10-22-2015 12:00 AM

On Enhancements of Physical Layer Secret Key Generation and Its Application in Wireless Communication Systems

Kang Liu, *The University of Western Ontario*

Supervisor: Dr. Xianbin Wang, *The University of Western Ontario*
Joint Supervisor: Dr. Serguei Primak, *The University of Western Ontario*
A thesis submitted in partial fulfillment of the requirements for the Master of Engineering Science degree in Electrical and Computer Engineering

ON ENHANCEMENTS OF PHYSICAL LAYER SECRET KEY GENERATION AND ITS APPLICATION IN WIRELESS COMMUNICATION SYSTEMS

(Thesis format: Monograph)

by

Kang Liu

Graduate Program in Electrical and Computer Engineering

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Engineering Science

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

Abstract

As an alternative and appealing approach to providing information security in wireless communication systems, secret key generation at physical layer has demonstrated its potential in terms of efficiency and reliability over traditional cryptographic methods. Without the necessity of a management centre for key distribution or reliance on computational complexity, physical layer key generation protocols enable two wireless entities to extract identical and dynamic keys from the randomness of the wireless channels associated with them.

In this thesis, the reliability of secret key generation at the physical layer is examined in practical wireless channels with imperfect channel state information (CSI). Theoretical analyses are provided to relate key match rate with channel's signal-to-noise ratio (SNR), degrees of channel reciprocity, and iterations of information reconciliation.

In order to increase key match rate of physical layer secret key generation, improved schemes in the steps of channel estimation and sample quantization are proposed respectively. In the channel estimation step, multiple observations of the wireless channels are integrated with a linear processor to provide a synthesized and more accurate estimation of the wireless channel. In the sample quantization step, a magnitude based quantization method with two thresholds is proposed to quantize partial samples, where specific quantization areas are selected to reduce cross-over errors. Significant improvements in key match rate are proven for both schemes in theoretical analysis and numerical simulations. Key match rate can even achieve 100% in both schemes with the assistance of information reconciliation process.

In the end, a practical application of physical layer secret key generation is presented, where dynamic keys extracted from the wireless channels are utilized for securing secret data

transmission and providing efficient access control.

Keywords: Secret key generation, physical layer, wireless channel, security

Acknowledgements

I owe my gratitude to a great many people who have contributed to my graduate study and thesis, and because of whom my graduate experience has been one that I will cherish forever.

I would like to express my most sincere gratitude to both of my supervisors, Professor Xianbin Wang and Professor Serguei Primak. Without their support and guidance, it would not be possible for me to develop my research and complete this thesis. And also thanks to their understanding and encouragement so that I can balance between my research in campus and internship in industry. It's their motivation, patience and insight that helps me through my two years graduate life, and it will definitely benefit my following study and research throughout the future.

Many thanks to Dr. Hao Li and my other colleagues in the laboratory in the University of Western Ontario. It has been a wonderful two year that we were working together and having fun together. We are more like friends than colleagues after work, and I will always remember the exciting moments we spent as a group.

My special thanks goes to my beloved parents. If I will make any contributions to science and human society, it's their education and selfless loving care that makes me so. Thank you Mum and Dad, I will always love you.

Contents

List of Tables

List of Figures

List of Appendices

List of Abbreviations

ADC	Analogue-to-digital converter
AWGN	Additive white Gaussian noise
CDF	Cumulative distribution function
CIR	Channel Impulse Response
CSI	Channel State Information
FPGA	Field-programmable gate array
LDPC	Low-density parity-check code
LTE	Long-Term Evolution
MIMO	Multiple-input and multiple-output
MMSE	Minimum mean square error
OFDM	Orthogonal frequency-division multiplexing
PHY	Physical layer
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
SNR	Signal-to-noise ratio
TDD	Time-division duplexing
TLS	Transport Layer Security

USRP	Universal Software Radio Peripheral
UWB	Ultra-wideband
WiMAX	Worldwide Interoperability for Microwave Access

Chapter 1

Introduction

1.1 Research Motivations

Wireless communication is now playing a prominent role in both civil and military data transmission. However, due to the inherent openness of wireless media, wireless communication systems face particular security vulnerabilities, and one of the most rigorous challenges lies in providing information integrity, confidentiality and access authentication. Traditional security mechanisms, which are inherited from wired communications, rely on cryptography and hash functions at higher layers. However, in wireless communication systems, they require secret key distribution in a wireless scenario, which may lead to possible leakage of secret keys. At the same time, physical layer security protocols associating with wireless channels are now emerging to complement the traditional security mechanisms.

In recent years, physical layer secret key generation, which exploits the randomness of wireless channels to extract secret keys, has attracted considerable attention. Based on the principle of channel reciprocity, and time and space varying characteristics of wireless channels,

this type of approaches can overcome the typical challenge of key distribution and dynamically refresh secret keys without heavy computational overhead. Thus, physical layer key generation can potentially provide a low cost and yet effective alternative to higher layer approaches.

Secret keys generated at the physical layer are assumed to provide communication security with an information-theoretic guarantee. Relying on the reciprocity of wireless channels, channel response of the forward channel (from the transmitter to the receiver) is identical, in theory, to the channel response of the backward channel (from the receiver to the transmitter), and exactly the same keys should be extracted at both ends of the transmitter-receiver pair. However, imperfect channel reciprocity generally presents in practice. Due to the existence of diverse noise, interference, estimation errors, and other non-reciprocity factors at both sides, secret key generation in practical applications suffers from low key match rate and insufficient key generation rate. While so far, little effort has been devoted to enhancing the overall performance of physical layer secret key generation, especially under conditions with imperfect channel state information. Since two remote terminals only obtain correlated but nonidentical estimates of the wireless channels for secrecy extraction, key bit mismatch happens, and finally leads to the severe degradation of transmission reliability. Therefore, it is of great significance and priority to investigate secret key generation schemes with imperfect channel conditions, as well as corresponding enhancing techniques for secret key generation.

1.2 Research Objectives

Much research in the literature discusses secret key generation at the physical layer from different aspects, including various random sources for secrecy extraction, protocol improvements

and its implementation in different systems. The objectives of this thesis are to provide insight of key generation reliability under practical channel conditions, increase key match rate from different procedures of key generation protocols and propose a practical application of physical layer secret key generation.

First, since in practical implementations secret keys are generated with limitations from estimation errors and channel non-reciprocity, the first objective of this thesis is to analyse the reliability of secret key generation with imperfect channel state information, discover the relations between probability of key bit mismatch and practical channel conditions such as SNR and degrees of channel reciprocity, and investigate the effect of information reconciliation on key match rate.

Second, as most of the existing key generation algorithms focus on key bit extraction from different random sources and implementation systems, the topic of increasing key match rate in key generation protocols requires much more attention. Therefore, our second objective is to explore current key generation protocols and develop new schemes regarding different procedures to improve key match rate in wireless communication systems.

Third, with the ultimate mission of contributing theoretical proposal in engineering domain to practical applications in human society, the third objective of this thesis is to facilitate existing security applications with physical layer secret key generation, and provide reliability and efficiency in security aspects with the advantages of secret key generation.

1.3 Research Contributions

The main contributions of the thesis are summarized as follows:

- We provide a general review of physical layer security. Significant theory development and security protocols are discussed with their strength and weakness. A literature survey is also conducted from different aspects of secret key generation at the physical layer.

- Theoretical insights are cast into secret key generation using physical channels with imperfect CSI. The reliability of secret key generation is analysed in practical channels with estimation requirements, and mathematical expressions are derived to relate channel SNRs, degrees of channel reciprocity and information reconciliation to the probability of key bit mismatch.

- An improved key generation protocol with multiple observations of wireless channels is proposed to increase key match rate. A linear processor is calculated and utilized to combine multiple observations of the wireless channels on both sides, and helps to obtain a synthesized and more accurate channel estimation for secret key generation.

- The key match rate is further improved by employing a quantization method with two thresholds in secret key generation. Magnitude based quantization with a dead region is proposed and only partial samples are quantized to reduce cross-over errors. The trade-off between key match rate and key generation rate in low SNR region is also analysed with this scheme.

- Practical application of secure data transmission is presented with the facilitation of physical layer secret key generation. Dynamic secret keys are generated to provide secure data transmission and reliable access control in wireless networks. A demonstration app for secure data transmission is also developed on iOS devices.

1.4 Thesis Outline

The rest of the thesis is organized as follows:

- **Chapter 2** describes the background of physical layer secret key generation. A general review on physical layer security is provided in the first section. From Shannon's perfect secrecy with his proposal of one-time pad cryptography, to information-theoretic security brought up by Wyner, till the most recent physical layer secret key generation, milestones of research development and significant proposals are addressed in each stage. A literature survey is conducted on the most appealing approach of secret key generation at physical layer, and research aspects of key extraction from different random sources, protocol improvements and implementation systems are all addressed. In the last section of chapter 2, parameters for performance evaluations are also discussed for physical layer secret key generation.

- **Chapter 3** analyses secret key generation using physical channels with imperfect CSI. In the system model, wireless channel model with imperfect CSI is first provided, secret key generation rate and channel estimation are also discussed in this scenario. Essential procedure of secret key reconciliation and the adversary model are briefly addressed. In the following section, theoretical analysis of bit error rate, probability of raw key mismatch and probabilities of key mismatch after reconciliation are provided with mathematical insight. A simple example of secret key generation with simulation results is presented in the last section of this chapter.

- **Chapter 4** discusses the basic procedures of physical layer secret key generation and

proposes an improved protocol with multiple observations of wireless channels to increase key match rate. Mathematical analyses of three procedures, channel estimation, sample quantization and key reconciliation, are provided. Simulation results for the general protocol are also presented. An improved key generation protocol with multiple observations of wireless channels is proposed in the next section. Simulation results of the proposed scheme are also compared with the performance of the general secret key generation protocol.

- **Chapter 5** further improves key match rate by proposing a key generation protocol with two thresholds quantization, where the magnitude based quantization has a dead region without key bits extracted. Simulation results demonstrate the improved key match rate with two thresholds quantization, and the trade-off between key generation rate and key match rate. In the end, a practical application of secure data transmission with secret key generation is provided in the scenario of mobile computing. Secret key generation is utilized for secure data transmission and provides efficient access control. A demonstration app on iOS devices is also presented.

- **Chapter 6** summarizes the thesis and discusses the future research topics in the area of physical layer secret key generation.

Chapter 2

Background

2.1 Physical Layer Security: Review

2.1.1 From Shannon's Perfect Secrecy

Shannon built the theoretical foundation of cryptography in [1], where his one-time pad gave an example of perfect secrecy. He proved that, mathematically, the priori probability of a plaintext message is the same as the posteriori probability of the plaintext message conditioned on the corresponding cipher text. In other words, as long as a secret key that is at least as large as the plaintext is shared by both users, the cipher text could be made independently from the plaintext, and perfect secrecy could be achieved with nothing revealed about the source information, even if the eavesdropper has his own observation. Unfortunately, one of the main drawbacks of this solution is the safe distribution of a sufficient large secret key, and the condition itself is impractical.

Due to the difficulty of secret key distribution, most of the existing cryptographic methods

lay the system security on the computational complexity of a certain mathematical problem. The computational hardness can to a large extent guarantee the secrecy of the information. The most classical and computational security based solution is the Diffie-Hellman [2] algorithm.

While most of these traditional security protocols employing public or private keys are implemented at the upper layers (data link layer, network layer, transport layer and application layer), they are designed separately from the physical layer where the cryptography and communications are actually executed, especially in wireless communication systems. However, physical characteristics of the communication channels could be exploited to strengthen the system security in an alternative way.

2.1.2 Information-theoretic Security

Starting in the 1970s, researchers followed up Shannon's work by exploring the information-theoretic security in wireless communication systems and characterized the theoretic limit of secure transmission over wireless channels. In particular, the concept of wire-tap channel brought by Wyner [3] has received considerable attentions. Wyner's work was later formalized by Csiszar in [4].

In this line of research, information theorists noted that perfect secrecy could be achieved by exploring advantages of wireless channels. For instance, if both legitimate users experience a better channel with higher SNR than the eavesdropper, there is non-zero secrecy capacity that allows finite information to be exchanged over the wireless channel without any information leakage to the eavesdropper [5, 6]. Further research indicated that even if the adversary dominates in the channel SNR, legitimate users can still ensure secure communication by utilizing

opportunistic signalling to create an effective SNR advantage. As a whole, given statistics of the communication channels that link between two legitimate users, and between legitimate users and the eavesdropper, one can use well defined codes and obtain secure transmission of the messages in a key-less way.

This kind of information-theoretic security is achieved without implementation of traditional cryptographic keys, which apparently avoids the problem of secret key distribution. However, there is always requirement of certain bound on the eavesdropper's channel quality in advantage based security protocols, which sometimes is an unmet condition.

2.1.3 PHY Based Key Generation

Later, another direction of physical layer security, first by Maurer [7] and almost simultaneously by Ahlswede and Csiszr [8], proved that secret and identical keys could be extracted for data encryption from the characteristic randomness of the wireless channels between two legitimate users. Certain public discussions were allowed over the channel, even with the presence of an eavesdropper. Additional fundamental research exploring wireless channels and systems to generate secret keys at the physical layer can be seen in [9–11], etc. The generated keys resulting from these protocols then could be utilized in a traditional way of symmetric key cryptography such as Shannon's one-time pad. However, different from Shannon's approach, both keys are generated directly by the pair of users, such that no further consideration of key distribution is required.

2.1.3.1 Source Model and Channel Model

Basically these research papers can be classified into two categories based on their models for secret key generation. One is the channel model, and the other is the source model. In the channel model, two parties *Alice* and *Bob*, both transmit common randomness information to each other, and apply reconciliation and privacy distillation to obtain identical keys. In the source model, both users observe and estimate a random process of the wireless channel between them, and followed by information reconciliation and privacy amplification over the public channel to generate the same keys. The observations of the random process from the legitimate users is distinct from the eavesdropper's observation, which ensures the secrecy of the shared key. A practical protocol due to Bloch et al. [6] utilized the channel model to generate secret keys. However, compared to source model based key generation, approaches based on channel model have some basic demands from *Alice* and *Bob*, that both users should be aware of the channel state information (CSI) of their own channels as well as eavesdropper's channel.

2.1.3.2 Channel Reciprocity

In all symmetric secret key generation protocols, the most significant and fundamental requirement of the schemes is the identicalness of the shared keys. Generally, in a time-division duplex (TDD) wireless communication system, such as 802.11, 802.16 (WiMAX), and LTE, the forward and backward channels are identical due to the property of channel reciprocity, and this characteristic provides potential possibility of identical secret key generation. Research exploring channel reciprocity to extract shared keys can be seen as early as in [12].

Channel reciprocity based key generation for the source model dominates in several ways than other approaches with channel model: (1) The near reciprocal channel observations in practical systems alleviate the reliability on following information reconciliation and privacy amplification, and improve the time and energy efficiency of secret key generation, (2) No fore-knowledge of the channel state information of the eavesdropper's channel is required, which lowers the complexity of the security protocol without any compromise of achieving perfect secrecy, and (3) *Alice* and *Bob* never need to transmit CSI as part of the protocol, allowing full use of the channel and exploiting extra randomness for key extraction.

However, one must realize that although the radio channels are reciprocal, the observations of the radio channels are not. A couple of facts contribute to this situation. Firstly, each received signal is inevitably corrupted by additive noise, and the measurements of these signals can't be exempted. Secondly, the hardwares of transceivers on both sides are nonidentical, which will affect the transmitting and receiving signals in different ways. Thirdly, additional interference may exist in the scenario, which can't be totally symmetric. Lastly, the channel estimation process made by both users are typically not simultaneous, while the wireless channels associated between them might change within this period.

2.1.3.3 Secrecy against Eavesdropper

On one hand, due to fluctuations and variations of channel states, secret keys can be generated dynamically as the channel evolves [12–14]. In addition, both users' random movement could results in different observations of the channel, which also leads to ever changing keys. As long as secret keys are generated dynamically with a certain length, it can efficiently resist the brute-force attacks from the eavesdropper. On the other hand, eavesdropper's estimation

of the channel is bound to be independent from the legitimate users' estimation owing to the rapid fading or multipath characteristics of the wireless channels. That is to say, the common randomness of evolving CSI for secret key generation [15, 16] can only be observed by the legitimate transceivers, but not the eavesdropper, which ensures the secrecy of extracted keys.

2.1.4 Remarks on Different Security Protocols

While several methods were developed to reduce probability of information interception [17, 18], the most widely used technique in information security is cryptographic methods, where plaintext messages are encrypted with ciphers before transmitting via the public channel. Both symmetric and asymmetric key based cryptography exist in this domain. As shown in Fig. 2.1, symmetric key encryption requires the secure distribution of a shared key between two legitimate users in advance [19], however, the key distribution itself remains a difficult problem in practice. An alternative cryptographic method as illustrated in Fig. 2.2 utilizes public keys to avoid key distribution, and pairs of asymmetric keys are applied by the users.

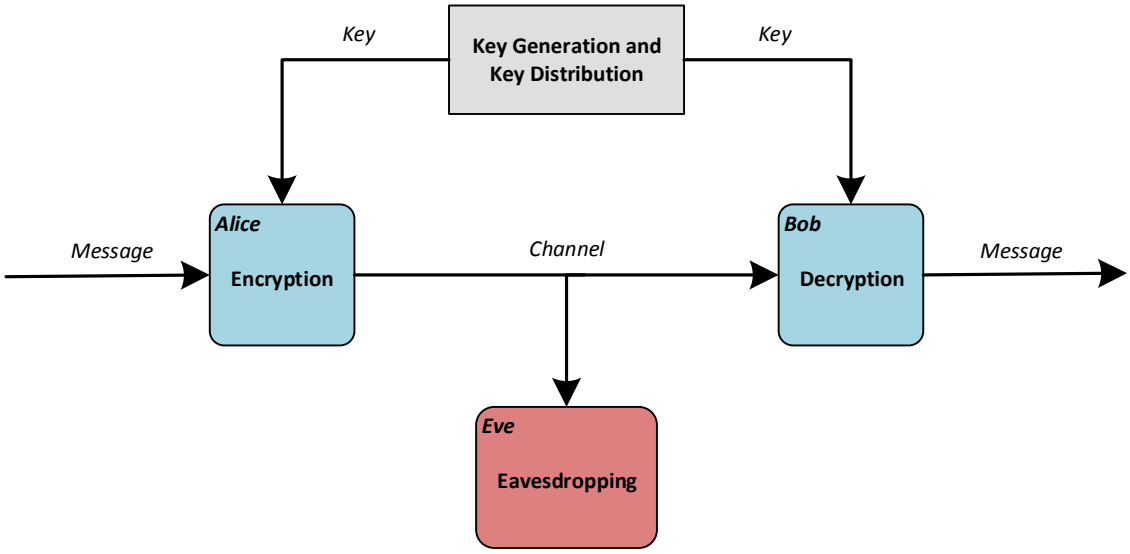

Figure 2.1: Illustration of symmetric encryption

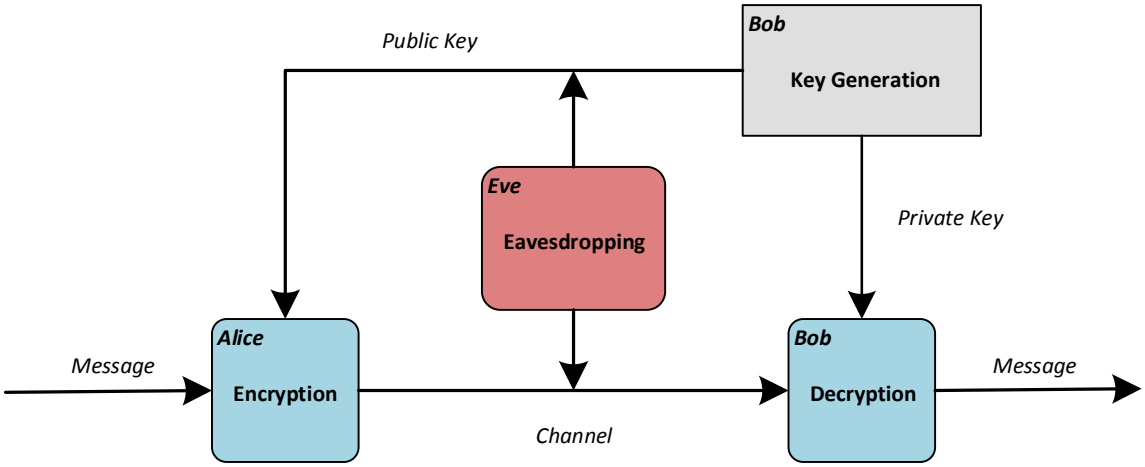

Figure 2.2: Illustration of asymmetric encryption

Generally, classical cryptographic techniques such as Diffie-Hellman build the system security on the computational hardness of mathematical problems, such as the discrete logarithm problem. While later proposed information-theoretic security assumes no bound on the available resources of the adversary, but it can still achieve perfect secrecy.

Most information-theoretic methods take advantage of the wireless channels to provide communication security, however, it's less practical than physical layer secret key generation protocols, since the latter only requires the channel independence other than channel advantages.

Along with the development of physical layer security, there has been discussions about the reliability and efficiency of secret key generation based on wireless channel characteristics [20–22]. Although observations of the physical randomness can not always provide identical keys each time even with additional reconciliation, we argue that such degradation is acceptable as long as the protocols can generate sufficiently long keys with time and energy efficiency. And the physical layer secret key generation can function as an alternative to the traditional cryptographic methods.

2.2 PHY Secret Key Generation: Literature Survey

So far much research effort has been devoted into the area of physical layer secret key generation. Some research papers focus on the wireless random sources for secret key extraction, some provide theoretical improvements for key generation protocols, and some apply these security algorithms into practical wireless systems.

2.2.1 Key Generation from Different Random Sources

Temporal and spatial variations of wireless channels can provide high probability for secret key generation, and different kinds of channel characteristics of wireless channels can be utilized as random sources to generate secret bits. The channel state information, magnitude, phase, joint magnitude and phase information, all are widely employed in secret key extraction based on the reciprocity of wireless channels [9–11, 13, 14, 23–32].

Mathur et al. [31] exploit the channel impulse response (CIR) to generate secret bits by a FPGA board in a real indoor environment, they achieve at most one secret bit per second in practical scenarios. Wilson et al. [11] utilize the CIR of rich multipath wireless channels in ultra wideband (UWB) systems for key generation.

Apart from CIR based key extraction, another intuitive way to generate secret bits is to quantize different channel coefficients themselves. Received Signal Strength (RSS) can be easily measured by most mobile devices and existing wireless infrastructures, and thus is widely used in many secret key generation protocols as a random source [33, 34]. RSS based methods such as [34] utilize a 802.11 board to collect the temporal and spatial variations of the channel to extract secret bits in both static and mobile situations. The moving velocity can provide ex-

tra randomness for secret key generation. However, as RSS can only reflect rough information of the wireless channel, the secret key generation rate cannot always meet the expectations for practical application.

Other than RSS based protocols, the phase reciprocity of wireless channels is also utilized for secret key extraction [9, 10]. In [14], the phase variations of each sub-channel is explored in a multipath OFDM communication system. Wang et al. [35] propose a scalable phase-based secret key generation with initial phase estimation of a random pilot sequence. The proposed scheme improves the key generation rate in a narrowband system. However, most of these protocols are less likely to be deployed in practical communication systems since the phase parameter of wireless channel is hard to estimate with the effect of time offset and frequency synchronization problems.

Tope et al. [36] utilize the characteristic of received signal's envelope to generate identical keys between both users, and in [24] an envelope detector is designed in the key generation platform.

The frequency selectivity of channel fading is also feasible to generate secret bits in a static wireless sensor network as shown in [37]. In [26], the deep fades caused by multipath wireless channels is considered as the random source for key extraction. Similarly, another secret key generation from deep fades of correlated observations is described in [23]. However, frequency selectivity based secret key generation rate cannot always be guaranteed in practical situations.

In [38], multiple antenna diversity is utilized to increase key generation rate within the coherence time of wireless channels. In [5], space-time technique in multiple antenna systems is introduced to generate secret keys. Since only spatial diversity is considered, improvement on key generation rate is extremely limited by the number of antennas.

Other random sources, such as the delay profiles of ultra wideband (UWB) channels are processed to generate key bits in [27]. Jointly Gaussian random variables are also utilized to extract secret keys as shown in [28–30].

There are also key generation protocols in which secret keys are not generated from channel randomness. In [39], the randomness of level crossing process is utilized for key extraction. In [6], the secret keys are pseudorandom sequences generated by users without exploiting the randomness of the wireless channels.

2.2.2 Key Generation Protocol Improvements

Secret key generation protocols are composed of several procedures, including channel estimation, sample quantization, information reconciliation and privacy amplification. Much effort has been devoted by different researchers into the improvement of key generation algorithms, with emphases on different procedures.

In the channel estimation stage, mutual information between two channel estimates on both sides are denoted as the upper bound on the number of extracted key bits per channel estimation sample, and is presented in the case of jointly Gaussian channels in [16, 28, 29]. Regarding correlated Gaussian channels, it could be the case where wireless communications experience Rayleigh fading. The mutual information and channel estimation process are also employed in the cases of MIMO channels [30] and UWB channels [11].

Research has also been conducted to utilize multiple measurements of the channels to obtain sufficient long keys for cryptography. Aono et al. in [13] use multiple beam patterns to obtain multiple observations. In [37, 40] different frequencies contribute to the multiple mea-

surements. Multiple antenna diversity [38] has also been proposed to extract arbitrarily long keys.

To reduce key bit errors, channel quantization method with a guard band is presented in [30]. An improved quantization scheme considering level crossing errors quantizes consecutive fading channels with a guard band in [31]. Adaptive quantization of channel estimates with noise corruption is investigated in [41] and [33].

In early research regarding key extraction, secret keys are generated from reciprocal channels without the use of a public channel to reconcile different bits. In [9], the phase difference of two orthogonal sinusoids in a received signal is quantized to generate the raw key, and a coding scheme is applied later to improve key match rate.

Since practical situations are often affected by noise, synchronization offset and channel estimation errors, key generation protocols with simple channel estimation and quantization usually have poor key match rate. To make the protocols more reliable and efficient, principles of information reconciliation and privacy amplification based on the discussions over a public channel are considered to improve key match rate [39, 42, 43].

The reconciliation procedures are basically aimed to correct the discrepancies of the raw key bits between two terminals. A common way to apply information reconciliation is to formulate the procedure as a problem of Slepian-Wolf lossless compress coding [6, 28–30, 44, 45]. Research in [13,23–29,32] all utilize the reconciliation procedure to correct raw key errors after direct channel sample quantization.

Some reconciliation proposals by Bloch et al. [6] and Ye et al. [39] strongly rely their error correcting ability on the utilization of low-density parity-check (LDPC) codes, and exploit the correlation between channel quantization samples on both sides to correct errors. However,

both system complexity and memory requirement for LDPC coding scheme are too high, which makes it unsuitable to be deployed in mobile devices. Other similar work such as [16] uses nested lattice codes and vector quantization for information reconciliation.

Some other research suggests an alternative to information reconciliation, Sayeed et al. [14] abandon the process of information reconciliation, but regenerate the secret keys as long as the disagreement bits are discovered. The energy trade-off between secret key regeneration and transmitting power with different SNR is explored.

Along with information quantization, privacy amplification is utilized in [6, 46] to distil secret keys available for practical cryptographic applications.

2.2.3 Key Generation Implementation Systems

Notably, many experiments and implementations of secret key generation protocols have been deployed in different scenarios. In [14] the randomness of channel phase is proposed for secret key extraction in OFDM systems. In [29] the channel impulse responses of ultra wideband radios are measured. While in [11], channel estimations are conducted in cellular environment. An implementation in [40] transmits and receives multi-carrier signals with the GNU software radio and universal software radio peripheral (USRP). In [31] two off-the-shelf 802.11a wireless devices are used to generate secret key bits and achieves key generation rate around 1 bit per second. In addition, Zigbee techniques are also utilized for key generation. In [13], the Zigbee radio hardware along with steerable directional antennas are presented to generate secret bits between two users.

2.3 PHY Secret Key Generation: Performance Evaluation

In evaluation of the reliability, efficiency, security and feasibility of physical layer secret key generation protocols, some parameters are analysed in terms of key match rate, key generation rate, key bit randomness and implementation complexity.

2.3.1 Reliability: Key Match Rate

Key match rate refers to the portion of successful generation of two identical keys in a given number of trials. For a set of identical keys, it requires that every single bit in two keys should be exactly the same in both value and order correspondingly. In this sense, key match rate shows higher and practical standards as an evaluation parameter than key bit agreement probability for key generation schemes. By the latter it means the probability of key bit agreement between two generated keys in one trial of key extraction. Apparently, as long as the key bit agreement probability is not 100% in one iteration of key generation, it adds no contribution to the key match rate, and the generated keys are unable to be utilized for upper layer encryption. From this perspective of view, only key match rate can truly reflect the reliability of certain key generation protocols in providing identical keys for applications. However, in some cases, the probability of raw key bit agreement is also meaningful in suggestion of later effort for key reconciliation. Relatively low probability of raw key bit agreement reduces the required resources for key generation protocols.

Usually, wireless channels in stationary scenarios could be more easily effected by environment noises, which will greatly undermine the reciprocity of channel estimations. With less reciprocal information shared between two users, their generated keys will certainly have lower

probability of raw key bit agreement. It is shown by experiments in [34] that the probability of key bit agreement is determined by the variations of the wireless channels in RSS based schemes. In [38] the utilization of multiple antennas improves the key bit agreement and key match rate by providing extra reciprocal information for key extraction.

2.3.2 Efficiency: Key Generation Rate

Key generation rate is a significant parameter in reflecting the efficiency of key generation protocols. Due to low level-crossing rate of Rayleigh fading channels, and bit cross-level errors in quantization, usually only one bit key can be extracted out of a consecutive measurements of the wireless channels. And that's the reason for low key generation rate in RSS based key generation methods.

One possible way to facilitate key generation rate is to oversample the channel estimations to make best use of every single measurement. However, high correlated estimates will result in low bit entropy, and thus low key bit randomness [31]. Again, an example of enhanced RSS based key generation protocol in [38], utilizes multiple antennas to achieve four times faster key generation rate than basic protocols. In [33], the technique of signal processing helps to achieve key generation rate as high as 22 bits per second.

2.3.3 Security: Key Bit Randomness

A high key bit randomness will greatly increase the complexity of brute-force attacks by adversaries, and key bit randomness evaluates the security level of such secret key generation protocols. A statistical test suite for random and pseudorandom number generators for cryp-

tographic applications is suggested by National Institute of Standards and Technology in [47]. Based on its decision rule, if the P-value is lower than 1%, then the generated keys is insufficient in randomness to serve as a cryptographic key.

Usually there is a trade-off for key generation rate and key bit randomness in RSS based key generation schemes. As we mentioned before, the desire for higher key generation rate leads to higher sampling rate of channel measurements. While at the same time, oversampled channel estimates share little entropy, and results in less key bit randomness. However, if the randomness of channel phase is employed to generate secret bits instead of received signal strength, such a constraint will be non-existent [35]. Since the randomness of initial phase can always be utilized for key extraction, even though the wireless channels remain constant.

2.3.4 Feasibility: Implementation Complexity

The last parameter in evaluation of key generation protocols is the implementation complexity of such schemes. As we know, the received signal strength is quite easy to obtain with almost every mobile devices or wireless infrastructures nowadays with a valid wireless card. However, for channel phase based key generation schemes, more complicated hardware will be needed. In order to estimate the phase of wireless channels, an analogue-to-digital converter (ADC) is required to work at Nyquist frequency of a single-tone carrier [48]. And the operating frequency of the wireless system will also ask for more necessary hardware.

2.4 Chapter Summary

In this chapter, a general review on the development of physical layer security is provided. Starting from Shannon's theory of perfect secrecy, to information-theoretic security, till most recent physical layer secret key generation, fundamental theories and significant proposals are addressed in each stage. Especially, PHY based key generation are classified into two categories, named source model and channel model. Channel reciprocity, the theoretical foundation of physical layer secret key generation, is also discussed. Secrecy against eavesdroppers in this scenario is also analysed from a theoretical point of view. Remarks on different security schemes within each stage are presented at the end of section.

Regarding the most appealing and up-to-date security protocols, a literature survey of secret key generation at physical layer is conducted from three aspects. Key generation from different random sources, protocol improvements and application scenarios are surveyed.

In the end, the most commonly used parameters for performance evaluations are introduced. The reliability, efficiency, security and feasibility of physical layer secret key generation protocols are reflected by parameters including key match rate, key generation rate, key bit randomness and implementation complexity, respectively.

Chapter 3

Secret Key Generation Using Physical Channels with Imperfect CSI

3.1 Introduction

One of the main tasks of communication networks is to assure the secrecy of messages transmitted through the network. This task is particularly important for wireless networks, where emitted signals could be easily intercepted and analysed by external parties. However, at the same time, the random nature of wireless channels could also lend itself to providing security of communications by a number of possible means. On one hand, one can use beamforming properties of antenna arrays to minimize the power intercepted by the eavesdropper. On the other hand, from the perspective of information security, using jammers in addition to legitimate transmitters will also allow to deliver information securely. Furthermore, using channel impulse response as a source of common randomness to provide extra security is also an appealing direction in prospective wireless communication systems. The fact that wireless chan-

nels are space varying allows to use the Channel State Information (CSI) to provide additional secrecy by means of secret key generation [49]. The latter approach is based on obtaining secret keys by direct sampling of the channel impulse response. For example, the instantaneous power [49], RSSI [50], phase [51] and full complex response [52] all could be used for this purpose.

In secret key generation approaches, since sampling and quantization are applied to slightly different channels by both transmitters, there is a non-zero probability that some secret bits are different on both sides. Therefore, some *reconciliation* is required by means of communication via a public channel. One of such procedures, CASCADE, is suggested in [53]. In this algorithm, *Alice* performs random permutation of the secret key bit stream, followed by division into small blocks. The permutation sequence and parity information of each block are then sent to *Bob* via the public channel. *Bob*, after performing the same permutation, block division and parity checking, performs a binary search on the block in attempt to recover parity. These steps are iterated to increase the probability of matching two keys.

The best condition for generating secret key bits is to have independent channel samples. However, in practical situations, this may not be a case due to low fading rate. Thus, in practical cases some additional measures have to be taken to further randomize the values extracted from the channel either by using universal hash functions [49] (privacy amplification) or by undersampling the channel.

While some theoretical and experimental work has demonstrated its potential, a number of issues remain unexplored. This chapter is concerned with the reliability of secret key generation in practical channels with estimation requirements. The influence of quality of channel estimation, asymmetry and partial loss of reciprocity on achievable key generation rate as well as

on the amount of information recovered by the eavesdropper are both investigated. We provide analytical expressions which relate the channel SNRs and degree of reciprocity to probability of key mismatch, as well as give some information theoretical insight into the amount of key rate reduction due to correlation.

This chapter is organized as follows. In Section 3.2 we describe the channel model, measurement of estimation quality and non-reciprocity of generated keys. Information-theoretical limit of secure key generation rate is also provided. In the following Section 3.3, we examine a simple quantization scheme based on binary quantization of the received power, and evaluate the probability of bit errors between transmitter and receiver, as well as effectiveness of the reconciliation stage. In Section 3.4, we provide an basic example of secret key generation with some numerical and simulation results. Finally, we give the conclusions.

3.2 System Model

3.2.1 Channel Model with Imperfect CSI

We assume that transceivers A and B (as shown in Fig. 3.1) are connected via zero mean, circularly symmetric complex Gaussian channels h_{AB} and h_{BA} with the same variance σ_h^2. Due to inherent discrepancies between the hardware of both sides and differences in time of access and carrier frequency, all non-reciprocity factors are assumed to be absorbed into the correlation coefficient ρ between the channels and the different SNR γ_{AB} and γ_{BA} experienced by the transceivers. In other words, differences in time of access and carrier frequency are reflected by ρ and the power imbalance is shown by SNR γ_{AB} and γ_{BA}. With these assumptions, we have

the following relation between the forward and reverse channels

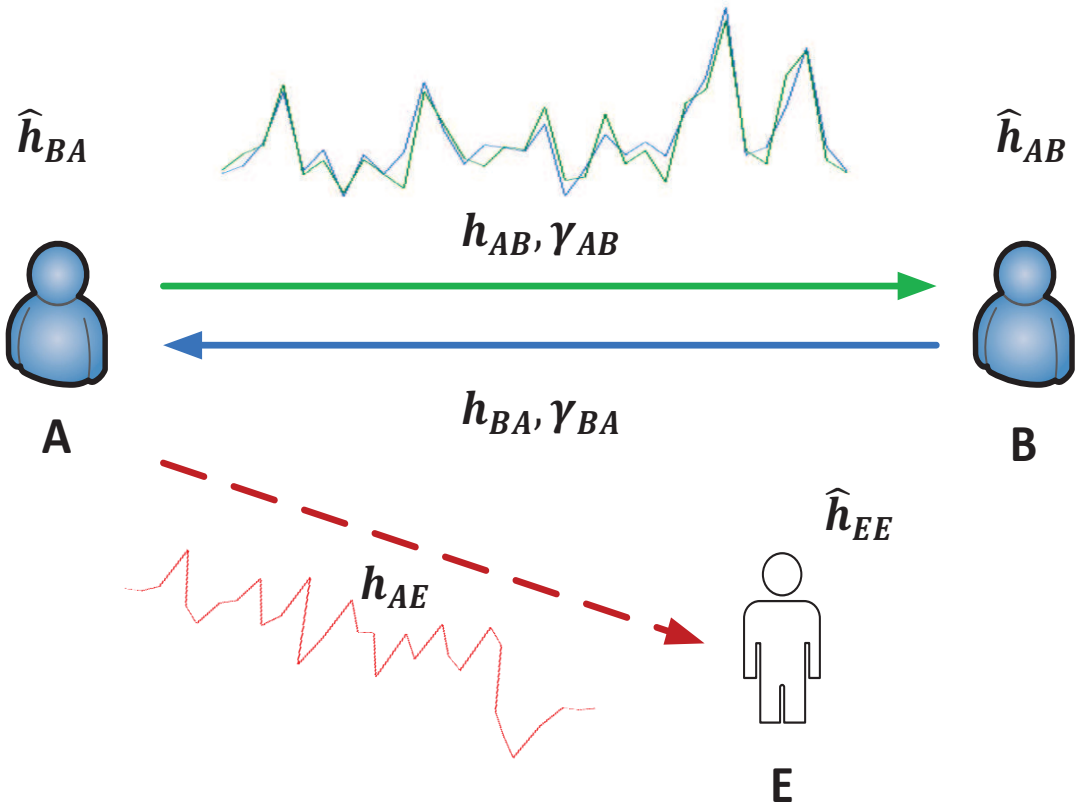

Figure 3.1: Illustration of secret key generation based on channel reciprocity

$$h_{BA} = \rho h_{AB} + \sqrt{1 - |\rho|^2}\sigma_h \xi \tag{3.1}$$

where ξ is additive white Gaussian noise (AWGN) with unity variance. In the case of perfectly reciprocal channels, $\rho = 1$ and γ_{AB}=γ_{BA}=γ.

Both transceivers A and B estimate the channels by sending pilots and performing minimum mean square error (MMSE) estimation. The estimates of the channels can be modelled as

$$\hat{h} = \frac{\gamma}{1+\gamma} h + \frac{\sqrt{\gamma}\sigma_h}{1+\gamma} \eta \tag{3.2}$$

where η is zero mean unit variance Gaussian noise. The variance of the estimates $\hat{h}$ is given by

$$\mathcal{E}\left\{|\hat{h}|^2\right\} = \frac{\gamma}{1+\gamma}\sigma_h^2 \tag{3.3}$$

Thus, the correlation between two estimates $\hat{h}_{AB}$ and $\hat{h}_{BA}$ can be found as

$$\tilde{\rho} = \frac{\mathcal{E}\left\{\hat{h}_{AB}\hat{h}_{BA}^*\right\}}{\sqrt{\mathcal{E}\left\{|\hat{h}|_{AB}^2\right\}\mathcal{E}\left\{|\hat{h}|_{BA}^2\right\}}} = \sqrt{\frac{\gamma_{AB}}{1+\gamma_{AB}}}\sqrt{\frac{\gamma_{BA}}{1+\gamma_{BA}}}\rho \tag{3.4}$$

3.2.2 Secret Key Generation Rate

According to jointly two dimensional distribution of two correlated Gaussian vectors, expressed in Appendix A.1, mutual information $I(\hat{h}_{AB}, \hat{h}_{BA})$, and thus the secrecy rate, can be found as

$$C_s = I(\hat{h}_{AB}, \hat{h}_{BA}) = -\frac{1}{2}\ln\left(1 - \frac{\gamma_{AB}}{1+\gamma_{AB}} \cdot \frac{\gamma_{BA}}{1+\gamma_{BA}}|\rho|^2\right) \tag{3.5}$$

It is obvious by inspection that either of the SNR γ_{AB}, γ_{BA}, or the correlation coefficient ρ approaches zero, the secrecy rate $C_s \longrightarrow 0$.

The derivations above only provide the secrecy rate for a single independent sample of the channel. However, due to random velocity of mobile devices, the correlation intervals of the channel impulse responses often vary in a significant range, and especially for cases of high carrier frequency. At the same time, the correlation of channel samples is highly undesirable for secret key generation and must be removed.

Generally a standard practice is to apply hash functions to already digitized and sampled channel impulse responses. We can utilize the idea of stochastic degrees of freedom [54],

to estimate the number of independent samples that could be extracted from a process with covariance function $\rho(\tau)$. In this case, it is assumed that the channel pilots, used for estimation and key extraction, are produced at the rate of $F_s = 1/T_s$ samples per second. Given $\rho(\tau)$ and an arbitrary number N, a $N \times N$ correlation matrix $\mathbf{R}_N$ can be defined such that $\mathbf{R}_N(i, j) = \rho((j - i)T_s$. The number of independent samples N_I then could be defined as (3.6).

$$N_I = \frac{\mathrm{tr}^2 \mathbf{R}_N}{\mathrm{tr}\left(\mathbf{R}_N \mathbf{R}_N^H\right)} \tag{3.6}$$

Finally, combining (3.5) and (3.6), the secrecy rate of key extraction from partially reciprocal channels can be obtained as

$$R_s = \frac{N_I}{N} C_s \tag{3.7}$$

Thus, it is approved that the secret key generation rate depends on the channel condition (SNR), fading rate and the shape of power spectrum.

3.2.3 Channel Estimation

While it is common to distinguish between interpolation, estimation, and prediction based on the location of the symbols of interest [55], in the following we use the term estimation for all three cases without loss of generality. The minimum mean square error (MMSE) estimator for the model above is linear [55] due to its optimality for underlying Gaussian process.

The frequency flat fading channel is modelled to be a complex zero mean circularly symmetric Gaussian random process $h(t)$ with covariance function $R(\tau) = \sigma_h^2 \rho(\tau)$, $\rho(0) = 1$. The

received signal $r(t)$ is given by

$$r(t) = h(t)s(t) + \xi(t),\ t = nT_s,\ n \in \mathcal{N}^+ \tag{3.8}$$

Here, T_s is the symbol duration and $s(nT_s)$ represents the signal (pilot) transmitted during the n-th time slot. The additive white Gaussian noise (AWGN) $\xi(t)$ has variance σ_n^2. The average SNR of pilot symbols is $\bar{\gamma}$ and the energy of the symbol pilot is then $E_p = \bar{\gamma} T_s \sigma_n^2$. The estimate of the channel $\hat{h}(t)$ is obtained by sending a sequence of predefined symbols, the pilots, with the following linear estimation

$$\hat{h}(t) = \sum_{l=-L_1}^{L_2} \alpha_l \left[\sqrt{\frac{E_p}{T_s}} h(lNT_s) + \xi(lNT_s) \right] \tag{3.9}$$

Here N is the length of a frame, *i.e.* the number of data symbols between two sequential pilot symbols. L_2 and L_1 are the numbers of blocks forward and backward used for estimation. The filter coefficient α_l obeys the Wiener-Hopf equation [55], which, in turn, is defined by the covariance function $\rho(\tau)$ of the channel.

3.2.4 Reconciliation of Keys

Since there would always be mismatch between the secret bits obtained on both sides of a legitimate link, some sort of reconciliation procedure is required. For example, the well known Maurer's approach as suggested in [7] contains information reconciliation and privacy amplification.

The information reconciliation [56] procedure aims to reconcile the different bits between

Alice and *Bob* through the public channel so that they can obtain identical keys. Owing to the fundamental and practical requirements for high secret key generation rate, as well as the confidentiality of the secret key bits, the entropy of the random source for key extraction must be maximized, while the amount of information exchanged for key reconciliation via the public channel must be minimized. This suggests an innate connection between the information reconciliation procedure and SlepianWolf data compression. This connection was analysed in the general setting of multi-terminal secret key generation in [57]. Moreover, in consideration of the duality between SlepianWolf data compression and channel coding (e.g., [58–62], etc.), we could establish the relation between information reconciliation and channel coding, such that available channel codes, such as Turbo codes or low-density parity check (LDPC) codes, could be utilized to reconcile the discrepancies between generated keys. A comprehensive application and optimality analysis of such channel codes in secret key generation algorithms can be found in [6] and [45].

In the general procedure of Maurers protocol, extra privacy amplification [63, 64] is required after the extraction of the secret key bits even if both keys obtained by *Alice* and *Bob* are identical after information reconciliation, in that the channel samples used for key generation might share insufficient independence. Privacy amplification can be implemented by linear mapping and universal hashing [64–67], or by an extractor [66,68–71]. The effort in combining both information reconciliation and privacy amplification can be seen in [46] and [72].

In this thesis, we assume the channel is properly and independently sampled, and only focus on the reconciliation of keys and secrecy leakage. A public channel is used in Maurer's procedure to communicate between *Alice* and *Bob* in attempt to reconcile the keys. Here we summarize this procedure as follows:

- Randomly permute the key on the *Alice* side
- Communicate permutation order (but not the permuted bits) to *Bob* over a public channel
- Divide the key into small blocks, encode them with a predefined code (such as BCH)
- Send only syndromes to *Bob* over the public channel
- When possible, *Bob* recovers proper bits in the key based on the syndromes and his permuted key. If some bits cannot be properly decoded, the procedure could be repeated

During the process of reconciliation, some information could be leaked to *Eve*. Since there is some probability of recovering a few bits in each block based on a syndrome communicated over the public channel, such amount of information leakage could be increased along with each iteration.

3.2.5 Adversary Model

In the adversary model, we assume that the Eavesdropper *Eve* can eavesdrop all the information exchanged between *Alice* and *Bob*. She is also able to perform channel estimation between herself and *Alice* or herself and *Bob*, even at the same time when the two transmitters are estimating the channel in between for secret key generation. It is also assumed that the eavesdropper is fully aware of all the procedures and parameters in secret key generation algorithms. However, there is a restriction in terms of *Eve*'s location that she cannot be too close to the transmitters, or more precisely, she must be away from *Alice* or *Bob* at least half of the wavelength of the radio signal while the transmitters are estimating channels and generating the secret keys [73]. This will ensure that *Eve* estimates a different and uncorrelated channel

than *Alice* or *Bob*. In addition, *Eve* can neither jam the communication channels between *Alice* and *Bob* nor can she modify any messages exchanged in between. It is also assumed that *Eve* is free to move objects between the transmitters and affect the communication channel. Essentially, *Eve* will not intentionally break the key generation process between *Alice* and *Bob*, and only acts as a passive adversary.

3.3 Key Match Rate in Secret Key Generation

3.3.1 Bit Error Rate

One of the possible ways to produce secret key bits is to quantize the power level of the received signal into Q equally probable intervals. Since in the case of Rayleigh fading this distribution is exponential, the corresponding levels of quantization could be easily found in Appendix A.2

$$I_q = \sigma^2 \ln \frac{Q}{Q-q},\ q = 1, \cdots, Q-1,\ I_0 = 0,\ I_Q = \infty \tag{3.10}$$

This is to ensure that all symbols are equally probable as required for a good key [52]. For the case of binary quantization, $I_1 = \sigma^2 \ln 2$.

Due to differences between channel estimations from *Alice* and *Bob*, there are some discrepancies between secret bits acquired on each side. The probability that both *Alice* and *Bob* recover the same bit is given by

$$P_{eq} = \sum_{q=1}^{Q} \int_{I_{q-1}}^{I_q} \int_{I_{q-1}}^{I_q} p_2(I_A, I_B) dI_A dI_B \tag{3.11}$$

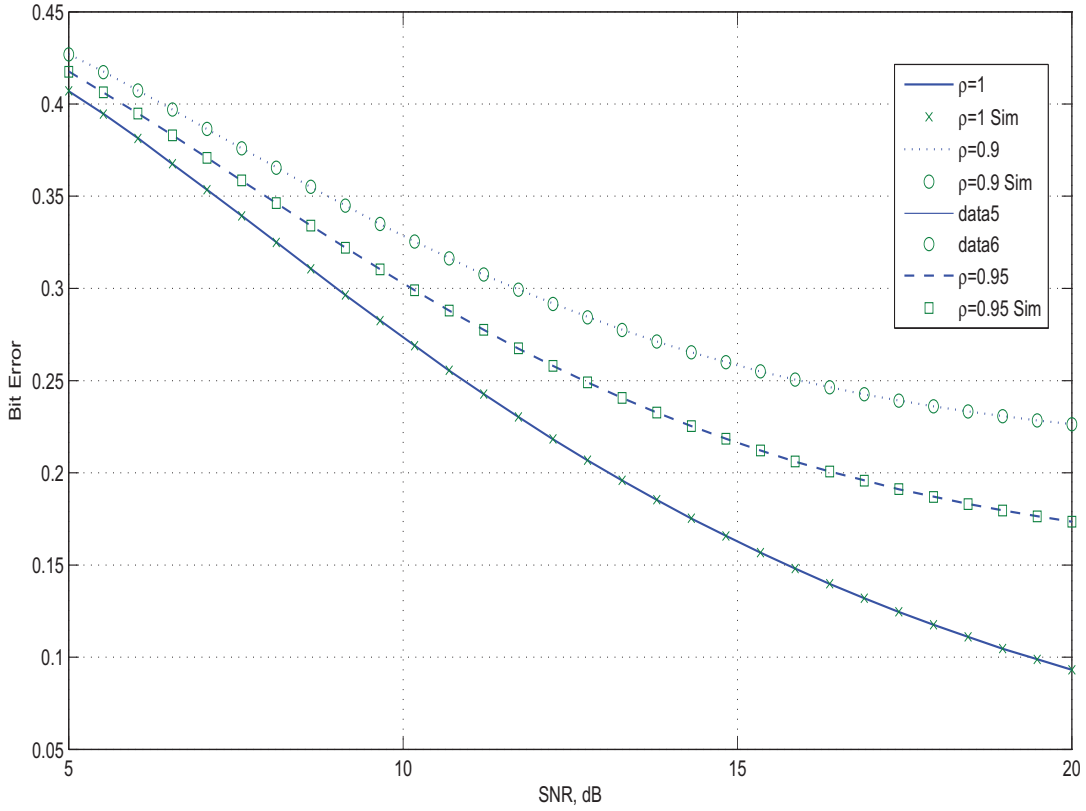

Figure 3.2: Raw bit error probability as function of SNR and channel reciprocity

where $p_2(I_A, I_B)$ is joint bivariate exponential distribution, defined by the average SNR on both sides and the power correlation coefficient. In the particular case of binary quantization, probability of one single bit mismatch is thus given by Appendix A.3

$$p = 1-2P(\ln 2, \ln 2, \rho_{eq}) = \frac{1}{2}Q_1\left(\sqrt{\frac{2\ln 2}{1-\rho^2}}, \sqrt{\frac{2\rho^2 \ln 2}{1-\rho^2}}\right)+\frac{1}{2}\left[1 - Q_1\left(\sqrt{\frac{2\rho^2 \ln 2}{1-\rho^2}}, \sqrt{\frac{2\ln 2}{1-\rho^2}}\right)\right] \tag{3.12}$$

where $Q_1(x, y)$ is the Marcum-Q function [74], $P(x, y; \rho)$ is CDF of normalized bivariate Gamma distribution as considered in the Appendix and the value of $\rho_{eq} = |\tilde{\rho}|^2$ is calculated by means of equation (3.4). Comparisons of simulation results and analytical expressions are shown in Fig. 3.2.

It can be shown that for low pilot SNR, the estimation error dominates the probability of key bit mismatch while the reciprocity plays a minor role. However, for higher SNR, non-reciprocity plays a dominant role while estimation error has less influence. Such observation leads to a conclusion that if SNR is low, one can improve secrecy rate by increasing the number of pilots used for channel estimation to obtain more accurate channel samples.

3.3.2 Probability of Raw Key Mismatch

Distribution of probabilities of the number of errors in the whole key of length K is given by the binomial distribution

$$P(k;K) = \begin{pmatrix} K \\ k \end{pmatrix} p^k (1-p)^{K-k},\ k = 0, \cdots, K \tag{3.13}$$

The average number of errors between two keys is thus given by $N_e = pK$ and the variance of number of errors $\sigma_e^2 = \sqrt{Kp(1-p)}$. It is well known that the binomial distribution can be well approximated by normal distribution with the same mean and variance. Corresponding distributions for different key length and link parameters are shown in Fig. 3.3.

3.3.3 Probabilities of Key Mismatch after Reconciliation

It is clear from the graph that the probability of key bit mismatch between two sufficiently long keys is almost a certainty. Therefore, some form of reconciliation via a public channel is required. One of the most important characteristics of key bit stream is the distribution of errors in sub-key blocks of different sizes. Such distribution has principal impact on performance of

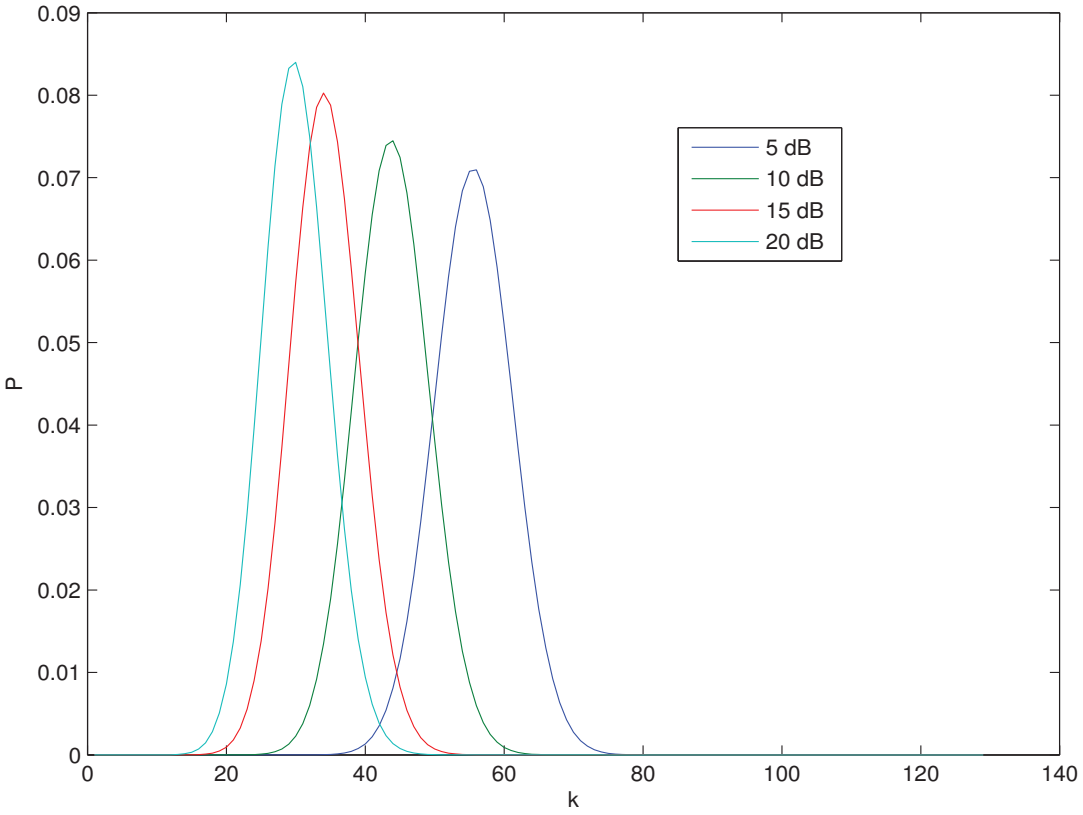

Figure 3.3: Distribution of unequal bits between keys

error correcting codes, which are applied to such blocks during reconciliation procedures (see Section 3.2.4).

Since channel samples used to obtain secret keys are independent, and the discrepancies between key bits are due to random noise, the distribution of errors can be well described by Bernoulli trials. Let a key of length K be split into N_B blocks of length L, *i.e.* $K = LN_B$. Therefore, the probability $P_e(l; L)$ of exactly l errors in a block is given by

$$P_e(l; L) = \begin{pmatrix} L \\ l \end{pmatrix} p^l (1-p)^{L-l}, \; l = 0, \cdots, L \tag{3.14}$$

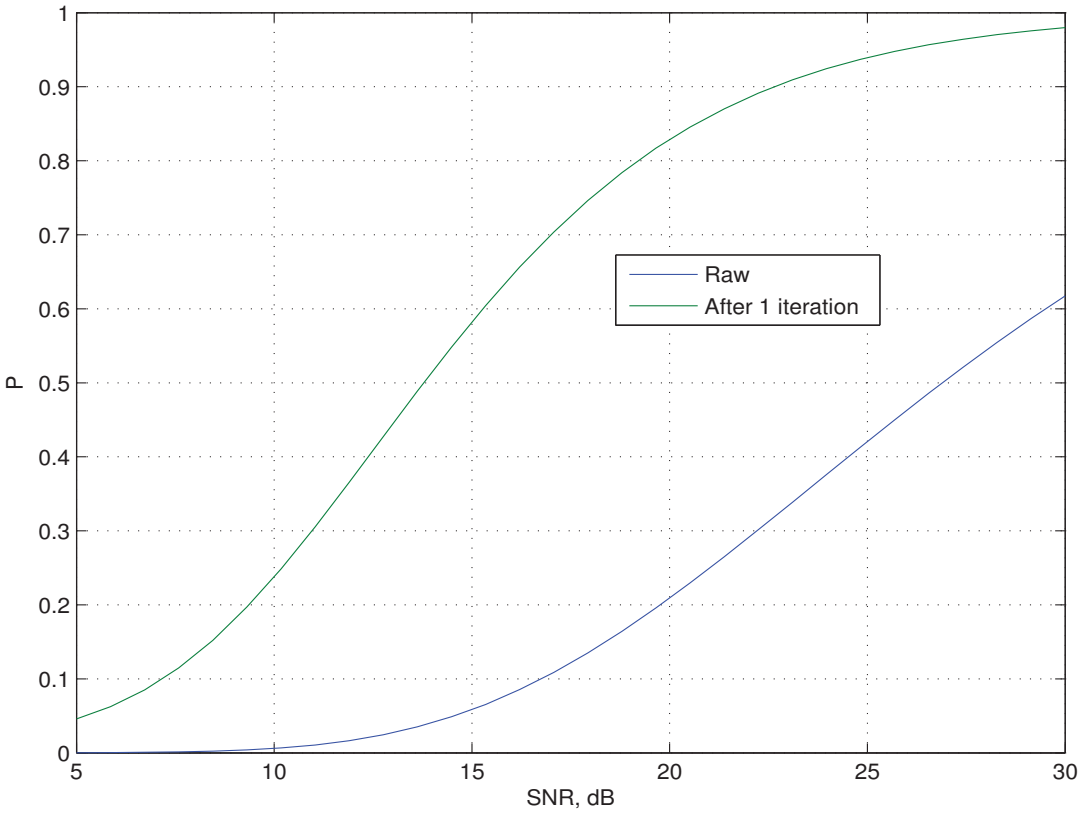

Figure 3.4: Probability of keys agreement before and after the first reconciliation

The probability that there is no more than t errors in the block is

$$P_0(t, L) = \sum_{l=0}^{t} P_e(l; L); \tag{3.15}$$

Thus, after the first round of reconciliation, the probability that both keys are identical is equal to

$$P = [P_0(t, L)]^{N_B} \tag{3.16}$$

Figure 3.4 shows the effectiveness of reconciliation after a single step for a perfectly reciprocal channel with various SNRs. As it could be expected, such a procedure can significantly increases the probability of key match rate on both sides of the *Alice-Bob* link. This is especial-

ly noticeable for higher SNR, when bit error probability is relatively low and the probability of correcting all the errors in the block is high. However, for low SNR, there could be a significant number of errors in a single block, which cannot be corrected in a single application of the reconciliation algorithm. In this case, such procedure could be repeated to improve the results. However, one has to be aware that such a procedure may lead to some information leakage to *Eve*.

3.4 Example and Simulation

In this section, we consider an example of key generation approach with 16 bits keys generated, and examine the reconciliation stage in correcting key bits errors. In this example, since each key has 16 bits, then $K = 16$. We apply Hamming $(7, 4)$ FEC correcting code in key reconciliation. In this case, $t = 1$, as this code can only correct one single error in each iteration. There will be 4 blocks of length 4 in this key. If the probability of a single bit error is p, then the probability of a single block that can be properly reconciled is

$$P_{1b} = (1 - p)^4 + 4p(1 - p)^3 = (1 + 3p)(1 - p)^3 \tag{3.17}$$

and the probability of keys agreement is

$$P = (1 + 3p)^4(1 - p)^{12} \tag{3.18}$$

Corresponding plots, as function of SNR, are shown in Fig.3.5. It could be seen that additional iterations of key reconciliation significantly improve match rate between keys, obtained

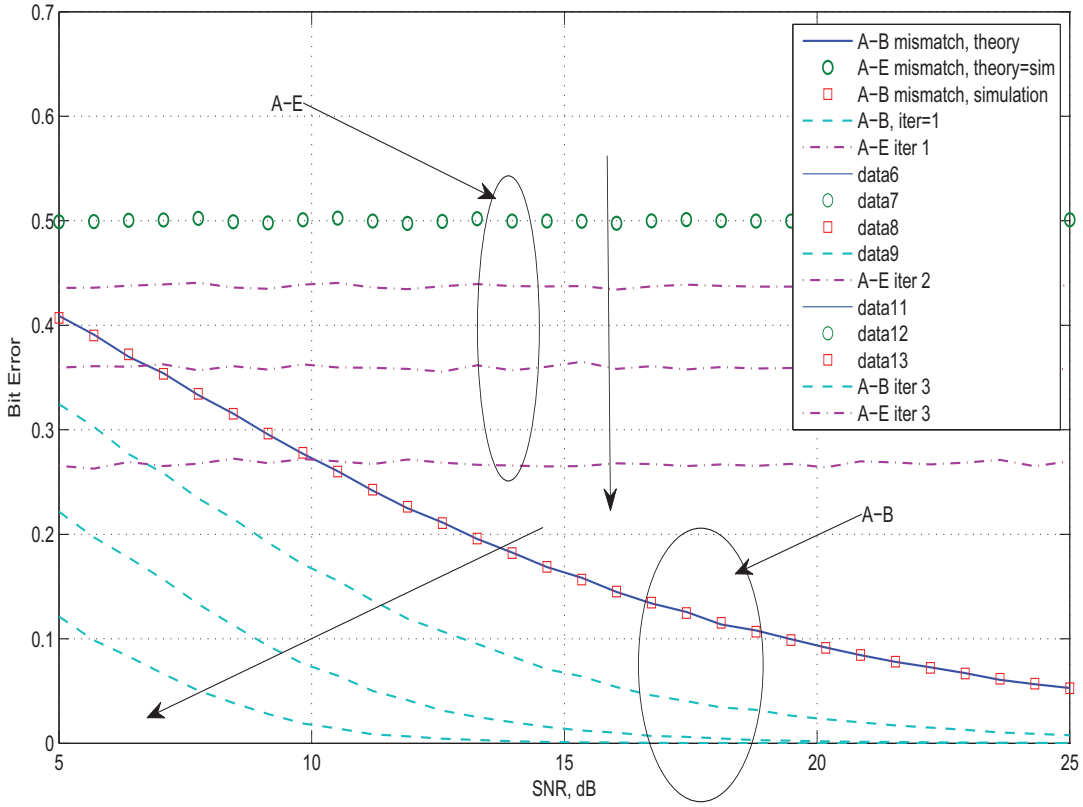

Figure 3.5: Probability of keys agreement between A and B, and A and E after a number of reconciliation iterations.

by *Alice* and *Bob*. This is especially efficient at higher SNR. However, it can be seen that the amount of leakage to *Eve* is also increasing. This amount does not depend on SNR, since initially, the key, generated by *Eve* is completely independent from that generated by *Alice*.

In practical applications, an acceptable key for secure data encryption would at least have 128 bits. Fig. 3.6 shows the key match rates of a 128-bit key before and after one iteration of reconciliation process utilizing BCH coding. Within the two processes with reconciliation, each of them has a different error correcting capability. Apparently, stronger error correcting code leads to higher key match rate, and keys with reconciliation process dominates the raw key, both with the same SNR.

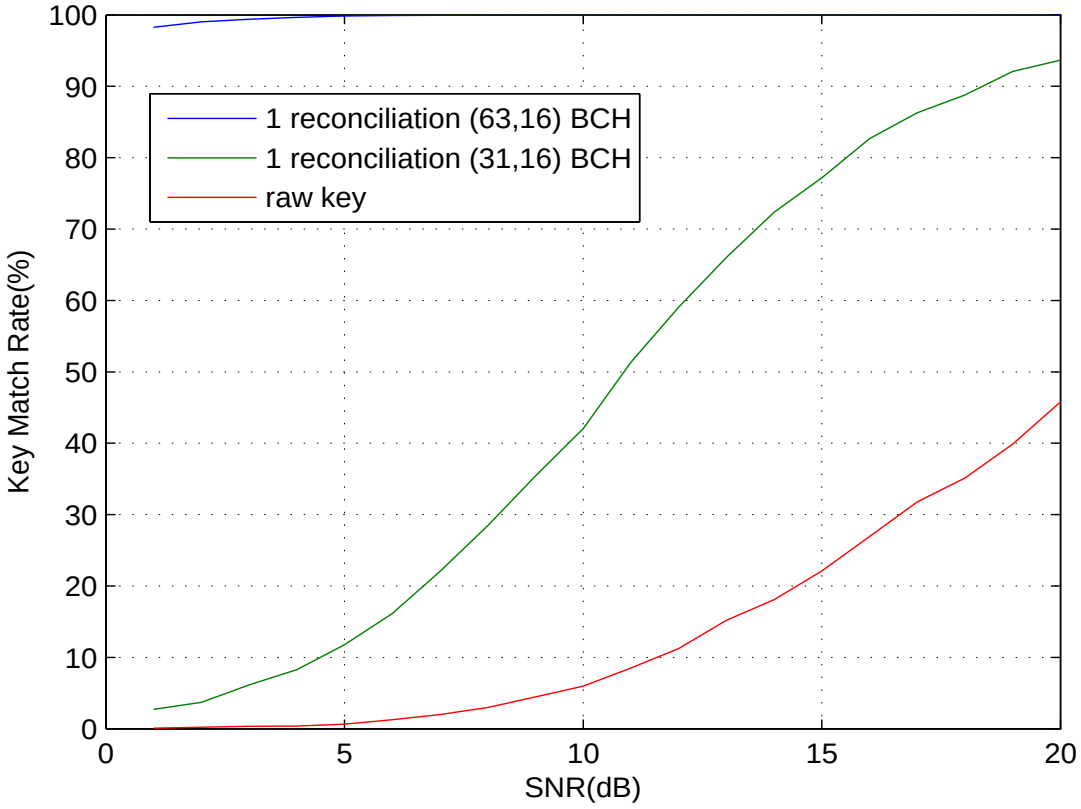

Figure 3.6: Key match rate with different parameters in BCH codes

3.5 Chapter Summary

We have investigated the impact of imperfect CSI and channel parameters, such as SNR and correlation function on generation and reconciliation of secret keys between legitimate users. A number of analytical expressions have been derived to gain insight into effect of channel parameters on key generation rate and quality of key match. It is found that SNR has predominant effect on probability of bit mismatch. This could be rectified by one of the following means: increasing the number of pilot symbols used for estimation, iterative reconciliation. Iterative reconciliation is the last resort since it leaks information to eavesdropper. Additional degradation of key match rate could be caused by non-reciprocity, either due to time or frequency division techniques. In this case the reconciliation via a public channel is the only option to

achieve required quality of key match.

Chapter 4

Secret Key Generation From Multiple Observations of Wireless Channels

4.1 Introduction

Secret key generation at physical layer has attracted more and more attentions as an emerging cryptography method. Compared to traditional security approaches, secret key generation at physical layer not only avoids the problem of key distribution, but also holds high efficiency and low complexity in application. Based on the principle of channel reciprocity, *Alice* and *Bob* both estimate the channel conditions and extract shared keys from this source of common randomness. In this paper, we first analyse the basic steps (channel estimation, sample quantization and key reconciliation) of secret key generation, key match rate with different quantization levels and key reconciliation times are also simulated. While in practical situations, different non-reciprocity factors affect the channel estimation step, key match rate can be greatly decreased and hardly meet real time cryptography requirements. In order to in-

crease the key match rate, the unified framework of physical layer key generation has been extended to utilizing multiple observations of wireless channels to generate secret keys. An improved key generation approach with multiple observations can well deal with discrepancies between transceivers and keep increasing the key match rate. Theoretical analysis and simulation results both validate the significant improvement due to multiple observations. With an increased number of observations on both sides, the desired key match rate can be achieved much greater than with a single observation, and also the probability of key recovery by *Eve* can be decreased.

Recently, a secret key generation approach based on physical layer channel reciprocity has attracted significant attentions [7], [75]. The reciprocity principle states that the channel impulse responses observed by transceivers over the uplink and downlink channel of a Time-Division duplex (TDD) system is approximately the same in both directions, assuming a slow varying channel. This is due to the fact that the link operates on the same carrier frequency in both directions, thus the signal would undergo the same perturbations [76]. This physical characteristic of wireless channels can be used by the transmitter and the receiver to generate secret keys [75].

Two wireless entities exploit the common randomness of the wireless channel and obtain two highly correlated estimates of channel states, from which, they can produce shared keys [75]. Usually, an eavesdropper exists in this scenario, it could eavesdrop but would only experience different channel conditions if it is more than half of a wavelength away from the transceivers. This feature also ensures the secrecy of the produced keys. Compared to traditional cryptographic techniques, secret key generation approaches at physical layer have some advantages. Firstly, there is no need for key management centres, since the secret keys are

directly generated by authorized parties without the process of key distribution, which is still quite an important weakness in traditional cryptography. Secondly, secret key generation at physical layer is based on the randomness of the wireless channels, thus it is independent of computational complexity, using simple hardware and achieving high efficiency can both be realized in this situation. Thirdly, the secret keys are generated dynamically because of the motions of transceivers and ever changing environment, which also improves the secrecy of shared keys.

Generally, while operating in a rich scattering environment, two authorized parties *Alice* and *Bob* (transmitter A and receiver B) both estimate the channel states of uplink and downlink channels with a specific estimation algorithm, respectively [77]. According to the feature of channel reciprocity, their estimations would be correlated to some extend and thus they can extract almost the same bits from their observations. By applying a certain quantization rule, these estimation samples will be translated into a sequence of binary bits, which is the raw key [7]. Due to some non-reciprocity factors and environment and estimation noise, two raw keys generated by transmitter A and receiver B would have some different bits. In this case, some form of key reconciliation is required. Due to the scattering environment, the only eavesdropper, *Eve*, located even in close proximity to *Bob*, will experience a significantly different physical channel. This fact makes it impossible for *Eve* to recover the same secret key.

In traditional key generation approaches, both the transmitter A and the receiver B estimate the channel conditions in one short period and each obtains one observation of the channel. Every time with their own observation, they each generate one key. While in practical situations, different non-reciprocity factors along with inevitable noises affect different steps of the whole key generation process, which can greatly corrupt reciprocity of channel estimations obtained

by transmitter and receiver. One random observation usually can not reflect the precise and integral conditions of the wireless channels as it is certainly affected by different non-reciprocity factors. In this case, the single observation of the uplink and downlink channel consists of much randomness, and keys generated from this only observation usually share many disagreement bits, and thus more efforts in later key reconciliation process will be required.

However, we can exploit more from channel reciprocity and dramatically increase key match rate by obtaining multiple observations instead of only one observation. Here, multiple observations can be obtained from channel conditions with different frequencies, antenna elements and some other methods. As all the observations describe the channel from one certain aspect or path, essentially they share much correlation. A synthesized estimation of the channel can be obtained by applying a linear combiner to these multiple observations and reduce the random factors of the estimation process to the minimum degree. Although imperfect channel reciprocity leads to more or less different estimations, highly correlated multiple observations can be combined together to overcome such random corruption. In this case, key generation approach with multiple observations yields higher key match rate than with only one observation, which we will analyse in detail in Section III.

The remainder of the paper will be organized as follows. Section II will give a brief review of traditional key generation approaches and analyse the existing key generation approaches with only one observation. An improved key generation approach with multiple observations will be proposed in Section III, detailed analysis and simulation results will also be provided. We conclude in Section IV.

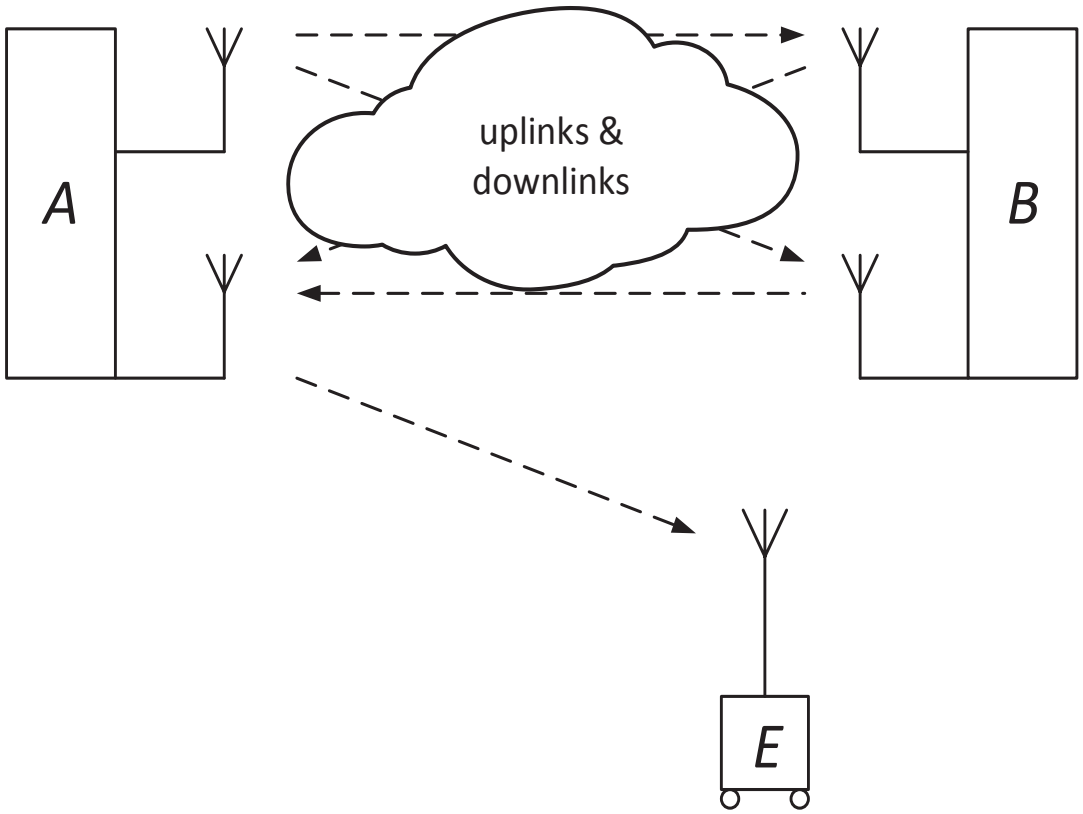

Figure 4.1: Illustration of secret key generation by *Alice* and *Bob*.

4.2 Fundamentals of PHY Key Generation

4.2.1 Channel and Data Model

Secret keys are generated from the common randomness described by the wireless channel state. In other words, both transmitter A and receiver B need to estimate the channel conditions to generate secret bits. The most common method used in this step is MMSE [77, 78] estimation. Such estimation is usually based on a single block of pilots, since independent samples of the channel are required to generate a proper key. In this case, the estimates of the channels can be modelled as [79]

$$\hat{h} = \frac{\gamma}{1+\gamma} h + \frac{\sqrt{\gamma}\sigma_h}{1+\gamma} \eta \tag{4.1}$$

where γ is the SNR, σ_h is the channel variance, and η is Gaussian noise with zero mean and unit variance. The variance of the estimate $\hat{h}$ is given by

$$\mathcal{E}\left\{|\hat{h}|^2\right\} = \frac{\gamma}{1+\gamma}\sigma_h^2 \tag{4.2}$$

The correlation between these two estimates $\hat{h}_{AB}$ and $\hat{h}_{BA}$ can be found as

$$\tilde{\rho} = \frac{\mathcal{E}\left\{\hat{h}_{AB}\hat{h}_{BA}^*\right\}}{\sqrt{\mathcal{E}\left\{|\hat{h}|_{AB}^2\right\}\mathcal{E}\left\{|\hat{h}|_{BA}^2\right\}}} = \sqrt{\frac{\gamma_{AB}}{1+\gamma_{AB}}}\sqrt{\frac{\gamma_{BA}}{1+\gamma_{BA}}}\rho \tag{4.3}$$

With the assumption of perfect channel reciprocity such that $\rho = 1$ and $h_{AB} = h_{BA} = h$, $\gamma_{AB} = \gamma_{BA} = \gamma$, the estimates of channel conditions by A and B are as follows

$$\hat{h}_{AB} = \frac{\gamma}{1+\gamma}h + \frac{\sqrt{\gamma}\sigma_h}{1+\gamma}\eta_{AB} \tag{4.4}$$

$$\hat{h}_{BA} = \frac{\gamma}{1+\gamma}h + \frac{\sqrt{\gamma}\sigma_h}{1+\gamma}\eta_{BA} \tag{4.5}$$

If we assume a more practical channel model that features channel non-reciprocity but also holds $\gamma_{AB} = \gamma_{BA} = \gamma$, we have $\rho \neq 1$, then

$$h_{BA} = \rho h_{AB} + \sqrt{1-|\rho|^2}\sigma_h\xi \tag{4.6}$$

where ξ is AWGN with unity variance. So we have

$$\hat{h}_{AB} = \frac{\gamma}{1+\gamma}h_{AB} + \frac{\sqrt{\gamma}\sigma_h}{1+\gamma}\eta_{AB} \tag{4.7}$$

$$\hat{h}_{BA} = \frac{\gamma}{1+\gamma}(\rho h_{AB} + \sqrt{1-|\rho|^2}\sigma_h \xi) + \frac{\sqrt{\gamma}\sigma_h}{1+\gamma}\eta_{BA} \tag{4.8}$$

4.2.2 Sample Quantization

Estimates of the channel conditions need to be further translated into binary bits to form the key. One possible way to generate key bits is to quantize the channel gains of the received signal according to a certain quantization rule. However, other features of the channels, such as phase, envelope, received signal impulse, can also be treated as the randomness source of key extraction. Here, by quantization of the magnitude of the uplink and downlink channel, channel conditions estimated by transmitter A and receiver B can be converted to two sequences of binary bits, which is the raw keys here. Furthermore, within a certain quantization rule, different choices of quantization levels can lead to different lengths of the raw keys and also different probabilities of deriving the same key by transmitter A and eavesdropper E. If we choose 2 level quantization, channel magnitudes of each sample can be quantized into 1 or 0. If we choose 4 level quantization, channel magnitudes of each sample can be quantized into 00, 01, 10, and 11, in which case, the key length will be twice of that in the case of two quantization levels. The case of 8 quantization levels is similar. From the perspective of security issues, the probability of deriving the same key by transmitter A and eavesdropper E, after the same procedure deployed on two independent channels, is given by

$$P_{k_A=k_B} = Q^{-D} \tag{4.9}$$

where Q is the quantization levels and D is the number of samples utilized to generate a key.

It is clear from this equation that by increasing the quantization level we can decrease the

probability that A and E generate the same key. However, with longer key length that caused by increased quantization level, the probability of disagreement bits between A and B will also increase.

4.2.3 Key Reconciliation

It is almost for sure that two raw keys with a certain length will have different bits, that is why raw keys usually can not be used directly in practical situations. As a result, some form of key reconciliation is required via a public channel between the transceivers to correct the raw key differences [7]. Here we adopt the t-error correcting code to reconcile the two raw keys.

At the first step, the transmitter A divides the raw key sequence of length K into N_B blocks of length L, $K = L \cdot N_B$. By performing a certain t-error correcting code (for example, (7,4) BCH code, then $L = 4$) on each block of the raw key, N_B codewords will be generated and each codeword consists of 4 bits from the original block and 3 bit syndrome. Secondly, transmitter A sends the N_B syndromes to receiver B over the public channel. We assume the public channel is noiseless and receiver B can receive all the syndromes without any mistake. In the meantime, an eavesdropper will also capture all the syndromes, but it can not learn much information about the raw keys only from the syndromes. Thirdly, receiver B also divides its raw key sequence into N_B blocks, of course, each block has the same length as blocks of transmitter A. B adds each received syndrome after each divided block, and then it also obtained N_B codewords. Fourthly, by applying the same t-error correcting code, B decodes all the N_B codewords and get N_B 4 bit sequences, all the sequences form the new Key_B after one iteration of key reconciliation.

Due to corruptions of random noise and capabilities of error correcting code, usually one iteration of key reconciliation is not sufficient to correct all the different bits between two raw keys. Therefore, more iterations of key reconciliation is necessary in order to achieve near 100% key match rate. However, before each extra key reconciliation, a random permutation of the key bits on both sides need to be performed to spread the error bits that were not corrected in the previous error correcting process.

In order to perform exactly the same permutation, transmitter *A* and receiver *B* also need to agree on the permutation. There is a mapping relationship between the permutation and the corresponding number, it is called the *Lehmer Code* [80]. By sending the specific number from *A* to *B*, transmitter and receiver can easily agree on the same permutation.

Table 4.1: Mapping table

Permutation	Permutation number
abc	0
acb	1
bac	2
bca	3
cab	4
cba	5

Here we briefly address how *Lehmer Code* works in agreeing on the permutation. As shown in Table I, for 3-digit permutations, the permutation numbers would be 0, 1, 2, 3, 4, 5, for the specific permutation cab, its corresponding number is 4. With this kind of relationships, when transmitter *A* performs a random permutation, she only needs to send the corresponding number to receiver *B* according to *Lehmer Code*. After receiving the number, *B* can employ the same permutation as *A* by decrypting the number into the corresponding permutation.

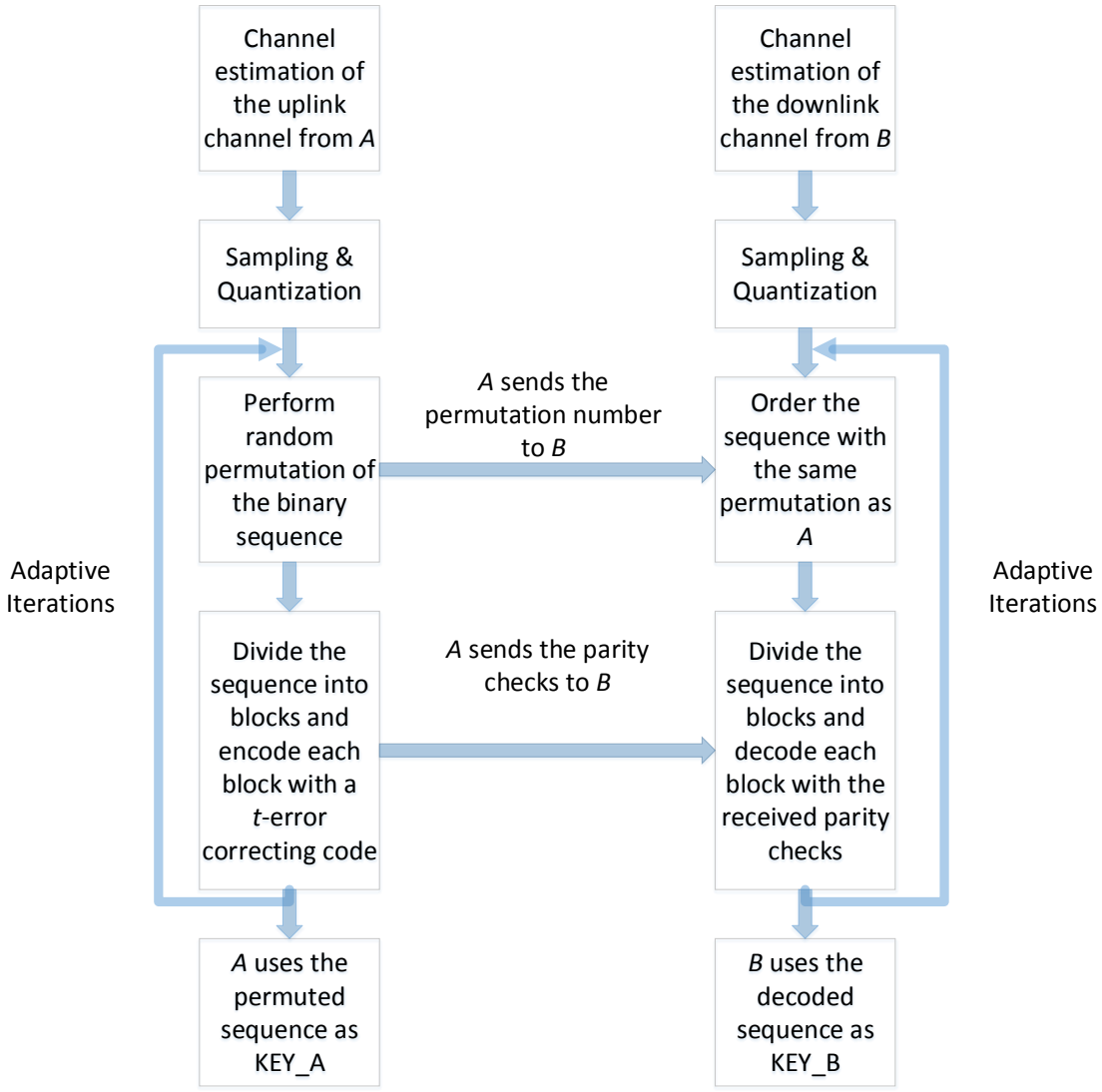

Figure 4.2: Physical layer secret key generation flowchart

4.2.4 Simulation Results

From Figure 4.3, we can see that key match rate after one time key reconciliation with 2, 4 and 8 quantization levels has different performance. Key match rate with 2 quantization level is higher than that with 4 quantization level, and key match rate with 4 quantization level is higher than that of 8 quantization level. Here we set the key length as 16 bits, adopt MMSE estimation and use (7,4) BCH as the t-error correcting code.

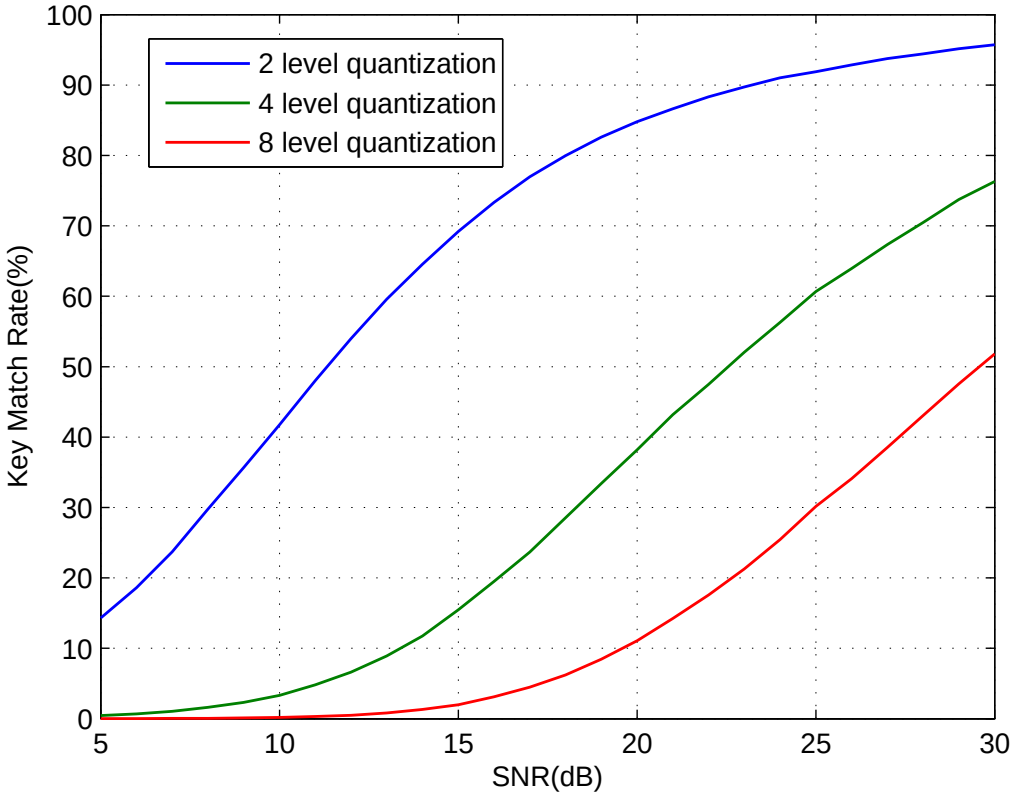

Figure 4.3: Key match rate with different quantization levels

As Figure 4.4 shows, more iterations of key reconciliation yield better performance of key match rate. Even with one time key reconciliation, key match rate can be greatly improved compared to that of raw keys. From the figure, we can see that with SNR greater than $20dB$, 5 iterations of key reconciliation can ensure nearly 100% key match rate. Also, we set the key length as 16 bits, adopt MMSE estimation and use (7,4) BCH as the t-error correcting code.

However, as it has been shown in [79], additional iterations of reconciliation process lead to increased degree of proper decoding by *Eve*. In order to avoid this situation, we suggest using multiple observations in key generation to reduce the bit mismatch rate at the first step.

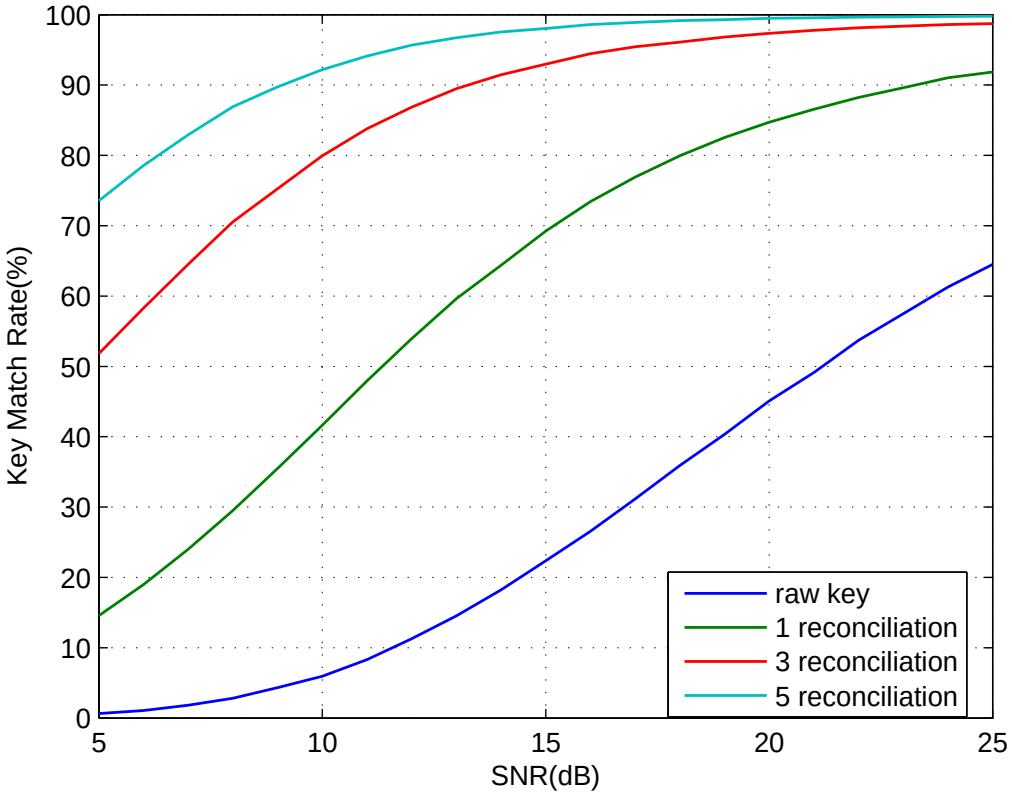

Figure 4.4: Key match rate with different iterations of key reconciliation

4.3 Key Generation from Multiple Observations

4.3.1 Normalized Correlation Coefficient of Multiple Observations

Let us extend the consideration of Section 4.2 to the case of multiple observations. This could be attributed to measurements of the channel at different frequencies, sequential observations, antenna elements or polarization, *etc*. In particular, let both *Alice* and *Bob* have access to N correlated but nonidentical observations

$$\mathbf{Y} = \tilde{\rho}\mathbf{X} + \sqrt{1 - |\tilde{\rho}|^2}\boldsymbol{\Xi} \tag{4.10}$$

Here, $\mathbf{X} = [x_1, x_2, \cdots, x_N]^T$ is the vector of observations on the *Alice* side, while $\mathbf{Y} = [y_1, y_2, \cdots, y_N]^T$ are measurements on the *Bob* side. The correlation coefficient $\tilde{\rho}$ is the same for all channels and is given by equation (4.3). Furthermore, statistical differences between these observations are provided by vector of WGN components $\boldsymbol{\Xi}$ with zero mean and unit variance. Let $\mathbf{w} = [w_1, w_2, \cdots, w_N]^T$, $\mathbf{w}^H\mathbf{w} = 1$. It is a linear processor used on both sides of the legitimate link. Our goal is to define $\mathbf{w}$ such that the correlation ρ_N between observations $z_A = \mathbf{w}^H\mathbf{X}$ and $z_B = \mathbf{w}^H\mathbf{Y}$ attains its maximum. Applying $\mathbf{w}^H$ to both sides of (4.10), one obtains

$$z_B = \mathbf{w}^H\mathbf{Y} = \tilde{\rho}\mathbf{w}^H\mathbf{X} + \sqrt{1-|\tilde{\rho}|^2}\mathbf{w}^H\boldsymbol{\Xi} = \tilde{\rho}z_A + \sqrt{1-|\tilde{\rho}|^2}\mathbf{w}^H\boldsymbol{\Xi} \tag{4.11}$$

The variance σ_A^2 of the observation z_A and the variance σ_B^2 of the observation z_B are thus given by

$$\sigma_A^2 = \mathcal{E}\left\{|z_A|^2\right\} = \mathbf{w}^H\mathbf{R}_{xx}\mathbf{w} \tag{4.12}$$

and

$$\sigma_B^2 = \mathcal{E}\left\{|z_B|^2\right\} = |\tilde{\rho}|^2\,\mathbf{w}^H\mathbf{R}_{xx}\mathbf{w} + \left(1-|\tilde{\rho}|^2\right) \tag{4.13}$$

At the same time, correlation R_{ab} between observations z_A and z_B is just

$$R_{ab} = \mathcal{E}\left\{z_B^* z_A\right\} = \tilde{\rho}\sigma_A^2 = \tilde{\rho}\mathbf{w}^H\mathbf{R}_{xx}\mathbf{w} \tag{4.14}$$

Finally, normalized correlation coefficient ρ_N is given by

$$\rho_N = \frac{R_{ab}}{\sigma_A\sigma_B} = \tilde{\rho}\sqrt{\frac{\mathbf{w}^H\mathbf{R}_{xx}\mathbf{w}}{|\tilde{\rho}|^2\,\mathbf{w}^H\mathbf{R}_{xx}\mathbf{w} + (1-|\tilde{\rho}|^2)}} \tag{4.15}$$

Since ρ_N is an increasing function of the quadratic form $\mathbf{w}^H\mathbf{R}_{xx}\mathbf{w}$, its maximum coincides with the maximum of $\lambda_1 = \mathbf{w}^H\mathbf{R}_{xx}\mathbf{w}$, where λ_1 is the biggest eigenvalue of $\mathbf{R}_{xx}$ and the vector $\mathbf{w}$ coincides with eigenvector, corresponding to λ_1. Therefore

$$\rho_N = \tilde{\rho}\sqrt{\frac{\lambda_1}{\lambda_1|\tilde{\rho}|^2 + 1 - |\tilde{\rho}|^2}} \tag{4.16}$$

If all the observations are independent, then $\mathbf{R}_{xx} = \mathbf{I}$ and $\lambda_1 = 1$. In this case $\rho_N = \tilde{\rho}$, *i.e.* there is no improvement, compared to the case of a single channel observation. In fact, the processing vector $\mathbf{w}$ has only one non-zero component, equal to 1. On the contrary, when observations are very correlated, $\mathbf{R}_{xx} = \mathbf{1^H 1}$, and thus $\lambda_1 = N$. In this case

$$\rho_N = \tilde{\rho}\sqrt{\frac{N}{N|\tilde{\rho}|^2 + 1 - |\tilde{\rho}|^2}} > \tilde{\rho} \tag{4.17}$$

If the number of observations increases to infinity $N \to \infty$, $\rho_N \to 1$, and observations on both sides become identical.

4.3.2 Feasibility of Multiple Observations of Wireless Channels

In a practical situation, assuming that both *Alice* and *Bob* possess two antennas, such that $N = 4$ estimations can be made from the 4 paths between the transceivers by each of *Alice* and *Bob*. In terms of correlation coefficient, the improvement of key generation with multiple observations over single observation can be expressed as

$$\frac{\rho_N}{\tilde{\rho}} = \sqrt{\frac{N}{N|\tilde{\rho}|^2 + 1 - |\tilde{\rho}|^2}} = \frac{2}{\sqrt{3|\tilde{\rho}|^2 + 1}} \tag{4.18}$$

The improvement of correlation coefficient obtains its maximum when $\tilde{\rho}$ gets its minimum. That means key generation with multiple observations has relatively greater improvement in situations with smaller correlation coefficient of the wireless channels, this can also be verified by later simulation results.

In practical applications, based on this idea of multiple observations, we can extend our key generation algorithm to fully utilizing more than one observation. In the channel estimation step, instead of estimating only one channel condition, multiple correlated channel conditions are estimated both by *Alice* and *Bob*. This is feasible, since during the information exchange, a number of packets using the same inscription key are transmitted (after the initialization) and the channel estimates could be stored for the next round of secret key generation.

Assuming that *Alice* and *Bob* have N correlated but nonidentical estimations of uplink and downlink channels, respectively. By calculating and utilizing a linear processor $\mathbf{w}$, *Alice* and *Bob* each get a synthesized channel estimation $\hat{h}_{AB}$ and $\hat{h}_{BA}$. With these two synthesized channel estimations, followed by quantization and key reconciliation in the standard steps of key generation, theoretically key match rate on both sides can be improved with the increase of observations, as we analysed before.

4.4 Simulations

Simulation results also demonstrate the improvement and efficiency of using multiple observations in key generation regarding key match rate. Three different cases have been simulated. In all cases, key length is 16 bits, and each side performs one time key reconciliation with (7,4) BCH code.

Case I: Key match rate with different numbers of observations. Assuming *Alice* and *Bob*

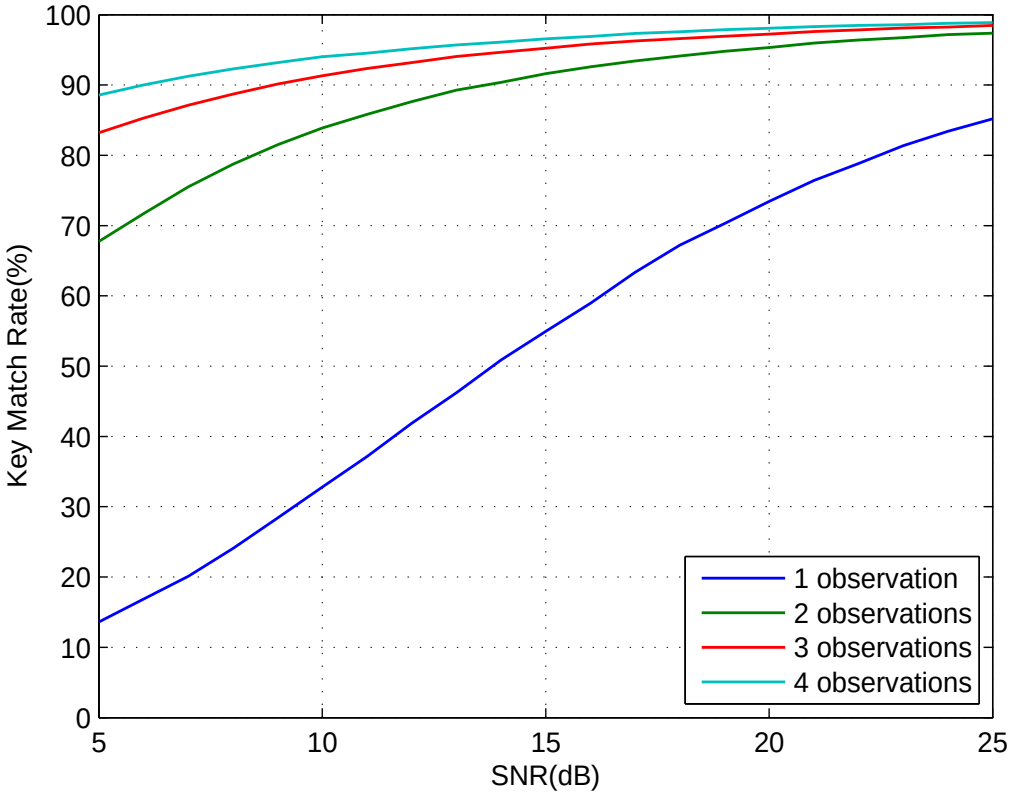

Figure 4.5: Key match rate with different numbers of observations.

experience perfect channel reciprocity, key match rates with $N = 1, 2, 3, 4$ channel observations are depicted in Fig. 4.5. This plot shows that with more channel observations on both sides, key match rate keeps growing. It is also clear that key match rate with two observations has significantly better performance than with only one observation. With 4 observations from each side, key match rate can achieve nearly 90% with signal to noise ratio as low as $5dB$.

Case II: Key match rate with different correlation coefficients. In this case, we assume that both *Alice* and *Bob* possess $N = 4$ channel observations, while correlation coefficient of uplink and downlink channel differs each time with $\rho = 1, 0.8, 0.6$. The performance of key match rate is depicted in Fig. 4.6. This plot shows that with higher correlation coefficient of uplink

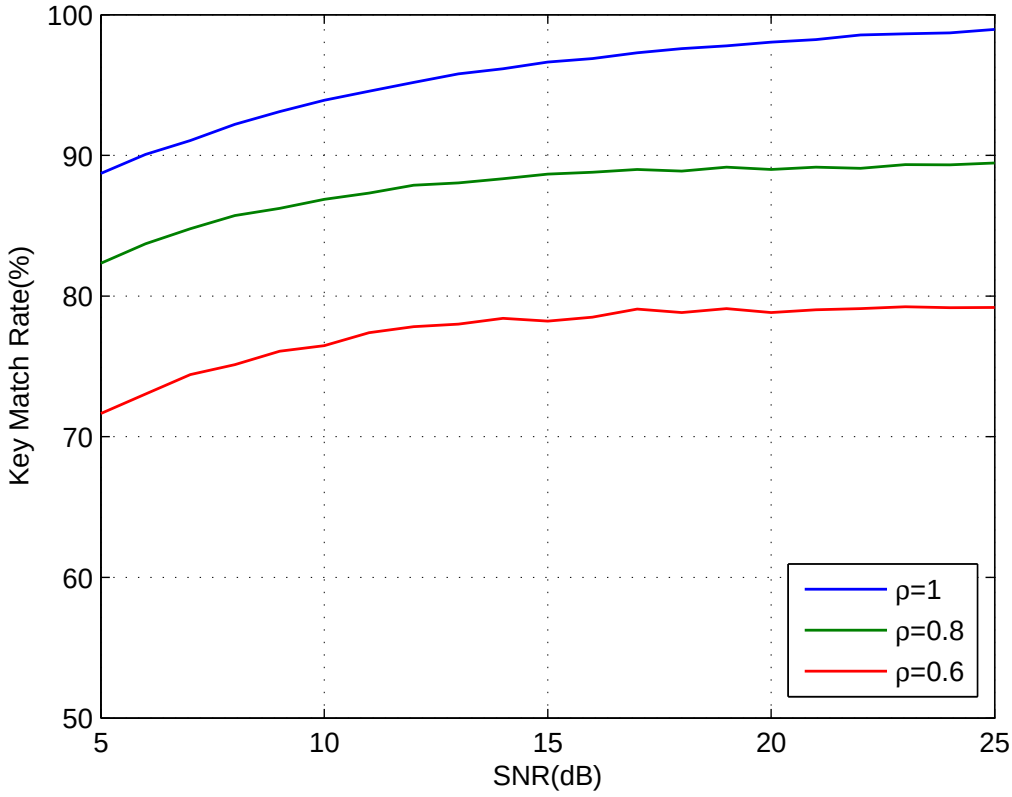

Figure 4.6: Key match rate with different correlation coefficients.

and downlink channel, key match rate keeps growing. Apparently key match rate with $\rho = 1$, which is perfect channel reciprocity, achieves best performance.

Case III: Key match rate with different numbers of observations and different correlation coefficients. In the third case, *Alice* and *Bob* both experience wireless channels with $SNR = 10dB$. The number of channel observations differs as $N = 1, 2, 3, 4$, and correlation coefficient of uplink and downlink channel changes as $\rho = 1, 0.8, 0.6$. Fig. 4.7. shows that with more channel observations on both sides and higher correlation coefficient of uplink and downlink channel, key match rate keeps growing.

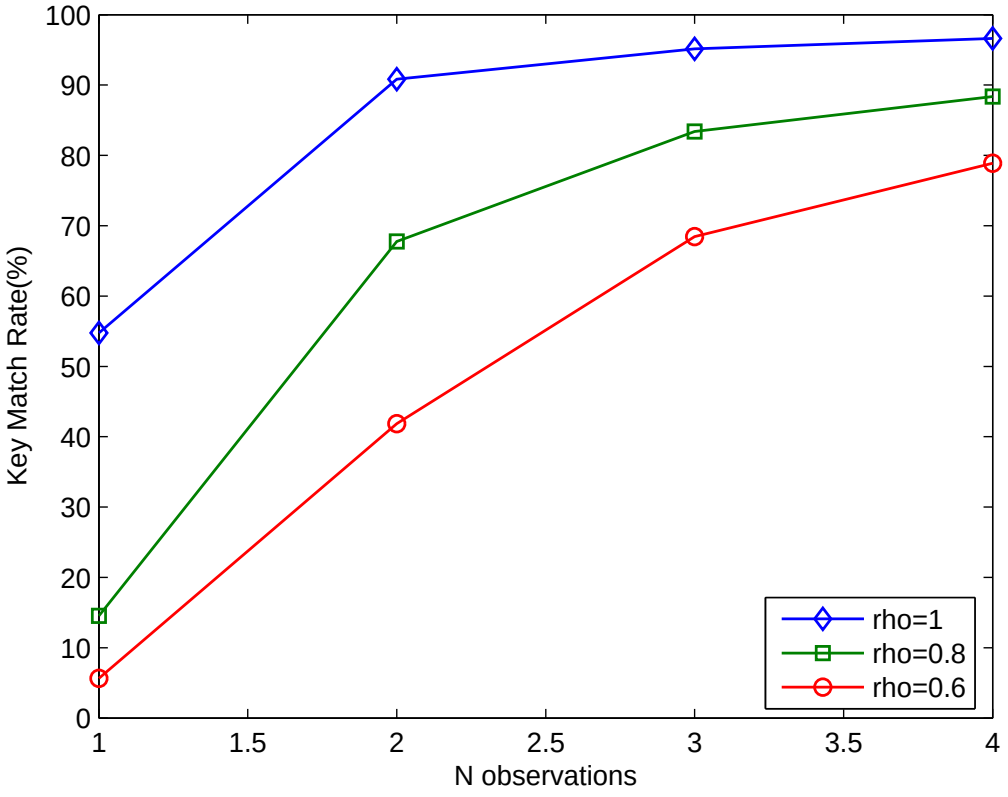

Figure 4.7: Key match rate with different numbers of observations and different correlation coefficients.

4.5 Chapter Summary

In this paper, a unified framework of key generation approach at physical layer has been improved to increase key match rate by utilizing multiple observations of wireless channels. We have derived mathematical equations to describe different steps of key generation, and also analysed the significant improvement of key match rate with multiple observations. Simulation results indicate the influence of quantization levels and key reconciliation times on key match rate. While in the step of channel estimation, key generation approach with multiple observations can further improve the key match rate. Multiple observations processed by a linear combiner can well counteract the non-reciprocity factors in practical wireless channels,

and thus increase key match rate. Simulation results also demonstrate that with an increasing number of observations on both sides, nearly 100% key match rate can be achieved.

Chapter 5

Secret Key Generation with Partial Quantization and Its Application in Wireless Networks

5.1 Introduction

In the previous key generation methods, sample quantizations are conducted based on a hard decision, such that magnitudes of channel samples are quantized to 1 or 0 when compared with the median magnitude of all channel samples in the most recent time period. Due to noise or interference corruptions to the channel reciprocity, measurements on both sides will have slight differences, and the median magnitude of all channel samples might fall into this difference area, and eventually lead to mismatch bit after key extraction with higher probability. For low SNR regime, this situation is more serious, and the relatively lower key match rate with low SNR has been proved both theoretically and experimentally in previous chapters.

Since one of the reasons that mismatch bits exist in keys generated by *Alice* and *Bob*, especially in low SNR scenarios, is the fact that a hard decision is made based on a single threshold, with relatively high probability of bit cross-over errors. One way to reduce these errors is to use two thresholds of quantization. In this section, we propose an improved key generation protocol with partial quantization, which means two level quantization will be applied with a dead region in the magnitudes.

After that, a practical application of secret key generation at physical layer is presented. Secret keys extracted from the wireless channels are converted to permutations for data scrambling. And secure data transmission and reliable access control in the scenario of mobile networks are both achieved with the exploitation of physical layer secret key generation.

5.2 Secret Key Generation with Partial Quantization

5.2.1 Magnitude based quantization with a dead region

In magnitude based quantization with a dead region, two levels $I_m < \sigma_2 \ln 2 < I_M$ are selected such that

$$P_0 = \int_0^{I_m} p(I)dI = \int_{I_M}^{\infty} p(I)dI = P_1 = q < 1/2 \tag{5.1}$$

Therefore

$$I_m = -\sigma^2 \ln(1-q),\ I_M = -\sigma^2 \ln(q) \tag{5.2}$$

It is clear that with probability $1 - 2q$ no key bit is obtained, which results in a lower key generation rate, and it is the price for improved reliability. We would also assume in the following that the fact that a decision is made (but not the decision itself) is communicated

properly among two sides over the public channel. After such reconciliation, with all erasures removed from consideration, the probability of 1 and 0 bits are equal to be $q/(q+q) = 0.5$ as required for good keys.

The next step is to calculate probability of bit mismatch p_q under a new quantization scheme. The probability p_M of symbols match in both keys is given by

$$p_M = \int_0^{I_m} \int_0^{I_m} p(I_A, I_B) dI_A dI_B + \int_{I_M}^{\infty} \int_{I_M}^{\infty} p(I_A, I_B) dI_A dI_B \tag{5.3}$$

The probability of erroneous decision p_E is given by

$$p_E = 2 \int_0^{I_m} \int_{I_M}^{\infty} p(I_A, I_B) dI_A dI_B \tag{5.4}$$

While the probability that at least one of the two sides is not able to make a decision is

$$P_N = 1 - P_M - P_E \tag{5.5}$$

Thus, the probability of a mismatch bit is given by

$$p = \frac{P_E}{P_E + P_M} \tag{5.6}$$

5.2.2 Simulations

Simulation results of the raw key match rate with different quantization thresholds are shown in Fig. 5.1. It is obvious that with wider dead region in the process of magnitude based

quantization, key extraction protocols result in higher key match rate. And even with a slight region eliminated from sample quantization, key match rate will get a significant increase than with one threshold quantization. For threshold parameter $q = 0.35$, key match rate will achieve 100% with SNR as low as 15dB.

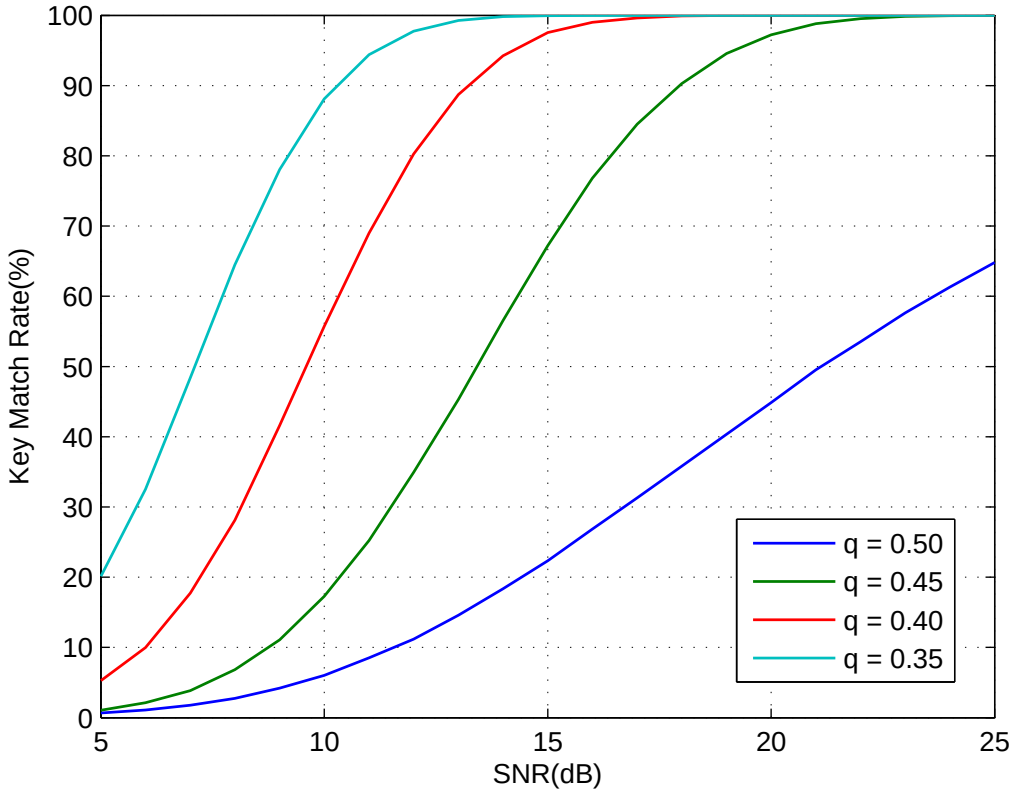

Figure 5.1: Key match rate with different quantization thresholds

Fig. 5.2 simulates the performance of key match rate with both procedures of two thresholds partial quantization and one iteration of information reconciliation. Undoubtedly, the final key match rate performs a near perfect result even with a very low SNR, and key generation protocols with both of these procedures can well meet practical requirements for 100% key match rate.

With two thresholds elected in the process of magnitude based quantization, it's inevitable

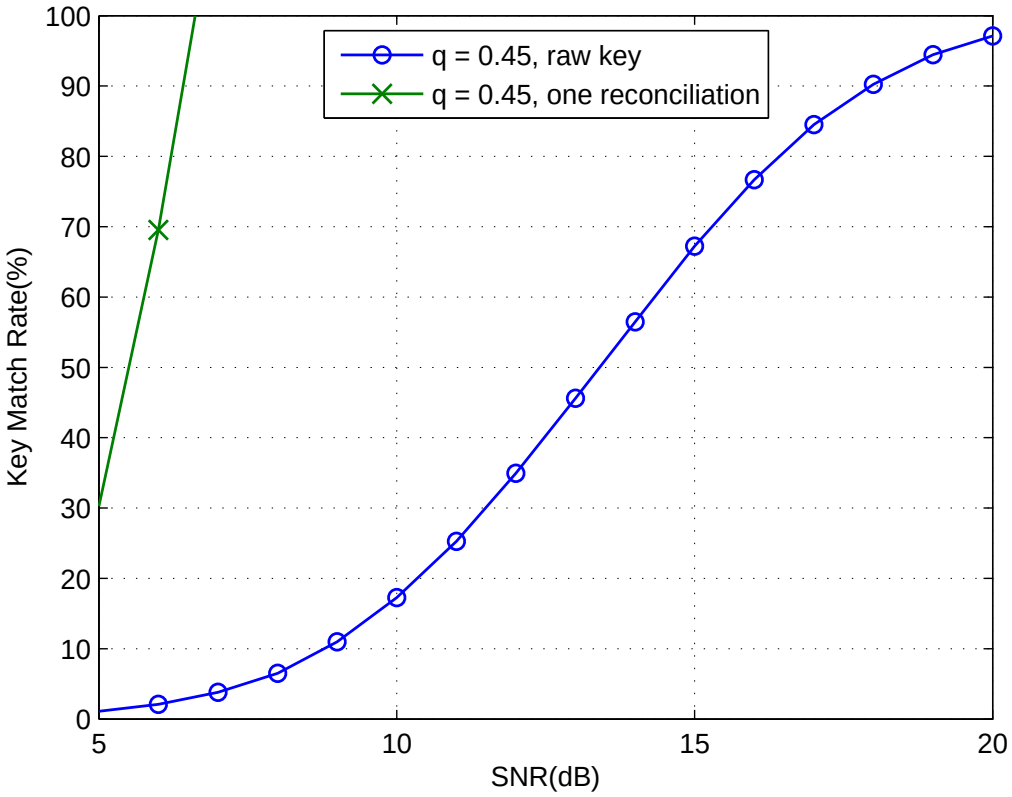

Figure 5.2: Key match rate with quantization threshold $q = 0.45$ before and after reconciliation

that fewer channel samples will be processed for key bit extraction. It can be foreseen that with a reasonable amount of samples eliminated from key generation, key match rate along with successful key generation rate will both increase. While with an extreme low q value that blocks most of the channel samples for key bit extraction, apparently key generation rate will reach as low as 0 bit per second. As a result, the successful key generation rate will reach the peak with a particular threshold parameter chosen for quantization. And with higher SNR, the highest successful key generation rate can be achieved with smaller dead region,and thus with greater q value.

Secret key generation with $15dB$, $20dB$ and $25dB$ SNR are all simulated to obtain the successful key generation rate with difference threshold parameters. And all three situations achieve the highest key generation rate with a particular q value. And as we analysed before, higher SNR requires smaller dead region, which means greater q value can be utilized.

5.3 Data Scrambling with Secret Keys

Physical layer secret key generation differs with traditional cryptographic methods only in the ways of key bits extraction, while all the generated keys will be utilized for data encryption without distinction. A natural and simple approach to encrypt secret information is to scramble the data bits with a predefined order. At the same time, it's possible and convenient to convert dynamic binary key bits to permutations for data encryption. In this section, we address a simple and efficient method to convert secret keys into permutation, and present how to combine multiple similar keys into one permutation in case of slow varying channels. An approach of similarity check for binary keys is also provided.

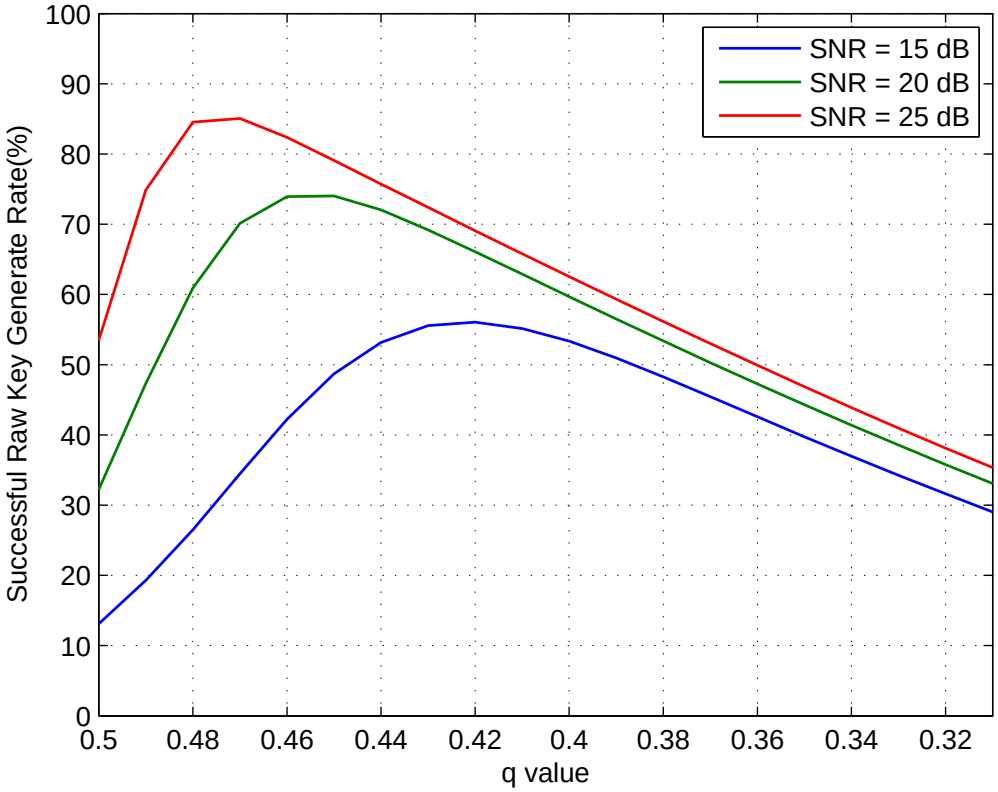

Figure 5.3: Successful raw key generation rate with different quantization thresholds and different SNR

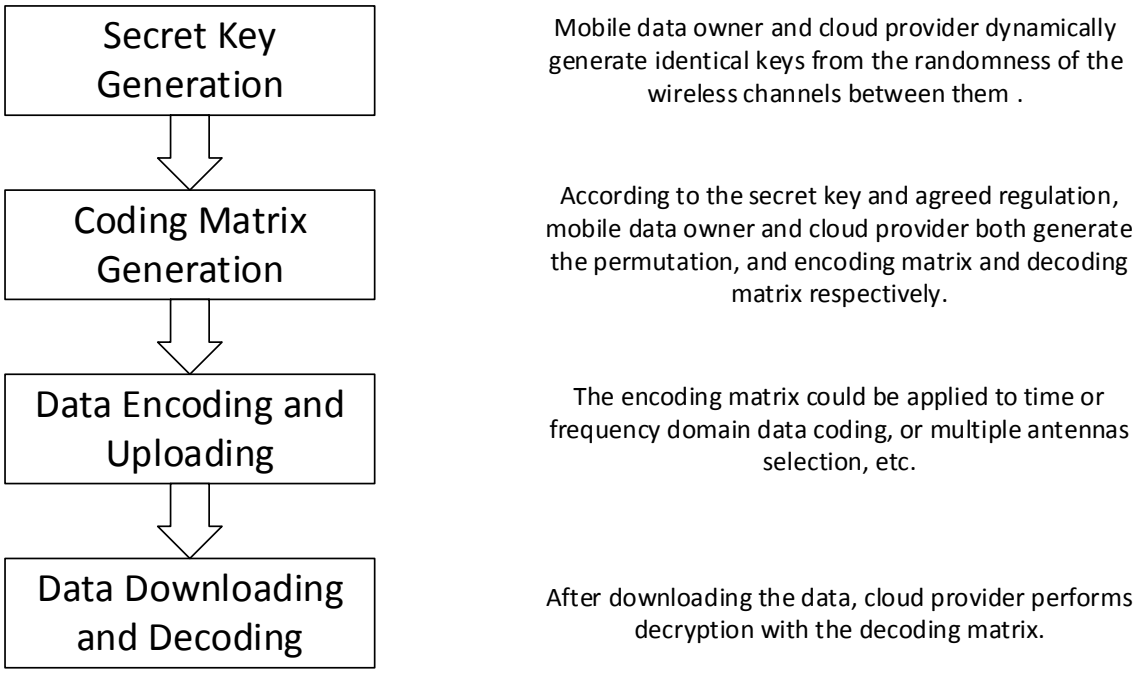

Figure 5.4: Data encryption model

5.3.1 Converting Secret Key to Permutation

Let us take the 16 bit keys for example, the binary sequence consists of eight '1's and eight '0's. There are $\binom{16}{8} = 12870$ such keys. They are:

0000000011111111

0000000101111111

0000000110111111

$\vdots$

1111111100000000

Our proposal is to convert the sequence into a specific number, the number, at the same time, can be transferred into a specific permutation. We decide the number to be the position of the specific key in the book of all possible keys. Now we arrange all the binary sequences

in the ascending order.

For instance, the position of key 1010101010101010 can be calculated as follows,

$$Position\ number = \binom{i_8}{8} + \binom{i_7}{7} + \binom{i_6}{6} + \binom{i_5}{5} + \binom{i_4}{4} + \binom{i_3}{3} + \binom{i_2}{2} + \binom{i_1}{1}$$

where i_n is the position of the n*th* '1' in the binary sequence. In this case,

$$Position\ number = \binom{15}{8} + \binom{13}{7} + \binom{11}{6} + \binom{9}{5} + \binom{7}{4} + \binom{5}{3} + \binom{3}{2} + \binom{1}{1} = 8788$$

which means key 1010101010101010 sits in the 8788*th* position of the key book.

At the same time, the position number can be regarded as a permutation number, which can be mapped into a specific permutation.

$$Permutation\ number = Position\ number = 8788$$

For example, all the six possible permutations for three digit *a b c*,

$$a \quad b \quad c$$

$$a \quad c \quad b$$

$$b \quad a \quad c$$

$$b \quad c \quad a$$

$$c \quad a \quad b$$

$$c \quad b \quad a$$

can be numbered as 0 to 5, respectively.

Inspired from this example, now we can convert the 16 digit binary sequence into an 8 digit permutation, with the permutation number linking in between. According to the permutation number, we can calculate the indexes of all the digits as follows,

$$
\begin{aligned}
8788/7! &= 1\cdot\cdot\cdot\cdot\cdot\cdot 3748 \\
3748/6! &= 5\cdot\cdot\cdot\cdot\cdot\cdot 148 \\
148/5! &= 1\cdot\cdot\cdot\cdot\cdot\cdot 28 \\
28/4! &= 1\cdot\cdot\cdot\cdot\cdot\cdot 4 \\
4/3! &= 0\cdot\cdot\cdot\cdot\cdot\cdot 4 \\
4/2! &= 2\cdot\cdot\cdot\cdot\cdot\cdot 0 \\
0/1! &= 0\cdot\cdot\cdot\cdot\cdot\cdot 0 \\
0/0! &= 0\cdot\cdot\cdot\cdot\cdot\cdot 0
\end{aligned}
$$

The quotients of each equation indicate the different positions of each digit in the permutation. As calculated above, the quotient (index) sequence is 15110200, which means *a* sits in the *2nd* position of the sequence, *b* sits in the *6th* position of the rest of the sequence, *c* sits in the *2nd* position in the rest of the 6 digits, *d* sits the *2nd* in the rest of the 5 digit sequence, *e* sits the *1st* among the rest 4 digits, *f* sits the *3rd* in the remaining 3 spots, *g* sits in the *1st* of the rest 2 spots, and the last empty digit for the permutation is *h*.

Following this rule, the converted permutation is *eacdghbf*, compared to the original sequence *abcdefgh*. In this way, we can transfer the binary sequence to a specific permutation.

5.3.2 Combining Similar Keys to One Permutation

In practical applications of secret key generation, extracted keys could be quite similar, since the wireless channels associated with the transceivers may change very slowly. In this case, multiple similar keys with minor differences need to be combined together to generate only one permutation. On one hand, it helps to increase the match rate of the permutations obtained by the transmitter and receiver, and thus to better secure data transmission; on the other hand, it lowers the possibility of violent cracking from the eavesdropper by reducing similar keys for data transmission in a consecutive time period.

Providing that we possess n similar or even identical secret keys on both sides, then we compare these n secret keys bit by bit. For each digit, if more 1s appear than 0s, we decide this digit of the combined key be 1, otherwise be 0. Because all these n keys are quite similar, either 1 or 0 should be the majority in this digit in all these n keys. In particular cases, if there happens to be equal numbers of 1 or 0, then we can manually set the digit to be 1 (or 0).

5.4 Secure Data Transmission with Secret Key Generation

Till now we have addressed reliable and efficient protocols to generate secret keys at physical layer and how to utilize theses keys for data scrambling. In this final section, we present a practical application for secret key generation in a wireless network environment. Secret keys generated at physical layer will be applied to secure data transmission and provide efficient access control with a cloud provider in the wireless network. In the end, a simple demonstration of secure data transmission is also presented on iOS devices.

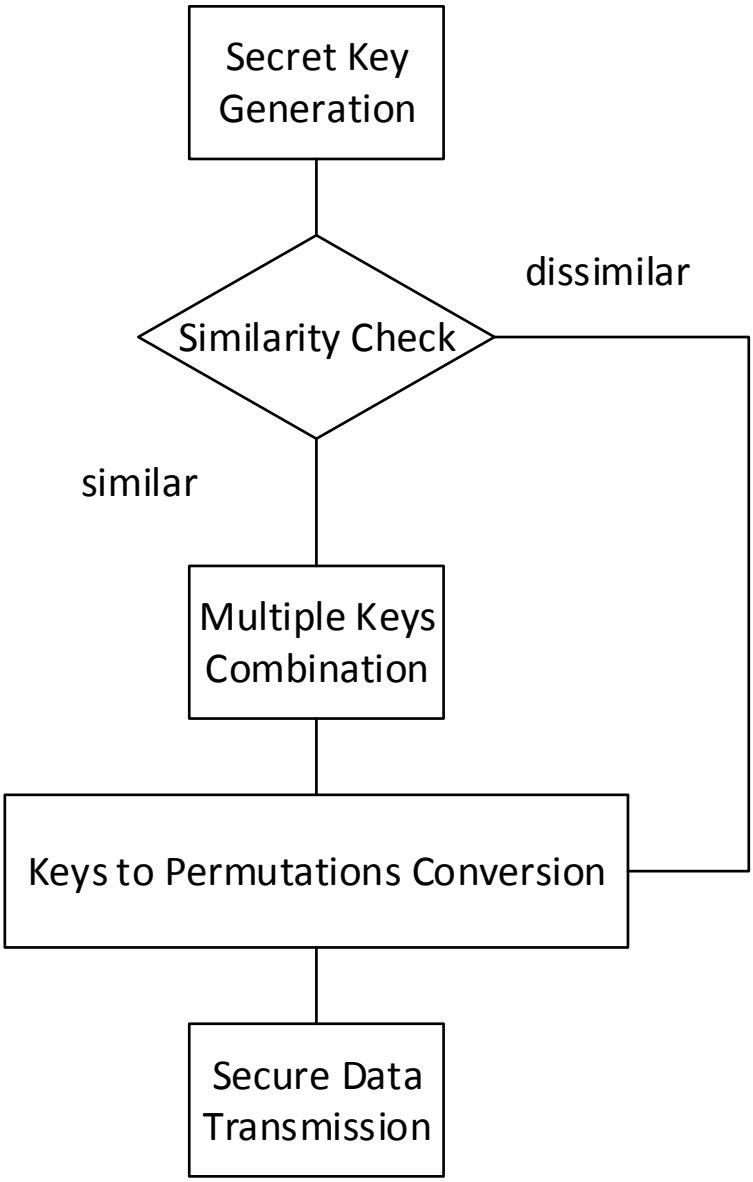

Figure 5.5: Combining similar keys to one permutation

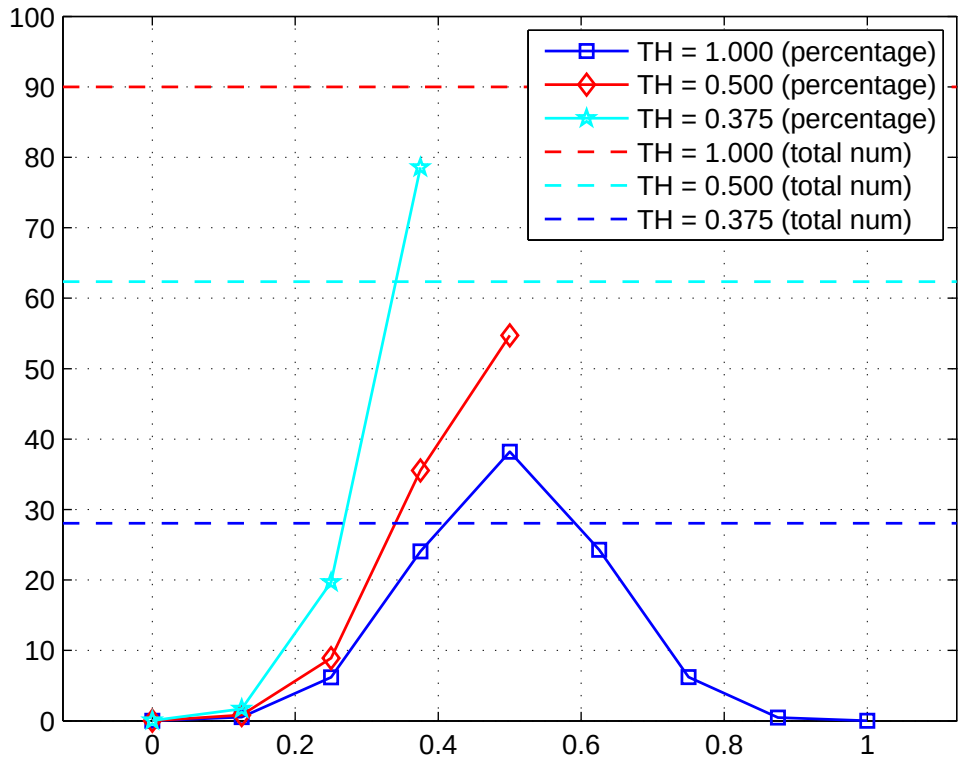

Figure 5.6: Similarity of generated keys with different thresholds

5.4.1 System Scenario

We assume that an honest-but-curious cloud provider, mobile data owner, mobile data sharers, a trusted third party and passive eavesdroppers exist in this scenario for data transmission and data sharing. In the whole process, the cloud provider is reliable enough to perform requested operations such as data encryption, data storage, data transmission, and message broadcasting. It will provide safe storage of the data from being stolen or being manipulated by unauthorized users, but itself might be curious of the plaintext of the data without intentions. Of course, we assume that the cloud provider won't use its resources to crack the data on purpose. A trusted third party can provide and manage critical information regarding key encryption and key distribution, the trusted third party and each authorized mobile user communicate through a secure connection, such as TLS etc. It is also feasible that the trusted third party could be a different cloud provider which is physically isolated from the cloud provider for data storage. Since it is meaningless to just hold the private key without possession of the original data, we could consider this cloud provider as a trusted third party. During the data transmission process, passive eavesdroppers are interested in eavesdropping and capturing the data transmitted between the cloud provider and the mobile users, and they will not perform active attacks to steal or manipulate the data.

Given the practical situation that mobile devices have limited calculation ability, short batter life, and insufficient storage, we assume that the data transmission and data sharing process are both associated with a small amount of data, and all the mobile devices are capable of performing encryption and decryption of the data with real-time service quality. For the cloud provider, we assume that it has unlimited calculation ability and power supply, and it will

always be online. Each mobile user can be recognized by the cloud provider with their own unique identities.

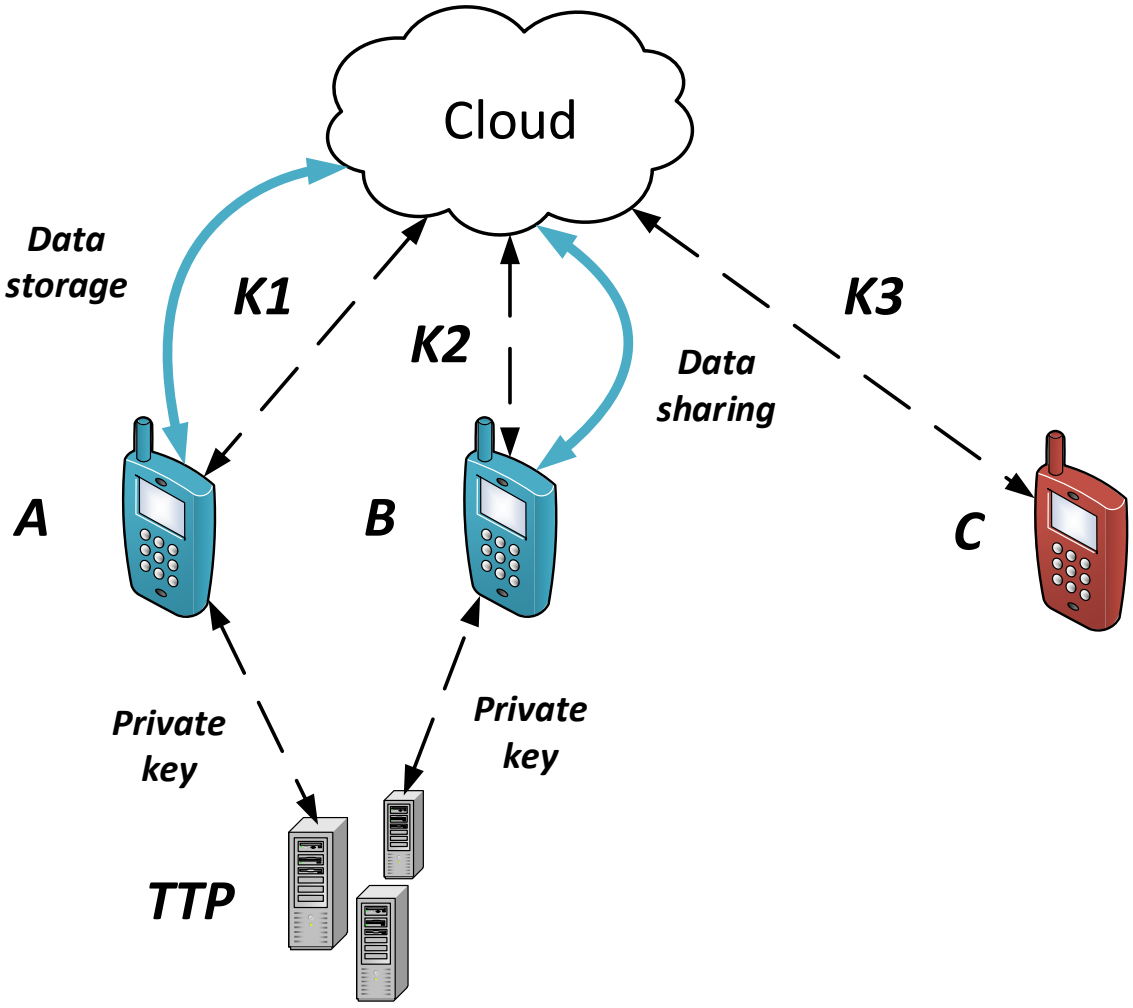

Figure 5.7: Data storage and data sharing model

5.4.2 Secure Data Transmission

During the process of data transmission and data storage, plaintext information can not be revealed to both cloud providers and passive eavesdroppers. Proposed solutions that protect original data information in data transmission and data storage are addressed below.

5.4.2.1 Avoid information leakage to honest-but-curious cloud provider

The trusted third party can provide data owner the private keys to encrypt the original data before further encryption and sending to the cloud provider. The private key is unknown to the cloud provider, which ensures that the plaintext of the data is secured from the cloud provider.

Trusted third party can change the private keys over different periods of time according to the sensitivity of the data. Upon receiving a new private key, data owner encrypts the original data with the new private key, and then performs further encryption and sends the well encrypted data to the cloud provider.

The cloud provider is unaware whether the data already exists in the cloud or not with new data uploading. Data owner can also send requests to remove any data in the cloud. The trusted third party can share the private keys with authorized data sharers upon receiving the request and permission from the data owner.

5.4.2.2 Avoid information leakage to passive eavesdroppers

Once the data uploading request from data owner, or the data sharing request from data sharer are sent to the cloud provider, mobile user and access point of the cloud provider start generating secret keys from the randomness the wireless channels between them. According to the principle of channel reciprocity, theoretically their secret keys are identical. That is to say, the mobile user and the cloud provider possess the same key at the same time. In this case, they don't need to exchange information regarding the secret key itself, which eliminates the possibility of key information leakage to the eavesdroppers.

Since the wireless channels between mobile user and access point of the cloud provider are

changing dynamically due to mobility or environment variation, different keys can be extracted from the wireless channels over time for further data encryption. The data owner can perform second encryption of the data package, which is already encrypted with the private key provided by the trusted third party. When the cloud provider receives the double encrypted data, if it is curious, it can decrypt the data for one step with the secret key generated from the wireless channels, however it cannot further decrypt the data since it has no idea of the private key of the data owner.

As for the passive eavesdroppers who intend to eavesdrop and capture the information transmitted between the mobile user and cloud provider, the wireless channels between themselves and the access point of the cloud provider are quite different from the wireless channel between the authorized mobile user and the cloud provider, even if they are only half of a wavelength away from the authorized mobile users. In this case, the eavesdroppers cannot access the secret keys and decrypt the transmitted data, not mention the encryption keys are changing over time. And of course, they also don't know the private key of the data owner, which makes it even impossible to get the original plaintext of the data.

5.4.3 Secure Access Control

Secure access control of data sharing ensures data availability for authorized mobile users and denies illegal access trials from unauthorized users. With dynamic physical layer secret key generation, the cloud provider and each mobile user possess the same secret keys which are only known to themselves. Of course, these secret keys can be updated over time with the ever changing wireless channels between them, and each new data sharing task will always trig-

ger the secret key generation process to provide independent keys for data encryption among multiple tasks.

We denote the data owner as A, the authorized data sharer as B, and the illegal data sharer as C, all are as shown in Figure 5.7. In addition, the secret keys between the cloud provider and mobile user A, B and C are denoted as $K1$, $K2$, and $K3$, respectively.

When A and B agree to share some data stored in the cloud, A and B both send a request to the cloud provider to share the data, and also send a request to the trusted third party to share the private key of A. Upon receiving both requests, the cloud provider broadcasts the message of key combination $K1 \oplus K2$, and starts transmitting the data which is double encrypted with the private key from A and the secret key $K1$. Since B holds the secret key $K2$, he can calculate $K1$ from the key combination $K1 \oplus K2$. After receiving the double encrypted data, B can decrypt the data with $K1$ and the private key provided by the trusted third party step by step.

If the unauthorized data sharer C camouflages itself with the identity of authorized user B, it might get the double encrypted data and the private key of A, but still, it cannot access the plaintext of the data since it cannot crack $K1$ out of the key combination $K1 \oplus K2$ only with his own key $K3$.

5.4.4 Demonstration of Secure Data Transmission on iOS devices

A simple mobile app on the iOS platform is designed to demonstrate the application of secure data transmission. Users can dynamically set the permutation as Fig. 5.8 shows at the first stage. With the predefined permutation order, transmitter can send the information after data scrambling as shown in Fig. 5.9 and receiver can then successfully decrypt the data as shown

in Fig. 5.10.

5.5 Chapter Summary

In this chapter, a secret key generation scheme with partial quantization is proposed to significantly improve key match rate, where two thresholds in the magnitude based quantization is applied to channel samples. In this case, cross-over errors could be efficiently reduced to further ensure the conformity of channel samples on both sides for key bit extraction. However, the trade off for the increased key match rate due to two thresholds partial quantization is the decreased key generation rate.

A practical application of physical layer secret key generation is presented in the end. Secure data transmission and reliable access control could be both realized with the employment of secret keys extracted from the wireless channels.

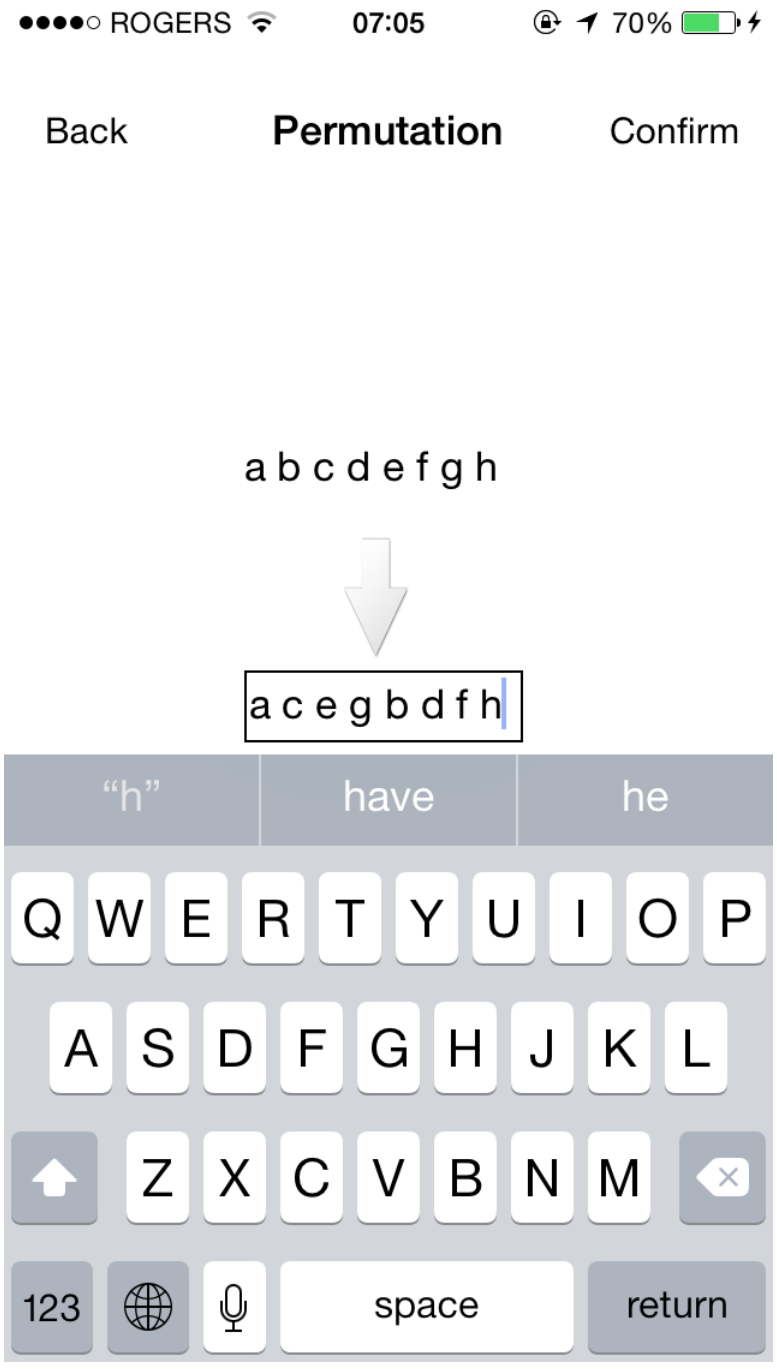

Figure 5.8: iOS App: permutation setting

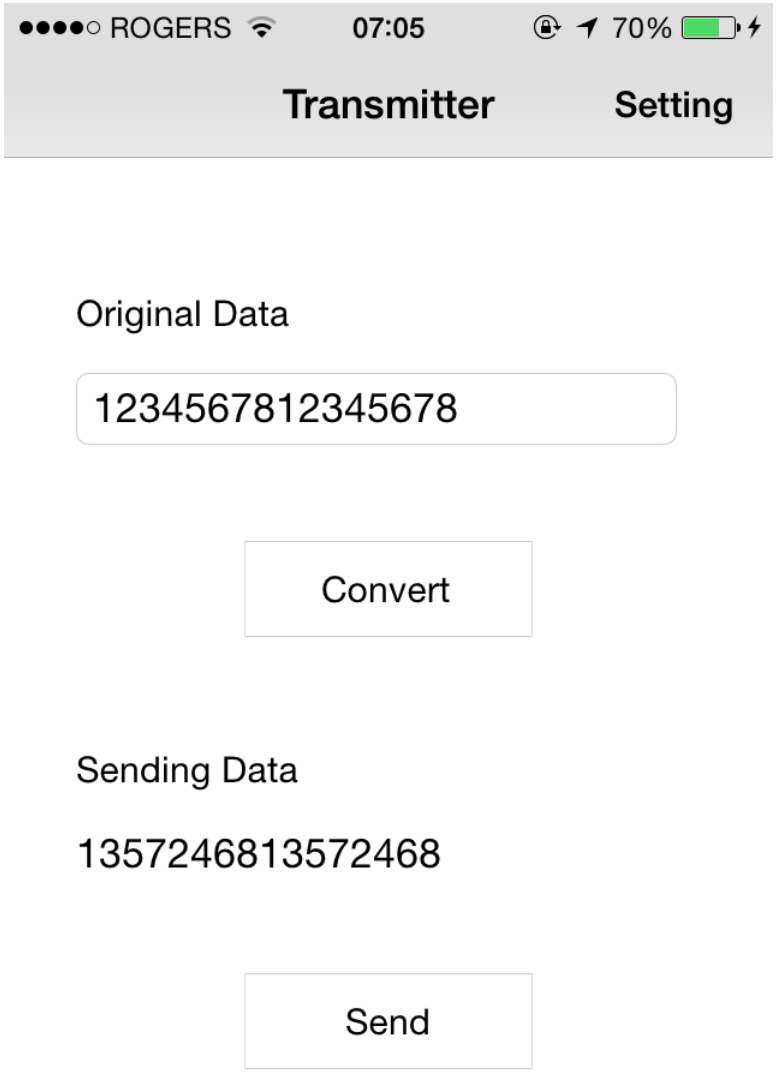

Figure 5.9: iOS App: data encryption and data transmission

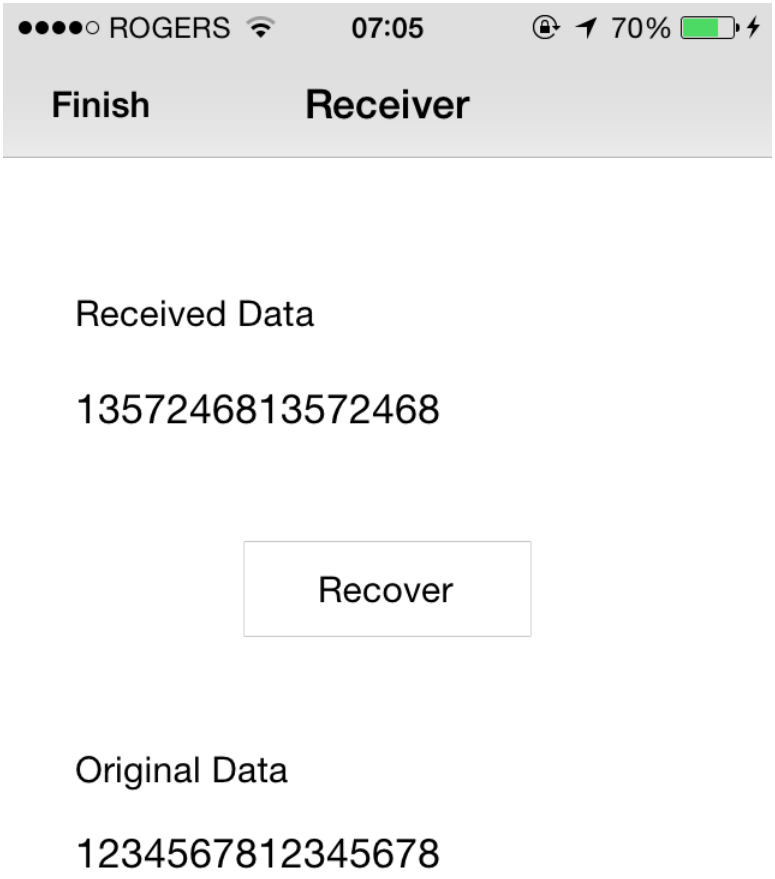

Figure 5.10: iOS App: data receiving and data decryption

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this thesis, the topic in the area of physical layer secret key generation has been discussed mainly in four aspects, including a general review and literature survey of PHY secret key generation protocols, theoretical analysis of the reliability and efficiency of secret key extraction with practical wireless channels, and proposed algorithms that aim to improve key match rate, and in the end a practical application of physical layer secret key generation.

More specifically, a general review of security protocols is first provided with introduction of theory development and significant protocols. From the very first Shannon's perfect secrecy, till Wyner's wire-tap channel and to the up-to-date physical layer secret key generation protocols, PHY based key extraction with the channel model is reviewed as the most promising cryptographic method nowadays. Literature survey on secret key generation at physical layer is also provided.

Secondly, the reliability and efficiency of physical layer secret key generation is examined

with practical wireless channels. Theoretical analysis relates secret key match rate with channel estimation quality. Channel SNRs and degrees to channel reciprocity are both found to have deep influence on the key match rate. One resort that exploits information reconciliation is proved to help achieve required quality of secret key match.

Thirdly, two algorithms aiming to improve key match rate are proposed in the process of channel estimation and sample quantization respectively. Multiple observations of the wireless channels are combined with a linear processor to obtain a synthesized estimation of the channel, and two thresholds in the magnitude based partial quantization are elected to avoid cross level errors, both schemes result in a significant increase in secret key match rate, and both are examined by theoretical analysis and numerical results.

Finally, a practical application of physical layer secret generation is presented in the s-cenario of mobile networks. Secret keys extracted from the wireless channels successfully provide secure data transmission and reliable access control.

6.2 Future Work

Some potential topics in the area of physical layer secret key generation remain uncovered in this thesis, but they are worthy of extra attention and further exploration. Some of the topics are described as follows:

- In the protocols of secret key generation, an error-free channel is utilized for public discussion and information reconciliation, all the messages exchanged between two terminals are carried over the channel without any mistakes. However, this could not be the case in practical wireless communication systems due to environment noise and in-

terference. As a result, the influence of noisy channel on the key reconciliation process remains further investigation.

- In this thesis, we explored the performance of information reconciliation in terms of channel reciprocity, SNR and number of observations. Apparently, one iteration of key reconciliation cannot always achieve identical keys on both sides. In future research, an adaptive information reconciliation process with different conditions of wireless communication systems requires further investigation.

- Most of the existing research assumes a passive eavesdropper in the whole process of secret key extraction. While in practical situations, adversaries and eavesdroppers could not only eavesdrop and capture the information, but also manipulate the messages transmitted between authorized users or even perform interference and brute attacks to the communication system. The protocols of physical layer secret key generation demand extra security approaches to against active eavesdroppers in the scenario.

- Key generation rate reflects the efficiency of secret key generation protocols, but it's not fully covered in this thesis. In the case of slow varying channels, one possible way to increase key generation rate is to utilize relay nodes between two terminals and discover the associated relay channels to obtain extra randomness for secret key extraction.

- Many secret key generation protocols are designed to extract secret keys between two users in a one to one mode. While in some situations a group of users require a common key shared in between. How to generate a group key in a one to many mode could be potential topic in physical layer secret key generation.

Bibliography

[1] C. E. Shannon, "Communication theory of secrecy systems," *Bell system technical journal*, vol. 28, no. 4, pp. 656–715, 1949.

[2] W. Diffie and M. E. Hellman, "New directions in cryptography," *Information Theory, IEEE Transactions on*, vol. 22, no. 6, pp. 644–654, 1976.

[3] A. D. Wyner, "The wiretap channel," *Bell System Technical Journal, The*, vol. 54, no. 8, pp. 1355–1387, 1975.

[4] I. Csiszár and J. Korner, "Broadcast channels with confidential messages," *Information Theory, IEEE Transactions on*, vol. 24, no. 3, pp. 339–348, 1978.

[5] A. O. Hero III, "Secure space-time communication," *Information Theory, IEEE Transactions on*, vol. 49, no. 12, pp. 3235–3249, 2003.

[6] M. Bloch, J. Barros, M. R. Rodrigues, and S. W. McLaughlin, "Wireless information-theoretic security," *Information Theory, IEEE Transactions on*, vol. 54, no. 6, pp. 2515–2534, 2008.

[7] U. M. Maurer, "Secret key agreement by public discussion from common information," *Information Theory, IEEE Transactions on*, vol. 39, no. 3, pp. 733–742, 1993.

[8] R. Ahlswede and I. Csiszár, "Common randomness in information theory and cryptography. part i: secret sharing," *IEEE Transactions on Information Theory*, vol. 39, no. 4, 1993.

[9] A. A. Hassan, W. E. Stark, J. E. Hershey, and S. Chennakeshu, "Cryptographic key agreement for mobile radio," *Digital Signal Processing*, vol. 6, no. 4, pp. 207–212, 1996.

[10] H. Koorapaty, A. Hassan, and S. Chennakeshu, "Secure information transmission for mobile radio," in *Information Theory, 1998. Proceedings. 1998 IEEE International Symposium on*. IEEE, 1998, p. 381.

[11] R. Wilson, D. Tse, R. Scholtz *et al.*, "Channel identification: Secret sharing using reciprocity in ultrawideband channels," *Information Forensics and Security, IEEE Transactions on*, vol. 2, no. 3, pp. 364–375, 2007.

[12] J. E. Hershey, A. Hassan, R. Yarlagadda *et al.*, "Unconventional cryptographic keying variable management," *Communications, IEEE Transactions on*, vol. 43, no. 1, pp. 3–6, 1995.

[13] T. Aono, K. Higuchi, T. Ohira, B. Komiyama, and H. Sasaoka, "Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels," *Antennas and Propagation, IEEE Transactions on*, vol. 53, no. 11, pp. 3776–3784, 2005.

[14] A. Sayeed and A. Perrig, "Secure wireless communications: Secret keys through multipath," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 3013–3016.

[15] I. Csiszár and P. Narayan, "Common randomness and secret key generation with a helper," *Information Theory, IEEE Transactions on*, vol. 46, no. 2, pp. 344–366, 2000.

[16] S. Nitinawarat and P. Narayan, "Secret key generation for correlated gaussian sources," *Information Theory, IEEE Transactions on*, vol. 58, no. 6, pp. 3373–3391, 2012.

[17] G. R. Cooper and C. D. McGillem, *Modern communications and spread spectrum*. McGraw-Hill, Inc., 1986.

[18] A. L. Swindlehurst, "Fixed sinr solutions for the mimo wiretap channel." in *ICASSP*, 2009, pp. 2437–2440.

[19] W. Stallings, *Network and internetwork security: principles and practice*. Prentice Hall Englewood Cliffs, 1995, vol. 1.

[20] N. Döttling, D. Lazich, J. Müller-Quade, and A. S. de Almeida, "Vulnerabilities of wireless key exchange based on channel reciprocity," in *Proceedings of the 11th International Conference on Information Security Applications*, ser. WISA'10. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 206–220. [Online]. Available: http://dl.acm.org/citation.cfm?id=1949945.1949964

[21] M. Zafer, D. Agrawal, and M. Srivatsa, "Limitations of generating a secret key using wireless fading under active adversary," *IEEE/ACM Transactions on Networking (TON)*, vol. 20, no. 5, pp. 1440–1451, 2012.

[22] 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*. ACM, 2011, p. 8.

[23] A. Kitaura and H. Sasaoka, "A scheme of private key agreement based on the channel characteristics in ofdm land mobile radio," *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, vol. 88, no. 9, pp. 1–10, 2005.

[24] M. G. Madiseh, M. L. McGuire, S. S. Neville, L. Cai, and M. Horie, "Secret key generation and agreement in uwb communication channels," in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*. IEEE, 2008, pp. 1–5.

[25] M. G. Madiseh, S. He, M. L. McGuire, S. W. Neville, and X. Dong, "Verification of secret key generation from uwb channel observations," in *Communications, 2009. ICC'09. IEEE International Conference on*. IEEE, 2009, pp. 1–5.

[26] 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*. ACM, 2007, pp. 401–410.

[27] A. Kitaura, T. Sumi, K. Tachibana, H. Iwai, and H. Sasaoka, "A scheme of private key agreement based on delay profiles in uwb systems," in *Sarnoff Symposium, 2006 IEEE*. IEEE, 2006, pp. 1–6.

[28] C. Ye, A. Reznik, and Y. Shah, "Extracting secrecy from jointly gaussian random variables," in *Information Theory, 2006 IEEE International Symposium on*. IEEE, 2006, pp. 2593–2597.

[29] C. Ye, A. Reznik, G. Sternberg, and Y. Shah, "On the secrecy capabilities of itu channels," 2007.

[30] J. Wallace, "Secure physical layer key generation schemes: performance and information theoretic limits," in *Communications, 2009. ICC'09. IEEE International Conference on*. IEEE, 2009, pp. 1–5.

[31] 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.

[32] X. Sun, X. Wu, C. Zhao, M. Jiang, and W. Xu, "Slepian-wolf coding for reconciliation of physical layer secret keys," in *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*. IEEE, 2010, pp. 1–6.

[33] N. Patwari, J. Croft, S. Jana, and S. K. Kasera, "High-rate uncorrelated bit extraction for shared secret key generation from channel measurements," *Mobile Computing, IEEE Transactions on*, vol. 9, no. 1, pp. 17–30, 2010.

[34] 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.

[35] Q. Wang, H. Su, K. Ren, and K. Kim, "Fast and scalable secret key generation exploiting channel phase randomness in wireless networks," in *INFOCOM, 2011 Proceedings IEEE*. IEEE, 2011, pp. 1422–1430.

[36] M. Tope, J. C. McEachen *et al.*, "Unconditionally secure communications over fading channels," in *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, vol. 1. IEEE, 2001, pp. 54–58.

[37] M. Wilhelm, I. Martinovic, and J. B. Schmitt, "Secret keys from entangled sensor motes: implementation and analysis," in *Proceedings of the third ACM conference on Wireless network security*. ACM, 2010, pp. 139–144.

[38] K. Zeng, D. Wu, A. J. Chan, and P. Mohapatra, "Exploiting multiple-antenna diversity for shared secret key generation in wireless networks," in *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.

[39] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam, "Information-theoretically secret key generation for fading wireless channels," *Information Forensics and Security, IEEE Transactions on*, vol. 5, no. 2, pp. 240–254, 2010.

[40] Z. Li, W. Xu, R. Miller, and W. Trappe, "Securing wireless systems via lower layer enforcements," in *Proceedings of the 5th ACM workshop on Wireless security*. ACM, 2006, pp. 33–42.

[41] J. W. Wallace, C. Chen, M. Jensen *et al.*, "Key generation exploiting mimo channel evolution: Algorithms and theoretical limits," in *Antennas and Propagation, 2009. EuCAP 2009. 3rd European Conference on*. IEEE, 2009, pp. 1499–1503.

[42] C. Chen, M. Jensen *et al.*, "Secrecy extraction from increased randomness in a time-variant mimo channel," in *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*. IEEE, 2009, pp. 1–6.

[43] ——, "Improved channel quantization for secret key establishment in wireless systems," in *Wireless Information Technology and Systems (ICWITS), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–4.

[44] M. Bloch, A. Thangaraj, S. W. Mc Laughlin, and J.-M. Merolla, "Ldpc-based gaussian key reconciliation," in *2006 IEEE Information Theory Workshop*, no. 1633793, 2006, pp. 116–120.

[45] M. Bloch, A. Thangaraj, S. W. McLaughlin, and J.-M. Merolla, "Ldpc-based secret key agreement over the gaussian wiretap channel," in *Information Theory, 2006 IEEE International Symposium on*. IEEE, 2006, pp. 1179–1183.

[46] C. Cachin and U. M. Maurer, "Linking information reconciliation and privacy amplification," *journal of Cryptology*, vol. 10, no. 2, pp. 97–110, 1997.

[47] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," DTIC Document, Tech. Rep., 2001.

[48] K. Ren, H. Su, and Q. Wang, "Secret key generation exploiting channel characteristics in wireless communications," *Wireless Communications, IEEE*, vol. 18, no. 4, pp. 6–12, 2011.

[49] 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*, ser. CCS '07. New York, NY, USA: ACM, 2007, pp. 401–410.

[50] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Security and Privacy, 2003. Proceedings. 2003 Symposium on*, 2003, pp. 197–213.

[51] H. Liu, Y. Wang, J. Yang, and Y. Chen, "Fast and practical secret key extraction by exploiting channel response," in *INFOCOM, 2013 Proceedings IEEE*, 2013, pp. 3048–3056.

[52] S. Premnath, J. Suman, J. Crof, L. Gowda, M. Clark, K. Kasera, N. Patwari, and S. Krishnamurthy, "Secret key extraction from wireless signal strength in real environments," *IEEE Trans. Mobile Comput.*, vol. 12, no. 5, pp. 917–930, 2013.

[53] G. Brassard and L. Salvail, "Secret-key reconciliation by public discussion," in *Advances in Cryptology EUROCRYPT-93*, ser. Lecture Notes in Computer Science, T. Helleseth, Ed. Springer Berlin Heidelberg, 1994, vol. 765, pp. 410–423.

[54] D.-S. Yoo and W. E. Stark, "Characterization of wssus channels: normalized mean square covariance and diversity combining," *IEEE transactions on wireless communications*, vol. 4, no. 4, pp. 1307–1310, 2005.

[55] H. L. Van Trees, *Detection, estimation, and modulation theory*. John Wiley & Sons, 2004.

[56] G. Brassard and L. Salvail, "Secret-key reconciliation by public discussion," in *advances in CryptologyEUROCRYPT93*. Springer, 1994, pp. 410–423.

[57] I. Csiszár and P. Narayan, "Secrecy capacities for multiple terminals," *Information Theory, IEEE Transactions on*, vol. 50, no. 12, pp. 3047–3061, 2004.

[58] J. Chen, D.-k. He, and E.-h. Yang, "On the codebook-level duality between slepian-woif coding and channel coding," in *Information Theory and Applications Workshop, 2007*. IEEE, 2007, pp. 84–93.

[59] T. P. Coleman, A. H. Lee, M. Médard, and M. Effros, "Low-complexity approaches to slepian&# 8211; wolf near-lossless distributed data compression," *Information Theory, IEEE Transactions on*, vol. 52, no. 8, pp. 3546–3561, 2006.

[60] J. Garcia-Frias, "Compression of correlated binary sources using turbo codes," *Communications Letters, IEEE*, vol. 5, no. 10, pp. 417–419, 2001.

[61] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using ldpc codes," *Communications Letters, IEEE*, vol. 6, no. 10, pp. 440–442, 2002.

[62] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (discus): Design and construction," *Information Theory, IEEE Transactions on*, vol. 49, no. 3, pp. 626–643, 2003.

[63] C. H. Bennett, G. Brassard, and J.-M. Robert, "Privacy amplification by public discussion," *SIAM journal on Computing*, vol. 17, no. 2, pp. 210–229, 1988.

[64] C. H. Bennett, G. Brassard, C. Crépeau, and U. M. Maurer, "Generalized privacy amplification," *Information Theory, IEEE Transactions on*, vol. 41, no. 6, pp. 1915–1923, 1995.

[65] J. L. Carter and M. N. Wegman, "Universal classes of hash functions," in *Proceedings of the ninth annual ACM symposium on Theory of computing*. ACM, 1977, pp. 106–112.

[66] U. Maurer and S. Wolf, "Secret-key agreement over unauthenticated public channels. ii. privacy amplification," *Information Theory, IEEE Transactions on*, vol. 49, no. 4, pp. 839–851, 2003.

[67] M. N. Wegman and J. L. Carter, "New hash functions and their use in authentication and set equality," *Journal of computer and system sciences*, vol. 22, no. 3, pp. 265–279, 1981.

[68] R. Cramer, Y. Dodis, S. Fehr, C. Padró, and D. Wichs, "Detection of algebraic manipulation with applications to robust secret sharing and fuzzy extractors," in *Advances in Cryptology–EUROCRYPT 2008*. Springer, 2008, pp. 471–488.

[69] Y. Dodis, J. Katz, L. Reyzin, and A. Smith, "Robust fuzzy extractors and authenticated key agreement from close secrets," in *Advances in Cryptology-CRYPTO 2006*. Springer, 2006, pp. 232–250.

[70] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in cryptology-Eurocrypt 2004*. Springer, 2004, pp. 523–540.

[71] R. Raz, O. Reingold, and S. Vadhan, "Extracting all the randomness and reducing the error in trevisan's extractors," in *Proceedings of the thirty-first annual ACM symposium on Theory of computing*. ACM, 1999, pp. 149–158.

[72] C. Ye and P. Narayan, "Secret key and private key constructions for simple multiterminal source models," *Information Theory, IEEE Transactions on*, vol. 58, no. 2, pp. 639–651, 2012.

[73] 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.

[74] M. Simon and M.-S. Alouini, *Digital Communication over Fading Channels:A Unified Approach to Performance Analysis*. New York: John Wiley & Sons, 2000.

[75] L. Lai, Y. Liang, and H. V. Poor, "A unified framework for key agreement over wireless fading channels," *Information Forensics and Security, IEEE Transactions on*, vol. 7, no. 2, pp. 480–490, 2012.

[76] C. Balanis, *Antenna Theory: Analysis and Design*. New York: John Wiley & Sons, 2012.

[77] M. Biguesh and A. Gershman, "Training-based MIMO channel estimation: a study of estimator tradeoffs and optimal training signals," *IEEE Trans. Signal Process.*, vol. 54, pp. 884–893, March 2006.

[78] J. Cavers, "An analysis of pilot symbol assisted modulation for Rayleigh fading channels," *IEEE Trans. Veh. Technol.*, vol. 40, no. 4, pp. 866–693, November 1991.

[79] S. Primak, K. Liu, and X. Wang, "Secret key generation using physical channels with imperfect csi," in *Proc. VTC-2014 Fall*, 2014.

[80] D. Lehmer, "Teaching combinatorial tricks to a computer," in *Proc. Symposium on Applied Mathematicical Combinatorial Analysis*, 1960.

[81] Middleton, *Introduction to Statistical Communications Theory*, 1st ed. New York: McGraw-Hill, 1960.

Appendix A

Equations of Theorems

A.1 Probabilities of key mismatch

Jointly two dimensional distribution of two correlated Gaussian vectors is well known [81] and given by

$$p_2(I_A, I_B) = \frac{1}{2\sigma^2(1-\rho^2)} \exp\left[-\frac{I_A + I_B}{\sigma^2(1-\rho^2)}\right] \times I_0\left(\frac{2\rho}{1-\rho^2}\frac{\sqrt{I_A I_B}}{\sigma^2}\right) \tag{A.1}$$

A.2 Quantization levels

If it is required to partition the range of values of I_A into Q equally probable intervals, such partition could be achieved at levels I_q such that

$$1 - \exp\left(-\frac{I_q}{\sigma^2}\right) = \frac{q}{Q},\ q = 1, \cdots, Q-1,\ I_0 = 0,\ I_Q = \infty \tag{A.2}$$

i.e.

$$I_q = \sigma^2 \ln\frac{Q}{Q-q},\ q = 1, \cdots, Q \tag{A.3}$$

In particular, if $Q = 2$, $I_1 = \sigma^2 \ln 2$.

A.3 Key bit mismatch after quantization

The probability of key mismatch could be expressed in terms of Marcum Q function [74]. In this case, CDF $P(x, y)$ of the standardized joint exponential distribution is given by

$$\begin{aligned} P(x,y) = \text{Prob}\,(I_A < x, I_B < y) = 1 - \exp(-x) Q_1\left(\sqrt{\frac{2y}{1-\rho^2}}, \sqrt{\frac{2\rho^2 x}{1-\rho^2}}\right) \\ - \exp(-y)\left[1 - Q_1\left(\sqrt{\frac{2\rho^2 y}{1-\rho^2}}, \sqrt{\frac{2x}{1-\rho^2}}\right)\right] \end{aligned} \tag{A.4}$$

Curriculum Vitae

Name:	Kang Liu
Post-Secondary Education and Degrees:	The University of Western Ontario London, Canada 2013 - 2015 M.E.Sc Central South University Changsha, China 2009 - 2013 B.Eng.
Related Work Experience:	Teaching Assistant The University of Western Ontario 2013 - 2015

Publications:

Kang Liu, Serguei Primak, Xianbin Wang, "On Secret Key Generation From Multiple Observations of Wireless Channels", *IEEE International Conference on Communication Systems*, IEEE ICCS 2014.

Serguei Primak, **Kang Liu**, Xianbin Wang, "Secret Key Generation Using Physical Channels with Imperfect CSI", *IEEE Vehicular Technology Conference*, IEEE VTC Fall 2014.

REN Ju, ZHANG Yaoxue, **LIU Kang**, "Multiple *k*-hop clusters based routing scheme to preserve source-location privacy in WSNs", *Journal of Central South University*, August 2014, Volume 21, Issue 8, pp 3155-3168.

Ju Ren, Yaoxue Zhang, **Kang Liu**, "An Energy-Efficient Cyclic Diversionary Routing Strategy against Global Eavesdroppers in Wireless Sensor Networks", *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 834245, 15 pages, 2013.