LDPC Codes for Physical Layer Security

Demijan Klinc*, Jeongseok Ha†, Steven W. McLaughlin*, João Barros‡, and Byung-Jae Kwak§
*School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, USA
Email: {demi, swm}@ece.gatech.edu
†School of Electrical and Computer Engineering, Information and Communications University
119, Munjiro, Daejeon, 305-732, Korea
Email: jsha@icu.ac.kr
‡Instituto de Telecomunicações, Departamento de Engenharia Electrotécnica e de Computadores
Faculdade de Engenharia da Universidade do Porto, Portugal
Email: jbarros@fe.up.pt
§Electronics and Telecommunications Research Institute
161 Gajeong-dong, Yuseong-gu Daejeon, 305-700, Korea
Email: bjkwak@etri.re.kr

Abstract—This paper presents a coding scheme for the Gaussian wiretap channel based on low-density parity-check (LDPC) codes. The messages are transmitted over punctured bits to hide data from eavesdroppers. It is shown that this method is asymptotically effective in the sense that it yields a BER very close to 0.5 for an eavesdropper whose SNR is lower than the threshold SNR_E, even if the eavesdropper has the ability to use a bitwise MAP decoder. Such codes also achieve high reliability for the friendly parties provided they have an SNR above a second threshold SNR_B. It is shown how asymptotically optimized LDPC codes can be designed with differential evolution where the goal is to achieve high reliability between friendly parties and security against a passive eavesdropper while keeping the security gap SNR_B/SNR_E as small as possible. The proposed coding scheme is applicable at finite block lengths and can be combined with existing cryptographic schemes to deliver improved data security by taking advantage of the stochastic nature of many communication channels.

I. INTRODUCTION

It was proved by Shannon in [1] that information-theoretically secure communication is possible only if the communicating parties, say Alice and Bob, share a secret key whose entropy is larger or equal to that of the message. In that case Alice and Bob can use the one-time pad scheme and any potential eavesdropper Eve who does not have access to the secret key is provably unable to extract any information about the message. Unfortunately, the one-time pad scheme only translates the problem of sharing a message to sharing a secret key. To circumvent this difficulty, a variety of cryptographic algorithms were invented that employ shorter secret keys, but rely on unproved mathematical assumptions and limited computational resources at Eve for secrecy.

Shannon’s assumption, though, was that Bob’s and Eve’s observations of the transmitted ciphertexts are identical. Quite often that assumption is not realistic due to the stochastic nature of many communication channels. A few decades after Shannon’s work it was shown in [2]–[5] that information theoretically secure communication is possible exclusively by means of coding at the physical layer if Eve has a worse channel than Bob.

Equivocation at Eve, which is an established metric for information theoretic security, is difficult to measure or analyze on noisy coded sequences, especially at finite block lengths. That may be one of the main reasons why no practical code constructions at finite block lengths for secure communication exist at this point. To get around his problem, the bit-error-rate (BER) over message bits, which is much easier to analyze and measure, is used as a measure for security in this paper. For example, if Eve observes data through a channel with BER close to 0.5 (the errors are IID), then she would be able to extract little information about the message. It should be noted at the outset that BER is a different metric than the equivocation, therefore this paper does not address information theoretic security, but rather physical layer security. Nevertheless, it is argued that a high BER at Eve is useful and can, possibly in conjunction with standard cryptographic techniques, deliver improved resilience against eavesdropping.

Consider the Gaussian wiretap model depicted in Figure 1. Alice wants to transmit an s-bit message $M^s$ to Bob. She uses an error-correcting code to encode $M^s$ to an n-bit codeword $X^n$ and transmits it over an AWGN channel to Bob. Eve listens to the transmission over a noisier, independent AWGN channel and tries to reconstruct the message $M^s$. She is
assumed to be passive, hence she is not allowed to transmit data, so as to jam or interfere in the communication between Bob and Eve. Let an average bit-error-rate (BER) over the Bob’s estimate $M_B^e$ be $P_e^B$ and let an average bit-error-rate over the Eve’s estimate $M_E^e$ be $P_e^E$. It is desired that $P_e^B$ be sufficiently low to ensure reliability and that $P_e^E$ be high. If $P_e^E$ is close to 0.5 and the errors are IID, then Eve will not be able to extract much information from the received sequence $Z^n$. Thus, for fixed $P_{e,max}^B(\approx 0)$ and $P_{e,min}^E(\approx 0.5)$, it must hold that

a) $P_e^B \leq P_{e,max}^B$ (reliability),
b) $P_e^E \geq P_{e,min}^E$ (security).

Let $SNR_{B,min}$, also called the threshold, be the lowest SNR for which a) holds and let $SNR_{E,max}$ be the highest SNR for which b) holds. It is assumed that Bob operates at $SNR_{B,min}$ and that Eve’s SNR is strictly lower than $SNR_{B,min}$. The security gap is defined as $SNR_{B,min}/SNR_{E,max}$ and can alternatively be expressed in dB. Thus, the size of the security gap in dB (see Figure 2) is the minimum required difference between Bob and Eve’s SNRs for which secure communication in our context is possible. Conventional error-correcting codes require large (> 20 dB) security gaps when $P_{e,min}^E > 0.4$. The focus of this paper is to design a coding scheme that exhibits a small security gap.

In the proposed coding scheme, messages are transmitted over punctured [8]. A puncturing distribution in this form is useful for an asymptotic analysis of message passing decoders. Bob and Eve are assumed to use the belief propagation decoder, which is asymptotically equal to the bitwise maximum a-posteriori (MAP) decoder and hence very powerful. It will be shown that transmitting messages over punctured bits can significantly reduce security gaps and can thus be efficiently used for increased security of data. Security gaps as low as few dB are sufficient to force Eve to operate at BER above 0.49. The suggested coding scheme is proposed to be employed in conjunction with existing cryptographic schemes which operate on higher layers of the protocol stack.

The outline of the paper is the following. Section II introduces the relevant definitions and notation. Section III discusses the design of secure LDPC codes both in the asymptotic case and at finite block lengths. For the asymptotic case it is shown how the BER over punctured message bits can be evaluated analytically. Using this result it is shown that transmitting messages over punctured bits significantly reduces security gaps. Further, it is shown how puncturing distributions that deliver minimized security gaps can be obtained by means of nonlinear optimization. Some results on secure LDPC codes at finite block lengths are presented. Finally, Section IV discusses the integration of the proposed coding scheme at the physical layer with existing cryptographic schemes at higher layers for increased security.

II. PRELIMINARIES

In this paper, punctured binary low-density parity-check codes are used as the coding scheme for improved data security. This section introduces the notation and some definitions that will be used in the remaining sections.

LDPC codes were introduced in [6]. They have been shown to reach or perform close to capacity over many channels [7]. An LDPC code can be specified by means of a bipartite graph, composed of variable nodes representing codeword bits and check nodes representing the constraints imposed on the codeword bits. An important parameter that describes a bipartite graph of an LDPC code is the degree distribution, which is given in the form of two polynomials $\lambda(x) = \sum_{d=2}^{d_v-x} \lambda_i x^{d_i-1}$ and $\rho(x) = \sum_{d=2}^{d_c} \rho_i x^{d_i-1}$. The values $d_v$ and $d_c$ represent the maximum variable and check node degrees, while $\lambda_i$ and $\rho_i$ denote the fractions of edges connected to variable and check nodes of degree $i$, respectively. From the node perspective, the fraction of variable nodes of degree $i$ is denoted by $\lambda_i$ and $\lambda_i = (\lambda_i/i)/\sum_{i=2}^{d_v} \lambda_i/i$.

An LDPC code can be punctured, which effectively means that some of its variable nodes are not transmitted. One way of describing how an LDPC is punctured is by means of a puncturing distribution $\pi(x) = \sum_{i=2}^{d_v-x} \pi_i x^{i-1}$, where $\pi_i$ denotes the fraction of variable nodes of degree $i$ that are punctured [8]. A puncturing distribution in this form is useful for an asymptotic analysis of punctured LDPC codes. Let $p$ denote the fraction of all punctured bits, so that $p = \sum_{i=2}^{d_v} \lambda'_i \pi_i$.

In the proposed coding scheme, messages are transmitted over punctured bits. The puncturing pattern must be such that no subset of punctured bits forms a stopping set, otherwise some
punctured (message) bits would not be recoverable in the decoder.

Let the dimension of an LDPC code be \(d\), let the number of message bits be \(s\), and let the number of transmitted codeword bits be \(n\). The design rate is defined as

\[
R_d = \frac{d}{n},
\]

while the secrecy rate is defined as

\[
R_s = \frac{s}{n}.
\]

Note that it is possible that the number of punctured (message) bits is smaller than \(d\). In such cases, punctured message bits are coupled with some randomly chosen dummy bits to occupy all independent bit locations in a codeword. Usually in such cases \(R_s < R_d\). On the other hand if a code is left unpunctured and assuming that all independent bit locations carry messages then \(R_s = R_d\).

There exist some powerful methods for the analysis of LDPC decoders. Most notably, the method of density evolution, developed in [9], tracks the evolution of the probability density function (PDF) of messages as they are passed between variable and check nodes during the decoding process. Density evolution is computationally demanding, but [10] showed that it can be simplified significantly if the messages are assumed to have Gaussian PDFs. Namely, in that case the density evolution can be reduced to tracking only one parameter: the average BER over punctured nodes. For the analysis of \(m_{u}^{(k)}\) it is convenient to define the following function

\[
\phi(x) = \left\{ \begin{array}{ll} 1 & \text{if } x > 0 \\ \frac{1}{\sqrt{4\pi x}} \int_{x}^{\infty} e^{-t^2/4} dt & \text{if } x \leq 0 \end{array} \right.
\]

which is used in the next section. Further details can be found in [10].

### III. Coding for Security

The use of puncturing for improved data security is investigated in this section. First, it is shown by means of asymptotic analysis that the security gap can be significantly reduced if messages are transmitted over punctured bits. Subsequently, some results for finite block lengths are presented.

#### A. Asymptotic analysis

Framework for the analysis of punctured binary LDPC codes over the AWGN channel was developed in [8]. If messages in the belief propagation decoder are assumed to be Gaussian, the mean value of check-to-variable node messages in the \(k\)-th iteration was shown to be

\[
m_u^{(k)} = \sum_{s=2}^{d} \rho_s \phi^{-1}
\left( 1 - \frac{1}{(1 - e^{(k)})^{s-1}} \right) \left[ 1 - \sum_{j=2}^{d_s} \left( \lambda_j \pi_j \sum_{i=0}^{j-1} \chi_i^{(k)} \phi \left( im_u^{(k-1)} \right) + \lambda_j (1 - \pi_j) \sum_{i=0}^{j-1} \chi_i^{(k)} \phi \left( im_u^{(k-1)} + m_{u_0} \right) \right) \right]^{-1},
\]

where \(n_k = \binom{n}{m} (1 - e^{(k)})^n - m (1 - e^{(k-1)})^m\), \(e^{(k)} = 1 - \rho (1 - e^{(k-1)})\), \(e^{(0)} = \sum_{i=2}^{d_s} \lambda_i \pi_i\), \(m_{u_0} = 2/\sigma^2\), and \(\sigma^2\) is noise variance.

Using \(m_{u}^{(k)}\), the BER over all variable nodes in the \(k\)-th iteration is

\[
P_c^{(k)} = P_{e_1}^{(k)} + P_{e_2}^{(k)}
\]

\[
= \sum_{j=2}^{d_s} \chi_j^{(k)} \sum_{i=0}^{j} \phi \left( im_u^{(k)} \right) + \sum_{j=2}^{d_s} \chi_j^{(k)} \sum_{i=0}^{j} \phi \left( im_u^{(k)} + m_{u_0} \right).
\]

The first term, \(P_{e_1}^{(k)}\), is the contribution from the punctured variable nodes, while the second, \(P_{e_2}^{(k)}\), is the contribution from the unpunctured variable nodes. Thus, if the fraction of punctured nodes is \(p\) the average BER over punctured nodes is

\[
P_{e,p}^{(k)} = \frac{1}{p} P_{e_1}^{(k)}
\]

and can be computed at arbitrary SNRs, both above and below the threshold.

Intuitively, the BER over message bits is expected to be higher if the message bits are punctured than if they are transmitted. This assumption is tested on the following example. An LDPC mother code with degree distribution

\[
\begin{align*}
\lambda(x) &= 0.25105x + 0.30938x^2 + 0.00104x^3 + 0.43853x^9, \\
\rho(x) &= 0.63676x^6 + 0.36324x^7
\end{align*}
\]

is chosen and punctured randomly according to

\[
\pi(x) = 0.4x + 0.4x^2 + 0.4x^3 + 0.4x^9.
\]

Since the fraction of punctured bits \(p\) equals 0.4 and all punctured bits are assumed to carry messages, the secrecy rate according to (2) amounts to \(p/(1-p) = 2/3\). Notice that \(s < d\), therefore some variable nodes must be set by random dummy bits in the encoder.

For comparison, an LDPC code of design rate 2/3 is chosen, where messages are transmitted over the channel along with
parities. Its degree distribution is
\[
\begin{align*}
\lambda(x) &= 0.17599x + 0.40223x^2 + 0.42178x^9, \\
\rho(x) &= 0.61540x^{10} + 0.38460x^{11}.
\end{align*}
\]
Note that for this code \( R_s = R_d = 2/3 \). The results for the BER over message bits for these two codes are shown in Figure 3.

It should be emphasized that both Bob and Eve are assumed to have perfect knowledge of \( \pi(x) \) and the exact puncturing locations within a codeword. Bob is operating at threshold \( \text{SNR}_{B,\text{th}} \), therefore his BER can be arbitrarily low. Further, Eve’s SNR is assumed to be lower than \( \text{SNR}_{B,\text{th}} \). Of interest is Eve’s BER in terms of her gap to threshold \( \text{SNR}_{B,\text{th}} \). Notice that Eve’s BER increases much faster as the gap between Bob and Eve grows, when the message bits are punctured. For instance, for \( P_{e,\text{min}} \) set at 0.40, 0.45 and 0.49, the security gaps amount to 2.5 dB, 4 dB and 8 dB, respectively. In contrast, if the message bits are transmitted over the channel, the security gaps are considerably larger at 15 dB, 21 dB and 35 dB, respectively. These results indicate the benefit of protecting the message bits by means of puncturing. Further, they indicate that relatively high BER at Eve are attainable for small security gaps. It should be noted that asymptotically, belief propagation equals bitwise MAP decoding. Thus, even if Eve has the capability of using a bitwise MAP decoder, her BER approaches 0.5 fast if her channel is worse than Bob’s.

The increased security is leveraged at the expense of increased transmit power. Given that the punctured LDPC code operates at a secrecy rate which is lower than the design rate, Bob requires a better signal to receive the data reliably than he would have with the unpunctured code. In the example from Figure 3, the threshold \( \text{SNR}_{B,\text{th}} \) for the punctured code is 2.28 dB, whereas it is \(-0.48\) dB for the unpunctured code. Thus, the power requirement is increased by 2.76 dB.

The puncturing distribution in (9) was random. It is natural to ask if smaller security gaps can be attained by using puncturing distributions that are specifically designed to improve security. With the above described framework such an optimization of puncturing distributions can be set up as follows. Let \( \text{SNR}_E \) be the maximum SNR for which \( P_{e,k}^{(i)} \geq P_E^{(i)} \) for any \( i \). Let \( \text{SNR}_B \) be the threshold of the punctured code. Then, for a given degree distribution \( \langle \lambda(x), \rho(x) \rangle \) and puncturing fraction \( p \), the problem at hand is:

subject to
\[
\begin{align*}
a) &\quad 0 \leq \pi_i \leq 1 \quad \text{for all} \ 2 \leq i \leq d_v, \\
b) &\quad \sum_{i=2}^{d_v} \lambda_i \pi_i = p, \\
c) &\quad \text{SNR}_B < \infty, \\
d) &\quad \text{SNR}_E > -\infty.
\end{align*}
\]

Differential Evolution [11] was used to optimize puncturing distributions for a mother code with the degree distribution \((7,8)\), puncturing fractions \( p = 0.1, 0.2, 0.3, 0.4 \) and \( P_{e,\text{min}} = 0.49 \). The optimized puncturing distributions for security are given in Table I and their performance is shown in Figure 4, where a comparison is drawn with random puncturing.

| Table I: Optimized puncturing distributions for security obtained by means of Differential Evolution. The parameter \( P_{e,\text{min}} \) was set to 0.49. |
|------------------|---|---|---|---|
| \( p \)         | 0.10 | 0.20 | 0.30 | 0.40 |
| \( R_s \)       | 0.1111 | 0.2500 | 0.4286 | 0.6667 |
| \( \pi_2 \)     | 0.1294 | 0.2959 | 0.4552 | 0.5326 |
| \( \pi_4 \)     | 0.0479 | 0.0120 | 0.0175 | 0.2066 |
| \( \pi_8 \)     | 0.0901 | 0.3801 | 0.4597 | 0.4896 |
| \( \pi_{10} \)  | 0.1336 | 0.3907 | 0.5196 | 0.4886 |
| security gap (dB) | 5.59 | 5.75 | 6.16 | 7.39 |

The benefit of using optimized puncturing distributions for security is most pronounced at high secrecy rates, where the puncturing fractions are high. The gains over random puncturing can be close to 0.5 dB, which is a notable improvement at asymptotic block lengths.

Another important remark is that the proposed method works only if Eve’s SNR is lower than Bob’s, which can be restrictive. It is interesting to study the opposite case, when Eve has a better SNR than Bob. It has been shown [12], [13] that even then, physical layer security is attainable by introducing a feedback channel between Alice and Bob (also accessible to Eve). However, this problem falls outside of the scope of this paper and shall be studied in the future.

B. Finite block lengths

The performance of codes from Figure 3 is revisited at finite block lengths and random puncturing. The results are presented in Figure 5, where the number of message bits is 1576, the number of transmitted bits in each block is 2364
and $P_{e,\text{max}}$ is set to $10^{-5}$. With random puncturing, security gaps as low as few dB can be attained for $P_{e,\text{min}} = 0.4$. Note that at finite block lengths Tanner graphs of LDPC codes usually have cycles, therefore belief propagation decoding does not yield exact bitwise MAP probabilities for codeword bits. Nevertheless, the belief propagation was shown to exhibit excellent performance.

Security gap and secrecy rate can be controlled by varying the number of message bits that are communicated per transmitted block. When selecting the locations of message bits priority is given to punctured variable nodes that require more iterations to recover, such that the security gap is reduced. Figure 6 shows the results.

Note that the puncturing pattern remains unchanged at all secrecy rates. A reduction in secrecy rate is achieved by placing dummy bits on punctured variable nodes that do not carry messages. The dummy bits can be chosen randomly in the encoder. They, along with the set of transmitted dummy bits, make the encoding process stochastic, as one set of message bits can be encoded to multiple different codewords.

In a realistic scenario the transmission power would ideally have to be controlled by Alice such that Bob operates close to $\text{SNR}_{B,\text{min}}$. Namely, notice that all measurements so far assume that Bob operates at $\text{SNR}_{B,\text{min}}$. In case his SNR is larger, the required gap for Eve grows equally larger as well, which is undesirable.

IV. SYSTEM ASPECTS

The proposed coding scheme operates at the physical layer of the protocol stack and can be viewed as a first step towards a somewhat non-conventional bottom-up architecture for secure communications. In contrast with the traditional cryptographic approach, in which the role of the physical layer is merely to provide the higher layers with a virtually error-free channel abstraction, it is argued that stronger levels of secrecy can be achieved by re-designing the channel coding modules under the aforementioned security metrics. The basic underlying intuition is that a noisy cipher text is more difficult to break than its error-free counterpart. At a bit error rate close to 0.5 there is little correlation left between the signals observed by Eve and the original cryptogram. To break the cipher, she would have to guess both the key and the random error sequence introduced by the channel, which leads to a significant increase in the search space she is forced to deal with while performing cryptanalysis.

Ultimately, a wireless link should be as difficult to eavesdrop as an Ethernet cable, in the sense that the attacker would have to gain physical access to the channel at very close proximity to be able to acquire information bearing signals. Even if this threshold is exceeded, the attacker would still have to break hard cryptographic primitives. In other words, the proposed coding scheme does not replace cryptography, yet it adds one more layer of protection that is targeted at the lower and
arguably most vulnerable stage of wireless devices.

While wiretap codes, whose practical construction is still elusive for most cases of interest, would use part of the rate to confuse the eavesdropper and to achieve information-theoretic security (at least asymptotically), the proposed LDPC codes can be readily implemented to induce higher bit error rate at the eavesdropper with a controlled reduction of the rate at which Alice communicates with Bob.

Alternatively, one could argue that cryptographic primitives such as the Advanced Encryption Standard (AES) are designed for the worst case scenario in which the eavesdropper acquires an error-free cryptogram. In applications, such as RFID systems and wireless sensor networks, where strong ciphers like AES are too costly from the point of view of computational complexity, the proposed codes for security can be combined with lightweight cryptographic primitives, while still ensuring sufficient levels of confidentiality. Thus, joint design of channel codes and cryptographic primitives emerges as a promising line of research [14], [15].

V. CONCLUSION

This paper addresses the design of practical error-correction codes for physical layer security. While most work on the topic considers equivocation at Eve as the security metric, this paper proposes the BER over message bits, as it is more amenable to analysis and measurement. Consequently is allows for an easier transition from asymptotical results to practical code constructions at finite-block lengths.

In particular, Gaussian wiretap channel was considered where Eve is assumed to (i) have a lower SNR than Bob and (ii) be passive. A practical code construction based on LDPC codes is proposed, where the messages are transmitted over punctured bits. The analysis of the decoder performance by means of the Gaussian Approximation shows that Eve is forced to operate at BERs very close to 0.5 as soon as her SNR is only a few dB lower than Bob’s, even if she has the capability of using a bitwise MAP decoder. It is shown how puncturing distributions can be optimized for security. The proposed method can naturally be extended to finite-length code constructions.

While beneficial in terms of security, the proposed code construction has its limitations. The benefit in security is yielded at the expense of increased transmission power and the method works only if Eve’s SNR is lower than Bob’s. Code constructions that do not incur power losses and solutions when Eve has a higher SNR than Bob remain to be studied in the future.

REFERENCES