TSG-RAN Working Group1 meeting #5   **TSGR1#5(99)735**
Cheju, KOREA, 1-4, June 1999

[1] Agenda Item:
[2] Source: SAMSUNG Electronics Co., NTT DoCoMo, and Nortel Networks
[3] Title: Agreement of incorporating PIL modification into 25.212, 25.222
[4] Document for: RAN WG-1 Approval

_____

1. Introduction

In [1], Samsung reported a Hamming weight analysis for the PIL interleaver that incorporated weight-1 input sequences. They showed that, under certain conditions, the free distance of Turbo code with PIL interleaver was low due to weight-1 input sequences, and proposed two possible revisions that improve the free distance of this case. [See Appendix.] No objection to the revisions was made from other companies. Thus, the chairman of Adhoc5 suggested that three companies have off-line discussion to settle down the above-mentioned problem.

2. Discussion and Conclusions

Samsung Electronics, NTT DoCoMo, and Nortel Networks had off-line discussion after the day 2 plenary session and reached the common consensus that the proposed modification of PIL is to be reflected into 25.212, 25.222, and [2]. They carefully investigated the modification and concluded that the modification neither influences any hardware structure nor increases hardware complexity. The final modification is the following.

In section 4.2.2.2.2.1,
(B-6) if $K = R \times C$ then, exchange $c_{R-1}(p)$ with $c_{R-1}(0)$

They agreed to make a new text proposal of PIL interleaver reflecting the method above. In order not to delay the current standardization process for new items, they also agreed to eliminate unnecessary future work by setting up certain criteria for further possible improvement of PIL, which means that, to incorporate further new modification into current PIL, the following three criteria should be satisfied.

- Change or increase in hardware should be negligible,
- Retain original structure of PIL interleaving algorithm , and
- Performance gain obtainable with the new method should be reasonable, for example, 0.1 dB in SNR.

3. Revised text proposal for Turbo code internal interleaver of 25.212, 25.222

4.2.2.2.2 Turbo code internal interleaver

<Editor's note: The following is a working assumption of Ad Hoc 5>

Figure 4-xx depicts the overall 8 State PCCC Turbo coding scheme including Turbo code internal interleaver. The Turbo code internal interleaver consists of mother interleaver generation and pruning. For arbitrary given block length K, one mother interleaver is selected from the 207 mother interleavers set. The generation scheme of mother interleavre is described in section 4.2.2.2.2.1. After the mother interleaver generation, _l_-bits are pruned in order to adjust the mother interleaver to the block length K. The definition of _l_ is shown in section 4.2.2.2.2.2.
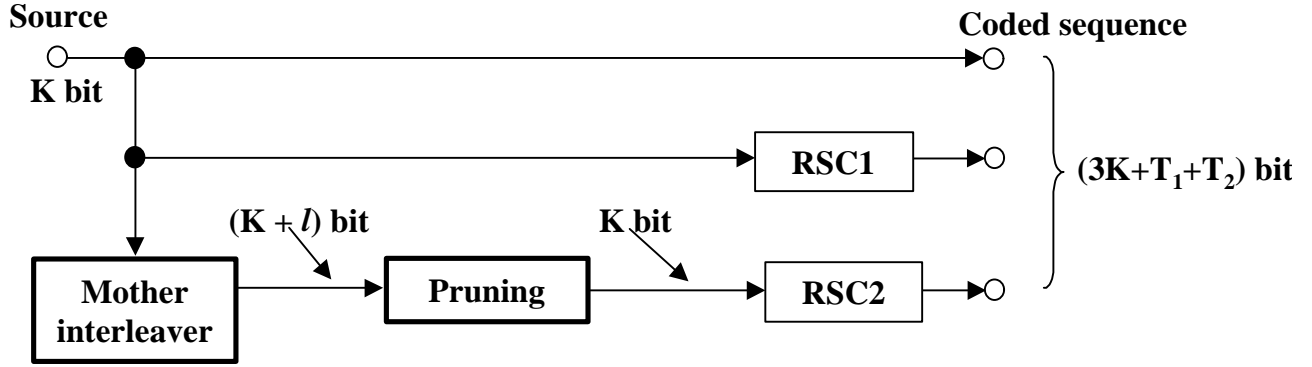
Figure 4-xx. Overall 8 State PCCC Turbo Coding

4.2.2.2.2.1 Mother interleaver generation

The interleaving consists of three stages. In first stage, the input sequence is written into the rectangular matrix row by row. The second stage is intra-row permutation. The third stage is inter-row permutation. The three-stage permutations are described as follows, the input block length is assumed to be K (320 to 8192 bits).

First Stage:

    (1)   Determine a row number R such that
           R=10 (K = 481 to 530 bits; Case-1)
           R=20 (K = any other block length except 481 to 530 bits; Case-2)
    (2) Determine a column number C such that
           Case-1; C = p = *53*
           Csae-2;
                 (i) Find minimum prime p such that,
                    $0 =< (p+1)$-K/R.
                (ii) If $(0 =< p$-K/R) then go to (iii),
                   else C = $p+1$.
                (iii) If $(0 =< p$-1-K/R) then C=p-1,
                   else C = p.
     (3) The input sequence of the interleaver is written into the R $\times$ C rectangular matrix row by

row.

Second Stage:

    A. If C = p
    (A-1) Select a primitive root g_0 from Table 4-yy.
    (A-2) Construct the base sequence c(i) for intra-row permutation as:
$$c(i) = [g_0 \times c(i-1)] \bmod p ,\qquad i = 1,2, \ldots (p\text{-}2)., c(0) = 1.$$

    (A-3) Select the minimum prime integer set $\{q_j\}$ (j=1,2, ..R-1) such that
           g.c.d$\{q_j, p\text{-}1\}$ =1
           $q_j > 6$
           $q_j > q_{(j-1)}$
       where g.c.d. is greatest common divider. And q_0 = 1.
    (A-4) The set $\{q_j\}$ is permuted to make a new set $\{p_j\}$ such that
           $p_{P(j)} = q_j$, j = 0, 1, ...R-1,
       where P(j) is the inter-row permutation pattern defined in the third stage.
    (A-5) Perform the j-th (j = 0, 1, 2, ... R-1) intra-row permutation as:
$$c_j(i) = c([i \times p_j] \bmod(p-1)) ,\quad i = 0, 1,2, \ldots (p\text{-}2)., \text{ and } c_j(p\text{-}1) = 0,$$

       where $c_j(i)$ is the input bit position of i-th output after the permutation of j-th row.

<u>B. If C = *p*+1</u>
(B-1) Same as case A-1.
(B-2) Same as case A-2.

(B-3) Same as case A-3.
(B-4) Same as case A-4.
(B-5) Perform the *j*-th (*j* = 0,1, 2, .., R-1) intra-row permutation as:

$$c_j(i) = c([i \times p_j] \bmod(p-1)), \quad i = 0,1,2, .., (p-2)., \quad c_j(p-1) = 0, \text{ and } c_j(p) = p,$$

where $c_j(i)$ is the input bit position of *i*-th output after the permutation of *j*-th row.]
<u>(B-6) If (K = C x R) then exchange $c_{R-1}(p)$ with $c_{R-1}(0)$.</u>

<u>C. If C = *p*-1</u>
(C-1) Same as case A-1.
(C-2) Same as case A-2.

(C-3) Same as case A-3.
(C-4) Same as case A-4.
(C-5) Perform the *j*-th (*j* = 0,1, 2, .., R-1) intra-row permutation as:

$$c_j(i) = c([i \times p_j] \bmod(p-1)) -1, \quad i = 0,1,2, .., (p-2),$$

where $c_j(i)$ is the input bit position of *i*-th output after the permutation of *j*-th row.

Third Stage:

(1) Perform the inter-row permutation based on the following P(*j*) (*j*=0,1, ..., R-1) patterns, where P(*j*) is the original row position of the *j*-th permuted row.

P_A: {19, 9, 14, 4, 0, 2, 5, 7, 12, 18, 10, 8, 13, 17, 3, 1, 16, 6, 15, 11} for R=20
P_B: {19, 9, 14, 4, 0, 2, 5, 7, 12, 18, 16, 13, 17, 15, 3, 1, 6, 11, 8, 10} for R=20
P_C: {9, 8, 7, 6, 5, 4, 3, 2, 1, 0} for R=10

The usage of these patterns is as follows:

Block length K: P(*j*)
320 to 480-bit: P_A
481 to 530-bit: P_C
531 to 2280-bit: P_A
2281 to 2480-bit: P_B
2481 to 3160-bit: P_A
3161 to 3210-bit: P_B
3211 to 8192-bit: P_A

(2) The output of the mother interleaver is the sequence read out column by column from the permuted R × C matrix.

Table 4-yy. Table of prime p and associated primitive root

p	G$_o$	*p*	g$_o$	*p*	g$_o$	*p*	g$_o$	*p*	g$_o$	*P*	g$_o$	*p*	g$_o$	*p*	g$_o$
17	3	59	2	103	5	157	5	211	2	269	2	331	3	389	2
19	2	61	2	107	2	163	2	223	3	271	6	337	10	397	5
23	5	67	2	109	6	167	5	227	2	277	5	347	2	401	3
29	2	71	7	113	3	173	2	229	6	281	3	349	2	409	21
31	3	73	5	127	3	179	2	233	3	283	3	353	3		
37	2	79	3	131	2	181	2	239	7	293	2	359	7		
41	6	83	2	137	3	191	19	241	7	307	5	367	6		
43	3	89	3	139	2	193	5	251	6	311	17	373	2		
47	5	97	5	149	2	197	2	257	3	313	10	379	2		
53	2	101	2	151	6	199	3	263	5	317	2	383	5		

3

4.2.2.2.2.2 Definition of number of pruning bits

The output of the mother interleaver is pruned by deleting the *l*-bits in order to adjust the mother interleaver to the block length K, where the deleted bits are non-existent bits in the input sequence. The pruning bits number *l* is defined as:

$$l = R \times C - K,$$

where R is the row number and C is the column number defined in section 4.2.2.2.2.1.

4. Reference

[1] Results of Hamming weight asymptote and performance analysis for Turbo interleaver candidates, TSGR1#5(99)559, Samsung Electronics
[2] Updated text for Turbo code internal interleaver of 25.212, 25.222, TSGR1#5(99)540, NTT DoCoMo and Nortel Networks
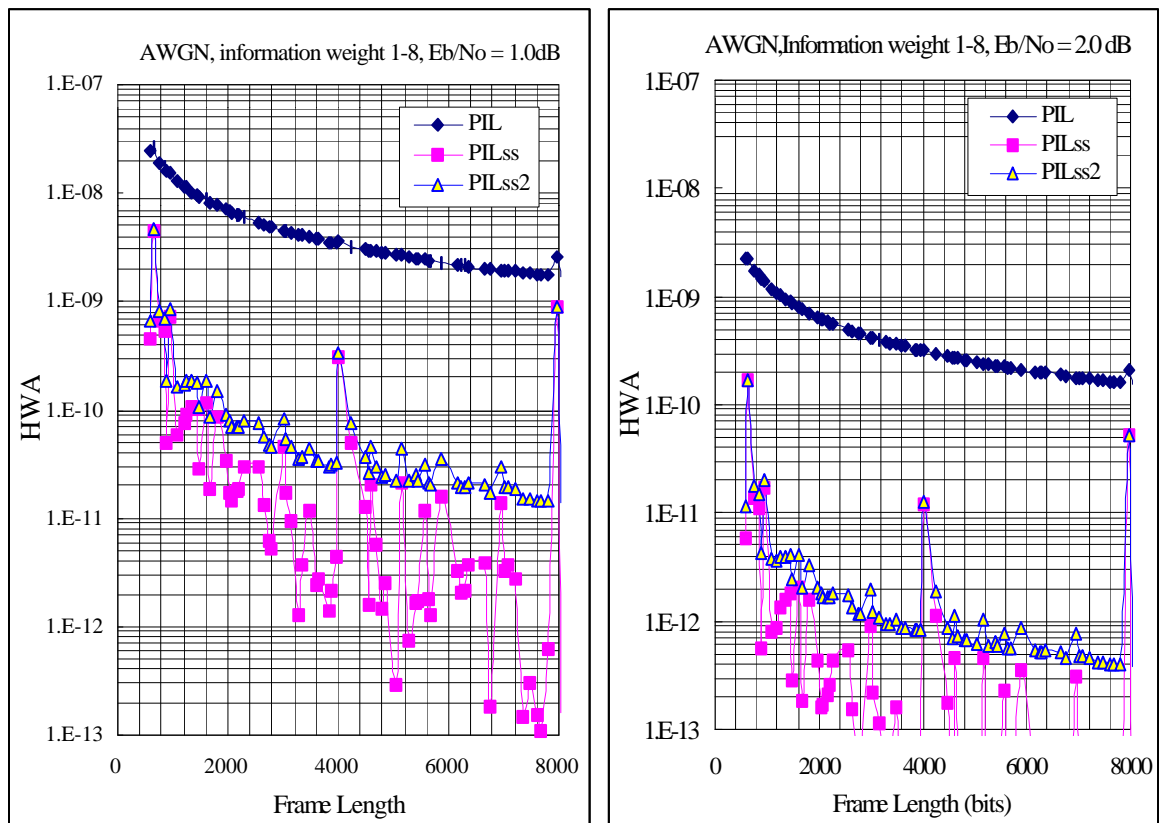
5. Appendix

Figure-1 HWA of PIL, PILSS, and PILSS2 at 1.0 and 2.0 dB. Refer to [1]