2010

Analysis of symmetric key establishment based on reciprocal channel quantization

David Wagner

Analysis of Symmetric Key Establishment Based on Reciprocal Channel Quantization

By

David M. Wagner

A Thesis Submitted
in
Partial Fulfillment of the
Requirements for the Degree of
Master of Science
in
Electrical Engineering

Supervised by:

Gill R. Tsouri,
Assistant Professor,
Dept. of Electrical and Microelectronic Engineering

Department of Electrical and Microelectronic Engineering

Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, New York

November 2010

Abstract

Analysis of Symmetric Key Establishment based on Reciprocal Channel Quantization

David M. Wagner

Supervising Professor: Dr. Gill R. Tsouri

Methods of symmetric key establishment using reciprocal quantization of channel parameters in wireless Rayleigh and Rician fading channels are considered. Two important aspects are addressed through generic analysis: impact of a proximity attack by a passive eavesdropper and achievable key establishing rates. The analysis makes use of the National Institute of Standards and Technology statistical test suite applied to the channel quantization bits as the outputs of a random number generator. For proximity attacks, a passive mobile eavesdropper with an ability to approach one of the communicating parties and a possible signal-to-noise ratio advantage is assumed. The minimal required distance from the eavesdropper in order to maintain perfect secrecy during key establishment is evaluated as a function of the Rician factor and quantization depth. For key establishing rates, the maximal rates are evaluated while ensuring that the generated secret key bits pass the entire statistical test suite. The generic analysis is applied to channel-phase quantization and performance in practical systems is considered as well.

Table of Contents

LIST OF FIGURES **5**

LIST OF TABLES **6**

SUMMARY OF CONTRIBUTIONS **7**

CHAPTER 1: INTRODUCTION **8**

1.1 Motivation **8**

1.2 Literature Survey **9**

1.3 Novelty **9**

1.4 Outline **10**

CHAPTER 2: BACKGROUND **12**

2.1 Theoretic Secrecy **12**

2.2 Random Number Generators **14**

2.3 Random Number Generator Evaluators **16**

CHAPTER 3: ANALYSIS **20**

3.1 Opening Remarks **20**

3.1.1 Proximity Attacks 21

3.1.2 Key Establishing Rates 23

3.2 Analysis of Key Establishing Rates in Rayleigh fading 23

3.3 Analysis of Proximity Attacks in Rayleigh fading 25

3.4 Analysis of Proximity Attacks in Rician fading 26

3.4.1 Supporting Lemma for ensuring independent eavesdropper channels 27

3.5 Analysis of Key Establishing Rates in Rician fading 30

3.6 Carrier-phase Quantization 32

3.6.1 Proximity Attacks using Carrier-Phase Quantization 32

3.6.2 Key Establishing Rates using Carrier-Phase Quantization 35

CHAPTER 4: CONCLUSION 39

4.1 Closing Remarks 39

4.2 Improvements and Future Work 40

REFERENCES 42

List of Figures

FIG. 1 – COMMUNICATION SYSTEM WITH EAVESDROPPER 13

FIG. 2 –RICIAN CHANNEL CORRELATION IN TIME 31

FIG. 3– MINIMUM REQUIRED DISTANCE AS FUNCTION OF RICIAN FACTOR FOR VARIOUS QUANTIZATION BITS 34

FIG. 4 – MAXIMUM KEY REFRESHING RATES AS A FUNCTION OF QUANTIZED BIT POSITION FOR $K = 0, 1, 3, 5, 10$. 37

List of Tables

TAB. 1 – NIST STATISTICAL TESTS **17**

TAB. 2 – PARAMETERS FOR NIST TESTS **36**

Summary of Contributions

- Generic approach of analyzing eavesdropper proximity attacks on key establishment methods that use reciprocal quantization of channel parameters. The analysis determines the minimal required distance from an eavesdropper to maintain perfect secrecy while establishing the key.
- Generic approach of analyzing achievable key refreshing rates for key establishment methods that use reciprocal quantization of channel parameters. The analysis determines maximum key establishing rates while insuring that the resulting key is a true random sequence.
- Determination of secure eavesdropper-receiver distances and key refreshing rates for carrier-phase quantization in Rician fading environments.
- Implementation in Matlab of entire NIST 2008 statistical test suite for Cryptographic Random Number Generators. The Matlab code would be made available online at:

  http://people.rit.edu/grteee/communicationLab.html

- Publication:

  D. Wagner and G. R. Tsouri, "Analysis of Symmetric Key Establishment Based on Reciprocal Channel Quantization: Proximity Attacks and Key Establishing Rates", submitted for review to *IEEE Transactions on Information Forensics and Security – special issue on Physical Layer Security*.

Chapter 1: Introduction

1.1 Motivation

The broadcast nature of wireless communication links exposes them to eavesdropping and therefore securing a wireless link is paramount in many applications. In traditional symmetric encryption systems, a large pre-deployed secret key is shared by the two communicating parties. The same key is used to encrypt and decrypt information. A prominent example is the *Advanced Encryption Standard* (AES) [1], where a 128 bit key is typically used. Asymmetric encryption is based on public-key cryptography where the public key is not secret and is used to encrypt information. Decryption can only be performed using a private key which is secretly kept. A prominent example is the Rivest-Shamir-Adleman (RSA) [2] algorithm. Both types of encryption methods are based on security by complexity and provide adequate security. Symmetric methods are characterized by lower algorithmic complexity, while asymmetric methods are characterized by lower key management complexity. To minimize complexity one could use a simpler symmetric encryption method such as a stream cipher [3] coupled with periodic key establishing to compensate for its weak encryption strength. To this end a method of securely establishing a symmetric encryption key is needed. A prominent method used in practice is the Diffie-Hellman algorithm [4] which reintroduces high algorithmic complexity.

AES and the Diffie-Hellman algorithm involve the use of considerable online computation power, memory space and communication overhead. Therefore, these methods could prove impractical in resource-constrained devices such as implanted medical devices, compact mobile devices and wireless enabled bio-sensors. The costs associated with securing a wireless link in resource-constrained devices received considerable attention in the past – see [5] for example. A

low-complexity alternative for establishing a symmetric key is attractive provided that it is secure from eavesdropping. Such an alternative would allow the use of low-complexity symmetric encryption coupled with frequent key establishment.

1.2 Literature Survey

In recent years there has been increased attention to the use of wireless physical layer security to establish information theoretic security as a low cost alternative to standard encryption methods which are based on computational complexity such as AES. Previous work on the secrecy capacity of wireless fading channels showed they have an intrinsic property of concealing information from an eavesdropper – see [6-12] for prominent examples. In addition, the literature depicts many attempts to practically use the secrecy-capacity to implement information theoretic security - see [13-21] and references therein for examples. We focus our attention on methods of symmetric key generation based on reciprocal quantization of channel parameters such as those reported in [15-21]. In [15] knowledge of the channel-phase is used to encrypt data with some arbitrary quantization. In [16] reciprocal random fluctuations in the signal amplitudes are quantized to generate keys. In [17] key generating is simulated for ultra wideband channels, while in [18-21] the channel phase and/or amplitudes are directly quantized to generate secret key bits.

1.3 Novelty

In this contribution we propose two generic analysis approaches applicable to key establishment methods which are based on reciprocal quantization of channel parameters. The first approach is for assessing the impact of proximity attacks by a mobile passive eavesdropper with possible

Signal to Noise Ratio (SNR) advantage. The second approach is for evaluating achievable key establishing rates. For the scope of this work we consider Rician fading channels, a passive eavesdropper, and no quantization errors. Note that quantization errors and key establishing rates are intimately tied, since failures to establish a key due to quantization errors means the communicating parties must perform multiple attempts to establish the key resulting in slower establishing rates. It follows that the analysis results in an upper bound on key establishment rates. Our analysis makes use of the Rician channel model reported in [22], the *National Institute of Standards and Technology* (NIST) random number generator test suite [23], a supporting lemma we define and prove, and Clarke's Rayleigh channel model in [24]. The model in [22] offers high accuracy with regard to the random nature of the Rician factor and was successfully used in the past to model Rician fading channels, and the NIST test suite [23] is used extensively to evaluate many cryptographic random number generators. We are unaware of previous attempts to use the NIST test suite to quantitatively evaluate the limits of key generating methods based on channel randomness. As an example, we apply the generic approach to key establishment based on single antenna reciprocal channel-phase quantization and use the result to evaluate the applicability for practical systems and standards.

1.4 Outline

Chapter 2 details the foundational concepts in cryptography and information theory required for analysis. Chapter 3 presents analysis of proximity attacks and key refreshing rates in a strong multipath environment with no line-of-sight (based on Clarke's Rayleigh fading model) and in multipath with line-of-sight channels (based on the novel Pop-Beaulieu [22] Rician fading model).

We consider the three *Industrial-Scientific-Medical* (ISM) frequency bands around 433MHz, 915MHz and 2.45GHz. Chapter 4 concludes the treatment and provides direction for future work.

Chapter 2: Background

2.1 Theoretic Secrecy

Most secrecy systems today rely on practical security by using a large key; for example AES uses a 128-bit key. For these systems, a brute force attack by an eavesdropper would require exhaustive search through 2^{128} different possible keys. Even with today's computational resources, the search duration would exceed the system's lifetime. However, these systems can theoretically be compromised since the eavesdropper gains a modicum of information from each ciphertext sample available to him.

In 1946, Claude Shannon published a seminal paper [25] on secrecy systems which addresses achieving theoretic secrecy in the presence of an eavesdropper with unlimited resources. In a keyspace $K = \{k_i\}_{i=1}^{|K|}$, a message space $M = \{m_j\}_{j=1}^{|M|}$, and a cryptogram space $E = \{e_k\}_{k=1}^{|E|}$, the function f_K is a rule that assigns elements of M to elements of E. For theoretic secrecy, the probability of e_k occurring must be the same as the probability of e_k occurring given that any m_j occurred previously, or $P_{m_j}(e_k) = P(e_k)$. Therefore, two conditions must be met: $|E| \geq |M|$ and $|K| \geq |M|$. If both of these are met, we may construct a mapping f_K that ensures $P_{m_j}(e_k) = P(e_k)$. All plaintext messages are equiprobable and so the eavesdropper may not glean any information from any individual ciphertext, and therefore from unlimited ciphertexts.

In most practical systems $|K| < |M|$ and so $P_{m_j}(e_k) \neq P(e_k)$. In this case, by intercepting a cryptogram the eavesdropper will obtain information about the probability distribution of the messages. The unicity distance is defined in [25] as the number of cryptograms required by the

eavesdropper to uniquely determine the message by using cryptanalysis. In systems without theoretic secrecy, the unicity distance is a finite number.

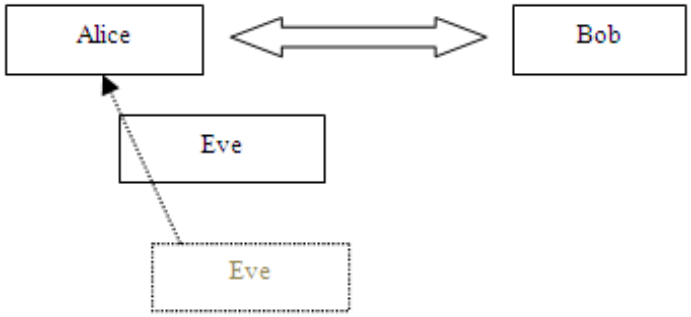

Fig. 1 – Communication System with Eavesdropper

In Fig. 1, we apply the concept of theoretic secrecy to a wireless communication system. An eavesdropper (Eve) is attempting to understand the communication between the transmitter (Alice) and the receiver (Bob). The channel formed between Alice and Bob is designated h_{Bob}, and the channel between Alice and Eve is h_{Eve}. Since the communication is over-the-air, Eve is able to receive the signal with her antenna. The secrecy capacity of the Alice-Bob channel is defined as the maximum quantity of information that may be transmitted over the channel with theoretic secrecy. Previous work [9] has shown the secrecy capacity of the Alice-Bob channel to be

$$C_s = \begin{cases} C_2 - C_1; & C_2 > C_1 \\ 0; & C_2 < C_1 \end{cases} \quad (1)$$

where C_{Eve} designates the channel capacity in [bits/Sec] of Eve's channel and C_{Bob} represents the channel capacity of Bob's channel. For a Gaussian Identity [9] Channel, we have

$$C = \log_2\left(1 + \frac{S}{N}\right) \quad (2)$$

where $\frac{S}{N}$ denotes the Signal-to-Noise Ratio (SNR) in which S is signal power and N is noise power. For a wireless channel, this relation is invalid because of the multipath, but C is still proportional to SNR. Assuming Eve has unlimited resources, she can design an optimum antenna and have a signal with extremely high SNR, and therefore $C_{Eve} \to \infty$. This would indicate that theoretic secrecy is impossible with a powerful eavesdropper. However, the relation in (1) does not hold if h_{Eve} and h_{Rx} are independent [21]. If the channels are independent, Eve's unicity distance will remain at infinity even if she gains an arbitrarily large SNR advantage.

2.2 Random Number Generators

A *True Random Number Generator* (TRNG) is an information source whose instantaneous outputs are selected from the states of an underlying random process. TRNGs are often based on observations of physical phenomena, for example the alpha emissions in a radioactive decay process, and measurements of atmospheric noise. Humans have many applications for TRNGs, including Monte-Carlo simulations of physical phenomena, random sampling among a population, generation of keys in cryptography, selecting lottery winners, and even for creation of content in the arts. However, harnessing physical processes is challenging and often does not provide the demanded quantity of random data. Also, the concept of randomness is counterintuitive to the human brain and thus cannot be synthesized by man. Therefore, humans have thoroughly investigated and developed deterministic means of approximating TRNGs. These generators are termed *Pseudo Random Number Generators* (PRNG). PRNGs produce a stream of numbers that strive to emulate properties of randomness. Starting with an initial number seed, each next number is generated by a deterministic transformation on the previous number.

A simple example of a PRNG is a *Linear Feedback Shift Register* (LFSR [26]). The LFSR of order n generates each n-bit number as a function of the previous number according to the *exclusive or* (XOR) gate connections between the registers. Depending on the initial seed, the LFSR progresses through different cycles of states. A LFSR which produces a maximal length sequence of $2^n - 1$ is called an m-sequence generator. The XOR connections of any m-sequence generator correspond to a primitive polynomial.

Another simple PRNG is the *Linear Congruential* Generator (LCG) [26], which generates subsequent numbers as residues of the previous number weighted and shifted by a constant value. Its deterministic expression is $X_{n+1} = aX_n + c\ (mod\ m)$, and it starts with a seed X_0. Even with carefully chosen values for a, c, m, X_0 the sequence has a period of at most m.

PRNGs can also be complex, consisting of a series of cumbersome deterministic transformations. One example is the Mersenne-Twister algorithm [27], which is currently implemented in Matlab as the rand() function. The Mersenne-Twister algorithm is a computationally intensive PRNG which has a period of $2^{19937} - 1$.

In some cases, it is desirable to have pseudo-randomness rather than pure randomness. For example, in *Code-Division-Multiple-Access* (CDMA) systems, *Pseudo*-Noise (PN) spreading sequences are used for coding and decoding messages for an individual user.

2.3 Random Number Generator Evaluators

Due to the high demand for random data, much research has been conducted on identifying previously untapped TRNGs and also on creating new PRNGs. Since humans cannot intuitively judge randomness, a need has arisen for RNG assessors which accurately determine where a particular RNG stands on the spectrum between deterministic and random. Humans do understand properties of deterministic sequences, and so these assessors are designed to filter out RNGs that generate sample sequences with deterministic properties. Typically, the assessors consist of a battery of tests, each of which detects a different type of underlying determinism or predictability.

One such RNG assessor is the *National Institute of Standards and Technology* (NIST) statistical test suite [23].The NIST statistical test suite consists of multiple tests designed to evaluate the effectiveness of a RNG which is specifically meant for use in cryptographic applications. The suite consists of 15 unique tests, each of which judges the randomness of an incoming bitstream, and returns one or more P-values. These values are typically obtained by transforming the input sequence and observing some properties of it, and then performing a chi-squared test to compare to the expected properties of a truly random sequence. The chi-squared test ensures that the sum of probabilistically weighted squares of the differences between the observed and expected values is less than a certain threshold. Statistically, the P-values represent the strength of the evidence against the null hypothesis; which is that the sequence is nonrandom. For each P-value, the sequence is statistically random with a significance level of α if $P_{value} \geq \alpha$. However, a Type I error can occur if a random sequence produces a P-value below the significance level. Also, a nonrandom sequence may occasionally produce a P-value which passes, which is a Type II Error. In order to reduce the effect of these statistical errors, NIST specifies [23] that at least $\frac{1}{\alpha}$ sequences be tested. To determine whether a generator is indeed random, one may either conduct a chi-square

test on the produced P-values to assess their uniformity, or simply observe whether the percentage of passing P-values is above a specified threshold determined by α.

The NIST tests are not completely independent in terms of the aspects of non-randomness they catch. They also don't span the entire testing space, since no battery of tests could conclusively prove that a sequence is random. Nonetheless, they are the industry standard of RNG and PRNGs, especially for those generators to be used in cryptographic applications.

Test Number	Test Title
1	Frequency
2	Block Frequency
3	Runs
4	Longest-run-of-ones in a block
5	Binary Matrix Rank
6	Discrete Fourier Transform
7	Non-overlapping Template Matching
8	Overlapping Template Matching
9	Maurer's "Universal Statistical"
10	Linear Complexity
11,12	Serial
13	Approximate Entropy
14, 15	Cumulative Sums
16-23	Random Excursions
24-41	Random Excursions Variant

Tab. 1 – NIST statistical tests

Tab. 1 shows a list of the tests available in the suite. Each test is designed to filter out a different kind of non-randomness. The Frequency test is the simplest one and can be used as a filter before applying any of the other tests. It detects whether the distribution of zeros to ones is uniform enough for randomness. The Block Frequency test assesses the uniformity of the bits in local blocks which are subsets of the bitstream. The Runs test detects abnormally large or small streaks of ones, and the Longest-run-of-ones-in-a-Block test is a local version of this test within blocks.

The Spectral test rejects sequences that have repetitive patterns. The Template Matching tests detect whether the frequency of occurrence of a specified bit sequence is atypical of that of a random sequence. The Universal Statistical test determines if the sequence's entropy is consistent with its length, i.e. if the sequence cannot be compressed. The Linear Complexity test determines whether the length of the sequence's generator linear feedback shift register is too small. The Serial test judges the uniformity of the distribution of overlapping subsequences of a certain length, and returns two P-values based on different sequence indices. The Approximate Entropy test employs a different method to test the same aspect of non-randomness as the Serial test. The Cumulative Sums test detects whether there a certain value is over-represented at the extremities of the sequence. It returns a P-value for traversing through the sequence forward and for traversing backward. The Random Excursions test creates a random walk out of the sequence, and examines the frequency of occurrence for each of 8 states, returning a P-value for every state. The Random Excursions Variant test creates multiple random walks and measures the occurrence rate of each of 18 states, also returning a P-value for every state.

NIST has a website [28] where one may download ANSI C implementation of the test suite. In order to better understand the tests in the suite, we wrote a Matlab implementation of each test. Several challenges were encountered in this pursuit. The biggest challenge was encountered with the Linear Complexity test, which required coding a binary version of the Berlekamp-Massey algorithm [29]. This algorithm detects the smallest size LFSR able to generate the given sequence. Finding the minimal LFSR for a sequence requires on the order of n^2 bit operations [26], where n is the sequence length. The test required dividing the sequence of length at least $n = 10^6$ into N blocks of M bits each, where$n = MN$, $500 \leq M \leq 5000$ and $N \geq 200$. The Berlekamp-Massey algorithm would then be run on each block and a table of minimal LFSR would be constructed,

after which a chi-squared test would be conducted on the table. Ignoring any processing associated with the chi-squared test, this test requires quadratic complexity with a constant times $O(M^2) * N$ bit operations. In the best case, this corresponds to $O(500^2) * 2000$ bit operations. On a $3GHz$ 32-bit architecture CPU with the maximum $2GB$ of Random Access Memory allocated to Matlab, this test took an average of approximately 8 seconds to execute, compared with a fraction of a second required by each other test on average. Evaluating a RNG with a significance level of 0.01 requires generating 100 sequences and running every test on each sequence. Therefore, the additional delay incurred by the Linear Complexity test drastically increased the time of a large amount of simulations.

In order to test the correctness of the Matlab implementation, we subjected random and deterministic sequences to the newly implemented tests. For the random sequence we used data from the Random.org [30] TRNG, which is based on atmospheric noise. We requested data in blocks of 10^4 bits until accumulating enough for a sequence of length 10^6. For the deterministic data we used a LFSR of length 27 with gate connections corresponding to the polynomial $1 + x^{17} + x^{22} + x^{23} + x^{27}$ to generate a sequence of length $2^{27} - 1$. We used 10^8 bits of this data to form 100 sequences of length 10^6. The Matlab implementation passed the sequence harvested from Random.org, and it failed the sequences generated by the LFSR.

Chapter 3: Analysis

3.1 Opening Remarks

We consider the scenario depicted in Fig. 1, where two communicating parties (Alice and Bob) are establishing a key using reciprocal quantization of some channel parameter by alternating the roles of transmitter and receiver. The eavesdropper (Eve) performs a proximity attack in attempt to decipher the key by approaching Bob or Alice during key establishment. Other than approaching one of the communicating parties, the eavesdropper is passive. We consider the distance of the eavesdropper from the current receiver, who is attempting to establish a key.

We assume that some efficient method is used by both legitimate communicating parties to accurately estimate a channel parameter. Following the assumptions made in [12-21], we too assume that the channel is reciprocal for sufficient time such that the transmitter and receiver estimate the same value. The channel estimate is quantized with an arbitrary quantization depth to generate encryption key bits. The process is periodically repeated to generate the necessary amount of secret bits to form the encryption key. For the sake of analysis, we consider each bit of the quantization separately as if the key is generated by accumulating a single bit per quantized estimate.

We assume that the reciprocal key generating method being used is designed such that maximal key entropy is achieved, i.e., all possible keys are equally probable [27]. This means that the probability for any generated key bit to be zero or one is the same. This could translate to performing non-uniform quantization depending on the *Probability Density Function* (PDF) of the parameter being quantized. In addition, note that since the eavesdropper and receiver are in close proximity their fading channel statistics are expected to be the same. We regard the quantized

channel parameter estimate at the receiver as a binary vector of B secret key bits denoted by $\boldsymbol{k^r} = [k_1^r, k_2^r, \dots, k_B^r]$.

Since we require perfect secrecy during key establishment and key establishing rates which remain secure, we decouple analysis of proximity attacks and key establishing rates. In what follows, we assume a secure key establishing rate is used when performing analysis of proximity attacks, and that sufficient space separation between receiver and eavesdropper is in place when performing analysis of key establishing rates.

3.1.1 Proximity Attacks

In most reported work on symmetric key generation, it was assumed that the eavesdropper is sufficiently distanced from the intended receiver so that the channel from transmitter to receiver is independent of the channel from transmitter to eavesdropper [13-20]. Under this assumption, channel estimates at the receiver are unique and the eavesdropper is blocked access to them due to space selectivity of the wireless channel, resulting in independent channels and therefore perfect secrecy for key establishment. In a real world scenario, an eavesdropper can make an attempt to near the intended receiver and compromise the basic assumption of independent channel estimates. In other words, the eavesdropper can perform a *proximity attack* to reduce the space selectivity of the wireless channel. As a result the eavesdropper would be able to gain knowledge of the channel estimates at the receiver based on its own channel estimates and thereby deduce the key being established with some certainty. In an extreme scenario the eavesdropper could attach its antenna to that of the receiver so that they would experience the same channel with the transmitter. This implies that an effective proximity attack would hinder any practical method based on channel randomness. The question is: what is the minimal required distance of an intended receiver from a potential eavesdropper to securely establish the key? An analysis of security

strength in the face of proximity attacks is crucial for evaluating the efficacy of encryption methods based on channel randomness and for promoting their possible acceptance as alternatives to traditional methods.

There is limited reported work on the vulnerability of practical symmetric key generation methods using channel randomness in the presence of a proximity attack. The most relevant work to date was recently reported in [21], where a measurement campaign was conducted to evaluate the limits of key establishment based on reciprocal quantization of *Multiple-Input-Multiple-Output* (MIMO) channels in the presence of a passive eavesdropper. In [21] information theoretic analysis is used to find the percent of vulnerable secret bits out of the total number of generated bits as a function of the distance between eavesdropper and receiver. The difference in SNR of the channels to eavesdropper and receiver, the number of multipath components, presences of line of sight and number of antenna being used are considered as system parameters and affect the ratio of vulnerable secret bits.

In this contribution we present a generic approach to evaluate the effect of proximity attacks on any practical method which makes use of reciprocal quantization of channel parameters. Our generic approach evaluates the minimum required distance between receiver (either one on the communicating parties) and eavesdropper for such methods to remain secure, regardless of a possible SNR advantage of the eavesdropper and the number of antennas being used. The analysis results in a threshold on the required separation between eavesdropper and the communicating parties to achieve perfect secrecy for key establishment as a function of the Rician factor and quantization depth. Such absolute thresholds are useful for practical systems where the channel environment changes dynamically resulting in variable and unknown SNR advantage for the eavesdropper or when the number of antennas at the eavesdropper is unknown.

3.1.2 Key Establishing Rates

Key establishment rates received considerable attention in the past [6-21]. In general, the achievable key refreshing rates depend on channel decorrelation in time. If key refreshing rates are too fast, the channel doesn't decorrelate sufficiently to ensure that successive channel estimates and subsequent generated secret bits are uncorrelated. The strength of the key is diminished if successive secret bits are correlated. Past reported work on achievable key refreshing rates applied an information-theoretic approach based on the secrecy capacity. Using this approach, the achievable key rates largely depend on channel conditions. For example, in a single antennas system if the capacity of the channel from transmitter to eavesdropper is higher than that from transmitter to receiver, the secrecy capacity is zero and secure key establishment is not possible.
In this contribution we present a generic approach to evaluate achievable key establishing rates of practical methods making use of reciprocal quantization of channel parameters. We treat the sequence of generated secret bits as the output of a *Random Number Generator* (RNG). Assuming the eavesdropper is sufficiently far from the communicating parties to render a proximity attack ineffective, we are left with the need to validate the output of our channel-based RNG. To this end we use the NIST statistical test suite [23] in its entirety as was previously done for other novel RNGs.

3.2 Analysis of Key Establishing Rates in Rayleigh fading

We use Clarke's Rayleigh fading model, assuming the channel is narrowband with infinite scattering [24]. The received signal can be decomposed into in-phase and quadrature components, which are on different dimensions and are therefore independent.

$$r_I(t) = \sum_{n=1}^{N(t)} \alpha_n(t) \cos\big(\vartheta_n(t)\big) \tag{3}$$

$$r_Q(t) = \sum_{n=1}^{N(t)} -\alpha_n(t) \sin\big(\vartheta_n(t)\big) \tag{4}$$

$$r(t) = \alpha e^{j\vartheta(t)} = r_I(t) + j r_Q(t) \tag{5}$$

The autocorrelation function in time of the components is [24]

$$A_{Zc}(\tau) = A_{\mathrm{Zs}}(\tau) = P_r J_0(2\pi f_D \tau) \tag{6}$$

$$\delta_t = \tau f_D \tag{7}$$

where Z_c and Z_s respectively indicate the in-phase and quadrature components of the received signal, P_r denotes the received power, and J_0 indicates the zero-order Bessel Function of the first kind.

After sampling the components in (3) and (4) with period $\mathrm{T_s}$, the goal is to obtain the vector of channel parameter samples $\{k_i{}^R\}_{i=1}^N$. To this end, we define the following covariance matrix of the jointly normal elements in the quadrature component:

$$\mathrm{C} = \begin{bmatrix} \sigma_1^2 & \rho_{21} & \cdots & \rho_{N1} \\ \rho_{12} & \sigma_2^2 & \cdots & \rho_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{1N} & \rho_{2N} & \ldots & \sigma_N^2 \end{bmatrix} \tag{8}$$

$$\forall i \neq j, \qquad \sigma_i^2 = \sigma_j^2 = P_r and \rho_{ij} = \rho_{ji} = A_{rQ}(\tau = |i-j|\mathrm{T_s}) \tag{9}$$

Since both components are drawn from this distribution, we may use (8) to independently generate samples of Z_c and Z_s. We may then extract a channel parameter by applying a function to these components. In the case studied in this work, we would extract the phase and perform quantization. This would verify the results of using the Rician model for $K = 0$.

3.3 Analysis of Proximity Attacks in Rayleigh fading

In order to incorporate decorrelation across distance, we invoke a low-pass equivalent model for the correlation between two antennas in a diversity system [18]

$$\rho_{\mathrm{d}} = A_{Zc}(d) = A_{Zs}(d) = J_0\left(\frac{2\pi d}{\lambda}\right) \tag{10}$$

$$\delta_d = \frac{d}{\lambda} = \frac{dc}{f_c}. \tag{11}$$

This model assumes no correlation in time, so we set $T_s = 1sec$ to eliminate correlation of samples in time. This is justified for a case when the devices wait long enough for the channel to de-correlate before estimating the next key bit. This leads to

$$\mathrm{C} = \begin{bmatrix} \sigma_1^2 & P_r J_0(2\pi f_D \tau) & \dots & P_r J_0(2\pi f_D (N-1)\tau) \\ P_r J_0(2\pi f_D \tau) & \sigma_2^2 & \dots & P_r J_0(2\pi f_D (N-2)\tau) \\ \vdots & \vdots & \ddots & \vdots \\ P_r J_0(2\pi f_D (N-1)\tau) & P_r J_0(2\pi f_D (N-2)\tau) & \dots & \sigma_N^2 \end{bmatrix} \approx \mathrm{I_{NxN}} \tag{12}$$

assuming $P_r = \{\sigma_{\mathrm{i}}^2\}_{i=1}^{N} = 1$. We define the following vectors of component samples, in which samples of the received components and samples of the eavesdropper components are generated:

$$[\{Z_c\ {}^{r}_{i}\}_{i=1}^{N}\} \ \big|\ \{\mathrm{Z_c}\ {}^{e}_{i}\}_{i=1}^{N}\}\,] \tag{13}$$

$$[\{Z_s\ {}^{r}_{i}\}_{i=1}^{N}\} \ \big|\ \{\mathrm{Z_s}\ {}^{e}_{i}\}_{i=1}^{N}\}\,] \tag{14}$$

We form the new covariance matrix

$$C^{re} = \begin{bmatrix} I_{NxN} & \rho_d I_{NxN} \\ \rho_d I_{NxN} & I_{NxN} \end{bmatrix}. \tag{15}$$

Due to space-time independence, (15) generates random variables in the form of (13) and (14).

Once again, after obtaining $\mathrm{Z_c^r}, \mathrm{Z_s^r}, \mathrm{Z_c^e}, \mathrm{Z_s^e}$ we may apply a given function to $\mathrm{Z_c^r}, \mathrm{Z_s^r}$ and to $\mathrm{Z_c^e}, \mathrm{Z_s^e}$ to obtain a channel parameter. If the parameter is phase, we could compare to the results from the Rician case where $\mathrm{K} = 0$.

3.4 Analysis of Proximity Attacks in Rician fading

We use the time-based model given in [22] to describe the varying channel in space. This is justified due to the channel duality between space and time [23]. We use the following variable translation between space and time:

$$\frac{d}{\lambda} = f_D t \tag{16}$$

where λ is the wavelength associated with the frequency of operation,f_D is the maximal Doppler shift and $\omega_D = 2\pi f_D$. This equivalency is also evident in [24] for the Rayleigh fading scenario. Further discussion on space-time duality in wireless channels is given in [31].

Using the model in [22] and we form the space-based model:

$$Z_c(d) = \frac{\frac{1}{\sqrt{N}}\sum_{n=1}^{N}\cos\left(\frac{2\pi d}{\lambda}cos(\alpha_n)\right) + \phi_n) + \sqrt{K}\cos\left(\frac{2\pi d}{\lambda}\cos(\theta_0) + \phi_0\right)}{\sqrt{1+K}} \tag{17}$$

$$Z_s(d) = \frac{\frac{1}{\sqrt{N}}\sum_{n=1}^{N}\sin\left(\frac{2\pi d}{\lambda}cos(\alpha_n\right) + \phi_n) + \sqrt{K}\sin\left(\frac{2\pi d}{\lambda}\cos(\theta_0) + \phi_0\right)}{\sqrt{1+K}} \tag{18}$$

where $Z_c(d)$ and $Z_s(d)$ represent the in-phase and quadrature components respectively at the eavesdropper, d is distance in meters, K is the Rician Factor, N is the number of multipath components, θ_0 is the angle-of-arrival of the *Line of Sight* (LoS) component, ϕ_0 is the initial phase of the LoS component, $\{\phi_n\}$ are the initial phases of the scattered components, and$\{\alpha_n\}$ are the angles-of-arrival of the scattered components. Note that the model in (17) and (18) allows for evaluating the correlation between any two points in space. This is useful for modeling single as well as multiple antenna scenarios.

The quantized channel parameter estimate at the eavesdropper is denoted by the vector $\boldsymbol{k^e} = [k_1^e, k_2^e, \dots, k_B^e]$. If $\boldsymbol{k^e}$ and $\boldsymbol{k^r}$are independent the eavesdropper would not be able to deduce $\boldsymbol{k^r}$.

We define the following binary random variable:

$$\Delta e = k_i^r \oplus k_i^e \tag{19}$$

where $\oplus$ is the modulo 2 sum operation (exclusive or) and i is chosen out of $1, \ldots B$ to reflect a specific bit in the quantized binary vector.

3.4.1 Supporting Lemma for ensuring independent eavesdropper channels

Let X and Y be discrete binary random variables each uniformly distributed and let $Z = X \oplus Y$. $X$and Y are independent if and only if Z is uniformly distributed.

Proof:

Uniformity of X and Y means that their PDFs are given by

$$f_X(x) = \frac{1}{2}\delta_x + \frac{1}{2}\delta_{x-1}$$

$$f_Y(y) = \frac{1}{2}\delta_y + \frac{1}{2}\delta_{y-1} \tag{20}$$

It follows that

$$f_X(0) = f_X(1) = f_Y(0) = f_Y(1) = \frac{1}{2} \tag{21}$$

Z is 0 only if X and Y have the same value. Using the joint PDF of X and $Y f_{X,Y}(x, y)$ gives

$$f_Z(0) = f_{X,Y}(1,1) + f_{X,Y}(0,0) \tag{22}$$

$$f_Z(1) = f_{X,Y}(0,1) + f_{X,Y}(1,0) \tag{23}$$

Using marginalization and the discrete nature of X and Y to derive $f_X(x)$ and $f_Y(y)$ from $f_{X,Y}(x, y)$ we have

$$f_X(x) = \int_{-\infty}^{\infty} f_{X,Y}(x, y)\, dy = f_{X,Y}(x, 1) + f_{X,Y}(x, 0) \tag{24}$$

$$f_Y(y) = \int_{-\infty}^{\infty} f_{X,Y}(x, y)\, dx = f_{X,Y}(1, y) + f_{X,Y}(0, y) \tag{25}$$

$$\frac{1}{2} = f_X(1) = f_{X,Y}(1,1) + f_{X,Y}(1,0) \tag{26}$$

$$\frac{1}{2} = f_Y(1) = f_{X,Y}(1,1) + f_{X,Y}(0,1) \tag{27}$$

$$\frac{1}{2} = f_X(0) = f_{X,Y}(0,1) + f_{X,Y}(0,0) \tag{28}$$

$$\frac{1}{2} = f_Y(0) = f_{X,Y}(1,0) + f_{X,Y}(0,0) \tag{29}$$

Equating (26) with (27) and (28) with (29) respectively results in the following symmetries

$$f_{X,Y}(0,0) = f_{X,Y}(1,1) \tag{30}$$

$$f_{X,Y}(1,0) = f_{X,Y}(0,1) \tag{31}$$

Using (30) in (22) and (31) in (23) gives

$$f_Z(0) = 2f_{X,Y}(1,1) = 2f_{X,Y}(0,0) \tag{32}$$

$$f_Z(1) = 2f_{X,Y}(0,1) = 2f_{X,Y}(1,0) \tag{33}$$

Case I. Assuming uniformity of Z means that

$$f_Z(1) = f_Z(0) = \frac{1}{2} \tag{34}$$

Using (34) in (22) and (23) gives

$$f_{X,Y}(0,1) = f_{X,Y}(1,0) = f_{X,Y}(0,0) = f_{X,Y}(1,1) = \frac{1}{4} \tag{35}$$

It follows that $f_{X,Y}(x,y)$ is given by

$$f_{X,Y}(x,y) = \frac{1}{4}\delta_x\delta_y + \frac{1}{4}\delta_x\delta_{y-1} + \frac{1}{4}\delta_{x-1}\delta_y + \frac{1}{4}\delta_{x-1}\delta_{y-1}$$

$$= \left(\frac{1}{2}\delta_x + \frac{1}{2}\delta_{x-1}\right)\left(\frac{1}{2}\delta_y + \frac{1}{2}\delta_{y-1}\right) \tag{36}$$

Using (20) in (36) gives

$$f_{X,Y}(x,y) = f_X(x)f_Y(y) \tag{37}$$

so X and Y are independent.

Case II. Assuming independence between X and Y means that

$$f_{X,Y}(x,y) = f_X(x)f_Y(y) \tag{38}$$

Using (20) in (38) gives

$$f_{X,Y}(x,y) = \left(\frac{1}{2}\delta_x + \frac{1}{2}\delta_{x-1}\right)\left(\frac{1}{2}\delta_y + \frac{1}{2}\delta_{y-1}\right) \tag{39}$$

$$= \frac{1}{4}\delta_x\delta_y + \frac{1}{4}\delta_x\delta_{y-1} + \frac{1}{4}\delta_{x-1}\delta_y + \frac{1}{4}\delta_{x-1}\delta_{y-1}$$

which is equivalent to

$$f_{X,Y}(0,1) = f_{X,Y}(1,0) = f_{X,Y}(0,0) = f_{X,Y}(1,1) = \frac{1}{4}. \tag{40}$$

Using (40) in (22) and (23) results in

$$f_Z(0) = f_Z(0) = \frac{1}{2} \tag{41}$$

so Z is uniformly distributed.

■

The quantized bits are binary random variables, each uniformly distributed. It follows from *Lemma 1* that if Δe is uniform, k_i^r and k_i^e are independent and the eavesdropper can gain no knowledge on the established key bit by observing its own channel estimates.

In order to test uniformity of Δe, we invoke the NIST statistical test suite [23]. Using the channel model, we generate a bitstream of a single bit position of Δe for a given distance, and then apply the NIST frequency monobit test to the bitstream. The frequency monobit test assesses the uniformity of a binary random variable. If the proportion pass-rate exceeds the threshold determined by the sequence length, the bit position of Δe is considered to be uniformly distributed.

It follows that the eavesdropper's key observations are independent to those of the receiver and the eavesdropper can gain no knowledge of the generated key. This means that the space selectivity of the wireless channel determined by the distance between eavesdropper and receiver is sufficient to securely generate an encryption key by quantizing the channel estimates.

3.5 Analysis of Key Establishing Rates in Rician fading

Consecutive samples of a single bit from the quantized channel parameter comprise a random bit sequence which is the secret key. We apply the entire NIST test suite from Tab. 1 to the bit sequence per quantization bit as if it originated from a RNG.

In order to formulate a testing strategy, we observe the channel in-phase and quadrature autocorrelation functions in the time-based Rician fading channel model in [22]:

$$R_{Z_c Z_c}(\tau) = R_{Z_s Z_s}(\tau) = \frac{J_0(\omega_d \tau) + K cos(\omega_d \tau cos\ (\theta_0)}{2 + 2K} \tag{42}$$

Where J_0 is the first kind Bessel function of the zeroth order. We plot these functions as a function time normalized to the Doppler shift and $K \in [0, 1, 3,5,10]$ in Fig. 2.

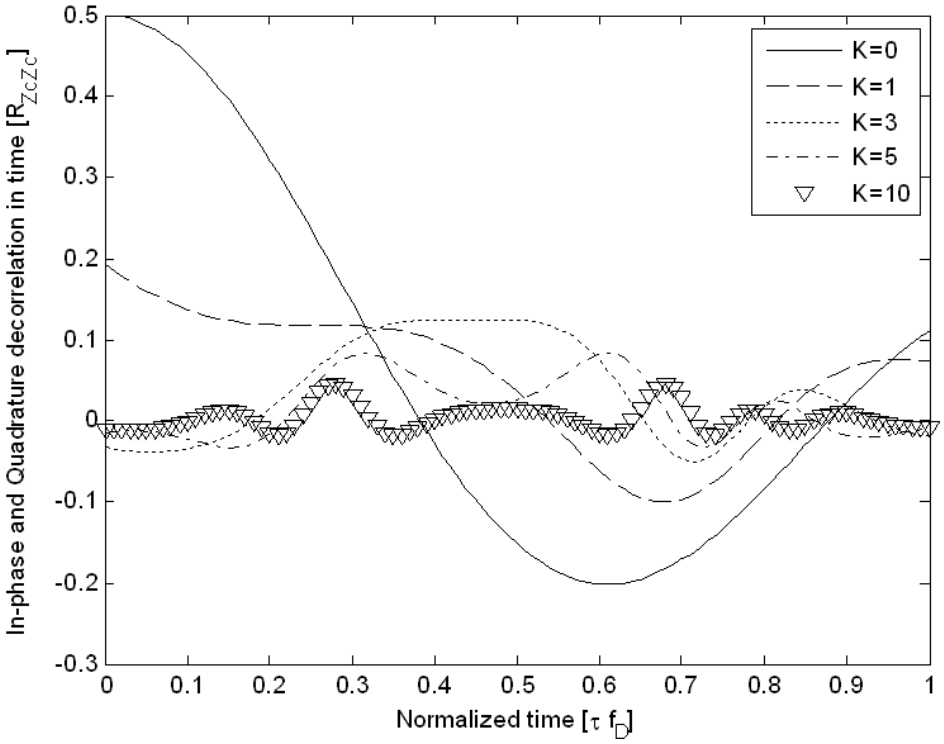

Fig. 2 –Rician channel correlation in time

The randomness of the phase for a particular sampling period is related to the component autocorrelation value at that time. We observe that sampling at a zero crossings in Fig. 2 would produce a channel estimate which it completely uncorrelated with the previous channel estimate. In an ideal world, we would sample at this zero-crossing and achieve an extremely high key refresh rate. However, sampling precisely at the zero-crossing would require impractical precision. For example, a Doppler shift of 100 Hz would produce a period in the phase decorrelation function of$10ms$.We assume the worst case of sampling on a peak or trough. Thus, for a particular Rician channel, we must extract and test the sequence of sampling periods corresponding to the extrema of the autocorrelation functions. For each sampling period a sequence of quantized channel estimates is generated using B bits per estimate. The quantized estimates are partitioned into separate sequences of random bits each corresponding to a specific bit in the

quantization $[k_1^r, k_2^r, \ldots, k_B^r]$. Each such sequence is evaluated using the entire NIST statistical test suite. The smallest extrema which passes all NIST tests is the smallest secure sampling period, since a small sampling offset would not increase the correlation across samples. The inverse of this sampling period is the maximum secret bit generating rate of a specific quantized bit position and is denoted $R_{b_{max}}$.

3.6 Carrier-phase Quantization

We now apply the two generic approaches to key establishing based on reciprocal quantization of the channel-phase. We assume that an accurate estimation method is used by both parties to accurately estimate the fading channel phase, using signals going back and forth in rapid succession [13-21]. The phase estimate is quantized to generate encryption key bits. The process is periodically repeated to generate the necessary amount of secret bits to form the encryption key.

Given a sampled channel phase $-\pi < \theta_R(nT) \leq \pi$, we shift and scale to

$$\theta_R'[n] = \frac{\theta_R(nT) + \pi}{2\pi} \tag{43}$$

and uniformly quantize these phases into B bits per phase,

$$\theta_R^Q = \frac{\lfloor \theta_R'[n] * 2^B \rfloor}{2^B}. \tag{44}$$

3.6.1 Proximity Attacks using Carrier-Phase Quantization

The phase at the eavesdropper and receiver is given respectively by

$$\theta_e = \tan^{-1}\left(\frac{Z_s(d)}{Z_c(d)}\right) \tag{45}$$

and

$$\theta_r = \tan^{-1}\left(\frac{Z_s(d_0)}{Z_c(d_0)}\right) \quad (46).$$

In order to generate the phase of a Rician fading channel, we first generate the received in-phase and quadrature components. Loosely stated, if the sign of Z_s and Z_c are considered, full phase mapping is obtained and $\theta_e, \theta_r \in [-\pi, \pi)$. The phases are uniformly quantized to obtain $\boldsymbol{k^e}$ and $\boldsymbol{k^r}$, where $B = 6$ bits.

Without loss of generality we assume the eavesdropper and receiver are at a distance of d and d_0 respectively from some reference point placed on a straight line going through receiver and eavesdropper positions, and that the receiver is at the reference position ($d_0 = 0$). For distances d and d_0, we used $N = 8$ multipath components, which was shown in [22] to be a sufficient number of components to model the channel. The frequency monobit test requires a bitstream length of at least 100, and a significance level of $\alpha = 0.01$ requires $\frac{1}{\alpha} = 100$ bitstreams. We generated 10^5 phases, which we then quantized to $B = 6$ bits. We formed Δe and generated 1000 bitstreams of sequence length 100 for each of the 6 bit-positions, which were then input into the NIST frequency monobit test.

For generality we normalize the distance d by the carrier wavelength λ. We considered a normalized distance of $0 < \frac{d}{\lambda} \leq 1$, assuming the eavesdropper is always able to be within a wavelength of the receiver. We found the largest distance in this range for which the NIST monobit test failed. The distance up to the failing distance is the minimal required distance to securely generate the key and is noted d_{min}. If a distance of $d = \lambda$ failed the NIST test, we declare key generation as a failure.

The aforementioned strategy was executed on each of the 6 quantized bit-positions with $K \in [0,10]$. Fig. 3 shows the results. For brevity we omit failed attempts ($d_{min} > \lambda$) from the

graph. It is apparent that as K increases d_{min} increases as well. This is because a higher K results in less multipath and hence less randomness of the channel. We observe that deeper quantization bits help increase d_{min}. This is because deeper quantization bits are sensitive to smaller channel variations across space. As long as the quantization noise is tolerable, the loss of channel randomness due to high K can be compensated by using a deeper quantization bit. Note the discrete levels of d_{min} for varying K.This is a manifestation of the hard-decision threshold output of the NIST frequency monobit test and is useful for determining clear requirements for d_{min} as a function of K.

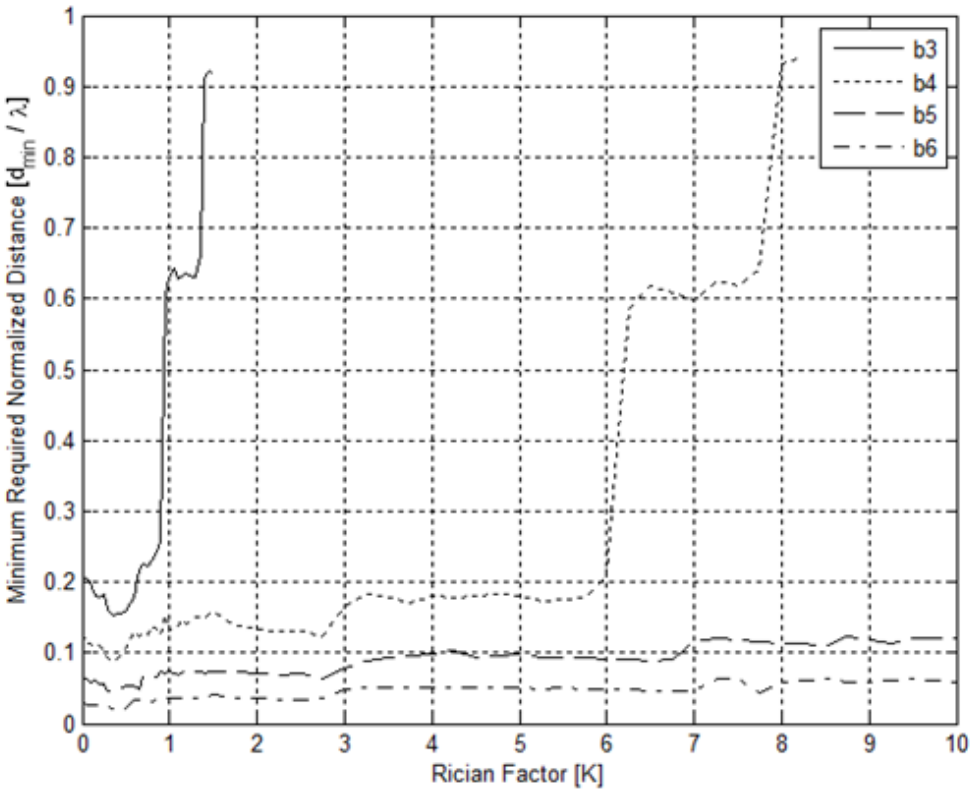

Fig. 3– Minimum required distance as function of Rician factor for various quantization bits

The results in Fig. 3 help determine how far a receiver must be from the eavesdropper to foil a proximity attack in practical systems. For example, transmission in the ISM bands $2.45GHz$,

$915MHz$ and $434MHz$ correspond to a wavelength of $12.2cm$, $32.7cm$ and $69.1cm$ respectively. The third *Most Significant Bit* (MSB#3) of the phase quantization can be used for$K < 1$if the receiver is at least $2.5cm$, $6.6cm$ and $13.8cm$ away from the eavesdropper for $2.45GHz$, $915MHz$ and $434MHz$ respectively.If MSB#4 is used the same distances ensure security for $K < 6$. If MSB#4is used in a$2.45GHz$ IEEE 802.15.4 system and the channel is known to be Rician fading with $K \leq 8$ a distance of at least $7.5cm$ between receiver and eavesdropper is required. These distances seem reasonable for many practical systems. For quantization depth higher than five bits the required distance is below $\lambda/10$, which corresponds to a minimal distance of $1.2cm$, $3.3cm$ and $6.9cm$ for $2.45GHz$, $915MHz$ and $434MHz$ respectively.

3.6.2 Key Establishing Rates using Carrier-Phase Quantization

The channel-phase using the time model in [22] is given by

$$\theta_R(nT) = \tan^{-1}\left(\frac{Z_s(nT)}{Z_c(nT)}\right);\ n = 1,2,\dots,z \tag{47}$$

We observe that (47) generates a sequence of consecutive phase of length z. We generate m total number of sequences of length z. We scale and quantize these phases according to (45) and (46). After quantizing, we have a matrix of bits of size m by z by B. We select a bit position $b \leq B$ and reshape the data into m bitstreams of length z.

We ran Monte Carlo analysis over a sweep of phase sampling period T_s. We took a quantization depth of $B = 8$ bits since that is a conservative estimate of common *Analog to Digital Converter* (ADC) depths. We set the number of multipaths equal to $N = 8$ as was done previously in [22]. We set the bit positions to $b \in [3,4,5,6,7,8]$ and the Rician factors to $K \in [0,1,3,5,10]$. We then applied the NIST test suite with sequence length $z = 10^6$ so that we could execute all the tests. We

used a significance level $\alpha = 0.01$, requiring $m = \frac{1}{\alpha} = 100$ sequences. Tab. 2 shows the parameters used for the tests.

Test	Parameter	Value
Block Frequency	block size	100000
	# blocks	10
Longest Run of Ones	block size	10000
	# blocks	75
Binary Matrix Rank	# matrix rows	32
	# matrix cols	32
Non-overlapping Template Matching	# blocks	8
	block size	125000
	template size	9
	Template	000000001
Overlapping Template Matching	template size	9
	Template	000000001
Maurer's "Universal Statistical"	block length	7
	# blocks	1280
Linear Complexity	block length	1000
	degrees of freedom	7
Serial	block length	3
Approximate Entropy	block length	2
Random Excursions	States	{-4..-1}{1..4}
Random Excursions Variant	States	{-9..-1}{1..9}

Tab. 2 – Parameters for NIST tests

We determined $R_{b_{max}} = 1/T_s$, which simultaneously meets the randomness threshold for every test, across the aforementioned space of (K, b). For generality, time is normalized by the Doppler shift. Fig. 4 shows the results.

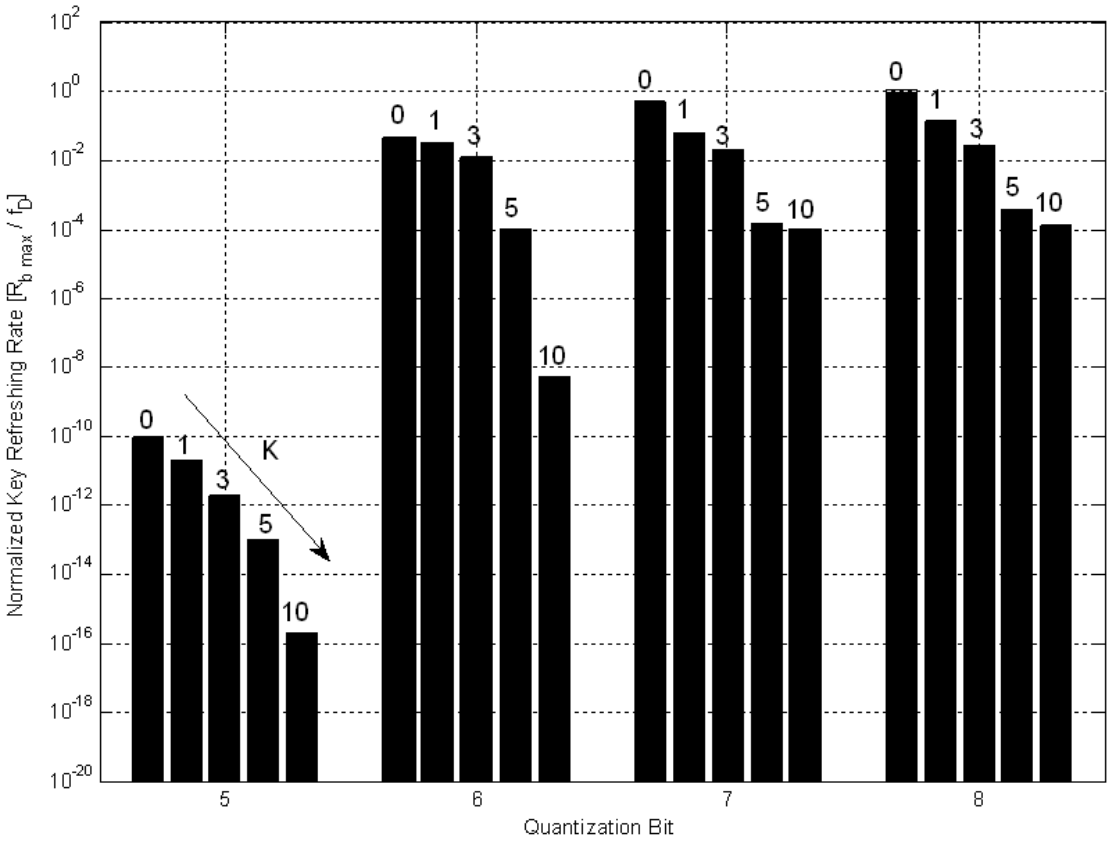

Fig. 4 – Maximum key refreshing rates as a function of quantized bit position for K = [0,1,3,5,10].

We note that $R_{b_{max}}$ varies inversely with K, since a higher K increases the ratio between LoS and scattered power resulting in reduced randomness. We also observe that $R_{b_{max}}$increases with a higher b, since a deeper quantization bit is more sensitive to small variation of the channel over time.

The results in Fig. 4 are useful for determining achievable key establishing rates in practical systems. For example, consider a stationary scenario with no LoS ($K = 0$), where changing environment corresponds to a low Doppler shift of $f_D = 5Hz$. In such a scenario, one may attain the following key refresh rates:$4\ x\ 10^{-2}\frac{bits}{sec} *\ 5 = 0.2\frac{bits}{sec}$ using $MSB\ \#6$ and $5\ x\ 10^{-1}\frac{bits}{sec} * 5 = 2.5\frac{bits}{sec}$ using $MSB\ \#7$. This means that it would take $320sec$ to establish a 64 bit key

usingonly $MSB\ \#6$, and $25.6sec$ to establish the same key using only $MSB\ \#7$.As another example, consider a mobile vehicular environment corresponding to $f_D = 100Hz$ with a LoS component corresponding to $K = 10$. In such a scenario, using only $MSB\ \#7$ to establish a 128 bit key would require $10^{-4} * 100 * 128 = 1.28sec$.

Chapter 4: Conclusion

4.1 Closing Remarks

Symmetric key establishment using reciprocal quantization of channel parameters in wireless Rician fading channels was considered. Two aspects were addressed through generic analysis: impact of a proximity attack by a passive yet mobile eavesdropper with possible SNR advantage and achievable key establishing rates. Analysis made rigorous use of the NIST statistical test suite applied to the channel quantization bits as the outputs of a random number generator. The analysis was applied to channel-phase quantization and performance in practical systems was considered as a special case.

For proximity attacks, the NIST frequency monobit test was used in conjuncture with a lemma that was defined and proved. The minimal required distance from the eavesdropper in order to maintain perfect secrecy during key establishment was evaluated as a function of the Rician factor and quantization depth. The analysis proved useful for determining the required distance from the eavesdropper to securely establish the key. For example, in the ISM bands $2.45GHz, 915MHz$ and $434MHz$perfect secrecy is achieved for environments with a Rician factor of $K \leq 8$ by using $MSB\ \#5$ with a minimum receiver-eavesdropper distance of $6.9cm, 3.3cm,$ and $1.2cm$ respectively.

For key establishing rates, we assumed that a proximity attack is not possible, i.e., the eavesdropper is sufficiently far from the legitimate parties. The maximal achievable key establishment rates were evaluated by treating a given quantization bit of the channel phase as a cryptographic RNG and applying the complete NIST statistical test suite to its output bitstream. The analysis proved useful for evaluating achievable key refreshing rates in practical scenarios.

For example, when using MSB #7 in a Doppler shift of 5 Hz and no LoS between transmitter and receiver, a 64 bit key can be established in $25.6s$. Alternatively, in a vehicular scenario where the Doppler shift is $100Hz$ and the Rician factor is 10, a 128 bit key is established in $1.28sec$.

4.2 Improvements and Future Work

The entropy inherent in a wireless channel is present in all the channel parameters. Therefore, the channel phase is only one possible keying variable. The case of using the phase was particularly convenient since its uniform distribution allowed uniform quantization. Any function on the channel parameters should be considered as a key generator. For example, the channel amplitude of the Rician channel may be used. This amplitude has Rice distribution

$$f_X(x) = \frac{2(K+1)x}{P_D} e^{-K-\frac{(K+1)x^2}{P_D}} J_0\left(2x\sqrt{\frac{K(K+1)}{P_D}}\right) \tag{48}$$

where P_D represents the LoS power, K is the ratio of LoS to scattered power, and J_0 is the zero order Bessel function of the first kind.

If using a quantization of this amplitude as a key generator, one would need to adjust the sampling such that the regions in a sampled Rice amplitude distribution would have equal area. In order to determine where to sample, we must solve this equation for $\{x_i\}_{i=1}^{res}$

$$\int_0^{x_1} f_X(x)dx = \int_{x_1}^{x_2} f_X(x)dx = \cdots = \int_{x_{res-1}}^{x_{res}} f_X(x)dx \tag{49}$$

where res represents the degree of granularity of the sampling and $\{x_i\}_{i=1}^{res}$ represent the sampling indices.

The results in this work have been generated with practical intent. It is our hope for the system analyst to use these results as a guideline for preventing proximity attacks while using the channel phase to generate keys for a symmetric cipher. Even if the channel has properties outside the range of those tested here, one may still use the trends we have outlined in Fig. 3 and Fig. 4. We have explained the general trends encountered when varying the environment, quantization bit, and frequency.

Many improvements could be made to the work here, especially for those with theoretical interest. One could perform additional simulations for more ISM frequencies, a deeper level of quantization bits, and a wider and higher resolution sweep of Rician K values. Future work could also be in the form of gathering more accurate channel statistics through a real world measurement campaign or through instrumentation which simulates a wireless channel.

References

[1] J. Daemen, V. Rijmen, The Design of Rijndael: AES - The Advanced Encryption Standard, Springer-Verlag, 2002.

[2] R. Rivest, A. Shamir, L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, vol. 21, pp. 120–126, 1978.

[3] M. Robshaw, O. Billet (Eds.), New Stream Cipher Designs – the eSTREAM Finalists, Springer, 2008.

[4] W. Diffie, M. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, vol.22, no.6, pp. 644- 654, Nov 1976.

[5] N. R. Potlapally, S. Ravi, A. Raghunath and N. K. Jha, "A Study of the Energy Consumption Characteristics of Cryptographic Algorithms and Security Protocols", IEEE Transactions on Mobile Computing, vol. 5, no. 2, pp. 128-143, Feb. 2006.

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

[7] Y. Liang, H. V. Poor, and S. Shamai, "Secrecy capacity region of parallel broadcast channels," in Proceedings of the IEEE Information Theory and Applications Workshop (ITA '07), pp. 245–250, San Diego, Calif, USA, February 2007.

[8] M. Debbah and M. Kobayashi, "On the secrecy capacity of frequency-selective fading channels: a practical vandermonde approach," in Proceedings of IEEE Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '08), pp. 1–5, Cannes, France, September 2008.

[9] P. K. Gopala, L. Lai and H. El Gamal, "On the Secrecy Capacity of Fading Channels", IEEE Transactions on Information Theory, vol. 54, issue 10, pp. 4687-4698, Oct. 2008.

[10] M. Bloch, J. Barros, M. R. D. Rodrigues, S. W. McLaughlin, "Wireless Information-Theoretic Security", IEEE Transactions on Information Theory, vol. 54, no. 6, pp. 2515-2534, June 2008.

[11] L. Dong, Z. Han, A. P. Petropulu, H. V. Poor, "Improving Wireless Physical Layer Security via Cooperating Relays", IEEE Transactions on Signal Processing, vol. 58, no. 3, pp. 1875-1888, March 2010.

[12] M. Kobayashi, M. Debbah, and S. Shamai, "Secured Communication over Frequency-Selective Fading Channels: A Practical VandermondePrecoding", Eurasip Journal on Wireless Communications & Networking – Special Issue on Physical Layer Security, 2009.

[13] G. R. Tsouri and D. Wulich, "Securing OFDM over Wireless Time-Varying Channels using Sub-Carrier Over-Loading with Joint Signal Constellations", Eurasip Journal on Wireless Communications & Networking – Special Issue on Physical Layer Security, 2009.

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

[15] H. Koorapaty, A. Hassan, and S. Chennakeshu, "Secure Information Transmission for Mobile Radio", IEEE Communication Letters, vol. 4, issue 2, pp. 52-55, Feb. 2000.

[16] Y. Liang, H. V. Poor and S. Shamai (Shitz), "Secure communication over fading channels," IEEE Transactions on Information Theory – Special Issue on Information Theoretic Security, 54(6), pp. 2470-2492, June 2008.

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

[18] G. R. Tsouri, "Securing Wireless Communication with Implanted Medical Devices", IEEE International WoWMoM Workshop on Interdisciplinary Research on E-Health Services and Systems, May 2009.

[19] C. Ye, S. Mathur, A. Reznik, Y. Shah, W. Trappe, and N. B. Mandayam, "Information-theoretically secret key generation for fading wireless channels," IEEE Transactions on Information Forensics and Security, vol. 5, no. 3, pp. 240–254, Jun. 2010.

[20] G. R. Tsouri and A. Sapio, "Method of Securing Resource-Constrained Wireless Enabled Devices via Channel Randomness", IEEE 28th International Conference on Consumer Electronics (ICCE), Jan. 2010.

[21] J. W. Wallace, R. K. Sharma, "Automatic Secret Keys from Reciprocal MIMO Wireless Channels: Measurements and Analysis", IEEE Transactions on Information Forensics and Security, Sept. 2010.

[22] C. Xiao, Y.R. Zheng, N.C. Beaulieu, "Novel Sum-of-Sinusoids Simulation Models for Rayleigh and Rician Fading Channels," IEEE Transactions on Wireless Communications, vol. 5, pp. 3667-3679, Dec. 2006.

[23] A. Rukhin, J. Soto, J, Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, S. Vo, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," NIST, August 2008.

[24] A. Goldsmith, Wireless Communications, University Press, 2005.

[25] C. E. Shannon, "Communication theory of secrecy systems," Bell Systems Technical Journal, vol. 28, pp. 656–715, 1949.

[26] Donald Knuth. The Art of Computer Programming, Volume 2: Seminumerical Algorithms, Third Edition. Addison-Wesley, 1997. ISBN 0-201-89684-2. Chapter 3, pp. 1–193. Extensive coverage of statistical tests for non-randomness.

[27] M. Matsumoto and T. Nishimura, "Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator", ACM Trans. on Modeling and Computer Simulation Vol. 8, No. 1, January pp.3-30 (1998).

[28] NIST.gov - Computer Security Resource Center <http://csrc.nist.gov/groups/ST/toolkit/rng/index.html>.

[29] Information and Communication <http://www.informationsuebertragung.ch/indexAlgorithmen.html>.

[30] RANDOM.org –Integer Generator. <http://www.random.org/integers/>.

[31] G. D. Durgin, Space-Time Wireless Channels, Prentice Hall, 2002.