Received December 29, 2018, accepted January 8, 2019, date of publication January 15, 2019, date of current version February 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2892969

Hash-Polar Codes With Application to 5G

PEIYAO CHEN[1], **(Student Member, IEEE), BAOMING BAI**[1], **(Senior Member, IEEE), ZHAOFENG REN**[1], **JIAQING WANG**[2], **AND SHAOHUI SUN**[2]

[1]State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China
[2]State Key Laboratory of Wireless Mobile Communications, China Academy of Telecommunications Technology, Beijing 100083, China

Corresponding author: Baoming Bai (bmbai@mail.xidian.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61771364 and Grant 61701368, and in part by the Key Research and Development Project of Shaanxi Province under Grant 2018ZDCXL-GY-04-05.

ABSTRACT This paper presents a kind of concatenated polar codes called hash-polar codes with flexible outer code lengths, in which a hash function-based encoder is used as an outer encoder. A partial hash-polar code is also proposed to enhance the error-correcting performance at high signal-to-noise ratios. Since polar codes have been recommended by 3GPP as the channel coding scheme for the 5G enhanced mobile broadband control channel, the design of hash-polar codes for 5G is considered, where both good error-correcting performance and low false alarm rate (FAR) are required. The simulation results show that, under the 5G FAR requirement, the proposed hash-polar codes have similar frame error rate performance to cyclic redundancy check (CRC)-polar codes and perform better than parity check polar codes. In order to support early termination (ET) for 5G coding, we then propose segmented hash-polar codes, which exhibit the advantages of the ET gain compared with both CRC-polar codes and distributed CRC-polar codes.

INDEX TERMS Concatenated polar codes, hash-polar codes, false alarm rate, early-termination, 5G channel codes.

I. INTRODUCTION

Polar codes [1], the first provably capacity-achieving codes discovered by Arıkan in 2009, are of great interest recently. Under recursive encoding structures in conjunction with successive cancellation (SC) decoding, channel polarization arises, where equivalent bit-channels polarize to be nearly noiseless or useless as the code length approaches infinity. The capacity can be achieved by employing the noiseless bit-channels for data transmission. Since the error-correcting performance under the simple SC decoding for polar codes with finite code length is inferior to other modern codes such as low-density parity-check (LDPC) codes and Turbo codes, successive cancellation list (SCL) decoding was proposed in 2011 [2]–[5], where L paths were kept at each decoding tree level to select the most likely estimation. In [6], an adaptive SCL decoding algorithm was provided. Compared with the conventional SCL decoder with constant list size, the adaptive SCL decoder can achieve similar performance but with significantly lower complexity.

An important direction of improving the finite performance of polar codes is to concatenate a polar code with a high-rate systematic outer code, such as cyclic redundancy check (CRC)-aided polar (CRC-polar) codes [2], [7], and parity check polar (PC-polar) codes [8], where check bits are appended at the end of information bits and scattered within information bits, respectively. For CRC-polar decoder, the check bits are regarded as unfrozen bits and decoded by the decoder. While, for PC-polar decoder, the check bits can be viewed as dynamic frozen bits, and will be computed from the decoded information bits. Recently, concatenated polar codes have been accepted by 3GPP as the channel coding scheme for 5G enhanced mobile broadband (eMBB) control channel, where both low frame error rate (FER) and false alarm rate (FAR) performance are required [9]. In [10], we proposed a concatenated polar coding scheme called hash-polar codes to reduce the FAR with satisfying error-correcting performance. Moreover, the distributed CRC polar (DCRC-polar) codes were proposed for downlink channel [11], [12] to reduce the energy consumption of the decoder. DCRC-polar codes can be considered as an instance of the PC-polar codes, in which the check bits are scattered within information bits by swapping rows and columns of the CRC-generator-matrix. These concatenated polar codes (including hash-polar codes) were recommended as candidates for coding the eMBB control information in 2017 [13]–[18].

Other concatenated polar coding schemes are also available [19]–[23], where multiple polar codes are considered to concatenate with other codes. In 2010, Bakshi *et al.* [19] demonstrated that concatenating polar codes with a high-rate outer Reed-Solomon (RS) code could improve the error decay rate. Then, in 2014, Mahdavifar *et al.* [20] showed that the concatenation of polar codes with interleaved RS codes could increase the error decay rate to be $O(2^{-N^{1-\varepsilon}})$ for any $\varepsilon > 0$, where N is the total code length. In [21]–[23], Bose-Chaudhuri-Hocquenghem (BCH) codes, convolutional codes, and single-parity-check (SPC) codes were considered instead of RS codes to increase the error decay rate and improve error-correcting performance. Furthermore, generalized concatenated polar codes with multiple outer codes were also designed in [24]–[27].

For a systematic outer code, increasing the number of check bits, on the one hand, may obtain better error detection performance, but on the other hand, may lead to increased effective code rate of the inner polar code, resulting in inferior error-correcting performance. Thus, in addition to the structure of outer codes, the number of check bits from the systematic outer code were also considered for the concatenated polar codes. Murata and Ochiai [28] and Hashemi *et al.* [29] showed that there is a trade-off between the FER performance and the CRC code length. It means that specific CRC code lengths are needed to accommodate different information lengths, code rates and signal-to-noise ratios (SNRs). Thus, to adapt those variations, it is preferred to use the outer codes with flexible code lengths implemented by one outer encoder.

In this paper, we attempt to use a hash encoder as the systematic outer encoder to obtain arbitrary outer code lengths for arbitrary input lengths, in which a hash function is used to generate various check bits. In addition, taking the advantage of the structure of the hash outer encoder, check bits can be scattered within information bits, which has an advantage, i.e. supporting early-termination [30], for 5G control channel. Although the 5G standardization of polar codes for eMBB control channel was finalized, since our proposed hash-polar codes had ever been recommended as a candidate for eMBB control channel, we introduce them with their application to 5G in this paper in more detail for exploring their potentials for other scenarios.

The main contributions of this paper are summarized as follows:

- We first introduce the encoding and decoding of hash-polar codes, where a hash function is used to generate various check bits for flexible outer code lengths. By analyzing the decoding error probability for systematic concatenated polar codes, we then propose an adaptive hash-polar coding scheme, where the number of check bits varies with SNRs according to the provided upper-bound. In particular, the decoding error of check bits by polar decoder is taken into account in our upper-bound.
- To improve the error correcting performance at high SNRs, we propose partial hash-polar codes, partial information bits of which are fed into the hash function twice, resulting in the reduction of collisions for each check sequence. Inspired by considering the minimum Hamming distance (for polar codes without outer codes), the partial hash-polar codes are constructed based on low row-weights and outperform that constructed based on low bit-channel reliabilities.
- We design and apply the proposed hash-polar codes for 5G eMBB control channel, where low FAR is required for uplink and both low FAR and early-termination (ET) are required for downlink. By analyzing the FAR performance, the proposed hash-polar codes with designed CRC bits can meet the 5G FAR target. To support ET, segmented hash-polar codes are proposed, and a length search algorithm is provided to design segmented hash-polar codes for satisfying the FAR requirement.

The remainder of this paper is organized as follows. Section II gives a brief introduction of polar codes. Section III presents hash-polar codes and the corresponding decoding algorithm. Improved partial hash-polar codes are introduced in Section IV. In Section V, we discuss the design of hash-polar codes for 5G eMBB control channel, where the analysis of FAR performance is also given. Under the requirement of ET, segmented hash-polar codes and the corresponding decoding algorithm are presented. Finally, conclusion is drawn in Section VI.

II. BACKGROUND

A. POLAR CODES

Polar codes, as a class of capacity-achieving linear block codes, are constructed based on the channel polarization. The polarization effect refers to the phenomenon that part of equivalent bit-channels become nearly noiseless while others become useless with the code length tending to infinity. The conventional polarization matrix (called Arıkan's kernel) is given as $\mathbf{F}_2 = \left[\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right]$. Then, the generator matrix $\mathbf{G}_N$ for a polar code of length $N = 2^n$ can be recursively obtained as

$$\mathbf{G}_N = \mathbf{F}_2^{\otimes n}, \tag{1}$$

where $\otimes$ represents the Kronecker power operation. Based on $\mathbf{G}_N$, N bit-channels $W_N^{(i)}$ $(1 \le i \le N)$ with different reliabilities can be obtained. Information bits are only transmitted over those equivalent bit-channels with the highest reliabilities. Let $\mathcal{A} \subseteq \{1, 2, ..., N\}$ denote the index set of bit-channels with the highest capacities, which is called the *information set*. The complementary set $\mathcal{A}^c = \{1, 2, ..., N\} \backslash \mathcal{A}$ is called the *frozen set*, which is known to the decoder. The selection of $\mathcal{A}$ refers to as the code construction. Many efficient construction methods, including Bhattacharyya bounds method, Monte-Carlo method and Gaussian approximation (GA) method, are summarized in [31].

Assume that the binary-sequence $\mathbf{u} = \{u_1, u_2, \ldots, u_N\}$ is fed into the polar encoder, where $\mathbf{u}_{\mathcal{A}} = \{u_i, i \in \mathcal{A}\}$ contains data. The resulting codeword $\mathbf{c} = \mathbf{u}\mathbf{G}_N$ is transmitted over the channel with the transition probability $W(y_i|c_i)$, where $\{y_i\}$ is the received sequence. For any non-negative integer $0 \le j \le 2^n - 1$, we express it in a binary expansion

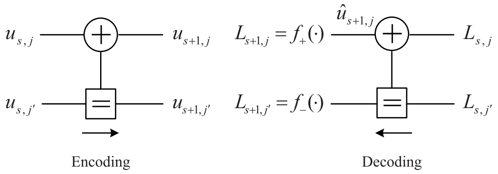

FIGURE 1. Encoding and decoding process.

as $j \triangleq (j_{n-1}, j_{n-2}, \ldots, j_0)_2$, i.e., $j = \sum_{s=0}^{n-1} j_s \cdot 2^s$, where $j_s \in \{0, 1\}$. Let (j, j') $(j < j')$ denote the index pair that their binary representations differ only in the s-th bit, where (j, j') is called s-complementary pair [32]. Thus, the polar encoding process can also be described as a "recursive computation" given by

$$\begin{cases} u_{s+1,j} = u_{s,j} \oplus u_{s,j'}, \\ u_{s+1,j'} = u_{s,j'}, \end{cases} \tag{2}$$

for each s-complementary pair (j, j') with the initialization $u_{0,j} = u_{j+1}$, where $0 \le j \le N-1$ and $0 \le s \le n-1$. The codeword is then given by $c_{j+1} = u_{n,j}$. Refer to Fig. 1.

Consider SC decoder in the log-likelihood ratio (LLR) domain. The estimate $\hat{u}_i$ with $i \in \mathcal{A}^c$ will be set to zero or a known bit, and the estimate $\hat{u}_i$ with $i \in \mathcal{A}$ will be determined by the decision LLRs

$$L_i = L(u_i) \triangleq \ln \frac{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1} | u_i = 0)}{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1} | u_i = 1)}, \quad 1 \le i \le N. \tag{3}$$

That is $\hat{u}_i = 0$ if $L_i \ge 0$, and $\hat{u}_i = 1$ otherwise.

Assume that $0 \le s \le n-1$, and $(s+1)$-complementary pair is equal to s-complementary pair when $s+1$ equals n. Let $L_{s,j}$ denote the recursive LLR in layer-s of the decoder. Based on the butterfly computational structure, the decision LLRs can be computed recursively as

$$\begin{cases} L_{s+1,j} = f_-(L_{s,j}, L_{s,j'}), \; for \text{ ``} \oplus \text{''} \; constraint, \\ L_{s+1,j'} = f_+(L_{s,j}, L_{s,j'}, \hat{u}_{s+1,j}), \; for \text{ ``} = \text{''} \; constraint, \end{cases} \tag{4}$$

with

$$\begin{cases} \hat{u}_{s,j} = \hat{u}_{s+1,j} \oplus \hat{u}_{s+1,j'}, \\ \hat{u}_{s,j'} = \hat{u}_{s+1,j'}, \end{cases} \tag{5}$$

where

$$\begin{cases} f_-(\alpha, \beta) \triangleq \ln \dfrac{e^{\alpha+\beta}+1}{e^{\alpha}+e^{\beta}}, \\ f_+(\alpha, \beta, u) \triangleq (-1)^u \cdot \alpha + \beta. \end{cases} \tag{6}$$

The initial value $L_{0,j} = \ln \frac{W(y_i|0)}{W(y_i|1)}$ with $j = i-1$, $1 \le i \le N$. With the recursive computation in (4), the decision LLRs are finally given by $L_i = L_{n,i-1}$. The encoding and decoding procedures are shown in Fig. 1.

Although polar codes with SC decoding are proved to be capable of achieving the symmetric channel capacity, SC decoding exhibits an inferior error-correcting performance for polar codes with short and moderate block lengths. Thus, the SC-based list decoding algorithm was proposed, where the maximum L decoding paths are kept at each decoding stage. The path metric for each path ℓ can be computed as follows with the initialization $\mathrm{PM}_0[l] = 0$ for all ℓ,

$$\mathrm{PM}_i[\ell] = \begin{cases} \mathrm{PM}_{i-1}[\ell], \text{ if } \hat{u}_i[\ell] = \dfrac{1}{2}(1 - \mathrm{sign}(L_i[\ell])), \\ \mathrm{PM}_{i-1}[\ell] + |L_i[\ell]|, \text{ otherwise.} \end{cases} \tag{7}$$

where $1 \le i \le N$, and $L_i[\ell] = \ln \frac{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1}[\ell] | u_i = 0)}{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1}[\ell] | u_i = 1)}$, which can be obtained by (4). Finally, the decoding path with the smallest path metric is chosen as the output.

B. HASH FUNCTION

Hash function is a nonlinear function exhibiting avalanche effect, where one bit variation of the input leads to a huge change of the output. There are many kinds of hash functions with one or more than one input, resulting in one output. Assume that $\mathrm{h}(a, b) \in \{0, 1\}^v$ denotes a two-input hash function with $a \in \{0, 1\}^k$, $b \in \{0, 1\}^v$. In order to guarantee a low number of collisions, the hash function is chosen randomly from a family of hash functions with uniform difference property, where for $a_i \in \{0, 1\}^k$ and $b_i \in \{0, 1\}^v$ with $i = 1, 2$, $\Pr\{\mathrm{h}(a_1, b_1) = \mathrm{h}(a_2, b_2) | (a_1, b_1) \ne (a_2, b_2)\} \le \frac{1}{2^v}$, and $\Pr\{(\mathrm{h}(a_1, b_1) - \mathrm{h}(a_2, b_2)) \bmod 2^v\}$ is uniformly distributed in 2^v. In this paper, the improved "one-at-a-time" hash function [33] with two 32-bit unsigned integer inputs (seen in Appendix-A), which only requires 10 additions, 15 shifts, and 6 XOR operations, is applied resulting in an unsigned integer $S \in \{0, 1\}^{32}$, called *hash state*.

III. HASH-POLAR CODES

On account of the fact that basic polar codes of finite length exhibit inferior performance with list decoding, systematic concatenated polar codes, with CRC-polar codes as typical examples, were proposed. Due to the trade-off between the FER and the outer code length, flexible outer code lengths are required, where different SNRs may correspond to different optimal code lengths. In this section, inspired by the application of the hash function in spinal codes [34], we introduce the hash-polar code, which is a concatenation of the hash function and a basic polar code, and various numbers of check bits (corresponding to various outer code lengths) can be obtained via such a hash function.

A. DESIGN OF HASH-POLAR CODES

1) ENCODING OF HASH-POLAR CODES

A hash-polar encoder is shown in Fig. 2, where the hash encoder has a trellis-coded modulation (TCM)-like structure. A K-bit source sequence $\mathbf{d}$ is divided into non-overlapped $I \ge 2$ segments, each of which is a bit-sequence of length r_i and is represented as an integer k_i $(1 \le i \le I)$. The splitting process is denoted by $\mathbf{d} \rightarrow \{k_i\}$. Note that $\sum_{i=1}^{I} r_i = K$, where r_i can be different for different segments. Let the two inputs of the hash function h be the integer k_i from the i-th segment and the prior *hash state* S_{i-1} obtained by

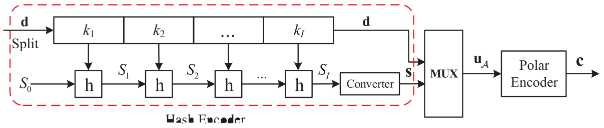

FIGURE 2. Hash-polar encoder.

the $(i-1)$-th segment. Then, the output is given by $S_i = \mathrm{h}(S_{i-1}, k_i)$. The initial state S_0 is known both to the transmitter and to the receiver, and here we assume that $S_0 = 0$.

After computing all the segments, the output state S_I is clamped as a J-bit integer $\Theta(S_I)$ ($J \leq v = 32$), which is represented as a J-bit binary sequence $\mathbf{s}$. The J-bit sequence $\mathbf{s}$, called the *hash state sequence*, is regarded as the check bits, thus the codeword given by the hash encoder in systematic form is $\mathbf{u}_{\mathcal{A}} = (\mathbf{d}, \mathbf{s})$ with length $K + J$. The attached sequence $\mathbf{u}_{\mathcal{A}}$ is then fed into the polar encoder. Therefore, for polar encoder, $|\mathcal{A}| = K + J$. Hash-polar codes are a kind of concatenated codes, in which the hash encoder and polar encoder act as the outer encoder and inner encoder, respectively. Moreover, the number of check bits J from the hash encoder is more flexible than that from the CRC encoder, in which preferred J-bit check sequences can be obtained by clamping and converting the output state S_I from the hash encoder.

2) DECODING OF HASH-POLAR CODES

With the use of the hash encoder, hash-aided SCL decoding method is proposed. Instead of selecting the path which can pass CRC as the output in SCL with CRC-aided decoder, hash-aided SCL decoder outputs the most likely decoding path with the smallest path metric among the paths whose states equal the output of the hash encoder. The main hash-aided SCL algorithm is described using pseudo-code in Algorithm 1, where $\mathcal{A}[i]$ represents the i-th index in the *information set* $\mathcal{A}$.

B. PROPERTIES OF HASH-ENCODER

Proposition 1: The undetected error probability of the hash-encoder is lower than 2^{-J}, which is independent of the number of segments, i.e., the value of I.

Proof: Let two messages $\mathbf{d}_1$ and $\mathbf{d}_2$ ($\mathbf{d}_1 \neq \mathbf{d}_2$) be encoded by a hash-encoder, resulting in the J-bit sequence $\mathbf{s}_1$ and $\mathbf{s}_2$, respectively. By the definition of the hash function, the collision probability is lower than 2^{-v} for each segment i ($1 \leq i \leq I-1$), and is lower than 2^{-J} for the last segment. Thus, the undetected error probability of the hash-encoder is given by

$$\begin{aligned} P_{\mathrm{ud}} &= \Pr\{\mathbf{s}_1 = \mathbf{s}_2 | \mathbf{d}_1 \neq \mathbf{d}_2\} \\ &\leq \sum_{i=0}^{I-1} C_{I-1}^{i} (1-2^{-v})^{i} (2^{-v})^{I-1-i} 2^{-J} \\ &= (1 - 2^{-v} + 2^{-v})^{I-1} 2^{-J} = 2^{-J}. \end{aligned} \tag{8}$$

□

Algorithm 1 Hash-Aided SCL Decoder

```
Input: Received channel probabilities.
Output: Estimated source sequence.
Flag ← 0.
for $i = 1$ to $N$ do
    • Calculate the path metric $\mathrm{PM}_i[\cdot]$ by (4) and (7).
    if $i \in \mathcal{A}^c$ then
        Set $\hat{u}_i[\ell] = 0$ for all $\ell$, and keep all paths.
    else
        Sort the path metrics, and reserve the $L$ most
        probable paths with $\mathrm{PM}_i[1] < \mathrm{PM}_i[2] < \cdots <$
        $\mathrm{PM}_i[L] < \mathrm{PM}_i[\cdot] \in \{\mathrm{PM}_i[\ell], \ell \geq L\}$.
for $\ell = 1$ to $L$ do
    • Map
    $\{\hat{u}_i[\ell], i \in \{\mathcal{A}[1], \ldots, \mathcal{A}[K]\}\} \to \{\hat{k}_j, 1 \leq j \leq I\}$,
    and $\{\hat{u}_i[\ell], i \in \{\mathcal{A}[K+1], \ldots, \mathcal{A}[|\mathcal{A}|]\}\} \to \hat{S}$.
    • Compute $\tilde{S}_I$ with $\tilde{S}_j = h(\tilde{S}_{j-1}, \hat{k}_j)$ $(1 \leq j \leq I)$ and
    $\tilde{S}_0 = 0$ .
    if $\Theta(\tilde{S}_I) == \hat{S}$ then
        $\hat{\mathbf{d}} \leftarrow \{\hat{u}_i[\ell], i \in \{\mathcal{A}[1], \ldots, \mathcal{A}[K]\}\}$.
        Flag ← 1.
        Break.
if Flag== 0 then
    $\hat{\mathbf{d}} \leftarrow \{\hat{u}_i[1], i \in \{\mathcal{A}[1], \ldots, \mathcal{A}[K]\}\}$.
```

The hash function can also be viewed as a mapping from the input to a J-bit *hash sequence*. When $K > J$ and $I = 1$, there are 2^{K-J} data sequences corresponding to the same check sequence (i.e., 2^{K-J} collisions for each check sequence), which may introduce errors when using the check sequence to select the path for polar decoder. Consider $I \geq 2$ and $\frac{K}{I} > v$, then the number of collisions is $2^{K/I-J} \prod_{i=1}^{I-1} 2^{K/I-v} = 2^{K-(I-1)v-J}$. We can see that with the increase of I, the number of collisions decreases. However, the I segments correspond to the I uses of the hash function, which indicates that the larger I will lead to the higher computational complexity. Thus, for low complexity and less collisions, $I = 2$ segments and $r_0 = r_1 = K/2$ are used for hash-polar codes in this paper. Note that for large r_i, we clamp each segment as a 32-bit unsigned integer k_i, where the clamping method is shown in Appendix-B.

Up to the present, the choice of the optimal value of J is also an open problem, due to a trade-off between the outer code length and the FER performance for a given SNR. In [28], the approximate upper-bound on the decoding error probability for the J-bit CRC concatenated system is given, but the bound only holds under the assumption that the estimation of CRC bits is correct. However, at the receiver, both information bits and check bits are decoded as unfrozen bits, leading to the potential decoding errors of check bits. Motivated by [28], the improved bound is discussed as follows.

Assume that the maximum L decoding paths are kept at each stage, and $\mathcal{L} = \{1, 2, \ldots, L\}$ represents the path index set. Consider a systematic concatenated polar code and denote the check sequence $\mathbf{s}$ from the outer encoder with input $\mathbf{d}$ by $\mathbf{s} = \Psi(\mathbf{d})$, then the output of the generalized systematic outer encoder can be written as $\mathbf{u}_{\mathcal{A}} = (\mathbf{d}, \Psi(\mathbf{d}))$. Let $\hat{\mathbf{u}}_{\mathcal{A}} = (\hat{\mathbf{d}}, \hat{\mathbf{s}})$ denote the estimate of $\mathbf{u}_{\mathcal{A}}$, generated by the polar decoder. Note that $\Psi(\hat{\mathbf{d}})$ is not necessarily equal to $\hat{\mathbf{s}}$. Suppose that ℓ^* represents the decoding path index corresponding to the correct data, i.e., $\hat{\mathbf{d}}[\ell^*] = \mathbf{d}$, and γ represents the SNR over the AWGN channel. Then, let $P_{\ell^*}(J, \gamma, \ell) = \Pr\{\ell^* = \ell \in \mathcal{L} \mid J, \gamma\}$ denote the distribution of ℓ^* for a given SNR γ and length J, and $P_\Psi(J, \gamma, \ell) = \Pr\{\hat{\mathbf{s}}[\ell] \neq \Psi(\mathbf{d}) \mid J, \gamma, \ell^* = \ell \in \mathcal{L}\}$ denote the probability that the check bits are decoded by polar codes incorrectly when the information bits are decoded correctly for a given SNR γ and length J.

Proposition 2: For a systematic concatenated polar code with the undetected error probability 2^{-J} of the outer code, the probability of decoding error $P_e(J, \gamma)$ for a given γ and J with list size L is bounded by

$$\begin{aligned} 1 - \sum_{\ell=1}^{L} P_{\ell^*}(J, \gamma, \ell) &\leq P_e(J, \gamma) \\ &\leq 1 - \sum_{\ell=1}^{L} P_{\ell^*}(J, \gamma, \ell)(1 - P_\Psi \\ &\quad \times (J, \gamma, \ell))(1 - 2^{-J})^{(\ell-1)}. \end{aligned} \tag{9}$$

Proof: We prove the left-hand side of (9) first. The perfect decoding for a finite list algorithm means that the correct path $\ell^* \in \mathcal{L}$ can always be found correctly for a given γ. Then, the decoding error probability is equal to $\Pr\{\ell^* \notin \mathcal{L} \mid J, \gamma\} = 1 - \sum_{\ell=1}^{L} P_{\ell^*}(J, \gamma, \ell)$. Thus, $1 - \sum_{\ell=1}^{L} P_{\ell^*}(J, \gamma, \ell) \leq P_e(J, \gamma)$. Now we prove the right-hand side of (9). Assume that the ℓ-th path is the correct path (i.e., $\ell^* = \ell$), then it as a correct output requires the correct decoding of $\hat{\mathbf{s}}[\ell]$ (for selection) and the perfect error-detection of the previous $\ell - 1$ paths, simultaneously. Suppose that $\ell' \leq (\ell - 1)$, which implies that the ℓ'-th path is not the correct path, i.e., $\ell^* \neq \ell'$. Let $(\hat{\mathbf{d}}[\ell'], \hat{\mathbf{s}}[\ell'])$ denote the estimate from the ℓ'-th path. We can see that if the check sequence $\Psi(\hat{\mathbf{d}}[\ell'])$ obtained by the estimate $\hat{\mathbf{d}}[\ell']$ is not equal to the estimated check sequence $\hat{\mathbf{s}}[\ell']$, then the ℓ'-th path will not be selected as the output, which means that it is a perfect error-detection of the ℓ'-th path. Thus, consider the error-detection probability of the ℓ'-th path, we have

$$\begin{aligned} &\Pr\{\Psi(\hat{\mathbf{d}}[\ell']) \\ &\neq \hat{\mathbf{s}}[\ell'] \mid J, \gamma, \ell^* \neq \ell'\} \\ &\geq \Pr\{\Psi(\hat{\mathbf{d}}[\ell']) \neq \hat{\mathbf{s}}[\ell'] \mid \hat{\mathbf{s}}[\ell'] = \Psi(\mathbf{d}), J, \gamma, \ell^* \neq \ell'\} \\ &= \Pr\{\Psi(\hat{\mathbf{d}}[\ell']) \neq \Psi(\mathbf{d}) \mid \hat{\mathbf{s}}[\ell'] = \Psi(\mathbf{d}), J, \gamma, \ell^* \neq \ell'\} \\ &= 1 - \Pr\{\Psi(\hat{\mathbf{d}}[\ell']) = \Psi(\mathbf{d}) \mid \hat{\mathbf{s}}[\ell'] = \Psi(\mathbf{d}), J, \gamma, \ell^* \neq \ell'\} \\ &\geq 1 - \Pr\{\Psi(\hat{\mathbf{d}}[\ell']) = \Psi(\mathbf{d}) \mid J, \gamma, \ell^* \neq \ell'\} \\ &\geq 1 - 2^{-J}. \end{aligned}$$

Then, according to the two requirements above, the correct decoding probability of $\mathbf{u}_{\mathcal{A}}$ is given by

$$\begin{aligned} &\sum_{\ell=1}^{L} \Pr\Big\{(\hat{\mathbf{d}}[\ell], \hat{\mathbf{s}}[\ell]) = (\mathbf{d}, \Psi(\mathbf{d})), \bigcup_{\ell'=1}^{\ell-1} \{\Psi(\hat{\mathbf{d}}[\ell']) \\ &\qquad \neq \hat{\mathbf{s}}[\ell'])\} \mid J, \gamma\Big\} \\ &\quad \geq \sum_{\ell=1}^{L} \Pr\{\ell^* = \ell \in \mathcal{L} \mid J, \gamma\} \Pr\{\hat{\mathbf{s}}[\ell] = \Psi(\mathbf{d}) \mid J, \\ &\qquad \gamma, \ell^* = \ell \in \mathcal{L}\} \prod_{\ell'=1}^{\ell-1} (1 - 2^{-J}) \\ &\quad \geq \sum_{\ell=1}^{L} P_{\ell^*}(J, \gamma, \ell)(1 - P_\Psi(J, \gamma, \ell))(1 - 2^{-J})^{(\ell-1)}. \end{aligned}$$

Thus, the decoding error probability is upper-bounded by

$$P_e(J, \gamma) \leq 1 - \sum_{\ell=1}^{L} P_{\ell^*}(J, \gamma, \ell)(1 - P_\Psi(J, \gamma, \ell))(1 - 2^{-J})^{(\ell-1)}.$$

□

Note that the bound holds for any systematic concatenated polar codes, in which the undetected error probability of the outer code is 2^{-J}. The distribution of $P_{\ell^*}(J, \gamma, \ell)$ and $P_\Psi(J, \gamma, \ell)$ can be obtained via the Monte-Carlo method (without concatenating the outer encoder, due to the assumption that the undetected error probability of the outer code is 2^{-J}, when the estimation of check bits is correct), in which the effective code rate of the single polar code is $\frac{K+J}{N}$, i.e., $K + J$ bits are decoded as unfrozen bits at the receiver. Taking into account the advantage of hash-polar codes with flexible numbers of check bits, the adaptive hash-polar codes can be obtained according to the upper-bound in Proposition 2, where J can vary with different SNRs. It is worth noting that all check sequences with the varied J are obtained from the same output state S_I. In this paper, we assume that the varied J for each SNR is known for both the transmitter and the receiver.

C. EXAMPLES OF HASH-POLAR CODES

Two examples of hash-polar codes are provided in this section. In all simulations, BPSK signaling over the AWGN channel is assumed. All codes are constructed by the GA method at -1.59 dB.

Example 1: Consider hash-polar codes with code rates $R = 1/2$ and $R = 1/4$ for $K = 32$. Fig. 3 shows the FER performance of hash-polar codes along with comparable CRC-polar codes, where the performance with different numbers of check bits are also compared. For all schemes, the decoding list size $L = 8$. Let $g_J(x)$ denote the generator polynomial of J-bit CRC. Then, $g_4(x) = x^4 + x + 1$ and $g_6(x) = x^6 + x^5 + x^2 + x + 1$ are used corresponding to CRC-4 and CRC-6, respectively. In addition, the proposed upper-bounds for different numbers of check bits J are

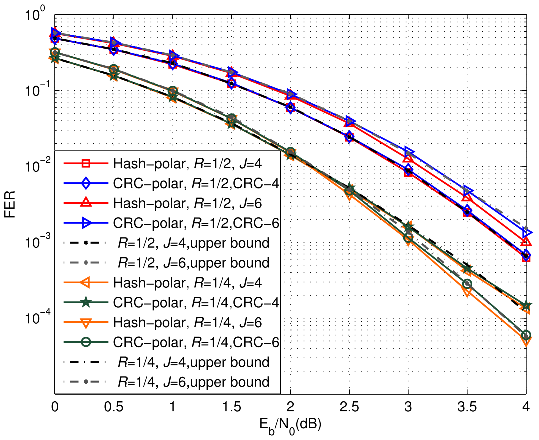

FIGURE 3. Performance comparison between hash-polar codes and CRC-aided polar codes with $K = 32$ over the AWGN channel.

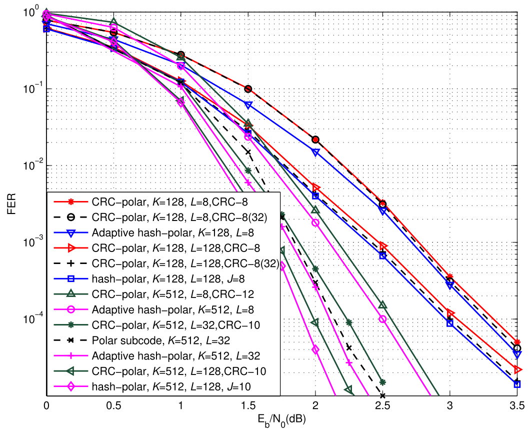

FIGURE 4. Performance comparison among hash-polar codes, adaptive hash-polar codes, polar subcodes and CRC-aided polar codes with $R = 1/2$ over the AWGN channel.

also given. The $P_{\ell^*}(J, \gamma, \ell)$ and $P_{\Psi}(J, \gamma, \ell)$ are obtained by the Monte-Carlo method for each SNR.

It can be seen that whatever the number of check bits is 4 or 6, the simulated FER performance of both hash-polar codes and CRC-polar codes can match the upper-bounds as shown in Fig. 3. From the upper-bound, it can also be seen that, for the same K, the optimal number of check bits is different for the different R. Furthermore, for the case of $R = 1/4$, we can see that the optimal number of check bits is 4 at low SNR, whereas the optimal number of check bits is 6 at high SNR. Thus, for further improvement of the error-correcting performance, it is necessary to apply an adaptive outer coding scheme.

Example 2: In Fig. 4, the performances of hash-polar codes and CRC-aided polar codes with $R = 1/2$ are compared, where the number of check bits is selected according to the upper-bound. For $K = 128$, the performance of CRC-aided polar codes with 8 check bits taking from CRC-32 is also considered and labeled by "CRC-8(32)" in Fig. 4. Furthermore, the performance of adaptive hash-polar codes with $R = 1/2$ is also given, where the maximum numbers of check bits are 8 and 12 for $K = 128$ and $K = 512$, respectively. Moreover, as a reference, the performance of randomized polar subcodes proposed in [26] with $L = 32$ is also given, where the number of check bits (i.e., the number of type-A dynamic freezing constraints in [26]) is 10. The related parameters for the remainder of those schemes are exhibited in Table 1.

TABLE 1. Related parameters for example 2.

K	L	Hash-Polar Codes	CRC-Polar Codes
128	8	$J = 4$ for $0 \sim 2.5$ dB $J = 6$ for 3.0 dB $J = 8$ for 3.5 dB	CRC-8: $g_8(x) =$ $x^8 + x^5 + x^4 + 1$
	128	$J = 8$	
512	8	$J = 6$ for $0 \sim 1.5$ dB $J = 8$ for 2.0 dB $J = 10$ for 2.5 dB $J = 12$ for 3.0 dB	CRC-12: $g_{12}(x) = x^{12} +$ $x^{11} + x^3 + x^2 + x + 1$
	32	$J = 6$ for $0 \sim 1.75$ dB $J = 10$ for $2.0 \sim 2.5$ dB	CRC-10: $g_{10}(x) = x^{10} + x^9 +$ $x^6 + x^3 + x^2 + x + 1$
	128	$J = 10$	

Although the undetected error probability of both the J-bit hash-polar code and CRC-polar code are lower than 2^{-J} [35], we can see from Fig. 4 that the hash-polar codes outperform CRC-polar codes when $L = 128$, which indicates that hash-polar codes have better check ability than CRC-polar codes and can perform well for the path selection. It is worth noting that the traditional CRC-32 check sum cannot be used to CRC-aided polar codes with "CRC-8(32)" at the receiver. Instead, for selecting paths, the estimated information sequence should be encoded by CRC-32, resulting in a 8-bit computed check sequence, which will be compared with the estimated check sequence, requiring 8 operations for a comparison. However, hash-polar codes only need to compare a clamped output state $\Theta(S_I)$ with low computational complexity. In addition, the adaptive hash-polar code with $K = 512, L = 32$ can provide up to 0.12 dB gain for $E_b/N_0 < 2$ dB and 0.2 dB gain for $E_b/N_0 > 2$ dB with respect to the optimal outer code length-based CRC-polar code, and 0.15 dB gain compared to the polar subcode at FER=10^{-5}, which reflects the necessity of flexible outer code lengths for finite-length concatenated polar codes.

Remarks: From the above examples, we can see that, compared with CRC-polar codes, hash-polar codes have the following advantages:

- For the same number of check bits, similar performance can be exhibited with a small list size, and better performance can be obtained with a large list size.
- More flexible numbers of check bits (corresponding to flexible outer code lengths) can be obtained, where only one hash encoder is required, while for CRC-polar

codes, different generator polynomials, i.e., different encoders, are required.

- With the advantage of clamping, the number of additions, shifts, and XOR operations of the hash encoder are only related to the number of segments and not related to the input length, while for CRC-polar codes, those operations of CRC encoder are related to the input length.

IV. PARTIAL HASH-POLAR CODES

It is known that outer codes play two roles in improving the performance of polar codes. For one thing, outer codes, such as CRC codes, can increase the minimum Hamming distance or improve the code-weight distribution [36] (unfortunately, for nonlinear hash-polar codes, it is hard to analyze the minimum Hamming distance); for another thing, they help the list decoder to select the most likely estimate from decoding candidates. At high SNRs, the correct path is always contained with a large probability in the candidate-list, and it will be found by the outer code among the paths who pass the check. If the estimations of check codes are correct, low collisions should be ensured to find the correct path successfully. In this section, partial hash-polar codes are proposed to improve the FER performance at high SNRs, where partial information bits are fed into the hash function to reduce the collisions.

A. ENCODING AND DECODING OF PARTIAL HASH-POLAR CODES

The partial hash-polar encoder is shown in Fig. 5. Assume that $\tilde{\mathcal{A}} \subseteq \mathcal{A}'$ denotes the index set of partial information bits, where $\mathcal{A}' \subseteq \mathcal{A}$ with $|\mathcal{A}'| = K$ denotes the *information set* excluding the indexes of check bits. Partial information bits $\mathbf{d}'$ are chosen from the source sequence $\mathbf{d}$ according to the subset $\tilde{\mathcal{A}}$. Similar to the use of termination processing for convolutional structure codes to improve the reliability, partial information bits $\mathbf{d}'$ are encoded successively twice by the hash function, where the other input for the first hash function is a known *hash state* S_0, and for the second hash function is the output of the first hash function. Similarly, the output of the second hash function S_2 is converted into a J-bit *hash state sequence* $\mathbf{s}$ as the check of the (partial) information sequence $\mathbf{d}$. We regard the sequence $\mathbf{u} = (\mathbf{d}, \mathbf{s})$ as the output of the partial hash encoder, and then it is encoded by a polar encoder. The key point of the partial hash encoder is to construct the subset $\tilde{\mathcal{A}}$ appropriately, i.e., to ensure that all estimated information bits on the selected path are correct, when the path is selected among the paths whose estimated bits belong to the subset can pass the check.

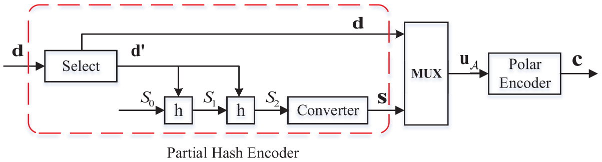

FIGURE 5. Partial hash-polar encoder.

Consider polar codes without outer codes. It is well known that the minimum Hamming weight of polar codes is equal to the minimum row-weight of $\mathbf{G}_{N_{\mathcal{A}'}}$, where $\mathbf{G}_{N_{\mathcal{A}'}}$ consists of the rows, the index of which belongs to the set $\mathcal{A}'$. We can see that if the *information set* changes, the matrix $\mathbf{G}_{N_{\mathcal{A}'}}$ and the minimum row-weight W_r will also change. Li *et al.* [6], [37] and Bioglio *et al.* [38] found that the performance of finite-length polar codes was dominated by the minimum Hamming distance, and the performance could be improved by improving the minimum Hamming distance to $2W_r$ via constructing the $\mathbf{G}_{N_{\mathcal{A}'}}$ based on Reed-Muller codes. The improvement suggests that both code construction (i.e., bit-channel reliabilities) and row-weights (which equal W_r or $2W_r$) can be considered for the set $\tilde{\mathcal{A}}$. The specific construction of the set $\tilde{\mathcal{A}}$ can be obtained as follows.

Step 1: Compute the i-th row-weight of $\mathbf{G}_{N_{\mathcal{A}'}}$ by Lemma 1, where $i \in \mathcal{A}'$.

Step 2: Find the minimum row-weight W_r among $|\mathcal{A}'|$ row-weights.

Step 3: If the i-th ($i \in \mathcal{A}'$) row-weight equals W_r or $2W_r$, let the i belong to the set $\tilde{\mathcal{A}}$.

Lemma 1: The i-th ($1 \leqslant i \leqslant N = 2^n$) row-weight of the generator matrix $\mathbf{G}_N$ is $2^{x(\tilde{i})}$, where $x(\tilde{i}) = \sum_{j=1}^{n} x_j$, and $\{x_j \in \{0, 1\}, 1 \leq j \leq n\}$ represents the binary expansion of $\tilde{i} = i - 1$.

Proof: This lemma has been proved in [39]. □

According to the $\tilde{\mathcal{A}}$, the partial hash encoder works. We note that the number of collisions for partial hash-polar codes is $2^{|\tilde{\mathcal{A}}|-v-J}$, where $|\tilde{\mathcal{A}}| \leq |\mathcal{A}'| = K$ (generally, $|\tilde{\mathcal{A}}| < |\mathcal{A}'|$). Thus, the coding gain may be obtained at high SNRs. Similar to the hash-aided SCL decoder, the partial hash-aided SCL algorithm can be used, where the decoder outputs the most likely decoding path with the smallest path metric by (7) among the paths whose states equal the output of the partial hash encoder.

B. EXAMPLES OF PARTIAL HASH-POLAR CODES

Two examples of partial hash-polar codes are provided in this section. In all simulations, BPSK signaling over the AWGN channel is assumed. All codes are constructed by the GA method at −1.59 dB. The number of check bits $J = 8$ is applied to all schemes and the generator polynomial of CRC-8 is the same as that in Fig. 3.

Example 3: Error-correcting performance comparison among partial hash-polar codes, hash-polar codes, and CRC-polar codes with different code rates are shown in Fig. 6, where the code length $N = 256$ and the list size $L = 8$. From Fig. 6, it can be seen that partial hash-polar codes outperform both the hash-polar codes, and CRC-polar codes, especially at high SNRs. Although the coding gain decreases as the code rate increases, the proposed partial hash-polar code with $R = 2/3$ also provides about 0.13 dB gain compared to the CRC-polar code at high SNRs.

Example 4: In Fig. 7, as a reference, the construction of the index set $\tilde{\mathcal{A}}$ based on the lowest bit-channel

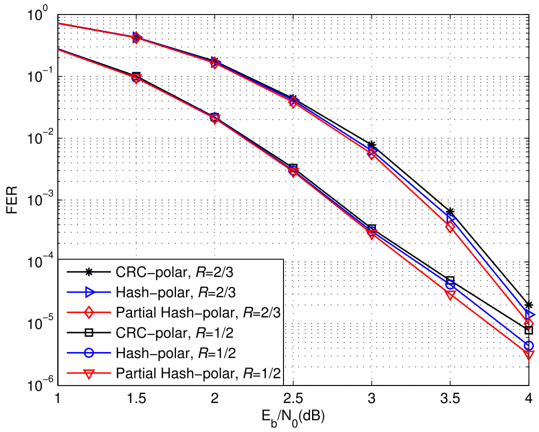

FIGURE 6. Performance comparison among partial hash-polar codes, hash-polar codes and CRC-polar codes with $N = 256$ over the AWGN channel.

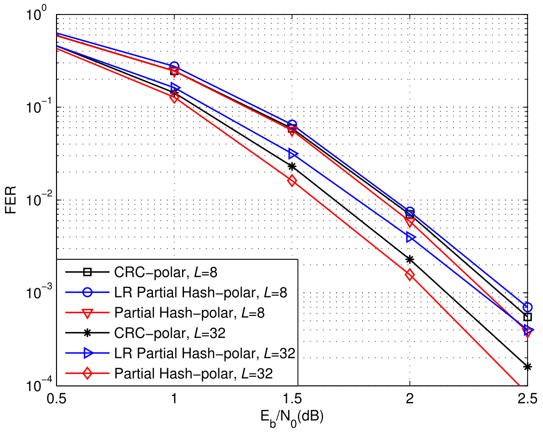

FIGURE 7. Performance comparison among partial hash-polar codes, LR partial hash-polar codes and CRC-polar codes with $N = 512$ and $R = 1/2$ over the AWGN channel.

reliabilities (LR) in $\mathcal{A}'$ is considered, where the bit-channel reliabilities are evaluated by the GA method, and the subset size $|\tilde{\mathcal{A}}|$ is equal to that in the proposed partial hash-polar codes. For all schemes, the code length $N = 512$ and the code rate $R = 1/2$. From Fig. 7, we can see that the proposed partial hash-polar codes outperform both the CRC-polar codes and LR partial hash-polar codes with the same list size, which indicates that the construction of the subset $\tilde{\mathcal{A}}$ based on the low row-weights is more efficient than that based on the low reliabilities. Fig. 7 also shows that the coding gain can increase as the list size increases.

V. DESIGN OF HASH-POLAR CODES FOR 5G

As the coding scheme for control information, the codes are also required to exhibit lower FAR performance and support ET (for downlink channel) apart from the outstanding FER performance. In this section, we discuss the design of hash-polar codes for the eMBB control channel. It is worth noting that the analysis and results can be extended to other channels (e.g. ultra-reliable low latency communication (uRLLC) control channel, where only the FAR target is different [40]).

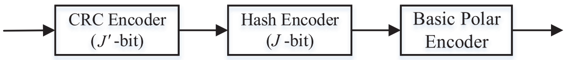

FIGURE 8. Structure of hash-concatenated polar codes for the blind detection.

A. DESIGN OF HASH-POLAR CODES FOR 5G FAR TARGET

In 3GPP LTE and 5G standards, blind detection is used by the user equipment (UE) to receive control information, where the UE, by attempting at decoding a set of candidate locations, identifies if one of the candidates holds its control information. Typical blind detection solutions for polar codes can be found in [41]–[43]. Due to the use of blind detection, low FAR performance is required for coding schemes over the control channel. According to LTE standard Release 8 [44], a 16-bit CRC is always applied to protect the information sequence and masked by a 16-bit radio network temporary identifier (RNTI). Thus, the scheme that a CRC code is used in conjunction with a concatenated polar codes are also considered for 5G control channel. The FAR is defined as

$$\mathrm{FAR} = \frac{E_{\mathrm{crc}}}{E_{\mathrm{total}}}, \tag{10}$$

where E_{crc} represents the number of the incorrectly decoded frames which pass the CRC, and E_{total} denotes the total number of the incorrectly decoded frames.

For hash-polar codes, it is not necessary to use a 16-bit CRC, and we showed that hash-polar codes with a 12-bit CRC can also work well and satisfy the FAR target in [10]. Assume that a J'-bit CRC encoder is inserted in front of the J-bit hash-polar encoder for the blind detection as shown in Fig. 8. Since the output of the hash function can also be viewed as a check word, the false alarm rate of hash-concatenated polar codes can be redefined as

$$\mathrm{FAR} = \frac{E_{\mathrm{hash\&crc}}}{E_{\mathrm{total}}}, \tag{11}$$

where $E_{\mathrm{hash\&crc}}$ represents the number of the incorrectly decoded frames which satisfy the check of the hash function and the CRC simultaneously.

Lemma 2: The FAR target P_t can be satisfied, if a J'-bit CRC encoder is applied for hash-concatenated polar codes with $J' \geq (\log_2 L - \log_2 P_t - J)$, where J denotes the check length from the hash encoder.

Proof: The undetected error probability of a J'-bit CRC and a J-bit hash sequence are lower than $2^{-J'}$ [35] and 2^{-J}, respectively. Thus, the undetected error probability of one path is lower than $2^{-(J+J')}$, i.e., $P_{\mathrm{ud}} \leq 2^{-(J+J')}$. According to the definition, the false alarm rate P_{far} equals the undetected error probability of L paths, thus, we have

$$\begin{aligned} P_{\mathrm{far}} &= 1 - (1 - P_{\mathrm{ud}})^L \\ &\leq L \cdot P_{\mathrm{ud}} \\ &\leq L \cdot 2^{-(J+J')}, \end{aligned} \tag{12}$$

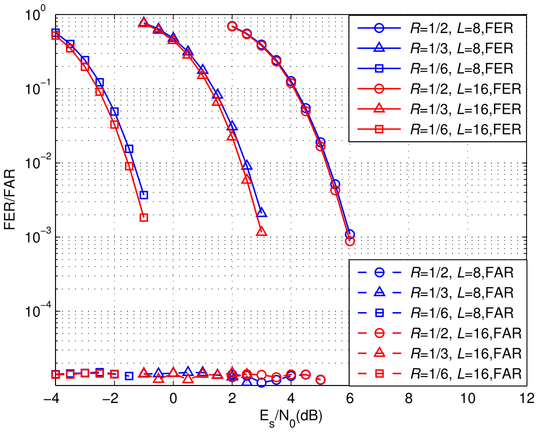

FIGURE 9. FER and FAR performance of hash-concatenated polar codes with different list sizes over the AWGN channel.

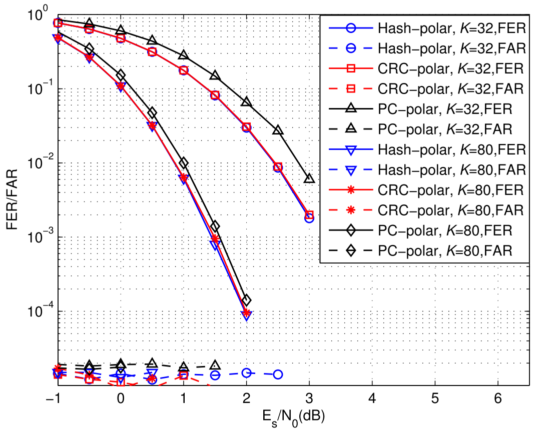

FIGURE 10. Performance comparison among hash-concatenated polar codes, CRC-polar codes and PC-polar codes with $R = 1/3$ over the AWGN channel.

where P_{far} tends to $L \cdot 2^{-(J+J')}$, when $P_{\text{ud}} = 2^{-(J+J')}$ with $2^{-(J+J')}$ tending to 0. Assume that $L \cdot 2^{-(J+J')} \leq P_t$, then we have $P_{\text{far}} \leq P_t$ and $J' \geq (\log_2 L - \log_2 P_t - J)$. Hence, the lemma is proved. □

It is worth noting that when $J = 0$, the code is degraded to a long CRC-polar code. Furthermore, before the error detection, only one path is reserved for PC-polar codes [8]. Therefore, the FAR of PC-polar codes is lower than $2^{-J'}$, where only CRC is employed for the error detection. From (12), it can also be seen that the FAR increases with the increase of L. For lower FAR, it is also an efficient method that only the $L_{\max} \leq L$ most likely paths at the last level are checked by the hash function and CRC. With the decrease of $L_{\max}$, the FAR will decrease, but the FER will increase. In order to avoid the loss of error-correcting performance, we assume that $L_{\max}$ equals L in this paper.

Example 5: According to LTE standard, for eMBB control channel, FER target is 10^{-2}, and FAR target is less than 2×10^{-5}. For high spectral efficiency, QPSK modulation is considered for the AWGN channel. In both Fig. 9 and Fig. 10, the codes are constructed by the GA method at 2.89 dB.

Fig. 9 shows the FER and FAR performance of hash-concatenated polar codes with different list sizes, where $K = 32$, $J = 9$ for codes with $L = 8$, and $J = 10$ for codes with $L = 16$. For the FAR target, according to Lemma 2, CRC-10 (with $g_{10}(x)$) is used. It can be seen that all codes can meet the FAR target with the designed CRC code length J'.

In Fig. 10, the FER and FAR performance of hash-concatenated polar codes along with comparable CRC-polar codes and PC-polar codes are exhibited. Both $K = 32$ and $K = 80$ with $R = 1/3$ are considered, and the quasi-uniform puncturing (QUP) scheme presented in [45] is used for rate matching. The SCL decoding with $L = 8$ is employed for all schemes. Thus, for hash-concatenated polar codes, $J = 9$ and $J' = 10$ (with $g_{10}(x)$) are applied. For CRC-polar codes and PC-polar codes, the CRC-19 (with $g_{19}(x) = x^{19}+x^{18}+x^{16}+x^{15}+x^{14}+x^{13}+x^{12}+x^{10}+x^{9}+x^{7}+x^{5}+x^{3}+x^{2}+x+1$) and CRC-16 (with $g_{16}(x) = x^{16}+x^{15}+x^{2}+1$) are employed, respectively. Fig. 10 shows that all codes are able to meet the FAR target with their critical numbers of check bits, and hash-concatenated polar codes have similar FER performance to CRC-polar codes, but outperform PC-polar codes, especially, when $K = 32$, about 0.35 dB coding gain can be obtained at FER=10^{-2}.

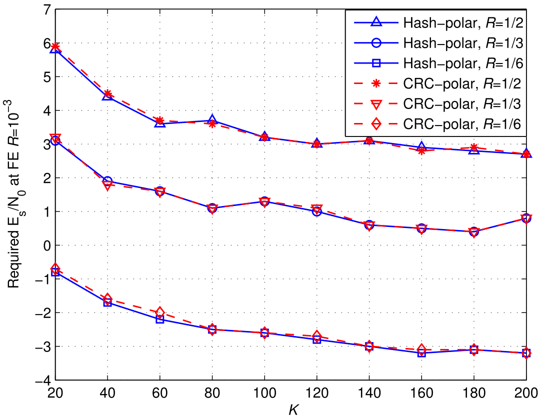

FIGURE 11. Performance comparison between hash-concatenated polar codes and CRC-polar codes with the parameters in the 5G standards.

Example 6: Recently, the agreement for 5G eMBB uplink control channel is reached, where the FAR target is 1.5×2^{-8} for $K \geq 20$ and $L = 8$. In Fig. 11, we compared our proposed hash-concatenated polar codes with CRC-polar codes in three code rates, i.e., 1/2, 1/3, 1/6 with K from 20 to 100. All codes are constructed by the sequence required in [40] with QPSK modulation. The puncturing and shorten methods proposed in [46] are used for rate matching, and the interleaver required in [46] is also applied. For the FAR target, CRC-11 (with $g_{11}(x) = x^{11} + x^{10} + x^{9} + x^{5} + 1$) is applied to the CRC-polar codes, and $J = 5$ and $J' = 6$ (with $g_6(x)$) are applied to the hash-concatenated polar codes. Fig. 11 shows that hash-concatenated polar codes also exhibit outstanding FER performance with the parameters in the 5G standards.

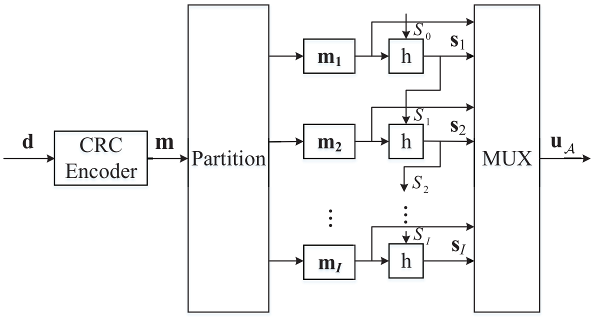

FIGURE 12. Segmented-hash-polar encoder.

B. SEGMENTED HASH-POLAR CODES FOR 5G EARLY-TERMINATION REQUIREMENT

In order to reduce the energy consumption per decoding attempt, the ability of ET is required in the control channel [30], especially for the downlink channel. For concatenated polar codes, the list decoder delivers the message in the path which satisfies the outer-check. However, if all the paths in the list fail to pass the check, the decoding is failed, where the all computations result in energy consumptions. Thus, the path test can be advanced to terminate the unnecessary decoding and reduce the computations, when the correct path is not in the list. To support ET, the check bits should be scattered within information bits.

1) ENCODING OF SEGMENTED HASH-POLAR CODES

Fig. 12 exhibits the segmented-hash-polar encoder. Similar to hash-polar codes, the sequence $(K + J')$-bit $\mathbf{m}$ is divided into I non-overlapped segments, i.e., $\mathbf{m} = (\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_I)$. Suppose that the length of sequence $\mathbf{m}_i$ is r_i, thus, $\sum_{i=1}^{I} r_i = K + J'$. Then, each segment is delivered into a hash function h, outputting a J_i-bit sequence $\mathbf{s}_i$ $(1 \leq i \leq I)$, the integer form of which is denoted by $\Theta(S_i)$. Then, each J_i-bit sequence will be attached to its corresponding segment $\mathbf{m}_i$ and form a $(K + J' + \sum_{i=1}^{I} J_i)$-bit sequence $\mathbf{u}_{\mathcal{A}}$. The obtained sequence $\mathbf{u}_{\mathcal{A}}$ is fed into the polar encoder, where the index order of the *information set* $\mathcal{A}$ is natural, which is different from the general polar encoder where the order based on reliabilities. Assume that a_i represents the end index of the i-th hash-sequence in $\mathcal{A}$, and $\hat{\mathcal{A}}$ denotes the set of a_i $(1 \leq i \leq I)$.

2) DECODING OF SEGMENTED HASH-POLAR CODES

Due to the scattered check bits, each decoded segment can be detected by the hash-sequence for each path. Once the decoded segment is failed to pass the check, the current path will be marked. If all paths are marked, the decoding termination exists. The main segmented hash-aided SCL algorithm presented by pseudo-code is shown in Algorithm 2.

The error-correcting performance comparison among the segmented hash-polar codes and the segmentation schemes, i.e., PSCL-CRC-polar codes, proposed in [47] and [48] with different numbers of segments I over the AWGN channel is shown in Fig. 13, where $N = 1024$, $R = 1/2$, $L = 8$, and the codes constructed by Monte-Carlo method at 2.0 dB. It can be seen that our proposed segmented hash-polar codes outperform PSCL-polar codes. Especially, when $I = 4$, about 0.15 dB and 0.1 dB gain can be obtained by the segmented hash-polar codes at FER $= 10^{-4}$ compared with the comparable PSCL-CRC-polar codes in [47] and [48], respectively.

Algorithm 2 Segmented Hash-Aided SCL Decoder

```
Input: Received channel probabilities.
Output: Estimated source sequence.
$Flag \leftarrow 0, j \leftarrow 1, \hat{j} \leftarrow 1$.
for $i = 1$ to $N$ do
    • Calculate the path metric $\mathrm{PM}_i[\cdot]$ by (4) and (7).
    if $i \in \mathcal{A}^c$ then
        Set $\hat{u}_i[\ell] = 0$ for all $\ell$, and keep all paths.
    else
        Sort the path metrics, and reserve the $L$ most probable paths with $\mathrm{PM}_i[1] < \mathrm{PM}_i[2] < \cdots < \mathrm{PM}_i[L] < \mathrm{PM}_i[\cdot] \in \{\mathrm{PM}_i[\ell], \ell \geq L\}$.
        if $i \in \hat{\mathcal{A}}$ then
            for $\ell = 1$ to $L$ do
                • Map $\{\hat{u}_t[\ell], t \in \{\mathcal{A}[\hat{j}], \ldots, \mathcal{A}[\hat{j} + r_i]\}\} \rightarrow \{\hat{k}_j\}$, and $\{\hat{u}_t[\ell], t \in \{\mathcal{A}[\hat{j} + r_i], \ldots, \hat{\mathcal{A}}[j]\}\} \rightarrow \hat{S}$.
                • Compute $\tilde{S}_j$ with $\tilde{S}_j = h(\tilde{S}_{j-1}, \hat{k}_j)$ and $\tilde{S}_0 = 0$.
                if $j == I$ then
                    if $\Theta(\tilde{S}_j) == \hat{S}$ then
                        $\hat{\mathbf{d}} \leftarrow \{\hat{u}_t[\ell], t \in \{\mathcal{A}[1], \ldots, \mathcal{A}[K]\}\}$.
                        $Flag \leftarrow 1$.
                        Break.
                else
                    if $(\Theta(\tilde{S}_j)) \neq \hat{S}$ then
                        The $\ell$-th path is marked.
            if all paths are marked then
                The decoding terminates.
            $\hat{j} \leftarrow \hat{j} + r_i + J_j$.
            $j \leftarrow j + 1$.
if $Flag == 0$ then
    $\hat{\mathbf{d}} \leftarrow \{\hat{u}_t[1], t \in \{\mathcal{A}[1], \ldots, \mathcal{A}[K]\}\}$.
```

3) ET GAIN

Based on the definition of ET, the ET gain representing the total saved computation ratio, can be given by [49]

$$\text{ET gain} = \Pr\{E_{\text{ET}}\} \cdot \eta_{\text{ET}},$$

where $\Pr\{E_{\text{ET}}\}$ and η_{ET} represent the probability of the ET event and the saved computation ratio from ET, respectively.

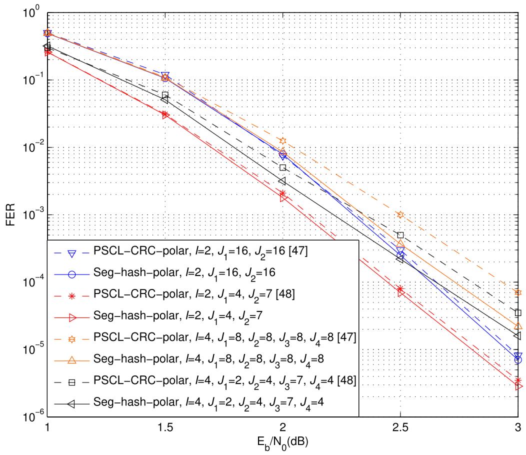

FIGURE 13. Performance comparison among segmented hash-polar codes and PSCL-CRC-polar codes with different numbers of segments over the AWGN channel, where $J' = 0$ for all segmented hash-polar codes.

Specifically, let E_{ET} represent the number of frames occurring ET, and E_{fail} denote the number of the incorrectly decoded frames. Then, the probability of the ET can be computed by

$$\Pr\{E_{\mathrm{ET}}\} = \frac{E_{\mathrm{ET}}}{E_{\mathrm{fail}}}.$$

Assume that N_l represents the number of undecoded unfrozen bits (include check bits) in l-th path. Then, the saved computation ratio from ET for segmented hash-polar codes is

$$\eta_{\mathrm{ET}} = \frac{\min\{N_l, 0 \le l \le L\}}{K + \sum_{i=1}^{I} J_i + J'}.$$

4) SELECTION OF SEGMENT LENGTH FOR FAR TARGET

Except for the supporting of the ET, segmented hash-polar codes also should achieve the FAR target for the control channel. For the blind detection and FAR requirement, a J'-bit CRC code is also employed to the segmented hash-polar codes to fulfill the detection. Then, the FAR of segmented hash-polar codes is defined as

$$\mathrm{FAR} = \frac{E_{\mathrm{hash_s\&crc}}}{E_{\mathrm{total}}}, \tag{13}$$

where $E_{\mathrm{hash_s\&crc}}$ represents the number of the incorrectly decoded frames which satisfy the check of all hash functions and the CRC simultaneously.

Lemma 3: Assume that the sequence is divided into two segments. Then,

$$\mathrm{FAR} \approx \frac{L \cdot \{\Pr\{E_1\} \cdot 2^{-(J_1+J_2+J')} + \Pr\{E_2\} \cdot 2^{-(J_2+J')}\}}{\Pr\{E_1\} + \Pr\{E_2\}}, \tag{14}$$

where $\Pr\{E_1\}$ represents the probability of the error occurring in the first segment, and $\Pr\{E_2\}$ represents the probability of the error only occurring in the second segment.

Proof: Consider one path. With the definition of $\Pr\{E_1\}$ and $\Pr\{E_2\}$, the probability of frame error is $\Pr\{E_1\} + \Pr\{E_2\}$. There are three cases, when error occurs.

The first one, the error occurs in the first segment (no matter whether the error exists in the second segment or not). Then, the error can be detected by all check bits, thus

$$\mathrm{FAR}(E_1) = \frac{\Pr\{E_1\} \cdot 2^{-(J_1+J_2+J')}}{\Pr\{E_1\} + \Pr\{E_2\}}.$$

The second case is that the error occurs only in the second segment. Since the first hash-function doesn't work, and the error can be detected by both the second hash-function and the CRC. Thus,

$$\mathrm{FAR}(E_2) = \frac{\Pr\{E_2\} \cdot 2^{-(J_2+J')}}{\Pr\{E_1\} + \Pr\{E_2\}}.$$

The last one is that the error occurs among the check bits. which can also be subdivided into two cases, where the one is that error only exists in check bits, and another is that error exists in both the information bits and check bits. For the case one, the FAR is zero, and for the case two, the FAR is approximately zero, due to the low probability of passing the check.

Based on the above, we can obtain that

$$\mathrm{FAR} \approx \frac{\Pr\{E_1\} \cdot 2^{-(J_1+J_2+J')} + \Pr\{E_2\} \cdot 2^{-(J_2+J')}}{\Pr\{E_1\} + \Pr\{E_2\}}.$$

Then consider L paths, the lemma is proved. □

Corollary 4: The case can also be extended to I segments, then

$$\mathrm{FAR} \approx \frac{L \cdot \sum_{i=1}^{I}\{\Pr\{E_i\} \cdot 2^{-(\sum_{k=i}^{I} J_k + J')}\}}{\sum_{i=1}^{I} \Pr\{E_i\}}, \tag{15}$$

where $\Pr\{E_i\}$ $(1 \le i \le I)$ represents the error occurs in the i-th segment and no error exists in the previous segments.

In practice, J_i and J' are fixed for all rates and code lengths. The above probabilities are associated with the segment lengths for $\sum_{i=1}^{I} r_i = K + J'$. Without a proper selection of each segment length, the FAR requirement may not be satisfied for specific SNRs. An off-line length search algorithm is given by pseudo-code in Algorithm 3, where the Monte-Carlo method is used to compute $\Pr\{E_i\}$ $(1 \le i \le I)$ for (14). Note that both the initialization algorithm and the search rule are not unique. For different search rules, the length allocations are also different. According to the undetected probability and Corollary 4, the segment length r_i may be proportional to the $\sum_{k=i}^{I} J_k$. However, the undecoded unfrozen bits will decrease as the code rate decreases, which implies that r_I can be small. Thus the coefficient $(1 + A - R)$ is used in the initialization process, where A is an offset to ensure that the initialization process works.

Proposition 3: The initialization process in Algorithm 3 works, where all segment lengths r_i are always greater than zero.

Proof: The proof is given in Appendix-C. □

Algorithm 3 Length-Search Algorithm

```
//Initial the length of $I$ segments
$J^* \leftarrow \sum_{i=1}^{I}\sum_{k=i}^{I} J_k,\ A \leftarrow \frac{J_I}{J^*}$.
for $i = 1$ to $I-1$ do
    $r_i \leftarrow \lfloor (K+J') \cdot \frac{\sum_{k=i}^{I} J_k}{J^*} \cdot (1+A-R) \rfloor$.
$r_I \leftarrow K + J' - \sum_{i=1}^{I-1} r_i$.
$Flag \leftarrow 0, j \leftarrow I, Count \leftarrow 1, t \leftarrow 0$.
//main loop
while $Flag$==0 do
    • Monte-Carlo method is used to compute all Pr$\{E\}$ defined in (15) at specific SNRs, where the source sequence is encoded by a $J'$-bit CRC and $I$ hash functions, and segmented hash-aided SCL decoder is also employed.
    • Calculate the FARs by (15) at specific SNRs.
    if one of the FARs fails to satify the target then
        $Flag \leftarrow 0$.
    else
        $Flag \leftarrow 1$.
    if $Flag$==0 then
        if $r_j > 1$ then
            $r_j \leftarrow r_j - 1$.
            $r_{j-1} \leftarrow r_{j-1} + 1$.
        if $j > 2$ then
            $j \leftarrow j - 1$.
        else
            for $k = 1$ to $I-1$ do
                $t \leftarrow \sum_{k}^{I-1} 1$.
                if $(Count \bmod t) == 0$ then
                    $j \leftarrow I + 1 - k$.
                    break.
        $Count \leftarrow Count + 1$.
    if $\sum_{i=2}^{I} r_i == (I-1)$ then
        $Flag \leftarrow 1$.
```

Consider Lemma 3, we note that if the Pr$\{E_2\}$ is small enough, the FAR will approach to $L \cdot 2^{-(J_1+J_2+J')}$. This means that, for given J_i and J', the FAR will decrease intensively with the decrease of Pr$\{E_I\}$, which suggests that r_I should be small, but smaller r_I leads to lower ET gain. Therefore, in Algorithm 3, the segment lengths $r_I, r_{I-1}, \ldots, r_2$ decrease successively until meet the FAR target.

Example 7: Assume that the FAR target is 2×10^{-5}, and QPSK signaling is transmitted over the AWGN channel. The FER and FAR performance of the proposed segmented hash-polar (seg-hash-polar) codes with different rates are shown in Fig. 14, where $I = 2$, $L = 8$, $J_1 = 3$, $J_2 = 7$, $J' = 9$ (with $g_9(x) = x^9 + x^5 + x^4 + x + 1$), and $K = 80$. For $R = 1/2, 1/3, 1/6$, the first segment length is determined by Algorithm 3 as 51, 57, 66, respectively. For comparison, the FER performance of CRC-polar codes is also given. All codes are also constructed by the GA method at 2.89 dB and the QUP scheme is used for rate matching. It can be seen that, under the FAR requirement, the segmented hash-polar codes have similar performance to CRC-polar codes, while supporting ET. We can also see that the computed FAR$_c$ by (15) is very close to the simulated FAR as shown in Fig. 14.

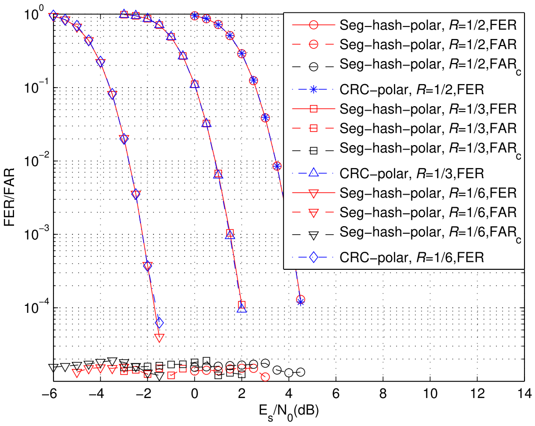

FIGURE 14. Performance comparison between segmented hash-polar codes and CRC-polar codes and $K = 80$ over the AWGN channel.

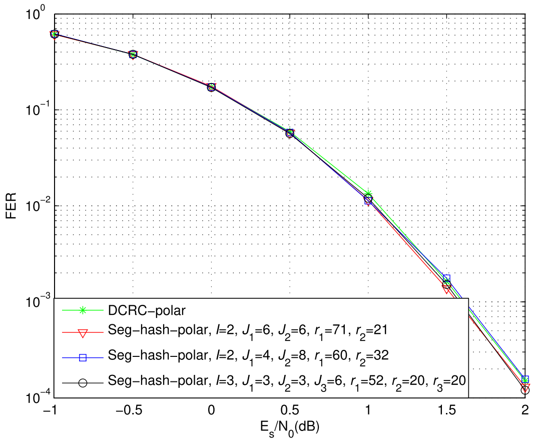

FIGURE 15. Performance comparison among segmented hash-polar codes with different numbers of segments and DCRC-polar codes with $K = 80$ and $R = 1/3$ over the AWGN channel.

Example 8:

In [40], the agreement for eMBB downlink control channel is also given, where the FAR target is 1.5×2^{-21} for $L = 8$. In Fig. 15 and Fig. 16, the error-correcting performance and ET gain of segmented polar codes and DCRC-polar codes are compared, respectively, where $K = 80$, $R = 1/3$, and different I and J_i for segment hash-polar codes are also considered. For the FAR target, CRC-24 (with $g_{24}(x) = x^{24} + x^{23} + x^{21} + x^{20} + x^{17} + x^{15} + x^{13} + x^{12} + x^8 + x^4 + x^2 + x + 1$) is applied to the DCRC-polar codes, and $J' = 12$ (with $g_{12}(x)$) is applied to the segment hash-polar codes. All codes

TABLE 2. Computed FAR of Seg-hash-polar in Example 8.

FAR_c $(\times 10^{-7})$ \ E_s/N_0 (dB) Schemes	-1.0	-0.5	0.0	0.5	1.0	1.5	2.0
I=2, J_1=6, J_2=6, r_1=71, r_2=21	5.35	5.56	5.96	5.80	5.17	5.40	5.23
I=2, J_1=4, J_2=8, r_1=60, r_2=32	5.04	5.07	5.51	5.63	5.29	6.60	6.47
I=3, J_1=3, J_2=3, J_3=6, r_1=52, r_2=20, r_3=20	5.43	5.96	5.86	6.55	6.66	6.80	5.51

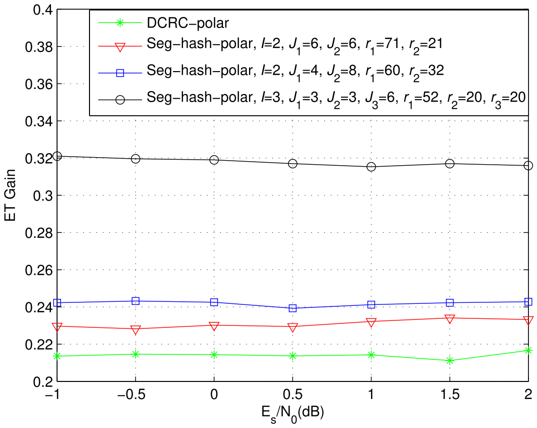

FIGURE 16. ET gain comparison among segmented hash-polar codes with different numbers of segments and DCRC-polar codes with $K = 80$ and $R = 1/3$ over the AWGN channel.

are constructed by the required sequence in [40] with QPSK modulation, and the method in [46] is used for rate matching.

From Fig. 15 and Fig. 16, we can see that, similar performance can be approached among segmented hash-polar codes with different I and J_i and DCRC-polar codes. For segmented hash-polar codes, the ET gain increases as the number of segments I increases. In addition, more ET gain can be obtained for segmented hash-polar codes compared with DCRC-polar codes. Due to the fact that the FAR is too low that cannot be simulated, the computed FAR_c is shown in Table 2. From Table 2, we can also see that segmented hash-polar codes by the given length allocations can meet the FAR requirement.

VI. CONCLUSION

A kind of concatenated polar codes named hash-polar codes has been proposed, where a hash encoder is used as an outer encoder for the flexible outer code length. It has illustrated that, under the 5G FAR requirement, the proposed hash-polar codes can obtain similar error-correcting performance to CRC-polar codes and outperform the PC-polar codes. By considering the collisions, partial hash-polar codes have been constructed based on low row-weights to improve the error-correcting performance at high SNRs. Furthermore, in order to support ET for 5G control channel, segmented hash-polar codes have been presented, where the ET gain can increase with the increase of the number of segments. Simulation results show that the proposed hash-polar codes and their variations can satisfy the requirements for the transmission of 5G control information, i.e., low FER and FAR performance and supporting ET.

A. "ONE-AT-A-TIME" HASH FUNCTION

The improved "one-at-a-time" hash function presented by pseudo-code is shown in Algorithm 4, where the parameters a, b, and h_s are 32-bit unsigned integers. The operations "$<<$" and "$>>$" represent left and right shifts, respectively.

Algorithm 4 Hash_Function

```
Input: $a$, $b$ (two 32-bit unsigned integers)
Output: $h_s$ (a 32-bit unsigned integer)
$h_s \leftarrow 3321836253$ XOR $b$
for $i = 1$ to $4$ do
    $h_s \leftarrow h_s + a$
    $h_s \leftarrow h_s + (h << 10)$
    $h_s \leftarrow h_s$ XOR $(h_s >> 6)$
    $a \leftarrow a >> 8$
$h_s \leftarrow h_s + (h_s << 3)$
$h_s \leftarrow h_s$ XOR $(h_s >> 11)$
$h_s \leftarrow h_s + (h_s << 15)$
```

Algorithm 5 Clamping

```
Input: $\mathbf{d_i}$, $r_i$
Output: $k_i$ (a 32-bit unsigned integer)
$t_1 \leftarrow (r_i - 1)/32 + 1$, $t_2 \leftarrow 31$, $t_3 \leftarrow 2^{32} - 1$,
    $count \leftarrow 1$, $k_i \leftarrow 0$.
for $i = 1$ to $t_1$ do
    if $i == t_1$ then
        $t_2 \leftarrow (r_i - 1) \bmod 32$.
    for $j = 0$ to $t_2$ do
        if $d_i[count] \neq 0$ then
            $k_i \leftarrow (k_i + (1 << j)) \bmod t_3$.
        $count \leftarrow count + 1$.
```

B. CLAMPING METHOD

Let the binary sequence $\mathbf{d}_i$ correspond to the i-th segment, and $d_i[j]$ $(1 \leq j \leq r_i)$ represent the j-th element in $\mathbf{d}_i$. The clamping method is described using pseudo-code in Algorithm 5.

C. PROOF OF PROPOSITION 3

When R tends to 1, $0 < r_i = \lfloor (K + J') \cdot \frac{\sum_{k=i}^{I} J_k}{J^*} \cdot A \rfloor$ for all $1 \leq r \leq I - 1$. Due to the fact that $\frac{\sum\limits_{i=1}^{I-1} \sum\limits_{k=i}^{I} J_k}{J^*} = \frac{J^* - J_I}{J^*} < 1$ and $A = \frac{J_I}{J^*} < 1$, then it is obvious that $r_I > 0$.

When R tends to 0, $0 < r_i = (K+J') \cdot \frac{\sum_{k=i}^{I} J_k}{J^*} \cdot (1+A)\rfloor$ for all $1 \leq r \leq I-1$. Then, $r_I = (K+J')(1 - \frac{\sum\limits_{i=1}^{I-1} \sum\limits_{k=i}^{I} J_k}{J^*} \cdot (1+A)) = (K + J')(1 - \frac{J^*-J_I}{J^*} \cdot \frac{J^*+J_I}{J^*}) = (K + J')(\frac{J_I}{J^*})^2 > 0$.
Thus, the initialization process works. □

REFERENCES

[1] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.

[2] I. Tal and A. Vardy, "List decoding of polar codes," in *Proc. IEEE Int. Symp. Inf. Theory*, St. Petersburg, Russia, Jul. 2011, pp. 1–5.

[3] K. Chen, K. Niu, and J. R. Lin, "List successive cancellation decoding of polar codes," *Electron. Lett.*, vol. 48, no. 9, pp. 500–501, Apr. 2012.

[4] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, "LLR-based successive cancellation list decoding of polar codes," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5165–5179, Oct. 2015.

[5] Z. Huang, S. Zhang, and F. Zhang, C. Duanmu, F. Zhong, and M. Chen, "Simplified successive cancellation decoding of polar codes with medium-dimensional binary kernels," *IEEE Access*, vol. 6, pp. 26707–26717, 2018.

[6] B. Li, H. Shen, and D. Tse, "An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check," *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2044–2047, Nov. 2012.

[7] K. Niu and K. Chen, "CRC-aided decoding of polar codes," *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668–1671, Oct. 2012.

[8] T. Wang, D. M. Qu, and T. Jiang, "Parity-check-concatenated polar codes," *IEEE Commun. Lett.*, vol. 20, no. 12, pp. 2342–2345, Sep. 2016.

[9] *Chairman's Notes Agenda Item 7.1.5 Channel Coding Modulation*, document R1-1613710, 3GPP, Nov. 2016. [Online]. Available: http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_87/Docs/R1-1613710.zip

[10] P. Chen, M. Xu, B. Bai, and J. Wang, "Design and performance of polar codes for 5G communication under high mobility scenarios," in *Proc. IEEE Veh. Technol. Conf.*, Sydney, NSW, Australia, Jun. 2017, pp. 1–5.

[11] *Polar Codes Design for UL Control*, document R1-1701033, 3GPP, Jan. 2017. [Online]. Available: http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_AH/NR_AH_1701/Docs/R1-1701033.zip

[12] J. Chen, Y. Chen, K. Jayasinghe, D. Du, and J. Tan, "Distributing CRC bits to aid polar decoding," in *Proc. IEEE Globecom Workshops*, Singapore, Dec. 2017, pp. 1–6.

[13] *Chairman's Notes Agenda Item 7.1.5 Channel Coding Modulation*, document R1-1701384, 3GPP, Jan. 2017. [Online]. Available: http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_AH/NR_AH_1701/Docs/R1-1701384.zip

[14] *Polar Codes Design for eMBB Control Channel*, document R1-1700242, 3GPP, Jan. 2017. [Online]. Available: http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_AH/NR_AH_1701/Docs/R1-1700242.zip

[15] *Chairman's Notes Agenda Item 8.1.4 Channel Coding*, document R1-1703980, 3GPP, Feb. 2017. [Online]. Available: http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_88/Docs/R1-1703980.zip

[16] *Design Polar Codes for eMBB Control Channel*, document R1-1702110, 3GPP, Feb. 2017. [Online]. Available: http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_88/Docs/R1-1702110.zip

[17] *Chairman's Notes Agenda Item 8.1.4 Channel Coding*, document R1-1706377, 3GPP, Apr. 2017. [Online]. Available: http://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_88b/Docs/R1-1706377.zip

[18] *Polar Codes Construction for eMBB Control Channel*, document R1-1704593, 3GPP, Apr. 2017. [Online]. Available: http://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_90/Docs/R1-1704593.zip

[19] M. Bakshi, S. Jaggi, and M. Effros, "Concatenated polar codes," in *Proc. IEEE Int. Symp. Inf. Theory*, Austin, TX, USA, Jun. 2010, pp. 918–922.

[20] H. Mahdavifar, M. El-Khamy, J. Lee, and I. Kang, "Performance limits and practical decoding of interleaved Reed–Solomon polar concatenated codes," *IEEE Trans. Commun.*, vol. 62, no. 5, pp. 1406–1417, May 2014.

[21] P. Trifonov and P. Semenov, "Generalized concatenated codes based on polar codes," in *Proc. IEEE Int. Symp. Wireless Commun. Syst.*, Aachen, Germany, Nov. 2011, pp. 442–446.

[22] Y. Wang, K. R. Narayanan, and Y.-C. Huang, "Interleaved concatenations of polar codes with BCH and convolutional codes," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 267–277, Feb. 2016.

[23] S. W. Yoon and J. Moon, "Low-complexity concatenated polar codes with excellent scaling behavior," in *Proc. IEEE Int. Conf. Commun.*, Paris, France, May 2017, pp. 3103–3115.

[24] P. Trifonov, "Efficient design and decoding of polar codes," *IEEE Trans. Commun.*, vol. 60, no. 11, pp. 3221–3227, Nov. 2012.

[25] P. Trifonov and V. Miloslavskaya, "Polar subcodes," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 2, pp. 254–266, Feb. 2016.

[26] P. Trifonov and G. Trofimiuk, "A randomized construction of polar subcodes," in *Proc. IEEE Int. Symp. Inf. Theory*, Aachen, Germany, Jun. 2017, pp. 1863–1867.

[27] H. Saber and I. Marsland, "Design of generalized concatenated codes based on polar codes with very short outer codes," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3103–3115, Apr. 2017.

[28] T. Murata and H. Ochiai, "On design of CRC codes for polar codes with successive cancellation list decoding," in *Proc. IEEE Int. Symp. Inf. Theory*, Aachen, Germany, Jun. 2017, pp. 1868–1872.

[29] S. A. Hashemi, C. Condo, F. Ercan, and W. J. Gross, "On the performance of polar codes for 5G eMBB control channel," in *Proc. IEEE 51st Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, USA, Oct./Nov. 2017, pp. 1764–1768.

[30] *Chairman's Notes Agenda Item 5.1.4 Channel Coding*, document R1-1711917, 3GPP, Jun. 2017. [Online]. Available: http://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_AH/NR_AH_1706/Docs/R1-1711917.zip

[31] H. Vangala, E. Viterbo, and Y. Hong. (Jan. 2015). "A comparative study of polar code constructions for the AWGN channel." [Online]. Available: https://arxiv.org/abs/1501.02473

[32] C. Liang, J. Hu, X. Ma, and B. Bai, "A new class of multiple-rate codes based on block Markov superposition transmission," *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4236–4244, Aug. 2015.

[33] B. Jenkins. *A Hash Function for Hash Table Lookup*. [Online]. Available: http://www.burtleburtle.net/bob/hash/doobs.html

[34] J. Perry, H. Balakrishnan, and D. Shah, "Rateless spinal codes," in *Proc. 10th ACM Workshop Hot Topics Netw.*, Cambridge, MA, USA, 2011, pp. 1–6.

[35] T. Kløve and V. I. Korzhik, *Error Detecting Codes*. Boston, MA, USA: Kluwer, 1995.

[36] M. Xu, P. Chen, B. Bai, and S. Tong, "Distance spectrum and optimized design of concatenated polar codes," in *Proc. IEEE Wireless Commun. Signal Process.*, Nanjing, China, Oct. 2017, pp. 1–6.

[37] B. Li, H. Shen, and D. Tse. (Jul. 2014). "A RM-Polar codes." [Online]. Available: https://arxiv.org/abs/1407.5483

[38] V. Bioglio, F. Gabry, I. Land, and J.-C. Belfiore, "Minimum-distance based construction of multi-kernel polar codes," in *Proc. IEEE Globecom*, Singapore, Dec. 2017, pp. 1–6.

[39] A. Eslami and H. Pishro-Nik, "On finite-length performance of polar codes: Stopping sets, error floor, and concatenated desig," *IEEE Trans. Commun.*, vol. 61, no. 3, pp. 919–929, Mar. 2013.

[40] *Chairman's Notes Agenda Item 6.1.4 Channel Coding*, document R1-1715121, 3GPP, Aug. 2017. [Online]. Available: http://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_90/Docs/R1-1715121.zip

[41] V. Bioglio, C. Condo, and I. Land. (Apr. 2018). "Design of polar codes in 5G new radio." [Online]. Available: https://arxiv.org/abs/1804.04389

[42] P. Giard, A. Balatsoukas-Stimming, and A. Burg, "Blind detection of polar codes," in *Proc. IEEE Int. Workshop Signal Process. Syst.*, Oct. 2017, pp. 1–6.

[43] C. Condo, S. A. Hashemi, and W. J. Gross, "Blind detection with polar codes," *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2550–2553, Dec. 2017.

[44] *Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures*, document 3GPP TS 36.213 V.8.2.0, 3GPP, Mar. 2009. [Online]. Available: http://www.3gpp.org

[45] K. Niu, K. Chen, and J.-R. Lin, "Beyond turbo codes: Rate-compatible punctured polar codes," in *Proc. IEEE Int. Conf. Commun.*, Budapest, Hungary, Jun. 2013, pp. 3423–3427.

[46] *Chairman's Notes Agenda Item 6.4 Channel Coding*, document R1-1716868, 3GPP, Sep. 2017. [Online]. Available: http://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_AH/NR_AH_1709/Docs/R1-1716868.zip

[47] S. Hashemi, A. Balatsoukas-Stimming, P. Giard, C. Thibeault, and W. J. Gross, "Partitioned successive-cancellation list decoding of polar codes," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Shanghai, China, Mar. 2016, pp. 957–960.

[48] S. A. Hashemi, M. Mondelli, S. H. Hassani, C. Condo, R. L. Urbanke, and W. J. Gross, "Decoder partitioning: Towards practical list decoding of polar codes," *IEEE Trans. Commun.*, vol. 66, no. 9, pp. 3749–3759, Sep. 2018.

[49] *Polar Codes Construction for NR Control Channel*, document R1-1712434, 3GPP, Aug. 2017. [Online]. Available: http://www.3gpp.org/ftp/TSG_RAN/WG1_RL1/TSGR1_90/Docs/R1-1712434.zip

PEIYAO CHEN (S'17) received the B.S. degree from Xidian University, China, in 2014, where she is currently pursuing the Ph.D. degree with the State Key Laboratory of Integrated Services Networks. From 2017 to 2018, she studied in McMaster University, Canada, as a Visiting Student. Her research interests include channel coding, modulation, and their applications to communication systems.

BAOMING BAI (S'98–M'00–SM'18) received the B.S. degree from the Northwest Telecommunications Engineering Institute, China, in 1987, and the M.S. and Ph.D. degrees in communication engineering from Xidian University, China, in 1990 and 2000, respectively. From 2000 to 2003, he was a Senior Research Assistant with the Department of Electronic Engineering, City University of Hong Kong. Since 2003, he has been with the State Key Laboratory of Integrated Services Networks, School of Telecommunication Engineering, Xidian University, where he is currently a Professor. In 2005, he was with the University of California at Davis as a Visiting Scholar. His research interests include information theory and channel coding, wireless communication, and quantum communication.

ZHAOFENG REN received the B.S. degree from the University of Electronic Science and Technology of China, in 2016. He is currently pursuing the M.S. degree with the State Key Laboratory of Integrated Services Networks, Xidian University, China. His research interests include channel coding, modulation, and their applications to communication systems.

JIAQING WANG received the Ph.D. degree in radio engineering from Southeast University, Nanjing, China, in 2006. He is currently a Senior Engineer with the State Key Laboratory of Wireless Mobile Communications, Telecommunications Science and Technology Institute. His research interests include communication standardization, channel coding, and wireless mobile communication.

SHAOHUI SUN received the M.S. degree in computer engineering and the Ph.D. degree in communication and information systems from Xidian University, Xi'an, China, in 1999 and 2003, respectively. From 2003 to 2006, he was a Post-Doctoral Fellow with the China Academy of Telecommunications Technology, Beijing, China. From 2006 to 2010, he was with Datang Mobile Communications Equipment Ltd., Beijing, where he has been deeply involved in the development and standardization of 3GPP. Since 2011, he has been the Chief Technical Officer with the Datang Wireless Mobile Innovation Center, CATT. His current research interest includes 5G and satellite communications.