TSG-RAN Working Group 1 meeting #11 *TSGR1#10(00)0105*

San Diego, USA

February 29 – March 3, 2000

Agenda item:

Source: Nokia

Title: CR 25212-030r1: Clarification on turbo internal interleaver

Document for: Decision

This CR introduces a clarification on the internal interleaver section of TS 25.212. The functionality is not changed, only presentation is made more consistent with other parts of TS 25.212. After looking at the current internal interleaver description in detail we also found some items missing. Furthermore, TS 25.212 has been changed a lot since June 1999. Thus, we feel that there really is a need to rewrite the current text in order to make it consistent. Our findings are as follows:

1) the range of indexes should be from 1 to K inclusive (section 4.2.2.2 in TS 25.212), the current text probably may generate indexes from 0 to K-1, but this is unclear.

2) an explicit description for pruning is missing, it is only said how many indexes are pruned but not in which way the pruning is done. This is a consequence of 1), it is difficult to specify pruned bits without explicit indexing.

3) the conditions for a minimum prime integer set in A-3, B-3, and C-3 are not tight enough, they do not quarantee a unique selection of q_j. Hence a new additional requirement for selection has to be included: q_j is a *least* prime satisfying the selection conditions.

Example: K=320, so R=20, C=16, p=17, and p-1=16. The correct q_j is 1,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79. However, the current text allows to take any inceasing sequnece of prime numbers with q_0 =1, q_1 =7, so q_2 =151, q_3 =199, and so on.

Moreover, the current text applies "bad notation", i.e. a symbol has multiple meanings. Also the current text contradicts with use of the global symbols.

For example:

-C is a number, there is a step C and the pattern P_C.

-there is a step A and the pattern P_A.

-there is a step B and the pattern P_B.

The split of the algorithm into three disjoint parts A, B, and C is artificial. Instead, the three are nested of nature.

Finally, the current text does not specify how bits coming from section 4.2.2.2 are interleaved in 4.2.3.2.3

The text in this change request solves these above-mentioned problems.

3GPP RAN WG1 Meeting #11
San Diego, USA, 29 Feb – 3 Mar 2000

Document R1-00-0105
e.g. for 3GPP use the format TP-99xxx
or for SMG, use the format P-99-xxx

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

25.212 CR 030r1 Current Version: 3.1.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to:	RAN #7	for approval	X	strategic	
list expected approval meeting # here ↑		for information		non-strategic	*(for SMG use only)*

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)*	(U)SIM	ME X	UTRAN / Radio X	Core Network

Source: Nokia **Date:** 13-Jan-2000

Subject: Update for 4.2.3.2.3 of 25.212 for consistent description

Work item:

Category:				**Release:**	
	F	Correction	X	Phase 2	
	A	Corresponds to a correction in an earlier release		Release 96	
(only one category	B	Addition of feature		Release 97	
shall be marked	C	Functional modification of feature		Release 98	
with an X)	D	Editorial modification		Release 99	X
				Release 00	

Reason for change: The current text explains the Turbo Code Internal Interleaver in a very complicated way.
The connection to 4.2.2.2 is missing, i.e., there is no text to describe how bits from 4.2.2.2 are interleaved in 4.2.3.2.3. Also the current text contains three formal errors: unclear range of indexes, how pruning is done is not told, and the selection process of q_j is not tight enough. These errors will be fixed.

Clauses affected: 4.2.3.2.3 of TS25.212

Other specs affected:			
	Other 3G core specifications		→ List of CRs:
	Other GSM core specifications		→ List of CRs:
	MS test specifications		→ List of CRs:
	BSS test specifications		→ List of CRs:
	O&M specifications		→ List of CRs:

Other comments:

<--------- double-click here for help and instructions on how to create a CR.

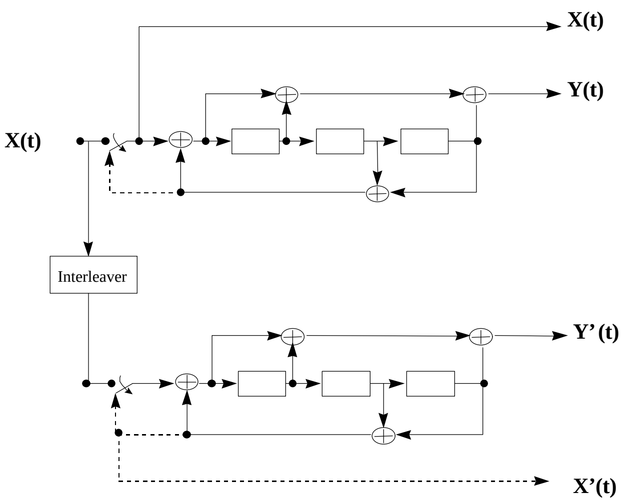


Figure 4: Structure of the 8 state PCCC encoder (dotted lines effective for trellis termination only)

The initial value of the shift registers of the PCCC encoder shall be all zeros.

The output of the PCCC encoder is punctured to produce coded bits corresponding to the desired code rate. For rate 1/3, none of the systematic or parity bits are punctured, and the output sequence is X(0), Y(0), Y'(0), X(1), Y(1), Y'(1), etc.

4.2.3.2.2 Trellis termination for Turbo coding

Trellis termination is performed by taking the tail bits from the shift register feedback after all information bits are encoded. Tail bits are added after the encoding of information bits.

The first three tail bits shall be used to terminate the first constituent encoder (upper switch of figure 4 in lower position) while the second constituent encoder is disabled. The last three tail bits shall be used to terminate the second constituent encoder (lower switch of figure 4 in lower position) while the first constituent encoder is disabled.

The transmitted bits for trellis termination shall then be

X(t) Y(t) X(t+1) Y(t+1) X(t+2) Y(t+2) X'(t) Y'(t) X'(t+1) Y'(t+1) X'(t+2) Y'(t+2).

4.2.3.2.3 Turbo code internal interleaver

Figure 4~~5~~ depicts the overall 8 state PCCC Turbo coding scheme including Turbo code internal interleaver. A length of a turbo code internal interleaver is allowed to take any value from 40 to 5114 inclusive assigned according to the rules desribed in 4.2.2.2. The length is denoted by K_i for a TrCH i. Elements of a turbo code internal interleaver are denoted by $T(k)$, $k = 1,2, ..., K_i$, and each of them stands for the original position of an k:th interleaved bit. The range of $T(k)$ is $1 \le T(k) \le K_i$.

The bits input to the turbo code internal interleaver are denoted by $o_{ir1}, o_{ir2}, o_{ir3}, \ldots, o_{irK_i}$ and the bits after interleaving are denoted by $x_{ir1}, x_{ir2}, x_{ir3}, \ldots, x_{irK_i}$, where i is a TrCH number and r is a code block number (for details see 4.2.2.2). The relationship between the two is defined by: $x_{irk} = o_{irT(k)}$ for $k = 1,2, ... ,K_i$.

Every interleaving index $T(k)$ shall satisfy the following stepwise algorithm:

4.2.3.2.3.1 Reference algorithm for turbo interleavers

The following section specific notation is used for the parameters in the algorithm:

K	Length of Turbo Code Internal Interleaver for a TrCH
A	Number of rows of an A times B matrix
B	Number of columns of an A times B matrix
l	Prime number
m	Primitive root for l
ROP	Row order pattern
BR	Base sequence
Q	Minimum prime integer sequence
MIS	Minimum row index sequence
i	Index in row dimension
j	Index in column dimension
i_0	Index in row dimension
z	Candidate index for Turbo Code Internal Interleaver

1. Assign values for the number of rows A, the number of columns B, the prime number l, and the primitive root m depending on K:

If $480 < K < 531$ then

- $A = 10$;

- $l = 53$;

- $B = 53$;

- $m = 2$;

else

- choose the number of rows A by

$$A = \begin{cases} 5 & if \quad 40 \le K < 160\ , \\ 10 & if \quad 160 \le K < 201\ , \\ 20 & otherwise. \end{cases}$$

- find a least prime l such that $K \le A*(l+1)$;

- select the number of columns B by

$$B = \begin{cases} l-1 & if \quad K \le A*(l-1), \\ l & if \quad A*(l-1) < K \le A*l\ , \\ l+1 & if \quad A*l < K \le A*(l+1). \end{cases}$$

- select m from Table 2 below on the right side of l.

endif

2. Select the row order pattern ROP out of $Pattern_1$, $Pattern_2$, $Pattern_3$, and $Pattern_4$ depending on K:

$$ROP = \begin{cases} Pattern_4 & if\ 39 < K < 160, \\ Pattern_3 & if\ 159 < K < 201\ or\ 480 < K < 531, \\ Pattern_2 & if\ 2280 < K < 2481\ or\ 3160 < K < 3211, \\ Pattern_1 & otherwise. \end{cases}$$

$Pattern_1$: {19, 9, 14, 4, 0, 2, 5, 7, 12, 18, 10, 8, 13, 17, 3, 1, 16, 6, 15, 11}.

$Pattern_2$: {19, 9, 14, 4, 0, 2, 5, 7, 12, 18, 16, 13, 17, 15, 3, 1, 6, 11, 8, 10}.

$Pattern_3$: {9, 8, 7, 6, 5, 4, 3, 2, 1, 0}.

$Pattern_4$: {4, 3, 2, 1, 0}.

3. Construct the base sequence $BR(j)$, j=0,1,2,..., $l - 2$, by $BR(0)=1$ and

$$BR(j) = (m * BR(j-1)) \text{ modulo } l \quad \text{for } j = 1,2,..., l-2.$$

4. Selection of the minimun prime integer sequence $Q(i)$ for i=0,1,..., $A - 1$: assign $Q(0)=1$ and choose a least prime number $Q(i)$ such that gcd($Q(i)$, $l - 1$)=1 , $Q(i)>6$, and $Q(i) > Q(i-1)$ for i=1,2,..., $A - 1$. Here gcd(x , y) is the greatest common divisor of integers x and y.

5. Calculate the minimum row index sequence $MIS(i)$, i=0,1,..., $A - 1$, by

$$MIS(i) = \begin{cases} ROP(i) * B & if\ B = l - 1, \\ ROP(i) * B + 1 & if\ B = l\ \ or\ B = l + 1. \end{cases}$$

6. All elements of $T(k)$ are same as ones obtained from the steps 6.1-6.3:

6.1. Set $i_0 = 0$ and $k = 1$;

6.1.1 if $K=A*B$ and $B=(l+1)$ then $T(k)=MIS(i_0) + l$; $k = k + 1$; $i_0 = i_0 + 1$; endif

6.1.2 for $j = 0, 1, 2, \ldots, l - 2$ do

6.1.3 for $i = i_0, i_0 + 1, i_0 + 2, \ldots, A - 1$ do

6.1.4 $z = MIS(i) + BR((j*Q(i))$ modulo $(l - 1))$;

6.1.5 if z £ K then $T(k) = z$; $k = k + 1$; else prune z; endif

6.1.6 endfor

6.1.7 $i_0 = 0$;

6.1.8 endfor

6.2. if $(l - 1) < B$ then

6.2.1 for $i = 0, 1, 2, \ldots, A - 1$ do

6.2.2 $z = MIS(i)$;

6.2.3 if z £ K then $T(k) = z$; $k = k + 1$; else prune z; endif

6.2.4 endfor

6.2.5 endif

6.3. if $l < B$ then

6.3.1 $i_0 = 0$;

6.3.2 if $K = A*B$ then $T(k) = MIS(i_0) + BR(0)$; $k = k + 1$; $i_0 = i_0 + 1$; endif

6.3.3 for $i = i_0, i_0 + 1, i_0 + 2, \ldots, A - 1$ do

6.3.4 $z = MIS(i) + l$;

6.3.5 if z £ K then $T(k) = z$; $k = k + 1$; else prune z; endif

6.3.6 endfor

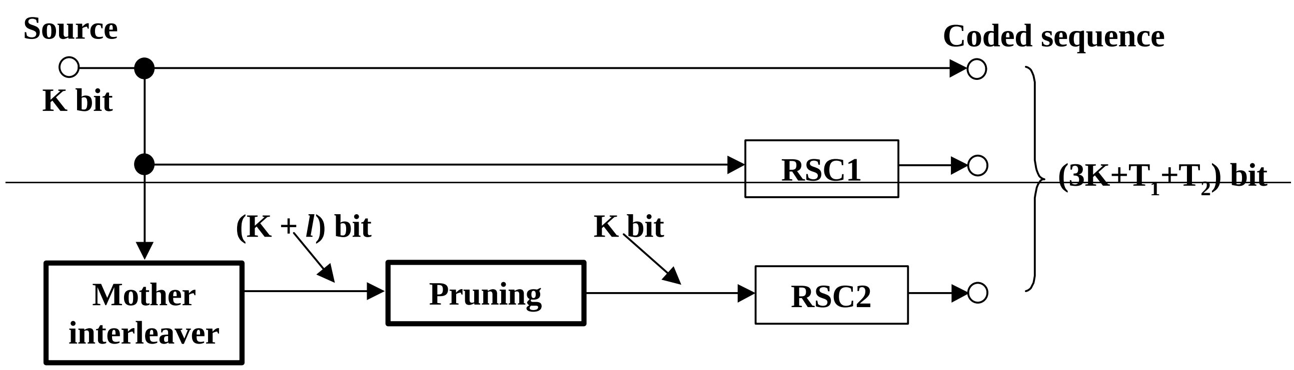
6.3.7 endif

The total number of pruned indexes is $A*B - K$.

Table 2: Table of prime l and associated primitive root m

l	m	l	m	l	m	l	m	l	m
7	3	47	5	101	2	157	5	223	3
11	2	53	2	103	5	163	2	227	2
13	2	59	2	107	2	167	5	229	6
17	3	61	2	109	6	173	2	233	3
19	2	67	2	113	3	179	2	239	7
23	5	71	7	127	3	181	2	241	7
29	2	73	5	131	2	191	19	251	6
31	3	79	3	137	3	193	5	257	3
37	2	83	2	139	2	197	2		
41	6	89	3	149	2	199	3		
43	3	97	5	151	6	211	2		

~~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 134 mother interleavers set. The generation scheme of mother interleaver is described in section 4.1.3.2.3.1. 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 4.1.3.2.3.2.~~



~~**Figure 5: Overall 8 State PCCC Turbo Coding**~~

~~4.1.3.2.3.1 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 5114 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 RxC rectangular matrix row by row.~~

~~**Second Stage:**~~

~~A. If C = *p*~~

~~(A-1) Select a primitive root~~ $\cancel{g_0}$ ~~from table 2.~~

~~(A-2) Construct the base sequence *c(i)* for intra-row permutation as:~~

$$\cancel{c(i) = [g_0 \times c(i-1)] \bmod p\,,\ i = 1,2,\ldots,(p\text{-}2).,\ c(0) = 1.}$$

~~(A-3) Select the minimum prime integer set~~ $\cancel{\{q_j\}\ (j=1,2,\ldots R\text{-}1)}$ ~~such that~~

$\cancel{\text{g.c.d}\{q_j, p\text{-}1\} = 1}$

$\cancel{q_j > 6}$

$\cancel{q_j > q_{(j\text{-}1)}}$

~~where g.c.d. is greatest common divider. And~~ $\cancel{q_0 = 1}$.

~~(A-4) The set~~ $\cancel{\{q_j\}}$ ~~is permuted to make a new set~~ $\cancel{\{p_j\}}$ ~~such that~~

$\cancel{p_{P(j)} = q_j,\ j = 0, 1, \ldots. R\text{-}1,}$

~~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:~~

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

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

~~B. If C = *p*+1~~

~~(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:~~

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

~~(B-6) If (K = C x R) then exchange $c_{R-1}(p)$ with $c_{R-1}(0)$.~~

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

~~C. If C = *p*-1~~

~~(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:~~

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

~~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 5114-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 2: Table of prime p and associated primitive root~~

~~p~~	~~g_0~~	~~p~~	~~g_0~~	~~p~~	~~g_0~~	~~p~~	~~g_0~~	~~p~~	~~g_0~~
~~17~~	~~3~~	~~59~~	~~2~~	~~103~~	~~5~~	~~157~~	~~5~~	~~211~~	~~2~~
~~19~~	~~2~~	~~61~~	~~2~~	~~107~~	~~2~~	~~163~~	~~2~~	~~223~~	~~3~~
~~23~~	~~5~~	~~67~~	~~2~~	~~109~~	~~6~~	~~167~~	~~5~~	~~227~~	~~2~~
~~29~~	~~2~~	~~71~~	~~7~~	~~113~~	~~3~~	~~173~~	~~2~~	~~229~~	~~6~~
~~31~~	~~3~~	~~73~~	~~5~~	~~127~~	~~3~~	~~179~~	~~2~~	~~233~~	~~3~~
~~37~~	~~2~~	~~79~~	~~3~~	~~131~~	~~2~~	~~181~~	~~2~~	~~239~~	~~7~~
~~41~~	~~6~~	~~83~~	~~2~~	~~137~~	~~3~~	~~191~~	~~19~~	~~241~~	~~7~~
~~43~~	~~3~~	~~89~~	~~3~~	~~139~~	~~2~~	~~193~~	~~5~~	~~251~~	~~6~~
~~47~~	~~5~~	~~97~~	~~5~~	~~149~~	~~2~~	~~197~~	~~2~~	~~257~~	~~3~~
~~53~~	~~2~~	~~101~~	~~2~~	~~151~~	~~6~~	~~199~~	~~3~~		

~~4.1.3.2.3.2 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:~~

$$\cancel{l = R \times C - K,}$$

~~where R is the row number and C is the column number defined in section 4.1.3.2.3.1.~~