BER Bounds on Parallel Concatenated Single Parity Check Arrays and Zigzag Codes

Xiaoling Huang[†]*,Nam Phamdo*[†] *and Li Ping*[‡]

[†]Department of Electrical and Computer Engineering, State University of New York at Stony Brook
Stony Brook, NY 11794-2350, xhuang@sbee.sunysb.edu, phamdo@sbee.sunysb.edu

[‡]Department of Electronics Engineering, City University of Hong Kong
Hong Kong, eeliping@cityu.edu.hk

ABSTRACT

A union bound analysis of the bit error probability of parallel concatenated single parity check (SPC) array and zigzag code is presented. It is shown that the union bounds for these codes can be generated very efficiently. It is also shown that the simple codes studied can achieve comparable performances as the turbo codes (yet has much lower decoding costs as discussed in an earlier paper). It illustrates that, for a fixed interleaver size, the concatenated code has increased code potential as its dimension, i.e., the number of interleavers, increases (which raises decoding costs linearly, not exponentially as for increased constraint length). Finally, the analysis shows that the zigzag codes with dimension greater than three have lower error floors than comparable two-dimensional turbo codes.

1. INTRODUCTION

Turbo codes[1][2], also known as parallel concatenated convolutional codes, have attracted the interests of many researchers due to their promising performance. In [3], Benedetto and Montorsi proposed a technique for generating performance bounds for turbo codes assuming ML (Maximum Likelihood) soft-decision decoding and ideal uniform interleavers. This method can be used to evaluate turbo codes well beyond the bit error rate (BER) range of simulation possibility. It gives much insight into the principle of turbo codes, e.g., it reveals that the excellent performance of turbo codes is a consequence of low coefficient multiplicity, rather than high minimum-distance. This provides a satisfactory explanation of the "error floor" characteristic of turbo codes, which can also be observed from simulation. From a programming point of view, the method of [3] requires the weight enumerating function of the constituent code.

One practical concern of turbo codes is the complexity of the iterative decoding algorithm involved. Recently, it has been shown that multi-dimensional concatenated SPC (single parity check) arrays [4] can be a low-cost alternative for high rate schemes. However, this method has a high error floor problem and is also relatively weak for medium to low rates. A solution to these two problems has been devised in [5], employing the so-called zigzag codes. The simulation based studies in [4] and [5] demonstrated that the parallel concatenated SPC arrays and zigzag codes can be decoded by very low complexity decoding rules which still produce near Shannon capacity limit performance.

This paper is concerned with the performance bounds of parallel concatenated SPC arrays and zigzag codes. Our purpose is twofold. First, we will present some very simple methods to generate the performance bounds for the above mentioned codes. Second, we will use numerical results to investigate the performance of these codes. In particular, we will examine the error floor behavior, which is difficult to study by the simulation approach due to the computing power limitation. We will show analytically that the concatenated zigzag codes can achieve lower error floor than standard turbo codes – a property that has been reported in [5] by simulation and can be important for high quality data communication channels.

2. DESCRIPTION OF SPC ARRAYS AND ZIGZAG CODES

Let $D = \{d(i,j)\}, i = 1,2,\ldots,I, j = 1,2,\ldots,J$, denote an $I \times J$ matrix of information bits and $P = \{p(i)\}, i = 1,2,\ldots,I$, denote a parity check (column) vector. The following is the encoding principle for the SPC array and the zigzag code, respectively.

2.1. Single Parity Check (SPC) Array

Each parity check bit $p(i)$ is generated such that $\{d(i,1),...,d(i,J),p(i)\}$ forms an even SPC code [4], i.e.,

$$p(i) = \left(\sum_{j=1}^{J} d(i,j)\right) \bmod 2, \qquad i = 1,2,\ldots,I. \qquad (1)$$

The code array is completely parameterized by the pair (I,J). It is a systematic code with coding rate $J/(J+1)$ and $(n,k) = (IJ+I, IJ)$.

2.2. Zigzag Code

The zigzag code is described in [5]. A *segment* is defined as $[p(i-1), d(i,1), d(i,2), \ldots, d(i,J), p(i)]$. The parity check bits are generated such that each segment contains even number of ones. The codeword contains $I \times J$ information bits and I parity check bits ($(n,k) = (IJ+I, IJ)$), which

0-7803-5796-5/99/$10.00 © 1999 IEEE

are formed as follows:

$$\begin{aligned} p(1) &= \left(\sum_{j=1}^{J} d(1,j)\right), \\ p(i) &= \left(\sum_{j=1}^{J} d(i,j) + p(i-1)\right), \quad i = 2,3,\ldots,I, \end{aligned} \tag{2}$$

assuming modulo-2 addition.

2.3. Parallel Concatenated Codes

Let D be the information bits for a code C and $D_m = \pi_m(D)$ be an interleaved version of D, $m = 1,2,\ldots,M$. For each D_m, we form a parity vector P_m using code C. We will call $[D, P_1, P_2, \ldots, P_M]$ the M-dimensional parallel concatenation of C, hereafter denoted by C^M.

A standard turbo code [1][2] is a two dimensional instance of the above definition with C being a convolutional code. The concatenated SPC array studied in [4] is another instance using an SPC array for C. Similarly, we can also use a zigzag code for C to form a concatenated zigzag code. For the latter two cases, the overall code rates are both $J/(J+M)$ and $(n,k) = (IJ+IM, IJ)$.

3. BER BOUNDS

3.1. General Parallel Concatenated Codes

The following is an outline of the procedure developed in [3] for finding the BER union bound of a general parallel concatenated code. Let

$$A^C(W,Z) \triangleq \sum_{w=0}^{k} \sum_{j=0}^{n-k} A^C_{w,j} W^w Z^j, \tag{3}$$

be the *input-redundancy weight enumerating function* (IRWEF) for an (n,k) linear systematic block code C, where $A^C_{w,j}$ is the number of codewords with input weight w, parity weight j, and codeword weight $w+j$. Numerically, the coefficients of $A^C(W,Z)$ can be represented by a matrix of size $(k+1) \times (n-k+1)$, with row and column indices corresponding to w and j, respectively.

The *conditional weight enumerating function* (CWEF) [3] for C is defined as

$$A^C_w(Z) \triangleq \sum_{j=0}^{n-k} A^C_{w,j} Z^j, \qquad w = 0,1,\ldots,k. \tag{4}$$

Numerically, the coefficients of $A^C_w(Z)$ form the w-th row of the matrix for $A^C(W,Z)$. Based on the uniform interleaver assumption of [3], the CWEF of C^M can be found from that of C as,

$$A^{C^M}_w(Z) = \frac{[A^C_w(Z)]^M}{\binom{IJ}{w}^{M-1}}, \qquad w = 0,1,\ldots,IJ, \tag{5}$$

where IJ is the interleaver size. Using the coefficients of $A^{C^M}_w(Z)$ as the w-th row, we can find $A^{C^M}(W,Z)$ for the concatenated code C^M. Finally, the union bound on BER for C^M is given by [3]:

$$P_b \leq \frac{1}{2} \sum_{w=0}^{k} \sum_{j=0}^{n-k} \frac{w}{k} A^{C^M}_{w,j} \cdot \operatorname{erfc}\left(\sqrt{(w+j)\frac{R_c E_b}{N_0}}\right), \tag{6}$$

where $R_c = k/n$ is the code rate and E_b/N_0 is the bit signal-to-noise ratio. (Note that w/k is the conditional BER given that a (w,j)-type decoding error has occurred.) In (6), we have assumed BPSK modulation, an additive white Gaussian noise (AWGN) channel and soft-decision ML decoding.

The key issue for the method outlined above is to derive the IRWEF for the constituent code C. This can be a complicated issue for a general code (e.g., a general convolutional code). However, we will show in the following that, due to the simple code structures of the SPC arrays and zigzag codes, their IRWEF can be generated by some very simple formulas.

3.2. Parallel Concatenated SPC Arrays

Let $\tilde{C}$ denote the $(n,k) = (J+1, J)$ SPC code. It is easy to show that

$$A^{\tilde{C}}(W,Z) = \sum_{j=0}^{\lfloor J/2 \rfloor} \binom{J}{2j} W^{2j} + \sum_{j=0}^{\lfloor (J-1)/2 \rfloor} \binom{J}{2j+1} W^{2j+1} Z. \tag{7}$$

Let C denote the (I,J) SPC array based on $\tilde{C}$. Since C is equivalent to $\tilde{C}$ used I times, the IRWEF for C is simply

$$A^C(W,Z) = [A^{\tilde{C}}(W,Z)]^I. \tag{8}$$

Numerically, the coefficients of $A^{\tilde{C}}(W,Z)$ form a $(J+1) \times 2$ matrix. The coefficient matrix of $A^C(W,Z)$ has size $(IJ+1) \times (I+1)$. It can be obtained from $A^{\tilde{C}}(W,Z)$ by applying the two-dimensional convolution $I-1$ times. $A^{C^M}_w(Z)$ can be readily obtained from equation (5).

3.3. Parallel Concatenated Zigzag Codes

In the following, we will develop a recursive method for generating the IRWEF of a zigzag code. Let $A^{C_i}(W,Z)$ be the IRWEF of a zigzag code with i rows and J columns. We can decompose $A^{C_i}(W,Z)$ into even and odd parts as

$$A^{C_i}(W,Z) = A^{C_i}_{even}(W,Z) + A^{C_i}_{odd}(W,Z), \tag{9}$$

where $A^{C_i}_{even}(W,Z)$(resp. $A^{C_i}_{odd}(W,Z)$) includes all the terms with even (resp. odd) powers of W. Clearly, for $i = 1$,

$$A^{C_1}_{even}(W,Z) = \sum_{j=0}^{\lfloor J/2 \rfloor} \binom{J}{2j} W^{2j} \triangleq B_{even}(W), \tag{10}$$

$$A^{C_1}_{odd}(W,Z) = \sum_{j=0}^{\lfloor (J-1)/2 \rfloor} \binom{J}{2j+1} W^{2j+1} Z \triangleq B_{odd}(W) Z. \tag{11}$$

Suppose that $A^{C_{i-1}}(W,Z) = A^{C_{i-1}}_{even}(W,Z) + A^{C_{i-1}}_{odd}(W,Z)$ is known and consider an extra i-th row. Notice that from (2)

$$p(i) = p(i-1) + \sum_{j=1}^{J} d(i,j) = \sum_{i'=1}^{i}\sum_{j=1}^{J} d(i',j). \qquad (12)$$

Thus $p(i)$ actually indicates the parity of the total input weight from row 1 to row i. Let n_i be the parity of $d(i,1),\ldots,d(i,J)$. $A^{C_i}_{even}(W,Z)$ corresponds to two situations: $p(i-1)$ and n_i are both even or they are both odd. In either case $p(i)$ is zero. This leads to

$$\begin{aligned} A^{C_i}_{even}(W,Z) &= A^{C_{i-1}}_{even}(W,Z)B_{even}(W) \\ &\quad + A^{C_{i-1}}_{odd}(W,Z)B_{odd}(W). \end{aligned} \qquad (13)$$

Similarly, $A^{C_i}_{odd}(W,Z)$ corresponds to the situation where $p(i-1)$ is odd and n_i is even, or vice versa. In either case $p(i)$ is one. This leads to

$$\begin{aligned} A^{C_i}_{odd}(W,Z) &= \Big[A^{C_{i-1}}_{odd}(W,Z)B_{even}(W) \\ &\quad + A^{C_{i-1}}_{even}(W,Z)B_{odd}(W)\Big] Z. \end{aligned} \qquad (14)$$

Finally, the IRWEF, $A^{C^M}(W,Z)$, of the parallel concatenated zigzag code can be obtained by applying the one-dimension convolution $M-1$ times to each row of $A^{C_I}(W,Z)$ and then dividing by $\binom{IJ}{w}^{M-1}$ — just as in equation (5).

3.4. Some Numerical Issues

It is worthwhile to mention that in computing the union bound, we may encounter two numerical problems when the interleaver size, IJ, is large. The first is that the values of $A^{C}_{w,j}$ may be extremely large (well beyond the limit of IEEE floating-point numbers). The second problem is that the number of terms in the double sum in (6) is very large — resulting in a significant amount of computation. The first problem is solved by representing each value of $A^{C}_{w,j}$ by a floating-point number (mantissa) and a 16-bit integer (exponent). Special C subroutines are written for multiplication/division and addition/subtraction. The second problem is solved by approximating the double sum in (6) by a smaller sum which includes only those values of (w,j) for which $0 \leq w+j \leq D_{max}$. Note that with this approach, the whole matrix $A^{C^M}(W,Z)$ need not be calculated. This approach is similar to [3] and is based on the observation that the upper bound is dominated by the codewords with low and moderate Hamming weights. To obtain an accurate analysis, the value of D_{max} must be sufficiently large. Fig. 1 reports the effects of D_{max} on the bound of parallel concatenated zigzag code with $(I,J,M) = (256,16,4)$. It illustrates that all the curves merge around 10^{-9} and stay together from there on. $D_{max} = 500$ produces almost same result as $D_{max} = 800$, thus we conclude that $D_{max} = 500$ is accurate enough for the performance of the $(I,J,M) = (256,16,4)$ concatenated zigzag code. In the following, we report some analytical results on parallel concatenated SPC arrays and zigzag codes according to this approximation scheme. In all cases, we choose $D_{max} = 500$.

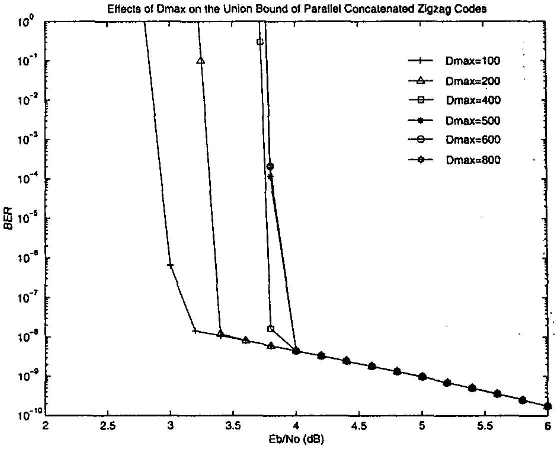

Figure 1. Effects of D_{max} on the Union Bound of Parallel Concatenated Zigzag Codes. $(I,J,M) = (256,16,4)$, $(n,k) = (5120,4096)$ and Rate=4/5.

4. NUMERICAL RESULTS

Fig. 2 presents the analytical union bounds of parallel concatenated SPC array and zigzag code with fixed rate 1/2 and different interleaver size ($IJ = 16, 64$ and 256). For the parallel concatenated SPC array, one can observe that increasing the interleaver size does not improve the performance much. The reason for this can be explained as follows.

Two terms of coefficients $A^{C^M}_{w,j}$ in (6) dominate the performance of parallel concatenated SPC array.

- The first term, $A^{C^M}_{1,M}$, is for the input sequences with weight of one, which result in codewords with parity weight of M ($P_1, P_2, \ldots, P_M$ each has Hamming weight one). There are IJ such codewords so that $A^{C^M}_{1,M} = IJ$.
- The second term, $A^{C^M}_{2,0}$, corresponds to the situation of weight-2 input sequences where the only 2 ones appear in the same row in every dimension (hence $P_1, P_2, \ldots, P_M$ are all zeros). It can be shown that $A^{C^M}_{2,0} = I\binom{J}{2}\left(\frac{J-1}{I\times J-1}\right)^{M-1}$.

We observed that the effects of these two terms are different in different E_b/N_0 ranges. For modest E_b/N_0, the first term dominates since typically (for sufficiently large IJ) $A^{C^M}_{1,M}$ is much larger than $A^{C^M}_{2,0}$. In this case, the BER

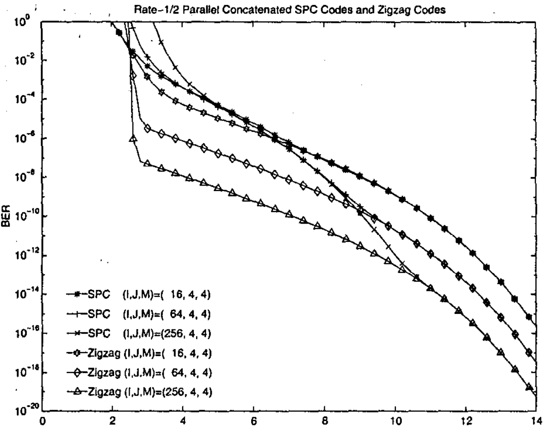

Figure 2. Upper Bounds to the Bit Error Probability of Parallel Concatenated SPC and Zigzag Codes with Different Interleaver Size. Fixed Dimension $M = 4$ and Rate $= 1/2$. $D_{max} = 500$.

performance can be approximated by:

$$P_b \approx \frac{1}{2} \cdot \frac{A_{1,M}^{C^M}}{IJ} \cdot \operatorname{erfc}\left(\sqrt{(M+1)\frac{R_c E_b}{N_0}}\right). \tag{15}$$

This approximation is shown by the dashed line in Fig. 3 for $(I, J, M) = (256, 4, 4)$, which is quite accurate for E_b/N_0 from 4 dB to about 10 dB. Notice that (15) is independent of the interleaver size IJ. Hence increasing interleaver size cannot improve the performance of a concatenated SPC array in this range.

At very high E_b/N_0, the effect of the second term takes over since it has lower weight. In this case, the BER performance can be approximated by:

$$P_b \approx \frac{1}{2} \cdot \frac{2}{IJ} \cdot A_{2,0}^{C^M} \cdot \operatorname{erfc}\left(\sqrt{\frac{2R_c E_b}{N_0}}\right). \tag{16}$$

The above approximation is shown by the dotted line in Fig. 3 for $(I, J, M) = (256, 4, 4)$. Increasing interleaver length can normally improve the performance in this range, since the larger IJ, the smaller $A_{2,0}^{C^M}$.

Notice that (15) does not apply to the concatenated zigzag code. In fact, for the latter case, a weight-1 input sequence results in a parity vector of the form $0, 0, 0, \ldots, 1, 1, \ldots, 1$. The non-zero section starts from the segment containing the only non-zero input bit. This implies that for the weight-1 input sequences, the codeword weight increases with high probability as the interleaver size increases. Consequently the performance of concatenated zigzag code is only dominated by (16). This results in much better performance for the zigzag code based schemes in median E_b/N_0 range, as seen in Fig. 3. At very high E_b/N_0, the performances of the concatenated SPC array and zigzag code converge since then both are dominated by (16).

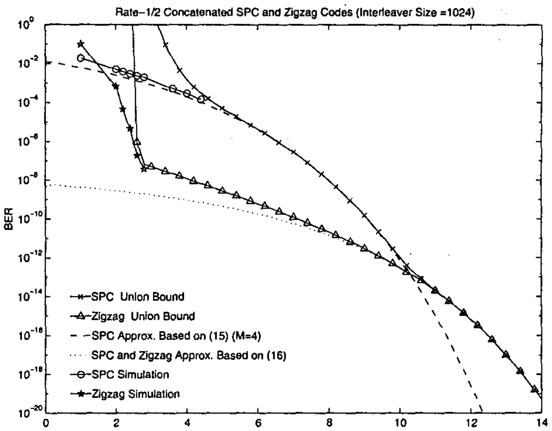

Figure 3. Union Bounds, Approximations of the Union Bounds, and Max-Log-MAP Simulation Results of Concatenated SPC and Zigzag Codes. $(I, J, M) = (256, 4, 4)$, $(n, k) = (2048, 1024)$, Rate $= 1/2$, and $D_{max} = 500$.

Fig. 4 compares the BER upper bounds for parallel concatenated zigzag codes with different dimension M, fixed interleaver size ($IJ = 1200$) and fixed code rate 1/2. This figure illustrates that as the dimension M increases, the code potential increases assuming optimal (ML) sequence decoding. This phenomenon can be attributed to the constant $\binom{IJ}{w}^{M-1}$ in equation (5) which grows exponentially in M. Similar results were found for other interleaver sizes, code rates and parallel concatenated SPC arrays. This finding suggests that to achieve a low BER with a small interleaver (hence low latency), a code with a large dimension can be used. However, it is worthwhile to mention that the sub-optimality of the iterative decoding algorithms described in [1], [2], [4], [5] becomes more enhanced as the dimension M increases. Furthermore, the decoding cost increases approximately linearly with M [5]. From our simulation experiments in [4] and [5], $M = 4$ seems to be the best compromise for rate-1/2 code.

In Fig. 4, we also show the union bound of the (37,21) [1], [2] and (23,35) [3] turbo codes with the same interleaver length (1200 bits). All the codes shown are not terminated. The rate-1/2 turbo code is achieved by puncturing all the parity bits with odd indices (the last parity bit is not punctured). This figure illustrates that the concatenated zigzag codes with dimension four or more can achieve lower error floor than standard (two-dimensional) turbo codes for the range of median E_b/N_0. It is noted that the union bound analysis yields an accurate assessment of the code performance only at reasonably high E_b/N_0. For very low E_b/N_0, the simulation results in [5] show that the (37,21) turbo code is better than the four-dimensional zigzag code by 0.3 - 0.4 dB at 10^{-4} BER. (However, the performance of concatenated zigzag code takes over for high E_b/N_0 and it also has much lower decoding complexity [5].)

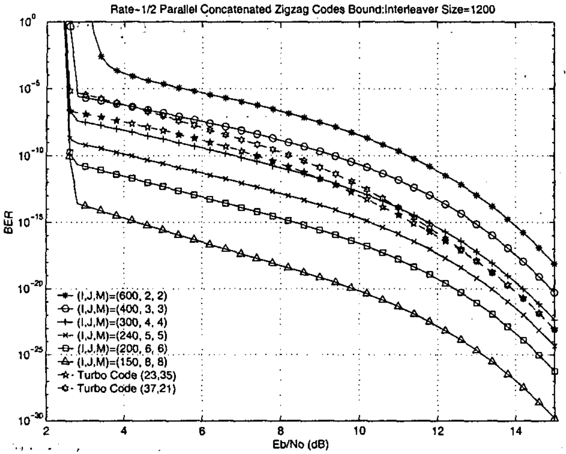

Figure 4. Upper Bounds to the Bit Error Probability of Parallel Concatenated Zigzag Codes with Different Dimension M and (37,21) and (23,35) Turbo Codes. $(n, k) = (2400, 1200)$, Rate $= 1/2$, and $D_{max} = 500$.

5. CONCLUSIONS

This paper assesses the upper bounds of the bit error probability of parallel concatenated SPC arrays and zigzag codes assuming ML decoding over an AWGN channel. Some very simple techniques for generating such bounds have been described and numerical results have been presented. The study verified the simulation results of [5] that these simple codes can achieve comparable performance as the turbo codes. It has also been shown that very low error floor can be achieved by going to higher dimensions.

REFERENCES

[1] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," *Proc. 1993 IEEE Int. Conf. on Communications*, pp. 1064-1070, May 1993.

[2] C. Berrou and A. Glavieux, "Near Optimum Error Correcting Coding and Decoding: Turbo Codes," *IEEE Trans. on Communications*, Vol. 44, No. 10, pp. 1261-1271, Oct. 1996.

[3] S. Benedetto and G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Trans. on Inform. Theory*, Vol. 42, No. 2, pp. 409-428, Mar. 1996.

[4] L. Ping, S. Chan and K. Yeung, "Iterative Decoding of Multi-Dimensional Concatenated Single Parity Check Codes," *Proc. 1998 IEEE Int. Conf. on Communications*, pp. 131-135, Jun. 1998.

[5] L. Ping and N. Phamdo, "Zigzag Codes and Concatenated Zigzag Codes," *IEEE Information Theory and Networking Workshop*, Metsovo, Greece, Jun. 1999.