Successive Cancellation Decoding of Single Parity-Check Product Codes

Coskun, Mustafa Cemil; Liva, Gianluigi; Graell i Amat, Alexandre; Lentmaier, Michael

Published in: IEEE International Symposium on Information Theory (ISIT), 2017

DOI: 10.1109/ISIT.2017.8006832

2017

Document Version: Publisher's PDF, also known as Version of record


General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Successive Cancellation Decoding of Single Parity-Check Product Codes

Mustafa Cemil Coşkun*, Gianluigi Liva*, Alexandre Graell i Amat† and Michael Lentmaier‡

* Institute of Communications and Navigation, German Aerospace Center, Weßling, Germany
†Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden
‡Department of Electrical and Information Technology, Lund University, Lund, Sweden

Abstract—We introduce successive cancellation (SC) decoding of product codes (PCs) with single parity-check (SPC) component codes. Recursive formulas are derived, which resemble the SC decoding algorithm of polar codes. We analyze the error probability of SPC-PCs over the binary erasure channel under SC decoding. A bridge with the analysis of PCs introduced by Elias in 1954 is also established. Furthermore, bounds on the block error probability under SC decoding are provided, and compared to the bounds under the original decoding algorithm proposed by Elias. It is shown that SC decoding of SPC-PCs achieves a lower block error probability than Elias’ decoding.

I. INTRODUCTION

In 1954, P. Elias showed that the bit error probability over the binary symmetric channel (BSC) can be made arbitrarily small with a strictly positive coding rate by iterating an infinite number of simple linear block codes, introducing the class of product codes (PCs) [1]. More recently, PCs, re-interpreted as turbo-like codes [2], and their generalizations (see, e.g., [3]–[6]) have attracted a large interest from both a research viewpoint [7]–[10] and an application [11] viewpoint.

In [7], PCs with single parity-check (SPC) component codes were decoded using iterative decoding algorithms based on Bahl, Cocke, Jelinek, and Raviv [12] a-posteriori probability decoding of the component codes. In [8], the asymptotic performance of SPC-PCs, whose component code length doubles with each dimension, was analyzed over the BSC, providing an improved bound on the bit error probability by using 2-dimensional SPC-PCs as the component codes of the overall PC. In [13], a bridge between generalized concatenated codes and polar codes is established.

In this paper, we establish a bridge between the original decoding algorithm of PCs, which we refer to as Elias’ decoder1 [1], and the successive cancellation (SC) decoding algorithm of polar codes [14], [15]. The link is established for SPC-PCs and for the binary erasure channel (BEC). We show that the block error probability of SPC-PCs can be upper bounded under both decoding algorithms using the evolution of the erasure probabilities over the decoding graph. As a byproduct of the analysis, it is shown that SPC-PCs do not achieve the capacity of the BEC under SC decoding. A comparison between Elias’ decoding and SC decoding of SPC-PCs is provided in terms of block error probability. We prove that SC decoding yields a lower error probability than Elias’ decoding. Finally, simulation results over the binary input additive white Gaussian noise (B-AWGN) channel under SC decoding are given for different SPC-PCs.

II. PRELIMINARIES

In the following, \( x_a^b \) denotes the vector \((x_a, x_{a+1}, \ldots, x_b)\) where \( b \geq a \). We use capital letters for random variables (RVs) and lower case letters for their realizations. In addition, we denote a binary-input discrete memoryless channel (B-DMC) by \( W : \mathcal{X} \to \mathcal{Y} \), with input alphabet \( \mathcal{X} = \{0, 1\} \), output alphabet \( \mathcal{Y} \), and transition probabilities \( W(y|x), \ x \in \mathcal{X}, \ y \in \mathcal{Y} \). We write BEC(\( e \)) to denote the BEC with erasure probability \( e \). The output alphabet of the BEC is \( \mathcal{Y} = \{0, 1, e\} \), where \( e \) denotes an erasure.

The generator matrix of an \((n, k)\) PC \( C \) is obtained by iterating binary linear block codes \( C_1, C_2, \ldots, C_m \) in \( m \) dimensions (levels) [1]. Let \( G_\ell \) be the generator matrix of the \( \ell \)-th component code \( C_\ell \). Then, the generator matrix of the \( m \)-dimensional PC is \( G = G_1 \otimes G_2 \otimes \ldots \otimes G_m \), where \( \otimes \) is the Kronecker product. Upon proper permutation, the generator matrix will permit to encode the message according to the labeling introduced in the next section.

Let \( C_\ell \) be the \( \ell \)-th component code with parameters \((n_\ell, k_\ell, d_\ell)\), where \( n_\ell, k_\ell, \) and \( d_\ell \) are the block length, dimension, and minimum distance, respectively. Then, the overall PC parameters are

\[
n = \prod_{\ell=1}^m n_\ell, \quad k = \prod_{\ell=1}^m k_\ell, \quad \text{and} \quad d = \prod_{\ell=1}^m d_\ell.
\]

Although the characterization of the complete distance spectrum of a PC is still an open problem even for the case where the distance spectrum of its component codes is known, the minimum distance multiplicity is known and equal to \( A_d = \prod_{\ell=1}^m A_d^{(\ell)} \), where \( A_d^{(\ell)} \) is that of the \( \ell \)-th component code. More on the distance spectrum of PCs can be found in [9], [10]. Thanks to the relationship between \( d \) and the minimum distances \( d_\ell \) of the component codes, PCs tend to have a large minimum distance. However, their minimum distance multiplicities are also typically very high [16]. Note finally that SPC-PCs are a special class of (left-regular) low-density parity-check codes [17], defined by a bipartite graph with girth 8.

1By Elias’ decoder, we refer to a decoding algorithm that treats the PC as a serially concatenated block code, where the decoding is performed starting from the component codes of the first dimension, up to those of the last dimension, in a one-sweep fashion.
III. SUCCESSIVE CANCELLATION DECODING OF SINGLE PARITY-CHECK PRODUCT CODES

Consider transmission over a B-DMC \( W \) using an \((n, k)\) SPC-PC with \( m \) dimensions (levels). Let the binary vectors \( u_1^k \) and \( x_1^k \) be the message to be encoded and the corresponding codeword, respectively, and let \( y_1^n \in \mathcal{Y}^m \) be the channel observation. The transmission by using the \((9, 4)\) SPC-PC, obtained by iterating \((3, 2)\) SPC codes, is illustrated in Fig. 1. We label the levels by numbers starting from right to left as it is seen for the 2-dimensional case in the figure. We denote by \( \eta_\ell \) the number of local SPC codes at level \( \ell \), \( 1 \leq \ell \leq m \), which is computed as

\[
\eta_\ell = \prod_{i=1}^{\ell-1} (n_i - 1) \prod_{i=\ell+1}^{m} n_i.
\]

SC decoding follows the schedule introduced in [15] for polar codes. Explicitly, the decision on the \( i \)-th information bit is made according to the soft-information obtained by performing a message-passing algorithm which propagates messages from the root of the decoding graph, along the edges of the tree rooted in \( u_1 \), where the operations at the local codes take into account the past decisions \( u_1^{i-1} \). The decoding graph for the \((9, 4)\) SPC-PC is provided in Fig. 2, by introducing check nodes (CNs) and variable nodes (VNs) to the encoder graph given in Fig. 1. We denote the soft-messages coming from the right of the decoding graph, along the edges performing a message-passing algorithm which propagates information obtained by iterating \((3, 2)\) SPC codes, is illustrated in Fig. 1. We label the levels by numbers starting from right to left as it is seen for the 2-dimensional case in the figure. We denote by \( \eta_\ell \) the number of local SPC codes at level \( \ell \), \( 1 \leq \ell \leq m \), which is computed as

\[
\eta_\ell = \prod_{i=1}^{\ell-1} (n_i - 1) \prod_{i=\ell+1}^{m} n_i.
\]

SC decoding follows the schedule introduced in [15] for polar codes. Explicitly, the decision on the \( i \)-th information bit is made according to the soft-information obtained by performing a message-passing algorithm which propagates messages from the root of the decoding graph, along the edges of the tree rooted in \( u_1 \), where the operations at the local codes take into account the past decisions \( u_1^{i-1} \). The decoding graph for the \((9, 4)\) SPC-PC is provided in Fig. 2, by introducing check nodes (CNs) and variable nodes (VNs) to the encoder graph given in Fig. 1. We denote the soft-messages coming from the right of the decoding graph, along the edges performing a message-passing algorithm which propagates information obtained by iterating \((3, 2)\) SPC codes, is illustrated in Fig. 1. We label the levels by numbers starting from right to left as it is seen for the 2-dimensional case in the figure. We denote by \( \eta_\ell \) the number of local SPC codes at level \( \ell \), \( 1 \leq \ell \leq m \), which is computed as

\[
\eta_\ell = \prod_{i=1}^{\ell-1} (n_i - 1) \prod_{i=\ell+1}^{m} n_i.
\]

SC decoding follows the schedule introduced in [15] for polar codes. Explicitly, the decision on the \( i \)-th information bit is made according to the soft-information obtained by performing a message-passing algorithm which propagates messages from the root of the decoding graph, along the edges of the tree rooted in \( u_1 \), where the operations at the local codes take into account the past decisions \( u_1^{i-1} \). The decoding graph for the \((9, 4)\) SPC-PC is provided in Fig. 2, by introducing check nodes (CNs) and variable nodes (VNs) to the encoder graph given in Fig. 1. We denote the soft-messages coming from the right of the decoding graph, along the edges performing a message-passing algorithm which propagates information obtained by iterating \((3, 2)\) SPC codes, is illustrated in Fig. 1. We label the levels by numbers starting from right to left as it is seen for the 2-dimensional case in the figure. We denote by \( \eta_\ell \) the number of local SPC codes at level \( \ell \), \( 1 \leq \ell \leq m \), which is computed as

\[
\eta_\ell = \prod_{i=1}^{\ell-1} (n_i - 1) \prod_{i=\ell+1}^{m} n_i.
\]

Consider the \( j \)-th local code at level \( \ell \), whose decoding graph is provided in Fig. 3. The soft output message for the \( i \)-th local information bit is computed as

\[
m_{j,i}^{(\ell)} = \rho_{j,i}^{(\ell)} + 2 \tanh \left[ \frac{\theta_{j,i}^{(\ell)}}{2} \right] \left( -1 \right)^{\sum_{i=1}^{\ell} \lambda_{j,i}^{(\ell)} }
\]

where \( \lambda_{j,i}^{(\ell)} \) is the hard input (i.e., bit) message for the \( z \)-th local information bit, coming from the left, with \( z = 1, \ldots, i-1 \), depending on the past decisions. The computed output message \( m_{j,i}^{(\ell)} \) is propagated leftwards over the tree edge, providing the next level with an input message. In particular, we set

\[
\rho_{j,i}^{(\ell+1)} = m_{j,i}^{(\ell)}
\]

where the assignment is made according to the graph connections, i.e., the PC structure. The decision is made as

\[
\hat{u}_{(j-1)(n_m-1)+i} = \begin{cases} 
0 & \text{if } m_{j,i}^{(n_m)} \geq 0 \\
1 & \text{if } m_{j,i}^{(n_m)} < 0
\end{cases}
\]

by breaking the ties in favor of zero. Over the BEC, ties are
made for input messages coming from the right. The final decision is
then the soft output message

According to the connections in the graph, we
illustrate an intermediate SC decoding step in Fig. 4, where
A block error event occurs if \( \hat{u}_1 \neq u_1 \).

**Example 1.** Consider the 2-dimensional (9, 4) SPC-PC obtained by iterating (3, 2) SPC codes. Its decoding graph is provided in Fig. 2. The number of local codes at levels 1 and 2 are computed respectively as \( \eta_1 = 3 \) and \( \eta_2 = 2 \). We illustrate an intermediate SC decoding step in Fig. 4, where the decisions for the first two information bits are already made and the decoder computes the soft message for the third bit. At level 1, the j-th local code has the hard input message \( \lambda_{j,1}^{(1)} \) and the soft input messages \( \rho_{j,1}^{(1)}, \rho_{j,2}^{(1)}, \rho_{j,3}^{(1)} \). Using (1) or (5) depending on the previous decisions, it computes the soft output message \( m_{j,2}^{(1)} \) providing the next level with a soft input message. According to the connections in the graph, we have the following assignments:

\[
\rho_{2,1}^{(2)} \leftarrow m_{1,2}^{(1)}, \quad \rho_{2,2}^{(2)} \leftarrow m_{2,2}^{(1)}, \quad \text{and} \quad \rho_{2,3}^{(2)} \leftarrow m_{3,2}^{(1)}.
\]

Then, the soft output message \( m_{2,1}^{(2)} \) is computed with the soft input messages coming from the right. The final decision is made for \( u_3 \) as in (3) or (4) depending on the channel.

We revise (1) under Elias’ decoder as

\[
m_{j,i}^{(2)} = \rho_{j,i}^{(2)} + 2 \tanh \left( \prod_{i' \neq i, i' = 1}^{n_i} \tanh \frac{\rho_{j,i'}^{(1)}}{2} \right).
\]

A. Analysis over the Binary Erasure Channel

We first analyze the behavior of a local \((n_\ell, n_\ell - 1)\) SPC (component) code under SC decoding over the BEC(\(\epsilon\)) \(W\). We denote by \(\epsilon^{(i)}\) the erasure probability of the \(i\)-th information bit after SC decoding conditioned on the correct decoding of the \(i-1\) preceding bits, with \(i = 1, \ldots, n_\ell - 1\). The relationship between the input-output erasure probabilities is given by

\[
\epsilon^{(i)} = \epsilon \left( 1 - (1 - \epsilon)^{n_\ell - i} \right) \quad \text{for} \quad i = 1, \ldots, n_\ell - 1.
\]

Denote the bits at the input of such a local SPC code encoder as \(v_1, \ldots, v_{n_\ell - 1}\) and the received values at the input of the corresponding local SPC decoder as \(b_1, \ldots, b_{n_\ell}\). Let \(I^{(i)}\) denote the mutual information between the RVs \(V_i\) and \((B_1^{n_\ell}, V_1^{n_\ell - 1})\), i.e., \(I^{(i)} \triangleq I(V_i; B_1^{n_\ell}, V_1^{n_\ell - 1})\). Then, the recursion (7) can be rewritten in terms of mutual information as

\[
I^{(i)} = 1 - (1 - I) \left( 1 - I^{n_\ell - i} \right) \quad \text{for} \quad i = 1, \ldots, n_\ell - 1
\]

with \(I \triangleq 1 - \epsilon\) and \(I^{(i)} \triangleq 1 - \epsilon^{(i)}\).

**Proposition 1.** The mutual information at the input of an SPC local decoder in the \(\ell\)-th dimension is not preserved at its output, i.e.,

\[
\sum_{j=1}^{n_\ell - 1} I^{(j)} < n_\ell I.
\]

**Proof.** We have that

\[
\sum_{j=1}^{n_\ell - 1} I^{(j)} = \sum_{j=1}^{n_\ell - 1} \left[ 1 - (1 - I) \left( 1 - I^{n_\ell - j} \right) \right] = (n_\ell - 1) I + (1 - I) \left( 1 - I^{n_\ell - 1} \right) \frac{1 - I^{n_\ell - 1}}{1 - I} = n_\ell I - I^{n_\ell} < n_\ell I.
\]

Proposition 1 provides also the loss of mutual information due to a local SPC code in the \(\ell\)-th dimension, which is \(I^{n_\ell}\). By recursively applying the transformation (7), one can derive the erasure probability associated with the information bits of an SPC-PC. Denote such erasure probabilities as \(q_i\), \(i = 1, \ldots, k\). The evolution of the corresponding mutual information values at each transformation level is illustrated in Fig. 5, where \(I = 0.3\), i.e., \(\epsilon = 0.7\), for the original channel.

The largest bit erasure probability is equal to that of the first decoded information bit, i.e., \(q_{\text{max}} \triangleq \max_{i=1,\ldots,k} q_i = q_1\). The block error probability under SC decoding, denoted by \(P_{\text{SC}}\), is bounded as \([15]\)

\[
q_{\text{max}} \leq P_{\text{SC}} \leq \sum_{i=1}^{k} q_i.
\]

A looser upper bound can be obtained by tracking only the largest erasure probability as

\[
P_{\text{SC}} \leq k q_{\text{max}}.
\]
Proof. The block error event is defined as
\[ E_E = \{(u_1, y_1^n) \in \mathcal{X}^k \times \mathcal{Y}^n : \hat{u}_1(y_1^n) \neq u_1^k\} \]
where \( \hat{u}_1(y_1^n) \) is the output of Elias’ decoder, obtained by using (2), (4) and (6). The bit error event is defined as
\[ B_i = \{(u_i^k, y_i^n) \in \mathcal{X}^k \times \mathcal{Y}^n : \hat{u}_i(y_i^n) \neq u_i\} \quad \text{for} \quad i = 1, \ldots, k. \]
The block error event satisfies \( E_E = \bigcup_{i=1}^{k} B_i \), which leads to
\[ \max_{i=1, \ldots, k} P(B_i) \leq P_E \leq \sum_{i=1}^{k} P(B_i). \] (14)
We conclude the proof by combining (14) and Lemma 1.

Theorem 1. The block error probability \( P_E \) of an SPC-PC over the BEC(\( \epsilon \)) under Elias’ decoding \([1]\) is bounded as
\[ q_{\max} \leq P_E \leq kq_{\max}. \] (13)
Proof. Over the BEC, both decoders can either make a correct decision or get an erasure for an information bit according to (4). Therefore, SC decoding cannot make use of any wrong bit decision. Recall (5) for SC decoding and (6) for Elias’ decoding. Under the former, the preceding decisions are exploited. However, each bit decision under the latter is made in the same manner as if one of the past decisions coming from left is an erasure for a local decoder under SC decoding. More precisely, assume that we apply both SC decoding and Elias’ decoding to an observation \( y_i^n \). Furthermore, assume at an intermediate step, the SC decoder computes the output message \( m_{j,i}^{(3)} \) corresponding to the \( i \)-th information bit of the \( j \)-th local code at the \( t \)-th dimension such that \( 1 \leq \ell \leq m, 1 \leq j \leq q_{\ell t}, 1 \leq i \leq n_{\ell t} \). Assume also that there exists at least one \( z, 1 \leq z \leq i-1, \) such that \( \lambda_{j,z}^{(3)} = \epsilon \). Note that having \( \lambda_{j,z}^{(3)} = \epsilon \) implies also that \( \rho_{j,z}^{(3)} = 0 \). Then, (5) computes the message as \( m_{j,i}^{(3)} = \rho_{j,i}^{(3)} \). For the same scenario under Elias’
In the figure, we also show the truncated union bound SC and Elias’ decoding algorithms for the B-AWGN channel. Iterating 3 times, the bounds are tight especially in the low error rate regime.

The given error probabilities are not simulation results, but are based on the block error probability under ML decoding given by

$$P_B \approx A_{\text{min}}e^d.$$

The given error probabilities are not simulation results, but are computed exactly under the three decoding algorithms thanks to the short length of the code. We observe that the bounds are tight especially in the low error rate regime.

### B. Performance over the binary input AWGN Channel

In Fig. 7, we provide the block error rate performance of the 3-dimensional (125, 64) and (216, 125) SPC-PCs, obtained by iterating (5, 4) and (6, 5) SPC codes, respectively, under both SC and Elias’ decoding algorithms for the B-AWGN channel. In the figure, we also show the truncated union bound

$$P_B \approx \frac{1}{2}A_{\text{min}}\text{erfc}\sqrt{dR\frac{E_b}{N_0}},$$

which, for high $E_b/N_0$, approximates tightly the ML decoding performance. Here, $E_b$ denotes the energy per information bit and $N_0$ is the single-sided noise power spectral density. For both codes, the difference of SC decoding to the truncated union bound is around 1 dB at block error rate of $10^{-4}$. In addition, SC decoding outperforms Elias’ decoding.

### IV. Conclusions

We introduced successive cancellation decoding of the class of product codes obtained by iterating single parity-check codes. Thanks to the structure of SPC-PCs, we showed how to compute the decision metrics recursively under SC decoding, which resembles SC decoding of polar codes. In addition, we introduced an analysis on the binary erasure channel, yielding lower and upper bounds on the block error probability. The performance of SC decoding is then compared with that of the original decoder of PCs introduced by Elias, showing that SC decoding yields lower block error probability than Elias’ decoding. Finally, it is concluded for the analyzed codes that the low-complexity SC decoding can outperform Elias’ decoding by $\sim 0.35$ dB.

### References