

Hybrid Decoding for Polar Codes

Soyeon Choi

Dept. of Electronics Engineering
Chungnam National University
Daejeon, South Korea
sychoi.cas@gmail.com

Hoyoung Yoo

Dept. of Electronics Engineering
Chungnam National University
Daejeon, South Korea
hyyoo.cnu@gmail.com

Abstract—Among various polar decoding algorithms, SC-List [2] and SC-Flip [3] suffer from high hardware complexity and long decoding latency, respectively. In this paper, a novel hybrid decoding algorithm is proposed to achieve affordable hardware complexity with a suitable decoding latency. According to the experimental results, the proposed method affords a comparable error-correcting performance to that of SC-List [2] and SC-Flip [3] counter parts.

Keywords—polar codes; SC-List decoding; SC-List decoding; hybrid decoding;

I. INTRODUCTION

The polar code [1] is the first error-correcting code that can achieve channel-capacity provably. Although Successive-Cancellation (SC) decoding [1] has been traditionally used to implement a polar decoder, the error-correcting performance is not suitable for next-generation communication and storage systems. To improve the error-correcting performance, SC-List decoding [2] and SC-Flip decoding [3]-[4] with flip are recently proposed by taking more chances to find a valid codeword. SC-List decoding [2] provides a good error-correcting performance, but it suffers from high hardware complexity. On the other hands, SC-Flip [3] requires a small hardware circuitry, but it deteriorates a decoding latency. In this paper, we propose a hybrid decoding algorithm that combines SC-List [2] and SC-Flip decoding [3] resulting in affordable hardware complexity with a suitable decoding latency. The proposed method suggests a good candidate for a polar decoder with a stringent hardware requirement.

II. REVIEW OF POLAR DECODING

A. Polar codes

Let us consider the polar (N, K) code, where N and K denote a code length and a message length, respectively. Among the N bit-channel, K most reliable bit-channels are used for information bits and $(N-K)$ remaining bit-channels are used for frozen bits. The index sets of information and frozen bits are denoted by A and A^c , respectively. The codeword x is generated by matrix multiplication of the generator matrix G and message vector u .

Given received vector y , Successive-Cancellation (SC) [1] decoding has been traditionally used to bring a polar decoder in practice. Due to its serial nature, SC decoding alternatively computes soft information corresponding to (1) and (2) in log-

likelihood ratio (LLR) domain. Although SC decoding [1] is the first decoding algorithm, its error-correcting performance is not comparable to other capacity-achieving codes such as LDPC and turbo codes since it makes a hard-decision when the message bit is estimated as (3).

$$f(LLR_i, LLR_{i+1}) \approx \text{sign}(LLR_i) \text{sign}(LLR_{i+1}) \min(|LLR_i|, |LLR_{i+1}|), \quad (1)$$

$$g(LLR_i, LLR_{i+1}, \hat{u}) = (-1)^{\hat{u}} LLR_i + LLR_{i+1}, \quad (2)$$

$$\hat{u}_i = \begin{cases} 0, & LLR_i \geq 0, i \in A \\ 1, & LLR_i < 0, i \in A \\ u, & i \in A^c \end{cases} \quad (3)$$

B. SC-List and SC-Flip decoding

To improve error-correcting performance, SC-List [2] decoding algorithm with list size of L inspects L most likely codewords simultaneously, whereas SC decoding [1] inspects only one most likely codeword. Since SC-List [2] searches more possible codewords, it always provides a stronger error-correcting capability compared to SC decoding [1]. As the list size of L increases, the error correcting capability is more improved. However, severe hardware complexity is inevitable in SC-List [2] decoding since the hardware to search valid codewords at the same time is proportionally increased as the larger L is adopted.

Similar to SC-List [2] decoding, SC-Flip [3] decoding with flipping bits of F provides a more chance to inspects possible codewords. Whereas SC-List [2] decoding searches L codewords simultaneously, SC-Flip [3] decoding searches 2^F codewords in a sequence. More precisely, the standard SC [1] decoding is firstly performed to estimate a message vector \hat{u}_1^N . If the CRC is success, the decoding is terminated. Otherwise, the SC decoding is performed for T additional attempts to identify which bit is an error by flipping the most unreliable bit. Although SC-Flip [3] decoding succeeded in reducing hardware complexity, it necessitates a long decoding latency. In general, the error-correcting performance of SC-Flip [3] with flipping bit of F is similar to that of SC-List [2] with list size of $L = 2^F$.

III. PROPOSED METHOD

From an implemental points of view, SC-List [2] and SC-Flip [3] decoding provides extreme candidates in terms of hardware complexity and decoding latency. In other words,

The proposed hybrid algorithm

```

Initialize:  $\hat{\mathbf{u}}_0^{N-l} \leftarrow \text{SC}(y_0^{N-l}, \emptyset)$ .
if CRC( $\hat{\mathbf{u}}_0^{N-l}$ ) = fail then
     $U \leftarrow i \in A$  of smallest  $|LLR_i|$ 
    for  $t = 0$  step 1 until  $T$  begin
         $k \leftarrow U(t)$ .
         $\hat{\mathbf{u}}_0^{N-l} \leftarrow \text{SCL}(y_0^{N-l}, k)$ .
        if CRC( $\hat{\mathbf{u}}_0^{N-l}$ ) = success then
            break.
        end if
    end for
end if
Output:  $\hat{\mathbf{u}}_0^{N-l}$ .

```

Figure 1. The proposed hybrid decoding algorithm

SC-List [2] is the best candidate in terms of decoding latency but the worst candidate in terms of hardware complexity. Similarly, SC-Flip [3] is the best candidate in terms of hardware complexity but the worst candidate in terms of decoding latency. In this paper, we propose a hybrid decoding algorithm for affordable hardware complexity with a suitable decoding latency by combining SC-List [2] and SC-Flip [3] decoding algorithms. Figure 1 describes the proposed hybrid decoding algorithm with list size l , flipping bit f , and additional attempt t . At first, standard SC [1] decoding is performed to find the set of flipping index U , which is a set of the least reliable bits. When CRC is successful, the proposed algorithm is terminated as SC-Flip [3]. Otherwise, additional t attempts are tried to identify which bit is an error by flipping the least reliable bit in U . Unlike SC-Flip [3] decoding which employs SC [1] decoding for additional T attempts, the proposed hybrid decoding algorithm employs SC-List [2] decoding to accelerate decoding latency. Note that SC and SCL(y_0^{N-l}, k) in Fig. 1 denote the SC and SCL decoding process with a flipped bit at the index of k . Since the proposed hybrid decoding provides an intermediate decoding algorithm between extreme SC-List [2] and SC-Flip [3] decoding algorithms, it can be a good design candidate for a polar decoder with a stringent hardware requirement in both area and time.

IV. EXPERIMENTAL RESULTS

We compare SC-List [2], SC-Flip [3], and the proposed decoding algorithms for Polar (1024, 512) codes with 16-bit CRC codes. AWGN channel is used for a transmission channel, and a codeword is modulated by BPSK. Figure 2 shows the frame error rate (FER) of various decoding algorithms for different channel environment. FER of the proposed decoding algorithm with $l = 2$ and $f = 1$ is similar to that of SC-List decoding with $L = 4$ and that of SC-Flip decoding with $F = 2$. In addition, FER of the proposed decoding algorithm with $l = 4$ and $f = 1$ is similar to that of SC-List decoding with $L = 8$ and that of SC-Flip decoding with $F = 3$. As a result, the proposed method affords a comparable error-correcting performance to that of SC-List and SC-Flip counter parts. Table I summarize a complexity comparison in terms of area and time. Generally

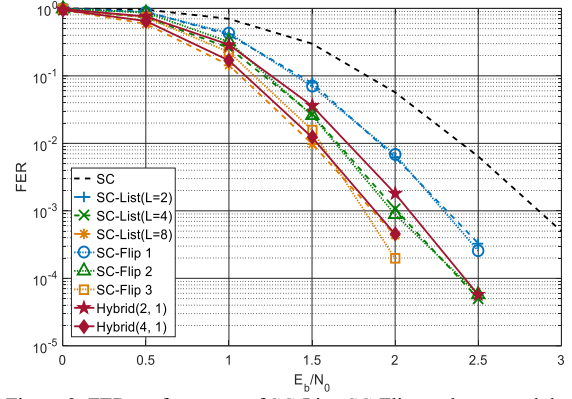


Figure 2. FER performance of SC-List, SC-Flip, and proposed decoding

TABLE I. Comparison of hardware complexity

	Area	Time	Area \times Time
SC-List Decoding	L	1	L
SC-Flip Decoding	1	$F \cdot T$	$F \cdot T$
Proposed Decoding	l	$f \cdot t$	$l \cdot f \cdot t$

speaking, $l \times f$ in the proposed decoding algorithm is the same as L in SC-List [2] and F is SC-Flip [3]. Note that l and f equipped in the proposed decoding algorithm are always smaller than L in SC-List [2] and F and T in SC-Flip [3].

V. CONCLUSION

In this paper, proposed decoding algorithm for polar codes has been newly proposed by combining SC-List [2] and SC-Flip [3] decoding algorithms. The proposed decoding algorithm requires a lower hardware complexity compared to SC-List [2] decoding and provides a shorter decoding latency compared to SC-Flip [3] decoding. Thus, the proposed algorithm can be used for a polar decoder with a stringent requirement.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation (NRF) grant funded by the Korea government (MSIP) (2017R1C1B5015962) and by the IC Design Education Center (IDEC).

REFERENCES

- [1] E. Arıkan, "Channel polarization: A method for constructing capacity achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, pp. 3051–3073, 2009.
- [2] I. Tal and A. Vardy, "List decoding of polar codes," in *Proc. IEEE Symp. Inform. Theory*, Saint Petersburg, Russia, 2011, pp. 1–5.
- [3] O. Afisiadis, A. Balatsoukas-Stimming, and A. Burg, "A low-complexity improved successive cancellation decoder for polar codes," in *Proc. 48th Asilomar Conf. Signals Syst. Comput.*, Nov. 2014, pp. 2116–2120.
- [4] L. Chandesaris, V. Savin, and D. Declercq, "An improved SCFlip decoder for polar codes," in *IEEE Global Communications Conference (GLOBECOM)*, Dec 2016, pp. 1–6.