Hybrid Decoding of CRC-Polar Codes

Wenxin Liu, Li Chen, Xingcheng Liu
School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou, China, 510006
Email: liuwx6@mail2.sysu.edu.cn, chenli55@mail.sysu.edu.cn, isslxc@mail.sysu.edu.cn

2021 13th International Conference on Wireless Communications and Signal Processing (WCSP) | 978-1-6654-0785-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/WCSP52459.2021.9613577

***Abstract*—Successive cancellation list (SCL) decoding of polar codes can yield excellent performance in error correction with the assistance of cyclic redundancy check (CRC) codes while its complexity remains high. This paper proposes a hybrid decoding for the CRC assisted polar codes, namely the CRC-polar codes. The proposed mechanism integrates the Fano decoding and the successive cancellation (SC) decoding. The Fano decoding will be first deployed to recover the message. However, when the received information is unreliable, it may stagger causing a large latency. To overcome this, we introduce a Fano decoding computation threshold. If a CRC validated estimation cannot be provided within the threshold, the SC decoding will be further deployed. As a result, both the decoding complexity and latency can be contained. The threshold-performance tradeoff insight of the hybrid decoding is also studied. Our simulation results demonstrate that the proposed hybrid decoding outperforms the SCL and the Fano decoding with a lower complexity.**

***Index Terms*—Cyclic redundancy check, Fano decoding, polar codes, successive cancellation list decoding.**

I. Introduction

Polar codes [1] can be theoretically proved to achieve the capacity of the binary input discrete memoryless channel (BI-DMC). In practice, this is realized by the successive cancellation (SC) decoding with a sub-quadratic complexity of $O(N\log_2 N)$, where N is the codeword length. However, in the finite length regime, the SC decoding performance remains inferior to that of the maximum likelihood (ML) decoding. By keeping multiple decoding paths, the SC list (SCL) decoding [2] was proposed to approach the ML performance with a complexity of $O(LN\log_2 N)$, where L is the list size. The use of cyclic redundancy check (CRC) codes can significantly enhance the SCL decoding performance [3]. This concatenation is often called the CRC-polar codes. The adaptive SCL (ASCL) decoding [4] was later proposed to avoid the unnecessary list expansion.

Projecting the SC decoding estimations over a binary tree, Fano decoding of polar codes was proposed [5]. It finds the correct decoding path over the decoding tree in a depth-first-search manner. The use of CRC in Fano decoding has also been considered. If a complete tree path cannot be validated by the CRC, the unreliable bits will be identified and flipped, triggering the subsequent decoding. However, when the received information is unreliable, the Fano decoding may stagger over the tree, causing in a large latency.

Addressing the above challenges, this paper proposes a hybrid message recovery mechanism that integrates the Fano decoding and the SC decoding, so that high performance of the CRC-polar codes can be achieved with a moderate cost of complexity and latency. The Fano decoding will be first invoked. To prevent it from lingering over the decoding tree, a decoding computation threshold is introduced. If the decoding cannot deliver a CRC validated codeword within the threshold, the SC decoding will be further invoked to find a complete path over the tree. The decoding threshold-performance tradeoff will be studied. Our simulation results will show that the hybrid decoding can outperform the SCL and the Fano decoding with a much lower complexity.

II. CRC-Polar Codes

A. Channel Polarization

Given a BI-DMC W: $\mathcal{X} \to \mathcal{Y}$, where $\mathcal{X} = \{0,1\}$ is the input alphabet and $\mathcal{Y} \in \mathbb{R}$ is the arbitrary output alphabet. Let $C(\mathrm{W})$ denote the capacity of channel W. Channel polarization can be explained by channel combing and splitting. The former combines N identical channels recursively into a vector channel W_N: $\mathcal{X}^N \to \mathcal{Y}^N$, where $N = 2^n$ and $n \in \mathbb{N}$. The latter splits the vector channel into N sub-channels $\mathrm{W}_N^{(i)}$: $\mathcal{X} \to \mathcal{Y}^N \times \mathcal{X}^{i-1}$, where $1 \le i \le N$. Let $C(\mathrm{W}_N)$ and $C(\mathrm{W}_N^{(i)})$ denote the capacity of the vector channel and the sub-channel i, respectively. They satisfy [1]

$$C(\mathrm{W}_N) = NC(\mathrm{W}) = \sum_{i=1}^{N} C(\mathrm{W}_N^{(i)}). \tag{1}$$

Let $[j]$ denote a permutation of $\{1, 2, \ldots, N\}$. Channel polarization leads to $C(\mathrm{W}_N^{[1]}) \le C(\mathrm{W}_N^{[2]}) \le \cdots \le C(\mathrm{W}_N^{[N]})$. Part of the N sub-channels become noisy as $C(\mathrm{W}_N^{(i)}) \to 0$, while the other part become noiseless as $C(\mathrm{W}_N^{(i)}) \to 1$. As $N \to \infty$, the above polarized portions become more significant. Let $P_{\mathrm{e}}^{(i)}$ denote the transmission error probability over sub-channel $\mathrm{W}_N^{(i)}$. Both $C(\mathrm{W}_N^{(i)})$ and $P_{\mathrm{e}}^{(i)}$ can be estimated via density evolution with a Gaussian approximation (GA) [6]. The N sub-channels can therefore be partitioned into the information set $\mathcal{A}$ and the frozen set $\mathcal{A}^c$. $\mathcal{A}$ collects the indices of the sub-channels with a large capacity, while $\mathcal{A}^c$ collects the remaining sub-channel indices. Information symbols and the redundancy (known as frozen symbols in polar coding language) will be transmitted through the sub-channels of $\mathcal{A}$ and $\mathcal{A}^c$, respectively.

Let $\mathbb{F}_2$ denote the binary field. The generator matrix of a length-N polar code is obtained by

$$\mathbf{G} = \mathbf{B}\mathbf{F}^{\otimes n}, \tag{2}$$

where $\mathbf{B} \in \mathbb{F}_2^{N\times N}$ is a bit-reversal permutation matrix [1], $\mathbf{F} = ((1,0),(1,1))^T \in \mathbb{F}_2^{2\times 2}$, and $\mathbf{F}^{\otimes n}$ is an n-th Kronecker power of $\mathbf{F}$.

978-1-6654-0785-4/21/$31.00 ©2021 IEEE

B. CRC-Polar Codes

Given a length-N vector $\underline{v} = (v_1, v_2, \ldots, v_N) \in \mathbb{F}_2^N$, we use $\underline{v}_{i_1}^{i_2}$ to denote its subvector $(v_{i_1}, v_{i_1+1}, \ldots, v_{i_2})$, where $1 \leq i_1 < i_2 \leq N$. Hence, $\underline{v}$ can also be denoted as $\underline{v}_1^N$. For an (N, K) CRC-polar code with a length-K_1 CRC, its inner code is an $(N, K+K_1)$ polar code and outer code is a $(K+K_1, K)$ CRC code. Given a message $\underline{m}_1^K = (m_1, m_2, \ldots, m_K) \in \mathbb{F}_2^K$, it can be written as $m(x) = m_1 + m_2 x + \cdots + m_K x^{K-1}$, the CRC codeword $\underline{s}_1^{K+K_1} \in \mathbb{F}_2^{K+K_1}$ is generated by

$$s(x) = x^{K_1} \cdot m(x) + (x^{K_1} \cdot m(x)) \bmod g(x), \tag{3}$$

where $s(x) = s_1 + s_2 x + \cdots + s_{K+K_1} x^{K+K_1-1}$ and $g(x)$ is the CRC generator polynomial. Let $\underline{u}_1^N \in \mathbb{F}_2^N$ and $\underline{c}_1^N \in \mathbb{F}_2^N$ denote information and codeword vectors of a CRC-polar code, respectively. Symbols of $\underline{s}_1^{K+K_1}$ will be placed in $\underline{u}_1^N$ at the positions indexed by $\mathcal{A}$ and $|\mathcal{A}| = K + K_1$. The rest positions of $\underline{u}_1^N$ (indexed by $\mathcal{A}^c$) are filled with frozen symbols. In this paper, we let the frozen symbols to be zero. The CRC-polar code can be generated by

$$\underline{c}_1^N = \underline{u}_1^N \mathbf{G} = \underline{u}_1^N \mathbf{B}\mathbf{F}^{\otimes n}. \tag{4}$$

III. The SC and Fano Decoding

A. The SC Decoding

It is assumed that codeword $\underline{c}_1^N$ is transmitted using binary modulation. Let $\underline{y}_1^N = (y_1, y_2, \ldots, y_N) \in \mathbb{R}^N$ and $\underline{\hat{u}}_1^N = (\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_N) \in \mathbb{F}_2^N$ denote the received symbol vector and the estimated message, respectively. If u_i is a frozen symbol, i.e., $i \in \mathcal{A}^c$, then $\hat{u}_i = 0$. Otherwise, its log likelihood ratio (LLR) will be computed as

$$L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1}) = \ln \frac{P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} | u_i = 0)}{P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} | u_i = 1)}, \tag{5}$$

where $P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} | u_i)$ is the conditional channel transition probability. The above LLR can be recursively calculated

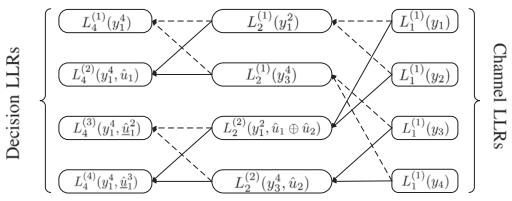

Fig. 1. The SC decoding trellis with $N = 4$. The dashed and solid arrows denote f_+ and f_- updates, respectively.

through a trellis that is shown in Fig.1. At each layer, $L_N^{(i)}$ denotes the LLR of symbol u_i of a length-N polar code. They can be calculated using those of two length-$N/2$ subcodes, which can be seen as the splitting results. Therefore, given a length-N polar code, after splitting $\log_2 N$ times, N symbols will be reached. Their LLRs can be determined by

$$L_1^{(1)}(y_i) = \ln \frac{P(y_i | u_1 = 0)}{P(y_i | u_1 = 1)}, \tag{6}$$

where $1 \leq i \leq N$. Let us define

$$f_+(\alpha, \beta) \triangleq \ln \frac{e^{\alpha+\beta} + 1}{e^\alpha + e^\beta}, \tag{7}$$

$$f_-(\alpha, \beta, \xi) \triangleq (-1)^\xi \alpha + \beta. \tag{8}$$

where $\alpha, \beta \in \mathbb{R}$ and $\xi \in \mathbb{F}_2$. The LLRs over the trellis can be recursively computed in a layer-by-layer manner as [7]

$$L_N^{(2i-1)}(\underline{y}_1^N, \underline{\hat{u}}_1^{2i-2}) = f_+(L_{N/2}^{(i)}(\underline{y}_1^{N/2}, \underline{\hat{u}}_{1,\mathrm{o}}^{2i-2} \oplus \underline{\hat{u}}_{1,\mathrm{e}}^{2i-2}), L_{N/2}^{(i)}(\underline{y}_{N/2+1}^N, \underline{\hat{u}}_{1,\mathrm{e}}^{2i-2})), \tag{9}$$

$$L_N^{(2i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{2i-1}) = f_-(L_{N/2}^{(i)}(\underline{y}_1^{N/2}, \underline{\hat{u}}_{1,\mathrm{o}}^{2i-2} \oplus \underline{\hat{u}}_{1,\mathrm{e}}^{2i-2}), L_{N/2}^{(i)}(\underline{y}_{N/2+1}^N, \underline{\hat{u}}_{1,\mathrm{e}}^{2i-2}), \hat{u}_{2i-1}), \tag{10}$$

where $\underline{\hat{u}}_{1,\mathrm{o}}^{2i-2}$ and $\underline{\hat{u}}_{1,\mathrm{e}}^{2i-2}$ are the odd and the even entries of $\underline{\hat{u}}_1^{2i-2}$, respectively. Finally, decision on the message symbols will be made based on $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$. That says for $i \in \mathcal{A}$, if $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1}) > 0$, $\hat{u}_i = 0$; otherwise, $\hat{u}_i = 1$.

B. The Fano Decoding

It is assumed that u_i ($i \in \mathcal{A}$) is uniformly distributed in $\mathcal{X}$. The *a posteriori* probability (APP) $P(u_i | \underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$ can be determined by

$$P(u_i | \underline{y}_1^N, \underline{\hat{u}}_1^{i-1}) = \left(1 + \left(\frac{P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} | u_i = 0)}{P(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1} | u_i = 1)}\right)^{(-1-2u_i)}\right)^{-1}. \tag{11}$$

The above SC decoding estimation can be projected over a binary decoding tree as Fig.2. Nodes of each layer denote the APPs of (11), while the branch leading to a node denotes the decision on the prior estimated symbol, i.e., $\hat{u}_{i-1}$. The binary tree that illustrates the decoding of a length-N polar code has $N + 1$ layers. At layer i ($i \in \mathcal{A}$), the branch that leads to a larger APP will be chosen. A complete root-to-leaf path corresponds to an SC decoding estimation. Fig.2 shows the estimated message is $\underline{\hat{u}}_1^4 = (0, 1, 1, 0)$, which is incorrect. This can be improved by revising the APPs.

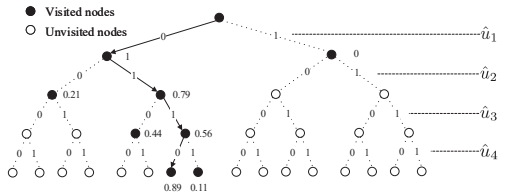

Fig. 2. Binary tree illustration of the SC decoding with $N = 4$, $\mathcal{A} = \{2, 3, 4\}$, $\mathcal{A}^c = \{1\}$ and $\underline{u}_1^4 = (0, 1, 0, 1)$.

Accumulating all APPs of a path from $\hat{u}_1$ to $\hat{u}_i$ results in

$$\prod_{t=1}^{i} \max_{u_t \in \mathbb{F}_2} \{P(u_t | \underline{y}_1^N, \underline{\hat{u}}_1^{t-1})\}, \tag{12}$$

which indicates the reliability of the estimation. This path reliability metric is upper bounded by the maximum *a posteriori*

probability (MAP) decoding metric as

$$\prod_{t=1}^{i}\max_{u_t\in\mathbb{F}_2}\{P(u_t|\underline{y}_1^N,\underline{\hat{u}}_1^{t-1})\}\le\max_{u_1^i\in\mathbb{F}_2^i}\{\prod_{t=1}^{i}P(u_t|\underline{y}_1^N,\underline{\hat{u}}_1^{t-1})\}. \tag{13}$$

Further assuming the codeword symbols are independent, the above upper bound can be approximated by

$$\max_{u_1^i\in\mathbb{F}_2^i}\{\prod_{t=1}^{i}P(u_t|\underline{y}_1^N,\underline{\hat{u}}_1^{t-1})\}\approx\prod_{t=1}^{i}(1-P_{\mathrm{e}}^{(t)}), \tag{14}$$

where $P_{\mathrm{e}}^{(i)}$ is the transmission error probability. This approximation can be used to tune the accumulated path metric of (12). In the LLR domain, they can be written as

$$\begin{aligned}\mathcal{M}(\underline{\hat{u}}_1^i)&=\ln\left(\prod_{t=1}^{i}\frac{P(u_t|\underline{y}_1^N,\underline{\hat{u}}_1^{t-1})}{1-P_{\mathrm{e}}^{(t)}}\right)\\&=\sum_{t=1}^{i}\ln\left(\frac{P(u_t|\underline{y}_1^N,\underline{\hat{u}}_1^{t-1})}{1-P_{\mathrm{e}}^{(t)}}\right)\\&=\mathcal{M}(\underline{\hat{u}}_1^{i-1})+\ln\left(\frac{P(u_i|\underline{y}_1^N,\underline{\hat{u}}_1^{i-1})}{1-P_{\mathrm{e}}^{(i)}}\right),\end{aligned} \tag{15}$$

with an initial value $\mathcal{M}(\hat{u}_1^0)=0$. Over the binary tree, $\mathcal{M}(\underline{\hat{u}}_1^{i-1})$ and $\mathcal{M}(\underline{\hat{u}}_1^i)$ are the path metrics of the nodes at layer $i-1$ and i, respectively. They have two realizations. E.g., the accumulated path metric $\mathcal{M}(\underline{\hat{u}}_1^i)$ can be realized as $\mathcal{M}(\underline{\hat{u}}_1^i|\hat{u}_i=0)$ and $\mathcal{M}(\underline{\hat{u}}_1^i|\hat{u}_i=1)$, which correspond to the path taking the left and the right branches from the node at layer $i-1$, respectively. A larger accumulated metric of $\mathcal{M}(\hat{u}_1^i)$ indicates the path that yields $\underline{\hat{u}}_1^i$ would be more reliable. The decoding finds a complete path, yielding $\underline{\hat{u}}_1^N$, in the manner of pursuing a larger accumulated metric. In practice, an accumulated metric threshold T is used to prevent the decoding from exploring too many paths. Only the paths with an accumulated metric greater than T will be considered. Note that T will be dynamically adjusted with a step size Δ, so that the decoding always yields a complete path.

Fig.3 demonstrates the Fano decoding with the threshold initialized as $T=-4$ and $\Delta=4$. Note that at layer i, if

$$\max_{\hat{u}_i\in\mathbb{F}_2}\{\mathcal{M}(\underline{\hat{u}}_1^i)\}\ge T, \tag{16}$$

the node with a greater accumulated path metric will be explored. However, if

$$\max_{\hat{u}_i\in\mathbb{F}_2}\{\mathcal{M}(\underline{\hat{u}}_1^i)\}<T, \tag{17}$$

the decoding path will move back to a nearest visited node which has a path metric

$$\min_{\hat{u}_{i'}\in\mathbb{F}_2}\{\mathcal{M}(\underline{\hat{u}}_1^{i'})\}\ge T, \tag{18}$$

where $i'<i$, and the two branches that stretch from the node have not been fully explored. Afterwards, the decoding will explore a new path from the node. A node update according to the above rules is called *an iteration of the Fano decoding*. Once the decoding yields a complete path, a codeword estimation $\underline{\hat{u}}_1^N$ will be provided. Fig.3 shows that with the same decoding task of Fig.2, the Fano decoding provides the correct estimation.

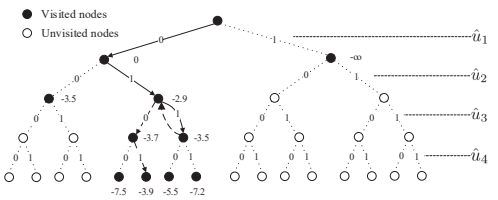

Fig. 3. Binary tree illustration of the Fano decoding with $N=4$, $\mathcal{A}=\{2,3,4\}$, $\mathcal{A}^c=\{1\}$ and $\underline{u}_1^4=(0,1,0,1)$.

It should be pointed out that if the step size $\Delta=\infty$, Fano decoding dissolves into the SC decoding. However, if Δ is too small, the Fano decoding may stagger, lingering over the binary tree without providing a complete path. This often happens when the received information is unreliable. To overcome this, the following hybrid decoding is proposed.

IV. The Hybrid Decoding

A. The Fano-SC Integration

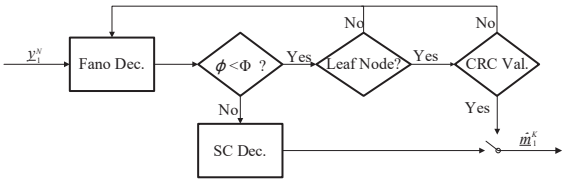

Fig. 4. Block diagram of the hybrid decoding.

Fig.4 shows the block diagram of the hybrid decoding. It integrates the above mentioned Fano decoding and SC decoding. The Fano decoding will be first deployed to recover the message. To prevent it from lingering over the binary tree, a decoding LLR computation threshold will be imposed, where the LLR computation is defined as the recursive operations of (9) and (10). In this work, the threshold is defined as a multiplicity of the SC decoding complexity, i.e.,

$$\Phi=\eta N\log_2 N, \tag{19}$$

where η is a positive integer. Once the number of LLR computations, denoted as ϕ, reaches Φ, and the Fano decoding still cannot deliver a CRC validated estimation, the SC decoding will be deployed. In particular, once ϕ reaches Φ and the Fano decoding has estimated $\underline{\hat{u}}_1^{i-1}$, the SC decoding will recover the remaining $\underline{\hat{u}}_i^N$. The estimated message $\underline{\hat{m}}_1^K$ can be further obtained. The following Algorithm 1 demonstrates the hybrid decoding process.

B. Decoding Insight

The above description shows that a larger LLR computation threshold Φ can lead to more Fano decoding efforts and a better decoding performance. We now study the threshold-performance relationship, aiming to provide more insight of the hybrid decoding.

Algorithm 1 The Hybrid Decoding.

Input: $\underline{y}_1^N$, Φ, Δ, T;
Output: $\underline{\hat{m}}_1^K$;
1: Initialize $\phi = 0$ and $i = 1$;
2: **If** $\phi < \Phi$ **do**
3: Compute $\mathcal{M}(\underline{\hat{u}}_1^i)$ as in (15) and update ϕ;
4: Perform an iteration of the Fano decoding;
5: Estimate $\hat{u}_{i'}$ with $i' \leq i$ and let $i = i' + 1$;
6: **If** $i = N + 1$ **do**
7: Retrieve $\underline{\hat{m}}_1^K$ from $\underline{\hat{u}}_1^N$;
8: **If** the CRC validates $\underline{\hat{m}}_1^K$ **do**
9: Terminate the decoding;
10: **Else**
11: Let $i = N - 1$ and go to 3;
12: **Else**
13: Compute $L_N^{(i)}(\underline{y}_1^N, \underline{\hat{u}}_1^{i-1})$ as in (5) and update ϕ;
14: Estimate $\hat{u}_i$ and let $i = i + 1$;
15: Retrieve $\underline{\hat{m}}_1^K$ from $\underline{\hat{u}}_1^N$ and terminate the decoding;

It is known that the decoding computation distribution of the sequential decoding (e.g., the Fano decoding) follows the Pareto distribution [8]. However, a closed form distribution for CRC-polar codes is yet to be developed. Instead, we provide the empirical results to reveal the computation distribution for Fano decoding of CRC-polar codes.

Let ψ denote the number of LLR computations required by the Fano decoding for delivering a CRC validated estimation. It can be normalized by the SC decoding complexity as

$$\tilde{\psi} = \left\lfloor \frac{\psi}{N\log_2 N} \right\rfloor. \tag{20}$$

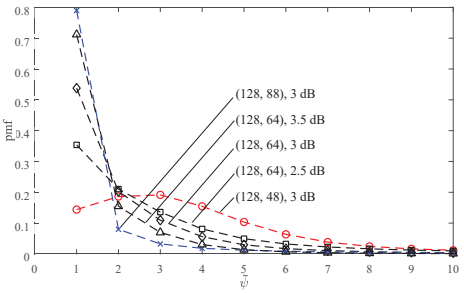

Fig. 5. The pmf of $\tilde{\psi}$ in decoding the length-128 CRC-polar code.

Fig. 5 shows the probability mass function (pmf) of $\tilde{\psi}$ for different rate CRC-polar codes. They are obtained over the additive white Gaussian noise (AWGN) channel with the signal-to-noise ratio (SNR) of 2.5 dB, 3 dB and 3.5 dB, respectively. It shows that with the same SNR, Fano decoding of a higher rate code is more likely to yield a CRC validated message with a lower decoding complexity. Meanwhile, for the same code, i.e., the (128, 64) CRC-polar code, less decoding computation would be needed if the SNR increases.

Given a decoding computation threshold Φ of (19), it can be normalized to η. Let $P_{\mathrm{F}}(\tilde{\psi} < \eta)$ denote the probability that a CRC validated message can be provided with less than $\eta N\log_2 N$ LLR computations. The average number of LLR computations of this hybrid decoding can be characterized as

$$\mathbb{E}_\eta[\phi] = \sum_{\tilde{\psi}=1}^{\eta-1} P_{\mathrm{F}}(\tilde{\psi} < \eta)\tilde{\psi} N\log_2 N + (1 - \sum_{\tilde{\psi}=1}^{\eta-1} P_{\mathrm{F}}(\tilde{\psi} < \eta))\eta N\log_2 N. \tag{21}$$

In the worst case, no CRC validated message can be yielded within the threshold, $\mathbb{E}_\eta[\phi] = \eta N\log_2 N$. Table I shows for the (128, 64) CRC-polar code, how the average LLR computations $\mathbb{E}_\eta[\phi]$ varies w.r.t. the threshold η and the AWGN channel condition. It can be seen that a larger computation threshold results in a higher average LLR computations. The average LLR computations decrease as the SNR increases.

TABLE I
COMPLEXITY $\mathbb{E}_\eta[\phi]$ IN DECODING THE (128, 64) CRC-POLAR CODE

η	1	4	16	64
SNR $= 1$ dB	1.73×10^3	4.10×10^3	1.12×10^4	2.79×10^4
SNR $= 2$ dB	1.56×10^3	3.22×10^3	5.52×10^3	9.04×10^3
SNR $= 3$ dB	1.32×10^3	2.08×10^3	2.41×10^3	2.57×10^3

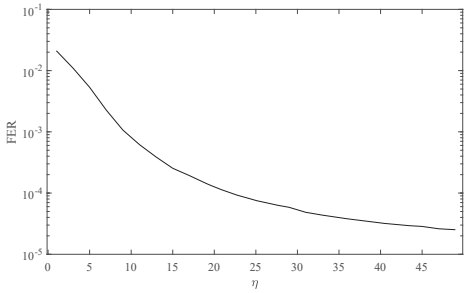

Fig. 6. The performance-threshold relationship.

Finally, Fig.6 shows the relationship between the decoding frame error rate (FER) and the computation threshold η for the (128, 64) CRC-polar code. It is simulated over the AWGN channel with SNR = 4 dB. It shows that better decoding performance can be obtained by increasing the computation threshold. For this code, if a decoding FER of 10^{-4} is targeted, a decoding threshold η of at least 20 would be needed.

V. Simulation Results

Our simulation results on decoding and complexity performances were obtained over the AWGN channel using BPSK. A length-8 CRC code with generator polynomial $g(x) = x^8 + x^2 + x + 1$ is used. The polar codes were designed using GA at the SNR of 0 dB.

A. Decoding Performance

Figs. 7 and 8 show the FER performance of the (128, 64) and the (256, 128) CRC-polar codes, respectively. Note that the SC and Fano decoding of polar codes of the same rate are used as comparison benchmarks. It can be seen that performance of the hybrid decoding can be improved by increasing the LLR computation threshold η. For the (128,

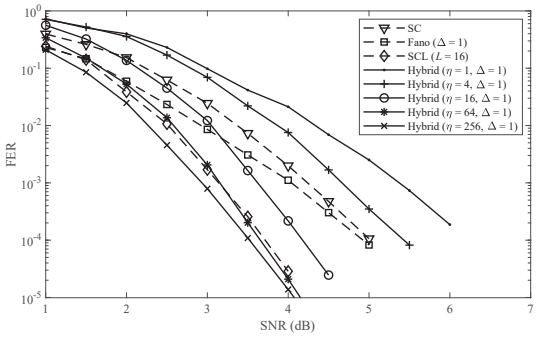

Fig. 7. Performance of the (128, 64) CRC-polar code.

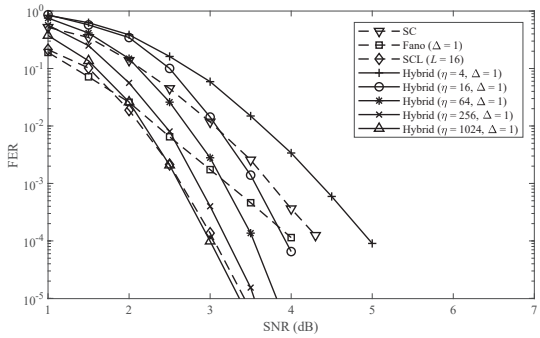

Fig. 8. Performance of the (256, 128) CRC-polar code.

64) CRC-polar code, when $\eta = 16$, the hybrid decoding can substantially outperform both the SC and the Fano decoding. When $\eta = 64$, it starts to outperform the SCL decoding with $L = 16$. A similar phenomenon can be observed for the (256, 128) CRC-polar code. It should be pointed out that the hybrid decoding exhibits a far smaller complexity than the SCL decoding, especially when the SNR increases, as demonstrated below.

B. Decoding Complexity

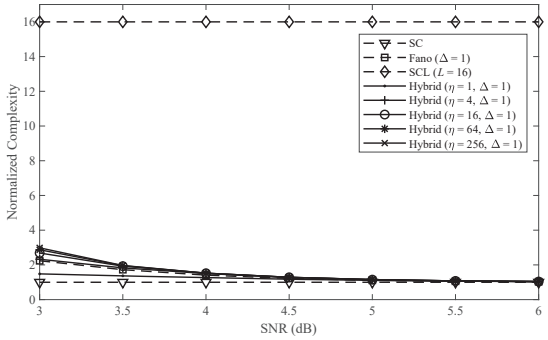

Fig. 9. Normalized complexity in decoding the (128, 64) CRC-polar code.

Complexity of all decoding algorithms are normalized by a factor of $N\log_2 N$. For the proposed hybrid decoding, it is $\phi/N\log_2 N$. Figs. 9 and 10 show the normalized complexity in decoding the (128, 64) and the (256, 128) CRC-polar codes, respectively. It can be seen that a larger LLR computation threshold results in a higher decoding complexity. But they converge as the SNR increases. For the (128, 64) CRC-polar code, Fig. 7 shows the hybrid decoding with $\eta = 64$ starts

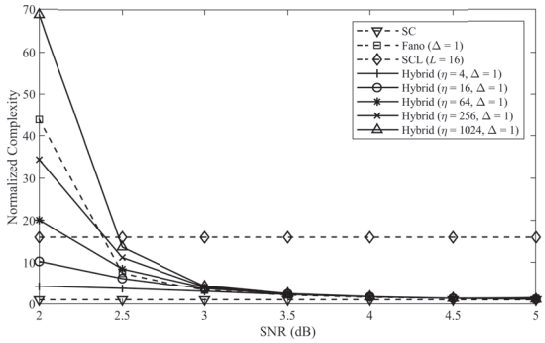

Fig. 10. Normalized complexity in decoding the (256, 128) CRC-polar code.

to outperform the SCL decoding with $L = 16$. Fig. 9 shows that over the interested SNR region (3 - 6 dB for the code), the proposal exhibits a far lower complexity than the SCL decoding. For the (256, 128) CRC-polar code, the interested SNR region is 2 - 5 dB, in which Fig. 10 shows the complexity advantage of the hybrid decoding with $\eta = 1024$ over the SCL decoding with $L = 16$ starts to appear with an SNR of 2.5 dB. Moreover, Fig. 8 shows the hybrid decoding with $\eta = 16$ outperforms the Fano decoding, while Fig. 10 shows the former has a lower complexity.

Since the SCL decoding can be processed in parallel, it has a lower average latency than the hybrid decoding. But this advantage will decrease rapidly as SNR increases.

VI. Conclusion

This paper has proposed a hybrid decoding for CRC-polar codes, realizing a high decoding performance with a moderate complexity and a contained latency. It integrates the Fano decoding and the SC decoding. If the Fano decoding cannot deliver a CRC validated estimation within an LLR computation threshold, the SC decoding will be further deployed. Insight of the hybrid decoding has also been studied, revealing its threshold-performance tradeoff. Simulation results have shown that the hybrid decoding can outperform the SCL and the Fano decoding with a lower complexity.

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," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213-2226, May 2015.

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

[4] 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, Dec. 2012.

[5] M. Jeong and S. Hong, "SC-Fano decoding of polar codes," *IEEE Access*, vol. 7, pp. 81682-81690, jun. 2019.

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

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

[8] I. Jacobs and E. Berlekamp, "A lower bound to the distribution of computation for sequential decoding," *IEEE Trans. Inf. Theory*, vol. 13, no. 2, pp. 167-174, Apr. 1967.