

RESEARCH

Open Access



Enhanced belief propagation decoding of polar codes by adapting the parity-check matrix

Guiping Li^{1*†} , Jianjun Mu^{1,2†}, Xiaopeng Jiao^{1†}, Junjun Guo³ and Xiaohang Liu⁴

Abstract

Though the performance of belief propagation (BP) decoder for polar codes is comparable with the successive cancellation (SC) decoder, it has significant performance gap when compared with the improved SC decoders, such as SC list (SCL) decoding. In this paper, we propose an improved BP decoding for polar codes with good performance by adapting their parity-check matrices. The decoding process is iterative and consists of two parts. Firstly, the parity-check matrix of polar codes is adjusted such that one of its submatrices corresponding to less reliable bits is in a sparse nature. Secondly, the BP decoding is applied to the adjusted parity-check matrix. Simulation results show that the proposed decoder, when accompanied with the early termination scheme, provides significant performance gains over the original polar BP decoder under a relatively low decoding complexity and even competes with the cyclic redundancy check (CRC)-aided SCL (CRC-SCL) decoder only by increasing a tolerate complexity.

Keywords: Polar codes, Belief propagation, Parity-check matrix adaptation, early termination

1 Introduction

Polar codes are shown to be capacity-achieving for binary input discrete memoryless channel (B-DMC) with explicit construction and decoding methods [1]. In the regime of finite block lengths with practical considerations, the performance of polar codes is not very appealing. One of the reasons is that the successive cancellation (SC) decoding for polar codes is suboptimal and it may cause error propagation due to the serial decoding architecture. Although there are some improved SC decoders [2, 3], the throughput of these decoders is low and the decoding latency is relatively high. Another interesting decoding method for polar codes is belief propagation (BP) decoding [4–11]. One advantage of BP decoder is its fully parallel architecture which is important for possible applications. However, the performance of BP decoders reported in the literatures is inferior to the improved SC decoders.

Since the parity-check matrix of polar codes is not sparse in general [12], running BP decoding directly on

it does not obtain good frame error rate (FER) performances. Thus, the proposed BP decoders in [4, 5] are based on the factor graph representation of the generator matrix. In this paper, based on the idea of iterative soft-input-soft-output decoding of Reed-Solomon (RS) codes [13], we propose an improved BP decoder for polar codes by adapting their parity-check matrices. In particular, we first adapt the parity-check matrix according to the bit reliability so that the unreliable bits correspond to a sparse submatrix. Then, the BP decoding algorithm is applied to the adapted parity-check matrix. The above process is iteratively performed until the predefined maximum number of iterations is reached or the stopping criterion is satisfied.

2 Preliminaries

2.1 Polar codes

Let W denote a B-DMC with input $u \in \{0, 1\}$, output $y \in \mathcal{Y}$, and transition probabilities $W(y|u)$. By recursively applying channel combining and splitting operations on N ($N = 2^n$, $n \geq 1$) independent copies of W , we can get a set of N polarized subchannels, denoted by $W_N^{(i)}(y_1^N, u_1^{i-1}|u_i)$, where $i = 1, 2, \dots, N$. The construction of an (N, K) polar code is completed by choosing

*Correspondence: 15693685@qq.com

†Equal contributors

¹The School of Computer Science and Technology, Xidian University, TaiBai Road, Xi'an, China

Full list of author information is available at the end of the article

the K best polarized subchannels which carry information bits and freezing the remaining subchannels to zero or some fixed values. According to the encoding rule of polar codes, the encoder can first generate the source block u_1^N . Then, polar coding is performed on the constraint $x_1^N = u_1^N G_N$, where G_N is the generator matrix and the vector x_1^N is the codeword. The matrix G_N is obtained by the formula $B_N F^{\otimes n}$, where $F^{\otimes n}$ is the n th Kronecker power of $F \triangleq \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ and B_N is the bit-reversal permutation matrix. The popular decoding algorithms of polar codes are SC algorithm and BP algorithm.

2.2 Successive cancellation decoding of polar codes

In [1], SC decoding is proposed as a baseline algorithm which has a very low complexity $N \log N$, where N is the codeblock length. Based on the recursive structure of the polar encoder, the SC decoder performs a series of interleaved step-by-step decisions in which a decision in each step heavily depends on the decisions in the previous steps and the received sequence y_1^N from channel. We define the log-likelihood ratio (LLR) of the i th bit as

$$L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) = \log \frac{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1} | 0)}{W_N^{(i)}(y_1^N, \hat{u}_1^{i-1} | 1)}. \quad (1)$$

Decisions are taken according to

$$\hat{u}_i = \begin{cases} 0 & \text{if } i \in I \text{ and } L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) \geq 0 \\ 1 & \text{if } i \in I \text{ and } L_N^{(i)}(y_1^N, \hat{u}_1^{i-1}) < 0 \\ 0 & \text{if } i \in I^c, \end{cases} \quad (2)$$

where the vector I denotes the index set of information subchannels and I^c is the complementary of the set I , and the decision LLR $L_N^{(i)}(y_1^N, \hat{u}_1^{i-1})$ can be straightly calculated using the following recursive formulas according to [1, 14]

$$\begin{aligned} L_N^{(2i-1)}(y_1^N, \hat{u}_1^{2i-2}) \\ = 2 \tanh^{-1} \left(\tanh \left(L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,e}^{2i-2} \oplus \hat{u}_{1,o}^{2i-2}) / 2 \right) \right. \\ \left. \times \tanh \left(L_{N/2}^{(i)}(y_{N/2+1}^N, \hat{u}_{1,o}^{2i-2}) / 2 \right) \right), \end{aligned} \quad (3)$$

or

$$\begin{aligned} L_N^{(2i)}(y_1^N, \hat{u}_1^{2i-1}) &= L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,e}^{2i-1}) \\ &+ (-1)^{\hat{u}_{1,e}^{2i-1}} L_{N/2}^{(i)}(y_1^{N/2}, \hat{u}_{1,e}^{2i-2} \oplus \hat{u}_{1,o}^{2i-2}), \end{aligned} \quad (4)$$

where $\hat{u}_{1,e}^{2i-2}$ and $\hat{u}_{1,o}^{2i-2}$ denote the subvectors which consist of elements of \hat{u}_1^{2i-2} with even and odd indices, respectively, and the symbol \oplus denotes modulo-2 addition. From

the above decoding process, we can see that the complexity is determined essentially by the complexity of computing the LLRs.

2.3 Belief propagation decoding of polar codes

To perform BP decoding of polar codes, the authors in [4] show that the factor graph of polar codes can be obtained by adding N check nodes to each column of the first n ($n = \log N$) columns from left to right in the encoding graph. During the whole BP iteration process, soft messages are updated and propagated among adjacent nodes from the rightmost column to the leftmost column. Then, the decoder reverses the course and updates schedule toward the rightmost column. This procedure makes one round of BP iteration. After reaching predefined number of BP iteration M_{BP} , the decision sequence \hat{u}_1^N is determined based on the hard decision of messages from the nodes in the leftmost column. However, the performance gains of BP decoder are not significant over the binary-input additive white Gaussian noise (BAWGN) channel [12].

3 Enhanced belief propagation decoding of polar codes by adapting the parity-check matrix

3.1 Enhanced belief propagation decoding of polar codes

Generally, the BP decoder for polar codes is based on the factor graph representation obtained by the encoding graph of polar codes [4]. In [12], the authors proved that the parity-check matrix H of polar codes is formed by the columns of G_N with indices in I^c , where I^c is the index set of frozen channels. However, no one considers to use H for the polar BP decoder since the density of H is relatively high. In general, BP decoder is not suitable for linear block codes with high-density parity-check (HDPC) matrix. However, in [13], Jiang and Narayanan proposed an iterative decoding scheme that performs well for RS code with HDPC matrix by adapting its parity-check matrix when BP decoding fails to converge. In this paper, we investigate whether the idea of parity-check matrix adaptation also performs well for polar codes or not.

Suppose a codeword x is transmitted over a BDMC and y is received. The LLR of the i th coded bit is defined as $\gamma(x_i) = \log \left(\frac{\Pr(y|x_i=0)}{\Pr(y|x_i=1)} \right)$. Let $\gamma = (\gamma(x_1), \gamma(x_2), \dots, \gamma(x_{N-K}), \dots, \gamma(x_N))$ denotes the LLRs of N coded bits. The improved polar BP decoder is described as follows.

Stage 1 matrix adaptation. Firstly, all absolute values $|\gamma(x_i)|$ ($i = 1, 2, \dots, N$) for γ are sorted by ascending order. This yields a permutation sequence $(j_1, j_2, \dots, j_{N-K}, \dots, j_N)$ of $(1, 2, \dots, N-K, \dots, N)$ with $|\gamma(x_{j_1})| < |\gamma(x_{j_2})| < \dots < |\gamma(x_{j_N})|$. Secondly, we

introduce a new vector $\mathbf{B} = (j_1, j_2, \dots, j_{N-K})$ which denotes the indices of the $N - K$ least reliable bits in γ . Let \mathbf{H}_B denotes the columns of \mathbf{H} with the indices chosen from \mathbf{B} . Finally, the Gauss elimination (GE) is applied to \mathbf{H}_B to reduce this submatrix to be an identity matrix. If \mathbf{H}_B is singular, we replace some elements of \mathbf{B} with the indices of less reliable information bits until \mathbf{H}_B becomes invertible. Then, a new matrix \mathbf{H}' of \mathbf{H} is obtained. The process can make \mathbf{H}_B as sparse as possible. We choose to use GE to \mathbf{H}_B based on the fact that the LLR value for a bit reflects its reliability. Therefore, the columns of \mathbf{H} corresponding to the most unreliable bits are reduced to a sparse matrix. This will help BP to correct these unreliable bits.

Stage 2 BP decoding. The BP algorithm is performed on the new obtained matrix \mathbf{H}' . During one round of BP iteration, the LLR value of the i th bit is updated as:

$$\gamma'(x_i) = \gamma(x_i) + \eta \gamma_{\text{ext}}(x_i), \quad (5)$$

where $0 < \eta \leq 1$ is the damping factor [13]. The extrinsic information of i th bit $\gamma_{\text{ext}}(x_i)$ is calculated as follows [13]:

$$\gamma_{\text{ext}}(x_i) = \sum_{j=1, H'_{ji}=1} 2 \tanh^{-1} \left(\prod_{p=1, p \neq i, H'_{jp}=1} \tanh \left(\frac{\gamma(x_j)}{2} \right) \right). \quad (6)$$

According to $\gamma'(x_i)$, the hard decision of i th bit can be made. After a fixed number of BP iterations, say M_{BP} , the decoder goes to stage 1 and continues to the next round of outer iteration. It should be noted that the matrix adaptation should use the latest updated LLRs γ' obtained by the inner BP decoder. The advantages of matrix adaptation are as follows: (1) It reduces the density of the original parity-check matrix \mathbf{H} and eliminates part of short cycles. (2) It takes LLR values of reliable bits into account when updating unreliable bits. Therefore, the decoding performance can be improved [13].

3.2 Early termination

It is well-known that the stopping criterion for the BP decoder is to determine if the constraint $\hat{\mathbf{x}}(\mathbf{H}')^T = 0$ is satisfied. However, according to the analysis in [15] and simulation results, the BP decoder does not provide a good error-correction performance if we only use $\hat{\mathbf{x}}(\mathbf{H}')^T = 0$ as the stopping criterion for polar codes. Hence, we consider the following simple but efficient stopping criterion for the proposed decoder based on the criterion described in [15], which also can reduce the number of matrix adaption.

minLLR-based stopping criterion: Let $\text{Min}|\text{LLR}|$ denotes the minimum absolute value of LLRs for γ' and β be a

positive real number. If $\text{Min}|\text{LLR}| > \beta$, then the decoder calculates $\hat{\mathbf{u}}$ according to $\hat{\mathbf{x}}$ and stops decoding. Otherwise, the decoder continues to perform the next round of matrix adaptation.

The above criterion is based on the stopping criterion described in [15], though it is proposed for BP decoder with factor graph representation of generator matrix. To illustrate the validity of this criterion, the proposed decoder only with $\hat{\mathbf{x}}(\mathbf{H}')^T = 0$ as the stopping criterion is used to decode the (256, 128) polar code at SNR = 2.5 dB. In order to avoid getting trapped for the decoder when BP iterations could not converge to a valid estimated sequence (i.e., $\hat{\mathbf{x}}(\mathbf{H}')^T \neq 0$), the total number of BP iterations should be limited to a predefined value M_{BP} . We generate 153 frames, and 50 frames among them cannot be decoded by the proposed decoder. It means that for 103 frames, the condition $\hat{\mathbf{x}}(\mathbf{H}')^T = 0$ is met before the total number of iterations is achieved and that for 50 frames, the condition $\hat{\mathbf{x}}(\mathbf{H}')^T = 0$ is still not met when the total number of iterations is achieved. Figure 1 shows the values of $\text{Min}|\text{LLR}|$ s for γ' which are gathered during the decoding of these 153 frames. In order to examine the results, $\text{Min}|\text{LLR}|$ s for the 50 error frames are plotted on the right side of Fig. 1. We can see that the $\text{Min}|\text{LLR}|$ of a correct frame is much larger than zero in general, while that of incorrect frames approach zero. The parameter β can be chosen according to simulations which are shown in the next section.

3.3 Algorithm description

Based on the above analysis, the proposed scheme is described in Algorithm 1, where M_{AP} denotes the predefined number of matrix adaptation. The function

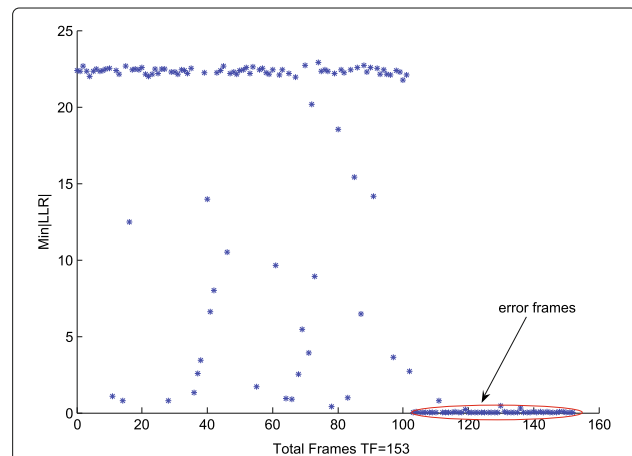


Fig. 1 Distribution of $\text{Min}|\text{LLR}|$ s for a (256, 128) polar code at SNR = 2.5 dB

AdaptiveMatrix() performs matrix adaptation which reduces \mathbf{H} to \mathbf{H}' . The function *BP()* performs BP algorithm based on \mathbf{H}' and stores its returned value into the boolean variable $\text{BP}_{\text{result}}$. If $\text{BP}_{\text{result}} = \text{false}$, it means the constraint $\hat{\mathbf{x}}(\mathbf{H}')^T = 0$ is still not satisfied when the number of iterations that BP used is increased to M_{BP} . Thus, the algorithm goes back to perform the matrix adaptation again. Otherwise, it means the constraint $\hat{\mathbf{x}}(\mathbf{H}')^T = 0$ is satisfied. Then, we compare $\text{Min}|\text{LLR}|$ with β , and the matrix adaptation will be performed again when $\text{Min}|\text{LLR}| < \beta$. It should be noted that we can use the cyclic redundancy check (CRC) to detect the output sequence of the BP algorithm when $\text{Min}|\text{LLR}| \geq \beta$ to further improve the performance of the proposed algorithm. Accordingly, it needs to set the last r unfrozen bits to hold the r -bit CRC when encoding, where r is a small constant. In addition, considering that there are lots of variables in Section 3, we first provide a variable list as shown in Table 1 to make the used variables more clearly. Specifically, the last four variables in Table 1 will be used in the following subsection.

3.4 Computational complexity

The computational complexity of Algorithm 1 is determined by three parts: (1) sorting in *AdaptiveMartix()*, (2) GE in *AdaptiveMartix()*, and (3) BP decoding. Since different parts are involved in different operations, we analyze the complexity by counting the number of binary operations (BOs), floating point comparisons (FPCs), and

Table 1 Symbol definition

Symbol	Meaning
$\hat{\mathbf{u}}$	The decision source block
$\hat{\mathbf{x}}$	The decision codeword
M_{BP}	The predefined number of BP iterations
M_{AP}	The predefined number of matrix adaptation
\mathbf{H}	The parity check matrix
\mathbf{H}'	The adjusted parity-check matrix
\mathbf{B}	The index set of the $N - K$ least reliable bits in $\boldsymbol{\gamma}$
$\boldsymbol{\gamma}$	The initial LLRs set of N coded bits
$\boldsymbol{\gamma}'$	The new LLRs set of N coded bits
η	The damping factor
$\text{Min} \text{LLR} $	The minimum absolute value of LLRs for $\boldsymbol{\gamma}'$
β	a positive real number
$\text{BP}_{\text{result}}$	The returned value of function <i>BP()</i>
C_{AP}	The average number of matrix adaptation
C_{BP}	The average number of iterations that <i>BP()</i> used
W_R	The average row weight
W_C	The average column weight

Algorithm 1 Enhanced Belief Propagation Decoding Algorithm of Polar Codes

Input: the original LLR vector $\boldsymbol{\gamma}$, the parity check matrix \mathbf{H} ;
Output: the encoding estimated sequence $\hat{\mathbf{u}}$;

```

1:  $i = 1$ ;
2: while  $i \leq M_{\text{AP}}$  do
3:   AdaptiveMatrix();
4:    $\text{BP}_{\text{result}} = \text{BP}()$ ;
5:   if  $\text{BP}_{\text{result}} = \text{false}$  then
6:     increment  $i$  by one;
7:     perform AdaptiveMatrix() on  $\mathbf{H}$ ;
8:   else
9:      $\text{Min}|\text{LLR}| = \min_{k=1}^N |\boldsymbol{\gamma}'(\mathbf{x}_k)|$ ;
10:    if  $\text{Min}|\text{LLR}| < \beta$  then
11:      increment  $i$  by one;
12:      perform AdaptiveMatrix() on  $\mathbf{H}$ ;
13:    else
14:      if  $\text{CRC}(\hat{\mathbf{x}}) = \text{Success}$  then
15:        compute  $\hat{\mathbf{u}}$  according to  $\hat{\mathbf{x}}$  and break;
16:      else
17:        increment  $i$  by one;
18:        perform AdaptiveMatrix() on  $\mathbf{H}$ ;
19:      end if
20:    end if
21:  end if
22: end while
23: if  $i > M_{\text{AP}}$  then
24:   declare a decoding failure;
25: end if

```

Algorithm 2 The function *AdaptiveMatrix()*

Input: the latest updated LLRs $\boldsymbol{\gamma}$, the original check matrix \mathbf{H} ;
Output: the new adapted matrix \mathbf{H}' ;

```

1: perform the sorting operation on the latest updated LLRs  $\boldsymbol{\gamma}'$ ;
2: perform matrix adaptation on  $\mathbf{H}$ ;

```

floating point operations (FPOs), respectively. For part (1), the sorting step needs $O(C_{\text{AP}}N \log N)$ FPCs, where C_{AP} denotes the average number of matrix adaptation. The GE in part (2) needs $O(C_{\text{AP}}N^3)$ BOs. Note that though the complexity of GE is cubic, it only needs BOs which are much simpler than FPOs. Moreover, simulations show that *AdaptiveMatrix()* is performed only a few times as SNR increases. When analyzing the complexity of BP decoding in part (3), we notice that an LLR for i th bit with odd index in SC decoding can be computed in the form $2 \tanh^{-1}(\tanh(\cdot) \times \tanh(\cdot))$, which is the same as that of the BP decoding [14]. For simplicity, we take the $\tanh()$ function as a black box and do

Algorithm 3 The function $BP()$

Input: the latest adapted matrix H' ;
Output: a boolean variable BP_{result} ;

```

1:  $j = 1$ ;
2:  $BP_{result} = false$ ;
3: while  $j \leq M_{BP}$  do
4:   update the message of the check nodes ;
5:   update the message of the variable nodes;
6:   if  $\hat{x}(H')^T = 0$  then
7:      $BP_{result} = true$ ;
8:     break ;
9:   else
10:    increment  $j$  by one ;
11:   end if
12: end while
13: if  $j > M_{BP}$  then
14:    $BP_{result} = false$ ;
15: end if

```

not count the number of FPOs for $\tanh()$ function when analyzing the complexity of different decoders. This is reasonable since all the involved decoders need $\tanh()$ function. According to Eq. (2), the N - K check nodes need $O(C_{AP}C_{BP}(N-K)(W_R-2))$ multiplication operations, where W_R is the average row weight and C_{BP} is the average number of iterations that $BP()$ used. The N variable nodes need $O(C_{AP}C_{BP}N(W_C-2))$ additive operations, where W_C is the average column weight. Table 2 summarizes the complexity analysis of Algorithm 1 and SC decoder. It should be noted that the complexity of the proposed algorithm decreases rapidly as SNR increases since C_{AP} and C_{BP} are very small. Actually, from Table 3, we can see that W_C and W_R after *AdaptiveMatrix()* are much smaller than K . Hence, the average complexity of BP decoding is relatively low.

3.5 Variations to the Proposed Algorithms

In this subsection, we give several variations of the proposed algorithm to further improve the performance or reduce the decoding complexity.

1. Improving the performance
The first variation inspired by [13] can further improve the performance by running the proposed algorithm several times each time with the same

initial LLRs from the channel but a different grouping of the less reliable bits. This is based on the fact that some bits with their |LLR|s close to those in the unreliable set B are also of the wrong sign and vice versa. Each time the proposed algorithm is run, a different estimate of codeword may be obtained due to the different parity-check matrix. The decoder keeps all the returned codewords in a list and chooses the one that minimizes Euclidean distance from the received vector of the channel. This variation can significantly improve the asymptotic performance of polar codes.

2. Reducing the complexity

- (a) Partial reliable bit updating
The complexity of the proposed algorithm mostly depends on the complexity of the FPCs. The main FPCs complexity comes from two ways: the computation of the extrinsic information in the reliable part and the adaptation of the parity-check matrix. Since only some bits in the boundary will be switched from the reliable part to the unreliable part in the adaptation of the parity-check matrix, we can use the rule of partial reliable bit updating proposed in [13] to reduce the complexity in the bit reliabilities update part.
- (b) Sophisticated update schemes
Using sophisticated update schemes reduces the complexity of matrix adaptation, such as the scheme that adapting the parity-check matrix from the previous ones proposed by El-Khamy and McEliece [16], which reduces the overall complexity by 75%.

4 Simulation results

In this section, we consider three polar codes: a (256, 128) rate-1/2 polar code C_1 , a (1024, 512) rate-1/2 polar code C_2 , and a (2048, 1536) rate-3/4 polar code C_3 . We set $M_{AP} = 10$ and $M_{BP} = 20/50$. According to the selection principle of η in [13], we set first the observed performances of the proposed decoder when $\eta = 0.5$. And a CRC-8 code for C_1 with generator polynomial $g(D) = D^8 + D^7 + D^6 + D^4 + D^2 + 1$ is used. While for C_2 and C_3 , we use a CRC-24 code with generator polynomial $g(D) = D^{24} + D^{23} + D^6 + D^5 + D + 1$. Since the length of CRC sequences is

Table 2 Complexity comparisons with different decoders

	BOs	FPCs	FPOs	
			\times/\div	$+/-$
SC	/	/	$O(N \log N)$	$O(N \log N)$
proposed	$O(C_{AP}N^3)$	$O(C_{AP}N \log N)$	$O(C_{AP}C_{BP}(N-K)(W_R-2))$	$O(C_{AP}C_{BP}N(W_C-2))$

Table 3 The average row and column weight of \mathbf{H}' for a (1024, 512) polar code

SNR	Avg. weight of column(W_C)	Avg. weight of row(W_R)
2.0 dB	26.08	52.67
2.5 dB	26.12	52.72
3.0 dB	26.10	52.73
3.5 dB	26.14	52.77
4.0 dB	26.16	52.79

very short when compared to that of information bits in a polar code, the rate loss caused by using CRC is almost negligible. All simulations are performed over an additive white Gaussian noise (AWGN) channel with binary phase shift keying (BPSK).

4.1 Choice of β

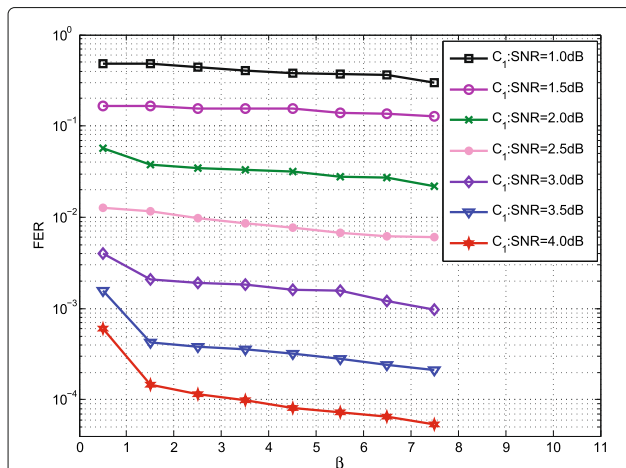
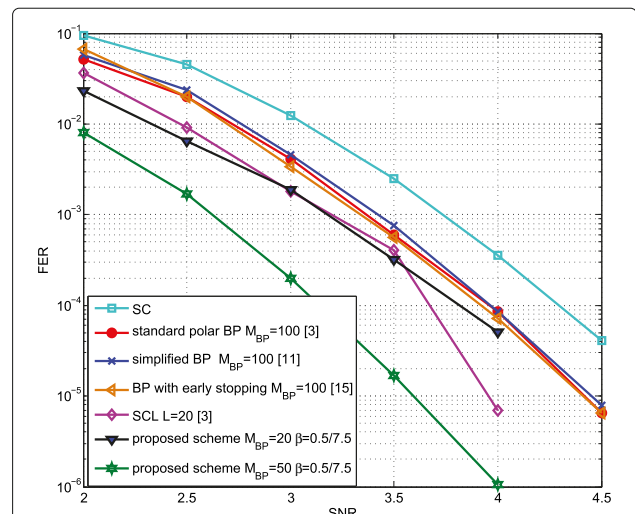
Figure 2 presents the FER performance as a function of β at different signal-to-noise ratios (SNRs) for C_1 . It can be seen that for $\text{SNR} < 3.5$ dB, the FER performances are almost the same as β increases. For $\text{SNR} \geq 3.5$ dB, the FER decreases gradually as β increases. Although the value of β has little influence on the FER performance of the proposed scheme for small SNRs, it is shown from the simulations that for the larger value of β , the decoder needs to perform a larger number of BP iterations. For example, if the value of β is 0.5, the average number of BP iterations for C_1 is about 49 at $\text{SNR} = 2.0$ dB. However, if the value of β is 7.5, the average number of BP iterations for C_1 is about 63 at the same SNR. Therefore, as in [15], to make a trade-off between the FER performance and decoding complexity, we choose $\beta = 0.5$ when $\text{SNR} < 3.5$ dB and $\beta = 7.5$ when $\text{SNR} \geq 3.5$ dB for C_1 . For C_2 and C_3 ,

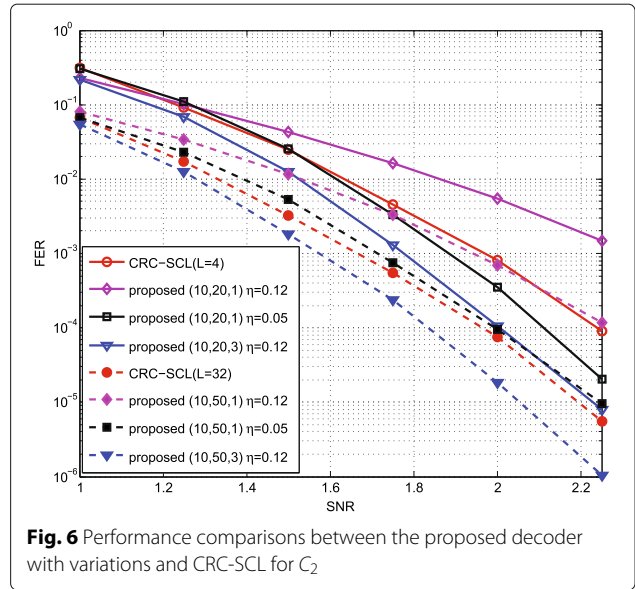
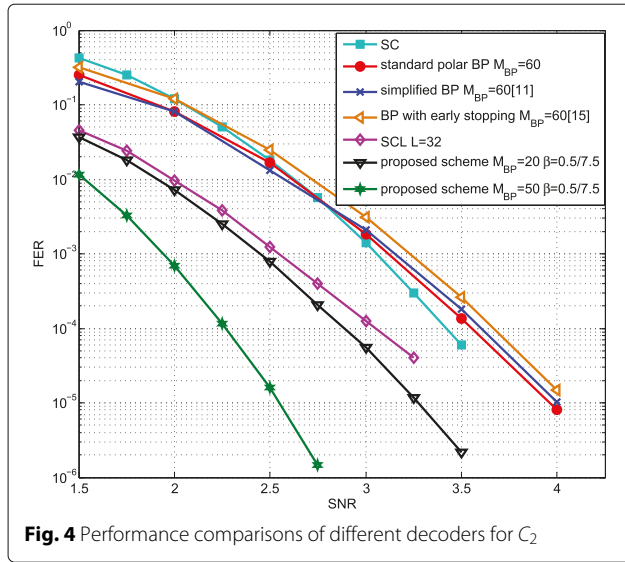
we choose $\beta = 0.5$ when $\text{SNR} < 3.5$ dB and $\beta = 15.5$ when $\text{SNR} \geq 3.5$ dB.

4.2 Comparison of the FER performance

Figures 3 and 4 show the FER performances for C_1 and C_2 with different polar decoders, respectively. We can see that the proposed decoder with $M_{BP} = 50$ performs significantly better than the SC decoder and the original BP decoder. Particularly, for C_2 , at $\text{FER} = 10^{-4}$, the performance gains for the proposed decoder with $M_{BP} = 50$ is about 1.2 dB when compared to the original BP decoder with $M_{BP} = 60$ and the improved BP decoders proposed in [11] and [15]. And also, compared to the SC list (SCL) decoder with the list size $L = 32$, the polar codes using the proposed decoder can obtain almost 0.7 dB performance gains. Figure 5 presents the FER performances for C_3 using different polar decoders. Although the code rate becomes large, the proposed decoder still can obtain about 0.5 dB performance gains at $\text{FER} = 10^{-4}$ compared to the original polar BP decoder. It shows that the proposed decoder can improve the FER performance of the original polar BP decoder significantly on the AWGN channel with short and moderate blocklength at different code rates. Another side, we notice that the algorithm proposed in [10] which the readers are referred to for more details also can improve the performance of the original polar BP decoder with an improvement in SNR of 0.3 dB. However, both Figs. 4 and 5 show the performance gains using the proposed decoder can be increased to 0.5 dB at least, such as 0.5 dB at a rate = 0.75 and 1.2 dB at a rate = 0.5.

We make a performance comparison between the proposed decoder with variations and CRC-aided SCL (CRC-SCL) decoder as shown in Fig. 6. Specifically, the notion proposed(M_{AP}, M_{BP}, q_1) refers to the proposed scheme

**Fig. 2** Performances of the proposed decoder for C_1 with different β **Fig. 3** Performance comparisons of different decoders for C_1



with variations which are presented in Section 3.5. The parameter q_1 refers to the number of decoding rounds with different groupings of the unreliable bits. The damping factor η is also specified on the plots. From Fig. 6, we can see that the performance of the proposed decoder using the different grouping method can be improved significantly. Compared with the CRC-SCL ($L = 32$) [17, 18], the proposed decoder with the parameters (10, 50, 3) provides about 0.2 dB performance gains at an FER = 10^{-4} . Additionally, the damping coefficients of the proposed algorithm must be carefully chosen to control the updating step width. In general, if the η is set to be larger, the performance of the proposed decoder has a flat slope (as shown in Fig. 6) which is mainly due to the overshooting

of the update scheduling such that the decoder tends to a wrong codeword, quickly.

4.3 Average iterative number and decoding complexity

Figure 7 shows the average number of total BP iterations ($= C_{AP}C_{BP}$) per frame for C_2 using the original BP decoder, BP decoder with G-matrix-based early stopping scheme [15], BP decoder with the freezing of connected sub-factor-graphs (CSFGs) [19], and the proposed decoder. From simulations, we can see that as SNR increases, the average number of BP iterations for the proposed decoder decreases faster than other decoders. It shows that the proposed algorithm can tend to a correct codeword with a faster convergence rate. To show the FPO

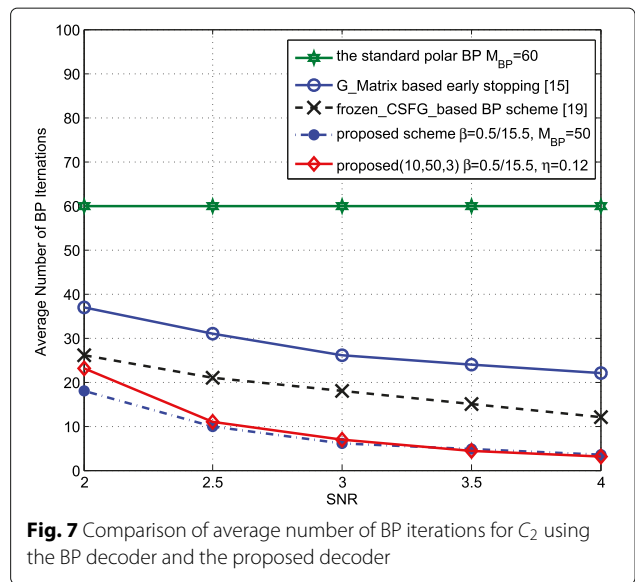
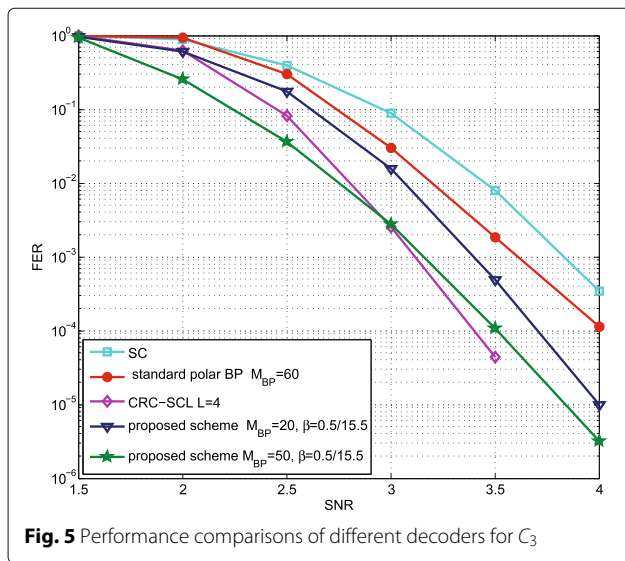


Table 4 The numbers of FPOs of different decoders ($\times 10^3$) for C_2

	SNR = 2.5dB		SNR = 3.0dB		SNR = 3.5dB		SNR = 4.0dB	
	M/D	A/S	M/D	A/S	M/D	A/S	M/D	A/S
SC	5	5	5	5	5	5	5	5
SCL(L = 32)	328	328	328	328	328	328	328	328
Proposed($M_{BP} = 50$)	286	272	182	173	130	124	78	74
Improved SCL(L = 32)[18]	45	45	23	23	10	10	6	6
Proposed(10, 50, 3)	214	204	136	130	98	93	58	55

complexity of the proposed decoder clearly, we use the complexity of SCL decoder as reference. Table 4 shows the numbers of FPOs using the original SC decoder, SCL decoder ($L = 32$) and the proposed decoder ($M_{BP} = 50$) for a (1024, 512) polar code. Since the average number of total BP iterations $C_{AP}C_{BP}$ becomes smaller as SNR increases, such as $C_{AP}C_{BP} = 5.01$ at SNR = 3.5 dB, the average numbers of FPOs of the proposed decoder decrease with SNR increasing. When SNR ≥ 3.5 dB, the proposed decoder has the lower complexity of FPOs than SCL decoder. It should be noted that all the numbers in Table 4 are rounding numbers. From the table, it shows that the FPO complexity of the proposed decoder decreases rapidly as SNR increases. And also, we can see that the complexity of the proposed decoder when using the variations mentioned in Section 3.5 is reduced quickly. However, compared with the reduced-complexity SCL [18], the proposed decoder with the variations still has a higher complexity. Considering the better performance and parallel architecture of the proposed decoder, if a larger complexity is tolerated, it will be a promising decoder to reduce the decoding latency of CRC-SCL decoder for polar codes and provide a good performance.

5 Conclusions

In this paper, we have proposed an improved BP decoder for polar codes by adapting their parity-check matrices. Though the idea of the proposed decoder is not new, it has never been used for polar codes before. More importantly, simulation results show that the proposed decoder can provide significant performance gains compared to the polar BP decoder and also can compete with CRC-SCL decoder when using the variations of the proposed algorithm described in Section 3.5. Although the proposed decoder still has a little larger complexity than that of those reduced-complexity CRC-SCL, the proposed decoder is a promising decoder as a fully parallel architecture to reduce the decoding latency of CRC-SCL decoder for polar codes with a tolerated complexity.

Acknowledgements

The authors would like to thank the editor and anonymous reviewers for their constructive comments which helped improve the quality of this paper. This

work is supported by the National Nature Science Foundation of China (no. 61471286, no. 61271004), the Fundamental Research Funds for the Central Universities, and the open research fund of Key Laboratory of Information Coding and Transmission, Southwest Jiaotong University (no. 2010-03).

Competing interests

The authors declare that they have no competing interests.

Author details

¹The School of Computer Science and Technology, Xidian University, TaiBei Road, Xi'an, China. ²The Key Laboratory of Information Coding and Transmission, Southwest Jiaotong University, ErHuan Road, Chengdu, China. ³The School of Computer Science and Technology, Xi'an Technological University, JinHua Road, Xi'an, China. ⁴Border Defence Academy, HuanShan Road, Xi'an, China.

Received: 5 April 2016 Accepted: 15 February 2017

Published online: 28 February 2017

References

1. E Arıkan, Channel polarization: a method for constructing capacity achieving codes for symmetric binary-input memoryless channels. *IEEE Trans. Inform. Theory*. **55**(7), 3051–3073 (2009)
2. I Tal, A Vardy, List decoding of polar codes. *IEEE Trans Inf Theory*. **61**(5), 2213–2226 (2015)
3. K Niu, K Chen, Stack decoding of polar codes. *Electron. Lett.* **48**(12), 695–697 (2012)
4. E Arıkan, A performance comparison of polar codes and Reed-Muller codes. *IEEE Commun. Lett.* **12**(6), 447–449 (2008)
5. A Eslami, H Pishro-Nik, On finite-length performance of polar codes: stopping sets, error floor, concatenated design. *IEEE Trans. Commun.* **61**(3), 919–929 (2013)
6. N Hussami, R Urbanke, SB Korada, in *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*. Performance of polar codes for channel and source coding (IEEE, Seoul, 2009), pp. 1488–1492
7. A Eslami, H Pishro-Nik, in *Proceedings of the 48th Annual Allerton Conference on Communication, Control, and Computing*. On bit error rate performance of polar codes in finite regime (AACC, Allerton, 2010), pp. 188–194
8. Y Zhang, A Liu, X Pan, Z Ye, C Gong, A modified belief propagation polar decoder. *IEEE Commun. Lett.* **18**(7), 1091–1094 (2014)
9. B Yuan, KK Parhi, in *Proceedings of the IEEE International Conference Acoustics, Speech and Signal Processing*. Architecture optimizations for BP polar decoders (ICASSP, Vancouver, British Columbia, 2013), pp. 2654–2658
10. J Guo, M Qin, A Guillén i Fàbregas, PH Siegel, in *Proceedings of the IEEE International Symposium on Information Theory*. Enhanced belief propagation decoding of polar codes through concatenation (ISIT, Honolulu, 2014), pp. 2987–2991
11. Y Zhang, Q Zhang, X Pan, Z Ye, C Gong, in *Proceedings of the IEEE International Wireless Symposium*. A simplified belief propagation decoder for polar codes (IWS, Xi'an, 2014), pp. 1–4
12. N Goela, SB Korada, M Gastpar, in *Proceedings of the IEEE Information Theory Workshop*. On LP decoding of polar codes (ITW, Dublin, 2010), pp. 1–5
13. J Jiang, K Narayanan, Iterative soft-input-soft-output decoding of Reed-Solomon codes by adapting the parity check matrix. *IEEE Trans. Inf. Theory*. **52**(8), 3746–3756 (2006)

14. R Mori, T Tanaka, Performance of polar codes with the construction using density evolution. *IEEE Commun. Lett.* **13**(7), 519–521 (2009)
15. B Yuan, KK Parhi, Early stopping criteria for energy-efficient low-latency belief-propagation code decoders. *IEEE Trans. Signal Proc.* **62**(24), 6496–6506 (2014)
16. M El-Khamy, RJ McEliece, J Harel, in *Proceedings of the IEEE International Symposium on Information Theory*. Performance enhancements for algebraic soft-decision decoding of Reed-Solomon codes (ISIT, Chicago, 2004), p. 419
17. K Chen, K Niu, J Lin, Improved successive cancellation decoding of polar codes. *IEEE Trans. Commun.* **61**(8), 3100–3107 (2013)
18. K Chen, K Niu, J Lin, in *Proceedings of the IEEE Vehicular Technology Conference*. A reduced-complexity successive cancellation list decoding of polar codes (VTC Spring, Dresden, 2013), pp. 1–5
19. S Mohsin Abbas, Y Fan, J Chen, C Tsui, Low complexity belief propagation polar code decoders [Online] (2015). Available, <http://arxiv.org/abs/1505.04979>

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com