



Energy Consumption of Polar Codes for Wireless Sensor Networks

Liping Li¹(✉), Quanyv Wang¹, Yanjun Hu¹, and Chuan Zhang²

¹ Key Laboratory of Intelligent Computing and Signal Processing of the Ministry of Education of China, Anhui University, Hefei, China

liping_li@ahu.edu.cn

² National Mobile Communications Research Laboratory, Southeast University, Nanjing, China

chzhang@seu.edu.cn

Abstract. Internet of Things (IoT) is becoming a strong force driving the evolution of mobile communications. Wireless sensor networks (WSN) are important parts of IoT. Link reliability and power consumption are two critical design constraints in WSN designs. Error control coding (ECC) is a classic approach to increase link reliability and thus to lower the required transmitting power, however typically at the cost of an increased decoding complexity and power. The idea of this paper is to assess the potentials of applying polar codes to WSNs. For comparison, the well developed Turbo and LDPC codes are studied in terms of the error performance and the power consumption. The results in this paper show that when the ratio of the transmitting power over the decoding power is smaller than 2.5, applying polar codes is favorable in terms of the power consumption compared with Turbo and LDPC codes at the same BER performance. When this ratio is larger than 2.5, polar codes are almost the same as Turbo and LDPC codes in terms of the energy consumption.

Keywords: Internet of Things (IoT) · Error control codes
Polar codes · LDPC codes · Turbo codes · Energy consumption

1 Introduction

Four typical technical scenarios of 5G are derived from the main application scenarios, service requirements, and key challenges of mobile internet and Internet of Things (IoT) [1]. Wireless sensor networks (WSNs) are important parts of IoT. The most significant challenge in sensor networks is to overcome the energy constraints since sensor nodes typically have limited power.

In WSNs, to increase the link reliability, Automatic Repeat reQuest (ARQ) can be deployed when errors occur. However, energy is wasted due to the retransmission in the network. A particularly undesirable situation occurs when the channel condition is bad, causing successive retransmissions. Forward error correction (FEC) [2] is an effective way to reduce the frame error rate and consequently reduce the number of retransmissions.

Block codes and convolutional codes with Viterbi decoding are applied to WSNs in [3–8]. Iterative decoding algorithm justifies the ability of Turbo codes in solving the hot-spot problem and prolong the network lifetime [6]. Previous work using error control coding (ECC) in wireless sensor networks focused primarily on codes such as Reed-Solomon and convolutional codes. A hybrid scheme choosing the energy-efficient combination of ECC and ARQ is considered in [7]. Convolutional codes with different rates and constraint lengths are studied in [8] for wireless microsensor networks. Other system-level techniques such as modulation and MAC protocols are also considered to reduce the energy consumption in [8].

The aforementioned coding scheme indeed can increase link reliability and lower the required transmitting power. However, extra power consumption of decoders can not be neglected especially with sparse capacity-approaching codes [6, 9]. It's shown in [9] that the transmitted power for short distances is only a few tens of the decoding power. Therefore, a coding scheme which can increase the link reliability but also consumes an acceptable level of decoding power is desired.

The discovery of channel polarization and polar codes by Arıkan [10] is universally recognized as a major breakthrough in coding theory. Polar codes provably achieve the capacity of binary-input discrete memoryless symmetric (B-DMS) channels, with a low encoding and decoding complexity. Moreover, polar codes have an explicit construction (there is no random ensemble to choose from) and a beautiful recursive structure that makes them inherently suitable for efficient implementation in hardware [11]. Therefore, in this paper, polar codes are studied when applied in WSNs. The focus of this paper is to qualify the ability of polar codes in increasing the link reliability while maintaining a low decoding power consumption. In this respect, two other codes are studied along with polar codes for comparison: Turbo codes [12] and LDPC codes [13, 14]. As mentioned in [9], the decoding power is dominating the processing power at the chip. A fair comparison is carried out in this paper by forcing the three coding schemes having the same bit-error-rate (BER) performance, which results in different transmitting power requirements for Turbo, LDPC, and polar codes. The decoding power of these codes are multiples of the transmitting power [9] depending on the distance of communications and the specific system parameters: center frequency, bandwidth, and throughput. From simulations of a simple WSN network with 100 nodes, the life time of the WSN network is found to be the longest when applying polar codes and the ratio of the transmitting power over the decoding power is smaller than 2.5.

The rest of the paper is organized as follows. Section 2 introduces the basics of polar codes. Section 3 discusses the energy model used in this paper. Section 4 provides the BER performance and power consumption of several error correction codes. Finally the conclusion remarks are provided at the end.

2 Polar Codes Basics

In this section, the relevant theories of non-systematic polar codes and systematic polar codes are presented based on [10, 15].

2.1 Non-systematic Polar Codes

Polar codes presented in [10] are in the non-systematic form. The generator matrix is $G_N = BF^{\otimes n}$ where B is a bit-reversal matrix, $F = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $n = \log_2 N$, and $F^{\otimes n}$ is the n th Kronecker power of the matrix F over the binary field \mathbb{F}_2 .

Throughout the paper, we use u_1^N to refer to a row vector with N elements: $v_1^N = (v_1, v_2, \dots, v_N)$. Let \mathcal{A} denote a set. Then $u_{\mathcal{A}}$ is a subvector of u_1^N with elements specified by \mathcal{A} . With these notations, the encoding of polar codes is: $x_1^N = u_1^N G_N$. Here the source vector is u_1^N which consists of the information bits and the frozen bits, denoted by $u_{\mathcal{A}}$ and $u_{\bar{\mathcal{A}}}$, respectively. The frozen bits are known to the receiver. The encoded bits in x_1^N are transmitted in N independent underlying channels. Let the underlying channel be W with a transition probability $W(x|y)$ where $x \in \mathcal{X} = \{0, 1\}$ and $y \in \mathcal{Y}$. The set \mathcal{X} and \mathcal{Y} contains the input and output alphabets, respectively. Note that W is a binary-input memoryless symmetric channel (B-MSC).

Transmitting the codeword x_1^N from N independent copies of W produces a vector channel:

$$W_N(y_1^N | u_1^N) = W^N(y_1^N | x_1^N) = W^N(y_1^N | u_1^N G_N) \quad (1)$$

This vector channel is split into N bit channels given by:

$$W_N^{(i)}(y_1^N, u_1^{i-1} | u_i) = \sum_{u_{i+1}^N \in \mathcal{X}^{N-i}} \frac{1}{2^{N-1}} W_N(y_1^N | u_1^N) \quad (2)$$

Polarization happens that when N is large enough, these bit channels either are noiseless, or completely noisy [10]. And the portion of the perfect bit channels is equal to the symmetric capacity of the underlying channel W .

To conclude this subsection, a note is necessary on the selection of the good bit channels in the set \mathcal{A} . Algorithms such as [16] can be used to sort the bit channels and the best of K bit channels can be selected when $K = NR$ (R being the code rate). Both set \mathcal{A} and its complementary set $\bar{\mathcal{A}}$ are in $\{1, 2, \dots, N\}$ for polar codes with a block length $N = 2^n$.

2.2 Construction of Systematic Polar Codes

For systematic polar codes, we also focus on a generator matrix without the permutation matrix B , namely $G = F^{\otimes n}$.

The source bits \mathbf{u} can be split as $\mathbf{u} = (u_{\mathcal{A}}, u_{\bar{\mathcal{A}}})$. The first part $u_{\mathcal{A}}$ consists of user data that are free to change in each round of transmission, while the second part $u_{\bar{\mathcal{A}}}$ consists of data that are frozen at the beginning of each session and made known to the decoder. The codeword can then be expressed as

$$\mathbf{x} = u_{\mathcal{A}} G_{\mathcal{A}} + u_{\bar{\mathcal{A}}} G_{\bar{\mathcal{A}}} \quad (3)$$

where $G_{\mathcal{A}}$ is the sub-matrix of G with rows specified by the set \mathcal{A} . The systematic polar code is constructed by specifying a set of indices of the codeword \mathbf{x} as the indices to convey the information bits. Denote this set as \mathcal{B} and the complementary set as $\bar{\mathcal{B}}$. The codeword \mathbf{x} is thus split as $(x_{\mathcal{B}}, x_{\bar{\mathcal{B}}})$. With some manipulations, we have

$$\begin{cases} x_{\mathcal{B}} = u_{\mathcal{A}}G_{\mathcal{A}\mathcal{B}} + u_{\bar{\mathcal{A}}}G_{\bar{\mathcal{A}}\mathcal{B}} \\ x_{\bar{\mathcal{B}}} = u_{\mathcal{A}}G_{\mathcal{A}\bar{\mathcal{B}}} + u_{\bar{\mathcal{A}}}G_{\bar{\mathcal{A}}\bar{\mathcal{B}}} \end{cases} \quad (4)$$

The matrix $G_{\mathcal{A}\mathcal{B}}$ is a sub-matrix of the generator matrix with elements $\{G_{i,j}\}_{i \in \mathcal{A}, j \in \mathcal{B}}$. Given a non-systematic encoder $(\mathcal{A}, u_{\bar{\mathcal{A}}})$, there is a systematic encoder $(\mathcal{B}, u_{\mathcal{A}})$ which performs the mapping $x_{\mathcal{B}} \mapsto \mathbf{x} = (x_{\mathcal{B}}, x_{\bar{\mathcal{B}}})$. To realize this systematic mapping, $x_{\bar{\mathcal{B}}}$ needs to be computed for any given information bits $x_{\mathcal{B}}$. To this end, we see from (4) that $x_{\bar{\mathcal{B}}}$ can be computed if $u_{\mathcal{A}}$ is known. The vector $u_{\mathcal{A}}$ can be obtained as the following

$$u_{\mathcal{A}} = (x_{\bar{\mathcal{B}}} - u_{\bar{\mathcal{A}}}G_{\bar{\mathcal{A}}\bar{\mathcal{B}}})(G_{\mathcal{A}\bar{\mathcal{B}}})^{-1} \quad (5)$$

From (5), it's seen that $x_{\bar{\mathcal{B}}} \mapsto u_{\mathcal{A}}$ is one-to-one if $x_{\bar{\mathcal{B}}}$ has the same elements as $u_{\mathcal{A}}$ and if $G_{\mathcal{A}\bar{\mathcal{B}}}$ is invertible. In [15], it's shown that $\mathcal{B} = \mathcal{A}$ satisfies all these conditions in order to establish the one-to-one mapping $x_{\bar{\mathcal{B}}} \mapsto u_{\mathcal{A}}$. In the rest of the paper, the systematic encoding of polar codes adopts this selection of \mathcal{B} to be $\mathcal{B} = \mathcal{A}$. Therefore we can rewrite (4) as

$$\begin{cases} x_{\mathcal{A}} = u_{\mathcal{A}}G_{\mathcal{A}\mathcal{A}} + u_{\bar{\mathcal{A}}}G_{\bar{\mathcal{A}}\mathcal{A}} \\ x_{\bar{\mathcal{A}}} = u_{\mathcal{A}}G_{\mathcal{A}\bar{\mathcal{A}}} + u_{\bar{\mathcal{A}}}G_{\bar{\mathcal{A}}\bar{\mathcal{A}}} \end{cases} \quad (6)$$

3 Energy Model

In this section, the average energy consumption of Turbo codes, LDPC codes, and polar codes are analyzed. As in [9], the energy is largely divided as the transmitting power and the decoding power. Traditionally, the transmitting power dominates the link budget as communications normally travel a long distance. As the distance of communications became short and the capacity-approaching error correction codes developed, the decoding power started to dominate the processing power.

Figure 1 is a reproduction of the two cases considered in [9] where the dashed line is the decoding power of a real implementation of a LDPC decoder [17]: 144 mW. From Fig. 1 it can be seen that the transmitting power is small than the decoding power for distances within 10 m. This motivates us to investigate a general energy consumption model which is not related to any specific implementation. Instead, the ratio of the transmitting and decoding power is $\alpha = E_{tx}/E_{dec}$. We study the effect of the ratio α to the overall energy consumption in a WSN setting.

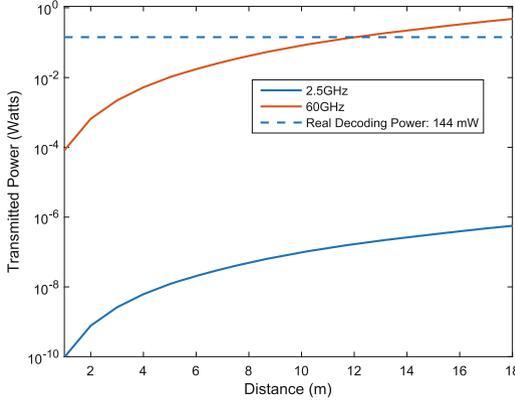


Fig. 1. Required power from Shannon prediction of two cases. Case 1 at 2.5GHz: bandwidth is 80 MHz and the throughput is 26 MHz. Case 2 at 60 GHz: bandwidth is 3 GHz and the throughput is 1.5 GHz. The real decoding power is from [17].

3.1 General Energy Model

Suppose there are N_{total} total of nodes in a WSN, labeled from 1 to N_{total} . Since this work does not deal with routing or any other optimization related to WSNs, we assume a simple protocol: nodes 1 to N_{total} transmitting and receiving (decoding) one by one in the natural order. Each complete transmission from node 1 to node N_{total} is called one round of transmission. With such a simplification, we can evaluate how many times of runs such a network can perform and how many nodes are still alive in each run. Here we assume a complete non-heterogenous network, meaning that the nodes can have different energy levels.

Denote E_{total}^i as the total energy of node i . Suppose each time a node transmits, it transmits one second. In other words, the energy consumed by transmitting is E_{tx} . We also assume that the decoder continuously works for one second when a node performs decoding. Therefore, the total energy E_{total}^i in a node will be completely consumed after N_i times of transmitting and decoding with

$$E_{total}^i = N_i(E_{tx} + E_{dec}) \tag{7}$$

With $\alpha = E_{tx}/E_{dec}$, E_{total}^i can be written as

$$E_{total}^i = N_i(1 + \alpha)E_{dec} \tag{8}$$

3.2 Transmitting and Decoding Energy of Three Codes

In this section, based on (7) and (8), the energy consumption of three codes are analyzed and compared: LDPC codes [13,14], Turbo codes [12], and polar codes [10]. For a fair comparison, both E_{tx} and E_{dec} need to be adjusted in (7) and (8).

To achieve the same bit error rate P_b , the required signal-to-noise ratio (SNR) per bit (E_b/N_0) for different codes are different. For comparison, let $(E_b/N_0)_{ldpc}$ be the required SNR per bit for LDPC codes. Then, the required SNR per bit for polar codes and Turbo codes can be expressed as:

$$(E_b/N_0)_{turbo} = \alpha_t(E_b/N_0)_{ldpc} \quad (9)$$

$$(E_b/N_0)_{polar} = \alpha_p(E_b/N_0)_{ldpc} \quad (10)$$

Since there is no closed form expression for all three codes, the ratio α_t and α_p can only be obtained from numerical simulations for any given P_b . Once the ratio α_t and α_p are obtained, the transmitting power of polar and Turbo codes can be obtained relative to the LDPC code.

With the same block length and code rate, the decoding power of the three codes are analyzed based on the number of iterations in the decoding process. Other factors such as the decoder structure, the parallelism, and the log likelihood ratio (LLR) calculations are not considered. Suppose the number of iterations of the LDPC code is I_l . In the same way, the number of iterations of polar and Turbo codes is I_p and I_t , respectively. Let $\beta_p = I_p/I_l$ and $\beta_t = I_t/I_l$. Then the decoding power of the three codes can be expressed as:

$$(E_{dec})_{turbo} = \beta_t(E_{dec})_{ldpc} \quad (11)$$

$$(E_{dec})_{polar} = \beta_p(E_{dec})_{ldpc} \quad (12)$$

With (9)–(12), the total energy in (7) for node i can be expressed as:

$$(E_{total}^i)_{turbo} = N_i(\alpha\alpha_t + \beta_t)(E_{dec})_{ldpc} \quad (13)$$

$$(E_{total}^i)_{polar} = N_i(\alpha\alpha_p + \beta_p)(E_{dec})_{ldpc} \quad (14)$$

where the subscript ‘turbo’ means node i employing Turbo encoding and decoding and ‘polar’ means polar codes are employed for node i . Note that the WSN considered only employs one code type: either a Turbo code is employed for all nodes, or a LDPC code, or a polar code. Therefore, the energy consumption in (13) and (14) is the energy model for all nodes in a WSN.

4 Simulation Results

In this section, simulation results are provided to verify the performance of different ECC in WSNs. The LDPC code used in this section is constructed by the Construction 1A in [14]. The LDPC code has a block length 512 and code rate 1/2. The two generators for the rate 1/2 recursive systematic convolutional (RSC) code is: $G_1 = 13$ and $G_2 = 15$. The number of iterations in the decoding of this Turbo code is set to be $I_t = 10$. The average number of iterations in the decoding of the LDPC code is recorded in the simulation as $I_l = 5$. As for the polar code (block length 512 and code rate 1/2), since successive cancellation

(SC) [10] decoding is applied, the number of iterations in the decoding is $I_p = 1$. Therefore, the decoding power ratio is:

$$\beta_p = I_p/I_t = 1/5 \tag{15}$$

$$\beta_t = I_t/I_t = 10/5 = 2 \tag{16}$$

Note that with the previous settings of the three codes, LDPC, Turbo, and polar codes have the same code rate. This is the basis for a fair comparison among these codes in terms of the error performance and the energy consumption.

To obtain the relative transmitting power in achieving a given BER P_b , the BER performance of these codes is simulated. Figure 2 displays the BER performance in an additive white Gaussian noise (AWGN) channel. Note that in Fig. 2, the BER performance of the schemes in three short distance communication protocols are also provided. The (15, 10) shortened Hamming code is used in the BlueTooth protocol. The coding scheme of 802.11g is the convolutional code with a code rate 1/2 and a constraint length 7. Uncoded BPSK is used in the ZigBee protocol.

It can be seen from Fig. 2 that the BER performance of the coding schemes in BlueTooth and the 802.11g is worse than the coding schemes of (512, 256) polar and LDPC codes. The rate 1/2 Turbo codes has the best BER performance among the codes considered. Now let us set a target $P_b = 10^{-3}$. Then compared with the LDPC code, Turbo code needs around 0.9 dB less and polar code needs around 0.5 dB more E_b/N_0 to achieve this P_b . Therefore, the transmitting power adjustments are the following:

$$\alpha_p = (E_b/N_0)_{polar}/(E_b/N_0)_{ldpc} = 1.122 \tag{17}$$

$$\alpha_t = (E_b/N_0)_{turbo}/(E_b/N_0)_{ldpc} = 0.813 \tag{18}$$

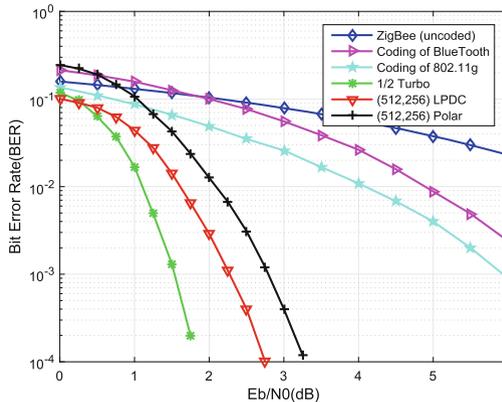


Fig. 2. The comparison of the BER performance of various codes in the AWGN channel. There is no coding in ZigBee. The code in BlueTooth is the shortened (15, 10) Hamming code. In 802.11g, the rate 1/2 convolutional code with a constraint length 7 is used. The rate 1/2 RSC Turbo code has two generators: $G_1 = 13$ and $G_2 = 15$.

With the transmitting power and the decoding power adjustments in (15)–(18), the only parameter in (13) and (14) is α , which is the ratio of the transmitting power relative to the decoding power. In the following simulations, this parameter α is set to different values to compare the energy consumption of the three coding schemes: polar code, LDPC code, and Turbo code described at the beginning of this section.

First of all, assume there are $N_{total} = 100$ nodes in the network. As discussed in Sect. 3.1, the nodes can have different energy levels. The total energy in node i is modelled as

$$E_{total}^i = 1 + e_i \quad (\text{Joules}) \quad (19)$$

where e_i is a random variable with uniform distribution. The random variables $\{e_i\}_{i=1}^{N_{total}}$ are independent. The transmission strategy described in Sect. 3.1 is used in the simulation: each node transmitting and receiving (decoding) one by one and each transmitting and decoding operation takes one second. Assume the decoding power for the LDPC code is $(E_{dec})_{ldpc} = 10^{-3}$ Watts. With the initial total energy distribution in this network, in the following operations in each second, all nodes in the network perform transmitting and decoding. As in Sect. 3.1, a complete transmission from node 1 to node $N_{total} = 100$ is one round of transmission. When the total energy E_{total}^i of node i is completely consumed from these transmitting and decoding operations, then this node dies. In each round, the total number of alive nodes are recorded for each decoding scheme.

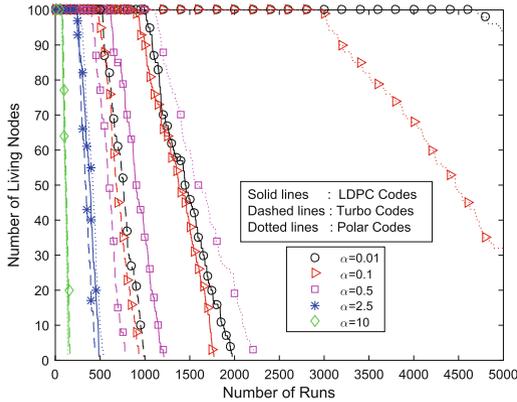


Fig. 3. Energy consumption comparison of three codes: polar code (512, 256), Turbo code, and LDPC code (512, 256). The Turbo code is rate 1/2 RSC code with $G_1 = 13$ and $G_2 = 15$. The LDPC code is constructed from Construction 1A from [14].

Figure 3 shows the energy consumption comparison of the three codes. The x-axis is the indices of the number of runs. The y-axis is the number of alive nodes in each run. For each code type, there are five curves: the solid lines are the alive nodes for the LDPC code, the dashed lines for the Turbo code, and the dotted lines for the Polar code.

and the dotted lines for the polar code. For each code type, the line with circles corresponds to $\alpha = 0.01$, meaning the transmitted power is only one hundredth of the decoding power. The other markers corresponds to the following: the line with triangles is for $\alpha = 0.1$, the line with squares is for $\alpha = 0.5$, the line with asterisks is for $\alpha = 2.5$, and the line with diamonds is for $\alpha = 10$.

For ease of description, a network is said to have a better energy efficiency if it has the most number of living nodes when operating with the same conditions compared with another network. From Fig. 3, it can be seen that the network employing polar code with $\alpha = 0.01$ and $\alpha = 0.1$ is very energy efficient: At runs 3000, all 100 nodes are still alive. However, all nodes are dead before run 2000 in this network employing LDPC (the solid line with circles) and Turbo codes (the dashed line with circles). At $\alpha = 0.5$, the network employing polar code displays a slightly better energy efficiency as that employing LDPC codes with $\alpha = 0.01$ $\alpha = 0.1$, which indicates that the network employing polar code can support a larger communication distance compared with that employing the LDPC code when the other parameters of the network are the same. This energy efficiency of polar code is even more pronounced when compared with the Turbo code: at $\alpha = 0.5$, the network life time with polar code is 1500 runs longer than the network employing Turbo codes.

However, when the transmitting power starts to dominate the energy consumption, all three codes converge in terms of the network lifetime. This can be seen from the three lines with asterisks (corresponding to the case of $\alpha = 2.5$) in Fig. 3. In this case, the transmitting power is 2.5 times larger than the decoding power. The networks employing polar, LDPC, and Turbo codes have almost the same energy efficiency at $\alpha = 2.5$. The same is true when $\alpha = 10$.

5 Conclusion

The use of forward error correcting codes can allow a system to operate at significantly lower SNR than an uncoded system for a given BER. But the choice of ECC is very important for a wireless sensor network. In this paper, we investigate the usage of polar codes applied in a WSN in a noisy environment. The main metric is the energy consumption when comparing polar codes with the existing capacity-achieving Turbo and LDPC codes. When the ratio of the transmitting power over the decoding power is below a threshold (in this paper, smaller than 2.5), polar codes are shown to be much energy efficient in prolonging the network lifetime compared with Turbo and LDPC codes. This shows that at low communication distances, polar codes are a very energy-efficient error correction code to be applied in WSNs.

Acknowledgement. This work was supported in part by National Natural Science Foundation of China through grant 61501002, in part by Natural Science Project of Ministry of Education of Anhui through grant KJ2015A102, in part by the Key Laboratory Project of the Key Laboratory of Intelligent Computing and Signal Processing of the Ministry of Education of China, Anhui University, in part by Talents Recruitment Program of Anhui University.

References

1. White paper on 5G concept: IMT-2020(5G) Promotion Group, February 2015
2. Lin, S., Costello, D.J.: Error Control Coding. Pearson Prentice Hall, Upper Saddle River (2004)
3. Kashani, Z.H., Shiva, M.: BCH coding and multi-hop communication in wireless sensor networks. In: IFIP International Conference on Wireless and Optical Communications Networks, pp. 1–5, April 2006
4. Kashani, Z.H., Shiva, M.: Channel coding in multi-hop wireless sensor networks. In: 2006 6th International Conference on ITS Telecommunications Proceedings, June 2006
5. Sankarasubramaniam, Y., Akyildiz, I.F., McLaughlin, S.W.: Energy efficiency-based packet size optimization in wireless sensor networks. In: IEEE International Workshop on Sensor Networks Protocols and Applications, May 2003
6. Vasudevan, S., Goeckel, D., Towsley, D.: Optimal power allocation in channel-coded wireless networks. In: Allerton Conference on Communication, Control and Computing, September 2004
7. Lettieri, P., Fragouli, C., Srivastava, M.B.: Low power error control for wireless links. In: 3rd Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM 1997), pp. 139–150, September 1997
8. Shih, E., Cho, S., Lee, F.S., Calhoun, B.H., Chandrakasan, A.: Design considerations for energy-efficient radios in wireless microsensor networks. *J. VLSI Sig. Process. Syst. Sig. Image Video Technol.* **37**(1), 77–94 (2004)
9. Grover, P., Woyach, K., Sahai, A.: Towards a communication-theoretic understanding of system-level power consumption. *IEEE J. Sel. Areas Commun.* **29**(8), 1744–1755 (2011)
10. Arikan, E.: Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels. *IEEE Trans. Inf. Theory* **55**(7), 3051–3073 (2009)
11. Sarkis, G., Giard, P., Vardy, A., Thibeault, C., Gross, W.J.: Hardware implementation of successive-cancellation decoders for polar codes. *IEEE J. Sel. Areas Commun.* **32**(5), 946–957 (2014)
12. Berrou, C., Glavieux, A.: Near optimum error correcting coding and decoding turbocodes. *IEEE Trans. Commun.* **44**(10), 1261–1271 (1996)
13. Gallager, R.G.: Low Density Parity Check Codes. MIT Press, Cambridge (1963). Monograph. Accessed Aug 2013
14. MacKay, D.J.C., Neal, R.M.: Near Shannon limit performance of low density parity check codes. *Electron. Lett.* **33**(6), 457–458 (1997)
15. Arikan, E.: Systematic polar coding. *IEEE Commun. Lett.* **15**(8), 860–862 (2011)
16. Tal, I., Vardy, A.: How to construct polar codes. *IEEE Trans. Inf. Theory* **59**(10), 6562–6582 (2013)
17. Zhang, Z., Anantharam, V., Wainwright, M.J., Nikolic, B.: An efficient 10GBASE-T ethernet LDPC decoder design with low error floors. *IEEE J. Solid-State Circ.* **45**(4), 843–855 (2010)