

LDPC Codes for the Gaussian Wiretap Channel

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

†Department of Electrical Engineering

Korea Advanced Institute of Science and Technology (KAIST)

Munji-dong Yuseong-gu, Daejeon, Korea Email: jsha@ee.kaist.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—A coding scheme for the Gaussian wiretap channel based on low-density parity-check (LDPC) codes is presented¹. The messages are transmitted over punctured bits to hide data from eavesdroppers. It is shown by means of density evolution that the BER of an eavesdropper, who operates below the code’s SNR threshold and has the ability to use a bitwise MAP decoder, increases to 0.5 within a few dB. 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* as small as possible. The proposed coding scheme is also efficiently encodable in almost linear time.

I. INTRODUCTION

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 . Let an average bit-error-rate (BER) over the Bob’s estimate \hat{M}_B^s be P_e^B and let an average bit-error-rate over the Eve’s estimate \hat{M}_E^s 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 a fixed $P_{e,\min}^E (\approx 0.5)$ and any $\epsilon > 0$ it must hold that

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

Let $\text{SNR}_{B,\min}$, also called the threshold, be the lowest SNR for which a) holds and let $\text{SNR}_{E,\max}$ and be the highest SNR for which b) holds. Throughout this paper it is assumed that Bob operates at $\text{SNR}_{B,\min}$, while Eve’s SNR is always lower than $\text{SNR}_{B,\min}$. The security gap is defined as $\text{SNR}_{B,\min}/\text{SNR}_{E,\max}$ and can alternatively be expressed in dB.

¹This work was partly supported by the IT R&D program of MKE/IITA. [2008-F-002-01, Development of original technology for next-generation Tactical Defense Communication Network]

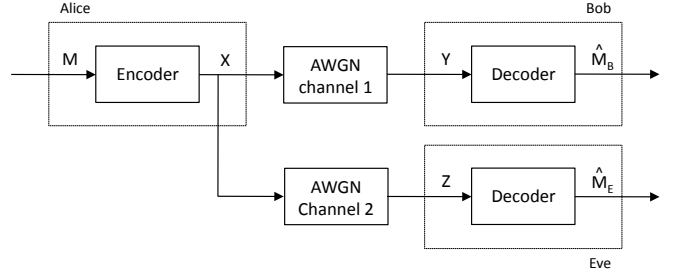


Fig. 1. The Gaussian wiretap channel.

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.

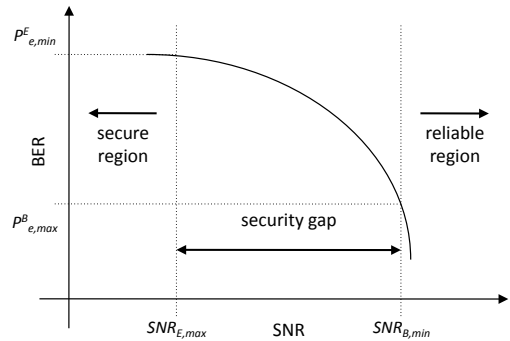


Fig. 2. The security gap. This curve shows the typical BER vs. SNR performance of an error correction code. $\text{SNR}_{B,\min}$ defines the lower end of the reliable region, whereas $\text{SNR}_{E,\max}$ defines the upper end of the secure region. The security gap (in dB) is the SNR difference between $\text{SNR}_{B,\min} - \text{SNR}_{E,\max}$ that must be maintained between Bob and Eve in order to achieve both reliability and security constraints in conditions a) and b).

This problem is related to the wiretap channel problem

introduced in [1], [2]. The idea is to exploit randomness introduced by communication channels to leverage improved data security by means of coding at the physical layer. One notable difference, though, is in the choice of the security metric. Equivocation at Eve, the metric in [1], [2], is well established, but hard to measure or analyze at practical block lengths. This is possibly the main obstacle toward practical code constructions for the wiretap channel problem. On the other hand, this paper considers BER over message bits as the metric for secrecy, as BER is considerably easier to analyze than the equivocation. For example, if Eve's BER after decoding is 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.

The main idea proposed in this paper is to hide data from Eve by means of puncturing. Instead of transmitting message bits, they are punctured in the encoder and must be deduced from the channel observations of the transmitted bits at the decoder. If the receiver (Eve) operates below the threshold $\text{SNR}_{B,\min}$, the channel observations are expected to be very noisy, therefore the reconstruction of punctured message bits is expected to be hard.

LDPC codes are chosen as the coding scheme for two reasons: (i) their excellent error-correcting performance and (ii) availability of powerful tools for the asymptotical 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 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 as follows. Section II introduces the relevant definitions and notation. Section III details the BER analysis over message bits by means of density evolution and demonstrates the benefit of hiding messages by means of puncturing. Some optimized LDPC code constructions for security are designed and compared with random LDPC code constructions. Section IV shows that the proposed constructions are efficiently encodable.

II. PRELIMINARIES

An LDPC code [3], [4] is 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. Its degree distribution, which is given in the form of two polynomials $\lambda(x) = \sum_i \lambda_i x^{i-1}$ and

$\rho(x) = \sum_i \rho_i x^{i-1}$. The coefficients λ_i and ρ_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 Λ_i and $\Lambda_i = (\lambda_i/i)/(\sum_i \lambda_i/i)$.

A puncturing distribution $\pi(x) = \sum_i \pi_i x^{i-1}$, where π_i denotes the fraction of variable nodes of degree i that are punctured [5]. 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 $p = \sum_{i=2}^{d_v} \Lambda_i \pi_i$.

Let s be the number of message bits, let d be the dimension of the LDPC code, and let n be the number of bits transmitted over the channel. Define the secrecy rate as $R_s = s/n$ and the design rate as $R_d = d/n$. Usually, the number of message bits s is equal to the dimension of the error-correction code d and thus $R_s = R_d$. However, in this paper all messages are transmitted exclusively over the punctured bits and since it is possible that the number of punctured bits is less than d , it may occur that $R_s < R_d$. In such cases, the unpunctured independent bit locations of a codeword are set randomly by dummy bits.

III. ANALYSIS OF BER OVER PUNCTURED BITS

The main idea in this paper is to hide information bits from the eavesdropper by means of puncturing. The objective in this section is to provide a better sense of the level of security that the proposed method delivers by analyzing the BER over (punctured) messages bits. In the following, it is assumed that the decoding algorithm is belief propagation and the analysis is asymptotic, where belief propagation decoding is equivalent to bitwise maximum a-posteriori decoding [4].

The analysis is performed with density evolution (DE) [6], which is known for its accurate analysis of the performance of the belief propagation decoder. In an earlier work [7], a similar attempt was made using the Gaussian approximation (GA), a computationally less demanding alternative to DE. While GA was shown to perform well around and beyond the threshold, the approximation errors can be considerable when decoder is operating below the threshold. To overcome this problem and for improved accuracy, the analysis is performed with DE.

The procedure of calculating the average BER performance over the punctured bits with DE is now briefly discussed. In the ℓ th iteration, the density of the message from a variable to a check node equals [6]

$$\begin{aligned} P_\ell &= \sum_{i \geq 2} ((1 - \pi_i)P_0 + \pi_i \delta) \otimes \lambda_i(Q_\ell)^{\otimes(i-1)} \\ &= P_0 \otimes \sum_{i \geq 2} \lambda_i^{(1-\pi)}(Q_\ell)^{\otimes(i-1)} + \sum_{i \geq 2} \lambda_i^\pi(Q_\ell)^{\otimes(i-1)}, \quad (1) \end{aligned}$$

where $\lambda_i^{(1-\pi)} \triangleq (1 - \pi_i)\lambda_i$, $\lambda_i^\pi \triangleq \pi_i\lambda_i$, P_0 is the density of the message from the channel observation, and Q_ℓ is the density of the averaged check node message over the right degree distribution ρ_i . In the initial iteration, the density from unpunctured variable nodes is Gaussian when the codeword is transmitted over an AWGN channel, while for punctured

variable nodes the density is the Dirac delta function. The first and the second term in (1) represent densities from unpunctured and punctured nodes, respectively. In the ℓ th iteration, the check node message is [6]

$$Q_\ell = \Gamma^{(-1)}(\rho(\Gamma(P_{\ell-1}))).$$

The punctured variable nodes get recovered during the decoding process if they receive at least one non-zero message from neighboring check nodes. The probability for the variable node message to be zero in the ℓ th iteration can be expressed as follows

$$P_\ell(0) = \sum_{i \geq 2} \lambda_i^\pi (1 - \sum_{j \geq 2} \rho_j (1 - P_{\ell-1}(0))^{j-1})^{i-1},$$

$$P_0(0) = \sum_{i \geq 2} \lambda_i^\pi$$

which tends to zero since it is assumed that punctured nodes do not contain a stopping set. Of most interest is the behavior of densities in the steady state, where all punctured bits are recovered.

The averaged bit error rate over the punctured variable nodes is

$$P_e^{\pi,(\ell)} = \int_{-\infty}^{0^-} \Phi(x) dx + \frac{1}{2} \Phi(0),$$

where

$$\Phi(x) = \frac{1}{\sum_{i \geq 2} \Lambda_i^\pi} \sum_{i \geq 2} \Lambda_i^\pi (Q_\ell)^{\otimes i},$$

and $\Lambda_i^\pi \triangleq \pi_i \Lambda_i$. The bit error probability is evaluated with the discretized density evolution technique [8].

Figure 3 shows the probability density function of an average check-to-variable message in the steady state, when the decoder is operating below the threshold. Note that estimates for the value of punctured variable nodes depend exclusively on the incoming check node messages, as no channel value is available to the decoder. It can be seen that the check node message density is not well approximated by a Gaussian distribution [9]. A sharp spike can be seen at 0, followed by an exponential falloff. In consequence, Eve's ability to reconstruct the message is overestimated by GA. As will be shown, the security gaps when calculated accurately with DE are even smaller than the estimates from GA.

Toward this end, the effectiveness of hiding message bits by means of puncturing is demonstrated on an example. A mother code of rate 1/2 with the degree distribution:

$$\lambda(x) = 0.25105x + 0.30938x^2 + 0.00104x^3 + 0.43853x^9, \quad (2)$$

$$\rho(x) = 0.63676x^6 + 0.36324x^7 \quad (3)$$

is chosen and punctured randomly according to

$$\pi(x) = 0.4x + 0.4x^2 + 0.4x^3 + 0.4x^9, \quad (4)$$

The overall fraction of punctured bits p is 0.4 and all punctured bits are assumed to carry messages, therefore $R_s = p/(1 -$

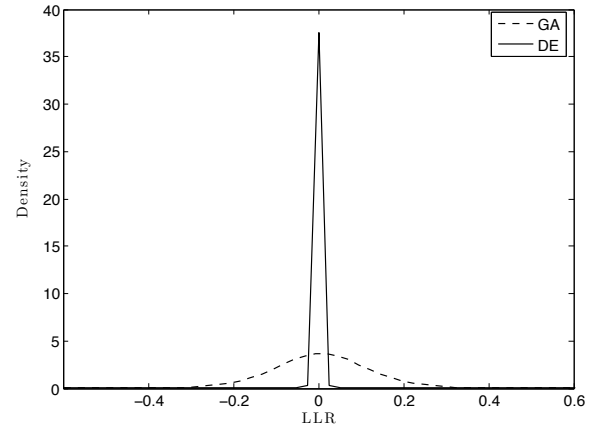


Fig. 3. Check-to-variable message PDF obtained by GA and DE when the decoder operates below the threshold.

$p) = 2/3$. Notice that $d = 0.5n$ and $s = 0.4n$, therefore $d - s = 0.1n$ variable nodes must be set by random dummy bits in the encoder.

For comparison, a 2/3-rate LDPC code is chosen, where messages are transmitted over the channel along with parities. Its degree distribution² is $\lambda(x) = 0.17599x + 0.40223x^2 + 0.42178x^9$, $\rho(x) = 0.61540x^{10} + 0.38460x^{11}$. The results for the BER over message bits for these two codes are shown in Figure 4.

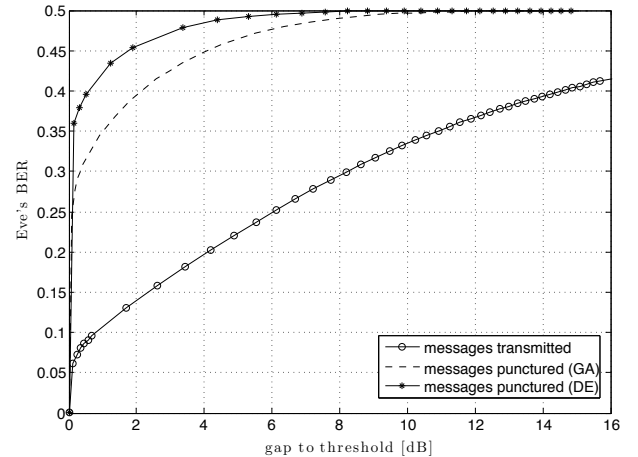


Fig. 4. Eve's BER performance when she operates below the threshold $\text{SNR}_{B,\min}$. Two methods are considered: (i) message bits are transmitted, and (ii) message bits are punctured.

Of most interest to us is the behavior of Eve's BER as her SNR declines from the threshold. If the message bits are punctured, Eve's BER increases much faster with the growing gap to threshold as it does when the messages are transmitted. For instance, for $P_{e,\min}^E$ set at 0.40, 0.45 and 0.49, the security gaps amount to 0.6 dB, 1.8 dB and 4.1 dB, respectively. In contrast, if the message bits are transmitted over the channel, the security gaps are considerably larger at 14 dB, 20 dB

²The degree distributions were obtained at <http://lthcwww.epfl.ch/research/ldpcopt/>

and 34 dB, respectively. These results manifest the benefit of protecting the message bits by means of puncturing. Further, they indicate that relatively high BERs at Eve are attainable for small security gaps. 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.

It must be noted that 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 4, the threshold SNR 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.

A natural question at this point is whether lower security gaps are achievable if the puncturing distribution is optimized for security instead of being random. Toward this end, a mother code with degree distribution (2), (3) was punctured in two different manners: (i) randomly, and (ii) according to an optimized puncturing distribution obtained by means of Differential Evolution [10]. The comparison, shown in Figure 5, was drawn at 4 different puncturing fractions: 0.1, 0.2, 0.3, and 0.4, which correspond to secrecy rates $1/9$, $2/8$, $3/7$, and $4/6$, respectively. The optimized puncturing distributions are given in Table I. Additionally, the increased power requirement at each considered rate is depicted as well, where the threshold of the punctured code was compared to the threshold of a well-performing unpunctured code of equal rate.

The benefit of using optimized puncturing distributions for security is most pronounced at high rates, where the puncturing fractions are high. The gains over random puncturing of up to 0.4 dB were achieved, which is a notable improvement at asymptotic block lengths. The security gap can be reduced at the expense of a lower secrecy rate (less punctured bits), however reducing the secrecy rate beyond a certain point ($R_s \approx 0.43$ in this case) is not reasonable due to negative effects both on the security gap and the price in power.

TABLE I
OPTIMIZED PUNCTURING DISTRIBUTIONS FOR SECURITY OBTAINED BY MEANS OF DIFFERENTIAL EVOLUTION. $P_{e,\min}^E$ WAS SET TO 0.49.

p	0.10	0.20	0.30	0.40
rate	0.1111	0.2500	0.4286	0.6667
π_2	0.1063	0.2965	0.4238	0.5527
π_3	0.1329	0.0009	0.0004	0.1361
π_4	0.7861	0.0159	0.0571	0.7492
π_{10}	0.0001	0.3930	0.6518	0.5816
security gap [dB]	4.346	4.222	4.026	4.386

Main reasons for the increased power requirement, or equivalently, reduced rate, are (i) losses inflicted by puncturing and (ii) rate loss due to $R_s < R_d$. The overall price in power can be reduced by using a mother code of rate lower than 0.5 and puncturing all independent bits. That would ensure that $R_s = R_d$ and consequently no inherent rate loss would be incurred. However, low rate mother codes also limit the size

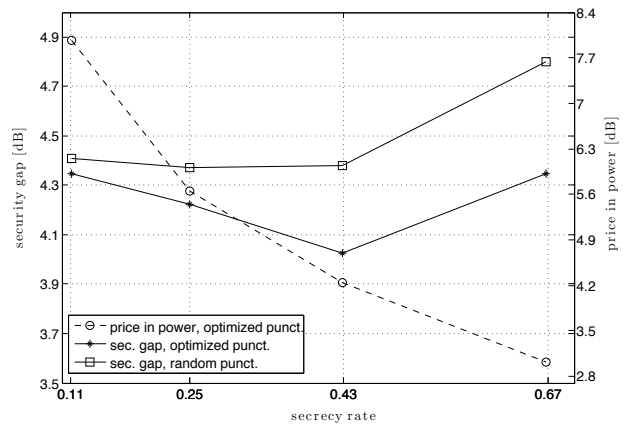


Fig. 5. The performance of puncturing distributions optimized for security. Comparison with random puncturing is drawn for $P_{e,\min}^E = 0.49$.

of the secrecy rate. Additionally, the attainable security gaps at low secrecy rates are expected to be higher.

IV. EFFICIENT ENCODING

It was shown in [11] that encoding in linear time is possible for many LDPC codes. However, in their analysis it is assumed that any variable node can be chosen to carry the message bits, which is not true in general for the case considered in this paper. This section addresses the encoding problem and shows that efficient encoding is possible if messages are transmitted in the proposed manner.

The main result from [11] states that any LDPC code, whose matrix can be rearranged by means of row and column permutations into the form shown in Figure 6, is encodable with complexity $O(n + g^2)$, where g is referred to as gap (see Figure 6). Efficient encoding is very closely related with

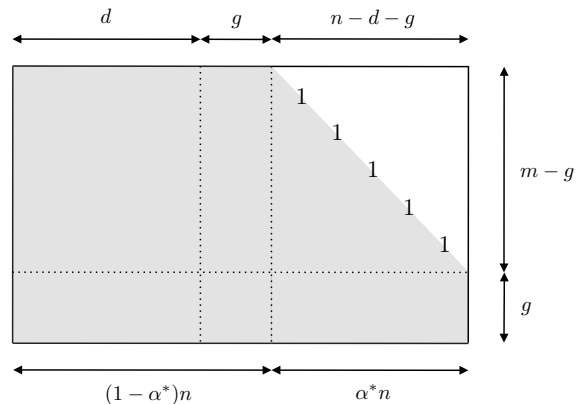


Fig. 6. Upper-triangular parity-check matrix suitable for efficient encoding.

code's ability to recover from erasures over the binary erasure channel (BEC). Let $\alpha^*(\lambda, \rho)$ denote the BEC threshold³ of a degree distribution pair $(\lambda(x), \rho(x))$. It was shown that using their greedy algorithm A,

³Note that the BEC threshold is defined differently than the threshold $\text{SNR}_{B,\min}$ from the previous Sections. See [4, p. 97] for details.

$$g = (1 - r(\lambda, \rho) - \alpha^*(\lambda, \rho))n \quad (5)$$

is achievable asymptotically. Sometimes, smaller gaps are attainable if triangulation is performed on the transpose of the parity check matrix. In that case the attainable gap becomes

$$g^{(T)} = \frac{1 - \alpha^*(\rho, \lambda)}{1 - r(\rho, \lambda)}n. \quad (6)$$

The letter T in the superscript indicates that the triangulation was performed on the transposed parity-check matrix. Note that in a transposed parity-check matrix the degree distributions of variable and check nodes are switched.

If puncturing is random, the greedy algorithm A from [11] can be applied directly. Consequently if a mother code has a BEC threshold close to capacity, the achievable gap g is small and the code is efficiently encodable. However, if puncturing is not random, the analysis from [11] has to be modified as the choice of variable nodes that carry messages is not random anymore.

Assume now that messages are transmitted over punctured nodes and the puncturing pattern is given by $\pi(x)$. The puncturing pattern $\pi(x)$ constraints the set of columns in the parity-check matrix that are subject to substitutions toward the triangular structure to only unpunctured variable nodes. Consequently, the methods from [11] must be applied to the residual parity-check matrix, which is obtained by deleting the columns that correspond to punctured bits. Let $(\lambda_{\text{res}}(x), \rho_{\text{res}}(x))$ be the degree distribution of the residual parity-check matrix. Further, let $\Lambda_{\text{res},i}$ denote the fraction of variable nodes of degree i in the residual parity-check matrix. Then

$$\Lambda_{\text{res},i} = \frac{(1 - \pi_i)\Lambda_i}{\sum_{j=2}^{d_v} (1 - \pi_j)\Lambda_j} \quad (7)$$

and

$$\lambda_{\text{res},i} = \frac{i\Lambda_{\text{res},i}}{\sum_{j=2}^{d_v} j\Lambda_{\text{res},j}}. \quad (8)$$

The average check node degree after puncturing is

$$\overline{d_c^p} = \frac{1 - \sum_{i=2}^{d_v} \pi_i \lambda_i}{(1 - r(\lambda, \rho)) \sum_{i=2}^{d_v} \lambda_i / i}. \quad (9)$$

In general, the residual check node distribution is not deterministic. To circumvent this problem it is assumed that it adheres to the uniform distribution as closely as possible in the sense that it comprises at most two terms. Then, the residual check node distribution from the node perspective is

$$\Gamma_{\text{res}}(x) = (1 - \lfloor \overline{d_c^p} \rfloor) \cdot x^{\lfloor \overline{d_c^p} \rfloor} + (\overline{d_c^p} - \lfloor \overline{d_c^p} \rfloor) \cdot x^{\lfloor \overline{d_c^p} \rfloor + 1}, \quad (10)$$

while from the edge perspective it is

$$\rho_{\text{res}}(x) = \frac{\Gamma'_{\text{res}}(x)}{\Gamma'_{\text{res}}(1)}. \quad (11)$$

With the residual degree distribution the attainable gap size can be computed according to

$$g = (1 - r(\lambda_{\text{res}}, \rho_{\text{res}}) - \alpha^*(\lambda_{\text{res}}, \rho_{\text{res}}))(1 - p)n \quad (12)$$

and

$$g^{(T)} = \frac{1 - \alpha^*(\rho_{\text{res}}, \lambda_{\text{res}})}{1 - r(\rho_{\text{res}}, \lambda_{\text{res}})}(1 - p)n. \quad (13)$$

Table II shows the encoding gaps for optimized puncturing distributions from Table I, which confirm that small gaps, and thus almost linear encoding, are indeed attainable for the proposed method.

TABLE II
ATTAINABLE GAPS.

p	g	$g^{(T)}$
0	0.0298n	0
0.1	0.0405n	0
0.2	0.0303n	0
0.3	0.0290n	0.0024n
0.4	0.0499n	0.0160n

V. CONCLUSION

This paper is an attempt to view at the Gaussian wiretap channel from a different perspective. Instead of equivocation, the secrecy metric of choice is the BER over message bits, which allows for easier analysis and transition to finite-block length constructions. A coding method based on LDPC codes is proposed where message bits are hidden from the eavesdropper by means of puncturing. It has been shown that eavesdroppers are forced to BERs close to 0.5 even if they have the ability to use a bitwise-MAP decoder, when they operate a few dB under the code's SNR threshold. In addition to providing physical layer security, the proposed coding scheme is shown to be efficiently encodable and can be directly applied to finite block length constructions.

REFERENCES

- [1] A. D. Wyner, "The wire-tap channel," *The Bell System Technical Journal*, vol. 54, no. 8, pp. 1355–1387, Oct 1975.
- [2] I. Csiszar and J. Korner, "Broadcast channels with confidential messages," *IEEE Transactions on Information Theory*, vol. 24, no. 3, pp. 339–348, May 1978.
- [3] R. G. Gallager, *Low-density parity-check codes*. MIT Press, 1963.
- [4] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.
- [5] J. Ha, J. Kim, and S. McLaughlin, "Rate-compatible puncturing of low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2824–2836, Nov 2004.
- [6] T. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 599–618, Feb 2001.
- [7] D. Klinc, J. Ha, S. W. McLaughlin, J. Barros, and B.-J. Kwak, "Ldpc codes for physical layer security," in *submitted for publication*, 2009.
- [8] S. Chung, "On the construction of some capacity-approaching coding schemes," Ph.D. dissertation, Massachusetts Institute of Technology, 2000.
- [9] S. Chung, T. Richardson, and R. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a gaussian approximation," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 657–670, Feb 2001.
- [10] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341–359, Dec 1997.
- [11] T. Richardson and R. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 638–656, Feb 2001.