to discussing the jamming attacks on Wi-Fi multicarrier systems, I present our SWiFi platform in Chapter 4. The discussion of efficient jamming strategy specifically to IEEE 802.11 communications is detailed in Chapter 5. Finally, future work is briefly discussed in Chapter 6.

Chapter 2

SAIM – Countering High-Power Jamming

In this chapter, we discuss our Steerable-separable Antenna for Interference Mitigation (SAIM) solution for countering high-power jamming attacks, where we assume that the spread spectrum and coding gain are not sufficient to counter the jammer. We focus our study on communications for a fairly narrowband (few MHz), where mechanical steering components are possible as is the case on many military vehicles or as widely used around the world in motorized dish antennas. Our system combines a novel mechanical beamforming design with a fast auto-configuration algorithm and a software radio digital interference cancellation algorithm. We start the discussion with a brief overview of our model and notations in Section 2.1. The key idea of our approach is presented in Section 2.2, and SAIM's details are described in Sections 2.3 and 2.4. Our fully functional protype is built based on GNU Radio platform with Ettus USRP devices. The implementation and building components are detailed in Section 2.5. We report the performance results in Section 2.6. We discuss the related work in Section 2.7 and Section 2.8 concludes our work.

2.1 Model

2.1.1 Communication and Adversarial Model

Our communication setup consists of two legitimate users and one adversary. We denote $s(t)$ as the signal containing the sender's modulated data. The sender's signal is sent at a transmit power $P_s = E[|s(t)|^2]$, where $E[\cdot]$ represents the expected (average) power over a unit time. Due to channel variations during signal propagation, the signal arrives at the receiver as

$$r(t) = s(t)h_s(t) + w(t)$$

Table 2.1: Notations used in SAIM's model.

$s(t)$	data signal of power P_s
$j(t)$	jamming signal of power P_j
$r(t)$	received signal at the receiver
$h_s(t)$	sender-receiver channel characteristics
$h_j(t)$	jammer-receiver channel characteristics

where $h_s(t)$ describes the channel between the sender and receiver, which includes the signal attenuation and phase variations, and $w(t)$ represents the additive white Gaussian noise (AWGN). When the AWGN is sufficiently small (i.e., good environment conditions), $w(t)$ is negligible and $s(t)$ can be recovered from $r(t)$ by the receiver estimating $h_s(t)$ with traditional channel estimation techniques.

In the presence of mlicious interference signal $j(t)$ generated by the jammer, the received signal becomes

$$r(t) = s(t)h(t) + j(t)h_j(t). \tag{2.1}$$

where $h_j(t)$ denotes the channel between the jammer and the receiver. In the model of high-power jamming attacks, the jamming signal power $P_j = E[|j(t)|^2]$ is much higher than legitimate user's signal power ($P_j \gg P_s$), preventing recovery of the original signal $s(t)$. Note that we already omitted $w(t)$ from Equation (2.1) as it is negligible to the malicious interference signal $j(t)$. In our model, we place no restrictions on the jammer's behavior. The jamming signal $j(t)$ can be either malicious AWGN or any modulated signal. The jammer can purposely start or stop jamming at any time, or adjust the transmit power to variable levels during the jamming period. Table 2.1 summarizes the notations used in this chapter.

2.1.2 Topology

In our system, nodes can operate independently without knowledge of the topology. More concretely, the sender and receiver are not necessarily aware of each other's location and even their own location. Similarly, location of the jammer is also unknown. However, all nodes (including adversary) are assumed to be static.

2.2 Approach

In a traditional system where the receiver has only one omni-directional antenna, the simultaneous transmission of both sender and jammer causes the receiver not able to recover the data signal $s(t)$ correctly, because a significantly large portion of the received signal $r(t)$ is the jamming signal $j(t)$. Our SAIM solution is a hybrid system consisted of two stages of anti-jamming: *antenna auto-configuration*, and *digital jamming cancellation*. The main components of SAIM are shown in Figure 2.1.

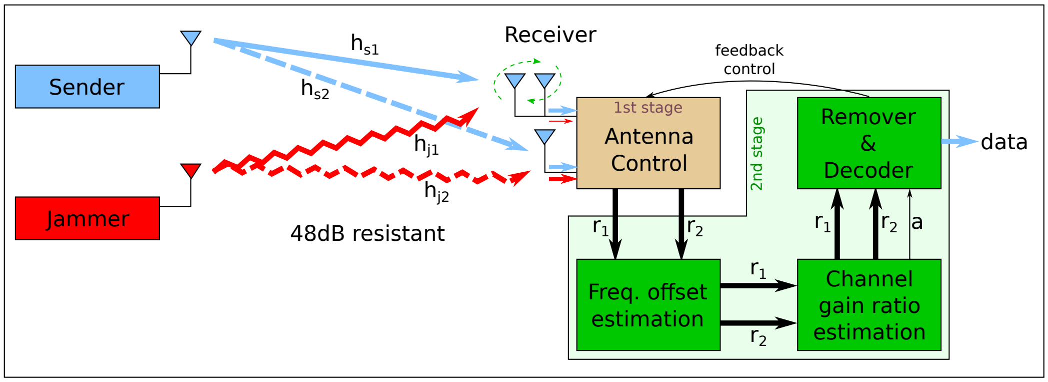


Figure 2.1: Block diagram of SAIM system.

First stage: The receiver uses for signal reception a specially designed two-element antenna capable of adjusting the *element separation* (distance between two elements) and the *angle* (rotational direction) around the central vertical axis. Our antenna prototype is shown in Figure 2.2. The characteristics of the antenna is that when its angle and element separation are adjusted, the receive pattern is changed (details in Section 2.3). In fact, we can construct a *large* number of different receive patterns, in comparison with fixed-position electronically steerable arrays.

At the heart of the first stage, we introduce an algorithm that dynamically configures the angle and the element separation of the antenna to increase legitimate sender's signal power, while, *at the same time*, reducing the jammer's interfering power. Concretely, the sender-receiver channel $h_s(t)$ and jammer-receiver channel $h_j(t)$ are changed accordingly to the antenna configuration such that $|h_j(t)/h_s(t)| \ll P_s/P_j$. As a result, $r(t) = h_s(t)s(t) + h_j(t)j(t) \approx h_s(t)s(t)$ allows the original data signal $s(t)$ to be decoded successfully. The flexibility of our custom-designed antenna allows the auto-configurability of the system to work effectively in both outdoor and indoor environments, where the latter often incurs problems to electronically steerable antenna arrays and results in poor performance. Our experiments show that our configuration algorithm can fast converge within 20 seconds, and the first stage can cope with a jammer with up to 28dB stronger power than legitimate users. However, we aim to increase the anti-jamming performance *beyond* the 28dB limit, for which we extend our model with the second stage.

Second stage: We use digital interference cancellation techniques to eliminate the jamming signal. For this goal, we attach a single-element antenna (in addition to the two-element antenna already used by the first stage) to the receiver in order to obtain an additional copy of the transmitted signal (cf., Figure 2.2). The operation of the second stage in conjunction with the first stage is illustrated in Figure 2.1. Equation (2.2)

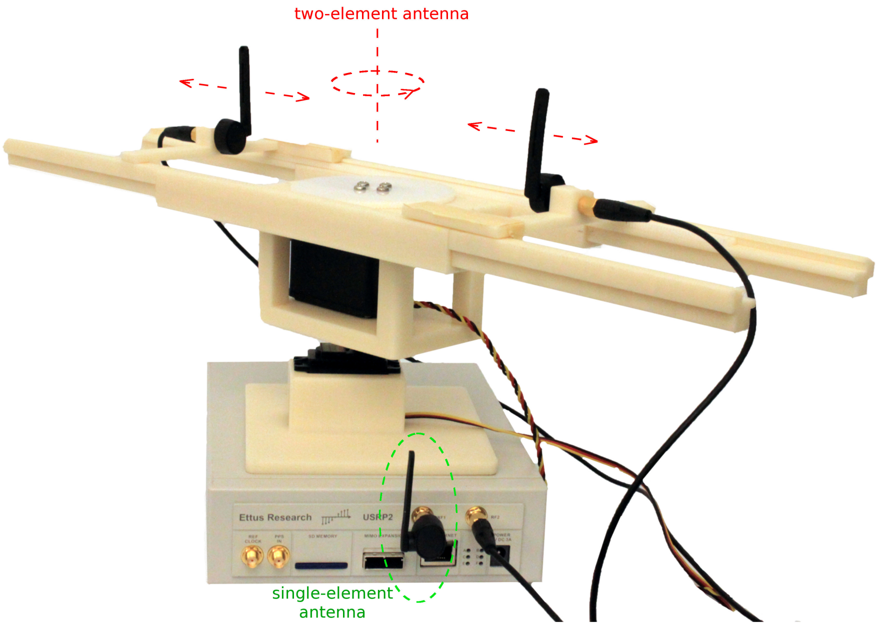


Figure 2.2: SAIM receiver with a two-element antenna (used for the 1st stage) capable of rotation and element separation, and a single-element antenna (used for the 2nd stage)

illustrates the idea of the jamming cancellation techniques applied to the received signal $r_1(t)$ at the single-element antenna and the received signal $r_2(t)$ at the two-element antenna.

$$\begin{aligned} r_1(t) &= h_{s1}(t)s(t) + h_{j1}(t)j(t) \\ r_2(t) &= h_{s2}(t)s(t) + h_{j2}(t)j(t) \end{aligned} \tag{2.2}$$

Our goal is to recover $s(t)$ from Equation (2.2) given only $r_1(t)$ and $r_2(t)$. While our setup resembles a MIMO 2x2 system, the major difference relies in the channel estimation and jamming removing techniques. Traditional MIMO systems use training sequences to estimate the channel gains. Unfortunately, this is not possible in our setup since we do not have control over the jamming signal. Instead, we propose a technique specific to this model which involves estimating the channel gain ratio $a(t) = h_{j2}(t)/h_{j1}(t)$. We observe that with the estimates of $a(t)$, we can decode $s(t)$ by

$$b(t)s(t) = a(t)r_1(t) - r_2(t), \tag{2.3}$$

where $b(t) = a(t)h_{s1}(t) - h(t)_{s2}(t)$. The ratio $a(t)$ depends on the channel characteristics such as attenuation, multipath and the power of the jamming signal. While $a(t)$ is required to be estimated, the factor $b(t)$ is considered as a new channel gain of the residual signal after eliminating the jamming signal, and does not introduce any difficulty for the decoder, thus requires no explicit estimation.

In summary, the high-level idea of our approach is to build a hybrid system consisting of two levels of anti-jamming techniques: first level of jamming cancellation by mechanical means of our custom-designed antenna and second level of jamming cancellation by software-based signal processing techniques. The robustness of our system highly depends on the performance of the configuration algorithm and digital interference cancellation algorithm. In the next sections, we will discuss the following problems:

- What is the optimal antenna configuration (separation, angle) that maximizes the SINR?
- How to estimate the channel characteristics to optimize the performance of the digital jamming cancellation technique against unknown jamming signals?

2.3 First stage – Antenna Auto-Configuration

In SAIM, the first stage employs the two-element antenna of a special design that allows the antenna not only to orientate, but also to move its elements relatively to the center, resulting in dynamic receive patterns. We will show that a *large* number of different receive patterns can be constructed, in comparison with fixed-position electronically steerable arrays. Later in this section, we present our efficient algorithm for auto-configuring the antenna adaptively to dynamic environments such that the received legitimate signal is enhanced, while – at the same time – the received jammer's signal is mitigated. In other words, we aim to maximize the received Signal-to-Jamming Ratio (SJR).

2.3.1 Pattern Analysis

We first study the basic characteristics of our two-element antenna. As the antenna structure can rotate around the vertical axis and two elements can move along the holding frame, signals received at two elements can be added constructively or destructively depending on the difference between phases arriving at two elements, which in turn depend on orientation and separation of elements. When two signals have the same phase, they add up together. When phases are opposite, signals eliminate each other. In the following, we characterize the receive pattern[1] of the antenna. Specifically, we are interested in the locations of lobes (where signals add up) and nulls (where signals eliminate). For ease of presentation, we introduce our notations:

[1] Receive pattern of an antenna shows the power of signal received at the antenna from a specific direction. We use polar coordinate system to depict the pattern, where the angle represents the incoming signal's direction, while the distance to origin indicates the level of signal power (dB scale).

Definition 2.3.1. *The antenna is said to be in a configuration (L,ϕ), if the element separation is equal to L and the frame position creates an angle ϕ with a referenced frame position. We assume that $L_{\min} \leq L \leq L_{\max}$ and $\phi_{\min} \leq \phi \leq \phi_{\max}$ for some system parameters $L_{\min}, L_{\max}, \phi_{\min}, \phi_{\max}$ depending on design and implementation constraints. We denote $P(L,\phi)$ as the signal power received with the corresponding configuration.*

Aiming to reducing interference, we are interested in such antenna configurations that result in minimum and maximum received power. Those special configurations are defined accordingly as follows.

Definition 2.3.2. *(L,ϕ) is a minimizing (resp. maximizing) configuration, if $P(L,\phi) \leq P(L',\phi')$ (resp. $P(L,\phi) \geq P(L',\phi')$) for all (L',ϕ'), $L' \in [L_{\min}, L_{\max}]$, $\phi' \in [\phi_{\min}, \phi_{\max}]$.*

Definition 2.3.3. *L_ϕ is called ϕ-minimizing separation, if $P(L_\phi,\phi) \leq P(L,\phi)$ for all $L \in [L_{\min}, L_{\max}]$. Similarly, L_ϕ is called ϕ-maximizing separation, if $P(L_\phi,\phi) \geq P(L,\phi)$ for all $L \in [L_{\min}, L_{\max}]$.*

Definition 2.3.4. *ϕ_L is called L-minimizing angle, if $P(L,\phi_L) \leq P(L,\phi)$ for all $\phi \in [\phi_L - \theta, \phi_L + \theta]$. The parameter θ denotes the local search range, which will be discussed later. Similarly, ϕ_L is called L-maximizing angle, if $P(L,\phi_L) \geq P(L,\phi)$ for all $\phi \in [\phi_L - \theta, \phi_L + \theta]$.*

Intuitively, minimizing angles are directions inside the nulls where the received power is minimized, and maximizing angles are directions inside the lobes where the received power is maximized. It is noted that there might exist more than one minimizing (or maximizing) angle for a specific separation L. Moreover, a minimizing (or maximizing) angle for a separation L is not necessarily a minimizing (or maximizing) angle for another separation L'. In contrast, there is only one minimizing (or maximizing) separation (global optimum) for an angle ϕ.

2.3.1.1 Analytical Pattern – Light-of-Sight Model

In light-of-sight propagation model, phases of signals received at two antenna elements only depend on the antenna configuration and the distance to transmitter. When distance to transmitter is significantly far in comparison to element separation, the following Theorems 2.3.1 and 2.3.2 give the locations and number of lobes and nulls as functions of the relative ratio $K = L/\lambda$ between element separation L and signal wavelength λ. Theorem 2.3.1 precisely determines the locations of lobes and nulls. It is interestingly noted that the angles 0 and π become nulls when the ratio's fractional part $\{K\} = K - \lfloor K \rfloor$ is less than half; otherwise, they become lobes. Theorem 2.3.2 determines the number of lobes and nulls expected for a configuration, which is useful for optimizing the antenna control algorithm shown later.

Theorem 2.3.1 (Locations of lobes and nulls)**.** *For a free-space communication at carrier wavelength λ with a receiver's two-element antenna configured at separation L, letting $\mathscr{M} = \{\phi, \cos\phi = k/K, |k| < K, k \in \mathbb{Z}\}$, where $K = L/\lambda$, the maximizing angles are located at*

$$\phi_L \in \mathscr{M}, \text{ if } \{K\} < \tfrac{1}{2} \qquad\qquad \phi_L \in \{0, \pi\} \cup \mathscr{M}, \text{ if } \{K\} \geq \tfrac{1}{2}.$$

Similarly, letting $\mathscr{N} = \{\phi, \cos\phi = \frac{k+1/2}{K}, |k+1/2| < K, k \in \mathbb{Z}\}$, the minimizing angles are located at

$$\phi_L \in \{0, \pi\} \cup \mathscr{N}, \text{ if } \{K\} < \tfrac{1}{2} \qquad\qquad \phi_L \in \mathscr{N}, \text{ if } \{K\} \geq \tfrac{1}{2}.$$

Proof. In Figure 2.3, we consider the antenna configuration (L, ϕ) and the signals transmitted from T and received at antenna elements A and B.

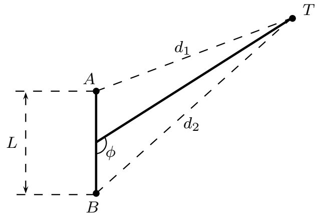


Figure 2.3: The two-element antenna in a free-space communication with one transmitter.

We assume a narrowband slow fading communication channel, therefore the signal received at A and B does not significantly differ in frequency offset, channel attenuation, fading, etc. The received signals at the two elements are represented by

$$\begin{aligned} r_A(t) &= g(x)\cos(2\pi f t + 2\pi\frac{d_1}{\lambda}) \\ r_B(t) &= g(x)\cos(2\pi f t + 2\pi\frac{d_2}{\lambda}) \end{aligned}$$

where $g(x)$ contains the transmitted data, f is the carrier frequency, λ is the carrier wavelength, and t denotes the receiving time. The sum of two signals at the output of the combiner, $r(t) = r_A(t) + r_B(t) = 2g(x)\cos(\pi\frac{d_1-d_2}{\lambda})\cos(2\pi f t + \pi\frac{d_1+d_2}{\lambda})$, is a signal of amplitude $|2g(x)\cos(\pi\frac{d_1-d_2}{\lambda})|$. Regardless of the transmitted data, the amplitude of $r(t)$ depends on the value of $|\cos(\pi\frac{d_1-d_2}{\lambda})|$. Since the distances between the transmitter and the receiver elements are much larger than the element separation, i.e., $d_1 \gg L$, $d_2 \gg L$, we have $d_1 - d_2 \approx L\cos\phi$. Let $h(\phi) = \cos^2(\pi K\cos\phi)$, $K = L/\lambda$. We investigate the amplitude of $r(t)$ indirectly by investigating $h(\phi)$. Note that by definition of maximizing angles and minimizing angles, the maximum and minimum values of $|r(t)|$ are not necessarily equal to 0 or 1. In fact, they are the roots of $h'(\phi) = 0$, where $h'(\phi) = 2\pi K \sin\phi \sin(2\pi K \cos\phi)$ is the derivative of $h(\phi)$. Roots of $h'(\phi) = 0$ satisfy the following conditions:

$$\sin\phi = 0 \tag{2.4}$$

$$\text{or} \quad \sin(2\pi K \cos\phi) = 0 \tag{2.5}$$

Letting $h_1(\phi) = 4\pi^2K^2 \sin^2\phi \cos(\pi K \cos\phi)$ and $h_2(\phi) = 2\pi K \cos\phi \sin(\pi K \cos\phi)$, we have $h''(\phi) = h_2(\phi) - h_1(\phi)$.

First, we consider Section 2.3.1.1. Let ϕ_1 be a root of Section 2.3.1.1, i.e., $\sin(\phi_1) = 0$, then $\cos(\phi_1) = \pm 1$, and $\phi_1 = 0$ or $\phi_1 = \pi$. As a result, $h_1(\phi_1) = 0$, and $h_2(\phi_1) = \pm 2\pi K \sin(\pm 2\pi K) = 2\pi K \sin(2\pi K)$ (the last equality is due to x having same sign as $\sin x$). Now that $h''(\phi_1) = h_2(\phi_1) = 2\pi K \sin(2\pi K)$.

- If $\{K\} \leq \frac{1}{2}$, $h''(\phi_1) \geq 0$, then ϕ_1 is a minimizing angle.
- If $\{K\} \geq \frac{1}{2}$, $h''(\phi_1) \leq 0$, ϕ_1 is a maximizing angle.

Now consider Section 2.3.1.1. Let ϕ_2 be a root of Section 2.3.1.1, i.e., $\sin(2\pi K \cos\phi_2) = 0$, then we have $h_2(\phi_2) = 0$, and $h''(\phi_2) = -h_1(\phi_2) = -4\pi^2K^2 \sin^2(\phi) \cos(\pi K \cos\phi)$. Note that $\cos(2\pi K \cos\phi_2) = \pm 1$.

- If $\cos(2\pi K \cos\phi_2) = 1$, or $\cos\phi_2 = k/K$, then $h''(\phi_2) < 0$, and ϕ_2 is a maximizing angle.
- If $\cos(2\pi K \cos\phi_2) = -1$, or $\cos\phi_2 = (k + \frac{1}{2})/K$, then $h''(\phi_2) > 0$, and ϕ_2 is a minimizing angle.

In conclusion, ϕ is a maximizing angle, if $\cos\phi = k/K$, or a minimizing angle, if $\cos\phi = (k + \frac{1}{2})/K$, $k \in \mathbb{Z}$. In addition, if $\{K\} \geq \frac{1}{2}$, we have two more maximizing angles at 0 and π; otherwise, they are two additional minimizing angles. □

Theorem 2.3.2 (Number of lobes and nulls)**.** *The number of maximizing angles of the two-element antenna in a free-space communication is equal to the number of minimizing angles, which is*

$$\begin{array}{ll} 4K, & \textit{if } K \in \mathbb{Z} \\ 2\lfloor 2K \rfloor + 2, & \textit{if } K \notin \mathbb{Z} \end{array}$$

Proof. First, we observe that there is always one null between two lobes, and one lobe between two nulls, that is the number of minimizing angles equals the number maximizing angles. Therefore, it is enough to only determine the number of maximizing angles of the receive pattern given ratio K between the separation and the carrier wavelength. We prove the theorem by counting the number of maximizing angles.

If K is integer, according to Theorem 2.3.1, we have maximizing angles at ϕ, for which $\cos\phi = \frac{k}{K}$, $k = -K, \ldots, 0, \ldots, K$, $k \in \mathbb{Z}$.

- For $k = \pm K$, we have maximizing angles at 0 and π.

- For each $k \in S_1 = \{-K+1, \ldots, 0, \ldots, K-1\}$, $|S_1| = 2K-1$, there are two maximizing angles at $\phi = \arccos\frac{k}{K}$ and $\phi = \pi - \arccos\frac{k}{K}$.

In total, we have $2 + 2|S_1| = 4K$ maximizing angles.

If K is a non-integer, for each $k \in S_2 = \{-\lfloor K \rfloor, \ldots, 0, \ldots, \lfloor K \rfloor\}$, $|S_2| = 2\lfloor K \rfloor + 1$, we have 2 maximizing angles at $\phi = \arccos\frac{k}{K}$ and $\phi = \pi - \arccos\frac{k}{K}$. The number of those maximizing angles is $2|S_2|$.

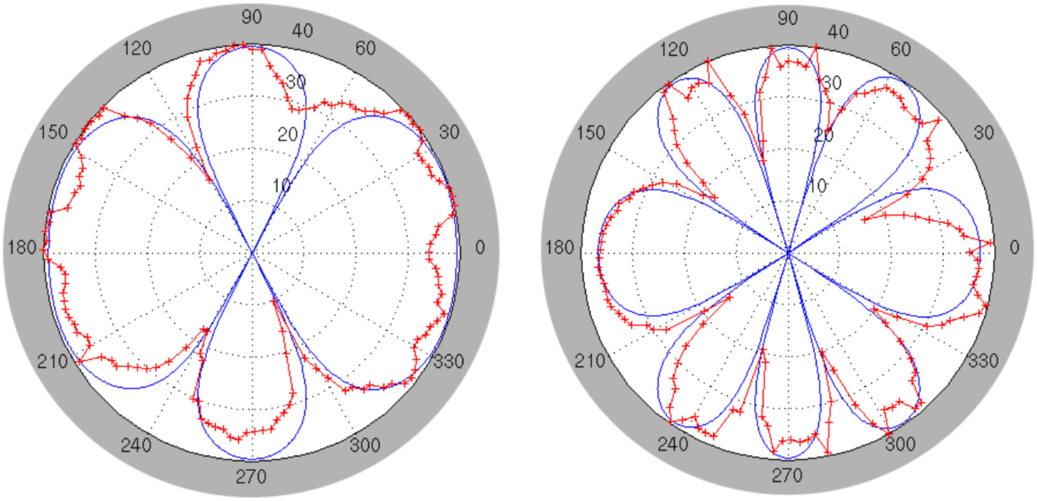
- If $\{K\} \leq \frac{1}{2}$, we have no more maximizing angles (Theorem 2.3.1), so the total number of maximizing angles is $2|S_2| = 2 \cdot (2\lfloor K \rfloor + 1) = 2\lfloor 2K \rfloor + 2$.
- If $\{K\} \geq \frac{1}{2}$, we have two additional maximizing angles at 0 and π (Theorem 2.3.1), which increase the total number of maximizing angles to $2|S_2| + 2 = 2 \cdot (2\lfloor K \rfloor + 1) + 2 = 4\lfloor K \rfloor + 4 = 2\lfloor 2K \rfloor + 2$.

Therefore, the total of maximizing angles for the case of non-integer K is $2\lfloor 2K \rfloor + 2$. Note that the above formulas are established based on the following claim: *"for any number x, if $\{x\} < \frac{1}{2}$, then $\lfloor 2x \rfloor = 2\lfloor x \rfloor$; otherwise $\lfloor 2x \rfloor = 2\lfloor x \rfloor + 1$".* □

2.3.1.2 Outdoor Experimental Pattern

To compare the two-element antenna's pattern in practice, we conducted experiments to measure the received power at the antenna in outdoor environments. The transmitter is placed at distance 10m to the receiver. Figure 2.4 shows the measured receive patterns for separation $L = \lambda = 12.5$cm and $L = 2\lambda = 25$cm ($f = 2.4$GHz). The results show that the outdoor environments have very similar characteristics to the theoretical patterns in free-space communications. Our antenna design is featured with the capability of adjusting the element separation, by which the two-element antenna can change the *locations* and the *number* of lobes and nulls in the receive pattern (according to Theorems 2.3.1 and 2.3.2).

To study the *change* of locations of lobes and nulls when adjusting the separation, we conducted another experiment, in which the separation is adjusted from minimum value $L_{\text{min}} = \lambda/4 = 3.1$cm to maximum value $L_{\text{max}} = 3\lambda = 37.5$cm. Figure 2.5 shows the locations of maximizing and minimizing angles for each separation value found in both experimental and theoretical cases. Note that, since the pattern is almost symmetric, only the maximizing and minimizing angles found in one half $[0, \pi]$ of the pattern are shown. As an example, the receive pattern for separation $L = \lambda$ has 4 minimizing angles at $\pm\pi/3$, $\pm 2\pi/3$ and 4 maximizing angles at 0, $\pi/2$, π, and $3\pi/2$, which imply 4 nulls and 4 lobes in Figure 2.4a. When the separation is increased by a small value to $L' = L + \Delta L$ with $\Delta L \approx 0.6$cm, 2 more minimizing angles and 2 more maximizing angles appear in the pattern (in Figure 2.5 we see 1 more minimizing angle and 1 more maximizing angle in $[0, \pi]$), which comply with the results of Theorem 2.3.2. In addition, Theorem 2.3.1



(a) $L = \lambda$

(b) $L = 2\lambda$

Figure 2.4: Outdoor receive patterns of the two-element antenna. Experimental gains (lines with plus signs) are compared to theoretical values (lines without markers).

implies that if $K' \approx K$, $\cos\phi' \approx \cos\phi$, then $\phi' \approx \phi$, i.e., the locations of the maximizing and minimizing angles deviate *slightly* from the previous locations.

2.3.1.3 Indoor Experimental Pattern

In an indoor environment, the receive patterns become more unpredictable due to reflecting and blocking objects. We carried out indoor experiments with our two-element antenna to characterize the receive pattern. Figure 2.6 shows that the indoor receive patterns (at different separation values) highly depend on the indoor environment (note that in contrast to outdoor scenarios, no theoretical patterns are shown). The locations and the number of lobes and nulls do not always comply with the results of Theorem 2.3.1 and Theorem 2.3.2. However, we can observe that more lobes and nulls are created when the separation between two elements is larger. For example, we obtained three lobes in one half of the receive pattern with separation value equal to twice the wavelength ($L = 2\lambda$) in comparison with five lobes with separation value equal to three times the wavelength ($L = 3\lambda$).

Similarly to the outdoor scenario, we also verified how pattern changes when the antenna separation is varied. Figure 2.7 shows that a small adjustment of separation results in a small change of maximizing angles and minimizing angles; in other words, the maximizing angles (or minimizing angles) of two close values of separation are likely not to deviate much from each other.

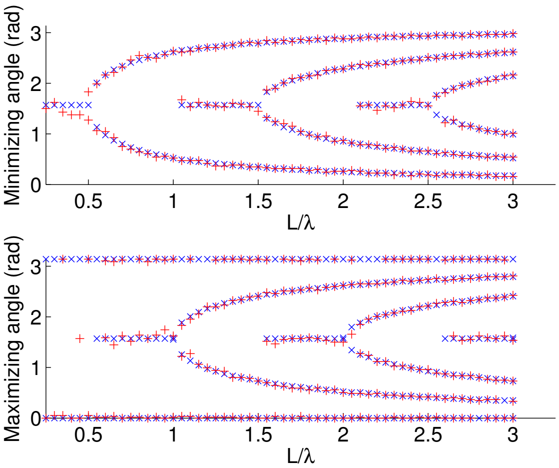


Figure 2.5: Locations of minimizing and maximizing angles for separations between $\lambda/4$ and 3λ in an outdoor environment. Plus (+) indicates experimental results, and cross (X) indicates theoretical predictions.

2.3.1.4 Pattern Continuity

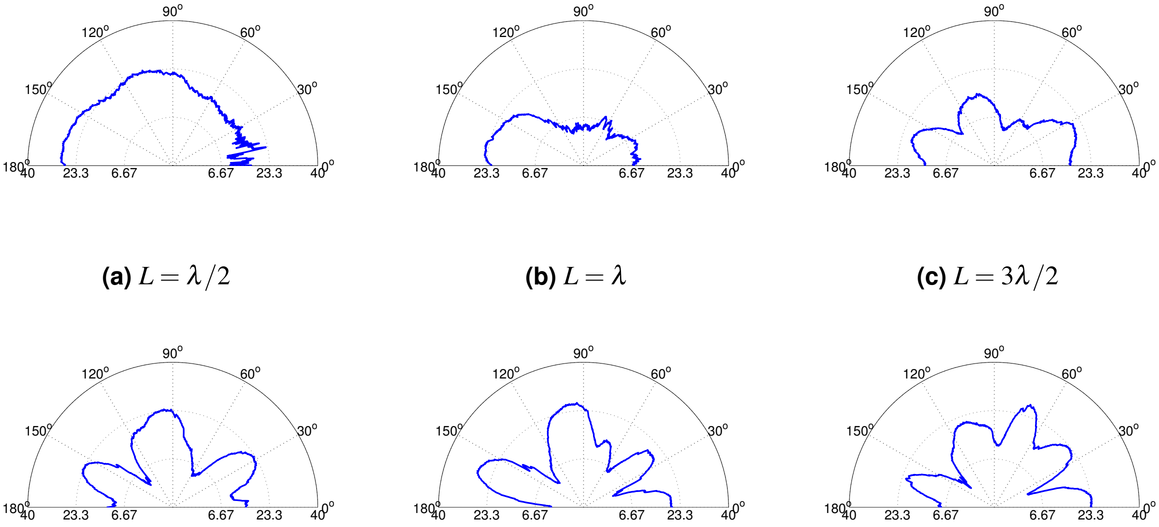
Based on the receive pattern of our two-element antenna obtained for both outdoor and indoor environments, we discover an important property of the antenna: a small change in antenna configuration results in only a slight change in receive pattern. We call this the *continuity* property of the two-element antenna's receive pattern. This property is utilized for the antenna configuration algorithm described next.

2.3.2 Antenna Configuration Algorithm

In this section, we derive the algorithm for controlling the two-element antenna to maximize the SJR at the receiver. We note that if both jammer and sender are in the same (or tiny range of) angles relatively to the receiver, Theorem 2.3.1 implies that there is no configuration resulting in significantly changing the portion of jamming power in the received signal, as the gains to the transmitters are always (almost) the same. We consider a situation in which the jammer is located in a different direction with respect to the sender.

2.3.2.1 Outdoor and Known Locations

In an outdoor environment, if the locations of the communicating and jamming nodes are known, we can maximize the SJR by determining the maximizing angles according to the relative locations between sender



(a) $L = \lambda/2$

(b) $L = \lambda$

(c) $L = 3\lambda/2$

(d) $L = 2\lambda$

(e) $L = 5\lambda/2$

(f) $L = 3\lambda$

Figure 2.6: Experimental indoor receive patterns.

and receiver in order to maximize the received power from the sender, and at the same time, determining the minimizing angles according to the relative locations between jammer and receiver in order to minimize the received power from the jammer. Based on the results of Theorem 2.3.1, the maximizing and minimizing angles can be precomputed, cf. Algorithm 1.

In Algorithm 1, minimizing angles and maximizing angles are computed based on the element separation L and relative locations of the nodes and returned as two sets: $A_J = [\phi_{m_1} - \theta, \phi_{m_1} + \theta] \cup \ldots \cup [\phi_{m_k} - \theta, \phi_{m_k} + \theta]$ for minimizing jammer's power and $A_S = [\phi_{k_1} - \theta, \phi_{k_1} + \theta] \cup \ldots \cup [\phi_{k_n} - \theta, \phi_{k_n} + \theta]$ for maximizing sender's power, where k and n are the number of minimizing and maximizing angles found by above theorems, respectively. As for each separation L, there are multiple positions that maximize the SJR, the SJR corresponding to each angle in the intersection of A_J and A_S are compared to find the best configuration. The advantage of Algorithm 1 is that the computations can be done offline, therefore requiring minimal setup time in a real-world deployment.

2.3.2.2 Unknown Locations

For outdoor environments and unknown locations of nodes, Algorithm 1 is not applicable. For indoor environments, even if the locations of nodes are known, the channel highly depends on the specific environment

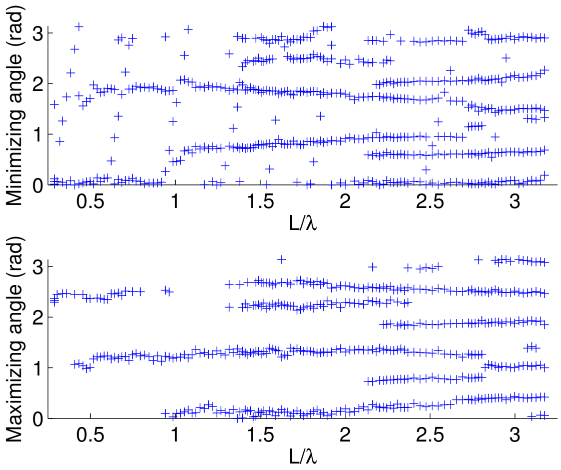


Figure 2.7: Locations of minimizing and maximizing angles for separation values between 0.25λ and 3.25λ measured in an indoor experiment.

Algorithm 1 Precomputable configuration for outdoor and known locations

```
PRECOMPUTE($(L_{\text{min}}, L_{\text{max}})$)
for $L \in [L_{\text{min}}, L_{\text{max}}]$        // rotating search while fixing separation
    $A_J$ = minimizing_angles_to_jammer($(L)$)
    $A_S$ = maximizing_angles_to_sender($(L)$)
    for $\phi \in A_J \cap A_S$
        if SJR$(L, \phi) >$ SJR$(L_{\text{opt}}, \phi_{\text{opt}})$
            $L_{\text{opt}} = L$
            $\phi_{\text{opt}} = \phi$
return $(L_{\text{opt}}, \phi_{\text{opt}})$
```

and results in unpredictable patterns. In this section, we present the antenna configuration algorithms that work for both outdoor and indoor environments.

Our goal is to maximize the SJR at the receiver. According to Theorem 2.3.1, changing separation results in new locations of maximizing and minimizing angles, therefore yielding different gains for the jammer and the sender (as they are not in the same direction). Consider a powerful jammer whose power dominates the received signal. Changing the antenna configuration to null the jammer, we would reduce the received signal's power. Thus, *maximizing* the SJR implies *minimizing* the total received power at the receiver. For low-power jammer, this implication is not applied, however the algorithms described below are still useful when combining with the digital cancellation technique to recover the user data.

Brute-force algorithm To minimize the total received power, a "brute-force" search would yield the best configuration: this search would measure the received power at the two-element antenna for all possible configurations and select the one corresponding to the minimum power.

Algorithm 2 Brute-force for unknown node locations

```
BRUTEFORCE($L_{\min}, L_{\max}, \phi_{\min}, \phi_{\max}$)
for $\phi \in [\phi_{\min}, \phi_{\max}]$
    for $L \in [L_{\min}, L_{\max}]$
        if $P(L, \phi) < P(L_{\text{opt}}, \phi_{\text{opt}})$
            $L_{\text{opt}} = L$
            $\phi_{\text{opt}} = \phi$
return $(L_{\text{opt}}, \phi_{\text{opt}})$
```

Without knowledge of node locations, we cannot rely on Theorem 2.3.1 to compute the optimal configuration. Instead, the brute-force approach tries each configuration by varying the rotational angle and the element separation within the physical limits and measuring the received power. Given the large number of separation values and angle values, such approach would take a significant amount of time to find the best configuration.

Fast algorithm Recall the continuity property of the receive pattern: continuously changing the separation results in new locations of maximizing angles and minimizing angles in the small vicinity of the previous ones. Based on this property, we propose the *fast algorithm*, cf. Algorithm 3.

Algorithm 3 Fast algorithm for unknown node locations

```
FastAntennaControl($(L_0, L_1, L_2, \phi_0, \phi_1, \phi_2)$)
$L^* = L_0, \quad \phi^* = \phi_0$                // initial configuration
repeat
    for $\phi = \phi_1$ to $\phi_2$              // rotating search while fixing separation
        if $P(L^*, \phi) < P(L^*, \phi^*)$
            $\phi^* = \phi$
    for $L = L_1$ to $L_2$                    // separating search while fixing orientation
        if $P(L, \phi^*) < P(L^*, \phi^*)$
            $L^* = L$
    // update search range
    $L_1 = L^* - \Delta L, \quad L_2 = L^* + \Delta L$
    $\phi_1 = \phi^* - \theta, \quad \phi_2 = \phi^* + \theta$
until $(L^*, \phi^*)$ unchanged
return $(L^*, \phi^*)$
```

Main idea Our algorithm is a local optimum search based on online measurement of received power at the receiver. The algorithm is initialized with a full search range $L_1 = L_{\min}$, $L_2 = L_{\max}$, $\phi_1 = \phi_{\min}$, $\phi_2 = \phi_{\max}$, which are determined by the antenna implementation constraints. The initial configuration (L_0, ϕ_0) are given as parameters. The configuration is iteratively improve in a series of steps, each comprises searching in only one dimension, either rotation or separation change. More precisely, the configuration search is, first, started by rotating the antenna between the given range while fixing the separation at the given separation value L_0. By measuring the received power at each angular position, we locate the angle ϕ^* that gives the minimum received power for the current separation value L_0. We know that ϕ^* found in this step is not necessarily the best one for other separation values. Therefore, in the next step, different separations within the given range $[L_1, L_2]$ are tried to improve the configuration. The configuration search in these two steps relies on the continuity property: if there is a better configuration, it is likely to be found in small vicinity of the most recently optimal configuration. We repeat these steps until no better configuration is found.

Due to the continuity property of the receive pattern, the algorithm converges in a local optimum point which corresponds to a good configuration, in which the SJR is maximized. The search range $(L_1, L_2, \phi_1, \phi_2)$ is updated after each iteration with system parameters ΔL and θ.

Algorithm 3 is much faster than brute-force, as it probes the optimal angle and separation values separately. We emphasize that the configuration returned by the fast algorithm is not essentially the best configuration, however as shown in Section 2.6, is comparable to brute-force.

Search range Recall the definition of minimizing angles in Definition 2.3.4, where θ depends on L. To determine θ, the search range's updating parameter, we observe that, according to Theorem 2.3.2, there are approximately $\frac{4L}{\lambda}$ lobes and $\frac{4L}{\lambda}$ nulls in the receive pattern depending on the separation value L and wavelength λ. Thus, the width of each lobe (or null) can be (approximately) averaged to $2\theta \approx \frac{2\pi}{2\times\frac{4L}{\lambda}} = \frac{\pi\lambda}{4L}$, or $\theta \approx \frac{\pi\lambda}{8L}$. Increasing the separation will reduce θ and consequently reduce the searching time.

Impact factors There are several factors that can impact the performance of this algorithm:

- *Local optimum:* Based on the continuity property, we can find a better configuration in the surrounding range of the current configuration, which yields the local minimum received power. However, since a receive pattern has multiple nulls and lobes, the algorithm only gives a local optimum solution which might not be the best configuration.
- *Mobility:* During the configuration search, if the nodes are moving, the channels can vary and result in an unoptimal configuration.
- *Low-power jamming:* Recall that we made an assumption of high-power jamming for this algorithm to work properly. If the jamming power is in the same order of the user signal's power, minimizing received power would not necessarily increase the SJR, because both user signal and jammer signal can be reduced, if by the best (minimum-power) configuration they are placed into the nulls of the receive pattern.

To solve the above issues, we apply the fast algorithm in parallel with the digital jamming cancellation process in the second stage. The fast algorithm is performed to reduce the received signal to such power levels that the jamming signal can be removed and original data can be decoded successfully.

A note on indoor environments The indoor channels highly depend on the specific environment and result in unpredictable patterns, even if the locations of nodes are known and fixed. However, Section 2.3.1.3 shows that the continuity property still holds. The difference of indoor environments from outdoor environments is the existence of "multipaths" of the signals. As we describe in more details in Section 2.4, in our system the multipath effect can be considered as a single path which acts as moving the jammer to a different location with different gain to the receiver. Since Algorithm 3 is designed, based on the continuity property, to identify and mitigate the jamming effect without knowing the jammer's location and power, it is also applicable for indoor environments. Our indoor experimental results (Section 2.6) demonstrate that the algorithm works efficiently for indoor environments. It is an adaptive algorithm relying on the real-time measurements during the operation of the system, without awareness of locations of the communicating and jamming nodes.

2.4 Second stage – Digital Jamming Cancellation

In the second stage (Figure 1.3), we extend our model by using digital interference cancellation to eliminate the jamming signal. Our cancellation technique requires an additional signal provided to the receiver. We use a single-element antenna for the additional signal reception. As a result, SAIM's receiver uses totally three single antennas, two of which are joined to construct the composite two-element antenna. In our simplified model, Equation (2.6) illustrates the idea of the jamming cancellation technique applied to the received signals at the single-element antenna (ANT-1) and the two-element antenna (ANT-2). We obtain two different copies R_1, R_2 of the transmitted signal at the receiver:

$$\begin{aligned} R_1 &= h_{S1}S + h_{J1}J, \qquad h_{S1}, h_{J1} : \text{channel gain of sender, jammer to ANT-1} \\ R_2 &= h_{S2}S + h_{J2}J, \qquad h_{S2}, h_{J2} : \text{channel gain of sender, jammer to ANT-2} \end{aligned} \tag{2.6}$$

We propose a technique specific to this model to estimate the channel gain ratio $a = h_{J2}/h_{J1}$ in order to recover the legitimate signal: $bS = aR_1 - R_2$, where $b = ah_{S1} - h_{S2}$. Knowing a, we can decode S. The channel gain ratio a depends on the channel characteristics such as attenuation, multipath and the power of the jamming signal. The factor b is considered as a new channel gain of the residual signal after eliminating the jamming signal, and does not introduce any difficulty for the decoder, thus requires *no* estimation. While the techniques used in this system are rooted in techniques developed for MIMO communication [112] and phased array antenna [69, 111], fields that have been extensively studied over several decades, the characteristics of our setup and design require new algorithms and techniques. Our digital jamming cancellation algorithms target *powerful and unknown jammers*, unlike traditional MIMO techniques that operate over user-designed transmission signals of similar powers, allowing adequate channel estimation through training sequences. In the following, we discuss our estimation of channel gain ratio a, the key challenge in our cancellation technique.

2.4.1 Channel Gain Ratio Estimation

In our model, the channel gains affected by the communication medium are represented as complex numbers which introduce magnitude and phase change in the received signals. Our digital processing techniques are applied to sequences of samples taken from the analog input at discrete time $t = t_0, t_0 + \tau, t_0 + 2\tau, \ldots$ where τ is the sampling period and t_0 is the time when the signals first arrive at the receiver input. Equation (2.2) can be explicitly rewritten in the time domain as follows:

$$\begin{aligned} R_1(t) &= h_{S1}(t)S(t) + h_{J1}(t)J(t) \\ R_2(t) &= h_{S2}(t)S(t) + h_{J2}(t)J(t) \end{aligned} \tag{2.7}$$

where the channel gains are complex functions of time t. Removing jamming signal involves the estimation of $a(t) = \frac{h_{J2}(t)}{h_{J1}(t)}$. We estimate $a(t)$ by separately computing its magnitude and phase over a small number of n samples. Since the jamming signal is unknown (i.e., $h_{J1}(t), h_{J2}(t)$ are unknown), our approach exploits the independence of stochastic processes.

Magnitude estimation The received power at ANT-1 in the past n samples before time t_0 is:

$$\begin{aligned} P_1(t_0) &\triangleq \frac{1}{n}\sum^{t_0} |h_{S1}(t)S(t) + h_{J1}(t)J(t)|^2 \\ &= \frac{1}{n}\left(\sum^{t_0} |h_{S1}(t)S(t)|^2 + \sum^{t_0} |h_{J1}(t)J(t)|^2\right) \\ &= \frac{1}{n}\left(|h_{S1}|^2 \sum^{t_0} |S(t)|^2 + |h_{J1}|^2 \sum^{t_0} |J(t)|^2\right), \end{aligned}$$

where the second equality is due to the independence between jamming signal and sender's signal, i.e., $\sum h_{S1}(t)h_{J1}(t)S(t)J(t) = 0$, while the third equality comes from the slow-fading characteristics in a narrow-band communication [112], i.e., $h_{S1}(t) = h_{S1}$, $h_{S2}(t) = h_{S2}$, $h_{J1}(t) = h_{J1}$, $h_{J2}(t) = h_{J2}$. Similarly, the power received at ANT-2 can be represented as

$$P_2(t_0) = \frac{1}{n}\left(|h_{S2}|^2 \sum^{t_0} |S(t)|^2 + |h_{J2}|^2 \sum^{t_0} |J(t)|^2\right).$$

If the portion of jamming power in $P_1(t_0)$ and $P_2(t_0)$ were significantly greater than that of the sender, one could estimate $|a| = \left|\frac{h_{J2}}{h_{J1}}\right| = \frac{P_2(t_0)}{P_1(t_0)}$. In order to estimate $|a|$ in more general cases, we apply another approach, in which the receiver is required to detect the signal from both sender and jammer.

Signal detection In this work, we consider such scenarios where the jamming signal is uncorrelated to the sender's signal (e.g., the jammer does not replay the sender's data but independently generates its own interference), and the communication between the sender and receiver is carried out at constant transmit power. To detect the signal, we monitor the average received power in a short period (several symbols) and seek for the abrupt change in magnitude. The detection is reported when the average received power suddenly increases by at least twice in a period of 1 or 2 symbols. For complex scenarios, if the jammer is capable of transmitting “user-like” data (e.g., the jammer is a compromised user), the system needs more sophisticated methods to identify whether the received signal is the jammer’s signal. We leave those complex scenarios for future work.

In order to estimate $|a|$, our approach is to measure the received power before and after the collision, and compare the recorded power in these two short periods to determine the ratio’s magnitude. The correctness of our method relies on the slow-fading assumption, by which the user/jammer’s signal is considered to have constant average power across the collision time.

Sender transmitted before collision If the sender transmitted before the jammer causes interference, the receiver estimates $|a|$ by the following steps:

- Measures $P_i(t_0)$ in the period t_0, which contains only the power of the sender’s signal received at both (two-element and single-element) antennas, $P_{Si}(t_0) = P_i(t_0) = \frac{1}{n}|h_{Si}|^2 \sum_{t_0} |S(t)|^2$ ($i = 1$ denotes the two-element antenna, and $i = 2$ denotes the single-element antenna). As the sender’s power is constant, we obtain $P_{Si}(t) = P_{Si}(t_0) = P_{Si}$ for any other period $t > t_0$.
- Measures $P_i(t_1)$ in the interference period t_1, $|a|$ can be computed by
$$|a| = \left|\frac{h_{J2}}{h_{J1}}\right| = \sqrt{\frac{P_2(t_1) - P_{S2}}{P_1(t_1) - P_{S1}}}.$$

Jammer transmitted before collision If the jammer is known to jam before the collision time t_0, the receiver measures the power at both (two-element and single-element) antennas before the collision, $P_i(t_0) = \frac{1}{n}|h_{Ji}|^2 \sum_{t_0} |J(t)|^2$. In the collision period t_1, the receiver measures $P_i(t_1)$. Since the time period immediately before and after the collision is short, the jammer’s power remains almost constant, i.e., $P_{Ji}(t_1) \approx P_{Ji}(t_0)$. This allows the sender’s power at each antenna to be found by $P_{Si} = P_i(t_1) - P_i(t_0)$. Knowing P_{Si}, $|a|$ can be estimated by the last step described in the first case.

Phase estimation The phase difference ϕ between $R_1(t)$ and $R_2(t)$ is determined by
$$\phi = \tan^{-1}\left(-\frac{\sum_t [I_1(t)Q_2(t) - I_2(t)Q_1(t)]}{\sum_t [I_1(t)I_2(t) + Q_1(t)Q_2(t)]}\right),$$
where $I_1(t) = \mathrm{Re}[R_1(t)]$, $Q_1(t) = \mathrm{Im}[R_1(t)]$, $I_2(t) = \mathrm{Re}[R_2(t)]$, $Q_2(t) = \mathrm{Im}[R_2(t)]$ represent the real and imaginary parts of the received signals. Similarly to the approach used in estimating the magnitude, we derive ϕ based on the phase difference ϕ in the periods before and after the collision. In software-defined radio, for both magnitude and phase estimation, the signal processing operations are done for chunks of n samples taken from the analog input.

2.4.2 Removing and Decoding

When the gain ratio a is estimated correctly, the jamming signal can be removed completely from the received signals by solving equation (2.3). The residual signal $b \cdot S$ is sent to the decoder to decode the data. The gain b of the residual signal is considered as a new channel gain of the signal after removing the jamming signal. Therefore, the data can be decoded by the decoder with well-known decoding techniques [16, 88] in software-defined radio. Consequently, estimation of b is not required.

2.4.3 Practical Issues

In practice, we need to address the issue of frequency offset between the received signals which are unavoidable in real devices. Moreover, the multipath problem is always an interesting part of systems working indoor.

Frequency offset estimation With the goal of providing a zero-knowledge anti-jamming system, manual calibration for compensating the frequency offset is not desired in our system. The frequency offset between the received signals is estimated in real-time by $\Delta f^* = \text{argmax}_{\Delta f} |\mathscr{F}\{R_1(t)R_2^*(t)\}|$, where $\mathscr{F}$ denotes the Fourier transform.

Dealing with multipath In this paragraph, we demonstrate that our estimation approach also works efficiently in indoor environments, where multipaths can occur. Intuitively, due to reflection, multiple copies of the transmitted signals arrive at the receive antennas:

$$\begin{aligned} R_1 &= \left(\sum_k h_{S1}^{(k)}\right) S + \left(\sum_k h_{J1}^{(k)}\right) J \\ R_2 &= \left(\sum_k h_{S2}^{(k)}\right) S + \left(\sum_k h_{J2}^{(k)}\right) J \end{aligned} \tag{2.8}$$

where $h_{Si}^{(k)}$, $h_{Ji}^{(k)}$ denote the channel gain of the k-th path from the sender and the jammer to the receiver, respectively. By letting $h_{Si} = \sum_k h_{Si}^{(k)}$ and $h_{Ji} = \sum_k h_{Ji}^{(k)}$, Equation (2.8) becomes equivalent to Equation (2.2). Thus, the sums R_1 and R_2 are now considered as line-of-sight signals transmitted from a different location. As a result, FASTANTENNACONTROL algorithm and our jamming cancellation technique are still applicable. Our experimental results for indoor environments (Section 2.6) confirm this conclusion.

Low-power jammer As mentioned in Section 2.3, the antenna control algorithms rely on the implication of minimum received power. In case of low-power jammer, minimizing total received power does not necessarily maximizes the SJR at the two-element antenna. However, the antenna algorithms result in the change in portion of jammer power in the total received power at the two-element antenna compared to

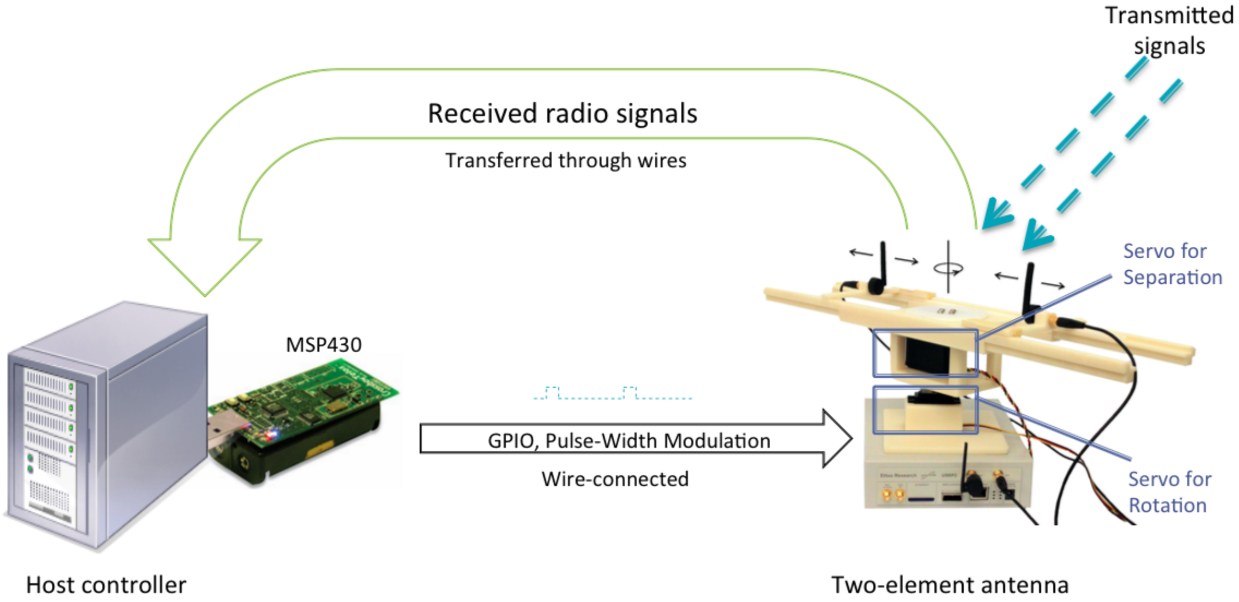


Figure 2.8: Receiver components and connections.

that at the single-element antenna, i.e., $h_{J1}/h_{S1} \neq h_{J2}/h_{S2}$, which allows obtaining the residual signal in Equation (2.3). Therefore, when combining with the digital stage, the antenna algorithms help eliminating the jamming signal. Although the first stage does not necessarily reduce the jamming power, it helps the second stage to derive the residual signal for successful decoding, thus is useful even for low-power jammers.

Variable-power jammer Recall the estimation of the gain ratio; as soon as the sender's power portion is determined, it can be used to derive the jammer's power portion (and hence their ratio a). Therefore, as long as the antenna remains in the same configuration, the power of the signal received from the sender is constant during the collision period, allowing the system to remove the variable-power jamming signal.

2.5 Prototype and Implementation

Our system consists of one receiver node and two transmitter nodes. We use GNU Radio SDR platform [16] to deploy our testbed. The digital signal processing is done by a host computer connected to the receiver.

Nodes Each node is deployed on an Ettus USRP device [30] with RFX2400 daughterboards. The jammer and sender use a single-element antenna for transmission. The receiver has a single-element and a two-element antenna for signal reception. All antenna elements are Titanis 2.4 GHz dipole Swivel SMA antennas.The receiver transfers digital samples to the host computer through an Ethernet link. The building

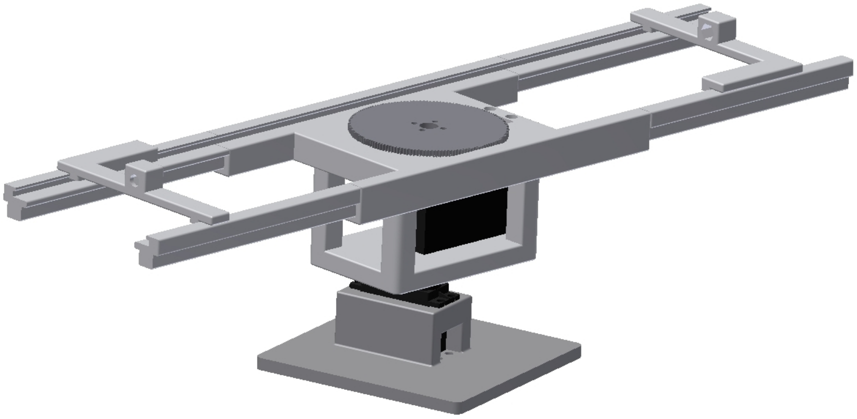


Figure 2.9: Antenna model.

components and connections of the receiver are depicted in Figure 2.8.

Two-element antenna Our two-element antenna (Figure 2.2) comprises two Titanis antennas. Signals received from two elements are added together through a HyperLink Technologies SCW02 combiner, which is then connected to one input of the receiver (the other input is connected to the single-element antenna). To build the antenna frame, we used Autodesk Inventor 2012 to design it (Figure 2.9) and built it using a uPrint Plus 3D printer [114]. The mechanical movement of the two-element antenna is controlled by two servos.

- **Rotation:** To rotate the antenna frame, we use a Hitec HS-485HB servo and attach the antenna frame to its rotating shaft. The HS-485HB servo is capable of rotating up to 200 degrees. However, we only need 180 degrees for half-circle rotation of the antenna, as two elements of the antenna are attached into the frame symmetrically with respect to the shaft. We set $\phi_{\min} = 0$ and $\phi_{\max} = \pi$ for the configuration algorithms.
- **Separation:** We use a Hitec HS-785HB servo (capable of rotating up to 3.5 circles) to transform the rotation movement to element separation by using a combination of gears and racks adjustable on the antenna frame. The frame allows the separation adjusted from $L_{\min} = 3.1\text{cm}$ to $L_{\max} = 37.5\text{cm}$.

The servos operate based on Pulse-Width Modulation signals given to their input. To generate those signals from the host computer, we use a Crossbow TelosB mote for receiving commands from the host and generating signals with appropriate pulse-width. The antenna controlling programs running on TelosB and the host computer are written in C and Python, and the digital signal processing stage is done with Matlab.

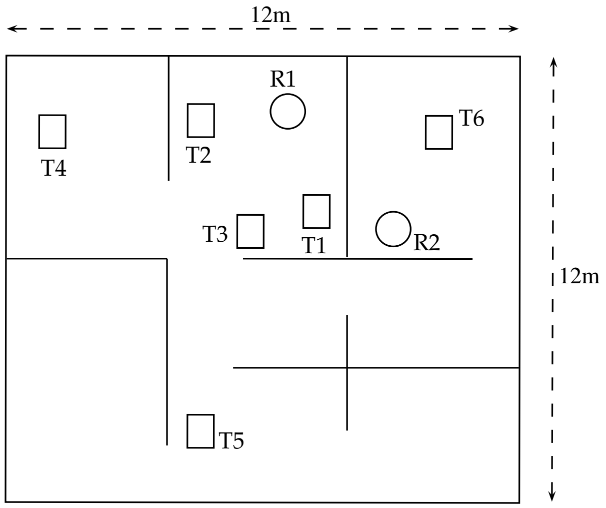


Figure 2.10: Experimental setup with 3 nodes in different locations.

2.6 Evaluation

In this section, we evaluate our system for indoor environments using three nodes: jammer, sender, and receiver. In our testbed environment, there are usual blocking objects and reflectors, such as walls, desks, metallic cabinets, and office space separators. We run the testbed at a fixed frequency of 2.4GHz (wavelength $\lambda \approx 12.5$cm). We carry out our experiments in different scenarios, each is configured to a different topology. The locations of transmitter and receiver are shown in Figure 2.10. We always fix the jammer at location T1, while varying locations of the transmitter to T2–T6, and locations of the receiver to R1 and R2. We report the experimental results based on the average performance over all experiments.

2.6.1 Antenna Configuration

2.6.1.1 Basic Operations

Two basic operations of the two-element antenna are the rotation and the separation adjustment. We measure the performance of those operations in terms of running time.

- **Rotation:** The half-circle rotation takes roughly 1 second to rotate the antenna frame from 0 to π. More precisely, it takes $T_R(\phi_1, \phi_2) = |\phi_1 - \phi_2|/\pi$ seconds to rotate from ϕ_1 to ϕ_2. The rotation servo is capable of rotating in sub-degree step.

- **Separation:** The separation adjustment takes about 2 seconds to increase the separation from 3.1cm to 37.5cm. More precisely, it takes $T_S(L_1, L_2) = \frac{2|L_1 - L_2|}{L_{\max} - L_{\min}}$ to change the distance between two antenna elements from L_1 to L_2. The minimal separation step is roughly 3.5 mm.

2.6.1.2 Brute-Force Algorithm

The brute-force algorithm is evaluated in terms of running time and capability of reducing jamming power. We deploy three nodes in a typical indoor environment. The jammer is set to transmit at 30dB higher power than the sender. Figure 2.11 shows the running time versus the power received at the two-element antenna relatively to the minimum value during the brute-force search. In this specific setup, using brute-force can eliminate up to almost 30dB of the jammer's power. Depending on the environment, the optimal configuration can be found at different time and the capability of reducing jamming power may vary. The total time to complete the brute-force search is more than 5 minutes as it tries all possible configurations.

2.6.1.3 Fast Algorithm

In order to evaluate the performance of the fast algorithm, we run the fast algorithm with the same setup (same node locations and same settings of transmit power). The capability of reducing jamming power is shown explicitly in two steps in Figure 2.12. While the first step (rotation only) can find a configuration that reduces the received power to more than 15dB, the second step (separation adjustment only) helps improving the power reduction of the jammer to roughly 25dB, which is not far compared to the performance of the brute-force algorithm. The running time of the fast algorithm in this experiment takes only 5 seconds to complete. We note that the running time of the fast algorithms depend on the environment. Table 2.2 summarizes the performance of the brute-force and fast algorithms in various experiments with different setups.

Table 2.2: Comparison of brute-force and fast algorithm

	Brute-force	Fast
Reduction of power	15-30dB	15-28dB
Reduction compared to brute-force in each experiment	–	< 6dB
Running time	> 5mins	5-18s

2.6.2 Anti-Jamming Performance

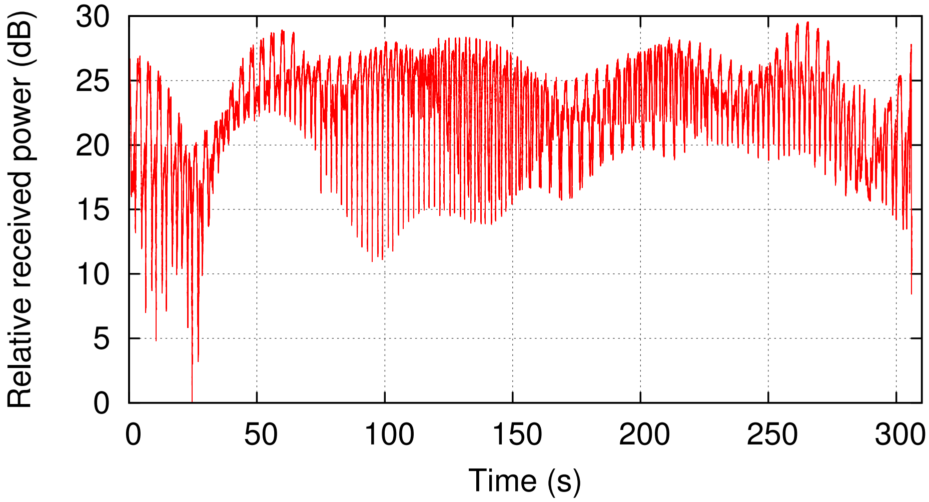


Figure 2.11: Brute-force: total received power relative to total received power's minimum value during search.

2.6.2.1 DBPSK Modulation

We investigate the performance of our system by examining the probability of bit error of the decoded data after removing the jamming signal. In this experiment, we use basic DBPSK modulation for data transmission between sender and receiver and for generating the jamming signal of the jammer. The bit rate used by sender is 500kbps. The receiver runs continuously during the experiment. In order to investigate the probability of bit error, sent and received signals are recorded at each node and later transferred to the host computer to compare and count the error bits. In the experiment, we keep the power of the sender constant and increase the power of the jammer gradually after each run to a threshold that the data becomes undecodable.

To evaluate our system's performance, we compare three cases: (a) decode the received signal directly from the receiver's single-element antenna, i.e. without any anti-jamming technique, (b) decode the received signal from the receiver's two-element antenna, and (c) decode the residual signal after applying the digital jamming cancellation. The average probability of bit error is presented in Figure 2.13. We visualize the BER in absolute (not log-scale) to make it easier to show the relative gain between combinations of techniques. Without the antenna auto-configuration capability (AA) and digital jamming cancellation (DC), the probability of bit error at the single-element antenna increases quickly when the jamming-to-signal ratio is greater than 3dB. Using the antenna auto-configuration with fast algorithm, the receiver can resist the jammer up to 28dB. The overall anti-jamming performance of the system is around 48dB when we combine two stages.

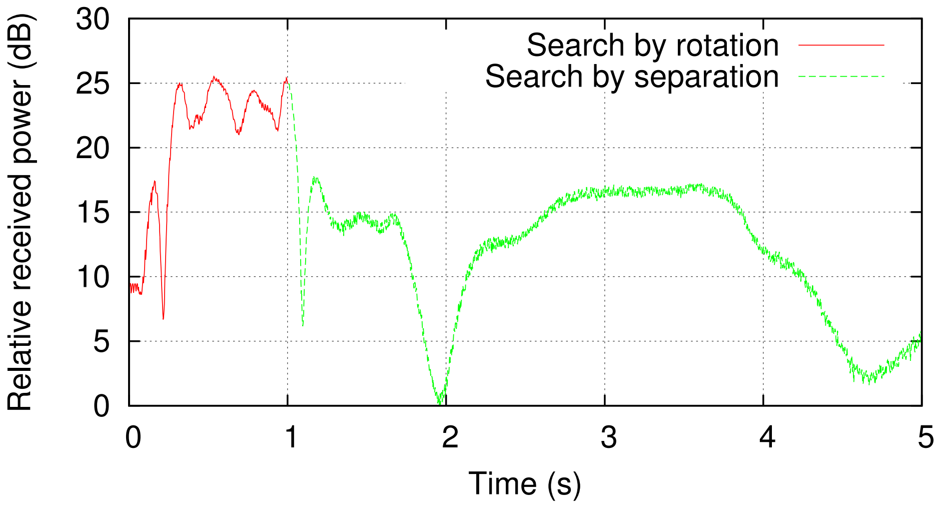


Figure 2.12: Fast: total received power relative to total received power's minimum value during search.

The results demonstrate that our system is able to work efficiently in indoor environments.

2.6.2.2 DQPSK Modulation

To study the effects of a higher-rate modulation on the performance of our system, we repeat the above experiments with DQPSK modulation at a doubled bit rate of 1Mbps.

Figure 2.13 compares the probability of bit error between DBPSK and DQPSK modulation. The performance of the system, when using DQPSK modulation, is around 42dB. Compared to the case of DBPSK modulation, the efficiency of the anti-jamming capability drops around 4 to 5dB. This is not surprising, since the constellation of the DQPSK modulation has a smaller minimum distance which results in higher probability of bit error [88]. Considering only the performance of the digital jamming cancellation, there is no significant difference in the capability of jamming cancellation between the two cases. This shows the efficiency of the estimation techniques applied in the second stage.

2.6.2.3 Variable Power Jammer

In the above experiments, the jammer transmitted at constant transmission power. To evaluate our system against a variable-power jammer, we modify the jammer such that after every 40 bytes it changes the transmit power to a random level within the range of 10 dB compared to the specified average power in each run. For this experiment, we use DBPSK modulation. We note that during the experiment, the antenna configuration

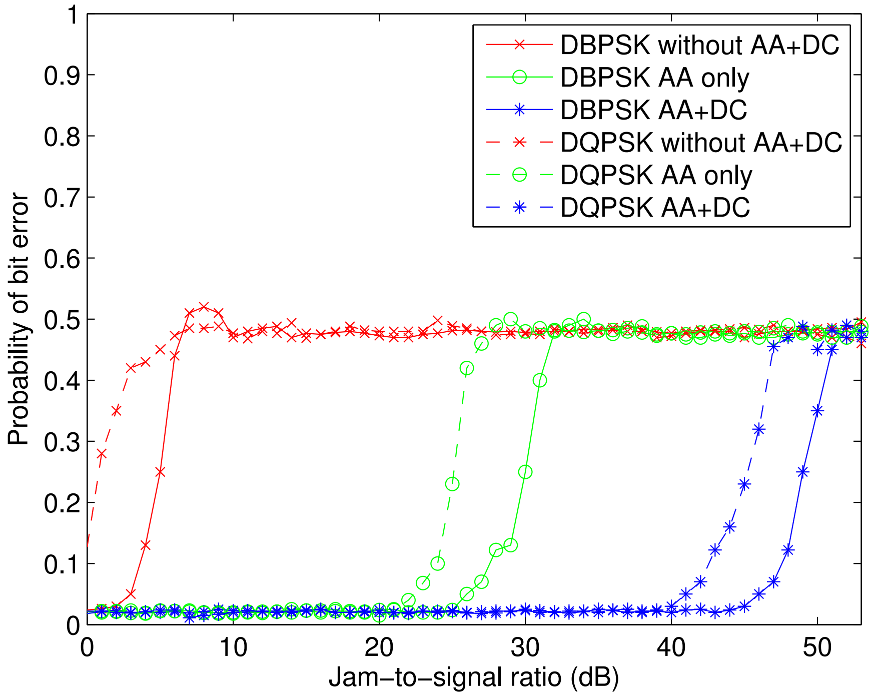


Figure 2.13: DBPSK and DQPSK modulation

does not change and is capable of removing a portion of about 28 dB in jamming power. Figure 2.14 shows the comparison between variable and constant jamming power cases in probability of bit error versus the average power in each run. The results show a performance degradation of 5-6 dB, demonstrating that the gain estimation is adaptive to the change of jamming power as long as the sender's power and the antenna remain unchanged.

2.7 Related work

Anti-jamming has been an active area of research for decades. Techniques developed at the physical layer [104] include directional antennas [60], spread spectrum communication, and power, modulation, and coding control. More recently, research has also addressed higher layers [4, 9, 25, 29, 41, 56, 59, 66, 67, 86, 105, 107, 108, 119, 122, 123]. However, given the ease of building *high power* jamming devices, there is still a strong need for efficient and flexible techniques operating at the physical layer. There is a demand for low-cost solutions mitigating the effects of jammers that are orders of power stronger than legitimate communication.

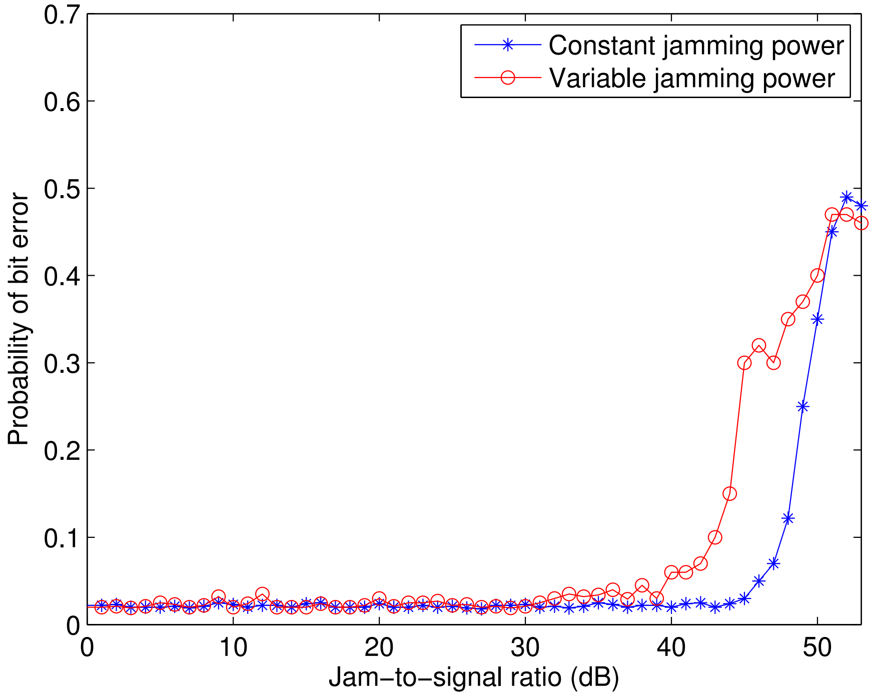


Figure 2.14: Variable vs. constant power jamming

While spread spectrum has been a solution of choice for anti-jamming, it suffers from a need for pre-shared secrets between the communicating nodes. Several solutions were recently proposed for alleviating the need for pre-shared secrets [2, 22, 41, 56, 65, 67, 107, 108]. However, they are not designed to tackle powerful jammers (meaning jammer with power 4-5 orders of magnitude higher than the transmitting node).

Other recent work has demonstrated mechanisms for cancelling interference. This work has found applications in protecting the confidentiality of communication [27, 44, 110]. However cancelling *powerful*, *unknown* jammers results in several challenging problems such as jammer signal identification and channel estimation.

The closest related work to our approach consists of phased array antennas and MIMO systems. Phased array antennas were very well studied since the 1950s [18, 19, 60, 111]. Likewise, MIMO systems were also very well studied since the mid 90s [112]. Our design and approach have unique characteristics that distinguishes them from prior work. Similar performance phased array antennas consist of a fairly large number of *fixed*-position elements aiming at creating a directed beam that can be electronically and digitally repositioned. Various adaptive beamforming algorithms have been studied with the goal of minimizing the impact of sidelobes. For example, MMSE (Minimum Mean Squared Error) approaches aim to adjust weights on array elements such that the error with respect to a referenced signal is minimized. Alternatively, approaches based on MVDR (Minimum Variance Distortionless Response) mitigate interference by minimizing the received signal given the knowledge of propagating channel. The effectiveness of these approaches heavily rely on the training phase, and they use a fairly large number of antennas and are considered to be more adequate for radar systems. In contrast, our system's goal is to create one or multiple nulls to minimize the jammer's impact while maximizing the legitimate user signal power and preparing the signal for a digital MIMO-like second stage of interference cancellation. Existing phased array antennas achieving a gain of 48dB require hundreds of elements even with high-end, expensive 7-bit phase shifters [18, 69]. Our two-elements mechanical steering can be controlled with low-cost micro controllers instead of requiring expensive DSP boards. Our second-stage digital jamming cancellation is in principle similar to MIMO. However, existing algorithms assume that the incoming signals are of similar power, transmitted by a cooperating node, with the possibility to embed training sequences for the channel estimation. Furthermore, MIMO-like digital beam-forming is not efficient against powerful jammers because of the limited dynamic-range of RF front-ends and ADCs (which are typically 12 to 14 bits).

2.8 Conclusion

The availability of software radios and commodity jammers are making jamming of wireless communication a problem of increasing importance for many cyber-physical applications. To mitigate the problem of jammers that are significantly more powerful than the transmitting nodes, we have designed, physically built, and evaluated a hybrid system of mechanical beam/null-forming and MIMO-like digital interference cancellation. Our novel antenna design and algorithms have several important characteristics and advantages compared to phased array antennas and MIMO techniques e.g., simplicity, low-cost, convergence speed. It allows a flexible creation of multiple nulls to cancel the effects of multi-path jamming. We have developed several techniques to effectively cancel the remaining interference digitally and verified their effectiveness in practice. To the best of our knowledge, this is the first academically published low-cost system that reduces the effects of powerful unknown jammers by almost five orders of power.

Chapter 3

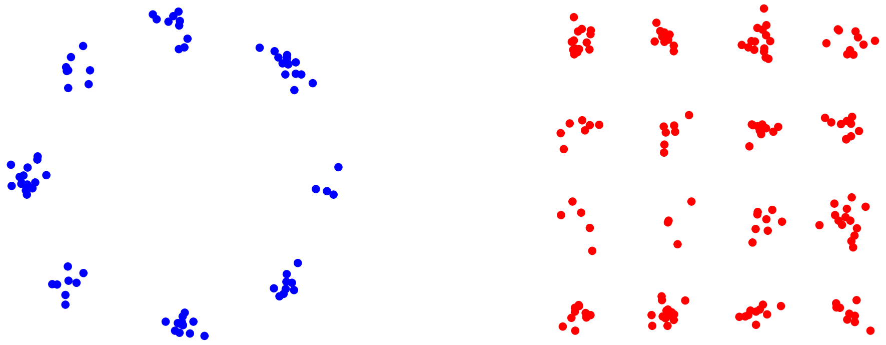
CBM – Concealing Rate Information and Boosting Resiliency

In this chapter, we consider jamming attack scenarios in which countering solutions based on smart steerable antennas are not applicable, e.g., Wi-Fi networks. In those scenarios, crippling jamming such as attacks on link rate adaptivity is very efficient even with limited-power adversaries. These attacks are enabled by the exposing of the rate information of wireless transmissions. In Section 3.1, we discuss the source and attacks that can be deployed to learn the rate information, then we show that it is challenging to hide the rate information from the adversary (Section 3.2). Our proposed solution is based on a generalization of Trellis Coded Modulation combined with Cryptographic Interleaving, whose details are discussed throughout Sections 3.3 to 3.6. Our evaluation is reported in Section 3.7. Finally, we summarize the related work in Section 3.8 and our work in Section 3.9.

3.1 Rate Information Leakage

3.1.1 Explicit rate information

In many communication protocols, the rate information of a transmission is unprotected. For instance in IEEE 802.11 networks, the rate is explicitly specified in the SIGNAL field of the physical layer's frames. An adversary can easily synchronize with the communication between two parties, analyze the data frames and extract the rate. This attack is very practical as demonstrated by [74]. Similarly, the rate information of a transmission in LTE cellular systems is exposed in the Modulation and Coding Scheme (MCS) field within the Downlink Control Information (DCI), which is itself encoded using a publicly known fixed rate $1/3$ convolutional code and QPSK modulation.



(a) 8-PSK **(b)** 16-QAM

Figure 3.1: Constellation guessing on USRP: Received symbols can be distinguished clearly after carrier synchronization and tracking.

3.1.2 Modulation guessing

Even if the rate information is not explicitly provided within the packet header, the adversary can analyze the received signal in complex I/Q form. After performing the carrier synchronization, frequency and phase offset correction, the adversary can trace the received constellation pattern and determine the modulation in use. This method does not require the knowledge of the protocol's frame structure. We demonstrate the guessing attack by implementing a modulation detector on USRP, which can in real time identify the transmission's modulation (Figure 3.1). It can be easily extended to build a practical rate-aware jammer [74, 119] to selectively jam the high rate packets.

3.1.3 Code guessing

With more sophisticated techniques, an adversary could manage to identify not only the transmission's modulation, but also the codes in use. One such technique is to track the sequence of received symbols to guess the codes based on the fact that different codes produce different transitions from one coded symbol to another. Since most communication standards specify a limited set of modulations and codes, guessing by matching and trial-and-error is efficient for the adversary.

3.2 Challenges to Rate Concealing

Given the possibilities of rate information leakage discussed in the previous section, we now provide some insights why it is hard to hide the rate information.

3.2.1 Data Encryption

First, to hide the explicit rate information in the packet's protocol header, a straight-forward solution would be to encrypt the header. However, doing so does not prevent the modulation guessing attack, as a transmission still exposes the modulation to the complex I/Q analyzing adversary, who does not even need to decrypt the header to learn the rate.

3.2.2 Using Single Modulation

Alternatively, in order to avoid modulation guessing, a naive solution could be always using only one modulation for communications. While this makes adversaries no more interested in the rate information, it would not benefit the system, because that requires the system to operate at low rate transmission to be resilient against jamming, which, nevertheless, results in low bandwidth efficiency. Using single modulation also lacks the flexibility and adaptivity to the environment, such as preventing the user from benefiting from high data rates at high SNR. As a consequence, the adversary immediately achieves the goal of degrading the network performance without the need of jamming.

3.2.3 Modulation Level Encryption

A very recent work [90] proposed a modulation level encryption technique to hide the rate of communications. In essence, this technique always transmits with the highest order modulation, but the communicating nodes cryptographically agree on a subset of the constellation points to be used for each symbol. For example, BPSK modulation can be hidden in 16-QAM by only considering eight pairs of points. For every symbol to be transmitted a pair is cryptographically selected by the transmitter and is also known to the receiver through a shared key. The information bit of BPSK determines which element of the pair is sent. Since the eight pairs cover the whole constellation points, the adversary cannot distinguish between a BPSK communication embedded within 16-QAM or a true 16-QAM. While this scheme conceals the rate information, it does so at the cost of degrading the robustness of the communication. First, one can analytically show that 1-2dB are lost because of the constrained selection of the constellation pairs. Several additional dB are lost due to the poor performance of frequency offset correction, and phase tracking techniques in higher order modulations.

Rate Concealing Techniques	Counter Against			Robustness
	Explicit Rate	Modulation Guessing	Code Guessing	
Data Encryption	✓			
Single Modulation	✓	✓		
Mod. Encryption	✓	✓		
Any above + Bin. Coding	✓	✓		Not guaranteed
This work	✓	✓	✓	✓

Table 3.1: Summary of rate concealing techniques against various rate information leakage attacks.

Recently, the modulation level encryption technique developed by Rahbari and Krunz [90] can hide the rate from modulation guessing by a random mapping into a higher-order modulation based on a shared secret between two parties. However, their solution sacrifices the resiliency by 1-2dB in simulation, and several dB in real world due to imperfect frequency offset correction.

3.2.4 Coding

One plausible approach to hide the implicit rate information, leaked by the constellation points, consists of always using the highest order modulation and combining it with a matching rate (e.g., uncoded BPSK can be hidden by using 16-QAM combined with a 1/4 rate coding scheme). As will be discussed and demonstrated in the next sections, such approach does not perform well, mainly because: (1) At low SNR several key components of the communication chain perform poorly, therefore degrading the performance of the system. These include the coding schemes, the frequency offset, and the phase tracking. This is one of the key reasons why most standards (both WLAN and Cellular) still rely on BPSK as a fallback solution for low SNR regimes. (2) Traditional codes maximize the Hamming distance between codewords and not the multi-dimensional Euclidean distance necessitated by the I/Q constellation of coded high order modulations. (3) An adversary can still guess the rate information by trying all possible modulation schemes. Traditional encryption schemes cannot prevent this attack since if they are applied post-coding/modulation they would render the error correction properties of the code useless (a single bit error would be amplified by the decryption process).

Table 3.1 summarizes the rate concealing challenges, effects and trade-offs of aforementioned countering methods.

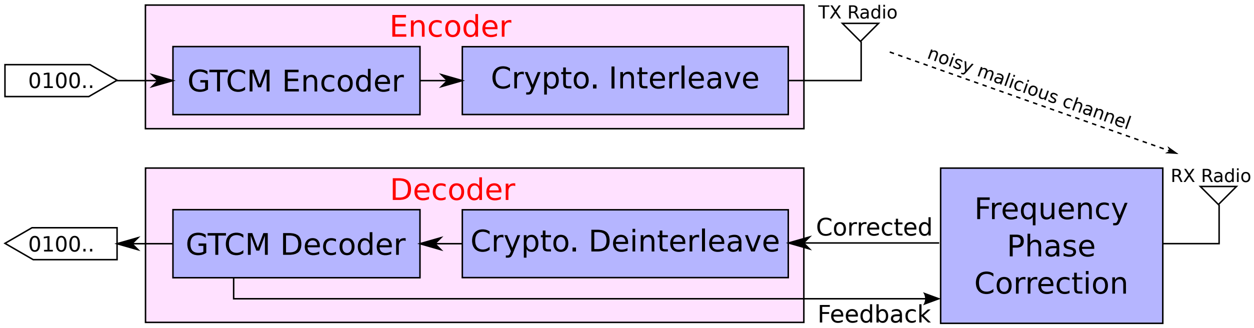


Figure 3.2: Overview of our CBM system.

3.3 Approach

Our scheme – CBM (Conceal and Boost Modulation) – is depicted in Figure 3.2. The General Trellis Coded Modulation (GTCM) module's functionalities are two-fold. First, it makes the constellation pattern indistinguishable to the adversary, therewith countering the modulation guessing attacks. Second, it boosts the system resiliency against interference. The Cryptographic Interleaving (CI) module conceals the rate information from explicit rate exposing and implicit code guessing attacks.

Our idea for hiding the constellation is to always use a *single unifying* modulation (the highest order) to transmit data in order to create the same constellation observed by the adversary. To preserve the bit rate and robustness supported by the original modulation, the GTCM module encodes the data by a suitable code of rate matching the bit rate ratio between the original modulation and the target modulation. To be precise, let's consider a system that supports a set of different modulations ordered by the number of bits per symbol (bps) $b_1 \leq \ldots \leq b_{\mathscr{N}}$, where $\mathscr{N}$ denotes the highest-order modulation of bit rate $n = b_{\mathscr{N}}$. Assume that the on-going transmission is desired to be carried at a bit rate $k = b_{\mathscr{K}}$ bps for some modulation $\mathscr{K}$. In order to conceal the constellation, we encode the data using an adequate code of rate k/n and transmit the encoded data using the target modulation $\mathscr{N}$. Since the adversary will always observe the same constellation $\mathscr{N}$, the actual rate is concealed from modulation guessing attacks.

To counter the code guessing attacks, we develop a cryptographic module, which interleaves the modulated symbols before transmission. We emphasize that the interleaving process is performed at the baseband samples level, i.e., complex symbols produced by the GTCM module are interleaved per block of transmitted symbols. We note that straightforward encryption of data *before* modulating does not conceal the rate information, as the adversary can clearly observe the constellation of encrypted data. We derive a specific method to efficiently generate cryptographic interleaving functions used for permuting the output symbols from the GTCM module in such a way that the transmit stream does not leak the rate information. For the

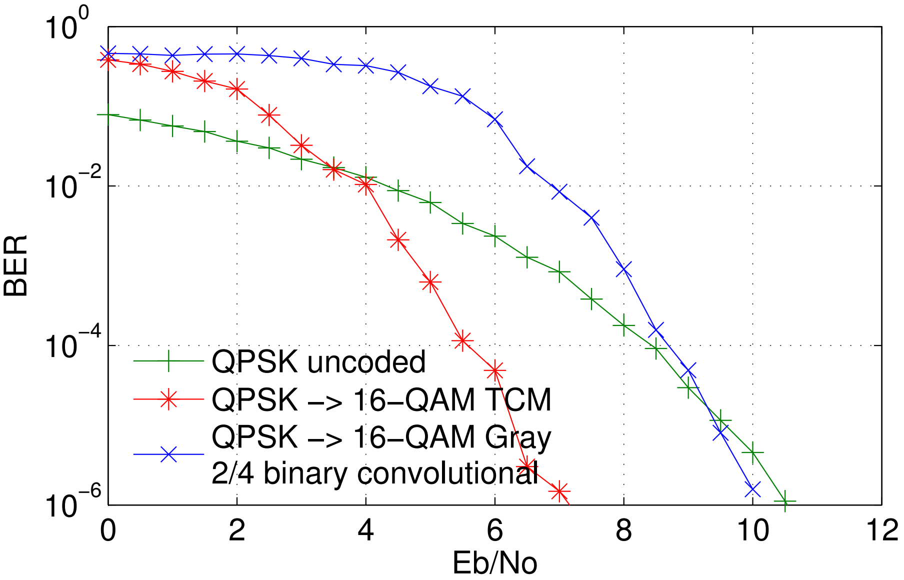


Figure 3.3: Performance comparison between (1) our TCM code with standard 16-QAM, and (2) best traditional binary code (from [24]) of rate $2/4$ with Gray coded 16-QAM.

receiver to be able to decode the data, the rate information is embedded into the packet in an encrypted form such that only the receiver, who shares the secret key with the transmitter, can decrypt the information.

It is important to understand the implications of rate hiding. On one hand, the highest-order modulation creates redundancy by the constellation expansion. On the other hand, the constellation points' pair-wise distances are closer than in the original constellation. Without good design specifically targeting to the upgraded modulation's constellation, the system can become less resilient against interference. For example, the modulation unification technique used in [90] results in the system robustness 1-2dB less than regular rate-exposing systems. The reduced resiliency is because no coding is used in their system. However, even using good traditional binary codes cannot guarantee the robustness of the system because they maximize the Hamming distance between codewords and are not designed for coded modulation. An illustration is shown in Figure 3.3. We take the best code $\begin{pmatrix} 17 & 13 & 05 & 02 \\ 10 & 03 & 17 & 15 \end{pmatrix}$ of rate $2/4$ from Table VII in [24], and use it with Gray coded 16-QAM modulation. Comparing it with our derived TCM code $\begin{pmatrix} 01 & 12 & 16 & 11 \\ 01 & 13 & 16 & 11 \end{pmatrix}$ of the same rate and constraint, we see a gain of about 4dB is achieved with our code (Figure 3.3), while the binary code almost gives no advantage over uncoded QPSK at BER $= 10^{-6}$. Therefore, with good codes designed for the target modulation, we can gain instead of losing.

Searching for good codes for the rate-hiding systems must take into account the constellation description defined by the highest-order modulation. This idea is rooted in the Trellis Coded Modulation (TCM) technique introduced by Ungerboeck [113], who focused on devising modulation codes of rate $k/(k+1)$. In the literature, finding good TCM codes is a challenging problem, for which only heuristic solutions have been studied such as the set partitioning rules established in [113]. Unfortunately, there is no polynomial time algorithm for constructing the optimal general TCM codes. In this work, we introduce a new heuristic approach for upgrading arbitrary modulations. Our heuristic solution is not based on the conventional set partitioning technique. Instead, we generate the code based on the general structure of a convolutional code. Specifically, we randomize the mapping between the inputs, shift registers, and the outputs. Our experimental results show that this approach can find codes at least as good as the ones found in [113]. In some cases, we obtain better codes than those in [113].

3.4 General TCM

In this section, we describe the fast search procedure for TCM codes of arbitrary rate k/n, which are used to encode data originally modulated by a modulation $\mathscr{K}$ of order 2^k (bit rate k) to the highest-order modulation $\mathscr{N}$ of order 2^n (bit rate n), without any uniformity restriction. For convenience, we give a brief overview of TCM codes. A TCM code is a convolutional code (k,n) defined by a set of k shift registers storing the code's k input bits, and a generator matrix which specifies the input-output relation (e.g., Figure 3.4). Deeper shift registers have higher potential gain and decoding complexity. Thus, we classify the codes by their constraint length v defined as $v = \sum_{i=1}^{k} v_i$, where v_i is the length of the i-th shift register. In our search procedure, we only consider the feed-forward construction of convolutional codes, because any construction with feedback can be transformed into a feedback-free construction that produces equivalent codewords [64]. To represent a code, we use the conventional generator polynomial form $G(D) = \{g_{ij}(D), i = 1\ldots k, j = 1\ldots n\}$, where $g_{ij}(D) = \sum_{l=0}^{v_i} a_l D^l$ is a univariate polynomial, and the indeterminate D represents the delay of the input bit in the corresponding shift register. If $a_l = 1$, the i-th input's current value (for $l = 0$) and past values (for $l > 0$) are $GF(2)$ added (exclusive-or) to the j-th output. For example, the convolutional code $(2,4)$ in Figure 3.4 has the generator matrix/polynomials $G = \begin{bmatrix} D+D^2 & D^3 & D+D^3 & 1+D+D^2 \\ 0 & 1 & 0 & 1+D \end{bmatrix}$.

Unlike binary convolutional codes whose performance depends on the Hamming distance of the binary output symbols, TCM codes' performance is determined by the free *Euclidean* distance d^∞, which is the minimum Euclidean distance of any two distinct complex-symbol sequences produced by the code and modulation $\mathscr{N}$. Since binary codes are not designed for coded modulation, they do not take into account the constellation mapping. The best binary code with regards to Hamming distance can have a significantly small Euclidean distance between transmitted complex symbols and result in poor performance when com-

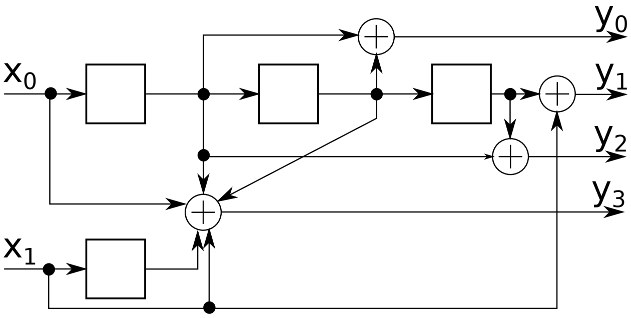


Figure 3.4: Our best TCM code of rate $2/4$ (QPSK $\rightarrow$ 16-QAM) and constraint length 4. Its boosting gain over uncoded QPSK is 3.8dB.

bined with a specific modulation. For example, Figure 3.3 shows that our TCM code (QPSK→16-QAM) outperforms the best binary $2/4$ convolutional code in [24] applied to the Gray-coded 16-QAM modulation, by 3dB at BER=10^{-6}. In the search for good TCM codes, we use as comparison metrics the asymptotic coding gain ratio measured by $\beta = d^{\infty}_{\mathscr{N}}/\Delta_{\mathscr{K}}$, where $\Delta_{\mathscr{K}}$ is the minimum Euclidean distance between constellation points in the original modulation $\mathscr{K}$. Good TCM codes must have high β ratio.

3.4.1 Code search algorithm

We introduce a new heuristic approach for searching for good TCM codes. For a given code specification $(k,n,\{v_i\})$, the coefficients of the generator polynomials g_{ij} are randomly selected. Since each combination of coefficients corresponds to a unique code construction, we check the generated code for its free Euclidean distance and if it satisfies additional critical properties such as structure information leakage. The search is performed for a fixed number of trials independent of the code specification, thus it is substantially faster than a full search which evaluates all possible codes. Yet, as shown in Section 3.4.3, our randomization approach can achieve the same results as a full search. The search procedure is illustrated in the RandomCodeSearch algorithm below.

Our random search is characterized by the number of trials T performed by RandomCodeSearch. A randomly generated code, might be (1) *catastrophic*, i.e., there exists a non-zero input sequence that can produce all-zero output sequence; or (2) *non-equiprobable*, i.e., the output values are not uniformly distributed, which can help the adversary deploy statistical attacks to distinguish the mapping we aim to conceal. Therefore, the generated code is first validated against above properties, then its free distance d^{∞} is computed.

```
RandomCodeSearch($k,n,\{v_i\},\mathscr{M},T$)
$d^\infty = 0$                // free distance of current best code
for $i = 1$ to $T$
    $C$ = generateCode($k,n,\{v_i\}$)
    if valid($C$)   // non-catastrophic and equiprobable check
        $d$ = ComputeDistance($C,\mathscr{M},d^\infty$)
        if $d > d^\infty$
            $d^\infty = d$      // update free distance
            $C^* = C$      // store new best code
return ($C^*,d^\infty$)
```

3.4.2 Free distance computation algorithm

The computational bottleneck of the code search lies in the computation of the Euclidean free distance, since it is performed for every generated code. In the conventional TCM code construction method based on set partitioning rules [113], computing the free distance only involves finding the minimum distance to the all-zero sequence. However, our GTCM approach does not restrict the search by the set partitioning rule, as we consider a higher-dimension space so that better codes can be found (including non-uniform ones). As a result, computing the free Euclidean distance involves all pairs of output sequences. Nevertheless, we devise an efficient algorithm – ComputeDistance – whose running time is on average less than 2ms, on a 3GHz CPU desktop computer, for the modulations and depths we consider. ComputeDistance's algorithm consists of traversing the trellis of the code and appropriately updating the *state-distances*, which is defined below.

First, we introduce some convenient notations. Let $I = \{0,\ldots,2^k-1\}$ be the set of inputs, $O = \{0,\ldots,2^n-1\}$ the set of output symbols, and $\Lambda = \{0,\ldots,2^v-1\}$ the set of possible states corresponding to a code C. A path P of length L is defined as a sequence of 3-tuples $P = \{(S_i,x_i,y_i), i = 0\ldots L-1\}$, where $S_i \in \Lambda, x_i \in I$ are respectively the state and input of the code at time i, and $y_i \in O$ is the output symbol due to input x_i at state S_i. The encoding can start from any initial state S_0. Since the output symbol is mapped to a specific constellation $\mathscr{M}$, the Euclidean distance between two paths P and $\tilde{P}$ of length L is dependent on $\mathscr{M}$ and computed by $d_{\mathscr{M}}(P,\tilde{P}) = \sum_{i=0}^{L-1} d_{\mathscr{M}}(y,\tilde{y})$, where $d_{\mathscr{M}}(y,\tilde{y})$ gives the Euclidean distance between two points y and $\tilde{y}$ on the target coded modulation $\mathscr{M}$'s constellation. Now, we define the state-distance $D[S,\tilde{S}] \triangleq \min\{d(P,\tilde{P})\}$ of two states S and $\tilde{S}$ as the minimum Euclidean distance between all possible paths of the same length ending at state S and $\tilde{S}$, respectively.

The key idea of our algorithm is that we update $D[S,\tilde{S}]$ gradually when traversing the trellis with increasing L. When two paths P and $\tilde{P}$ merge at the same state $S = \tilde{S}$, the free distance d^∞ is checked and updated with $D[S,\tilde{S}]$.

```
COMPUTEDISTANCE($C, \mathscr{M}, d^{\infty}_{best}$)
 1  $D[S,\tilde{S}] = \infty$ for all $(S,\tilde{S}) \in V^2$          // state-distances
 2  $d^{\infty} = \infty$                                  // $C$'s free distance
 3  for each $S \in \Lambda$, $(x,\tilde{x}) \in I^2$, $x \neq \tilde{x}$
 4      UPDATEDISTANCE($S,x,S,\tilde{x}$)
 5  repeat
 6      for each $(S,\tilde{S}) \in \Lambda^2, S \neq \tilde{S}, D[S,\tilde{S}] < d^{\infty}$
 7          for each $(x,\tilde{x}) \in I^2$
 8              UPDATEDISTANCE($S,x,\tilde{S},\tilde{x}$)
 9              if $d^{\infty} \leq d^{\infty}_{best}$
10                  return $d^{\infty}$                    // not the best, return now
11  until $(S,\tilde{S})$ not found in line 6
12  return $d^{\infty}$                                    // we found a better code
```

The algorithm COMPUTEDISTANCE starts by initializing the state-distances to the distance between any path P and $\tilde{P}$ starting from any *same* state S (line 1–4). We make the paths diverge from the same state (line 3), then compute the distance between them (line 4). In the main loop (line 5–11), the state-distances are repeatedly updated for each new segment added (line 7) to the paths until there exist no more state pairs $(S,\tilde{S})$ whose state-distance $D[S,\tilde{S}]$ is less than d^{∞} (line 11). The maintenance and update of state-distances in both the initialization and the main loop are performed by UPDATEDISTANCE, which keeps records of $D[S,\tilde{S}]$ for all $S,\tilde{S}$. Whenever two paths P and $\tilde{P}$ merge at a state S, the corresponding state-distance $D[S,S]$ is checked to update d^{∞}.

```
UPDATEDISTANCE($S,x,\tilde{S},\tilde{x}$)
$T = C.nextState(S,x)$; $y = C.output(S,x)$
$\tilde{T} = C.nextState(\tilde{S},\tilde{x})$; $\tilde{y} = C.output(\tilde{S},\tilde{x})$
if $S = \tilde{S}$
    $d = d_{\mathscr{M}}(y,\tilde{y})$                     // update at initialization
else
    $d = d_{\mathscr{M}}(y,\tilde{y}) + D[S,\tilde{S}]$          // update loop

if $d < D[T,\tilde{T}]$
    $D[T,\tilde{T}] = d$
    if $d < d^{\infty}$ and $T = \tilde{T}$                // two paths merge
        $d^{\infty} = d$
```

Furthermore, in the random search procedure discussed previously, where each generated code is computed for the free distance, we speed up the search by storing the best free distance d_{best}^{∞} associated to the best code C^* discovered so far in order to quickly eliminate codes of free distance shorter than d_{best}^{∞} (line 9 in COMPUTEDISTANCE).

Correctness. To prove the correctness of the algorithm, we show that the state-distances $D[S,\tilde{S}]$ keep records of the distances of all possible "close" paths. The proof is based on the following lemma.

Lemma 3.4.1. *At any time i on the code trellis, for any pair of paths P and $\tilde{P}$, if there exists another pair of paths Q and $\tilde{Q}$ such that $S_i^{(P)} = S_i^{(Q)}$, $S_i^{(\tilde{P})} = S_i^{(\tilde{Q})}$, and $D[Q,\tilde{Q}] < D[P,\tilde{P}]$, then P and $\tilde{P}$ can be eliminated.*

(sketch). By the lemma's assumption, P merges with Q and $\tilde{P}$ merges with $\tilde{Q}$ at time i. It is followed that at time $i+1$, any new pair evolved from P and $\tilde{P}$ will find a similar new pair evolved from Q and $\tilde{Q}$. Therefore, the pair $(P,\tilde{P})$ cannot have shorter distance and can be eliminated from searching. □

Proof of correctness. At initialization of COMPUTEDISTANCE, $D[S,\tilde{S}]$ are set to non-infinity values only for pairs of paths starting from the same state (line 3). This means that $D[S,\tilde{S}]$ properly reflect the distances of paths at initial states. In the main loop, the algorithm traverses every transition of the trellis and updates the state-distances. By Lemma 3.4.1, the macro UPDATEDISTANCE will discard paths corresponding to greater distance $D[S,\tilde{S}]$ and keep the ones corresponding to the shortest distance so far (line 9). Therefore, no closest pairs are eliminated by the algorithm.

To see that the algorithm terminates, we show that there exists a time such that $D[S,\tilde{S}] \geq d^{\infty}$ for all state pairs. It is enough to show that $D[S,\tilde{S}]$ are increasing while d^{∞} is decreasing. The former is correct because evolving paths always contain transitions that results in positive increment in distance. The latter is due to the update in UPDATEDISTANCE. This concludes the proof.

Computational Complexity. The time complexity $g(t)$ of COMPUTEDISTANCE depends on the length L of paths where the free distance is found. We estimate $g(t)$ in the worst case as follows. First, since UPDATEDISTANCE requires a constant number of operations, we judge the time complexity in terms of number of calls to UPDATEDISTANCE. The initialization of $D[S,\tilde{S}]$ requires 2^{v+2k} updates (line 3–4). At each iteration of time i in the main loop (line 5–11), the number of updates is at most 2^{2v+2k}. Therefore, the worst-case complexity of COMPUTEDISTANCE is $g(t) = 2^{v+2k} + 2^{2v+2k}L = O(2^{2(v+k)}L)$. In our experimental search results, we observe that the value of L can be bounded by $L \leq 3v$ for any code. The running time of the algorithm on a 3GHz CPU desktop computer is less than 2ms, which is significantly faster than the naive approach that compares all pairs of paths.

BPSK → QPSK			BPSK → 8-PSK		
v	β	G	v	β	G
1	1.76	(3 1)	2	3.72	(5 2 4)
2	3.98	(7 2)	3	4.77	(10 2 17)
3	4.77	(13 4)	4	5.36	(31 4 2)
4	5.44	(35 4)	5	6.02	(10 2 71)
5	6.02	(64 33)	6	6.53	(107 20 12)
6	6.99	(135 56)	7	6.99	(251 102 4)
7	6.99	(374 147)	8	7.24	(661 102 30)
8	7.40	(457 142)	9	7.63	(1715 336 400)
9	7.78	(1312 665)	10	7.78	(3575 1400 14)
10	8.45	(2175 1256)			

Table 3.2: BPSK → QPSK/8-PSK TCM codes.

3.4.3 Search results

In this section, we list the GTCM codes found by our randomization approach (Tables 3.2 to 3.6), and discuss their performance in comparison with traditional uniform TCM codes, as well as with a full search approach. The lists are compiled for codes of constraint length up to $v = 10$ and for each pair of original modulation $\mathscr{K}$ and target coded modulation $\mathscr{N}$ in {BPSK, QPSK, 8-PSK, 16-QAM, 64-QAM}. We note that previous work [113] discovered codes for only 1 bit constellation expansion. The asymptotic coding gain β is measured in dB, and the generator matrix G is presented in the standard octal form adopted from [24, 64]. Our symbol mapping of m-PSK constellations is $p(s) = e^{j2\pi s/m}$, where $s = 0 \dots m-1$ is the transmitted symbol, $j = \sqrt{-1}$. For square m-QAM constellations, our mapping is $p(s) = (x_0 + s_L\Delta_m) + j(y_0 + s_H\Delta_m)$, where Δ_m is the minimum separation in m-QAM, (x_0, y_0) are coordinates of the zero symbol ($s = 0$) located at the bottom-left corner of the constellation, and $s_L = s \bmod (\sqrt{m})$, $s_H = (s - s_L)/\sqrt{m}$ correspond to low-order and high-order bits of s. We note that (1) for some cases of short constraint length, there are no good codes with positive boosting gain; (2) our code search algorithm is independent of symbol mapping, thus can be used to find good generalized TCM codes for any constellation required by the application.

Comparison with uniform codes search. As an example shown in Table 3.4 for QPSK → 8-PSK, we achieve better codes than the ones in [113] for constraint lengths $v = 6, 8, 10$, which confirms the intuition that better codes can be discovered if the uniform mapping property of the set partitioning rules is relaxed.

Comparison with full search. To verify that the codes discovered using our randomization techniques

BPSK → 16-QAM			BPSK → 64-QAM		
v	β	G	v	β	G
3	3.42	(17 4 5 10)	5	3.94	(63 10 40 47 4 2)
4	4.15	(33 10 27 2)	6	4.63	(117 10 40 55 104 100)
5	5.05	(67 10 55 4)	7	4.91	(227 4 60 371 44 210)
6	5.31	(31 100 167 2)	8	5.17	(573 12 52 745 100 362)
7	5.8	(212 20 327 40)	9	5.35	(1747 16 344 1277 60 10)
8	6.13	(725 14 757 200)	10	5.61	(2473 12 3436 3341 62 6)
9	6.33	(1453 346 1137 30)			
10	6.63	(2653 16 3103 774)			

Table 3.3: BPSK → 16-QAM/64-QAM TCM codes.

QPSK → 8-PSK			QPSK → 16-QAM			QPSK → 64-QAM		
v	β	G	v	β	G	v	β	G
1	1.12	(3 2 1) (1 0 0)	2	2.55	(3 0 1 1) (1 2 2 1)	4	3.01	(1 0 2 3 0 1) (12 4 6 1 6 6)
2	3.01	(1 5 2) (1 0 0)	3	3.42	(5 2 0 6) (1 0 3 0)	5	3.41	(5 0 2 1 0 4) (12 4 4 15 2 10)
3	3.6	(1 4 2) (2 1 0)	4	3.8	(7 2 2 7) (2 0 5 0)	6	3.68	(1 0 2 3 0 0) (75 30 76 72 2 40)
4	4.13	(13 10 6) (3 1 0)	5	4.15	(1 2 3 0) (37 14 10 2)	7	3.94	(5 0 0 3 1 0) (66 44 20 71 26 6)
5	4.59	(1 4 0) (13 17 4)	6	4.47	(7 16 13 1) (11 16 12 1)	8	4.1	(61 54 34 46 32 34) (15 7 6 3 4 0)
6	5.01	(20 1 12) (3 4 0)	7	5.05	(17 0 4 0) (2 5 25 12)	9	4.26	(12 2 4 7 0 4) (73 62 6 110 16 24)
7	5.01	(7 2 0) (30 75 10)	8	5.05	(10 24 35 12) (25 12 3 15)	10	4.63	(220 46 140 231 102 330) (17 0 4 7 2 12)
8	5.75	(1 6 0) (134 165 42)	9	5.56	(10 24 35 12) (25 12 3 15)			
9	5.75	(311 250 122) (6 1 0)	10	5.56	(116 10 27 70) (23 30 31 5)			
10	6.02	(763 227 376) (7 6 0)						

Table 3.4: QPSK → 8-PSK/16-QAM/64-QAM TCM codes.

are actually the best for each category, we also perform a full search for some "small" tuples[1] $(\mathscr{K}, \mathscr{N}, v)$ and compare the results with codes found by RANDOMCODESEARCH. As we conjectured, the full search does not find any code better than RANDOMCODESEARCH. Yet, RANDOMCODESEARCH is extremely fast (cf. Table 3.7). The search results indicates that good codes are distributed randomly in the search space, thus randomized searching is a viable approach, and especially useful for large constraint length and high-order modulations.

Asymptotic coding gain. The search results show that with a large enough constraint length there are codes such that in addition to modulation hiding, the resiliency of the system can be boosted up. For example,

[1] Larger values of $\mathscr{K}, \mathscr{N}$ or v make the *full* search intractable.

8-PSK → 16-QAM		
v	β	G
1	3.11	(1 3 0 1) (0 0 1 0) (1 0 1 0)
2	4.36	(3 3 2 3) (1 0 3 1) (1 0 0 0)
3	5.33	(3 5 1 7) (0 1 3 1) (1 0 1 0)
4	6.12	(5 6 6 4) (4 1 2 3) (1 0 1 0)
5	6.12	(5 4 2 2) (0 3 7 3) (2 0 1 0)
6	6.79	(16 3 6 7) (16 17 13 13) (1 0 1 0)
7	7.37	(23 17 11 17) (2 16 15 10) (1 0 1 0)
8	7.37	(21 20 15 16) (15 5 17 3) (1 0 2 0)
9	7.37	(77 336 10 332) (2 1 5 1) (1 0 1 0)
10	7.37	(15 35 33 5) (13 16 10 15) (16 10 0 15)

8-PSK → 64-QAM		
v	β	G
3	4.15	(5 1 7 5 3 2) (2 3 3 1 2 0) (1 0 0 0 0 0)
4	4.66	(0 14 1 0 15 4) (2 0 0 1 1 0) (1 0 0 1 0 0)
5	5.12	(37 21 13 24 35 4) (3 1 0 1 0 0) (1 0 0 1 0 0)
6	5.53	(71 75 36 70 53 12) (2 2 0 1 2 0) (1 0 0 1 0 0)
7	5.73	(6 47 4 10 27 12) (3 2 0 1 2 0) (2 2 0 3 2 0)
8	5.91	(124 157 6 270 254 172) (3 0 0 0 0 2) (1 0 0 1 0 1)
9	6.26	(26 21 300 303 272 1) (5 3 0 1 2 0) (1 0 0 1 0 0)
10	6.42	(471 475 236 670 653 12) (6 6 0 1 2 0) (1 0 0 1 0 0)

Table 3.5: 8-PSK → 16-QAM/64-QAM TCM codes.

16-QAM → 64-QAM		
v	β	G
2	3.31	(0 0 1 2 3 0) (1 3 1 1 3 2) (1 0 0 1 0 0) (1 0 0 0 0 0)
3	3.31	(1 2 1 1 3 2) (0 3 0 1 0 2) (2 0 1 2 3 2) (0 0 0 1 0 0)
4	4.18	(0 5 6 11 10 1) (1 2 0 0 3 0) (1 0 0 1 0 0) (0 0 0 1 0 0)
5	4.56	(10 3 4 17 24 14) (1 2 1 2 3 0) (0 0 0 1 0 0) (1 0 0 1 0 0)
6	4.91	(63 57 30 12 14 32) (0 2 1 1 3 0) (0 0 0 1 0 0) (1 0 0 0 0 0)
7	5.23	(10 3 4 17 24 14) (11 16 5 12 17 14) (0 0 0 1 0 0) (1 0 0 1 0 0)
8	5.53	(24 0 15 26 23 0) (31 37 15 5 27 26) (1 0 0 1 0 0) (1 0 0 0 0 0)
9	5.81	(73 177 42 123 25 55) (10 7 4 2 12 1) (1 0 0 1 0 0) (1 0 0 0 0 0)
10	5.81	(36 737 302 641 151 317) (4 7 2 6 6 1) (1 0 0 1 0 0) (1 0 0 0 0 0)

Table 3.6: 16-QAM → 64-QAM TCM codes.

$\mathscr{K}$	$\mathscr{N}$	$T_{\text{full}}(s)$	$T_{\text{random}}(s)$
BPSK	QPSK	1	≈ 0
BPSK	8-PSK	4	≈ 0
BPSK	16-QAM	263	15
QPSK	8-PSK	57	44
QPSK	16-QAM	7101	54

Table 3.7: Running time comparison for searching codes of constraint length $v = 5$: random search is significantly faster than full search.

concealing BPSK in QPSK is about 8.5dB more robust than uncoded BPSK systems.

3.5 Cryptographic Interleaving

While encoding the data with the highest-order constellation can hide the modulation, information about the trellis codes still leaks. A powerful adversary can attempt to decode the received stream of samples with all possible codes (which are publicly known) in parallel. The attacker can then infer the rate from the code that has the highest likelihood (i.e., lowest errors). It is necessary to prevent the adversary from being able to try the decoding. Our key idea is to randomize the transitions between the coded symbols. However, we note that naive randomization does not work. For instance, simply applying conventional encryption algorithms on the coded symbols (post-GTCM) will significantly reduce the system performance, as an error occurring during the transmission can spread out to many errors after decryption (by definition of a good encryption algorithm), which would exceed the code's error correcting capability. We therefore propose "Cryptographic Interleaving" as a solution for the code concealing problem. In our approach, we only permute the coded symbols while preserving the symbol values. This ensures that any error occurring in the received symbol is known to be caused by the signal propagation in the noisy environment, but not due to the side effect of decrypting and spreading out the error symbol. We note that our secure permutation problem is a special case of a broader class Format Preserving Encryption (FPE), in which symbols are encrypted into the same domain as the original plaintext's [8, 96]. In our special case, we require to preserve the symbol values. There have been several known methods to randomize the sequence of symbols without changing individual symbol values. First, one can consider the classical Fisher-Yates method, where symbols from a set are randomly picked out based on a seed to create a new ordered sequence. However, this requires initializing the randomization oracle with a new seed for every block of coded symbols. Alternatively, another FPE approach called prefix cipher can be used, where symbol indices are encrypted and then ranked to produce a new ordered sequence [12]. Nevertheless, this provably-secure solution requires more overhead with m operations required per m-symbol block plus $m \log m$ comparisons for ranking. In the following, we will describe our solution, and as we can see later, it is designed specifically to permuting the coded symbol sequence with only 2 encryptions required per block.

Crypto-Interleaving Process: In order to prevent an adversary from distinguishing between sequences encoded by different codes, we require the following: (1) the transmitted symbols should be indistinguishable from a sequence produced by a random code, (2) symbols belonging to different packets are permuted differently, and (3) the user identity is not revealed. We assume a shared key between the transmitter and receiver, which will be used as the seed to generate a secret random interleaving function.

For convenience, we assume that the coded symbols produced by the GTCM Encoder can be divided into multiple blocks, each has m symbols, and each frame contains b blocks. A block is identified by the tuple (K, s, i), where K is the shared secret belonging to the communication session, s is the frame number, and i denotes the block index within the frame.

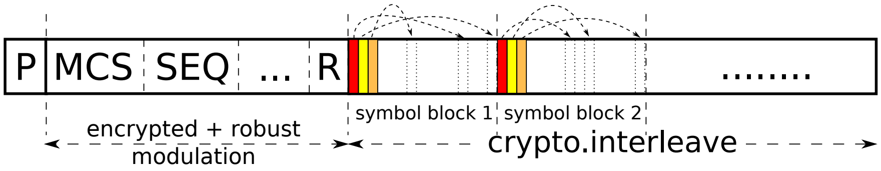
Let's consider a pool of N generated interleaving functions $\mathscr{P} = \{f_0 \ldots, f_{N-1}\}$, where N is the security parameter, and each interleaving function $f_n : I \to I$, $I = \{0, \ldots, m-1\}$, $n \in \{0, \ldots N-1\}$ is an invertible index-mapping of symbols within a block. The permutation of symbols in a block (K, s, i) is performed in two steps. First, an interleaving function $f = f_n$ is selected from the pool $\mathscr{P}$ by $n = h_K(s|i) \bmod N$, where h is a key-hashed pseudorandom function (e.g., truncated HMAC-SHA3). Then, the symbol sequence $(y_0, \ldots, y_{m-1})$ is permuted to $(y_{f^{-1}(0)}, \ldots, y_{f^{-1}(m-1)})$.

Generating Crypto-Interleaving Functions: The efficiency of the crypto-interleaving process depends on the computation of the permutation. While a naive solution can precompute the pool of index-mapping, it does not scale with the block size. Moreover, significant memory is required for the precomputation. In our CBM system, we propose an efficient method for generating the interleaving functions. Our technique assumes that the number of symbols per block, m, is a prime. We note that while this constraint can be easily overcome to support various frame size, e.g., by padding, it has the side effect of preventing length-based attacks, which exploit the observations of different frame size to infer the code information. We define the following linear interleaving functions: $f_{A,B}(x) = Ax + B \bmod m$, where $A \in \{1..m-1\}, B \in \{0..m-1\}$. It is easy to see that any pair (A,B) corresponds to a bijective function $I_{A,B}(x)$ with respect to x; therefore, $f_{A,B}(x)$ is a proper interleaving function (i.e., invertible). The interleaving process is carried out by first generating the coefficients A, B based on the block identifier (K,s,i) by

$$A = (h_K(|s|i|0) \bmod (m-1)) + 1, \quad B = h_K(s|i|1) \bmod m.$$

The interleaving function f is selected as $f = f_{A,B}$. The symbol sequence $\{y_0, \ldots, y_{m-1}\}$ is accordingly permuted to $\{y_{f^{-1}(0)}, \ldots, y_{f^{-1}(m-1)}\}$.

Header Format and Encoding: Since the interleaving processing on the coded symbols of the user data involves using not only the secret key K, but also the packet number s and block index i, the transmitter needs to embed this information along with the rate information into the transmitted frame.



At the packet beginning, the preamble P assists with the frame detection and synchronization at the receiver. The MCS field identifies the modulation and coding scheme for the payload. The packet number required for cryptographic interleaving is specified by SEQ. The R field stores a random number generated per packet by the transmitter. The frame header is encrypted by $E_K(MCS|SEQ|\ldots|R)$ using AES encryption E with the shared secret key K. The header is encoded by a public robust coding scheme and along with the preamble is modulated by a public robust modulation.

Security: Since the interleaving functions are generated using a key-hashed pseudorandom function h with secret key K applied on the packet number s and block index i, the coefficients A and B are indistinguishable [7], thus the interleaved symbol sequences are also indistinguishable.The header is also semantically secure due to the use of random R with AES encryption.

3.6 Synchronization Mechanisms

As we will show in Section 3.7, the simulation results exhibit a significant gain in system robustness while the rate information is concealed from the adversary. However, when evaluating our CBM scheme over a real radio system with a standard set of synchronization mechanisms, we experience a substantial performance loss. Chief among the mechanisms impacted by the imperfections of the radio front end are the frequency and phase correction. At low SNR, such imperfections impact high order modulation much more severely than low order modulation. As in many other communication systems, synchronization with transmitter is required at the receiver in order to decode the data. Traditional synchronization techniques for symbol timing, frequency and phase recovery are realized by means of phase locked loop (PLL) circuits [49, 88]. However, in the context of our work, where the channel SNR can be very low due to adversarial interference, conventional methods perform poorly, making synchronization mechanisms bottleneck of the system. To make our CBM system robust, we developed a set of new efficient digital signal processing algorithms for coarse and fine frequency and phase offset correction relying on an integrated process of estimations based on preambles and iterative soft decoding. Our algorithms also exploit the fact that today's radio receivers can have ample memory to store a whole packet, and many standards already require multi-pass iterative soft-decoding, e.g., LDPC decoding in DVB-S2, IEEE 802.11n/ac. In Section 3.7, we show that our techniques achieve significantly better performance than traditional solutions. It is also worth mentioning that the impact of asynchronization attacks, which focus jamming energy on the beginning of each frame to destroy the synchronization information, is also alleviated in our system by using a long preamble with good correlation properties (cf., Section 3.6.2).

3.6.1 Overview of Transmitter and Receiver

We designed and implemented our transmitter and receiver on the USRP N210 SDR [30] using GNU Radio [16]. The transmit and receive chains of our CBM system are depicted in Figure 3.5.

Transmitter: The key components of the transmitter are the GTCM Encoder and Cryptographic Interleaving blocks. On transmission of a packet, the binary payload is encoded with an appropriate trellis code and target modulation (Tables 3.2 to 3.6) by the GTCM Encoder to produce coded complex symbols. The sequence of symbols is then permuted using the Cryptographic Interleaving block according to the shared key between the two parties. The packet modulating process is completed by prepending the payload with the header and preamble. Finally, it is resampled by the root-raised cosine TX filter before transmission by the RF front end.

Receiver: As in many communication systems, our receiver first obtains the sequence of symbols from

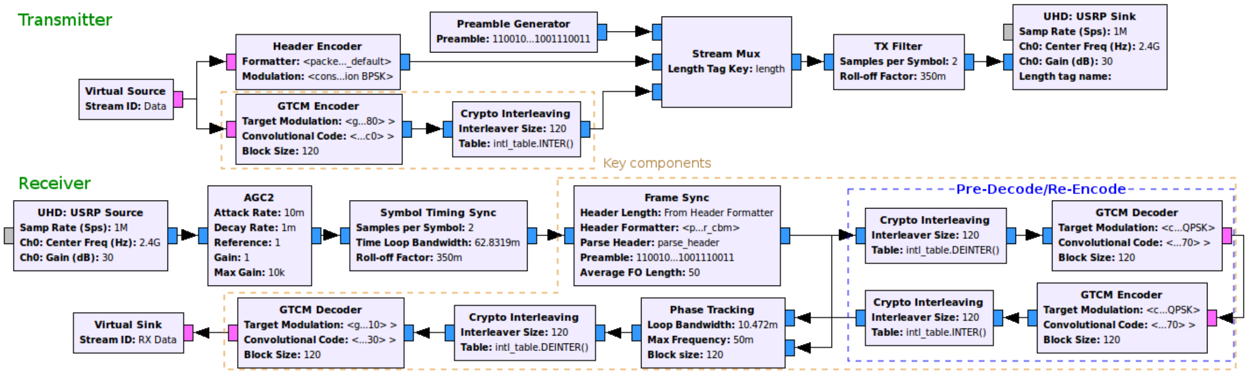


Figure 3.5: Transmitter and Receiver block diagram

the RF front end after they are preprocessed with the Automatic Gain Control (AGC) and symbol timing synchronizer to stabilize the attenuated input signal and lock to the receiver sampling clock.

The key to improving the robustness of our CBM system to overcome the RF front end imperfections consists of improving the accuracy of frequency offset and phase correction. Our mechanisms start with the Frame Synchronizer. First, we estimate the coarse frequency offset based on the preamble of each received packet. The header and payload symbols are then corrected with the estimates. To improve the estimation accuracy, we employ a phase locked loop (PLL) combined with a soft pre-decoding of the packet. This integration results into a 2-pass decoding process. The feedback from the soft pre-decoding re-encoding loop is applied to the PLL to improve the correction. Finally, the corrected sequence passes through the Cryptographic Deinterleaver and GTCM Decoder to recover the original data. In the following subsections, we discuss in details our efficient algorithms of those mechanisms.

3.6.2 Frame Synchronization

The Frame Synchronizer uses the preamble for both frame detection and frequency offset estimation. The principle of our synchronization mechanisms is to analyze the phase-difference sequence of the received symbols. While our technique for frequency offset estimation partly shares similarity with previous work [57], we improve the estimation by averaging over multiple packets. More importantly, we can identify the frame and estimate the phase offset at the same time, and fine-tune the estimation with feedback from the soft decoding process. We consider the received symbol $y_i = x_i e^{j(\theta i+\phi)} + w_i$ at discrete sampling time i, after the signal has been normalized by the AGC, and the symbol sampling period has been synchronized with the receiver clock. The transmitted symbol x_i is distorted by the unknown frequency offset θ, phase offset ϕ due to the mismatched clock between the transmitter and receiver, and interference w_i. The unknown parameters

θ and ϕ are considered as constants during a short period of preamble, and zero-mean variables during the packet transmission period.

Frame Detection Let $\{p_1,\ldots,p_L\}$ denote the complex valued preamble, and $\Delta p_i = p_{i+1}p_i^*$ be the phase-difference indicator between two adjacent preamble symbols. The transmitted stream consists of multiple frames: $\{x_i\} = \{p_1,\ldots,p_L,d_{1,1},\ldots,d_{1,m},\ldots,p_1,\ldots,p_L,d_{s,1},\ldots,d_{s,m},\ldots\}$, where $d_{s,j}$ denotes the s-th frame's j-th data symbol. To detect the start of each transmitted frame in the received signal, we first compute the phase-difference sequence of received symbols $\Delta y_i = y_{i+1}y_i^* = x_{i+1}x_i^* e^{j\theta} + z_i$, where z_i represents the phase-difference with interference. We then match $\{\Delta y_i\}$ with $\{\Delta p_i\}$, i.e., computing the cross-correlation

$$C = \sum_{i=1}^{L-1} \Delta y_i \Delta p_i^* = \sum_{i=1}^{L-1} x_{i+1}x_i^* \Delta p_i^* e^{j\theta} + \sum_{i=1}^{L-1} z_i \Delta p_i^*.$$

We select the preamble as a maximum length sequence such that $\sum p_i x_i^* = L$ only if $\forall i, x_i = p_i$. When the preamble is present in the received signal, i.e., $x_i = p_i$, we obtain $C \approx (\sum_i^{L-1} |\Delta p_i|^2)e^{j\theta}$. When it is not present, C becomes small due to uncorrelation between $\{p_i\}$ and $\{y_i\}$. Our experimental evaluation suggests that the frame be best detected, if

$$|C| \geq \alpha \sum_{i=1}^{L-1} |\Delta p_i|^2$$

with $\alpha = 0.8$.

Frequency Offset Estimation Once the frame is detected, the frequency offset is immediately estimated from the cross-correlation:

$$\tilde{\theta} = \angle C \approx \theta.$$

Under low SNR conditions, $\tilde{\theta}$ may not closely approximate the actual θ. Instead, we compute the average $\hat{\theta} = E[\tilde{\theta}]$ over multiple packets, and use $\hat{\theta}$ for succeeding processing.

Phase Offset Estimation Knowing the frequency offset $\hat{\theta}$, we estimate the phase offset by correcting the received preamble y_i with $\hat{\theta}$: $\hat{y}_i = y_i e^{-j\hat{\theta}i}$, and computing the cross-correlation with the expected preamble p_i as:

$$A = \sum_{i=1}^{L} \hat{y}_i p_i^* \approx \sum_{i=1}^{L} p_i e^{j(\theta-\hat{\theta})i+\phi} p_i^* \approx \sum_{i=1}^{L} |p_i|^2 e^{j\phi}.$$

The phase offset $\hat{\phi}$ is estimated as $\hat{\phi} = \angle A \approx \phi$.

3.6.3 Phase Tracking

While preamble based frequency and phase offset estimation can provide accurate estimates at the frame beginning, the environment variations make these initial estimates deviated from the actual values over time, especially when reaching the end of the frame. To over come this issue, standard phase tracking mechanisms usually deploy decision-aided methods, which rely on the distance between the received symbols and the reference constellation points. BPSK modulation has less symbol errors and therefore the phase tracking works quite well. For higher order modulations, the density of the constellation makes the estimation of the phase errors harder at low SNR, as incorrect constellation points are guessed more frequently. These errors accumulate and end up exceeding the error correction capability of the GTCM code. This partially explains for low SNR scenarios why many current standards still rely on low order modulations instead of coded high order modulations. In pilot-aided synchronization systems such as OFDM, low SNR can result in loss of orthogonality of the carriers, preventing the receiver to recover the signal's phase and amplitude.

To improve the resiliency of our system, we first add a PLL logic in the Frame Synchronizer to keep track of minor changes of the frequency and phase. The loop bandwidth of the PLL is set to a reasonably small value in order to be sufficient to track the small variations, but not to be disturbed by interference.

The key technique to improve the estimation accuracy, consists of a pre-decoding and re-encoding mechanism, in which we first decode the phase-corrected symbols (by running them one first time through the Cryptographic Deinterleaver and GTCM Decoder), then re-encode the resulted binary data back to complex symbols. Now by comparing the re-encoded symbols (which have few symbol errors) and the previously received signal, we can rerun the phase tracking mechanism overcoming synchronization mistakes of the first pass. During the second-pass phase tracking, we carefully skip all the symbols for which there is a discrepancy between what was received and what was decoded. We observe that this significantly reduced the phase tracking errors. This frequency/phase corrected sequence of symbols is then sent to the Cryptographic Deinterleaver and GTCM Decoder for the final decoding.

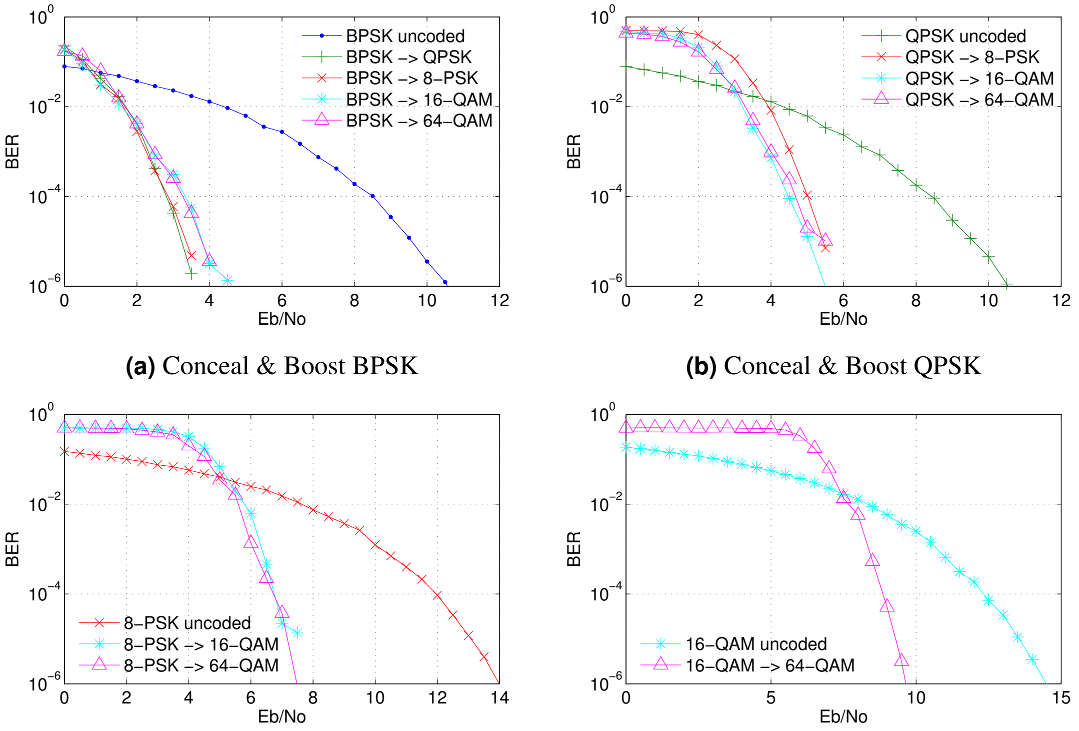
3.7 Evaluation

We report on the evaluation of our CBM system with both simulation results in MatLab and experimental results in our USRP N210 SDR testbed. We use the bit error rate (BER) and normalized signal-to-noise ratio E_b/N_0 as metrics for the system robustness.

3.7.1 Simulation Results

First, we assess the robustness of our codes found in Section 3.4 by simulation in MatLab to avoid impact of imperfect RF front ends such as frequency offset or phase noise. We simulate the transmission of packets encoded by the best TCM codes (constraint length $v = 10$) for each pair of original modulation $\mathscr{K}$ and target modulation $\mathscr{N}$. The simulated noise is additive white Gaussian.

At BER $= 10^{-6}$, Figure 3.6a shows that, in addition to hiding the rate, our codes provide up to 7dB



(a) Conceal & Boost BPSK

(b) Conceal & Boost QPSK

(c) Conceal & Boost 8-PSK

(d) Conceal & Boost 16-QAM

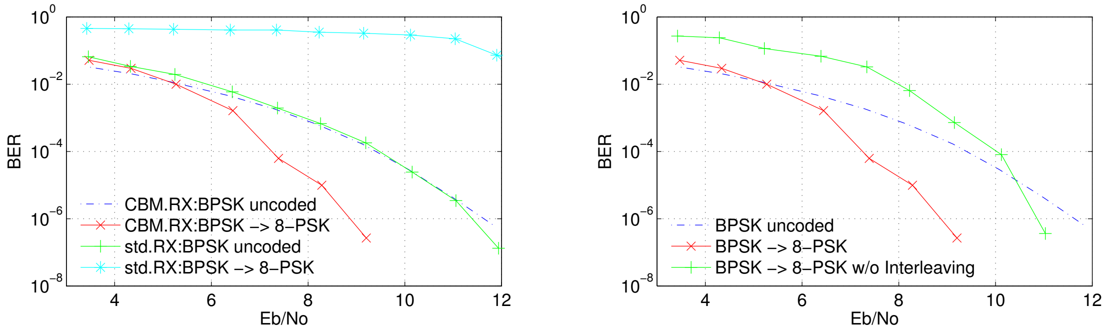
Figure 3.6: Simulation results: Resiliency Boost of Coded over Uncoded Modulations

gain over uncoded BPSK modulation when concealing it into any higher-order modulation. In Figures 3.6b to 3.6d, we also achieve significant improvements when concealing QPSK, 8-PSK, and 16-QAM to higher-order modulations.

The results also show that when 64-QAM modulation is used for rate concealing, we obtain at least 5dB resiliency over any original uncoded modulation. Compared to the modulation level encryption technique proposed in recent related work [90] whose performance degrades by about 1.2dB for hiding BPSK modulation in 64-QAM modulation, we gain up to 6.2dB.

3.7.2 Experimental Results

As noted by previous work [113], the frequency offset and phase noise, due to channel variations and clock mismatch between the transmitter and receiver, can severely reduce the decodability of TCM codes. To assess our system in realistic conditions, we implemented our solution in our USRP N210 SDR testbed to evaluate our system performance. The setup consists of one transmitter, one receiver, and one jammer. The



(a) Synchronization: CBM Receiver vs. Standard Receiver.

(b) Robustness with and without Cryptographic Interleaving.

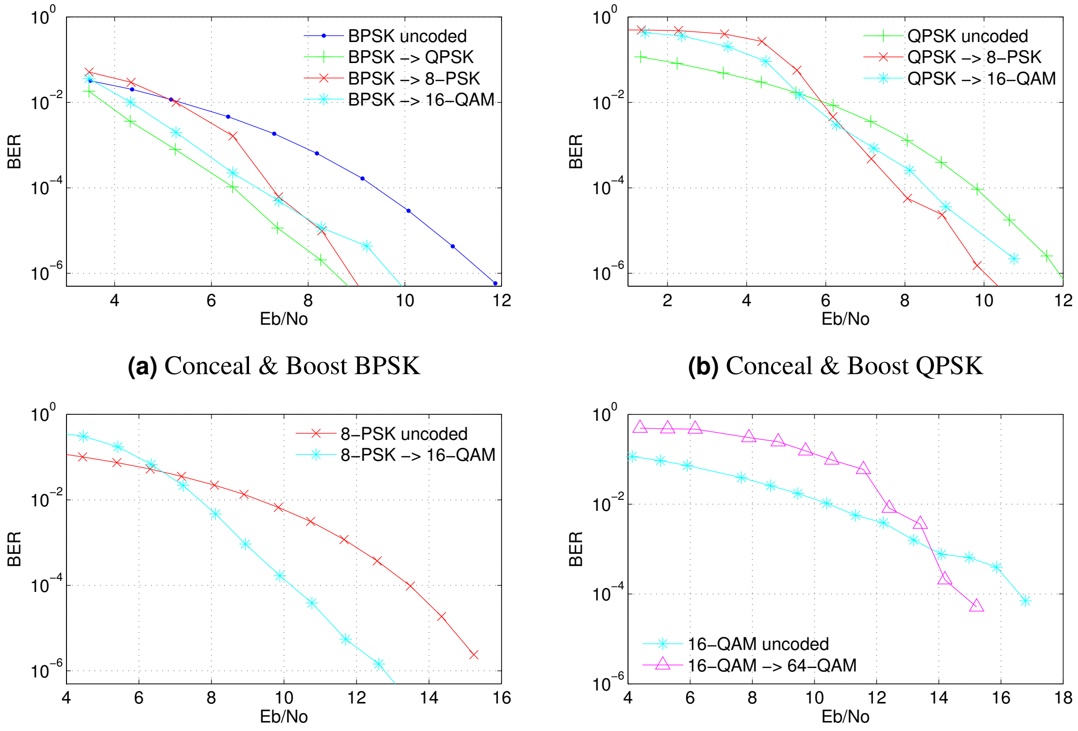
Figure 3.7: Performance of CBM's Synchronization and Cryptographic Interleaving mechanisms.

experiments were both carried through an RF cable and a combiner adding the TX signal to the jammer signal (for a precise control of the SNR), and over the air as a second check.

Impact of Synchronization Mechanisms As discussed in Section 3.6, the low SNR conditions prevent traditional synchronization techniques from performing well. Figure 3.7a shows the performance comparison between two systems: (1) one uses the standard synchronization techniques with band-edge filter combined with phase locked loop circuits [48, 49], (2) CBM receiver with our synchronization mechanisms. We observe that the coded modulation with standard synchronization techniques performs worse than the uncoded modulation due to low SNR. With our 2-pass synchronization, the accuracy of frequency and phase offset correction is much improved, resulting in the resiliency boost of the system.

Effect of Interleaving While the main goal of Cryptographic Interleaving is to conceal the underlying modulations and codes from the adversary, it has a side effect of increasing the system robustness by scattering the burst errors over the whole block. To show the contributions of Cryptographic Interleaving in terms of resiliency boost, we compare the performance of our CBM system with and without interleaving. Figure 3.7b shows that interleaving increases the performance gain by roughly 2dB. We note that without our synchronization mechanism, the interleaving does not help.

CBM vs. Uncoded Now we evaluate our CBM system in comparison with the uncoded system. We conduct the experiments for concealing any modulation $\mathscr{K}$ under any higher-order modulation $\mathscr{N}$ from the set of 5 different supported modulations. The results are shown in Figures 3.8a to 3.8d. First, we observe that in the real world environments, our system resiliency drops roughly $4-6$dB due to the unpleasant frequency and phase offset. Yet, comparing to the uncoded system, we obtain up to 4dB gain in resiliency



(a) Conceal & Boost BPSK

(b) Conceal & Boost QPSK

(c) Conceal & Boost 8-PSK

(d) Conceal & Boost 16-QAM

Figure 3.8: Experimental results: Resiliency Boost of Coded over Uncoded Modulations

while simultaneously concealing the rate information.

3.8 Related Work

Characterizing the vulnerabilities of Rate Adaptation algorithms to jamming and developing countermeasures received increasing interest over the last few years from the first observations and mitigations through fixed rate communication during jamming [21, 86], to randomized mitigations [80, 81], to game theoretic randomization strategies [35–39]. However, previous work assumes that either the adversary is not able to selectively jam packets based on their rates [21, 80, 81, 86] (e.g., due to the slow reaction time of the jammer), or because the transmitter is able to effectively hide the rate [37, 39] (without proposing a concrete mechanism for rate-hiding). Unfortunately, over the last couple of years several recent work demonstrated that rate-selective reactive jammers are feasible on Ettus Software Defined Radio platforms, either on the host [74], or on the FPGA [119]. RAA attacks are therefore fairly easy to implement on custom chips. As we will discuss in the next section, a key unresolved challenge in preventing RAA attacks is how to prevent an adversary from guessing the rate (Modulation, Coding information) during the transmission and therefore from selectively interfering with a packet.

A very recent work [90] proposed a modulation level encryption technique to hide the rate of communications. In essence, this technique always transmits with the highest order modulation, but the communicating nodes cryptographically agree on a subset of the constellation points to be used for each symbol. For example, BPSK modulation can be hidden in 16-QAM by only considering eight pairs of points. For every symbol to be transmitted a pair is cryptographically selected by the transmitter and is also known to the receiver through a shared key. The information bit of BPSK determines which element of the pair is sent. Since the eight pairs cover the whole constellation points, the adversary cannot distinguish between a BPSK communication embedded within 16-QAM or a true 16-QAM. While this scheme conceals the rate information, it does so at the cost of degrading the robustness of the communication. First, one can analytically show that 1-2dB are lost because of the constrained selection of the constellation pairs. Several additional dB are lost due to the poor performance of frequency offset correction, and phase tracking techniques in higher order modulations.

3.9 Conclusion and Discussion

We proposed a solution to the problem of hiding the rate of a communication while simultaneously increasing its robustness to interference. To the best of our knowledge, this is the first system that can achieve this goal. Our approach relies on algorithms for discovering new General TCM codes, and a cryptographic interleaving scheme. These algorithms include new efficient techniques to determine the free distance of non-uniform TCM codes. We explicitly derived 85 codes for upgrading any modulation in {BPSK, QPSK, 8-PSK, 16-QAM, 64-QAM} into any higher order modulation. These are the best codes among uniform and non-uniform TCM codes specifically designed for coded modulation and that conceal the underlying rate information. The GTCM codes and Crypto-Interleaving are complemented by new frequency correction and phase tracking techniques. We demonstrate analytically, and through simulations and experimentation, that beyond achieving rate-hiding, an order of magnitude improvement of energy efficiency is achieved in comparison with recent related work. The proposed solution is easily deployable in software defined radios. Our implementation source code is available at [116].

Acknowledgement

Our special thanks to Erik-Oliver Blass for his many useful comments, especially on the Cryptographic Interleaving mechanism. This material is based upon work supported by the National Science Foundation under Grant CNS-1409453.

Chapter 4

SWiFi: Wi-Fi Network Analysis with SDR

In this chapter, we discuss the design and implementation of our SWiFi system – the first open source SDR that can decode packets for all Modulation Coding Schemes reaching 54Mbps. Using this platform, we will later show in Chapter 5 that we can devise an efficient multicarrier jamming strategy that causes huge impacts on Wi-Fi communications. We start our discussion with an overview of the OFDM mechanism in Section 4.1. Our implementation incorporates novel frequency offset and frequency domain equalization techniques to overcome the limitations of the SDR RF Front End. The details of design and implementation of SWiFi Transmitter and Receiver are described in Sections 4.2 and 4.3. In Section 4.4, we rigorously evaluate the performance of the SWiFi Receiver demonstrating a performance closely matching or outperforming several commercial Wi-Fi cards with the latest Wi-Fi chipsets. In Section 4.5, we also demonstrate the potential of SWiFi to support research on Wi-Fi networks and devices characterization spanning the Medium Access Control (e.g., SIFS/CSMA/Backoff) and Link Layer (e.g., rate adaptation). We summarize the related work in Section 4.6 and conclude our work in Section 4.7.

4.1 Overview of OFDM

We start our discussion with a brief overview of the OFDM principle. In later sections, we then describe in detail our GNU Radio implementation of SWiFi-OFDM Transceiver with a focus on the receiver part.

The principle of Orthogonal Frequency Division Multiplexing (OFDM) is to transmit data over multiple

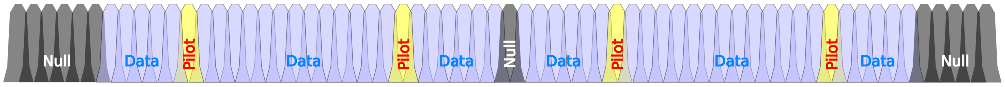


Figure 4.1: IEEE 802.11 OFDM subcarriers mapping.

subcarriers in a way that the subcarriers can overlap, but create no interference to each other. In other words, the wide-band channel is split into multiple *orthogonal* narrowband subcarriers which provide two main advantages:

- *Bandwidth efficiency:* By orthogonality, carriers in OFDM can overlap, resulting in efficient frequency reuse (Figure 4.1), while in non-orthogonal FDM, carriers are separated with significantly large gaps to avoid ICI (Inter-Carrier Interference) issue, leading to bandwidth-inefficiency.
- *Robustness in dynamic environments:* As carriers in OFDM are narrowbands, they are relatively flat (frequency-nonselective), allowing a simpler implementation of the receiver in comparison with single-carrier systems, where frequency-selective channel requires complex equalization techniques to deal with ISI (Inter-Symbol Interference) issue.

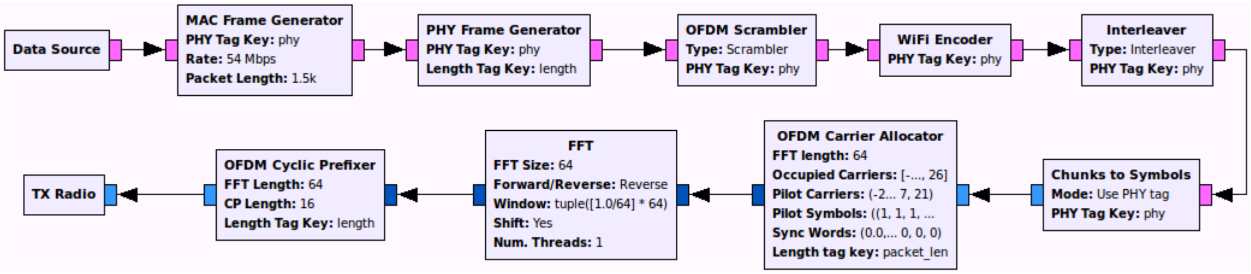
These features render OFDM as the core technique for boosting the performance of today's wireless systems in wideband communications such as 4G LTE/WiMax & IEEE802.11agn. OFDM's success is proven with the dense deployment of Wi-Fi networks today.

In the following, we briefly introduce our notations that will be used throughout the discussion of our estimation techniques which have been verified (Section 4.4) to achieve comparable performance to commercial Wi-Fi cards. Let $X_1, \ldots, X_N$ be the data symbols that will be transmitted at each time period T. We view X_k as values in the frequency domain, where each symbol X_k corresponds to the k-th subcarrier. The chunk $\mathbf{X} = \{X_1, \ldots, X_N\}$ forms an OFDM symbol in the frequency domain. To transmit the data, every X_k symbol is transformed to the time-domain signal $x_n^{(k)} = X_k e^{j2\pi kn/N}$ corresponding to the k-th subcarrier, where n indicates the n-th slot time in an OFDM symbol period. The sum of all subcarriers' signal in the n-th slot is represented by

$$x_n = \sum_{k=1}^{N} x_n^{(k)} = \sum_{k=1}^{N} X_k e^{j2\pi kn/N}. \tag{4.1}$$

The time-domain OFDM symbol, denoted by $\mathbf{x}$, includes N samples: $\mathbf{x} = \{x_1, \ldots, x_N\}$. Since $\sum_{n=1}^{N} x_n^{(k)} (x_n^{(m)})^* = 0$ for $k \neq m$ (where * denotes the complex conjugate), N subcarriers are orthogonal, resulting in no inter-carrier interference. This is in contrast with non-orthogonal FDM system, where inter-channel spacing must be reserved in order to avoid mutual interference. The computation in Equation (4.1) can be efficiently performed by Inverse Fast Fourier Transform (IFFT) technique. At the receiver, the original data symbols are recovered by an FFT operation on the time-domain signal x_n, specifically

$$X_k = \frac{1}{N} \sum_{n=1}^{N} x_n e^{-j2\pi nk/N}.$$



Data Source

MAC Frame Generator
PHY Tag Key: phy
Rate: 54 Mbps
Packet Length: 1.5k

PHY Frame Generator
PHY Tag Key: phy
Length Tag Key: length

OFDM Scrambler
Type: Scrambler
PHY Tag Key: phy

WiFi Encoder
PHY Tag Key: phy

Interleaver
Type: Interleaver
PHY Tag Key: phy

Chunks to Symbols
Mode: Use PHY tag
PHY Tag Key: phy

OFDM Carrier Allocator
FFT length: 64
Occupied Carriers: [-..., 26]
Pilot Carriers: (-2... 7, 21)
Pilot Symbols: ((1, 1, 1, ...
Sync Words: (0.0,... 0, 0, 0)
Length tag key: packet_len

FFT
FFT Size: 64
Forward/Reverse: Reverse
Window: tuple([1.0/64] * 64)
Shift: Yes
Num. Threads: 1

OFDM Cyclic Prefixer
FFT Length: 64
CP Length: 16
Length Tag Key: length

TX Radio

Figure 4.2: SWiFi Transmitter OFDM GNU Radio block diagram

The orthogonality is the key feature of OFDM systems. On one hand, orthogonality allows overlapping subcarriers, therefore increasing the channel efficiency. On the other hand, loss of orthogonality dramatically affects the system performance. To enable practical OFDM receivers, IEEE 802.11ag specifies $N = 64$ subcarriers for 20MHz bandwidth, among which only $N_D = 48$ subcarriers are used for data, while other $N_P = 4$ pilot subcarriers are placed equally in between data subcarriers to assist the receiver's channel estimation, and the rest $N_0 = 12$ subcarriers are left unused (null carriers) for avoiding DC offset and inter-channel interference (See Figure 4.1). During the signal propagation in the environment, blocking and reflecting objects can create multiple copies of the transmitted signal and distort the subcarrier orthogonality. To reduce the multipath effect, a guard interval with cyclic prefix is introduced at the beginning of each OFDM symbol. In particular, the last $N_G = 16$ symbols $x_{N-N_G+1}, \dots, x_N$ of the time-domain OFDM symbol $\mathbf{x}$ are copied and put before $\mathbf{x}$ to result in $N + N_G = 80$ samples $\{x_{N-N_G+1}, \dots, x_N, x_1, \dots, x_N\}$ to be transmitted per OFDM symbol.

The concrete estimation, equalization and decoding techniques for the receiver, however, are left to the specific implementation by the chipset manufacturers. In the following subsections, we describe our specific algorithms with comparison to the state-of-the-art techniques. First, we present the IEEE 802.11 OFDM frame structure and the transmit chain of SWiFi.

4.2 Transmitter Design

We develop the SWiFi-OFDM Transmitter using the built-in GNU Radio blocks of FFT, Carrier Allocator and Cyclic Prefixer along with our own blocks for generating and encoding the OFDM PHY frame, as depicted with GNU Radio Companion – a GUI signal processing flowgraph development environment – in Figure 4.2.

PHY Frame Structure: The time-domain OFDM PHY frame, illustrated in Figure 4.3, consists of OFDM header and payload, which are prepended by a special training sequence of short and long preambles. The

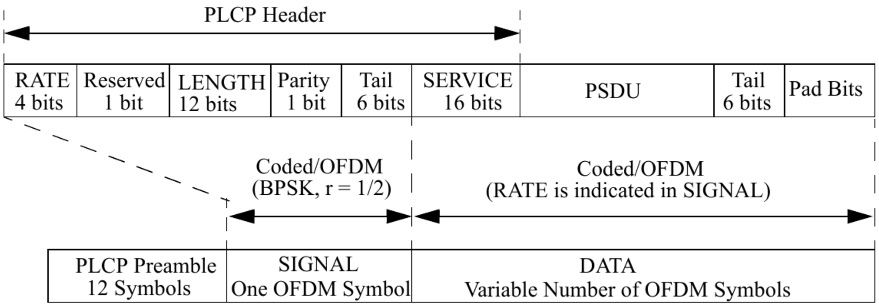


Figure 4.3: OFDM PHY frame format [54].

short preamble is composed of 10 repeated symbols each of $N_{LP} = 16$ samples, while the long preamble contains 2.5 repeated symbols each of $N_{SP} = 64$ samples. The resulting preamble duration is 320 samples, equal to the length of 4 OFDM symbols. The repeated patterns present in both short and long symbols allow the receiver to locate the frames inside the received stream and to perform frequency offset correction (described in Section 4.3.1).

Following the preamble are the SIGNAL field, which specifies the rate and size of the payload. The DATA field contains a 16-bit SERVICE subfield used to synchronize the receiver and transmitter's scrambling seed. The payload, tail and paddings are placed in the rest of the DATA field. Since the encoding process is different on SIGNAL and DATA fields, our PHY Frame Generator employs GNU Radio tags to inform the scrambling, encoding and interleaving blocks of the rate and length corresponding to each part.

Scrambling: For each raw PHY frame, only the DATA field is scrambled by a synchronous (additive) scrambler, while the SIGNAL field is left untouched. The 802.11ag scrambler has 7 registers and requires both transmitter and receiver to synchronize on the same scrambling seed in order to correctly decode the packet. This is assisted by the transmitter by setting the first 7 bits of the SERVICE subfield to all zeros before scrambling. Since wrong detection of the seed results in an undecodable frame, it is important to properly receive the first OFDM symbol in the DATA field. We provide a more detailed discussion in later subsections on channel estimation and equalization.

Encoding: The encoding of the frame header and payload relies on a convolutional code of coding rate 1/2 defined by 802.11ag. While the header is always encoded at coding rate 1/2, the payload coding rate can be increased to 2/3 and 3/4 by puncturing, i.e., periodically omitting several bits in the encoded bit stream. The WiFi Encoder block comprises two sub-blocks for handling convolutional encoding and puncturing tasks.

Interleaving: The interleaving of encoded bits is performed within each OFDM symbol. To minimize the computation cost, we predefine interleaving tables for different modulation and coding rate accordingly to

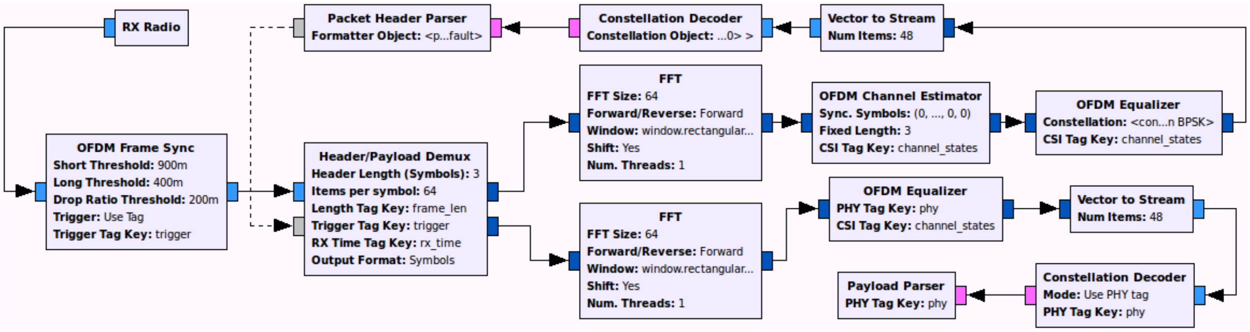


Figure 4.4: SWiFi Receiver OFDM GNU Radio block diagram

the rules by 802.11ag.

Modulation: Each OFDM symbol of interleaved bits is mapped to $N_D = 48$ data subcarriers by the modulation defined in the SIGNAL field. In 802.11ag, the same modulation is used across data subcarriers. Therefore, we employ only one common modulator to map serial bit stream to constellation symbols.

Pilot insertion: At this stage, symbols on data subcarriers are ready to be separated into groups with $N_P = 4$ pilot subcarriers inserted between them. The pilot symbols are defined as BPSK modulated values. The built-in GNU Radio Carrier Allocator block inserts not only the pilot but also null subcarriers into the carrier map to create 64-subcarrier OFDM symbols.

Generating time-domain signal: Finally, each OFDM symbol $\mathbf{X}$ is ready to be transformed into the time domain $\mathbf{x}$ by the use of FFT. The cyclic prefix of N_G symbols are prepended to the beginning of $\mathbf{x}$ by the GNU Radio Cyclic Prefixer block, resulting in $N + N_G = 80$ samples for every transmission of an OFDM symbol.

4.3 Receiver Design

SWiFi-OFDM Receiver block diagram is shown in Figure 4.4, in which the core components are:

- The OFDM Frame Sync block is responsible for frequency offset estimation and frame synchronization. For each recognized frame, it sends the time-domain samples to the Header/Payload Demux, which removes the cyclic prefix on every OFDM symbol and demultiplexes to the header and payload receive chains, where the FFT blocks transform the samples back to the frequency domain for decoding and parsing.

- The OFDM Channel Estimator is only present on the header chain, since it performs an initial channel estimation based on the training sequence defined in the preamble. The initial channel state information (CSI) is crucial for demodulating the subcarriers.

- The OFDM Equalizer establishes the equalization process with initial CSI, then it handles the dynamic channel variations along the reception of the frame.

4.3.1 Frequency Offset Correction

Due to the common clock mismatch between the transmitter and receiver RF front ends, the original signal s_n is rotated at the receiver by an amount of $n\theta + \phi$, where θ represents the frequency offset between the transmitter and receiver, and ϕ denotes the unknown phase offset. In SWiFi, we compensate the frequency offset as a first step before any other signal processing tasks. At the same time of frequency offset estimation, a coarse detection of OFDM frames is also achieved. Our algorithm is based on Schmidt-Cox method [99], which utilizes the repeated pattern in short and long preamble symbols.

Coarse estimation: Let $\mathbf{p} = \{p_1, p_2, \ldots, p_L\}$ be a time-domain short symbol consisted of $L = 16$ samples defined in 802.11ag preamble. The principle of the method is based on the assumption that the frequency offset θ is relatively smaller than $2\pi/L$ and remains constant in the duration of an OFDM symbol, by which a phase difference of $L\theta$ will be observed between two consecutive preamble symbols. The limit $2\pi/L$ will be made clear shortly. At the receiver, we observe the received signal as

$$\{r_n\} = \ldots, \underbrace{\hat{p}_1, \ldots, \hat{p}_L, \hat{p}_{L+1}, \ldots, \hat{p}_{2L}, \ldots}_{\text{preamble}}, \underbrace{\ldots, \hat{x}_k, \ldots}_{\text{data}}$$

where $\hat{p}_k = p_{[k]_L} e^{j(k\theta+\phi)}$, $[k]_L = k \bmod L$, and $\hat{x}_k = x_k e^{j(k\theta+\phi)}$ denote the rotated version of preamble and data samples. We compute the correlation A_n between two consecutive chunks of L samples and the energy E_n of the current chunk:

$$A_n = \sum_{k=0}^{L-1} r_{n+k+L} r^*_{n+k}, \quad E_n = \sum_{k=0}^{L-1} |r_{n+k}|^2$$

The ratio $|A_n/E_n|$ determines whether r_n contains a preamble sample p_k for some k. Specifically, the preamble is found if $|A_n/E_n| \geq \alpha$, where the parameter $\alpha = 0.9$ is used in SWiFi implementation. At the same time of identifying the preamble, we observe that $A_n = \sum_{k=0}^{L-1} |p_{[n+k]_L}|^2 e^{jL\theta}$. Now, due to the assumption $\theta < 2\pi/L$, we have $\arg(A_n) = L\theta$, and obtain the estimate $\hat{\theta} = \frac{1}{L}\arg(A_n)$. Considering the Wi-Fi short preamble symbols with length $L = 16$, the maximum correctable frequency offset by SWiFi is $\pi/8 \approx 0.4$ rad/sample, which is acceptable for today's RF front ends.

We emphasize that in the GNU Radio platform, we implement the mechanism in a single block rather than using multi-threaded multiple blocks as the typical GNU Radio approach. Our solution is to maintain a state machine in order to control the estimation and correction logic more efficiently. In particular, we stop the computation for the estimation once the frame is detected, and resume it when there is an energy drop in the signal indicating the frame end. During the packet processing, small variations of the frequency offset are handled by the equalization which will be described in Section 4.3.2.

Fine estimation: After a coarse estimation based on the short preamble, we repeat the same algorithm on the long preamble to compensate for any residual offset not detected by short symbols. Though more computation is required for working with the long preamble, it is only performed after the threshold for the coarse estimate is exceeded, thus only a slight overhead is introduced.

Frame synchronization: While the algorithm for frequency offset correction can detect the frame, it is ambiguous to locate exactly where the frame is started, as the ratio $|A_n/E_n|$ remains high in the duration of repeated symbols and gradually decreases when the preamble is passed. To precisely reveal the exact location of the first sample of the frame, we correlate the received samples (after frequency offset correction) with the time-domain long preamble to find the peak corresponding to the repeated pattern of the long preamble. This approach gives an accurate synchronization with the OFDM symbols, as also shown in [14].

4.3.2 Frequency Domain Equalization

Frequency offset correction in the time domain is not enough for a successful demodulation of the OFDM symbols, because in a wireless environment, the dynamic channel causes the subcarriers to experience different attenuations and phase offsets. Figure 4.5 shows the frequency-domain OFDM symbol corresponding to the SIGNAL field captured over the air. It can be seen that the subcarriers are distorted in both amplitude and phase, which result in the SIGNAL field's BPSK constellation points being rotated and deviated from their original points. More importantly, even though the frequency offset has been corrected in the time domain by the OFDM Frame Sync block, a small amount of frequency offset can still be observed in the frequency domain, which accumulate and rotate the symbols from their original locations.

While amplitude correction is not required for PSK modulations, it is crucial for demodulating QAM signals/ In the following, we review existing equalization methods for OFDM systems and present our new techniques that allow the SWiFi receiver to perform comparably to commodity Wi-Fi cards. We use similar notations introduced in Section 4.1 for our discussion, i.e., $\mathbf{X}$ denotes the frequency-domain transmitted OFDM symbol, $\mathbf{Y}$ the frequency-domain received OFDM symbol. For each frame, we index the OFDM symbols as follows: the two long preamble symbols are $\mathbf{X}_{-2}, \mathbf{X}_{-1}$, the SIGNAL field is $\mathbf{X}_0$, and the DATA field starts from $\mathbf{X}_k, k = 1, \ldots, n$.

Least Square (LS): The Least Square estimation is based on the long preamble symbols, namely the channel

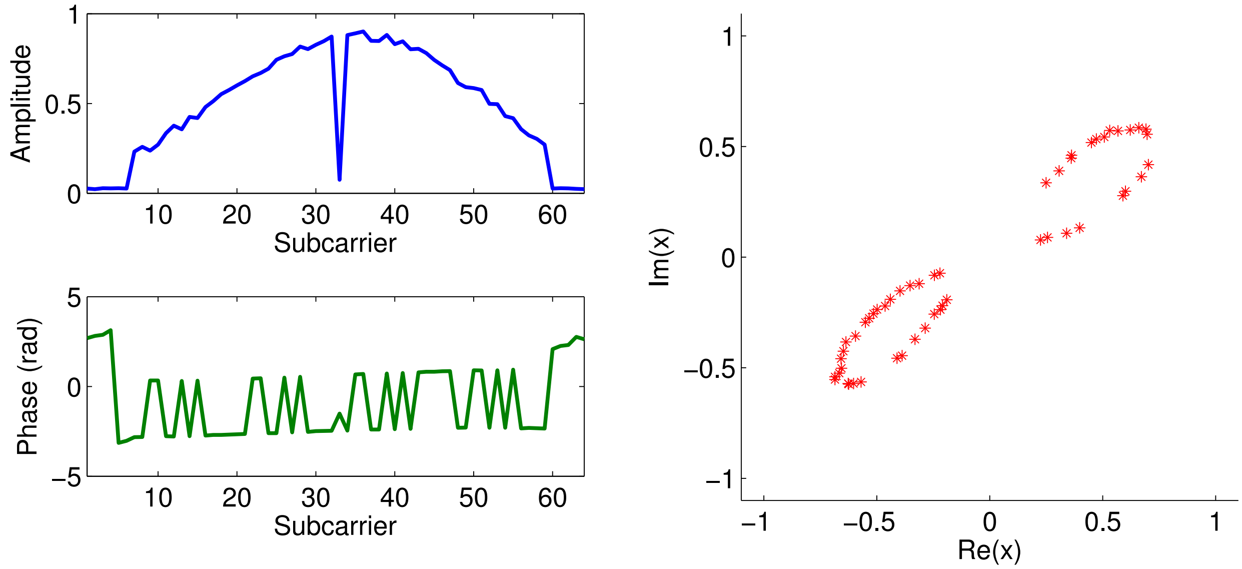


Figure 4.5: Analysis of BPSK-modulated SIGNAL field: 64 subcarriers experience different attenuations and phase offsets (on the left), resulting in the leaves-shape on the constellation (on the right).

is estimated as

$$\mathbf{H} = \frac{1}{2}\left(\frac{\mathbf{Y}_{-2}}{\mathbf{X}_{-2}} + \frac{\mathbf{Y}_{-1}}{\mathbf{X}_{-1}}\right) \tag{4.2}$$

and is used to equalize the rest of all OFDM symbols. Due to frequency and phase offset occurring in the transmission, the LS method cannot keep track of the phase of the symbols over the frame duration. As a result, a rotated constellation can be observed as shown in Figure 4.6.

Linear Regression (LR): In the Linear Regression approach, the pilot symbols on 4 pilot subcarriers are used to infer, by linear regression, the phase offset of the subcarriers based on the observation that there is a constant phase offset from subcarrier to subcarrier within an OFDM symbol (as seen previously in Figure 4.5). To compensate the amplitude, a scaling factor is derived from the average amplitude of pilot subcarriers and it is used to restore the amplitude of data subcarriers. While this approach can adaptively recover the symbols modulated by PSK modulations [14], the QAM signal cannot be decoded. This is explained by the fact that subcarriers are attenuated by different gains (See Figure 4.5), as a result, a common scaling factor does not properly recover the amplitudes for all subcarriers, which are required for amplitude-sensitive modulations like QAM.

Low Pass Interpolation (LPI): Based on the assumption that the received OFDM symbol have a sinc shape (e.g., Figure 4.5), the Low Pass Interpolation method uses the pilot subcarriers to reconstruct the

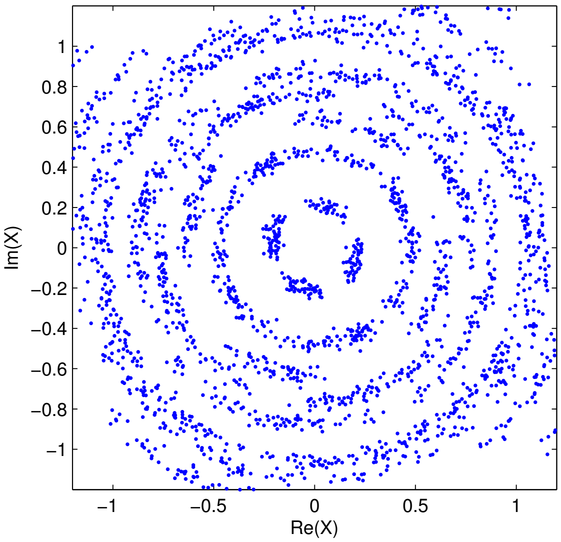


Figure 4.6: Phase tracking by LS Equalization results in rotated constellation.

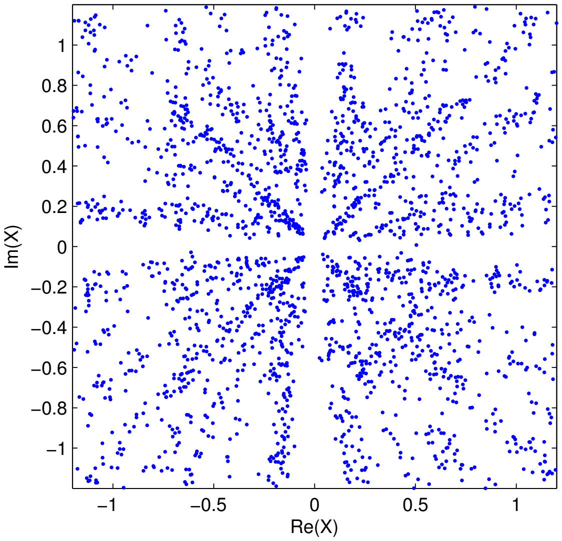


Figure 4.7: QAM symbol amplitudes are not recovered properly by LR Equalization.

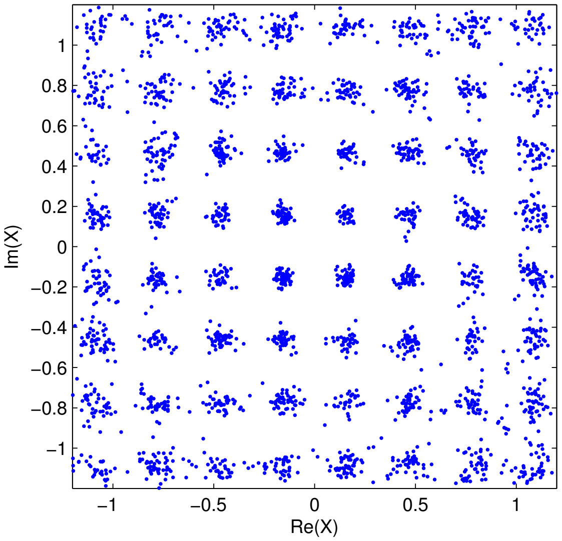


Figure 4.8: Effect of LPI equalization: phase and amplitudes are recover, but a significant amount of incorrect estimation is still present.

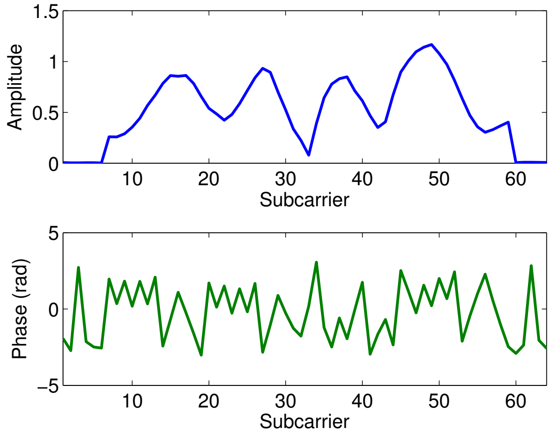


Figure 4.9: OFDM symbol is affected by unpredicted channel distortions .

data subcarriers by applying an ideal low pass filter on the pilot subcarriers. Specifically, let $X_{p_1},\dots,X_{p_4}$ be the known symbols on pilot subcarriers in the current OFDM symbol. The channel state for those pilot subcarriers are derived by:

$$H_{p_k} = \frac{Y_{p_k}}{X_{p_k}}, k = 1,\dots,4 \tag{4.3}$$

Let $\mathbf{H}^{(P)} = \{H_k^{(P)}\}$, where $H_k^{(P)} = H_k$ if $k \in \{p_1,\dots,p_4\}$, and $H_k^{(P)} = 0$ otherwise. Now using a low pass filter $\mathbf{F}$ with predefined coefficients $F_1,\dots,F_N$, the channel states for data subcarriers are estimated by:

$$H = H^{(P)} * \mathbf{F} \tag{4.4}$$

where $*$ denotes the convolution operation. The effect of LPI equalization is shown in Figure 4.8, where subcarrier symbols are reconstructed to their original phase and amplitude. However, there are a considerable amount of symbols not recovered correctly, which can be seen as noise in between constellation points in Figure 4.8. By experimentation, we conclude that this fraction of noise is, unfortunately, higher than the error correction capability of the Wi-Fi convolutional code, resulting in incorrect packet reception. We find that the LPI equalization method fails to reconstruct the signal because the sinc shape assumption does not always hold. Figure 4.9 shows an example of an abnormal shape of an OFDM symbol, where subcarriers are distorted in a unpredictable manner.

Decision Directed (DD): We have seen that the above discussed equalization methods only rely on the pilot symbols or preamble symbols to estimate the channel for data subcarriers in subsequent OFDM symbols. This limits the estimation accuracy because a channel distortion on training and pilot sequences results in a critical impact on the data subcarrier channel recoverability.

The principle of Decision Directed equalization [17, 92] is to use the statistical characteristics of data subcarriers to estimate the channel. In particular, each subcarrier is decoded according to the known modulation of the current OFDM symbol, and the decoded symbol is used as a training symbol to estimate the channel for the next OFDM symbol. It is illustrated by the following steps. First, the received symbol $Y_k^{(n)}$ of the k-th subcarrier at time n is equalized with the previously estimated channel state $H_k^{(n-1)}$ to yield the equalized symbol $\hat{X}_k^{(n)} = \frac{Y_k^{(n)}}{H_k^{(n-1)}}$. Using $\hat{X}_k^{(n)}$, the demodulator finds the closest constellation point $X_k^{(n)}$ and decides it as the decoded symbol. Now with the belief of $X_k^{(n)}$ as the original data symbol, the channel state of the k-th subcarrier for the next OFDM symbol is estimated by

$$H_k^{(n)} = \frac{Y_k^{(n)}}{X_k^{(n)}}. \tag{4.5}$$

The drawback of DD method is that the error occurring at the current OFDM symbol can propagate to subsequent symbols and destroy the whole packet.

Spectral Temporal Averaging (STA): The Spectral Temporal Averaging method [33] extends the Decision Directed equalization by performing averaging of the channel estimates in both frequency and time domain. Namely, after channel estimates $H_k^{(n)}$ are obtained in Equation (4.5), the channel states corresponding to adjacent subcarriers are spectral averaged, i.e., $\hat{H}_k^{(n)} = \sum_{m=-\beta}^{\beta} \gamma_m H_{k+m}$. Finally, a temporal averaging is performed to obtain the channel states for the next symbol: $H_k^{(n)} = \alpha \hat{H}_k^{(n)} + (1-\alpha) H_k^{(n-1)}$. The STA method relies on the assumption of likelihood of adjacent subcarriers and channel states changing slowly over time. However, as we show later, this approach does not improve the equalization quality, and in most of the cases, it performs worse than the DD method.

Our approach During our experimentation, we observe that while the amplitudes of pilot subcarriers can fluctuate significantly due to the channel variations, the pilots' phase is more stable. This motivates us to use pilot subcarriers only for phase tracking. Our idea is that we first derive the phase offset of the current OFDM symbol based on the pilot information. After compensating for the phase offset, the amplitudes of data subcarriers are recovered by applying the principle of Decision Directed method. However, different from the basic Decision Directed equalization, we only update the channel states if the mean squared error of the decoded symbols does not exceed a threshold β. In addition, to avoid the wrong channel estimates of the current OFDM symbol leading to error propagation in subsequent symbols, we update the channel states by a moving averaging over the previous states.

Our detailed solutions contains of the following steps:

- First, based on the long preamble symbols, we compute the initial channel states:

$$\mathbf{H}^{(-1)} = \frac{1}{2}\left(\frac{\mathbf{Y}_{-2}}{\mathbf{X}_{-2}} + \frac{\mathbf{Y}_{-1}}{\mathbf{X}_{-1}}\right). \tag{4.6}$$

  This step is handled by the OFDM Channel Estimator in the receive chain.

- Let $X_{p_k}^{(n)}$ denote the transmitted training symbol in the pilot subcarrier p_k of the n-th OFDM symbol, and ϕ_{n,p_k} be the phase of the received symbol in the corresponding subcarrier. We have $Y_{p_k}^{(n)} = H_{p_k}^{(n)} X_{p_k}^{(n)} = |H_{p_k}^{(n)}| e^{j\phi_{n,p_k}} X_{p_k}^{(n)}$. Using $\mathbf{H}^{(-1)}$ from Equation (4.6), we already obtain $\phi_{-1,p_k} = \angle H_{p_k}^{(-1)}$. The phase ϕ_{n,p_k} is rotated over each OFDM symbol by $\phi_{n,p_k} = \phi_{-1,p_k} + (n+1)\delta$, where δ is the sampling frequency offset created due to the mismatched sampling frequency between the transmitter and the receiver. To estimate ϕ_n, we first compute

$$A = \sum_{k=1}^{4} \frac{Y_{p_k}^{(n)}}{X_{p_k}^{(n)}} \left(H_{p_k}^{(-1)}\right)^*$$

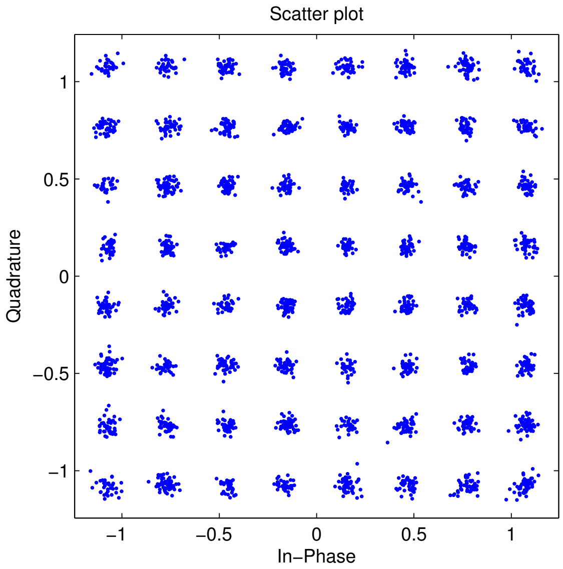


Figure 4.10: The constellation symbols can be clearly seen by our equalization method.

and derive the estimate

$$\varphi = \angle A = \angle \sum_{k=1}^{4} |H_{p_k}^{(n)}| e^{j\phi_{n,p_k}} |H_{p_k}^{(-1)}| e^{-j\phi_{-1,p_k}} = \angle \sum_{k=1}^{4} e^{j(\phi_{n,p_k} - \phi_{-1,p_k})} = (n+1)\delta.$$

- Now using φ, we compensate the phase offset for all data subcarriers k to obtain

$$\hat{Y}_k^{(n)} = Y_k^{(n)} e^{-j\varphi},$$

then use the estimated channel $\mathbf{H}^{(-1)}$ to derive the estimated transmitted symbol

$$\hat{X}_k^{(n)} = \frac{\hat{Y}_k^{(n)}}{H_k^{(-1)}}.$$

- Based on the constellation, we find the closest symbol $\tilde{X}_k$ to $\hat{X}_k$ and evaluate the mean square error (MSE) $\varepsilon = \frac{1}{N}\sum_{k=1}^{N} |\tilde{X}_k - \hat{X}_k|^2$. We update the channel states for the next symbol only if the computed MSE does not exceed a threshold β. The update of channel states is based on the closest symbol $\tilde{X}_k^{(n)}$ and performed by averaging out:

$$H_k^{(n)} = \alpha \frac{\hat{Y}_k^{(n)}}{\tilde{X}_k^{(n)}} + (1-\alpha) H_k^{(n-1)},$$

where $\mathbf{H}^{(n-1)}$ are the previously estimated channel states for the $(n-1)$-th OFDM symbol.

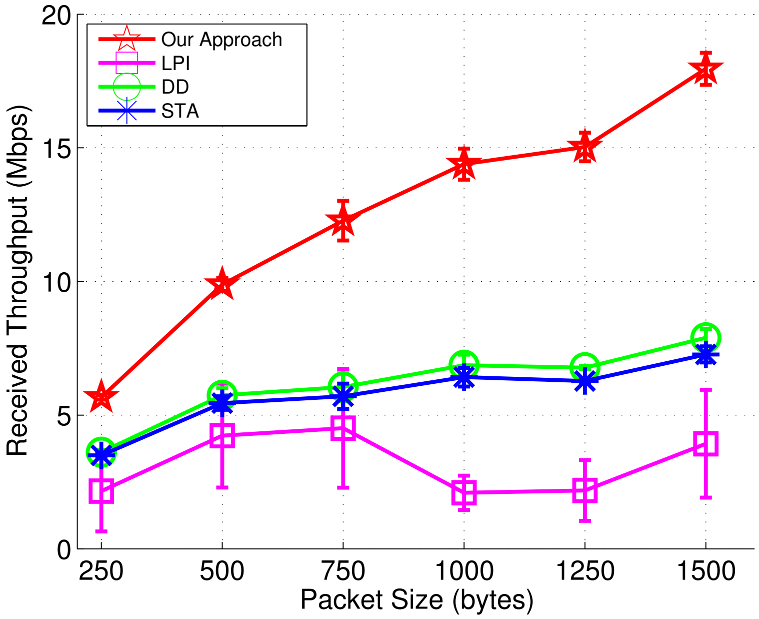


Figure 4.11: Performance comparison of equalization methods.

Our experimental evaluation shows that the performance of our technique in comparison with LS, LR, LPI, DD, STA, we report an improvement of almost 100% in comparison with the second best technique (DD) for any of the considered rates and packet sizes (Figure 4.11). We note that the performance of LS and LR is very low, and therefore is not reported here. From the results, we can observe an increasing improvement of performance as a function of packet size for all equalization methods except for the LPI method. With the LPI equalization, the performance is quite dependent on the dynamic environment, because the channels are estimated and interpolated solely based on the pilot subcarriers. The dynamic environment causes the OFDM symbols belonging to the same packet to experience completely different channel attenuations and phase rotations, resulting in a drop in performance when the packet is longer than an environment-dependent threshold; for instance, increasing packet size from 750 bytes to 1000 bytes reduces the overall received throughput. We note that when the packet size is increased beyond the threshold, the LPI performance is increased slightly due to the larger packet size.

4.4 Evaluation

In order to accurately characterize SWiFi, we methodically compared its performance to several commercial IEEE 802.11abg cards. We focussed on the most challenging aspects namely the performance of the OFDM receiver. We carried out measurements for both typical over the air communications and wired communication with controlled attenuators. The experiments considered the impact of packet size, rate, attenuation, and location.

4.4.1 Components and common methodology

We first summarize the various components used in our measurement experiments. For comparing with commercial Wi-Fi devices, we selected cards from well known and popular manufacturers (D-Link, Linksys, TP-Link) based on the widely used Atheros and Ralink chipsets. We selected cards for which it is possible to connect a single external antenna and which have Linux drivers available. The three selected cards have a PCI interface. Table 4.1 lists the various components, manufacturers, models, and key characteristics.

Table 4.1: Components used for over the air and wired testbeds.

Component	Brand/Model	Misc.
Hosts	Dell XPS 8500	i7 quad-core 3.4GHz, 16GB RAM
Traffic generator	Iperf	Ubuntu 14.04
SDR 1	Ettus USRP N210	SBX 0.4-4.4GHz
SDR 2	HackRF One	10MHz-6GHz
Access Point	TP-Link TL-WDR4300	Atheros AR9344
Wi-Fi USB	TP-Link TL-WN722N	Atheros AR9271 (ath9k_htc)
Wi-Fi NIC 1	TP-Link TL-WN751ND	Atheros AR9227 (ath9k)
Wi-Fi NIC 2	D-Link DWA-525	Ralink RT5360 (rx2x00pci)
Wi-Fi NIC 3	Linksys WMP54G	Ralink RT2500 (rt2x00pci)
Antenna	Antenova Titanis Swivel	2.2dBi (peak)
Splitter	L-com Hypergain	
Attenuators	Mini-Circuits 1-30dB	
RF Cables	L-com SMA	

In both the over the air and wired experiments, the devices under comparison operate as sniffers of the traffic between a Laptop transmitter and the Access Point (See Figures 4.12 and 4.16). The traffic is generated using the Iperf throughput measurement tool from the laptop to a PC connected to the Access Point. To obtain meaningful and fair results, all the devices under comparison are connected to a splitter to obtain a copy of the same RF signal.

Each experiment is run for 10 seconds and repeated 5 times. For each experiment we filter all the packets that are received without errors (correct CRC) and derive the throughput as seen by each device. We run experiments for each of the following IEEE802.11ag rates 18 Mbps (QPSK), 36 Mbps (16-QAM), and 54 Mbps (64-QAM), all with 3/4 convolutional code. For each experiment, we fixed the rate of the transmitter (driver) and verified that all received and counted packets are at the specified rate. We focused on rates with high order modulations because they are the most challenging. We considered both packets of size 1000 Bytes and 1500 Bytes. Based on the number of correctly received packets we computed the net throughput (IP and above) as seen by each Wi-Fi card/SDR. Note that this throughput is typically smaller than the raw physical layer rate since it does not include the IEEE802.11 overhead (preamble, header, ACK, SIFS, DIFS, Backoff) and concurrent traffic (for over the air experiments). All experiments are over the 2.4GHz band. We plan to comprehensively evaluate SWiFi over the 5GHz band once we setup a testbed with a USRP CBX daughterboard that can reach 6GHz, splitters (for 5GHz), and 802.11a transmitters with external antennas.

Our comprehensive evaluation focused on the USRP N210. Our early preliminary evaluation of the HackRF indicates that it achieves a slighly lower performance than the USRP N210 despite a significantly lower quality 8 bits ADC instead of a 14 bits for the N210. Note that 8 bit ADCs are typical for commercial Wi-Fi cards. The main obstacle to carry a comprehensive evaluation of the HackRF is that it intermittently overflows the USB link when using a sampling rate of 20 Msps. This is because the HackRF transfers 8 byte per sample (a 4 bytes float for I and respectively 4 bytes for Q) resulting in 1.28Gbps bandwidth requirements which far exceeds the 480 Mbps theoretical limit of USB. However, this is not a fundamental problem as a reprogramming of the HackRF CPLD/microcontroller (ARM LPC4330 Cortex M4/M0) can reduce the bandwidth requirements by a factor of 4, sending the ADC values as 2 bytes instead of 8. The preliminary values we obtained for windows of samples that did not experience overflows indicates a slight but not substantial performance degradation in comparison with the USRP N210.

4.4.2 Performance over the air

We started with a set of measurements over the air. Our goal was to confirm that, in a typical environment, SWiFi has a similar performance as commercial cards. Given that the RF environment is highly sensitive to location and time, the RF signal is sniffed by a single antenna and connected to a splitter that feeds the USRP and other cards under comparison.

We considered four locations for the transmitter, while keeping the receivers fixed (Figure 4.12). The transmiter locations were selected to create different types of link, from short distance un-obstructed to the most challenging ones with no line-of-sight. We evaluated the performance with packets of 1000 bytes and 1500 bytes. We note that most of the time SWiFi slightly outperforms the commercial cards. One interesting observation is that at low SNR, the performance of commercial cards is very unstable in comparison with

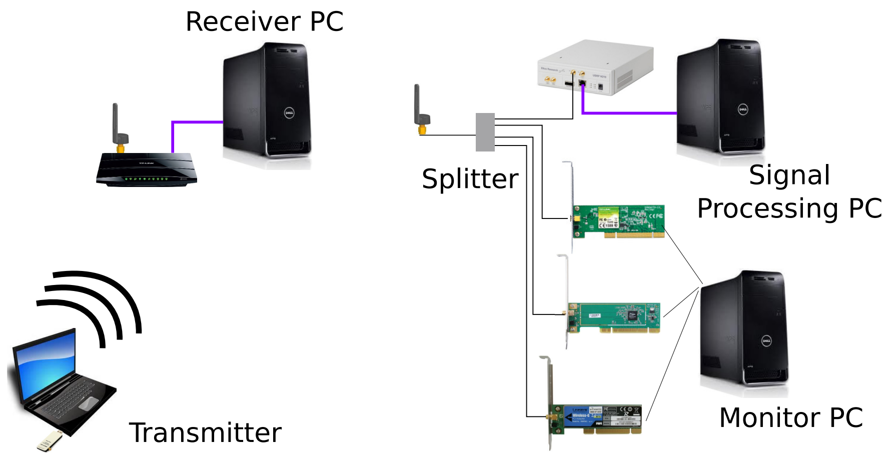


Figure 4.12: Over the air testbed setup.

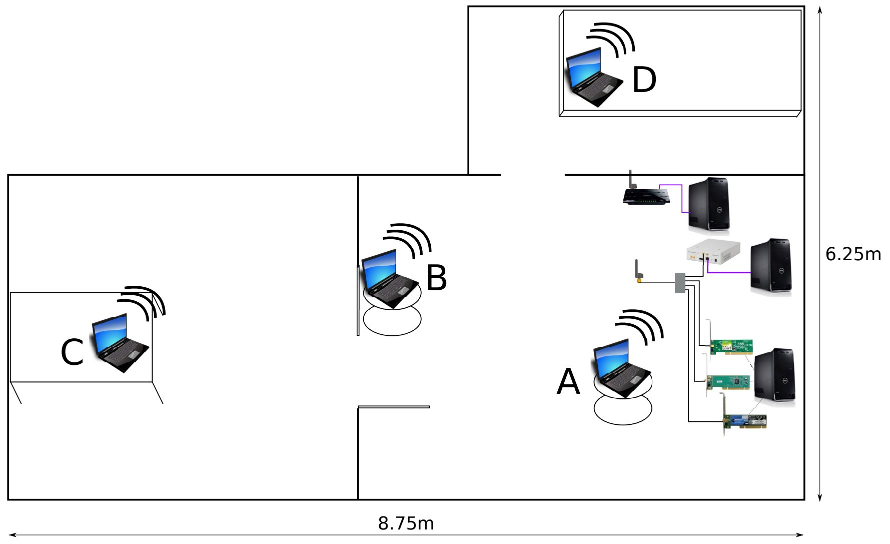
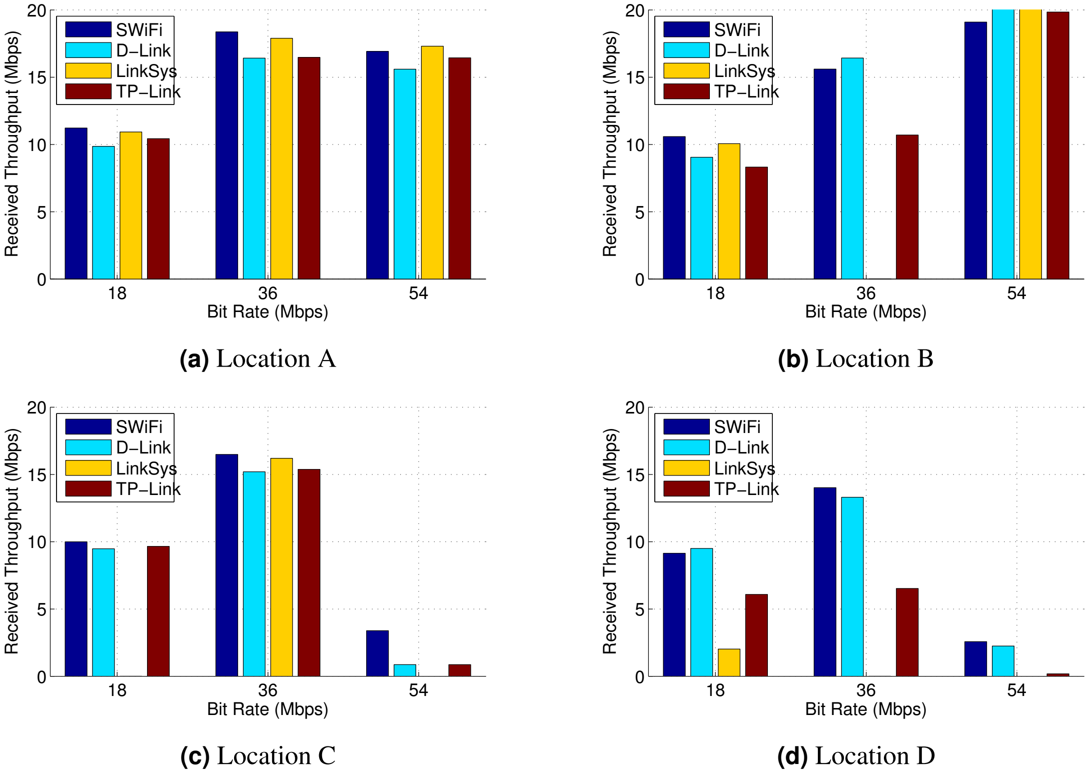


Figure 4.13: Locations of transmitters/receivers for over the air experiment.



(a) Location A

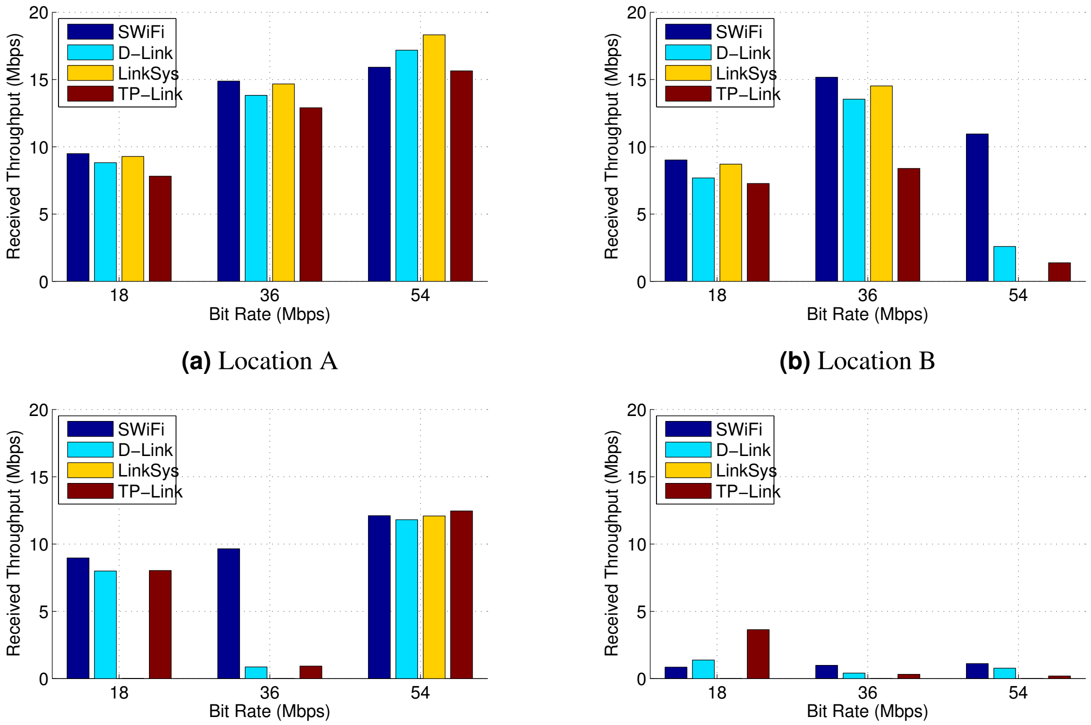
(b) Location B

(c) Location C

(d) Location D

Figure 4.14: Throughput comparison between SWiFi and commercial Wi-Fi cards in wireless setup with 1500-byte packet transmission.

SWiFi. In particular the Linksys WMP54G card was the most unstable which might be due to its relatively older chipset Ralink RT2500. It has the best performance of the three cards in good channel conditions, and the worst in harsh channel conditions. The D-Link DWA-525 with Ralink RT5360 chipset performed the closest to SWiFi except in one configuration where SWiFi significantly outperformed all the commercial cards. As is well knows over the air performance evaluation is very sensitive to time, location, and instantaneous interference. It is therfore possible to compare the receivers to each other, as they experience the same environment, but hard to predict performance of individual cards. An illustration of this is that in our experiments 1500 bytes packets seem to result in better throughput. This can only be partially explained by the lower overhead (i.e., headers, DIFS, backoff, ACK), an other explanation is that the environment was different. This also motivate our evaluation and comparison in the controlled attenuation/propagation setup.



(a) Location A

(b) Location B

(c) Location C

(d) Location D

Figure 4.15: Throughput comparison between SWiFi and commercial Wi-Fi cards in wireless setup with 1000-byte packet transmission.

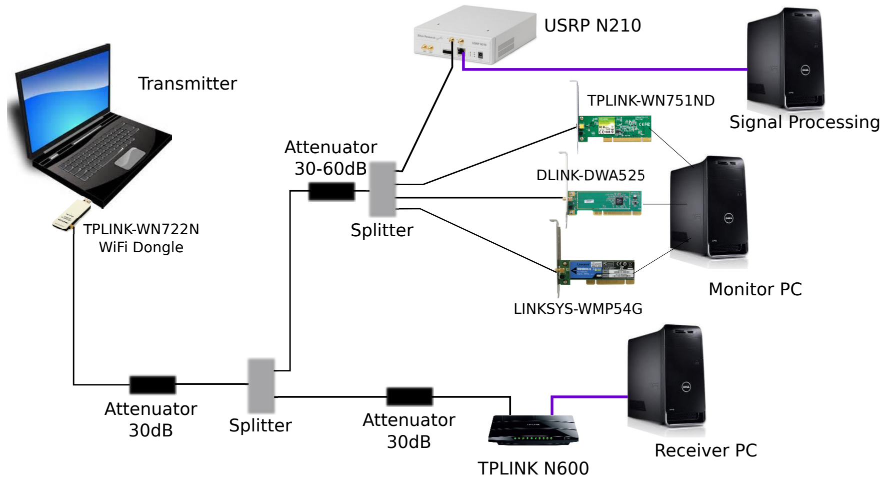


Figure 4.16: Controlled attenuation testbed setup.

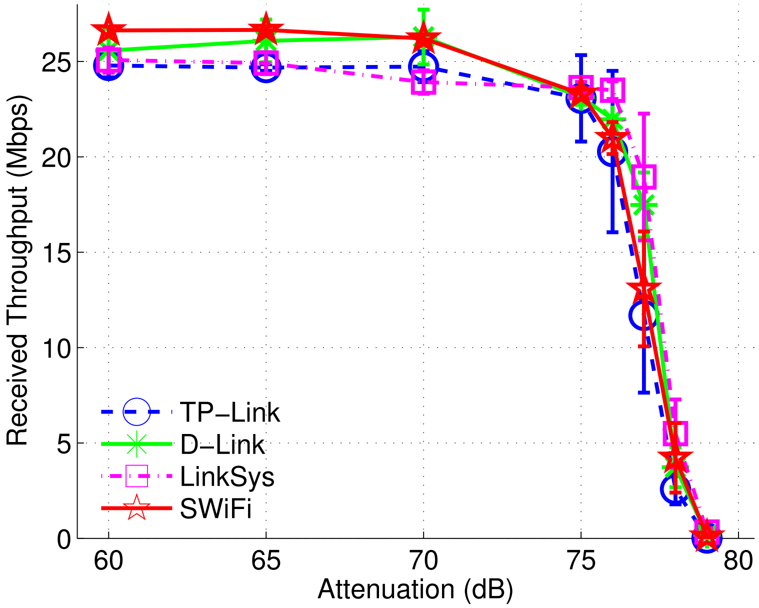


Figure 4.17: Throughput comparison between SWiFi and commodity WiFi cards in wired setup with 1500-byte packet transmission.

4.4.3 Controlled attenuation evaluation

In order to obtain a clearer view on how the different cards fair against SWiFi as the SNR decreases, we carried a set of controlled experiments where the transmitter's signal is first attenuated by 30 dB, then connected to a first splitter. One output of this splitter is further attenuated and connected to the access point, while the other output is attenuated within a range of 30-60dB before being split and connected to the sniffing devices under comparison. This allowed us to understand how the performance of each of the cards degrades as a function of signal attenuation, packet size, and rate.

We observe that at high SNR all cards perform well. SWiFi achieves slightly better performance than the commercial cards. When the attenuation is increased beyond 75 dB the performance of all the cards drops to 0 at 79-80dB attenuation. The similarity in performance of all the cards is remarkable for 1500 bytes packets (See Figure 4.17). It is also interesting to note that for packets of length 1000 bytes, the Linksys card achieves a slightly better performance than the other cards (1-2dB See Figure 4.18). Combined with the results for over the air evaluation, it seems that the Linksys is sensitive to interference and propagation effects more than to low values of SNR. This is consistent with the high volatility in the over the air results of Linksys as such environments are highly dynamic in the crowded 2.4GHz band. A second observation is

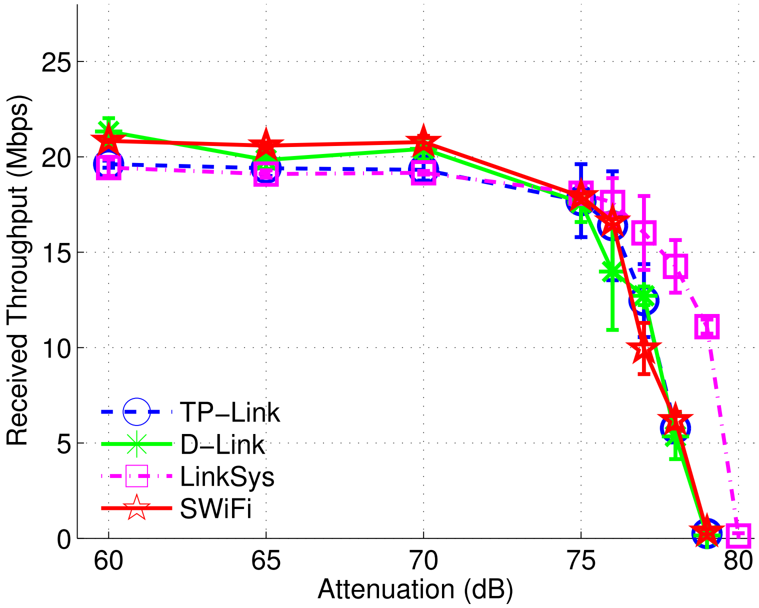


Figure 4.18: Throughput comparison between SWiFi and commodity WiFi cards in wired setup with 1000-byte packet transmission.

that 1000 bytes packets achieve lower throughput consistently at high SNR due to the unnecessary overhead of PHY/MAC headers, SIFS/ACK/DIFS and backoffs.

4.5 Enabling Wi-Fi Analysis

We believe that SWiFi will enable more sophisticated analysis of Wi-Fi networks in particular for high order modulation rates, in addition to enabling the implementation of novel techniques in a Wi-Fi compatible physical layer. To support our claim, we started the development of several companion tools that we plan to make available to the research community as open source.

4.5.1 Timing Analysis

Our first tool can analyze the SWiFi trace and output timing measurements on a per card basis. Most cards exhibit a 16us response time which is the SIFS value for IEEE 802.11an on the 5GHz band and IEEE 802.11g Extended Rate Mode on 2.4GHz band. Note that while a tool such as wireshark reports a timestamp as part of the radiotap header, that is supposed to correspond to the time when the first bit of a packet is received, our

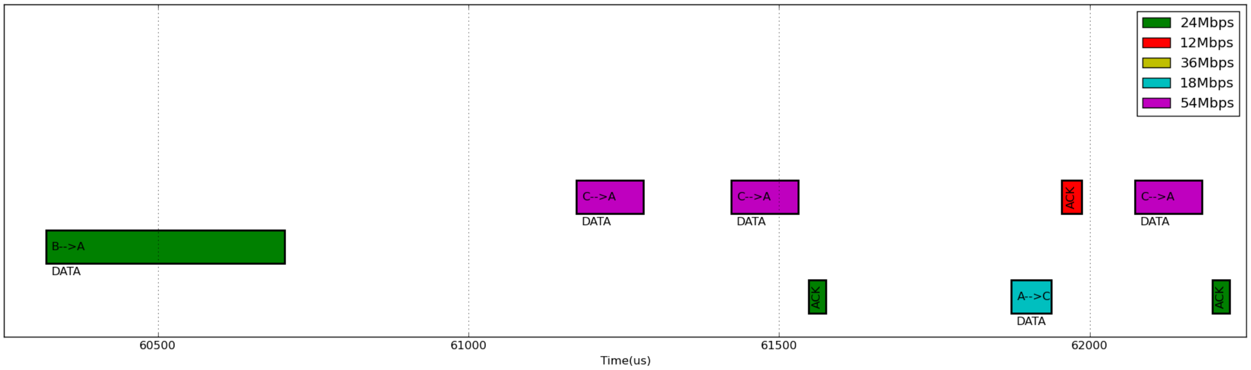


Figure 4.19: Packet flow visualization with rate and precise timing information.

analysis indicates that it is highly inaccurate leading to estimated values for SIFS sometimes corresponding to several hundred microseconds.

This tool also allow to visualize and navigate a SWiFi trace providing information about the src, destination, rate, and most importantly an exact timing information(See Figure 4.19). We plan to extend it to also visualize collisions and overlapping packets (based on some of the preliminary results we have for extracting the header of overlapping packets using successive interference techniques).

4.5.2 Open Source Release to the Community

Once the restrictions on the anonymous submission are lifted, we will make the SWiFi platform available to the research community. Beyond making the source code available, we plan to create an ORBIT image with the source code of SWiFi, companion tools, and scripts running experiments to validate and confirm our claims. Examples of such experiments would consist of an ORBIT configuration with Wi-Fi traffic between two nodes, and sniffed by the USRP N210 and by the Wi-Fi interface of some of the ORBIT nodes. The output of SWiFi can be compared to the output of the Wi-Fi sniffers. Such results can easily be reproduced. Unfortunately, ORBIT does not allow hardware reconfiguration (e.g., installation of splitters) to enable testbeds such as the one we used for comparing commercial cards to SWiFi. However, we hope that the detailed description of our testbed will enable others to reproduce and validate our results.

4.6 Related work

The success of the IEEE802.11 standard and the ubiquity of Wi-Fi networks, attracted a significant amount of research, devoted to their modeling, analysis and evaluation. Early work modeled 802.11 MAC layer using Markov chains and developed an analytical formulation for saturation throughput [10]. Despite the elegant formulation and analytical power, these results were limited to a simple interference model that assumes that all nodes are within range of each other, are IEEE802.11 compliant, and that collisions are binary. More sophisticated models extended the Markov chain framework by considering the hidden terminal scenarios [55], multi-rate physical layer [26], and worst-case interference (jamming) [5]. However, these extensions still retained many of the limitations of the original model, namely a simplistic physical layer and a prediction power that focuses on the steady state behavior and saturation throughput. A large body of work focused on analyzing IEEE802.11 networks using discrete event network simulators such as ns-2 [76], ns-3 [77], OPNET [95], GloMoSim [125], and QualNet [98]. While such environments simplified the analysis of more complex network topologies and accounted for some of the RF signal propagation effects, they were still limited by models that did not correspond to (1) realistic propagation environments, (2) realistic models of IEEE802.11 devices, and (3) realistic models of Bit Error Rate and Frame Error Rate as a function of the considered modulation, coding, noise, interference, and propagation. This can be illustrated by the early models of ns-2 that assumed that all packets with Signal to Noise Ratio exceeding a given ratio will be correctly received, and even ns-3 still retains several flaws in computing the frame error rate of IEEE802.11. Note that discrete event simulators typically focus on the MAC and higher layers behavior, while simulators such as Mathworks Matlab/Simulink are better at simulating the physical and link layers but are not able to scale to networks for nodes with realistic propagation environments.

The limitations of analytical frameworks and simulators motivated wireless systems researchers to develop experimental methods, platforms, and testbed to characterize and develop better models for Wi-Fi networks, but also to evaluate the performance of new algorithms while accounting for realistic propagation, interference, and RF front ends limitations. The ORBIT project provides several flexible wireless networks [47, 82]. Its largest testbed consists of 400 Wi-Fi nodes arranged in a 20 by 20 grid and is supplemented by a limited number of Ettus USRP SDR peripherals. Orbit was successful in enabling large scale, fairly reproducible Wi-Fi experiments, in a controlled environment. It is however not designed for analyzing Wi-Fi networks in the wild. In parallel, several researchers developed measurement-driven approaches to analyze and experiment with dense Wi-Fi networks in real world setting such as [20]. Measurement-driven approaches to Wi-Fi led to better protocols for frequency/channel auto-configuration, load-balancing, power-control, and rate adaptation [73, 102, 121]. However, most of the early work was limited by the limited view of the Wi-Fi channel and link provided by the Hardware Abstraction Layer and the driver API (e.g., RSSI, number of retransmissions). More recently, the availability of a relatively richer set of information about the Wi-Fi channel (RSSI per OFDM sub-carrier at KHz rate) combined with clever algorithms enabled a finer grain characterization of the channel. In particular, it became possible to detect and characterize a wide variety of non-Wi-Fi devices sharing the ISM band [84, 85, 93, 94].

Several research groups considered an alternative approach, to analyzing Wi-Fi networks and their coexistence with other wireless devices over ISM bands. They relied on flexible software defined radios to obtain a much finer grain characterization of the RF spectrum. The introduction, by BBN, of a first SDR implementation of IEEE802.11 (1 & 2Mbps), that runs on the popular Ettus USRP, was a first step to enabling a wide variety of projects from measurement, analysis and optimization of Wi-Fi protocols, to localization, wireless security, and cognitive radios (See [42] for a list of projects). For example RFDump used this implementation to develop a real time wireless multi-protocol analysis tool [62]. Others demonstrated the feasibility of stealthy man-in-the-middle attacks against previously believed to be secure WPA-Enterprise networks [23]. Key to this sophisticated attack is the capability to detect & jam Wi-Fi probes, sent by targeted devices, before the transmission of the CRC, making them invisible to neighboring Access Points. The main limitation of the BBN implementation (besides only supporting rates 1 & 2 Mbps) was the constrained bandwidth of the USRP USB link used to transfer baseband samples to the host PC. This forced the developers to down-sample the baseband signal from 11MHz to 4MHz, therefore significantly degrading the quality of the signal. An alternative implementation by the University of Utah utilized the FPGA capability to detect the frame preamble and despread the baseband signal before transfer to the host computer over USB [34]. This significantly reduced the bandwidth requirement on the USB link, however it still focused on IEEE802.11b at rates 1 & 2 Mbps. Very recently several attempts were made to develop an IEEE802.11abg compatible stack that reaches 54Mbps [13–15]. However, this open source GNU Radio based implementation is limited to QPSK and not able to correctly receive neither 16-QAM nor 64-QAM modulated signals and the reported evaluation only considers packets below 100 Bytes [14]. This is probably due to the limited performance of the Frequency Offset correction and Frequency Domain Equalization throughout the frame. In February 2015, National Instruments announced a commercial OFDM implementation based on the IEEE802.11 standard [72]. Despite its price of $5K, this implementation is not fully compatible with the IEEE802.11 as it implements a simplified PHY frame. The only programmable platform supporting IEEE802.11abg is Rice University's WARP that is commercialized by Mango Communications [118]. Besides its high cost, WARP requires an FPGA implementation of IEEE802.11 which limits the flexibility, ease of programmability and leveraging of the computation capability of the host computer.

To the best of our knowledge, SWiFi is the first IEEE 802.11abg protocol stack that can receive Wi-Fi packets from commodity hardware at rates reaching 54Mbps. We were able to overcome the RF front end limitations of the considered SDR thanks to a carefully designed Frequency Offset correction, and Frequency Domain Equalization. We have rigorously evaluated and compared the performance of our receiver implementation to other commodity Wi-Fi cards demonstrating superior performance over the air and similar performance in wired setups. We show that SWiFi even functions on the low-cost HackRF SDR platform, making IEEE802.11abg networks analysis affordable for a larger community of researchers, and even portable. We believe that further improvements can be achieved using successive interference cancellation techniques to better understand capture effects, and collisions. This open source platform will also make it possible to embed various interference mitigation techniques such as [43, 45] and to inter-operate with commodity Wi-Fi cards.

4.7 Conclusion

We introduced SWiFi, to the best of our knowledge, the first Open Source Wi-Fi SDR stack capable of successfully decoding high order modulation packets (16-QAM and 64-QAM). SWiFi relies on a combination of algorithms for frequency offset correction, and frequency domain equalization utilizing pilot-based phase tracking, and decision directed method for amplitudes equalization. We evaluate the performance of SWiFi on the Ettus USRP N210 over the 2.4 GHz band and compare its performance to three commercial Wi-Fi cards from well known manufacturers and based on popular chipsets. We demonstrate that SWiFi performs at least as well as the commercial cards and exhibits a higher stability in harsh environments. We also demonstrate the potential of this platform for enabling cross-layer research such as timing analysis.

Chapter 5

Multicarrier Jamming on Wi-Fi Communications

In this chapter, we investigate the impact of jamming on Wi-Fi communications with the focus on multi-subcarriers jamming. We consider both continuous and bursty jamming approach with equal power distributed to subcarriers. Our study is based on our SWiFi platform introduced in Chapter 4. We discover that by exploiting the interleaving pattern specified by the IEEE 802.11 standard, an efficient jamming strategy can be devised to completely block the Wi-Fi communication link at very low cost. First, we briefly review the IEEE 802.11's interleaving process in Section 5.1, then we discuss how an efficient jamming strategy is devised in Section 5.2. Our comprehensive evaluation of the jamming strategy in various scenarios is described in Section 5.3. We conclude our work in Section 5.4.

5.1 Overview of Interleaving Mechanism

As specified by the IEEE 802.11 standard, after the Physical Layer frame payload is encoded the convolutional encoder, the coded bit sequence is interleaved. The interleaving process helps scatter the burst errors that might happen during the signal propagation, therefore allowing the convolutional decoder at the receiver to correct the scattered errors.

The interleaving process is handled by the following steps:

- *Grouping:* First, the coded bit sequence produced by the convolutional encoder is divided into multiple groups, each consisting of the same number of bits, *m*, determined by the rate specified in the RATE field of the Physical Header (cf. Table 5.1). Bits within each group are interleaved by two rounds of permutations.

Table 5.1: Rate dependent interleaving parameters

Rate [Mbps]	Modulation & Coding rate	Bits/subcarrier	Interleaving size m [bits]
6	BPSK 1/2	1	48
9	BPSK 3/4	1	48
12	QPSK 1/2	2	96
18	QPSK 3/4	2	96
24	16-QAM 1/2	4	192
36	16-QAM 3/4	4	192
48	64-QAM 2/3	6	288
54	64-QAM 3/4	6	288

- *First-round permutation:* The purpose of the first-round permutation is to scatter adjacent coded bits into non-adjacent subcarriers in order to avoid burst errors that can be caused by unwanted channel distortion in adjacent subcarriers. This permutation is done by first dividing every group of m bits into 16 subgroups. Within each i-th subgroup, the j-th input bit is moved to the i-th bit of the j-th subgroup. Mathematically, if K denotes the index of the bit within a group of m bits, and let $i = \lfloor K/16 \rfloor$, $j = K \bmod 16$, then the location of the bit after the permutation is

$$K' = j\frac{m}{16} + i.$$

- *Second-round permutation:* The second-round permutation's purpose is to place adjacent coded bits into different bits of constellation points in order to avoid biased distortion that might occur on the same bit of multiple constellation points. The permutation rule is defined by the following formula

$$K'' = s\lfloor \frac{K'}{s} \rfloor + (K' + m - \lfloor 16\frac{K'}{m} \rfloor) \bmod s$$

  where $s = \max(b/2, 1)$, and b is the rate-dependent number of bits per subcarrier (Table 5.1).

We can see from the two rounds of permutation defined above that while the first-round permutation separates any two adjacent bits into two different subcarriers, the second round permutes the first-round permuted bits within the same subcarrier. Thus, the interleaving process employed by IEEE 802.11 can be viewed as an outer permutation followed by an inner permutation mechanism.

5.2 Jamming Strategy

5.2.1 Understanding Interleaving Pattern

In this section, we study the most effective jamming pattern that creates a huge impact on data subcarriers (DSC). Based on this, we can achieve an efficient jamming strategy to Wi-Fi communications.

First, we consider an example of interleaving operation for the rate 54Mbps with Modulation and Coding scheme 64-QAM 3/4. According to the interleaving mechanism described above, we can construct the full interleaving table in Table 5.2 (mapping results after two rounds of permutation), which contains 16 rows and 3 columns corresponding to 48 data subcarriers. This table defines the same mapping for 288 coded bits within every OFDM symbol. Each cell in the table represents a subcarrier which is used to transmit 6 interleaved bits. The numbers in each cell indicate the original indices of the pre-interleaved bits (e.g., the first cell shows that after deinterleaving, its 6 bits will be placed back to the original indices 0, 16, 32, 48, 64, and 80). From the mapping, we observe that any two adjacent coded bits belonging to the same constellation symbol are *always permuted into subcarriers of distance 3*, which we call *distance-3 pattern*. For instance, the coded bits 0, 1, 2, 3, 4 (before interleaving) are mapped into bits 0, 20, 37, 54, 74 (after interleaving) corresponding to data subcarriers 0, 3, 6, 9, 12, respectively.

It is interesting to note that this interleaving pattern not only holds for the rate 54Mbps, but for all rates as well. To see this, we consider two bit indices $K_1 = K$ and $K_2 = K + 1$ (before interleaving) which belong to the same subcarrier index, i.e., $i_1 = i_2$ and $j_2 = j_1 + 1$.

After interleaving, the new indices of two bits are $K'_1 = j_1 \frac{m}{16} + i_1$ and $K'_2 = j_2 \frac{m}{16} + i_2$ corresponding to subcarrier indices $M_1 = \left\lfloor \frac{K'_1}{b} \right\rfloor$ mod 48 and $M_2 = \left\lfloor \frac{K'_2}{b} \right\rfloor$ mod 48. We have

$$\begin{aligned} M_2 = \left\lfloor \frac{K'_2}{b} \right\rfloor \bmod 48 &= \left\lfloor \frac{(j_1+1)\frac{m}{16} + i_1}{b} \right\rfloor \bmod 48 \\ &= \left\lfloor \frac{K'_1}{b} + \frac{m}{16b} \right\rfloor \bmod 48 \\ &= \left\lfloor \frac{K'_1}{b} + 3 \right\rfloor \bmod 48 \\ &= (M_2 + 3) \bmod 48 \end{aligned}$$

where the last equality comes from the fact that $m = 48b$ for all rates (as observed from Table 5.1). As a consequence, two consecutive coded bits are always interleaved into subcarriers of distance 3, which confirms the *distance-3 pattern* rule.

5.2.2 Jamming Strategy

The performance of Wi-Fi communications is determined by the error-correcting capability of the convolutional code, whose characteristics is that although it can correct a lot of scattered errors, it is unable to

Table 5.2: Illustration of interleaving for a group of $m = 288$ bits for 54Mbps (64-QAM 3/4). Each cell in the table indicates a subcarrier consisting of 6 bits per constellation symbol.

DSC	Data Subcarrier (DSC) indices increase from left to right, top to bottom																	
0-2	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240	256	272
3-5	17	33	1	65	81	49	113	129	97	161	177	145	209	225	193	257	273	241
6-8	34	2	18	82	50	66	130	98	114	178	146	162	226	194	210	274	242	258
9-11	3	19	35	51	67	83	99	115	131	147	163	179	195	211	227	243	259	275
12-14	20	36	4	68	84	52	116	132	100	164	180	148	212	228	196	260	276	244
15-17	37	5	21	85	53	69	133	101	117	181	149	165	229	197	213	277	245	261
18-20	6	22	38	54	70	86	102	118	134	150	166	182	198	214	230	246	262	278
21-23	23	39	7	71	87	55	119	135	103	167	183	151	215	231	199	263	279	247
24-26	40	8	24	88	56	72	136	104	120	184	152	168	232	200	216	280	248	264
27-29	9	25	41	57	73	89	105	121	137	153	169	185	201	217	233	249	265	281
30-32	26	42	10	74	90	58	122	138	106	170	186	154	218	234	202	266	282	250
33-35	43	11	27	91	59	75	139	107	123	187	155	171	235	203	219	283	251	267
36-38	12	28	44	60	76	92	108	124	140	156	172	188	204	220	236	252	268	284
39-41	29	45	13	77	93	61	125	141	109	173	189	157	221	237	205	269	285	253
42-44	46	14	30	94	62	78	142	110	126	190	158	174	238	206	222	286	254	270
45-47	15	31	47	63	79	95	111	127	143	159	175	191	207	223	239	255	271	287

cope with long burst errors. With the application of interleaving, burst errors occurring during the signal propagation can be spread out to reduce the number of consecutive bit errors. In non-malicious interference environment, this encoding scheme can yield a high performance as typical noise in the environment usually introduces symbol errors at random subcarrier locations. However, in malicious interference scenarios, which we are considering, the jammer can actively corrupt a specific subset of subcarriers to create long burst errors to the received bit sequence. Based on the *distance-3 pattern* property above, we derive the Interleaving jamming strategy as follows.

Interleaving jamming: Select any $i \in \{0,1,2\}$, the adversary jams the data subcarriers at $\{i, i+3, i+6, \ldots, i+3n, \ldots\}$. The number of jammed locations, n, is a parameter dependent on the jamming requirements.

Applying this strategy, the jammed subcarriers will result in jammed adjacent coded bits after deinterleaving the demodulated bit sequence. In the next section, we will experimentally evaluate the impact of this jamming strategy.

5.3 Experimental Results

The performance of Wi-Fi communications in the real testbed is dependent on various factors such as natural noise, malicious jamming, and CSMA mechanism. While the natural noise in the environment is at least 40dB lower than the Wi-Fi signal power (thus not causing much interference to the Wi-Fi transmission – as we verified in our evaluation), the CSMA mechanism employed by Wi-Fi cards can block the transmitter from transmitting. As our goal is to justify only the impact of multicarrier jamming, we remove the side effect of CSMA by carrying out our extensive experiments in a controlled setup as follows. We use two commercial Wi-Fi cards, one as the transmitter and one as the receiver. We put our SWiFi receiver (cf. Chapter 4) at the same location of the receiving Wi-Fi card in order to capture the traffic between the nodes. Without jamming, we verify that our SWiFi receiver receives at least 95% of the transmitted packets for every run of the experiment. This allows us to have a jamming-free baseline for our comparison to jamming scenarios we evaluate later. Now to evaluate the jamming impact, we inject artificially generated noise into the captured samples and run our SWiFi receiver again to collect the statistics. For each experiment, we select a subset of data subcarriers for jamming. Additive white Gaussian noise (AWGN) is generated to each subcarrier in the subset. The generated interfering signals on selected subcarriers are combined and transformed into the time-domain signal for jamming the Wi-Fi communications. In all experiments, we use 1500-byte packets for the Wi-Fi transmissions.

To justify the jamming impact, we use the packet error rate (PER) as the comparison metrics. The experiments are performed in different jamming scenarios, where we adjust the total jamming power, which is the total power of jamming signals in all subcarriers in the selected subset. It is noted that the total jamming power is distributed equally to each selected subcarrier; that is, with the same total jamming power constraint, jamming on more subcarriers will result in less jamming power in each individual subcarrier.

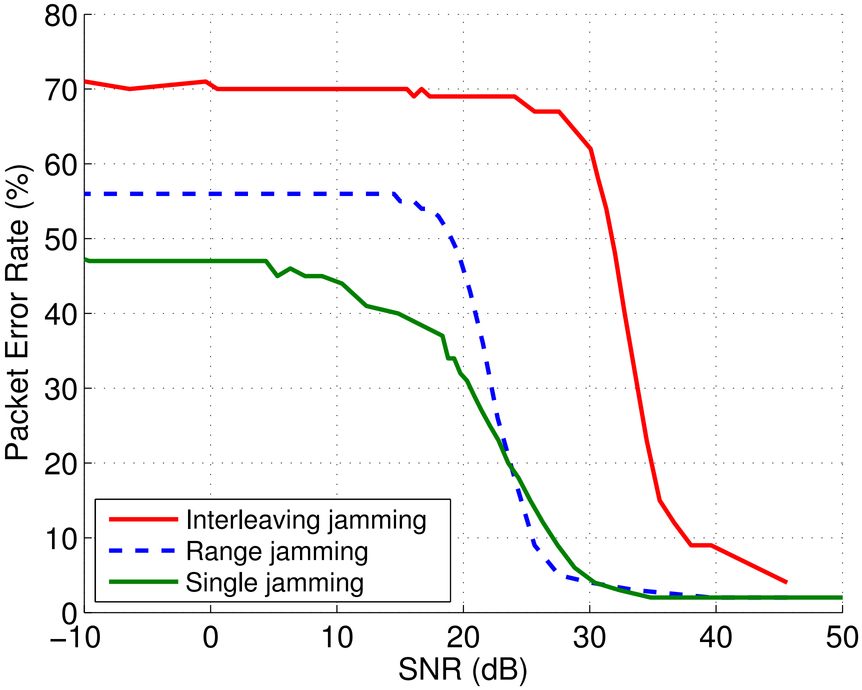


Figure 5.1: Impact of narrow-band jamming strategies: (a) Single jamming on DSC 0; (b) Range jamming on DSC 0, 1, 2; (c) Interleaving jamming on DSC 0, 3, 6.

5.3.1 Narrow-band Jamming

First, we evaluate the impact of narrow-band jamming, where the jamming signal covers only a few subcarriers. Three different scenarios are considered:

- *Single jamming:* This scenario illustrates a narrow-band jammer who can only jam in one single data subcarrier.
- *Range jamming:* In this scenario, the adversary can jam on a few adjacent data subcarriers.
- *Interleaving jamming:* This scenario models a more complex adversary, who can jam on a few non-adjacent data subcarriers following the interleaving jamming pattern.

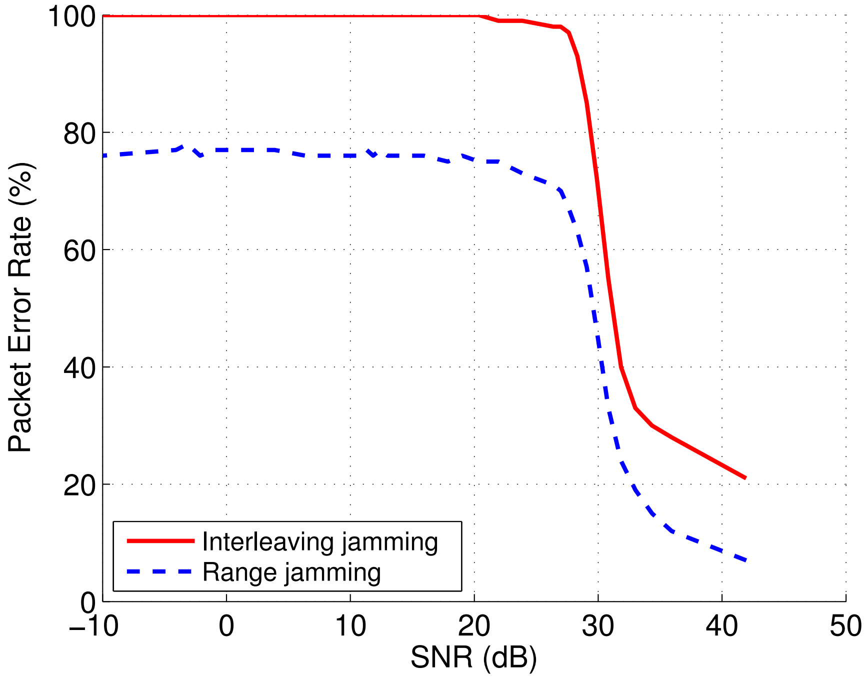


Figure 5.2: Impact of wide-band jamming strategies: (a) Range jamming on DSC 0–6; (b) Interleaving jamming on DSC 0, 3, 6, 9, 12, 15, 18.

We perform the experiment with various selections of jamming subcarriers for each scenario, and we see that for narrow-band jamming, a specific selection of subcarriers for jamming does not result in much different impact on the Wi-Fi reception. Therefore, we only report results for the following selection of jamming subcarriers: (a) for Single jamming, we jam on the first DSC (data subcarrier) 0; (b) for Range jamming, we jam the first three DSC 0, 1, 2; (c) and finally for Interleaving jamming, we jam on DSC 0, 3, 6. Impact on the performance of the Wi-Fi link between the transmitter and the receiver is shown in Figure 5.1. First, it is seen that jamming on a single data subcarrier requires at least 15dB more jamming power to achieve PER of 40% in comparison with Interleaving jamming strategy.

Interestingly, the Range jamming strategy creates slightly less harm (PER 10%) than Single jamming when the SNR is higher than 25dB. It can be explained that with small jamming power, spreading the jamming power over a small range of subcarriers weakens the interference in individual subcarriers. In this case, since the jammer is not aware of the interleaving pattern, these low-power individual jamming subcarriers cannot cooperate to create an effective combination to destroy the packets. In contrast, with the same low power constraint, the Interleaving jamming is able to corrupt 70% of the transmitted packets at the total jamming power 25dB less than the transmitted signal power.

Now we look at lower SNR conditions (less than 20dB), although all three jamming strategies have considerable impacts on the PER (more than 40%), there are specific PER thresholds such that a higher degradation of performance cannot be achieved by the adversary regardless of increasing jamming power. This is explained by the narrowband jamming constraint, which leaves a large enough portion of data subcarriers intact so that there are always some fraction of packets decoded correctly. In this experiment, the PER threshold for Single jamming, Range jamming, and Interleaving jamming are 48%, 56%, 70%, respectively.

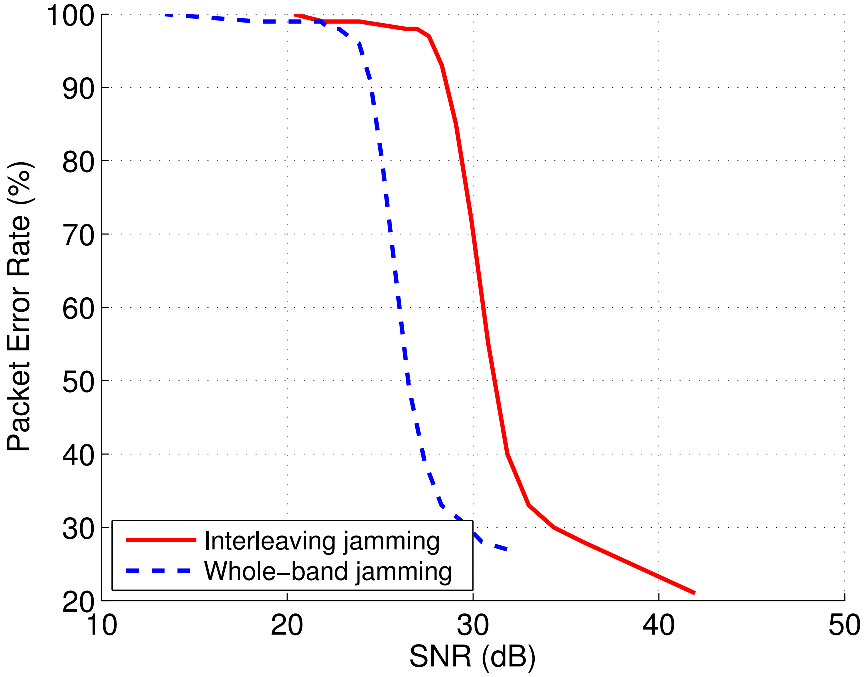


Figure 5.3: Comparison between Interleaving jamming on DSC 0, 3, 6, 9, 12, 15, 18 and Whole-band jamming on all data subcarriers.

5.3.2 Wide-band Jamming

With the goal of achieving higher jamming impact, we use a wide-band jammer which can now jam on a subset of more subcarriers. In particular, we compare the Range jamming and Interleaving jamming with the number of subcarriers selected for jamming increased to 7 subcarriers. We configure the Range jamming on DSC 0–6 and the Interleaving jamming on DSC 0, 3, 6, 9, 12, 15, 18.

Figure 5.2 shows that at high SNR around 30dB, there is a little difference (roughly 2dB) in the required jamming power between Range jamming and Interleaving jamming strategies to block up to 50% of packets. Compared to the narrow-band experiment, there is no more clear advantage of Interleaving jamming over

Range jamming at low-power jamming. This is because on one hand the wide-band Range jamming now also covers three data subcarriers at 0, 3, 6, effectively destroying three consecutive bits after deinterleaving, thus creating more impact than narrow-band Range jamming. On the other hand, Interleaving jamming with expanded number of subcarriers from 3 to 7 only adds little impact due to low power constraint.

However, at SNR around or lower than 20dB, the wide-band Interleaving jamming can now totally corrupt all transmitted packets. In contrast, the Range jamming can never block all the packets, of which 20% still get through. This means the PER threshold disappears for wide-band Interleaving jamming, but still holds for wide-band Range jamming, proving the more efficiency of Interleaving jamming.

5.3.3 Whole-band Jamming vs. Interleaving Jamming

To further justify the efficiency of Interleaving jamming, we conduct an experiment in which we compare its performance to the Whole-band jamming's. For the Whole-band jamming, we generate AWGN over 20MHz of the Wi-Fi channel. The results in Figure 5.3 show that to achieve the same jamming impact, the Whole-band jamming requires about 5dB more power than the Interleaving jamming with a careful selection of subcarriers.

5.3.4 Effect of Number of Jamming Subcarriers

In this experiment, we focus on evaluating the impact of Interleaving jamming in terms of number of data subcarriers selected for jamming. We increase the number of jammed data subcarriers from 2 to 16. As we observed in our results that the impact on Wi-Fi performance only slightly differs when increasing the number of subcarriers beyond 7, we only show the results for Interleaving jamming on $2,3,4,5,6,7$ and 16 (the maximum) data subcarriers. We note that all Interleaving jamming scenarios follow the distance-3 pattern. Figure 5.4 shows that except for Interleaving jamming with a few number (less than or equal to 4) of subcarriers, we can almost block all the packets of the communication link by jamming 5 or more interleaved subcarriers.

It is also interesting to note that Interleaving jamming with maximum 16 data subcarriers is slightly less efficient than using $5,6$ or 7 subcarriers due to more use of subcarriers resulting in less power in individual ones.

5.3.5 Jamming on Pilot Subcarriers

In IEEE 802.11, pilot subcarriers are located among the data subcarriers with equal spacing, and used for channel estimation and equalization. Interference mitigation for pilot subcarriers is, therefore, very

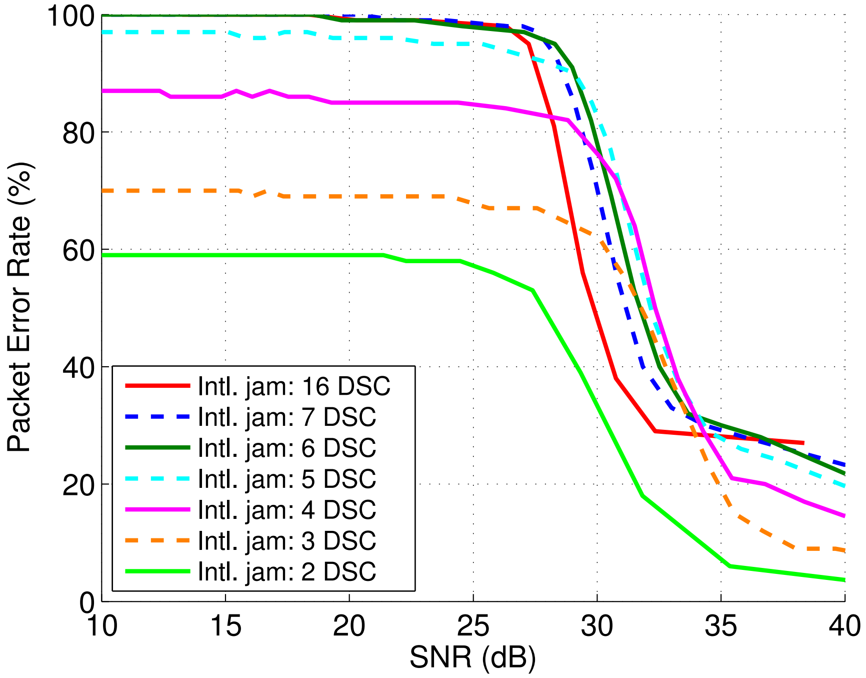


Figure 5.4: Impact of Interleaving jamming strategies with different number of subcarriers.

important for OFDM systems [28, 78, 106]. In this subsection, we investigate the impact of jamming on pilot subcarriers and compare it to Interleaving jamming. For Pilot jamming, the adversary is configured to jam on all four pilot subcarriers. For Interleaving jamming, we select to jam on DSC 0, 3, 6, 9, 12, 15, 18 similarly as in previous experiments. Figure 5.5 shows that jamming on pilot subcarriers results in a slightly less impact than Interleaving jamming, where the latter needs roughly 2dB less power to achieve the same impact. It is noted that both strategies are more efficient than the previously presented Range jamming and Whole-band jamming strategies.

5.3.6 Short-burst Jamming

We have so far considered continuous jamming in our study on multicarrier jamming. To have a more insight on how efficient Interleaving jamming is, we now perform another series of experiments, in which we design a more complex jammer, which generates interference only in a short period of a few OFDM symbols but with high power to target to every transmitted packet. In this subsection, we study the two most efficient jamming strategies discussed above: Interleaving jamming and Pilot jamming.

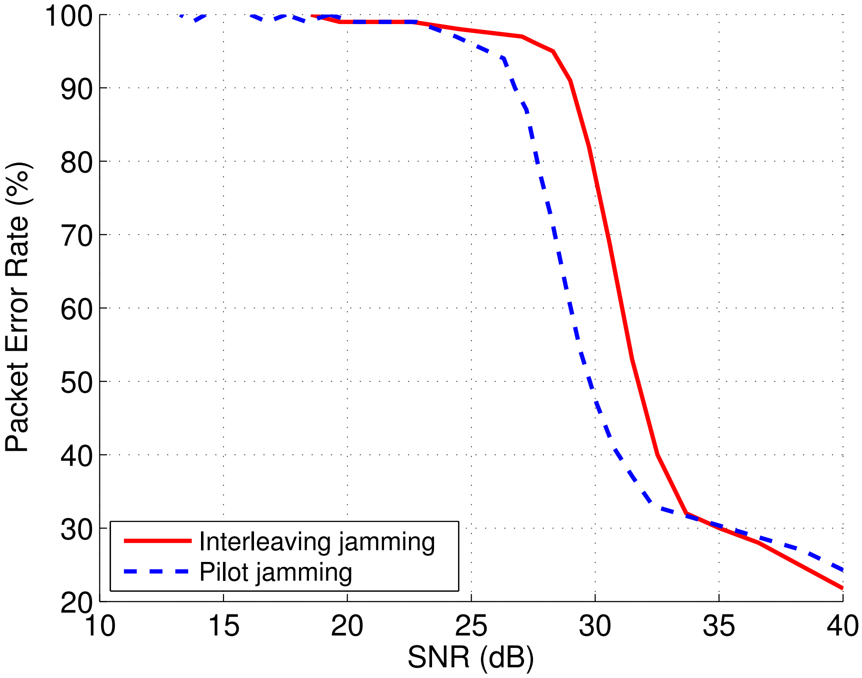


Figure 5.5: Comparison of performance impact by jamming on pilot subcarriers and Interleaving jamming on 7 data subcarriers.

Short-burst Jamming on Pilot Subcarriers

First, we compare the performance of the Wi-Fi transmission under bursty Pilot jamming of different periods. In this experiment, we configure the jammer to jam on all four pilot subcarriers but only on a few OFDM symbols at the beginning of every transmitted packet. The results are shown in Figure 5.6, where we also display the performance impact caused by the continuous Pilot jamming for comparison. We can immediately see that jamming only on the first OFDM symbol of every packet appears as the least efficient burst, and continuously jamming is also less effective than short bursts. In our experiments, we find that jamming on the first 4 OFDM symbols results in the highest efficiency for the adversary, which can corrupt over 80% of the packets using only 0.1% of power compared to the regular transmit power, and 99% of the packets if 1% power is used.

Short-burst Interleaving Jamming

Now we conduct a similar experiment as above to study the impact of short-burst Interleaving jamming strategy. We also consider four different burst lengths from 1 to 4 OFDM symbols for jamming at the

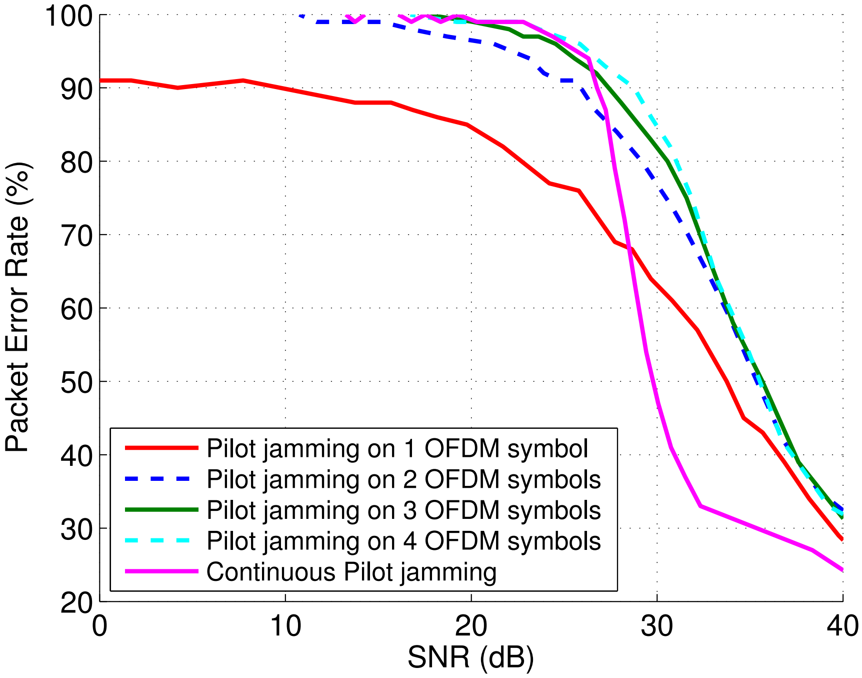


Figure 5.6: Impact of short-burst jamming on pilot subcarriers.

beginning of every packet. For the sake of convenience, the performance impacts caused by the continuous Interleaving jamming and Pilot jamming with burst of 4 OFDM symbols are included in the results shown in Figure 5.7.

First, we see that in contrast to short-burst Pilot jamming strategy, the short-burst Interleaving jamming strategy can cause over 95% of packets incorrectly received by using a burst of only 1 OFDM symbol. Also, this uses a jamming power as low as 0.1% of (30dB less than) the transmitted signal power. Moreover, when the burst length is increased to cover 2, 3, or 4 OFDM symbols' period, the adversary can block 99% of the transmitted packets. In comparison with bursty Pilot jamming, the Interleaving jamming is at least 5dB more power-efficient. The gap of 5dB is also observed when compared to continuous Interleaving jamming.

In summary, the most efficient multicarrier jamming strategy against Wi-Fi communications in our evaluating context is the short-burst Interleaving jamming.

5.4 Conclusion

We devised an efficient jamming strategy that exploits the IEEE 802.11's interleaving mechanism in order to actively introduce burst errors to the Wi-Fi receiver's convolutional decoder resulting in huge impact

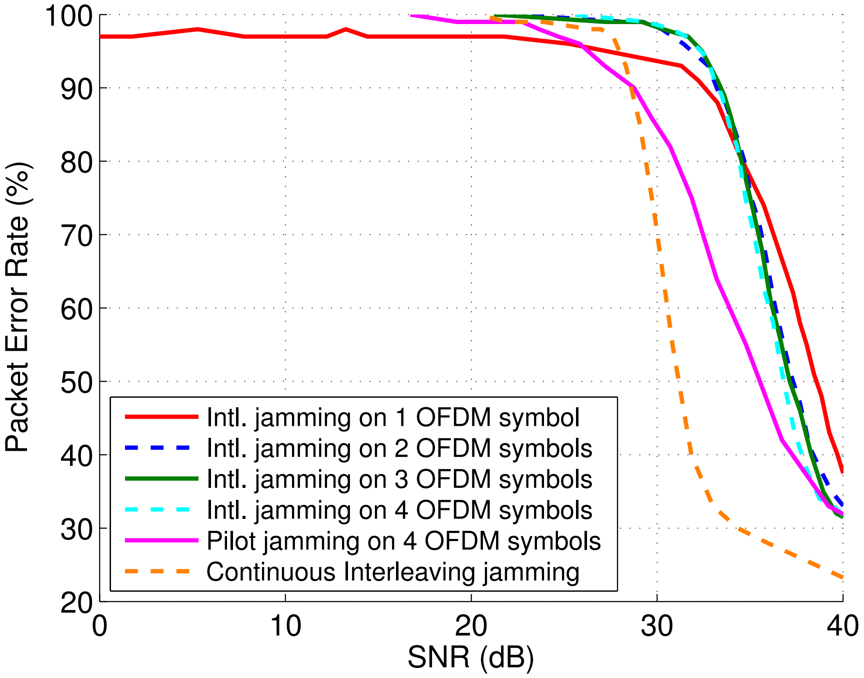


Figure 5.7: Impact of short-burst Interleaving jamming.

on the transmission performance. Our short-burst Interleaving jamming strategy can destroy 99% of the transmitted packets by using a jamming power equal to only 0.1% of regular transmitted signal power. When the jamming power is increased to the fraction of 1%, our strategy can completely block all packets. In comparison with jamming strategies that are unaware of the interleaving structure, we can achieve the same jamming impact with at least 5dB and up to 15dB more power efficiency.

Chapter 6

Future Work

This thesis work has shown the potentials of developing practical and efficient solutions to harden the wireless communication systems against three types of today's increasingly serious jamming threats: high-power jamming attacks, rate-based attacks, and multi-carrier jamming attacks. In the following, we discuss some of possible future work that could be extended based on the proposed solutions in this dissertation.

High-power anti-jamming The proposed scheme for extracting legitimate signals from the jammed received signals has been designed and evaluated for BPSK/QPSK modulation. The jamming cancellation technique, however, can be naturally extended to higher-order modulation as long as the transmit power of the sender remains constant during each interval of extraction process. This condition usually holds for today's communication systems, e.g. IEEE 802.11.

While our current solution applies to one jammer scenarios, it is promising that we can simultaneously mitigate impacts from multiple jammers by extending our antenna algorithm. It is also interesting to add more antennas to the design while keeping the cost low (compared to electronically steerable antenna arrays) and analytically quantify the worst-case gain loss in comparison with the brute-force configuration.

I also believe that the unique beam-forming characteristics of our system results in new research problems in the context of multi-hop wireless network topology control in the presence of malicious interference.

Mitigating rate-based attacks The achieved results in the proposed CBM system show that by protecting the rate information (which is usually exposed in today's communication systems) we actually gain the system robustness instead of losing resiliency. An interesting avenue of future research is to explore generalizations of turbo and LDPC codes to higher order modulations with an integrated frequency and phase correction mechanism. Beyond defending against rate adaptation attacks and boosting the performance of wireless systems at a time where RF spectrum is scarce, our techniques have the potentials to mitigate passive attacks against users traffic analysis.

SWiFi I believe that SWiFi can be extended in many ways for Wi-Fi networks analysis. For example, the incorporation of soft-decoding decoding mechanism can potentially reduce the packet error rate at the receiver. On the other hand, extending the system with successive interference cancellation may recover packets from collision and enable a much better view of capture and hidden terminal effects in dense networks. As another useful application, the framework can be customized for Wi-Fi fingerprinting based on the physical layer's detailed information, which is usually not provided by current Wi-Fi chipsets.

Wi-Fi interleaving jamming The preliminary results on Wi-Fi interleaving jamming in my dissertation have shown a new kind of efficient attacks specifically to Wi-Fi communications. It is interesting to study more thoroughly the jamming impacts according to a more dynamic pattern, e.g., jamming more carriers on a few first OFDM symbols while jamming less on the last few OFDM symbols. The reason behind this idea is that Wi-Fi packets usually have padding at the end; thus jamming at the end might not have as much impact as in the beginning.

For Wi-Fi anti-jamming, while it seems reasonable to mitigate the interleaving jamming attacks by applying the cryptographic interleaving mechanism proposed in this dissertation, it requires more study on protection mechanisms without changing the IEEE 802.11 standard.

Bibliography

[1] 3GPP TS 24.312. Access Network Discovery and Selection Function (ANDSF) Management Object (MO). `http://www.3gpp.org/DynaReport/24312.htm`, 2014.

[2] Naveed Ahmed, Christina Pöpper, and Srdjan Capkun. Enabling short fragments for uncoordinated spread spectrum communication. In Mirosław Kutyłowski and Jaideep Vaidya, editors, *Computer Security - ESORICS 2014*, volume 8712 of *Lecture Notes in Computer Science*, pages 488–507. Springer International Publishing, 2014.

[3] J.S. Atkinson, O. Adetoye, M. Rio, J.E. Mitchell, and G. Matich. Your wifi is leaking: Inferring user behaviour, encryption irrelevant. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 1097–1102, April 2013.

[4] Baruch Awerbuch, Andrea Richa, and Christian Scheideler. A jamming-resistant mac protocol for single-hop wireless networks. In *Proceedings of the Twenty-seventh ACM Symposium on Principles of Distributed Computing*, PODC '08, pages 45–54, New York, NY, USA, 2008. ACM.

[5] Emrah Bayraktaroglu, Christopher King, Xin Liu, Guevara Noubir, Rajmohan Rajaraman, and Bishal Thapa. Performance of IEEE 802.11 under jamming. *Mobile Networks and Applications*, pages 1–19, 2011.

[6] BBN Technologies. ADROIT GNU Radio development. `https://moo.cmcl.cs.cmu.edu/trac/cgran/wiki/BBN80211`.

[7] Mihir Bellare. New proofs for nmac and hmac: Security without collision-resistance. In *Proceedings of the 26th Annual International Conference on Advances in Cryptology*, CRYPTO'06, pages 602–619, Berlin, Heidelberg, 2006. Springer-Verlag.

[8] Mihir Bellare, Thomas Ristenpart, Phillip Rogaway, and Till Stegers. Format-preserving encryption. In Jr. Jacobson, MichaelJ., Vincent Rijmen, and Reihaneh Safavi-Naini, editors, *Selected Areas in*

Cryptography, volume 5867 of *Lecture Notes in Computer Science*, pages 295–312. Springer Berlin Heidelberg, 2009.

[9] Michael A. Bender, Martin Farach-Colton, Simai He, Bradley C. Kuszmaul, and Charles E. Leiserson. Adversarial contention resolution for simple channels. In *Proceedings of the Seventeenth Annual ACM Symposium on Parallelism in Algorithms and Architectures*, SPAA ’05, pages 325–332, New York, NY, USA, 2005. ACM.

[10] G. Bianchi. Performance analysis of the ieee 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, March 2000.

[11] John Charles Bicket. Bit-rate selection in wireless networks. Master’s thesis, Massachusetts Intitute of Technology, 2005.

[12] John Black and Phillip Rogaway. Ciphers with arbitrary finite domains. In Bart Preneel, editor, *Topics in Cryptology — CT-RSA 2002*, volume 2271 of *Lecture Notes in Computer Science*, pages 114–130. Springer Berlin Heidelberg, 2002.

[13] Bastian Bloessl, Michele Segata, Christoph Sommer, and Falko Dressler. A GNURadio Based Receiver Toolkit for IEEE 802.11a/g/p. In *19th ACM International Conference on Mobile Computing and Networking (MobiCom 2013), 5th Wireless of the Students, by the Students, for the Students Workshop (S3 2013), Demo Session*, Miami, FL, October 2013. ACM.

[14] Bastian Bloessl, Michele Segata, Christoph Sommer, and Falko Dressler. An IEEE 802.11a/g/p OFDM Receiver for GNU Radio. In *ACM SIGCOMM 2013, 2nd ACM SIGCOMM Workshop of Software Radio Implementation Forum (SRIF 2013)*, pages 9–16, Hong Kong, China, August 2013. ACM.

[15] Bastian Bloessl, Michele Segata, Christoph Sommer, and Falko Dressler. Decoding IEEE 802.11a/g/p OFDM in Software using GNU Radio. In *19th ACM International Conference on Mobile Computing and Networking (MobiCom 2013), Demo Session*, pages 159–161, Miami, FL, October 2013. ACM.

[16] Eric Blossom. GNU Radio: Tools for Exploring the Radio Frequency Spectrum. *Linux J.*, 2004(122), June 2004.

[17] A. Bourdoux, H. Cappelle, and A. Dejonghe. Channel tracking for fast time-varying channels in ieee802.11p systems. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1–6, Dec 2011.

[18] E. Brookner. Phased arrays and radars – past, present and future. *Microwave Journal*, 49(1):24–46, 2006. ISSN 01926225.

[19] E. Brookner. Phased-array radar: Past, astounding breakthroughs, and future trends. *Microwave Journal*, 51(1):30–50, 2008. ISSN 01926225.

[20] Ioannis Broustis, Konstantina Papagiannaki, Srikanth V. Krishnamurthy, Michalis Faloutsos, and Vivek P. Mhatre. Measurement-driven guidelines for 802.11 WLAN design. *IEEE/ACM Transactions on Networking*, 18(3):722–735, June 2010.

[21] Ioannis Broustis, Konstantinos Pelechrinis, Dimitris Syrivelis, Srikanth V. Krishnamurthy, and Leandros Tassiulas. Fiji: Fighting implicit jamming in 802.11 wlans. In Yan Chen, TassosD. Dimitriou, and Jianying Zhou, editors, *Security and Privacy in Communication Networks*, volume 19 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 21–40. Springer Berlin Heidelberg, 2009.

[22] A. Cassola, Tao Jin, G. Noubir, and B. Thapa. Efficient spread spectrum communication without preshared secrets. *Mobile Computing, IEEE Transactions on*, 12(8):1669–1680, Aug 2013.

[23] A. Cassola, W. Robertson, E. Kirda, and G. Noubir. A practical, targeted, and stealthy attack against wpa enterprise authentication. In *Proceedings of NDSS*, 2013.

[24] Jinn-Ja Chang, Der-June Hwang, and Mao-Chao Lin. Some extended results on the search for good convolutional codes. *Information Theory, IEEE Transactions on*, 43(5):1682–1697, Sep 1997.

[25] J.T. Chiang and Yin-Chun Hu. Dynamic jamming mitigation for wireless broadcast networks. In *INFOCOM*, INFOCOM'08, pages 1211–1219, Piscataway, NJ, USA, 2008. IEEE Press.

[26] Jaehyuk Choi, Kihong Park, and Chong-Kwon Kim. Cross-layer analysis of rate adaptation, dcf and tcp in multi-rate wlans. In *IEEE INFOCOM*, pages 1055–1063, May 2007.

[27] Jung Il Choi, Mayank Jain, Kannan Srinivasan, Phil Levis, and Sachin Katti. Achieving single channel, full duplex wireless communication. In *Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking*, MobiCom '10, pages 1–12, New York, NY, USA, 2010. ACM.

[28] AJ. Coulson. Narrowband interference in pilot symbol assisted ofdm systems. *Wireless Communications, IEEE Transactions on*, 3(6):2277–2287, Nov 2004.

[29] Jing Dong, Reza Curtmola, and Cristina Nita-Rotaru. Practical defenses against pollution attacks in intra-flow network coding for wireless mesh networks. In *Proceedings of the Second ACM Conference on Wireless Network Security*, WiSec '09, pages 111–122, New York, NY, USA, 2009. ACM.

[30] Ettus Research. Universal Software Radio Peripheral. `http://www.ettus.com/`.

[31] FCC. Jammer enforcement, 2012. `http://www.fcc.gov/encyclopedia/jammer-enforcement`, `http://transition.fcc.gov/eb/News_Releases/DOC-304575A1.html`.

[32] FCC. Marriott hotels fined $600,000 by FCC for jamming Wi-Fi hotspots. `http://transition.fcc.gov/Daily_Releases/Daily_Business/2014/db1003/DA-14-1444A1.pdf`, October 2014.

[33] J.A. Fernandez, K. Borries, Lin Cheng, B.V.K.V. Kumar, D.D. Stancil, and Fan Bai. Performance of the 802.11p physical layer in vehicle-to-vehicle environments. *Vehicular Technology, IEEE Transactions on*, 61(1):3–14, Jan 2012.

[34] Hamed Firooz, Neal Patwari, Junxing Zhang, and Sneha K. Kasera. Implementation of full-bandwidth 802.11b receiver. `http://span.ece.utah.edu/pmwiki/pmwiki.php?n=Main.80211bReceiver`, 2008.

[35] K. Firouzbakht, G. Noubir, and M. Salehi. Superposition coding in an adversarial environment. In *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, pages 1–5, March 2011.

[36] K. Firouzbakht, G. Noubir, and M. Salehi. On the performance of multi-layer superposition coding scheme under constrained jamming. In *Information Sciences and Systems (CISS), 2013 47th Annual Conference on*, pages 1–6, March 2013.

[37] K. Firouzbakht, G. Noubir, and M. Salehi. On the performance of adaptive packetized wireless communication links under jamming. *Wireless Communications, IEEE Transactions on*, 13(7):3481–3495, July 2014.

[38] K. Firouzbakht, G. Noubir, and M. Salehi. Packetized wireless communication under jamming, a constrained bimatrix game. In *Global Communications Conference (GLOBECOM), 2014 IEEE*, pages 740–745, Dec 2014.

[39] Koorosh Firouzbakht, Guevara Noubir, and Masoud Salehi. On the capacity of rate-adaptive packetized wireless communication links under jamming. In *Proceedings of the Fifth ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WISEC '12, pages 3–14, New York, NY, USA, 2012. ACM.

[40] Free Forfait Mobile. FreeWiFi secure EAP-SIM. `http://mobile.free.fr/assistance/261.html`, August 2010.

[41] Seth Gilbert, Rachid Guerraoui, and Calvin Newport. Of malicious motes and suspicious sensors: On the efficiency of malicious interference in wireless networks. In MariamMomenzadehAlexanderA. Shvartsman, editor, *Principles of Distributed Systems*, volume 4305 of *Lecture Notes in Computer Science*, pages 215–229. Springer Berlin Heidelberg, 2006.

[42] GNU Radio. The comprehensive GNU Radio archive network. `https://moo.cmcl.cs.cmu.edu/trac/cgran/wiki/Projects`.

[43] Shyamnath Gollakota, Fadel Adib, Dina Katabi, and Srinivasan Seshan. Clearing the RF smog: making 802.11n robust to cross-technology interference. In *Proceedings of the ACM SIGCOMM 2011 conference*, SIGCOMM'11, pages 170–181, New York, NY, USA, 2011. ACM.

[44] Shyamnath Gollakota, Haitham Hassanieh, Benjamin Ransford, Dina Katabi, and Kevin Fu. They can hear your heartbeats: Non-invasive security for implantable medical devices. In *Proceedings of the ACM SIGCOMM 2011 Conference*, SIGCOMM '11, pages 2–13, New York, NY, USA, 2011. ACM.

[45] Shyamnath Gollakota and Dina Katabi. Zigzag decoding: Combating hidden terminals in wireless networks. In *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication*, SIGCOMM '08, pages 159–170, New York, NY, USA, 2008. ACM.

[46] Great Scott Gadgets. Hackrf one. `https://greatscottgadgets.com/hackrf/`.

[47] George C. Hadjichristofi, Avi Brender, Marco Gruteser, Rajesh Mahindra, and Ivan Seskar. A wired-wireless testbed architecture for network layer experimentation based on ORBIT and VINI. In *Proceedings of the ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization*, WinTECH '07, pages 83–90, New York, NY, USA, 2007. ACM.

[48] Fred Harris. Let's assume the system is synchronized. In Ramjee Prasad, Sudhir Dixit, Richard van Nee, and Tero Ojanpera, editors, *Globalization of Mobile and Wireless Communications*, Signals and Communication Technology, pages 311–325. Springer, 2011.

[49] Fredric J. Harris. *Multirate Signal Processing for Communication Systems*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004.

[50] H. Haverinen and J. Salowey. Extensible Authentication Protocol Method for Global System for Mobile Communications (GSM) Subscriber Identity Modules (EAP-SIM). RFC 4186 (Informational), January 2006.

[51] Gavin Holland, Nitin Vaidya, and Paramvir Bahl. A rate-adaptive MAC protocol for multi-hop wireless networks. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, MobiCom '01, pages 236–251, New York, NY, USA, 2001. ACM.

[52] iClinks. Scada and industrial automation, ethernet scada and ethernet i/o, 2011. `http://www.iclinks.com/`.

[53] IEEE. IEEE 802.11 Interworking with External Networks. `http://standards.ieee.org/findstds/standard/802.11u-2011.html`.

[54] IEEE. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, February 2012.

[55] Beakcheol Jang and M.L. Sichitiu. IEEE 802.11 saturation throughput analysis in the presence of hidden terminals. *IEEE/ACM Transactions on Networking*, 20(2):557–570, April 2012.

[56] Tao Jin, Guevara Noubir, and Bishal Thapa. Zero pre-shared secret key establishment in the presence of jammers. In *Proceedings of the Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '09, pages 219–228, New York, NY, USA, 2009. ACM.

[57] S. Kay. A fast and accurate single frequency estimator. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 37(12):1987–1990, Dec 1989.

[58] Jongseok Kim, Seongkwan Kim, Sunghyun Choi, and D. Qiao. CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–11, April 2006.

[59] Chiu-Yuen Koo, Vartika Bhandari, Jonathan Katz, and Nitin H. Vaidya. Reliable broadcast in radio networks: The bounded collision case. In *Proceedings of the Twenty-fifth Annual ACM Symposium on Principles of Distributed Computing*, PODC '06, pages 258–264, New York, NY, USA, 2006. ACM.

[60] John D. Kraus, Ronald J. Marhefka, and Ronald J. Marhefka. *Antennas*. Mcgraw Hill Higher Education, 3rd edition, 2001.

[61] Mathieu Lacage, Mohammad Hossein Manshaei, and Thierry Turletti. IEEE 802.11 rate adaptation: A practical approach. In *Proceedings of the 7th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '04, pages 126–134, New York, NY, USA, 2004. ACM.

[62] Kaushik Lakshminarayanan, Samir Sapra, Srinivasan Seshan, and Peter Steenkiste. Rfdump: An architecture for monitoring the wireless ether. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, ACM CoNEXT'09, pages 253–264, New York, NY, USA, 2009. ACM.

[63] Barry Levine. Who is putting up 'interceptor' cell towers? The mystery deepens. *Venturebeat*, September 2014. `http://venturebeat.com/2014/09/02/who-is-putting-up-interceptor-cell-towers-the-mystery-deepens/`.

[64] Shu Lin and Daniel J. Costello. *Error Control Coding*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2 edition, 2004.

[65] An Liu, Peng Ning, Huaiyu Dai, Yao Liu, and Cliff Wang. Defending dsss-based broadcast communication against insider jammers via delayed seed-disclosure. In *Proceedings of the 26th Annual Computer Security Applications Conference*, ACSAC '10, pages 367–376, New York, NY, USA, 2010. ACM.

[66] S. Liu, L. Lazos, and M. Krunz. Thwarting inside jamming attacks on wireless broadcast communications. In *In Proceedings of ACM WiSec*, WiSec'11, pages 29–40, New York, NY, USA, 2011. ACM.

[67] Y. Liu, P. Ning, H. Dai, and A. Liu. Randomized differential dsss: jamming-resistant wireless broadcast communication. In *INFOCOM*, INFOCOM'10, pages 695–703, Piscataway, NJ, USA, 2010. IEEE Press.

[68] MadWifi-Project. Bit-rate selection algorithms. `http://madwifi-project.org/wiki/UserDocs/RateControl`, 2012.

[69] R.J. Mailloux. *Phased Array Antenna Handbook*. Artech Print on Demand, 2005.

[70] R. Miller. FCC steps up crackdown on cell jammers, 2012. `http://www.securitysystemsnews.com/article/fcc-steps-crackdown-cell-jammers`.

[71] Minstrel. MadWifi rate control. `http://madwifi-project.org/browser/madwifi/trunk/ath_rate/minstrel`, 2011.

[72] National Instruments. LabVIEW communications 802.11 application framework. `http://www.ni.com/product-documentation/52533/en/`, February 2015.

[73] Michael Neufeld, Jeff Fifield, Christian Doerr, Anmol Sheth, and Dirk Grunwald. SoftMAC - Flexible Wireless Research Platform. In *Fourth Workshop on Hot Topics in Networks (HotNets-IV)*, November 2005.

[74] Guevara Noubir, Rajmohan Rajaraman, Bo Sheng, and Bishal Thapa. On the robustness of IEEE 802.11 rate adaptation algorithms against smart jamming. In *Proceedings of the Fourth ACM Conference on Wireless Network Security*, WiSec '11, pages 97–108, New York, NY, USA, 2011. ACM.

[75] NPR. Congress passes FAA bill that speeds switch to GPS, 2 2012. `http://www.npr.org/`.

[76] ns 2. The network simulator - ns-2. `http://www.isi.edu/nsnam/ns/`.

[77] ns 3. Nsnam - ns-3. `https://www.nsnam.org`.

[78] Shuichi Ohno, Emmanuel Manasseh, and Masayoshi Nakamoto. Preamble and pilot symbol design for channel estimation in ofdm systems with null subcarriers. *EURASIP Journal on Wireless Communications and Networking*, 2011(1), 2011.

[79] Onoe. MadWifi rate control. `http://madwifi-project.org/browser/madwifi/trunk/ath_rate/onoe`, 2011.

[80] C. Orakcal and D. Starobinski. Jamming-resistant rate control in wi-fi networks. In *Global Communications Conference (GLOBECOM), 2012 IEEE*, pages 1048–1053, Dec 2012.

[81] Cankut Orakcal and David Starobinski. Jamming-resistant rate adaptation in wi-fi networks. *Performance Evaluation*, 75–76(0):50 – 68, 2014.

[82] M. Ott, I. Seskar, R. Siraccusa, and M. Singh. ORBIT testbed software architecture: supporting experiments as a service. In *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on*, pages 136–145, Feb 2005.

[83] David Pacholok. ATV Transmitter from a Microwave Oven! Low-cost high-power microwave operation has arrived. *Amateur Radio*, Jul 1989.

[84] Ashish Patro, Srinivas Govindan, and Suman Banerjee. Observing home wireless experience through WiFi APs. In *Proceedings of the 19th Annual International Conference on Mobile Computing and Networking*, MobiCom'13, pages 339–350, New York, NY, USA, 2013. ACM.

[85] Ashish Patro, Shravan Rayanchu, and Suman Banerjee. Mobicom 2011 poster: AirTrack: Locating non-WiFi interferers using commodity WiFi hardware. *SIGMOBILE Mob. Comput. Commun. Rev.*, 15(4):52–54, March 2012.

[86] Konstantinos Pelechrinis, Ioannis Broustis, Srikanth V. Krishnamurthy, and Christos Gkantsidis. Ares: An anti-jamming reinforcement system for 802.11 networks. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '09, pages 181–192, New York, NY, USA, 2009. ACM.

[87] PG & E. Smart meters by the numbers. http://www.pge.com/myhome/customerservice/smartmeter/deployment/, 2011. `http://www.pge.com/myhome/customerservice/smartmeter/deployment/`.

[88] John G. Proakis and Masoud Salehi. *Digital Communications*. McGraw-Hill, 5 edition, 2007.

[89] Qualcomm. 3G LTE Wifi offload framework: Connectivity Engine (CnE) solution, July 2013. `http://www.qualcomm.com/media/documents/3g-lte-wifi-offload-framework`.

[90] Hanif Rahbari and Marwan Krunz. Friendly CryptoJam: A Mechanism for Securing Physical-layer Attributes. In *Proceedings of the 2014 ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WiSec '14, pages 129–140, New York, NY, USA, 2014. ACM.

[91] Maisie Ramsay. Wi-Fi offload rising amid soaring data traffic. `http://www.wirelessweek.com/News/2012/07/technology-WiFi-Offload-Rising-Amid-Soaring-Data-Traffic/`, July 2012.

[92] Jianjun Ran, R. Grunheid, H. Rohling, E. Bolinth, and R. Kern. Decision-directed channel estimation method for ofdm systems with high velocities. In *Vehicular Technology Conference, 2003. VTC 2003-Spring. The 57th IEEE Semiannual*, volume 4, pages 2358–2361 vol.4, April 2003.

[93] Shravan Rayanchu, Ashish Patro, and Suman Banerjee. Airshark: Detecting non-WiFi rf devices using commodity WiFi hardware. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC'11, pages 137–154, New York, NY, USA, 2011. ACM.

[94] Shravan Rayanchu, Ashish Patro, and Suman Banerjee. Catching whales and minnows using WiFiNet: Deconstructing non-WiFi interference using WiFi hardware. In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, NSDI'12, pages 5–5, Berkeley, CA, USA, 2012. USENIX Association.

[95] Riverbed. OPNET modeler. `http:www.riverbed.com/products/performance-management-control/`.

[96] Phillip Rogaway. A synopsis of format-preserving encryption, 2010.

[97] Ben Rooney. Data-hungry 4G users gorge on Wi-Fi, report finds. `http://blogs.wsj.com/tech-europe/2013/09/19/data-hungry-4g-users-gorge-on-wi-fi-report-finds/`, September 2013.

[98] Scalable Network Technologies. QualNet communications simulation platform. `http://scalable-networks.com/content/qualnet`.

[99] T.M. Schmidl and D.C. Cox. Robust frequency and timing synchronization for OFDM. *Communications, IEEE Transactions on*, 45(12):1613–1621, Dec 1997.

[100] Bruce Schneier. Fake cell phone towers across the US. `https://www.schneier.com/blog/archives/2014/09/fake_cell_phone.html`, Sep. 2014.

[101] SEMAPHORE. Integrated scada, control, and communication solutions. `http://www.cse-semaphore.com/`, 2011.

[102] Ashish Sharma, M. Tiwari, and Haitao Zheng. MadMAC: Building a reconfiguration radio testbed using commodity 802.11 hardware. In *Networking Technologies for Software Defined Radio Networks, 2006. SDR '06.1st IEEE Workshop on*, pages 78–83, Sept 2006.

[103] Daniel P. Shepard, Todd E. Humphreys, and Aaron A. Fansler. Evaluation of the vulnerability of phasor measurement units to gps spoofing attacks. *International Journal of Critical Infrastructure Protection*, 5(3-4):146–153, 2012.

[104] Marvin K. Simon, Jim K. Omura, Robert A. Scholtz, and Barry K. Levitt. *Spread Spectrum Communications Handbook*. McGraw-Hill, 2001.

[105] David Slater, Patrick Tague, Radha Poovendran, and Brian J. Matt. A coding-theoretic approach for efficient message verification over insecure channels. In *Proceedings of the Second ACM Conference on Wireless Network Security*, WiSec '09, pages 151–160, New York, NY, USA, 2009. ACM.

[106] A. Stamoulis, S.N. Diggavi, and N. Al-Dhahir. Intercarrier interference in mimo ofdm. *Signal Processing, IEEE Transactions on*, 50(10):2451–2464, Oct 2002.

[107] M. Strasser, S. Capkun, S. Capkun, and M. Cagalj. Jamming-resistant key establishment using uncoordinated frequency hopping. In *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pages 64–78, May 2008.

[108] Mario Strasser, Christina Pöpper, and Srdjan Čapkun. Efficient uncoordinated FHSS anti-jamming communication. In *Proceedings of the Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '09, pages 207–218, New York, NY, USA, 2009. ACM.

[109] Synetcom. Synetcom industrial wireless systems. `http://www.synetcom.com/`, 2011.

[110] N.O. Tippenhauer, L. Malisa, A Ranganathan, and S. Capkun. On limitations of friendly jamming for confidentiality. In *Security and Privacy (SP), 2013 IEEE Symposium on*, pages 160–173, May 2013.

[111] H.L. Van Trees. *Detection, Estimation, and Modulation Theory, Part I.* Wiley & Sons, 2001.

[112] David Tse and Pramod Viswanath. *Fundamentals of wireless communication*. Cambridge University Press, New York, NY, USA, 2005.

[113] G. Ungerboeck. Channel coding with multilevel/phase signals. *Information Theory, IEEE Transactions on*, 28(1):55–67, Jan 1982.

[114] uPrint. Uprint plus. `http://www.uprint3dprinting.com/`.

[115] vMonitor. Scada wireless systems. `http://www.vmonitor.com/`, 2011.

[116] Triet D. Vo-Huu and Guevara Noubir. CBM source code. `http://www.ccs.neu.edu/home/noubir/projects/cbm`.

[117] Mythili Vutukuru, Hari Balakrishnan, and Kyle Jamieson. Cross-layer wireless bit rate adaptation. In *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication*, SIGCOMM '09, pages 3–14, New York, NY, USA, 2009. ACM.

[118] WARP. Wireless open-access research platform. `http://warp.rice.edu/`.

[119] Matthias Wilhelm, Ivan Martinovic, Jens B. Schmitt, and Vincent Lenders. Short paper: Reactive jamming in wireless networks: How realistic is the threat? In *Proceedings of the Fourth ACM Conference on Wireless Network Security*, WiSec '11, pages 47–52, New York, NY, USA, 2011. ACM.

[120] Starsky H. Y. Wong, Hao Yang, Songwu Lu, and Vaduvur Bharghavan. Robust rate adaptation for 802.11 wireless networks. In *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking*, MobiCom '06, pages 146–157, New York, NY, USA, 2006. ACM.

[121] Haitao Wu, Yunxin Liu, Qian Zhang, and Zhi-Li Zhang. SoftMAC: Layer 2.5 collaborative MAC for multimedia support in multihop wireless networks. *Mobile Computing, IEEE Transactions on*, 6(1):12–25, Jan 2007.

[122] Wenyuan Xu, Ke Ma, W. Trappe, and Y. Zhang. Jamming sensor networks: attack and defense strategies. *Network, IEEE*, 20(3):41–47, May 2006.

[123] Wenyuan Xu, Wade Trappe, and Yanyong Zhang. Channel surfing: defending wireless sensor networks from interference. In *Proceedings of IPSN*, pages 499–508, New York, NY, USA, 2007. ACM.

[124] Der-Yeuan Yu, Aanjhan Ranganathan, Thomas Locher, Srdjan Capkun, and David Basin. Short Paper: Detection of GPS Spoofing Attacks in Power Grids. In *Proceedings of the 2014 ACM Conference on Security and Privacy in Wireless & Mobile Networks*, WiSec '14, pages 99–104, New York, NY, USA, 2014. ACM.

[125] Xiang Zeng, Rajive Bagrodia, and Mario Gerla. GloMoSim: A library for parallel simulation of large-scale wireless networks. In *Proceedings of the Twelfth Workshop on Parallel and Distributed Simulation*, PADS '98, pages 154–161, Washington, DC, USA, 1998. IEEE Computer Society.

[126] Wei Zhang, M. Kamgarpour, Dengfeng Sun, and C.J. Tomlin. A hierarchical flight planning framework for air traffic management. *Proceedings of the IEEE*, 100(1):179–194, Jan 2012.

[127] Zhenghao Zhang, Shuping Gong, AD. Dimitrovski, and Husheng Li. Time Synchronization Attack in Smart Grid: Impact and Analysis. *Smart Grid, IEEE Transactions on*, 4(1):87–98, March 2013.