TSG-RAN Working Group 1 meeting #9
Dresden, Germany
November 30 – December 3, 1999

TSGR1#9(99)i58

Agenda item:

Source: Ericsson

Title: CR 25.212-014: Update of channel coding sections

Document for: Decision

The sections in TS 25.212 dealing with channel coding needs to be updated, since they by mistake has received wrong section numbers. In addition, figure 3 showing the convolutional coders is a bit "messy", so it is proposed to replace that figure with a more clean version. Finally, the PCCC may be applied for any service, depending on terminal capabilities, regardless of QoS, so that statement is removed.

3GPP TSG RAN WG1 Meeting #9
Dresden, Germany, Nov 30 – Dec 3, 1999

Document ??99???
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**	**014**	Current Version:	3.0.0

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

For submission to:	TSG-RAN #6	for approval	X	strategic		(for SMG use only)
list expected approval meeting # here ↑		for information		non-strategic		

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:	Ericsson	**Date:**	1999-11-18
Subject:	Update of channel coding sections		
Work item:			

Category:	F	Correction		**Release:**	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	X		Release 99	X
					Release 00	

Reason for change:	The sections in TS 25.212 dealing with channel coding needs to be updated, since they by mistake has received wrong section numbers. In addition, figure 3 showing the convolutional coders is a bit "messy", so it is proposed to replace that figure with a more clean version. Finally, the PCCC may be applied for any service, depending on terminal capabilities, regardless of QoS, so that statement is removed.
Clauses affected:	Incorrectly numbered 4.1.3.1 and 4.1.3.2 with sub-clauses

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:	

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

~~ACS~~	~~Add, Compare, Select~~
ARQ	Automatic Repeat Request
BCH	Broadcast Channel
BER	Bit Error Rate
BLER	Block Error Rate
BS	Base Station
CCPCH	Common Control Physical Channel
CCTrCH	Coded Composite Transport Channel
CRC	Cyclic Redundancy Code
DCH	Dedicated Channel
DL	Downlink (Forward link)
DPCH	Dedicated Physical Channel
DPCCH	Dedicated Physical Control Channel
DPDCH	Dedicated Physical Data Channel
DS-CDMA	Direct-Sequence Code Division Multiple Access
DSCH	Downlink Shared Channel
DTX	Discontinuous Transmission
FACH	Forward Access Channel
FDD	Frequency Division Duplex
FER	Frame Error Rate
GF	Galois Field
MAC	Medium Access Control
Mcps	Mega Chip Per Second
MS	Mobile Station
OVSF	Orthogonal Variable Spreading Factor (codes)
PCCC	Parallel Concatenated Convolutional Code
PCH	Paging Channel
PRACH	Physical Random Access Channel
PhCH	Physical Channel
~~QoS~~	~~Quality of Service~~
RACH	Random Access Channel
RX	Receive
SCH	Synchronisation Channel
SF	Spreading Factor
SFN	System Frame Number
SIR	Signal-to-Interference Ratio
SNR	Signal to Noise Ratio
TF	Transport Format
TFC	Transport Format Combination
TFCI	Transport Format Combination Indicator
TPC	Transmit Power Control
TrCH	Transport Channel
TTI	Transmission Time Interval
TX	Transmit
UL	Uplink (Reverse link)

4.2.3 Channel coding

Code blocks are delivered to the channel coding block. They are denoted by $o_{ir1}, o_{ir2}, o_{ir3}, \ldots, o_{irK_i}$, where i is the TrCH number, r is the code block number, and K_i is the number of bits in each code block. The number of code blocks on TrCH i is denoted by C_i. After encoding the bits are denoted by $y_{ir1}, y_{ir2}, y_{ir3}, \ldots, y_{irY_i}$. The encoded blocks are serially multiplexed so that the block with lowest index r is output first from the channel coding block. The bits output are denoted by $c_{i1}, c_{i2}, c_{i3}, \ldots, c_{iE_i}$, where i is the TrCH number and $E_i = C_iY_i$. The output bits are defined by the following relations:

$$c_{ik} = y_{i1k} \quad k = 1, 2, \ldots, Y_i$$

$$c_{ik} = y_{i,2,(k-Y_i)} \quad k = Y_i + 1, Y_i + 2, \ldots, 2Y_i$$

$$c_{ik} = y_{i,3,(k-2Y_i)} \quad k = 2Y_i + 1, 2Y_i + 2, \ldots, 3Y_i$$

$$\ldots$$

$$c_{ik} = y_{i,C_i,(k-(C_i-1)Y_i)} \quad k = (C_i - 1)Y_i + 1, (C_i - 1)Y_i + 2, \ldots, C_iY_i$$

The relation between o_{irk} and y_{irk} and between K_i and Y_i is dependent on the channel coding scheme.

The following channel coding schemes can be applied to TrCHs:

- Convolutional coding
- Turbo coding
- No channel coding

The values of Y_i in connection with each coding scheme:

- Convolutional coding, ½ rate: $Y_i = 2*K_i + 16$; 1/3 rate: $Y_i = 3*K_i + 24$
- Turbo coding, 1/3 rate: $Y_i = 3*K_i + 12$
- No channel coding, $Y_i = K_i$

Table 1: Error Correction Coding Parameters

Transport channel type	Coding scheme	Coding rate
BCH	Convolutional code	1/2
PCH		
FACH		
RACH		
CPCH		1/3, 1/2 or no coding
DCH		
CPCH	Turbo Code	1/3 or no coding
DCH		

4.2~~1~~.3.1 Convolutional coding

~~4.1.3.1.1 Convolutional coder~~

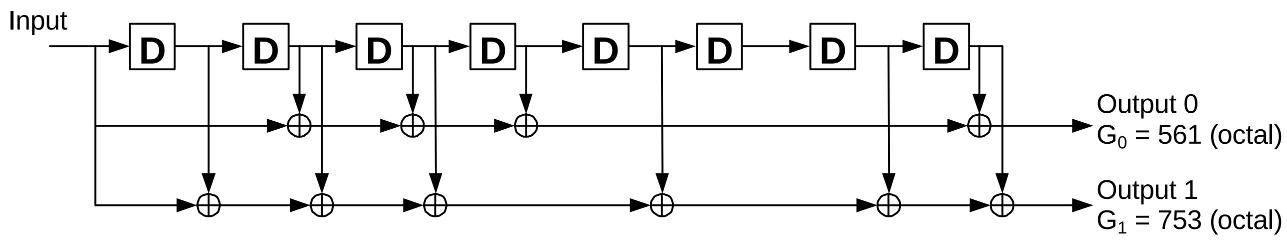
-Convolutional codes with ~~C~~constraint length ~~K=~~9 and ~~. C~~coding rates 1/3 and 1/2~~½~~ are defined.

The configuration of the convolutional coder is presented in figure 3.

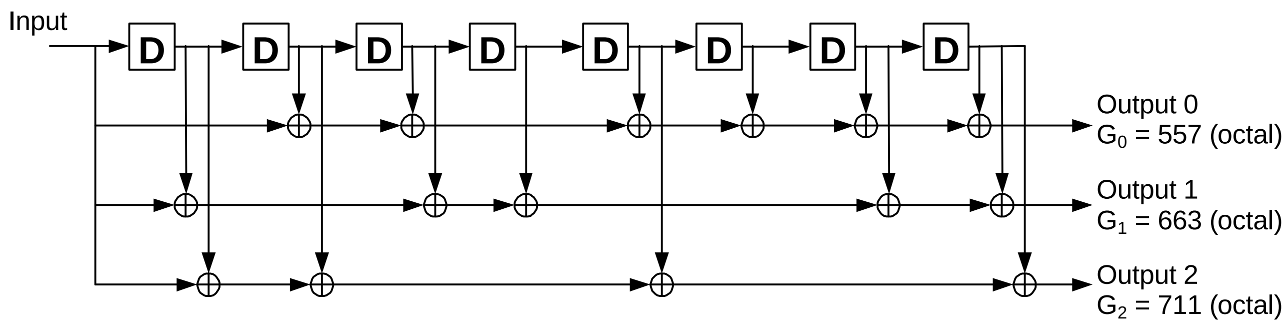
~~- The o~~Output from the rate 1/3 convolutional coder shall be done in the order output0, output1, output2, output0, output1, output 2, output 0, …,output2. ~~(When coding rate is 1/2, output is done up to output 1)~~Output from the rate 1/2 convolutional coder shall be done in the order output 0, output 1, output 0, output 1, output 0, …, output 1.

~~- K-1~~8 tail bits with binary ~~(~~value 0~~)~~ shall be added to the end of the code block before encoding.

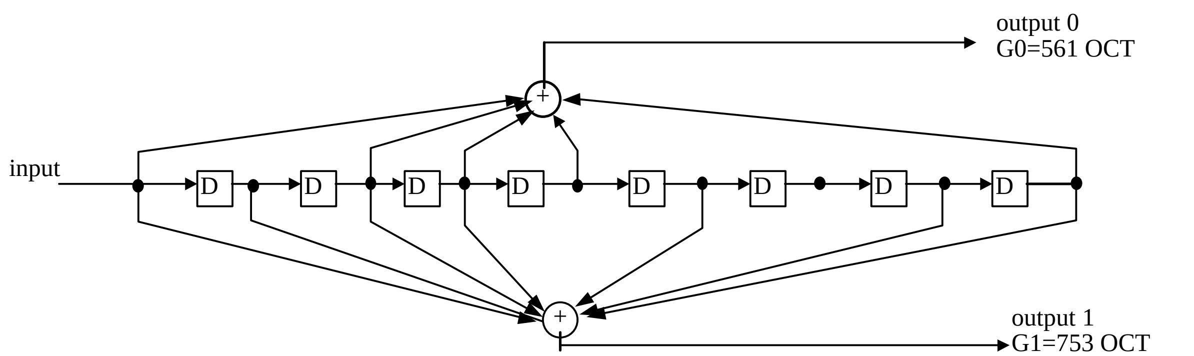
~~-~~ The initial value of the shift register of the coder shall be "all 0" when starting to encode the input bits.



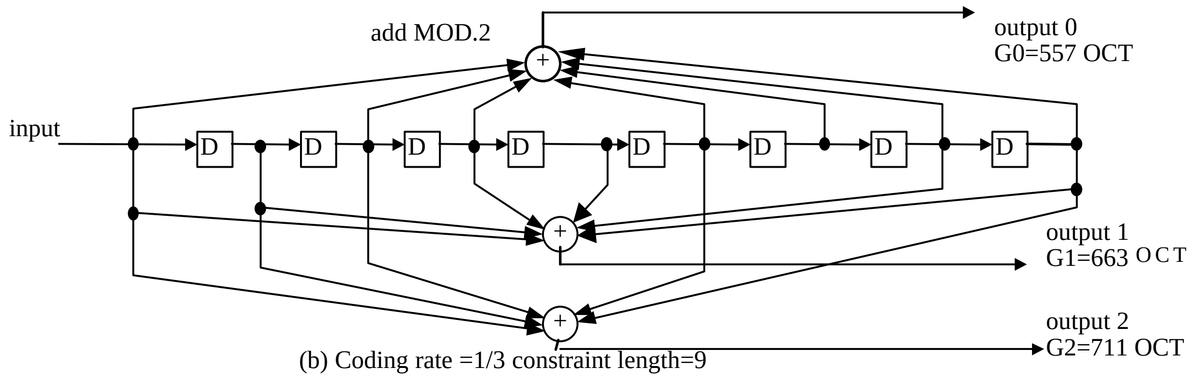
(a) Rate 1/2 convolutional coder



(b) Rate 1/3 convolutional coder



(a) Coding rate =1/2 constraint length=9



(b) Coding rate =1/3 constraint length=9

Figure 3: Rate 1/2 and rate 1/3 ~~C~~convolutional c~~C~~oders.

4.2~~1~~.3.2 Turbo coding

4.2~~1~~.3.2.1 Turbo coder

~~For data services requiring quality of service between 10^{-3} and 10^{-6} BER inclusive, p~~The turbo coding scheme is a parallel concatenated convolutional code (PCCC) with 8-state constituent encoders~~ is used~~.

The transfer function of the 8-state constituent code for PCCC is

$$G(D)=\left[1, \frac{n(D)}{d(D)}\right]$$

where,

$$d(D)=1+D^2+D^3$$

$$n(D)=1+D+D^3.$$

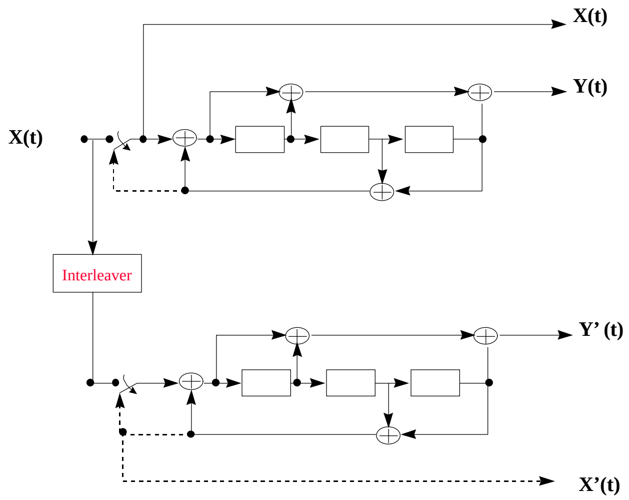


Figure 4: Structure of the 8 state PCCC encoder (dotted lines effective for trellis termination only)

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 1/3. 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),

4.2~~1~~.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~~1~~.3.2.3 Turbo code internal interleaver

Figure 5 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 134 mother interleavers set. The generation scheme of mother interleaver is described in section 4.2~~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 4.2~~1~~.3.2.3.2.

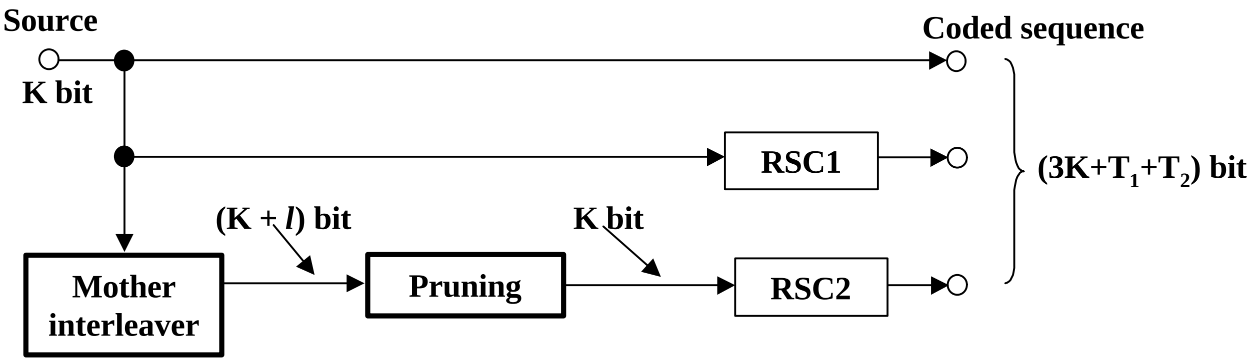


Figure 5: Overall 8 State PCCC Turbo Coding

4.2~~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 g_0 from table 2.

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

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

(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

$p_{P(j)} = q_j$, $j = 0, 1, \ldots$ 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:

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

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

<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, 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.

<u>Third Stage:</u>

(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 $\times$ C matrix.

Table 2: Table of prime p and associated primitive root

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
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		

4.2~~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:

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

where R is the row number and C is the column number defined in section 4.2~~1~~.3.2.3.1.