TECHNISCHE UNIVERSITÄT DRESDEN

Faculty of Computer Science Institute of Systems Architecture, Chair of Privacy and Security

Towards Practical and Secure Channel Impulse Response-based Physical Layer Key Generation

Paul Walther

Born on: 14th August 1986 in Freiberg

Dissertation

to achieve the academic degree

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

First referee

Prof. Dr. Thorsten Strufe

Second referee

Prof. Dr. Matthias Hollick

Submitted on: 22nd July 2021
Defended on: 15th November 2021

TECHNISCHE UNIVERSITÄT DRESDEN

Faculty of Computer Science Institute of Systems Architecture, Chair of Privacy and Security

Abstract

The current trend towards “smart” devices brings with it a multitude of Internet-enabled and thereby connected devices. The corresponding communication of these devices must inevitably be secured by suitable measures, in order to meet the privacy and security related requirements for the transmitted information. However, the large number of security-critical incidents in the context of “smart” devices and the Internet of Things indicates, that this safeguarding of communication is currently only inadequately implemented.

There are many reasons for this: Essential security measures are sometimes not taken into account in the design process or are not implemented due to price pressure. In addition, the nature of the devices used hinders the application of classic security procedures. In this context, solutions are primarily tailored to specific use cases, and due to the specific hardware used, they usually have only limited computing and energy resources available.

With respect to these limitations, physical layer security (PLS) solutions offer an alternative to classic cryptography. In the context of wireless communication, the properties of the transmission channel between the legitimate communication partners can be used to implement security primitives and thus realize security goals. Specifically, for example, the channel properties can be used to generate a trust anchor in the form of a shared symmetric secret. This approach is called *channel reciprocity based key generation* (CRKG).

Due to its widespread availability, CRKG is usually implemented using the channel property of received signal strength indicator (RSSI). However, this comes with the disadvantage that all physical channel properties are broken down to a single value and thus most of the available information is neglected.

This is contrasted with the use of complete channel state information (CSI). Current technical developments increasingly make it possible to provide this information also in everyday devices and thus to reuse it for PLS.

In this work, we analyse issues arising from the shift towards CSI as a basis for the key material used. Specifically, we investigate CSI in the form of ultra-wideband channel impulse responses (CIR).

For these investigations we conducted extensive real world measurements in realistic settings. These measurements allowed to analyse to what extent the basic assumptions of PLS and CRKG are fulfilled and whether CIRs are suitable for key generation. We show that the CIRs of the legitimate communication partners exhibit a higher similarity than those of an attacker, and that therefore an advantage over the attacker exists at the physical layer that can be exploited for key generation.

Based on the results of these initial analyses, we propose basic procedures, which are necessary to improve the similarity of the legitimate measurements and thus to enable the key generation. Specifically, procedures are presented that remove the temporal offset between reciprocal measurements and thus increase the similarity, and methods that remove the noise that is inevitably present in the measurements.

At the same time, we examine to which extent the fundamental security assumptions are fulfilled from the point of view of an attacker. For this purpose, we present, implement and analyse different practical attack scenarios. These procedures include, for example, approaches that use deterministic channel models or *ray tracing* to predict the legitimate CIRs. Furthermore, we investigate machine learning approaches that aim to infer the legitimate CIRs directly from the observations of an attacker. Especially with the help of the last method, it can be shown here that large parts of the CIRs are deterministically predictable. This leads to the conclusion that raw CIRs should not be used as input for security primitives without adequate preprocessing.

Based on these findings, we then conclude by designing and implementing procedures that are resistant to the attacks presented. The first solution builds from the insight, that the attacks are possible due to predictable parts within the CIRs. Hence, we propose a classical preprocessing approach, which removes these deterministically predictable parts and thus secures the input material. We implement and analyse this solution and show its effectiveness as well as resistance against the proposed attacks. In a second solution, we leverage the capabilities of machine learning by introducing them into the system design as well. Building on their strong pattern recognition performance, we design, implement and analyse a solution that learns to extract the random parts from the raw CIRs that define channel reciprocity and discard all other, deterministic parts. Thus, not only the key material is secured, but also reconciled at the same time, as any differences are efficiently removed. All proposed solutions completely omit the exchange of information between legitimate communication partners, thereby the associated information leakage as well as energy consumption is avoided by design.

Kurzfassung

Der derzeitige Trend hin zu “smarten” Geräten bringt eine Vielzahl an Internet-fähigen und verbundenen Geräten mit sich. Die entsprechende Kommunikation dieser Geräte muss zwangsläufig durch geeignete Maßnahmen abgesichert werden, um die datenschutz- und sicherheitsrelevanten Anforderungen an die übertragenen Informationen zu erfüllen. Jedoch zeigt die Vielzahl an sicherheitskritischen Vorfällen im Kontext von “smarten” Geräten und des Internets der Dinge auf, dass diese Absicherung der Kommunikation derzeit nur unzureichend umgesetzt wird.

Die Ursachen hierfür sind vielfältig: so werden essentielle Sicherheitsmaßnahmen im Designprozess mitunter nicht berücksichtigt oder auf Grund von Preisdruck nicht realisiert. Darüber hinaus erschwert die Beschaffenheit der eingesetzten Geräte die Anwendung klassischer Sicherheitsverfahren. So werden in diesem Kontext vorrangig stark auf Anwendungsfälle zugeschnittene Lösungen realisiert, die auf Grund der verwendeten Hardware meist nur eingeschränkte Rechen- und Energieressourcen zur Verfügung haben.

An dieser Stelle können die Ansätze und Lösungen der Sicherheit auf physikalischer

Schicht (physical layer security, PLS) eine Alternative zu klassischer Kryptografie bieten. Im Kontext der drahtlosen Kommunikation können hier die Eigenschaften des Übertragungskanals zwischen zwei legitimen Kommunikationspartnern genutzt werden, um Sicherheitsprimitive zu implementieren und damit Sicherheitsziele zu realisieren. Konkret können etwa reziproke Kanaleigenschaften verwendet werden, um einen Vertrauensanker in Form eines geteilten, symmetrischen Geheimnisses zu generieren. Dieses Verfahren wird *Schlüsselgenerierung basierend auf Kanalreziprozität* (channel reciprocity based key generation, CRKG) genannt.

Auf Grund der weitreichenden Verfügbarkeit wird dieses Verfahren meist mit Hilfe der Kanaleigenschaft des Empfangsstärkenindikators (received signal strength indicator, RSSI) realisiert. Dies hat jedoch den Nachteil, dass alle physikalischen Kanaleigenschaften auf einen einzigen Wert heruntergebrochen werden und somit ein Großteil der verfügbaren Informationen vernachlässigt wird.

Dem gegenüber steht die Verwendung der vollständigen Kanalzustandsinformationen (channel state information, CSI). Aktuelle technische Entwicklungen ermöglichen es zunehmend, diese Informationen auch in Alltagsgeräten zur Verfügung zu stellen und somit für PLS weiterzuverwenden.

In dieser Arbeit analysieren wir Fragestellungen, die sich aus einem Wechsel hin zu CSI als verwendetes Schlüsselmaterial ergeben. Konkret untersuchen wir CSI in Form von Ultrabreitband-Kanalimpulsantworten (channel impulse response, CIR).

Für die Untersuchungen haben wir initial umfangreiche Messungen vorgenommen und damit analysiert, in wie weit die grundlegenden Annahmen von PLS und CRKG erfüllt sind und die CIRs sich grundsätzlich für die Schlüsselgenerierung eignen. Hier zeigen wir, dass die CIRs der legitimen Kommunikationspartner eine höhere Ähnlichkeit als die eines Angreifers aufzeigen und das somit ein Vorteil gegenüber diesem auf der physikalischen Schicht besteht, der für die Schlüsselgenerierung ausgenutzt werden kann.

Basierend auf den Ergebnissen der initialen Untersuchung stellen wir dann grundlegende Verfahren vor, die notwendig sind, um die Ähnlichkeit der legitimen Messungen zu verbessern und somit die Schlüsselgenerierung zu ermöglichen. Konkret werden Verfahren vorgestellt, die den zeitlichen Versatz zwischen reziproken Messungen entfernen und somit die Ähnlichkeit erhöhen, sowie Verfahren, die das in den Messungen zwangsläufig vorhandene Rauschen entfernen.

Gleichzeitig untersuchen wir, inwieweit die getroffenen fundamentalen Sicherheitsannahmen aus Sicht eines Angreifers erfüllt sind. Zu diesem Zweck präsentieren, implementieren und analysieren wir verschiedene praktische Angriffsmethoden. Diese Verfahren umfassen etwa Ansätze, bei denen mit Hilfe von deterministischen Kanalmodellen oder durch *ray tracing* versucht wird, die legitimen CIRs vorherzusagen. Weiterhin untersuchen wir Machine Learning Ansätze, die darauf abzielen, die legitimen CIRs direkt aus den Beobachtungen eines Angreifers zu inferieren. Besonders mit Hilfe des letzten Verfahrens kann hier gezeigt werden, dass große Teile der CIRs deterministisch vorhersagbar sind. Daraus

leitet sich der Schluss ab, dass CIRs nicht ohne adäquate Vorverarbeitung als Eingabe für Sicherheitsprimitive verwendet werden sollten.

Basierend auf diesen Erkenntnissen entwerfen und implementieren wir abschließend Verfahren, die resistent gegen die vorgestellten Angriffe sind. Die erste Lösung baut auf der Erkenntnis auf, dass die Angriffe aufgrund von vorhersehbaren Teilen innerhalb der CIRs möglich sind. Daher schlagen wir einen klassischen Vorverarbeitungsansatz vor, der diese deterministisch vorhersagbaren Teile entfernt und somit das Eingabematerial absichert. Wir implementieren und analysieren diese Lösung und zeigen ihre Effektivität sowie ihre Resistenz gegen die vorgeschlagenen Angriffe. In einer zweiten Lösung nutzen wir die Fähigkeiten des maschinellen Lernens, indem wir sie ebenfalls in das Systemdesign einbringen. Aufbauend auf ihrer starken Leistung bei der Mustererkennung entwickeln, implementieren und analysieren wir eine Lösung, die lernt, die zufälligen Teile aus den rohen CIRs zu extrahieren, durch die die Kanalreziprozität definiert wird, und alle anderen, deterministischen Teile verwirft. Damit ist nicht nur das Schlüsselmaterial gesichert, sondern gleichzeitig auch der Abgleich des Schlüsselmaterials, da Differenzen zwischen den legitimen Beobachtungen durch die Merkmalsextraktion effizient entfernt werden. Alle vorgestellten Lösungen verzichten komplett auf den Austausch von Informationen zwischen den legitimen Kommunikationspartnern, wodurch der damit verbundene Informationsabfluss sowie Energieverbrauch inhärent vermieden wird.

Contents

Acronyms

ANN	Artificial Neural Network
ASBG	Adaptive Secret Bit Generation
AWGN	Additive Gaussian White Noise
BDR	Bit Disagreement Rate
CCE	Channel Characteristics Estimation
CDF	Cumulative Density Function
CIR	Channel Impulse Response
CNN	Convolutional Neural Network
COTS	Commercial Of-The-Shelf
CRKG	Channel Reciprocity-based Key Generation
CSI	Channel State Information
CTF	Channel Transfer Function
DMS	Discrete Memoryless Source
DoS	Denial of Service
ECC	Error Correction Code
FIR	Finite Impulse Response
GBQ	Guard Band Quantizer
iid.	independent and identically distributed
IIoT	Industrial Internet of Things
IoMT	Internet of Medical Things
IoT	Internet of Things

IQR	interquartile range
IR	Information Reconciliation
KGR	Key Generation Rate
KLT	Karhunen Loèven transformation
kNN	k-Nearest Neigbours
LOS	Line-of-Sight
MI	Mutual Information
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MSE	Mean Squared Error
NIC	Network Interface Card
NIST	National Institute of Standards and Technology
NLOS	Non-Line of Sight
OFDM	Orthogonal Frequency Division Multiplexing
PA	Privacy Amplification
PCA	Principal Component Analysis
PLS	Physical Layer Security
PUF	Physical Unclonable Function
RSSI	Received Signal Strength Indicator
SNR	Signal-to-Noise Ratio
ToF	Time of Flight
UWB	Ultra Wideband
WSN	Wireless Sensor Network

Part I.

Preliminaries

1. Introduction

As the Internet of Things (IoT) and its descendants and evolutions continue to grow, so does the need for adequate protection of these systems and of their communication. Recent attacks and high-impact security incidents impressively demonstrate the importance of protecting such connected environments, the structures built upon them as well as the messages exchanged between and within them.

Since the end devices in these environments are primarily solution specific and, hence, rather simplistic and resource-constrained products, protective measures must be found that take this very nature of the devices into account. In concrete terms, this means that lightweight and resource-efficient solutions have to be applied to protect these infrastructures and their communication.

In this scope, the concepts and approaches of Physical Layer Security (PLS) offer a suitable alternative to classical cryptography, as they provide very lightweight security primitives with proven information-theoretic security. In the area of key generation in particular, these methods rely on simple physical properties such as link quality indicators.

In this work, we investigate the question of how to realize the practical transition from these simple channel properties to the use of the complete information contained in the channel, paying particular attention to the underlying assumptions and the resulting practical security of the methods.

Throughout the remainder of this introduction, we describe the motivation for the work in more detail and specifically formulate the resulting research questions. Finally, we present the achieved contributions of this work as well as outline the structure of the work.

1.1. Motivation and Use Cases

The number of interconnected devices is growing steadily. Ever since the days of Wireless Sensor Networks (WSNs) the notion of “ubiquitous computing” has become, in the most literal sense of the word, ubiquitous. Although the mostly ambitious predictions for the number of connected devices have not been met — for example, in 2012 IBM predicted one trillion devices in 2015 and in 2017 Cisco predicted 25 billion devices in 2020 — the actual number of connected devices is growing steadily, reaching an estimated 17 billion devices in 2016 [150]. This is accompanied by a huge global market share, which size is estimated around 1 to 10 trillion US dollar [217, 63].

Regardless of these numbers, the IoT is still expanding and brought its concepts and ideas to a wide variety of domains, spawning further networks of things like Industrial Internet of Things (IIoT) [28], Internet of Medical Things (IoMT) [66], or the whole area of vehicular connections (V2x) [68, 128]. In parallel, the underlying concepts and approaches are constantly improving and evolving, creating the next generations of networked devices and networks, such as the Tactile Internet [59, 60]. Due to this constant expansion of application areas as well as the continuous technical advances, networked devices are becoming pervasive in every area of our daily lives.

Nevertheless, this persistently growing area comes with its own set of challenges and issues. Since the early days of IoT to the present, the biggest challenge in this area is to reliably ensure security. A considerable number of studies and surveys describe the multifaceted causes and corresponding manifestations of this security problem [99, 62, 217, 7, 116, 203, 202, 91, 68, 101, 128]. The impact of this issue extends well beyond the scientific community and the relevant circle of experts: Various security incidents, whose original attack vectors can be located in the IoT domain, have not only attracted considerable media attention but also caused significant economic damage. Examples range from targeted attacks on industrial facilities such as Stuxnet in 2010 [58, 110] or the German steel mill attack in 2014 [32], to a wide variety of botnets such as Mirai [11] and its various mutations [105], Hajime [55] or the Mikrotik network [35], to straight forward exploits against "smart" devices like doorbells [188], light bulbs [159], web cameras [83], fitness devices [41] or even children's toys [187, 40, 173].

One of the core reasons for this security problem is the very nature of the edge devices in use. Due to the intended use cases as well as the ever-present price pressure of the market, these devices are often strongly optimized for specific applications as well as minimal maintenance. As a result, supporting but equally important topics, like security, are sometimes neglected during product development [62]. In addition, due to their low complexity design, the corresponding devices usually offer only limited computational and energy resources hindering the straight forward application of classical cryptography [217].

With all of this in mind, providing reliable security for such IoT devices requires, besides an increased attention and focus, the use of lightweight, resource-efficient, yet effective security measures.

At this point, PLS presents itself as a promising alternative to classical cryptography to meet all these requirements. The core idea of PLS is to facilitate the properties of the physical environment to provide security primitives and achieve the respective security goals. Due to its origins in the realm of Information Theory, the corresponding primitives and processes are well analysed theoretically and information theoretic security has been proven — given that the respective underlying assumptions are met in practice [238, 26, 114].

PLS approaches are especially applicable in the scope of wireless communication, where the characteristics of the transmission layer shared between two legitimate communication partners can be employed [26, 228, 92]. Here, the physical properties of this shared channel are unique to the specific environment and terminal positions of these legitimate partners. Furthermore, these properties are *reciprocal*, meaning that the observations are approximately identical for both legitimate communication partners. A possible attacker, on the other hand, can only obtain weakly correlated observations of these properties. Additionally, it is assumed that an attacker cannot predict such properties accurately.

Building on this basic concept, these channel properties can be used as input values for security primitives. Specifically since these properties are unique to specific terminal positions and are assumed to be hard to predict by adversaries, they can be used to create a common trust anchor for the legitimate partners by generating a shared secret or symmetric key. This process is known as Channel Reciprocity-based Key Generation (CRKG) [228, 92].

The basic idea of CRKG was already raised in the 90s and is by now well established both theoretically and practically. CRKG builds on a comprehensive theoretical foundation, which proves both its feasibility and information-theoretic security [26].

In terms of practical implementations, most systems rely on link quality indicators such as the Received Signal Strength Indicator (RSSI) value, since these are provided by virtually all wireless interface devices. However, using this channel property as input material for CRKG entails certain drawbacks. The most noticeable is the exclusion of most of the actual properties contained in the physical channel by condensing all of them into a single value. This in turn causes a low effective Key Generation Rate (KGR) of less than a single bit per channel usage.

An alternative strategy here is to use the complete Channel State Information (CSI) and thereby facilitate all available information contained in the physical channel. This approach and its practical feasibility is fuelled by current technical development. Concretely, special firmwares for IEEE 802.11 interface cards have been developed and adapted so that they report the channel characteristics per channel subcarrier and thus provide rich information about the channel. Examples are firmwares developed for Intel [77] and Atheros [214] chips or the *nexmon* firmware patching framework for Braodcom chips [73]. In addition, Ultra Wideband (UWB) chips are being installed in more and more end-user devices, such as the latest flagship models of Apple [94] and Samsung [11] smartphones or in current indoor localization solutions [46]. Through this technology and especially through its high spatial resolution, particularly rich channel information can be provided.

In this work, we explore the practical challenges associated with the transition to CRKG based on complete CSI. Here, we put a special focus on CSI provided in the form of UWB Channel Impulse Responses (CIRs). The specific issues are analysed through use case driven real world measurements, with particular attention to the underlying security assumptions. Based on these analyses, we then design and analyse Channel Impulse Response (CIR)-based CRKG systems that provide inherent security and efficiency by design.

1.2. Research Questions

With this motivation and use cases in mind, we derived the following core research questions for this work. Here, we present the core reasoning for each research question — a detailed background for each can be found in the state of the art analysis in Chapter 3.

1. **Enabling the transition to CIR-based CRKG.**
   Previously, compared to RSSI-based CRKG systems, only a limited number of works have proposed and evaluated practical CRKG system based on CIRs. These works have given little attention to the special form of this input data. However, in order to make comprehensive practical use of the complete channel information contained in

CIRs, these special CIR properties must be considered and exploited. Here we raise the questions of how this can be done in an efficient and secure manner.

2. **Ensure secure design of CIR-based CRKG.**
   Analogously, previous works have placed their primary focus on efficient processing, leaving aside the fundamental security of the respective processes. This is particularly negligent considering that already the initial works in the field of CRKG postulated fundamental attacks such as the *predictable channel attack*. Therefore, we investigate to what extent CRKG and specifically CIRs can be attacked and possibly corrupted by adversaries. Building on such findings, CRKG systems can then be designed to be inherently secure against such attacks.

3. **Increase efficiency through reduced communication overhead.**
   In order to achieve efficient processing, different solution approaches were added to CRKG often requiring additional communication. Within the scope of our use cases, additional communication comes with several drawbacks: it leaks information to an attacker, reduces the achievable secure key rate, and consumes time and energy reducing key rates and energy efficiency. Moreover, it requires an authenticated communication channel between to legitimate partners. Therefore, minimizing the communication throughout the processing would be beneficial for CRKG. Thus, we want to investigate how and to which extent communication efforts can be reduced. The ultimate goal here would be to completely avoid all interaction between the communication partners, which would not only eliminate the disadvantages, but also the basic but very high requirement that an additional, authenticated communication channel is available.

4. **Implications of Machine Learning (ML) primitives used by adversaries.**
   Considering the remarkable results of ML approaches in a variety of different domains, their effectiveness must be conceded. This inevitably raises the question of how ML primitives affect CRKG. From an attacker's perspective, ML simply represents another tool with which CRKG can be attacked. Thus, the question arises to what extent the use of ML gives an adversary advantages and whether they are more potent than "conventional" attacks.

5. **Inclusion of ML primitives into CRKG system design.**
   The questions of the effectiveness of ML in the CRKG context and its implications do not only extend to the attacker part. Using the same rationale as an attacker, system designers of CRKG processes can also incorporate ML primitives into the corresponding systems. With the requirements regarding security and reduced communication in mind, we therefore want to investigate how CIR-based CRKG can be improved using ML.

These questions are addressed within this work.

1.3. Structure and Contributions

By pursuing the research questions outlined above, we were able to make the following contributions:

1. **Real world CIR measurements and data sets.**
   To adequately analyse the research questions with a special focus on the real world applicability, we conducted several extensive measurement studies. The setups and scenarios of these measurements are designed to suit the use cases of CRKG. We conducted different measurement campaigns aligned with the research questions in order to closely examine the specific aspects of each question. The benefits of the individual measurements were combined in a final data set, which was made available to the research community in open access. Thus, the verifiability of our results as well as further research in this area is enabled and supported.

2. **Fundamental analysis of CIRs as input material for CRKG.**
   We analyse the properties of the obtained measurements with respect to PLS and CRKG. Specifically, we analyse the suitability of such measurements as input for key generation. This includes investigation concerning the respective required properties, e.g., reciprocity of the legitimate observations or an advantage over the eavesdropper in the sense of Wyner's wiretap channel [212].

3. **General approaches to facilitate CIRs for CRKG.**
   The initial analysis of the CIR measurement data reveals certain challenges regarding the use of such data in the context of CRKG. To overcome these and remove the errors caused by them, we propose and analyse methods and procedures to basically enable the use of CIR within CRKG.

4. **Security analysis of the key material.**
   In order to work out the design criteria of a secure key generation from scratch, we thoroughly investigate to what extent an attacker can attack the input material, i.e., CIR measurements, and thus the key generation. We investigate different approaches an adversary can utilize to directly infer or predict the key material based on the information available to him. This analysis clearly shows that an attacker can predict large portions of deterministic CIR parts and thus corrupt corresponding security primitives. Based on this, we strongly advise against the direct use of unprocessed CIRs as key material, but recommend preprocessing that removes the deterministic and thus predictable parts of the key material. Only in this way CIRs can be used securely as key material.

5. **Proposal of attack resilient CIR-based solutions.**
   Based on the insights of the security analysis, we propose two different processing pipelines providing secure CIR-based CRKG. One solution is purely preprocessing based and can be used as drop-in solution with existing approaches. A second, ML-based solution provides an "all-in-one" solution for CIR-based CRKG. We show that both approaches are resilient against the presented attacks. Further, all proposed solutions work without communication between the legitimate partners and provide better key generation rates than the current state of the art.

6. **Introduction of machine learning into the adversary and system model.**
   Given the recent success of ML in a wide variety of domains, we introduce ML primitives both in the design of CRKG and as an attack method. Our results show that even basic neural network perform significantly better than their "conventional" counterparts on both the offensive and defensive side. To the best of our knowledge, we are

the first to introduce these ML primitives in the CRKG context, especially as attack approach.

These contributions are also reflected in the publications associated with this thesis, which are included in the author publication list on page 9.

The content of this thesis if organized as follows. In the upcoming Chapter 2 the required background of this thesis and the core methodology is presented. Sections 2.1 to 2.4 describes the concepts of CRKG as well as the adopted channel model and derives a system model for this work. Section 2.5 presents the CRKG specific metrics used. In Section 2.6 we give a high level introduction to machine learning, specifically neural networks. Chapter 3 presents the state of the art regarding CRKG as well as CIR-based CRKG and highlights the open research questions.

Chapter 4 details the measurement campaigns conducted in the scope of this work. Correspondingly, Chapter 5 details the analysis of the obtained CIR measurements.

In Part III we present the general approaches to CIR-based CRKG. This includes the fundamental design decisions made for all solutions presented in this work.

The Part IV provides details about our security analysis. Within, Chapter 8 shows "conventional" attack approaches and their results, whereas Chapter 9 introduces ML for the attacker and shows the respective results.

Based on these results, Part V proposes solutions for CIR-based CRKG resilient against these attacks. In correspondence to the former part, an approach without ML is presented in Chapter 10 and one with ML in Chapter 11. This includes the analyses of the respective attack resilience. Chapter 12 embeds these solutions in the greater context of CRKG.

Finally, Part VI summarizes the results of this thesis and outlines possible directions for future research as well as research questions that arise directly from the results.

Publication list

Paul Walther, Carsten Janda, Elke Franz, Mathias Pelka, Horst Hellbrück, Thorsten Strufe, and Eduard Jorswieck. "Improving Quantization for Channel Reciprocity Based Key Generation". In: *2018 IEEE 43rd Conference on Local Computer Networks (LCN)*. IEEE, 2018, pp. 545–552

Paul Walther, Elke Franz, and Thorsten Strufe. "Blind Synchronization of Channel Impulse Responses for Channel Reciprocity-Based Key Generation". In: *2019 IEEE 44th Conference on Local Computer Networks (LCN)*. IEEE, 2019, pp. 76–83

Paul Walther, Robert Knauer, and Thorsten Strufe. *Passive Angriffe auf kanalbasierten Schlüsselaustausch*. Gesellschaft für Informatik e.V., 2020. ISBN: 978-3-88579-695-4. DOI: 10.18420/sicherheit2020_03

Paul Walther and Thorsten Strufe. "Blind Twins: Siamese Networks for Non-Interactive Information Reconciliation". In: *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE, 2020, pp. 1–7

Best paper award

Paul Walther and Thorsten Strufe. "Machine Learning Aided Inference Attacks on Channel State Information". In: *2020 IEEE 19th International Conference On Trust, Security And Privacy In Computing And Communications (TrustCom)*. IEEE, 2020

Paul Walther, Markus Richter, and Thorsten Strufe. "Ray-Tracing Based Inference Attacks on Physical Layer Security". In: *2021 International Conference on Networked Systems (NetSys)*. 2021, pp. 1–4

Best paper award

Paul Walther, Robert Knauer, and Thorsten Strufe. *Ultra-Wideband Channel State Information and Localization for Physical Layer Security*. Feb. 2021. DOI: 10.21227/0WEJ-BC28

Further publications

Paul Walther, Stefan Köpsell, Frederick Armknecht, Gene Tsudik, and Thorsten Strufe. "Chains and Whips-An Approach to Lighweight MACs". In: *crypto day matters 28* (2018). Gesellschaft für Informatik eV/FG KRYPTO

Frederik Armknecht, Paul Walther, Gene Tsudik, Martin Beck, and Thorsten Strufe. "Pro-MACs: Progressive and Resynchronizing MACs for Continuous Efficient Authentication of Message Streams". In: *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 2020, pp. 211–223

The two first authors contributed equally to this work.

Sadia Moriam, Elke Franz, Paul Walther, Akash Kumar, Thorsten Strufe, and Gerhard Fettweis. "Protecting Communication in Many-Core Systems against Active Attackers". In: *Pro-*

ceedings of the 2018 on Great Lakes Symposium on VLSI. 2018, pp. 45–50

Julian Harttung, Elke Franz, Sadia Moriam, and Paul Walther. "Lightweight Authenticated Encryption for Network-on-Chip Communications". In: *Proceedings of the 2019 on Great Lakes Symposium on VLSI*. 2019, pp. 33–38

Sadia Moriam, Elke Franz, Paul Walther, Akash Kumar, Thorsten Strufe, and Gerhard Fettweis. "Efficient Communication Protection of Many-Core Systems against Active Attackers". In: *Electronics* 10.3 (2021). Multidisciplinary Digital Publishing Institute, p. 238

Nirav Annavarapu, Mircea Catuneanu, Paul Walther, Mohsen Razavi, Elke Franz, Thorsten Strufe, and Kambiz Jamshidi. "Numerical Modeling of Dual Pump Amplifiers in Standard Silicon-on-Insulator Platform for Random Number Generation". In: *Frontiers in Optics*. Optical Society of America, 2020, JTh4B–16

2. Background and Methodology

This chapter introduces the necessary background concepts on which this work is based as well as the applied methodology.

The main part is a high level presentation of Physical Layer Security and its application in Sec. 2.1, followed by a more detailed introduction to Channel Reciprocity-based Key Generation, the main topic of this work, in Sec. 2.2. Subsequently, we take a step back again and discuss the electrical engineering fundamentals of wireless transmission that are relevant to the work. These introductory chapters are then integrated in Sec. 2.4 and a system model is established, which is applied in the upcoming measurement campaigns as well as solution approaches. Based on this system model we describe the metrics used throughout the evaluation process. In addition, an overview of basic machine learning subjects that are relevant to some of the proposed attacks and solutions is presented in Sec. 2.6.

The Fig. 2.1 shows the dependencies between the individual sections of this chapter.

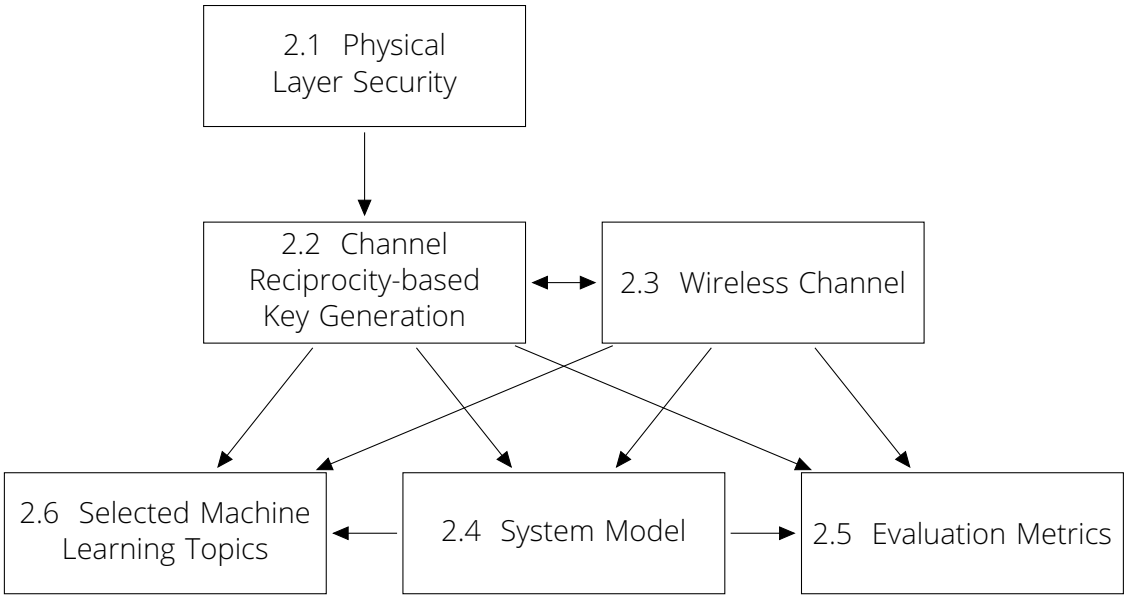
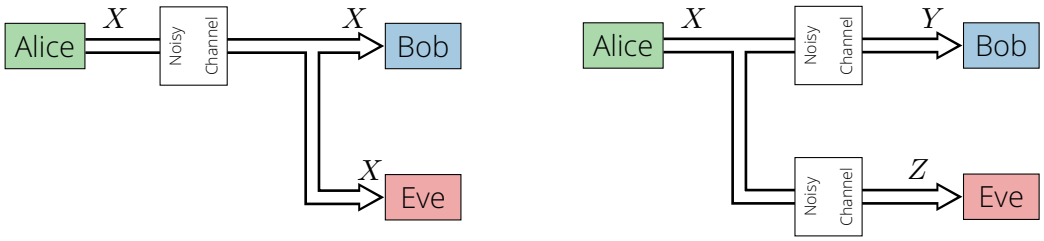


Figure 2.1.: Overview of the background chapter and the relations between its sections.

2.1. Physical Layer Security

The beginnings of Physical Layer Security (PLS) can be traced to Claude Shannon's definition of information theoretic security [171]. In reference to the key used in the cryptosystem, Shannon defined a transmission model for secure communication. The basic idea of this can be seen in Fig. 2.2a.



(a) Shannon's channel model (b) Wyner's channel model

Figure 2.2.: The channel models as developed by Shannon and Wyner [171, 212].

Based on this, Wyner defined the so-called "wiretap channel" which is shown in Fig. 2.2b [212]. Here a distinction is made between the channel used by Alice and Bob, and the channel between Alice and Eve: that is, there is an inherent difference between these two transmissions. In general, Alice and Bob are assumed to have a "better" transmission channel than Eve, since Eve's observation Z is *degraded* relative to Y. Thus, the legitimate communicators have an advantage over the attacker. In the context of wireless transmission, this advantage may be, e.g., a better Signal-to-Noise Ratio (SNR). This basic property of the degraded channel was confirmed in later works [43].

Wyner posited that these better characteristics of the legitimate channel can be exploited to conceal information from an attacker and thus achieve a secure transmission. The basic functionality of this approach was successfully demonstrated analytically. Furthermore, it underwent rigorous security analyses and its information-theoretic security could be proven.

This core idea that there is a physical difference between the legitimate communication partners and the attacker was then used in a variety of ways for security purposes. In the realm of wireless communication the different physical propagation properties, e.g., fading, interferences or spatial diversity, are exploited to achieve different security goals. Examples are Wyner's wiretap codes achieving confidentiality [212] or Simmon's identification schemes providing authentication [176, 175].

In addition to the application in the field of wireless communication, other concepts based on physical differences are now considered part of the PLS domain. Examples of this are Physical Unclonable Functions (PUFs) or biometric data.

The concepts of PLS and the approaches developed from them can now provide a variety of security-critical applications and solutions. A selection is shown in Fig. 2.3. In this work, we are primarily concerned with confidentiality, specifically secure key derivation based on wireless physical properties, as well as certain aspects of wireless identification.

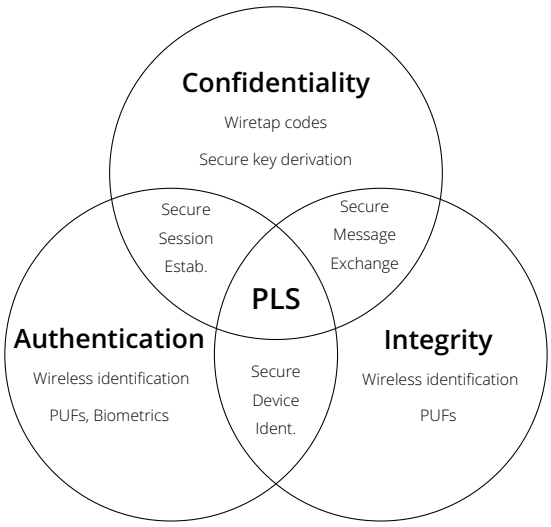


Figure 2.3.: Possible applications of Physical Layer Security.

2.2. Channel Reciprocity-based Key Generation

In this section we introduce the basics of the key derivation scheme *CRKG*. The information theoretic perspective onto this topic will be prevalent. Practical considerations, e.g., channel properties and reciprocity, is presented in Section 2.3 and 2.4. From the information theoretic point of view, CRKG is an implementation of the *source model* for key derivation, combined with a *sequential key derivation*. Both concepts are presented next.

It should be noted that CRKG is also referred to by other names, mostly concerning the last part — examples are *Channel Reciprocity based Key Agreement* [92] or *Channel Reciprocity based Key Extraction* [230, 229, 231, 228].

2.2.1. Source Model for Key Derivation

The source model for secret sharing was independently conceptualized by Maurer [134] as well as Ahlswede and Csiszar [1] in 1993. Both works addressed the problem, how two legitimate parties can derive a common shared secret from correlated observations of a randomness source, while a third party tried to infer the derived secret from its own correlated observations. The core of the source model is a common shared randomness source P_{XYZ} from which all participants receive realizations. Both original works assume this source to be Discrete Memoryless Source (DMS), which implies that the respective observations are independent and identically distributed (iid.). In the general description, the actual physical process acting as randomness source is completely abstracted away. This in turn means that no assumptions about the respective correlations between the observations are made.

To realize the further processing throughout the sequential key derivation, the source model also includes a public channel over which the legitimate partners perform the communication, which is assumed necessary for the key derivation. The set of messages exchanged during the key exchange is denoted as M_{ex}. It is assumed that the eavesdropper has full listening access to this channel and, in turn, to M_{ex}. Additionally, it is assumed that this channel is authenticated, i.e., the eavesdropper cannot modify the protocol messages

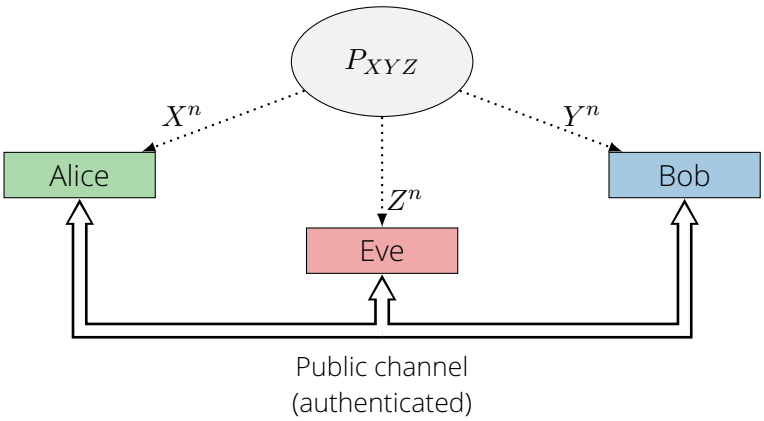


Figure 2.4.: The source model of key derivation from common randomness P_{XYZ}. Adapted from Fig. 4.1 [26].

without detection. This assumption guarantees that the key derivation protocol can be executed correctly by the legitimate partners.

Figure 2.4 depicts the *source model* [26, adapted from Fig. 4.1]. The name *source model* originates from the common source of randomness providing its realization to all participants. By convention the legitimate partners are Alice A and Bob B, receiving the respective observations X and Y, and the eavesdropper is Eve E with observation Z. X, Y and Z sometimes have a superior letter n denoting the length of the respective observations.

Maurer [134] as well as Ahlswede and Csiszar [1] analysed the source model for key derivation from the information theoretic point of view and thereby derived lower and upper bounds as well as fundamental requirements of this key derivation. Most relevant for this work are the derived lower and upper bounds for the achievable secret key capacity C_s as theoretic bounds for the key generation rate:

$$\mathbb{I}(X;Y) - \min(\mathbb{I}(X;Z), \mathbb{I}(Y;Z)) \leq C_s \leq \min(\mathbb{I}(X;Y), \mathbb{I}(X,Y|Z)) \tag{2.1}$$

Here, $\mathbb{I}(\cdot,\cdot)$ represents the Mutual Information (MI).

Additionally, both works derive security requirements for the keys derived, namely *reliability*, *secrecy* and *uniformity*:

Reliability

$$Pr(K_A \neq K_B) < \varepsilon \tag{2.2}$$

Reliability states that the keys derived by Alice and Bob, i.e., K_A and K_B, have a low probability of being unequal. This means, the legitimate partners should derive the same key.

Secrecy

$$\frac{1}{n}\mathbb{I}(K; M_{ex}) < \varepsilon \tag{2.3}$$

Secrecy requires that only a negligible amount of information is leaked towards the attacker. Given the resulting key K and the set of exchanged messages M_{ex}, the

mutual information between those should be small.
Here, the notion of *weak* secrecy is shown, which is normalized over the length of the resulting key n; without this normalization, *strong* secrecy is required.

Uniformity

$$\frac{1}{n}H(K) \geq \frac{1}{n}\log_2 |\mathcal{K}| - \varepsilon \tag{2.4}$$

Uniformity requires that the derived keys are uniformly distributed in the key space $\mathcal{K}$, i.e., every possible key has the same occurrence probability. Again, the equation shows the *weak* uniformity normalized by the key length n, whereas the *strong* uniformity is not normalized.

Although those requirements are explicitly formulated in the context of this key derivation, they probably apply to all key derivations due to their fundamental nature.

Independently of these information theoretic approaches, Hershey et al. proposed the usage of the channel between two wireless transceivers as keying variable [84]. This proposal combines the following two observations: first, the wireless transmission between two transceivers can be considered fixed for a certain amount of time (cf. Section 2.3) and additionally, the observations of the channel properties at those two transceiver are highly correlated due to *channel* and *antenna reciprocity*. Second, due to the scattering of the multipath propagation it is very hard for an attacker to measure the same channel properties, which is commonly denoted as *spatial decorrelation*.

Within the reciprocity observation, antenna reciprocity is based on the radiation characteristics of the respective antennas, which are reciprocal according to the Lorentz Reciprocity Theorem directly derived from Maxwell's equations [17]; channel reciprocity, on the other hand, results from the characteristics of wireless wave propagation, as described in [95].

The effect of spatial decorrelation is founded in *Uniform Scattering* model described by Jakes [95], At its core, this states that with increasing spatial distance the correlation of observed CIRs decreases. This concept is described in detail in Section 2.3.

This proposal connects the abstract source model with a real physical phenomenon. This connection allows the qualitative estimation of the correlations between the respective observations of the common source: In a loose analogy to the wiretap channel, the observations of the legitimate partners Alice and Bob, X and Y, are considered highly correlated, whereas the observations of an eavesdropper Eve Z is assumed to be less correlated. Hence, Alice and Bob can consider their observations a shared secret and use them to generate a common key.

2.2.2. Sequential Key Derivation

The source model alone describes the setup only regarding the common randomness source and the public communication channel. After observing their respective realizations, the legitimate partners still have to derive a shared key from it without disclosing information to the attacker.

Following the explanations in [26, Chapter 4.3], this derivation is realized by performing the following four steps at the legitimate partners Alice and Bob: *Randomness Sharing*,

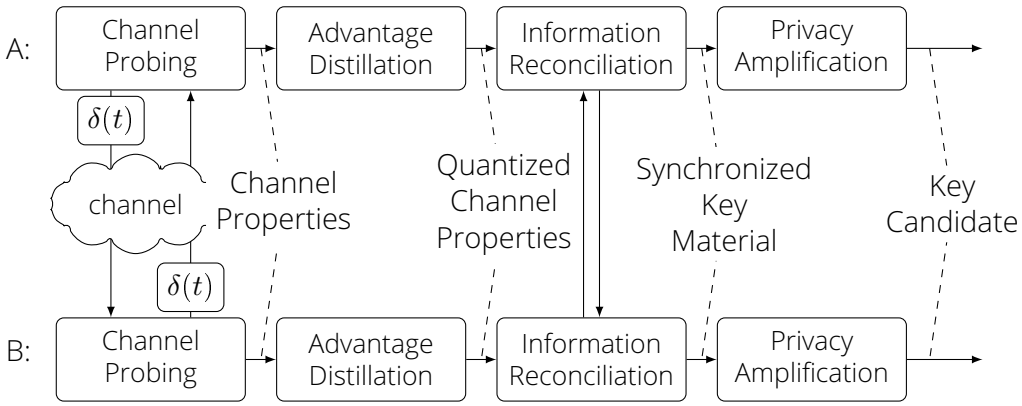


Figure 2.5.: Processing steps of sequential key derivation at the legitimate communication partners A and B as proposed in theory.
Adapted from Fig. 2.7 [228].

followed by *Advantage Distillation* and *Information Reconciliation*, and finally *Privacy Amplification*. Some of these steps might require information exchange between the legitimate partners, which is then performed over the authenticated channel assumed by the *source model*. The general progression of these steps at Alice and Bob is depicted in Fig. 2.5 [228, Adapted from Fig. 2.7]. This describes the key derivation in the context of CRKG, i.e., the *Randomness Sharing* is denoted as *Channel Probing*.

The initial step of the key derivation is **Randomness Sharing**. Here, all parties observe their respective realization of the shared randomness source P_{XYZ}. As the processing itself as well as the source model do not assume any particular randomness source, the acquisition of said realizations cannot be further specified. In the context of CRKG, the randomness sharing is realized either explicitly, through bidirectional exchange of suitable probing messages, or implicitly, by using channel estimations and alike of regular traffic. Thus, for CRKG, randomness sharing is also commonly termed *channel probing*. In the general model, the only assumption about the Randomness Sharing is the DMS nature of the source. Since the source itself is unspecified, no further assumptions prevail.

During **Advantage Distillation** the legitimate partners generate an advantage over the eavesdropper. As no assumption about the respective realization is made, it is possible that an eavesdropper has "more information" than the legitimate partners about the keying material. Hence, Alice and Bob have to use their observations to generate an advantage over the attacker. One possibility to achieve this, is to use only those values for key derivation, which were obtained with high confidence, e.g., with high Signal-to-Noise Ratio (SNR) [26]. Depending on the actual implementation, this selection of a subset of the measurement values might need to be aligned between Alice and Bob, implying optional communication about the selected key material. After this step, the legitimate partners will have highly correlated key material, about which they "know more" than the eavesdropper.

Subsequently, **Information Reconciliation (IR)** is performed to transform these highly correlated values into equal ones. In this step Alice and Bob exchange further information about the current key material, which allows them to remove any difference between their key candidates. Common approaches for this are the usage of parity information, Error Correction Code (ECC) or secure sketches/fuzzy commitments. It is important to notice, that such solutions are very dependent on the actual key material and the respective differences. Since, by design, within this step information about the key material is commu-

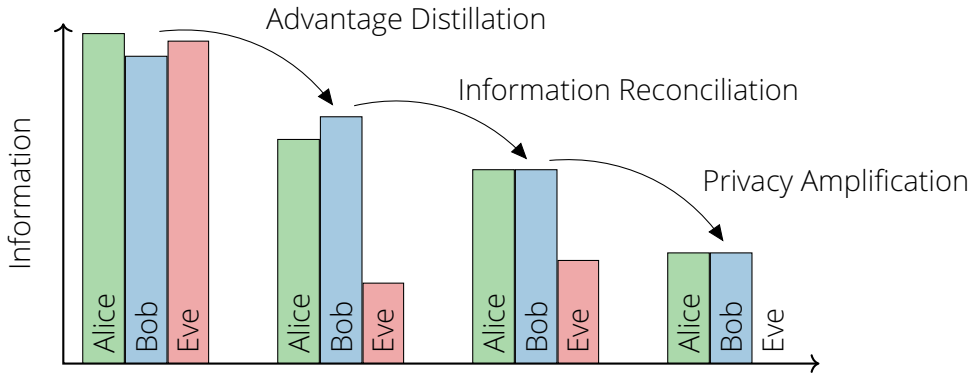


Figure 2.6.: Qualitative representation of the information about the key material at the different participant during sequential key derivation. Adapted from Fig. 4.4 [26].

nicated over the public channel, this information is directly leaked to the attacker. Hence, it needs to be carefully considered, which information are disclosed and whether the attacker can gain advantage from this additional knowledge. After reconciliation, Alice and Bob will have completely equal key candidates and Eve will have gained some information about it.

This knowledge at Eve is completely equalized in the final step, **Privacy Amplification**, in which the key candidate is processed to remove any possible advantage of the attacker. The approach proposed in theory is to remove exactly as much information from the respective key candidate as can have been leaked to the attacker up to that point. One possible realization is the usage of *Universal Hash Functions*, where the respective output length is chosen in such a way, that the difference between input and output contains as much entropy/information as were leaked to Eve. With this reduction, Privacy Amplification effectively reduces the possible knowledge of Eve to zero and thereby guarantees the security of the key candidate. Nevertheless, for practical implementations this step is challenging, since the information leakage to the attacker is unknown. In order to better estimate this parameter, the usage of the min-entropy or the collision-entropy were proposed [26]. Nevertheless, in practice the actual values are not available at runtime making this step, which is crucial for the keys security, very challenging.

A qualitative representation of the knowledge or information about the respective key material is visualized in Fig. 2.6 [26, adapted from Fig. 4.4]. It depicts the key points of the sequential key derivation: the randomness sharing does not assume a direct advantage, i.e., Eve might know more than Alice or Bob. Only after Advantage Distillation Alice *and* Bob have more information than Eve. This knowledge becomes equal by Information Reconciliation and finally Privacy Amplification reduces Eve's knowledge to zero.

The combination of the *source model* and the respective key derivation allows the legitimate partners to derive a shared key. Nevertheless, the associated proofs and models come with certain assumptions, which need to be considered when realizing key derivations based on those. The following core assumptions have to be considered:

Randomness of source The *source model* itself assumes a common source of randomness P_{XYZ} as input for the key derivation. Although this assumption might seem trivial, this source P should generate truly random realizations. If the source is not

random, i.e., it produces predictable events, the resulting keys have to be considered predictable as well, which results in an insecure key exchange.

In the context of CRKG, this means that regardless of the specific channel properties used, the Alice-Bob channel itself must undergo variations in order for new true random alterations to be induced for key generation.

Independence of consecutive realizations The proofs concerning the common source of randomness collectively assume a *Discrete Memoryless Source* for P_{XYZ}. This means that subsequent realizations, i.e., the measurements of the channel, need to be independent and identically distributed. If such realizations are produced, e.g., by the respective statistical channel models, this identical distribution might be given.

However, in practical CRKG settings the consecutive measurements are mainly influenced by the respective movements of the terminals, which represents a continuous movement with respect to space and time. Hence, the independence of the individual measurements cannot be guaranteed. On the contrary, a certain correlation must be assumed, which, in turn, must be removed or at least adequately reduced by appropriate measures.

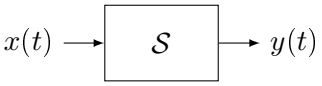
Discrete realizations Again originating from the assumption of a DMS, the realizations are assumed to be discrete. Actual channel properties are in fact continuous values. An initial quantization with high resolution is performed throughout the measurement of the channel properties. Extended models, like the *Satellite Source Model* even assume the source observations to be binary values. In the general model, at some point of the processing a mapping into the target alphabet $[0, 1]$ must be performed.

2.3. Wireless Channel

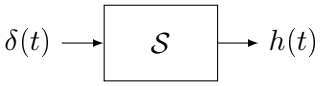
Within this section the fundamental ideas of wireless transmission in the scope of CRKG are described. First, the wireless transmission itself as viewed from the system theoretic point of view is introduced. Then, those properties of the wireless channel are detailed, which are especially relevant for the CRKG system model, e.g., multipath propagation, spatial decorrelation and channel coherence. Finally, the channel properties usable in CRKG are described.

2.3.1. Wireless Transmission

In the following we describe the basic process of wireless transmission and the respective adopted assumptions. The most notable difference to classical wireless communication is the focus on the channel itself, instead of the transmitted signals: The main purpose of wireless signals in the context of CRKG is not the transmission of information, but the collection of keying material, i.e., properties of the wireless channel.



(a) General linear system.



(b) Exciting the linear system using the Dirac impulse yields the systems *impulse response*.

Figure 2.7.: Wireless transmission as linear system $\mathcal{S}$.

From a system theoretic point of view, the wireless transmission of signals is the mapping of an input signal to an output signal through the wireless system. Generally speaking, wireless transmission transforms the input signal $x(t)$ through the system $\mathcal{S}$ into the output signal $y(t)$. This transformation is visualized in Fig. 2.7 and can be represented as

$$y(t) = h(t) * x(t) + n(t). \tag{2.5}$$

Here, $h(t)$ stands for the systems impulse response which represents alterations caused by the wireless transmission, i.e., the wireless propagation and the transmitter/receiver hardware. The operator $*$ denotes the convolution. $n(t)$ represents the noise present in the system, which additionally changes the signal. Within this work, we assume $n(t)$ to be *Additive Gaussian White Noise (AWGN)*. Further it should be noted, that we represent the wireless transmission in the respective *equivalent lowpass representation*, to remove the carrier frequency from the analysis.

For the actual wireless transmission system we assume multipath propagation of the wireless signal as defined in [69]. As visualized in the Fig. 2.8 the multipath propagation causes different versions of the original signal to arrive delayed and altered by their respective propagation path. Distortions along this path could be reflections and scattering, diffractions like knife edge diffraction or even refraction in the troposphere. These distortions cause power loss and time delay. Hence, the finally received signal $y(t)$ consists of the different time delayed copies of the original signal. This can be expressed as sum over

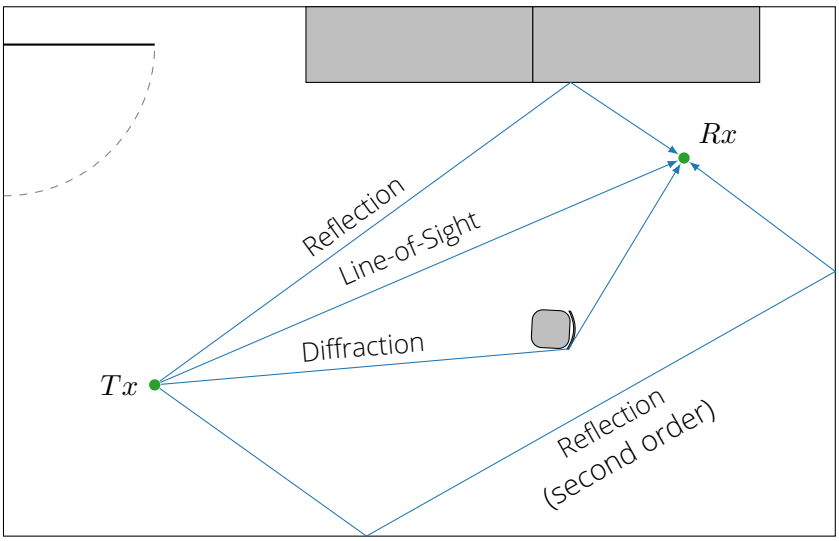


Figure 2.8.: Example indoor multipath propagation of a wireless signal from transmitter Tx to receiver Rx.

those different multipath components (noise is omitted for brevity):

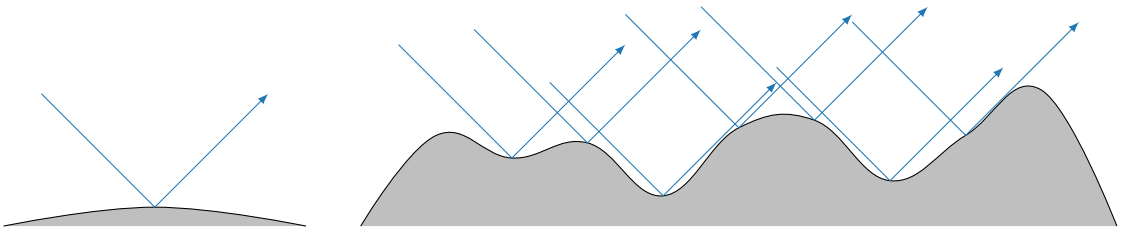
$$y(t) = \sum_{n=0}^{N} \alpha_n e^{j\phi_n} x(t - \tau_n) \tag{2.6}$$

Here, $x(t - \tau_n)$ is the n^{th} copy of the original signal $x(t)$ arriving after time delay τ_n, i.e., the n^{th} multipath component. This component is distorted by the respective multipath propagation. More specific, its power might be reduced and its phase shifted compared to the original signal, which is represented by the properties α_n and ϕ_n.

This description of the received signal $y(t)$ contains the input signal $x(t)$ of the wireless transmission as well as an inherent representation of the wireless channel $h(t)$. In typical wireless communication systems, the aim is to reconstruct the sent signal as accurate as possible. With CRKG, the aim is to estimate the channel properties $h(t)$ as accurate as possible. To estimate the channel as accurately as possible, a globally known input signal is used, which means that the expected output signal is also known. In theory, this known input signal would be the *Dirac impulse* δ. Since the Dirac impulse is a theoretical construct, in praxis a known signal suitable for the respective communication scheme is used. By exciting the wireless system with the known signal and subsequently removing this known input from the received signal, an estimation of the *Channel Impulse Response* can be obtained. As shown in Fig. 2.7b, this changes Eq. (2.6) to:

$$h(t) = \sum_{n=0}^{N} \alpha_n e^{j\phi_n} \delta(t - \tau_n) \tag{2.7}$$

It is worth noting, that this representation assumes a time-invariant channel. This means, that the number of multipath components N as well as amplitude α_n, phase ϕ_n and delay τ_n of the multipath clusters are assumed to be static over time. In praxis, these properties do change over time due to e.g., mobility of the terminals, changes in the environment or interferences. Moreover, following the requirement of fresh keying material, such time-variant channels are explicitly required in order to generate non-predictable key material.



(a) Point scatterer (b) Distributed scatterer

Figure 2.9.: Schematic differences between ideal reflector/point scatterer and a reflector cluster/distributed scatterer.

Hence, in spite of the notation used above, this work generally assumes time-variant channels.

The estimation of the channel properties as described in Eq. (2.7) are the core of the subsequent processing schemes, since it yields the input data used in the key derivation. The estimations obtained by the participants of the key derivation scheme are the realizations of the common source of randomness described in Section 2.2.

Multipath clusters

The preceding description of multipath propagation only considers single specular reflectors, which produce single resolvable reflections. This originates from the idealizations of a single ray hitting a perfect reflector. In practice, this assumption does not hold, mainly for the following two reasons: first, the reflector does not have a perfectly planar surface, but its surface structure consists of irregular unevenness. And second, the radio wave propagates not as single ray, but as a radial wave front. By integrating these two aspects into the reflection, the single reflector, or point scatterer, becomes a reflector cluster, or distributed scatterer, as shown in Fig. 2.9.

This differentiation is relevant, if the single rays from a reflector cluster are resolvable. In general, two multipath components are resolvable, if the respective delay between them is significantly larger than the inverse channel bandwidth [69]. So, two components with delay τ_i and τ_j are resolvable, if it holds that

$$|\tau_i - \tau_j| \gg \frac{1}{B}. \tag{2.8}$$

In this formula, B represents the channel bandwidth. If this requirement is not fulfilled, the respective components are non-resolvable, since $x(t-\tau_i) \approx x(t-\tau_j)$ in Eq. (2.6). This means, that those components are "merged" together and appear as one. Such non-resolvable components are varying quickly compared to resolvable ones, due to the combined interferences of the underlying components. However, if the requirement in Eq. (2.8) is fulfilled, the respective multipath component cannot be considered a single ray, but becomes a multipath cluster consisting of several rays.

As an example, if we consider an UWB transmission with $500\,\mathrm{MHz}$, the inverse of the bandwidth is $\frac{1}{B} = \frac{1}{500\,\mathrm{MHz}} = 2\,\mathrm{ns}$. So, as long as the delay between multipath components is longer than $2\,\mathrm{ns}$, they are resolvable with this channel. Given an appropriate sampling frequency, such multipath components can be recorded accordingly.

This transition from single reflectors to reflector clusters is also expressed in the statisti-

cal modelling of the corresponding channels. Models for wideband channels, which come with bandwidths capable of resolving different rays of clusters, incorporate this cluster/ray distinction.

In the realm of UWB, the channel model proposed by Saleh and Valenzuela [163] was the first indoor model to include the cluster nature of ray arrival. This model was later extended into the UWB reference model for indoor propagation by the IEEE 802.15.3 Working Group for Wireless Personal Area Networks [144].

The core of the Saleh-Valenzuela (S-V) model is the extension of the CIR as defined in Eq. (2.7) into a double sum to express the clustered arrival of the rays. This adaption yields to the following definition of the CIR [163]:

$$h(\tau) = \sum_{l=0}^{L} \sum_{k=0}^{K_l} \alpha_{k,l} e^{j\phi_{k,l}} \delta(\tau - T_l - \tau_{k,l}) \tag{2.9}$$

This is again the time-invariant representation of the channel, for time variations the parameters L, K_l, $\alpha_{k,l}$, $\phi_{k,l}$, T_l and $\tau_{k,l}$ would become time-dependent. Within this equation, L represents the number of multipath clusters, whereas K_l represents the rays within the respective cluster. Further, T_l is the cluster arrival time, i.e., the arrival time of the first ray of the l^{th} cluster; $\tau_{k,l}$ is the respective delay of the k^{th} ray in the l^{th} cluster.

Additionally, and for CRKG more importantly, the S-V model also defines a general progression of the received ray magnitudes and their respective arrival times. For this, the following additional variables are needed:

Λ cluster arrival rate

λ ray arrival rate, i.e., the arrival rate within a cluster

Γ cluster decay factor

γ ray decay factor

With this, the arrival times of clusters and rays are defined [163]:

$$P(T_l|T_{l-1}) = \Lambda e^{-\Lambda(T_l - T_{l-1})} \tag{2.10}$$

$$P(\tau_{k,l}|\tau_{k-1,l}) = \lambda e^{-\lambda(\tau_{k,l} - \tau_{k-1,l})} \tag{2.11}$$

And further, the magnitudes of the arriving rays can be defined depending on the first arrived ray [163]:

$$\alpha_{k,l}^2 = \alpha_{0,0}^2 \cdot e^{\frac{-T_l}{\Gamma}} e^{\frac{-\tau_{k,l}}{\gamma}} \tag{2.12}$$

By combining these definitions, this channel model describes the basic shape and course of a channel impulse response of UWB channels. A schematic representation thereof is depicted in Fig. 2.10. It clearly demonstrates the double exponential decay of the respective ray power: first, the clusters arrive with exponentially decaying power; and second, within each cluster, the single rays again have exponentially decaying power.

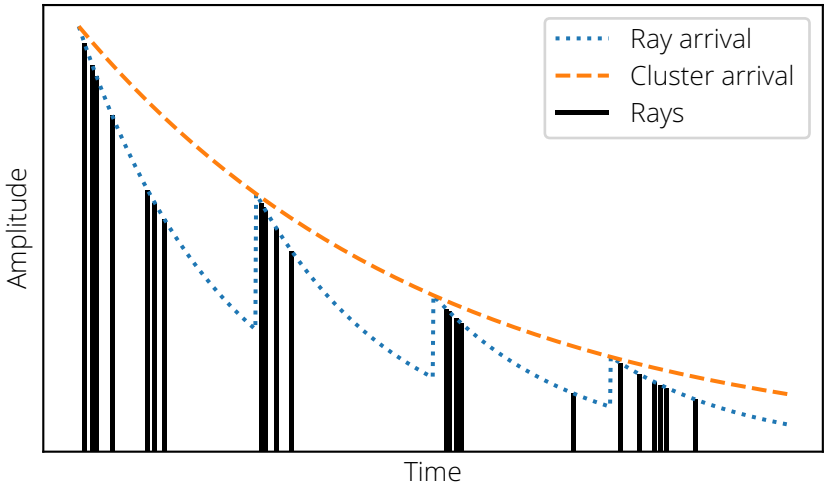


Figure 2.10.: General progression of a CIR as described by the Saleh-Valenzuela model.

Spatial Decorrelation

Spatial decorrelation is the core security assumption of CRKG. The general system and adversary model states, that the channel coefficients of the reciprocal channel are considered to be a shared secret used to derive a symmetric key. Hence, to have a secure key exchange, the attacker should not have access to this shared secret. In terms of the wireless transmission, this corresponds to observing uncorrelated channel coefficients during the channel measurement. Since the concrete point in time of the channel measurement is known to any eavesdropper due to the broadcast nature of wireless communication, this decorrelation can only be achieved by a spatial displacement of the attacker.

In the literature concerning PLS it is generally assumed that with spatial distances larger than half a wavelength of the carrier frequency an eavesdropper can only observe uncorrelated channel coefficients. This assumption is based on the *uniform scattering model* developed by Jakes [95].

By following the descriptions of Goldsmith [69, Chapter 3], this decorrelation argument can be derived as follows. First, we denote that the autocorrelation A for the in-phase and quadrate components for a given time offset τ are each described by

$$A(\tau) = \frac{1}{2} \sum_{n} \mathbf{E}[\alpha_n] \cos(2\pi v \tau \cos \theta_n / \lambda), \tag{2.13}$$

where v is the movement speed, θ_n is the angle of arrival of the respective scattered multipath cluster and λ is the wavelength. Now, by adopting the *uniform scattering model* it is assumed, that N scatterer are distributed uniformly and densely over all possible angles of arrival. It is further assumed, that all multipath components have the same share of the total received power P_r, i.e., $\mathbf{E}[\alpha_n^2] = \frac{2P_r}{N}$, and due to the uniform distribution the respective

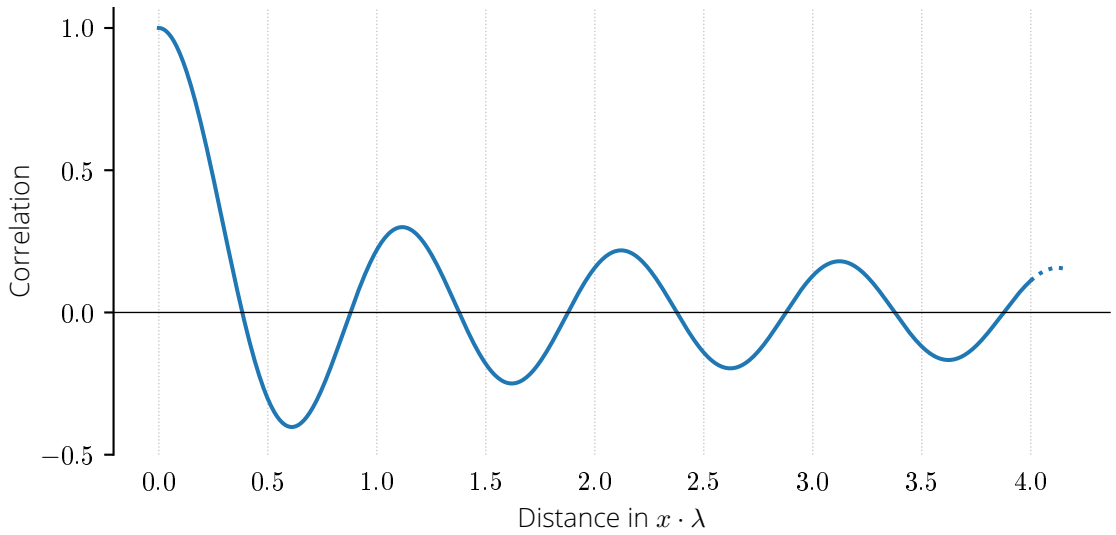


Figure 2.11.: First order Bessel function, which depicts the relation between expected correlation and distance (in multiples of wavelength λ) following *Uniform Scattering* as derive by Jakes [95].

angle offsets are $\Delta\theta = \frac{2\pi}{N}$. Thereby, the autocorrelation becomes

$$A(\tau) = \frac{P_r}{N} \sum_{n}^{N} \cos(2\pi v\tau \cos n\theta/\lambda) \tag{2.14}$$

$$= \frac{P_r}{2\pi} \sum_{n}^{N} \cos(2\pi v\tau \cos n\theta/\lambda)\Delta\theta \tag{2.15}$$

To represent the uniform scattering from all direction, the number of scatterers tends to infinity $N \to \infty$, whereas $\Delta\theta \to 0$. Hence, the summation becomes the following integral:

$$A(\tau) = \frac{P_r}{2\pi} \int \cos(2\pi v\tau \cos\theta/\lambda)\, d\theta \tag{2.16}$$

By substituting the integral with the Bessel function of 0^{th} order, $J_0(x)$, we finally have

$$A(\tau) = P_r J_0(2\pi v\tau \cos\theta/\lambda). \tag{2.17}$$

This states, that the correlation as function of the displacement $v\tau$ follows the progression of the Bessel function of 0^{th} order $J_0(x)$, which is shown in Fig. 2.11.

It is visible in the plot that the correlation reaches 0 at $v\tau \approx 0.383\lambda$, which is the basis for the decorrelation argument. In the literature, this value is occasionally rounded up to 0.4λ whereas in the vast majority of cases "distances greater than half a wavelength", i.e., 0.5λ, is reported.

Since this derivation is the basis for the security argument regarding spatial decorrelation, it is reasonable to consider the corresponding assumptions of the individual steps as described above:

- The propagation model assumes that there is no dominant Line-of-Sight (LOS) component in the impulse response.

- The scatterers are densely and uniformly distributed over all possible angles of arrival.
- All multipath clusters have the same power.
- The phase offset ϕ_n of the multipath clusters is uniformly distributed over $[-\pi, \pi]$, due to rapid changes.

Only if all these underlying assumptions are fulfilled, the derived security assumption of the spatial decorrelation can be accepted in good conscience.

Finally, it is noteworthy, that with the further progression of the Bessel function the correlation again increases. Hence, although the correlation reaches zero at $\approx 0.383\lambda$, it again rises to values $\neq 0$ with increasing distance. Therefore, a correlation of zero for distances $> 0.383\lambda$ cannot be assumed.

Channel Coherence and Sampling Interval

Considering the assumed channel model, there is a fundamental discrepancy regarding the time variance of the channel: with respect to the realizations of the reciprocal measurements, the channel is assumed to be quasi time invariant, so that the channel symmetry is fulfilled. Contrary to this, the requirement of fresh entropy for the key material requires the channel to change to generate new input. Both requirements can be met simultaneously if the reciprocal measurements are taken within the *coherence time* of the channel whereas consecutive measurements are far apart in time.

In a real world scenario, the channel is always time-variant. Due to interference, noise as well as movement of the terminals and of other surrounding objects, the channel coefficients change continuously over time (see Eq. (2.7)). Nevertheless, if the considered time interval is small enough, the respective changes becomes smaller as well — up to the point where the coefficients can be considered to be static. The time interval at which this is true is the *coherence time* T_{coh}, which at a general scope can be expressed as:

$$h(t) \approx h(t+\tau) \qquad ,\forall \tau < T_{coh} \tag{2.18}$$

$$h_{AB}(t) \approx h_{BA}(t+\tau) \qquad ,\forall \tau < T_{coh} \tag{2.19}$$

The second form incorporates the CRKG system setup, in which the legitimate partners acquire their channel estimates h_{AB} and h_{BA} in a half-duplex fashion. This means, h_{AB} represent the estimate recorded by B after the input from A.

There are different approximations for the concrete value of the channel coherence time Since the changes of the channel coefficient are primarily caused by movements, the different approximations are commonly based on the Doppler spread or the Doppler frequency caused by the respective movements. The maximum Doppler frequency f_D can be calculated in dependence of movement speed and the respective transmission wavelength:

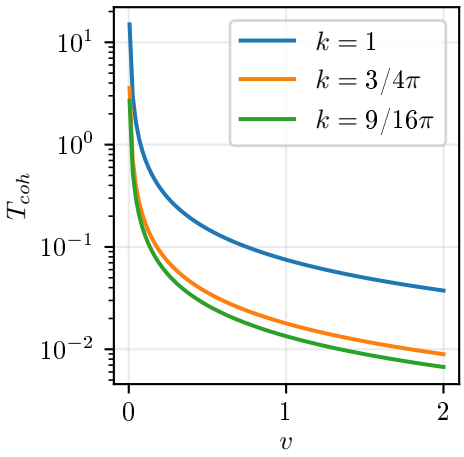
$$f_D = \frac{v}{\lambda} = \frac{v \cdot f_c}{c} \tag{2.20}$$

Based on this maximum, the channel coherence time can be approximated in dependence of k [69]:

$$T_{coh} = \frac{k}{f_D} = \frac{c * k}{f_c v} \tag{2.21}$$

Speed	T_{coh}		
	$k = 1$	$k = \frac{3}{4*\pi}$	$k = \frac{9}{16*\pi}$
$\mathrm{m/s}$	s	s	s
0.005	14.9971	3.5803	2.6852
0.05	1.4997	0.3580	0.2685
0.1	0.7499	0.1790	0.1343
0.15	0.4999	0.1193	0.0895
0.2	0.3749	0.0895	0.0671
0.25	0.2999	0.0716	0.0537
0.5	0.1500	0.0358	0.0269
0.75	0.1000	0.0239	0.0179
1.0	0.0750	0.0179	0.0134
1.25	0.0600	0.0143	0.0107
1.5	0.0500	0.0119	0.0090
1.75	0.0428	0.0102	0.0077
2.0	0.0375	0.0090	0.0067

(a)



(b)

Figure 2.12.: Example values for $T_{coh} = \frac{k}{f_D}$ for transmission at $3.998\,\mathrm{GHz}$ and varying speeds. In Fig. (b) the speed v is given in $\mathrm{m/s}$ and the Y axis is in log scale.

A straight forward approximation is $k = 1$ as described in [69]. Also, more restrictive approximation have been proposed, e.g., $k = \frac{3}{4*\pi}$ or $k = \frac{9}{16*\pi}$ in [179]. Table 2.12a and Fig. 2.12b show the implications of these different approximations for an assumed transmission at $3.998\,\mathrm{GHz}$ center frequency.

As long as the reciprocal measurements of the legitimate partners are conducted within the time interval T_{coh} the coefficients of their shared channel can assumed to be fixed. Hence, the respective channel estimation will be highly correlated as implied by Eq. (2.19).

In contrast to the reciprocal measurements, the consecutive measurements should be spaced far enough apart in time that the channel coefficients have changed sufficiently. If the measurements are too close together and the channel coefficients have changed only slightly, limited entropy can be provided for key generation. This in turn means either slow key generation or low entropy key material, i.e., easily predictable and thereby insecure key material. Hence, the interval at which successive estimates are taken should be large enough to yield uncorrelated observations. In accordance to the general description of the coherence time, the *sampling time* T_s can be expressed as:

$$h(t) \not\approx h(t+\tau) \qquad ,\forall \tau > T_s \tag{2.22}$$

$$h_{AB}(t) \not\approx h_{AB}(t+\tau) \qquad ,\forall \tau > T_s \tag{2.23}$$

As this requirement is less common in classical wireless communication, there is no generally accepted approximation of T_s.

A reasonable option to estimate the sampling interval is to use the limits derived in the context of channel sounding. Here, a common assumption is that a channel is identifiable over time, if it is sampled with frequency at least twice the maximum Doppler frequency $f_s \geq 2f_D$ [143, 151]. This implies that for sampling frequencies *below* this threshold, the

subsequent channel estimation should not be identifiable, i.e., they can be considered uncorrelated. Thereby, a very defensive lower bound for the sampling interval can be estimated with:

$$f_s < 2f_D \tag{2.24}$$

$$T_s > \frac{1}{2f_D} \tag{2.25}$$

This requirement follows the same decorrelation argument as presented above stating that for distances larger than half a wavelength the observations will be uncorrelated. Thereby, a terminal should move at least $\Delta x = \frac{\lambda}{2}$ between subsequent measurements to yield uncorrelated channel coefficients. By combining this required distance with the classic definition of speed $v = \frac{\Delta x}{t}$ and the definition of the maximum Doppler frequency in Eq. (2.20), it is clear that this is the same requirement as in Eq. (2.24):

$$f_s < \frac{v}{\frac{\lambda}{2}} \tag{2.26}$$

$$< 2\frac{v}{\lambda} \tag{2.27}$$

$$< 2f_D \tag{2.28}$$

In accordance to Eq. (2.20) this allows an estimation of the minimum sampling interval with respect to movement speed and transmission frequency:

$$T_s > \frac{c}{2f_c v} \tag{2.29}$$

It is worth noting, that this contradicts the approximation of T_{coh} with $k = 1$, since this would yield a coherence time twice as large as T_s. Hence, the more restrictive approximations should be applied.

In summary, regarding the acquisition of channel estimations, there a two different time intervals, which need to be considered. On the one hand, there is the *coherence time* T_{coh} of the channel, which is the time interval within which the channel properties can be considered static. The legitimate partner of CRKG should acquire their estimation within this interval to maintain high correlation between their realizations. On the other hand, there is the *sampling time* T_s of the participants, which denotes the interval between subsequent measurements at one partner. Following the assumption that subsequent realizations are distributed independently, this interval should be large compared to T_{coh} in order to reduce the respective correlation. Both of these aims should be realized together.

2.3.2. Channel Properties

The channel itself is completely described by the Channel Impulse Response. This is a direct continuation of the system theory view of wireless communications, based on which the wireless channel can be modelled as a Finite Impulse Response (FIR) filter. This leads to the consequence that the channel itself is completely described by its respective Channel Impulse Response, since all relevant channel properties are contained in the CIR.

It is worth noting, that other representations of the Channel State Information, i.e., of

the channel and its properties, exist. Following the analyses of Bello, a set of transfer functions can be constructed [19], which allow transforming the different representations into each other. Depending on whether the input and output values of the channel are examined in their respective representation in the time or frequency domain, the following additional transfer functions can be described besides the CIR: time varying transfer function (also Channel Transfer Function (CTF)), delay Doppler-spread function and frequency domain function. All four functions can be transformed into each other by applying (inverse) Fourier-Transformations [19]. This means, although the functions visualize different features of the wireless transmission, at their base they all contain the same information, or relevant for CRKG the same entropy. Following the argumentations of the *Data Processing Inequality* the transformation between the different representations might even reduce the available entropy [18, 102]. Hence, we only consider the CIR in the following.

By looking at the CIR as shown in Eq. (2.7), all the relevant channel properties can be described. Equation (2.9) could be used as well, but it introduces no more suitable properties. CRKG processing derives fresh entropy for the key derivation from the differences between successive measurements, i.e., the time variant parts of the CIR. Hence, we look at exactly those time-varying parts of the respective CIR representation:

$$h(t,\tau) = \sum_{n=0}^{N(t)} \alpha_n(t) e^{j\phi_n(t)} \delta(t - \tau_n(t)) \tag{2.30}$$

All properties with dependence on the time t are in theory suitable as entropy source, i.e., N, τ, α and ϕ. From a practical point of view, some are more suitable than others:

The number of multipath clusters N is closely related to the time delay τ of the respective paths. Both are determined by the respective capture process of the CIR: the total measurement time of the CIR capture limits the maximum of τ. Paths with a long time of flight might simply not be recorded, which in turn would affect the number of multipath clusters. Additionally, the measurement bandwidth determines the granularity of resolvable multipath components. If paths are non-resolvable, this directly alters N. In theory, N and τ were proposed as entropy source for CRKG [90, 89]. Nevertheless, in practice these properties are hard to obtain robustly, in part because there is no robust way to identify multipath clusters within the CIR.

The phase of the respective multipath components ϕ is time-variant and was also proposed as input for CRKG [104]. Practical evaluations in this realm showed, that it is hard to measure the phase accurately enough to facilitate an efficient key derivation [104, 133].

Finally, there is the amplitude α of the multipath components. Several studies rely on this property and successfully use it for CRKG, e.g., [79, 234, 31]. As for the amplitude, no work has reported practical problems regarding robustness and accuracy of recording this property.

Apart from using the features present in the CIR directly, there is also the possibility to derive aggregated metrics. Due to its availability, the most prominent aggregated metric is the *Received Signal Strength Indicator (RSSI)*.

RSSI is a single value indicating the overall link quality of the wireless transmission. In general, it has no formal definition but is a vendor specific indicator and the actual implementation is up to the hardware manufacturer. Nevertheless, since the RSSI is an indicator

of the total received energy, it is inherently derived from the CIR. A possible way of calculating a RSSI from the CIR is by summing up the single values of the CIR [218]:

$$\text{RSSI} = 10log(||h||^2) \tag{2.31}$$

As RSSI values are reported by nearly every wireless transceiver, its usage for CRKG was proposed several times. Nevertheless, in direct comparison to the CIR it contains much less entropy. This already manifests itself when the RSSI as a single scalar value is contrasted with the CIR as a vector of values. Further, the summation into one single value inevitably discards all information about the multipath clusters contained in the CIR. Additionally, as RSSI is only an *indicator*, it comes with two more drawbacks: its actual implementation is not defined and therefore up to the respective manufacturer, and due to the multipath cluster summation the RSSI value varies strongly even in static immobile settings (up to $5\,\text{dB}$ in [210] and up to $7\,\text{dB}$ in [115]). Hence, we refrain from using RSSI values and similar link quality indicators as entropy source for CRKG.

2.4. System Model

In this section we define the system model used throughout the remaining work. The core of this model is derived from the explanations and arguments detailed in Section 2.3 and Section 2.2. This model is also used to design the practical experiments and measurements in the upcoming parts.

The general setup for the key derivation is shown Fig. 2.13. The arrangement as a triangle in the figure is not obligatory in practice, but serves only for illustration.

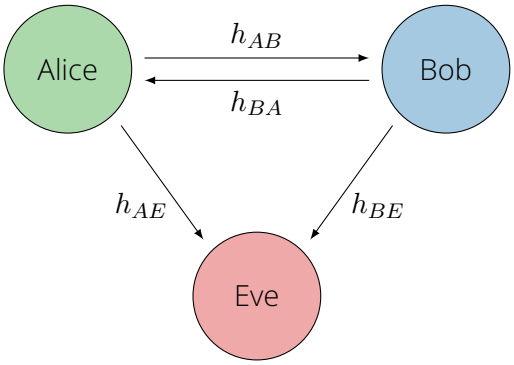


Figure 2.13.: General system setup considered in this work

To keep the system as general and simple as possible, we assume that all terminals are single-antenna systems with omnidirectional radiation patterns. Thus, due to the respective half-duplex radio mode, Alice and Bob use a ping-pong protocol for randomness sharing. Specifically, Alice sends an initial probing message to Bob, who upon receipt sends a probe of his own to Alice. Due to the broadcast nature of the wireless medium, Eve can listen to this message exchange and record her own channel measurements.

Following the reasoning in Section 2.3, we use the Channel Impulse Response (CIR) as a measured channel property and thus as input to the subsequent key derivation. The measurement messages thus provide the respective CIRs h_{XY}, with X and Y in $[A, B, E]$, as shown in Fig. 2.13. More precisely, the measurement hardware estimates the actual CIR h

based on the known input signal x (see Eq. (2.5) and Eq. (2.6))[1]. This estimate is acquired with respect to the hardware's sampling rate, i.e., one value for each sampling step. Typically, the hardware delivers the complex channel properties. Following the arguments in Section 2.3 we focus on the amplitude of the CIR. Hence, we calculate the magnitude of each sample within the CIR by taking its real $\mathrm{Re}(\cdot)$ and imaginary $\mathrm{Im}(\cdot)$ parts and combine them to $|h_i| = \sqrt{\mathrm{Re}(h_i)^2 + \mathrm{Im}(h_i)^2}$, for each time index i within the estimate. Thereby, the resulting h is a vector of scalars v

$$h = \{v_0, v_1, v_2, \ldots, v_n\} \quad v_i \in \mathbb{R}, \tag{2.32}$$

where the indices $i = \{0, 1, 2, \ldots, n\}$ represent the respective acquisition time τ_i with respect to the sampling interval T_s, i.e., $\tau_i = T_s \cdot i$. An example of this vector resulting from the obtained CIR, i.e., the input data for CRKG used in this work, is display in Fig. 2.14.

Due to the rich multipath information in their CIR, we use Ultra Wideband (UWB) transmissions. Specifically, system-dependent, our measurements take place at $4\,\mathrm{GHz}$ center frequency, $500\,\mathrm{MHz}$ bandwidth, and a sampling rate of $1\,\mathrm{ns}$, which can be easily implemented by the Commercial Of-The-Shelf (COTS) components used. Given that UWB technology has recently been integrated into various end-user devices, a practical implementation of the presented solution is realistic.

To introduce the required fresh entropy for new keys, we assume that the channel properties change adequately. Typically, this is realized by keeping one of the legitimate terminals, typically Alice, in motion. Alternatively, changes can be introduced through the transmission channel itself, i.e., objects or persons passing through the channel's LOS.

It is worth mentioning, that for other PLS primitives, e.g., wireless identification and authentication, the opposite has to be assumed: here, the channel should change as little as possible to keep the respective identifiers stable.

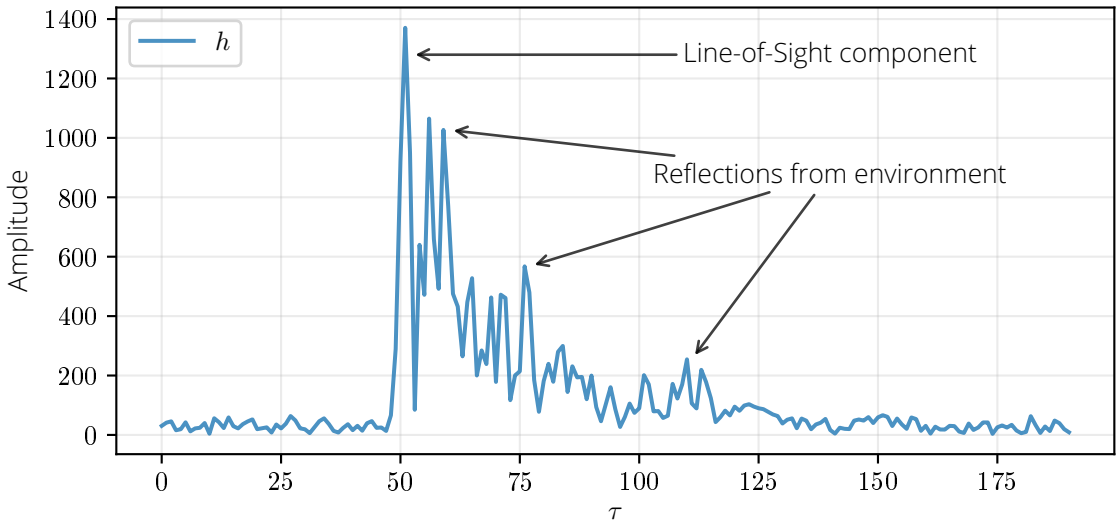


Figure 2.14.: Example of a CIR as described in Eqs. (2.9), (2.30) and (2.32).

[1]The estimated CIR is commonly denoted as $\hat{h}$. Here, we keep the notation of h for brevity.

2.5. Evaluation Metrics

In this section, we describe the metrics used to evaluate the performance of CRKG. In the different processing steps, different metrics are utilized to best evaluate the effectiveness of the particular step.

2.5.1. Measured Channel Properties

The main evaluation goal after the acquisition of the channel properties is to assess the correlation between the different realizations of the common source of randomness. Hence, we need a metric computing the correlation between two signals g and h, which in our case are CIRs.

Considering CIRs as input material, we want to have two special requirements for this metric: First, the metric should be self-aligning. Since the CIRs of different devices are not necessarily synchronized in time, the metric should ignore such time offset. And second, the metric should be normalizing to ignore potential offsets or linear scaling effects.

Common approaches to this are the Mean Squared Error (MSE), the Pearson correlation coefficient or the cross correlation. The MSE is defined as [172]:

$$MSE = \frac{1}{N}\sum_{i=0}^{N-1}(g_i - h_i)^2 \tag{2.33}$$

In this version, the MSE is neither normalizing nor self-aligning. There is a normalizing version, normalized MSE (NMSE) , defined in [145]:

$$\begin{aligned} NMSE &= \min_{\beta} \frac{\|\vec{g} - \beta\vec{h}\|^2}{\|\vec{g}\|^2} \\ &= 1 - \left(\frac{\vec{g}^T\vec{h}}{\|\vec{g}\|\|\vec{h}\|}\right)^2 \end{aligned} \tag{2.34}$$

Here, $\vec{g}$ and $\vec{h}$ are g and h interpreted as vectors.

A widely used metric in the field of RSSI based CRKG is the Pearson correlation coefficient. It is defined as

$$\rho(g,h) = \frac{\sum_{i=0}^{N-1}(g_i - \bar{g})(h_i - \bar{h})}{\sqrt{\sum_{i=0}^{N-1}(g_i - \bar{g})^2 \sum_{i=0}^{N-1}(h_i - \bar{h})^2}} \tag{2.35}$$

with

$$\bar{g} = \frac{1}{N}\sum_{i=0}^{N-1} g_i \qquad \bar{h} = \frac{1}{N}\sum_{i=0}^{N-1} h_i \tag{2.36}$$

The Pearson correlation is normalizing, but not self-aligning. Additionally, its intended use case does not match the usage with CIRs: originally the correlation is calculated for N pairs of scalar values as g and h. The application to g and h as CIRs is not justified here.

Finally, the **cross correlation** can be used as metric. The cross correlation between two

signals is defined as [141]:

$$r_{gh}(k) = \sum_{i=-\infty}^{\infty} g_i h_{i-k} \tag{2.37}$$

The input signals g and h are padded with zeros to matching length. If the cross correlation is normalized by the energy of the signals (E_g and E_h) and the maximum is used as result, it is applicable as normalizing, self-aligning metric:

$$\begin{aligned} r_{gh}(k) &= \max_k \frac{\sum_{i=-\infty}^{\infty} g_i h_{i-k}}{\sqrt{E_g E_h}} \\ &= \max_k \frac{\sum_{i=-\infty}^{\infty} g_i h_{i-k}}{\sqrt{\sum_{i=0}^{N_g-1} g_i^2 \sum_{i=0}^{N_h-1} h_i^2}} \end{aligned} \tag{2.38}$$

We use the normalized cross correlation r_{gh} as metric for the correlation between two CIRs. We introduced the other two, as they will become relevant in the scope of ML.

The best metric to evaluate the suitability of the estimated CIRs for use in CRKG would actually be Mutual Information, as defined in Information Theory [42]. However, the special properties of CIRs significantly complicate the computation of MI. Especially the possible existence of correlations between single/adjacent values within one CIR, as well as between consecutive CIRs make this calculation difficult. To the best of our knowledge, there is currently no robust approach to calculate the MI of CIRs, which is why we currently have to refrain from employing this metric.

To visualize the cross correlation results, we use a combination of boxplots and split violinplots. An example is shown in Fig. 2.15. The main feature of this plot is the Gaussian density estimation, which visualizes the respective result distribution. The three solid black lines are the minimum, maximum and median; the dotted line is the mean. The darker area in the middle represented the interquartile range (IQR). Especially in cases where the attacker data should be directly opposed to those of the legitimate partners, we employed the split version of the violin plot. Here, the results of the legitimate partners (upper split, green) are directly compared with those of the eavesdropper (lower split, red). It is worth noting, that both splits are normalized independently to the maximum violin width. Hence, the area under the curve with respect to the estimated density is equal to 1 in both splits — although the actual sizes might differ, e.g., a "flat" and "wide" distribution might appear bigger, than a "narrow" one. If multiple violinplots are combined into one plot, the Y axis will denote the respective scenario.

In addition to correlation, we examine the raw measurement data primarily for the **time synchronization** of reciprocal measurements, or, respectively, the temporal shift between them. The concrete problem and the corresponding implications are described in more detail in Sections 5.3 and 7.1. To describe the metric for this analysis, it is sufficient to know that the reciprocal measurements are not time synchronous but are shifted with respect to each other by a random value that we call absolute offset δ. This δ has to be interpreted with respect to the sampling rate of the measurement and actually represents time shift of δt_s. For instance, if we have $\delta = 2$ and a sampling rate of $f_S = 1\,\mathrm{GHz}$, the actual time delay is $\delta \frac{1}{f_S} = 2 \frac{1}{1 \times 10^9/\mathrm{s}} = 2\,\mathrm{ns}$.

We now introduce the metrics we propose to use for the analyses of this problem:

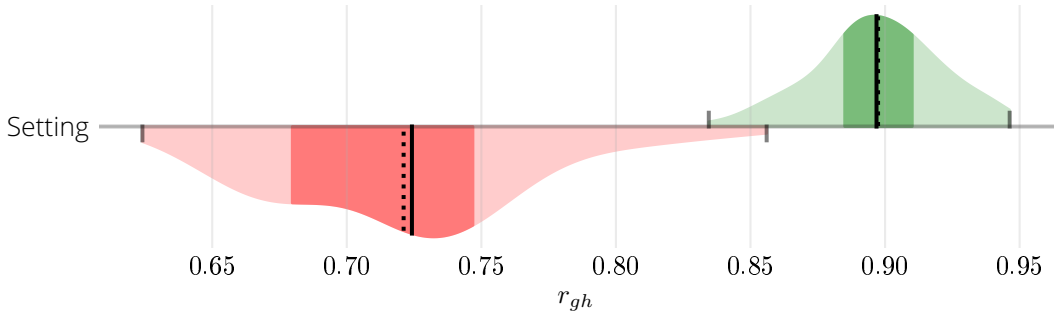


Figure 2.15.: Example plot to visualize the cross correlation r_{gh} of the legitimate partners and the respective attacker.

To remove the time shift between the reciprocal measurements, we need an approach which correctly identifies the offset δ. The optimal solution for determining this temporal shifts or time offsets between two signals is a function which perfectly identifies the actual offset of two given signals. This means in turn, that such a functions uniquely identifies a time shift for which the "overlapping" of the given signals is maximal — since the given signals would overlap maximally in case of $\delta = 0$.

Equations (2.37) and (2.38) concerning the correlation already define the cross correlation, which calculates the offset with a maximum overlap of the reciprocal measurements. Hence, we can also use the cross correlation as a reference point or ground truth for the calculation of the time offset. However, it cannot be used directly for the calculation of the necessary time synchronization, since both measurements are required for the formula evaluation — which is not possible in practice, since here the respective communication partners have only their own observations available.

Equation (2.38) actually delivers the maximum cross correlation r_{gh} over the range $k \in [-\infty, \infty]$. To determine the optimal time offset, we are actually using the value of the argument k. Hence, in accordance to Eq. (2.38), we defined the reference offset Δ_{opt} between two CIRs g and h:

$$\Delta_{opt}(g,h) = \arg\max_{k} \frac{\sum_{i=-\infty}^{\infty} g_i h_{i-k}}{\sqrt{E_g E_h}} \tag{2.39}$$

Here, we assume that Δ_{opt} is sufficiently close to δ. Keep in mind, that this is only useful during the evaluation of different time synchronization approaches, as both reciprocal CIRs are needed.

To actually employ this as metric for different approaches, we proceed as follows: The different locally executed approaches A define a unique anchor in time at each communication participant. This anchor is a unique data point within the measurement, which acts as the origin of the time axis for this measurement. By using the global view of this analysis, these local anchors can be related to each other to calculate a time offset for this actual approach A. The closer a given approach's time offset is to the optimal shift Δ_{opt}, the better is the respective approach's synchronization performance.

$$\Delta_A = \Delta_A(g,h) - \Delta_{opt}(g,h) \tag{2.40}$$

$$\vec{\Delta}_A = (\Delta_A^1, \Delta_A^2, \dots \Delta_A^n) \tag{2.41}$$

$$\Delta_A^0 = \frac{|\{d \in \vec{\Delta}_A : d = 0\}|}{|\vec{\Delta}_A|} \tag{2.42}$$

We implement the metrics as follows: The difference between the optimal time offset Δ_{opt} and approach A's time offset Δ_A for a given pair of CIR measurements g and h is shown in Eq. (2.40). By applying this difference to all n available CIR observations we obtain the vector $\vec{\Delta}_A$. This vector is used in a twofold manner: First, it is used to calculate the mean and standard deviation of the time differences for this approach. And second, it is used to computed the final metric $\Delta_{A'}^0$ which is the fraction of optimal time offset generated by the current approach (Eq. (2.42)). An optimal solution is an approach with $\vec{\Delta} = (0, \dots 0)$, i.e., $\Delta^0 = 1$.

2.5.2. Quantized Key Material

Usually, the measured channel characteristics are quantized into bit strings after optional preprocessing. In formal terms, the quantization function f_Q maps the real-valued measurements (see Eq. (2.32)) to integers:

$$f_Q : \mathbb{R}^n \to \mathbb{Z}^n \tag{2.43}$$

In the context of CRKG, this function usually maps directly to the binary number space:

$$f_Q : \mathbb{R}^n \to \{0,1\}^m \tag{2.44}$$

That is, after quantization, the observed CIRs h_{XY} at Alice, Bob, and Eve are bit strings, which we denote as b_{XY}. The processing steps subsequent to quantization typically all utilize bit strings. Hence, the following metrics are applicable at all of these stages.

The most important properties security-wise are still the similarity or non-similarity of the respective observations as well as their randomness.

To assess the **similarity of bit strings**, we employ the Bit Error Rate (BER). In the context of CRKG, this metric is often called Bit Disagreement Rate (BDR), which despite the different name is the same metric.

Generically, the BDR is defined as

$$BDR = \frac{\text{Erroneous bits}}{\text{Total number of bits}}. \tag{2.45}$$

In our case and with respect to the quantized bit string b_X and b_Y, this is realized as

$$BDR = \frac{w(b_X \oplus b_Y)}{|b_X|}. \tag{2.46}$$

Here, $\oplus$ denotes the XOR operation, $w(\cdot)$ the Hamming weight, and $|\cdot|$ the length of the bit string. Since we assume the same processing at all participants, the respective bit

strings are assumed to be of equal length. Hence, the length of b_X is representative for the length of all bit strings.

It is worth noting, that for multiple bit string pairs with a fixed length, the BDR is equal to the average Hamming distance of all pairs. So, assuming n pairs of b_X and b_Y with length m each, the BDR can equally be computed as

$$BDR = \frac{1}{n} \sum_{i=1}^{n} |\{j \in \{1, \dots, m\} : b_{X,i}[j] \neq b_{Y,i}[j]\}| \tag{2.47}$$

In the scope of CRKG, the BDR of the legitimate partners should be as low as possible, ideally 0, whereas those between one of the legitimate partners and an attacker should optimally be 0.5. If the attacker has a BDR of 0.5, it means that each outcome is equally probable, which in turn means the attackers chance of getting a correct bit is no better than with mere guessing.

In addition to this similarity metric, it is easier to calculate or estimate the Mutual Information with quantized values. Specifically, this means that for short bit strings ($leq\ 16\,\mathrm{bit}$) the MI can be calculated by setting up the joint distribution $p(x, y)$ of the respective realizations x,y. From this, the marginal distributions $p(x)$ and $p(y)$ can then be derived, which allows the direct calculation of the mutual information $\mathbb{I}$:

$$\mathbb{I}(X;Y) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x,y) \frac{p(x,y)}{p(x)p(y)} \tag{2.48}$$

For bit strings whose length is greater than 16 bits, setting up the joint distribution quickly leads to exhaustion of the computing capacity. Therefore, the analytical calculation cannot be performed for these instances. Instead, estimators can be used here. An established estimator in this area is the KSG estimator developed by Kraskov, Stögbauer, and Grassberger [106]. It is based on the work of Leonenko and Kozachenko and employs the k-Nearest Neigbours (kNN) algorithm [113]. Thereby, this estimator already provides valid results from 1000 realizations. Nevertheless, it must be mentioned that this procedure only provides an estimate and thus does not necessarily yield correct MI results [65, 14].

Finally, a common metric to assess the **efficiency** of CRKG is the Key Generation Rate (KGR), often also called Secure Key (Generation) Rate (SKR/SKGR). In the literature this metric is often employed as $\mathrm{bit/s}$, e.g. [12, 133, 96, 157, 74, 129, 222, 120, 31]. Here, we explicitly deviate from this notion of $\mathrm{bit/s}$ and instead use generated bits per channel usage, i.e., $\mathrm{bit/cu}$.

The reason for this is that the inclusion of an explicit time frame, $1\,\mathrm{s}$, yields an unreliable comparison metric regarding the actual processing. This is illustrated by the fact, that different solutions sample the channel at different rates. In this case, if a solution increases the channel sampling rate without changing the processing, the metric $\mathrm{bit/s}$ would increase whereas the actual processing efficiency of the key generation $\mathrm{bit/cu}$ stays the same. Hence, the $\mathrm{bit/s}$ does not reflect the processing efficiency correctly, but is heavily influenced by the respective sampling rate. It is worth noting that in initial works regarding CRKG like [134] the unit for key generation rate is proposed to be $\mathrm{bit/cu}$ and not $\mathrm{bit/s}$.

In summary, for evaluation and comparison of CRKG efficiency we use KGR in terms of $\mathrm{bit/cu}$.

2.6. Selected Machine Learning Topics

Since we apply Machine Learning (ML) methods in different solution designs and realizations, we now give a high level introduction to relevant topics of this field. Since this field of research is extremely diverse and extensive, only a rough overview can be given here. For a more detailed and in-depth introduction, please refer to the relevant literature [24, 142, 6]. The following descriptions are mainly based on [24, Chapter 5].

Within this work, we primarily use the ML approach of Artificial Neural Networks (ANNs), more specifically Convolutional Neural Networks (CNNs), which are trained via backpropagation. At its core, an ANN is a representation of a complex mapping of an input vector x_k to an output vector y_k. This mapping is realized by employing artificial neurons in a multi-layer architecture. Hence, classical ANNs are also referred to as *multilayer perceptrons*. An example of this multi-layer architecture is shown in Fig. 2.16.

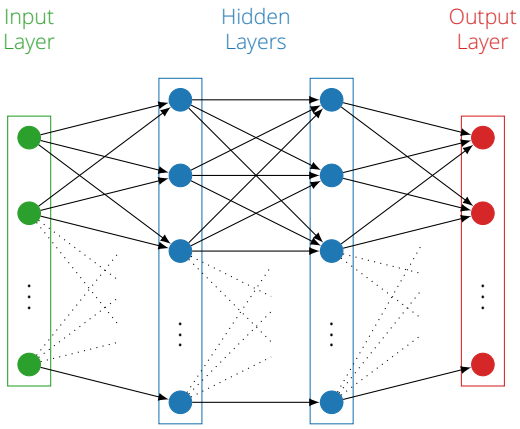


Figure 2.16.: Simple example of a multi layer Artificial Neural Network.

The actual "neurons" within the single layers are functions of their respective inputs, weights and local biases. Based on these inputs and weights the respective sum is calculated which is passed to the activation function z. With the first layer (1) the output of the k^{th} neuron can be expressed as

$$y_k^{(1)} = \sum_{j=0}^{N} w_{kj}^{(1)} x_j^{(1)} + b_k \tag{2.49}$$

Here, w_{kj} are the weights and b_k is the bias of the respective neuron. The result of this function is then passed to a *non-linear activation function* $h(\cdot)$. This function is typically sigmoidal. Thereby the actual output of the neuron is

$$z_k = h(y_k). \tag{2.50}$$

The core idea is presented in Fig. 2.17.

The multilayer nature of the network can be shown by composing the function of the different layers. So, if at layer (i) the neurons are activated by function $h^{(i)}$ and the weights $w_{kj}^{(i)}$ of the single neurons are represented as vector $W^{(i)}$, the resulting vector $z^{(i)}$ can be

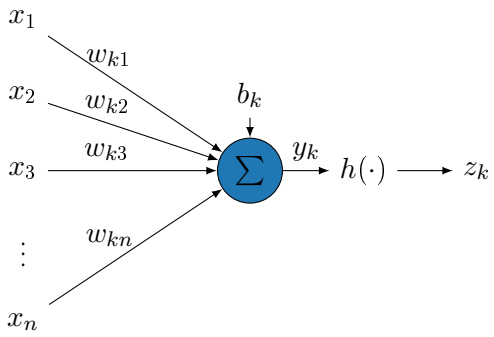


Figure 2.17.: Concept of a single artificial neuron in an ANN.

written as

$$z^{(i)} = h^{(i)}(W^{(i)} z^{i-1}) \tag{2.51}$$

Hence, the output y of a network with M layers with input x is

$$y = h^M(W^M h^{M-1}(W^{M-1} \cdots h^1(W^1 x) \cdots)) \tag{2.52}$$

This calculation is executed for all neurons in all layers to compute the final output, i.e., the result vector of the last layer. This is the *forward* pass through the network.

To actually facilitate training and adaption of the network the resulting error, i.e., the difference between the current result vector and aimed for vector, the ground truth, has to be propagated back through the network. This so called *backpropagation* enables the training of the neural network and is based on two core principles: first, the *loss function* L which quantifies the actual error of the current prediction; and second, the *gradient descent* which enables the minimization of the loss.

The loss function L "compares" the result of the forward pass y with a known ground truth $\hat{y}$. Thereby the error of the respective calculation is quantized and the learning process can be adapted. The loss itself is a function of the input vector x, the current weights W and the ground truth $\hat{y}$, i.e., $L(W, x, \hat{y})$. Typical loss functions are the Mean Square Error or Cosine Similarity for regression, and Binary or Categorical Cross-Entropy for classification [30, 148].

Building from this loss value, gradient descent enables the training of the network by adapting the networks weights W in dependence of the current loss L. For this, the influence of each neuron's weight w_{kj} to the final loss L is computed as partial derivation of the final loss, i.e.

$$\frac{\partial L}{\partial w_{kj}} \tag{2.53}$$

The overall gradient in dependence of the weights W is written as $\nabla L(W)$. As our goal is to minimize the error of the networks results, we want to find those values for W for which L becomes minimal. Since it is impossible to find an analytical solution to this, this minimization of L is performed iteratively. In a single iteration step τ the values w are adapted as

$$w^{\tau+1} = w^{\tau} + \delta w^{\tau} \tag{2.54}$$

The actual value of the update δ depends on the concrete implementation. But at the core all updates incorporate the gradient $\nabla L(W)$ in order to determine the "direction" within the weight space.

This iterative processing is the so called *training* of the ANN. It is repeated until a minimum is reached in which, ideally, $\nabla L(W) = 0$ or at least no further improvement can be achieved through the weight adaption. In such a minimum, the update of the weights no longer reduces the final loss L, i.e., the gradient descent has converged. Since no further improvement or learning is possible, the training of the network is completed. It must be kept in mind that this minimum does not necessarily represent the optimal solution — only with respect to the current network, its weights and the available data, a minimum has been reached here from which no further improving step is possible.

Typically, the training process incorporates different measure to ensure *generalization* of the network. This is required to avoid *overfitting* of the network. As neural networks are generally intended to solve a problem for a certain class of data, the weight adaption should not be a perfect match to the given training data. In such case, the network could solve the respective problem *exactly* for this set of data, but not for similar data. Hence, generalization measures are employed, like e.g., *Dropout* layers or *Batch Normalization*.

By combining the different described parts it is possible to devise powerful neural networks, which can significantly outperform "classical" approaches.

3. State of the Art

In this section we introduce the current state of the art with respect to the topics addresses in this thesis. We start with notable works, both theoretical and practical, proposing complete CRKG systems. Subsequently, we introduce approaches related to the different single processing steps of CRKG. Finally, we present practical attack approaches proposed to compromise this key generation scheme.

3.1. Implementations of Channel Reciprocity-based Key Generation

In the following, we present notable works in the realm of CRKG. Given their prevalence in the research literature, we first talk about RSSI-based schemes. Although this is not the major focus of this work, the ideas originating from this part of CRKG were often the first of their kind and reused in works handling different input data.

Thereafter, we present the state of the art concerning CIR-based CRKG. This is divided into theoretical and practical works: in the former several core ideas of current CRKG implementations were proposed. Nevertheless, given the drawbacks of simulation-based input data, the respective schemes and their evaluations might not represent real world performances. Hence, we subsequently focus on works, which realize their ideas in practical implementations and evaluate them with real world data.

3.1.1. Received Signal Strength Indicator

RSSI is one of the most common channel characteristic used for CRKG. Due to its general availability at almost every wireless transceiver, multiple research groups focussed on this characteristic. In the following, we give an overview over practically oriented solutions in the field of RSSI-based CRKG.

One of the first implementations of CRKG in a real world setting was those of Mathur et al. [133]. This work proposes a RSSI-based solution, which employs an IEEE 802.11a/b/g development board, equipped with additional custom logic to extract further channel properties beside the RSSI. The main contribution with respect to RSSI-based CRKG is the proposal of the level crossing-based quantization scheme, dubbed Guard Band Quantizer (GBQ), which is widely employed in later RSSI-based CRKG implementations. Beside quantization

it inherently achieves reconciliation. The real world measurements of this RSSI-approach yielded a KGR of $1.3\,\mathrm{bit/s}$ with 22 samples per second.

With a focus on quantization Jana et al. [96] realized a further RSSI-based scheme. They proposed a different quantization scheme, called Adaptive Secret Bit Generation (ASBG), which used blockwise processing to generate the statistical moments required for the actual quantization. Further, they proposed to use Gray Code [72] based multi-level quantization to increase the generation rate of secret bits. In their schemes they relied on the Cascade scheme [29] for *Information Reconciliation*, which was adopted by several upcoming works. Later, they expanded their schemes to Multiple Input Multiple Output (MIMO) settings as well as IEEE 801.15.4 based technologies [157] and IEEE 802.15.1, i.e., Bluetooth [158].

Zenger et al. proposed a system based on IEEE 802.11 RSSI values, designed as plugin solution for existing IEEE 802.11 systems [229]. This work also showed the connection between the RSSI correlation and the resulting BDR — in their experiments, a correlation of 0.75 results in BDR below 0.03. Additionally, they analysed the computation effort of CRKG compared to Elliptic Curve Diffie-Hellman (ECDH) and showed that CRKG requires up to 61.3 times less energy than ECDH.

Guillaume et al. analysed different transmitter/receiver setups in varying indoor and outdoor scenarios [74] and thereby confirmed that the legitimate channel has an advantage over the eavesdropper channel in practice. They achieved effective key generation rates of $0.01\,\mathrm{bit/s}$ to $15.45\,\mathrm{bit/s}$. It is worth noting, that they use a very high sampling rate of up to $122\,\mathrm{samples/s}$ to $230\,\mathrm{samples/s}$, which implies that at most $0.127\,\mathrm{bit}$ were extracted per channel use.

In [236] an adaptive version of Mathurs GBQ scheme is proposed. Here, the guard bands are adapted over the course of application in accordance to the current mean and standard deviation of the RSSI values. They implemented a complete scheme with Cascade [29] as Information Reconciliation (IR) solution and tested it with real world measurements of 802.11g transmissions. With a sample rate of 20 samples per second, they achieved a KGR of $18.8\,\mathrm{bit/s}$, which equals $0.94\,\mathrm{bit/cu}$.

Finally, Lin et al. proposed a multilevel version of GBQ [118]. Again in combination with Cascade, their approach yields a rather high BDR of $14.3\,\%$. They authors claim a KGR of $4.3967\,\mathrm{bit/s}$ without stating a sampling rate.

The core approach of using link quality indicators as input for CRKG was also proposed for technologies apart from IEEE 802.11.

In the realm of IEEE 802.15.4 several RSSI-based solutions have been proposed.

The most notable work in this realm is of Aono et al. [12]. Beside the proposal of different quantization schemes for RSSI streams, they achieve a high key rate of $128\,\mathrm{bit/s}$. Nevertheless, to realize this key rate they proposed and employed special ESPAR antennas and beam-forming, making this scheme highly specialized. The respective high key rate was achieved with $\approx 384\,\mathrm{samples/s}$, implying $0.333\,\mathrm{bit/cu}$.

In [206] the authors investigated RSSI values on multiple channels. In their approach they proposed a multilevel quantization scheme based on blockwise statistics. Additionally, an IR scheme is proposed, which transmits the offset of the current blocks mean to the partner, which adjusts his measurements accordingly. The main evaluation metric is the BDR, where they achieved 0.03.

Using the same technology in the context of Body Area Network, Ali, Sivaraman, and

Ostry proposed to employ Savitzky-Golay filters to enhance reciprocity [4]. Employing the quantization scheme of Mathur et al. and no IR at all, they still claim a BDR below 4% using fast changing components of the RSSI stream. By sampling the channel with 25 samples per second, their maximum KGR is reported as $10.18\,\mathrm{bit/s}$, resulting in an effective BDR of $0.4072\,\mathrm{bit/cu}$. The authors later published an extended analysis of their results [3]

RSSI-based systems were also proposed in the realm of LoRaWAN. Xu, Jha, and Hu present a system facilitating multibin guardband quantization [215]. The authors conclude that due to the low data rate, an efficient key generation is challenging with LoRaWAN.

Different works are proposing further preprocessing steps to either reduce the correlation of subsequent samples or to enhance the reciprocity.

Patwari et al. proposed the *HRUBE* system, which includes mechanisms for both: for reciprocity enhancement they proposed the application of a fractional interpolation filter; for temporal decorrelation they applied the discrete Karhunen Loèven transformation (KLT) [154]. In their evaluation they showed the effectiveness of both approaches for RSSI vectors. The overall system achieved a KGR of $22\,\mathrm{bit/s}$, with a BDR of 0.022. Nevertheless, as this system works with 50 samples per second, the effective KGR is actually $0.44\,\mathrm{bit/cu}$.

Yasukawa, Iwai, and Sasaoka proposed a system based on ESPAR antennas and special beamforms [219]. To decorrelate subsequent measurements they proposed the usage of the discrete cosine transformation (DCT) [220]. Under optimal conditions, this system reaches a KGR of $\approx 0.9\,\mathrm{bit/cu}$, with a BDR of zero for SNR $> 30\,\mathrm{dB}$. The application of DCT was also re-introduced later [129].

Finally, we present the works of Yuliana et al. [226, 224, 225]. In their works they employ Kalman filters as preprocessing (as proposed in [8]) and proceed with Gray code-based multilevel quantization. What is special about their work is, that they completely omit the IR phase and thereby the respective communication overhead. Since they do not adapt the processing accordingly, their achieved KGR is very low with 0.93 bits/s — as they measured ≈ 9.1 RSSI samples per second, this represents $0.102\,\mathrm{bit/cu}$. Further, the effectiveness of this processing is suboptimal — from 4000 samples only 4 keys could be derived. Additionally, the BDR between the legitimate partners appears only slightly better than those of the attacker — this suggests that the derived keys may be also obtained by the attacker. Since no evaluation to this end is conducted, this cannot be verified. Nevertheless, this approach is the only one so far trying to omit all communication cost of the CRKG processing.

In Table 3.1 we give an overview of the state of the art regarding RSSI-based CRKG approaches. Beside the Key Generation Rate reported in many works, we also show the key bits generate per *probe* or *channel use*. As discussed in Section 2.5, this exclusion of the data acquisition time frame yields a much more meaningful metric than KGR. This is vividly illustrated by the entries [12] and [154], both reporting high KGR values. If their probing frequency is considered, $384\,\mathrm{MHz}$ and $50\,\mathrm{MHz}$ respectively, the schemes effectiveness in terms of bits/channel use is much more realistic compared to the related works.

Often different works implement different metrics without defining clear relations to other approaches. Therefore, certain values not directly reported in the cited works were manually derived from the mentioned metrics or extracted from visual representations.

The *communication* column indicates the minimally required communication between the legitimate partners, in terms of *repetitions* $\times$ *directionality*. This means, e.g., 1×2 represents one bidirectional exchange.

Table 3.1.: Overview over selected RSSI-based solutions.

Ref.	Technology		$\mathrm{bit/s}$	$\mathrm{bit/cu}$	Communication
[133]	802.11a	RSSI	1.3	< 0.059	1×2
[96]	802.11g		15.45	< 0.127	1×2
[157]	802.11g		0.9	< 0.250	$n \times 2$
[74]	802.11n		15.5	< 0.127	1×2
[236]	802.11g		18.8	< 0.940	$n \times 2$
[118]	802.11g		4.4	$?$	$n \times 2$
[158]	802.15.1		0.2	< 0.276	$n \times 2$
[154]	802.15.1		22.0	< 0.440	1×1
[12]	802.15.4		128.0	< 0.333	1×2
[3]	802.15.4		0.3	< 0.250	1×2
[129]	802.15.4		1.6	< 0.930	1×1

The main drawback of using RSSI as channel characteristic is its inherent simplification of the channel: it ignores the multipath nature of the wireless propagation and reduces these rich information into a single value. Further, as the calculation of RSSI values is not standardized, it is up to each hardware provider to define its calculation, which can lead to non-reciprocity of the legitimate channel, which is shown in [74]. For reasons like this, the application of richer channel characteristics should be favoured.

3.1.2. Channel State Information

Although CSI are less frequently available as a common source than RSSI, there are several approaches based on this characteristic — typically in the form of CIRs. This is primarily due to the fact that very systematically CIRs can deliver a significantly higher bit rate per channel usage than RSSI. However, given the only starting practical availability, primarily theoretical approaches supported by simulation data exist, and only some practical implementations are available. In the following, we present both important theoretical work and practical realizations that use CIRs.

Theoretical approaches and simulations

One of the first theoretical works to propose the usage of reciprocal channel state information was those of Ye et al. [221]. In their work they proposed direct quantization to generate bits from the assumed Gaussian random variables. For IR, a LDPC code with rate 0.5 and block size $4800\,\mathrm{bit}$ is proposed, where syndromes are transmitted for reconciliation. The main contribution is the analysis of the theoretical secret key rate with respect to the channel SNR.

The authors later extended their scheme to ITU channels [223]. Here, they proposed a scheme to decompose the channel and to extract the respective complex channel taps. The subsequent CRKG related processing is equal to that in [221].

Following this initial work, the usage of several different channel properties were proposed for key derivation.

Kitaura et al. [104] proposed a system based on time delays of different simulated UWB multipath components. Using syndrome-based decoding they achieve perfect matches

between Alice and Bob and at least 2 bits differences to Eve (at 30 bit key length). Nevertheless, the usage of time delays implicitly requires robust multipath cluster detection, which is an open question for real world measurements.

This approach was adapted by Huang et al. [90, 89]. In their work they proposed to employ *Rake Receivers* to robustly extract the time delays between different multipath clusters. After this extraction, they applied to CASCADE protocol for IR, without any Privacy Amplification. They evaluated their scheme by adapting the Saleh-Velenzuela channel model and simulating a LOS and a NLOS channel. The evaluation mainly targets the achieved BDR, which ranges from $1\,\%$ to $10\,\%$, depending on the simulated SNR. Huang and Jiang additionally applied the NIST test suite to the results, indicating randomness of the resulting key material [89].

It was also proposed to facilitate *Bit Error Rate* fluctuation as entropy source [104]. Simulations of the respective channel properties yielded potential key rates of $10.31\,\mathrm{bit/s}$. Considering the sampling time of $10\,\mathrm{ms}$ this system extracted $0.10\,\mathrm{bit}$ from each channel observation.

An alternative approach is to use to occurrence of deep fades in the Channel State Information [15]. The major challenge here is to synchronize the generated bitstreams. For this, the authors design a scheme relying on the exchange of *Key Verification Information* calculated via a keyed hash function. Concretely, they transmit a subsequence of the bitstream to the legitimate partner, who, in turn, performs an exhaustive search on their own bitsequence. Due to this brute force approach this scheme comes with high computational effort. The evaluation solely shows the effectiveness of the scheme, but none of the typical metrics are reported. In an exemplary calculation it is shown, that a $55\,\mathrm{bit}$ key could be generated in 42 seconds, which represents a secret key rate of $1.31\,\mathrm{bit/s}$.

With a focus on UWB transmissions, different theoretical works derived bounds regarding the achievable secret key rates.

Wilson, Tse, and Scholtz analysed the upper bound of the secret key rate [208]. Simulation with the Saleh-Valenzuela channel model yield key rates ranging from $95\,\mathrm{bit}$ to $105\,\mathrm{bit}$, given $30\,\mathrm{dB}$ SNR. Further they derived the secret key capacity of this channel and showed that the Mutual Information represents the upper bound for this capacity. Finally, they analysed the key rate with respect to the channel bandwidth and showed a non-monotonic increase of the key rate with increasing channel bandwidth.

The theoretical analyses of Liu, Draper, and Sayeed confirmed these results for Orthogonal Frequency Division Multiplexing (OFDM) channels [121]. Following their analysis, an effective key rate of $104\,\mathrm{bit/cu}$ is theoretically possible for such transmissions.

A very different result was deduced by Wallace: in their analysis single antenna system can achieve at most $\approx 2\,\mathrm{bit/cu}$. Only by increasing either the number of antennas into a MIMO system or by increasing the number of multipath components, this value can be increased to at most $25\,\mathrm{bit/cu}$ for $4x4$ MIMO systems [190]. In this work the channel quantization with guardbands (CQG) is proposed (not to be confused with GBQ from [133]), which is later extended into the more general channel quantization alternating (CQA). Both schemes rely on global knowledge of the channel properties Cumulative Density Function (CDF) and assume zero mean Gaussian channel properties. The CGQ scheme was further analysed theoretically in [182]. The authors showed that CQG yields at most $0.14\,\mathrm{bit/cu}$ at $8\,\mathrm{dB}$ SNR.

Finally, Zhang et al. analysed the autocorrelation of subsequent OFDM-based CTFs [233].

They showed that the $50\,\%$ coherence time, i.e., the time until the autocorrelation falls to $50\,\%$, does not yield uncorrelated subsequent measurements. They showed this by applying a CDF-based quantization scheme and analysing the output with the NIST randomness test suite. Here, sampling intervals lower than the $12\,\%$ coherence time yielded bit strings which pass all National Institute of Standards and Technology (NIST) tests.

The works of Madiseh et al. [125, 124, 126], which are cumulated in the respective thesis [123], also employed real world UWB measurements to evaluate the initial key material. Special waveform generators and oscilloscopes with a sampling frequency of $40\,\mathrm{GHz}$ were used to analyse the temporal and spatial correlation of CIR observations. The analysis was performed only on the legitimate partners for which correlations in the range $\rho = 0.90$ to 0.97 were reported. The two most important results of this work are that the CIR data were found to be suitable for key generation with respect to their temporal and spatial correlation, and that even small synchronization errors between reciprocal CIRs reduce the respective MI and thereby the achievable key rate very quickly.
Although the correlation analysis was performed using real measured values, these measured values were unfortunately not used for the actual key generation. For this purpose the statistical Saleh-Valenzuela model was used: here, the authors customized the CM1 model so that 2000 values per CIR were observed. Thus, a very high key rate of $\approx 10 \times 10^4\,\mathrm{bit/cu}$ could be generated in these simulations, which however do not reflect realistic values.

Real World Measurements

The first proposed practical system using CIRs is again the work of Mathur et al. [133]. In the same work in which the RSSI-based solution was presented, a CIR-based solution was proposed as well. Although it is dubbed a CIR-based solution, it does not exploit the potential of this property: instead of using the complete vector of the CIR, solely the magnitude of the highest peak within the CIR is facilitated. This implicit reduction from CIR to pseudo-RSSI by considering only the peak value dismisses all effectiveness advantages of CIR usage. This is also reflected in the evaluation of this approach: The authors report a KGR of $1.13\,\mathrm{bit/s}$. Since the channel observations are recorded with 9.091 samples per second, the effective key generation rate is $0.1243\,\mathrm{bit/cu}$. Nevertheless, considering the goal of omitting the need for an authenticated channel in CRKG, it can be said that this work already aimed in the same direction as ours (even if its solution still required communication).

Hamida et al. [79] extended the work of [133] by applying adaptivity to the guardband quantizer. They evaluate their scheme using 23 UWB measurements, for which correlation coefficients of $\rho_{AB} = 0.95$ and $\rho_E = 0.00$ are calculated. After application of the proposed quantizer and ECC-based IR, they report perfect agreement between the legitimate partners. Nevertheless, in their approach an attacker also correctly derives $\approx 40\%$ key bits, making this scheme inherently insecure.

The system of Mathur et al. was later expanded in [222]. The main improvement is the introduction of an LLR-based quantizer, which is based on the CDF of the input data. Thereby, over-quantization could be facilitated, which significantly improved the key rate. Further, an explicit IR step was added, where a (3,6)-LDPC code was used on blocks of 400 bits for syndrome-based reconciliation. Hence, 200 bits are revealed as they are exchanged as syndrome. Although the overall performance of the system was considerably improved,

the main drawback of the original system is still retained: the majority of the input entropy is discarded, as only the peak value of the CIR is used for key generation.

In [234] the authors try to account for the nature of the CIR input data and the respective differences to RSSI processing. For this, a quantization scheme dubbed *Jigsaw Encoding* is proposed: here, each time step of the CIR is assigned 2 vectors of random numbers of the size equalling the intended quantization width, which are called zero- and one-map. Then, the values at each time step are quantized as usual with the intended width. But the quantized value is used as index, indicating from which vector the random values are taken. This means, for indices below the index random values from the zero-map are taken, for indices above they are taken from the one-map. Afterwards, they proceed with RS-based IR, where parity bits are exchanged. The scheme is solely evaluated regarding secret key rate, which is $\approx 5\,\mathrm{bit/cu}$.
The proposed scheme implicitly requires an additional exchange of the zero- and one-maps, which also leaks this information to the attacker. Considering the well-defined underlying structure of the input data, this exchange leaks substantial portions of the preliminary key material to the attacker. No evaluation towards this end was conducted.

With general CRKG as motivation, Marino et al. [130] conducted real world UWB CIR measurements with special hardware nodes. The main focus of their work is the comparison of different quantization schemes, which are variations of Adaptive Secret Bit Generation (ASBG). The evaluation regarding BDR showed, that this guardband-based quantization achieves $9 - 41\%$ mismatches for the legitimate partners and $46 - 52\%$ for the eavesdropper. Additionally, they reported the correlations of the raw material, with $0.89 \leq \rho_{AB} \leq 0.97$ and $0.03 \leq \rho_E \leq 0.43$, respectively. No processing beyond quantization was conducted or analysed. Nevertheless, it is one of the few works, which highlight the importance of time synchronization of raw CIRs.

Similar questions were tackled by Bulenok et al. [31]. In this paper the authors also investigate the effect of quantization and temporal synchronization on key exchange. For temporal synchronization, they first evaluate a leading edge-based solution, which is deemed insufficient regarding the resulting synchronization. Hence, they apply a system based on maximizing the correlation coefficient, which is not feasible in practice as both parties need to know both input vectors. For quantization, they apply direct quantization as well as quardband-based approaches to their measurement. As both approaches yield BDR between $14\,\%$ and $41\,\%$, they proposed to concatenate the data after cropping the noise and subsequently apply a moving average filter of size $10\,\mathrm{ns}$ Thereby they reduced the BDR to "less than 10%" [31]. Finally, with their respective measurement setup, they achieved a key rate of $\approx 2.7\,\mathrm{bit/cu}$. No Information Reconciliation or Privacy Amplification was performed or evaluated.

Besides approaches where the CIRs are recorded directly, there are also tools allowing to extract channel estimates from regular WiFi Network Interface Cards (NICs). Examples are the respective tools for Intel 5300 NICs [77] or for Atheros NICs [214], which are based on custom Intel firmware and, respectively, the ath9k kernel driver. Here the channel coefficients are analysed and reported for the respective OFDM subcarriers, resulting in a CTF with low resolution [228]. It is worth noting, that in [213] a non-negligible correlation between neighbouring subcarriers was shown — hence, the authors concluded that systems using CTFs from subcarriers are inherently prone to attacks. In the following we present work based on these tools.

Table 3.2.: Overview over selected practical CSI-based solutions.

Ref.	Technology		$\mathrm{bit/s}$	$\mathrm{bit/cu}$	Communication
[133]	802.11	CTF	1.1	< 0.124	1×2
[222]	802.11	CTF	13.0	< 1.430	2×1
[237, 213]	802.11	CTF	?	< 0.600	$n + 2 \times 2$
[79]	UWB	CIR	?	?	1×2
[234]	UWB	CIR	?	< 5.000	2×2
[31]	UWB	CIR	18.0	< 2.713	1×1

A system based on the Intel 5300 NICs is presented by Zhao et al. [237], which was further analysed and evaluated in [213]. The input data for each subcarrier is treated separately with the approach proposed in [133]. Afterwards, for each subcarrier a hash is exchanged to verify if this bitstream is already equal. If this is not the case, a random subset of all bits is picked, which is expected to have a low mismatch ratio. If the bit strings of all subcarriers are unequal, IR is performed by iteratively dividing the bitstrings into blocks and comparing the respective parities. Although less communication overhead is claimed, in at least 80% of the key exchanges 10 message exchanges and more are required. In [213] the system was compared to different other approaches and their results were confirmed, e.g., for [133]. The evaluation shows, that up to $0.6\,\mathrm{bit}$ could be extracted per channel use.

A system based on the same hardware is proposed in [120]. Here, the quantization is performed by calculating the CDF over all observations and subsequently determine bins for a multi-bit quantization. They further propose a method to reduce the non-reciprocity of the legitimate observations by subtracting the Channel Gain Complement. To obtain this, the legitimate partners exchange a set of full CIRs to estimate their non-reciprocal components. Although IR and Privacy Amplification (PA) are mentioned, they are not included in the implementation.
This system comes with certain drawbacks making a practical deployment unfeasible: The statistics for the quantization are calculated with global knowledge, i.e., with all recorded observations, which are not timely available in practice. Further, exchanging full CIRs leaks this information to potential attackers — this was not discussed or evaluated at all. The subsequent learning of the non-reciprocal parts are solely a snapshot of these components — given that the channel components need to change to generate "fresh" entropy, this estimation will become outdated quickly and no update function is specified. Last but not least, an independent evaluation in [213] reports secret key rate of $1.5\,\mathrm{bit/cu}$, which is significantly below the authors claims, and additionally showed that this system is inherently attackable.

Finally, Hamida, Pierrot, and Castelluccia conducted a study regarding reciprocity of different channel features [80]. They measured 13 different CIR realizations and derived the power delay profile and the envelope thereof. The subsequent reciprocity analysis showed clearly that the CIR should be preferred over the other features. Although no actual key derivation scheme was implemented, this work still demonstrates that UWB CIRs are optimal for key derivation.

A summary of all practical approaches facilitating CSI as input material is shown in Table 3.2. It is worth noting that the effective key rate for OFDM CTF-based systems is only slightly higher than those of RSSI-based systems. The UWB CIR-based systems achieve bet-

ter key rates of up to $5\,\mathrm{bit/cu}$. Nevertheless, when compared with the theoretically derived possible key rates, these results fall well short of expectations.

3.1.3. Further CRKG Approaches

In the following we present additional approaches which apply the CRKG concept to further wireless channel properties.

Although the **phase information** of the CSI is hard to extract accurately in practice [104, 133], some authors have analysed theoretically the possibility of key generation based on phase information.

Sayeed and Perrig were one of the first to propose the usage of the channel estimates phase as input for CRKG [164]. For this, the quantization of different coherence bands is explored. A great focus is put on the theoretical derivation of the error probability with respect to the SNR. Finally, the minimum energy required for a key exchange with this system is deduced. No practical evaluation of this approach is provided.

This theoretical work was further extended in [204] by proposing a potential extraction scheme for such key material. The evaluation is focused on the probability of different output results, which is analysed theoretically in terms of the respective SNR. Simulations yield key rates of up to $10 \times 10^4\,\mathrm{bit/s}$, where samples were taken every $100\,\mu\mathrm{s}$ — hence, per channel use less than $0.001\,\mathrm{bit}$ is extracted.

A similar study analysing time division duplex (TDD) OFDM was presented in [155]. Here, the authors proposed the usage of guard band quantizers for amplitude and phase in parallel. The authors analytically derive upper and lower bounds for their key derivation schemes efficiency. By simulating the respective TDD transmission, key rates in the realm of $15 \times 10^4\,\mathrm{bit/s}$ are claimed. Again, a very high sampling frequency of $15.2\,\mathrm{MHz}$ is used, which delivers an actual efficiency of $<0.001\,\mathrm{bit/s}$.

Extending from there, Alhasanat et al. propose to combine the channel estimation and especially the phase information with different modulation approaches [2]. Thereby, different interpretations of the same signal can be realized depending on the angle of arrival. This in turn is then facilitated to thwart approaches of intelligent attackers. Finally, they analyse the effectiveness of the approach by evaluating the BDR with respect to the SNR for the legitimate partners as well as for different attackers.

Considering the effective key rates per channel use, these schemes validate the analyses in [104, 133] stating that the phase information does not deliver high quality input material for CRKG.

The CRKG principles were also applied to **Multiple Input Multiple Output (MIMO)** settings.

Wallace and Sharma build such a MIMO system and are the first to back such a solution with real world measurements. In their measurements they used antenna arrays consisting of 8 antennas. Extending from their channel quantization with guardband (CQG) presented in [191], they proposed the channel quantization alternating (CQA) scheme for quantization, which relies on the CDF of the channel properties. Further steps like IR or PA are not described. Their system achieves key rates of $7.3\,\mathrm{bit/cu}$ to $16.9\,\mathrm{bit/cu}$, implying a single antenna key rate of $\approx 2.1\,\mathrm{bit/cu}$. Unrelated to the MIMO aspects of their solution, they analysed the effect of different numbers of paths and found that an increased number of multipaths result in a higher MI between Alice and Bob.

Chen and Jensen proposed a similar scheme in [36], which was later extended in [38]. They also employed CQA for quantization and a LDPC-based IR scheme. Using this system, they achieved $\approx 12\,\mathrm{bit/cu}$ for a $2x2$ antenna system with SNR $>30\,\mathrm{dB}$.

Overall, the MIMO results seem to validate the effectiveness and performance of SISO system, by achieving comparable results scaled linear by number of antennas.

3.2. Processing Steps

In the following we present notable concept for the single different processing steps of typical CRKG systems.

3.2.1. Preprocessing

Practical implementations of Channel Reciprocity-based Key Generation commonly include a preprocessing step. At its core this step serves one of two goals: increase reciprocity or temporal decorrelation. The former addresses the entropy sources realizations at the legitimate partners. The preprocessing aims to extract with high probability the properties from the realizations that define the reciprocity of the observations. Alternatively, this goal can be achieved by removing explicitly non-reciprocal properties. The second goal, temporal decorrelation, is based on the requirement of the input data that the observations should be independent and identically distributed — especially on the property of independence. To ensure this property, attempts are made during pre-processing to minimize, if not eliminate, the correlation between successive observations.

Besides, there are different approaches to specific problems, e.g., in dependence of the input data. For example, recorded Channel Impulse Response observations do not have a defined starting point but are preceded and succeeded by a varying length of noise. A preprocessing might aim for the removal of such noise-dominated parts of the observation.

Below, we describe essential approaches proposed in the realm of preprocessing.

With focus on **reciprocity enhancement** the following approaches were proposed:

One of the first ideas is the usage of Savitzy-Golay Filters as proposed in [3]. The authors proposed to pass the sequence of RSSI values through a Savitzky-Golay filter, which acts as a low pass filter, to extract the slow-moving features of the RSSI trace. Subsequently, this filtered version of the trace is subtracted from the original one, to extract the fast changes. This approach increased the correlation coefficient between Alice and Bob by up to 4.47 percent points. Nevertheless, the filtering also reduces the KGR significantly — it drops from $0.23\ bits/s$ to as low as $0.001\ bits/s$.

Similar approaches with the main intend being smoothing of the original signal include interpolation based preprocessing [154], curve fitting [8], and discrete cosine transformations [129].

A further approach in the realm of RSSI processing is the usage of a Kalman filter to increase the reciprocity of the traces [8]. This approach was later extended to the usage of autoregression based Kalman filters [136]. The main aim is to transform the traces of RSSI values into sequences of independent Gaussian vectors. To make this approach work, buffering and blockwise processing is necessary — concretely they evaluated this method on blocks of 1000, 2000 and 4000 samples. Although the feasibility of the approach is shown, no comparison to baseline approaches is conducted. Due to the computational

complexity of $\mathcal{O}(n^3)$ for the straight forward Kalman filter, these approaches might not be suitable for resource constrained devices.

Similar approaches use wavelet transformations with different wavelets to enhance the reciprocity of RSSI traces. This method was first proposed in [232] and later studies compared different families of wavelets regarding their efficiency [5, 118]. It was shown, that Haar wavelets performed best, which increased the correlation by up to ≈ 5 percent points.

A detailed comparison of the core approaches of the described works can be found in [71].

In the area of CIR based CRKG only few works propose preprocessing.

In [31] a method to remove noise around the actual CIR is proposed. This is achieved by determine a sample within the CIR by both partners, which is then used as "starting point". For this, two methods are proposed: the first is a threshold based algorithm, where the starting point is the first occurrence passing a certain threshold. Within the paper, the authors themselves dismiss this approach due to its subpar performance. The second approach the starting sample is identified by determining a shift, which maximize the correlation between Alice and Bob's observation. Although this approach exhibits strong performance, it is not applicable in practical implementations, since the calculation of the correlation requires full knowledge of both observations.

A similar problem is approached in [130]. In order to achieve synchronization in time for each key candidate, it is proposed to exchange the first 100 bits of the quantized key material. Based on these 100 bits, the respective partner determines an offset resulting in maximal correlation. Subsequently, this offset is used to shift their own raw key material and thereby achieve synchronization. The exchanged raw bits are discarded. It is explicitly stated, that these exchanged bits can be used to approximate statistics about the key material.

To achieve the **temporal decorrelation** of successive observations, typical approaches to decorrelation are applied.

In [154] the authors propose the application of discrete Karhunen Loèven transformation (KLT), due to its decorrelating property. The transformation would provide an orthogonal basis for the input data, which could be used to decorrelate the input vectors. A major assumption for this calculation is a model of the original input's covariance. As such a model is not known for RSSI traces, the authors derive it based on their measurements — which, as the authors state, is no general model, but specific to their current setup.

This idea was later adapted in [38], where der eigenvectors of the covariance matrix are used for decorrelation. Furthermore, the theoretical foundation of this approach was analysed. Of particular interest is the analysis of the computational complexity of this approach, which is $\mathcal{O}(n^3)$.

Another approach is the usage of discrete cosine transformations (DCT) for decorrelation. This was initially proposed in [220] were the typical application of images compression is transferred to this use case. For IEEE802.11a transmissions, they interpret the matrix of the 52 subcarriers times 256 time frames as a single image. The DCT than removed the low power components from this "image" and the resulting compressed representation was used for further processing.

Similar approaches with different primitives at their core were proposed in [71]. In their analysis, KLT and DCT were compared to Haar transformations and Walsh-Hadamard transformations. The final analysis showed, that no significant difference between these

approaches can be observed.

In recent works, Principal Component Analysis (PCA) is proposed to achieve decorrelation [177]. Here, blocks of RSSI samples are passed to PCA, which in turn gives the principal components. In their implementation 200 samples are used to generate one eigenvector and only the first 4 components are correlated enough for further processing. Furthermore, the authors state that the analysis is highly computational expensive — hence, it should be conducted on an unconstrained device and the resulting eigenvector should be transmitted to the partner nodes.

No decorrelation approaches for CIR have been proposed so far.

In summary, the presented approaches achieve effectiveness to a certain degree. Nevertheless, it has to be noted, that such approaches like Kalman filtering or KLT are very computation intensive, with computational complexities of $\mathcal{O}(n^3)$ for both. Considering the intended use case of resource contained edge devices and the general aim of energy efficient key derivation, such approaches cannot be considered applicable here. Additionally, all the proposed schemes focus on RSSI traces and their direct applicability to CIR based processing is not necessarily given, since different assumptions regarding the input data need to be considered.

In the realm of CIR based CRKG only few works explicitly addressed preprocessing for enhancing the input data. Unfortunately, these come with major drawbacks: either the proposed method is not applicable in praxis as both partner's observations would be needed for the respective calculation, or the method leaks a considerable amount of key bits to the attacker and therefore has to be considered insecure.

3.2.2. Quantization

In the following we describe quantization approaches proposed in the realm of CRKG. We survey works not only with the target channel characteristic CIR in mind, but also notable works for other input data.

The simplest approach is the one bit quantization based on a fixed threshold. Here, all values above the threshold are mapped to 1 and all below to 0, or vice versa. The threshold can be determined by applying an averaging function to the current data, e.g., the arithmetic mean [31] or the median [12].

This one bit quantization was extended with a guardband in [133] as Guard Band Quantizer. The core functionality stays the same, but all samples which lie within the guardband are dropped, where the width of the guardband is determined by the standard deviation. To use the same values at Alice and Bob, a vector of indices has to be exchanged, which denotes the dropped values. The GBQ approach is probably the most widely used approach for RSSI based systems.

Extensions of this scheme were proposed in [186], [79] or [96]. All works aim for optimization of quantization thresholds by incorporating blockwise statistics like the CDF about the seen data into their calculation.

Several quantization schemes for general zero-mean Gaussian input data were proposed, e.g. channel quantization with guardband (QCG) [190], its extension channel quantization alternating (QCA) [191] or multi-bin adaptive quantization (MAQ) [154]. At their core these schemes work similar. First, different quantization maps are generated based on

known channel properties. For each observed sample, the best matching map is selected and the respective selection is stored in a vector. This vector is transmitted to the partner, so that the same quantization map is in use. The approaches differ in the creation and border selection of the quantization maps. It is worth noting, that these approaches rely on pre-known distribution of the channel properties, which are unknown in practice [228].

The CQA scheme was later generalized by [37] for multiple bins and further by [86], where the general scheme was interpreted as variant of vector quantization. The major drawback of this solution is that the choice of cell size allows either low BDR but high information leakage to an attacker, or vice versa. Both choices are suboptimal for the overall system, since either the basic functionality or the security of the process is compromised.

To the best of our knowledge only one quantization scheme was proposed which directly addressed CIRs. Zhang, Kasera, and Patwari presented the so-called *jigsaw encoding* to quantize CIRs [234]. The idea is to generate two maps of random values, where the X axis is the time scale of the CIR and the Y axis is the respective amplitude. The quantization results of the approach are picked column-wise: for every time column the content of one of the generated maps is picked. If the current X value, i.e., the current amplitude step, is below the current CIR value, the content of the first map is picked. If it is above, the content of the second map is picked. Again, this approach comes with major security concerns: first of all, the maps generated in advance are exchanged in clear to ensure that both partners have the same map content. Since the general slope of a CIR is known, this means that an adversary thereby obtains the majority of the resulting bits, just by observing the maps. Further, the scheme claims to enhance the similarity of reciprocal measurements, which is shown via lower BDR. But the lower BDR in fact originates from mapping the majority of values to fixed values due to the general CIR shape. Nevertheless, the authors did not include an adversary in their system model and, hence, no evaluation to this end is conducted.

3.2.3. Information Reconciliation

In this section we present commonly applied Information Reconciliation schemes.

In extension of the initial work on Quantum Key Derivation Bennet et al. proposed an Information Reconciliation scheme, which was later dubbed *BBBSS* after the respective authors [21]. This work first introduced a scheme to remove a certain low number of differences from two binary sequences via public discussion in the presence of an eavesdropper. The core building block is the *bisect* algorithm, which combines parity check based error detection with a divide-and-conquer strategy. Each of the binary sequences is broken into smaller blocks, while for each block the parity bits are exchanged and compared. As this algorithm obviously cannot detect all errors, it is applied within a wrapping algorithm, which performs *bisect* in several rounds and additionally permutes the input sequences to increase the success probability. Nevertheless, the *BBBSS* scheme is not guaranteed to find all errors in a given sequence, it leaks up to 100% of the input sequences and is highly interactive [92].

This scheme was later improved by the proposal of the *CASCADE* protocol [29]. *CASCADE* removed one of the major drawbacks of BBBSS — the dropping of bits to remove errors. Thereby it increased the processing efficiency of the protocol. Nevertheless, CASCADE is still computationally complex [26] and since at its core it still relies on *bisect*, the main drawbacks remained. Most notable, the full disclosure of the reconciled sequence for high error

rates and high interactivity with typically 30 rounds of bi-directional interaction.

Calver conducted an extensive study of the CASCADE protocol and showed concrete numbers for the information leakage to an attacker. Since the number of exchanged messages depends on the BDR of the input material, the leakages does as well. The author showed, that for a very low bit mismatch of $1\,\%$, CASCADE leaks $\approx 10\,\%$ of the reconciled bits. For BDRs of $10\,\%$ $\approx 64\,\%$ are leaked and for error rates of $15\,\%$ $\approx 87\,\%$ leakages occurs; For error rates above $18\,\%$ the complete reconciled bit string is leaked to the attacker [34]. Considering that the expected BDRs for CIR input material is in the range of $10\,\%$ to $20\,\%$, CASCADE has to be considered insecure for this application and should not be used.

As an alternative, the application of Error Correction Codes (ECCs) was proposed.

The simplest variant here is the direct application of an ECC to the input material [190, 121]. Here, the incoming bit string is directly decoded as a would be codeword of the respective ECC, e.g., LDPC codes [121]. As it is tried to directly decode random bit strings, this approach has a low success rate.

In extension of this, the transmission of certain side information is proposed. This side information can then be used to reduce the distance of the codeword or the resulting word at the reconciliation partner. Typical side information are the parity bits of the ECC codeword [234] or the error syndromes after decoding [104]. By combining this side information with the respective observation of the communication partner, he might shift his observation into the vicinity of the other observation and the subsequent decoding yields the same decoded word. Such schemes were proposed with a multitude of different ECCs: repetition codes [134], Hamming codes [33], Bose-Chaudhuri-Hocqenghem (BCH) codes [184, 228], Reed-Solomon (RS) codes [234], Golay codes [132], Turbo codes [149] or Low-Density-Parity-Check (LDPC) codes [27, 56]. With respect to RSSI based CRKG, the effectiveness of all these approaches is shown with slight differences regarding the achieved BDR and key generation rate.

In addition to the direct transmission of the syndrome or parity, there are also derived methods with further steps. For instance, the *Bit Mismatch Mitigation Algorithm* randomly flips up to three bits in the transmitted syndrome [174]. This is intended to diminish Eve's chance for successful reconciliation. Nevertheless, the authors themselves indicate that the transmission of the syndrome causes a non-negligible information leakage to Eve.

It is worth noting, that since parity and syndromes are calculate directly from the respective codewords, their exchange inevitably leaks information about the key material to eavesdropping attackers. Mathur et al. analysed that for the Golay code they used and showed that every parity bit transmitted corresponds to one bit leaked about the key material [132].

Special applications of ECCs are *Secure Sketch* schemes and *Fuzzy Extractors* based on Secure Sketches as proposed by Dodis, Reyzin, and Smith [49]. The core idea is to apply a fuzzy commitment scheme like the Jules-Wattenberg commitment scheme [100] to the key candidate. Thereby the transmission of the key material itself is intended to be cloaked and, therefore, protected from eavesdroppers. This scheme was used in practice in e.g., [15].

To reconcile two realizations of bit strings at Alice and Bob, i.e., W_A and W_B, the scheme would work as follows: The scheme builds on an ECC with encoding function $enc()$ and decoding function $dec()$. Assuming that Alice is the leader of the scheme, she picks a random number x from the source alphabet of the ECC used. This x is encoded with the ECC and

XORed to W_A to produce the *Secure Sketch* $SS(W_A) = enc(x) \oplus W_A$. This sketch is then transmitted to Bob. Bob proceeds by XORing his own observation with the sketch $SS' = SS \oplus W_B$ and subsequently decodes the result SS'. This yields $x' = dec(enc(x) \oplus W_A \oplus W_B)$.

The core assumption for effectiveness is that the Hamming distance between W_A and W_B is smaller than the error correction capabilities of the ECC. Then $W_A \oplus W_B$ yields an error for $enc(x)$ which is correctable and, hence, $x' = x$. If this is not given, the scheme will not work. Likewise, the security relies on the Hamming distance between W_A and W_E, i.e., Eves bit string, is larger the error correction capability.

Currently, no analysis regarding effectiveness and security of *Secure Sketches* using real world CIR data is available. Further, it is worth noting, that with this scheme not the physical observations W_A and W_B are used for further key generation, but the chosen x — the values W_A and W_B are only used to "hide" the value of x in the fuzzy commitment.

3.3. Attacks

Attacks against CRKG can be divided into two categories: first, *jamming* attacks which primarily aim to obstruct legitimate key exchange and thus either prevent successful key generation or force it into a predetermined way. And secondly, *key recovery* attacks that aim to derive the same key as the legitimate partners. The majority of the attacks presented are against RSSI based systems, being simpler for the attacker due to the lower level of detail of the key material. A classification of different attacks is presented in [229].

The first kind of attacks aiming for direct interference with the wireless transmission between Alice and Bob are primarily realized via jamming approaches. The simplest approach here is constant jamming acting as Denial of Service (DoS) attack with the sole aim of prohibiting the key exchange [222, 227, 52, 139, 140]. Zafer, Agrawal, and Srivatsa showed in their theoretical analysis that the effective key rate is rapidly declining with adversarial signal power [227].

Several approaches construct dedicated jamming schemes which aim for a direct influence of the key exchange [52, 98, 97]. By coordinating the jamming with the probing phase of CRKG, the attacker can force the values of single bits in the preliminary key material. With such a reactive jamming scheme Eberz et al. attacked the CRKG scheme of Mathur et al. Using real world measurements, they could force the key material to be predictable and could thereby recover up to $47.4\,\%$ of the legitimate key [52]. Similar approaches are so-called *session hijacking* attacks, where the attacker injects high power signals to force the values of the key material [88]. It is worth noting that such reactive jamming approach will not work against CIR based CRKG — the jamming would here alter the whole CIR vector, which makes the dedicated change of single bits infeasible.

Studies of key recovery attacks cover a wide range of sub-topics from analyses of eavesdropper correlation to active side channel attacks.

Several analyses of eavesdropper correlation have been conducted [53, 231, 82, 54], primarily targeting the assumption of uncorrelated observations for distances $> \frac{\lambda}{2}$. Edman, Kiayias, and Yener studied the correlation of eavesdropper observations for locations $2.0, 5.0$ and 7.5 wavelengths away from a legitimate node [53]. They showed that a passive eavesdropper's RSSI values reach correlations up to 0.9, indicating that the $\frac{\lambda}{2}$ assumption does not hold in practice. Zenger et al. investigate the correlation of RSSI measurements in dependence of the eavesdropper's distance [231]. The authors showed, that there are

still notable correlation for distances greater than half a wavelength and noted, that the respective channel models need to be adapted for CRKG applications. These results are confirmed by similar studies, e.g., [82, 54].

The usage of multiple antennas at the attacker can significantly improve the attack's success chance. Steinmetzer, Schulz, and Hollick investigated the achievable secrecy rate of the CRKG system proposed in [70] with attackers using up to four antennas [180]. They showed that the usage of a second antenna effectively halves the secrecy rate of the system and that it drops to zero for four antennas. In the same vein, practical attacks against PLS encryption schemes were developed and shown to be practically feasible [168, 167]. Similar results were obtained by a theoretical study for colluding attackers [205]. The respective simulations show with increasing number of colluding attackers, the achievable conditional min-entropy of Alice and Bob is significantly reduced.

There are also side channel attacks against CRKG, though side channel is defined relatively broadly here. The most notable side channel attack is the so-called *re-radiation* as proposed in [50]. Here, the reflections of a receiving antenna are facilitated to learn about the legitimate partners communication. Döttling et al. performed experiments in an anechoic chamber and showed that re-radiations are measurable [50]. In a theoretical analysis, [189] verified these results and further derived that an attacker, given full CSI and high SNRs, can reduce the effectively achievable key rate to zero. Another side channel attack is the *repetition attack* [229], where an attacker mimics the movements of Alice and Bob to derive the same key. The practical evaluation shows low success probability of this attack, with a maximum correlation of 0.7 for the attack. Literal side channel attacks are attacks against IR phase messages and their information leakage, which are presented in Section 3.2.3.

A notable subtopic is the class of *predictable channel attacks* which was already proposed in the early works regarding CRKG by Jana et al. [96]. This approach relies on the core idea that the channel between the legitimate partners can be predicted or inferred by an adversary, with or without additional knowledge. Jana et al. already showed that the majority of RSSI based systems are prone to this attack [96]. This approach was further investigated in [157]: here, the authors showed that via targeted manipulation of the transmission environment, e.g., deliberately blocking the LOS, predictable channel attacks are feasible against RSSI based CRKG. Several works claim that CSI based CRKG is not prone to predictable channel attacks due to the higher detail level of the channel properties and the complexity it entails [120, 157, 152].

To our knowledge, no work combines the working mechanisms of different attacks to achieve higher attack effectiveness.

It is important to note here that the previous attacks were directed exclusively against RSSI-based systems. Due to the significantly higher complexity of the target, attacks against CIR systems are rare.

Notable works are the *room reconstruction* attack of Döttling et al. [50] and the ray tracing attacks by Ben Hamida et al. [20] and Madiseh [123].

Döttling et al. proposed the so called *room reconstruction* attack [50] against CIRs. The idea is that an adversary can reconstruct reflector positions, i.e., the room, based on timing difference of her own observations. Given the time delay of a multipath component, which represents the respective time of flight, and the position of Alice, Eve can derive an ellipsis, on which the respective reflector has to reside. Combining this information with the same

from Bob, Eve can construct two ellipses with potential reflector positions for each multipath cluster. The actual reflector is then assumed to reside at the intersection of these two ellipses. This attack is presented more as a theoretical construct and only backed up with highly simplified simulations, as it is based on strong assumptions. Especially robustly extracting the single multipath clusters and matching them accordingly between the Alice and Bob observation seems problematic. Nevertheless, this approach shows the capabilities of a technically advanced adversary.

Ben Hamida et al. as well as Madiseh both presented ray tracing attacks against CIR based CRKG [20, 123]. Both assume that the room geometry and the transmitter positions of the legitimate partners are known to the attacker. The attacker then tries to predict the current CIR for this setting by using a ray tracing tool like WISE [61]. In [20] the 30 most significant multipath clusters were considered. Thereby, the ray traced CIRs could achieve cross correlations to the legitimate one up to 0.7. Considering the whole CIR, the cross correlation only reaches 0.44. In [123], with respect to their specific processing, the author claim that ray tracing produces results en par with complete channel state information. It is worth noting, that although the authors call this approach ray tracing, they are actually using the Saleh-Valenzuela model [163] to solely calculate the single cluster decays. For the actual processing, they start with the CIR measured by Alice and Bob and extract the first path of each multipath cluster. Then, the Saleh-Valenzuela model is used to re-calculate the expected decay for the clusters; the result is then treated as attacker prediction. Given this specific processing, this cannot be considered a realistic or practical attack.

To our knowledge, these are the only direct inference attacks against CIR based CRKG. However, these works are only theoretical approaches or use small actual sample sizes, hence the respective practical implications are hard to qualify. Further, none of the existing works leverage the capabilities of machine learning approaches to attack CRKG or CIRs themselves.

3.4. Adjacent Topics

The source model of key distribution works with a multitude of shared entropy sources and is not necessarily bound to reciprocal wireless transmissions. As long as the participating legitimate partners have access to highly correlated realizations of the common source, the key derivation might work. In the following we present further common sources of entropy used for key generation as described above.

Mathur et al. use wireless radio or TV broadcast as general input, where they employ a TV signal ($584.34\,\mathrm{MHz}$) and an FM signal ($88.7\,\mathrm{MHz}$) [132]. Given that A and B are close, $< 0.1\lambda$ (5.1cm and 33.79cm, respectively), and Eve is further than 0.4λ away from them, they can use the signal changes as input for key derivation. They use direct median-based 1bit quantization or alternatively a scheme encoding maxima/minima into index lists. For reconciliation, they use perfect binary Golay-Codes in an offset-based decoding approach (direct decoding transmit error). They achieve a KGR of $\approx 4\,\mathrm{bit/s}$, with a BDR of 10^{-4}.

Different systems also use *acoustic input data* as input for the key derivation:

In [122] the *FREE* system is proposed. Here, the core principles like channel reciprocity and spatial decorrelation are adapted for sound waves. The respective sound channel is modelled analogously to the wireless channel and it is shown, that acoustic data is also a usable input. The processing itself does not provide new approaches: quantization is

realized via the ASBG quantizer, which realizes a guard-band approach. For reconciliation, they employ a direct ECC approach, with exchange of the syndromes. Although they reach KGR of up to $250\,\mathrm{bit/s}$, the approach only works for distances $<1\,\mathrm{m}$ — for greater distances the BDR between Alice and Bob quickly reaches 0.5, making the system unusable. Further, with additional sources of noise closer than $6\,\mathrm{m}$ to either Alice or Bob, their BDR is also 0.5.

Bala and Raman follow the same approach of using ambient noise for secure key generation [16]. They employ guard-band based quantizer and CASCADE as reconciliation approach. Using experimental data they claim key rates of $\approx 80\,\mathrm{bit/s}$ with a BDR of 0.25. They further propose an echo based approach in analogy to CIR based solutions, where the echoes are the multipath clusters — here, they reach KGR of $8\,\mathrm{bit/s}$ and a BDR of 0.05.

Another research direction proposes the application of CRKG principles in the context of *powerline transmissions*. With the *VoltKey* approach, the noise of the powerline transmission is used as source of randomness [112]. This means, noise within the powerline communication itself, originating e.g., from fluctuation in the power grid, is used as reciprocal input. Again, the core principles of CRKG are used: quantization is based on Mathur's guard band quantizer and an offset-based ECC information reconciliation using Golay codes is employed. The feasibility of the setup is shown successfully.

A survey covering approaches employing *powerline communication* is given in [156].

Finally, different approaches in the realm of Body Area Networks have been proposed.

In [216] the usage of electricity generated by human muscle contraction is proposed. Here, the shared source of randomness is an electromyogram (EMG) of the muscles in the wrist and forearm of the user. Slight movements of the fingers, hand or forearm induce the necessary randomness. The EMG graph is quantized using by detecting turning points, i.e., the change between rises and falls. Subsequently, the employ an ECC based IR using Golay codes. This approach yields a KGR of $5.51\,\mathrm{bit/s}$, but only generates matching keys in $88.84\,\%$ of the cases.

Another input source can be the person's heart beat measured by piezo sensor [117]. In their work, they also analysed the usage of ECC codes for reconciliation by using a (15,3) Reed Solomon code. Although the ECC approach was not successful, the overall architecture could generate matching keys with a probability of 95.6%.

Further approaches in the realm of Body Area Networks employ gait recognition [169, 183, 211], further heart beat measurements [160, 87] or accelerometer data [103, 135].

3.5. Summary

From the perspective of the present work, the current state of the art is as follows:

There is a multitude of approaches leveraging RSSI measurement as key material and only few practical CIR based systems have been proposed. Nevertheless, these systems clearly demonstrate that CIRs can deliver much more entropy for key generation — this is expressed by a significantly larger effective KGR. Due to the high resolution of the respective CIR measurements, UWB based system are especially effective. Given the fact that tools are available to extract CSI from NICs of common technologies like IEEE 802.11 and that current COTS start to incorporate UWB transceiver, considerably more focus should be given to UWB CIR based CRKG development.

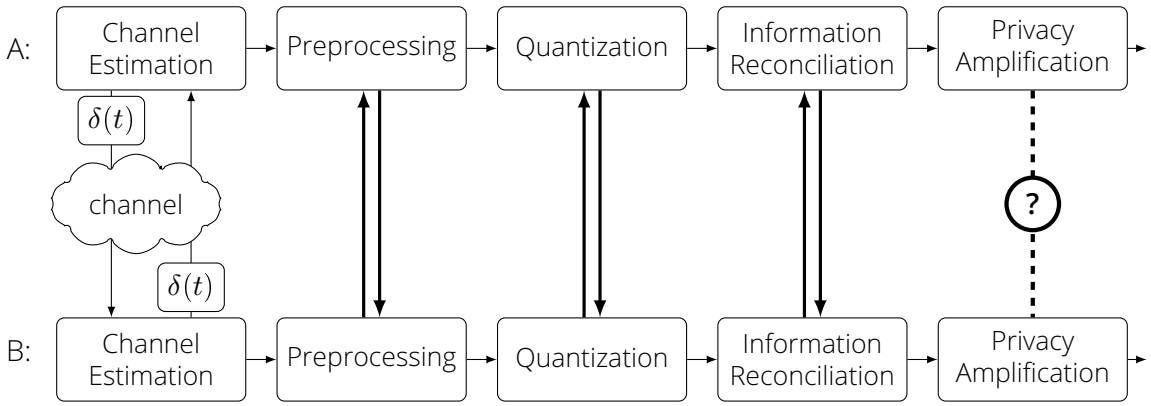


Figure 3.1.: CRKG processing steps at the legitimate nodes A and B following current state of the art.

Of the few current CIR based solutions, only one takes the special shape of CIRs into account. Considering the quantization phase, this yields two major drawbacks: first, by applying threshold based approaches to CIRs a biased outcome is generated. And second, neglecting the CIR shape inevitably causes information to be lost, reducing the overall effectiveness of the procedures. Therefore, the first step of CIR based CRKG, i.e., either preprocessing or quantization, should definitely take into account the special form of CIRs, such as described in [163].

The currently proposed solutions address all individual steps of CRKG from preprocessing to IR. However, the solutions proposed here are primarily accompanied by their own information exchange and, thereby, communication overhead. As a result, current CRKG solution work as shown in Fig. 3.1, in contrast to the originally envisioned processing depicted in Fig. 2.5. This additional communication effort comes with distinct disadvantages for CRKG: First of all, information exchanges always leak information to potential attackers — this not only degrades the basic security of the system, but also reduces the achievable key rate, since the achievable secret key capacity C_s is upper bounded by the information leak to an adversary (see Eq. (2.1)). Furthermore, communication slows down the protocol itself, since every exchange is accompanied by the corresponding round trip times. Considering the intended application scenario of resource-constrained IoT devices, communication is also disadvantageous due to the comparatively high energy consumption for such devices. Sending and receiving transmission consume up to $10x$ the power of typical operation [45]. Finally, information exchange requires an additional, authenticated communication channel itself. Summarizing, due to the detrimental effects of intensive information exchange, excessive communication should be avoided and the overall communication exchanges must be reduced as much as possible.

Further, the current approaches do not take recent developments of adjacent fields into account. Most notably the effectiveness of ML approaches are not yet considered in the CRKG domain.

Last but not least, state-of-the-art approaches subordinate the security aspects of their respective solutions to efficiency. In fact, most works do not evaluate their proposed systems with respect to known attack vectors such as *predictable channel* attacks for RSSI systems. Hence, all the points mentioned up to here must be equally considered from an attacker's point of view, in order to investigate their true capabilities. Only by including all these attacker perspectives a newly proposed system can be effectively evaluated for its

security.

In consequence, we identified the following aspects, which we address in this work: First, we want to enable the shift towards CIR based CRKG. For this, we intend to leverage the very nature of CIRs and extract and use the rich channel contained in them. Additionally, we analyse how and to which extent the communication between the legitimate partners can be reduced, to increase the performance and efficiency of CRKG. Besides "classical" approaches, we further investigate how CRKG can benefit from ML capabilities. This is not only relevant in the system design, but also for potential attackers — hence, we also incorporate ML primitive in attacks against CRKG system. Finally, we want to enable a secure design of CRKG approaches by comparing them to known and newly designed attacks.

Part II.

Measurements

4. Data Sets

Parts of this Chapter are published in [200, 193, 197, 194, 198, 196, 195].

This chapter describes the real world measurement campaigns conducted in the scope of this thesis. Four different sets of real world measurements were created. Following the system model defined in Sec. 2.4, all measurements are recordings of Ultra Wideband Channel Impulse Responses, taken in typical indoor communication setups. Each setup consists of two legitimate terminals and at least one eavesdropper.

The following sections describe the four measurement setups:

- The first data set, dubbed *scenarios*, is designed to capture a variety of different communication setups. For this, seven different constellations of the three terminals in an office space are set up and used for measurements.
- The second set, called *longterm*, aims for a larger data acquisition to generate more meaningful evaluations. The setup is a static one, in which environmental changes are mainly induced by movements of third objects.
- The third data set, *attack*, is designed to facilitate the deterministic attacks. Hence, it is the first to combine the CIR realization with concrete location information.
- The fourth and final data set, called *robot* combines the merits of the former ones into a final realistic setting. Here, an autonomous robot moves in an office space employing different settings and configurations while recording CIRs along with location information. This set exceeds all former ones regarding the total number of measurements with $\approx 1.5 \times 10^6$ recordings.

The data sets *scenarios*, *longterm* and *attack* were designed and realized in cooperation with the Center of Excellence CoSa (Communications - Systems - Applications) of the Lübeck University of Applied Sciences.

4.1. Data Set: Scenarios

In the following we describe the data set *scenarios*.

4.1.1. Rationale

The core idea of the *scenarios* data set is to provide a range of different communication settings, which are used to evaluate the general feasibility of different schemes as well as the influence of the respective settings. Hence, the measurements are taken in a typical environment for CRKG, i.e., an indoor office space.

To enable the usage with different interpretations, no roles were predetermined for each terminal. Hence, the respective measurements could be either used for the legitimate partners or for the eavesdropper.

4.1.2. Realization

The **measurement setup** consisted of Tektronix DPO72304DX oscilloscope[1] with three antennas and a Tektronix AWG70002A signal generator[2]. All three oscilloscope antennas were treated equally, i.e., none was explicitly interpreted as Alice, Bob or Eve. The measurement were conducted as Impulse Radio UWB at a baseband of $4\,\mathrm{GHz}$ and with a bandwidth of $500\,\mathrm{MHz}$, which satisfies UWB requirements. The oscilloscope recorded the CIRs with a resolution of $1\,\mathrm{ns}$, which corresponds to a spatial resolution of $30\,\mathrm{cm}$.

To excite the wireless system and initiate the channel estimation, the signal generator sent a short time domain impulse. This pulse is designed in compliance to the IEEE 802.15a channel specification [162]. It has a duration of $3\,\mathrm{ns}$ and is constructed of a set of ternary symbols, which exhibit perfect autocorrelation properties — thereby, the CIR can be identified as peaks in the periodic correlation sequence.

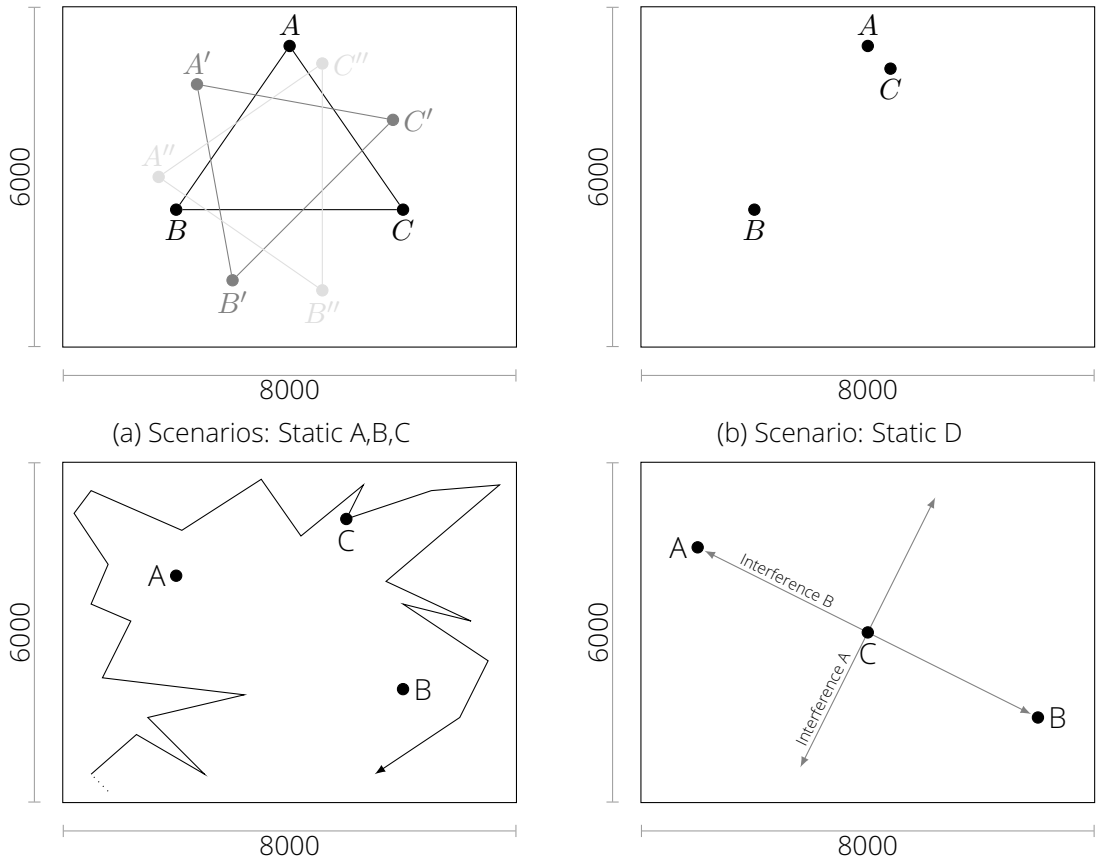
The **measurement environment** is an indoor office room, which is $8\,\mathrm{m}$ long and $6\,\mathrm{m}$ wide. Within this room the following seven scenarios were realized (see also Fig. 4.1):

Static A,B,C In this scenario, the three terminals form an equilateral triangle with side length $4\,\mathrm{m}$ within the room. The respective variants A, B, and C correspond to rotations of this triangle by angles of $0°$, $45°$ and $90°$, respectively. A schematic view of this scenario is shown in Fig. 4.1a.

Static D In this scenario, two terminals reside close to each other as shown in Fig. 4.1b. The distance between the two nodes is $\approx 30\,\mathrm{cm}$, which corresponds with the spatial resolution of the measurements. This is to investigate whether a short spatial distance of the attacker gives her an advantage.

Moving Eve Here, one of the terminals is moving randomly in the room without crossing the LOS of the other two terminals. The speed of its movement is $\approx 20\,\mathrm{cm/s}$. Although the scenario is called *Moving Eve*, the moving terminal could be interpreted as any of the participants. Fig. 4.1c depicts this scenario.

[1] https://de.tek.com/oscilloscope/dpo70000-mso70000
[2] https://de.tek.com/signal-generator/awg70000-arbitrary-waveform-generator



(a) Scenarios: Static A,B,C

(b) Scenario: Static D

(c) Scenario: Moving Eve

(d) Scenarios: Interference A,B

Figure 4.1.: Schematic floor plans of the different setups in the *scenarios* data set. All lengths are given in mm.

Interference A,B In these scenarios one terminal is moving in a predetermined fashion: In case A, it is moving back and forth on the line connecting the other two nodes, i.e., their LOS. In case B, it is moving perpendicular to the center of the other nodes LOS. Again, the speed of its movement is $\approx 20\,\mathrm{cm/s}$. It is expected, that the inferring movements will reduce the similarity of the legitimate terminals CIR realizations. This case is depicted in Fig. 4.1d

Within this environment the following **measurement procedure** was conducted. For each measurement, the signal generator produces a signal which is sent via one of the antennas. The respective CIRs is recorded at the other two antennas. To capture the full set of channel pairs, each antenna is used as sender once:

- Antenna A sends an impulse, which is recorded at antenna B and C
- Antenna B sends an impulse, which is recorded at antenna A and C
- Antenna C sends an impulse, which is recorded at antenna A and B

Thereby, the setup records CIRs for all possible reciprocal antenna pairs, i.e., A-B, A-C and B-C. Obtaining one set of measurements, i.e., the six recordings as described above, takes $\approx 90\,\mathrm{ms}$. Afterwards, the system has to process and store the acquired data and reset the internal state, which takes $\approx 100\,\mathrm{ms}$. In combination, every $200\,\mathrm{ms}$ a set of CIRs can be obtained. The general acquisition schema for this data set is visualized in Fig. 4.2.

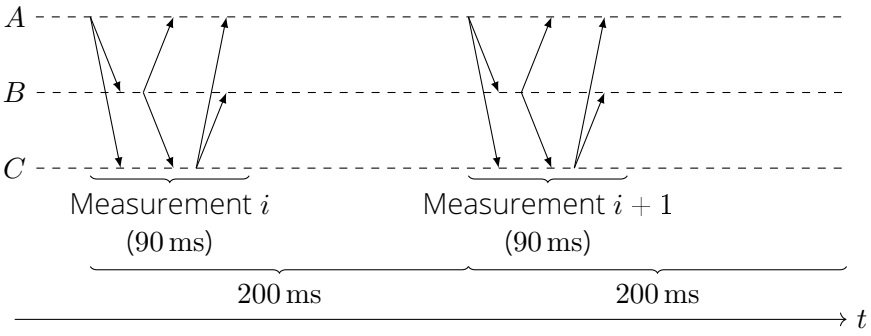


Figure 4.2.: The general measurement procedure for data set *scenarios* (also applied for data set *longterm*).

The internal processing of the CIRs includes a leading edge detection, which removes noise *before* the actual CIR. The length of the CIR itself is fixed to 200 samples, i.e., $200\,\mathrm{ns}$. Considering the room size, the maximum distance which can be travelled by the wave is $\sqrt{(6\,m)^2 + (8\,m)^2 + (2.5\,m)^2} \approx 10.3\,m$. Given the speed of light c, the electromagnetic wave travels this distance in $10.3\,m/c \approx 34.38\,ns$. Hence, $200\,\mathrm{ns}$ allows for approximately six reflections in this room. Considering that the amplitude and thereby the impact of the multipath clusters decreases exponentially, we consider six reflections to be sufficient.

4.1.3. Resulting Data

An example of the recorded data in the time domain is plotted in Fig. 4.3.

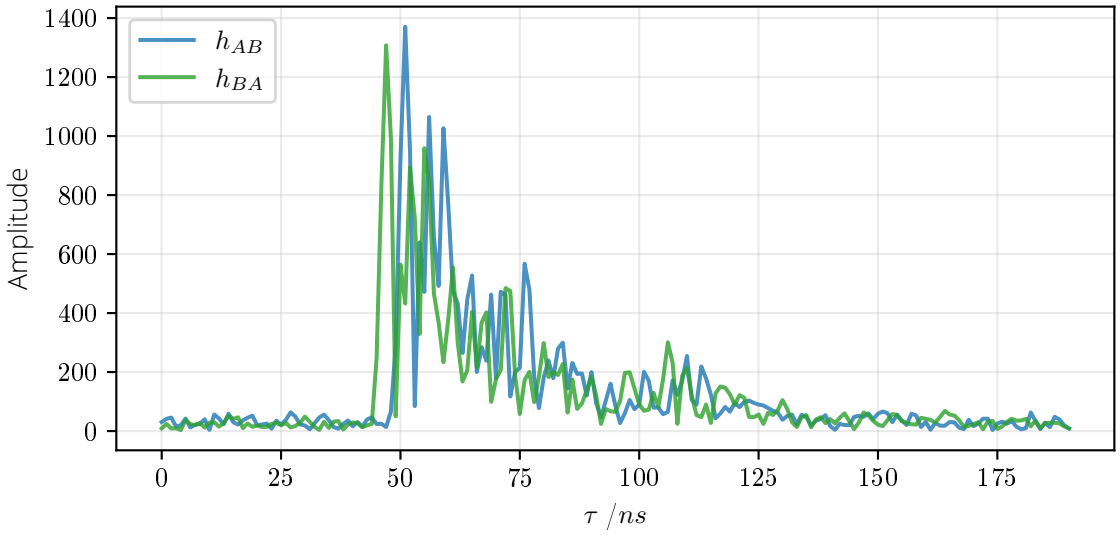


Figure 4.3.: Example record of the data set *scenarios*.

For each scenario, 201 measurements were taken, each consisting of 6 CIRs or respectively, 3 reciprocal pairs and a corresponding "overheard" CIR. Hence, in each setting, there are 603 pairs and overall 4221 pairs (plus the respective observation at Eve). The CIRs are taken every $200\,\mathrm{ms}$, thus providing five CIRs per second.

4.2. Data Set: Longterm

In the following we describe the data set *longterm*.

4.2.1. Rationale

The main purpose of the *longterm* data set is to verify the findings of the set *scenarios* with more extensive measurements. The former data set, intended for proof-of-concept and general feasibility studies, only provides few data per scenario. Hence, the respective evaluations might not be generally applicable. Therefore, we created another data set with the same hardware to have the possibility to verify the findings of the *scenarios* data set.

4.2.2. Realization

The **measurement's setup** as well as the **measurement's procedure** is the same as with the *scenarios* data set. That is, the same signal generator and the same oscilloscope were used and the measurements were performed in the same sequence per measurement point.

Different from the previous setup is the **measurement environment**. In general, a long term measurement in the CRKG context only provides useful data, if there are changes in the channel. A long term measurement of a static channel would yield slight variations of the respective channel, without any "fresh" entropy for key generation. Hence, the measurement environment needs to change for varying realization of the CIRs.

For the actual setup, including oscilloscope, signal generator and wired antennas, autonomous movements were not realizable. Hence, the movement scenarios of the former data set could not be realized as long term measurement.

As an alternative, we set up the measurement hardware in an environment which contained changes through movements in the environment itself. Concretely, the setup was installed in a busy hallway, where people crossing the respective LOSs cause interferences in the wireless propagation. The setup of this measurement as realized by our partners is depicted in Fig. 4.4.

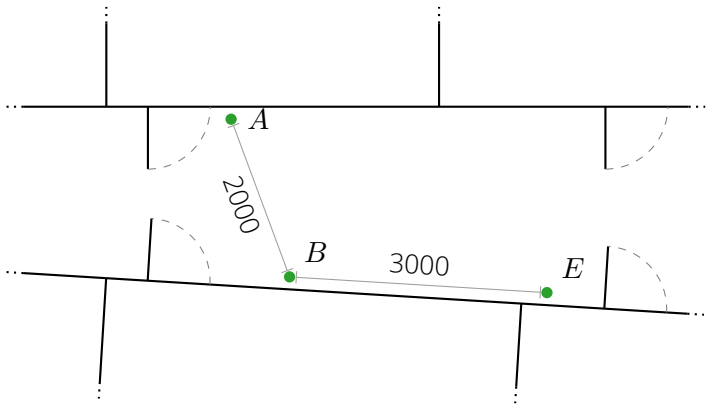


Figure 4.4.: Measurement environment for the *longterm* data set. All length are given in mm.

4.2.3. Resulting Data

An example of the recorded data in the time domain is plotted in Fig. 4.5. Given that the same hardware was used, the resulting data is similar to those of data set *scenarios*.

Differences are a slightly more pronounced LOS component and fewer multipath components. This is mainly due to the open ends of the hallway — here reflections have fewer possibilities to "return" to the respective terminals.

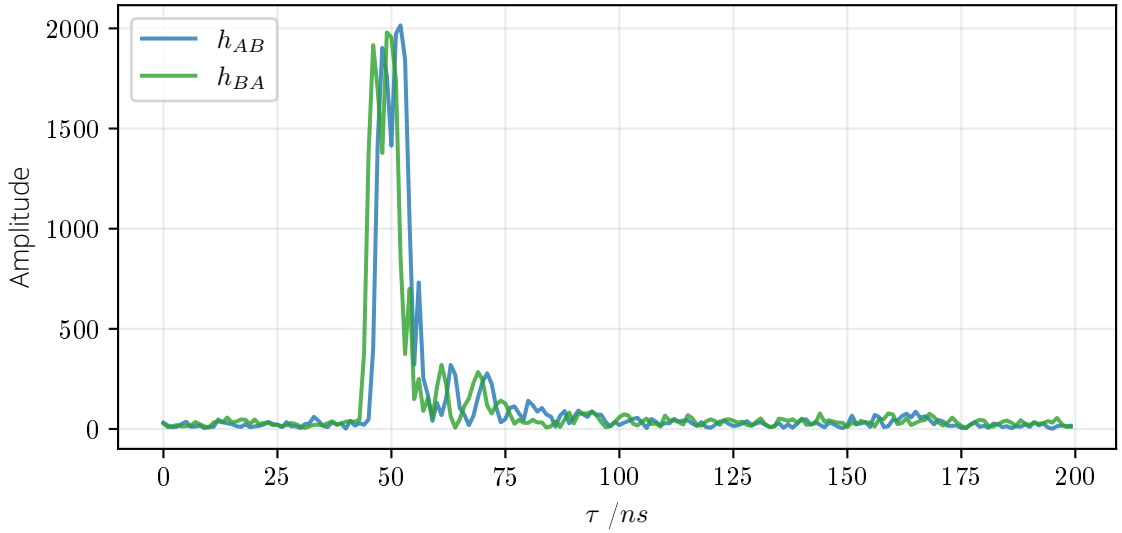


Figure 4.5.: Example record of the data set *longterm*.

Overall, the *longterm* measurement ran for one work day, approximately from 8:30 to 18:15. In this time, 175 005 CIR pairs were recorded, combined with the respective observations at the eavesdropper. As the same hardware setup was used, again five CIRs per second were recorded.

4.3. Data Set: Attack

In the following we describe the data set *attack*.

4.3.1. Rationale

As the name suggests, the main driver for this data set is the intended usage for attacking scenarios. One of the core ideas is to pre-calculate the resulting CIR of the legitimate partners only by using knowledge about their terminal positions and about the current environment. For such an attack, measurements of the legitimate partners channel properties need to be taken, in combination with their current position within the environment.

This data set is intended to be facilitated for proof-of-concept realizations of such deterministic attacks. For this reason, only a limited number of measurements were performed.

Additional to the attack idea, this data set also prepares the transition to realistic end user hardware. The former two data sets were both acquired with high-end signal processing hardware. This is justified by the intended use for proof-of-concept and feasibility studies. Nevertheless, as the final key derivation should be feasible on consumer hardware, the corresponding measurements must also be performed with such hardware. This data set is a first step towards this goal since it employs hardware designed for and used in consumer products.

4.3.2. Realization

The **measurement setup** is composed of two separate DecaWave EVB1000 UWB development boards[3]. These boards consist of the DecaWave DW1000 wireless transceiver accompanied by an ARM microprocessor, STM32F105[4]. The DW1000 itself is a fully integrated UWB transceiver compliant to the IEEE 802.15.4-2011 specification [93].

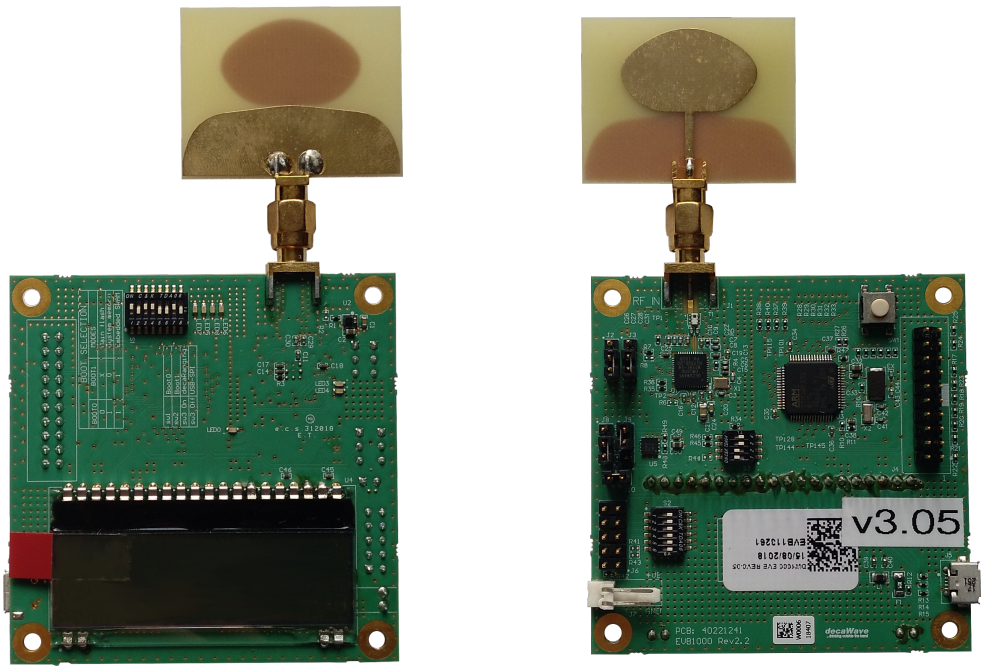
The DW1000 chip on the EVB1000 is calibrated to its channel 2, which corresponds to a center frequency of $3.9936\,\mathrm{GHz}$ and a $499.2\,\mathrm{MHz}$ bandwidth. For debugging purposes the DW1000 stores the internal CIR estimation in an accessible memory register. This estimate is at most 1016 samples long and is recorded with a sample interval of $T_S = 1\,\mathrm{ns}$. Each tap is recorded as complex values, more precisely a 16-bit real integer and a 16-bit imaginary integer [47].

To record the respective CIR, the DW1000 chip generates a short time domain pulse about $2.5\,\mathrm{ns}$ in duration [78]. The shape of the pulse is defined by the IEEE 802.15.4 standard [93]; the actual realization is depicted in Fig. 13 of the datasheet [45]. This short pulse is then transmitted to the respective receiver. As the receiver knows the transmitted pulse, it can use this to estimate its respective CIR. The result of this estimation is subsequently stored in the debugging register `ACC_MEM`.

The EVB1000 itself is shown in Fig. 4.6.

The following **measurement environment** was realized to record CIRs with associated position data: within a cuboid room, 12 positions were defined. First, the transmitting terminal is fixed in position I. The first measurement was with the receiving terminal at position

[3] https://www.ecawave.com/product/evk1000-evaluation-kit/
[4] https://www.st.com/en/microcontrollers-microprocessors/stm32f105-107.html



(a) front (b) back

Figure 4.6.: The EVB1000 evaluation boards used for the data sets *attack* and *robot*.

XII. Then, the receiving terminal moves in a quarter circle around the transmitter over positions II-XI. Finally, the transmitter is set to positions II, IV and VI and for each position measurements were conducted with points XI and XII. Throughout the evaluation, these measurements are numbered sequentially. The measurement setup and the respective measurement runs are depicted in Fig. 4.7.

The **measurement procedure** for this data set is straightforward: After setting up the respective measurement setup manually, the transmitting EVB1000 boards sent impulses every $200\,\mathrm{ms}$. The corresponding receiver boards recorded the CIRs, where the whole content of the `ACC_MEM` register is stored, i.e., all 1016 samples. It was intended to record at least 1000 CIRs per measurement run. With respect to the intended use case, no movement or intentional interferences are performed or induced.

4.3.3. Resulting Data

An example of the recorded data in the time domain is plotted in Fig. 4.8. Beside the example data itself, this figure also visualizes the transition from the high-end oscilloscope to consumer hardware: no internal noise removal like on the oscilloscope was done, which is clearly visible by the noise before the actual CIR.

For each of the 17 described pairs, a set of CIRs was recorded, where each set consists of at least 1000 CIR realizations. Due to the manual processing, the actual set sizes vary from 1014 to 1112 realizations. Within these sets, each single CIR consists of 1016 samples The CIRs are taken every $200\,\mathrm{ms}$, thus providing five CIRs per second.

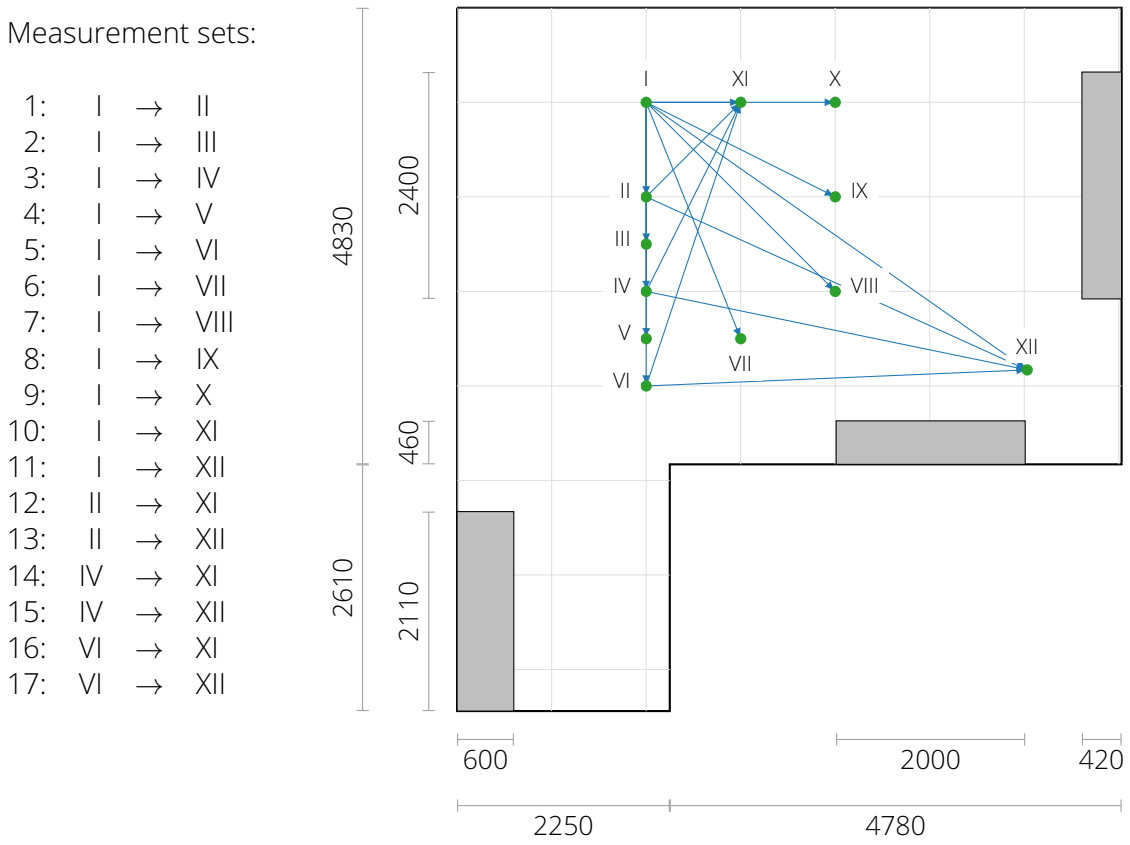


Figure 4.7.: Measurement environment for the *attack* data set. All distances are given in mm. The grid step size is 1 m.

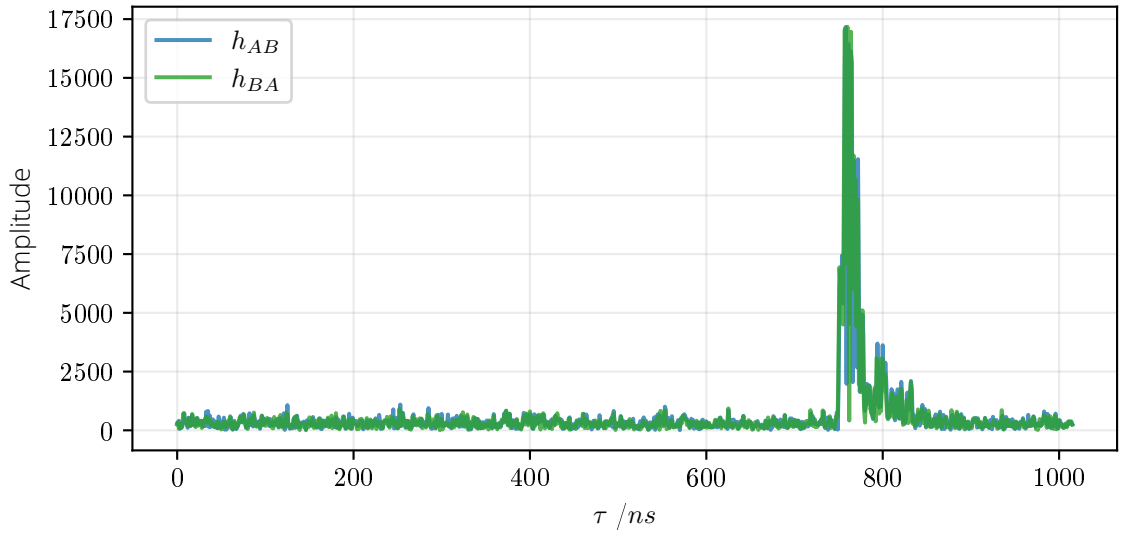


Figure 4.8.: Example record of the data set *attack*.

4.4. Data Set: Robot

In the following we describe the last data set *robot*. This dataset was released under open access [195].

4.4.1. Rationale

This data set is intended to integrate the useful features of the former sets, e.g., simultaneous eavesdropper realizations and position information, while eradicating the drawbacks like manual terminal movements.

The main design goals for this data set are:

- Autonomous movement, independent of room geometry
- Longterm measurements possible
- Simultaneous measurements of the legitimate partners as well as the eavesdropper[5]
- No fixed interpretation of recorded CIR, i.e., no predefined roles.
- Measurements with consumer hardware
- Adaptable parameters

The goal that all terminals should be interpretable as any role, combined with the goal of COTS equipment usage, also means that the attacker uses such technology. We accept this apparent weakening of the attacker, since this in turn makes the respective attacks per se stronger.

To achieve all of these goals we build an autonomous robot platform which is used to semi-deterministically move COTS measurement hardware through the determined measurement environment.

4.4.2. Realization

The **measurement setup** itself consists of three main parts: (1), the hardware used to obtain the actual CIR realizations, (2), the autonomous robot moving one of the terminals within the environment and (3), the hardware facilitating the aggregation of the different measurements.

The first part (1) is realized with the same EVK1000 hardware as used for the data set *attack*. Here, we expanded the setup by two additional boards, to a total of 4, to achieve two goals: first, the additional nodes enable the simultaneous recording of multiple CIRs, which can be interpreted as, e.g., multiple eavesdroppers, and second, with four nodes the EVK1000 boards can autonomously perform localization of the moving terminal and thereby provide position information. For the latter, the EVK1000 were set up in a positioning setup as described in the DecaWave manual. Here, three nodes are at fixed positions (*anchors*) and one is moving in the room (*tag*). The detailed implementation of this process is described below in *measurement procedure*.

The second part (2), the autonomous movements, is realized with a Lego MINDSTORMS EV3 set[6]. The core components of the set are used to build a driving robot platform as depicted in Fig. 4.9.

The main components used in this robot are: the EV3 brick itself, a gyroscope, an ultrasonic sensor and two touch sensors. The EV3 brick controls the movements of the robot

[5] Simultaneous in the sense that they fall within the coherence time.
[6] https://www.lego.com/de-de/product/lego-mindstorms-ev3-31313

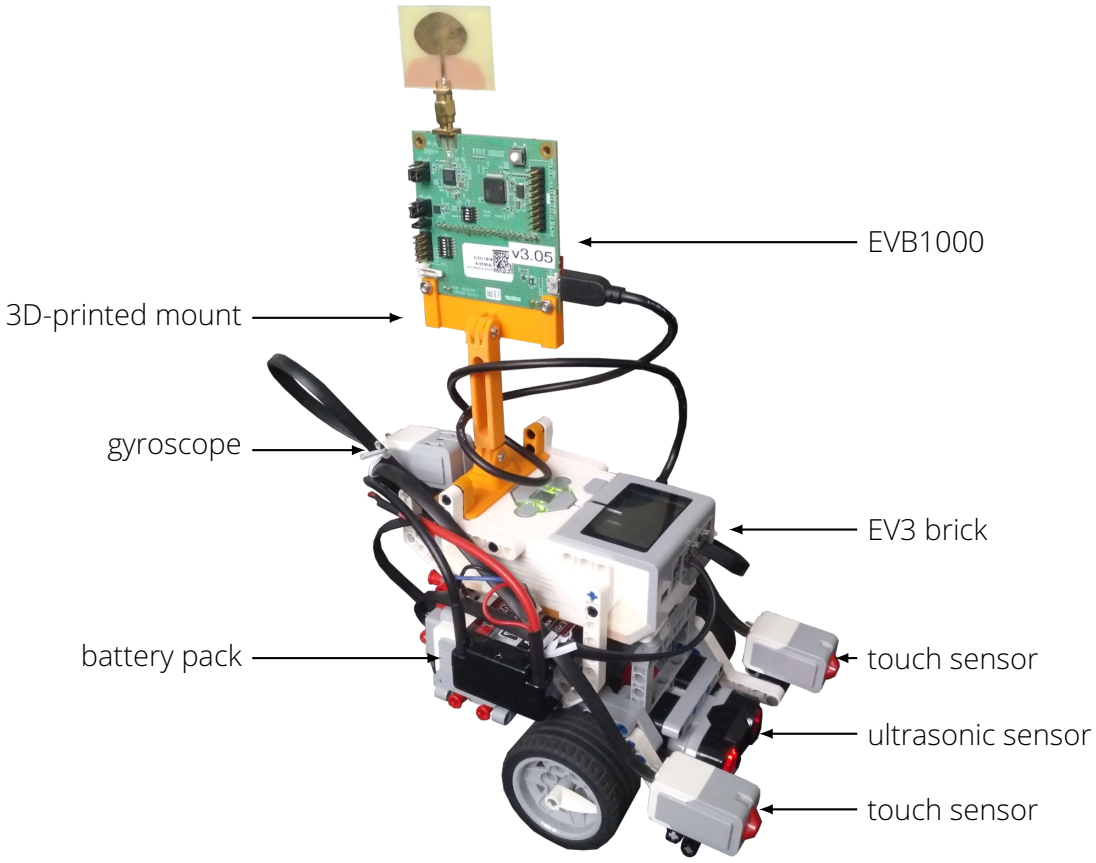


Figure 4.9.: The robot built from the Lego EV3 set, equipped with the measurement hardware and the custom extensions.

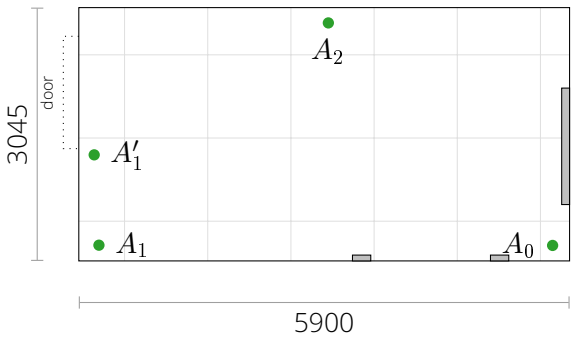
and processes the sensor data. In addition, the EVB1000 is connected to it via USB and the EV3 relays the recorded measurements of the EVB1000. The gyroscope is used to determine the orientation of the robot and to rotate it precisely. The ultrasonic sensors detect the distance to obstacles in the robots way — the touch sensors are a backup solution, which would trigger in the case of a collision.

To adapt the EV3 to our special requirements, the following two custom modifications were added: To attach the EVB1000 to the EV3 brick, we 3D-printed a mount consisting of a base plate, an arm and a bracket for the EVB1000. Further, to facilitate longterm measurements, we adapt the power supply of the robot. The standard Lithium-Polymer battery of the EV3 has a capacity of $2050\,\mathrm{mA\,h}$. In our intended use case, the robot consumes up to $3.4\,\mathrm{W}$ (including movements, powering the EVB1000 and data transmission). Hence, the standard battery can power the whole robot for max. $(7.2\,\mathrm{V} \cdot 2.05\,\mathrm{A\,h})/3.4\,\mathrm{W} \approx 4.3\,\mathrm{h}$. In order to enable longer measurements, we adapted the battery slot of the EV3 with a 3D-printed plate, which provide connectors for external battery packs. We then incorporated a $5300\,\mathrm{mA\,h}$ Lithium-Polymer battery pack[7]into the robot (visible in Fig. 4.9). Thereby, we extend the expected runtime to $\approx 10.6\,\mathrm{h}$.

All 3D-printed parts are produced using Polylactic Acid (PLA). As PLA is non-conductive, we expect negligible interference with the signal propagation [228].

Finally, with part (3), the single components need to be connected to aggregate the single measurements. The aggregation is executed on an additional measurement PC. The three

[7]Turnigy NC5300.2S2P.30



(a) Schematic floor plan of the measurement room.

(b) Picture of the room with the reflector set up.

Figure 4.10.: Measurement environment for the data set *robot*.

anchors are directly connected to the PC via USB and communicate over serial ports. The *Tag* is connected to the EV3 brick via USB and the respective serial port. The EV3 brick then receives the messages from the EVK1000 and simply relays them via Bluetooth to the measurement PC.

The **measurement environment** is an indoor office room, which is $5.9\,\mathrm{m}$ long and $3.05\,\mathrm{m}$ wide as shown in Fig. 4.10a.

Within this measurement room the following obstacles are present:

- the doorway on the right side of the schema is an inset into the wall $1350\,\mathrm{mm}$ wide and $186\,\mathrm{mm}$ deep
- two cable channels at the bottom, each $220\,\mathrm{mm}$ wide, $70\,\mathrm{mm}$ deep and $600\,\mathrm{mm}$ high
- a radiator on the left, which is $1400\,\mathrm{mm}$ wide, $95\,\mathrm{mm}$ deep and $600\,\mathrm{mm}$ high, while sitting $155\,\mathrm{mm}$ above the ground

The three *anchors* $A0, A1, A2$ are attached to the walls at $1150\,\mathrm{mm}$ height. To realize the recommended distance to the wall of at least $150\,\mathrm{mm}$, we attached them using 3D-printed mounts (visible in Fig. 4.10b). The coordinates of the anchor position are given in Table 4.1.

Table 4.1.: Anchor positions in mm measured from lower left corner of the room.

Anchor	x	y
A0	5696	186
A1	242	189
A1′	185	1275
A2	3000	2861

To break the symmetry between $A1$ and $A0$, a subset of the measurement were taken with $A1$ in position $A1'$.

Within this room we realized different measurement scenarios. For all scenarios, the robot moves randomly within the room at $0.15\,\mathrm{m/s}$. The pseudo-random movement is realized by stopping and rotating over a certain angle at random points in time. For this, each second the robot decides whether to rotate or not with probability $\mathcal{B}(1/60)$, i.e., approximately once in a minute. The rotation angle to rotate is sampled from $\mathcal{U}(10°, 90°)$. If

the robot detects an obstacle in its way, e.g., a wall, it backs off $10\,\mathrm{cm}$, rotates randomly for $\mathcal{U}(100°, 180°)$ and then proceeds its movement. The following different scenarios were realized:

symmetric This is the basic setting. The robot operates as described above. The anchors are positioned at locations $A0$, $A1$ and $A2$. Due to the symmetry of $A0$ and $A1$ this scenario is dubbed "symmetric".

asymmetric The same as above, but $A1$ resides at position $A1'$.

varying speed The overall setting is as in the first one. Instead of fixing the speed at $0.15\,\mathrm{m/s}$, we started at 50% of the robots possible speed, i.e., $0.25\,\mathrm{m/s}$. The speed was then reduced by $0.05\,\mathrm{m/s}$ steps until it reaches the final value of $0.05\,\mathrm{m/s}$.

reflector The robot movement and the anchor setup are as in the asymmetric setting. Additionally, a moving reflector is mounted in the room to induce further interferences (see Fig. 4.10b). The reflector consists of $750\,\mathrm{mm}$ long aluminium foil strips attached to a $1000\,\mathrm{mm}$ long wire. A stepper motor in the middle of the wire rotates the reflector with a constant speed of 2 rotations per minute. The construction is suspended from the ceiling with simple string, so that the wire is at $1200\,\mathrm{mm}$ height.

no movement We predefined spots in the room which form a $1\,\mathrm{m} \times 1\,\mathrm{m}$ grid. The grid is visualized in Fig. 4.10a — the grid intersections are the predefined spots. The robot is manually put into the single spots and does not move.

For all measurements, the **measurement procedure** followed the DecaWave two-way ranging procedure. At its core this is a Time of Flight (ToF) based localization procedure, where the distances between the *tag* and each *anchor* are determined. Based on these single distances and the known anchor positions, the tag can perform a multilateration to calculate its own position.

To estimate the respective ToFs, the *tag* T broadcasts a start signal. Upon reception of this signal, each *anchor* $A0 - A2$ immediately replies to the tag. After the final anchor message arrives at the *tag*, it sends a final message to conclude the ranging [44]. The general procedure is depicted in Fig. 4.11.

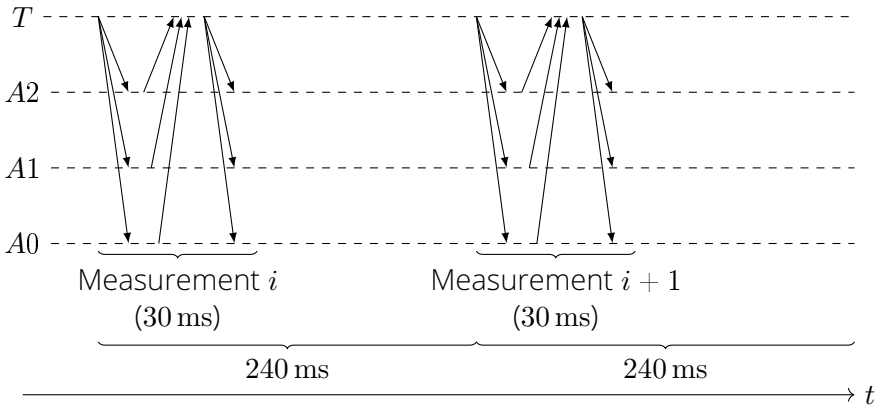


Figure 4.11.: The general measurement procedure for data set *robot*.

For our intended use case, we adapted the DW1000 firmware realizing this regular localization procedure in the following ways: Upon reception of any message, all DW1000 chips record the content of the internal registers describing CIR and RSSI of this transmission. The DW1000 calculates RSSI values from the *RX Level* and the *First Path Power Level*. To increase the processing speed, we do not record the full 1016 possible samples of the CIR register. Instead, we facilitate the internal first path detection of the DW1000 and record only 50 samples before and 200 sample after the first path. These values are selected in accordance to the rationale in Section 4.1. With this adaption, we could reduce the time required for the actual ranging to $30\,\mathrm{ms}$. Additionally, we reduced the time slots for participating anchors from 10 to 3, which allows for faster consecutive measurements every $240\,\mathrm{ms}$.

4.4.3. Resulting Data

An example of the recorded data in the time domain is plotted in Fig. 4.12. The different time scale again demonstrates the adapted processing: instead of recording the completed CIR register as in the data set *attack*, we only record the 251 values around the detected first path.

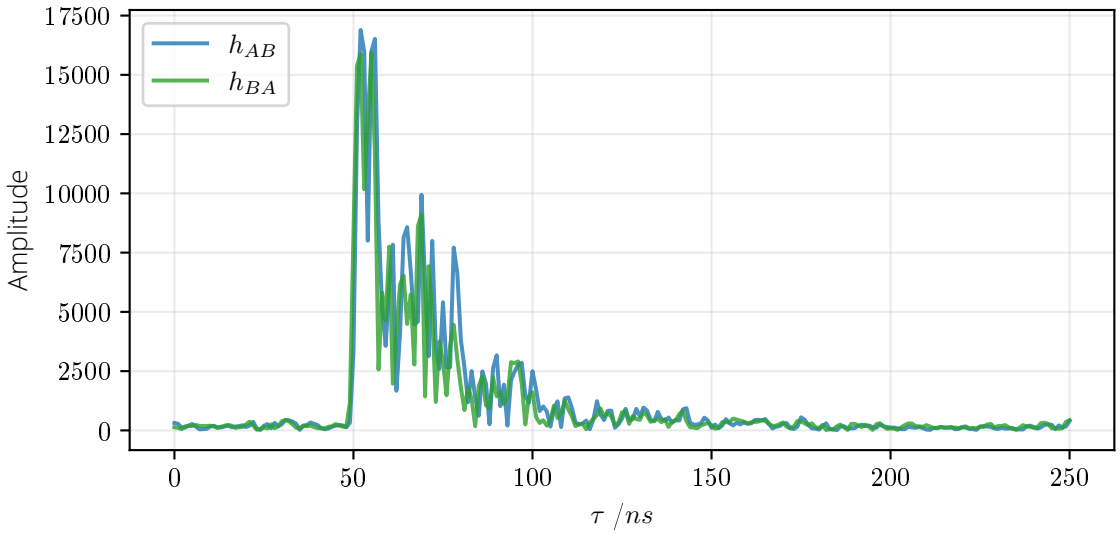


Figure 4.12.: Example record of the data set *robot*.

For all settings used, the following parameters are constant: Every $240\,\mathrm{ms}$ a set of CIRs is recorded, resulting in 4.1667 CIRs per second. Each ranging consists of 12 CIRs (6 possible channels, each measured reciprocally). Each single CIR consists of 251 samples, taken every $1\,\mathrm{ns}$. Table 4.2 shows the number of realizations in each setting.

Table 4.2.: Data recorded in the *robot* data set. Every realization consists of 12 CIRs (3 for each of the 4 terminals) with 251 samples each.

Data Set		Duration	Realizations
robot	symmetric	56 h	845 250
	asymmetric	19 h	281 595
	reflector	20 h	299 730
	var. speed	16 h	237 424
	no movement	40 min	5418
	total	111 h 40 min	1 669 417

5. Initial Analysis

In the following, we perform an initial analysis of the key material acquired in the measurement campaigns described in Chapter 4. First, we analyse the core property with respect to CRKG, the correlation between the participating parties. Then, we proceed to analyse the results from the attacker's point of view. Subsequently, the time synchronization of the raw CIRs is evaluated. Finally, we summarize and discuss the findings in an intermediate summary.

For this analysis, we use the metrics introduced in Section 2.5.1.

5.1. Correlations

In this section we evaluate the cross-correlation of raw, unprocessed measurement data, i.e., the CIRs of Alice, Bob and Eve. This gives us insight into whether the basic assumption of CRKG is true: that the legitimate communication partners have an advantage over the attacker by the very nature of their shared channel. This means, following the respective assumptions, we expected that the CIRs of the eavesdropper will have a much lower correlation to the CIRs of Alice and Bob than their CIRs to each other. If this is the case, we denote the shared channel as *advantageous*

The correlations for the measurements of data set ***scenarios*** are depicted in Fig. 5.1. The main message of this figure is that the fundamental assumption that the legitimate partners have an advantageous channel is fulfilled in all scenarios: In all scenarios there is a clear separation between the majority of the legitimate observations and those of the attackers, i.e., between the respective IQRs.

The following artefacts should be noted: On average, the eavesdropper achieves the best results in scenario *Static D*. This is in line with expectations, since Eve is positioned right next to one of the legitimate terminals. Despite this advantageous position, there is still a clear separation, thus no direct advantage through this position can be assumed here.

The eavesdropper observations appear to yield bimodal distributions in *Static B* and *Static C*. No clear reason for this could be established. The most plausible explanation is that this is a direct result of the terminal positions with respect to the measuring room.

Finally, there are small overlaps between the distribution tails in scenarios *Static C*, *Static D*, *Interference A* and *Interference B*. In turn, this is also visible in the *Total* summary row.

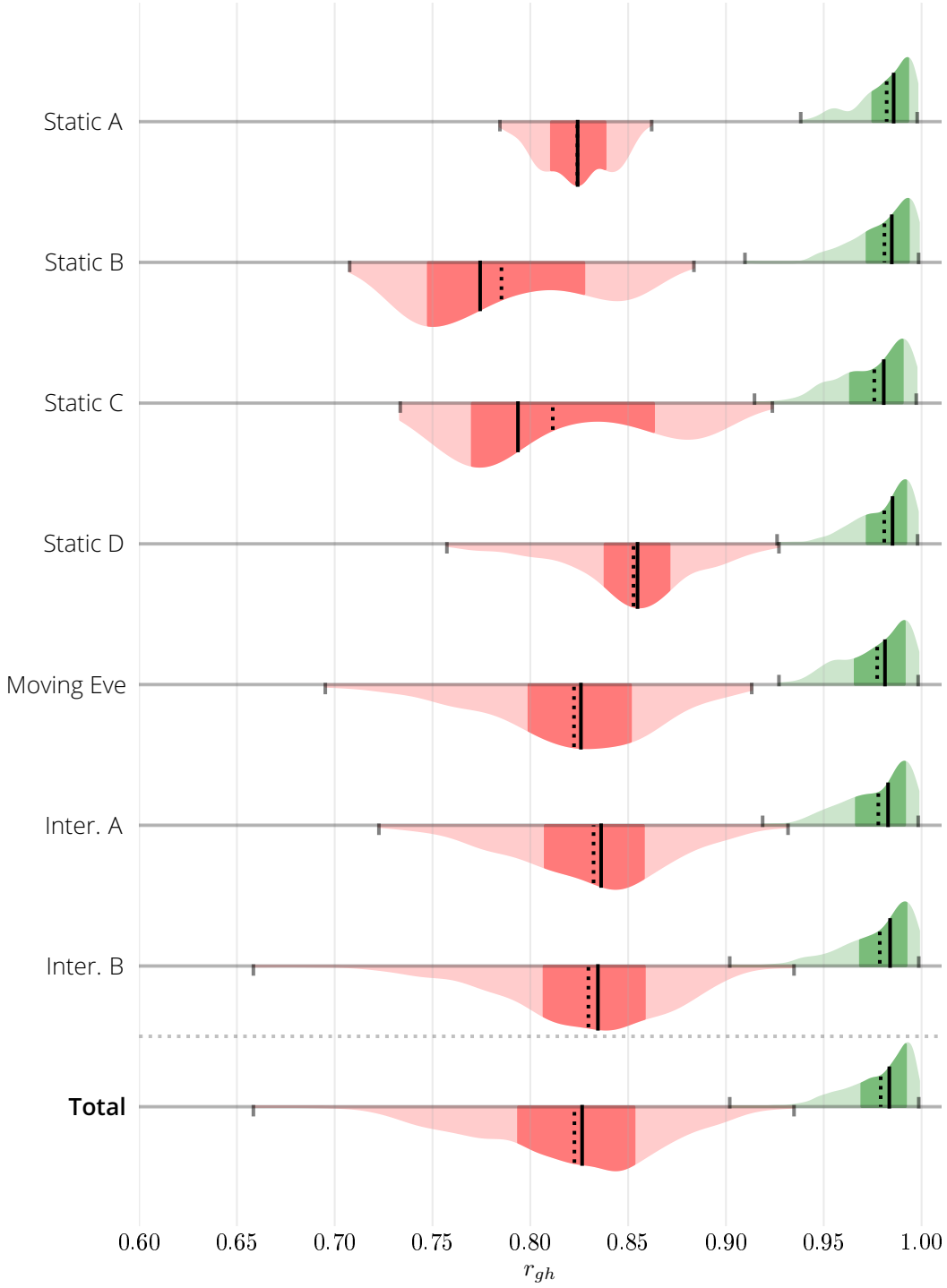


Figure 5.1.: Correlations of the different scenarios of the *scenarios* data set.

This overlap means that for this concrete distribution estimate it would be possible for the eavesdropper to measure a CIR observation whose correlation with the legitimate CIR is higher than that of the reciprocal measurement. In the actual measurements no such pair occurred in any scenario.

Nevertheless, the fundamental possibility of such observation pairs must be considered when designing CRKG systems. Concretely, not all possible combinations of reciprocal

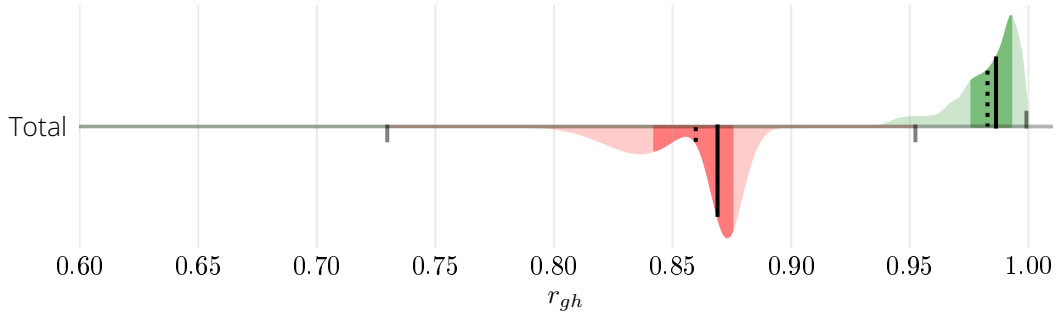


Figure 5.2.: Correlations of the *longterm* data set.

measurements must be accepted for key generation — if the deviation between the CIRs becomes too high, the process must not generate a valid key any more. This is practicable in the IR phase: here not arbitrarily high differences may be eliminated. Instead, a concrete lower limit for the accepted difference has to be defined and implemented.

In total, the legitimate partners achieve correlations from 0.902 to 0.997, with a mean at 0.979 ± 0.015 and a median at 0.984. Accordingly, the eavesdropper correlations range from 0.615 to 0.935, with mean of 0.820 ± 0.045 and median of 0.826. The complete values for all scenarios can be found in Table 5.1.

The correlations for the measurements of data set ***longterm*** are depicted in Fig. 5.2. In general it confirms the observations of the data set *scenarios*: There is a clear separation between the respective IQRs indicating the advantageous channel between the legitimate partners. Again, there is a small overlap between the extrema of the density estimations. The minimum whisker of the legitimate partners is not visible, as there is a single outlier at 0.5731.

Overall, the eavesdropper achieves slightly higher correlations than in the data set *scenarios*. This corresponds to the expectation as the environment for this data set is much more static and thus provides fewer changes and thereby "fresh" entropy.

In total, the legitimate partners achieve correlations from 0.573 to 0.999, with a mean at 0.983 ± 0.013 and a median at 0.986. The eavesdropper correlations range from 0.598 to 0.952, with mean of 0.875 ± 0.032 and median of 0.889. The complete values are shown in Table 5.1.

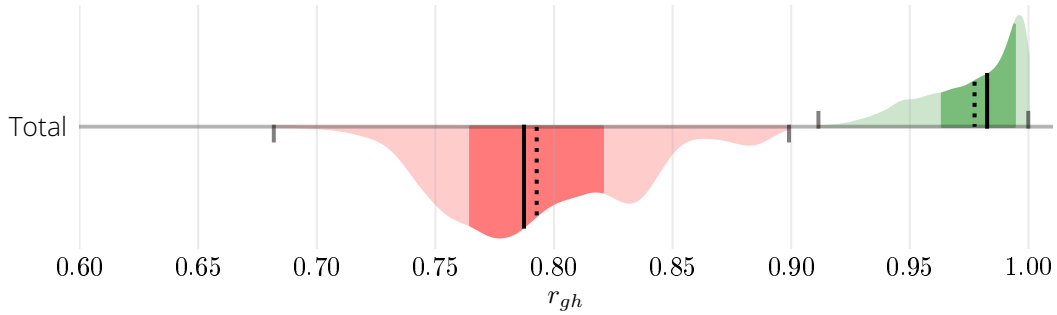


Figure 5.3.: Correlations of the *attack* data set.

The correlations for the measurements of data set ***attack*** are depicted in Fig. 5.3. Here, the legitimate correlations have values in the range of 0.9115 to 0.9999, with mean and standard deviation of 0.9773 ± 0.0193 and median at 0.9826. The eavesdropper achieves correlations in the range from 0.6819 to 0.8991, with mean 0.7927 ± 0.0372 and median 0.7873.

The correlations for the measurements of data set ***robot*** are depicted in Fig. 5.4. Overall, the legitimate partners achieve correlations from 0.8177 up to 0.9991, the mean and standard deviation are 0.9783 ± 0.0159 and the median 0.9828. The eavesdroppers CIRs have correlations from 0.6003 to 0.97, mean of 0.8231 ± 0.0503 and median of 0.8277. Again, the complete values for all scenarios can be found in Table 5.1.

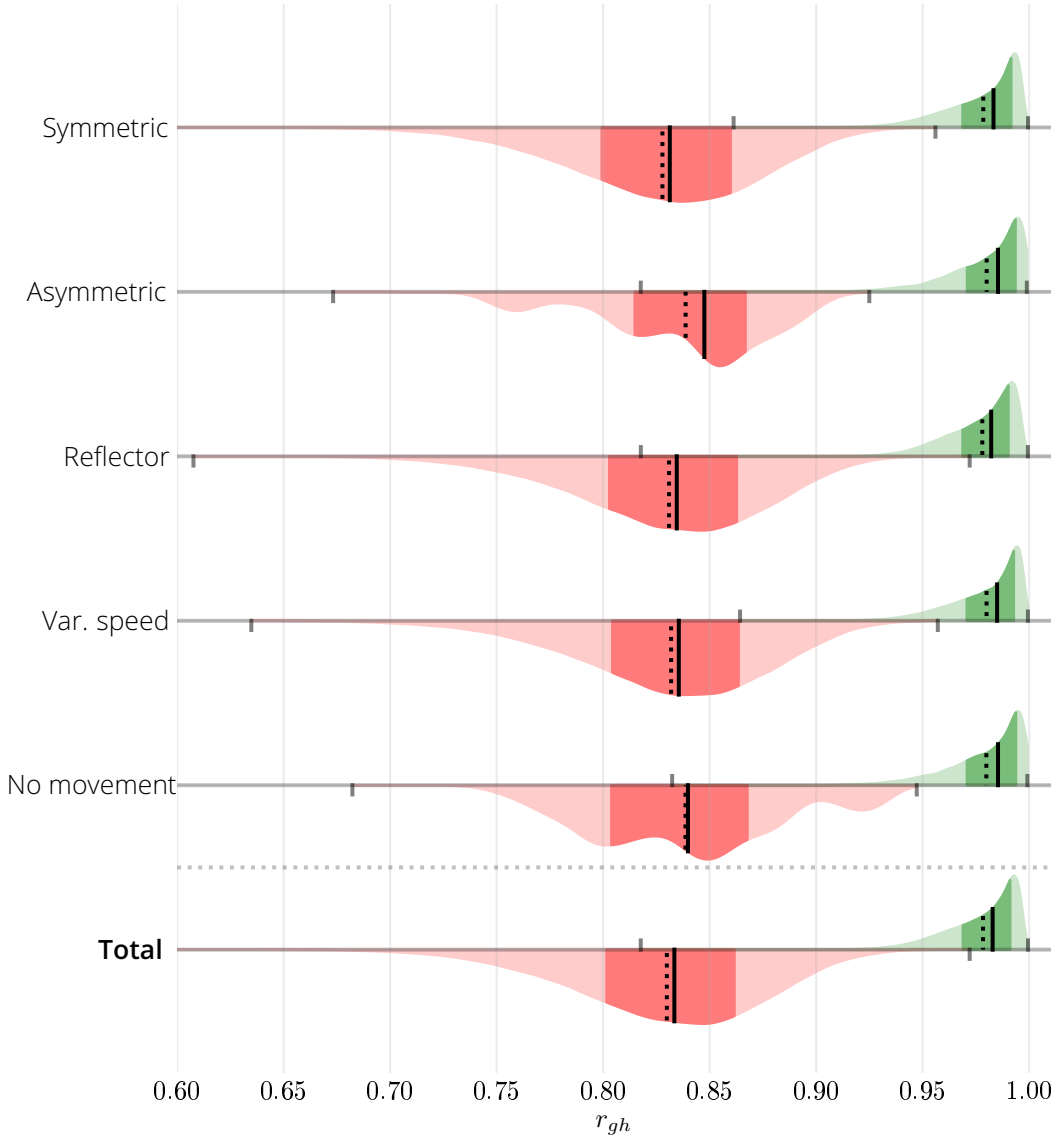


Figure 5.4.: Correlations of the *robot* data set.

Overall, the results indicate that for a secure, functional CRKG system, legitimate partners should observe CIRs with a cross correlation > 0.95, whereas the correlation of attacker observations should be < 0.90.

5.2. Spatial and Temporal Decorrelation

The rich information in the *robot* data set allows to further analyse the correlation of both the legitimate partners and the attackers. In this context, changes caused by temporal or spatial shifts of the attackers are of particular importance.

The **spatial decorrelation** relates the achieved cross correlation to the available position information. This allows to further inspect the CRKG security argument of spatial decorrelation based on *uniform scattering*. Following this argument it is expected that the achieved correlation's magnitude is decreasing with increasing spatial distance. Additionally, it should follow the Bessel function with respect to distance between eavesdropper and legitimate terminal (cf. Section 2.3).

To visualize the achieved correlations as well as the spatial decorrelation we show the achieved cross correlations in a heatmap imposed over the room geometry displayed in Fig. 4.10a. For all heatmaps, if not stated otherwise, the moving robot is interpreted as legitimate terminal Alice; the anchor in the lower right corner of the map is Bob (denoted as B).

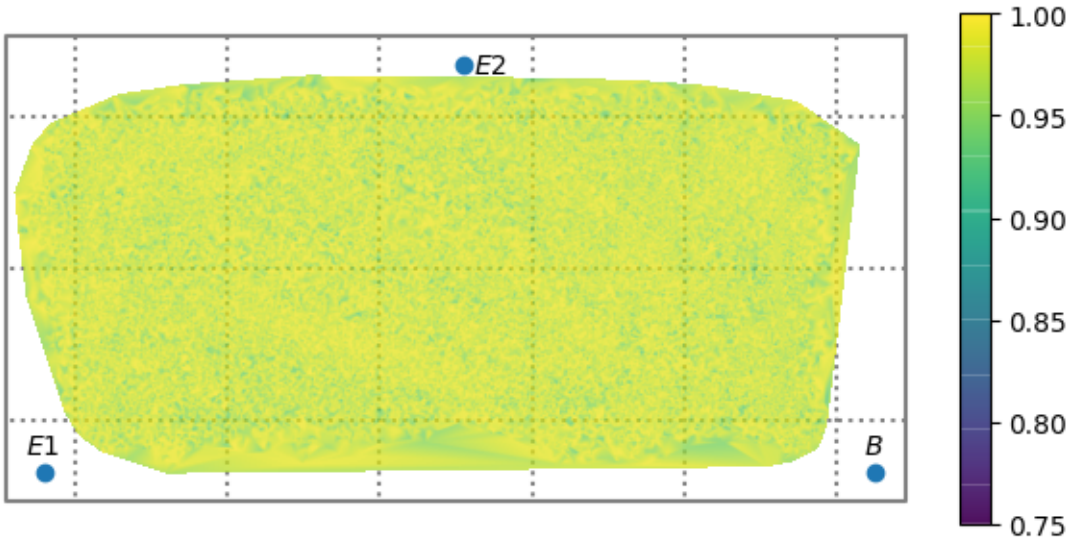
We first show the actual cross correlations in dependence of the current position of the robot platform. The heatmap for the legitimate partners Alice and Bob is shown in Fig. 5.5a. It shows the cross correlation in the range of 0.75 to 1.00. Figure 5.5b shows the respective cross correlation between the CIRs obtained at $E2$ and those of Alice and Bob. It can be seen that, in general, the obtained results are significantly lower. In the heatmap, no region can be discerned where the attacker has a noticeable advantage in terms of the achieved correlation.

In addition to the straight forward comparison of cross correlations, we also look at the correlation with the *uniform scattering* assumption in mind. If this assumption is fulfilled in practice and the correlation periodically decreases and increases with growing distance as predicted with the Bessel function, concentric circles around a legitimate terminal or the respective eavesdropper should be recognizable in the correlation heatmap. Considering the center frequency of $3.99\,\mathrm{GHz}$, circular structures repeating with period $\lambda \approx 7.5\,\mathrm{cm}$ should be recognizable around A or $E2$. Since terminal A is constantly moving, these artefacts should be recognizable around the attacker position $E2$ in Fig. 5.5b. In the actual heatmap, no clearly identifiable circular structures can be found.

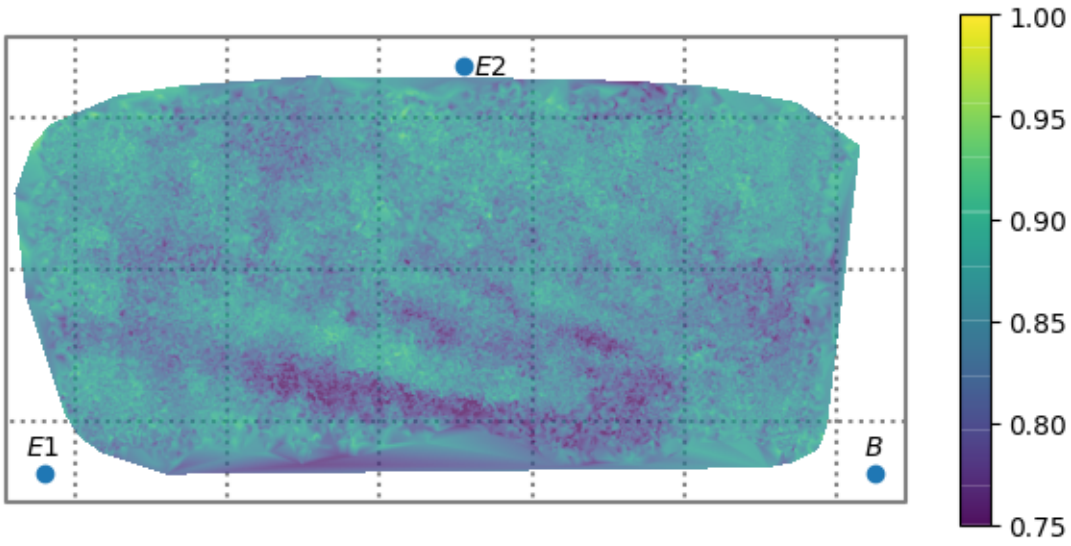
The respective heatmap for eavesdropper position $E1$ is shown in Appendix A.2.1. This generally confirms the findings of Fig. 5.5b, with an overall low correlation for the eavesdropper, no explicit advantage regions and no circular but radial patterns.

Within the attacker heatmap in Fig. 5.5b, some regions of particularly low correlation can be seen — such as in the lower left region or fairly centrally below $E2$. Such regions appear in all heatmaps and change depending on the evaluated channels. Due to the dependence on the evaluated channel and the spatial propagation radially away from the source, these artifacts are probably attributable to the radiation characteristics of the antennas used. A corresponding image with Bob at position $A2$ can be found in the appendix Appendix A.2.2, where the origin of the radial pattern changes accordingly to this position.

In order to investigate the spatial decorrelation in more detail, we analysed distances smaller than two wavelengths separately. For this purpose, we searched for CIR measurements for channel Alice-Bob, independent of the respective measurement time, whose distance is smaller than 2λ. That is, here we compare the correlation between Alice and



(a) Correlations r_{AB} between Alice (robot) and Bob (B).



(b) Correlations r_{AE2} at the attacker E2.

Figure 5.5.: Cross correlation in dependence of terminal positions between the legitimate partners CIRsand between the legitimate and eavesdropped CIRs, respectively, superimposed on the room geometry.

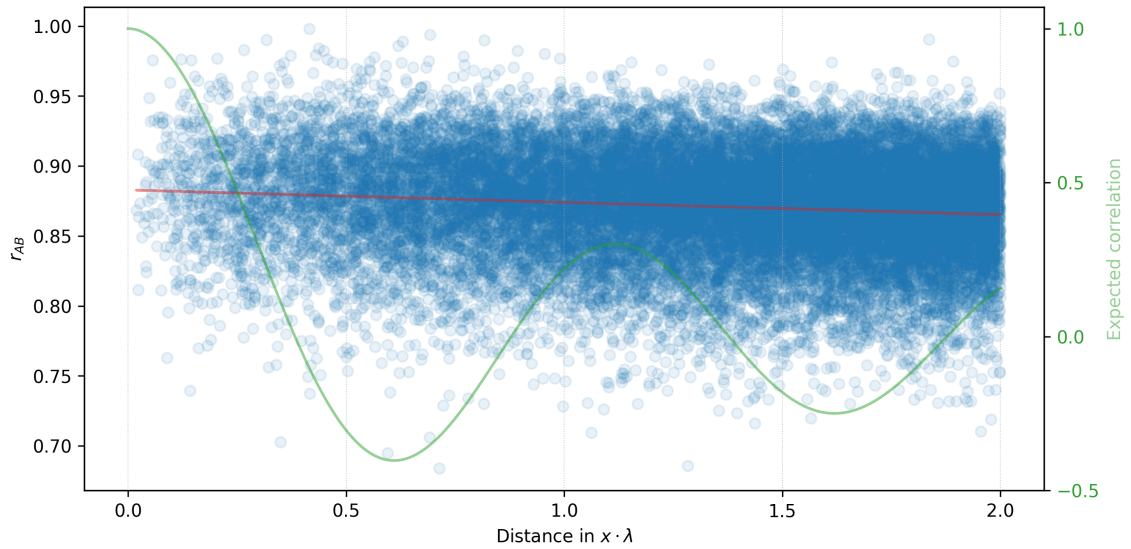


Figure 5.6.: Cross correlation r_{AB} in dependence of spatial displacement. The red line is the trend of the data, the green line is the expected correlation from the *uniform scattering*.

Bob at different times, but at similar positions. Following the argument of spatial decorrelation, the same effects should occur here as for an attacker. Concretely, the values of the correlation should also follow the Bessel function in Fig. 2.11.

We have visualized the concrete results for a sample of 25000 such pairs as a scatter plot in Fig. 5.6. Overall, the correlations range 0.684 to 0.998, with a mean of 0.871 and standard deviation of 0.038. Hence, the correlations are below the actual legitimate correlation of 0.978, but are higher than the plain eavesdropped ones of 0.830. The red line represents the trend within this data, representing the local average, which drops from 0.876 to 0.870. It is clearly visible that the correlation data does not follow the trend predicted by *uniform scattering*. The plot of the individual values also shows that sometimes very high correlations are obtained even for distances larger than 0.38λ. In $0.152\,\%$ of all cases the correlation is larger than the average correlation of the legitimate partners, i.e., >0.97.

In addition to the spatial decorrelation we also analysed the **temporal decorrelation**. To analyse these correlations, we compare the correlation of a particular CIR realization with time shifted variants of the same channel. This is accomplished by facilitating the successive measurements obtained during the measurements. Concretely, we take a single CIR pair, e.g., those of Alice and Bob, and calculate the correlation to the same pair's realizations obtained later in time. Since, the acquisition time of the single measurements is $240\,\mathrm{ms}$ the temporal shifts will be multiples of this value. Following the assumptions about the wireless channel, especially the channel coherence time, we expect the channel to decorrelate quickly with increasing time differences.

This analysis for the legitimate partners as well as for the two eavesdropper positions is shown in Fig. 5.7. The figure shows the average cross correlations for all CIRs in the dataset *robot*. The shift of 0 denotes the average correlations of the respective observations. Hence, it is in line with the analyses before, showing average correlation between Alice and Bob of 0.98 and to the eavesdroppers, 0.85 and 0.84 respectively.

It is clearly visible that the correlation of the legitimate partners drops rapidly with in-

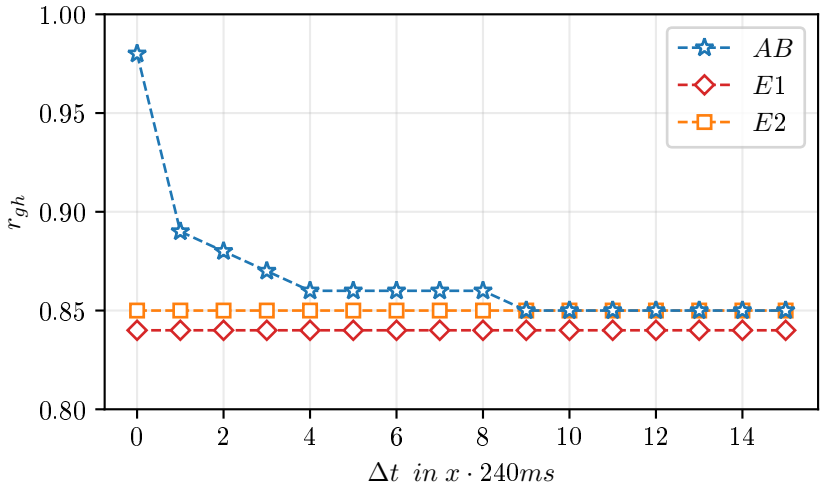


Figure 5.7.: Cross correlation r_{gh} for different CIR measurements. As the recordings were made in $240\,\mathrm{ms}$ time steps, the X axis is given in multiples of this time step.

creasing time lag. At an time offset of 1, i.e., $240\,\mathrm{ms}$, the average correlation already drops to 0.89. From there, it drops further to 0.86 at $1\,\mathrm{s}$, reaching the value range of the attacker's correlation here. The attackers experience neither an improvement nor a worsening of the correlation due to the time shifts — the average correlation remains low regardless of the offset.

In this analysis, it must be taken into account that the temporal displacement inevitably includes a spatial displacement due to the robots continuous movement. Specifically, at the standard speed of $0.15\,\mathrm{m/s}$, the robot moves about $0.036\,\mathrm{m}$ between two measurements. The influence of the spatial displacement becomes apparent when the different velocities are considered, as shown in Fig. 5.8. Here, the different velocities cause decorrelation at different speeds, with slower motion also causing slower decorrelation. The lowest speed

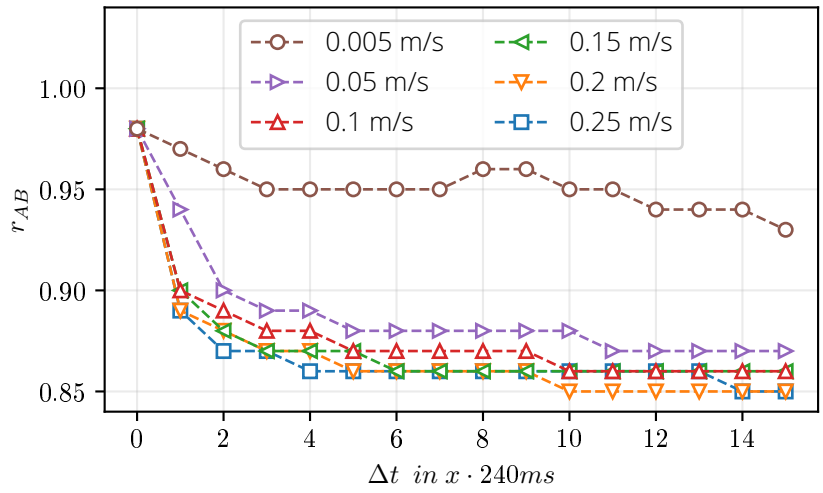


Figure 5.8.: Cross correlation r_{AB} in dependence of movement speeds and time shifts.

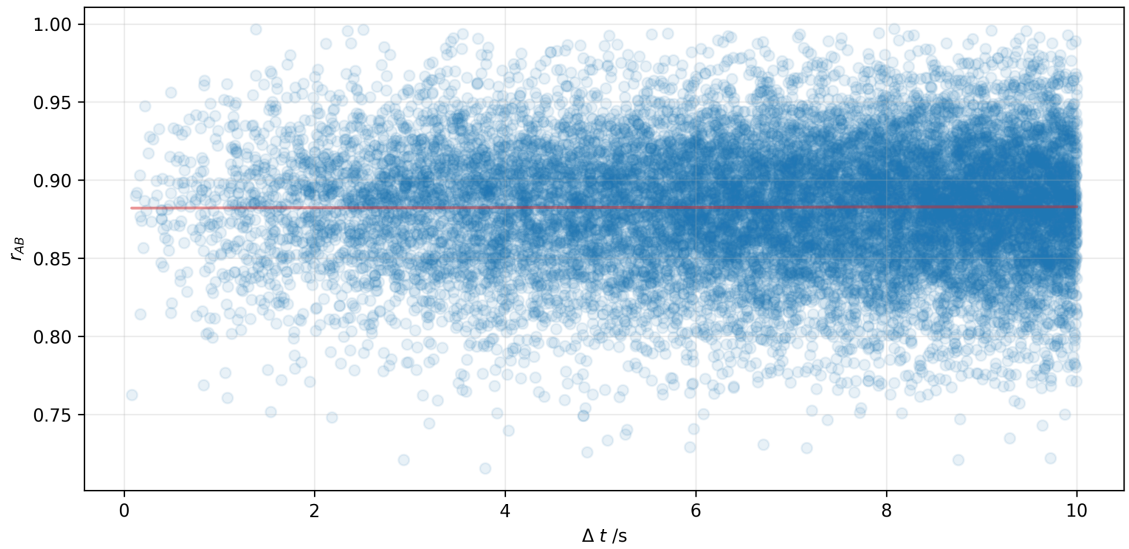


Figure 5.9.: Cross correlation r_{AB} in dependence of temporal displacement. The red line is the trend of the data.

of $0.005\,\mathrm{m/s}$ decorrelates so slowly, that correlations <0.9 are only reached for shifts >47, i.e., $>11.28\,\mathrm{s}$, which corresponds to a spatial displacement of $0.0564\,\mathrm{m}$. This indicates that the spatial displacement has a greater influence on the correlation than the temporal shift.

To remove the influence of the spatial displacement, we finally looked at CIR pairs with low spatial but high temporal distance. For this, we analysed pairs with arbitrary time offset and a maximum spatial displacement of $50\,\mathrm{mm}$, which is the average localization error. In the same vain as Fig. 5.6, we show the results as scatter plot in Fig. 5.9. The red line is again the trend of the correlation data. It can be seen that the correlation does not change with respect to a purely temporal shift. The trend changes only minimally from 0.8824 to 0.8816. These results confirm that the correlation is primarily determined by the spatial shift.

This figure provides another insight regarding the potential for attack. Due to the small spatial offset, even with high temporal shifts, i.e., $>20\,\mathrm{s}$, correlations can still be achieved that are in the range of legitimate CIR pairs. In $1.15\,\%$ of the cases, this correlation exceeds the average value of 0.97 for the legitimate partners. This is not only significantly more frequent than with spatial change but also suggests that attacks like the stalker attack [119] might also be possible in the context of CIR based CRKG.

5.3. Time Synchronization

In the following section we analyse the *time synchronization* of the measured data.

In this scope, *time synchronization* is considered to be the following problem: If we consider a typical CIR as described in Chapter 2, the multipath clusters arrive at certain points in time. Considering the CIR as returned by the hardware, we have a vector of scalars

$$h = \{v_0, v_1, v_2, \ldots, v_n\}. \tag{5.1}$$

With this vector, the indices $i = \{0, 1, 2, \ldots, n\}$ of the single scalars are the respective points in time, when this value was recorded. So, in dependence of the sampling interval T_s, we

can say that the respective arrival times τ_i are offsets from the first recorded value v_0:

$$\tau_i = T_s \cdot i \qquad , i \in \{0, 1, 2, \ldots, n\} \tag{5.2}$$

After the reciprocal measurements of Alice and Bob are conducted, each has a realization h^A and h^B. For the single multipath component as well as for the LOS component, to be *time synchronous* means that if the component arrives at time offset τ_x^A at Alice, it also arrives at the same time offset at Bobs, i.e., τ_x^B.

As the time difference between the single multipath clusters can be interpreted as part of the CIR information itself, this requirement can be relaxed to only the LOS component. This means, that in our scope, *time synchronization* means, that

$$\tau_{LOS}^A = \tau_{LOS}^B, \tag{5.3}$$

where τ_{LOS} is the offset of the LOS component.

If CIRs are not synchronous and $\tau_{LOS}^A \neq \tau_{LOS'}^B$ a difference between these τ occur, which we call an *offset*. The absolute offset between two CIRs is denoted by δ. This offset is calculated as difference between the respective times at Alice and Bob. So, if $\tau_{LOS}^A = T_s \cdot i$ and $\tau_{LOS}^B = T_s \cdot j$, then the offset δ is

$$\delta = |i - j| \tag{5.4}$$

If *time synchronization* is not given and the reciprocal CIRs are not synchronous, processing after quantization will be significantly impaired because this will cause an additional, comparatively large mismatched between the quantized reciprocal bit strings. The actual implications are considered precisely in the appropriate context in Section 7.1.

In all measurement setups, and likewise in all practical CRKG scenarios, Alice's and Bob's wireless terminals are independent participants. This means in particular that they do not have a common clock or clock signal available. Hence, we cannot assume that the estimated CIRs are synchronous in time, i.e., have an offset of $\delta = 0$ for all measurements. Thus, we analysed the absolute offset δ for each data set.

The results of this analysis for every data set are shown in Fig. 5.10. All data sets have a median of 2. The median is exactly at 2, since the offsets are integer values. This also causes the "spiky" progression of the distribution *longterm* and *robot*. Further, the data sets *longterm* and *robot* achieve comparable results, with means at 2.027 and 2.076 as well as standard deviations of 1.584 and 1.449.

A notable deviation here is the data set *attack*: With a mean offset of 4.074 and a corresponding standard deviation of 16.457, it shows significantly greater differences than the other data sets. This clearly proves the influence of adequate processing with regard to time synchronization. In this data set no fine granular synchronization was performed — the data was used as recorded. On the other hand, in the case of the data set *robot*, which uses the same hardware, the internal leading edge detection algorithm was utilized. The resulting improved synchronicity is clearly visible in the data.

The plot is cut at offset $\delta = 10$ and thereby does not show the respective maximum outliers. This is intentional as the outliers can become quite large and, in turn, the actual plot would be very small. For the respective data set the maximum outliers are at: *scenarios* at 25, *longterm* at 56, *attack* at 243, and *robot* at 33.

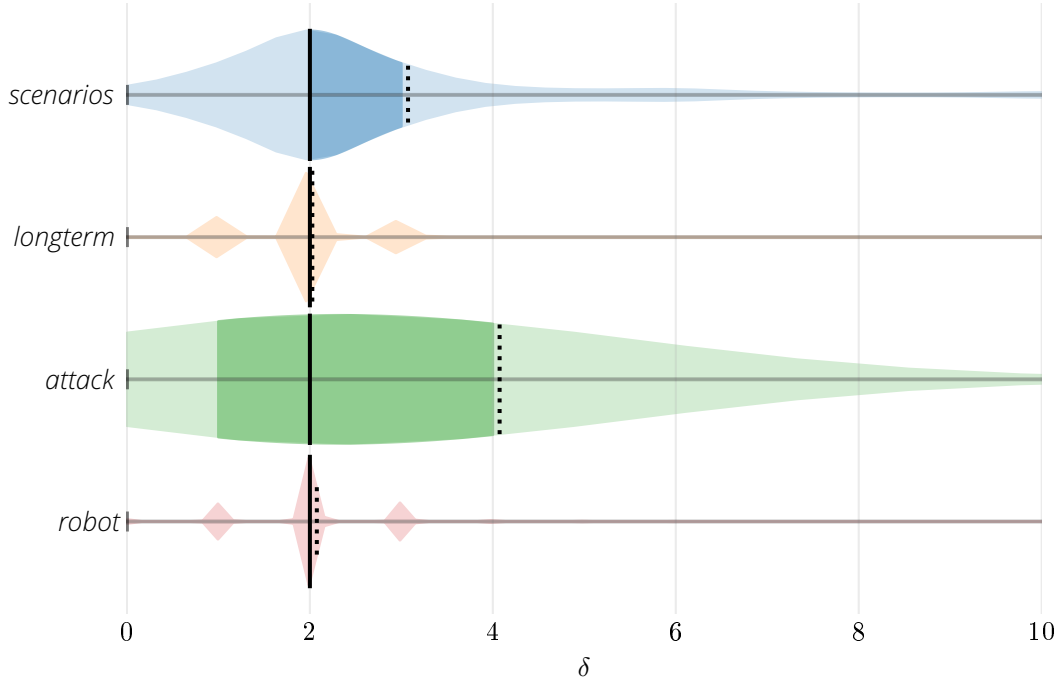


Figure 5.10.: The summarized distributions of the absolute time offset of all data sets. This summary is calculated over all settings of the respective data sets.

The analysis of the *time synchronization* showed that for all data sets the raw measurements are not synchronous. This means, for an effective quantization and in turn key generation itself, the single reciprocal measurements must be preprocessed to be synchronized in time. The complete results including the single settings are shown in Table 5.1.

5.4. Intermediate Summary

The initial analyses of raw CIR measurement data generally show that CIRs are suitable as input data for CRKG.

The analysis of the cross correlations in particular showed that the basic assumption of an advantageous channel for the legitimate communication partners or a degraded channel for the attacker, in the sense of the wiretap channel, is confirmed. This is shown by a consistently high correlation of the legitimate reciprocal measurements, which is contrasted by a comparatively low correlation of the eavesdropped measurements. Thus, in general, higher correlations can be assumed for Alice and Bob and thus also appropriate key material.

Nevertheless, it can be seen that in isolated cases the eavesdropper can also achieve high correlations. This is visible in the correlation plots by means of the outliers in the distribution. Consequently, this means that not the complete range of Alice-Bob observations should be used, but a lower limit for the acceptable correlation should be defined. Concretely, this means that, for example, in the IR step the parameters of the respective solution should be selected in such a way that legitimate observations below the defined correlation lower limit are also not successfully reconciled. This is a valid result in the con-

text of IR and leads here to an increased security — which is the highest criterion for CRKG.

However, the initial analysis also shows a need for further processing, to prepare the use of CIRs. The recorded measurements show a high proportion of noise. Since by definition these measurement points do not carry any information and certainly no reciprocal information, they must be removed before further processing. This has to be done in a deterministic manner on both legitimate communication partners to ensure consistent key material at both parties.

Analogously, the analysis also shows a temporal offset between the reciprocal measurements. This must also be removed so that no additional, but avoidable differences between the reciprocal measurements, or their quantizations, are induced.

Finally, the analysis of the correlation as a function of the spatial placement of the terminal questions the accepted security assumption of *uniform scattering*. Although the eavesdropper has low correlations even in proximity of the legitimate nodes, their correlations do not follow the progression predicted by the *uniform scattering*. Considering that this assumption itself is based on strong assumptions about the physical propagation environment, these results can be interpreted as an indication that the *uniform scattering* assumption does not hold in practice. This in turn implies the assumption of decorrelation for distances $> 0.5\lambda$ is *not* justified in practice either.

The core results of the initial analysis are summarized in Table 5.1.

Table 5.1.: Summary of the initial analysis of the acquired data sets.

Data Set		Realizations	Length	CIRs	Cross Correlation						Time offset	
					A-B		A-E		B-E			
				$1/\mathrm{s}$	μ	σ	μ	σ	μ	σ	μ	σ
	Static A	603	200	5	0.9822	0.0130	0.8240	0.0168	0.8377	0.0111	3.1957	5.3732
	Static B	603	200	5	0.9810	0.0141	0.7852	0.0446	0.7966	0.0358	3.0210	3.3925
	Static C	603	200	5	0.9758	0.0167	0.8115	0.0516	0.7997	0.0664	5.9818	7.6473
scenarios	Static D	603	200	5	0.9809	0.0136	0.8527	0.0312	0.8433	0.0368	2.0155	0.8896
	Moving Eve	603	200	5	0.9773	0.0159	0.8223	0.0412	0.8136	0.0444	2.4467	2.4050
	Interf. A	603	200	5	0.9779	0.0161	0.8323	0.0377	0.8138	0.0477	2.5202	3.8880
	Interf. B	603	200	5	0.9786	0.0166	0.8297	0.0398	0.8233	0.0421	2.3300	1.8913
	total	4221	200	5	0.9791	0.0154	0.8225	0.0434	0.8183	0.0464	3.0730	4.2130
longterm		175 005	200	5	0.9827	0.0129	0.8598	0.0212	0.8902	0.0426	2.0265	1.5841
attack		17 445	2000	5	0.9773	0.0193	0.7927	0.0372			4.0740	16.4473
	Symmetric	845 250	251	4	0.9785	0.0162	0.8278	0.0447	0.7875	0.0520	2.1264	1.6466
	Asymmetric	281 595	251	4	0.9801	0.0174	0.8387	0.0395	0.8618	0.0316	2.0114	0.8656
robot	Reflector	299 730	251	4	0.9780	0.0153	0.8309	0.0446	0.8465	0.0424	2.0313	1.2496
	Var. speed	237 424	251	4	0.9800	0.0159	0.8319	0.0445	0.8247	0.0475	1.8181	1.3776
	No movem.	5418	251	4	0.9799	0.0178	0.8387	0.0438	0.7825	0.0508	1.9971	1.1505
	total	1 669 417	251	4	0.9783	0.0159	0.8299	0.0445	0.8164	0.0561	2.0759	1.4478

Part III.

Channel Reciprocity-based Key Generation using Channel Impulse Responses

6. General Objectives and Design Choices

The general processing during CRKG as well as the usage of CIRs as common source of randomness come with certain requirements regarding the processing to assure the overall goal of secure and efficient key exchange. In the following we describe these requirements as well as constraints and highlight the high level approaches to the solutions.

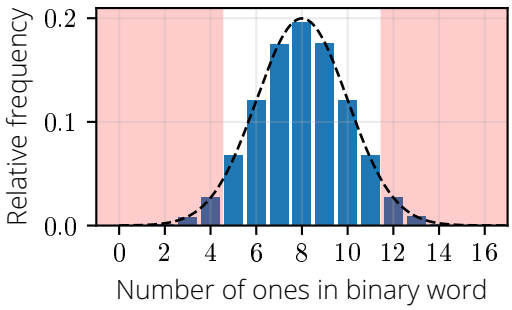
The **requirements** for key derivation based on reciprocal channel properties can be divided into two groups: first, the requirements derived from the information theoretic proofs as developed by Maurer [134] and Ahlswede and Csizar [1]. And second, the requirements derived from the specific use cases and respective practical realization of such key derivation schemes.

The requirements of the supporting information theoretic proofs are the following (cf. Section 2.2): *Reliability* states that the probability for Alice and Bob to derive different keys is negligible, i.e., they derive the same key with high probability. *Secrecy* means that the messages of the key derivation protocol only leak negligible amount of information about the key material to an attacker. This is most commonly defined via the set of exchanged messages, but also includes the initial randomness sharing. *Uniformity* requires that the resulting keys are distributed uniformly over the key space.

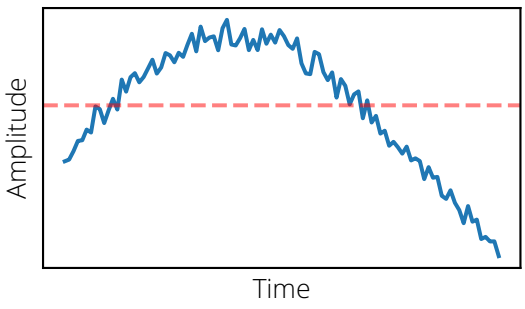
Another set of requirements is derived from the use cases of CRKG and practical questions regarding actual implementations. These can be divided into *resource centric* and *efficiency centric* requirements.

The *resource centric* requirements are primarily derived from the actual use case of CRKG implementations. In our case, the use case consists primarily of resource-constrained endpoints, such as in the IoT context or in Tactile Internet scenarios. Hence, in parallel to the actual functionality of such devices, energy-efficient processing is always an additional goal. The most important drivers of energy consumption are, on the one hand, computing effort and, on the other hand, communication by means of wireless technology. This in turn means that practical realizations in this use case scenario should save energy at exactly these two points.

In terms of *efficiency*, specifically performance, of the key derivation protocol, the main requirement is a high key generation rate. However, this apparently simple requirement is subject to influence by a wide variety of measures. One important point, for instance,



(a) The distribution of ones in quantized uniformly distributed values.



(b) Small scale vs. large scale fading and the resp. average calculated as threshold.

Figure 6.1.: Visualization of the potential biases introduced by blockwise statistics.

is the utilization of the available entropy in the key material — the better the entropy is harnessed, the greater the number of secure key bits that can be generated. This requirement also influences practical concerns, such as ensuring that the necessary entropy is provided by changes to the reciprocal channel. Finally, the efficiency of the overall protocol is directly influenced by the time efficiency of the single processing steps, i.e., highly complex computations can actually hinder the protocol efficiency.

From these requirements actual **constraints** for the solution design can be derived:

The first two constraints are straight forward derivations of the resource centric and the efficiency centric requirements of the practical implementations. As mentioned above the main drivers of energy consumption are computing effort and wireless transmissions. Here, communication is especially resource demanding: considering the measurement devices used, with up to $126\,\mathrm{mA}$ communication consumes up to $10\times$ the power of typical operation with $12\,\mathrm{mA}$ [45] and up to $\approx 3.3\times$ the power of *maximum* computing effort of $38\,\mathrm{mA}$ (at default frequency of $36\,\mathrm{MHz}$) [138]. Additionally, the time that must be spent on transmissions and communications is the single largest contributor to time delays, as analysed by Zenger [228] and Huth [92]. To achieve better energy and time efficiency, actual implementations should pay particular attention to these points. Specifically, this means that both highly complex calculations and heavily interactive protocols should be avoided.

The constraint to avoid excessive communication carries two additional aspects: First, exchanging messages about the key material reveals information about it to the attacker. In the worst case, the key material gets completely revealed [92]. This means that the attacker might directly learn information about the key material from this message exchange. And second, leaking information about the key material reduces the achievable secure key rate C_s. Equation (2.1) defines the upper bound for the secure key rate, which in terms of knowledge at the participants A, B and E can be stated as $C_s \leq \min(\mathbb{I}(A;B), \mathbb{I}(A,B|E))$. Hence, leaking information about the key material to the attacker E effectively reduces the achievable key rate.

The next constraint is that blockwise processing of the key material is reduced or even completely removed. This is again a direct consequence of the requirement for efficiency: blockwise processing requires buffering of key material, which can only be processed once the block is full. Hence, it implies wait times and thus delays.

As an extension of this, another constraint is that the usage of blockwise statistics should

Table 6.1.: Summary of the defined constraints and their respective objectives.

Constraint	Objective	Targeted Requirement
Avoid (excessive) communication	Reduce information leakage	Secrecy
	Reduce processing complexity	Efficiency, Reliability
	Increase time efficiency	Efficiency, Key rate
	Increase energy efficiency	Efficiency
Avoid complex computations	Increase time efficiency	Efficiency, Key rate
	Increase energy efficiency	Efficiency
Avoid blockwise processing	Reduce delays	Efficiency
No blockwise statistics	Avoid potential biases	Uniformity
No global statistics	Avoid potential biases	Uniformity
	General feasibility	(Feasibility)

be avoided. This primarily fulfills the requirement for uniformity of the keys, since biases can be avoided with this. To illustrate how the utilization of blockwise statistics can introduce biases, we present two examples in which this is exactly the case: first an approach where the total number of zeros and ones in the quantized measurements is forced to be equal to generate a uniform-like distribution [200]; and second a commonly applied approach where blockwise averages are used for quantization used, e.g., in [133, 12, 192, 75].

The quantization approach in the first example assumes that in a given block of quantized measurements the numbers of zeros and ones are roughly equal. Hence, this approach adapts itself until this equal number is realized in the output. This assumption might hold for very large block size by the *law of large numbers*. However, if we consider a practical block size of 16, this actually introduces the following bias. Given that the original random numbers follow the uniform distribution $\mathcal{U}(0, 2^{16})$, the numbers of ones of the quantized values follow a normal distribution $\mathcal{N}(8, 2)$. This transition to normal distribution is due to the binary representation of the real original values, i.e., many ones and few zeros for small input values and vice versa for large ones. If the quantization algorithm now excludes values with too few or too many ones, the tails of the normal distribution are essentially "cut off". This bias is visualized in Fig. 6.1a. Since the values represented by the red zones in the figure are removed by the algorithm, the range of possible results is significantly reduced. Put simply, the results containing primarily ones or zeros, i.e., large or small input values, respectively, are simply discarded. This reduction of possible results also narrows the search space for an attacker — which in turn increases the attacker's chances of success.

The second example concerns quantizations approaches which calculate averages over blocks of measurements to determine quantization thresholds. Such approaches can lead to biases if the different effects of small scale and large scale fading are not carefully differentiated. The core idea of this is shown in Fig. 6.1b: you can see the progression of a channel characteristic that is subject to large scale fading due to movement in space — this is the coarse progression of the line. The changes of the small scale fading which are actually interesting for CRKG are shown as smaller fluctuations. A blockwise quantization approach now might treat the whole progression as a single block for quantization. Given that the mean is used as threshold for quantization, the respective calculation of the mean

over this block yields the red dashed line in the figure. If now values above this line are quantized as ones and values below as zeros, this inevitably leads to a distorted result. The result will be dominated by the more predictable large scale fading, i.e., the coarse progression of the line. However, CRKG and its security relies on the usage of the small scale fading, which is completely neglected. Hence, such approach using blockwise statistics might introduce biases affecting CRKG itself as well as its security.

Finally, we exclude the usage of global statistics, i.e., statistics calculated over the complete data set. In addition to the same reasoning as the previous constraint, this one is primarily due to plain feasibility: in practice, respectively in the application of the protocols, the complete data sets are simply not available, meaning that such calculations are practically not realizable.

A summary of the defined constraints along with the objectives sought in each case can be found in the Table 6.1.

In accordance with these requirements and constraints, we define the following two **general design decisions**, which are taken into account in all solution approaches and further design work.

Blind solutions Every solution design is intended to operate in what we call a *blind* mode. In this context, *blind* means that the legitimate communication partners do not have to exchange messages in order to achieve the respective goal. They therefore do not see the input of their partner, hence the term *blind*. This design decision is primarily a direct implementation of the constraint that communication should be avoided. Achieving this constraint would directly yield the advantages mentioned above. In addition, a completely blind solution would have the advantage that the previously necessary authenticated channel for this communication would be omitted as a requirement for CRKG. As a result, the corresponding solution would be much easier to implement in practice and may find acceptance more quickly. Therefore, this can be considered the most important design decision — all solutions should work preferably *blind*, that is, without any kind of interaction between the legitimate communication partners.

Online solutions All processing steps of a solution should be executed in an *online* mode. This means that measurements of channel properties are processed immediately, in order to avoid potential biases of blockwise processing, any kind of buffering or the like. For our specific use case, that means that as soon as a CIR is acquired, it is instantly processed by the corresponding CRKG pipeline. This applies analogously to the respective substeps, such as preprocessing, quantization or IR, as well as to their substeps in turn.

By following these two design principles, the defined constraints are effectively taken into account and thereby the corresponding requirements are pursued.

7. Enabling CIR based CRKG

Parts of this Chapter are published in [193, 194].

In this chapter, we present methods and solutions necessary for basic processing of CIR in the context of CRKG. Since CIR have a different basic structure than, for example, RSSI values, a certain amount of preprocessing is necessary to make the measured values suitable for further processing.

Mainly, we consider two key issues here: first, how to synchronize the CIR observation in time at the legitimate participants — this is required to avoid additional errors as described in Section 5.3. And second, how the parts relevant for key derivation can be extracted from the overall measurement of a CIR. The second point is mainly due to the fact that the different measurement methods record varying amounts of noise before and/or after the parts of the CIR that actually carry information. For correct processing, this noise must be removed or at least reduced to a minimum.

7.1. Blind Synchronization of CIRs

In the following we describe a *blind* solution to synchronize CIRs at the legitimate communication partners. To the best of our knowledge, this problem has not been considered before and we present here the first solution.

7.1.1. Problem Statement

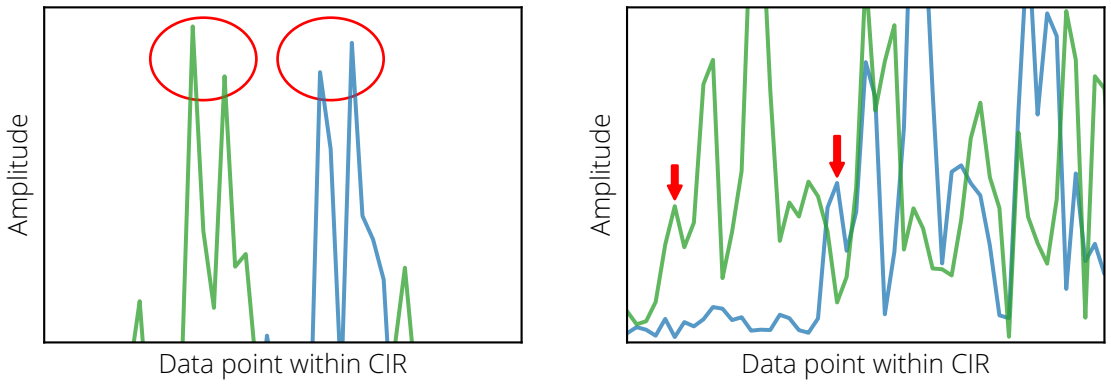
Our initial analysis of the measurement data in Section 5.3 has shown that the raw measurement data are not synchronous — on average there are absolute deviations of $\delta = 2$, with the mean value for the data set *attack* even being $\delta = 4$. Such a deviation means that the subsequent quantization inevitably generate bit strings which contain significant differences between Alice's and Bob's instance. The extent of this additional error can be illustrated intuitively, if one starts from actually perfectly equal CIRs and quantizes each measuring point within. After this quantization into a bitstring, the δ corresponds to a shift of the reciprocal CIRs by exactly δ bits. The following small example illustrates how such simple shifts produce significant differences in the quantized measurements:

Alice bits	01011011 011010100
Bobs bits (shifted by 2 positions)	00010110 110110101
Induced errors	01001101 101100001

These errors would further degrade the reciprocity of the bit strings in addition to the differences that exist anyway due to interference and non-reciprocal hardware paths.

Basically, these time shifts are due to the fact that the respective devices per se do not operate time-synchronously. As a result, the measurements are recorded at minimally different times, which causes the corresponding shifts within the measurement vectors.

But in addition, our analysis shows that state-of-the-art as well as common approaches to related problems like maximum based algorithms (MAX), leading edge detection algorithms (LED), and their modern derivates employed in millimeter range UWB location applications [137, 107] (max ration, MR) do not perform sufficiently in the described scenario. Due to the specific progressions of the respective CIRs, these methods do not achieve an effective synchronization of the measurements. Figure 7.1 depicts examples in which the straight forward approaches fail and their respective causes: The first example in Fig. 7.1a shows LOS components of two reciprocal measurements at Alice and Bob. It is visible that either this component is composed of two peaks or the LOS component is closely followed by another component of similar power. In any case, this strong component consists of two peaks with approximately the same amplitude. The reason why the MAX algorithm generates wrong synchronizations in such cases is that the amplitudes of these two peaks are *interchanged*. Therefore, the MAX approach would deliver one peak as maximum on one side and vice versa on the other side. The second example in Fig. 7.1b depicts a case, where leading edge algorithms like LED and MR fail. Here, the CIR does not have a continuous strictly monotonically increasing first edge. In addition, the respective discontinuities have different amplitudes. This results in the case that the edge of the "smaller" discontinuity is not recognized as a peaks rising edge by one communication partner and only the following edge is used. In the corresponding reciprocal measurement, on the other hand, the slightly higher amplitude of the discontinuity means that it is recognized and used as the leading edge.



(a) Exchanged maxima points

(b) Leading edges which neither are strictly monotonous increasing nor have reciprocal amplitudes

Figure 7.1.: Examples for cases where straight forward synchronization fails. The green line is Alice's CIR, the blue one Bob's.

Since CRKG, especially in terms of secret key rate and secure key bits, is driven by the reciprocity of the respective channel estimations, it is crucial to minimize any influence diminishing this reciprocity. Mismatches in synchronization inherently lead to decreased reciprocity, since even equal reciprocal measurements differ, if they are out of sync. In state-of-the-art approaches to CRKG, this problem has not been addressed yet.

Therefore, to make CRKG as efficient as possible, an approach is needed which can determine a sufficient synchronization while keeping the information leakage low. To fulfill the second part and to adhere to our general design decisions, we only consider completely *blind* approaches

In consequence, this means that such *blind* approaches have solely local knowledge, i.e., no information about the reciprocal measurement is available. Hence, there is no feedback about the performance of the applied algorithm.

7.1.2. Solution Design

In order to present the solution design adequately, we first describe the requirements that must be met by a corresponding solution. Then we specify our concrete solution.

Requirements

Our analysis of common as well as state-of-the-art approaches showed, that in the worst case, i.e., the worst analysed scenario, the best of these approaches only achieves $\Delta^0_{MR} = 0.49$. This means, that none of the traditional and modern approaches achieves optimal synchronization in more the 49% of the observed CIRs. The reasons for this lacking performance are shown in Fig. 7.1: First, the existence of multiple local maxima within the first arriving cluster with non-reciprocal amplitude differences (Figure 7.1a), which cause errors for maximum based synchronization approach. Second, non-strictly monotonous increasing first edges (Figure 7.1b), which are especially hard for leading edge detection algorithms.

In relation to the described error causes, an optimal algorithm would expose 2 major properties:

Uniqueness A *single data point* within an observation needs to be identified, which thereupon acts as this CIRs anchor for synchronization.

Robustness The *same* unique time anchor needs to be identified in the reciprocal measurements, irregardless of noise and interference.

An explicit non-requirement is the preservation of edges. Since the main error causes in Fig. 7.1b are superfluous edges in the leading edge, it is not necessary to preserve such artefacts. Even more, the removal of such interference based edges would be favourable. In consequence, this non-requirement rules out filtering solution, which aim for perfect waveform structure preservation at the cost of computation complexity, e.g., wavelet filtering [209].

Finally, we disregard solutions which exchange information about the received CIRs. Although they might exhibit good performance, following the argumentation in Chapter 6 surplus information exchange about the CIR should be avoided. Hence, we only consider *blind* approaches.

Solution

The two major requirements for the solution are achieved by the following means.

Uniqueness can be achieved by selecting the global maximum of a given CIR. As shown above this approach alone does not perform well, since it is lacking *robustness*. Hence, we propose to combine it with an algorithm which provides this second requirement.

The *robustness* property can be achieved through two approaches: The first approach is the application of *noise reduction*. This can be realized by employing classical digital Finite Impulse Response (FIR) filters like Butterworth or Chebyshev filters in low- or band-pass mode [153]. Although they are very powerful, they need careful fine-tuning regarding the transmission properties, e.g., baseband and bandwidth. Nevertheless, we still include FIR filters in this evaluation.

The second approach, we are propose, is to use blurring or smoothing filters originating from the realm of image processing. These have the advantage of independence of transmission properties and, additionally, they fulfill the requirements defined above. Through the application of such filters with appropriate parameters, the noisy artifacts of the signal can be eradicated.

Due to their proven strong performance in the fields of image noise reduction and image edge detection [48], as well as signal processing for localization [127, 57], we propose to use a **one dimensional Gaussian filter** for the time synchronization.

According to the system model the single CIR are vectors of values. Hence, we apply a *one dimensional* Gaussian filter to this signal, which is defined as:

$$G_{1D}(x) = \frac{1}{\sqrt{2\pi\sigma}} exp(-x^2/2\sigma^2) \tag{7.1}$$

By applying this filter to the single CIRs, a unique synchronization anchor can be identified by taking the maximum of the resulting filtered signal. This constitutes the overall processing of the newly proposed approach: 1. Filter the obtained CIR with a Gaussian 1D filter. 2. Take the maximum of the filtered signal as time anchor for synchronization. This effectively determines a single point within the vector of values, which we call the *time anchor* τ_{sync}.

Figure 7.2 shows an example CIR after applying a Gaussian 1D filtering. It depicts that the filter successfully removed the aforementioned major error causes. In addition, the Gaussian filtering yields a robust time root for the corresponding CIRs, the robustness of which is thoroughly evaluated in the upcoming section.

7.1.3. Evaluation

We are interested in both the performance and robustness of our approach compared to alternative solutions. Employing the metrics as introduced in Sec. 2.5, we assess to which extent it independently identifies identical anchors in corresponding CIR observations.

Methodology We set out to evaluate the alternative approach on a data set as diverse as possible to cover as many scenarios as possible. Hence, we use the data set *scenarios* for the real world measurement evaluation. Nevertheless, all data sets are representative only for the given scenario. For a broader evaluation of the robustness we hence subsequently

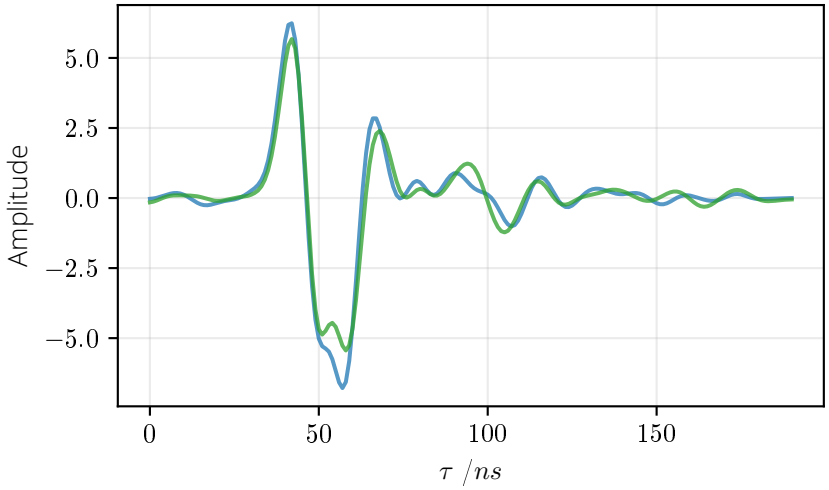


Figure 7.2.: Example of a CIR after applying the Gaussian 1D filter.

evaluate all approaches on synthetic data, generated with accepted UWB models from literature and simulating broad ranges of interference.

We finally evaluate the impact of the improved synchronization for the actual CRKG case, comparing the achieved BDR after quantization with algorithms tailored to key generation [200].

The **synthetic impulse response** used to evaluate robustness in situations as diverse as possible were generated as follows:

The basis for these synthetic data is the IEEE 802.15 UWB channel model as presented in [144], which is a slight modification of the Saleh-Valenzuela UWB model [163]. In addition to the channel model itself, [144] also defines 3 parameter sets for typical indoor UWB propagation. The first parameter set represents a strong LOS scenario, whereas the second and third models describe typical Non-Line of Sight (NLOS) setups. Finally, a fourth parameter set is defined, which is artificially generated and aims to resemble an extreme NLOS case. We call these four parameter sets Model 1-4.

Since the channel model itself does not support the generation of correlated measurements, we adapted it to support this in the following manner: In accordance with the channel model described in Section 2.3, the correlated observations were realized by adding two different realizations of independent AWGN to a single observation. This allows us to generate highly correlated observations X_i and Y_i, which also include edge cases as described above.

Since the main task of the approaches is time synchronization, an additional time offset was added to one of the generated observations. Theoretically, the value of this offset does not matter, since the algorithms identify the unique data points within a single realization as anchors. This means, they do not have a global view on both realizations and thereby the actual time offset is irrelevant. It would suffice to show that the approach identifies the same data point in both observations, to demonstrate their feasibility. But to keep the results comparable to the real world, we added similar offsets between the two observations.

Analysing the real world data indicated typical offsets in $t \sim \mathcal{N}(8.3, 11.4)$. We hence used this distribution to generate artificial offsets in the synthetic data.

Various other 1D filters exist that can be used as **comparable approaches**. Within the domain of CRKG the time synchronization problem has not been addressed. Nevertheless, there exist alternative solutions for, e.g., time calibration. We apply these approaches to the same data sets for comparison.

A simple maximum detection (*MAX*) and state of the art leading-edge detection mechanisms (*LED* and *MR*) [137, 107] are used as baselines.

In addition, we compare our one-dimensional Gaussian filter to other, more complex digital FIR filters, for which we choose Butterworth and Chebyshev Type II filters, both realized as low-pass and band-pass, as representative candidates. These filters as parameterized to the acquired real world measurements — additionally, low and high cut frequencies as well as the filter order were determined manually to minimize $1 - \Delta^0$.

To assess the difference between Gaussian and other smoothing filters, we also compare to moving average or one dimensional uniform filter, Savitzky-Golay filter, Sobel filter, Hilbert filter, Wiener filter, and the spline interpolation. Again, the respective algorithm parameters are determined a priori, to optimize their performance with respect to $1 - \Delta^0$.

Results with real world impulse responses The results for the physical measurements used for this comparison are shown in Table 7.1. Each result shows Δ^0 as well as mean μ and standard deviation σ of the respective vector $\vec{\Delta}$. The different compared approaches are listed in the rows, with the last row showing the proposed 1D Gauss filter. For each scenario we highlight the best results with respect to Δ^0.

Our Gaussian filter outperforms all other approaches in all but one scenario. Only in scenario $StaticD$, the manually optimized Chebyshev band-pass performs slightly better (0.95 vs. 0.93). Both band-pass approaches (Butterworth Band and Chebyshev Band) fare equally well in the other static scenarios.

In the dynamic scenarios with interference, which are most relevant to the CRKG problem, our Gaussian filter performs better than all competing algorithms with advantages up to 13%, even over the *best* baseline approach with global knowledge.

It is also the most robust approach with consistently high performance across all different measurement scenarios.

Results with synthetic impulse responses The results for simulated data used for comparing the different approaches are shown in Table 7.2. Again, the new approach shows highest robustness, outperforming all others in all but one model.

By recalling the model setup of Model 1 is clear, why the simple leading edge detection algorithm is slightly better in this single case: Model 1 represents a strong LOS channel, as described in [144], which is characterized by a strong first cluster in the CIR. This is distinguished by an unusually strong leading edge, which is also strictly monotonously rising. These two properties are highly advantageous for leading edge detection algorithms. Nevertheless, the Gaussian 1D approach still performs almost on par, even in this case ($0.88/0.86$).

For the generic NLOS models, our Gaussian filter again significantly outperforms all other approaches, with advantages of up to 31% over the *second-best* approach.

In conclusion, the Gaussian 1D approach delivers best results in most cases: it comes in second only for two scenarios that have minor relevance for CRKG, achieving almost identical performance with the respective winner. It achieved optimal synchronization for

Table 7.1.: Solution performance in the different scenarios of data set *scenarios*

Algorithm	Δ^0 and (μ, σ) of Δ						
	Scenarios						
	Inter. A	Inter. B	Moving Eve	Static A	Static B	Static C	Static D
MAX	0.61 (0.19, 1.20)	0.65 (0.57, 2.74)	0.56 (0.09, 0.65)	0.64 (0.00, 0.59)	0.66 (0.10, 0.57)	0.59 (0.00, 0.63)	0.69 (0.05, 0.96)
LED	0.69 (0.04, 0.61)	0.67 (-0.54, 9.25)	0.59 (0.12, 10.19)	0.61 (-0.60, 9.52)	0.59 (1.15, 18.1)	0.56 (0.45, 12.11)	0.56 (0.64, 14.06)
MR	0.82 (0.06, 0.41)	0.74 (-0.04, 0.62)	0.82 (0.09, 0.40)	0.83 (0.07, 0.39)	0.77 (0.10, 0.46)	0.85 (0.07, 0.38)	0.81 (0.09, 0.42)
Butterworth Low	0.61 (0.20, 1.21)	0.65 (0.57, 2.74)	0.57 (0.10, 0.65)	0.65 (0.01, 0.59)	0.66 (0.11, 0.57)	0.60 (0.01, 0.64)	0.70 (0.06, 0.96)
Butterworth Band	0.87 (0.38, 3.41)	0.77 (0.02, 1.95)	0.93 (0.00, 0.25)	**0.96 (0.00, 0.18)**	**0.91 (-0.00, 0.29)**	**0.92 (0.01, 0.27)**	0.92 (0.13, 0.82)
Cheby2 Low	0.83 (0.17, 0.81)	0.76 (0.37, 1.89)	0.93 (0.13, 0.81)	0.95 (0.04, 0.21)	0.87 (0.09, 0.45)	0.91 (0.06, 0.30)	0.79 (0.98, 2.55)
Cheby2 Band	0.88 (0.04, 0.78)	0.77 (0.38, 1.66)	0.93 (0.05, 0.25)	**0.96 (0.01, 0.18)**	**0.91 (0.00, 0.29)**	0.91 (0.01, 0.29)	**0.95 (0.03, 0.20)**
Uniform	0.80 (0.24, 1.33)	0.72 (0.38, 1.87)	0.91 (0.10, 0.47)	0.90 (0.04, 0.30)	0.88 (0.03, 0.33)	0.89 (0.05, 0.31)	0.68 (0.95, 1.98)
Hilbert	0.61 (0.19, 1.20)	0.65 (0.57, 2.74)	0.56 (0.09, 0.65)	0.64 (0.00, 0.59)	0.66 (0.10, 0.57)	0.59 (0.00, 0.63)	0.69 (0.05, 0.96)
SavGol	0.82 (0.16, 0.87)	0.76 (0.18, 1.20)	0.91 (0.10, 0.47)	**0.96 (0.02, 0.18)**	0.86 (0.07, 0.35)	0.89 (0.08, 0.32)	0.90 (0.12, 0.93)
Sobel	0.42 (0.27, 2.90)	0.34 (0.10, 5.08)	0.39 (0.16, 2.26)	0.43 (0.14, 2.42)	0.34 (0.78, 6.17)	0.41 (0.01, 2.06)	0.28 (0.01, 6.31)
Wiener	0.64 (0.20, 1.19)	0.66 (0.38, 1.76)	0.57 (0.10, 0.64)	0.66 (0.01, 0.58)	0.72 (0.10, 0.51)	0.62 (0.00, 0.61)	0.76 (0.07, 0.92)
Spline Interpolation	0.77 (0.61, 2.06)	0.67 (0.57, 2.28)	0.85 (0.27, 1.06)	0.65 (0.92, 2.07)	0.78 (0.10, 0.45)	0.80 (0.51, 1.50)	0.76 (0.43, 1.63)
Gaussian 1D	**0.90 (0.12, 1.54)**	**0.78 (0.13, 1.39)**	**0.94 (0.00, 0.23)**	**0.96 (0.01, 0.19)**	**0.91 (0.03, 0.29)**	**0.92 (0.02, 0.27)**	0.93 (0.04, 0.26)

Table 7.2.: Solution performance in different synthetic scenarios

Algorithm	Δ^0 and (μ, σ) of Δ			
	Saleh-Valenzuela Models			
	Model 1	Model 2	Model 3	Model 4
MAX	0.50 (0.01, 3.47)	0.42 (-0.03, 5.96)	0.36 (0.53, 9.63)	0.36 (0.23,16.37)
LED	0.76 (0.04, 1.26)	0.66 (0.02, 1.80)	0.61 (-0.11,12.12)	0.49 (0.18,13.01)
MR	**0.88 (-0.03, 1.37)**	0.77 (0.16, 5.56)	0.62 (0.46,12.38)	0.44 (-0.03,17.11)
Butterworth Low	0.47 (-0.05, 4.33)	0.40 (0.01, 6.20)	0.41 (-0.22, 9.27)	0.36 (0.23,16.37)
Butterworth Band	0.56 (0.15, 3.79)	0.46 (-0.08, 5.42)	0.42 (0.47,10.36)	0.37 (0.29,17.73)
Cheby2 Low	0.53 (-0.03, 3.12)	0.45 (0.06, 5.71)	0.40 (0.24, 9.24)	0.38 (0.19,15.57)
Cheby2 Band	0.54 (0.21, 4.36)	0.46 (0.21, 6.16)	0.41 (-0.01,10.87)	0.38 (0.77,16.31)
Uniform	0.68 (-0.08, 2.42)	0.49 (-0.01, 4.84)	0.46 (0.05, 8.26)	0.43 (-0.26,13.59)
Hilbert	0.50 (0.01, 3.47)	0.42 (-0.03, 5.96)	0.36 (0.53, 9.63)	0.36 (0.23,16.37)
SavGol	0.83 (0.02, 2.30)	0.59 (0.90, 2.69)	0.55 (0.32, 7.86)	0.49 (-0.42,15.17)
Sobel	0.33 (0.24, 8.57)	0.27 (-0.34, 9.39)	0.27 (-0.17,16.85)	0.26 (-0.90,22.32)
Wiener	0.51 (-0.02, 3.29)	0.42 (0.05, 5.67)	0.38 (0.27, 9.18)	0.37 (0.14,13.88)
Spline Interpolation	0.50 (0.01, 3.47)	0.42 (-0.03, 5.96)	0.36 (0.53, 9.63)	0.36 (0.23,16.37)
Gaussian 1D	0.86 (0.45, 11.61)	**0.83 (-0.01, 15.72)**	**0.82 (-0.21,11.27)**	**0.80 (-0.82,16.49)**

over 90% of the comparisons in measured data sets, and for over 80% even of the synthetic inputs simulating artificially bad conditions. This consistent good performance indicates a good general applicability, since it performs stable and robust under a broad variety of potential settings.

Bit Disagreement Rate To visualize the effect of time synchronization and the advantages of applying our solution in the context of CRKG , we show the BDR of the quantized *scenarios* measurements, with and without our synchronization approach.

Figure 7.3 shows the bit disagreement rates when applying our filter before quantization according to CRKG . The upper part of violin plots visualizes the BDR without the application of our proposed approach, whereas the lower green part shows the results after applying it. While the BDR remains high for all scenarios, the results clearly demonstrate the benefit of applying our approach: Single improvements reach up to 21% reduction in BDR for scenario *Static A*. Averaging over all obtained measurements the improvement exceeds 18%.

This resulting BDR over the total measurements verifies that the slightly higher standard deviation our approach exhibits has no significant influence on the final performance.

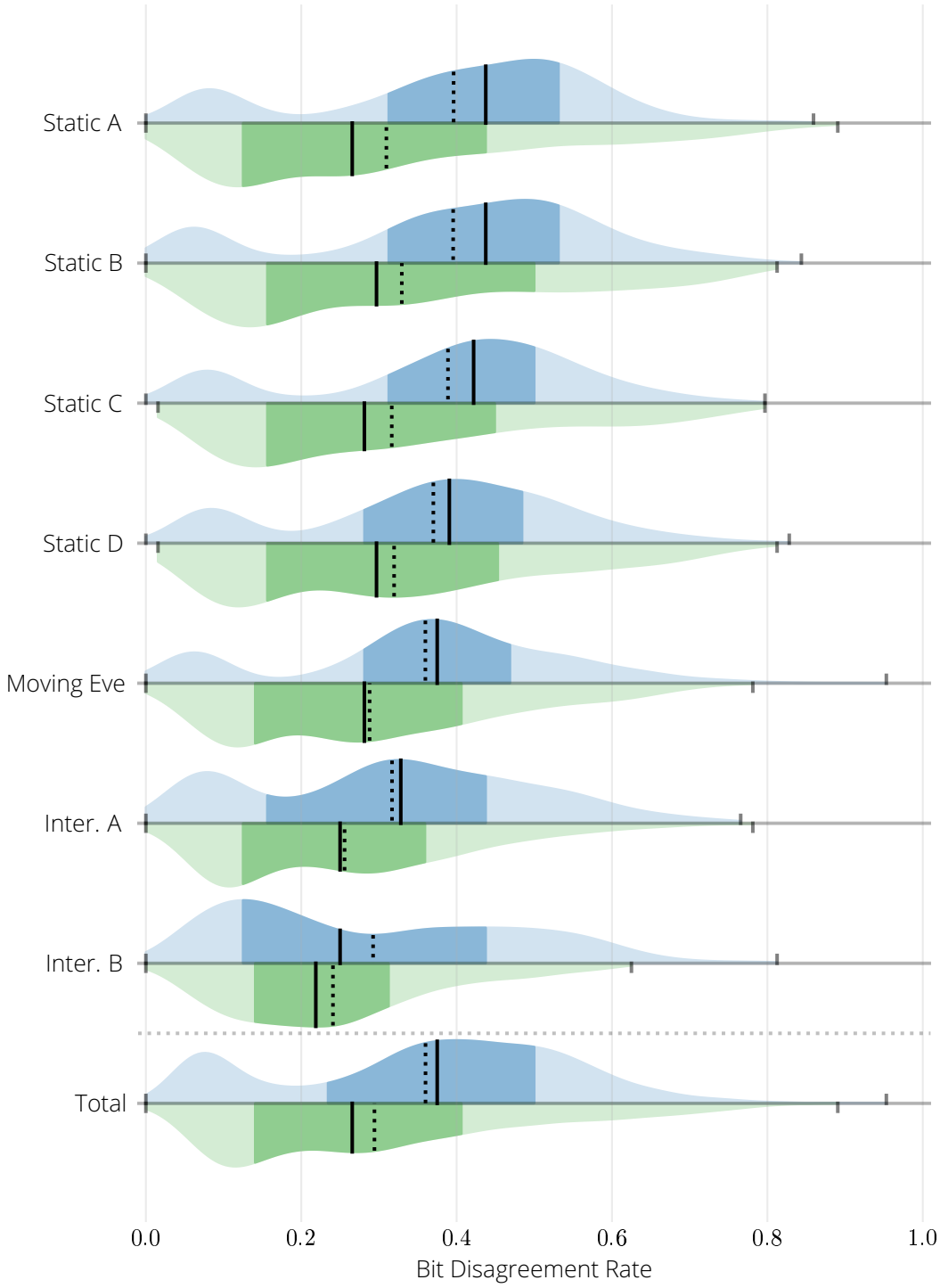


Figure 7.3.: Improved BDR after application of the proposed algorithm. The blue upper part shows the BDR without our approach; the lower green part with our approach respectively.

7.2. Noise Removal

By definition noise is "[...] any unwanted signal that tends to interfere with the required signal." [39]. Thus, to minimize the effects of such interference, the impact of noise on processing must be minimized.

The impulse responses, as they are estimated respectively captured by the hardware, contain not only information carrying parts, i.e., the actual impulse response itself, but also additional noise *before* and *after* the information bearing parts. The corresponding examples for this are to be seen for instance in Fig. 4.3 and even more pronounced in Fig. 4.8. Following the notation of Eq. (2.5), a certain part of the measurement before h consists exclusively of noise n, followed by the actual impulse response $h + n$, which then changes again into a part consisting of n only. Since we assume AWGN, the noise cannot be expected to be reciprocal in any way. This means that after quantization, there is a high probability that bit mismatches will occur in the noise-only parts. In turn, this means that the more pure noise is included in the processing, the more the respective BDR increases. Since a high BDR complicates information reconciliation to the point of failure, this should be avoided at all costs. Thus, the parts of the CIR measurements consisting entirely of noise must be removed for the basic effectiveness and functionality of the CRKG key exchange.

With reference to Fig. 4.3, the Fig. 7.4 shows the aim of such noise removal. The red areas are considered pure noise and need to be removed from the measurement data.

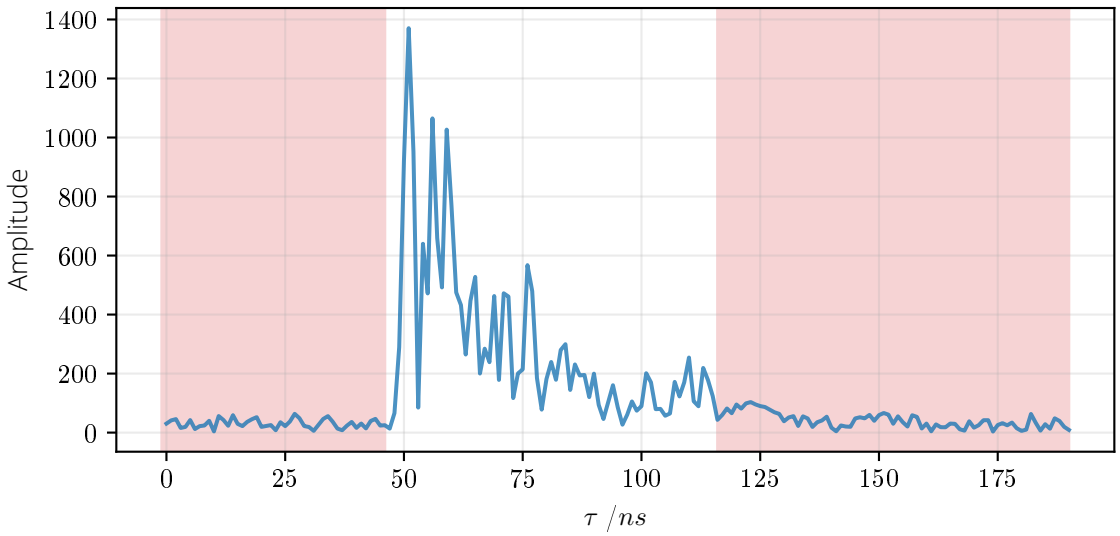


Figure 7.4.: Example of the aimed for noise removal — the red shaded parts need to be removed from the measurement data.

Basically, this problem consists of finding the information-bearing part, i.e., the impulse response itself, and its boundaries within the measurement. Thus, it can be divided into the two partial questions, how to locate the beginning and the end of the impulse response, respectively.

With respect to the data set *attack* the following approach was proposed in [194]. In this data set, it holds for each measurement that the actual impulse response is in the back half of the measurement. Thus, the front half could be used to determine a threshold t from the

noise level by calculating the average of these values. Recalling the vector representation of h in Eq. (5.1), this can be written as

$$t = \frac{1}{N/2} \sum_{i=0}^{N/2} v_i, \tag{7.2}$$

where N is the number of data points within the recorded CIR. Based on this, a thresholding procedure was used to determine both the beginning and the end of the CIR: if two consecutive values exceeded the threshold t, this defines the beginning of the impulse response; two consecutive values below define the end.

Considering all available data sets, this approach has the following drawbacks: first, it cannot be generally assumed that the front half consists exclusively of noise and is thus available as a basis for determining the noise level. Second, this method basically implements a leading edge detection approach, which we saw in the previous section can be ineffective on general measurement data.

Starting from the last argument, we therefore adapt the processing for general measurements as follows: Instead of the leading edge procedure described, we use the time anchor determined by time synchronization to define both the beginning and end of the CIR: For the starting point, we include five more values before the time anchor to account for any preceding fluctuation. Starting from this index, the next 64 values including the starting point are used as CIR. This value is based on the reflection argument from Section 4.1 — within the largest measurement room used, this value still allows for at least two complete reflections over the longest diagonal. Most reflections arrive much earlier and within the selected interval. In addition, reflections after this limit are usually of such low power that distinguishing them from noise is difficult. Both of these values were determined empirically.

Originating from the time anchor τ_{sync}, we determine the start and end points, τ_A and τ_B, of the CIR as follows:

$$\tau_A = \tau_{sync} - 5 \tag{7.3}$$

$$\tau_B = \tau_{sync} + 58 \tag{7.4}$$

$$= \tau_A + 63 \tag{7.5}$$

Including the values at indices τ_A and τ_B this yields a CIR as vector of 64 values.

It is worth noting, that both parameters of the noise removal, i.e., 5 and 64, depend on the sampling rate of the hardware in use, which is $1\,\mathrm{ns}$ in our case. If these actual values are to be used for differing sampling rates, the parameters must be adjusted to represent the respective absolute times.

Part IV.

Attacks against Channel Impulse Responses

8. “Classical” Attacks on CIR-CRKG

Parts of this Chapter are published in [194, 198, 196].

In the following we discuss two different approaches to attack CRKG processing at the common source of randomness, specifically whether this common source is as unpredictable as assumed. The designation as “classical” here is solely to distinguish it from the next chapter, where machine learning-based approaches are presented.

The basic question posed by all these attacks is invariably the same, namely whether the assumption is justified that an attacker can record exclusively uncorrelated observations of the legitimate channel impulse responses.

If this assumption does not hold, a key derivation-based on these physical properties or properties derived from them would have to be considered insecure. This is due to two main properties of key derivation based on channel properties:

First, the key generation is completely deterministic. In the case of our attacks, this means that the same input into the CRKG process will generate the same key. This property explicitly allows Alice and Bob to generate the same key, and also requires the channel to change, resulting in new entropy leading to new keys. Hence, if the attacker has the same input material as Alice or Bob the same key can also be generated.

And second, the input materials at Alice and Bob are not perfectly the same, so the CRKG process itself removes a certain amount of differences. The specific amount depends on the implementation used. This fact, however, also favours attacks against the input material — the attacker thus does not need to have a perfect replica of the key material as in Alice or Bob, but it is sufficient to get hold of reasonably similar input material. The CRKG processing itself then removes a certain number of differences, which makes the corresponding attack much easier.

Following this argument, the attacks presented here try to predict or infer the input material of CRKG allowing the attacker to derive the same key as Alice and Bob. We first present the two attacks, their ideas, realizations and results, and then put the achieved results into perspective by showing their implications and compare them to the state-of-the-art.

8.1. Rationale and Attack Idea

We now describe the core idea of the presented attacks and the rationale behind it. The key idea behind the attacks is that certain parts of a CIR can be computed deterministically and that the assumption of uncorrelated observations at the attacker is based on very strong assumptions. By combining the knowledge of what the attacker can derive from each part, a potential attack surface is created — specifically, that an attacker can predict the input material for the key derivation or derive it from her observations.

Regardless of the key derivation itself, the following observations can be made about the channel properties between the respective communication partners:

CIRs are well described statistically Taking existing channel models into account, for example the Saleh-Valenzuela model or the IEEE channel model (cf. Section 2.3), we know what general progression can be expected from a CIR: Under LOS conditions, there will be a dominant LOS component. This is followed by the different multipath clusters, which arrive with exponentially distributed time intervals and with exponentially decaying power. Within the multipath clusters the single rays exhibit the same behaviour as the cluster: arriving with exponentially distributed time delays and exponentially decaying power — only with differing rates. In NLOS environments no LOS component is present.
Keeping such statistical knowledge about the impulse responses in mind, the general progression of a CIR measurement is known. An example is visualized in Fig. 2.10.

Main CIR components can be calculated deterministically Proceeding from the previous point, the main components of the impulse response are the LOS component and the multipath components. Given a known transmission environment and known terminal positions, these components can be calculated deterministically: The LOS component is primarily determined by the spatial distance between the communication partners. The multipath components result from the corresponding reflections of the environment and the thereby extended distance. This extended distance in turn determines the multipath clusters amplitude and arrival time through an increased path loss and propagation time.
Figure 8.1 illustrates a simplified and idealized shape of an impulse response and how it can be derived from the current environment.

Interferences originating from the environment affect all CIRs equally That is, interferences from sources outside the system model are present at all terminals in the same way. For example, if such causes yield a signal drop between Alice and Bob, it will also be visible in Eve's observation.
Intuitively, this can be expressed in reference to Eq. (2.5): here we can decompose the noise term n into two parts. A local part n_{loc} for noise, which depends specifically on the terminal, and a global part n_{glob}, which stands for system-wide or system-external causes. Thus, $n = n_{glob} + n_{loc}$, where n_{glob} is the same for all participants.
Based on this, the attacker is potentially able to infer abnormalities in the legitimate CIR from abnormalities in her observation.

More concretely associated with CRKG is the security assumption that an attacker located further than half a wavelength away from the legitimate participants can only observe un-

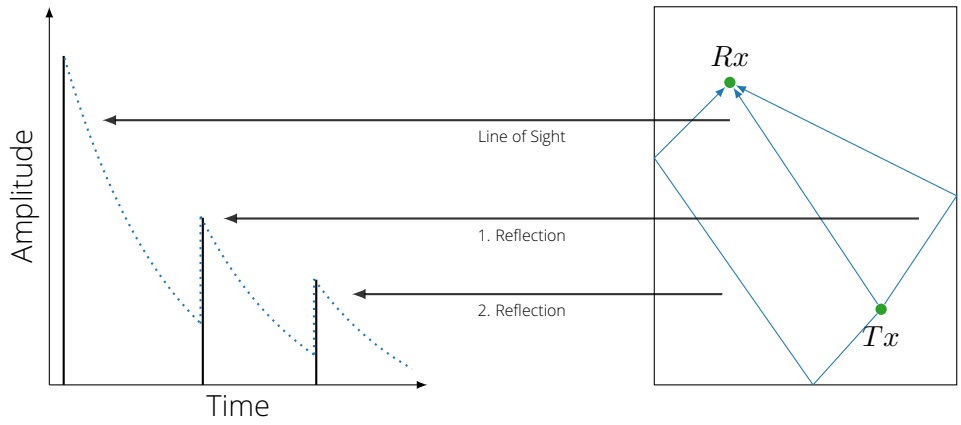


Figure 8.1.: The simplified general course of a CIR and how it can be reconstructed from the transmission environment.

correlated channel properties. However, it must be pointed out that this assumption is in turn based on further assumptions, which in practice must be considered quite improbable (cf. Section 2.3). These assumptions are: no dominant LOS component, uniformly and densely distributed scatterers and all multipath clusters having the same power (cf. Section 2.3).

Considering a typical application environment for CRKG, indoor office space, the concrete assumptions have to be evaluated as follows: The scatterers will not be densely and uniformly distributed over all possible angles of arrival. Possible scatterers, e.g., furniture, typically represent only a small part of the room geometry, i.e., they are sparsely distributed. Given the cuboid shape of typical indoor rooms, the walls as scatterers are also not uniformly distributed. As described by the statistical channel models, the multipath clusters in indoor scenarios do not have the same power. And finally, within a room a direct LOS connection between two terminals and the respective LOS component in the CIR is likely.

Thus, it is justified to question the security assumption of uncorrelated attacker observations.

Further, the CRKG processing itself eases attacks against the common source of randomness used: due to non-reciprocal properties of the estimated CIR, caused by, e.g., non-reciprocal noise and differing hardware paths, the observations of the legitimate partners themselves are not perfectly equal. Hence, in the IR phase, a certain amount of differences is removed by design. Therefore, an adversary attacking the source of randomness does not need to predict or infer perfectly matching CIRs. As long as the attackers resulting CIRs are within the range of error of the legitimate partners, the subsequent key derivation will generate the same key. Thus, it is sufficient for the attacker to reproduce the CIR within this fuzziness to successfully compromise the key exchange.

In summary, the issues described represent a non-negligible attack surface: If the attacker has knowledge about the transmission environment, she can predict certain parts of the CIR and thus the input data for the key derivation. Additionally, due to the assumptions used, completely uncorrelated observations of the channel properties cannot be postulated at the attacker. The attacks presented here address precisely the combination of these two aspects. It is investigated to what extent an attacker can exploit these points to predict or infer the key material and what accuracy can be achieved.

8.2. Attacker Model

All attacks presented in this chapter are modelled using the same attacker model. To keep the presented attacks as general as possible, the model restricts the attacker as little as necessary.

In general, we assume a plain passive, eavesdropping attacker. This means in particular:

- The attacker only observes the respective system, in our case Alice, Bob and the environment they are residing in.
  This explicitly excludes modifying interactions with the communication infrastructure as well as with the physical environment. Here, "interaction with the physical environment" refers to active changes of this environment with the goal of influencing the transmission properties of the wireless communication, such as proposed in [96]. This restriction is included to render the attack both as strong and as practical as possible.

- The attacker is computationally restrained, i.e., he does not have infinite computing power.
  Regardless of how realistic the assumption of infinite computational power is, such an assumption would mean that the attacker has the ability to perfectly simulate wireless transmissions to any level of detail — this would be equivalent to directly breaking CRKG as described above. Therefore, we must exclude infinite computational power for the attacker in our context.

In addition to this basic model, individual extensions to this model become necessary due to the concrete procedure during the presented attacks. For the two attacks presented in this chapter, this means in concrete terms:

Channel model-based attack The attacker needs representative CIR measurements with the help of which the corresponding channel model is adjusted to the actual environment (see Section 8.3).
This can take place completely independently of the actual attack, which significantly increases feasibility. Such measurements can easily be realized as follows, in ascending order of effort for the attacker, conspicuousness, and expected representativeness of the measurements: passive eavesdropping on a legitimate communication; own dedicated measurements depending on the environment; or own dedicated measurements with special hardware.

Ray Tracing-based attack For this attack, the attacker does not even need actual wireless measurements, but only knowledge about the physical layout of the current wireless transmission environment.
Such knowledge can be easily obtained by inspecting blueprints or by plain visual inspection through video cameras, photos, or mere physical presence.

In summary, the attacker model even with the respective adaptations for the presented attacks is realistic and practically implementable, not only for state-level attackers but also for less powerful attackers.

8.3. Channel Model-based Attack

In the attack presented here, we investigate whether deterministic propagation models are suitable for predicting the main characteristics of a channel impulse response. For this purpose, we use such a channel model and fit it to the actual conditions of our measurements. Based on these adjustments, we then evaluate how well the model can predict legitimate CIRs for the specific measurement space.

In the following, we present the concept and procedure for this attack, the concrete realization as well as our evaluation.

8.3.1. Concept

The attack concept is based on the practically oriented sequence of the attack, which defines three distinct phases in which the attacker acts differently. These three phases can be described as follows:

Phase 1 - Data acquisition A local attacker is physically present in the environment where the future communication of the key exchange will take place. Within this environment the attacker obtains representative CIR measurements with known terminal positions.
This might be realized by performing CRKG itself and recording the measurements accompanied by the respective terminal positions. Alternatively, the attacker can perform active reference measurements, where concrete measurement scenarios are set up and the respective observations are recorded.

Phase 2 - Model preparation During a second preparatory step, the attacker used the obtained reference measurements to adapt his local representation of the propagation environment, i.e., the deterministic channel model used in the attack.

Phase 3 - Actual attack To carry out the actual attack on the random source of key generation, the information from the preparation is combined with the position information from the actual communication to compromise the key generation. The attacker predicts the CIR based on the channel model and the terminal position of the legitimate nodes. This predicted CIR is then used as input for CRKG by the attacker. If the prediction was sufficiently close to the real one, the attacker will inevitably derive the same key as Alice and Bob.
To do this, the attacker does not need to be in the vicinity and theoretically does not need to interact either actively or passively with the legitimate communication partners[1]. Based on the representation developed in phase 2, an attempt is made to reconstruct the readings of the legitimate communication partners. Apart from the terminal locations, no other information has to be available for this step.

All phases are easy to implement in practice, and the attacker does not need to have any special resources in terms of the technology used, computing effort or time. Even the last phase places only little demands on the attacker, since position data can be easily collected (e.g., through physical proximity or visually through windows or surveillance cameras).

[1] From a practical point of view, it would of course be reasonable to record data processed with the attacked key to realize a higher-level target of the attacker, e.g., a breach of confidentiality. For the presented attack on the key exchange itself, however, this is not necessary.

8.3.2. Realization

The main points of realization are the selection of a suitable deterministic channel model and a strategy how to adapt it to the reference measurement. Once this has been selected, the attack can be implemented analogously to the procedure described in the concept.

The main criterion for **model selection** is the determinism of the model, so that bidirectionally reciprocal CIRs can be computed from known positions. Therefore, well-known statistical channel models, such as the Saleh-Valenzuela model [163] or the IEEE UWB channel model [144] are unsuitable for the present case — these could only be used to generate randomized CIRs. In the field of deterministic channel models, the model of Kunisch and Pamp has been well established [108]. Since it was designed primarily for the description of radio channels in buildings it corresponds exactly to the system model used.

At its core the Kunisch-Pamp model calculates the CIR by resolving the respective reflections of the transmitted signal. This is achieved by unwrapping the reflections into so called *virtual sources* or *echos* — based on the sender and receiver position as well as the mirror point, the point of the reflection, a virtual source can be determined. The distance between the virtual source and the receiver can then be used to determine the time of flight of this transmission. This in turn is used as path delay for the respective multipath cluster. The generation of the *virtual source* is visualized in Fig. 8.2.

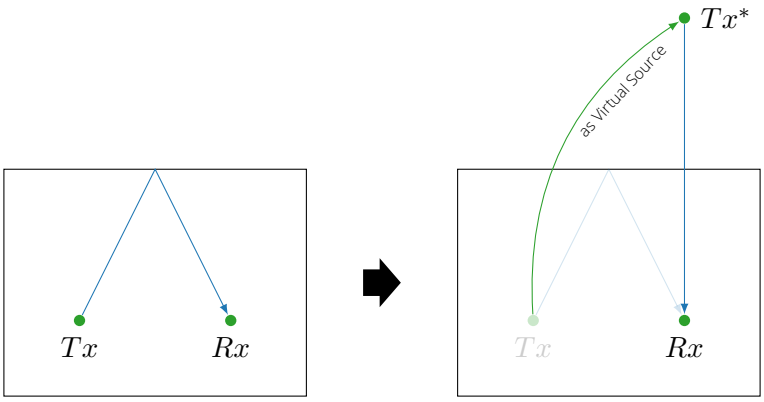


Figure 8.2.: Transformation from reflection to *virtual sources*.

The model can handle multiple reflections by unwrapping the *virtual sources* accordingly. The depth to which this is done is determined by the parameter Q — if the value is set to $Q = x$, then paths with up to x reflections are integrated into the transmission simulation.

One disadvantage of this model worth mentioning is that it supports only rectangular spaces in the current implementation.

The Kunisch-Pamp channel model has a large number of **parameters** relevant to the calculation of channel properties. For our attack, these can be divided into free and fixed parameters as follows:

free, locations These parameters are the concrete input values for the impulse response calculation. Specifically, these are the positions of the legitimate communication partners. Since they will or can move in space, these parameters remain open and potentially change for each calculation of the model.

free, environment These parameters are determined by fitting the model to the reference measurements and represent the adaptation of the transmission environment

by the model. Specifically, these are the selected virtual sources K, the reference distance d_0, the slope of the path loss function α, the cluster decay γ and the cluster gains G_{MP} and $G_{MP,LOS}$.

fixed These parameters are either static for all transmission or have no significant influence on the success of the presented attack. This includes parameters like the center frequency and bandwidth of the transmission or the sampling interval.

A comprehensive description of the channel model's system parameters can be found in the corresponding publication [108].

In addition, value ranges are given in the description of the channel model for the individual parameters, which was derived from the corresponding measurements during model development. The proposed value ranges are listed in Table 8.1.

Table 8.1.: Value ranges of the system parameters of the Kunisch-Pamp channel model as proposed by Kunisch and Pamp [108].

parameter	value range
Q	3
α	2.0 to 3.0
G_{MP}	$-20\,\mathrm{dB}$ to $-16\,\mathrm{dB}$
$G_{MP,LOS}$	$-13\,\mathrm{dB}$ to $0\,\mathrm{dB}$
γ	$9.5\,\mathrm{ns}$ to $12.5\,\mathrm{ns}$
β	1.01 to 1.3

To **adapt the channel model** to the references measurements, i.e., fit the model parameters to them, we proceed as follows:

In general, we use a set of measurements, which in turn consists of different measurement runs r, as the basis for the fitting. Here we denote the set of measurement runs as R and the respective measurements of a single run as M. For the fitting, the mentioned free environment parameters have to be fitted to these measurements. For this purpose, we define a scoring function $f_{score}(\vec{p})$ based on the metrics defined in Section 2.5, which takes a parameter set $\vec{p}$ as input Internally, the function performs the following substeps: First, a CIR h_{syn} is simulated with the particular given parameter set p. This is transferred into the time domain by calculating the magnitude $|h_{syn}|$. Then, the average cross correlation is calculated for all measurements M_r of a measurement run $r \in R$ as

$$s_r = \frac{1}{M_r} \sum_{i=0}^{M_r-1} r(h_{syn}, M_{r,i}) \tag{8.1}$$

Finally, the mean value of the achieved cross correlations is calculated for all measurement runs as

$$f_{score} = \frac{1}{|R|} \sum_{r \in R} s_r \tag{8.2}$$

This is the final value of the scoring function.

The input of the score function is the set of free environment parameters described

above.

$$\vec{p} = \{K, d_0, \alpha, \gamma, G_{MP}, G_{MP,LOS}\} \tag{8.3}$$

By encapsulating the computation of the metric in this way, we can view it as a black box model and use Bayesian optimization to determine the optimal set of parameters [170]. The goal here is to find the set of parameters p^* that maximizes the scoring function:

$$\vec{p^*} = \arg\max_{\vec{p}} f_{score}(\vec{p}) \tag{8.4}$$

The implementation is based on the open source tool *Hyperopt*[2] proposed and developed by Bergstra, Yamins, and Cox [22]. For the optimization, we define the ranges of the parameters as described in Table 8.2b.

As the channel model itself calculates the CIR without taking noise into account, we additionally analyse whether the addition of noise to the simulation would alter the attack success. For this we add pseudo-random noise to the resulting simulated CIR. The values of this noise are sampled in accordance to an additional noise variance parameter V_N. This parameter defines a Normal distribution $\mathcal{N}(0, 10^{V_N/10dB})$, from which the actual values are sampled. For these experiments, V_N is included in the set of optimized parameters.

Finally, there are the fixed parameters of the model. According to the system model, the center frequency and bandwidth of the transmission are set to $4\,\mathrm{GHz}$ and $500\,\mathrm{MHz}$ respectively, and the sampling interval to $2\,\mathrm{ns}$. The model parameters gain G and frequency domain decay exponent β did not yield any effect on the attack success. Hence, we set them manually to $120\,\mathrm{dB}$ and 1.2 in accordance to the proposed values [108]. Likewise, the number of considered echoes Q is set to 3.

To summarize the model parameters used for simulation and optimization, we present them consolidated in Table 8.2.

Table 8.2.: Parameter values and ranges for the Kunisch-Pamp channel model used for simulation and optimization.

(a) Fixed parameters

Parameter	Value
Q	3
f	$4\,\mathrm{GHz}$
f_s	$500\,\mathrm{MHz}$
n	$2\,\mathrm{ns}$
G	$120\,\mathrm{dB}$
β	1.2

(b) Optimized parameters

Parameter	Value
K	Echos / virtual sources
d_0	0.5 to $1.5 d_{LOS}$
α	2.0 to 3.0
G_{MP}	$-25\,\mathrm{dB}$ to $-10\,\mathrm{dB}$
$G_{MP,LOS}$	$-15\,\mathrm{dB}$ to $0\,\mathrm{dB}$
γ	$9\,\mathrm{ns}$ to $13\,\mathrm{ns}$

8.3.3. Evaluation

As this kind of attacks requires location data of the respective sender and receiver terminals, only the data set *attack* and *robot* can be used for evaluation. Thus, we use the *attack* data set for a general feasibility study and the *robot* data set for verification.

[2] Available at https://github.com/hyperopt/hyperopt

Measurement sets:

1: I → II
2: I → III
3: I → IV
4: I → V
5: I → VI
6: I → VII
7: I → VIII
8: I → IX
9: I → X
10: I → XI
11: I → XII
12: II → XI
13: II → XII
14: IV → XI
15: IV → XII
16: VI → XI
17: VI → XII

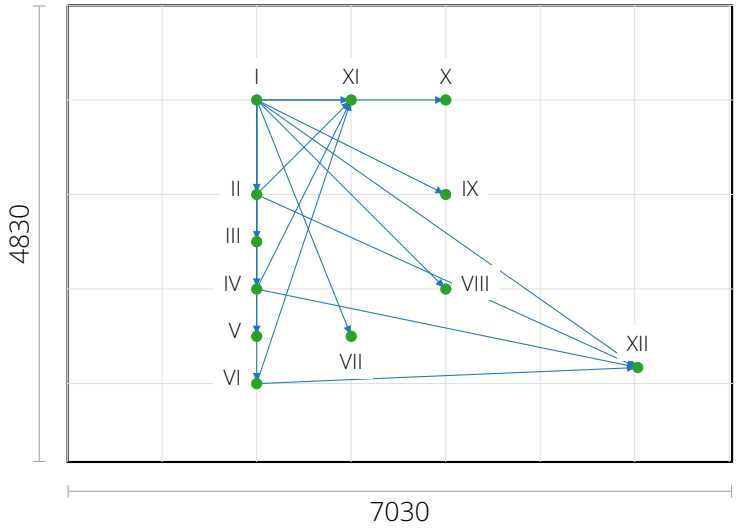


Figure 8.3.: Simplified room model, adjusted to adhere to the Kunisch-Pamp channel models restrictions.

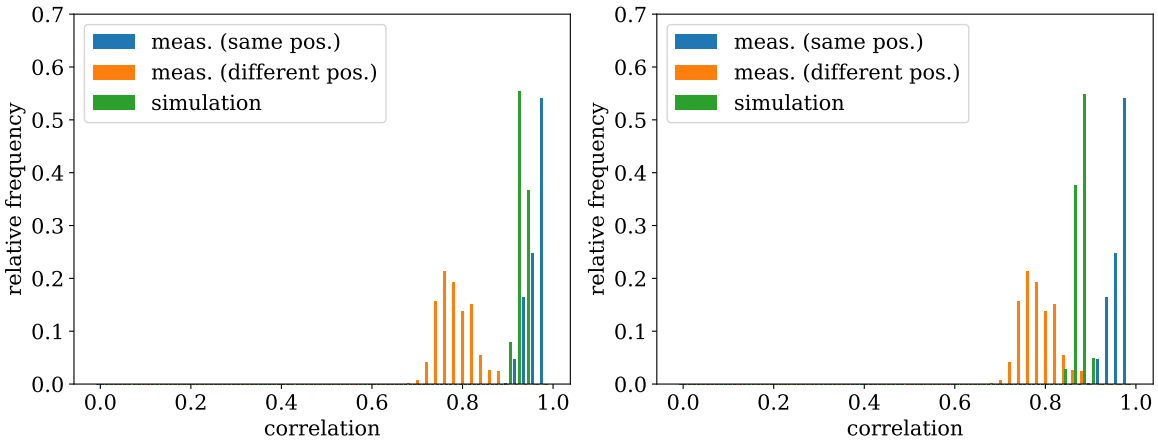
Data Set *attack*

To investigate the general feasibility of the attack, we use two different optimization strategies for this attack: First, a so-called *individual optimization*, where the parameters are adjusted for exactly one of the 17 measurement setups. Subsequently, the achieved cross-correlations are calculated again for the same measurement setup. This serves as basic investigation whether the Kunisch-Pamp model can simulate sufficiently accurate CIRs at all. Since this approach fits the whole channel model to one concrete measurement setup, we expect considerably higher cross correlations for the simulation than for the eavesdropped CIRs.

In a second step, the so-called *attacker-oriented optimization*, we select four measurement setups that are used to adapt the model. Here, the four selected measurements represent the previously known measurements of the attacker. The evaluation is then performed on the 13 remaining measurement setups. Concretely, the setups $R = \{11, 13, 15, 17\}$ are used for the attacker-oriented optimization — these are the setups involving the lower right position XII in the scheme (see Fig. 8.3). As this optimization approach is much more general, we expect a lower accuracy for the results. Hence, the respective achieved cross correlations are expected to be lower than for individual optimization.

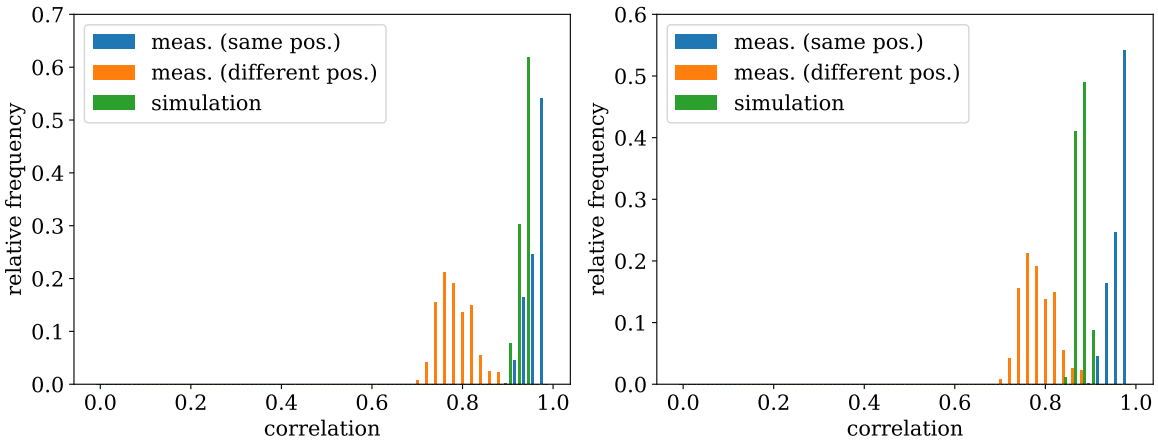
It should be mentioned that individual optimization does not provide robust information about the applicability of the attack, since it cannot be performed in practice by an adversary. This is only the case with attacker-oriented optimization.

For this data set, the following constraint must be considered: Since the channel model only supports rectangular spaces, the modelling of the space must be adjusted accordingly. Thus, instead of the real space shown in Fig. 4.7, a simplified model shown in Fig. 8.3 is used for the simulation.



(a) Best setup #14 (b) Worst setup #17

Figure 8.4.: Histogram of the achieved cross correlation between attacker simulation and measurement. Best and worst run using the individual optimization.



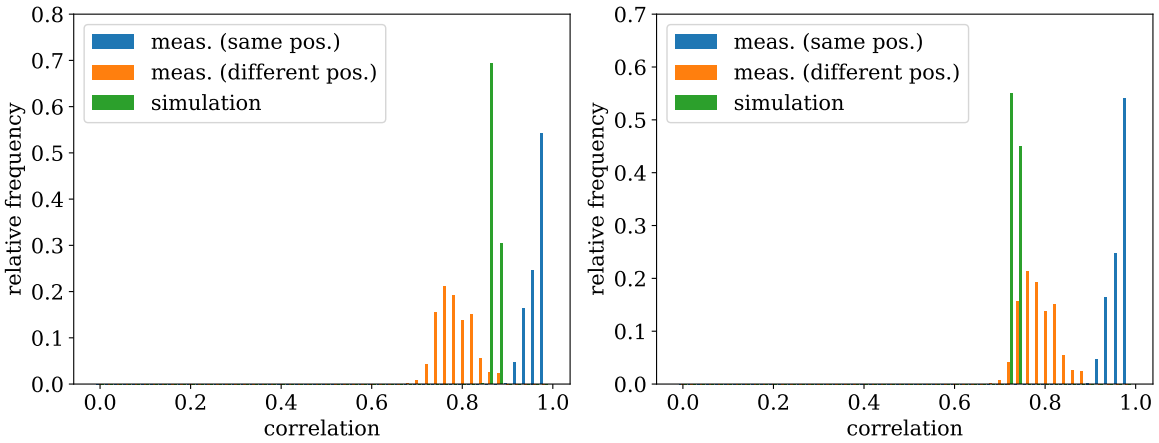
(a) Best setup #9 (b) Worst setup #17

Figure 8.5.: Histogram of the achieved cross correlation between attacker simulation and measurement. Best and worst run using the individual optimization with additional noise.

To visualize the general prediction performance of this attack we show histograms of the achieved cross correlation. For each optimization approach, we present the performance of the best and the worst measurement setup.

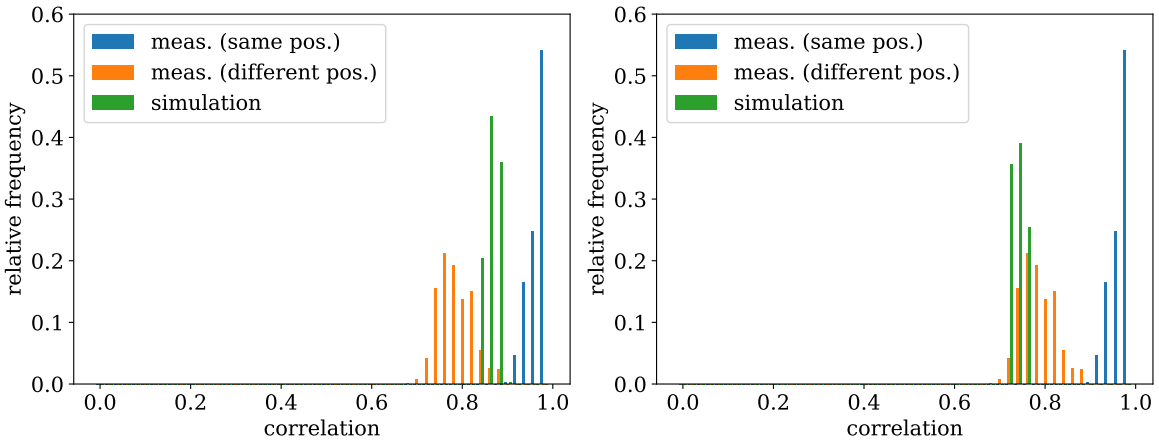
Each figure consists of three actually shown histograms: the first, "measurements (same pos.)" shown in blue, shows the cross correlation within the current measurement run. The second, "measurements (different pos.)" shown in orange, shows the cross correlation of the analysed measurement to *all* other measurements. And finally, "simulation" shown in green, depicts the cross correlation of the attacker simulation to the analysed measurement.

First, we show the performance based on the individual optimization. The respective best and worst results, from the attackers point of view, are shown in Fig. 8.4. The best attacker results are achieved for the setup #14, $IV \rightarrow XI$, with an average cross correlation 0.936. The worst results are recorded for setup #17, $VI \rightarrow XII$, with 0.883. In general the results



(a) Best setup $\#16$ (b) Worst setup $\#12$

Figure 8.6.: Histogram of the achieved cross correlation between attacker simulation and measurement. Best and worst run using the attacker-oriented optimization.



(a) Best setup $\#12$ (b) Worst setup $\#4$

Figure 8.7.: Histogram of the achieved cross correlation between attacker simulation and measurement. Best and worst run using the attacker-oriented optimization with additional noise.

here closer to the correlations of the legitimate CIRs then to the eavesdropped ones — the overall average cross correlation for the simulated CIRs are at 0.92.

By adding pseudo-random noise as described above, the results of the individual optimization could be slightly enhanced. In the best setup, here run $\#9$, $I \rightarrow X$, the attacker simulation achieves an average cross correlation of 0.941. The worst results are again obtained in setup $\#17$, $VI \rightarrow XII$ with 0.883 as well. Histograms for this optimization strategy are shown in Fig. 8.5.

The results of the parameter optimization, i.e., the parameter sets for each individual optimization, are listed in Appendix A.1. Table A.1 shows the parameters for the individual optimization without noise, Table A.2 with noise respectively.

After the individual optimization, we evaluated the **attacker-oriented optimization**. The achieved cross correlation with and without noise are shown in Fig. 8.6 and Fig. 8.7, respectively. As expected, the overall results of this optimization strategy are lower than for the

Table 8.3.: Optimized parameters for Kunisch-Pamp channel model using the attacker-oriented optimization.

(a) Without noise

Parameter	Value
K	$\{172, 123, 165, 173, 179, 221, 130, 164, 170, 174, 178, 180, 214, 270\}$
d_0	$5.45\,\mathrm{m}$
α	2.39
G_{MP}	$-10.24\,\mathrm{dB}$
$G_{MP,LOS}$	$-14.06\,\mathrm{dB}$
γ	$12.99\,\mathrm{ns}$

(b) With noise

Parameter	Value
K	$\{172, 171, 173, 179, 221, 116, 170, 178, 180, 214, 270\}$
d_0	$2.88\,\mathrm{m}$
α	2.14
G_{MP}	$-20.68\,\mathrm{dB}$
$G_{MP,LOS}$	$-2.23\,\mathrm{dB}$
γ	$12.83\,\mathrm{ns}$
V_N	$14.47\,\mathrm{dB}$

individual optimization. With or without noise, the best results of this approach achieve approximately the same cross correlation as the worst results of the individual optimization. The best setup without noise, setup $\#16$, $VI \rightarrow XI$ achieves 0.877. With noise, the best setup is $\#12$,$II \rightarrow XI$ with 0.874. The overall average of this optimization approach is 0.825 without and 0.836 with noise.

The final optimized model parameters for the attacker-oriented approach are listed in Table 8.3

A summary of the cross correlations of all 17 measurement runs is depicted in Fig. 8.8.

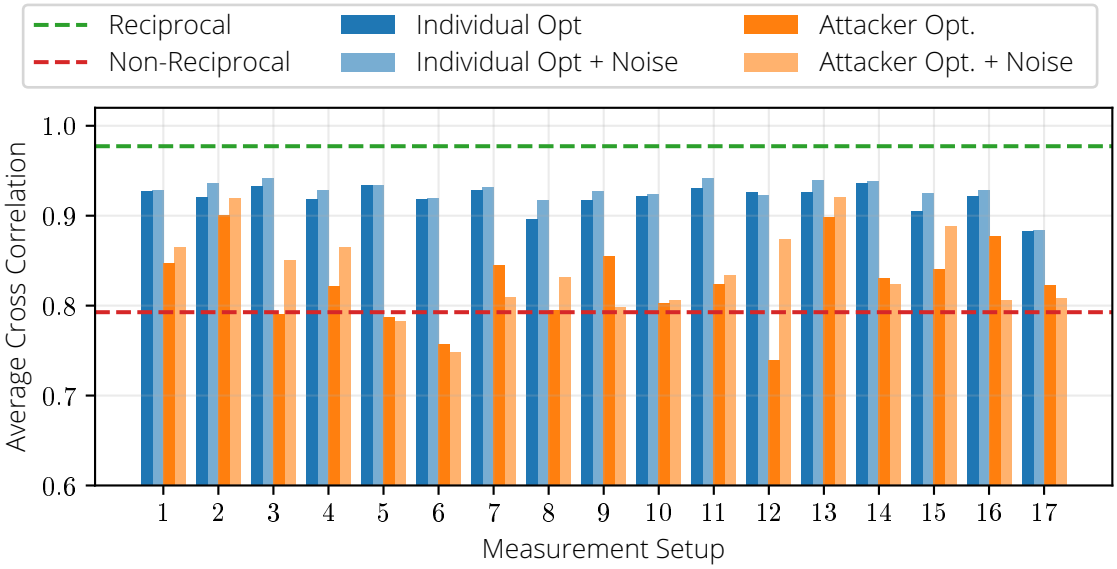


Figure 8.8.: Achieved average cross correlations for all 17 measurement setups of data set *attack*.

Overall these result show, that the deterministic channel model is capable of predicting major parts of the legitimate CIRs. This is demonstrated by the performances of the *individual optimization*. Hence, in settings where the adversary can collect samples of actual future transmission positions, this attack poses a non-negligible risk. Nevertheless, the re-

sults of the *attacker-oriented optimization* shows, that the model itself does not generalize very well. This means that the final channel model predicts CIRs at different positions in the transmission environment with only low precision.

Data Set *robot*

To verify the results obtained with the data set *attack*, we also applied this attack to the *robot* data set.

Since this data set is significantly more extensive, not all individual location combinations can be considered. Therefore, we apply an adapted strategy: In general, we only use the attacker-oriented adjustment. For this we use $10\,000$ random measurements, which are chosen from all measurements of one of the two eavesdroppers. These $10\,000$ measurements form the set R, which is used for the adaptation. Then, we use these optimized parameters to simulate 1000 randomly chosen measurements between Alice and Bob. Finally, the correlation between the simulation and the actual measurement is taken.

The resulting correlations of this evaluation are visualized in Fig. 8.9. The average cross correlation achieved with this approach is 0.835. It can be seen from the histogram that the attack does not provide any advantage over mere eavesdropping, i.e., the attack predictions do not yield higher correlations than the eavesdropped CIRs. The only difference to mere eavesdropping is a slightly narrower distribution. Nevertheless, this does not help the attacker in any particular way.

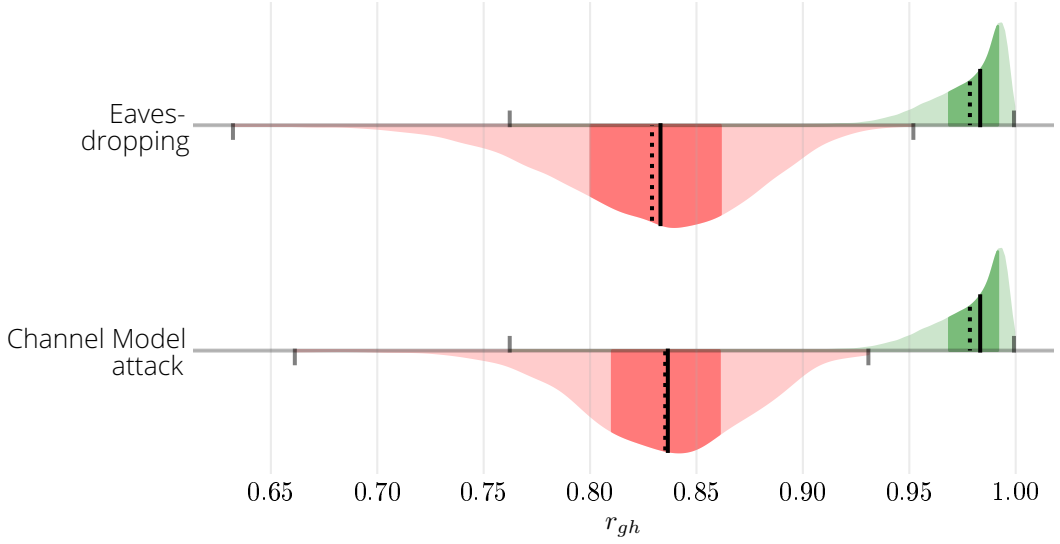


Figure 8.9.: Histogram of the achieved cross correlations for the data set *robot*

The final optimized model parameters for the attacker-oriented approach given the *robot* data set are listed in Table 8.4.

The results of this data set confirm the findings that the optimized channel model does not generalize very well. In this dynamic environment the predictions of this attack do not yield higher correlations than the CIRs obtained by eavesdropping.

Table 8.4.: Optimized parameters for Kunisch-Pamp channel model using the *robot* data set.

Parameter	Value
K	$\{172, 123, 165, 171, 173,$ $179, 122, 124, 130, 164,$ $166, 170, 178, 186\}$
d_0	$8.01\,\mathrm{m}$
α	2.33
G_{MP}	$-10.59\,\mathrm{dB}$
$G_{MP,LOS}$	$-6.29\,\mathrm{dB}$
γ	$9.30\,\mathrm{ns}$

8.4. Ray Tracing-based Attack

In the following we describe the ray tracing-based approach to the CIR pre-calculation attack. The basic idea of this attack is to reduce the complexity and error-proneness of the attack execution by choosing an approach that does not require substantial adaptation. We investigate to which extent this is achieved by applying ray-tracing concepts.

8.4.1. Concept

Ray tracing is an alternative approach to channel modelling to deterministically calculate location-specific parameters of a wireless transmission such as receive strengths or path delays. Thus, it is also an approach to deterministically predict CIRs depending on the transmission environment.

For this purpose, ray tracing considers the propagation of electromagnetic waves as quasi-optical, which means that a ray-optical approach can be adopted for the propagation calculation. This assumption is considered fulfilled as long as the reflecting objects are much larger than the wavelength of the transmission. Compared to the evaluation of a full field solution, this simplifies the calculations considerably and thus reduces the complexity.

It is already shown that this approach is very suitable for the calculation of CIRs and can achieve high prediction accuracies. In comparison with empirically recorded measurements, it could be shown that ray tracing approaches produce results with a very high matching. Additionally, the corresponding statistical and structural properties of the CIRs were reproduced very well [67]. This is also true for UWB transfers [181].

Since ray tracing assumes only the space geometry and terminal positions in addition to the basic transmission parameters, it provides a much simpler approach for the deterministic computation of CIRs compared to the Kunisch-Pamp channel model.

Using such an approach also changes the underlying concept of the attack built upon it. Since only the environment and the terminal position are required for the evaluation in addition to the center frequency and the bandwidth, the step of adapting the channel model to the actual environment can be omitted. This in turn means that no preparatory data acquisition is required. Thus, the preparatory steps of the preceding attack are completely unnecessary and can therefore be removed.

Nevertheless, the attacker must prepare the evaluation of the corresponding ray tracing tool. Thus, we have the following procedure for this attack:

Phase 1 - Tool preparation The attacker models the transmission environment in accordance to the tool's requirements.
Typically, this means, that a representation of the environment is generated, i.e., a model of the actual room. Considering that most indoor rooms are cuboids varying only in width, length and height, this step requires negligible effort. Increased effort is only expected if highly complex environments are to be modelled or if material properties are to be taken into account in addition to geometry.

Phase 2 - Actual attack The attacker predicts CIRs using the ray tracing tool and uses the prediction as input for his evaluation of the CRKG protocol. This step resembles *Phase 3* of the preceding attack.

Both steps can again be implemented in practice without major problems. Since no preparatory measurements are necessary here, this attack must even be considered easier to realize than the previous one.

8.4.2. Realization

For the implementation of this attack we employ the wireless propagation simulator *PyLayers*[3] [9]. Since this tool is completely open source, it is also fully available to a potential attacker.

PyLayers uses a graph-based approach to ray tracing [109] According to the authors, based on this approach, on the one hand a significantly faster execution can be achieved and on the other hand, environment-dependent transmission characteristics can be realized more effectively. Specifically for the application in the domain of localization, e.g., an improved calculation of the arrival times of multipath components is claimed [109]. Since this is also directly applicable to the use case of our attack, *PyLayers* is an appropriate candidate for implementation.

For the actual implementation, we modelled the concrete transfer environment for *PyLayers* . The corresponding transfers were then simulated in this environment. The relevant parameters for the simulation were the center frequency, the bandwidth and the terminal positions. With reference to our collected measurement data, the frequency is fixed at $3.9\,\mathrm{GHz}$ and the bandwidth at $500\,\mathrm{MHz}$. The positions are set in dependence of the concretely examined measurement.

Furthermore, the parameter *cutoff* influenced the evaluation significantly — it determines the depth to which the graph is explored. This corresponds to the number of permissible reflections. This parameter required a cost-benefit trade-off: for spaces that are not simple cuboids, such as the space for the data set *attack*, the runtime and memory requirements increase rapidly for higher cutoff values. The step from *cutoff* $= 4$ to 5 increased the average runtime from typically $0.66\,\mathrm{s}$ to $674.48\,\mathrm{s}$ and the memory consumption from $510.97\,\mathrm{MB}$ to $2231.26\,\mathrm{MB}$, while changing the cross correlation by only $<0.01\,\%$. Therefore we set the parameter *cutoff* fixed to 4.

In the course of our investigations, we found that the scaling of the time axis of the resulting CIR was partly non-deterministically distorted. As a result, we added the following post-processing step to the actual simulation:

The time axis was stretched or compressed by multiplying the originally simulated axis by a scaling factor s, in the range 0.5 to 9.5. In order to keep the post-processing as simple and easy as possible, the necessary adjustment of the simulated values based on the scaling was realized by linear interpolation. This correction of the time axis significantly improved the results of the attack.

Since the distortion of the time axis did not follow from any traceable cause, the concrete values of the scaling factor were found by exhaustive search within the half-open interval $[0.5, 9.5)$ with a step size of 0.1. The corresponding distribution of the resulting factors can be seen in Fig. 8.10.

For the attacker, this step does not represent a significant additional effort. Since the relevant interval includes only 90 values, an exhaustive search is feasible without any problems.

[3] Available at https://github.com/pylayers/pylayers

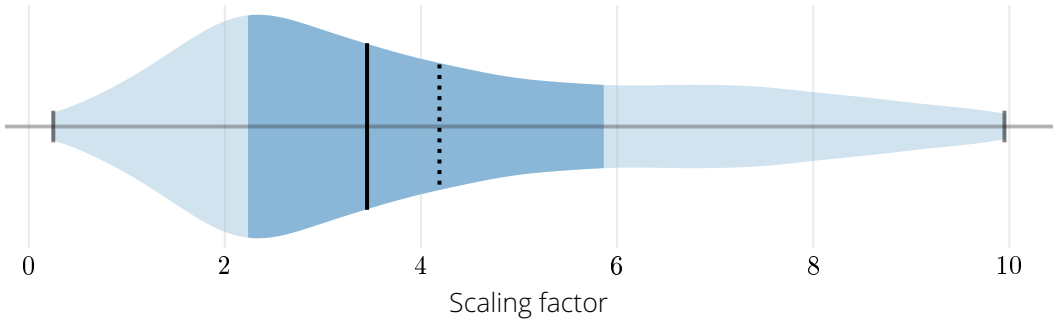


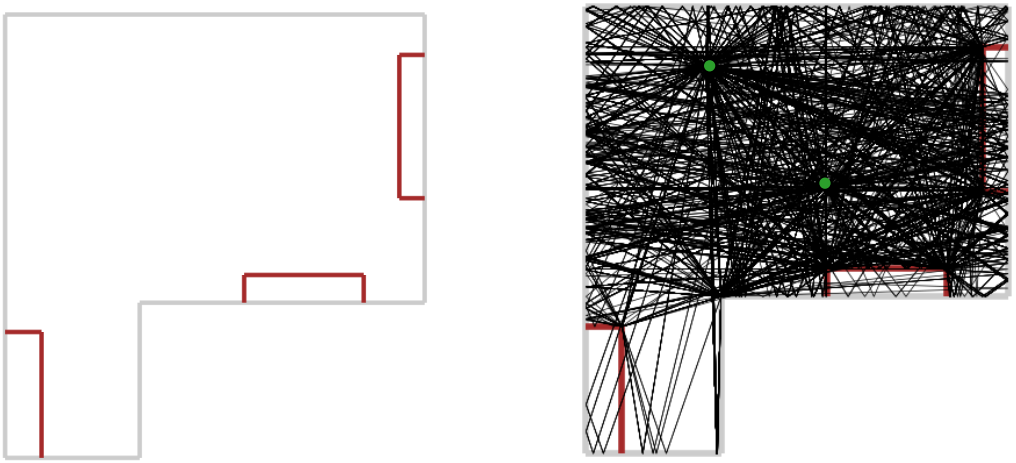
Figure 8.10.: Distribution of the scaling factor used.

8.4.3. Evaluation

Since the evaluation of this attack also requires position information of the terminals, we use the same data sets as in the previous attack, i.e., data sets *attack* and *robot*. The data sets *scenarios* and *longterm* do not contain location information and are therefore not applicable here.

Data Set *attack*

The modelling of the environment and a corresponding exemplary ray tracing simulation, concretely of attack setup $7 : I \rightarrow VIII$, are depicted in Fig. 8.11 In Fig. 8.11b, rays passing through the modelled furniture are results of the configured material, i.e., wood, and its predefined complex relative permeability.



(a) Model of the transmission environment.

(b) Simulated rays within the model.

Figure 8.11.: Environment model and ray tracing simulation example for data set *attack*.

For the *attack* data set, as there are only 17 different setups, we present the correlations achieved on average for all setups for a comprehensive presentation. In analogy to Fig. 8.8, we compare the results of the ray tracing attack to those of attacker oriented optimization of the channel model-based attack. As we are primarily interested in generally applicable attacks, we are omitting the results of the *individual optimization*. The corresponding results

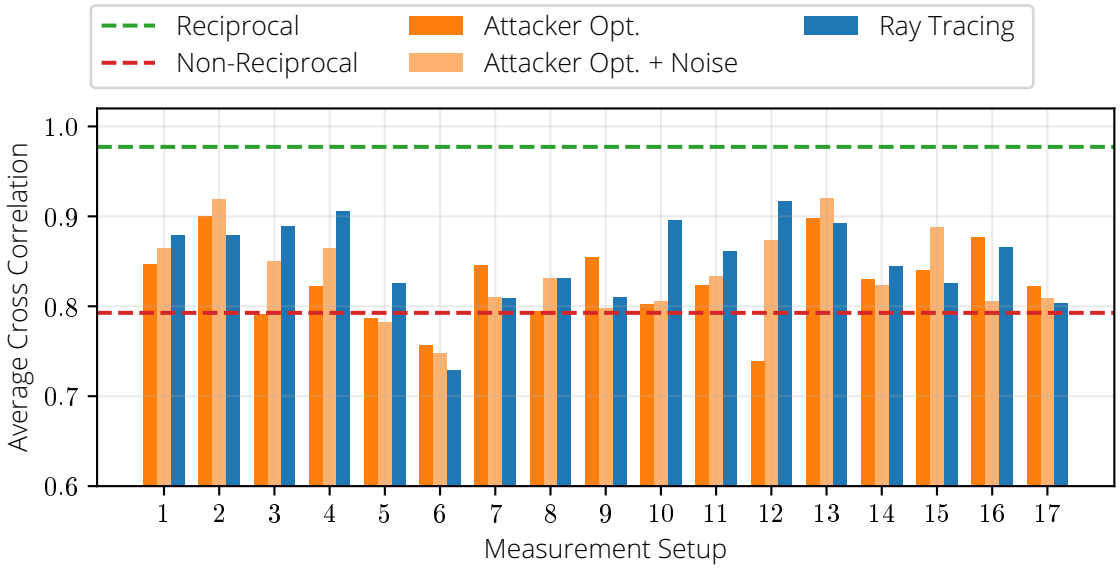


Figure 8.12.: Achieved average cross correlation of the ray tracing-based attack for data set *attack*, compared to the channel model-based attack.

related to the results of the channel model-based attack, can be seen in Fig. 8.12.

In 9 of the 17 setups the Ray Tracing-based attack achieved equal or better results than the Channel Model-based attack. For the remaining cases, the achieved results are only slightly behind. In total the average results are better than those of the general Channel Model-based attack, with a mean and standard deviation of 0.850 ± 0.046, compared to 0.825 ± 0.043.

In summary these results show, that the Ray Tracing-based attack is slightly more suitable for a general attack. Nevertheless, the overall performance in terms of generalization capabilities is still lacking.

Data Set *robot*

Since the *robot* data set is too large for a detailed evaluation of the single positions, we show the overall results here. The achieved correlations of this attack are presented in Fig. 8.13. To visualize the effect of the time scaling, we included the unscaled ray tracing results.

The raw results of the ray tracing clearly visualize the scaling issues of the simulation. With a mean of $\mu = 0.690$ and standard deviation of $\sigma = 0.078$ the raw ray tracing simulation perform significantly worse than plain eavesdropping, with $\mu = 0.827$ and $\sigma = 0.044$ respectively. By adding the described scaling factor, the correlations could be improved significantly. The scaled versions achieve cross correlations with $\mu = 0.848$ and $\sigma = 0.037$. Overall, the range of the scaled version is smaller with 0.741 to 0.940, compared to eavesdropping 0.652 to 0.949.

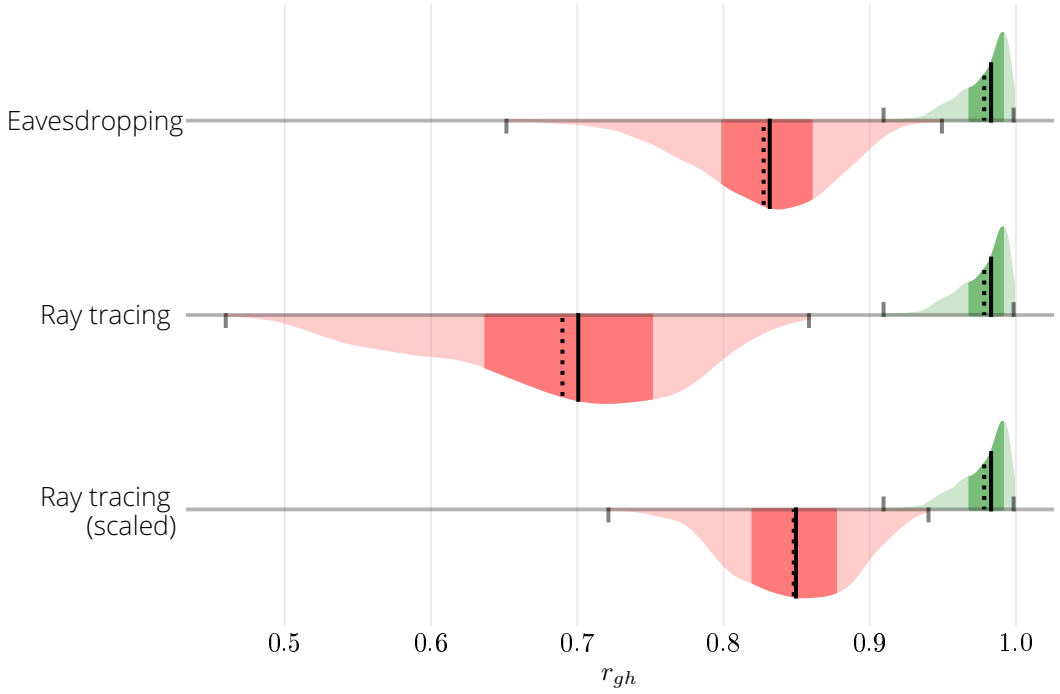


Figure 8.13.: Achieved cross correlations of the ray tracing-based attack for data set *robot*.

8.5. Discussion

The results of the two attacks presented yield the following insights regarding the security of CRKG.

The Channel Model based attack provides two findings:

First, deterministic channel modelling is very well capable of predicting actual channel impulse responses given the actual propagation environment and actual measurements. CIRs calculated in this way provide a significant advantage over plain eavesdropping on the communication. This is verified by the *individual optimization* approach of this attack.

And second, the channel model approach does not generalize well. Despite the high accuracy achieved through individual optimization, the general optimization approaches yield low cross correlations. This is already indicated by the attacker-oriented optimization of the *attack* data set and confirmed by the evaluation of the *robot* data set.

This means, the deterministic channel model attack, on the one hand, provides high prediction accuracy for concrete transmission situations, i.e., those where environment and the actual positions are used in the optimization, and on the other hand low prediction accuracy for generalized transmission situations, i.e., those where only the environment and different positions are used for the optimization.

For the adversary, this means that for a promising attack, she must either have reference measurements for the concretely attacked positions or he must improve the generalization of the optimization

In general, the Ray Tracing based attack confirmed these insights. In case of the *attack* data set, this attack on average achieved higher results than the one based on the Channel Model. Nevertheless, with the data set *robot*, which includes a lot more variations and different settings, the attack does not yield an advantage over mere eavesdropping. This verifies the missing generalization capabilities of this attack.

Hence, in the context of CRKG, this leads to the same conclusion as above: the attack can be successful in favourable settings, but is lacking with respect to generalization performance.

Looking at the larger context of PLS, these attacks are nonetheless a threat. Considering PLS primitives relying on static or slowly changing channel properties, these attacks are capable of predicting relevant features. For example, physical layer authentication methods require comparably stable identifiers, which are based on static or slowly changing environments. Here, the adaptation of the presented attacks to the actual positions is quite feasible.

Additionally, the attacker may be able to implement an iterative optimization procedure. Thus, the now separate steps of data acquisition, model optimization, and attack can be combined into one pipeline and executed continuously.

9. Machine Learning-assisted Attack

Parts of this Chapter are published in [194, 198].

In this chapter, we extend the attacker's capabilities to include the powerful approaches of machine learning. The basic goal of the attack is the same as in the previous chapter, with a slight twist on the attacker model and underlying assumptions.

Although the powerful capabilities of machine learning have already been used on the system design side, for example in the context of cognitive radio [25] or to design efficient wiretap codes [23], to the best of our knowledge they have not yet been used on the attacker side. Thus, we are the first to introduce the concept of machine learning assisted attacks on Physical Layer Security.

9.1. Adapted Attacker Model

Basically, we assume the same attacker model for attacks of this class as in the previous chapter — a passive eavesdropping attacker.

Additionally, the following two slight alterations to the model are imposed:

First, the attacker employs the capabilities of machine learning. Although, this is not directly a modification of the attacker model itself — the attacker per se is still computationally restrained — this significantly enhances the attacker capabilities.

Second, the application of machine learning approaches typically requires the acquisition of appropriate training data. Hence, the attacker models include access to such data. This assumption is comparable to the modification applied throughout the Kunisch-Pamp based attack: there, the attacker would also have access to CIRs acquired in the respective transmission environment.

The first adaption is easily justifiable — the relevant knowledge and tools are primarily public knowledge and open source. Hence, there is only a small hurdle to the practical use of these tools by attackers.

The second adaption is somewhat less clear, but here the same argument can be made as for the Kunisch-Pamp based attack. Specifically, this means that we consider it very realistic for an attacker to have access to the environment of the actual attack in order to record the corresponding measurements as training data. Since these measurements are completely independent in time from the actual attack, this only increases practical feasibility. Concretely, such recordings could be performed, for example, already during the construction phase of the corresponding buildings or at any time by service personnel [161, 165].

One notable difference to the attacks described before is the transition to a *model-free* attack. This attack does not require any knowledge or makes any assumptions about the actual physical properties of the attacked wireless communications. Specifically, no prior knowledge about the room, the terminal positions or the wireless properties like center frequencies, bandwidth and alike is required. Following the notation presented in [201] this changes the adversary model from *model-based* to *model-free*, as we no longer need a predefined physical model of environment and transmission, but instead learn them implicitly, based on prior observations of the adversary. Further, in difference to the state of the art, our attack does not require any optimization with respect to the terminal positions.

Summarizing, it is fair to say that this attacker model must be considered close to reality and relevant to practice as well.

9.2. Machine Learning-assisted Inference Attack

In this section we describe the proposed Machine Learning assisted Inference Attack. We start with the concept of and the reasoning behind this attack. Subsequently, we describe the actual realization of the attack. Then we evaluate this attack in comparison to the attacks described before and to the state-of-the-art. Finally, we put this attack into perspective regarding CRKG.

9.2.1. Concept and Rationale

The basic idea of this attack is rooted in the following three observations: *First*, as already shown in the preceding attacks, the CIRs recorded by the respective participants have an inherent structure, in the sense that the individual features follow a well-defined pattern [69, 163]. These are exactly the features used as key material in CRKG, for example. As demonstrated in Chapter 8, these features can be estimated with deterministic calculations to a certain extent, given sufficient information.

Second, the core assumption, that an eavesdropper who resides more than $\lambda/2$ away from the legitimate partners measures that lead to completely uncorrelated channels estimates, might not hold in practice. This assumption has already been questioned in recent work [185, 228], and in theory it is possible to reconstruct parts of the room geometry solely from overheard features [50]. Hence, we postulate that it may be possible to reconstruct parts of the key material only from features overheard in larger distances.

And finally, as the CRKG processing itself corrects certain differences between input data during *Information Reconciliation*, an attack does not need to predict the input data perfectly. For a successful attack it is sufficient to predict input values which do not have greater differences than the respective h_{AB}, h_{BA} pair. In this work, we hence want to investigate to which extent these three facts combined facilitate a practical inference attack.

Our attack aims to predict the CIR accurate enough to be within a distance that subsequently is corrected successfully by *Information Reconciliation*. To achieve this, we propose a machine learning model, which implicitly performs geometry reconstruction as well as the subsequent prediction of legitimate channel characteristics. It allows an attacker to predict the input values for the CRKG processing, which in turn enables her to derive the supposedly secret key of Alice and Bob.

The core concept is equal to those of the Channel Model based attack, presented in Section 8.3: the attacker collects samples in the room where the attack will be performed, uses this samples to adapt, here train, her local model, and finally employs the resulting model in the attack itself. This approach comes with the same three phases as the Channel Model based attack:

Phase 1 - Data acquisition The adversary collects CIR samples within the environment where legitimate communication will occur.
Again, these samples are used to adapt the selected model to this actual environment. More specifically, in terms of the machine learning approach, these samples are used at train the selected ML model. Different to the preceding attack, the adversary needs to collect samples of h_{AB} and h_{BA} as well as the corresponding h_{AE} and h_{BE}. Since the attack directly uses h_{AE} and h_{BE} to infer h_{AB}/h_{BA}, all four observations need to be acquired.

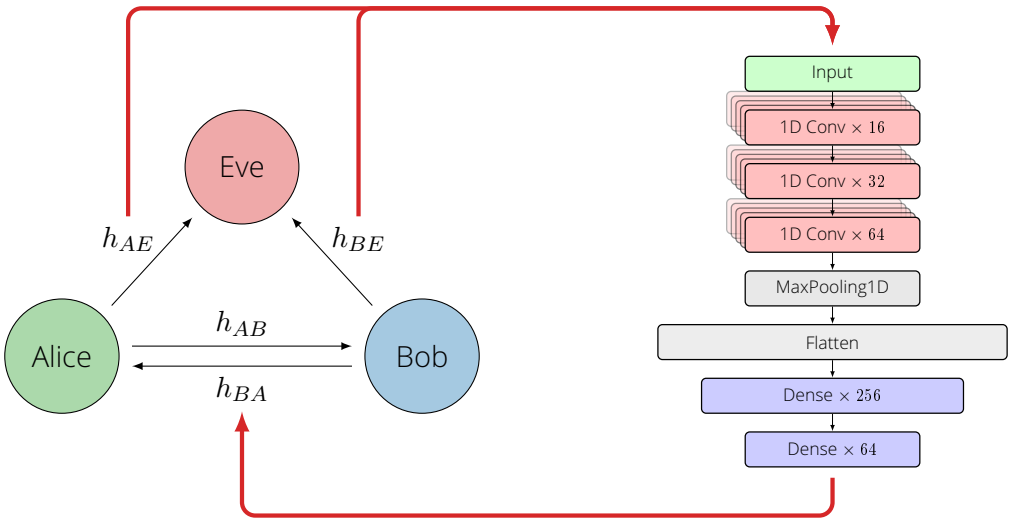


Figure 9.1.: Core idea of the ML based inference attack: use a trained model to directly infer the legitimate CIR from the overheard ones.

Phase 2 - Model training The acquired samples are used to train the chosen machine learning primitive.

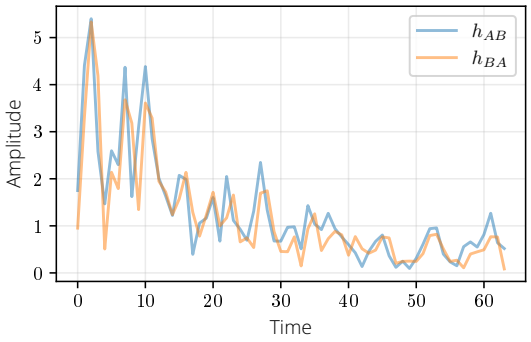
Phase 3 - Actual attack The trained model is used on live captured CIRs to infer the respective legitimate CIR.
During the attack itself, the adversary is completely passive and is only listening to the legitimate communication between Alice and Bob. The attacker locally processes the CIRs overheard with the trained model and infers the CIR between Alice and Bob. Since all messages of the subsequent CRKG processing are sent in clear text, the attacker can subsequently use the inferred CIR to derive the same key as the legitimate partners.

The core idea of the attack is visualized in Fig. 9.1. With a trained model the attacker tries to directly infer the legitimate input of the key derivation, i.e., h_{AB}/h_{BA}. All steps of the attack are carried out with COTS hardware — thus, no special capabilities are needed for the adversary.

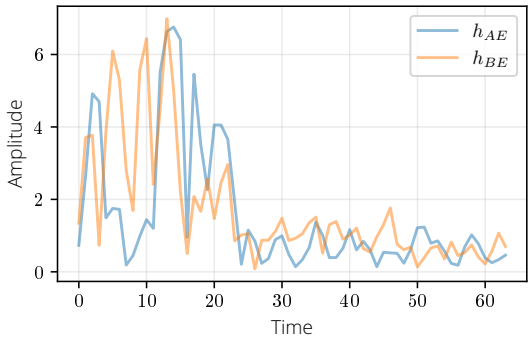
9.2.2. Realization

In the following we describe how we implemented the presented attack. First, the data used and the respective preprocessing are presented. Second, we describe the core architecture of our machine learning approach. And finally, we show how these parts are combined to successfully mount the corresponding attack.

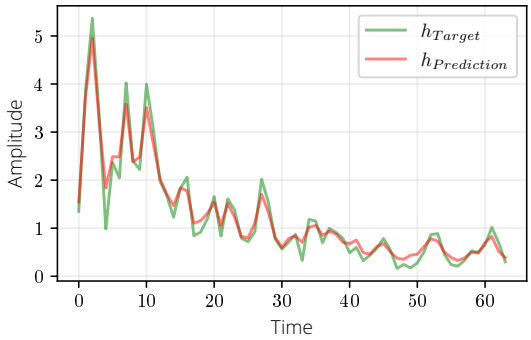
Data With exemption of data set *attack*, we are using all data sets available for the realization of this attack approach. Since the training of a machine learning model requires a comprehensive representative set of samples, this excludes the use of the data set *attack*, which contains only 17 different measurement points. The other three data sets were used in this attack.



(a) Reciprocal measurements



(b) Measurements at Eves terminal



(c) Target values and CNN generated prediction

Figure 9.2.: Exemplary CIR realizations at different nodes as well as the derived target values and the values predicted by the attacking CNN.

To prepare the data for the processing, the following preliminary steps were conducted for all measurements: we synchronized the CIR pairs by using the maximum of the cross correlation, in accordance to the argumentation in Section 2.5 and Section 5.3. This is valid, because during training the attacker has all samples locally at hand, so no blind synchronization needs to be applied.

Subsequently, we purge the parts containing only noise to extract those that bear information of the CIR, based on the approach described in Section 7.2. As last preparatory step, the CIRs were scaled by the standard deviation of the measurements.

Fig. 9.2a depicts the acquired measurements h_{AB} and h_{BA} after this processing. In Fig. 9.2b the corresponding measurements at Eve, h_{AE} and h_{BE}, are shown. The measurements of the legitimate, reciprocal channel h_{AB} and h_{BA} are called *reciprocal*, whereas Eves CIRs (overheard measurements and predictions) are called *non-reciprocal*, since they are not part of the reciprocal channel.

Finally, we defined the mean of h_{AB} and h_{BA} as target for the model training process. This decision is based in the CRKG processing itself: one of the main steps is *Information Reconciliation*, which removes slight differences between h_{AB} and h_{BA}, as those are not perfectly equal either. Hence, for an attack to succeed, it suffices to solely have the same amount or fewer differences as Alice and Bob.

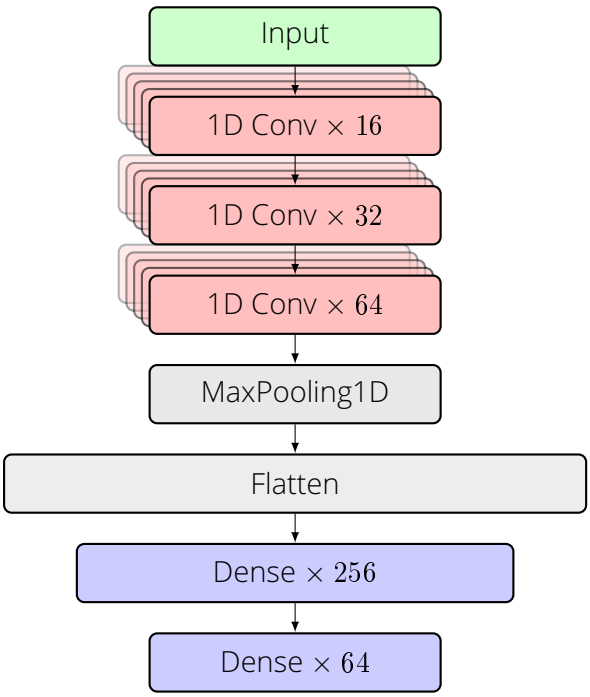


Figure 9.3.: The core architecture of the *Convolutional Neural Network* used in the inference attack.

Architecture The main aim is to reconstruct the channel characteristics of the legitimate channel from the features observed by Eve. Literature indicates that CNNs excel at this task [111, 235]. We hence chose them when designing our core architecture. Since our input data is one dimensional, we use 1D convolutional layers (Conv).

The architecture of the final network used in the attack is depicted in Fig. 9.3. It is a straight forward CNN realization consisting of three 1DConv layers with $16, 32, 64$ feature maps respectively, followed by a 1D MaxPooling layer. Afterwards a Flatten layer reshapes the data for the subsequent processing done by two fully connected layers (Dense). The intermediate Dense layer has size 256 and the final layer has the size of the targeted output, i.e., 64 elements. After each 1DConv and before each Dense layer, a Dropout layer with rate 0.5 is applied, to avoid overfitting and to provide generalized learning results. All 1DConv and Dense Layers have Rectified Linear Activation functions.

Although this architecture might seem simple, it is sufficient to extract relevant features from the CIRs as the results show. Note that we also tested more complex architectures like VGG, DenseNet or InceptionNet, but despite their significantly higher complexity, none of those achieved better results than the architecture presented here.

Training To train the network, we split the respective data set by 0.7 and used the $70\ \%$ for training and the remaining $30\ \%$ for evaluation. In the case of the data set *scenarios*, this division was made separately for each scenario. Since we intend to fit real valued output data, we used the *Mean Squared Error* as loss function and *Mean Absolute Error* as accuracy metric. Dozat suggested training the network using the *ADAM* optimizer with Nesterov momentum for such tasks, as it can achieve the best performance [51].

Fig. 9.2c shows an example of the prediction performances of the trained network.

9.2.3. Evaluation

To evaluate our attack, we first assess the accuracy of the attack predictions. Subsequently, we compare the results with those of similar attacks, i.e., the two attacks presented in Section 8.3 and Section 8.4 as well as the work of Ben Hamida et al. [20]. For both evaluations we employ the metric of *normalized cross correlation*, as introduced in Section 2.5, to ease comparison.

Prediction accuracy To assess the accuracy of the CNNs prediction, we first show the measurement metrics as split violin plots. As the data sets *scenarios* and *robot* are very diverse, we expect a spread out distribution of the resulting cross correlations. In contrast, the *longterm* data is more stable, hence, the histograms are expected to be more consolidated.

In the plots of Fig. 9.4, the upper split violin plot show the results for plain eavesdropping, i.e., the cross correlation achieved by the attackers overheard CIRs. The respective lower violin plot titled "Inference attack" presents the results of this attack. These cross correlations are between the legitimate CIRs and the output of the trained CNN. Due to the training towards the mean of h_{AB} and h_{BA}, we expect to see a distinctively higher cross correlation than for the predicted CIRs.

The upper plot, Fig. 9.4a, shows the results for the data set *scenarios*: for the reciprocal measurements the mean cross correlation is 0.980, for the non-reciprocal ones 0.807. The fact that the non-reciprocal distribution appears to be multi-modal apparently results from the different measurement scenarios: some scenarios include much interference, which translates to significantly worse results for the non-reciprocal measurements.

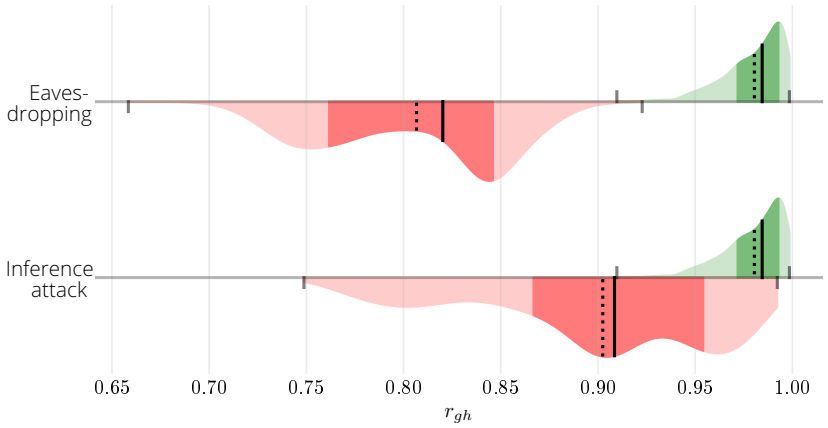
For the inference attack, the mean cross correlation of the predicted CIRs lies at 0.902. The CIRs predicted by the attack are closer to the reciprocal than to the non-reciprocal ones, which means that the inferred CIRs substantially match the legitimate ones. Again, the histogram appears to be multi-modal due to the differing performance in the varying scenarios.

The middle plot, Fig. 9.4b, presents the results for the data set *robot*. In this setting the reciprocal measurements cross correlations have an average of 0.912, whereas the non-reciprocal ones achieve 0.823. The distribution of the cross correlations do not appear multimodal as in the *scenarios* results, which indicates that the different settings of this data set achieve similar result distributions.

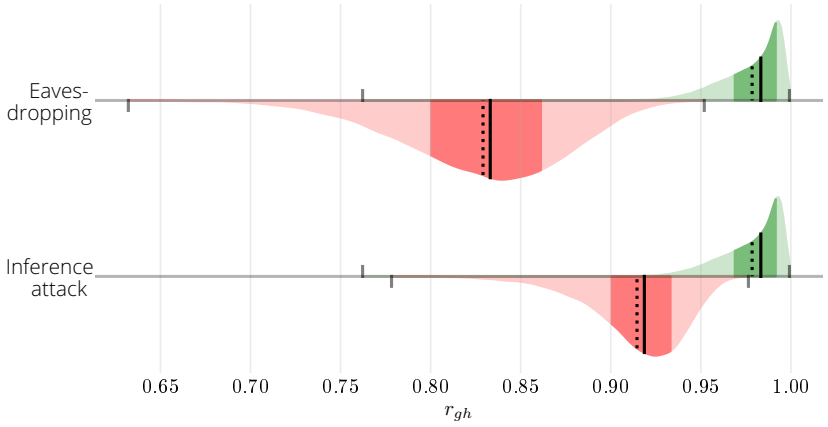
In this setting, the presented inference attack could again increase the correlations for the attacker: the predicted CIRs achieve cross correlations of 0.915. Additionally, the distribution is much more consolidated, with a standard deviation of 0.026, nearly half of the eavesdropped one, 0.050. It is also apparent that the distribution of the attack CIRs clearly overlaps with that of the legitimate communication partners. This means that in this data set, too, the attacker can infer CIRs whose correlation with the legitimate ones is comparable or even better than that of the reciprocal measurements of Alice and Bob.

The plot Fig. 9.4c depicts the results for the *long-term* data. The cross correlations are more stable and higher, the non-reciprocal mean is at 0.871, the reciprocal one at 0.990, as expected.

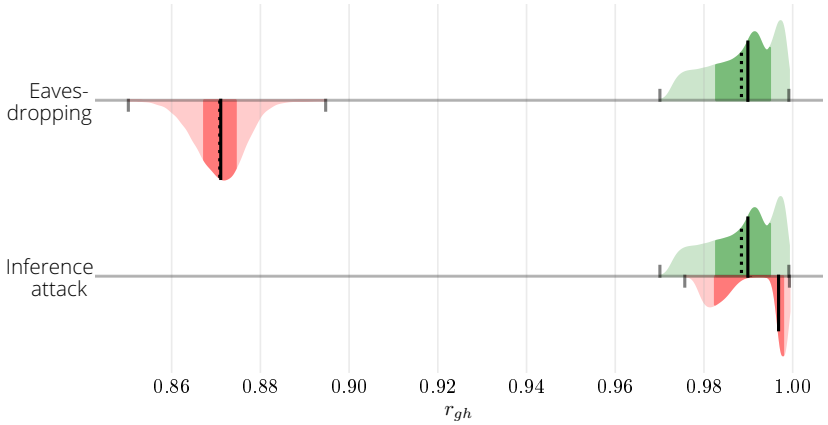
For these measurements, the average attacker correlation is 0.997 — so in fact *higher* than the average correlation of the legitimate CIRs, which is at 0.990. This is possible, as the



(a) Data set *scenarios*



(b) Data set *robot*



(c) Data set *longterm*

Figure 9.4.: Achieved normalized cross correlation between legitimate terminals, eavesdroppers, and predictions for current attack.
Mind the different scales of the X axis.

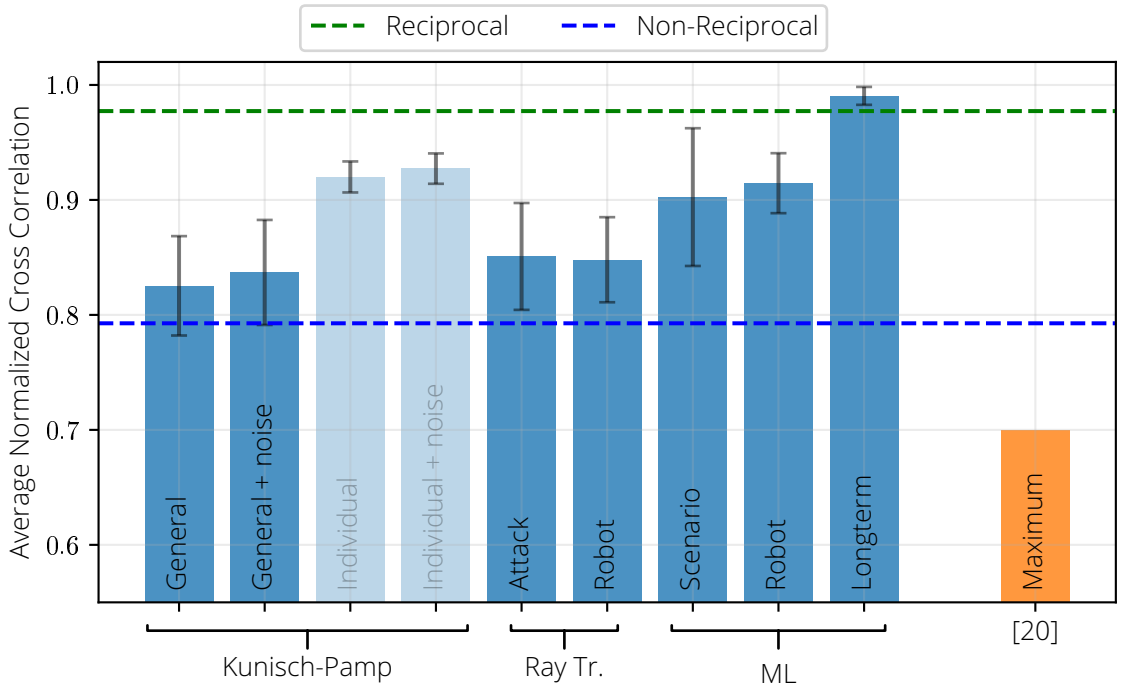


Figure 9.5.: Comparison of the current attack with the state-of-the-art and the presented attacks.

CNN learns to predict the mean between h_{AB} and h_{BA}. Hence, a very precise prediction is closer to h_{AB} than h_{BA}. Consequently, this result means that the attacker in this setting can predict the CIRs on average so well that he has better knowledge of the reciprocal channel than the legitimate participants — and thus also of the implicit shared secret of them.

The bimodal shape of the attacker results are presumably also results of the training target: apparently, the CNN predicts h_{AB} slightly better than h_{BA}, or vice versa. Hence, during evaluation, the bimodal distribution is generated — one "peak" for h_{AB} and h_{BA} each.

Overall, these results clearly demonstrated the high prediction accuracy of the trained attack CNN. This accuracy allows the attacker to significantly expand his information about the legitimate channel up to concrete knowledge about the channel characteristics.

Comparison to "classical" attacks In Fig. 9.5 we compare the prediction performance of the ML inference attack to those of the baseline attack [20] as well as to the Channel Model and Ray Tracing based attacks.

Compared to [20], our presented attack provides substantially better results for the attacker. Since only the maximum can be reliably extracted from their paper, we compare here our average value with their maximum. The attack presented here provides better average cross correlations, with 0.902 to 0.997, than the maximum of ≈ 0.7 in [20], despite this unfavourable comparison.

The Channel Model based attacks need to be differentiated regarding their optimization approach: Only results originating from the "general" or attacker-oriented optimization are comparable to the current results, as only those are generalized to be applicable to all measurements in this room. With this optimization the previous work achieves average

cross correlations of 0.825 and 0.837. Again, the attack presented here outperforms the previous work with a cross correlation of 0.902 to 0.997.

The individual optimizations are each specifically adapted to a concrete terminal position and room geometry. Hence, higher correlations are achieved here, with 0.920 and 0.927. However, since this specific optimization cannot be applied to other measurements but to the concrete one, these results are not comparable to the results of our generally applicable attack.

Nevertheless, in the context of the presented attack, a fair analogy to "Individual Optimization" would be an evaluation regarding the training data, since these are also specialized and not generally applicable — here our attack would reach correlations of 0.975 to 0.999, thus again outperforming the Channel Model attack. However, as we are only interested in generally applicable attacks and as evaluating the training data is no meaningful analysis, these results are shown more transparent in Fig. 9.5 to indicate that these are not generally applicable. Furthermore, our attack, even in the generic version, achieves significantly better cross correlation with 0.997 for the *long-term* measurements.

Finally, it should be mentioned that Fig. 9.5 shows the achieved *average* cross correlations. The single approaches might still yield CIR predictions accurate enough to corrupt the subsequent key exchange. This is further investigated in the upcoming discussion.

9.3. Discussion

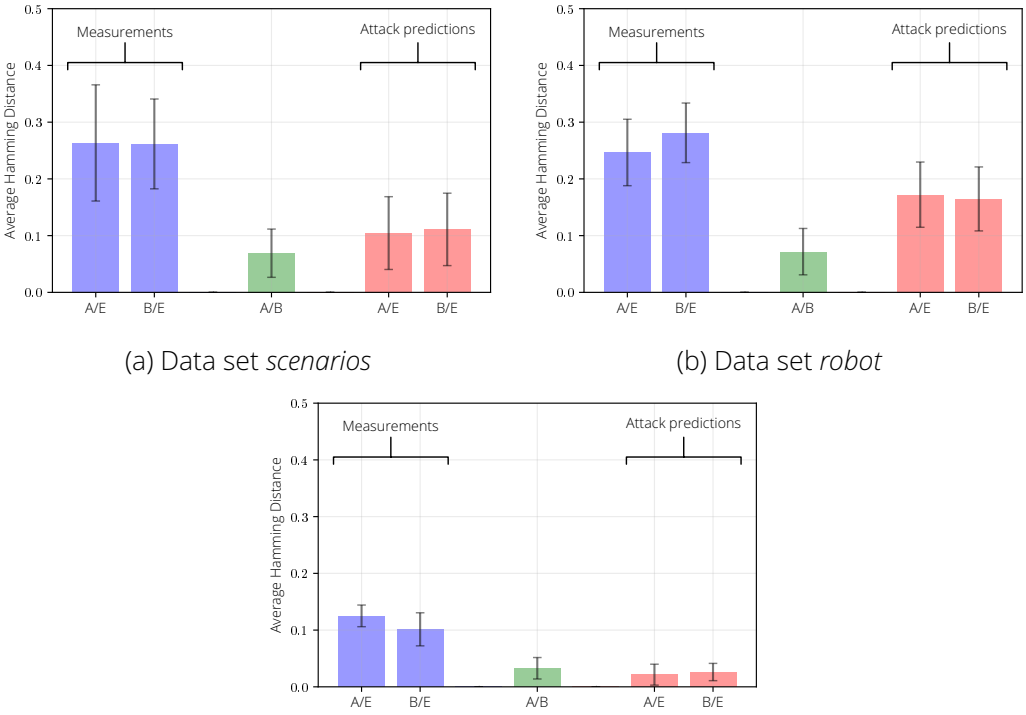
In general, the ML assisted attack increases the attacker's chances of success once again significantly compared to the previously presented attack as well as to comparable attacks.

Considering that even the straight forward architecture applied in our attacks can achieve such significant results, this class of attacks pose a considerable risk to PLS primitives relying on CIRs as input material. To further assess the implications of this prediction performance, we further evaluate the inferred CIRs in the context of CRKG.

Security of keying material The processing subsequent to the channel measurement is currently not standardized and many different solutions for the key generation exist. Hence, a definite analysis of the attack's impact on the key exchange is difficult. Nevertheless, to still show the possible implications of this attack, we implement a threshold based quantization scheme, as used in, e.g., [12, 96, 133], and analyse the resulting Hamming distances between Alice/Eve and Bob/Eve. Such threshold based quantization approaches were also used as baseline in previous quantization related work, e.g., [229, 75]. Note, that Alice and Bob would aim to achieve very low Hamming distance between each other, and as high Hamming distances as possible to the sequence Eve infers. Since the resulting bit vectors are of equal length, the average Hamming distance equals the BDR.

As we trained our CNN to predict the mean of h_{AB} and h_{BA}, the resulting quantized bit vector of the predicted CIRs are expected to have significantly lower Hamming distances than quantized h_{AB}/h_{AE} or h_{AB}/h_{BE}.

Figure 9.6a presents the Hamming distances for the observations obtained by Eve, Alice and Bob in the *scenarios* data set as well as for the predictions generated by the presented attack. The reference is the green bar in the middle showing the average Hamming distance of 0.069 for the quantized CIRs of Alice and Bob. The two blue bars show the average Hamming distance between the overheard CIR measurements at Eve and Alice's/Bob's CIR



(a) Data set *scenarios*

(b) Data set *robot*

(c) Data set *longterm*

Figure 9.6.: Average Hamming Distances for overheard CIRs (blue bars), legitimate CIRs (green bar) and the values predicted by the attack (red bars).

after quantization, with 0.263 and 0.261, respectively. On the right, the red bars show the Hamming distance between the attack predictions and Alice's/Bob's quantized CIRs, achieving 0.105 and 0.110, respectively. It is visible, that the attack generates values well within the standard deviation of the legitimate channel, i.e., binary sequences which should be successfully corrected during *Information Reconciliation*. Further, the attack yielded a perfect match, i.e., a Hamming distance of 0, in 1.2% of all cases and Hamming distances below those of Alice and Bob in 39.1% of all cases.

In combination, this means that an adversary carrying out this attack can derive the same key bits as the legitimate communication partners in at least 39.1% of all cases.

Figure 9.6b shows the Hamming distances for the quantized inferred CIRs of the data set *robot*. In this setting, the ML based attack could not achieve such results as with the other data sets. Nevertheless, the attackers Hamming distances could still be improved: the eavesdropped CIRs achieve average Hamming distances of 0.247 and 0.281, for Alice/Eve and Bob/Eve, respectively. This is opposed by the results of the inference attack with 0.172 and 0.165. Although these results are not as clearly within the standard deviation of the legitimate partners as in the previous results, the attacker can still use these results to significantly corrupt the key exchange: in 14.6% of all cases the achieved Hamming distance is lower than Alice's/Bob's, meaning that the attacker can derive the same key as them.

Finally, Fig. 9.6c depicts the same values as measured for the *longterm* data set. Since these measurements, unlike the previous ones, do not contain highly dynamic interference, all distances are considerably lower. The green Alice/Bob reference is 0.032; the respective results for Alice/Eve and Bob/Eve are 0.125 and 0.101. In this setting, the CNN attack achieved significantly better results: with average Hamming distances of 0.021 and 0.026,

the predictions achieved even better results, than the legitimate Alice/Bob CIR pairs. This is possible, because we trained the network to predict the mean of Alice and Bobs measurements. Hence, if the attack prediction is accurate enough, the resulting Hamming distance at Eve can even be lower than those of Alice and Bob. Further, the attack yields Hamming distances of 0 in 33.5% of all cases and distances below the legitimate one in 83.5% of all cases.

Again, this means that the attack can derive the correct key bits in 83.5% of all cases.

In conclusion, these results show that a significant proportion of the key material can be successfully inferred, using the proposed attack. Within all data sets, the CIRs predicted by the attack show high correlation to the legitimate CIRs and in many cases even exceed the correlation of the reciprocal measurements, resulting in a significant proportion of the keying material being compromised.

Using RSSI instead of CIR-based schemes cannot improve security, as indicated above. To the contrary, since RSSI can directly be derived from CIR with loss of information, RSSI-based systems will be much more vulnerable to our attack.

Considering that other PLS primitives have different input data requirements, the results of the *long-term* measurement reveal a further problem: since these measurements were recorded with comparatively little dynamic range in the channel, they would also be suitable for PLS primitives requiring more stable channel characteristics, such as authentication. The obtained results of the presented attack imply that the underlying assumption of uncorrelated observations is not unconditionally acceptable. Hence, similar PLS primitives relying on the same core assumption may be affected in the same way as the presented CRKG.

Part V.

Attack Resistant Solutions

10. Deterministic Preprocessing

Parts of this Chapter are published in [200].

Despite the considerable success of the attacks, we still believe that CRKG can be secured successfully. This is rooted in the fact that the attacks succeed because they predict parts of the CIR that are deterministically driven by the environment. If these predictable parts are removed from the raw input data, this would inherently thwart the attacks.

In this chapter, we design and implement a generic preprocessing step, called Channel Characteristics Estimation (CCE), that has two main goals: The most important goal is to reduce or even remove the weaknesses that enable the attacks presented in the previous part. For this purpose, we concretely consider the special properties of the common randomness sources realizations, which are the main reason for the success of the attacks, for our solution design.

As an additional goal, the preprocessing should handle the special properties of CIRs in such a way that, on the one hand, the contained entropy is preserved as well as possible and made available to the key derivation, and, on the other hand, an effective quantization is enabled, taking into account the special properties of CIRs.

Both the resilience to the presented attacks and the consideration of the special CIR characteristics have not yet been investigated in the literature.

10.1. Channel Characteristics Estimation

We describe CCE by first introducing the main reasoning and design decisions. Then follows the description of the actual realization and how the respective parameters are selected. Finally, we evaluate the approach using our data sets.

10.1.1. Concept

First of all, the approach obeys to the general design principal detailed in Chapter 7. This means, it operates in a *blind* modus, i.e., no message exchange between the legitimate parties, and it runs *online*, i.e., without buffering of measurements or alike, to avoid information leakage and to increase efficiency.

The core idea of CCE is based on the following two observations: first, and as seen in the previous attacks, the main properties of CIRs are determined by the physical environment and the resulting reflections, and second, these properties change comparatively slowly due to the natural movement of the terminals in space.

By combining these two observations, one can conclude that the measured CIR consists of a "static" part and a "dynamic" part. Here, the static part results from reflections by the environment. Due to the slow, continuous movement of the terminal this static part changes slowly as well. Additionally, since the static part is deterministically generated based on the environment and the terminal position, we conjecture that this part does not carry entropy usable for key generation. In terms of wireless communication systems, this static part can be interpreted as slow fading process. Hence, between successive measurements, this static part will only change little.

The dynamic part, on the other hand, consists of quickly changing channel properties, i.e., is fast fading. These are changes in the channel happening quicker than the respective sampling interval. Thereby, these changes alter the respective properties in between two successive measurements. Hence, they are denoted as non-static or dynamic.

The CCE approach now intends to approximate the static part of the current CIR and then remove it before the actual processing.

With the removal of the static part the following goals will be achieved:

- Increase attack resilience of the respective processing.
  Since the presented attacks are successful due to their prediction capabilities regard the general CIR shape, the removal of such predictable features will inherently increase the resilience against such attacks.

- Increase security and quality of key material. Since the static parts of the CIR are predetermined by the environment, they are by definition low entropy. The CRKG processing itself cannot create entropy. Hence, a low entropy input material will lead to low entropy keys, i.e., keys with bad randomness or respectively predictable keys. Thereby, the removal of predictable features itself increases the entropy of the respective key material.

- Enabling established processing approaches.
  This mainly addresses the fact, that common approaches, e.g., regarding quantization, assume the input material of CRKG as stationary process, at least regarding its mean. Given the existence of the typical CIR signal progression, this cannot be assumed for raw CIR measurement. Concretely, the values at the "beginning" of a CIR will always be higher than those at the "end". The removal of the predictable features also removes the typical progression of CIRs and generates a stationary process with mean 0

The actual characteristics estimation is conducted by computing the ensemble average over the latest recorded CIRs. This process is described in the next section.

10.1.2. Realization and Parameter Selection

The realization of CCE builds from the fact that the static part of the recorded CIRs is only changing slowly. This allows for the approximation of this part by building an element-wise ensemble average of the CIRs over a given time window.

Concretely, we proceed as follows. First, we define a system parameter w, which denotes the size of the time window. The actual value of w is discussed below. Using this size, we instantiate a ring buffer intended to hold the last w CIR observations. During the operation and upon acquisition of a CIR realization, the following steps are performed: At time step i, the CIR realization h_i is acquired. With respect to Eq. (2.32), this CIR is a vector of scalars.

$$h_i = \{v_i^0, v_i^1, v_i^2, \ldots, v_i^n\} \quad v_i^j \in \mathbb{R}, \tag{10.1}$$

Here, the subscript $_i$ describes the acquisition time, the superscript ^{j} denotes the time index within a single CIR. This current realization h_i is pushed to the ring buffer, which also removes the entry h_{i-w}. Hence, the buffer now holds the last w CIR realizations, ranging from time i to $i-(w-1)$. Now, for each element v^j within the CIRs the arithmetic mean $\bar{v}^j$ over the ensemble in the buffer is computed.

$$\bar{v}^j = \frac{1}{w} \sum_{l=0}^{w-1} v_{i-l}^j \tag{10.2}$$

This results in a vector $\bar{h}$ consisting of the respective elements $\bar{v}^j$.

$$\bar{h} = \{\bar{v}^0, \bar{v}^1, \bar{v}^2, \ldots, \bar{v}^n\} \tag{10.3}$$

For a given time i, the respective vector $\bar{h}_i$ represents the approximation of the static part, or Channel Characteristics Estimation, at that time.

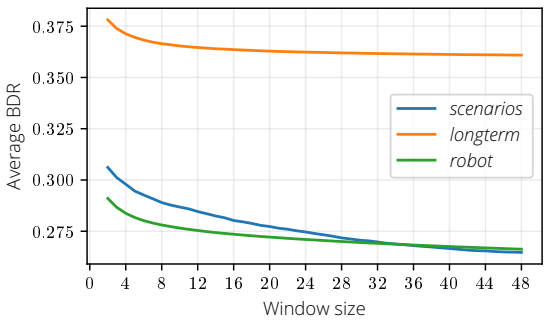
To finally remove this approximated static part from the current realization, we simply element-wise subtract $\bar{h}_i$ from h_i which delivers the final result h_i^*.

$$\begin{aligned} h_i^* &= h_i - \bar{h}_i \\ &= \{v_i^j - \bar{v}_i^j\} \quad j \in \{0, 1, \ldots, n\} \\ &= \{v_i^0 - \bar{v}_i^0, v_i^1 - \bar{v}_i^1, v_i^2 - \bar{v}_i^2, \ldots, v_i^n - \bar{v}_i^n\} \end{aligned} \tag{10.4}$$

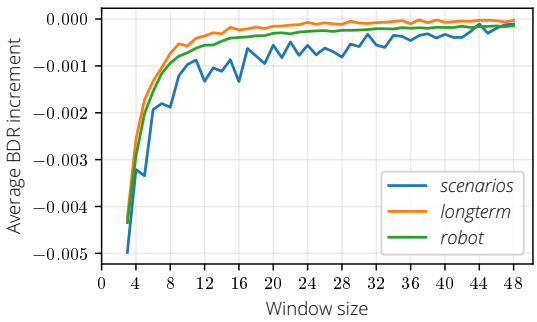
This process delivers a simple yet effective way of preprocessing the acquired CIRs. Further, the processing provides flexibility regarding the applied averaging function. Instead of Eq. (10.2) the values of $\bar{v}^j$ could also be computed through different averaging functions, e.g., the median. Nevertheless, the usage of the arithmetic mean as averaging function allows for an effective implementation as moving average.

Regarding the **parameter selection**, the relevant parameter to determine is the window size w. To actually find an appropriate value for w we use two approaches: first, and more practically oriented, we combine CCE with a quantization method and aim for the lowest BDR of the quantized bit strings, and second, in a more analytical approach, we examine the resulting entropy of the single bits in the bit strings.

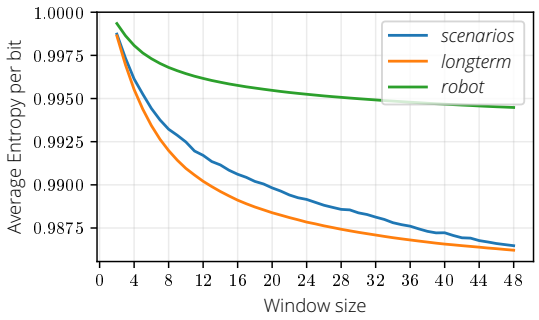
The practically oriented approach takes the subsequent step of quantization in advance and tries to optimize its result. Concretely, we execute the CCE preprocessing and the quantization for each of the legitimate partners. Subsequently, the BDR of the resulting bitstrings is analysed. To find to optimal value, we execute this analysis for window sizes $w \in \{2, \ldots, 20\}$ and determine the value with the minimal BDR. The window size 1 is excluded, since $w = 1$ would include the current CIR only — hence, the respective CCE approximation would be equal to the current CIR, $\bar{h}_i = h_i$ and the resulting vector h_i^* would only consist



(a) Average BDR for different w.



(b) Change between different w.



(c) Average entropy of single bits wrt. w.

Figure 10.1.: Analysis of parameter window size w with respect to the resulting BDR and single bit entropy of the resulting bit strings.

of 0 elements. Therefore, the window size $w = 0$ has to be excluded as well.

The results of this analysis are shown in Fig. 10.1. Figure 10.1a shows the achieved BDR for the respective window size. It is visible that the general trend is indicating a decreasing BDR for increasing window sizes.

However, this result has to be taken with caution: as shown in Fig. 10.1c increasing the window size w also decreases the entropy of the single bits of the resulting bit vector. This means, that with high values of w the resulting bits tend to be more static and less random. This is perfectly in line with the results in Fig. 10.1a: as more values are mapped to *static* results, the corresponding BDR decreases – at the expense of entropy. Hence, the resulting window size should not be increased as much as possible, but in contrary be kept as small as possible.

To find the right balance between these two opposing goals, we analysed the gains of increasing the window size, which is shown in Fig. 10.1b. Here, the respective changes of the BDR for increasing the window size are shown. The progression suggests that above a window size of $w = 6$, the changes become marginal.

It is worth noting, that the windows size is dependent on the sampling interval of single CIRs imposed by the hardware in use. A high sampling interval implies more changes in between the single acquired CIRs — hence, the window size should be smaller, to not generate static values as described above. Smaller sampling rates, on the other hand, allow for greater window sizes. Therefore, the actual window size has to be adapted to the sampling rate used.

In our concrete case, we employ a window size of $w = 6$.

10.1.3. Evaluation

The evaluation here mainly targets the general requirements for application in context of CRKG, i.e., the requirements detailed in Section 2.2. The resilience against the presented attacks is analysed in the upcoming Section 10.2.

The CRKG requirements we want to analyse are, specifically, the reciprocity of the quantized bit strings, their uniformity and their temporal independence. Hence, we analyse the following core aspects of the CCE results.

First, we inspect the achieved Mutual Information (MI) between Alice and Bob using this scheme. MI serves as metric for two core aspects of CRKG: it includes the reciprocity of the bit strings at the legitimate partners and additionally, due to its computation from the respective entropies, shows the extracted entropy as well. We expect that the application of CCE increases the achieved MI.

Second, we analyse the distribution of the resulting bit strings. More specifically, we analyse how close to a uniform distribution the achieved results are. This again serve two purposes: it exposes potential biases introduced, i.e., non-uniformly distributed resulting bit strings indicate a biased quantization, and it again hints at the quality of the entropy extraction, since the maximum entropy is achieved with a uniform distribution. We expect that the application of CCE moves the distribution of the resulting bit strings closer to the uniform distribution.

And finally, we inspect the autocorrelation of the resulting bit strings. This shows, whether subsequent bit strings are dependent on one another. Here, we expect that the autocorrelation will have a single peak at lag 0.

The *data sets* used for evaluation are data sets *scenarios*, *longterm* and *robot*. We have to consider, that CCE by design relies on successive measurements of natural movements or changes. Data set *attack* with the "simulated movement" by setting up terminals in different locations and conducting a multitude of measurements there does not provide such data. Hence, data set *attack* has to be excluded from the CCE evaluation.

Mutual Information In order to study the influence of CCE on MI as closely as possible, we perform the analytical calculation of MI. Since computing the analytical solution for bitstring of length 64 exceeds the hardware capabilities, we need to reduce the length of the bit vector. For this we proceed as follows: First we performed a subsampling of the incoming vector of length 64 to reduce its size to 16. The subsampling is realized by selecting every 4^{th} value of the vector and dropping everything else. Subsequently, we perform a threshold based quantization of the vectors, resulting in bit strings of length $16\,\text{bit}$. The reciprocal realizations of these are the input for the MI analysis.

This processing is performed once with the proposed CCE preprocessing and once without. The corresponding results are shown in Fig. 10.2. It is visible how the preprocessing increased the MI of the reciprocal measurements for all data sets. With data set *scenarios* CCE increased the MI by $3.67\,\text{bit}$, from $7.59\,\text{bit}$ to $11.28\,\text{bit}$. For data sets *longterm* and *robot*, the increases are large: with *longterm* the MI is increased from $5.46\,\text{bit}$ to $10.91\,\text{bit}$; for data set *robot* it was more than doubled from $5.95\,\text{bit}$ to $12.71\,\text{bit}$.

Notice that with a bit string length of $16\,\text{bit}$ the maximum entropy and MI is $16\,\text{bit}$ as well. This means, CCE could achieve up to 79.43% of the maximum possible MI. Concretely,

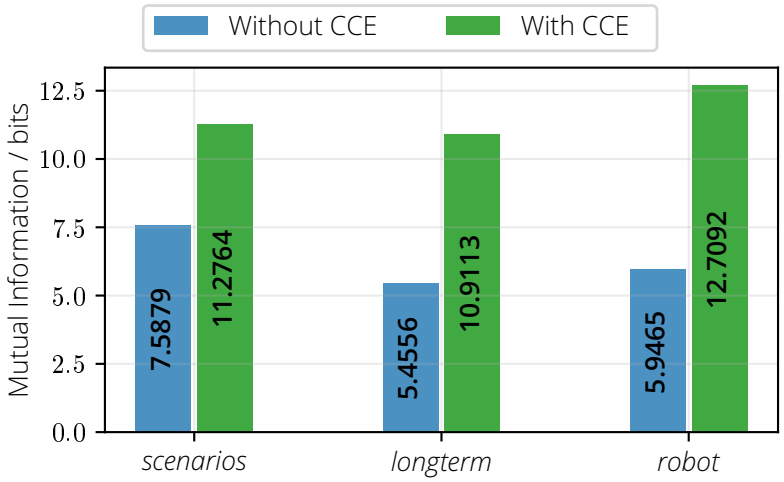


Figure 10.2.: Resulting mutual information with and without CCE preprocessing.

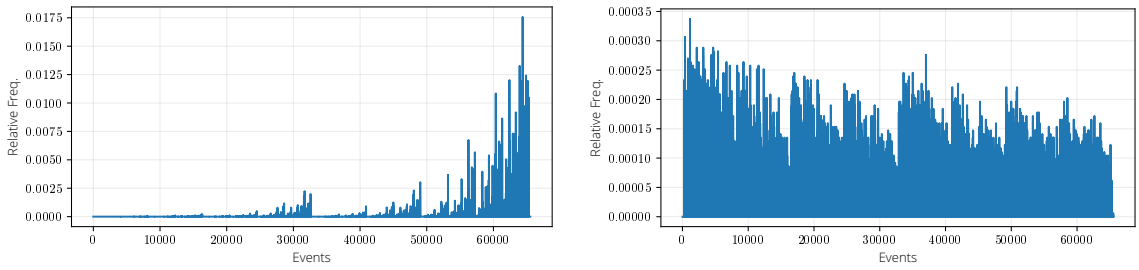
70.48%, 68.20% and 79.43% are achieved for data sets *scenarios*, *longterm* and *robot* respectively.

Overall, the MI results show that the CCE preprocessing effectively extracted the reciprocal features of the CIRs and additionally enable the better preservation of the contained entropy.

Uniformity To assess the uniformity of the resulting quantized bit strings, we start with the histograms of the respective quantization. For data set *robot*, these histograms of the results with and without the application of CCE are depicted in Fig. 10.3. The respective results for *scenarios* and *longterm* are shown in Appendix A.3.

The two plots illustrate how the application of CCE changes the distribution of the resulting bit strings and that this distribution is thus much closer to uniformity than without CCE.

These results also highlight that a straight forward application of threshold based quantization approaches lead to significantly biased distributions of the outcome: Due to the typical progression of CIRs, a simple threshold based approach will inherently result in disproportionally many one bits in the most significant bits. Hence, the larger numbers of the event space, i.e., the bit strings, are significantly overrepresented in the resulting distribution. Such, biased distributions of key material should be avoided for key generation as it



(a) Without CCE

(b) With CCE

Figure 10.3.: Histogram of the 2^{16} possible quantization outcomes.

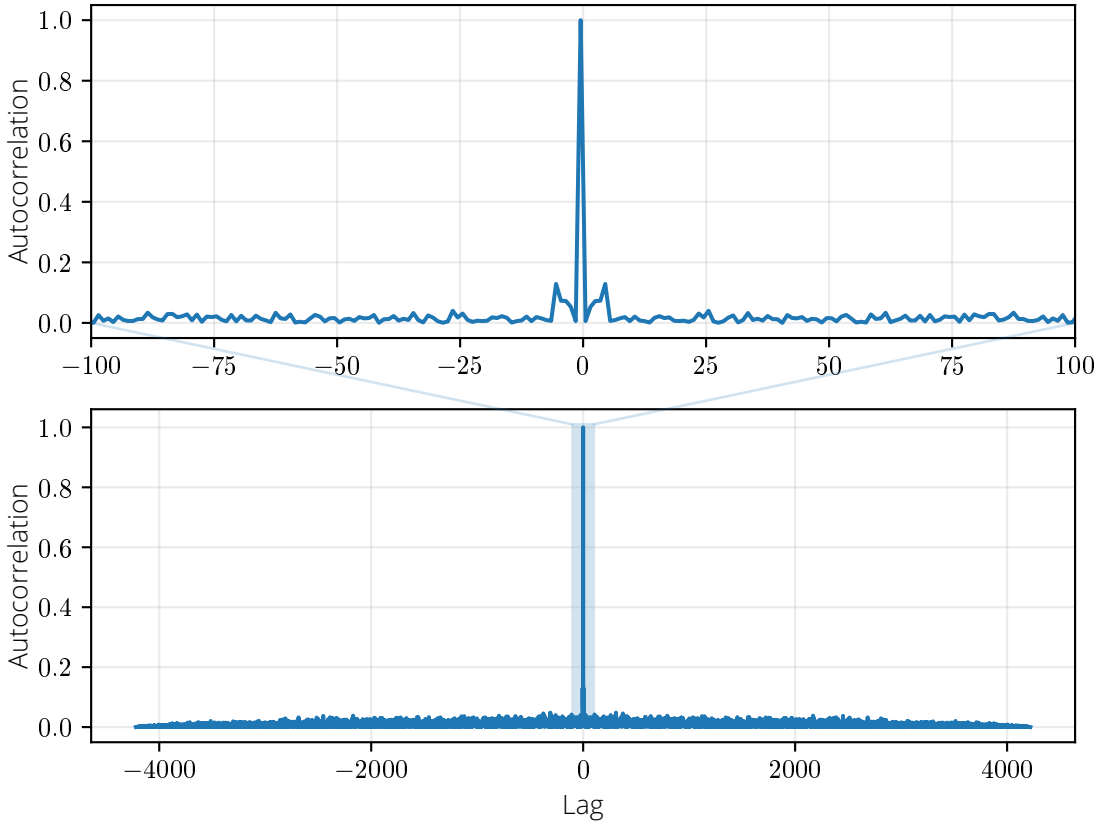


Figure 10.4.: Autocorrelation of the quantized results of data set *scenarios* after the application of CCE.

inherently weakens the schemes security.

Autocorrelation The plot of the resulting autocorrelation for data set *scenarios* is shown in Fig. 10.4. The single spike at lag 0 is clearly visible. Additionally, the zoomed in section shows that smaller lags do not exhibit strong correlations either. The second-largest correlation is at log 12 with 0.1288.

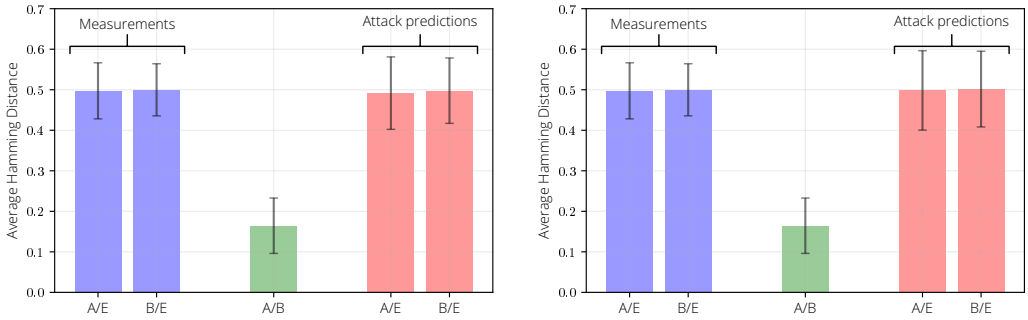
Overall, these low correlation coefficients for lags $\neq 0$ suggest that no linear relation between the subsequent bit strings exist. The very low correlation combined with the single peak at lag 0 strongly indicates an independent distribution of the underlying sequence. Data sets *longterm* and *robot* resulted in similar autocorrelation curves, which are display in Appendix A.4

10.2. Attack Resilience

In the following we evaluate the CCE preprocessing with respect to the attacks presented. As we are looking at a preprocessing and quantization scheme, we focus on the resulting hamming distance of the quantized bit strings.

The general approach for the evaluation is the following:

- The legitimate partners operate in accordance to the CCE design.
  Alice and Bob process the incoming CIRs as they are recorded. Based on this suc-



(a) Channel Model-based attack (b) Ray Tracing-based attack

Figure 10.5.: Average Hamming Distances for overheard CIRs (blue bars), legitimate CIRs (green bar) and the values predicted by the two "classical" attacks (red bars) after execution of CCE on data set *robot*.

cession of measurements the CCE is calculated and subtracted. Afterwards, a one bit quantization is executed.

- The adversary performs the attacks as described in the preceding sections and perform CCE on the attack results.
  Eve executes the attack under evaluation as presented in the respective section in Part IV. Afterwards, CCE is applied to the resulting CIRs predictions.

- Compute the hamming distance, i.e., the BDR, between the results.
  To actually inspect the implications of CCE we analyse the BDR between the bit strings quantized by the legitimate partners and those resulting from Eves attack.

Following the reasoning of CCE, we expect that the application will significantly increase the BDR for the attacker. Optimally, the resulting BDR for the predictions of Eve should be 0.5 — this means, that Eve has no better chance than guessing each bit of the result, implying that the performed attack does not generate any advantage at all.

Regarding the *data sets* used for evaluation of the attack resilience, we have the following restrictions: As described during the evaluation of CCE we have to exclude data set *attack* for CCE. This also affects the evaluation of the attack resilience, since the channel model-based as well as the ray tracing-based attack rely on location information which are only available in data sets *attack* and *robot*. This in turn means that for these two attacks we evaluate CCE on the data set *robot*. The final and most effective attack based on machine learning can be evaluated on the same data set as described in Section 9.3, i.e., data sets *scenarios*, *longterm* and *robot*.

The resilience of CCE against the two "classical" attacks is visualized in Fig. 10.5. Since both are evaluated against the same data set *robot*, the results of the legitimate partners and of the overheard CIRs are the same. Concretely, after the application of CCE the legitimate partners achieve a BDR of 0.1648, with a standard deviation of 0.0639. The eavesdropped CIRs have BDRs of 0.4972 and 0.4999 compared to the realizations at Alice and Bob, respectively. Their deviations are similar with 0.0692 and 0.0643.

As for the single attacks, the results for the **channel model-based attack** are shown in Fig. 10.5a. After applying this attack, the adversary achieves BDRs of 0.4917 and 0.4978. The

distribution of the results become more spread out with standard deviations of 0.0894 and 0.0807.

In the same vain, the results for the **ray tracing-based attack** are depicted in Fig. 10.5b. The resulting CIRs deliver BDRs of 0.4983 and 0.5016, again compared to Alice's and Bob's CIR respectively. The standard deviation became a little wider compared to the measurements with 0.0980 and 0.0943.

Both results show, that the application of CCE effectively removed any attacker advantage.

The results for the **machine learning-based attack** are presented in Fig. 10.6. Overall, they confirm the results from the previous attacks: the CCE preprocessing successfully thwarted the attack completely. The adversary achieves an average hamming distance of 0.5 for both the measurements themselves and after the attack is executed — which is the worst result for the attacker and the best for Alice and Bob.

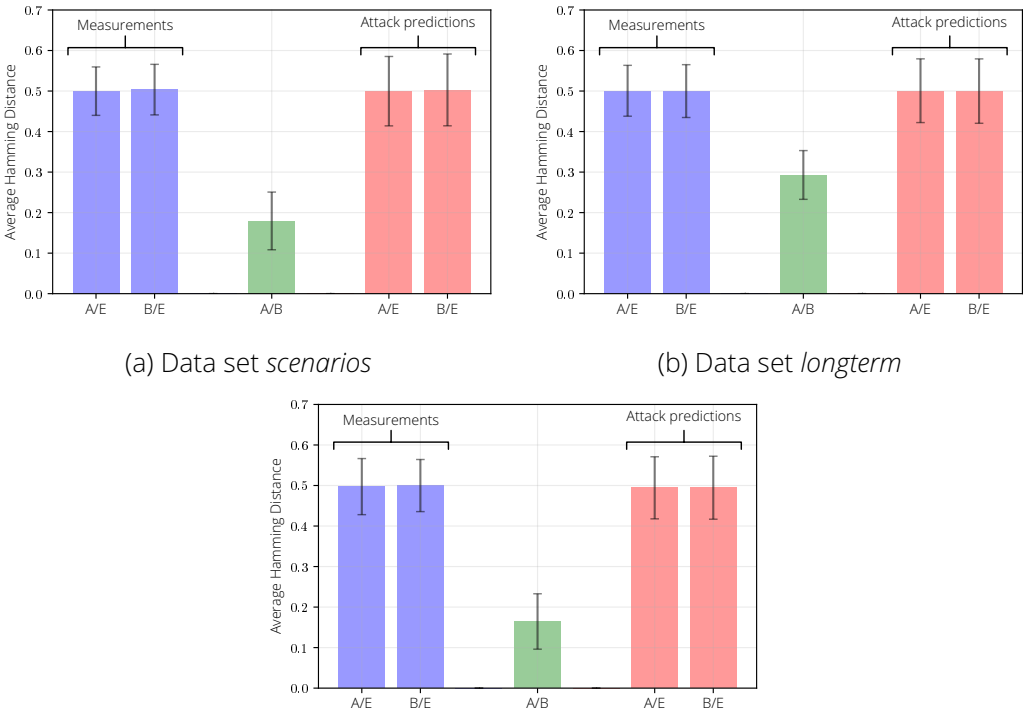
Concretely the following BDRs are achieved: For data set *scenarios* the attacker measurements after CCE have a BDR of 0.4997 and 0.5035 for Alice's and Bob's CIR, respectively. After execution of the ML attack, these values are at 0.4997 and 0.5028. The standard deviation of the attack results are higher than for the measurements with $0.0856/0.0886$ compared to $0.0598/0.0624$.

With data set *longterm*, the values before and after the attack change from 0.5007 to 0.5008 for Alice/Eve and are constant at 0.4999 Bob/Eve. Here, the spread became also wider with 0.0786 and 0.0795 compared to 0.0628 and 0.0652.

Finally, for the data set *robot*, the average BDR decreased marginally. As shown in for the previous two attacks, the BDR with CCE before executing an attack is 0.4973 and 0.4999, for Alice and Bob respectively. After the ML-based attack it is at 0.4943 and 0.4945. These values are noticeable because it is slightly below 0.5 for *both* partners. Nevertheless, these results still do not show any advantage for the attacker and are clearly above the legitimate communication partners' value of 0.164.

It is worth noting, that the ML-based attack can also be adapted to directly infer the results of the CCE preprocessing. However, using this approach does not yield attacker success either: We used the same attack approach to directly infer the CCE results and still could not generate an advantage for the attacker. Here, BDRs of 0.4920 and 0.4913 are achieved for Alice's and Bob's CIR, respectively. Although a minimal improvement of 0.02 points was achieved, the overall results still indicated that CCE is resilient against this attack.

Overall, these results clearly demonstrate, that the proposed CCE preprocessing successfully thwarts all presented attacks. None of the attacks achieved a value significantly below 0.5. This means that using CCE gives Alice and Bob a clear advantage in terms of security. After using it, the attacker has no reliable way to predict or infer the legitimate bit values.



(a) Data set *scenarios*

(b) Data set *longterm*

(c) Data set *robot*

Figure 10.6.: Average Hamming Distances for overheard CIRs (blue bars), legitimate CIRs (green bar) and the values predicted by the Machine Learning-based attack (red bars) after execution of CCE.

11. Introduce Machine Learning to the System Model

Parts of this Chapter are published in [197].

In the same vein as attackers can use the powerful capabilities of machine learning primitives to attack CRKG, system designers can leverage these methods to make the key exchange more secure and efficient. In the following, we present a machine learning-based solution, called *Blind Twins*, that aims to solve two problems at once: first and foremost, to be secure against the attacks described earlier. And additionally, we want to reduce the communication between Alice and Bob as much as possible.

Again, the same design principles of the general solution approaches described in Chapter 6 apply. This means that we want to design a solution which operates blindly, i.e., without interaction, as well as online, meaning every collected CIR sample can be processed right away.

11.1. Blind Twins

In this section, we introduce the machine learning approach for CRKG. Although the primary focus is on Information Reconciliation, the specific processing also implicitly incorporates preprocessing and quantization, making this an "all-in-one" solution for CIR-based CRKG. As described above the main design focus of this approach is twofold:

Attack resilience The approach should be resilient against the described attacks.
This is the core idea — to prevent the attacks described in Part IV, ideally completely thwarting them.

Communication avoidance Since superfluous communication is detrimental to CRKG, we aim for a solution which avoids communication completely.
The complete removal of communication would not only remove the negative effects of information leakage, surplus energy usage and efficiency reduction through wait times. Further, the requirement of an additional authenticated communication channel to exchange the respective messages can be dropped. By removing this strong

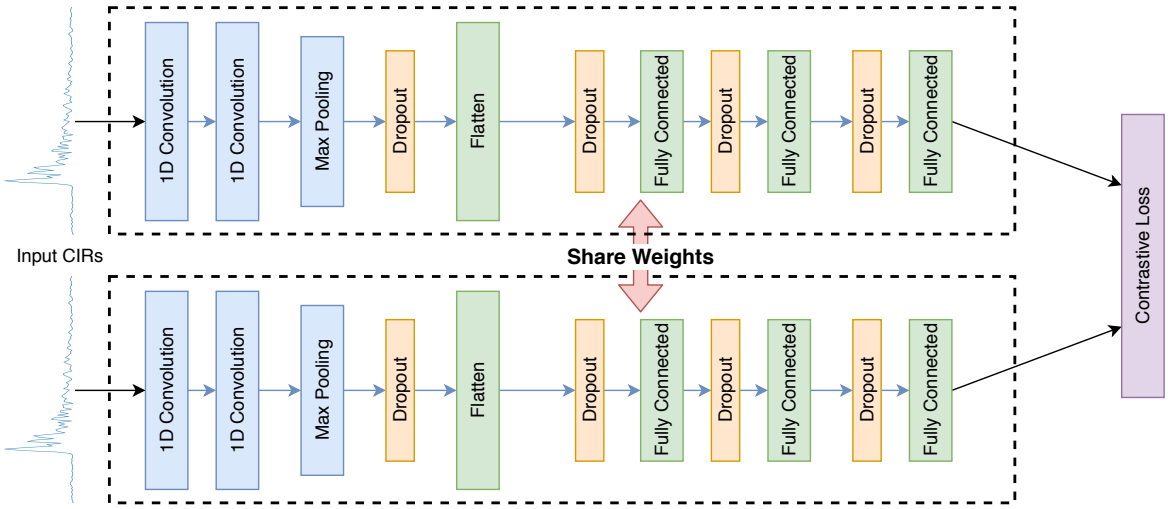


Figure 11.1.: The core setup of the proposed Siamese Network — the CNNs within the dotted lines share their weights and are essentially one and the same network. This CNN becomes the resulting network deployed at CRKG nodes in practice.
For the *triplet loss*, three instances of the base CNN would be created and their respective outputs combined in the triplet loss function.

requirement, CIR-based CRKG becomes significantly more realistically applicable and practical.

In addition to these two main goals, we naturally also want to create an effective solution that achieves competitive BDR results.

We address all of these goals *together* with the approach presented here.

To achieve these goals, we design a neural network that is trained to efficiently recognize and extract those features of the CIRs that specifically drive the channel reciprocity. Further the network discard the deterministic features, which are easily inferable by an attacker, i.e., those features upon which the presented attacks thrive.

To the best of our knowledge, this is the first IR approach facilitating machine learning primitives and, thereby, achieves reconciliation without communication.

11.1.1. Concept: Siamese Networks for Information Reconciliation

The **core idea** of our solution is to blindly extract the reciprocal channel characteristics that are unique to the positions of the legitimate partners. We train a machine learning model that distinguishes reciprocal components of legitimate measurements from those overheard by the adversary, for this purpose. The model can be trained in advance, once, and subsequently be used for blind information reconciliation. To make sure that Eve, even in possession of trained models, cannot approximate the sequences reconciled by Alice and Bob, we aim at extracting exactly those features, which represent the characteristics of the legitimate channel well.

We leverage two ML concepts for this purpose. For extraction, the task is to take uni-dimensional, sequential data and to output a sequence of bits. The input data at Alice and Bob will be subject to transformations, most importantly due to gain differences and temporal shifting as described in Section 2.3. According to literature and as shown in the

ML-based attack, such feature extraction tasks are well performed using convolutional neural networks (CNNs) [111, 235]. Their real-valued output at the last layer can be quantized using a simple threshold function. Thereby, a complete solution is established taking real valued CIRs as input and yielding reconciled bit strings as output.

We also want to maximize the advantage of the extraction at legitimate parties versus extraction at an adversary in an arbitrary position (different to the exact location of Alice or Bob). We hence want to train the network to project correlated input sequences nearer, and decorrelated input sequences further apart in the output space. For this purpose, we train our CNN in a Siamese Network setup. In combination with a discriminating loss function, like *contrastive* [76] or *triplet* loss [85, 166], this architecture is explicitly designed to enable the learning of discriminative features within a single network. Here, the base network learns to extract and use the unique features that represent the CIR reciprocity, and to disregard all others.

Together, the architecture of suitable CNNs trained in a Siamese Network with contrastive/triplet loss, provide the properties needed for interaction-free IR.

Our **concrete realization** combines these concepts as follows: The Siamese network itself is instantiated by creating two of the above CNNs as Siamese twins with shared weights as shown in Fig. 11.1. Since the networks share their weights, they effectively are two views of the same network. To train the network with *contrastive loss*, the data is prepared by defining pairs of data, which are flagged as collected from reciprocal measurements or not. This flagging represents, whether the observations pairs originate from either Alice and Bob or from Alice and Eve or Bob and Eve. Each of the Siamese CNN instances then processes one CIR of this pair. By feeding the output of both CNNs combined with the similarity flag to the *contrastive loss*, reciprocal pairs generate low losses for similar outputs, whereas non-reciprocal ones are generating low losses for non-similar outputs. Hence, through the respective backpropagation, the shared base network learns to discriminate between the different input pairs. Thus, it effectively learns to extract those feature which are reciprocal and discard non-reciprocal ones.

To train with *triplet loss*, the input data is not augmented with a similarity flag but arranged into triplets of anchor (Alice), positive (Bob) and negative (Eve) samples. Subsequently, *three* Siamese instances are used to learn the discriminating features. The core setup is similar to Fig. 11.1, but with three instead of two coupled network instances.

After one-time training, the base CNN is deployed at the terminals and used to generate reconciled outputs without seeing the partners input.

An attacker with access to the CNN, e.g., a malicious insider, is still not capable of reconciling the same sequence since the overheard messages are different: the obtained h_{AE} is not reciprocal to h_{AB} (h_{BE} and h_{BA} accordingly). Hence, the output of the attackers CNN is different to those of the legitimate terminals[1].

A **deployment** then is done in the three following steps:

1. Instantiate the Siamese Network with an appropriate base CNN.
2. Train the network once with *contrastive* or *triplet* loss.
3. Deploy the base CNN at the terminal and use them for IR on live CIRs.

[1] In fact, the potential CIR pair h_{AE} and h_{EA} would reconcile into the same output. But this is a valid key exchange scenario between reciprocal terminals and not an attack.

As the CNN are trained to distinguish reciprocal from non-reciprocal features in the CIR locally, this information reconciliation does not require any interaction.

11.1.2. Instantiation of the Siamese Network

The architecture of the model has three core properties that are relevant for actual instantiations: the base network with its respective architecture, the activation function of the final layer and finally the loss function used for learning.

For the base network, we implemented two different approaches: first, a rather simple network to test feasibility of the approach in general, and second, a more generalized network to show applicability. Both approaches build from the basic idea of an initial convolutional layer, intended to capture the relevant features of the input data, followed by fully connected layers, which combine those feature into the final results.

Since the first network, **CNN1**, is intended solely to demonstrate the approach's feasibility, we omitted all measures for generalizations. The network is compromised of a single *1D convolutional layer* (1DConv) with 128 kernels of width 3, followed by 3 *fully connected layers* (FC) with 1024, 512 and 256 nodes, respectively. The final layer, whose output we call embedding, is again *fully connected*, with the targeted embedding size chosen to be 16. All layers are activated as *rectified linear units* (ReLU). It is expected, that this network overfits the training data and is not generally applicable.

To demonstrate effective general applicability, we devise a second, more generalizing network **CNN2**. It consists of 2 *1DConvs* with respectively 32 and 64 kernels of width 3, followed by a 1D MaxPooling layer of size 2. After flattening, there are 2 FC layers with 256 nodes and the final FC layer sized to the embedding. We include a dropout with rate 0.5 before each FC layer (including the embedding layer). Again, all layers are ReLU activated.

For both base networks, the activation of the embedding layer, i.e., the last layer of the base network, is especially important, due to its direct impact on the loss function. In the first iteration, this activation was set to ReLU as with the intermediate layers. This is reasonable when used in combination with the Euclidean distance or L2-Norm as distance function D_W.

Our goal, however, is to reconcile the input into a *binary* sequence, so we changed the activation to a *sigmoid* function, later. Restricting the final output to $[0, 1]$, the quantization is implicitly taken care of in this case: for a robust solution, we simply chose a threshold of $\theta = 0.5$ for quantization of the final embedding. This yields the great advantage that this processing implicitly takes care of the quantization step as well.

Nevertheless, this quantization design invalidates the L2 norm as chosen distance metric. It also does not reflect the intention of the IR process and its final data: the main goal is to have equal *binary* sequences for the legitimate partners and different ones for the attacker. We hence would prefer the Hamming distance between the binary sequences to measure the final loss.

Including this quantization and Hamming distance calculation would result in a non-differentiable function, meaning that learning would not be possible. Therefore, instead of the classical Hamming distance, we chose its continuous variant as distance function D_W:

$$D_W(x, y) = y(1 - x) + (1 - y)x \tag{11.1}$$

This metric can now be optimized to reach 0 for reciprocal/similar pairs (Alice and Bob) and 0.5 for non-reciprocal/dissimilar pairs (Eve).

Finally, the loss function used to learn the separation is of essential importance. Following the reasoning above, the two candidates are *contrastive loss* (CL) and *triplet loss* (TL).

The **contrastive loss** is applied as

$$L(W, Y, \vec{X_1}, \vec{X_2}) = (1 - Y)\frac{1}{2}D_W^2 + Y\frac{1}{2}(\max(m - D_W, 0))^2 \tag{11.2}$$

Here, $\vec{X_i}$ are the respective outputs of the Siamese instances, D_W is $D_W(\vec{X_1}, \vec{X_2})$ and Y is the similarity flag. The margin m is set to 0.5, as we aim for a Hamming distance of 0.5 for "non-similar" pairs, i.e., Eves observations.

The **triplet loss** is realised as

$$L(W, \vec{X_A}, \vec{X_P}, \vec{X_N}) = \max(D_W^P - D_W^N + m, 0) \tag{11.3}$$

D_W^P denotes the distance between the anchor $\vec{X_A}$ and the positive sample $\vec{X_P}$, D_W^N is the distance between the anchor and the negative sample $\vec{X_N}$. The margin m is again set to 0.5 with the same reasoning as above.

Training and Data The data for training and evaluation of the *Blind Twins* approach were obtained in two different ways: on the one hand we employ the extensive real world measurements obtained to show the practical applicability of the approach, and on the other hand, we used synthetically generated CIRs to demonstrate that the approach generalizes well.

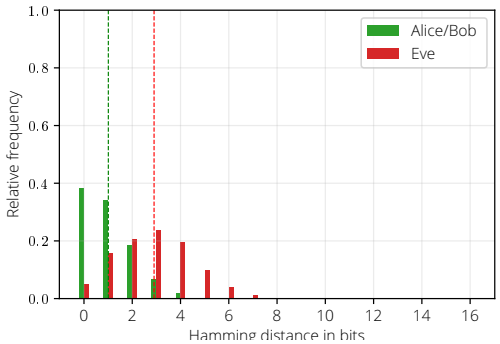
For the **real world measurements** we used the data set *scenarios* to explore the general feasibility of this approach. In the upcoming Section 11.2 we extend the evaluation to the other data sets.

The samples of the different scenarios were split into 70% training and 30% evaluation sets. All training samples were combined into on data set and permuted before the actual training. The same procedure was followed with the evaluation data.

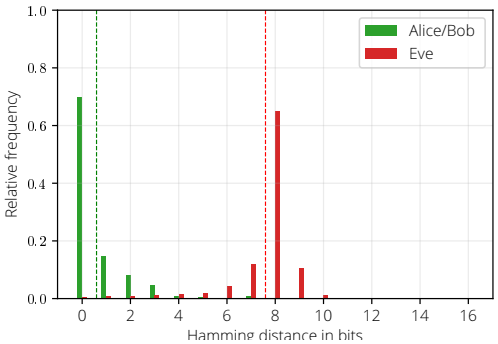
In analogy to Section 8.3, the **synthetic data** was created by adapting the deterministic Kunisch-Pamp channel model for UWB CIRs. Given a predefined environment and setting for transceiver position and properties, this model allows to deterministically generate noisy impulse responses with correlations similar to real world measurements. We used the obtained real world measurements to perform a Bayesian optimization of the model parameters (cf. Section 8.3). Thereby, the resulting parameter set resembles the environment of the real world measurements. Given the model and this specific parameter set, we can create arbitrary CIRs for this environment.

11.1.3. Evaluation

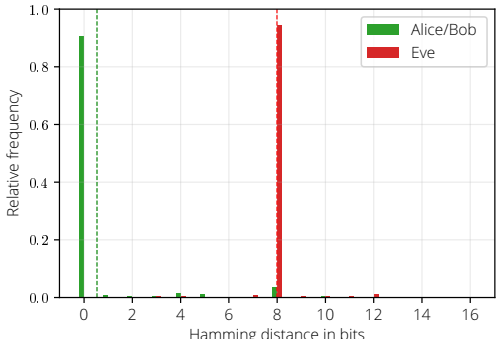
For our evaluation we employ the BDR as describe in Section 2.5. As we have a constant vector length, the BDR is equal to the average Hamming distance between the binary vectors. We depict the histogram of achieved Hamming distances to convey their actual distribution, instead of the mere average. The length of our embedding vector, $16\,bit$, is also the maximum distance, i.e those of two inverted sequences. For an attacker, the worst case



(a) Contrastive Loss, D_W is Euclidean distance



(b) Constrastive Loss, D_W as in (11.1)



(c) Triplet Loss, D_W as in (11.1)

Figure 11.2.: Effectiveness of different **CNN1** parameter realizations. The dotted lines represent the BDR.

would be a distance of 0.5, in our case equal to a distance of $8\ bit$, because in this case the attacker can only guess which of her bits are correct. In the plots, the Y-axis always shows the relative frequency and the X-axis the Hamming distance in bits; the dashed line is the BDR.

Different Network Parameters and Losses

We show the evaluation results for different parameters of **CNN1** in Fig. 11.2. The interpretation of these results is twofold:

First, the plots demonstrate the general capability of the proposed network architecture to differentiate between CIRs of legitimate partners and those of an eavesdropper. Especially, Fig. 11.2c depicts the clear discriminative strength of the network: the learned embeddings of the legitimate partners Alice/Bob have very low Hamming distances (0.033 BDR), whereas the eavesdropper Eve has a Hamming distance close to 0.5 (0.469 BDR) for her observations.

Second, the results show the influence of the different described network parameters. Figure 11.2a shows contrastive loss with Euclidean distance as D_W. The general effectiveness of the network is visible, but not strongly expressed: the legitimate partners reconcile to the same bit sequence in only 38% of cases, and the distribution of the attacker, albeit clearly distinct, is still very close. The network in Fig. 11.2b employs the same loss, but with the continuous Hamming instead of the Euclidean distance. This clearly increases the sep-

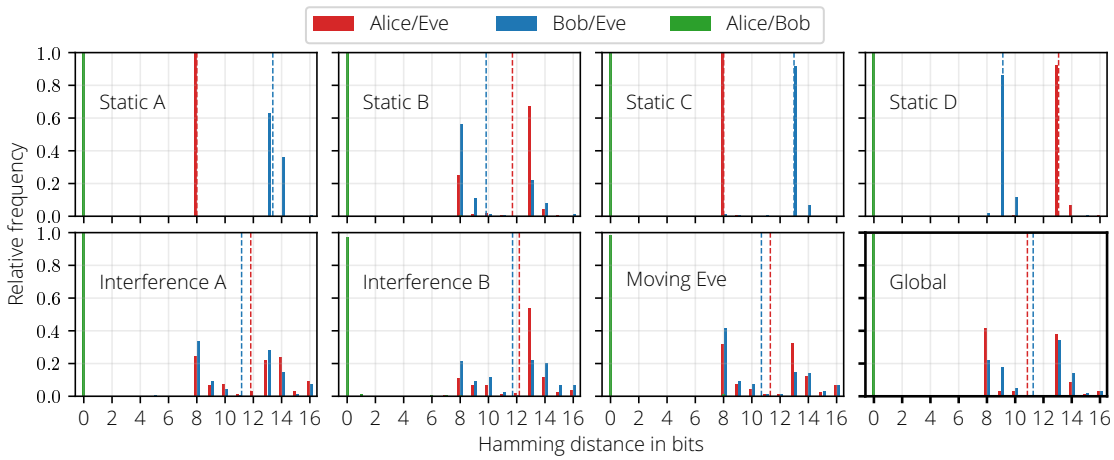


Figure 11.3.: The histograms of achieved Hamming distance of data set *scenarios* after application of the trained model. The dotted lines represent the BDR.

aration between the legitimate and the adversarial observations: Alice/Bob have a BDR of 0.037, whereas Eve is restrained to a BDR of 0.475. Finally, Fig. 11.2c shows the increased performance of applying triplet loss: separation is significantly better, with BDRs of 0.033 and 0.499, respectively, and a decreased standard deviation, e.g., from 0.087 to 0.049 for Eve.

It is worth noting that with **CNN1**, the attacker sometimes achieves a Hamming distance of 0, i.e., extracts the same binary sequence as Alice/Bob. This originates from the simple architecture and overfitting of **CNN1**. The more powerful and generalizing network **CNN2** completely removes this artifact as we show in the next section.

General Real World Performance

Network **CNN2** was then used with continuous Hamming distance and triplet loss to evaluate the approach's applicability in real world scenarios.

The results of the evaluation with the real world measurements are depicted in Fig. 11.3. The lower right plot is a summary of all scenarios. The most notable outcome is that for all cases the reconciled sequences of the legitimate partners are equal in nearly all cases $(99,7\%)$, whereas the attacker reach at most a distance of 0.5. This means, that the legitimate partners robustly are reconciling into the same sequence, while the attacker is unable the gain any significant insight into this sequence, despite full knowledge of the used trained network.

Using the triplet loss with a margin $m = 0.5$, we expect the following: First, through the "pull" of the positive samples, the reciprocal observations will have a Hamming distance of 0. Second, since only negative samples with distance $< m$ contribute to the loss, there will be few attacker results lower than 0.5, i.e., $8\,bits$, because these are "pushed" to the upper half of the histogram. This should be particularly apparent in static scenarios, as these yield relatively stable observations. Finally, due to the generalizing measures, the results will be tolerant of movement and interference, which will be particularly visible in scenarios with such characteristics.

The evaluation results of the data set *scenarios* accurately confirm these expectations:

Within the **static scenarios** of this data set, the network achieves a distinct separation between the reciprocal and non-reciprocal results. Alice and Bob achieve a BDR of 0, i.e., Hamming distance of 0 in all cases, with an overall BDR of 0.003. Only scenarios *static B* and *static C* deviate slightly, with a BDR of 0.005. The attacker achieved an average hamming distance of 0.692, while no result has a distance lower than $8\,bit$. As the static scenarios are also static for Eve, her reconciled sequences have rather stable distributions.

The results of the **dynamic scenarios** verify the general effectiveness: Despite the presence of unfavourable movements[2], the legitimate partners reach perfect reconciliation with frequencies of 1.0 (IA), 0.97 (IB) and 0.99 (ME). The attacker's observations again are located in the histograms "upper" half, with a BDR of 0.717. Due to the additional interferences, the attacker's binary sequences are much more scattered than in the static scenarios.

To further assess the security of this approach, we used the *Kunisch-Pamp* channel model for **synthetic attacks** to rule out that we missed an advantageous attacker position: We used the real-world measurements to optimize the parameter of the channel model to our measurement environment. Then we generated attacker observations for all positions in this room, in steps of at most $\lambda/2$, using this optimized model. This synthetic attack data was then again processed with the trained network. The results are shown in Fig. 11.4. The average Hamming distance for this synthetic attack is 0.43. It is again visible, that in no positions the attacker reconciles into the same sequences as Alice/Bob. Nevertheless, the distribution of the histograms indicates that there are in fact more advantageous positions for the attacker. These might be positions in very close vicinity of the legitimate partners or positions where multipath clusters are shadowing each other.

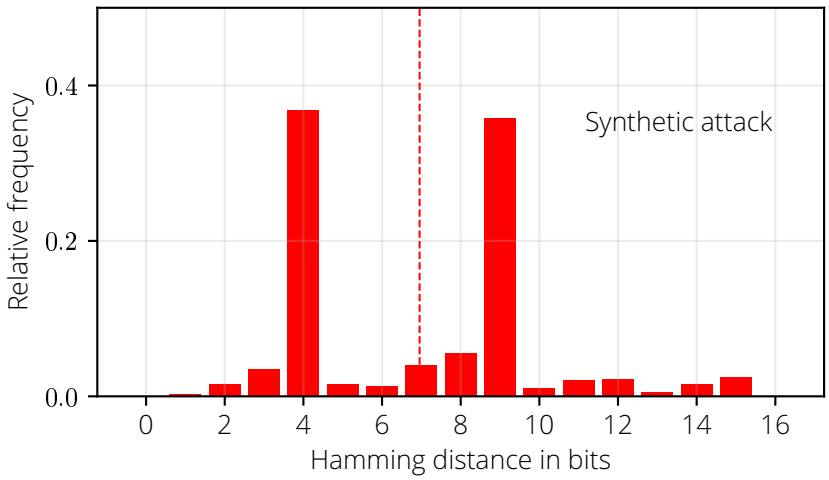


Figure 11.4.: The histograms of Eves Hamming distances for synthetic attack data.

Overall, the results obtained show that the trained network effectively implements a non-interactive Information Reconciliation. The synthetic attack even shows that attackers with knowledge of the network in especially favourable positions are not capable of obtaining the same sequence as the legitimate partners. Further, the average entropy of the single reconciled bits is $0.99\,bit$. Hence, the network has not learnt something static, but in fact

[2] Movements per se are beneficial for CRKG, as they generate entropy to successive CIRs. Our experiment also includes detrimental movement, like Eve moving directly on the Line-of-Sight between Alice/Bob.

extracts the reciprocal randomness. The overall success rate for the proposed IR is 0.992, which in turn can be represented as a BDR of 0.003. As invalid IR is a valid CRKG protocol outcome [26], successful reconciliation in 99.2% of all interaction cases is a very good result. Compared to state of the art CIR solutions, our results are very competitive: current solutions reach as low as 84% for successful IR [224]. But even high performing approaches like [120] reach at most a BDR of 0.004, which is still higher than the 0.003 achieved by our approach. This means, our solution is, in terms of effectiveness, at least as good as state of the art solutions. Additionally, it is completely interaction free, i.e., it leaks no information at all and can be processed quicker.

11.2. General Blind Twins

In the preceding section the *Blind Twins* scheme was applied to the data set *scenarios* to show the effectiveness of the scheme. Nevertheless, the straight forward implementation presented there comes with two drawbacks: first, the generalization capabilities are not sufficient for diverse data sets like *robot*. This means, that in this instantiation *Blind Twins* does not work for the data set *robot*. Concretely, the separation between the legitimate CIRs and the overheard ones could not be guaranteed. And second, a network trained in this manner is specialized to the exact room in which the training data was captured.

To fix these two problems, we extended the Blind Twins approach to a general one. For this, two modifications were put in place: first, the input data was extended, and second, the training itself was adapted. Using these adaptations, we successfully implement a solution that works for all data sets, and most importantly, generalizes well enough that after training on a single data set, the network also performs successfully for the other datasets.

We now proceed to describe the modifications in detail.

11.2.1. Extension of Input Data

With the same reasoning as for CCE, we expand the input to Blind Twins to the latest six recorded CIRs. Thereby the natural progression of the terminal through the physical environment is accounted for. Additionally, this inclusion helps to determine the static parts of the respective CIR — such parts need to be identified and removed as countermeasure against the attacks.

This alteration is represented in the network, by expanding the input tensor from former shape $(n, l, 1)$ to $(n, l, 6)$. Here, n is the batch size of the training and l is the length of the current CIR. To accommodate for the slow fading effects, e.g., multipath component which slowly "move" within the CIR, we also expanded the kernel width of the 1DConv layers from 3 to 7 and 5, respectively.

11.2.2. Training Adaptions

To ensure a successful training even on diverse data sets, we adapted the training in two ways: first, we made the training as hard as possible for the desired loss function and second, we took measures to prevent the network from collapsing into a trivial solution.

To increase the difficulty of the training we in turn implemented the following two measures. First, we ensured a non-negative loss, which would be ignored by the triplet loss

through the maximum, throughout the entire training process by implementing *hard mining* as proposed by [166]. Since our specific use case resembles *one-shot* training, i.e., every location specific CIR triplet is only seen once, we cannot employ online hard mining but have to use offline hard mining. Further, to establish a correct ground truth for the mining, we did not employ the distance of the embedding, but our known distance metric of normalized cross-correlation with the raw CIRs. Concretely, for a triplet of anchor h_{AB}, positive sample h_{BA} and negative sample h_{AE}, we calculated the hardness score f_{hard} as

$$f_{hard} = r(h_{AB}, h_{BA}) - r(h_{AB}, h_{AE}) \tag{11.4}$$

Here, $r(g, h) = r_{gh}$ is the cross correlation as defined in Eq. (2.38). With this calculation, "easy" samples will have a large score, since $r(h_{AB}, h_{BA})$ becomes large due to the similarity, whereas $r(h_{AB}, h_{AE})$ becomes small due to decorrelation. For hard samples, it will be the other way around resulting in a lower score. Hence, for the actual mining, we sort the triplet according to their hardness score f_{hard} in ascending order and then use only the first n triplets for the actual training. In our case, we selected the first $100\,000$ triplets.

Additionally to the offline hard mining, we also decreased the batch size for the training process. In contrast to typical batch sizes for triplet networks, which are in the order of 1000 samples per batch [166], we employ batch sizes typical for gradient descent training, i.e., at most 64 samples per batch [207, 131]. This is again rooted in the difference between online and offline hard triplet mining — the large batch size for triplet networks follows the reasoning, that large batch sizes will include several hard triplets for *online* mining. Since we are using *offline* mining, there is no reason to increase the batch size. Hence, we employ a batch size of 64.

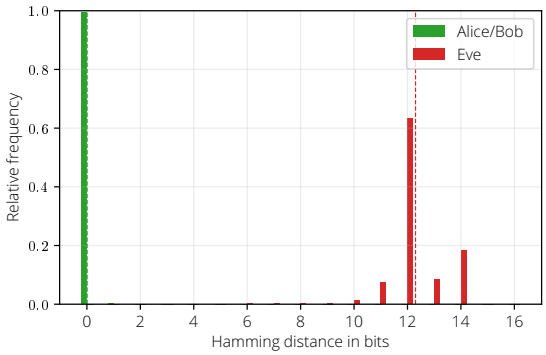
Finally, we applied the following data augmentation to increase the generalization capabilities of the CNN. For all available CIR triplets we created a copy of this triplet where the time axis is reversed. This effectively "flips" the CIR along the time axis. Thereby, the network cannot rely on the general progression over time, but has to actually identify and extract the features within the CIR.

To prevent the training process from collapsing into a trivial solution we also adapt the optimizer in use. With the increased hardness of the training, the risk of a collapsed solution increases as well. This means, it becomes more probable that the network statically maps all samples to fixed value, irregardless of the actual input. To avoid such collapsed solutions we reduce the learning rate of the Nadam optimizer in use from originally 1×10^{-3} to 1×10^{-5}. In our concrete case, this effectively prevents collapsed solutions.

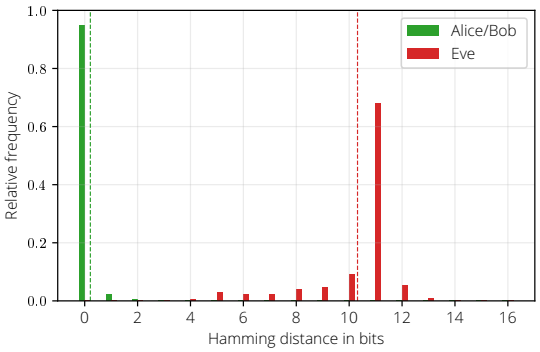
11.2.3. Evaluation

The evaluation of the general Blind Twins approach is performed analogously to the Blind Twins above. This time we use not only the *scenarios* data set, but all data sets appropriate for it, i.e., *longterm*, *robot* and *scenarios*. The evaluation focuses on the one hand on the functionality of the diverse data set *robot* and on the other hand on the transferability from one data set to others.

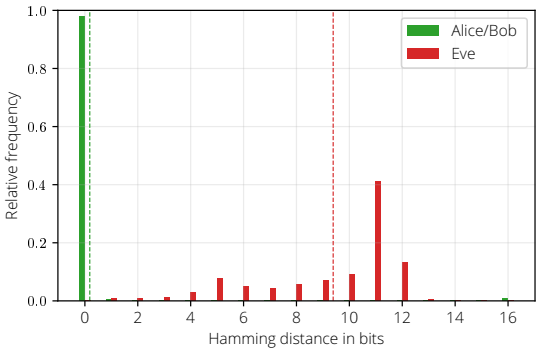
Since we need diverse input samples to effectively facilitate the offline hard mining, we use the data set *robot* for training. Here, we facilitate the measurements of the attacker in position $E1$. From these measurements the hard triplets are mined. The training uses exclusively these mined triplets of data set *robot*.



(a) Data set *robot*



(b) Data set *longterm*



(c) Data set *scenarios*

Figure 11.5.: Effectiveness of the generalized Blind Twins approach with different data sets.

For the actual evaluation of the trained general Blind Twins all three data sets are employed. This means, after the training with the *robot* data, we evaluate the network on the data sets *robot*, *longterm* and *scenarios*. Here, the measurements of *robot* with the attack in position $E2$ is used, to exclude any overlapping.

As with the evaluation of Blind Twins before, the metric in use is the BDR of the reconciled bit strings.

The results of this evaluation are shown in Fig. 11.5.

The first plot in Fig. 11.5a shows the results for the evaluation with data set *robot*. First of all, the network successfully provides reconciliation for the legitimate CIRs pairs: in $99.47\,\%$ of all cases the reconciliation is effective and delivers a perfect match between the Alice and Bob CIR. Overall, the BDR for the legitimate partners is $0.000\,55$, which is significantly lower than the BDR of simple Blind Twins.

Furthermore, this system also ensures security, since Eve does not obtain a result with a Hamming distance less than 4. The smallest distances for Eve are 4 and 5, with relative frequencies 0.0001 and 0.001, respectively. The majority of Eve's observations have a distance of 12. This is also reflected in the respective BDR of $0.768\,83$.

The remaining two plots of Fig. 11.5 visualize the transferability of the trained network to other physical scenarios. Here, the network trained with *robot* data is applied to the data set *longterm* (Fig. 11.5b) and *scenarios* (Fig. 11.5c). Overall, the results of the *robot* data set are confirmed, indicating a successful transferability of the trained network.

For data set *longterm*, a perfect reconciliation was achieved in $94.88\,\%$ of all legitimate

cases. This is also reflected in the respective BDR of $0.013\,36$. With data set *scenarios* the legitimate partner successful reconcile into the same bit string in $97.99\,\%$ of the cases, resulting in a BDR of $0.011\,58$. Although, these results are not as good as for *robot*, they are still better than comparable state-of-the-art approaches, having BDRs of 0.04 [120].

For both data sets, the result histogram for the eavesdropper is shifted more towards the optimal BDR of 0.5. With data set *longterm*, the majority, $68.06\,\%$, of the eavesdropped CIRs have a Hamming distance of 11. The respective BDR is $0.644\,74$. With data set *scenarios*, the distribution is even more spread out. Here, the largest peak is again at distance 11, with $41.03\,\%$. The BDR is $0.586\,75$. In no case the eavesdropper reconciles into the same sequence as the legitimate partners.

Overall, these results demonstrate that this general *Blind Twins* approach effectively eradicates the drawbacks of the simple Blind Twins approach. This generalized approach can effectively handle very diverse data sets, as shown with the results of data set *robot*. Additionally, a network trained in this manner is transferable to different physical transmission environments. This is demonstrated by the results of data sets *longterm* and *scenarios*. Especially the *longterm* results are noteworthy, since the respective environment is not even cuboid.

As with the simple Blind Twins approach, the results achieved here outperform current approaches like, e.g., [120, 224]. Additionally, these results are achieved without any communication between the legitimate partners.

11.3. Attack Resilience

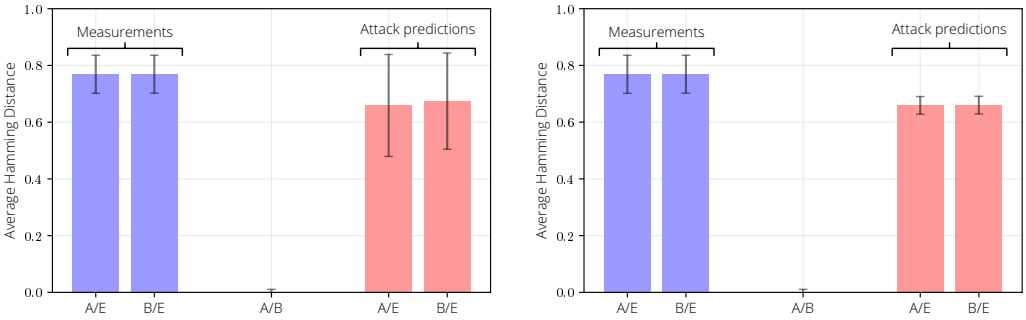
To evaluate the attack resilience of the *Blind Twins* approach we proceed as with the respective CCE evaluation. This means in particular: the legitimate partners apply general Blind Twins as defined in the previous section, while the attacker executes the attack currently evaluated as described before. Subsequently, Eve applies Blind Twins to the obtained attack results. For the final evaluation, we compare the BDR of resulting bit strings.

With the same reasoning as for the CCE approach, we evaluate the channel model as well as the ray tracing-based attacks on data set *robot*, and the machine learning-based one on *scenarios*, *longterm* and *robot*.

The resilience against the "classical" attacks is shown in Fig. 11.6. Since the data set *robot* was used for both attacks, the results for the overheard CIRs (blue bars) and for the legitimate ones (green bars) are equal. The bars of the legitimate partners are actually so close to zero, that they are not visible in the plots — the respective BDR is $0.000\,55$ with a standard deviation of $0.010\,09$. The eavesdropper CIRs after application of general Blind Twins achieve BDRs of $0.768\,69 \pm 0.067\,24$ and $0.768\,97 \pm 0.066\,97$, for h_{AE} and h_{BE} respectively.

The results of the **channel model-based attack** are shown in Fig. 11.6a. After applying this attack, the BDR were slightly lower than for the raw observed CIR. Concretely, BDRs of $0.658\,87 \pm 0.179\,67$ and $0.674\,06 \pm 0.169\,64$ are achieved.

The **ray tracing-based attack** achieved similar results. Again, the predictions of the attack achieve lower results as the measurements, without yielding advantageous results for the attacker. The actual BDRs are $0.659\,28 \pm 0.031\,10$ and $0.660\,09 \pm 0.031\,20$.



(a) Channel Model-based attack (b) Ray Tracing-based attack

Figure 11.6.: Average Hamming Distances for overheard CIRs (blue bars), legitimate CIRs (green bar) and the values predicted by the two "classical" attacks (red bars) after execution of general Blind Twins on data set *robot*.

It is worth noting, that both attacks achieve similar results in this setting. This is assumed to originate from the underlying approach of the attacks: since both attacks predict the main deterministic components of the CIR, they achieve similar results.

Although both attacks lowered the respectiveBDR, they could not create any advantage for the attacker. This demonstrates that Blind Twins provides protection against these attacks.

The results for the **machine learning-based attack** are shown in Fig. 11.7. Overall, the results of the "classical" attacks are confirmed: Blind Twins does effectively protect from this attack.

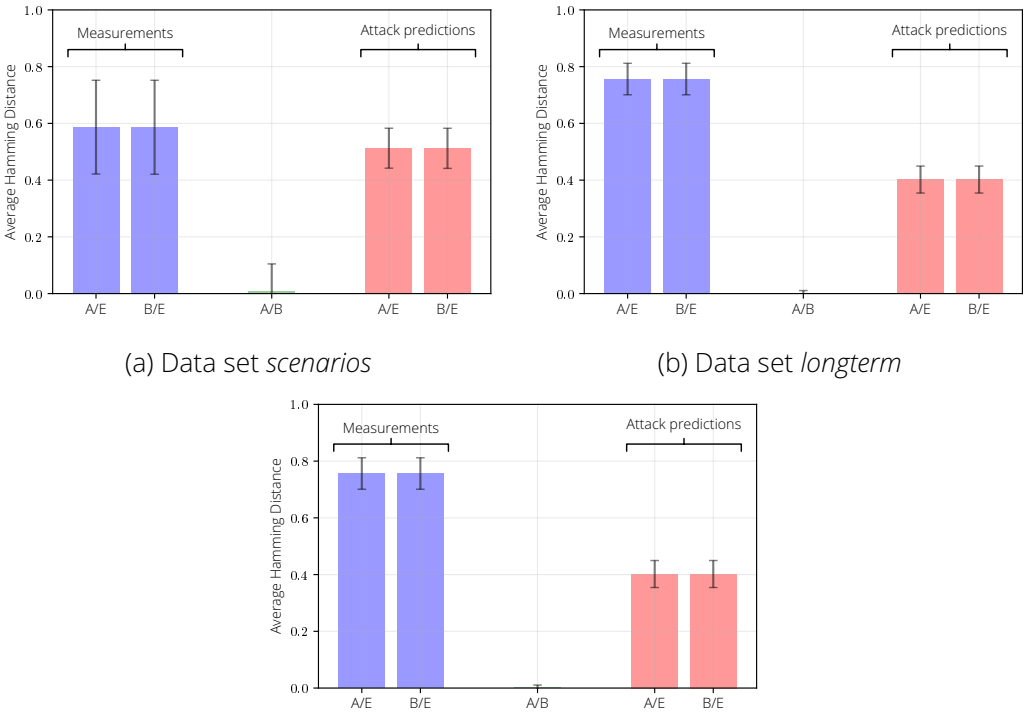
As for the different data sets used in this evaluation: With data set *scenarios*, the legitimate partners achieve a BDR of $0.011\,58 \pm 0.094\,21$. By applying simple eavesdropping the attacker reaches BDRs of $0.586\,74 \pm 0.165\,23$ and $0.586\,37 \pm 0.165\,64$, for h_{AE} and h_{BE} respectively. After application of the machine learning-based inference attack, the respective BDRs were lowered to $0.512\,53 \pm 0.070\,42$ and $0.512\,07 \pm 0.070\,72$.

For data set *longterm*, the legitimate partners reach a BDR of $0.003\,74 \pm 0.051\,55$. The application of the attack lowered the BDRs achieved by the adversary from $0.664\,60 \pm 0.084\,95$ and $0.664\,61 \pm 0.084\,67$ to $0.437\,93 \pm 0.012\,62$ and $0.437\,94 \pm 0.012\,40$.

Finally, with data set *robot*, the legitimate partners achieve a BDR of $0.000\,55 \pm 0.010\,09$. Using the machine learning attack Eve decreased her BDRs from $0.756\,33 \pm 0.055\,51$ and $0.756\,33 \pm 0.055\,51$ to $0.402\,00 \pm 0.047\,62$ and $0.402\,00 \pm 0.047\,62$.

As with the "classical" attacks, the Blind Twins approach completely thwarts the effectiveness of the machine learning-based attack. Nevertheless, the former effectiveness of this attack is still hinted within these results. This attack could reduce the BDRs stronger than the other approaches, which indicates that the respective predictions are closer to the legitimate CIR. Nevertheless, considering the actually achieved BDRs, this attack is still effectively prevented by Blind Twins. This is also highlighted by the fact, that in no case the intermediate key sequence could be inferred correctly.

In summary, these results clearly show that Blind Twins provide effective protection against the described attacks. The best attacker outcomes are achieved for the machine learning attack with BDRs around 0.4. These poor results for the attacker show that the



(a) Data set *scenarios*

(b) Data set *longterm*

(c) Data set *robot*

Figure 11.7.: Average Hamming Distances for overheard CIRs (blue bars), legitimate CIRs (green bar) and the values predicted by the Machine Learning-based attack (red bars) after execution of general Blind Twins on data sets *scenarios*, *longterm* and *robot*.

method assures effective protection against the attacks.

Furthermore, the BDRs obtained confirm the strong results regarding the general effectiveness as well as the transferability of Blind Twins. The plots show that Blind Twins effectively generates matching bitstrings for Alice and Bob. These results are achieved for *all* datasets, although training was only performed on the *robot* data set. Finally, we highlight, that all these results were achieved without any communication between the legitimate partners.

12. Resulting Pipelines

To summarize the proposed fixes, we present their practical application in the context of CRKG. For this we show the proposed schemes in the respective processing pipelines. This is to be understood with reference to Fig. 3.1, which shows the processing chain of the current state of the art. With reference to this figure, we illustrate the proposed changes.

12.1. Preprocessing-based Pipeline

The first proposed comprehensive solution incorporates the general CIR processing and the CCE preprocessing steps. This combined solution is depicted in Fig. 12.1.

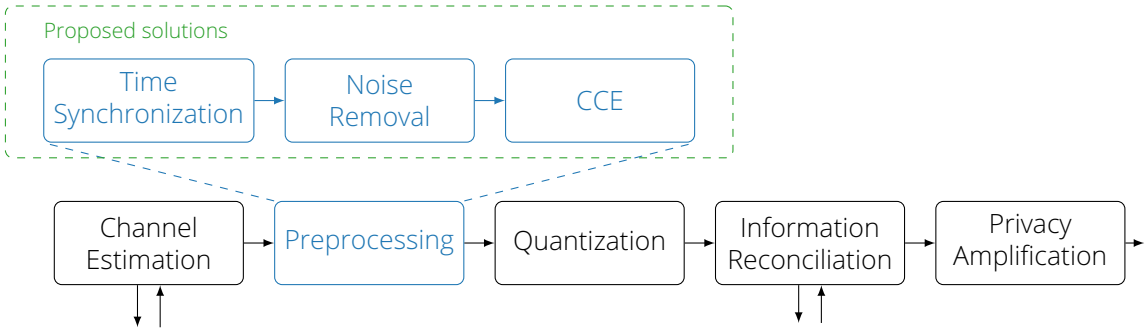


Figure 12.1.: CRKG processing steps at the legitimate nodes A and B following current state of the art.

This solution consists of the successive execution of the single steps proposed in this work. These are specifically:

1. Time synchronization
   This step facilitates the synchronization within the single CIR measurements based on 1D Gaussian filters as proposed in Section 7.1. The result of this preprocessing are CIRs measurements at Alice and Bob with no or minimal time shift in between them.

2. Noise removal
   Based on the time anchor defined by the time synchronization, the noise surrounding the actual, information-bearing part of the CIR is removed, following the procedure proposed in Section 7.2. This step results in CIRs with a fixed length of 64.

3. Channel Characteristics Estimation
   In the final step of preprocessing, the CCE is performed as described in Section 10.1. Thereby, the static predictable parts of the CIR are removed.

Subsequent to this preprocessing is the quantization step. Since CCE delivers vectors with a stable mean, the quantization can be a straight forward threshold based quantization. The threshold can be chosen as zero or as an average over the vector, e.g., the mean or the median. Alternatively, a guard band-based quantizer could be employed — but this re-introduces communication into the pipeline.

In terms of the original theoretic CRKG approach as presented in Fig. 2.4 and with the purpose and design of CCE in mind, these preparatory data processing steps realize an effective *Advantage Distillation*.

Following the design decisions from Chapter 6 these sub-steps operate completely without communication between the legitimate partners. Additionally, they do not facilitate buffering or statistics relying on global knowledge or large sample sizes. Nevertheless, this approach effectively thwarts the presented attacks and, thereby, provides secure key material for the subsequent key derivation.

The implementation of the single steps is efficiently possible, allowing for fast execution of this pipeline. Using an ARM-based embedded device which facilitates a Broadcom BCM2837 System-on-Chip, the execution of the preprocessing took on average $0.6576\,\mathrm{ms}$ with a standard deviation of $0.013\,67\,\mathrm{ms}$. Considering the data sets sampling time of $200\,\mathrm{ms}$ and $240\,\mathrm{ms}$, this pipeline inflicts no additional delays and can be executed in real time of the measurement acquisition.

The final result of this preprocessing is a $64\,\mathrm{bit}$ vector for each acquired CIR measurement. It is worth noting that this $64\,\mathrm{bit}/\mathrm{cu}$ is not the final KGR of this pipeline, since the subsequent steps of quantization, information reconciliation and privacy amplification are still to be performed. Depending on the respective implementations used, additional delays, e.g., during IR, or reductions of the number of bits during PA might occur.

12.2. Blind Twins-based Pipeline

Additionally, we propose a pipeline which is based on the general Blind Twins approach. This solution is visualized in Fig. 12.2.

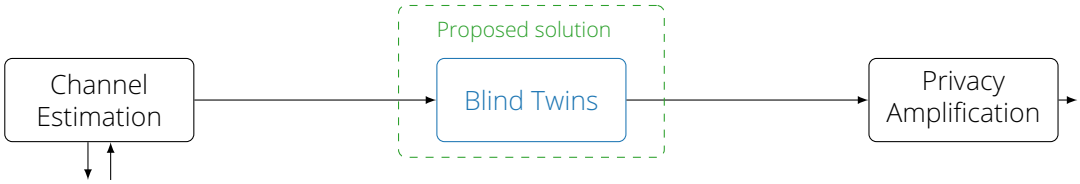


Figure 12.2.: CRKG processing steps at the legitimate nodes A and B following current state of the art.

The Blind Twins based solution incorporates the single processing steps of CRKG into a single solution. Blind Twins handles all the necessary steps including CIR preprocessing, i.e., time synchronization and noise removal, quantization as well as information reconciliation. This is achieved without requiring communication between the legitimate partners at all. Thereby, the implicit requirement of an authenticated communication channel between Alice and Bob is removed as well.

Although the overall architecture of CRKG becomes significantly less complex with this pipeline, the Blind Twins solution itself re-introduces complexity. Nevertheless, due to the quite simple architecture of the underlying CNN, this solution is still performant enough for real time application. Concretely, the execution of Blind Twins on a single CIR takes $5.394\,\mathrm{ms}$, with a standard deviation of $0.6136\,\mathrm{ms}$. It is worth noting, that although this $\approx 8\times$ the time of the preprocessing based pipeline, this is no fair comparison, since this result is the time for the *complete* pipeline, including preprocessing, quantization and IR. Again, considering the acquisition time of the single CIRs, this step can be executed in real time.

The final result of this solution is a $16\,\mathrm{bit}$ vector for each CIR measurement. Hence, before execution of Privacy Amplification, the KGR is at $16\,\mathrm{bit}/\mathrm{cu}$.

Part VI.

Summary and Outlook

13. Achieved Results

In this work we investigated the basic question, how Channel Reciprocity-based Key Generation (CRKG) based on Channel Impulse Response (CIR) measurements can be made more practical. The motivation for this question is twofold: on the one hand there is the ever-increasing number of connected and communicating devices, which require secure and efficient communication. On the other hand, there are recent hardware developments providing additional and powerful capabilities.

For the first point, we have seen increasing numbers for connected devices for recent decades, with even further growing numbers prospected for upcoming years. The communication between those devices needs to be secured properly to avoid and mitigate risks associated with this communication as well as with the physical processes connected with this communication.

The second point puts increasing system capabilities into the hands of system designers allowing for more complex and efficient solution approaches. In our case, we facilitate key generation protocols building on CIRs instead of the more classical Received Signal Strength Indicator (RSSI) as source of common randomness. This change is supported by the incorporation of Ultra Wideband (UWB) transceivers into consumer grade hardware. Such wireless systems allow for the acquisition of CIRs with high temporal resolution, providing access to the rich multipath information within.

In combination, the recent technological advances can be facilitated to effectively meet the demand for secure connections between an increasing number of devices.

To approach the question about practical realization of UWB CIR-based CRKG, three subquestions were investigated: the fundamental challenge to show practical relevance of all analyses is to provide representative data sets for the evaluation. Based on such data sets, the security of the initial key material can be assessed, especially with focus on the attacker side and the respective capabilities. Finally, based on the potential attack possibilities of an adversary and their chances of success, a secure system can be designed that prevents exactly these possibilities by its very design.

For the first task, we recorded four different sets of real world UWB Channel Impulse Response measurements in realistic scenarios for CRKG. With focus on different partial aspects of the CRKG topic, the first three data sets were recorded. Based on the insights generated with these, an autonomous measurement setup was designed and built, combining the different advantages of the former data sets. With this setup, the fourth data set

was recorded, which was also released under open access.

The initial analysis of the raw CIR measurements confirmed the overall approach feasibility: in all data sets the legitimate communication partners have an "advantage" over the eavesdropper in the sense of Wyner's wiretap channel. This advantage is expressed in terms of the cross correlation between the respective CIRs. In all data sets the legitimate partners Alice and Bob record CIR measurements with average cross correlations >0.93. The eavesdropper Eve, on the other hand, observes CIRs with average correlation to the legitimate one of 0.79 to 0.82. Hence, Alice and Bob do have an advantage over Eve which can be exploited to derive a common secret key.

The initial analysis of the recorded CIR measurement data furthermore highlighted that additional steps are necessary to prepare the raw measurement data for CRKG processing. Here, it is necessary to consider the time synchronization of the reciprocal measurements, which cannot be solved by straight forward approaches like maximum or leading edge-based synchronization.

To investigate whether an attacker can overcome the advantage of the legitimate communication partners, we propose three different inference attacks against the raw CIR measurements.

The first attack facilitates a deterministic channel model, concretely the Kunisch-Pamp UWB channel model. By using sample CIR measurements, the attacker adapts the model parameters to the current physical setting. This model can than in turn be used to predict CIRs of the legitimate communication partners based on their current position. The second attack operates in similar vein, but instead of building on a deterministic channel model, ray tracing is used to predict the legitimate CIRs. Through the usage of ray tracing the attacker model can be simplified as no explicit adaption to the current physical environment is necessary. The third and final attack facilitates machine learning to directly infer the legitimate CIR from the overheard ones at Eve's position.

The results of the first two attacks provide two insights: first, these deterministic approaches are capable of predicting CIRs with high accuracy, and second, both approaches do not generalize well. The former part is particularly evident in the channel model-based attack — by employing the *individual optimization*, which is based on CIR samples from the current positions, this attack can achieve average cross correlations up to 0.941. Thus, with this attack, attackers can infer CIRs whose cross correlation to the legitimate CIRs is as high as for the legitimate ones among each other. This is contrasted by the generalization capability of these attack approaches. Especially with the data set *robot*, which incorporates many different settings and high variance, both approaches achieve correlations with are only slightly better than those of mere eavesdropping. Concretely, eavesdropping results in CIRs with average cross correlation of 0.827; the attacks predict CIRs with correlations of 0.848 and 0.835, for the channel model and ray tracing-based attack respectively.

The shortcomings of these two attacks are eradicated by the machine learning-based attack, which achieves high accuracy as well as general applicability. The design of this attack, i.e., the used architecture of the underlying neural network, and the employed training procedure inherently makes this approach generally applicable. Nevertheless, this attack still predicts CIRs with high accuracy: for all analysed data sets average cross correlations >0.9 were achieved. For data set *longterm*, this attack yielded average cross correlations of 0.997, thus a higher correlation than the legitimate CIRs to each other.

The performance of these attacks poses a significant risk to the respective Physical Layer

Security (PLS) primitives. Even the results of the individual optimized approach need to be considered — in cases where terminal positions are known in advance or when the terminals move only occasionally, these approaches are applicable in practice. An even greater threat arises from the machine learning-based approach and its respective, general performance. To further investigate this threat in the context of CRKG we analysed the respective Bit Disagreement Rate (BDR) after quantization. This analysis showed that the adversary's predictions have *lower* BDRs than the legitimate partners in up to $83.5\,\%$ of all cases. Since the subsequent CRKG processing is deterministic, this means that in $83.5\,\%$ of all cases the attacker can derive the same key as the legitimate communication partners. These results lead to the conclusion, that the unprocessed, raw CIR measurements should not be used directly in the respective security primitives.

This insight begs the question of how the measurements and the contained entropy can still be used for key generation in a secure way. Hence, we propose two different ways, how such a processing can be designed and realized to provide a key exchange which is efficient and still resilient against the presented attacks. These two processing approaches are, on the one hand, a preprocessing based one, which incorporates different "classical" processing steps, and, on the other hand, a machine learning-based one, which presents an "all-in-one" solution.

In a preparatory step, we defined design guidelines for this based on the requirements of the system to be designed, which are to be implemented for all solution approaches. This includes for example the avoidance of buffering, of global statistics and especially of excessive communication to prevent biases, latencies and additional energy consumption.

Proceeding from there, we designed the processing steps of the preprocessing based solution. The first step of pre-processing is to synchronize the respective CIR observations at Alice and Bob in time. To solve this, we proposed a synchronization approach based on one-dimensional Gaussian filters. Applying this solution lead to an $18\,\%$ improvement of the respective BDR, while being completely interaction-free. This is followed by noise removal, which essentially discards surplus measurements within the single CIRs which do not hold reciprocal information. Finally, to ensure the resilience against the presented attacks, we proposed the *Channel Characteristics Estimation (CCE)* technique. This approach is designed to remove those parts of the single CIRs which are predictable and upon which the presented attacks thrive. By applying a moving average over the single data point within the last n CIR observations the static, deterministic parts could be estimated and removed. This approach not only increased the mutual information and yielded results closer to the uniform distribution, but also provided protection against the attacks: after applying Channel Characteristics Estimation (CCE), the attacker consistently has a BDR of ≈ 0.5 to the legitimate bits, the worst possible result for her, regardless of whether she merely eavesdrops or applies one of the presented attacks. This is achieved without interaction between the legitimate partners. The final outcome of the preprocessing solution is a $64\,\mathrm{bit}$ vector for each observed CIR, which than can be passed to Information Reconciliation (IR).

As an alternative, we propose a second solution, which incorporates machine learning capabilities, and thereby provides IR as well. The core idea is to train a neural network in a discriminative manner so that it learns to use exactly those properties of the CIRs that define reciprocity of legitimate measurements or non-reciprocity of attacker measurements. To achieve a generally applicable solution, we combined convolutional neural networks with a training based on triplet loss, hard triplet mining and a CCE-like history. Thus, after

one-time training on a single data set, this network can be deployed to the respective devices and used in different settings. Due to the shift invariance of the Convolutional Neural Network (CNN) and its feature extraction capabilities, this solution inherently handles time synchronization and noise removal. Due to the triplet loss-based training, IR is provided: by learning the discriminatory features of the reciprocity, the network maps similar, reciprocal CIRs to equal bit strings, whereas non-reciprocal CIRs are mapped to dissimilar bit strings. This is confirmed by the achieved results: after training with the *robot* data set and evaluation on all data sets, the BDR of the legitimate partners is <0.012 in all cases, whereas the attacker using eavesdropping or the presented attack has BDRs in the range 0.402 to 0.756. The final outcome of this solution is a $16\,\mathrm{bit}$ vector for each observed CIR, which is already reconciled.

It is worth noting, that all the proposed solutions are completely interaction-free, i.e., no communication between the legitimate partners is required. Further, both overall solution pipelines can be implemented efficiently allowing for a real time execution during the acquisition of the CIR measurements. Thereby, with a Key Generation Rate (KGR) of at least $16\,\mathrm{bit/cu}$ (for the machine learning-based solution), or in the respective settings $38.4\,\mathrm{bit/s}$, the presented approaches are more efficient than the current state of the art solutions. Finally, both presented solutions are secure against all the presented attacks.

14. Outlook

The analyses and approaches presented here present a comprehensive view of the situation of CIR-based CRKG. However, the studies conducted and their results certainly raise further research questions that should be addressed in future investigations. In the following, we outline the most important of these issues.

With regard to the analyses of the acquired measurement data, the question inevitably arises as to how the information-theoretical view underlying most of the analyses here can be better implemented. Specifically, this would translate into analysing the raw CIR observations of Alice, Bob, and Eve in terms of their respective Mutual Information (MI). Current MI estimators have not yielded consistent results in our analyses to date.

Additionally, the measurement data could be used to generate insight into the trade-offs governing the different processing steps of the CRKG pipeline. Specific trade-offs of interest here include the effect of different quantization algorithms on entropy, randomness, and resulting BDR of the quantization result, or the influence of IR algorithm in use on information leakage to the attacker contrasted by its effectiveness depending on the BDR of the input data.

Likewise, current developments in adjacent areas and its impact on CRKG or CIRs, respectively, should be considered. For instance, reconfigurable surfaces as proposed in [178] can provide additional entropy for key generation if terminal movement alone cannot ensure this.

In the domain of attacks, the results obtained have so far merely outlined the possibilities of the respective attack vectors. Here, a refinement of the individual attacks as well as the combination of the respective methods should be investigated with regard to the resulting security implications. For the channel model-based attack, further possibilities of parameter adaptation should be investigated in order to achieve a better generalization performance and thus to achieve the prediction performance of the individual optimization in the general setting. For this purpose, alternatives to Bayesian optimization for fitting per se could be investigated, for example, or continuous improvement and adaptation of the parameters with respect to the actual situations during the actual attack. Regarding the ray tracing-based attack, the investigation of better transmission space modelling, which e.g., takes into account the transmission properties of the respective materials, is suggested. Likewise, alternative ray tracing tools, e.g., the developments from [64], should be compared to achieve the best results. The machine learning-based attack provides a variety

of options for improvement: For example, more complex architectures can be used and trained with more and diverse data to improve the prediction accuracy and/or generalization of the attack. Similarly, the generalization techniques of Blind Twins could be used to improve this attack, such as the use of history or the appropriate data augmentation. Finally, fundamentally different approaches from this area, such as recurrent networks, can be investigated for their suitability as attack tools.

A promising approach in this area is to combine the individual attacks. In the simplest case, all three attacks can be executed and evaluated in parallel so that the best attack is used in each situation. In terms of a true combination, the "classical" attacks can also be used for a "rough" prediction, which is then improved using Machine Learning (ML) methods. Thus, denoising techniques like autoencoders could be used to improve the attack predictions.

Equally promising is the approach of collaborating attacks. Given the success of a single adversary ML attack, the combination of eavesdropped information from different positions might yield even stronger attacks.

To be able to guarantee the security of the respective CRKG methods, such improved attacks have to be investigated and the resilience of the CRKG methods against them has to be shown.

The success of the machine learning approaches raises also questions about the channel modelling: as the ML-based attacks could infer much more information about the CIR properties than the channel model it seems reasonable to assume that the channel model is insufficient for such security analyses. Considering the corresponding expertise from this domain, an improved channel model could possibly be developed, which on one hand would enable better attacks, but on the other hand would also enable a more precise security analysis and thus the design of more secure CRKG systems.

The area of solutions presented also offers some points that should be further investigated to find any improvements and prevent any problems.

For CCE, the structure of the recorded data raises the question whether the moving average is the best solution. To account for the physical movements of the terminals, the exponentially weighed moving average or even a one dimensional Gaussian filter along the time axis might deliver good results.

In the domain of Blind Twins, the most important connecting point is the verification of the results, especially of the generalization capabilities, by further measurements from diversified setups and scenarios. Such a validation based on additional and new measurements not only increases confidence in the capabilities and performance of Blind Twins, but also provides additional training material to train the model in the generalizing manner. Further and similar to the enhancements of the ML-based attack, the Blind Twins approach could also be shifted to other core architectures. Considering the incorporation of a CCE like history, the application of recurrent architectures seems possible here as well.

Last but not least, an effective approach for Privacy Amplification needs to be implemented. A real world implementation inevitably needs a policy by what proportion the key information must be reduced so that Eve can no longer derive any information about the key material. This can be done, for example, by a suitable online entropy estimator quantifying the respective information content at Alice and Bob and deriving a guideline value from it. Alternatively, reference can be made here to the MI estimation mentioned above — a corresponding precise analysis of the raw measured CIR values MI could provide limits

and boundaries for Eve's information gain, from which corresponding practically oriented "rules of thumb" can arise, always in consideration of practical security.

By addressing these points and analysing the corresponding questions, the working solutions presented here can be further optimized, their practicality further enhanced, and their adaptation supported.

Appendix

A. Additional Evaluation Results

A.1. Complete Parameter Sets of Individual Optimization

Table A.1.: Complete individually optimized parameters of the Kunisch-Pamp channel model.

r	K	d_0	α	G_{MP}	$G_{MP,LOS}$	γ
1	$\{172, 123, 116, 158, 178, 214, 220, 228, 270\}$	5.33 m	2.02	−10.02 dB	−13.69 dB	10.91 ns
2	$\{172, 123, 165, 179, 130, 164, 166, 174, 178, 186, 228, 270\}$	1.07 m	2.02	−11.14 dB	−2.80 dB	9.25 ns
3	$\{172, 179, 221, 122, 124, 130, 166, 174, 186, 214, 220, 228, 270\}$	1.82 m	2.02	−18.63 dB	−12.31 dB	11.93 ns
4	$\{172, 123, 171, 221, 116, 122, 124, 130, 166, 170, 174, 178, 180, 214, 220, 222\}$	2.56 m	2.98	−10.64 dB	−3.12 dB	12.43 ns
5	$\{172, 173, 74, 116, 124, 130, 158, 164, 178, 180, 186, 214, 220, 222, 228, 270\}$	3.29 m	2.70	−14.37 dB	−7.60 dB	12.39 ns
6	$\{172, 123, 171, 221, 74, 116, 122, 170, 178, 180, 186, 220, 222\}$	1.50 m	2.98	−12.33 dB	−14.28 dB	12.79 ns
7	$\{172, 165, 179, 221, 74, 116, 130, 166, 174, 180, 186, 214, 220, 270\}$	1.82 m	2.67	−13.57 dB	−14.27 dB	12.99 ns
8	$\{172, 123, 173, 179, 116, 122, 124, 166, 174, 180, 186, 270\}$	3.83 m	2.86	−10.01 dB	−13.90 dB	13.00 ns
9	$\{172, 165, 171, 221, 74, 116, 124, 130, 158, 178, 180, 186, 214, 222, 228, 270\}$	1.53 m	2.88	−11.11 dB	−12.52 dB	12.87 ns
10	$\{172, 171, 173, 221, 116, 122, 124, 130, 164, 166, 170, 174, 178, 222, 228\}$	2.31 m	2.20	−13.17 dB	−7.86 dB	12.84 ns
11	$\{172, 123, 171, 173, 179, 221, 166, 170, 178, 180, 186, 220, 228, 270\}$	0.61 m	2.00	−17.81 dB	−13.62 dB	12.94 ns
12	$\{172, 123, 165, 171, 221, 74, 122, 124, 164, 166, 174, 178, 186, 270\}$	4.03 m	2.05	−12.49 dB	−13.21 dB	12.54 ns
13	$\{172, 165, 171, 173, 74, 116, 122, 124, 130, 170, 178, 180, 214\}$	1.93 m	2.06	−10.59 dB	−1.07 dB	9.39 ns
14	$\{172, 123, 165, 171, 173, 179, 221, 122, 130, 158, 164, 170, 178, 180, 186, 222, 228, 270\}$	3.14 m	2.32	−10.29 dB	−6.37 dB	12.86 ns
15	$\{172, 123, 165, 221, 122, 124, 174, 186, 214, 222\}$	3.10 m	2.74	−10.03 dB	−13.27 dB	10.38 ns
16	$\{172, 123, 165, 171, 221, 116, 130, 158, 164, 166, 174, 178, 180, 220, 222, 270\}$	5.18 m	2.02	−10.82 dB	−1.41 dB	13.00 ns
17	$\{172, 171, 179, 221, 116, 124, 130, 158, 164, 166, 170, 186, 220, 222\}$	4.50 m	2.52	−13.21 dB	−6.05 dB	11.97 ns

Table A.2.: Complete individually optimized parameters of the Kunisch-Pamp channel model (with noise).

r	K	d_0	α	G_{MP}	$G_{MP,LOS}$	γ	V_N
1	$\{172, 123, 165, 173, 179, 221, 74, 116, 122, 130, 158, 164, 166, 170, 174, 178, 180, 186, 214, 228, 270\}$	$5.28\,\mathrm{m}$	2.55	$-10.02\,\mathrm{dB}$	$-9.98\,\mathrm{dB}$	$12.17\,\mathrm{ns}$	$14.79\,\mathrm{dB}$
2	$\{172, 123, 165, 179, 122, 158, 166, 170, 174, 180, 186, 214, 270\}$	$1.15\,\mathrm{m}$	2.03	$-11.90\,\mathrm{dB}$	$-3.21\,\mathrm{dB}$	$9.26\,\mathrm{ns}$	$18.60\,\mathrm{dB}$
3	$\{172, 173, 221, 116, 122, 124, 130, 164, 174, 186, 222, 228, 270\}$	$2.18\,\mathrm{m}$	2.32	$-19.65\,\mathrm{dB}$	$-10.78\,\mathrm{dB}$	$13.00\,\mathrm{ns}$	$17.00\,\mathrm{dB}$
4	$\{172, 123, 165, 173, 130, 158, 164, 174, 180, 220\}$	$1.46\,\mathrm{m}$	2.04	$-16.34\,\mathrm{dB}$	$-11.64\,\mathrm{dB}$	$10.09\,\mathrm{ns}$	$16.13\,\mathrm{dB}$
5	$\{172, 173, 122, 124, 130, 158, 164, 166, 170, 174, 178, 186, 222, 228, 270\}$	$2.01\,\mathrm{m}$	2.96	$-12.18\,\mathrm{dB}$	$-12.74\,\mathrm{dB}$	$12.33\,\mathrm{ns}$	$12.45\,\mathrm{dB}$
6	$\{172, 116, 122, 164, 170, 178, 220\}$	$3.89\,\mathrm{m}$	2.86	$-12.51\,\mathrm{dB}$	$-14.98\,\mathrm{dB}$	$9.06\,\mathrm{ns}$	$13.99\,\mathrm{dB}$
7	$\{172, 123, 171, 179, 116, 122, 124, 158, 164, 178, 186, 214, 220\}$	$2.75\,\mathrm{m}$	2.92	$-14.03\,\mathrm{dB}$	$-9.85\,\mathrm{dB}$	$11.50\,\mathrm{ns}$	$14.38\,\mathrm{dB}$
8	$\{172, 123, 173, 179, 221, 116, 122, 158, 166, 178, 180, 220, 222, 270\}$	$2.60\,\mathrm{m}$	2.77	$-12.30\,\mathrm{dB}$	$-10.96\,\mathrm{dB}$	$12.43\,\mathrm{ns}$	$19.99\,\mathrm{dB}$
9	$\{172, 123, 173, 179, 221, 74, 116, 122, 130, 158, 174, 186, 214, 220, 222, 228\}$	$2.37\,\mathrm{m}$	2.65	$-19.10\,\mathrm{dB}$	$-10.41\,\mathrm{dB}$	$12.86\,\mathrm{ns}$	$12.62\,\mathrm{dB}$
10	$\{172, 165, 122, 130, 164, 174, 214, 222, 270\}$	$1.87\,\mathrm{m}$	2.85	$-10.28\,\mathrm{dB}$	$-13.75\,\mathrm{dB}$	$12.61\,\mathrm{ns}$	$18.42\,\mathrm{dB}$
11	$\{172, 165, 173, 179, 221, 74, 116, 122, 164, 166, 170, 180, 186, 214, 270\}$	$1.01\,\mathrm{m}$	2.06	$-15.75\,\mathrm{dB}$	$-8.35\,\mathrm{dB}$	$10.62\,\mathrm{ns}$	$18.50\,\mathrm{dB}$
12	$\{172, 123, 165, 173, 221, 116, 122, 130, 158, 174, 178, 180, 186, 222, 228\}$	$5.88\,\mathrm{m}$	2.20	$-20.28\,\mathrm{dB}$	$-5.69\,\mathrm{dB}$	$13.00\,\mathrm{ns}$	$14.04\,\mathrm{dB}$
13	$\{172, 165, 171, 173, 74, 116, 122, 124, 130, 158, 170, 174, 180, 214, 228\}$	$2.08\,\mathrm{m}$	2.74	$-10.03\,\mathrm{dB}$	$-5.93\,\mathrm{dB}$	$9.02\,\mathrm{ns}$	$17.94\,\mathrm{dB}$
14	$\{172, 165, 171, 173, 179, 221, 74, 116, 122, 158, 164, 170, 174, 186, 220, 228\}$	$3.15\,\mathrm{m}$	2.93	$-15.29\,\mathrm{dB}$	$-13.90\,\mathrm{dB}$	$13.00\,\mathrm{ns}$	$15.97\,\mathrm{dB}$
15	$\{172, 171, 221, 116, 122, 130, 158, 164, 178, 214, 220, 222, 228\}$	$1.99\,\mathrm{m}$	2.12	$-10.68\,\mathrm{dB}$	$-5.27\,\mathrm{dB}$	$10.57\,\mathrm{ns}$	$17.67\,\mathrm{dB}$
16	$\{172, 123, 116, 158, 164, 166, 170, 174, 178, 180, 186, 214, 220, 228, 270\}$	$2.49\,\mathrm{m}$	2.08	$-15.61\,\mathrm{dB}$	$-4.45\,\mathrm{dB}$	$13.00\,\mathrm{ns}$	$18.66\,\mathrm{dB}$
17	$\{172, 123, 173, 122, 158, 166, 170, 174, 178, 180, 186, 214, 220, 222\}$	$4.05\,\mathrm{m}$	3.00	$-12.27\,\mathrm{dB}$	$-10.13\,\mathrm{dB}$	$11.28\,\mathrm{ns}$	$15.65\,\mathrm{dB}$

A.2. Cross Correlation Heatmaps

A.2.1. Attacker Position $E1$

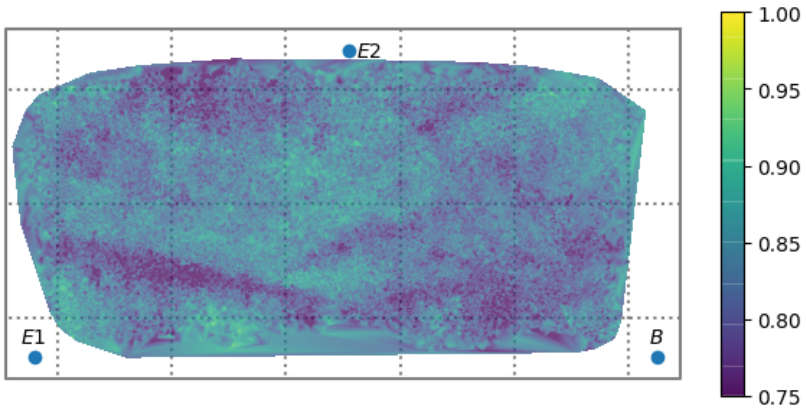


Figure A.1.: Cross correlation in dependence of terminal positions between the legitimate partners CIRs and eavesdropped CIRs at $E1$ superimposed on the room geometry.

A.2.2. Bob at Position $A2$

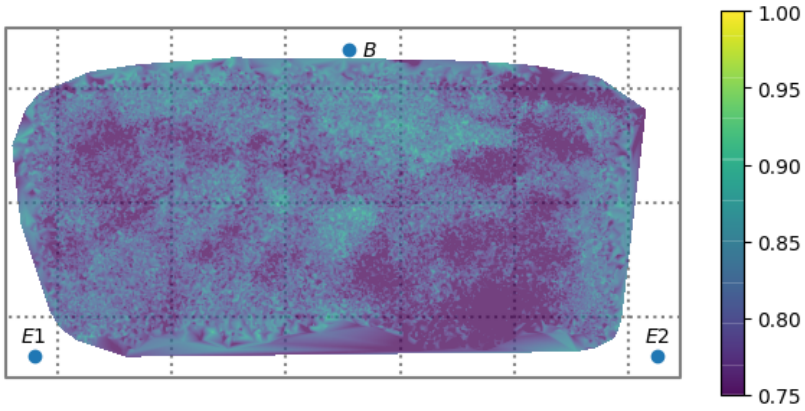


Figure A.2.: Cross correlation in dependence of terminal positions between the legitimate partners CIRs and eavesdropped CIRs at $E2$ superimposed on the room geometry.

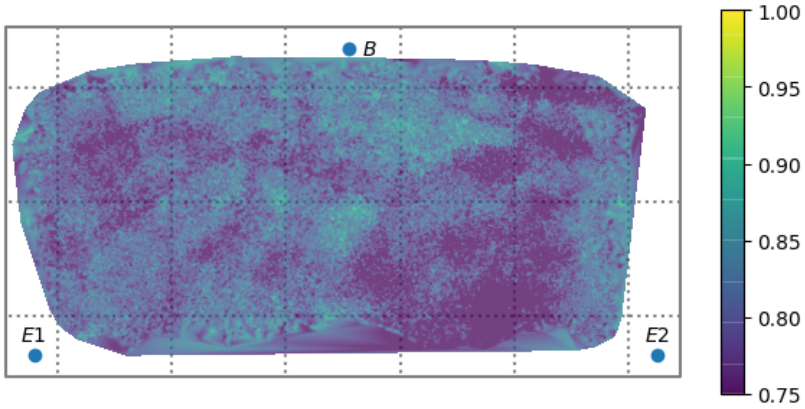
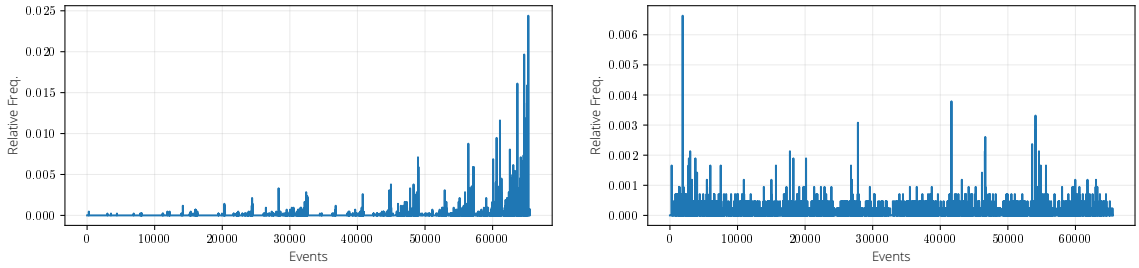


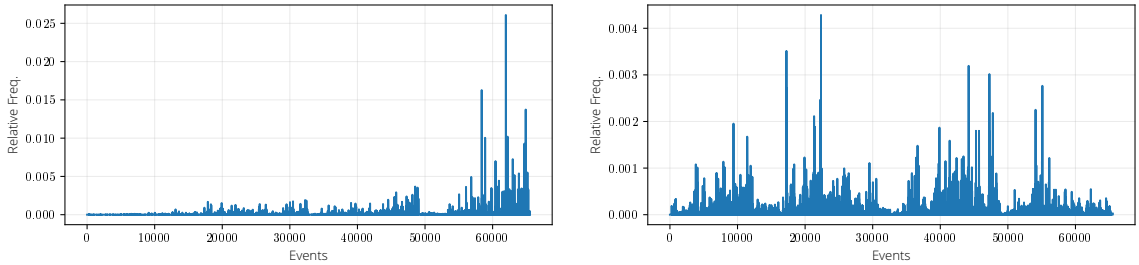
Figure A.3.: Cross correlation in dependence of terminal positions between the legitimate partners CIRs and eavesdropped CIRs at $E1$ superimposed on the room geometry.

A.3. Uniformity of Bit Strings after CCE



(a) Without CCE (b) With CCE

Figure A.4.: Histogram of the 2^{16} possible quantization outcomes for data set *scenarios*.



(a) Without CCE (b) With CCE

Figure A.5.: Histogram of the 2^{16} possible quantization outcomes for data set *longterm*.

A.4. Autocorrelation of Bit Strings after CCE

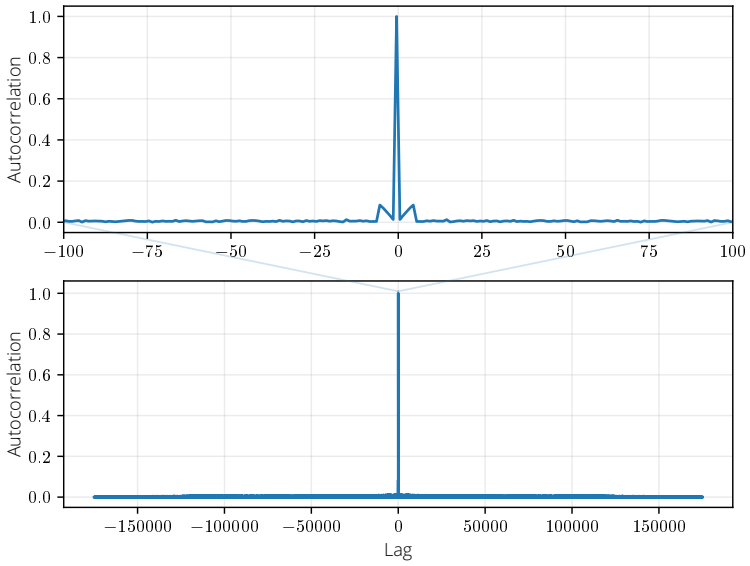


Figure A.6.: Autocorrelation of the quantized results of data set *longterm* after the application of CCE.

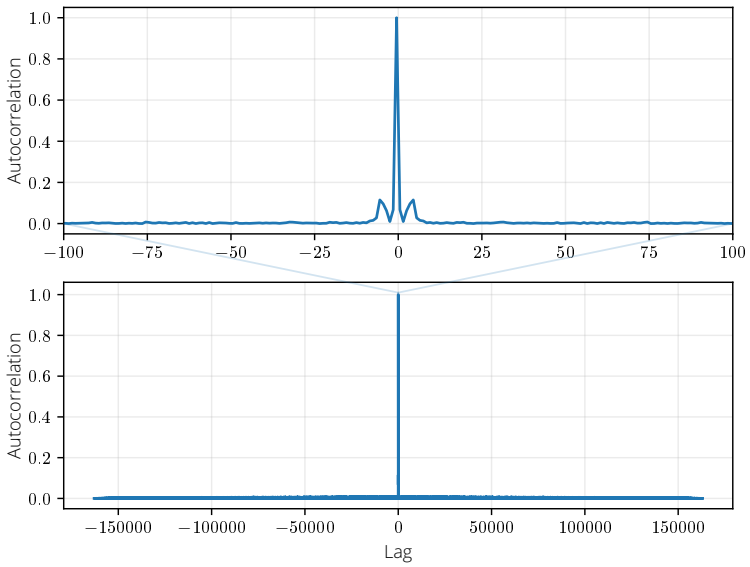


Figure A.7.: Autocorrelation of the quantized results of data set *robot* after the application of CCE.

Bibliography

[1] R. Ahlswede and I. Csiszar. "Common Randomness in Information Theory and Cryptography. I. Secret Sharing". In: *IEEE Transactions on Information Theory* 39.4 (July 1993), pp. 1121–1132. DOI: 10.1109/18.243431.

[2] M. Alhasanat, S. Althunibat, K. A. Darabkh, A. Alhasanat, and M. Alsafasfeh. "A Physical-Layer Key Distribution Mechanism for IoT Networks". In: *Mobile Networks and Applications* 25.1 (Feb. 2020), pp. 173–178. DOI: 10.1007/s11036-019-01219-5.

[3] S. T. Ali, V. Sivaraman, and D. Ostry. "Eliminating Reconciliation Cost in Secret Key Generation for Body-Worn Health Monitoring Devices". In: *IEEE Transactions on Mobile Computing* 13.12 (Dec. 2014), pp. 2763–2776. DOI: 10.1109/TMC.2013.71.

[4] S. T. Ali, V. Sivaraman, and D. Ostry. "Secret Key Generation Rate vs. Reconciliation Cost Using Wireless Channel Characteristics in Body Area Networks". In: *2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*. Dec. 2010, pp. 644–650. DOI: 10.1109/EUC.2010.103.

[5] O. Alp Topal, Z. Liang, G. Ascheid, G. Dartmann, and G. Karabulut Kurt. "Using of Wavelets for Secret Key Generation: A Measurement Based Study". In: *2018 26th Telecommunications Forum (TELFOR)*. Nov. 2018, pp. 1–4. DOI: 10.1109/âĎąFOR.2018.8611930.

[6] E. Alpaydin. *Introduction to Machine Learning*. MIT press, 2020.

[7] F. Alsubaei, A. Abuhussein, and S. Shiva. "Security and Privacy in the Internet of Medical Things: Taxonomy and Risk Assessment". In: *2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops)*. IEEE, 2017, pp. 112–120.

[8] A. Ambekar and H. D. Schotten. "Enhancing Channel Reciprocity for Effective Key Management in Wireless Ad-Hoc Networks". In: *2014 IEEE 79th Vehicular Technology Conference (VTC Spring)*. May 2014, pp. 1–5. DOI: 10.1109/VTCSpring.2014.7022913.

[9] N. Amiot, M. Laaraiedh, and B. Uguen. "PyLayers: An Open Source Dynamic Simulator for Indoor Propagation and Localization". In: *2013 IEEE International Conference on Communications Workshops (ICC)*. June 2013, pp. 84–88. DOI: 10.1109/ICCW.2013.6649206.

[10] N. Annavarapu, M. Catuneanu, P. Walther, M. Razavi, E. Franz, T. Strufe, and K. Jamshidi. “Numerical Modeling of Dual Pump Amplifiers in Standard Silicon-on-Insulator Platform for Random Number Generation”. In: *Frontiers in Optics*. Optical Society of America, 2020, JTh4B–16.

[11] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, and M. Kallitsis. “Understanding the Mirai Botnet”. In: *26th USENIX Security Symposium (USENIX Security 17)*. 2017, pp. 1093–1110.

[12] T. Aono, K. Higuchi, M. Taromaru, T. Ohira, and H. Sasaoka. “Wireless Secret Key Generation Exploiting the Reactance-Domain Scalar Response of Multipath Fading Channels: RSSI Interleaving Scheme”. In: *Wireless Technology, 2005. The European Conference On*. IEEE, 2005, pp. 173–176.

[13] F. Armknecht, P. Walther, G. Tsudik, M. Beck, and T. Strufe. “ProMACs: Progressive and Resynchronizing MACs for Continuous Efficient Authentication of Message Streams”. In: *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 2020, pp. 211–223.

[14] V. Arzamasov, K. Böhm, and I. Rutter. “Minimizing Bias in Estimation of Mutual Information from Data Streams”. In: (2019), p. 12.

[15] B. Azimi-Sadjadi, A. Kiayias, A. Mercado, and B. Yener. “Robust Key Generation from Signal Envelopes in Wireless Networks”. In: *Proceedings of the 14th ACM Conference on Computer and Communications Security*. CCS ’07. New York, NY, USA: ACM, 2007, pp. 401–410. ISBN: 978-1-59593-703-2. DOI: 10.1145/1315245.1315295.

[16] D. Q. Bala and B. Raman. “PHY-Based Key Agreement Scheme Using Audio Networking”. In: *2020 International Conference on COMmunication Systems NETworkS (COMSNETS)*. Jan. 2020, pp. 129–136. DOI: 10.1109/COMSNETS48256.2020.9027340.

[17] C. A. Balanis. *Antenna Theory: Analysis and Design*. 3rd ed. Hoboken, NJ: John Wiley, 2005. ISBN: 978-0-471-66782-7.

[18] N. J. Beaudry and R. Renner. “An Intuitive Proof of the Data Processing Inequality”. In: *arXiv:1107.0740 [quant-ph]* (July 2011). Comment: 10 pages. arXiv: 1107.0740 [quant-ph].

[19] P. Bello. “Characterization of Randomly Time-Variant Linear Channels”. In: *IEEE transactions on Communications Systems* 11.4 (1963), pp. 360–393.

[20] S. Ben Hamida, J.-B. Pierrot, B. Denis, C. Castelluccia, and B. Uguen. “On the Security of UWB Secret Key Generation Methods against Deterministic Channel Prediction Attacks”. In: Sept. 2012. DOI: 10.1109/VTCFall.2012.6399358.

[21] C. H. Bennett, F. Bessette, G. Brassard, L. Salvail, and J. Smolin. “Experimental Quantum Cryptography”. In: *Journal of cryptology* 5.1 (1992), pp. 3–28.

[22] J. Bergstra, D. Yamins, and D. Cox. “Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures”. In: *International Conference on Machine Learning*. PMLR, 2013, pp. 115–123.

[23] K.-L. Besser, P.-H. Lin, C. R. Janda, and E. A. Jorswieck. “Wiretap Code Design by Neural Network Autoencoders”. In: *IEEE Transactions on Information Forensics and Security* 15 (2019), pp. 3374–3386.

[24] C. M. Bishop. *Pattern Recognition and Machine Learning*. springer, 2006.

[25] M. Bkassiny, Y. Li, and S. K. Jayaweera. "A Survey on Machine-Learning Techniques in Cognitive Radios". In: *IEEE Communications Surveys & Tutorials* 15.3 (2012), pp. 1136–1159.

[26] M. Bloch and J. Barros. *Physical-Layer Security: From Information Theory to Security Engineering*. Cambridge University Press, 2011.

[27] M. Bloch, A. Thangaraj, and S. W. McLaughlin. "Efficient Reconciliation of Correlated Continuous Random Variables Using LDPC Codes". In: *arXiv preprint cs/0509041* (2005). arXiv: cs/0509041.

[28] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson. "The Industrial Internet of Things (IIoT): An Analysis Framework". In: *Computers in industry* 101 (2018), pp. 1–12.

[29] G. Brassard and L. Salvail. "Secret-Key Reconciliation by Public Discussion". In: *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer, 1993, pp. 410–423.

[30] A. Buja, W. Stuetzle, and Y. Shen. "Loss Functions for Binary Class Probability Estimation and Classification: Structure and Applications". In: *Working draft, November* 3 (2005).

[31] M. Bulenok, I. Tunaru, L. Biard, B. Denis, and B. Uguen. "Experimental Channel-Based Secret Key Generation with Integrated Ultra Wideband Devices". In: *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. Wichtig! Sept. 2016, pp. 1–6. DOI: 10.1109/PIMRC.2016.7794705.

[32] *Die Lage der IT-Sicherheit in Deutschland 2014*. Tech. rep. Nov. 2014, p. 44.

[33] W. T. Buttler, S. K. Lamoreaux, J. R. Torgerson, G. H. Nickel, C. H. Donahue, and C. G. Peterson. "Fast, Efficient Error Reconciliation for Quantum Cryptography". In: *Physical Review A* 67.5 (2003), p. 052303.

[34] T. I. Calver. "An Empirical Analysis of the Cascade Secret Key Reconciliation Protocol for Quantum Key Distribution". In: (2011).

[35] J. M. Ceron, C. Scholten, A. Pras, E. Lastdrager, and J. Santanna. "Characterising Attacks Targeting Low-Cost Routers: A MikroTik Case Study (Extended)". In: *arXiv:2011.01685 [cs]* (Nov. 2020). arXiv: 2011.01685 [cs].

[36] C. Chen and M. Jensen. "Random Number Generation from Multipath Propagation: MIMO-Based Encryption Key Establishment". In: (June 2009), pp. 1–4. DOI: 10.1109/APS.2009.5172006.

[37] C. Chen and M. A. Jensen. "Improved Channel Quantization for Secret Key Establishment in Wireless Systems". In: *2010 IEEE International Conference on Wireless Information Technology and Systems*. Aug. 2010, pp. 1–4. DOI: 10.1109/ICWITS.2010.5611930.

[38] C. Chen and M. A. Jensen. "Secret Key Establishment Using Temporally and Spatially Correlated Wireless Channel Coefficients". In: *IEEE Transactions on Mobile Computing* 10.2 (Feb. 2011), pp. 205–215. DOI: 10.1109/TMC.2010.114.

[39] J. S. Chitode. *Communication Theory*. Technical Publications, 2010. ISBN: 978-81-8431-763-3.

[40] G. Chu, N. Apthorpe, and N. Feamster. "Security and Privacy Analyses of Internet of Things Children's Toys". In: *IEEE Internet of Things Journal* 6.1 (2018), pp. 978–985.

[41] J. Classen, D. Wegemer, P. Patras, T. Spink, and M. Hollick. "Anatomy of a Vulnerable Fitness Tracking System: Dissecting the Fitbit Cloud, App, and Firmware". In: *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 2.1 (2018), pp. 1–24.

[42] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, 2012.

[43] I. Csiszar and J. Korner. "Broadcast Channels with Confidential Messages". In: *IEEE Transactions on Information Theory* 24.3 (May 1978), pp. 339–348. DOI: 10.1109/TIT.1978.1055892.

[44] Decawave. *DecaRangeRTLS ARM Source Code Guide*. 2015.

[45] Decawave. *DW1000 Datasheet*. 2017.

[46] Decawave. *DW1000 Radio IC*.

[47] Decawave. *DW1000 User Manual*. 2017.

[48] G. Deng and L. W. Cahill. "An Adaptive Gaussian Filter for Noise Reduction and Edge Detection". In: *1993 IEEE Conference Record Nuclear Science Symposium and Medical Imaging Conference*. IEEE, 1993, pp. 1615–1619.

[49] Y. Dodis, L. Reyzin, and A. Smith. "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data". In: *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2004, pp. 523–540.

[50] N. Döttling, D. Lazich, J. Müller-Quade, and A. S. de Almeida. "Vulnerabilities of Wireless Key Exchange Based on Channel Reciprocity". In: *Information Security Applications*. Ed. by Y. Chung and M. Yung. Vol. 6513. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 206–220. ISBN: 978-3-642-17954-9. DOI: 10.1007/978-3-642-17955-6_15.

[51] T. Dozat. "Incorporating Nesterov Momentum into Adam". In: (2016).

[52] S. Eberz, M. Strohmeier, M. Wilhelm, and I. Martinovic. "A Practical Man-In-The-Middle Attack on Signal-Based Key Generation Protocols." In: *ESORICS*. Vol. 12. Springer, 2012, pp. 235–252.

[53] M. Edman, A. Kiayias, and B. Yener. "On Passive Inference Attacks against Physical-Layer Key Extraction?" In: *Proceedings of the Fourth European Workshop on System Security*. EUROSEC '11. Salzburg, Austria: Association for Computing Machinery, Apr. 2011, pp. 1–6. ISBN: 978-1-4503-0613-3. DOI: 10.1145/1972551.1972559.

[54] M. Edman, A. Kiayias, Q. Tang, and B. Yener. "On the Security of Key Extraction From Measuring Physical Quantities". In: *IEEE Transactions on Information Forensics and Security* 11.8 (Aug. 2016), pp. 1796–1806. DOI: 10.1109/TIFS.2016.2543687.

[55] S. Edwards and I. Profetis. "Hajime: Analysis of a Decentralized Internet Worm for IoT Devices". In: *Rapidity Networks* 16 (2016), pp. 1–18.

[56] J. Etesami and W. Henkel. "LDPC Code Construction for Wireless Physical-Layer Key Reconciliation". In: *2012 1st IEEE International Conference on Communications in China (ICCC)*. IEEE, 2012, pp. 208–213.

[57] B. Etzlinger and H. Wymeersch. "Synchronization and Localization in Wireless Networks". In: *Foundations and Trends® in Signal Processing* 12.1 (2018), pp. 1–106.

[58] N. Falliere, L. O. Murchu, and E. Chien. "W32. Stuxnet Dossier". In: *White paper, Symantec Corp., Security Response* 5.6 (2011), p. 29.

[59] G. P. Fettweis. "The Tactile Internet: Applications and Challenges". In: *IEEE Vehicular Technology Magazine* 9.1 (2014), pp. 64–70.

[60] F. H. Fitzek, S.-C. Li, S. Speidel, and T. Strufe. "Tactile Internet with Human-in-the-Loop: New Frontiers of Transdisciplinary Research". In: *Tactile Internet*. Elsevier, 2021, pp. 1–19.

[61] S. J. Fortune, D. M. Gay, B. W. Kernighan, O. Landron, R. A. Valenzuela, and M. H. Wright. "WISE Design of Indoor Wireless Systems: Practical Computation and Optimization". In: *IEEE Computational science and Engineering* 2.1 (1995), pp. 58–68.

[62] M. Frustaci, P. Pace, G. Aloi, and G. Fortino. "Evaluating Critical Security Issues of the IoT World: Present and Future Challenges". In: *IEEE Internet of Things Journal* 5.4 (Aug. 2018), pp. 2483–2495. DOI: 10.1109/JIOT.2017.2767291.

[63] K. Fu, T. Kohno, D. Lopresti, E. Mynatt, K. Nahrstedt, S. Patel, D. Richardson, and B. Zorn. "Safety, Security, and Privacy Threats Posed by Accelerating Trends in the Internet of Things". In: *arXiv:2008.00017 [cs]* (July 2020). Comment: A Computing Community Consortium (CCC) white paper, 9 pages. arXiv: 2008.00017 [cs].

[64] F. Fuschini, E. M. Vitucci, M. Barbiroli, G. Falciasecca, and V. Degli-Esposti. "Ray Tracing Propagation Modeling for Future Small-Cell and Indoor Applications: A Review of Current Techniques". In: *Radio Science* 50.6 (2015), pp. 469–485. DOI: 10.1002/2015RS005659.

[65] S. Gao, G. V. Steeg, and A. Galstyan. "Efficient Estimation of Mutual Information for Strongly Dependent Variables". In: (2015), p. 10.

[66] A. Gatouillat, Y. Badr, B. Massot, and E. Sejdić. "Internet of Medical Things: A Review of Recent Contributions Dealing with Cyber-Physical Systems in Medicine". In: *IEEE internet of things journal* 5.5 (2018), pp. 3810–3822.

[67] G. German, Q. Spencer, L. Swindlehurst, and R. Valenzuela. "Wireless Indoor Channel Modeling: Statistical Agreement of Ray Tracing Simulations and Channel Sounding Measurements". In: *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*. Vol. 4. May 2001, 2501–2504 vol.4. DOI: 10.1109/ICASSP.2001.940509.

[68] A. Ghosal and M. Conti. "Security Issues and Challenges in V2X: A Survey". In: *Computer Networks* 169 (2020), p. 107093.

[69] A. Goldsmith. *Wireless Communications*. Cambridge university press, 2005.

[70] S. Gollakota and D. Katabi. "Physical Layer Wireless Security Made Fast and Channel Independent". In: *2011 Proceedings IEEE INFOCOM*. Apr. 2011, pp. 1125–1133. DOI: 10.1109/INFCOM.2011.5934889.

[71] S. Gopinath, R. Guillaume, P. Duplys, and A. Czylwik. "Reciprocity Enhancement and Decorrelation Schemes for PHY-Based Key Generation". In: *2014 IEEE Globecom Workshops (GC Wkshps)*. Dec. 2014, pp. 1367–1372. DOI: 10.1109/GLOCOMW.2014.7063624.

[72] F. Gray. “Pulse Code Communication”. US2632058 (A). Mar. 1953.

[73] F. Gringoli, M. Schulz, J. Link, and M. Hollick. “Free Your CSI: A Channel State Information Extraction Platform for Modern Wi-Fi Chipsets”. In: *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*. 2019, pp. 21–28.

[74] R. Guillaume, F. Winzer, A. Czylwik, C. T. Zenger, and C. Paar. “Bringing PHY-Based Key Generation into the Field: An Evaluation for Practical Scenarios”. In: *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall)*. Sept. 2015, pp. 1–5. DOI: 10.1109/VTCFall.2015.7390857.

[75] R. Guillaume, A. Mueller, C. T. Zenger, C. Paar, and A. Czylwik. “Fair Comparison and Evaluation of Quantization Schemes for Phy-Based Key Generation”. In: *OFDM 2014; 18th International OFDM Workshop 2014 (InOWo'14); Proceedings Of*. VDE, 2014, pp. 1–5.

[76] R. Hadsell, S. Chopra, and Y. LeCun. “Dimensionality Reduction by Learning an Invariant Mapping”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*. Vol. 2. IEEE, 2006, pp. 1735–1742.

[77] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. “Tool Release: Gathering 802.11N Traces with Channel State Information”. In: *SIGCOMM Comput. Commun. Rev.* 41.1 (Jan. 2011), pp. 53–53. DOI: 10.1145/1925861.1925870.

[78] M. Haluza and J. Vesely. “Analysis of Signals from the DecaWave TREK1000 Wideband Positioning System Using AKRS System”. In: *2017 International Conference on Military Technologies (ICMT)*. May 2017, pp. 424–429. DOI: 10.1109/MILTECHS.2017.7988797.

[79] S. T.-B. Hamida, J.-B. Pierrot, and C. Castelluccia. “An Adaptive Quantization Algorithm for Secret Key Generation Using Radio Channel Measurements”. In: *2009 3rd International Conference on New Technologies, Mobility and Security*. Dec. 2009, pp. 1–5. DOI: 10.1109/NTMS.2009.5384826.

[80] S. T.-B. Hamida, J.-B. Pierrot, and C. Castelluccia. “Empirical Analysis of UWB Channel Characteristics for Secret Key Generation in Indoor Environments”. In: *21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*. Sept. 2010, pp. 1984–1989. DOI: 10.1109/PIMRC.2010.5671596.

[81] J. Harttung, E. Franz, S. Moriam, and P. Walther. “Lightweight Authenticated Encryption for Network-on-Chip Communications”. In: *Proceedings of the 2019 on Great Lakes Symposium on VLSI*. 2019, pp. 33–38.

[82] X. He, H. Dai, W. Shen, P. Ning, and R. Dutta. “Toward Proper Guard Zones for Link Signature”. In: *IEEE Transactions on Wireless Communications* 15.3 (Mar. 2016), pp. 2104–2117. DOI: 10.1109/TWC.2015.2498621.

[83] C. Heffner and T. N. Solutions. “Exploiting Surveillance Cameras”. In: *Tactical Network Solutions, Tech. Rep* (2013).

[84] J. E. Hershey, A. A. Hassan, and R. Yarlagadda. “Unconventional Cryptographic Keying Variable Management”. In: *IEEE Transactions on Communications* 43.1 (Jan. 1995), pp. 3–6. DOI: 10.1109/26.385951.

[85] E. Hoffer and N. Ailon. “Deep Metric Learning Using Triplet Network”. In: (2014). arXiv: 1412.6622 [cs.LG].

[86] Y.-W. P. Hong, L.-M. Huang, and H.-T. Li. "Vector Quantization and Clustered Key Mapping for Channel-Based Secret Key Generation". In: *IEEE Transactions on Information Forensics and Security* 12.5 (May 2017), pp. 1170–1181. DOI: 10.1109/TIFS.2017.2656459.

[87] C. Hu, X. Cheng, F. Zhang, D. Wu, X. Liao, and D. Chen. "OPFKA: Secure and Efficient Ordered-Physiological-Feature-Based Key Agreement for Wireless Body Area Networks". In: *2013 Proceedings IEEE INFOCOM*. IEEE, 2013, pp. 2274–2282.

[88] Q. Hu and G. P. Hancke. "A Session Hijacking Attack on Physical Layer Key Generation Agreement". In: *2017 IEEE International Conference on Industrial Technology (ICIT)*. Mar. 2017, pp. 1418–1423. DOI: 10.1109/ICIT.2017.7915573.

[89] J. Huang and T. Jiang. "Dynamic Secret Key Generation Exploiting Ultra-Wideband Wireless Channel Characteristics". In: *2015 IEEE Wireless Communications and Networking Conference (WCNC)*. Mar. 2015, pp. 1701–1706. DOI: 10.1109/WCNC.2015.7127724.

[90] J. Huang, T. Jiang, and S. Zhai. "Secret Key Generation Exploiting Ultra-Wideband Indoor Wireless Channel Characteristics". In: *MILCOM 2013 - 2013 IEEE Military Communications Conference*. Nov. 2013, pp. 254–259. DOI: 10.1109/MILCOM.2013.51.

[91] S. R. Hussain, E. Bertino, and S. Nirjon. "Securing the Insecure Link of Internet-of-Things Using Next-Generation Smart Gateways". In: *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. May 2019, pp. 66–73. DOI: 10.1109/DCOSS.2019.00032.

[92] C. S. F. Huth. "Physical-Layer Security Architectures for the Internet of Things". PhD thesis. Ruhr Universität Bonn, 2018.

[93] "IEEE Standard for Local and Metropolitan Area Networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)". In: *IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006)* (Sept. 2011), pp. 1–314. DOI: 10.1109/IEEESTD.2011.6012487.

[94] *iPhone 11's Ultra-Wideband Chip Helps You AirDrop with the Right Person*. Sept. 2019.

[95] W. C. Jakes, ed. *Microwave Mobile Communications*. Nachdr. An IEEE Press Classic Reissue. New York, NY: IEEE Press [u.a.], 1995. ISBN: 978-0-7803-1069-8.

[96] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. "On the Effectiveness of Secret Key Extraction from Wireless Signal Strength in Real Environments". In: *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*. ACM, 2009, pp. 321–332.

[97] R. Jin and K. Zeng. "Manipulative Attack Against Physical Layer Key Agreement and Countermeasure". In: *IEEE Transactions on Dependable and Secure Computing* 18.1 (Jan. 2019), pp. 475–489. DOI: 10.1109/TDSC.2019.2895325.

[98] R. Jin and K. Zeng. "Physical Layer Key Agreement under Signal Injection Attacks". In: *2015 IEEE Conference on Communications and Network Security (CNS)*. Sept. 2015, pp. 254–262. DOI: 10.1109/CNS.2015.7346835.

[99] Q. Jing, A. V. Vasilakos, J. Wan, J. Lu, and D. Qiu. "Security of the Internet of Things: Perspectives and Challenges". In: *Wireless Networks* 20.8 (Nov. 2014), pp. 2481–2501. DOI: 10.1007/s11276-014-0761-7.

[100] A. Juels and M. Wattenberg. "A Fuzzy Commitment Scheme". In: *Proceedings of the 6th ACM Conference on Computer and Communications Security - CCS '99*. Kent Ridge Digital Labs, Singapore: ACM Press, 1999, pp. 28–36. ISBN: 978-1-58113-148-2. DOI: 10.1145/319709.319714.

[101] H. Kayan, M. Nunes, O. Rana, P. Burnap, and C. Perera. "Cybersecurity of Industrial Cyber-Physical Systems: A Review". In: *arXiv:2101.03564 [cs]* (Jan. 2021). Comment: 32 pages, 10 figures. arXiv: 2101.03564 [cs].

[102] J. B. Kinney and G. S. Atwal. "Equitability, Mutual Information, and the Maximal Information Coefficient". In: *Proceedings of the National Academy of Sciences of the United States of America* 111.9 (Mar. 2014), pp. 3354–3359. DOI: 10.1073/pnas.1309933111.

[103] D. Kirovski, M. Sinclair, and D. Wilson. "The Martini Synch". In: *Microsoft Research, Cambridge, UK, Tech. Rep. MSR-TR-2007-123* (2007).

[104] A. Kitaura, T. Sumi, T. Tango, H. Iwai, and H. Sasaoka. "A Private Key Sharing Scheme Based on Multipath Time Delay in UWB Systems". In: *2006 International Conference on Communication Technology*. Nov. 2006, pp. 1–4. DOI: 10.1109/ICCT.2006.341996.

[105] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas. "DDoS in the IoT: Mirai and Other Botnets". In: *Computer* 50.7 (2017), pp. 80–84.

[106] A. Kraskov, H. Stögbauer, and P. Grassberger. "Estimating Mutual Information". In: *Physical review E* 69.6 (2004), p. 066138.

[107] M. J. Kuhn, J. Turnmire, M. R. Mahfouz, and A. E. Fathy. "Adaptive Leading-Edge Detection in UWB Indoor Localization". In: *Radio and Wireless Symposium (RWS), 2010 IEEE*. IEEE, 2010, pp. 268–271.

[108] J. Kunisch and J. Pamp. *Radio Channel Model for Indoor UWB WPAN Environments*. Tech. rep. IEEE P802.15-02/281. IEEE 802.15.3a, 2002.

[109] M. Laaraiedh, N. Amiot, and B. Uguen. "Efficient Ray Tracing Tool for UWB Propagation and Localization Modeling". In: *2013 7th European Conference on Antennas and Propagation (EuCAP)*. Apr. 2013, pp. 2307–2311.

[110] R. Langner. "Stuxnet: Dissecting a Cyberwarfare Weapon". In: *IEEE Security & Privacy* 9.3 (2011), pp. 49–51.

[111] H. Lee, P. Pham, Y. Largman, and A. Ng. "Unsupervised Feature Learning for Audio Classification Using Convolutional Deep Belief Networks". In: *Advances in neural information processing systems* 22 (2009), pp. 1096–1104.

[112] K. Lee, N. Klingensmith, S. Banerjee, and Y. Kim. "VoltKey: Continuous Secret Key Generation Based on Power Line Noise for Zero-Involvement Pairing and Authentication". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3.3 (Sept. 2019), 93:1–93:26. DOI: 10.1145/3351251.

[113] N. N. Leonenko and L. F. Kozachenko. "Sample Estimate of the Entropy of a Random Vector". In: *Problems of Information Transmission* 23 (1987), pp. 95–101.

[114] Y. Liang, H. V. Poor, and S. Shamai. *Information Theoretic Security*. Now Publishers Inc, 2009.

[115] H. Lim, L.-C. Kung, J. C. Hou, and H. Luo. "Zero-Configuration, Robust Indoor Localization: Theory and Experimentation". In: *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*. Apr. 2006, pp. 1–12. DOI: 10.1109/INFOCOM.2006.223.

[116] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao. "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications". In: *IEEE Internet of Things Journal* 4.5 (Oct. 2017), pp. 1125–1142. DOI: 10.1109/JIOT.2017.2683200.

[117] Q. Lin, W. Xu, J. Liu, A. Khamis, W. Hu, M. Hassan, and A. Seneviratne. "H2B: Heartbeat-Based Secret Key Generation Using Piezo Vibration Sensors". In: *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*. 2019, pp. 265–276.

[118] R. Lin, L. Xu, H. Fang, and C. Huang. "Efficient Physical Layer Key Generation Technique in Wireless Communications". In: *EURASIP Journal on Wireless Communications and Networking* 2020.1 (Jan. 2020), p. 13. DOI: 10.1186/s13638-019-1634-7.

[119] H. Liu, J. Yang, Y. Wang, and Y. Chen. "Collaborative Secret Key Extraction Leveraging Received Signal Strength in Mobile Wireless Networks". In: *2012 Proceedings IEEE INFOCOM*. IEEE, 2012, pp. 927–935.

[120] H. Liu, Y. Wang, J. Yang, and Y. Chen. "Fast and Practical Secret Key Extraction by Exploiting Channel Response". In: *2013 Proceedings IEEE INFOCOM*. Apr. 2013, pp. 3048–3056. DOI: 10.1109/INFCOM.2013.6567117.

[121] Y. Liu, S. C. Draper, and A. M. Sayeed. "Exploiting Channel Diversity in Secret Key Generation From Multipath Fading Randomness". In: *IEEE Transactions on Information Forensics and Security* 7.5 (Oct. 2012). csi >> rssi, pp. 1484–1497. DOI: 10.1109/TIFS.2012.2206385.

[122] Y. Lu, F. Wu, S. Tang, L. Kong, and G. Chen. "FREE: A Fast and Robust Key Extraction Mechanism via Inaudible Acoustic Signal". In: *Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. Mobihoc '19. Catania, Italy: Association for Computing Machinery, July 2019, pp. 311–320. ISBN: 978-1-4503-6764-6. DOI: 10.1145/3323679.3326529.

[123] M. G. Madiseh. "Wireless Secret Key Generation versus Capable Adversaries". In: 2011.

[124] M. G. Madiseh, M. L. McGuire, S. W. Neville, and A. A. B. Shirazi. "Secret Key Extraction in Ultra Wideband Channels for Unsynchronized Radios". In: *6th Annual Communication Networks and Services Research Conference (Cnsr 2008)*. Sync error bad. May 2008, pp. 88–95. DOI: 10.1109/CNSR.2008.52.

[125] M. G. Madiseh, M. L. McGuire, S. S. Neville, L. Cai, and M. Horie. "Secret Key Generation and Agreement in UWB Communication Channels". In: *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*. Nov. 2008, pp. 1–5. DOI: 10.1109/GLOCOM.2008.ECP.356.

[126] M. G. Madiseh, S. He, M. L. Mcguire, S. W. Neville, and X. Dong. "Verification of Secret Key Generation from UWB Channel Observations". In: *2009 IEEE International Conference on Communications*. June 2009, pp. 1–5. DOI: 10.1109/ICC.2009.5199564.

[127] R. K. Mahapatra and N. S. V. Shet. "Localization Based on RSSI Exploiting Gaussian and Averaging Filter in Wireless Sensor Network". In: *Arabian Journal for Science and Engineering* 43.8 (2018), pp. 4145–4159.

[128] A. U. Makarfi, K. M. Rabie, O. Kaiwartya, K. Adhikari, G. Nauryzbayev, X. Li, and R. Kharel. "Toward Physical-Layer Security for Internet of Vehicles: Interference-Aware Modeling". In: *IEEE Internet of Things Journal* 8.1 (Jan. 2021), pp. 443–457. DOI: 10.1109/JIOT.2020.3006527.

[129] G. Margelis, X. Fafoutis, G. Oikonomou, R. Piechocki, T. Tryfonas, and P. Thomas. "Efficient DCT-Based Secret Key Generation for the Internet of Things". In: *Ad Hoc Networks*. Special Issue on Security of IoT-Enabled Infrastructures in Smart Cities 92 (Sept. 2019), p. 101744. DOI: 10.1016/j.adhoc.2018.08.014.

[130] F. Marino, E. Paolini, and M. Chiani. "Secret Key Extraction from a UWB Channel: Analysis in a Real Environment". In: *2014 IEEE International Conference on Ultra-WideBand (ICUWB)*. Wichtig. Paris, France: IEEE, Sept. 2014, pp. 80–85. ISBN: 978-1-4799-5396-7. DOI: 10.1109/ICUWB.2014.6958955.

[131] D. Masters and C. Luschi. "Revisiting Small Batch Training for Deep Neural Networks". In: *arXiv:1804.07612 [cs, stat]* (Apr. 2018). arXiv: 1804.07612 [cs, stat].

[132] S. Mathur, R. Miller, A. Varshavsky, W. Trappe, and N. Mandayam. "Proximate: Proximity-Based Secure Pairing Using Ambient Wireless Signals". In: *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*. 2011, pp. 211–224.

[133] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik. "Radio-Telepathy: Extracting a Secret Key from an Unauthenticated Wireless Channel". In: *Proceedings of the 14th ACM International Conference on Mobile Computing and Networking*. ACM, 2008, pp. 128–139.

[134] U. M. Maurer. "Secret Key Agreement by Public Discussion from Common Information". In: *IEEE Transactions on Information Theory* 39.3 (May 1993), pp. 733–742. DOI: 10.1109/18.256484.

[135] R. Mayrhofer and H. Gellersen. "Shake Well before Use: Authentication Based on Accelerometer Data". In: *International Conf. on Pervasive Computing*. Springer, 2007, pp. 144–161.

[136] M. McGuire. "Channel Estimation for Secret Key Generation". In: *2014 IEEE 28th International Conference on Advanced Information Networking and Applications*. May 2014, pp. 490–496. DOI: 10.1109/AINA.2014.60.

[137] B. Merkl. "The Future of the Operating Room: Surgical Preplanning and Navigation Using High Accuracy Ultra-Wideband Positioning and Advanced Bone Measurement". In: *Doctoral Dissertations* (Dec. 2008).

[138] S. Microelectronics. *STM32F105xx, STM32F107xx - Datasheet*. Mar. 2017.

[139] S. M. MirhoseiniNejad, A. Rahmanpour, and S. M. Razavizadeh. "Phase Jamming Attack: A Practical Attack on Physical Layer-Based Key Derivation". In: *2018 15th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology (ISCISC)*. Aug. 2018, pp. 1–4. DOI: 10.1109/ISCISC.2018.8546920.

[140] M. Mitev, A. Chorti, E. V. Belmega, and M. Reed. "Man-in-the-Middle and Denial of Service Attacks in Wireless Secret Key Generation". In: *2019 IEEE Global Communications Conference (GLOBECOM)*. Dec. 2019, pp. 1–6. DOI: 10.1109/GLOBECOM38437.2019.9013816.

[141] S. K. Mitra. *Signals and Systems*. Oxford Series in Electrical and Computer Engineering. Oxford University Press, 2016. ISBN: 978-0-19-024529-0.

[142] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning*. MIT press, 2018.

[143] A. F. Molisch. *Wireless Communications*. 2nd ed. Chichester, West Sussex, U.K: Wiley : IEEE, 2011. ISBN: 978-0-470-74187-0.

[144] A. F. Molisch, K. Balakrishnan, C.-C. Chong, S. Emami, A. Fort, J. Karedal, J. Kunisch, H. Schantz, U. Schuster, and K. Siwiak. "IEEE 802.15. 4a Channel Model-Final Report". In: *IEEE P802* 15.04 (2004), p. 0662.

[145] D. R. Morgan, J. Benesty, and M. M. Sondhi. "On the Evaluation of Estimated Impulse Responses". In: *IEEE Signal processing letters* 5.7 (1998), pp. 174–176.

[146] S. Moriam, E. Franz, P. Walther, A. Kumar, T. Strufe, and G. Fettweis. "Efficient Communication Protection of Many-Core Systems against Active Attackers". In: *Electronics* 10.3 (2021). Multidisciplinary Digital Publishing Institute, p. 238.

[147] S. Moriam, E. Franz, P. Walther, A. Kumar, T. Strufe, and G. Fettweis. "Protecting Communication in Many-Core Systems against Active Attackers". In: *Proceedings of the 2018 on Great Lakes Symposium on VLSI*. 2018, pp. 45–50.

[148] H. V. Nguyen and L. Bai. "Cosine Similarity Metric Learning for Face Verification". In: *Asian Conference on Computer Vision*. Springer, 2010, pp. 709–720.

[149] K.-C. Nguyen, G. Van Assche, and N. J. Cerf. "Side-Information Coding with Turbo Codes and Its Application to Quantum Key Distribution". In: *arXiv preprint cs/0406001* (2004). arXiv: cs/0406001.

[150] A. Nordrum. "The Internet of Fewer Things [News]". In: *IEEE Spectrum* 53.10 (Oct. 2016), pp. 12–13. DOI: 10.1109/MSPEC.2016.7572524.

[151] P. Pagani, F. T. Talom, P. Pajusco, and B. Uguen. *Ultra-Wideband Radio Propagation Channels: A Practical Approach*. John Wiley & Sons, 2013.

[152] F. Pan, Z. Pang, M. Luvisotto, M. Xiao, and H. Wen. "Physical-Layer Security for Industrial Wireless Control Systems: Basics and Future Directions". In: *IEEE Industrial Electronics Magazine* 12.4 (Dec. 2018), pp. 18–27. DOI: 10.1109/MIE.2018.2874385.

[153] T. W. Parks and C. S. Burrus. *Digital Filter Design*. Wiley-Interscience, 1987.

[154] N. Patwari, J. Croft, S. Jana, and K. Kasera Sneha K. "High-Rate Uncorrelated Bit Extraction for Shared Secret Key Generation from Channel Measurements". In: *IEEE Transactions on Mobile Computing* 9.1 (Jan. 2010), pp. 17–30. DOI: 10.1109/TMC.2009.88.

[155] Y. Peng, P. Wang, W. Xiang, and Y. Li. "Secret Key Generation Based on Estimated Channel State Information for TDD-OFDM Systems Over Fading Channels". In: *IEEE Transactions on Wireless Communications* 16.8 (Aug. 2017), pp. 5176–5186. DOI: 10.1109/TWC.2017.2706657.

[156] A. Pittolo and A. M. Tonello. "Physical Layer Security in Power Line Communication Networks: An Emerging Scenario, Other than Wireless". In: *IET Communications* 8.8 (May 2014), pp. 1239–1247. DOI: 10.1049/iet-com.2013.0472.

[157] S. N. Premnath, S. Jana, J. Croft, P. L. Gowda, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy. "Secret Key Extraction from Wireless Signal Strength in Real Environments". In: *IEEE Transactions on Mobile Computing* 12.5 (May 2013), pp. 917–930. DOI: 10.1109/TMC.2012.63.

[158] S. N. Premnath, P. L. Gowda, S. K. Kasera, N. Patwari, and R. Ricci. "Secret Key Extraction Using Bluetooth Wireless Signal Strength Measurements". In: *2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. June 2014, pp. 293–301. DOI: 10.1109/SAHCN.2014.6990365.

[159] E. Ronen, C. O'Flynn, A. Shamir, and A.-O. Weingarten. *IoT Goes Nuclear: Creating a ZigBee Chain Reaction*. Tech. rep. 1047. 2016.

[160] M. Rostami, A. Juels, and F. Koushanfar. "Heart-to-Heart (H2H) Authentication for Implanted Medical Devices". In: *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*. 2013, pp. 1099–1112.

[161] J. Rutkowska, A. Tereshkin, and R. Wojtczuk. "Thoughts about Trusted Computing". In: *Invisible Things Lab, EuSecWest May* (2009), pp. 27–28.

[162] Z. Sahinoglu and S. Gezici. "Ranging in the IEEE 802.15. 4a Standard". In: *Wireless and Microwave Technology Conference, 2006. WAMICON'06. IEEE Annual*. IEEE, 2006, pp. 1–5.

[163] A. A. M. Saleh and R. A. Valenzuela. "A Statistical Model for Indoor Multipath Propagation". In: *IEEE Journal on Selected Areas in Communications* 5.2 (Feb. 1987), pp. 128–137. DOI: 10.1109/JSAC.1987.1146527.

[164] A. Sayeed and A. Perrig. "Secure Wireless Communications: Secret Keys through Multipath". In: *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. Mar. 2008, pp. 3013–3016. DOI: 10.1109/ICASSP.2008.4518284.

[165] B. Schneier. "Evil Maid Attacks on Encrypted Hard Drives". In: *Crypto-Gram Newsletter* (2009).

[166] F. Schroff, D. Kalenichenko, and J. Philbin. "FaceNet: A Unified Embedding for Face Recognition and Clustering". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015, pp. 815–823. DOI: 10.1109/CVPR.2015.7298682.

[167] M. Schulz, A. Loch, and M. Hollick. "Demonstrating Practical Known-Plaintext Attacks against Physical Layer Security in Wireless MIMO Systems". In: *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks*. 2016, pp. 201–203.

[168] M. Schulz, A. Loch, and M. Hollick. "Practical Known-Plaintext Attacks against Physical Layer Security in Wireless MIMO Systems." In: *NDSS*. 2014.

[169] D. Schürmann, A. Brüsch, S. Sigg, and L. Wolf. "BANDANA—Body Area Network Device-to-Device Authentication Using Natural gAit". In: *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2017, pp. 190–196.

[170] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. "Taking the Human out of the Loop: A Review of Bayesian Optimization". In: *Proceedings of the IEEE* 104.1 (2015), pp. 148–175.

[171] C. E. Shannon. "Communication Theory of Secrecy Systems". In: *The Bell system technical journal* 28.4 (1949), pp. 656–715.

[172] Z. Sharif and A. Z. Sha'ameri. "The Application of Cross Correlation Technique for Estimating Impulse Response and Frequency Response of Wireless Communication Channel". In: *2007 5th Student Conference on Research and Development*. Selangor, Malaysia: IEEE, 2007, pp. 1–5. ISBN: 978-1-4244-1469-7. DOI: 10.1109/SCORED.2007.4451386.

[173] S. Shasha, M. Mahmoud, M. Mannan, and A. Youssef. "Playing with Danger: A Taxonomy and Evaluation of Threats to Smart Toys". In: *IEEE Internet of Things Journal* 6.2 (2019), pp. 2986–3002.

[174] C. Shen, H. Li, G. Sahin, H.-A. Choi, and Y. Shah. "Golay Code Based Bit Mismatch Mitigation for Wireless Channel Impulse Response Based Secrecy Generation". In: *IEEE Access* 7 (2019), pp. 2999–3007. DOI: 10.1109/ACCESS.2018.2888489.

[175] G. J. Simmons. "A Survey of Information Authentication". In: *Proceedings of the IEEE* 76.5 (1988), pp. 603–620.

[176] G. J. Simmons. "Authentication Theory/Coding Theory". In: *Workshop on the Theory and Application of Cryptographic Techniques*. Springer, 1984, pp. 411–431.

[177] A. Soni, R. Upadhyay, and A. Kumar. "Dimensionality Reduction in Wireless Physical Layer Key Generation". In: *2019 IEEE 16th India Council International Conference (INDICON)*. Dec. 2019, pp. 1–4. DOI: 10.1109/INDICON47234.2019.9029059.

[178] P. Staat, H. Elders-Boll, M. Heinrichs, R. Kronberger, C. Zenger, and C. Paar. "Intelligent Reflecting Surface-Assisted Wireless Key Generation for Low-Entropy Environments". In: *arXiv:2010.06613 [cs, eess]* (Mar. 2021). arXiv: 2010.06613 [cs, eess].

[179] R. Steele. *Mobile Radio Communications*. Pentech Press, John Wiley, 1992.

[180] D. Steinmetzer, M. Schulz, and M. Hollick. "Lockpicking Physical Layer Key Exchange: Weak Adversary Models Invite the Thief". In: *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks - WiSec '15*. New York, New York: ACM Press, 2015, pp. 1–11. ISBN: 978-1-4503-3623-9. DOI: 10.1145/2766498.2766514.

[181] C. Sturm, W. Sorgel, T. Kayser, and W. Wiesbeck. "Deterministic UWB Wave Propagation Modeling for Localization Applications Based on 3D Ray Tracing". In: *2006 IEEE MTT-S International Microwave Symposium Digest*. IEEE, 2006, pp. 2003–2006.

[182] X. Sun, W. Xu, M. Jiang, and C. Zhao. "Improved Generation Efficiency for Key Extracting from Wireless Channels". In: *2011 IEEE International Conference on Communications (ICC)*. June 2011, pp. 1–6. DOI: 10.1109/icc.2011.5962502.

[183] Y. Sun, C. Wong, G.-Z. Yang, and B. Lo. "Secure Key Generation Using Gait Features for Body Sensor Networks". In: *2017 IEEE 14th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*. IEEE, 2017, pp. 206–210.

[184] W. Traisilanun, K. Sripimanwat, and O. Sangaroon. "Secret Key Reconciliation Using BCH Code in Quantum Key Distribution". In: *2007 International Symposium on Communications and Information Technologies*. IEEE, 2007, pp. 1482–1485.

[185] W. Trappe. "The Challenges Facing Physical Layer Security". In: *IEEE Communications Magazine* 53.6 (2015), pp. 16–20.

[186] I. Tunaru, B. Denis, and B. Uguen. "Reciprocity-Diversity Trade-off in Quantization for Secret Key Generation". In: *2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC)*. Sept. 2014, pp. 134–138. DOI: 10.1109/PIMRC.2014.7136147.

[187] J. Valente and A. A. Cardenas. "Security & Privacy in Smart Toys". In: *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy*. 2017, pp. 19–24.

[188] N. Vigdor. "Somebody's Watching: Hackers Breach Ring Home Security Cameras". In: *The New York Times* (Dec. 2019).

[189] H. Vogt and A. Sezgin. "Secret-Key Generation from Wireless Channels: Mind the Reflections". In: *2014 IEEE International Conference on Communications Workshops (ICC)*. June 2014, pp. 783–788. DOI: 10.1109/ICCW.2014.6881295.

[190] J. Wallace. "Secure Physical Layer Key Generation Schemes: Performance and Information Theoretic Limits". In: *2009 IEEE International Conference on Communications*. Dresden, Germany: IEEE, June 2009, pp. 1–5. DOI: 10.1109/ICC.2009.5199440.

[191] J. W. Wallace, C. Chen, and M. A. Jensen. "Key Generation Exploiting MIMO Channel Evolution: Algorithms and Theoretical Limits". In: *2009 3rd European Conference on Antennas and Propagation*. Mar. 2009, pp. 1499–1503.

[192] J. W. Wallace and R. K. Sharma. "Automatic Secret Keys From Reciprocal MIMO Wireless Channels: Measurement and Analysis". In: *IEEE Transactions on Information Forensics and Security* 5.3 (Sept. 2010), pp. 381–392. DOI: 10.1109/TIFS.2010.2052253.

[193] P. Walther, E. Franz, and T. Strufe. "Blind Synchronization of Channel Impulse Responses for Channel Reciprocity-Based Key Generation". In: *2019 IEEE 44th Conference on Local Computer Networks (LCN)*. IEEE, 2019, pp. 76–83.

[194] P. Walther, R. Knauer, and T. Strufe. *Passive Angriffe auf kanalbasierten Schlüsselaustausch*. Gesellschaft für Informatik e.V., 2020. ISBN: 978-3-88579-695-4. DOI: 10.18420/sicherheit2020_03.

[195] P. Walther, R. Knauer, and T. Strufe. *Ultra-Wideband Channel State Information and Localization for Physical Layer Security*. Feb. 2021. DOI: 10.21227/0WEJ-BC28.

[196] P. Walther, M. Richter, and T. Strufe. "Ray-Tracing Based Inference Attacks on Physical Layer Security". In: *2021 International Conference on Networked Systems (NetSys)*. 2021, pp. 1–4.

[197] P. Walther and T. Strufe. "Blind Twins: Siamese Networks for Non-Interactive Information Reconciliation". In: *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE, 2020, pp. 1–7.

[198] P. Walther and T. Strufe. "Machine Learning Aided Inference Attacks on Channel State Information". In: *2020 IEEE 19th International Conference On Trust, Security And Privacy In Computing And Communications (TrustCom)*. IEEE, 2020.

[199] P. Walther, S. Köpsell, F. Armknecht, G. Tsudik, and T. Strufe. "Chains and Whips-An Approach to Lighweight MACs". In: *crypto day matters 28* (2018). Gesellschaft für Informatik eV/FG KRYPTO.

[200] P. Walther, C. Janda, E. Franz, M. Pelka, H. Hellbrück, T. Strufe, and E. Jorswieck. "Improving Quantization for Channel Reciprocity Based Key Generation". In: *2018 IEEE 43rd Conference on Local Computer Networks (LCN)*. IEEE, 2018, pp. 545–552.

[201] C. Wan, L. Wang, and V. V. Phoha. "A Survey on Gait Recognition". In: *ACM Computing Surveys (CSUR)* 51.5 (2018), pp. 1–35.

[202] N. Wang, P. Wang, A. Alipour-Fanid, L. Jiao, and K. Zeng. "Physical-Layer Security of 5G Wireless Networks for IoT: Challenges and Opportunities". In: *IEEE Internet of Things Journal* 6.5 (Oct. 2019), pp. 8169–8181. DOI: 10.1109/JIOT.2019.2927379.

[203] Q. Wang, W. U. Hassan, A. Bates, and C. Gunter. "Fear and Logging in the Internet of Things". In: *Network and Distributed Systems Symposium*. 2018.

[204] Q. Wang, H. Su, K. Ren, and K. Kim. "Fast and Scalable Secret Key Generation Exploiting Channel Phase Randomness in Wireless Networks". In: *2011 Proceedings IEEE INFOCOM* (2011). DOI: 10.1109/INFCOM.2011.5934929.

[205] X. Wang, Y. Hou, X. Huang, D. Li, X. Tao, and J. Xu. "Security Analysis of Key Extraction from Physical Measurements with Multiple Adversaries". In: *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*. May 2018, pp. 1–6. DOI: 10.1109/ICCW.2018.8403770.

[206] M. Wilhelm, I. Martinovic, and J. B. Schmitt. "Secure Key Generation in Sensor Networks Based on Frequency-Selective Channels". In: *IEEE Journal on Selected Areas in Communications* 31.9 (Sept. 2013), pp. 1779–1790. DOI: 10.1109/JSAC.2013.130911.

[207] R. D. Wilson and T. R. Martinez. "The General Inefficiency of Batch Training for Gradient Descent Learning". In: *Neural Networks* 16.10 (Dec. 2003), pp. 1429–1451. DOI: 10.1016/S0893-6080(03)00138-2.

[208] R. Wilson, D. Tse, and R. A. Scholtz. "Channel Identification: Secret Sharing Using Reciprocity in Ultrawideband Channels". In: *IEEE Transactions on Information Forensics and Security* 2.3 (Sept. 2007), pp. 364–375. DOI: 10.1109/TIFS.2007.902666.

[209] A. B. Wiltschko, G. J. Gage, and J. D. Berke. "Wavelet Filtering before Spike Detection Preserves Waveform Shape and Enhances Single-Unit Discrimination". In: *Journal of neuroscience methods* 173.1 (2008), pp. 34–40.

[210] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. M. Ni. "FILA: Fine-Grained Indoor Localization". In: *2012 Proceedings IEEE INFOCOM*. Mar. 2012, pp. 2210–2218. DOI: 10.1109/INFCOM.2012.6195606.

[211] Y. Wu, Q. Lin, H. Jia, M. Hassan, and W. Hu. "Auto-Key: Using Autoencoder to Speed Up Gait-Based Key Generation in Body Area Networks". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4.1 (Mar. 2020), 32:1–32:23. DOI: 10.1145/3381004.

[212] A. D. Wyner. "The Wire-Tap Channel". In: *Bell system technical journal* 54.8 (1975), pp. 1355–1387.

[213] W. Xi, Xiang-Yang Li, C. Qian, J. Han, S. Tang, J. Zhao, and K. Zhao. "KEEP: Fast Secret Key Extraction Protocol for D2D Communication". In: *2014 IEEE 22nd International Symposium of Quality of Service (IWQoS)*. May 2014, pp. 350–359. DOI: 10.1109/IWQoS.2014.6914340.

[214] Y. Xie, Z. Li, and M. Li. "Precise Power Delay Profiling with Commodity Wi-Fi". In: *IEEE Transactions on Mobile Computing* 18.6 (2018), pp. 1342–1355.

[215] W. Xu, S. Jha, and W. Hu. "Exploring the Feasibility of Physical Layer Key Generation for LoRaWAN". In: *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. Aug. 2018, pp. 231–236. DOI: 10.1109/TrustCom/BigDataSE.2018.00044.

[216] L. Yang, W. Wang, and Q. Zhang. "Secret from Muscle: Enabling Secure Pairing with Electromyography". In: *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*. SenSys '16. Stanford, CA, USA: Association for Computing Machinery, Nov. 2016, pp. 28–41. ISBN: 978-1-4503-4263-6. DOI: 10.1145/2994551.2994556.

[217] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao. "A Survey on Security and Privacy Issues in Internet-of-Things". In: *IEEE Internet of Things Journal* 4.5 (Oct. 2017), pp. 1250–1258. DOI: 10.1109/JIOT.2017.2694844.

[218] Z. Yang, Z. Zhou, and Y. Liu. "From RSSI to CSI: Indoor Localization via Channel Response". In: *ACM Computing Surveys (CSUR)* 46.2 (2013), pp. 1–32.

[219] S. Yasukawa, H. Iwai, and H. Sasaoka. "A Secret Key Agreement Scheme with Multi-Level Quantization and Parity Check Using Fluctuation of Radio Channel Property". In: *2008 IEEE International Symposium on Information Theory*. July 2008, pp. 732–736. DOI: 10.1109/ISIT.2008.4595083.

[220] S. Yasukawa, H. Iwai, and H. Sasaoka. "Adaptive Key Generation in Secret Key Agreement Scheme Based on the Channel Characteristics in OFDM". In: *2008 International Symposium on Information Theory and Its Applications*. Dec. 2008, pp. 1–6. DOI: 10.1109/ISITA.2008.4895646.

[221] C. Ye, A. Reznik, and Y. Shah. "Extracting Secrecy from Jointly Gaussian Random Variables". In: *2006 IEEE International Symposium on Information Theory*. July 2006, pp. 2593–2597. DOI: 10.1109/ISIT.2006.262101.

[222] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam. "Information-Theoretically Secret Key Generation for Fading Wireless Channels". In: *IEEE Transactions on Information Forensics and Security* 5.2 (June 2010), pp. 240–254. DOI: 10.1109/TIFS.2010.2043187.

[223] C. Ye, A. Reznik, G. Sternberg, and Y. Shah. "On the Secrecy Capabilities of ITU Channels". In: *2007 IEEE 66th Vehicular Technology Conference*. Sept. 2007, pp. 2030–2034. DOI: 10.1109/VETECF.2007.426.

[224] M. Yuliana, Wirawan, and Suwadi. "A Simple Secret Key Generation by Using a Combination of Pre-Processing Method with a Multilevel Quantization". In: *Entropy* (2019). DOI: 10.3390/e21020192.

[225] M. Yuliana, Wirawan, and Suwadi. "An Efficient Key Generation for the Internet of Things Based Synchronized Quantization". In: *Sensors* (2019). communication overhead
computation overhead. DOI: 10.3390/s19122674.

[226] M. Yuliana, Wirawan, and Suwadi. "Performance Improvement of Secret Key Generation Scheme in Wireless Indoor Environment". In: *Int. J. Commun. Networks Inf. Secur.* (2017).

[227] M. Zafer, D. Agrawal, and M. Srivatsa. "Limitations of Generating a Secret Key Using Wireless Fading Under Active Adversary". In: *IEEE/ACM Transactions on Networking* 20.5 (Oct. 2012), pp. 1440–1451. DOI: 10.1109/TNET.2012.2183146.

[228] C. Zenger. "Physical-Layer Security for the Internet of Things". PhD thesis. Ruhr Universität Bonn, 2017.

[229] C. T. Zenger, J. Zimmer, M. Pietersz, J.-F. Posielek, and C. Paar. "Exploiting the Physical Environment for Securing the Internet of Things". In: ACM Press, 2015, pp. 44–58. ISBN: 978-1-4503-3754-0. DOI: 10.1145/2841113.2841117.

[230] C. T. Zenger, A. Ambekar, F. Winzer, T. Pöppelmann, H. D. Schotten, and C. Paar. "Preventing Scaling of Successful Attacks: A Cross-Layer Security Architecture for Resource-Constrained Platforms". In: *Cryptography and Information Security in the Balkans*. Lecture Notes in Computer Science. Springer, Cham, Oct. 2014, pp. 103–120. ISBN: 978-3-319-21355-2. DOI: 10.1007/978-3-319-21356-9_8.

[231] C. T. Zenger, H. Vogt, J. Zimmer, A. Sezgin, and C. Paar. "The Passive Eavesdropper Affects My Channel: Secret-Key Rates under Real-World Conditions". In: *2016 IEEE Globecom Workshops (GC Wkshps)*. Dec. 2016, pp. 1–6. DOI: 10.1109/GLOCOMW.2016.7849064.

[232] F. Zhan and N. Yao. "On the Using of Discrete Wavelet Transform for Physical Layer Key Generation". In: *Ad Hoc Networks* 64 (Sept. 2017), pp. 22–31. DOI: 10.1016/j.adhoc.2017.06.003.

[233] J. Zhang, A. Marshall, R. Woods, and T. Q. Duong. "Secure Key Generation from OFDM Subcarriers' Channel Responses". In: *Globecom Workshops (GC Wkshps), 2014*. IEEE, 2014, pp. 1302–1307.

[234] J. Zhang, S. K. Kasera, and N. Patwari. "Mobility Assisted Secret Key Generation Using Wireless Link Signatures". In: *2010 Proceedings IEEE INFOCOM*. IEEE, 2010, pp. 1–5.

[235] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu. "Convolutional Neural Networks for Time Series Classification". In: *Journal of Systems Engineering and Electronics* 28.1 (Feb. 2017), pp. 162–169. DOI: 10.21629/JSEE.2017.01.18.

[236] H. Zhao, Y. Zhang, X. Huang, and Y. Xiang. "An Adaptive Physical Layer Key Extraction Scheme for Smart Homes". In: *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. Aug. 2019, pp. 499–506. DOI: 10.1109/TrustCom/BigDataSE.2019.00073.

[237] J. Zhao, W. Xi, J. Han, S. Tang, X. Li, Y. Liu, Y. Gong, and Z. Zhou. "Efficient and Secure Key Extraction Using CSI without Chasing down Errors". In: *arXiv preprint arXiv:1208.0688* (2012). arXiv: 1208.0688.

[238] X. Zhou, L. Song, and Y. Zhang. *Physical Layer Security in Wireless Communications*. CRC Press, Nov. 2013. ISBN: 978-1-4665-6700-9.

List of Tables

List of Figures