Review of Deterministic Interleavers

Kwame Ackah Bohulu

25-05-2017

1 Introduction

Turbo codes are created by the parallel concatenation of two Recursive Systematic Convolutional (RSC) codes via an interleaver. The performance of turbo codes is dependent on it's effective free distance. The effective free distance is the minimum distance associated with input weight 2 error events [2]. Input weight 2 error events that are small multiples of the cycle length of the component code tend to produce low-weight codewords. It is desirable to prevent the occurance of such error events in both component codes to prevent low-weight Turbo codewords. The interleaver reorders input information bits before feeding them into the second component encoder. By designing an interleaver in such a way that it prevents input weight 2 error events present in the first component encoder from occuring in the second component encoder we will be able to increase the effective free distance of the turbo code and improve its performance.

2 Case 1: Prevention of duplication of weight 2 error events of length d

An input weight error event is defined as an error event with two information bits in error[2]. Each input weight error event in a turbo code corresponds to one input weight error event in each of the two component codes. Consider a typical input weight 2 error event shown in the diagram below. In the first component code, it begins at position x and ends at position $x+d$ where

$$x \in \mathbb{Z}$$

and

$$d \in \tau \cdot \mathbb{Z} \triangleq \mathbb{C}$$

where τ is the cycle length of the component encoder. We represent the start and end of the error event with the integer pair $(x, x+d)$These points are then rearranged via the interleaver to positions αx and $\alpha x + d$ and the distance between these positions is represented by

$$(\alpha(x+d), \alpha(d)) \triangleq \alpha(x+d) - \alpha(d)$$

. Since $d \in \mathbb{C}$, we may rewrite d as $a\tau$, where a is a small integer number.

Assuming the weight 2 error events in both component codes have lengths that are small multiples of τ (type 1 error events)we wish to investigate its effect on the bit error rate (BER) performance of the Turbo code. The BER performance of a convolutional code with maximum-likelihood (ML) decoding on an additive white Gaussian noise (AWGN) channel can be upper-bounded using a union bound technique by [2]

$$P_b \leq \sum_{i=1}^{2^N} \frac{w_i}{N} Q\left(\sqrt{d_i \frac{2RE_b}{N_o}}\right) \tag{1}$$

where w_i and d_i are the information weight and total Hamming weight, respectively of the ith codeword. Since the Turbo code is composed of convolutional codes, the above formula can also be used to find the upper bound on the BER performance of the Turbo code.

For type one error events the total output weight in relation to Turbo codes can be calculated using the equation [1]

$$w_{(t_i,s_i)} = 6 + \left(\frac{\sum |t_i|}{\tau} + \frac{\sum |s_i|}{\tau}\right) w_o \tag{2}$$

Where t_i and s_i are the lengths of the type one error events in the first and second component codes respectively and w_o is the weight of output sequence of either component codes when the input sequence is of the form $1 + D^{\tau}$. t_i and s_i can be written as $a\tau$ where a =1,2,3,4

By adjusting the values of t_i and s_i in (2) we can collect codewords of the same total Hamming weight and define the average information weight per codeword as [1]

$$w_d = \frac{W_d}{N_d}$$

Where W_d is the total information weight of all codewords of weight d and N_d is the multiplicity of codewords of weight d. We can then rewrite (1) as

$$P_b \leq \sum_{d=d_{(a=1)}}^{d_{(a=4)}} \frac{N_d w_d}{N} Q\left(\sqrt{d \frac{2RE_b}{N_o}}\right) \tag{3}$$

The BER performance graph for Turbo codes using six different recursive systematic convolutional (RSC) codes is shown in figure 1 below.

We wish to design an interleaver such that

$$(\alpha(x+d), \alpha(d)) \notin \mathbb{C} \triangleq \mathbb{E}(\alpha(x+d), \alpha(d)) \ \forall x \in \mathbb{Z}, d \in \mathbb{C}$$

2.1 Linear Interleaver design

The index mapping equation for the proposed linear interleaver is given by (1)

$$\alpha(x)_{\mathfrak{L}_N} = x(\tau^2 - z\tau + 1) \mod N \tag{4}$$

where $N = 2^n, n \in \mathbb{R}$ is the interleaver size, τ is the cycle length of the component encoder, z is the largest odd integer within the range 1 to $\sqrt{2N}$.

$N = 2^n$, $(\alpha(x+d), \alpha(x))$								
n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	τ
5	13	25	55	113	235	481	981	1
6	6	6	30	70	174	390	854	2
1	1	13	59	5	79	245	649	3
4	4	20	20	52	212	52	372	4
5	5	1	47	89	67	329	29	5
2	10	26	18	122	162	58	650	6
1	9	5	3	29	247	269	193	7
0	8	8	8	72	72	456	712	8

Table 1: Values for $(\alpha(x+d), \alpha(x))$, a=1

$N = 2^n$, $(\alpha(x+d), \alpha(x))$								
n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	τ
2	10	18	46	98	214	450	938	1
4	12	12	60	12	92	268	684	2
2	2	26	54	10	158	490	274	3
0	8	8	40	104	168	104	744	4
2	10	2	30	50	134	146	58	5
4	4	20	36	116	68	116	276	6
2	2	10	6	58	238	26	386	7
0	0	16	16	16	144	400	400	8

Table 2: Values for $(\alpha(x+d), \alpha(x))$, a=2

$N = 2^n$, $(\alpha(x+d), \alpha(x))$								
n=3	n=4	n=5	n=6	n=7	n=8	n=9	n=10	τ
7	7	11	37	83	193	419	895	1
2	2	18	26	82	10	146	514	2
3	3	7	49	15	237	223	923	3
4	12	28	60	28	124	156	92	4
7	15	3	13	11	201	475	87	5
6	14	14	54	110	230	174	926	6
3	11	15	9	87	229	295	579	7
0	8	24	24	88	216	344	88	8

Table 3: Values for $(\alpha(x+d), \alpha(x))$, a=3

3 References

[1] Oscar Y. Takeshita, Member, IEEE, and Daniel J. Costello ,"New Deterministic Interleaver Designs for Turbo Codes",IEEE Trans. Inform. Theory, vol. 46,pp. 1988-2006,Nov. 2000

[2] L. C. Perez, J. Seghers, D. J. Costello, Jr., "A distance spectrum interpretation of turbo codes", IEEE Trans. Inform. Theory, vol. 42, pp. 1698-1709, Nov. 1996.

[3] Jing Sun, Oscar Y. Takeshita " Interleavers for Turbo Codes Using Permutation Polynomials over Integer Rings" , IEEE Trans. Inform. Theory, vol. 51, pp. 101 - 119 Jan. 2005