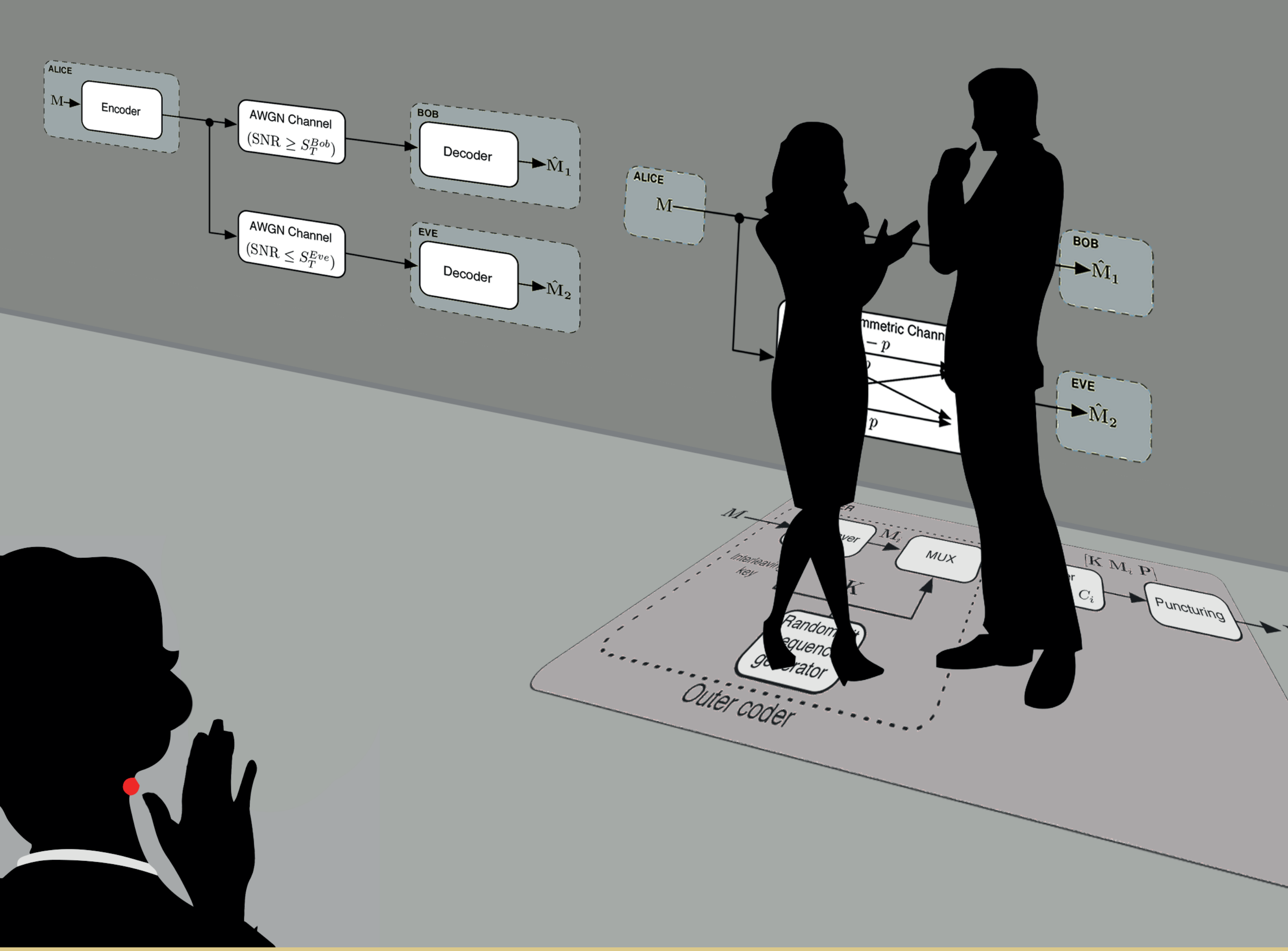

Dinis Cambraia Lopes Sarmento Pereira

Design of Finite Length
Interleaved Coding Schemes for Secrecy

Dissertação submetida para a satisfação parcial dos requisitos do grau de
Mestre em Engenharia Electrotécnica e de Computadores, Área de Especialização em Telecomunicações

Julho de 2015

· U C ·

UNIVERSIDADE DE COIMBRA

Design of Finite Length Interleaved Coding Schemes for Secrecy

Dinis Cambraia Lopes Sarmento Pereira

Dissertação para obtenção do Grau de Mestre em
Engenharia Electrotécnica e de Computadores

Orientador:	Doutor Marco Alexandre Cravo Gomes
Co-Orientador:	Doutor João Paulo da Silva Machado Garcia Vilela

Júri

Presidente:	Doutor Mário Gonçalo Mestre Veríssimo Silveirinha
Orientador:	Doutor Marco Alexandre Cravo Gomes
Vogal:	Doutor Vítor Manuel Mendes da Silva

Julho de 2015

Agradecimentos

Gostaria de começar por agradecer ao professor Marco Gomes e ao professor João Vilela, pela oportunidade de participar neste projeto. A orientação dada por ambos foi incansável e o conhecimento por eles partilhado fez com que esta experiência se tornasse bastante enriquecedora a nível pessoal.

Agradeço também a Willie Harrison pela disponibilidade que demonstrou para ajudar, desde o primeiro dia.

Ao Instituto de Telecomunicações por todos os meios disponibilizados e por ter levado a que partilhasse um ambiente de trabalho descontraído, com um grupo de colegas sempre dispostos a ajudar.

Aos meus pais e irmãos por sempre acreditarem em mim e pelo privilégio de os ter como exemplo a seguir.

A ti Adriana, por fazeres com que cada dia valha a pena.

A todos os amigos e familiares que me acompanharam e apoiaram ao longo desta importante etapa,

Muito Obrigado.

Abstract

With the increasing use of wireless communications, came a growing concern for obtaining confidentiality in the transmission of data. Recently, methods for applying secrecy at the physical layer have been in the focus of attention from the scientific community. In this dissertation we provide secrecy metrics applicable to physical-layer coding techniques with finite blocklengths. Our metrics go beyond some of the known practical secrecy measures, so as to make lower bound probabilistic guarantees on error rates over short blocklengths. These techniques are especially useful in cases where application of traditional information-theoretic security measures is either impractical or simply not yet understood. We also propose two coding for secrecy schemes based on the combination of interleaving with systematic channel codes. The basic idea consists of generating a random interleaving key that is used to shuffle/interleave information at the source. The message and the interleaving key are then both encoded with a systematic code. On a first approach a jamming signal is generated with the intent to degrade the eavesdropper's channel during the transmission of the key bits. Then we consider a different setup where the part related to the interleaving key is removed/punctured before being sent to the channel, hence operating as a hidden key for any receiver (legitimate or not) that needs to deinterleave the message. Successfully obtaining the original message then depends on determining the interleaving key, which can only be done through the parity bits that result from jointly encoding the interleaving key and the message. Leveraging on the proposed security metrics, we provide a method to determine the necessary signal-to-noise ratio difference that enables successful reception at the legitimate receiver without the eavesdropper having access to the message. In addition, we provide evidence that this scheme may also be used to turn a realistic channel into a discrete memoryless channel that can be employed with a wiretap code to provide information-theoretic security guarantees.

Keywords

Security, Reliability, Short Blocklengths, Concatenated Schemes, Interleaving, Practical Security Metrics

Resumo

O forte crescimento observado na última década, na presença das comunicações sem fios no dia-a-dia, trouxe consigo a necessidade de assegurar a confidencialidade de dados transmitidos. Neste contexto, no seio da comunidade científica têm vindo, recentemente, a serem propostos métodos que visam implementar segurança na camada física. Nesta dissertação propomos métricas de segurança para avaliação/desenho de esquemas de codificação na camada física, fazendo uso blocos de comprimento curto. As nossas métricas são relevantes face a algumas das métricas de segurança práticas, visto que avaliam a probabilidade de garantir taxas elevadas de erros em cada bloco. Estas métricas são especialmente úteis para aplicação em esquemas onde seja impraticável efetuar uma análise com base nas métricas baseadas em teoria da informação. Propomos também dois esquemas de codificação com vista a obter segurança na camada física. A ideia base destes esquemas consiste em gerar aleatoriamente uma chave de *interleaving* que é utilizada para baralhar a mensagem. Após este baralhamento a mensagem é concatenada com a chave e a palavra resultante é codificada com um código sistemático. Numa primeira abordagem é gerado um sinal de interferência que visa degradar o canal de transmissão de um eventual recetor ilegítimo, durante a transmissão dos *bits* correspondente à chave. Numa segunda abordagem, a secção da palavra de código referente à chave não é transmitida, sendo a chave escondida para qualquer utilizador (ilegítimo ou não). De modo a desembaralhar corretamente a mensagem será necessário obter uma estimativa correta da chave através dos bits de paridade da palavra de código. Com base nas métricas de segurança definidas, a metodologia proposta permite determinar a diferença de relações sinal ruído necessária para que o receptor legítimo obtenha uma comunicação fiável e o receptor indesejado não seja capaz de obter uma estimativa correta da mensagem. Por fim, é feita uma análise e são tecidas considerações que visam mostrar que este esquema pode ser utilizado para emular um canal discreto sem memória onde possa ser aplicado um código *wiretap* que proporcione garantias de segurança baseadas em teoria de informação.

Palavras-Chave

Confidencialidade, Fiabilidade, Blocos de Comprimento Curto, Esquemas Concatenados, *Interleaving*, Métricas Práticas de Confidencialidade

Contents

List of Figures

List of Acronyms

ARQ	automatic repeat request
AWGN	additive white gaussian noise
BCH	bose-chaudhuri-hocquenghem
BEC	binary erasure channel
BER	bit error rate
BPSK	binary phase-shift keying
BSC	binary symmetric channel
CDF	cumulative distribution function
DMC	discrete memoryless channel
ECC	error correcting code
FEC	forward error correction
PMF	probability mass function
LDPC	low-density parity-check
LSPA	logarithmic sum product algorithm
SNR	signal-to-noise ratio

1

Introduction

The steady growth on the usage of wireless communications, propelled by the impact caused by the Internet, raises the necessity of having methods that grant confidentiality during the transmission of data. To face this problem and obtain security on communications, cryptographic techniques are applied on the upper protocol layers [1].

Physical-layer security, which lately has been the focus of much attention, follows a different approach by aiming to obtain secrecy at the physical layer, through the exploitation of the physical characteristics and imperfections of communication channels [1]. The objective of physical-layer security is not the substitution of cryptography protocols, but rather to act as a complement, by adding an extra layer of security for communications.

This idea is not new and dates back to 1975, introduced by A. D. Wyner in [2], where a coding method and a metric for evaluating secrecy known as *weak secrecy* were proposed. Wyner's propositions are based on information theory, and consider a system setup which is referred as *wiretap channel*. On this setup, an eavesdropper is passively listening to the communication between two legitimate users, through a degraded channel.

Developing codes (known as wiretap codes), for the coding method from [2], is not a trivial problem. It took more than 30 years for the emergence of the first code construction that achieves weak secrecy [3], since then, further progress has been made [4] [5] [6]. However, such designs are based on simplified channel models (e.g. binary erasure channel (BEC) or binary symmetric channel (BSC) models) and often assume that the legitimate users share a perfect channel, which means that the problem of assuring a reliable communication is neglected. These assumptions are far from reality. Approaches for attaining secrecy on more realistic channel models have been made and include puncturing for secrecy [7] [8] or the use of scrambling techniques over blocks of concatenated frames [9].

Information-theoretic metrics have been the chosen medium for evaluating physical-layer security coding schemes [10]. However, analyzing realistic channel models with such metrics is impractical for most cases. On the other hand, evaluating the bit error rate (BER) can be done by simulation for various channel models, but it's not regarded as reliable in terms of security. These aspects motivate the need of new ways to evaluate secrecy, when the objective is the construction of schemes for real world scenario uses, that often require short blocklengths.

1.1 Objectives and Main Contributions

The investigation developed on the course of this dissertation took on physical-layer security from a functional perspective. The main objective of this thesis is the proposal of secrecy schemes for application to real world scenarios, that consider both the reliability and secrecy aspects of a communication. Our approach is based on using concatenated coding schemes, where an outer coder (that may consist of any number of coding operations) is used to provide secrecy, and an inner coder to account for reliability.

The impracticability of applying information-theoretic secrecy metrics to realistic channel models and the shortcomings of the BER as a way to evaluate security, led us to the development of secrecy metrics that are based on the distribution of the number of errors per block. Contrary to other secrecy metrics, these enable the evaluation of coding schemes on the short blocklength regime, and can be used to identify operable regions of signal-to-noise ratio (SNR) for which bit-error rates, even over a short number of bits, are guaranteed to be near 0.5. Using these new metrics, we also aim to design coding schemes that can produce an effective wiretap channel over which the code constructions of wiretap codes can be applied to.

Summing up, the main contributions of this thesis are:

1. J. P. Vilela, M. Gomes, W. Harrison, D. Sarmento, F. Dias, “Interleaved Concatenated Coding for Secrecy in the Finite Blocklength Regime”, IEEE Signal Processing Letters, submitted for publication, March 2015.

2. W. Harrison, D. Sarmento, J. P. Vilela, M. Gomes, “Analysis of Short Blocklength Codes for Secrecy”, IEEE Transactions on Information Forensics and Security, submitted for publication, June 2015.

3. D. Sarmento, J. P. Vilela, W. Harrison, M. Gomes, “Interleaved Coding for Secrecy with a Hidden Key”, IEEE GLOBECOM 2015 - Workshop on Trusted Communications with Physical Layer Security, submitted for publication, July 2015.

1.2 Dissertation Outline

This thesis is composed of six chapters. Following the Introduction, Chapter 2 introduces concepts on information theory and conditions for reliable and secure transmissions, that are necessary for the comprehension of the succeeding chapters. Chapter 3 details the current metrics for evaluating secrecy as well as their limitations, and contains the description of the new proposed metrics. On chapter 4 a concatenated coding for secrecy scheme that uses interleaving to hide information, and where security is obtained through the use of jamming is described. The analysis of its performance is executed using the proposed metrics from chapter 3. Chapter 5 introduces a secrecy scheme that shares similarities with the scheme from chapter 4, but where a hidden key is punctured before transmission, and shows that such scheme may be used to emulate the necessary conditions for applying code constructions made for the wiretap channel, on real world scenarios. Finally, Chapter 6 presents the main conclusions drawn from this thesis and presents some suggestions for future work.

2

Preliminary Concepts

In this chapter we will overview some theoretical concepts that will be important for the comprehension of the work developed during the course of this dissertation. We'll start by introducing some notions on information theory, which will be relevant for the understanding of the secrecy metrics presented on section 3.1.

The central focus of this dissertation is the evaluation/proposition of coding schemes that achieve reliability for the legitimate user and secrecy against unintended eavesdroppers. Therefore, on section 2.2, some aspects on how reliability can be achieved in digital communications will be addressed and later on section 2.3, some concepts of physical layer security will be introduced.

2.1 Information Theory Concepts

The scientific area of information theory was greatly developed by Claude E. Shannon [11], with the publication of the paper *The Mathematical Theory of Communication*, in 1948.

Shannon introduced the notion of *entropy*, which plays a central role in information theory as a measure of information, choice and uncertainty [11]. The entropy $H(X)$, of a discrete random variable[1] X with probability mass function (PMF) $p(x)$, is defined by[2]:

$$H(X) = -\sum_{x} p(x) \log p(x). \tag{2.1}$$

Entropy corresponds to the average number of *bits* (binary digits) required to describe a random variable [12]. Its value is maximum when $p(x)$ is uniform. In that situation $H(X) = \log q$, where q is equal to the number of possible values X can take.

As in probabilities, where there are joint probabilities and conditional probabilities, we can also define similar types of entropies. The *joint entropy* $H(X;Y)$, of discrete random variables X and Y with PMFs $p(x)$ and $p(y)$, respectively, and joint PMF p(x,y) is defined by:

$$H(X;Y) = -\sum_{x}\sum_{y} p(x,y) \log p(x,y). \tag{2.2}$$

This definition can be extended to any number of random variables. The *conditional entropy* $H(X \mid Y)$, also known as *equivocation*, can be defined as the uncertainty of X conditional to the knowledge of Y, and is given by:

$$H(X \mid Y) = -\sum_{x}\sum_{y} p(x \mid y) \log p(x \mid y). \tag{2.3}$$

[1] In the context of information theory, this random variable refers to probabilities that underlie in the process of communication or data compression.

[2] For the formulas on this section, all the logarithms are to base 2.

One important concept is the one of *mutual information*. Formerly known as *transinformation*, mutual information $I(X;Y)$, refers to the reduction in uncertainty of X, due to the knowledge of Y, and is defined by:

$$I(X;Y) = H(X) - H(X \mid Y). \tag{2.4}$$

or, alternatively:

$$I(X;Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}. \tag{2.5}$$

Its value can never be negative and is equal to zero if and only if X and Y are independent.

Figure 2.1 illustrates de relations between the entropies $H(X)$ and $H(Y)$, joint entropy $H(X;Y)$, conditional entropies $H(X \mid Y)$ and $H(Y \mid X)$, and mutual information $I(X;Y)$.

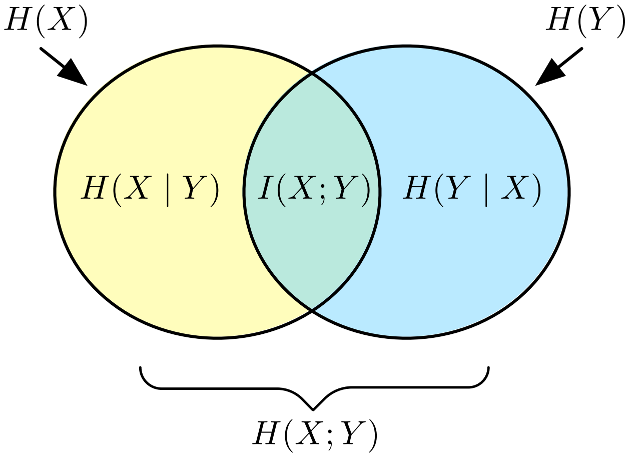

Figure 2.1: Venn diagram with the relations between the mutual information and the different types of entropies.

2.2 Error Control Coding

Most of the time, communication is done over noisy channels. This brings the necessity of having methods for detection and correction of transmission errors. In this section we are going to do a brief introduction on important aspects of error control coding.

Figure 2.2 illustrates a typical digital communication system. The channel code is designed to perform error detection and/or correction, aiming to convert a noisy channel into a reliable one. There are two types of systems involving this type of encoder:

- *Automatic repeat request* (ARQ): in this type of system, codes are designed to detect the occurrence of transmission errors, and in that event, the receiver requests the retransmission of the information received with error. It is required that the source and the destination can communicate at the same time, and in a two way mode [13].

- *Forward error correction* (FEC): in FEC systems, transmission errors are detected and corrected at the receiver.

We'll focus our attention on FEC systems. There are two main categories of codes that can be used for this type of systems: *block codes* and *convolution codes* [13]. Along this dissertation only block codes will be considered.

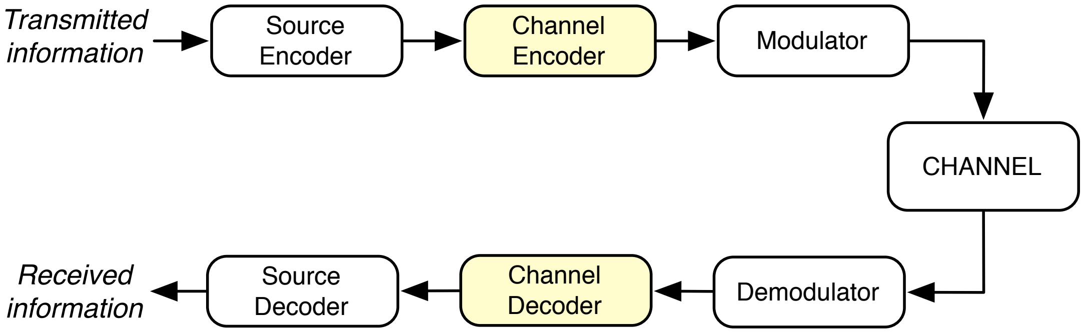

Figure 2.2: Generic digital communication system

2.2.1 Block codes

When using a block code (n,k), the message to be transmitted (in binary format) is divided into blocks of k bits. The encoder takes each of these blocks and converts it into a block of a fixed size of n bits ($n > k$), named as the *codeword*. The added $n-k$ bits, known as *parity bits*, add redundancy to the message, which is used by the decoder in the detection and correction of errors. A *systematic code* is such whose codewords are composed by the unaltered message concatenated to the parity bits.

The *code rate* of a code, $R = k/n$, which corresponds to the ratio between the number information bits and the number of bits that compose a codeword, is a measure on how much redundancy is added to the message and serves as an indicator on the performance of a code. Generally, the lower the code rate, the higher are the capabilities of error detection and correction of the code (note that $0 < R < 1$). This value is closely related to the increased bandwidth needed in the transmission when coding is used [13].

For binary codes, there are a total of 2^k codewords that belong to a code, however, due to errors generated by the transmission through the channel, one of 2^n possible words can be received. The decoder's task is to find the codeword that was most likely to be transmitted, based on the received word. A decoder that makes a decision based on data whose value is taken from a finite set, ({0,1} for binary codes), is classified as a *hard-decision* decoder. Decoders can also be of the *soft-decision* type. These decoders receive information from the demodulator about the certainty of its decisions and make use of this knowledge when obtaining an estimation of the transmitted data.

An important concept that is related to the number of errors a code is guaranteed to detect/correct is the one of *minimum distance*. For two codewords $\mathbf{c}_a$ and $\mathbf{c}_b$ the *Hamming distance* corresponds to the number of positions for which $\mathbf{c}_a$ is different from $\mathbf{c}_b$. The minimum distance d_{min} of a code can then be defined as the minimum Hamming distance between all possible distinct pairs of codewords. If a code can correct up all error patterns up to t errors, then the minimum distance of that code is such that: $d_{min} \geq 2 \times t + 1$.

Linear block codes form an important class of error correcting codes (ECCs). One possible definition, is that a code is linear if the sum (using modulo-2 arithmetic) of two or more arbitrary codewords, results on a codeword. The *parity check matrix* **H**, with dimensions $(n-k) \times n$, of a linear block code describes the linear relations that the bits of a codeword must satisfy. Each line of **H**, $\mathbf{h}_i$, contains the coefficients $(h_{i1}, h_{i2}, ..., h_{in})$ of a *parity check equation*. A restriction of this type is satisfied by an arbitrary codeword $\mathbf{c}$=$(c_1, c_2, ..., c_n)$ if[3]:

$$c_1 \bullet h_{i1} \oplus c_2 \bullet h_{i2} \oplus ... \oplus c_n \bullet h_{in} = 0. \tag{2.6}$$

Given that there are $n-k$ independent equations with n variables, there's a total of 2^k binary vectors of size n, the codewords, that are possible solutions to the system formed by the parity check equations of a parity check matrix. Therefore, for every codeword **c** of a code with a parity check matrix **H**:

$$\mathbf{c} \times \mathbf{H}^T = \mathbf{0}. \tag{2.7}$$

Lets now consider a received word $\hat{\mathbf{c}} = \mathbf{c} + \mathbf{e}$, where **e** represents an error vector of length n:

$$\hat{\mathbf{c}} \times \mathbf{H}^T = \mathbf{c} \times \mathbf{H}^T + \mathbf{e} \times \mathbf{H}^T = \mathbf{0} + \mathbf{e} \times \mathbf{H}^T = \mathbf{s}. \tag{2.8}$$

The $(n-k)$ length vector **s** is called the *syndrome* and corresponds to the sum of the columns of **H** given by the positions of the 1s in the vector **e**. There are 2^k different error vectors that can produce the same syndrome, and the subset formed by these vectors is called a *coset* of the code. *Syndrome decoding* works by calculating the syndrome of the received word and assuming that the error vector that occurred is the one with the least number of 1s, from the coset corresponding to the obtained syndrome [13].

A codeword **c** and its respective message $\mathbf{m} = (m_1, m_2, ..., m_k)$ are related by the following expression:

$$\mathbf{m} \times \mathbf{G} = \mathbf{c}, \tag{2.9}$$

[3] The $\oplus$ and $\bullet$ operators refer to a modulo-2 arithmetic sum and product, respectively.

where the matrix $\mathbf{G}$ is known as the *generator matrix*. For systematic codes the generator matrix is of the form $\mathbf{G} = [\mathbf{P} \mid \mathbf{I_k}]$, and the equivalent parity check matrix is $\mathbf{H} = [\mathbf{I}_{n-k} \mid \mathbf{P}^T]$, where $\mathbf{P}$ is known as the *parity matrix* and $\mathbf{I}_i$ represents the identity matrix of dimensions $i \times i$.

We'll follow now with a brief description on a couple important classes of block codes which will be relevant on later chapters:

- *Bose-Chaudhuri-Hocquenghem* (BCH):
  Invented by R. C. Bose, D. K. Ray-Chaudhuri [14], and A. Hocquenghem [15], these are codes with well defined error correcting capabilities. BCH are cyclic codes, which is a special class of linear codes. Cyclic codes are such that cyclically shifted versions of codewords are also codewords. For any positive integer $i \geq 3$ and $t < 2i - 1$, there exists a binary BCH code with the following properties [13]:

  - Codeword length: $n = 2^i - 1$
  - Number of parity bits: $n - k \leq i \times t$
  - $d_{min} \geq 2 \times t + 1$
  - Error-correction capability: t errors on a codeword

- *Low-Density Parity-Check* (LDPC):
  These are powerful linear codes characterized by a sparse, (i.e. with low density of 1s), parity check matrix [16]. The decoding uses soft information and follows an iterative process. These type of channel codes will be utilized on coding schemes on chapters 4 and 5. Among the various decoding algorithms [17], the Logarithmic Sum Product Algorithm (LSPA) [18] will be employed. The utilized codes are from the WiMAX standard [19].

2.2.2 Interleaving and concatenated schemes

Some ECCs are designed to correct a limited number errors, assumed to be uncorrelated and randomly distributed [20]. However, because of the physical nature of wireless channels, bursts of correlated errors might occur. A technique known as *interleaving* can be applied to deal with this issue.

Interleaving works on the following principal:

1. An *interleaver* is placed after the channel encoder, and randomizes the order of encoder bits before passing them to the modulator. Note that this randomization can be done with bits from 1 or more codewords.

2. In the receiver, a *deinterleaver* that performs the reverse operation of the interleaver is placed before the channel decoder, spreading eventual bursts of errors and increasing the likelihood of obtaining correctable received codewords.

Interleaving is also applied on concatenated schemes. In this type of schemes, the channel encoder from figure 2.2 may consist on an inner and an outer encoder separated by an interleaver, as depicted on figure 2.3. The inner code is typically a powerful code, therefore, decoder failure usually implies a great amount of errors on the message bits. The interleaver is used to spread the burst of errors caused by a potencial failure of the inner decoder, so that the outer decoder (usually a code with a well defined error correcting capability) is able to correct them.

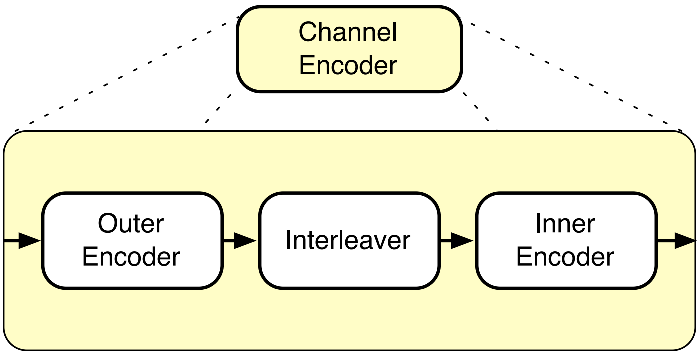

Figure 2.3: Channel encoder of a concatenated scheme using two ECCs and an interleaver.

There are different techniques of implementing interleaving:

- *Block interleaving:* bits are written on a buffer array of dimensions $R \times C$, column-wise. After the array is full they are transmitted row-wise. Note that a delay proportional to the buffer size is introduced.

- *Convolutional interleaving:* A convolutional interleaver consists of a set of shift registers, each with a different delay (usually for the first shift register the delay is 0, for the second is D, for the third is $2D$, etc). Each new bit from the input stream is shifted into the next shift register and the oldest bit in that register is shifted out.

- *Random interleaving:* a block of N input bits is written into the interleaver, and then the bits from the block are read out in a random manner. Typically, the permutation of the input bits is defined by a uniform distribution [20].

Figure 2.4 portrays an example on how interleaving can spread a burst of errors along a bit sequence. Note that for the depicted example, block interleaving with a buffer of dimensions 5×4 was utilized.

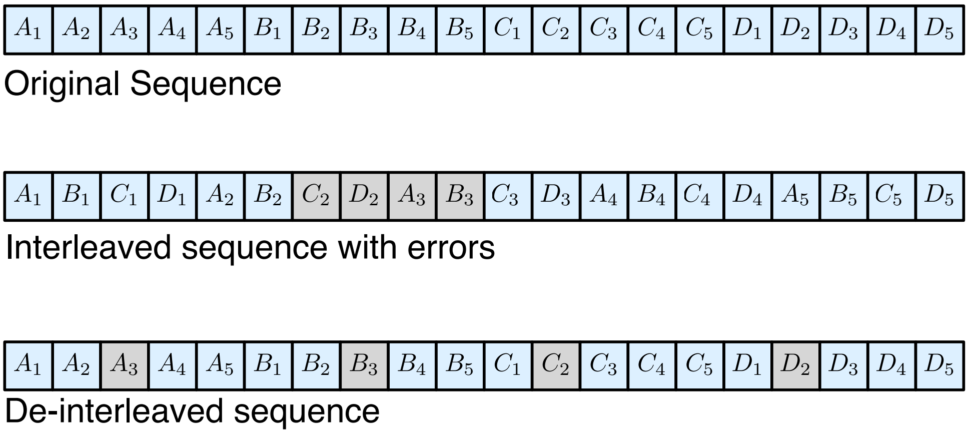
A_1 A_2 A_3 A_4 A_5 B_1 B_2 B_3 B_4 B_5 C_1 C_2 C_3 C_4 C_5 D_1 D_2 D_3 D_4 D_5
Original Sequence
A_1 B_1 C_1 D_1 A_2 B_2 C_2 D_2 A_3 B_3 C_3 D_3 A_4 B_4 C_4 D_4 A_5 B_5 C_5 D_5
Interleaved sequence with errors
A_1 A_2 A_3 A_4 A_5 B_1 B_2 B_3 B_4 B_5 C_1 C_2 C_3 C_4 C_5 D_1 D_2 D_3 D_4 D_5
De-interleaved sequence

Figure 2.4: Example of block interleaving and on how the use of interleaving spreads bursts of errors. The grey blocks represent bit errors

2.3 Coding for Secrecy

On the previous section we gave some insight on how coding techniques can be used to face the issue of having errors during a transmission. We'll now consider the problem on how to keep the information transmitted during a communication secret for undesirable parties, on the context of physical-layer security [10].

Security in communications is usually achieved through the use of cryptographic methods applied on upper layers [1]. These methods are based on the unscalability of computing mathematical operations such as the prime factorization of large numbers. The efforts made to accomplish secrecy at the physical layer follow a different approach. Secrecy is attained through the utilization of signal processing and coding techniques that aim to explore inherent characteristics of the communication such as the errors introduced by a noisy channel.

On this dissertation we will only regard the case where the eavesdroppers are passive listeners of the communication. For that we will consider the *wiretap channel* model, introduced in 1975 by Aaron Wyner [2].

2.3.1 Wiretap Channel

The wiretap channel model considers two legitimate communicators, *Alice* and *Bob*, and a passive eavesdropper *Eve*[4]. The channel that separates Alice and Bob is often referred as *main channel*, and the channel through where Eve observes Alice's transmission

[4] Throughout this dissertation we'll continue to use the names Alice, Bob and Eve when referring to a legitimate transmitter, receiver and adversary eavesdropper, respectively.

is known as *wiretapper channel* or *eavesdropper's channel.* A system for this model is depicted on figure 2.5.

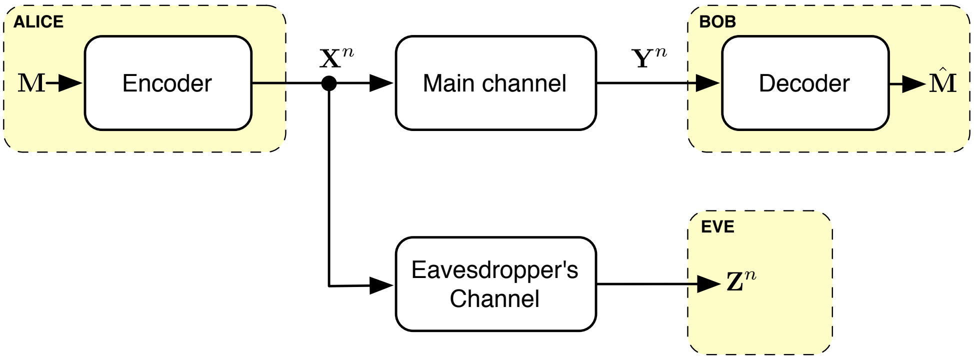

Figure 2.5: The wiretap channel

Alice wants to send a message M composed of k bits to Bob, so she encodes M into a codeword X^n of length n and transmits it onto the main channel from where Bob receives an estimated version Y^n. Eve observes through the eavesdropper's channel an estimation Z^n of the sent codeword, which is assumed to be more degraded than the one received by Bob. Alice and Bob can decide the encoding and decoding processes, however Eve has full knowledge on these mechanisms and is assumed to not possess computational restraints. This model introduces the problem on how to explore the difference of quality between the main and eavesdropper's channels to assure that secrecy and reliability constraints are fulfilled.

The original model proposed by Wyner considered that both the main and eavesdropper's channels were discrete memoryless channels (DMCs) and introduced the notion of *secrecy capacity*, which corresponds to the maximum achievable information rate k/n that can guarantee secrecy and reliability, as a function of the channels' parameters.

Many variations of wiretap channels have been considered. For example the authors of [3] and [4] treated the main channel as a perfect channel and the eavesdropper's channel as a BEC. On [21] the main channel was also considered as perfect but on the other hand the eavesdropper's channel was treated as a BSC. Finally the authors of [9] and [7] considered both channels as additive white Gaussian noise (AWGN) channels.

Constructing a code that fulfills a secrecy constraint proves to be a different challenge than constructing an ECC. On the next subsection we will introduce a coding method for the wiretap channel which was also presented in [2]. Many codes were developed for application to this coding method [3] [4] [5], that satisfy secrecy constraints. Such codes are commonly known as *wiretap codes*.

2.3.2 Nested coding method

To understand how this coding method works, let's take a look at the original example from [2].

Consider that the main channel is perfect and that the eavesdropper's channel is a BEC with erasure probability ε. Alice wants to transmit a message $M \in \{0,1\}$ $(k=1)$ so she encodes it into a codeword X^n. Suppose that the encoder works in such a way that when $M=0$, X^n is chosen at random among all the binary sequences of length n with even parity, i.e. with an even number of 1s. On a similar fashion when $M=1$, X^n is chosen at random among all the binary sequences of length n with odd parity. Since the main channel is perfect, Bob could easily decode by checking the parity of the received word. Eve however would have an average number of $n \times \varepsilon$ erasures and only 1 erasure would be sufficient for preventing Eve to know for sure the parity of the sent codeword.

The general idea that can be taken from the previous example is that in the presence of a noisy channel, secrecy might be obtained by randomly selecting for transmission one of several possible codewords related to the message. The random factor of selecting a codeword coupled with the fact that the code is constructed in such a way that different messages possess similar associated codewords, makes it almost impossible to guess the message in the presence of a noisy channel. Therefore, on a wiretap code, each individual message M_j, $j \in \{0,1,...,2^k-1\}$ is associated to a fixed number of codewords of length n, $(n>k)$, that form a subcodebook C_j of the code[5]. The nested code structure of a wiretap code is illustrated on figure 2.6.

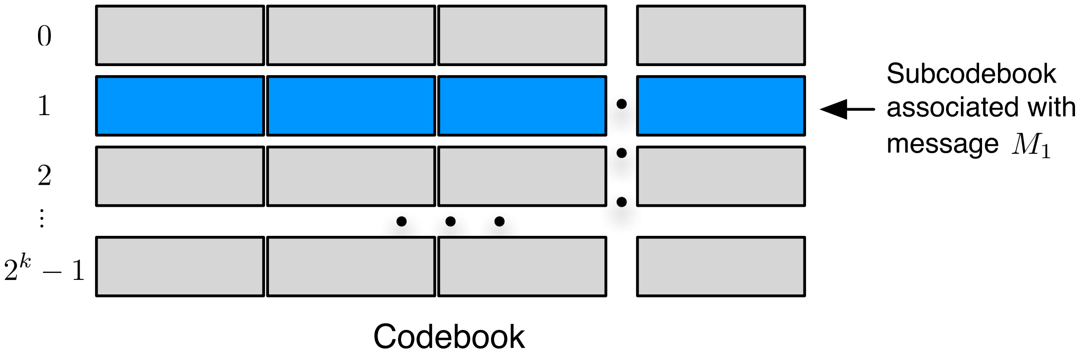

Figure 2.6: Nested structure of a wiretap code. Each block represents a distinct codeword.

On [22] Ozarow proposed the use of linear block codes for the nested coding scheme. This implementation of the nested coding scheme is sometimes referred as *coset coding*

[5] Note that C_j is a vector subset from $\{0,1\}^n$ and that the two subsets corresponding to any pair of arbitrary messages must be disjoint.

scheme and consists on the following steps:

1. A linear block code $\mathscr{C}\ (n, n-k)$ with parity check matrix $\mathbf{H}$ is selected.

2. Alice encodes a message $\mathbf{m}$ of k bits into $\mathbf{X}$, by randomly selecting a word from the coset correspondent to the syndrome equal to $\mathbf{m}$.

3. Bob receives $\mathbf{X}$ (it is assumed he possesses a perfect channel) and obtains the message by calculating the syndrome of the received word, i.e, $\mathbf{X} \times \mathbf{H}^T = \mathbf{s} = \mathbf{m}$.

The only way for Eve to obtain the correct message is if the received word $\mathbf{Z}$ belongs to the coset from where $\mathbf{X}$ was randomly chosen. For that to happen the error vector generated by the channel would have to be a codeword of $\mathscr{C}$, since every pair of arbitrary words that belong to the same coset differ on a codeword.

2.3.3 Other approaches for attaining secrecy

An obvious issue with the coding method presented on the previous subsection is that only the secrecy aspect of the communication is considered. If Bob doesn't possess a perfect channel then he would also be unable to obtain the information sent by Alice. We will return to this problem on chapter 5 where we suggest a possible solution. Anyway, different methods have been proposed for obtaining secrecy at the physical layer. For example, the authors of [23] investigated on how to protect a communication against passive eavesdroppers by generating artificial noise. In [8] the random puncturing of information bits using LDPC codes was proposed. The use of scramblers has also been considered while ARQ techniques provide reliability for Bob [9].

The approach we take on this dissertation aims to take into account both the reliability and secrecy facets of a communication. To do so, along the following chapters we will consider the concatenated scheme of figure 2.7. The outer code is there to provide secrecy and may consist on any number of coding operations (scrambling, puncturing, interleaving, etc...), while the inner code is an ECC, considering that on most real scenarios, Bob doesn't possess a perfect channel. These two codes must be matched in such a way that the difference in qualities between Bob and Eve's channels is explored so that Bob has a reliable transmission and a desirable level of security is obtained against Eve. One question that might arise, is how to quantify the secrecy level of a system, and which metrics are used to do so. On the next chapter we will provide answers for such questions.

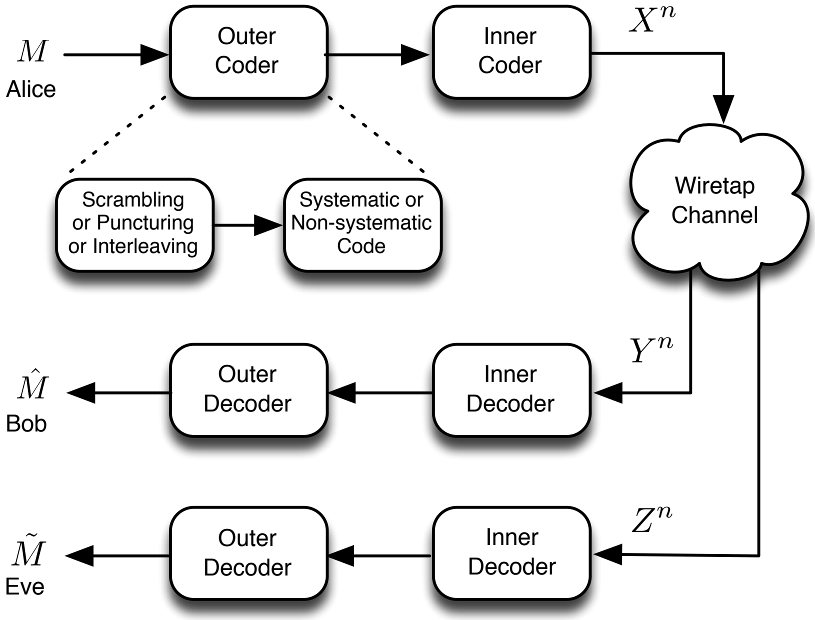

Figure 2.7: Wiretap channel model assuming a concatenated coding scheme, where the outer code is for secrecy and the inner for reliability. The outer coder may consist on any set of encodings.

3

Secrecy Metrics

Metrics that evaluate security are necessary when constructing a code for secrecy or developing a system with this purpose. In this section we will overview the current secrecy metrics as well as the limitations they present. In the last two sections, we will propose new methods for evaluating security, which can be applicable to specific physical-layer coding techniques over realistic channel models such as the Gaussian or fading wiretap channel models.

3.1 State of the Art

Proposed by Claude Shannon in 1949 the first secrecy metric was *perfect secrecy* [24]. For a coding system to achieve perfect secrecy, the mutual information between the message M and it's correspondent encoder output X^n must be exactly zero, i.e.,

$$I(M;X^n) = 0. \tag{3.1}$$

Perfect secrecy indicates that M and X^n are statistically independent which means that the knowledge of X^n doesn't provide any extra information to an attacker about M. Shannon concluded that to achieve perfect secrecy a secret key at least as long as the message would be required, highly limiting the use of this metric.

Aaron Wyner proposed in 1975 the use of a weaker requirement for secrecy [2]. This metric known as *weak secrecy* requires that the rate of the mutual information between M and the eavesdropper's observed codeword Z^n vanishes as the codeword length n goes to infinity, i.e.,

$$\lim_{n\to\infty} \frac{1}{n} I(M;Z^n) = 0. \tag{3.2}$$

Instead of imposing that the codeword X^n doesn't leak any information about M, this criterion requires that the amount of information about M leaked by Z^n is sufficiently small such that the $1/n$ factor can still drive the quantity to zero. Note that it's implied that the legitimate user has some sort of advantage over the eavesdropper.

Weak secrecy was not deemed completely satisfactory. In fact, some code constructions that achieve weak secrecy show evident flaws in terms of security [10]. Being so, Ueli Maurer [25] introduced the concept of *Strong Secrecy* where secrecy is achieved if the mutual information between M and Z^n is asymptotically zero in n, i.e.,

$$\lim_{n\to\infty} I(M;Z^n) = 0. \tag{3.3}$$

The list of information-theoretic metrics presented above proved to be impracticable for application to realistic channel models such as Gaussian or fading channels. Therefore, some authors evaluated the performance of code constructions through the BER at the output of the decoder [7], [9]. This analysis is practical and can be done by simulation the same way ECC are evaluated.

The authors of [7] introduced the concept of *security gap*. Target values of BER are chosen for Eve and Bob, and the respective operation points in terms of SNR are identified. The security gap corresponds to the difference in dB between these two values of SNR. This metric is useful for code/system design since it evaluates the required advantage the legitimate user needs over the eavesdropper as well as the threshold operation points for reliability and security.

3.2 Shortcomings

The metrics introduced in the previous section provide many tools for evaluating specific coding schemes security wise. Yet, the applicability of these metrics is very dependent on the type of channel models in question.

When considering more practical wiretap channel models, performing the security analysis with information-theoretic metrics is an unmanageable task for most cases. For example, some authors managed to construct codes for the nested coding scheme described in section 2.3.2, that achieve strong secrecy [6]. However, these codes were constructed for the erasure wiretap channel model which considers a perfect channel for the legitimate user, an assumption which is not suitable for most real-world scenarios. One could be misguided and think that by concatenating an ECC to the security code in question, a perfect channel would be achieved for Bob and the system would provide both reliability and security. This doesn't work because the ECC also reduces the noise on Eve's channel so, when constructing a code for a realistic scenario, both reliability and secrecy factors must be considered together.

Another limitation on using information-theoretic metrics is that the code is designed to fulfill a secrecy condition as the codeword length goes to infinity. This shows to be a shortcoming when the designer wants to work in real-world systems that require short blocklength codes. Nonetheless, information-theoretic metrics are still the most desirable when possible to apply.

Using only the BER to perform security analysis is not always advisable because high error rates don't strictly imply that no information has been leaked to the eavesdropper. A BER analysis is performed through simulation. The best known decoder/attack assumed to be used by the eavesdropper is emulated, and the value for the BER is obtained by averaging the outcome of a large amount of independent runs. Since the result is an average, the reliability of this metric is uncertain when working with short blocklengths.

Motivated by these shortcomings and in order to evade some of them, in the next two sections we will introduce new methods for security evaluation. These are for application on security schemes that follow the model depicted in figure 2.7, with the purpose of providing practical tools to aid in the construction of models that achieve secrecy and reliability in realistic scenarios.

The following methods are based on the BER but instead of focusing on the average value of error rates, an approximation of the PMF of the number of errors is obtained by simulation. The knowledge of the CDF of errors can then be used to comprehend how the system behaves and to provide thresholds of operation. The main interest of these methods is that, contrary to the ones presented on the previous section, they can be used to evaluate the performance of systems on the short blocklength regime. While the security bounds provided by information-theoretic metrics are still the most reliable ones, the following methods continue to be relevant since the ease of simulation-based characterization of security allows them to be applied to realistic channel models where it's not always known how to perform an information-theoretic analysis.

3.3 The Bit Error Cumulative Distribution Function

Let's consider the model illustrated on figure 2.7, where the outer decoder is meant to provide secrecy. The failure of this decoder is a determinant factor to prevent the eavesdropper from obtaining a good estimate of the message, $\tilde{M}$. In a similar fashion, the legitimate user should reliably succeed the decoding. The first proposed metric is the Bit Error Cumulative Distribution Function. It evaluates the system by measuring the likelihood of decoding success, when the *outer* coder is a t-error correcting code (i.e. a code with minimum distance $t \times 2 + 1$).

Definition 1 (Bit Error Cumulative Distribution Function)**.** The bit error cumulative distribution function, BE-CDF^{bc}(t, SNR, $\mathscr{S}_m$, $\mathscr{C}_i$), gives the probability of having t or less errors, $\Pr(E \leq t)$, as a function of the SNR for a message of size $\mathscr{S}_m$, encoded with a code $\mathscr{C}_i$ (refers to the optional *inner* code).

Being able to predict the rate at which the secrecy decoder fails or succeeds is of crucial importance when constructing a system for providing both reliability and secrecy.

Working with the distribution of the number of errors allows us to overcome the shortcomings of the BER, when evaluating the performance of t-error correcting codes. Assuming uniform error distribution and using the BER measured before the outer decoder to evaluate the likelihood of decoder failure is not a reliable method because, when the blocklength is short, errors are not guaranteed to occur so uniformly.

The BE-CDF^{bc} provides useful information when choosing possible SNR operation points for the legitimate user and the eavesdropper, by evaluating the effect of the channel and inner code. One the other hand, if Bob and Eve are expected to operate at given SNR values, this metric can also be used to provide information about which t-error correcting codes (of the BCH class for example) could be used as an outer code, to fit the given restraints.

For example, let's consider a simplified system without an inner code, that uses BPSK modulation over an AWGN channel. The BER at the output of the channel is given by [26]

$$P_b = \frac{1}{\sqrt{\pi}} \int_{\sqrt{SNR}}^{\infty} e^{-t^2} dt \tag{3.4}$$

Using blocks of size n and assuming that the bits transmitted are independent, the probability of having t or less independent errors in a word, $\Pr(E \leq t)$, can be modelled as a binomial distribution and is given by:

$$\Pr(E \leq t) = \sum_{i=0}^{t} \binom{n}{i} {P_b}^i (1 - P_b)^{n-i}. \tag{3.5}$$

Let's now consider that we intend to use as an outer code, a t-error correcting code of length 127 that is able to correct up to 10 errors. Figure (3.1) shows the behave of functions (3.4) and (3.5) for this set of parameters.

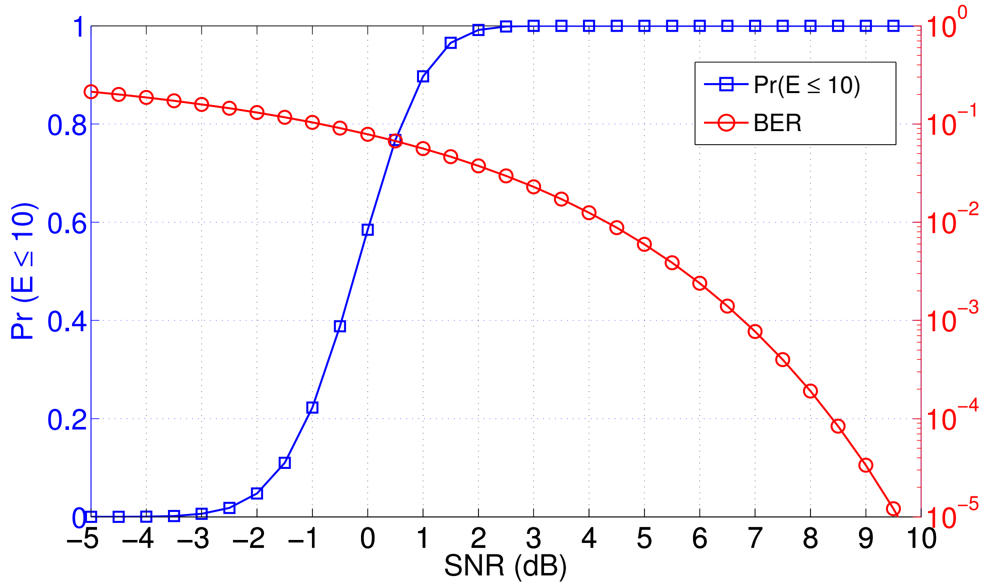

Figure 3.1: Bit error probability and probability of having 10 or fewer errors on a word of length 127, for an AWGN channel with BPSK modulation.

We can now evaluate the range of values of SNR where Bob should operate to achieve a reliable communication. For example, if we consider the communication reliable if Bob successfully decodes at least 99,9% of the blocks, i.e. $\Pr(E \leq 10) \geq 0.999$, Bob's SNR should be over 2.6 dB as indicated in figure 3.1. On the other hand, if we consider the communication secure if Eve fails the decoding 99,9% of the time, i.e. $\Pr(E \leq 10) \leq 0.001$, Eve would need to operate at SNR below -3.78 dB. Note that this simple example is not suggested to be used as a possible coding scheme for secrecy, it simply illustrates how the BE-CDF^{bc} can be used to evaluate a system.

This metric uses the probability of successful decoding of the outer coder to measure the reliability and security aspects of the system. However, decoder failure doesn't necessarily imply that the eavesdropper can't obtain most of the message bits. To address this issue, a new metric is presented in the next section.

3.4 The Bit Error Rate Cumulative Distribution Function

As stated in the previous section, decoder failure doesn't necessarily guarantee that most of the bits aren't leaked to the eavesdropper. The new metric we propose in this section, named Bit Error Rate Cumulative Distribution Function, fortifies the security guarantee by measuring the probability of having a decoder failure that generates a BER close to 0.5 in the estimated message bits. Let $\hat{P}_b$ be the proportion of errors measured over $\mathscr{S}_b$ message bits at the output of the outer decoder, the metric introduced in this section evaluates the probability that $\hat{P}_b > 0.5 - \delta$ for any δ specified.

Definition 2 (Bit Error Rate Cumulative Distribution Function)**.** The Bit Error Rate Cumulative Distribution Function, BER-CDF$^{ac}(\delta, E_b/N_0, \mathscr{S}_b, \mathscr{C})$ is the quantity

$$\Pr(\hat{P}_b > 0.5 - \delta) \tag{3.6}$$

calculated over $\mathscr{S}_b$ estimated message bits for a code $\mathscr{C}$ as a function of E_b/N_0, where $\mathscr{C}$ may be the concatenation of an (optional) inner code $\mathscr{C}_i$ and an outer code $\mathscr{C}_o$.

The BER-CDF^{ac} measures secrecy by evaluating the probability of the eavesdropper being kept from obtaining useful information on a block. The security guarantees obtained from this method are much stronger than just considering the BER.

Similarly to the BE-CDF^{bc}, for this metric the user must specify a required level of security. For example, the designer could consider a system secure if $\Pr(\hat{P}_b > 0.4) \geq 0.999$, and design it with the purpose to fulfill this restriction while keeping the required E_b/N_0 on the range of levels expected by the eavesdropper.

Note that the BER-CDF^{ac} is actually the complement of the CDF, being the chosen nomenclature consistent with that of BE-CDF^{bc}. Also, because we are calculating this metric after the decoder, it makes sense to use E_b/N_0, rather than SNR, although the conversion can be made if desired. The superscripts *bc* and *ac* indicate that the metrics are measure respectively *before* and *after* the outer decoder, as depicted in figure 3.2.

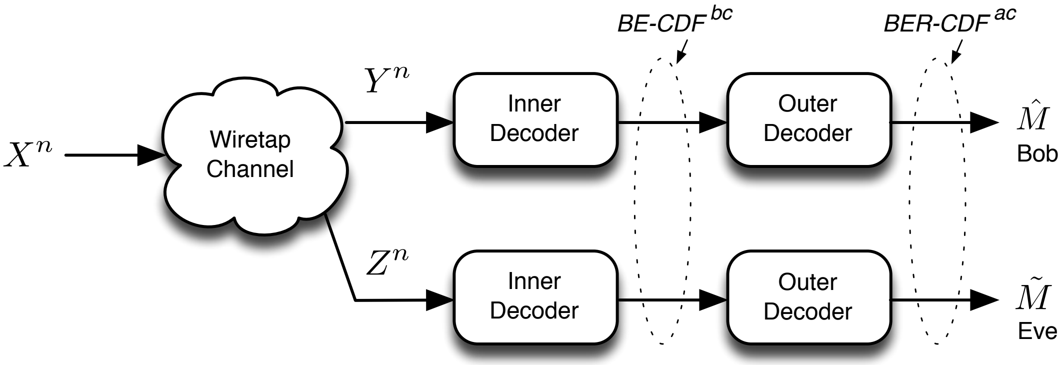

Figure 3.2: Wiretap channel model assuming a concatenated coding scheme, where the outer code is for secrecy and the inner for reliability. The application points of the proposed new metrics are illustrated.

The two methods introduced can be applied as a pair to help in the design of systems that aim to provide both reliability and secrecy. The BE-CDF^{bc} can be used to identify regions of operation for Bob, in terms of SNR, that provide a high rate of decoding success and therefore, reliability. It also provides information of acceptable regions of operation for Eve that guarantee a high probability of decoder failure. The BER-CDF^{ac} can then be used to evaluate the contribution of the outer code in terms of generating a considerable BER when decoder failure occurs.

In the next chapter we will put these two metrics to use and analyze a proposed security scheme, showing how the information obtained from the metrics can be used to tune and further develop the scheme.

4

Interleaved Coding for Secrecy

Having proposed new possible methods for analyzing schemes that intend to achieve secrecy, in this chapter we propose a concatenated coding scheme and use the metrics introduced in sections 3.3 and 3.4 to make its security evaluation. The scheme, which will be presented in section 4.1, was first introduced in [27], although a security evaluation is yet to be made and there's still no clearance on which codes to use. Along this chapter we'll aim to fill these gaps and suggest a methodology for selecting possible codes to apply for this scheme.

4.1 Coding for Secrecy Scheme

The system we will present this section follows the concatenated coding scheme of figure 2.7. An inner code is utilized to correct transmission errors and provide reliability, while an outer code is employed to provide secrecy.

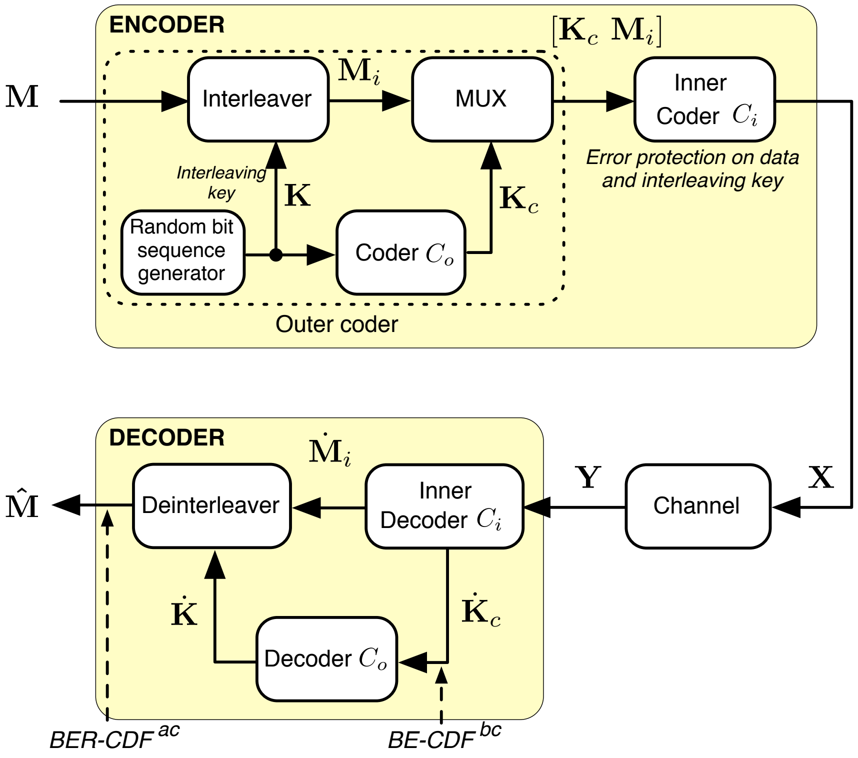

Figure 4.1: Encoder and decoder processes of the analyzed scheme

The outer code consists of a set of encodings. A word[1] K^k, is randomly generated and used as a permutation key to interleave the message M^m, producing a shuffled message

[1] Along this section, we will use the notation X^x to denote a word X formed by x bits. The superscript label x will be dropped, whenever the size of X is clear from the context.

M_i^m. The key is then encoded for extra error protection into K_c^l by a code C_o, of dimensions (l,k), and concatenated to the interleaved message. This concludes the outer encoding.

The concatenated encoded key and interleaved message, i.e. $[K_c\ M_i]$, are then encoded by the inner coder, a systematic code C_i of size $(n, l+m)$, and transmitted onto the channel. The receiver performs soft decoding on the received codeword, obtaining estimations $\dot{M}_i$ and $\dot{K}_c$, of the interleaved message and encoded key respectively. The estimated coded key is then decoded and used to deinterleave $\dot{M}_i$ generating the estimated message $\hat{M}$.

Figure 4.1 illustrates the encoder and decoder processes described. It should be clear by now that getting an estimated key $\dot{K}$ without errors is crucial to obtain a good estimation of the message. Due to the mapping between keys and permutations, any errors on the key can result in high error rates on the deinterleaved message. It is assumed that the distribution of the number of ones (or zeros) of the message is somewhat uniform, in order to avoid messages composed only of ones or zeros, which would make $M = M_i$.

The point of this scheme is to prevent the eavesdropper from obtaining a correct estimation of the key. Being so, for this system we are going to consider a variant of the wiretap channel, depicted in figure 4.2. A friendly jammer is active during the transmission of bits associated with K_c producing extra interference with power equal to a fraction α of Alice's transmit power, intending to degrade the eavesdropper's channel. The idea is to give the legitimate user some sort of advantage, due to his location or knowledge of the jamming signal, so that the degradation on his channel is minimal compared to the eavesdropper's. Some amount of interference might be present on Bob's channel, hence the use of code C_o on the key.

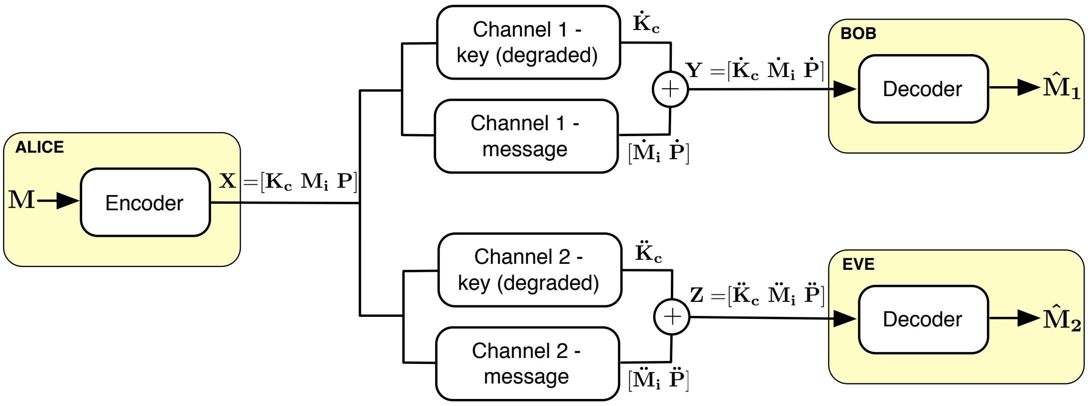

Figure 4.2: Variant of the wiretap channel where a jammer is present during the transmission of the key, hence the degraded channels. P represents the parity bits of the codeword generated by the inner code C_i.

This usage of a jammer over the key, only, requires strict synchronization with Alice, and that the eavesdropper doesn't have any information on the jamming signal and/or is in a geographic position that doesn't grant him the same advantage that Bob has. These factors are out of the scope of this thesis, and, therefore, won't be taken into consideration in the following sections.

Now that we presented the coding for secrecy scheme, we'll use the methods introduced in sections 3.3 and 3.4 to help us understand the behavior of the scheme and make some considerations about the proper codes to be used.

4.2 Code Selection

The literature presents a vast amount of error correcting codes. For the inner code C_i, given that we need a powerful systematic code, we decided to focus on LDPC codes. However, the inner code can't be powerful enough to correct all the errors on the coded key, even when in the presence of a high power jamming signal, or else Eve would be able to reliably get a correct estimation of the key. Given these constraints, we chose, as example and without loss of generalization, a LDPC code of dimensions $(1248, 1040)$ to be used as the inner code, even though there are many other options that would be acceptable.

Selecting a suitable code C_o is an important task. This code must be able to correct eventual errors left on the coded key after the inner decoding done by Bob. On the other hand, it is crucial for security that this code isn't strong enough to correct Eve's errors. For these reasons we chose to use BCH codes, as these type of codes possess a well defined t error correcting capability. To avoid the possibility of Eve having such a high SNR that allows the inner code to correct all the errors on the coded key, independently of the power of the jamming signal, the codeword length of code C_o must be greater than the number of parity bits of code C_i. In order to maximize the code rate of the system, we'll consider BCH codes with codeword length 255, since this is the lowest value the codeword length of a BCH can take that is greater than 208 (the number of parity bits of the chosen inner code).

Throughout this section we'll make some considerations on possible options to use as code C_o by analyzing the selected inner code C_i performance for varying values of SNR and jamming power.

4.2.1 Notes on simulations

The figures shown on the following sections were obtained through simulation. The simulations were performed in MATLAB due to the vast amount of toolboxes it gives access to. The PMFs necessary for the construction of some of the figures were obtained by normalization of histograms with never less than 10^5 samples. BER values were calculated using Monte Carlo.

Simulations will be done for transmission over a Gaussian channel using BPSK modulation. In simulations that involve jamming, the emulated eavesdropper has knowledge on the variance of the jamming AWGN signal when performing the demodulation.

The number of maximum iterations of the LSPA, chosen for the decoding of LDPC codes, will be 50.

4.2.2 Analysis for a fixed point of SNR

The performance of this system is dependent on various factors such as the SNR, the power of the jamming signal and the utilized codes. We'll start analyzing this system by choosing a fixed operation point of SNR and observe the effects of the extra interference on the affected bits, as a function of the jamming signal power.

Figure 4.3 shows de behavior of the chosen inner code when performing on an AWGN channel using BPSK modulation.

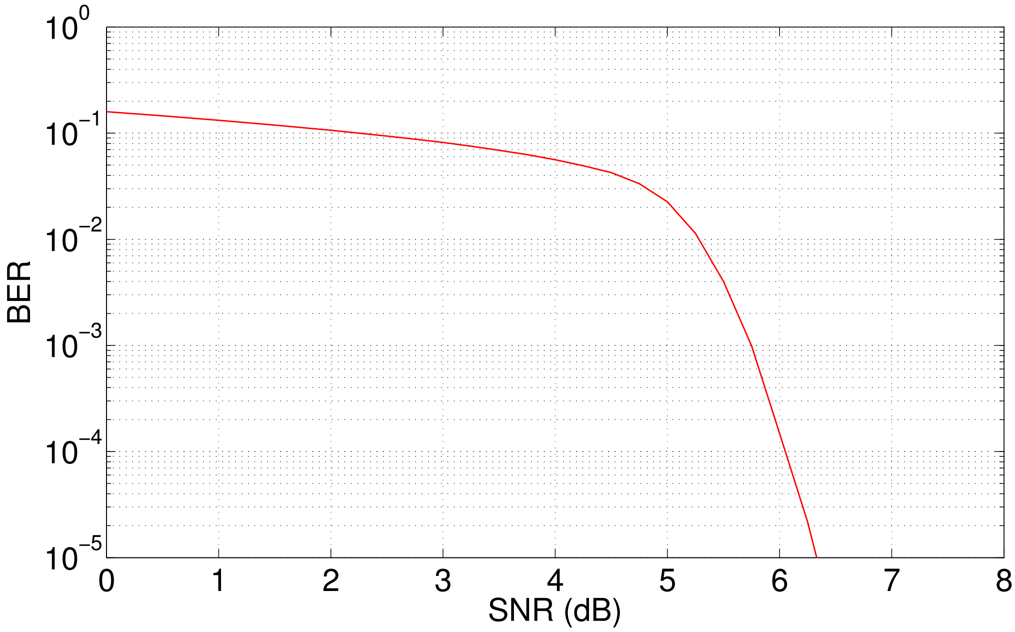

Figure 4.3: BER as a function of SNR for LDPC$(1248, 1040)$ over an AWGN channel using BPSK modulation.

Let's consider the operating point of 6.33 dB which results on a BER approximately 10^{-5}. On these circumstances we assume that the LDPC$(1248, 1040)$ provides a reliable transmission, so we'll start by evaluating this value of SNR as a possible operation point for Bob and Eve.

Given that we chose to use a BCH of length 255 as the code C_o, we'll evaluate the error correcting capabilities of the inner coder for the first 255 bits of a decoded word. Remember that only these bits will be affected by the extra interference caused by a jamming AWGN signal, with power $P_j = \alpha P_a$, where P_a is Alice's transmit power.

Figures 4.4 and 4.5 represent the probability $\Pr(X \leq t)$ as a function of α, where X represents the number of errors on the key section of the decoded codeword. The considered values of t are such that exists a BCH code with codeword length equal to 255 that can correct up to t errors.

Observing the figures we can estimate the behavior of the system depending on the used code C_o. For example, if both Bob and Eve have an SNR of 6.33 dB on the main channel, and the code C_o is a BCH$(255, 87)$, which can correct up to 26 errors, Bob could reliably get an error free key as long as he was affected by a jamming interference with power corresponding to an α less than 0.25. Eve on the other hand would have to suffer interference with power matching an α of around 1.5, for reliably being kept away from obtaining a correct key. Using a BCH$(255, 47)$, which can correct up to 42 errors, would guarantee reliability of transmission for Bob even when affected by a jamming signal with power corresponding to $\alpha = 0.4$, at the cost of requiring the eavesdropper to be affected by a much higher interference.

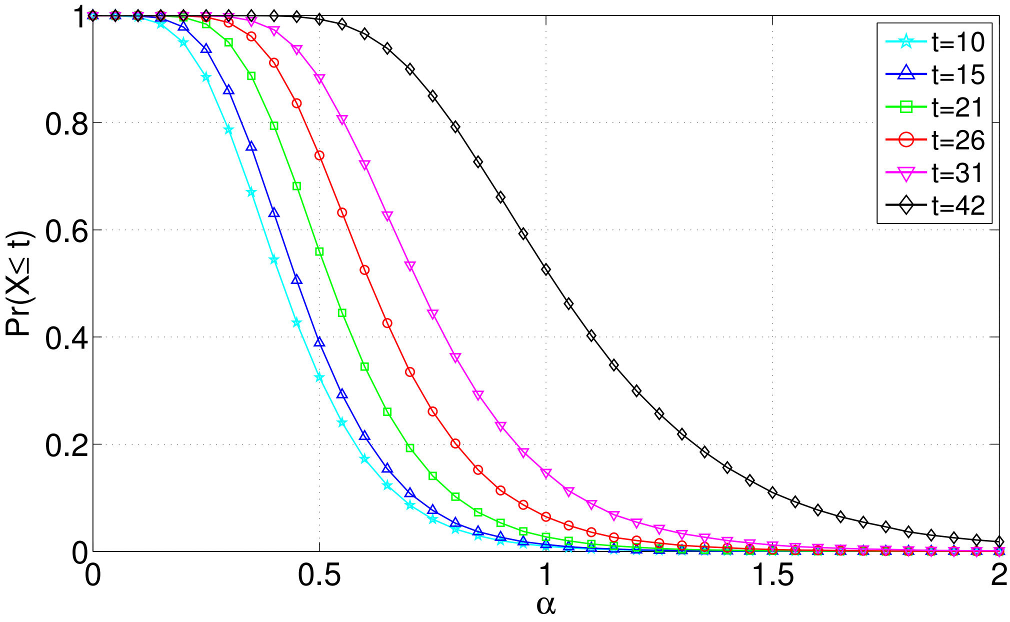

Figure 4.4: Probability of having fewer than t errors as a function of a jamming power of $P_j = \alpha P_a$ applied over 255 bits, for LDPC(1248,1040) decoding at the selected SNR of 6.33 dB for reliability.

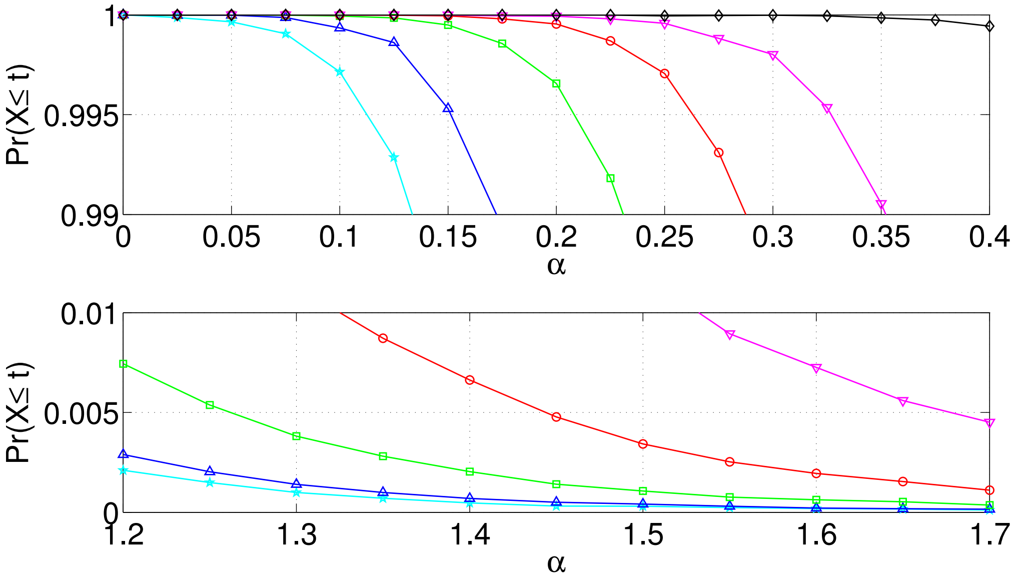

Figure 4.5: Close up of figure 4.4 on the aceptable regions of operation in terms of interference for Bob (upper figure) and Eve (bottom figure).

The previous example highlights the type of compromise one must make when calibrating this system. For a fixed code C_o this type of analysis provides bounds for the interference acceptable for both Bob and Eve. On the other hand, if the values of expected interference on both the legitimate party and eavesdropper are defined for a specific scenario, this analysis provides information about which code C_o should be used to fit the given restraints.

Figure 4.4 also gives us an idea on the levels of interference this system requires to function. It becomes clear that for this value of SNR, Bob shouldn't suffer from interference with α greater than 0.5 or else he won't be able to reliably decode the key, regardless of the chosen outer code. On a similar fashion, Eve should be affected by jamming interference with α greater than 1.2. Figure 4.5 shows in more detail an example of acceptable levels of interference for both parties.

4.2.3 Analysis for fixed values of α

Now that we have an idea on how the system behaves when varying the jamming power, we'll observe how the system responds when the SNR of the receiver varies for fixed values of α, considering the error correcting capabilities of the code C_o. The values of $\alpha = 0.2$ and $\alpha = 1.6$ will be evaluated as the levels of interference that affect Bob and Eve respectively. We'll also consider the use of $\text{BCH}(255,139)$, $\text{BCH}(255,115)$ and $\text{BCH}(255,87)$ as the code C_o, which respectively can correct up to 15, 21 and 26 errors.

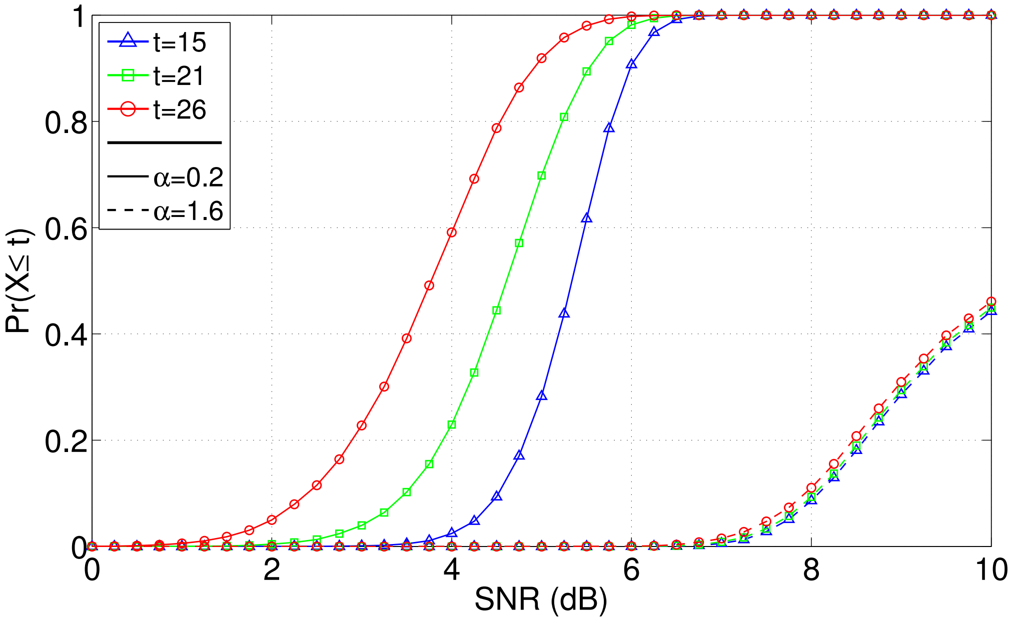

Figure 4.6: BE-CDF^{bc} for the first 255 bits coded by LDPC$(1248,1040)$. These bits are affected by a jamming AWGN signal of power $P_j = \alpha P_a$.

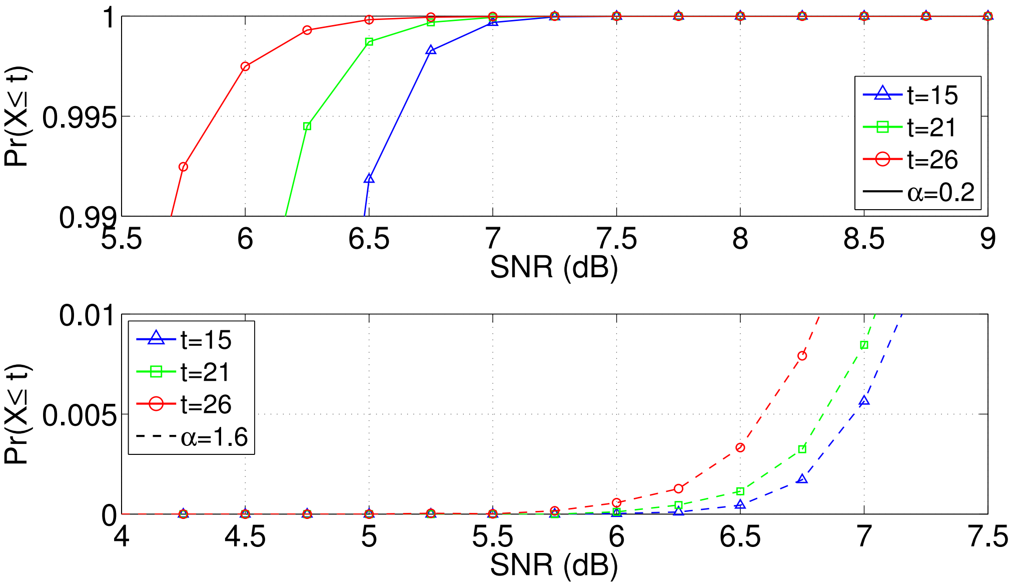

Figure 4.7: Close up of figure 4.6 on the aceptable regions of operation in terms of SNR for Bob (upper figure) and Eve (bottom figure).

Figures 4.6 and 4.7 show the results of the BE-CDF^{bc} metric defined in section 3.3, when applied to this scenario. For the examined values of interference, using a BCH $(255,115)$, which can correct up to 21 errors, would allow Bob to successfully decode the key with probability 0.9997 when the SNR over his channel is above 6.75 dB. For the same code and value of SNR, the probability of Eve getting a correct key would only be

0.0032.

Using a BCH$(255,87)$, which can correct up to 26 errors, a SNR of 6 dB would make Bob get an errorless key with probability 0.9975, while for a SNR of 6.5, Eve's probability would be 0.0033. This example illustrates that it's possible to achieve reliability and secrecy, even if Eve has a better main channel than Bob, depending on the levels of jamming interference that affect them.

Note that until now we only evaluated the probability of obtaining a key without errors. While this type of analysis permits us to evaluate points of operation that assure reliability for Bob and brings closure on how Eve's key decoding capability breaks down, we still can't predict the effects of a bad estimated key on Eve's decode message. Also keep in mind that BCH codes are not perfect codes, which means that there's the possibility of successfully decoding certain error patterns with more than t errors, although this is unlikely to occur.

To give us more insight on how Eve is affected by this scheme, on the next section we'll provide simulation results of the full system, and perform the security evaluation with the BER-CDF^{ac} metric defined in section 3.4, showing the usefulness of the proposed metric.

4.3 Security Considerations

The analysis done on the previous sections gave us an idea on the desirable levels of jamming interference, SNR operation points and possible outer codes that can be used for this system, when fixing the inner code C_i. We'll now evaluate the security brought by this scheme, through the analysis of distribution of error proportion in Eve's message bits.

Figure 4.8 shows the BER-CDF^{ac} calculated for this system, when the code C_o is a BCH$(255,115)$ which can correct up to 21 errors. The values of α used on section 4.2.3 for Bob and Eve were considered (i.e. 0.2 and 1.6 respectively). BER curves are also shown for comparison, and to check the required level of E_b/N_0 Bob needs to operate at, for having reliability of transmission. Note that it doesn't make sense to use the BER-CDF^{ac} for evaluating Bob's performance (the objective is for Bob to have a reliable transmission), nonetheless, we chose to keep the curves of $\Pr(\hat{P}_b > 0.5 - \delta)$ for $\alpha = 0.2$ as example.

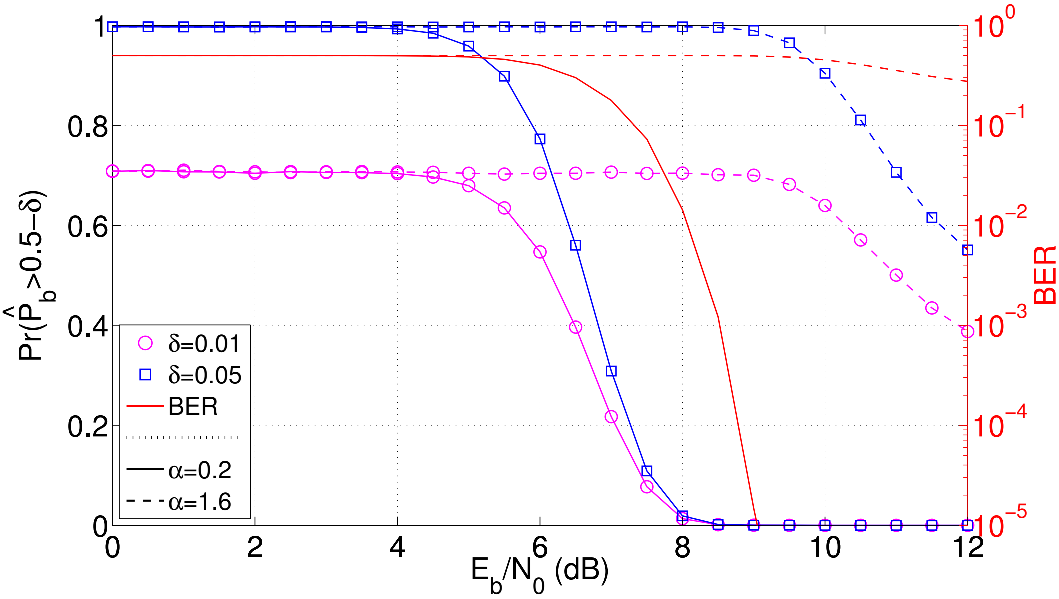

Figure 4.8: BER-CDF^{ac} and BER for the coding for secrecy scheme presented on section 4.1, when the inner code is a LDPC$(1248,1040)$ and the code C_o a BCH$(255,115)$.

For $\delta = 0.05$ we verify that this scheme ensures a $\Pr(\hat{P}_b > 0.5 - \delta)$ close to 1 for a wide range of E_b/N_0 values, when Eve is affected by a jamming signal with $\alpha = 1.6$.

On section 4.2.3 we saw that a SNR of 6.75 dB would cause Eve to fail the decoding of the key with probability 0.9968. From figure 4.8 we note that for an E_b/N_0 of 8.76 dB (equivalent to a SNR of 6.75 dB[2]) the probability $\Pr(\hat{P}_b > 0.45)$ is 0.9959. This result indicates that assuring that Eve fails the decoding of the key might be sufficient for insuring a high proportion of errors on Eve's decoded message.

Note that these results fortify the statement of section 3.2, that the BER can be misguiding when used to evaluate the security of a system with short blocklenghts. E_b/N_0 equal to 10 provides a BER of 0.453 however, the BER-CDF^{ac} shows that for the same value of E_b/N_0, the probability $\Pr(\hat{P}_b > 0.45)$ is around 0.90, meaning that roughly 1 out of every 10 blocks wouldn't possess such a high error rate. A better way to understand this is by looking at the distribution of the number of errors on the decoded message, for this set of parameters, which is presented on figure 4.9.

[2] $SNR_{dB} = (E_b/N_0)_{dB} + 10 \times \log_{10}(m/n)$

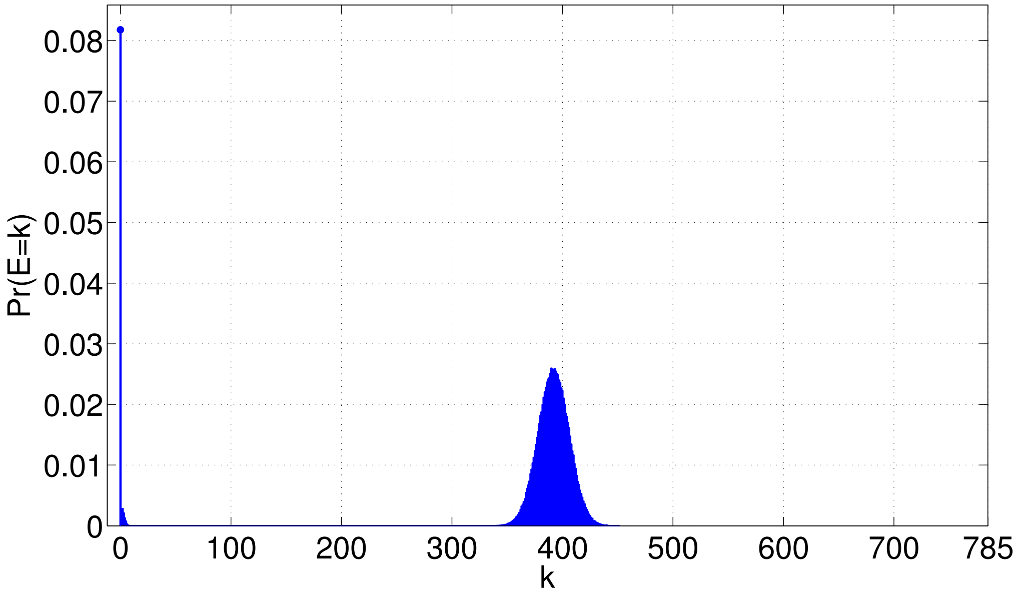

Figure 4.9: PMF of the random variable E that defines the number of errors on the decoded message bits for the security scheme presented on section 4.1, when the inner code is a LDPC$(1248,1040)$, the code C_o a BCH$(255,115)$, the value of α is 1.6 and the E_b/N_0 of operation is 10.

Even with a BER of 0.453 we can see on figure 4.9 that there's a probability of 0.083 that Eve gets a correct estimation of the message, which is far from desirable from a security perspective. The Gaussian centered on $k = 392$ is the consequence of a key with errors, while the residue close to $k = 0$ is the result of the situation when the key is correctly decoded but the LDPC doesn't correct all errors on the interleaved message.

Nevertheless the BER can still be used to evaluate the reliability aspect of the scheme, and to identify regions of operation for Bob. We conclude the analysis of this scheme with the graphic representation of the BER as a function of the E_b/N_0 and the power of the received jamming signal. Once again the inner code C_i is the LDPC$(1248,1040)$ and the code C_o is the BCH$(255,115)$.

We see on figure 4.10 that Bob can still have reliability of transmission for some values of α greater than 0.2, at the cost of a small penalty in terms of E_b/N_0. However, for α greater than 0.6, reliable transmission becomes impractical.

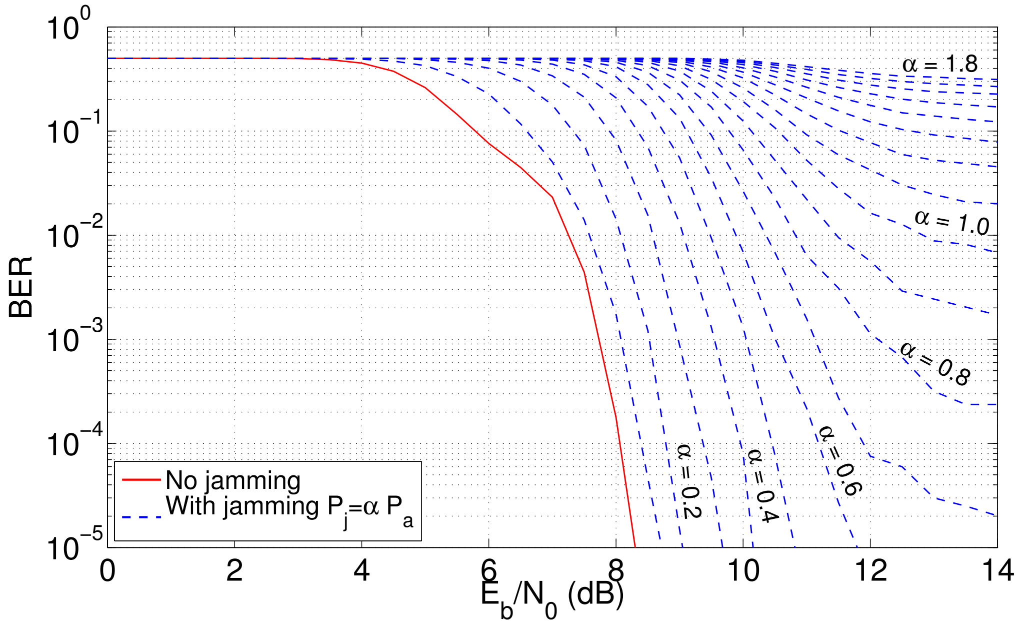

Figure 4.10: BER as a function of E_b/N_0 and jamming power, for the security scheme presented on section 4.1, when the inner code is a LDPC$(1248,1040)$ and the outer code a BCH$(255,115)$.

On the following chapter the discussion developed so far will come in handy, since the scheme presented on the next chapter shares some similarities with this one, although serving a different purpose.

5

Creating DMC/BSC for Secrecy

Throughout this dissertation, physical-layer security was approached from a practical perspective. On Chapter 3 we observed how the current secrecy metrics are impractical to apply to real world scenarios, and suggested new metrics to evaluate coding for secrecy schemes. Then, on Chapter 4, a scheme with the objective to provide both reliability and secrecy was introduced, and analyzed using the previously defined metrics.

We will now use the new metrics introduced in Chapter 3 for a different purpose. A great amount of work has been developed in attempts to provide code constructions of wiretap codes that satisfy information-theoretic security constraints [6], [3]. However, these code constructions exist only for discrete memoryless wiretap channels, and require either a noiseless channel for Bob and/or a degraded wiretap channel for Eve [1]. In order to utilize these code constructions in real-world scenarios, it becomes relevant to design channel coding schemes that aim to produce an effective wiretap channel over which these codes can be applied to.

On this chapter we will show how the metrics from sections 3.3 and 3.4 can be used to help us fulfill this objective. The point is to achieve an information-theoretic security result on a real-world scenario, by concatenating a coding scheme that can be modeled as a BSC when working on a realistic channel, to an additional code that can achieve strong secrecy over a discrete memoryless channel (e.g. [5]), as shown on figure 5.1.

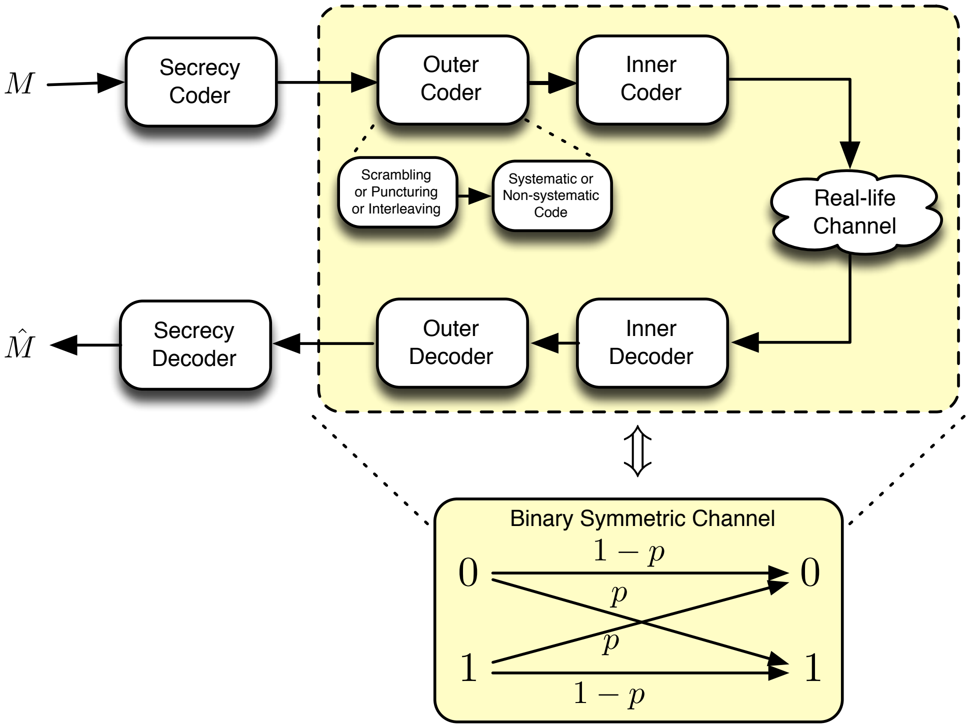

Figure 5.1: A concatenated coding scheme may be utilized to emulate a BSC over which known secrecy codes may operate for information-theoretic security.

First we will present an example coding scheme, and use the BE-CDF^{bc} and BER-CDF^{ac} to analyze it's performance. On the final section of this chapter we will make the necessary considerations to show that the scheme can be used to model a BSC.

5.1 Interleaved Coding for Secrecy with a Hidden Key

In this section we will present the concatenated coding scheme that will be the object of focus for the remainder of the chapter.

This scheme shares several similarities with the one presented on section 4.1. Once again, a word K^k is randomly generated and used as a permutation key to interleave the message M^m, giving origin to a shuffled message M_i^m. Then, the key is concatenated to the interleaved message and coded by the inner coder, a systematic code C_i of size $(n, k+m)$. Note that until now the encoder process is mostly the same as the one from section 4.1, the only difference being the non presence of a code C_o that first encodes the key.

After the encoder process, only the last $n-k$ bits from the obtained codeword are transmitted onto the channel. This means that the first k bits of C_i, referring to the key K, are not transmitted, being the information on those bits only embedded on the parity bits. Figure 5.2 illustrates the presented scheme.

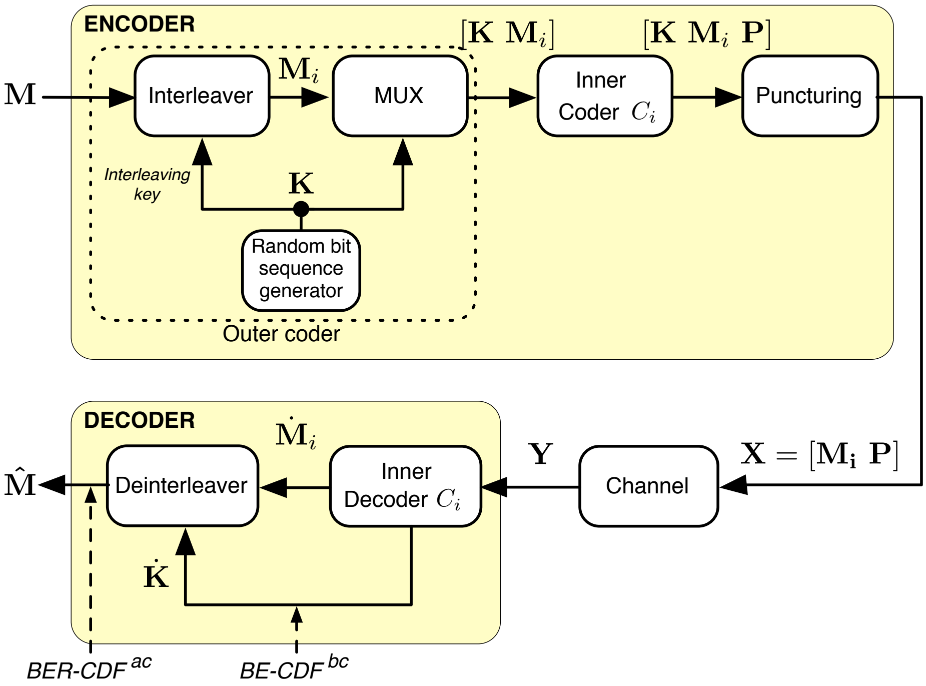

Figure 5.2: Encoder and decoder processes of the described scheme. Note that the bits of K at the output of the inner coder are not transmitted onto the channel. P refers to the parity bits of the codeword at the output of C_i.

Due to the non transmission of the key word, the code C_i must be adequately powerful and have enough parity bits to allow a legitimate/unintended receiver to be able/unable to obtain an errorless key, depending on the SNR.

An analogy that can be done is that this scheme corresponds to the one from section 4.1, when both Bob and Eve are affected by jamming interference of infinite power during the transmission of the bits corresponding to the key.

On the following section we will perform a concise analysis on this scheme and identify the advantage in terms of SNR Bob needs to possess over Eve, as well as threshold operation points, for some chosen inner code.

5.1.1 Performance analysis

On a similar fashion to the decision taken on section 4.2, on which codes to use for simulations, we will consider as example a LDPC$(1536, 1280)$ for the inner code C_i, even though there are many other options that could still be considered without loss of generalization. Once again transmission will be over a AWGN channel using BPSK modulation.

The BE-CDF^{bc} applied onto the bits of K at the output of the inner decoder will help us identify regions of operation for Bob and Eve as a function of the SNR and the size of the key (k). Note that no code is used on the key prior to the inner encoding, which means the t parameter from definition 1 is equal to zero for this case, therefore, figure 5.3 represents $\Pr(E = 0)$, where E is the number of errors on the key after inner decoding.

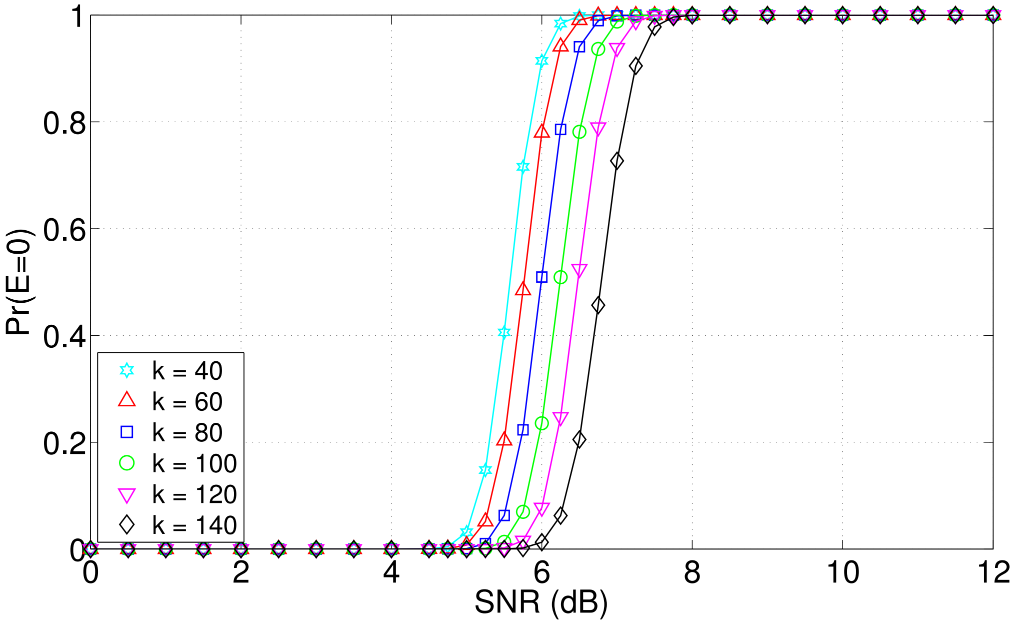

Figure 5.3: BE-CDF^{bc} for the first k bits coded by a LDPC$(1536, 1280)$. These bits are not transmitted. The remainder of the codeword is transmitted onto a AWGN channel using BPSK modulation.

Examining figure 5.3 it's easy to identify, for each curve, the SNR region that displays a probability close to 1 of obtaining a key with errors, and the one where it is likely to get an errorless key. These will be the regions of operation for Eve and Bob, respectively. The gap (≈ 2.5dB) between the thresholds of these regions of SNR corresponds to rough estimation on the minimum advantage Bob has to possess over Eve in terms of channel quality, not varying much with the key size.

On figure 5.4 an example of a distribution of number of errors on the key, considering the case with $k = 120$, for a value of SNR close to Bob's threshold operation point is depicted. The probability of obtaining a key with a short amount of errors (e.g. $0 < E \leq 10$) is negligible. This is the reason why an ECC isn't applied on the key before the concatenation with the interleaved message, contrary to what was done on the scheme discussed on Chapter 4, as the usage of this code wouldn't provide significant improvements, but on the other hand would put constraints on the size of the key.

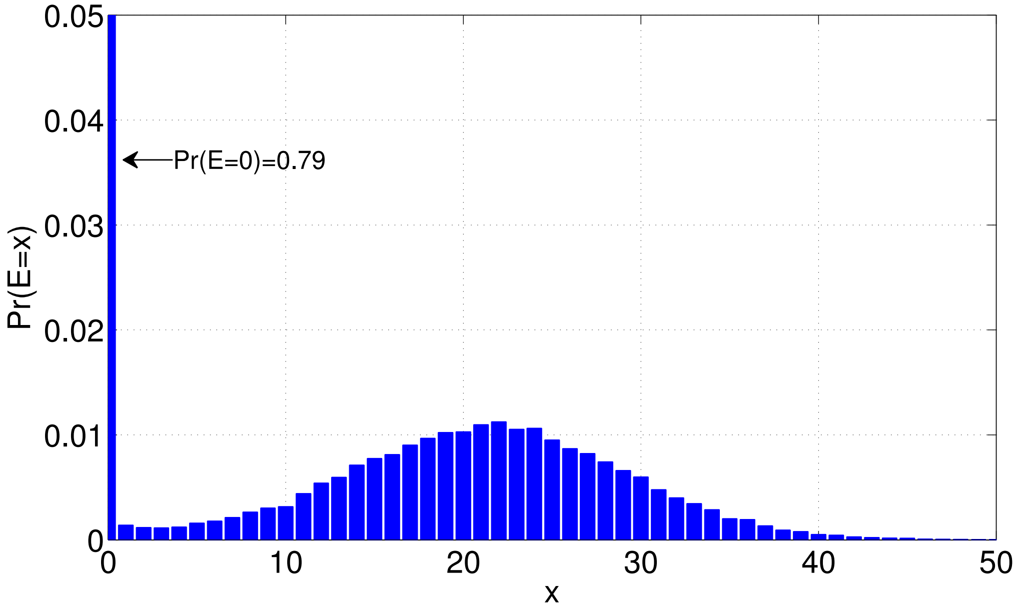

Figure 5.4: PMF of the number of errors on the key K (considering $k = 120$) after soft inner decoding of a LDPC$(1536, 1280)$. These bits are not transmitted. The remainder of the codeword is transmitted onto an AWGN channel using BPSK modulation, with $\text{SNR} = 6.75$ dB.

The BE-CDF^{bc} gave us an idea on the advantage in terms of SNR Bob needs to have over Eve. In order to get a more precise value for this gap of SNR and have more closure on the security brought by this scheme, we'll analyze the BE-CDF^{bc} and BER, depicted on figure 5.5. As example we picked the cases in which the key is composed of 60 and 100 bits.

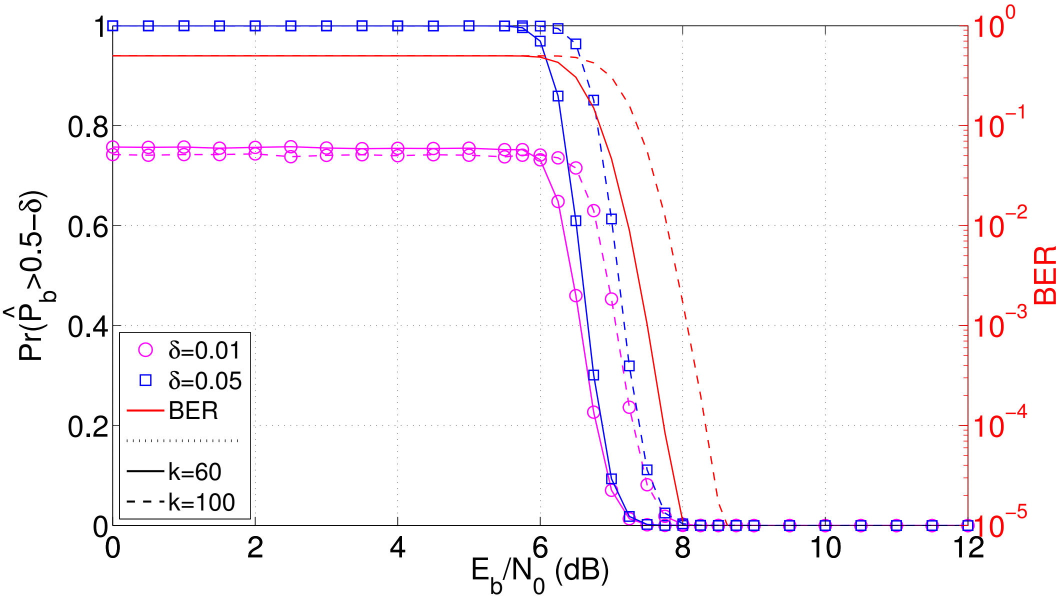

Figure 5.5: BER-CDF^{ac} and BER for the coding for secrecy scheme presented on section 5.1, when the inner code is a LDPC$(1536, 1280)$ and using a key with k bits.

Considering the transmission secure if the eavesdropper's decoding generates a $\Pr(\hat{P}_b > 0.45) \geq 0.999$, the security restriction would be fulfilled if Eve operates at $E_b/N_0 \leq 5.5$ dB, for $k = 60$, and at $E_b/N_0 \leq 6$ dB, for $k = 100$. These threshold values of E_b/N_0 are consistent with the values of SNR given by the BE-CDF^{bc} (see figure 5.3) that grant a $\Pr(E = 0) \leq 0.001$, after making the necessary units conversion[1]. If we recognize the transmission as reliable if the BER over the message bits is below 10^{-5}, Bob would have to operate at $E_b/N_0 \geq 8$ and $E_b/N_0 \geq 8.6$ for a key of size 60 and 100, respectively. From here on we'll refer to S_T^{Bob} and S_T^{Eve} as the threshold values of SNR that limit the regions of operation of Bob and Eve, respectively.

With Bob and Eve operating at S_T^{Bob} and S_T^{Eve}, respectively, of the previously defined operating regions, the advantage of E_b/N_0 (or SNR) Bob needs over Eve for assuring reliability and security is 2.5 dB for $k = 60$ and 2.6 dB for $k = 100$. The similarity of these values is interesting, because it introduces the notion of selecting the most appropriated key size (i.e. the one that assures reliability and has the highest possible value of S_T^{Eve}), when applying this security scheme to a scenario where Bob's expected SNR is characterized.

Now that we have an idea on how this scheme performs, we'll move on to the main focus of this chapter, which is the introduced concept of using a coding scheme to emulate a discrete memoryless channel, more specifically, a BSC.

[1] $(E_b/N_0)_{dB} = SNR_{dB} - 10 \times \log_{10}[m/(n-k)]$. For the considered example with LDPC$(1536, 1280)$, $n = 1536$ and $m = 1280 - k$.

5.2 Using the Scheme to Emulate a BSC

Previously on this chapter we mentioned the existence of code constructions that fulfill information-theoretic restrictions [3] [4] [5] [6]. These code constructions often require that the legitimate user receives from a perfect channel, and the eavesdropper from a discrete memoryless channel. We'll now make the necessary deliberations for showing that the previously introduced scheme can emulate this situation, when Bob and Eve's receive through a AWGN channel with a SNR greater than S_T^{Bob} and less than S_T^{Eve}, respectively, as illustrated on figure 5.6.

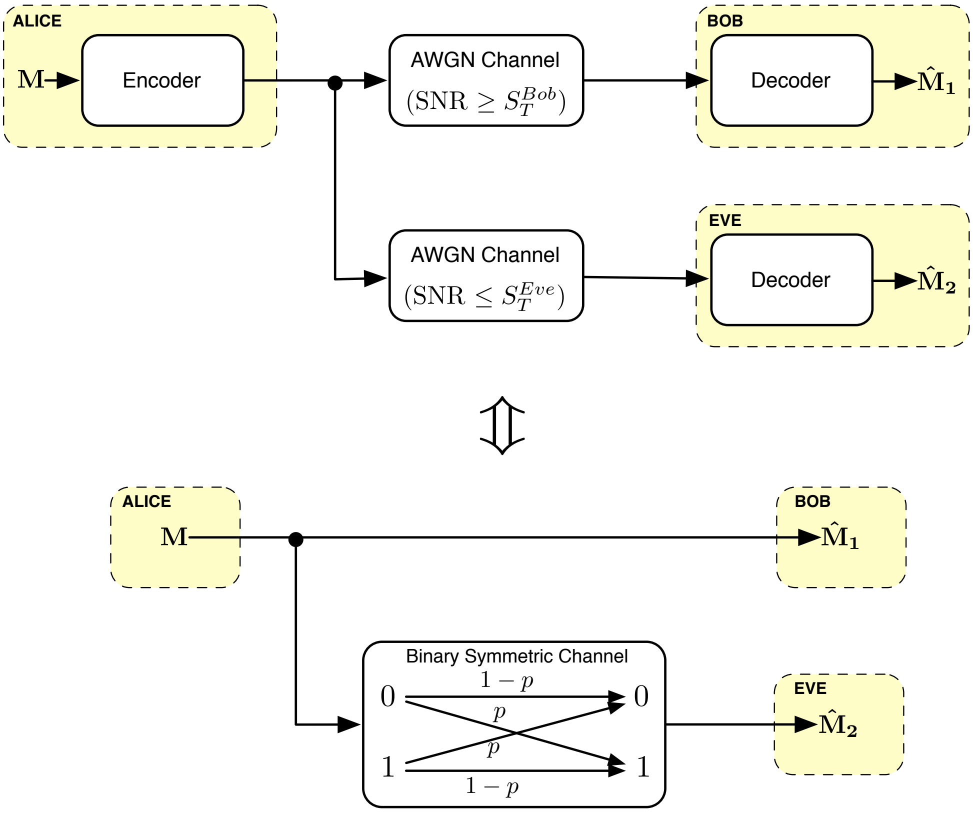

Figure 5.6: Using the encoder and decoder from figure 5.2 a perfect channel is emulated for Bob when having a AWGN channel and operating at a value of SNR $\geq S_T^{Bob}$. For Eve a BSC is emulated when having a AWGN channel and operating at a value of SNR $\leq S_T^{Eve}$.

We'll start by defining the bounds for the emulated channels. Bob's channel will be considered as perfect if the probability of having errors on the message bits after the decoding of a block is at least fewer than 10^{-4}, i.e. $1 - P(E_X = 0) \leq 10^{-4}$, where E_X represents the number or errors on the message bits. The BER-CDF^{ac} with $\delta = 0.5$ allows us to evaluate this probability, i.e. $1 - P(E_X = 0) = \Pr(\hat{P}_b > 0)$. On a similar fashion, Eve's channel will be considered as a BSC, if the probability of it possessing the properties of a

BSC, for the transmission of a block, is at least 0.9999.

The properties that need to be verified for considering that Eve's channel is modeled as an effective BSC are:

1. the probability p of flipping each bit over the channel should be identical for all bits;

2. each bit should be flipped independently from all other bits.

For the following analysis we'll return to the example from last section, i.e. using a LDPC$(1536,1280)$ as the inner code and considering the key sizes of 60 and 100. Figure 5.7 represents the BER-CDF^{ac} for this scenario as a function of the SNR. The values of $\delta = 0.1$ and $\delta = 0.5$ were chosen for evaluating Eve and Bob's performances, respectively.

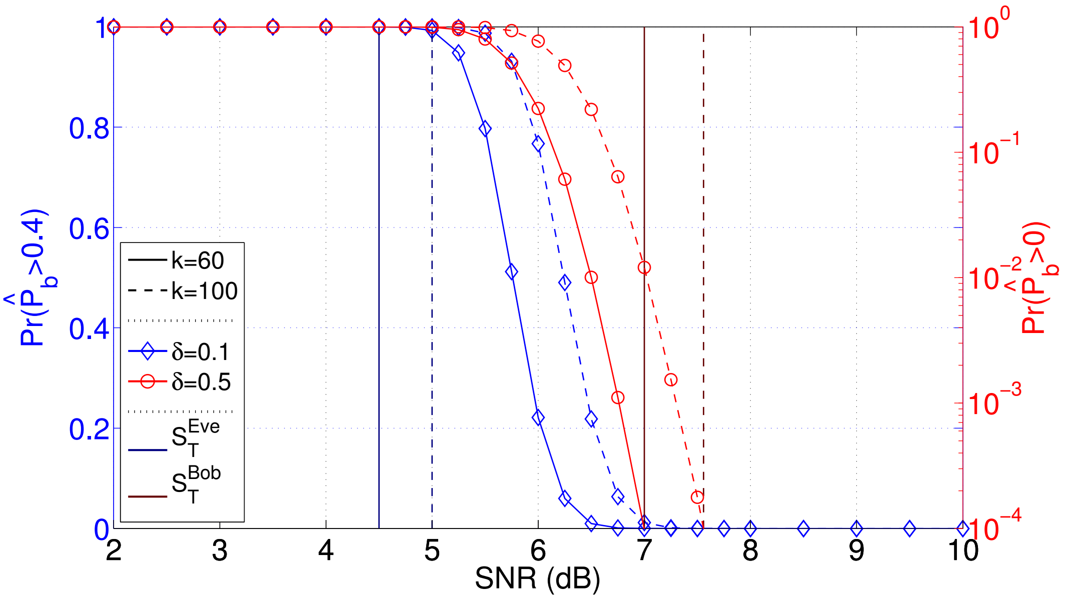

Figure 5.7: BER-CDF^{ac} for the coding for secrecy scheme presented on section 5.1, when the inner code is a LDPC$(1536,1280)$ and using a key with k bits. The thresholds values of operation for Bob and Eve are also included.

The vertical dark red lines indicate the minimum values of SNR, (7 dB for $k=60$ and 7.56 dB for $k=100$), that satisfy the previously stated requirement for considering Bob's channel as perfect. The values of S_T^{Eve}, (4.5 dB for $k=60$ and 5 dB for $k=100$), marked by the vertical dark blue lines, are the maximum SNR values that guarantee $\Pr(\hat{P}_b > 0.4) \geq 0.9999$. Recall that $\hat{P}_b$ is the proportion of estimated message bits in error over a single block of data, so this guarantee indicates that all blocks maintain at least a 40% error rate.

We'll evaluate the first property for considering that Eve's channel is a BSC, when Eve operates at SNR $\leq S_T^{Eve}$. This will be done by performing the analysis of the probability of error of each message bit for SNR $= S_T^{Eve}$. On figure 5.8, we see that for both cases, the probability of flipping each message bit over the channel is approximately identical for all message bits, with value $p \approx 0.5$. This result verifies the first stated property.

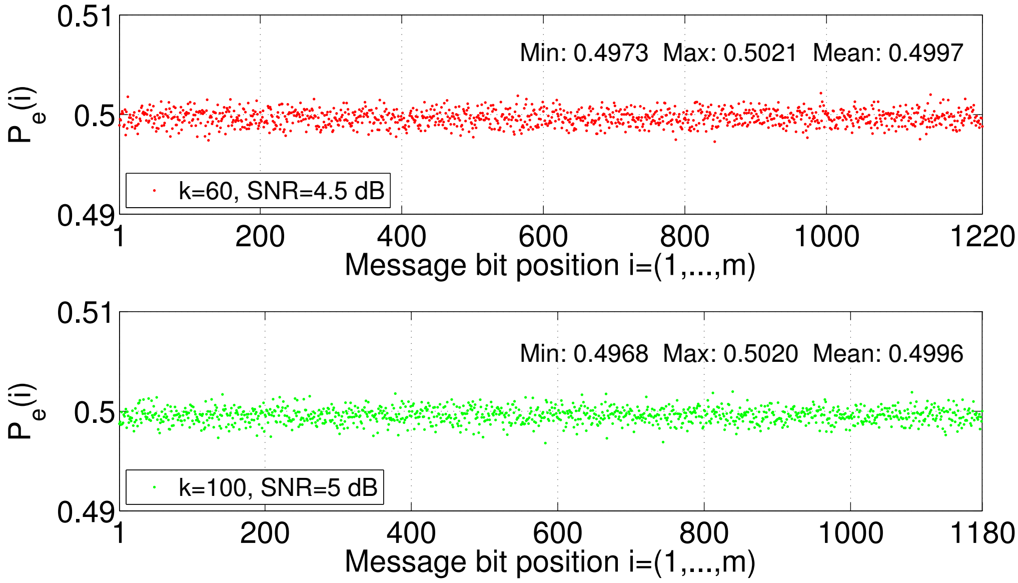

Figure 5.8: Probability of error of a bit from a decoded message, P_e, as a function of the position within the message length m. Considering a LDPC$(1536, 1280)$, $m = 1280 - k$. These values were obtained after the evaluation of 5×10^5 blocks, with random messages following an uniform distribution.

One possible way to guarantee that each bit is flipped independently from all other bits, is by using an inter-block interleaver, spreading the information around as in [28]. On the limit case, each bit from a message is transmitted on a different block, assuring independency even if there's some correlation between the bits flipped on a block. Therefore, the usage of an inter-block interleaver allows the emulation of a DMC by any coding scheme for which the BER-CDF^{ac} assures a probability close to 1 of having high error rates.

We still wish to see if the scheme may provide this property in the absence of the additional interleaver. However, evaluating if each bit is erased independently from all other bits proves to be a more difficult challenge. Let E_X be the random variable that defines the number of errors on a word of size m, received through a BSC with probability of flipping a bit P_f, then due to the errors being independent, $E_X \sim B(m, P_f)$, and the probability of having x errors on a received word is given by:

$$\Pr(E_X = x) = \binom{m}{x} {P_f}^x (1 - P_f)^{m-x}. \tag{5.1}$$

When $P_f = 0.5$, which corresponds to the value of p we identified on figure 5.8, equation

5.1 can be simplified into:

$$\Pr(E_X = x) = \binom{m}{x} 0.5^m. \tag{5.2}$$

On figures 5.9 and 5.10 the obtained through simulation PMFs that model the number of errors on the decoded message bits for the examples we are considering ($k = 60$ and $k = 100$), are depicted for comparison with the PMF[2] of X for the respective values of m and P_f. Although this comparison is not enough to claim that the second property is verified without the additional interleaver, it serves as an indicator on how the scheme from section 5.1 approaches the behavior of a BSC for the example parameters evaluated, even without the addition of an inter-block interleaver.

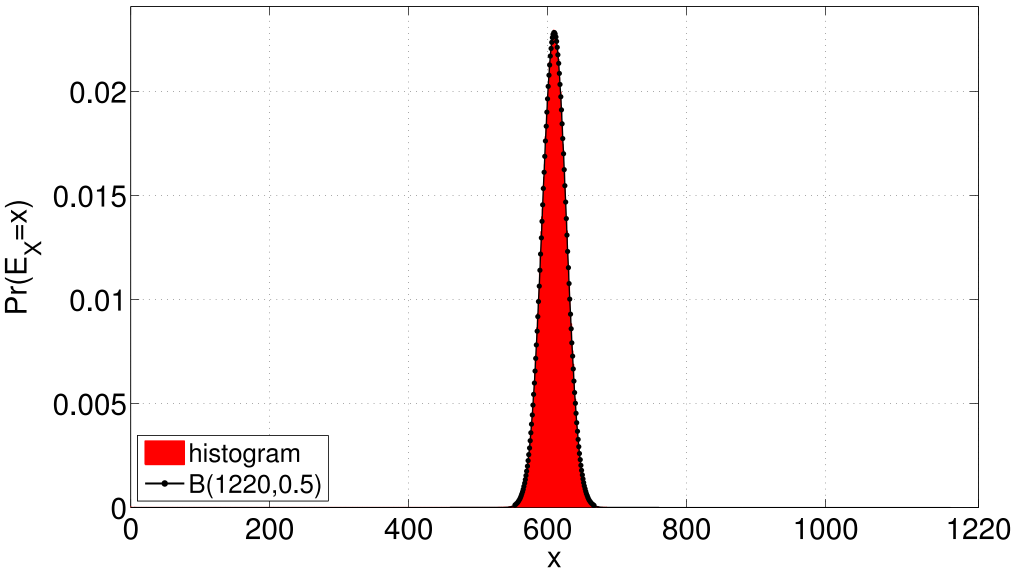

Figure 5.9: Normalized histogram that approximates the probability of having x errors on the decoded message, for the scheme presented on section 5.1 when the inner code is a LDPC$(1536, 1280)$, k=60 bits and the SNR is 4.5 dB. The curve from equation 5.2 when $m = 1220$ is shown for comparison.

[2] Due to the complexity of calculating equation 5.2 for large values of m, the curves on figures 5.9 and 5.10 were obtained by approximation to a normal distribution. The Central Limit Theorem states that for large values of m and/or P_f close to 0.5, $X \sim B(m, P_f)$ approaches $X \sim \mathcal{N}(m \times P_f,\ m \times P_f(1 - P_f))$.

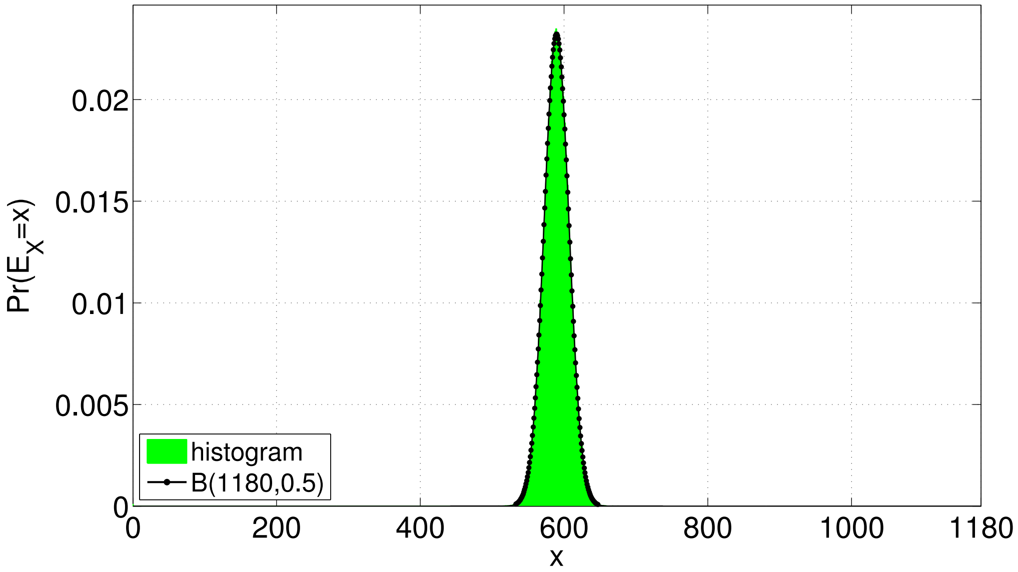

Figure 5.10: Normalized histogram that approximates the probability of having x errors on the decoded message, for the scheme presented on section 5.1 when the inner code is a LDPC$(1536, 1280)$, k=100 bits and the SNR is 5 dB. The curve from equation 5.2 when $m = 1180$ is shown for comparison.

The mutual information over a BSC with $p = 0.5$ is zero and, therefore, if the probability of a flipped bit can indeed be assumed to be 0.5, then this scheme could provide secrecy by itself. However, we feel that the proper approach to achieve secrecy in practice is to apply a wiretap code on top of the emulated BSC, while assuming the lower bound of the error rate over smaller blocks of $p = 0.5 - \delta$, ($p = 0.4$ in this case) given by the BER-CDF^{ac}. Any secrecy codes appended to our system would then be designed to provide information-theoretic security on this lower bound p value, and would thus provide it in practice on every (possibly short) secrecy codeword since we have designed for the worst case error rate over a single small block of data. We also point out that these results are more general than the specific code that has been used for example in this section, and any code that leads to similar properties (steep waterfall region) could be applied to our scheme with the accompanying analysis to identify the required SNR gap between Bob and Eve.

6

Conclusions

In this thesis a practical approach on physical-layer security was taken. We started by pointing out the limitations of the state of the art physical-layer security metrics, and in order to circumvent some of this shortcomings, we have proposed two new metrics for evaluation of schemes with finite blocklength. Through the calculation of the distribution of the number of bit errors per block, we can use CDF to provide a lower bound on the security levels based on BER. This approach retains the simplicity of calculation of the BER while providing a much stronger secrecy guarantee.

Later, on chapter 4, we used the new metrics to design and evaluate a proposed concatenated coding scheme for secrecy, that takes into account both reliability and security factors. On this scheme, an inner code is used to provide typical levels of information reliability, while security is obtained on the premise than an eavesdropper can't obtain a correct estimation of an interleaving key, that is used to shuffle the message, before being encoded with an ECC and transmitted concatenated to the interleaved information bits. Our analysis shows how the system behaves for varying parameters of SNR, additional expected interference during the transmission of key bits at the eavesdropper's channel (e.g. due to a jammer) and error correcting capability of the code applied on the key.

Finally, on chapter 5 we have proposed a coding scheme that functions on a similar fashion to the one from chapter 4, with the difference that the interleaving key is punctured before being sent through the channel, meaning that for any receiver the only information about the key is in the transmitted parity bits. The presented methodology allows us to determine the SNR advantage Bob needs to possess over Eve, as well as threshold operation points for which a reliable and secure communication is achieved. We have also outlined arguments and given evidence for the possibility of using this scheme to generate an effective DMC from a Gaussian wiretap channel. Therefore, the scheme can be concatenated with existing wiretap codes that require such a channel to provide information-theoretic security guarantees on scenarios where such guarantees were not yet achieved.

6.1 Future Work

The design of secrecy coding schemes has much room for development. For the schemes proposed on this dissertation, other key based methods for hiding the information could be employed. For example, the interleaving could be substituted by a scrambler for which the scrambling matrix is chosen randomly from a codebook indexed by the key. It also might be of interest to see how the proposed schemes behave on a real transmission. One way this can be tested is through the implementation of such schemes using a software-defined radio system.

Bibliography

[1] W. K. Harrison, J. Almeida, M. R. Bloch, S. W. McLaughlin, and J. Barros, "Coding for secrecy: An overview of error-control coding techniques for physical-layer security," *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 41–50, September 2013.

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

[3] A. Thangaraj, S. Dihidar, A. R. Calderbank, S. W. McLaughlin, and J.-M. Merolla, "Applications of LDPC codes to the wiretap channel," *IEEE Trans. Inf. Theory*, vol. 53, no. 8, pp. 2933–2945, August 2007.

[4] A. T. Suresh, A. Subramanian, and A. Thangaraj, "Strong secrecy for erasure wiretap channels," *IEEE Information Theory Workshop*, 2010.

[5] H. Mahdavifar and A. Vardy, "Achieving the secrecy capacity of wiretap channels using polar codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6428–6443, Oct. 2011.

[6] M. Bloch, M. Hayashi, and A. Thangaraj, "Error-control coding for physical-layer secrecy," *Submitted to Proc. IEEE*, pp. 1–37, 2015.

[7] D. Klinc, J. Ha, S. W. McLaughlin, J. Barros, and B.-J. Kwak, "LDPC codes for the Gaussian wiretap channel," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 532–540, Sept 2011.

[8] J. Almeida and J. Barros, "Random puncturing for secrecy," *Asilomar Conference on Signals, Systems and Computers*, 2013.

[9] M. Baldi, M. Bianchi, and F. Chiaraluce, "Coding with scrambling, concatenation, and harq for the awgn wire-tap channel: A security gap analysis," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 883–894, 2012.

[10] M. Bloch and J. Barros, *Physical Layer Security: From Information Theory to Security Engineering*. Cambridge University Press, 2011.

[11] C. E. Shannon, “The mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, 1948.

[12] T. M. Cover, *Elements of information theory*. Wiley, 1991.

[13] J. C. Moreira and P. G. Farrell, *Essentials of Error-Control Coding*. Wiley, 2006.

[14] R. C. Bose and D. K. Ray-Chaudhuri, “On a class of error correcting binary group codes,” *Information and Control*, vol. 3, no. 1, 1960.

[15] A. Hocquenghem, “Codes correcteurs d’erreurs,” *Chiffres*, vol. 2, 1959.

[16] R. G. Gallager, *Low-Density Parity-Check Codes*. MIT Press, Cambridge, 1963.

[17] *Error Control Coding*, 2nd ed. Pearson Prentice Hall, 2004.

[18] J. Chen and M. P. C. Fossorier, “Near optimum universal belief propagation based decoding of low-density parity check codes,” *IEEE Transactions on Communications*, vol. 50, no. 3, 2002.

[19] L. Nuaymi, *WiMAX: Technology for Broadband Wireless Access*. Wiley, 2007.

[20] S. S. Haykin, *Digital Communication Systems*. Wiley, 2014.

[21] R. Liu, Y. Liang, H. V. Poor, and P. Spasojevic, “Secure nested codes for type II wiretap channels,” in *IEEE Information Theory Workshop*, Lake Tahoe, CA, USA, September 2007.

[22] L. H. Ozarow and A. D. Wyner, “Wire tap channel ii,” *AT and T Bell Laboratories Technical Journal*, vol. 63, no. 10, 1984.

[23] S. Goel and R. Negi, “Guaranteeing secrecy using artificial noise,” *IEEE Transactions on Wireless Communications*, vol. 7, no. 6, pp. 2180–2189, June 2008.

[24] C. E. Shannon, “Communication theory of secrecy systems,” *Bell System Technical Journal*, vol. 28, pp. 656–715, Oct. 1949.

[25] U. M. Maurer, “The strong secret key rate of discrete random triples,” in *Communications and Cryptography: Two Sides of One Tapestry*. Kluwer Academic Publishers, 1994, pp. 271–285. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.26.1065

[26] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2005.

[27] F. Dias, “Sistemas de codificação para segurança na camada física baseados em técnicas de interleaving aleatório e jamming,” Master’s thesis, Universidade de Coimbra, 2014.

[28] Y. Liang, H. V. Poor, and L. Ying, “Secrecy throughput of MANETs under passive and active attacks,” *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6692–6702, Oct. 2011.