

Article

Optimized Polar Codes as Forward Error Correction Coding for Digital Video Broadcasting Systems

Karim El-Abbasy ^{1,2,*} , Ramy Taki Eldin ³, Salwa El Ramly ² and Bassant Abdelhamid ² 

¹ Spectrum Management and Regulatory Affairs Department, The Egyptian Satellite Company (Nilesat), Cairo 12568, Egypt

² Electronics and Communications Engineering Department, Faculty of Engineering, Ain Shams University, Cairo 11517, Egypt; salwa_elramly@eng.asu.edu.eg (S.E.R.); bassant.abdelhamid@eng.asu.edu.eg (B.A.)

³ Physics and Mathematics Department, Faculty of Engineering, Ain Shams University, Cairo 11517, Egypt; Ramy.Farouk@eng.asu.edu.eg

* Correspondence: k.alabbasy@nilesat.com.eg

Abstract: Polar codes are featured by their low encoding/decoding complexity for symmetric binary input-discrete memoryless channels. Recently, flexible generic Successive Cancellation List (SCL) decoders for polar codes were proposed to provide different throughput, latency, and decoding performances. In this paper, we propose to use polar codes with flexible fast-adaptive SCL decoders in Digital Video Broadcasting (DVB) systems to meet the growing demand for more bitrates. In addition, they can provide more interactive services with less latency and more throughput. First, we start with the construction of polar codes and propose a new mathematical relation to get the optimized design point for the polar code. We prove that our optimized design point is too close to the one that achieves minimum Bit Error Rate (BER). Then, we compare the performance of polar and Low-Density Parity Check (LDPC) codes in terms of BER, encoder/decoder latencies, and throughput. The results show that both channel coding techniques have comparable BER. However, polar codes are superior to LDPC in terms of decoding latency, and system throughput. Finally, we present the possible performance enhancement of DVB systems in terms of decoding latency and complexity when using optimized polar codes as a Forward Error Correction (FEC) technique instead of Bose Chaudhuri Hocquenghem (BCH) and LDPC codes that are currently adopted in DVB standards.

Keywords: polar codes; LDPC codes; adaptive successive cancellation list decoder; decoding latency; DVB systems



Citation: El-Abbasy, K.; Taki Eldin, R.; El Ramly, S.; Abdelhamid, B. Optimized Polar Codes as Forward Error Correction Coding for Digital Video Broadcasting Systems. *Electronics* **2021**, *10*, 2152. <https://doi.org/10.3390/electronics10172152>

Academic Editor: Rafael P. Torres

Received: 5 August 2021

Accepted: 27 August 2021

Published: 3 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

More than a decade ago, polar codes have been introduced by Arikan as a Shannon limit capacity achieving codes for symmetric binary-input discrete memoryless channels. The main idea of polar codes is to create virtual synthetic polarized noise-free or pure-noisy channels from unpolarized equally likely independent channels. The polarized noise-free channels are used to carry the information bits, while the polarized pure-noisy channels are used to carry the known frozen bits [1].

The most significant feature of polar codes is the low encoding and decoding complexity, which make it suitable for control channels of 5G New Radio (NR) communications system [2]. In addition, polar codes inherently support the adaptation of the code rate by changing only the number of frozen bits while using the same encoder and decoder [3]. Thus, it would be a good candidate for other communication systems and is worth investigating.

Polar codes construction proposed by Arikan is based on the simplest 2×2 binary kernel matrix G_2 [1]. Therefore, code lengths of such polar codes are constrained to 2^n , which makes them unsuitable in low-rate real-time communication systems. In fact,

such systems require flexible medium and short code lengths, such as real-time video communication systems. This obstacle was overcome by constructing polar codes with arbitrary code length through various puncturing strategies, as shown in [4,5]. In all these strategies, the punctured bits are obtained by removing some of the encoder output bits according to a pre-determined puncturing pattern optimized by one of these strategies. The punctured bits are not sent over the channel, and before decoding process the corresponding Log-Likelihood Ratios (LLRs) for these removed bits are set again according to a pre-defined value in puncturing algorithm. While these methods allow for the construction of arbitrary length polar codes, the wrong selection of puncturing bits location could alter the polarization process of the codes and could decrease decoding error-rate performance [6].

On the other hand, the generation of polar codes using different higher dimensions $m \times m$ binary kernel matrix has been introduced in [7–11]. These methods describe the choice of the best kernel matrix with maximum exponent that provides the highest performance. A family of polar codes based on multi-kernel (MK) constructions have been presented in [12,13], which showed that the channel polarization condition still applies for such MK polar codes.

In recent years, the concept and design principles of MK polar codes have become a significant research area [6,14,15]. Therefore, flexible code lengths can be obtained using MK construction by applying the Kronecker product for different base kernels matrices with various dimensions.

Another concern related to polar codes is their channel-specific nature, which requires optimizing polar codes construction for the given channel [3]. Actually, polar code optimization is about selecting the right indices for the most reliable channels from the synthesized channels. This can be determined by the Bhattacharyya parameter bounds [1]. However, closed-form expressions for the Bhattacharyya parameter bounds of the synthesized channels are usually unavailable for general channels [16]. Moreover, the reliability of the synthesized channels can be acquired by other three popular techniques which are Monte-Carlo estimation approach [1], density evolution (DE) technique [17], and Gaussian approximation (GA) technique [18]. These methods define each synthetic channel as a pure-noise channel to be reserved for a frozen bit or noise-free channel to be reserved for an information bit. However, all these techniques depend on the conditions of the transmission channel, which makes it necessary to construct polar codes separately for each signal to noise ratio (SNR) [19]. This can be solved by modelling the channel with all characteristics and designing the code for the worst operating SNR [20]. In general, the optimized SNR design point is very important to construct polar codes where the construction of polar codes plays a significant role in improving its performance, but the major role is related to the decoding method of polar codes.

Over the past decade, decoding of polar codes has gone through some improvement stages. Polar codes decoding started from the simple Successive Cancellation Decoding (SCD) method proposed by Arikan [1]. Subsequently, it has been enhanced several times using advanced decoding techniques such as SCL decoding [21]. In SCL, a list of size L of SCD paths is concurrently generated at each decoding stage to select the best paths. SCL decoding was improved later in [22] to Cyclic Redundancy Check (CRC) Aided SCL (CA-SCL), where CRC detector selects the correct codewords from the list. This detector proceeds the candidate codewords and feeds back again SCL decoder with check results.

CA-SCL decoding technique for polar codes achieves nearly a similar performance as LDPC codes and turbo codes [23]. However, its computational complexity leads to higher latency and lower throughput. Recently, fast-adaptive CA-SCL decoders are proposed in [24], named Fully Adaptive SCL (FA-SCL), Partially Adaptive SCL (PA-SCL). Both of them use the CRC to reduce the decoding processing time by gradually increasing the list size L .

Using these decoding techniques for polar codes, a potential trade-off can be achieved between throughput, latency, and decoding performance over other channel coding techniques. Also, other channel coding techniques such as LDPC codes and turbo codes suffer

from an error floor region [25]. On the contrary, polar codes do not show these error floors [26]. Therefore, polar codes with flexible, fast-adaptive SCL decoders can be used as an alternative for LDPC codes to enhance performance of wireless systems.

Since the field of data communications continues to grow rapidly due to the ever-increasing use of video, this has led to the demand of new technologies such as 4K, 8K, virtual reality and augmented reality. Millions of set-top boxes, tablets and smartphones need efficient broadcasting and reception of TV programs with minimum access time, minimum hardware processing time, maximum throughput, and best performance. Accordingly, in this paper, we propose polar codes to be utilized in DVB systems. We propose to replace the LDPC inner coding and the BCH outer coding of DVB systems by polar codes to take advantages of the benefits in the new coding technique.

The motivation in this paper is to enhance the performance of DVB systems using polar codes as FEC coding. To design an optimized polar code construction, a simple mathematical relation to get approximately the optimized SNR design point is proposed. The proposed relation avoids complex iterative calculations or long algorithms executions as in [27,28]. In addition to this, the present paper has compared the performance of the proposed optimized polar codes with LDPC codes not only in terms BER performance as in [29], but in terms of other important aspects such as the decoding latency and system throughput. In this comparison, the recently proposed FA-SCL decoding technique [24] is utilized for polar codes to achieve the desired minimum decoding latency while maintaining the error-correction performance almost unchanged. According to the obtained results, we proposed using optimized polar codes with FA-SCL decoding technique in DVB systems to enhance the overall throughput and decrease the latency. Furthermore, we utilize long code lengths which are more efficient in terms of BER performance for any DVB system, in contrast to [30–32] which only use short code lengths in mobile communications for 4G Long Term Evolution (LTE) and 5G New Radio (NR) systems.

The main novelty of our paper is proposing the optimized polar codes as FEC technique for DVB systems. To prove this, we make the following contributions: (1) We propose a new mathematical relation to get the optimized SNR design point for polar code construction. (2) We compare the optimized polar codes with LDPC codes using different code rates and showed the advantages and disadvantages in terms of encoder and decoder parameters. According to the simulation results, it is shown that polar codes have a superior performance than LDPC in terms of the decoding latency and the overall throughput. (3) We compare the optimized polar codes with LDPC and BCH codes in terms of BER, latency, throughput and proved that it can be proposed as a FEC coding technique instead of LDPC and BCH codes in wireless communication systems such as DVB systems.

The rest of the paper is organized as follows: Section 2 briefly presents an overview of polar codes construction, encoding, decoding, and some related preliminaries. Section 3 presents a derivation to get the optimized design SNR. Section 4 includes an overview for the simulation system model for simulation measurements. Section 5 presents the simulation results for optimized design point of polar codes followed by the comparison between LDPC and optimized Polar codes. Section 6 presents an overview for DVB standards describing the proposed functional block that are changed to enhance the performance of DVB systems. The system complexity comparison is discussed in Section 7. Finally, the paper is concluded in Section 8.

2. Preliminaries

Throughout this paper, the following notations are adopted. $W: X \rightarrow Y$ indicates a Binary Input Additive White Gaussian Noise (BI-AWGN) channel with input alphabet X and output alphabet Y . $W_X(x)$ is the probability that $x \in X$ is sent across W . $W(y|x)$ is the transition probability from the input x to the received output of the channel $y \in Y$. The input alphabet X are always $\{0, 1\}$, the output alphabet and the transition probabilities may be arbitrary. The notation $\mathbf{u}_1^N$ is used as shorthand for denoting a row vector $(u_1, u_2, \dots, u_N)$, $\mathbf{u}_i^j$ denotes the sub-vector $(u_i, u_{i+1}, \dots, u_j)$ from $\mathbf{u}_1^N$, $1 \leq i \leq j \leq N$. Bhat-

tacharyya parameter Z is used as a measure of reliability, where it is the upper limit on the maximum-likelihood (ML) decision error probability of the binary input channel. The mathematical asymptotic notation O has been used to describe the limiting behaviour of the function complexity.

2.1. Polar Codes Construction

Polar codes construction is an ordering process to synthetic channels according to their reliability and selecting good channels for information transmission. In this paper, Polar code is constructed using GA method for BI-AWGN channel proposed by Trifonov [18]. For the BI-AWGN original channels with noise variance σ^2 , the transition probability $W(y|x)$ can be written as:

$$W(y|x) = \begin{cases} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-x_0)^2}{2\sigma^2}}, & x = 0 \\ \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y+x_0)^2}{2\sigma^2}}, & x = 1 \end{cases} \quad (1)$$

where x_0 is the mean value for the transmitted signal x , assuming that bits are modulated using binary phase-shift keying (BPSK), then the LLR of each received symbol y is given by:

$$l(y) = \ln \frac{W(y|0)}{W(y|1)} = \frac{2yx_0}{\sigma^2} \quad (2)$$

Taking into consideration that y is a Gaussian random variable with $y \sim N(x_0, \sigma^2)$, thus $l(y)$ can be considered also as a Gaussian random variable with $l(y) \sim N\left(\frac{2x_0^2}{\sigma^2}, \frac{4x_0^2}{\sigma^2}\right)$. For a data block of N bits, N synthetic channels are created by polar code transformation with a different error probability for each sub-channel i . GA method assumes that LLR for each sub-channel i follows a Gaussian distribution constraint in every recursion step of the code tree, in which the Expected value E is half of the variance [18]. This enables one to compute only the mean value of LLR for each synthetic channel $l_N^{(i)}$, drastically reducing the complexity to obtain the reliability of each sub-channel. The recursion process is applied on the coding tree according to following calculations [18]:

$$E[l_N^{(2i-1)}] = \phi^{-1} \left(1 - \left(1 - \phi \left(E[l_{N/2}^{(i)}] \right) \right)^2 \right) \quad (3)$$

$$E[l_N^{(2i)}] = E[l_{N/2}^{(i)}] \quad (4)$$

where the expected value of the original channel $E[l_1^{(1)}] = \frac{2x_0^2}{\sigma^2}$, and:

$$\phi(t) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi t}} \int_{-\infty}^{\infty} \tanh \frac{u}{2} e^{-\frac{(u-t)^2}{4t}} du, & t > 0 \\ 1, & t = 0 \end{cases} \quad (5)$$

However, the exact calculation of complex integration in $\phi(t)$ requires a high computational complexity. Therefore, an approximation function of $\phi(t)$, denoted by $\psi(t)$ is introduced, as follows [33]:

$$\psi(t) = \begin{cases} e^{-0.4527t^{0.86}+0.0218}, & t \leq 10 \\ \sqrt{\frac{\pi}{t}} e^{-\frac{t}{4}} \left(1 - \frac{10}{7t} \right), & t > 10 \end{cases} \quad (6)$$

which is often used to simplify the construction of polar codes.

2.2. Polar Codes Encoder

Polar code is a linear block code subspace of the binary vector space $\mathbb{F}_2^N$, which is an (N, K) block code of length N over the binary Galois field $\mathbb{F}_2$, and forms a K -dimensional vector subspace of $\mathbb{F}_2^N$. Polar codes encoder adds frozen bits and transforms the initial

information message vector M of length K into a codeword vector C of length N , where the fraction $R = K/N$ is called the code rate.

On the transmitter side, the codeword is created by inserting the information bits in the positions corresponding to noiseless channels, while frozen bits are allocated at the positions corresponding to pure-noisy channels. These bits are transformed using generator matrix G and transmitted over the communication channel. The location selection for frozen bits in the codeword depends on polar code construction. Therefore, the encoder of the polar codes can be mathematically represented as follows:

$$C = M' G \quad (7)$$

where M' is the message vector with length N bits consisting of K information bits and $N - K$ frozen bits, and the generator matrix G is defined as follows:

$$G = G_m^{\otimes n} \quad (8)$$

which has been constructed using Kronecker power denoted by $\otimes^n$ for $m \times m$ binary kernel matrix G_m , where $n = \log_m N$, m is the base number. For 2×2 polarizing binary kernel matrix $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ proposed by Arikan, the generator matrix G is as follows:

$$G = G_2^{\otimes n} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & \cdots & \cdots & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & \cdots & \cdots & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & \cdots & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & \cdots & \cdots & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & \cdots & \cdots & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & \cdots & \cdots & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & \cdots & \cdots & 1 & 1 & 1 & 1 \end{bmatrix} \quad (9)$$

2.3. Polar Codes Decoder

At the receiver side, the initial LLR of a received signal is measured using the same definition used in Equation (2) that is, $l(y) = \ln [W(y|0)/W(y|1)]$, where $W(y|0)$, $W(y|1)$ are measured practically using the distance between the position of the received bit and ideal position of bits (0, 1). SCD generally starts from the vector l that consists of all initial LLRs of the received signals and ends with the decoded information bits. This can be done by recursively calculating the intermediate LLRs and broadcasting of the decision bits and frozen bits [1,11] as shown in Figure 1, where red arrows describe the flow of decision bits. The intermediate LLRs are calculated by the recursive formulas corresponding to the polarizing kernel matrix used in polar code construction.

The recursive formulas of 2×2 binary kernel matrix $G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ have been proved by Arikan as follow [1]:

$$l_N^{(2i-1)}(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{2i-2}) = 2 \tanh^{-1} \left[\tanh \left(\frac{l_1}{2} \right) \tanh \left(\frac{l_2}{2} \right) \right] \quad (10)$$

$$l_N^{(2i)}(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{2i-1}) = (-1)^{\hat{u}_i} l_1 + l_2 \quad (11)$$

and $l_1 = l_{\frac{N}{2}}^{(i)}(\mathbf{y}_1^{\frac{N}{2}}, \hat{\mathbf{u}}_{1,o}^{2i-2} \oplus \hat{\mathbf{u}}_{1,e}^{2i-2})$, $l_2 = l_{\frac{N}{2}+1}^{(i)}(\mathbf{y}_{\frac{N}{2}+1}^N, \hat{\mathbf{u}}_{1,e}^{2i-2})$ for shorthand, and using the symbols o, e to symbolize for $u \equiv 0, 1 \pmod{2}$, respectively.

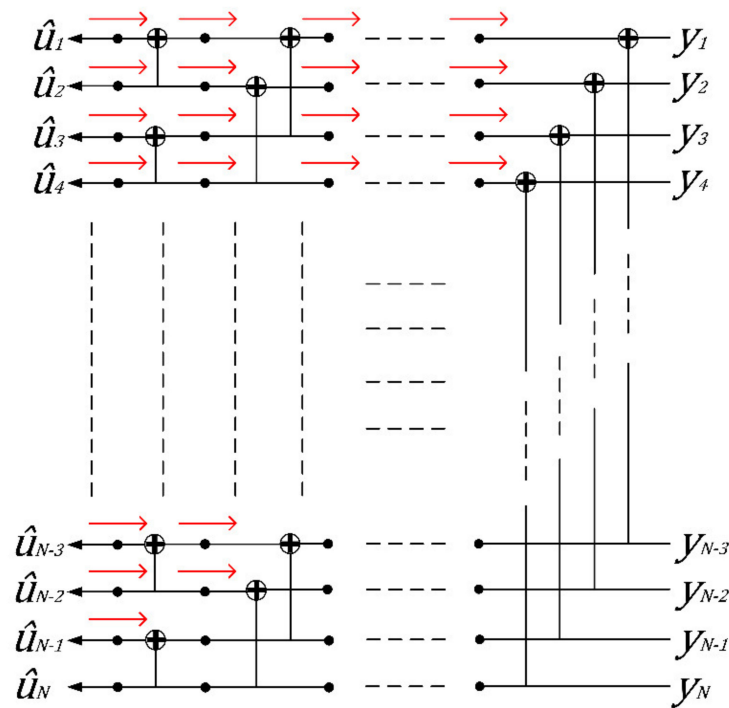


Figure 1. Successive cancellation decoding.

The decision bit $\hat{u}_i$ is determined from frozen bits, otherwise it is given by:

$$\hat{u}_i = \begin{cases} 0, & \text{if } l_N^{(i)}(\mathbf{y}_1^N, \hat{\mathbf{u}}_1^{i-1}) \geq 0 \\ 1, & \text{otherwise} \end{cases} \quad (12)$$

SCD performance has been improved by using the SCL decoder, which assumes for each decoded decision bit, the two possibilities of being decoded as 1 or 0. This process divides the current decoding path into two new paths, one for each possibility. By using multiple SCDs over the same code tree, the chance of finding the correct decoding path is significantly improved. Here the number of SCDs components is referred to as the list size L .

Repeatability of this process for each decision bit increases the number of paths exponentially, which creates a huge decoding tree. Therefore, a pruning criterion has been used to limit the number of possibilities. This criterion has been chosen in [34] to be the smallest computed LLR-based path metric, which is properly normalized for fairer comparisons of the partial paths having different lengths and be able to capture the reliabilities of the associated partial paths.

Moreover, SCL is improved by adding CRC bits K_{CRC} to information bits K_{info} during the construction of the polar code frame and before adding $N - K$ frozen bits during the encoding process of information and CRC K bits, as shown in Figure 2, and performing a CRC check on the chosen paths during the decoding process to detect the wrong paths that do not pass the CRC check and discards it.

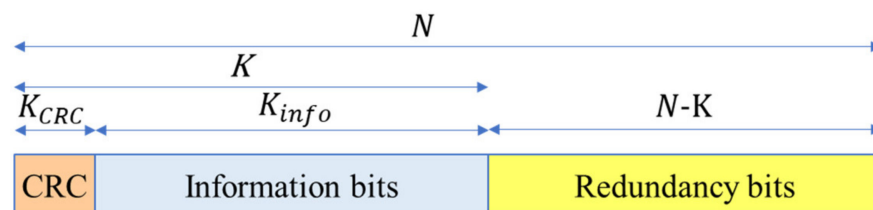


Figure 2. Polar code Frame after adding CRC bits.

After this filtration, the most likely path among the remaining correct ones is selected. The computational complexity of the decoder is not changed due to CRC efficiently computed [35]. This adding of CRC bits modifies the definition of code rate R to be the ratio between the number of information bits K_{info} and the code length N . Furthermore, the decoding complexity has been reduced by using an adaptive SCL decoder for polar codes with CRC [36].

The adaptive SCL decoder presented in [36] is summarized in the flowchart shown in Figure 3. It initially uses list size ($L = 1$) which is equivalent to simplified SC decoder, and then iteratively doubles L (if there is no correct path passing CRC), until L reaches a predefined number L_{max} , and outputs the path with the highest probability and exits the decoding process.

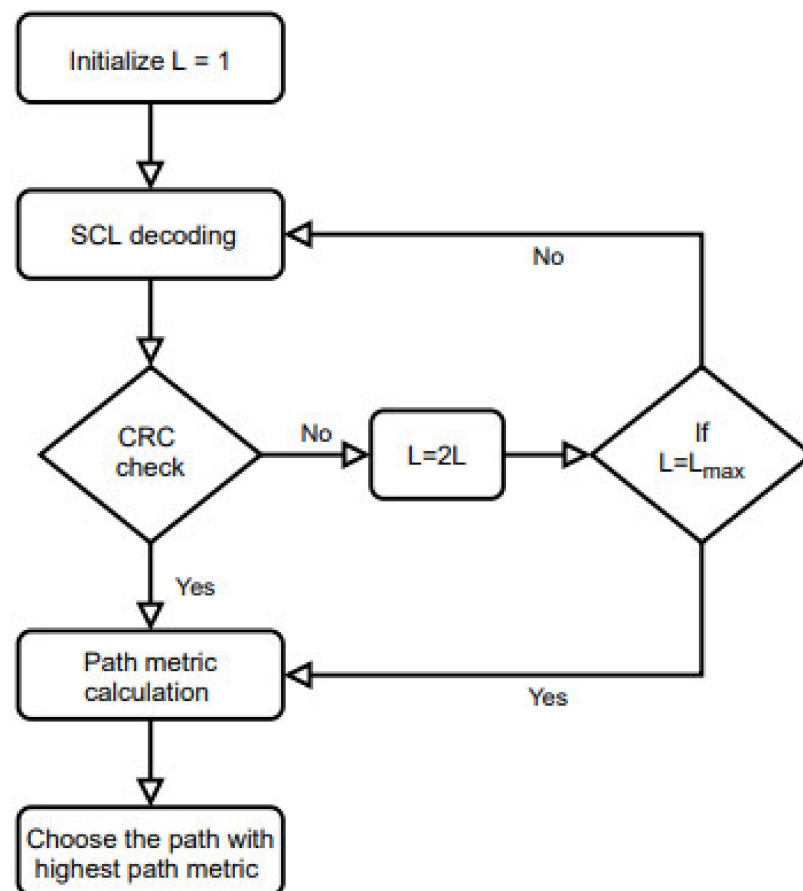


Figure 3. Adaptive SCL Decoder flowchart.

3. Design Point Selection Method

This section discusses a selection method for the optimized SNR design for the AWGN channel. Polar code construction is dependent on the selected value of $2x_0^2/\sigma^2$ as shown before in Equations (3) and (4), which is related to $\text{SNR} = x_0^2/2\sigma^2$. However, for AWGN channels, the Bhattacharyya Z parameter can be calculated as follows:

$$\begin{aligned}
 Z &= \int_{-\infty}^{\infty} \sqrt{W(y|0)W(y|1)} dy \\
 &= \int_{-\infty}^{\infty} \sqrt{\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y-x_0)^2}{2\sigma^2}} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y+x_0)^2}{2\sigma^2}}} dy \\
 &= e^{-\frac{x_0^2}{2\sigma^2}}
 \end{aligned} \tag{13}$$

This equation demonstrates that polar code construction based on Bhattacharyya parameter depends also on the designed SNR $= x_0^2/2\sigma^2$ value, which is related to the mean value for the transmitted signal energy x_0 , and the noise standard deviation σ . The ratio of the standard deviation σ to the mean value x_0 defines the coefficient of variation C_v , also known as relative standard deviation (RSD). It measures the dispersion in the probability distribution and shows the correlation between the average value and the amount of variation around it. The main idea of polar codes is the creation of synthetic channels, and each synthetic channel becomes either a pure-noise channel or noise-free channel. Our target is to construct good polar codes with optimum channel polarization with respect to any change in RSD at the design point. Thus, the rate of change in the channel reliability measured by Bhattacharyya parameter Z with respect to RSD should be maximized. To carry out this maximization, we compute Bhattacharyya parameter rate of change as the first derivative of Z with respect to C_v as follows:

$$\frac{dZ}{dC_v} = \frac{1}{C_v^3} e^{-\frac{1}{2C_v^2}} \quad (14)$$

Thus, the second derivative of Bhattacharyya parameter Z equals to zero as follows:

$$\frac{d^2Z}{dC_v^2} = \frac{-3}{C_v^4} e^{-\frac{1}{2C_v^2}} + \frac{1}{C_v^6} e^{-\frac{1}{2C_v^2}} = 0 \quad (15)$$

Accordingly, the optimum design value of the coefficient of variation equals:

$$C_v = \frac{1}{\sqrt{3}} \quad (16)$$

This mathematical relation is used to get the optimized SNR design point that is required to construct the optimized polar code according to its code-rate. Moreover, in Figure 4, it is observed that Z is an increasing function with an inflection point at $C_v = \frac{1}{\sqrt{3}}$ from concave up to concave down which coincides with Equation (16).

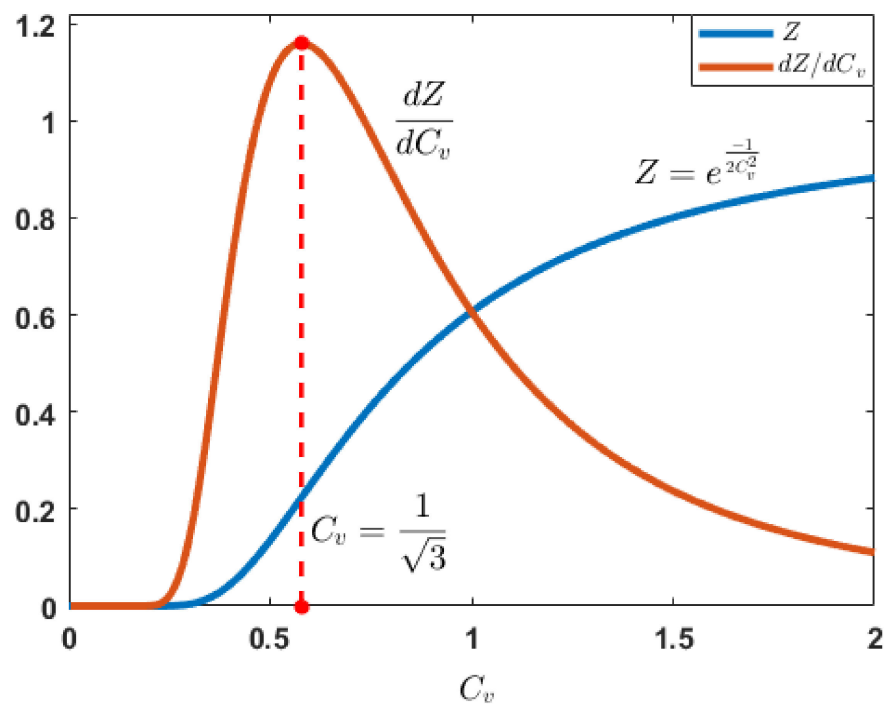


Figure 4. Bhattacharyya parameter optimized design point for AWGN channel.

To confirm the result of the proposed mathematical relation, a simulation comparison for polar code construction for different coefficient of variation C_v is established in simulation results section.

4. System Model

The description of the polar code system model used in the simulation, is shown in Figure 5.

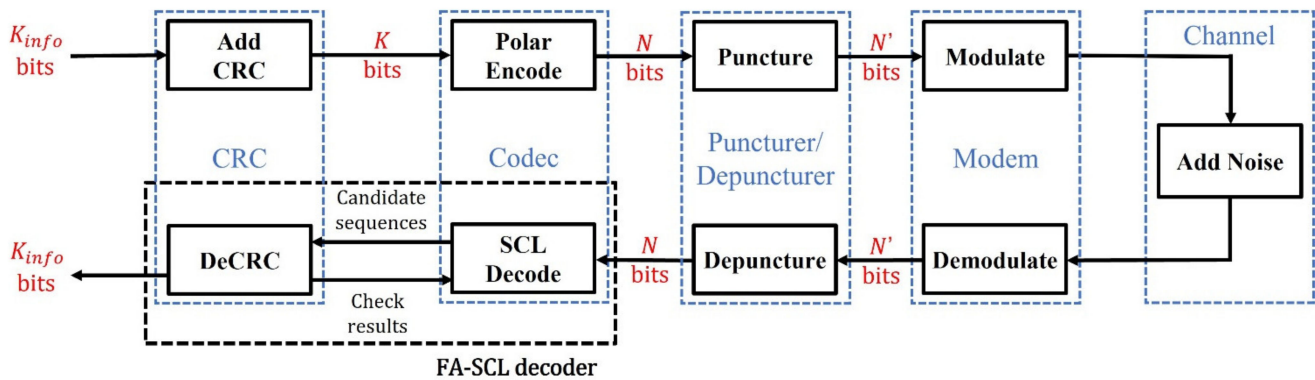


Figure 5. Block Diagram for modules and tasks in simulation for polar code system model.

To understand the organization of the parameters in the simulator, it is important to be aware of the simulator structure. The simulation contains a set of modules (CRC, Codec, Puncture, Modem, channel and monitor as shown with dashed blue boxes). A module can contain one or more tasks (with solid black boxes). A task can be assimilated to a process which is executed at runtime. The transmitter blocks start from a source which generates K_{info} information bits and CRC bits can be concatenated to the information bits to help the decoding process to know if the decoded bit sequence is valid or not. The used CRC polynomial in the simulation is the popular CRC32, which has been mentioned in [37], defined as follows:

$$g_{CRC32}(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1 \quad (17)$$

with hexadecimal representation 0x04C11DB7. After that, K information bits are encoded by introducing some frozen bits to the binary sequence and create a frame with N bits. The encoder output includes $N - K$ redundancy bits to overcome the effects of the channel. If necessary, a puncturer module can be used to match between encoder output $N = m^n$ and frame size N' . The puncture selects only coded bits which have the fewest number of stopping trees criteria to be punctured [5]. After that, the modulator transforms a frame with a sequence of bits into a suitable form to be ready to be transmitted over the physical medium. This physical medium is represented by communication channel, which randomly adds noise like AWGN channel. The receiver blocks are the reverse blocks of the transmitter which try to decode the noisy frames. The depuncture estimates LLRs of these punctured bits by setting a pre-fixed value in the puncturing algorithm for all of them [5]. The DeCRC is used as defined in the flowchart of adaptive SCL decoder shown in Figure 3 and to remove the added CRC bits at the end.

The process of transmission and emitting frames continue until a fixed number of frame errors are achieved. A frame error occurs when the receiver decoded frame differs from the transmitter original frame. Consequently, when the SNR increases, the number of frames to be simulated increases as well as the overall simulation time. According to the principle of power conservation and assuming ideal Nyquist filter with zero roll-off factor, the relationship between SNR and E_b/N_0 can be described as follows:

$$SNR = R \times \log_2 M \times \frac{E_b}{N_0} \quad (18)$$

where $R \log_2 M$ is the number of information bits per symbol, which might be influenced by the size of the modulation alphabet (M -ary signaling) or the code rate R . To unify the comparison criteria between different modulations, E_b/N_0 is used in all simulations.

The description of the LDPC code system model used in simulation is similar to polar code system model. However, the LDPC code uses the optimized encoding matrices defined in DVB-S2 standard [38], and belief propagation (BP) horizontal layered algorithm [39] is used for decoding, which exchanges soft-information iteratively between variable and check nodes, and implementation of the decoder algorithm uses Attenuated Min-Sum (AMS) update rule [40]. The main module of the FEC (Polar/LDPC) code simulation measurements is summarized below in Algorithm 1.

Algorithm 1 Polar / LDPC-based simulated transceiver system

INPUT: Code type (Polar or LDPC), Frame Size N' , Number of information bits K_{info} , Max. List Size L_{max} , Max. iterations i_{max} , CRC Size, CRC polynomial (g_{CRC}), Optimum coefficient of variation C_v , $min E_b/N_0$, step E_b/N_0 , and $max E_b/N_0$, Number of simulated frames N_f

OUTPUT: BER, encoding latency, decoding latency and overall throughput

STEPS:

1. **For** $E_b/N_0 = min E_b/N_0$: step E_b/N_0 : $max E_b/N_0$
 2. **For** $j = 1 : N_f$
 3. **Get** K_{info} information bits for frame j
 4. **if** code type = Polar
 5. **Concatenate** CRC bits generated using g_{CRC} to the information bits
 6. **Construct** polar code using GA technique with optimum design value of the coefficient of variation C_v using Equation (16), where $K = K_{info} + \text{CRC bits}$, and $max N = 2^n < N'$
 7. **Encode** using Polar code for $(K_{info} + \text{CRC})$ bits and generate N encoded bits
 8. **if** $N' \neq N$
 9. **Use** a puncturing pattern
 10. **end if**
 11. **elseif** code type = LDPC
 12. **Encode** using LDPC code for K_{info} bits and generate N' encoded bits
 13. **end if**
 14. **Modulate** the frame size N' bits for the transmission signal form
 15. **Pass** the modulated data through the channel
 16. **Demodulate** the received signal to the frame size N' bits
 17. **if** code type = Polar
 18. **if** $N' \neq N$
 19. **Remove** the puncturing bits
 20. **end if**
 21. **Decode** using FA-SCL decoder with max. list size L_{max} for N bits to get K bits
 22. **Remove** CRC bits to get K_{info} decoded bits
 23. **elseif** code type = LDPC
 24. **Decode** using LDPC decoder with max. iteration i_{max} for N bits to get K_{info} bits
 25. **end if**
 26. **end For**
 27. **Calculate** the performance metrics like BER, encoding/decoding latency and overall throughput
 28. **end For**
-

It is worth mentioning that since Release 16 (released in 2017) Third Generation Partnership Project (3GPP) has been interested in using satellites in the development of 5G networks by considering and integrating SatCom technology in standardization [41]. However, 3GPP uses other non-3GPP standards for the communications between the ground station and satellites. Such non-3GPP standards are the DVB-S2, DVB-S2X which were considered in this paper. This makes our work a potential candidate to be considered in the satellite communication aspect of 3GPP future standards.

5. Simulation Results

In this section, simulation results are implemented using A Fast Forward Error Correction Toolbox (AFF3CT) simulator and a library dedicated to channel coding [42]. During the simulations, all the error performance, throughput and latency measurements have been obtained on a single core of an Intel i7-4510U Central Processing Unit (CPU), which is based on the Haswell architecture and is manufactured in 22 nm with a base clock frequency of 2 GHz and a maximum turbo frequency of 2.6 GHz. The description has been compiled on Windows with the C++ Clang compiler for AFF3CT 2.3.5 version.

The number of simulated frames is up to 250,000 frames with typically 100 frame errors to stop the simulation, the transmitted bits are modulated using Binary Phase Shift Keying (BPSK). The channel model is AWGN channel, with zero mean and variance $N_0/2$. The simulations results have been done for different code rates R .

5.1. Optimized Design Point for Polar Codes

In this sub-section, a simulation comparison for polar code construction using GA method for different coefficient of variation C_v is established. Table 1 summarizes the used simulation parameters.

Table 1. Simulation parameters for polar code.

Construction	GA Technique
Decoder	FA-SCL
Max. List size (L_{max})	$L_{max} = 32$
Implementation	Full
CRC Size in bits	32
Codeword size (N)	65,536
Frame size (N')	64,800
Modem Type	BPSK
Channel Type	AWGN

Figure 6 shows the BER performance versus E_b/N_0 at code-rate $R = 3/5$ for various coefficients of variation C_v . It is observed that the minimum BER performance happened at coefficient of variation $C_v = \frac{1}{\sqrt{3}} = 0.577$. In addition, to check the validity of the relation proved in Equation (16), BER performance is evaluated versus coefficient of variation C_v for different code-rates R .

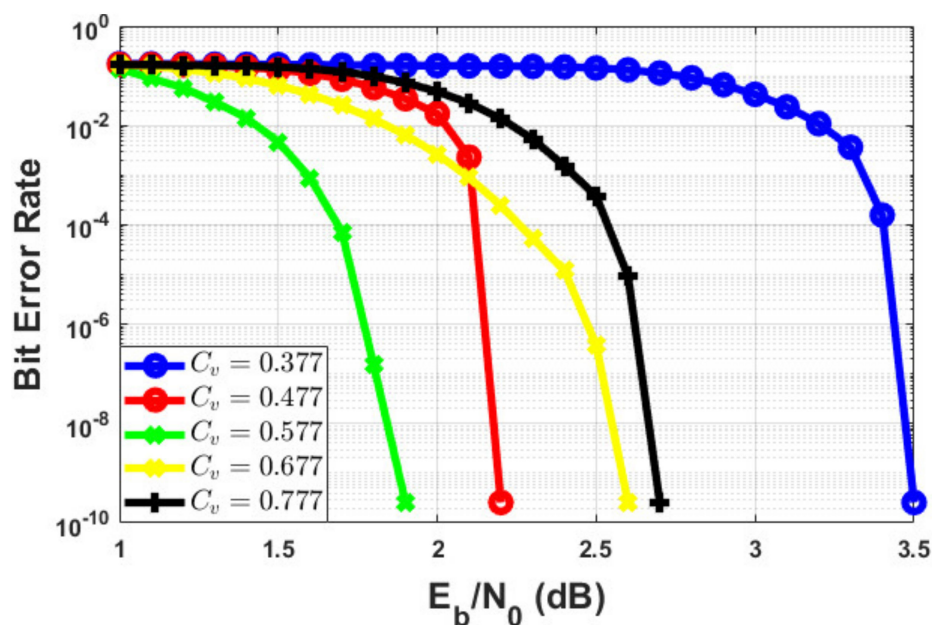


Figure 6. Polar code BER versus E_b/N_0 for code rate $R = 3/5$ and different coefficient of variation C_v .

The results shown in Figure 7 demonstrate that the coefficient of variation C_v that achieves minimum BER performance is very close to the calculated $C_v = \frac{1}{\sqrt{3}}$. This small variation is expected since the GA method approximates the exact transition probability of each binary input channel to overcome difficulties in calculating the actual values as shown in Equations (5) and (6). However, in our mathematical calculations we assume that the exact relation is used. Therefore, this value of C_v can be used in the construction of polar code with GA method after a small fine tuning for each code rate as we did in the upcoming simulation results for optimized polar codes.

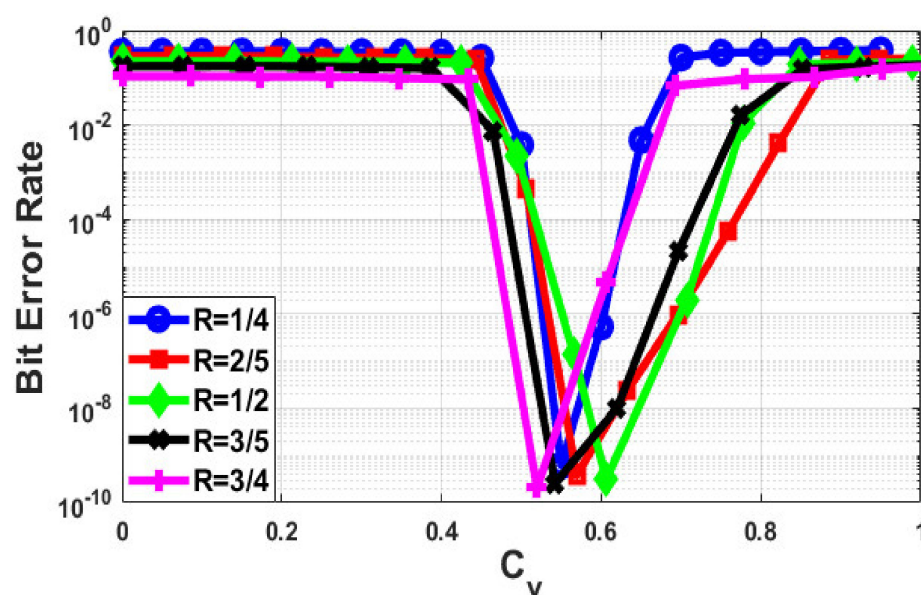


Figure 7. Polar code BER versus coefficient of variation C_v for different code rates R .

5.2. LDPC and Polar Codes Comparison

In this sub-section, optimized polar code performance is compared with the well-known LDPC code. First, the comparison is only restricted with respect to LDPC code, because it is the main inner error correcting technique for all advanced DVB systems. Table 2 summarizes the simulation parameters for both channel coding techniques.

Table 2. Simulation parameters for polar and LDPC codes.

Code Type	Polar Code	LDPC Code
Construction	GA technique	DVB-S2 standard
Decoder	FA-SCL	BP horizontal layered
Max. List size (L_{max})/ Max. iterations (i_{max})	$L_{max} = 32$	$i_{max} = 50$
Implementation	Full	AMS
Frame size (N')	64,800	64,800
Puncturing pattern	yes	No
Modem Type	BPSK	BPSK
Channel Type	AWGN	AWGN

Figures 8 and 9 show the error performance comparison between LDPC used in DVB-S2 standard and optimized polar codes for normal frame ($N = 64,800$) and short frame ($N = 16,200$), respectively. The short frame is more efficient for minimum latency interactive applications. From the results, it is concluded that optimized polar codes can achieve a comparable performance to LDPC where the difference of E_b/N_0 for $BER = 10^{-7}$ in worst case is about 0.3 dB. Moreover, optimized polar code has better performance than LDPC for code rate 5/6 for short frames, the reason for this performance degradation of LDPC code is due to the method of composition for sparse parity check matrix used in LDPC code, which depends on empirical approaches and iterative algorithms, which is not necessarily optimized for all code rates.

On the other hand, Tables 3 and 4 present the measured encoding and decoding average latency comparisons between LDPC code and optimized polar code for normal frame and short frame at QEF selected E_b/N_0 points for reception conditions of different mentioned code rates. The selected E_b/N_0 is the first value of E_b/N_0 at which PER is less than 10^{-7} for both coding techniques at each code rate. It is worth to note that short frame with code rate $R = 9/10$ is not applicable according to DVB-S2 standard, and for that no measured values for it.

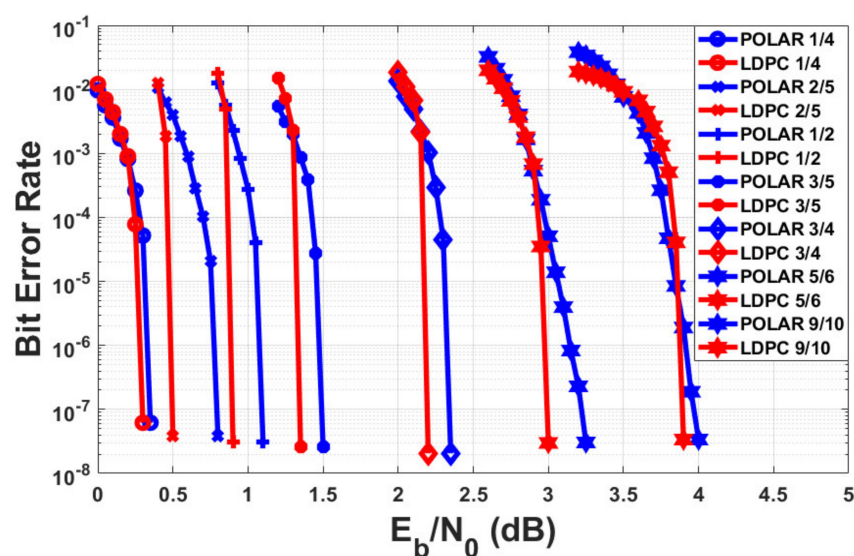


Figure 8. BER versus E_b/N_0 for LDPC & optimized polar codes for normal frame with $N = 64,800$ and different code rates R in DVB-S2.

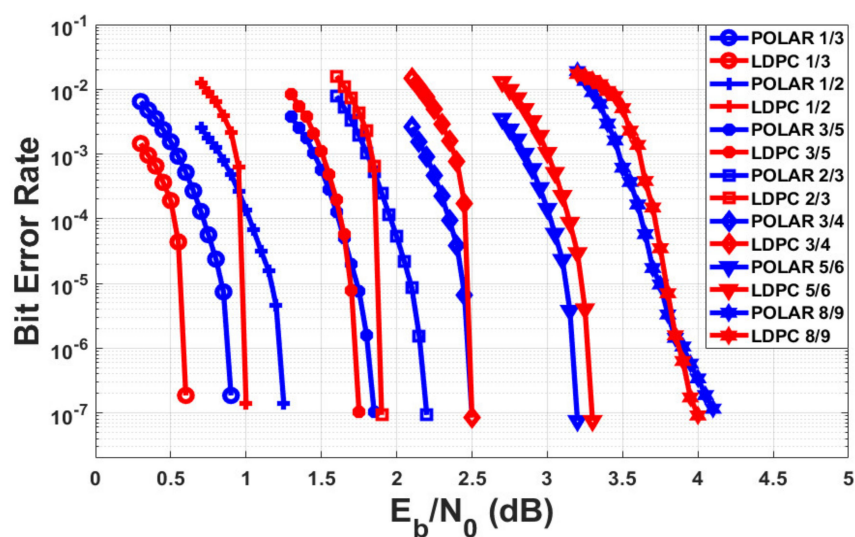


Figure 9. BER versus E_b/N_0 for LDPC & optimized polar codes for short frame with $N = 16,200$ and different code rates R in DVB-S2.

Table 3. Encoding average latency comparisons between LDPC code and optimized Polar code for normal frame and short frame. (bold cells represent the best values).

Frame Type	Normal Frame (64,800) @ QEF		Short Frame (16,200) @ QEF	
Code rate R	LDPC Latency (ms)	Polar Latency (ms)	LDPC Latency (ms)	Polar Latency (ms)
1/4	0.312	0.999	0.066	0.202
1/3	0.399	0.985	0.092	0.200
2/5	0.442	0.836	0.109	0.201
1/2	0.488	0.825	0.097	0.202
3/5	0.630	0.822	0.162	0.205
2/3	0.553	0.845	0.134	0.203
3/4	0.618	0.812	0.132	0.203
4/5	0.662	0.837	0.133	0.203
5/6	0.673	0.815	0.148	0.204
8/9	0.640	0.824	0.154	0.203
9/10	0.641	0.819	-	-

Table 4. Decoding average latency comparisons between LDPC code and optimized Polar code for normal frame and short frame. (bold cells represent the best values).

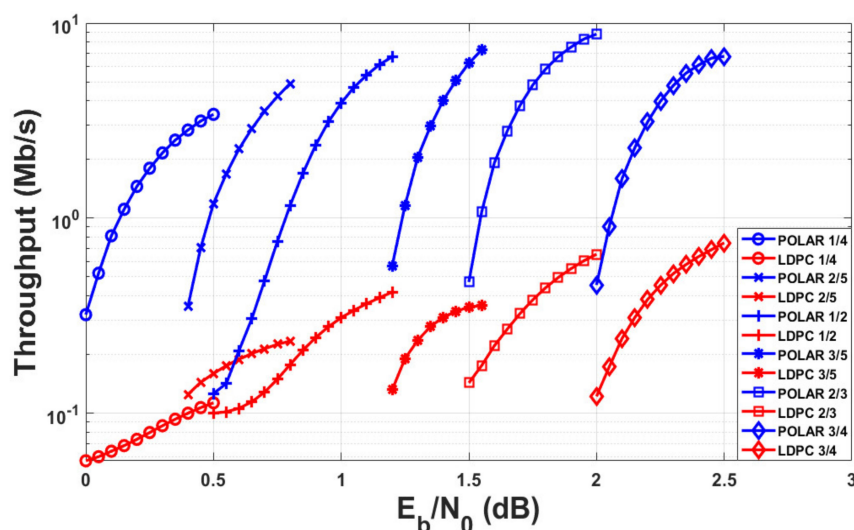
Frame type	Normal Frame (64,800) @ QEF		Short Frame (16,200) @ QEF	
	Code Rate R	LDPC Latency (ms) Polar Latency (ms)	LDPC Latency (ms) Polar Latency (ms)	
	1/4	71.023 0.519	13.184 0.108	
	1/3	65.079 0.587	11.223 0.111	
	2/5	56.459 0.540	10.468 0.120	
	1/2	45.587 0.484	8.129 0.108	
	3/5	44.816 0.506	8.803 0.118	
	2/3	39.091 0.546	6.636 0.121	
	3/4	33.115 0.521	5.129 0.114	
	4/5	28.185 0.505	3.742 0.114	
	5/6	26.395 0.540	3.913 0.118	
	8/9	16.411 0.501	2.533 0.105	
	9/10	15.383 0.474	- -	

It is observed from Table 3 that the encoding latency for both channel coding techniques is almost small, and from Table 4 that the decoding latency of polar code is at least twenty-five times less than the decoding latency of LDPC code for both frame types. Moreover, it is observed from both Tables 3 and 4 that LDPC code latency depends completely on code rate related to the parity check matrix. Furthermore, it is shown that the decoding latency decreases as the code rate increases, while polar code latency is independent of code rate. Decoding latency is the main factor, which dominates in the overall system latency T , which is defined as the total sum of all measured latencies for all modules in the full communication chain. The overall latency T is related to system throughput δ as follows:

$$\delta = \frac{K_{info} \times S}{T} \quad (19)$$

where S is the number of simulated frames.

Figures 10 and 11 show the comparison between simulation throughput of the LDPC code used in DVB-S2 and optimized polar code with different code rates for normal frame and short frame, respectively. As can be seen, the throughput of optimized polar code is higher than that for LDPC code for the same SNR values, which indicates the effectiveness of replacing LDPC code by optimized polar code.

**Figure 10.** Throughput versus E_b/N_0 for LDPC and optimized polar codes using normal frame with $N = 64,800$ and different code rates R in DVB-S2.

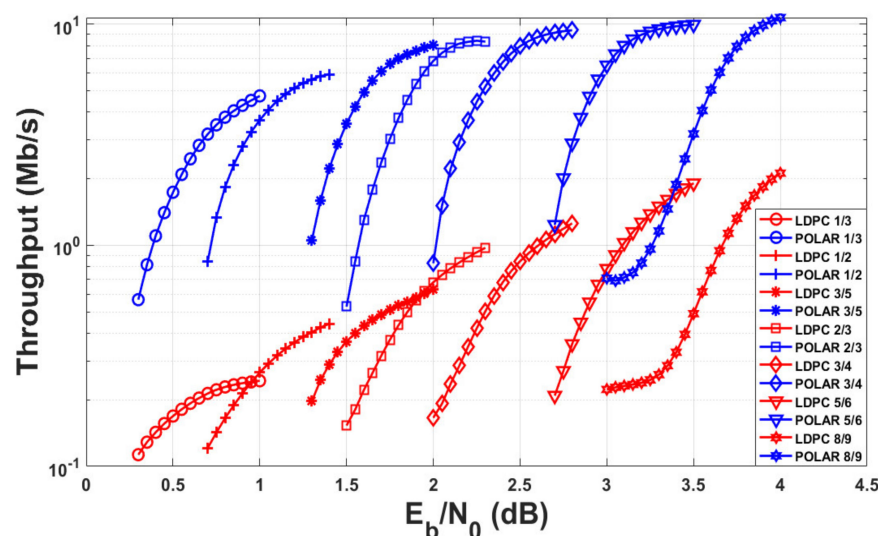


Figure 11. Throughput versus E_b/N_0 for LDPC and optimized polar codes using short frame with $N = 16,200$ and different code rates R in DVB-S2.

Finally, Table 5 summarizes all simulation qualitative results of a comparison between LDPC code and optimized Polar code. The point of decoder latency is really significant for all interactive communication systems, especially for satellite systems which are already affected by the high propagation delay of more than 250 milliseconds (ms) for one-way transmission channel or 500 ms for a round trip with respect to geostationary satellites at about 36,000 kilometers above the equator.

Table 5. Qualitative simulation results summarization.

Code Type	Polar Code	LDPC Code
BER Performance	Good	Good
Encoder Latency	Low	Low
Decoder Latency	Low	High
System Throughput	High	Low

6. Proposed DVB System Using Polar Codes

Digital Video Broadcasting (DVB) is a collection of standards produced by European Telecommunications Standards Institute (ETSI) under the auspices of the DVB project. It aims to unite the specifications of the devices used to transmit and receive multimedia via cable (DVB-C), terrestrial (DVB-T) or satellite (DVB-S). DVB Systems use different modulation schemes and several channel coding techniques, inner and outer coding. Inner coding is utilized as the main error correcting coding like Turbo codes or LDPC codes. However, outer coding is utilized to remove residual errors after inner coding like Reed-Solomon (RS) or BCH codes. Over the years, DVB systems are regularly upgraded to meet the increasing demand for more bandwidth/bitrate and new advanced interactive services. The FEC schemes used in DVB systems have evolved over the last two decades.

In the beginning, DVB systems were using convolutional code for inner coding and RS coding for outer coding [43–45]. As the size of carriers for broadcasting increased, the need for a powerful FEC scheme was recognized. Moreover, DVB created another standard in 1999 called Digital Video Broadcasting-Digital Satellite News Gathering (DVB-DSNG) [46], which added and standardized 8PSK and 16QAM modulation (more efficient than QPSK but requiring more link margin) to be used for professional applications especially news gathering live show. However, the DVB-DSNG does not support any enhancement in channel coding, which is applied to protect information bits from errors and provide FEC with different code rate ratios. Therefore, LDPC and BCH are adopted by the DVB committee as the inner and outer coding, respectively. These codes are now part of the DVB-S2, DVB-T2, DVB-C2, and DVB-S2X standards [38,47–49]. The increase of the demand on interactive services leads to more need for optimizing more aspects like decoding latency and complexity.

In the proposed enhanced model, as shown in Figure 12, optimized polar code is introduced into DVB systems to replace both BCH code and LDPC code in the FEC coding block in these systems standards, where optimized polar codes can achieve a comparable error performance to LDPC code

with less latency and more throughput, and as polar codes are not suffering from error floor [26], hence no need to use BCH for outer coding.

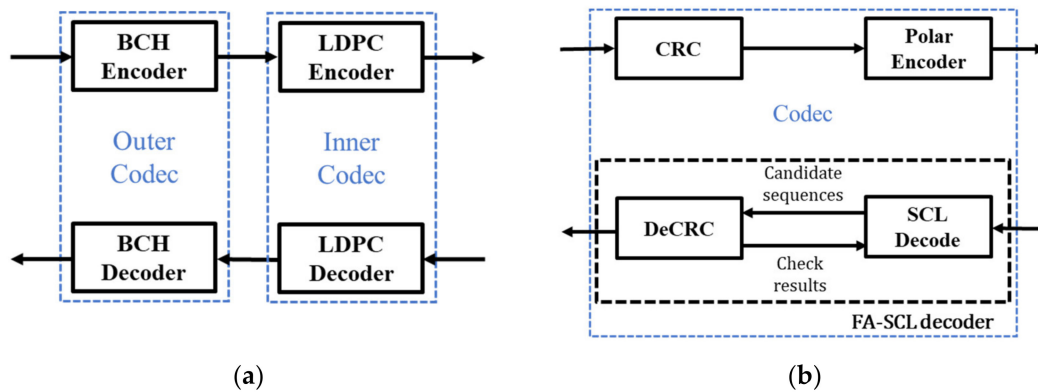


Figure 12. Forward Error Correction encoder and decoder for: (a) Original DVB systems.; (b) Proposed DVB systems.

To make sure of that, a comparison between simulation results of LDPC code concatenated with BCH code as described in DVB-S2 standard [38,50] and optimized polar codes using AFF3CT simulator has been executed. The simulation setting is compliant with DVB-S2 standard [38] with release (ETSI EN 302307-1) as an example for comparison. Figure 13 shows the packet error rate (PER) versus E_s/N_0 for coded blocks of 64,800 bits (normal frame) with QPSK modulation same as in DVB-S2 standard for different code rates in the AWGN channel. From these results, it is concluded that optimized polar codes can achieve a comparable error performance to BCH+LDPC where the difference ranges between 0.3 dB to 0.6 dB.

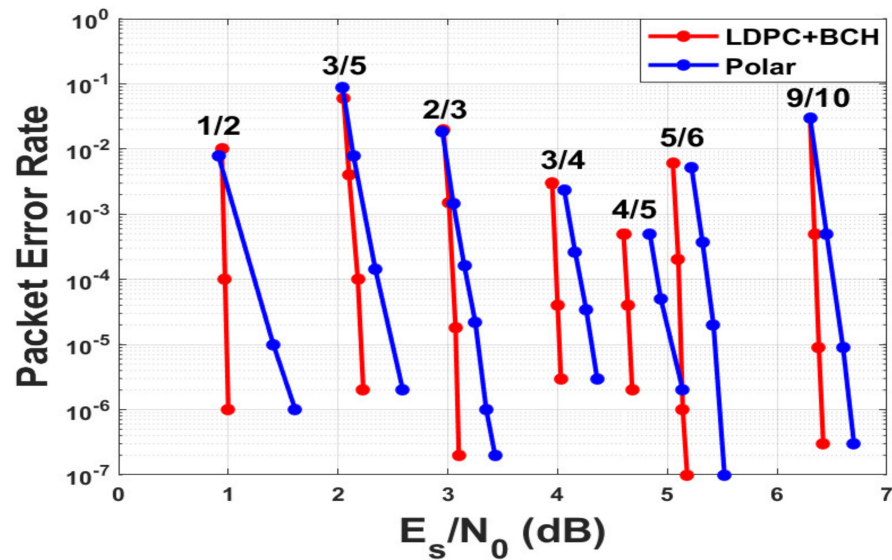


Figure 13. PER versus E_s/N_0 for DVB-S2 FEC (BCH+LDPC) and optimized polar codes using normal frame and different code rates.

Table 6 presents measured average latency comparisons using AFF3CT simulator between LDPC and BCH codes as used in DVB-S2 and optimized polar code for normal frame at QEF points for reception conditions on an AWGN channel for different mentioned code rates. According to these latency simulation results, optimized polar code overall processing time is smaller than LDPC and BCH codes, with a powerful decoding latency enhancement.

Table 6. Average latency comparisons between (LDPC+BCH) codes and optimized Polar code for normal frame at different code rates R . (bold cells represent the best values).

Code Rate	BCH + LDPC Encoding Latency (ms)	BCH + LDPC Decoding Latency (ms)	Polar Encoding Latency (ms)	Polar Decoding Latency (ms)	Encoding Latency Enhancement	Decoding Latency Enhancement
1/4	2.7	74.2	1.00	0.52	2.7	142.8
1/3	3.9	69.8	0.98	0.59	3.9	118.0
2/5	4.6	62.1	0.84	0.54	5.5	115.1
1/2	5.7	52.7	0.83	0.48	6.9	108.9
3/5	7.2	54.1	0.82	0.51	8.7	106.9
2/3	7.8	49.4	0.84	0.55	9.2	90.4
3/4	8.8	44.7	0.81	0.52	10.8	85.7
4/5	9.4	40.5	0.84	0.51	11.2	80.2
5/6	8.3	37.1	0.81	0.54	10.2	68.8
8/9	7.3	25.9	0.82	0.50	8.9	51.7
9/10	7.4	25.0	0.82	0.47	9.1	52.6

7. System Complexity Comparison

Polar coding transformation can be represented as a graph with $N[1 + \log_2 N]$ variables. This graph contains $[1 + \log_2 N]$ levels with N variables in each level. Computation starts at the source level and can be executed from one level to another. Thus, the total number of calculations needed for the polar code is $N[1 + \log_2 N]$; the encoding and SCD complexity of polar code is $O(N \log_2 N)$ as proved in [1], CA-SCL decoding complexity is $O(\bar{L}N \log_2 N)$ as proved in [22]. However, FA-SCL decoding complexity is $O(\bar{L}N \log_2 N)$, where $\bar{L}$ is the mean of L under L_{max} . It is seen that under $L_{max} = 32$, the mean $\bar{L} = 2.04$, which gives about 16 times complexity reduction with respect to CA-SCL decoding, but with similar performance [36].

On the other hand, The encoding complexity of LDPC code is $O(N + g^2)$, where g is the representation gap in the parity-check matrix in approximate lower triangular form [51] which differs for each code rate, and the decoding complexity is $O(i_{max}(N\rho + (N - K)\gamma))$, where i_{max} is the maximum number of iterations and ρ, γ represent the average degree of the variable and parity nodes respectively in the parity check matrix [52], which also differ for each code rate. DVB-S2 QEF performance is achieved after a pre-determined number of iterations $i_{max} = 50$ [38]. Moreover, the computational complexity of BCH encoding, and decoding are $O(t\sqrt{K_{bch}})$, $O(t\sqrt{N_{bch}})$ respectively [53–55], where t is the BCH error correction capability.

Table 7 summarizes the computational complexity of different channel coding methods. It is shown that polar code complexity is less than (LDPC + BCH) codes. Because, by comparison between mean $\bar{L} = 2.04$ and $i_{max} = 50$, the polar code decoding complexity is less than LDPC code decoding complexity by twenty-five times at least. Moreover, BCH code adds another layer of complexity to LDPC code. Therefore, polar code usage reduces the complexity of DVB systems.

Table 7. Comparison of computational complexity of coding and decoding of Polar, LDPC, and BCH codes.

Code Type	Coding Complexity	Decoding Complexity
Polar Code	$O(N \log_2 N)$	$O(\bar{L}N \log_2 N)$
LDPC Code	$O(N + g^2)$	$O(i_{max}(N\rho + (N - K)\gamma))$
BCH Code	$O(t\sqrt{K_{bch}})$	$O(t\sqrt{N_{bch}})$

8. Conclusions

In this paper, a polar code encoder and decoder are presented. For polar code construction, it has been proven a relation to get the optimized design SNR point required to have the best performance. Moreover, a comparison between optimized polar codes and LDPC code is conducted. Results show that optimized polar codes achieve comparable bit error

rate performance compared to LDPC code for different code rates, while being free of error floor. Although the encoding latency of polar codes is slightly higher than LDPC codes, however, the overall encoding and decoding latency of polar codes is lower than LDPC codes by at least twenty-five times. Polar code takes advantage of low decoding latency, high throughput and flexibility provided by FA-SCL decoding of polar codes. As a result, in this paper, optimized polar code is proposed to replace the LDPC and the BCH codes in the new standards of DVB systems to enhance their performance in terms of latency and throughput. Finally, a complexity comparison show that polar code complexity is less than (LDPC + BCH) codes, which reduces DVB systems complexity.

As a future work, the performance of the proposed system can be shown for different channel models. Moreover, a hardware implementation for any DVB system after exchanging the existing encoder/decoder by optimized polar encoder/decoder can be established and field measurements can be presented. Another future research direction, we propose to replace the LDPC channel coding in 5G by optimized polar code to benefit from the low latency and high throughput of polar codes over LDPC as proved in our paper.

Author Contributions: Conceptualization, K.E.-A., R.T.E., S.E.R. and B.A.; methodology, K.E.-A., R.T.E., S.E.R. and B.A.; software, K.E.-A.; resources, K.E.-A.; writing—original draft preparation, K.E.-A.; writing—review and editing, K.E.-A., R.T.E., S.E.R. and B.A.; supervision, R.T.E., S.E.R. and B.A.; All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by The Egyptian Satellite Company (Nilesat).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Arikan, E. Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels. *IEEE Trans. Inf. Theory* **2009**, *55*, 3051–3073. [\[CrossRef\]](#)
2. Chen, P.; Xu, M.; Bai, B.; Wang, J. Design and Performance of Polar Codes for 5G Communication under High Mobility Scenarios. In Proceedings of the 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), Sydney, NSW, Australia, 4–7 June 2017; pp. 1–5.
3. Babar, Z.; Egilmez, Z.B.K.; Xiang, L.; Chandra, D.; Maunder, R.G.; Ng, S.X.; Hanzo, L. Polar Codes and Their Quantum-Domain Counterparts. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 123–155. [\[CrossRef\]](#)
4. Miloslavskaya, V. Shortened Polar Codes. *IEEE Trans. Inf. Theory* **2015**, *61*, 4852–4865. [\[CrossRef\]](#)
5. Wang, R.; Liu, R. A Novel Puncturing Scheme for Polar Codes. *IEEE Commun. Lett.* **2014**, *18*, 2081–2084. [\[CrossRef\]](#)
6. Bioglio, V.; Gabry, F.; Land, I.; Belfiore, J.-C. Multi-Kernel Polar Codes: Concept and Design Principles. *IEEE Trans. Commun.* **2020**, *68*, 5350–5362. [\[CrossRef\]](#)
7. Korada, S.B.; Sasoglu, E.; Urbanke, R. Polar Codes: Characterization of Exponent, Bounds, and Constructions. *IEEE Trans. Inf. Theory* **2010**, *56*, 6253–6264. [\[CrossRef\]](#)
8. Presman, N.; Shapira, O.; Litsyn, S.; Etzion, T.; Vardy, A. Binary Polarization Kernels From Code Decompositions. *IEEE Trans. Inf. Theory* **2015**, *61*, 2227–2239. [\[CrossRef\]](#)
9. Yao, H.; Fazeli, A.; Vardy, A. Explicit Polar Codes with Small Scaling Exponent. In Proceedings of the 2019 IEEE International Symposium on Information Theory (ISIT), Paris, France, 7–12 July 2019; pp. 1757–1761.
10. Serbetci, B.; Pusane, A.E. Practical polar code construction using generalized generator matrices. *IET Commun.* **2014**, *8*, 419–426. [\[CrossRef\]](#)
11. El-Abbasy, K.; Taki ElDin, R.F.; Elramly, S.; Abdelhamid, B. Polar codes Bhattacharyya parameter generalization. *IET Commun.* **2020**, *14*, 1718–1729. [\[CrossRef\]](#)
12. Presman, N.; Shapira, O.; Litsyn, S. Mixed-Kernels Constructions of Polar Codes. *IEEE J. Sel. Areas Commun.* **2016**, *34*, 239–253. [\[CrossRef\]](#)
13. Benammar, M.; Bioglio, V.; Gabry, F.; Land, I. Multi-kernel polar codes: Proof of polarization and error exponents. In Proceedings of the 2017 IEEE Information Theory Workshop (ITW), Kaohsiung, Taiwan, 6–10 November 2017; pp. 101–105.
14. Gabry, F.; Bioglio, V.; Land, I.; Belfiore, J.-C. Multi-kernel construction of polar codes. In Proceedings of the 2017 IEEE International Conference on Communications Workshops (ICC Workshops), Paris, France, 21–25 May 2017; pp. 761–765.
15. Bioglio, V.; Land, I. On the Marginalization of Polarizing Kernels. In Proceedings of the 2018 IEEE 10th International Symposium on Turbo Codes & Iterative Information Processing (ISTC), Hong Kong, China, 3–7 December 2018; pp. 1–5.
16. Hanif, M.; Ardakani, M. Polar Codes: Bounds on Bhattacharyya Parameters and Their Applications. *IEEE Trans. Commun.* **2018**, *66*, 5927–5937. [\[CrossRef\]](#)
17. Moriand, R.; Tanaka, T. Performance of polar codes with the construction using density evolution. *IEEE Commun. Lett.* **2009**, *13*, 519–521. [\[CrossRef\]](#)

18. Trifonov, P. Efficient Design and Decoding of Polar Codes. *IEEE Trans. Commun.* **2012**, *60*, 3221–3227. [\[CrossRef\]](#)
19. Vangala, H.; Viterbo, E.; Hong, Y. A comparative study of polar code constructions for the AWGN channel. *arXiv* **2015**, arXiv:1501.02473.
20. Benkhrouya, R.; Chana, I.; Hadi, Y. Study of the operational SNR while constructing polar codes. *Int. J. Electr. Comput. Eng. (IJECE)* **2020**, *10*, 3200–3207. [\[CrossRef\]](#)
21. Tal, I.; Vardy, A. List Decoding of Polar Codes. *IEEE Trans. Info. Theory* **2015**, *61*, 2213–2226. [\[CrossRef\]](#)
22. Niu, K.; Chen, K. CRC-Aided Decoding of Polar Codes. *IEEE Commun. Lett.* **2012**, *16*, 1668–1671. [\[CrossRef\]](#)
23. Liang, X.; Yang, J.; Zhang, C.; Song, W.; You, X. Hardware Efficient and Low-Latency CA-SCL Decoder Based on Distributed Sorting. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; pp. 1–6.
24. Léonardon, M.; Cassagne, A.; Leroux, C.; Jégo, C.; Hamelin, L.-P.; Savaria, Y. Fast and Flexible Software Polar List Decoders. *J. Signal Process. Syst.* **2019**, *91*, 937–952. [\[CrossRef\]](#)
25. Richardson, T. Error floors of LDPC codes. In Proceedings of the Annual Allerton Conference on Communication Control and Computing, Monticello, IL, USA, 1–3 October 2003; pp. 1426–1435.
26. Mondelli, M.; Hassani, S.H.; Urbanke, R.L. Unified Scaling of Polar Codes: Error Exponent, Scaling Exponent, Moderate Deviations, and Error Floors. *IEEE Trans. Inf. Theory* **2016**, *62*, 6698–6712. [\[CrossRef\]](#)
27. Sun, H.; Wang, Y.; Tian, R.; Zhao, H. A Nearly Optimal Method of Polar Code Constructions for the AWGN Channel. *IEEE Access* **2021**, *9*, 17266–17274. [\[CrossRef\]](#)
28. Dai, J.; Niu, K.; Si, Z.; Dong, C.; Lin, J. Does Gaussian Approximation Work Well for The Long-Length Polar Code Construction? *IEEE Access* **2017**, *5*, 1. [\[CrossRef\]](#)
29. Tahir, B.; Schwarz, S.; Rupp, M. BER comparison between Convolutional, Turbo, LDPC, and Polar codes. In Proceedings of the 2017 24th International Conference on Telecommunications (ICT), Limassol, Cyprus, 3–5 May 2017; pp. 1–7.
30. Balatsoukas-Stimming, A.; Giard, P.; Burg, A. Comparison of Polar Decoders with Existing Low-Density Parity-Check and Turbo De-coders. In Proceedings of the 2017 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), San Francisco, CA, USA, 19–22 March 2017; pp. 1–6.
31. Shao, S.; Hailes, P.; Wang, T.-Y.; Wu, J.-Y.; Maunder, R.G.; Al-Hashimi, B.M.; Hanzo, L. Survey of Turbo, LDPC, and Polar Decoder ASIC Implementations. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2309–2333. [\[CrossRef\]](#)
32. Pratschner, S.; Tahir, B.; Marijanovic, L.; Mussbah, M.; Kirev, K.; Nissel, R.; Schwarz, S.; Rupp, M. Versatile mobile communications simulation: The Vienna 5G Link Level Simulator. *EURASIP J. Wirel. Commun. Netw.* **2018**, *2018*, 1–17. [\[CrossRef\]](#)
33. Chung, S.-Y.; Richardson, T.; Urbanke, R. Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation. *IEEE Trans. Inf. Theory* **2001**, *47*, 657–670. [\[CrossRef\]](#)
34. Balatsoukas-Stimming, A.; Parizi, M.B.; Burg, A. LLR-Based Successive Cancellation List Decoding of Polar Codes. *IEEE Trans. Signal Process.* **2015**, *63*, 5165–5179. [\[CrossRef\]](#)
35. MacWilliams, F.J.; Sloane, N.J.A. *Codes, Designs and Perfect Codes In The Theory of Error Correcting Codes*, 1st ed.; North-Holland Mathematical Library: North-Holland, Holland; Elsevier: Amsterdam, The Netherlands, 1977; Volume 16, pp. 155–187.
36. Li, B.; Shen, H.; Tse, D. An Adaptive Successive Cancellation List Decoder for Polar Codes with Cyclic Redundancy Check. *IEEE Commun. Lett.* **2012**, *16*, 2044–2047. [\[CrossRef\]](#)
37. Wikipedia Organization, Cyclic Redundancy Check. Available online: http://en.wikipedia.org/wiki/Cyclic_redundancy_check (accessed on 15 August 2020).
38. ETSI Standard: ETSI EN 302307-1. *Digital Video Broadcasting (DVB); Second Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering and other Broadband Satellite Applications; Part I: (DVB-S2)*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 2013.
39. Pakzad, Y.P.; Nikolic, B.; Anantharam, V. High throughput low-density parity-check decoder architectures. In Proceedings of the Globecom'01 IEEE Global Telecommunications Conference, San Antonio, TX, USA, 25–29 November 2001; pp. 3019–3024.
40. Chen, J.; Dholakia, A.; Eleftheriou, E.; Fossorier, M.P.C.; Hu, X.-Y. Reduced-Complexity Decoding of LDPC Codes. *IEEE Trans. Commun.* **2005**, *53*, 1288–1299. [\[CrossRef\]](#)
41. 3GPP TR 22.822 V16.0.0 (2018-06), *The 5G System—Phase 2: 5G Enhancements, NR-based Access to Unlicensed Spectrum (NR-U), Satellite Access*; 3rd Generation Partnership Project (3GPP): Valbonne, France, 2018.
42. Cassagne, A.; Hartmann, O.; Léonardon, M.; He, K.; Leroux, C.; Tajan, R.; Aumage, O.; Barthou, D.; Tonnellier, T.; Pignoly, V.; et al. AFF3CT: A Fast Forward Error Correction Toolbox! *SoftwareX* **2019**, *10*, 100345. [\[CrossRef\]](#)
43. ETSI Standard: EN 300 744 V1.5.1. *Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 2004.
44. ETSI Standard: ETSI EN 300 429 V1.2.1 (1998-04). *Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for Cable Systems*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 1998.
45. ETSI Standard: ETSI EN 300 421 V1.1.2. *Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for 11/12 GHz Satellite Services*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 1997.
46. ETSI Standard: ETSI EN 301 210 (V1.1.1). *Framing Structure, Channel Coding and Modulation for Digital Satellite News Gathering (DSNG) and other Contribution Applications by Satellite*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 1999.

47. ETSI Standard: ETSI EN 302 755 (V1.4.1). *Frame Structure Channel Coding and Modulation for a Second-Generation Digital Terrestrial Television Broadcasting System (DVB-T2)*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 2015.
48. ETSI Standard: ETSI EN 302 769 (V1.3.1). *Frame Structure Channel Coding and Modulation for a Second-Generation Digital Transmission System for Cable Systems (DVB-C2)*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 2015.
49. ETSI Standard: ETSI EN 302307-2. *Digital Video Broadcasting (DVB); Second Generation Framing Structure, Channel Coding and Modulation Systems for Broadcasting, Interactive Services, News Gathering and other Broadband Satellite Applications: Part II: S2-Extensions (DVB-S2X)*; European Telecommunications Standards Institute (ETSI): Valbonne, France, 2014.
50. Morello, A.; Mignone, V. DVB-S2: The Second Generation Standard for Satellite Broad-Band Services. *IEEE* **2006**, *94*, 210–227. [[CrossRef](#)]
51. Richardson, T.; Urbanke, R. Efficient encoding of low-density parity-check codes. *IEEE Trans. Inf. Theory* **2001**, *47*, 638–656. [[CrossRef](#)]
52. Youssef, A.A.; Abdelhamid, B.; El-Ramly, S.H.; Elattar, H.M.; Ali, H.H. LDPC Decoding Algorithms for Implant to Implant Wireless Body Area Network. *IEEE Access* **2018**, *6*, 13200–13212. [[CrossRef](#)]
53. El-Din, R.F.T.; El-Hassani, R.M.; El-Ramly, S.H. A Novel High-Speed Systematic Encoder for Long Binary Cyclic Codes. *IEEE Commun. Lett.* **2013**, *17*, 984–987. [[CrossRef](#)]
54. Schipani, D.; Elia, M.; Rosenthal, J. On the decoding complexity of cyclic codes up to the BCH bound. In Proceedings of the 2011 IEEE International Symposium on Information Theory Proceedings, St. Petersburg, Russia, 31 July–5 August 2011; pp. 835–839.
55. El-Din, R.F.T.; El-Hassani, R.M.; El-Ramly, S.H. Optimizing Chien Search Usage in the BCH Decoder for High Error Rate Transmission. *IEEE Commun. Lett.* **2013**, *17*, 741–744. [[CrossRef](#)]