TSGR1#7(99)e18

TS 25.212 V2.2.0 (1999-09)

Technical Specification

3rd Generation Partnership Project (3GPP); Technical Specification Group (TSG) Radio Access Network (RAN); Working Group 1 (WG1); Multiplexing and channel coding (FDD)

Reference

<Workitem> (<Shortfilename>.PDF)

Keywords

<keyword[, keyword]>

3GPP

Postal address

Office address

Internet

secretariat@3gpp.org

Individual copies of this deliverable
can be downloaded from
http://www.3gpp.org

Contents

Intellectual Property Rights

<IPR notice shall be provided once correct notice is available within 3GPP>

Foreword

This Technical Specification has been produced by the 3GPP.

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of this TS, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version 3.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 Indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the specification.

1 Scope

This specification describes the documents being produced by the 3GPP TSG RAN WG1and first complete versions expected to be available by end of 1999. This specification describes the characteristics of the Layer 1 multiplexing and channel coding in the FDD mode of UTRA.

The 25.2series specifies Um point for the 3G mobile system. This series defines the minimum level of specifications required for basic connections in terms of mutual connectivity and compatibility.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

[1] 3GPP RAN TS 25.201: “Physical layer – General Description”

[2] 3GPP RAN TS 25.211: “Transport channels and physical channels (FDD)”

[3] 3GPP RAN TS 25.213: “Spreading and modulation (FDD)”

[4] 3GPP RAN TS 25.214: “Physical layer procedures (FDD)”

[5] 3GPP RAN TS 25.221: “Transport channels and physical channels (TDD)”

[6] 3GPP RAN TS 25.222: “Multiplexing and channel coding (TDD)”

[7] 3GPP RAN TS 25.223: “Spreading and modulation (TDD)”

[8] 3GPP RAN TS 25.224: “Physical layer procedures (TDD)”

[9] 3GPP RAN TS 25.231: “Measurements”

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the [following] terms and definitions [given in ... and the following] apply.

<defined term>: <definition>.

example: text used to clarify abstract rules by applying them literally.

3.2 Symbols

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

<symbol> <Explanation>

éxù round towards ¥, i.e. integer such that *x* £ *éxù* < *x*+1
ëxû round towards -¥, i.e. integer such that *x*-1 < *ëxû* £ *x*
çxç absolute value of *x*

Unless otherwise is explicitly stated when the symbol is used, the meaning of the following symbols is:

i TrCH number
j TFC number
k Bit number
l TF number
m Transport block number
n_i Radio frame number of TrCH i.
p PhCH number
r Code block number
I Number of TrCHs in a CCTrCH.
C_i Number of code blocks in one TTI of TrCH i.
F_i Number of radio frames in one TTI of TrCH i.
M_i Number of transport blocks in one TTI of TrCH i.
P Number of PhCHs used for one CCTrCH.
PL Puncturing Limit for the uplink. Signalled from higher layers
RM_i Rate Matching attribute for TrCH i. Signalled from higher layers.

3.3 Abbreviations

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

<ACRONYM> <Explanation>

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 Multiplexing, channel coding and interleaving

4.1 General

Data stream from/to MAC and higher layers (Transport block / Transport block set) is encoded/decoded to offer transport services over the radio transmission link. Channel coding scheme is a combination of error detection, error correcting, rate matching, interleaving and transport channels mapping onto/splitting from physical channels.

4.2 Transport-channel coding/multiplexing

Data arrives to the coding/multiplexing unit in form of transport block sets once every transmission time interval. The transmission time interval is transport-channel specific from the set {10 ms, 20 ms, 40 ms, 80 ms}.

The following coding/multiplexing steps can be identified:

- Add CRC to each transport block (see Section 4.2.1)
- Transport block concatenation and code block segmentation (see Section 4.2.2)
- Channel coding (see Section 4.2.3)
- Rate matching (see Section 4.2.7)
- Insertion of discontinuous transmission (DTX) indication bits (see Section 4.2.9)
- Interleaving (two steps, see Section 4.2.4 and 4.2.11)
- Radio frame segmentation (see Section 4.2.6)
- Multiplexing of transport channels (see Section 4.2.8)
- Physical channel segmentation (see Section 4.2.10)
- Mapping to physical channels (see Section 4.2.12)

The coding/multiplexing steps for uplink and downlink are shown in Figure 1 and Figure 2 respectively.

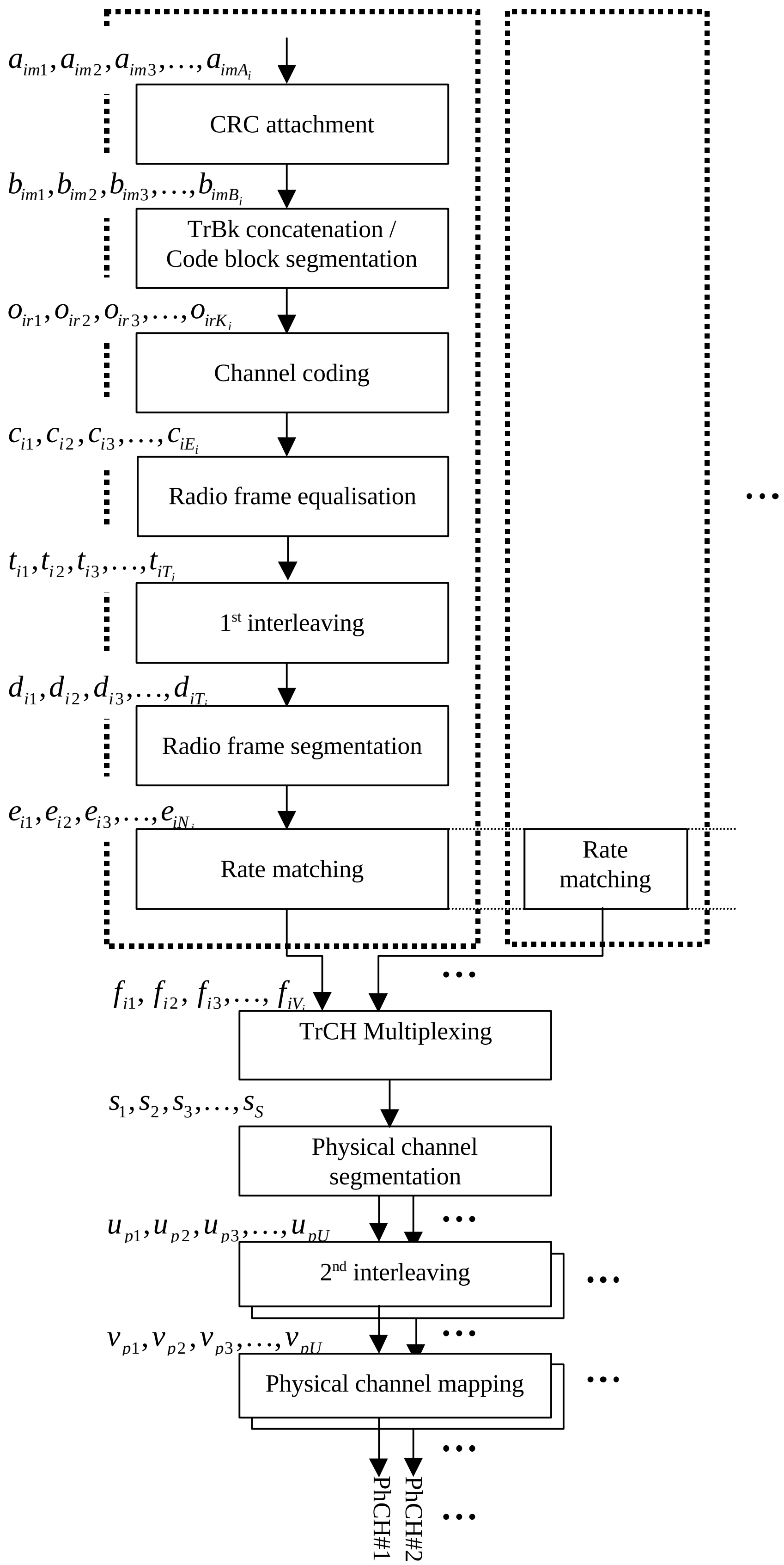


Figure 1: Transport channel multiplexing structure for uplink

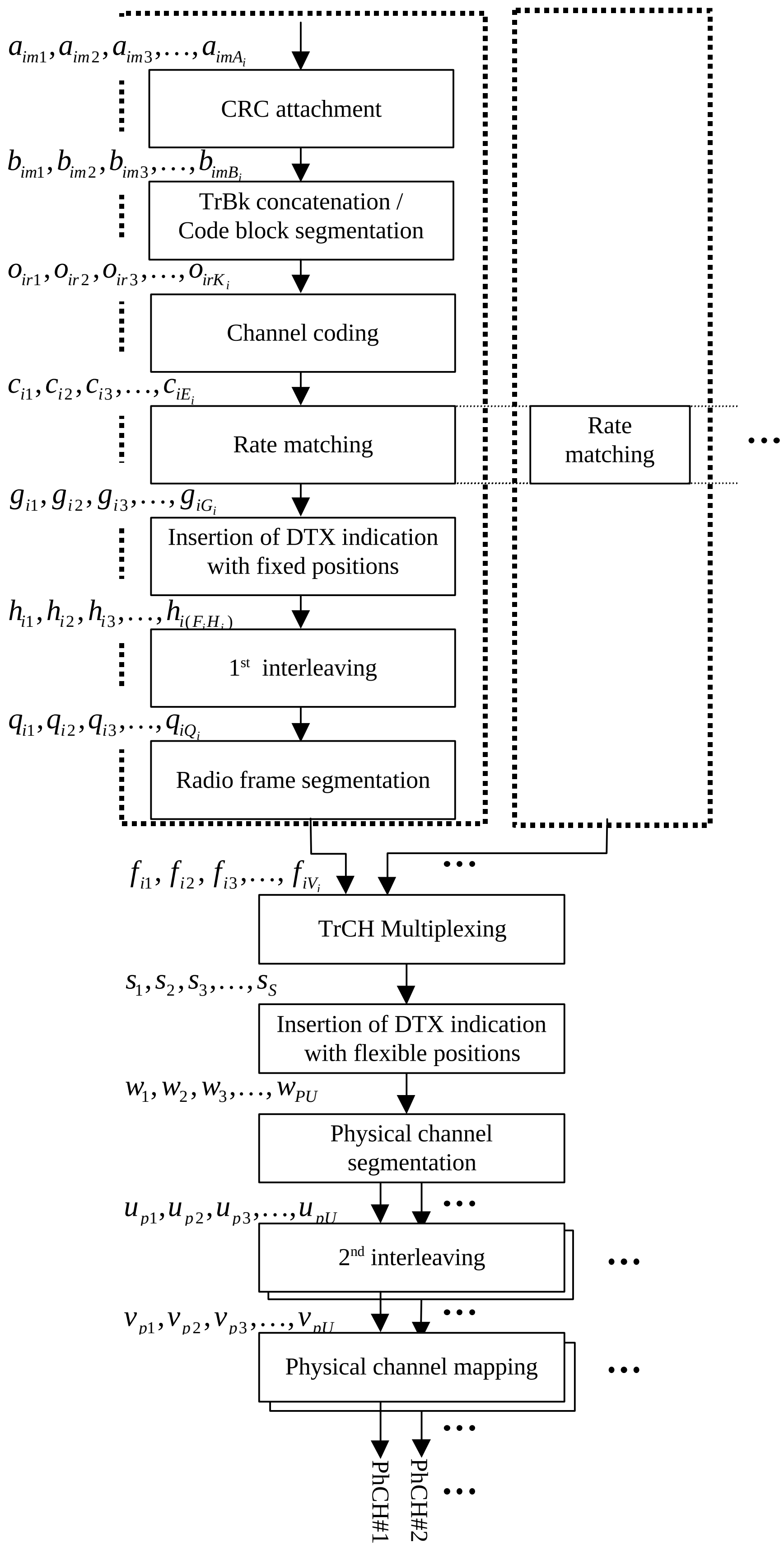


Figure 2: Transport channel multiplexing structure for downlink

The single output data stream from the TrCH multiplexing is denoted *Coded Composite Transport Channel (CCTrCH)*. A CCTrCH can be mapped to one or several physical channels.

4.2.1 Error detection

Error detection is provided on transport blocks through a Cyclic Redundancy Check. The CRC is 24, 16, 8 or 0 bits and it is signalled from higher layers what CRC length that should be used for each TrCH.

4.2.1.1 CRC Calculation

The entire transport block is used to calculate the CRC parity bits for each transport block. The parity bits are generated by one of the following cyclic generator polynomials:

$g_{CRC24}(D) = D^{24} + D^{23} + D^{6} + D^{5} + D + 1$

$g_{CRC16}(D) = D^{16} + D^{12} + D^{5} + 1$

$g_{CRC8}(D) = D^{8} + D^{7} + D^{4} + D^{3} + D + 1$

Denote the bits in a transport block delivered to layer 1 by $a_{im1}, a_{im2}, a_{im3}, \ldots, a_{imA_i}$, and the parity bits by $p_{im1}, p_{im2}, p_{im3}, \ldots, p_{imL_i}$. A_i is the length of a transport block of TrCH *i*, *m* is the transport block number, and L_i is 24, 16, 8, or 0 depending on what is signalled from higher layers.

The encoding is performed in a systematic form, which means that in GF(2), the polynomial

$$a_{im1}D^{A_i+23} + a_{im2}D^{A_i+22} + \ldots + a_{imA_i}D^{24} + p_{im1}D^{23} + p_{im2}D^{22} + \ldots + p_{im23}D^{1} + p_{im24}$$

yields a remainder equal to 0 when divided by $g_{CRC24}(D)$, polynomial

$$a_{im1}D^{A_i+15} + a_{im2}D^{A_i+14} + \ldots + a_{imA_i}D^{16} + p_{im1}D^{15} + p_{im2}D^{14} + \ldots + p_{im15}D^{1} + p_{im16}$$

yields a remainder equal to 0 when divided by $g_{CRC16}(D)$ and polynomial.

$$a_{im1}D^{A_i+7} + a_{im2}D^{A_i+6} + \ldots + a_{imA_i}D^{8} + p_{im1}D^{7} + p_{im2}D^{6} + \ldots + p_{im7}D^{1} + p_{im8}$$

yields a remainder equal to 0 when divided by $g_{CRC8}(D)$.

4.2.1.2 Relation between input and output of the Cyclic Redundancy Check

The bits after CRC attachment are denoted by $b_{im1}, b_{im2}, b_{im3}, \ldots, b_{imB_i}$, where $B_i=A_i+L_i$. The relation between a_{imk} and b_{imk} is:

$b_{imk} = a_{imk}$ $\quad k = 1, 2, 3, \ldots, A_i$

$b_{imk} = p_{im(L_i+1-(k-A_i))}$ $\quad k = A_i + 1, A_i + 2, A_i + 3, \ldots, A_i + L_i$

4.2.2 Transport block concatenation and code block segmentation

All transport blocks in a TTI are serially concatenated. If the number of bits in a TTI is larger than *Z*, then code block segmentation is performed after the concatenation of the transport blocks. The maximum size of the code blocks depend on if convolutional or turbo coding is used for the TrCH.

4.2.2.1 Concatenation of transport blocks

The bits input to the transport block concatenation are denoted by $b_{im1}, b_{im2}, b_{im3}, \ldots, b_{imB_i}$ where *i* is the TrCH number, *m* is the transport block number, and B_i is the number of bits in each block (including CRC). The number of transport blocks on TrCH *i* is denoted by M_i. The bits after concatenation are denoted by $x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iX_i}$, where *i* is the TrCH number and $X_i=M_iB_i$. They are defined by the following relations:

$x_{ik} = b_{i1k}$ k = 1, 2, …, B_i

$x_{ik} = b_{i,2,(k-B_i)}$ $k = B_i + 1, B_i + 2, \ldots, 2B_i$

$x_{ik} = b_{i,3,(k-2B_i)}$ $k = 2B_i + 1, 2B_i + 2, \ldots, 3B_i$

…

$x_{ik} = b_{i,M_i,(k-(M_i-1)B_i)}$ $k = (M_i - 1)B_i + 1, (M_i - 1)B_i + 2, \ldots, M_iB_i$

4.2.2.2 Code block segmentation

Segmentation of the bit sequence from transport block concatenation is performed if $X_i>Z$. The code blocks after segmentation are of the same size. The number of code blocks on TrCH i is denoted by C_i. If the number of bits input to the segmentation, X_i, is not a multiple of C_i, filler bits are added to the last block. The filler bits are transmitted and they are always set to 0. The maximum code block sizes are:

convolutional coding: $Z = 512 - K_{tail}$
turbo coding: $Z = 5120 - K_{tail}$

The bits output from code block segmentation 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.

Number of code blocks: C_i = éX_i / Zù
Number of bits in each code block: K_i = éX_i / C_iù
Number of filler bits: $Y_i = C_iK_i - X_i$

If $X_i \le Z$, then $o_{i1k} = x_{ik}$, and $K_i = X_i$.
If $X_i \ge Z$, then

$o_{i1k} = x_{ik}$ k = 1, 2, …, K_i

$o_{i2k} = x_{i,(k+K_i)}$ k = 1, 2, …, K_i

$o_{i3k} = x_{i,(k+2K_i)}$ k = 1, 2, …, K_i

…

$o_{iC_ik} = x_{i(k+(C_i-1)K_i)}$ $k = 1, 2, \ldots, K_i - Y_i$

$o_{iC_ik} = 0$ $k = (K_i - Y_i) + 1, (K_i - Y_i) + 2, \ldots, K_I$

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 $x_{ir1}, x_{ir2}, x_{ir3}, \ldots, x_{irX_i}$. 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 $E_i = C_iX_i$. The output bits are defined by the following relations:

$c_{ik} = x_{i1k}$ k = 1, 2, …, X_i

$c_{ik} = x_{i,2,(k-X_i)}$ $k = X_i + 1, X_i + 2, \ldots, 2X_i$

$c_{ik} = x_{i,3,(k-2X_i)}$ $k = 2X_i + 1, 2X_i + 2, \ldots, 3X_i$

…

$c_{ik} = x_{i,C_i,(k-(C_i-1)X_i)}$ $k = (C_i - 1)X_i + 1, (C_i - 1)X_i + 2, \ldots, C_iX_i$

The relation between o_{irk} and x_{irk} and between K_i and X_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

Table 1: Error Correction Coding Parameters

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

<Editor's note: Removal of 1/2 Turbo code rate is a working assumption.>

4.2.3.1 Convolutional coding

4.2.3.1.1 Convolutional coder

- Constraint length K=9. Coding rate 1/3 and ½.
- The configuration of the convolutional coder is presented in Figure 3.
- The output from the convolutional coder shall be done in the order output0, output1, output2, output0, output1, …,output2. (When coding rate is 1/2, output is done up to output 1).
- K-1 tail bits (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".

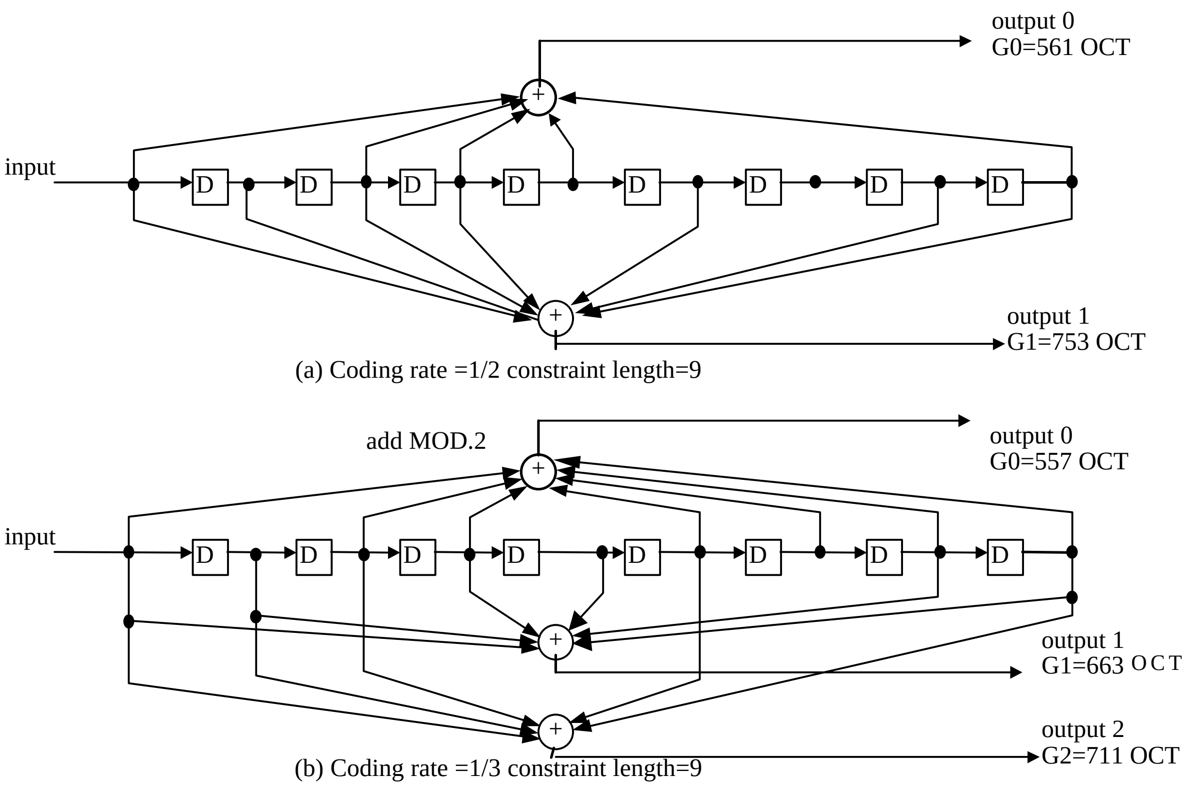


Figure 3: Convolutional Coder

4.2.3.2 Turbo coding

4.2.3.2.1 Turbo coder

For data services requiring quality of service between 10^{-3} and 10^{-6} BER inclusive, 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,

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

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

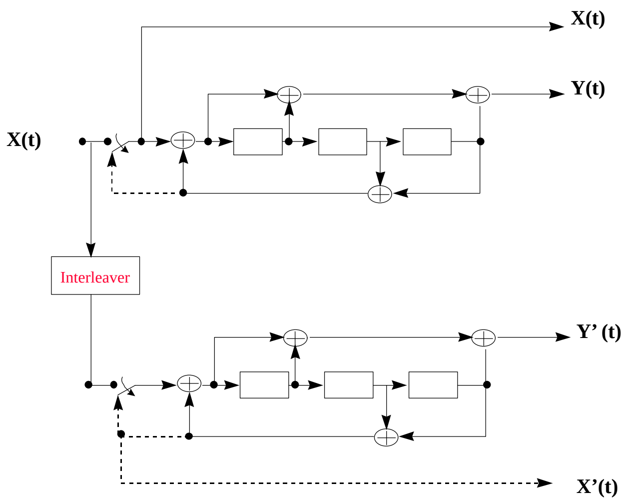


Figure 4: Structure of the 8 state PCCC encoder (dotted lines effective for trellis termination only)

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),

4.2.3.2.2 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 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 4.2.3.2.3.1. 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 4.2.3.2.3.2.

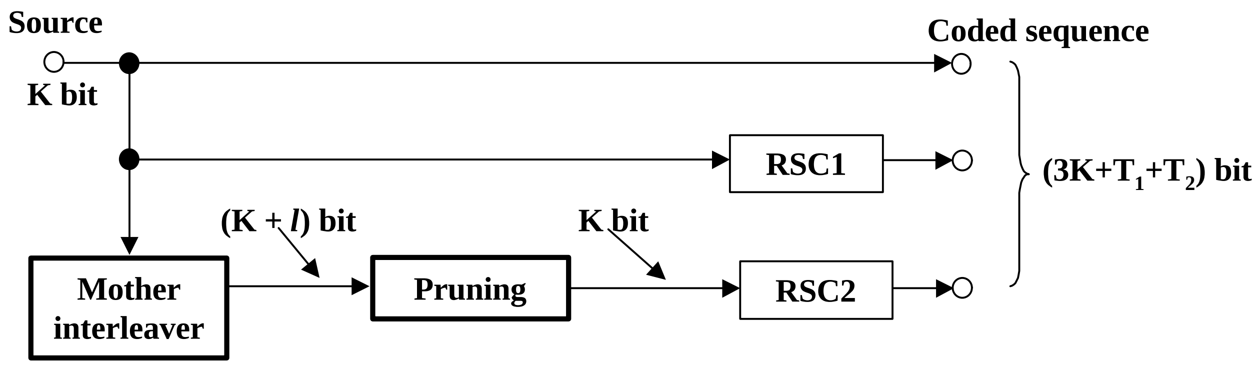


Figure 5: Overall 8 State PCCC Turbo Coding

4.2.3.2.3.1 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$\{q_j, p-1\}$ =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:

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

where $c_j(i)$ 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,…, (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,…, (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 $\times$ 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		

4.2.3.2.3.2 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 4.2.3.2.3.1.

4.2.4 Radio frame size equalisation

Radio frame size equalisation is padding the input bit sequence in order to ensure that the output can be segmented in F_i data segments of same size as described in section 4.2.6. Radio frame size equalisation is only performed in the UL (DL rate matching output block length is always an integer multiple of F_i)

The input bit sequence to the radio frame size equalisation is denoted by $c_{i1}, c_{i2}, c_{i3}, \ldots, c_{iE_i}$, where *i* is TrCH number and E_i the number of bits. The output bit sequence is denoted by $t_{i1}, t_{i2}, t_{i3}, \ldots, t_{iT_i}$, where T_i is the number of bits. The output bit sequence is derived as follows:

$t_{ik} = c_{ik},$ for k = 1… E_i and

$t_{ik} = \{0 \mid 1\}$ for k= E_i +1… T_i, if $E_i < T_i$

where

$T_i = F_i * N_i$ and

$N_i = \lfloor (E_i - 1)/F_i \rfloor + 1$ is the number of bits per segment after size equalisation.

4.2.5 1st interleaving

The 1st interleaving is a block interleaver with inter-column permutations. The input bit sequence to the 1st interleaver is denoted by $x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iX_i}$, where *i* is TrCH number and X_i the number of bits (at this stage X_i is assumed and guaranteed to be an integer multiple of TTI). The output bit sequence is derived as follows:

(1) Select the number of columns C_I from Table 3.

(2) Determine the number of rows R_I defined as

$R_I = X_i/C_I$

(3) Write the input bit sequence into the $R_I \times C_I$ rectangular matrix row by row starting with bit $x_{i,1}$ in the first column of the first row and ending with bit $x_{i,(R_I C_I)}$ in column C_I of row R_I:

$$\begin{bmatrix} x_{i1} & x_{i2} & x_{i3} & \dots & x_{iC_I} \\ x_{i,(C_I+1)} & x_{i,(C_I+2)} & x_{i,(C_I+3)} & \dots & x_{i,(2C_I)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ x_{i,((R_I-1)C_I+1)} & x_{i,((R_I-1)C_I+2)} & x_{i,((R_I-1)C_I+3)} & \dots & x_{i,(R_I C_I)} \end{bmatrix}$$

(4) Perform the inter-column permutation based on the pattern $\{P_1(j)\}$ (j=0,1, ..., C-1) shown in Table 3, where $P_1(j)$ is the original column position of the j-th permuted column. After permutation of the columns, the bits are denoted by y_{ik}:

$$\begin{bmatrix} y_{i1} & y_{i,(R_I+1)} & y_{i,(2R_I+1)} & \dots & y_{i,((C_I-1)R_I+1)} \\ y_{i2} & y_{i,(R_I+2)} & y_{i,(2R_I+2)} & \dots & y_{i,((C_I-1)R_I+2)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ y_{iR_I} & y_{i,(2R_I)} & y_{i,(3R_I)} & \dots & y_{i,(C_I R_I)} \end{bmatrix}$$

(5) Read the output bit sequence $y_{i1}, y_{i2}, y_{i3}, \dots, y_{i,(C_I R_I)}$ of the 1st interleaving column by column from the inter-column permuted $R_I \times C_I$ matrix. Bit $y_{i,1}$ corresponds to the first row of the first column and bit $y_{i,(R_I C_I)}$ corresponds to row R_I of column C_I.

Table 3

TTI	Number of columns C_I	Inter-column permutation patterns
10 ms	1	{0}
20 ms	2	{0,1}
40 ms	4	{0,2,1,3}
80 ms	8	{0,4,2,6,1,5,3,7}

4.2.5.1 Relation between input and output of 1st interleaving in uplink

The bits input to the 1st interleaving are denoted by $t_{i1}, t_{i2}, t_{i3}, \dots, t_{iT_i}$, where i is the TrCH number and E_i the number of bits. Hence, $x_{ik} = t_{ik}$ and $X_i = T_i$.

The bits output from the 1st interleaving are denoted by $d_{i1}, d_{i2}, d_{i3}, \dots, d_{iT_i}$, and $d_{ik} = y_{ik}$.

4.2.5.2 Relation between input and output of 1st interleaving in downlink

If fixed positions of the TrCHs in a radio frame is used then the bits input to the 1st interleaving are denoted by $h_{i1}, h_{i2}, h_{i3}, \dots, h_{i(F_i H_i)}$, where i is the TrCH number. Hence, $x_{ik} = h_{ik}$ and $X_i = F_i H_i$.

If flexible positions of the TrCHs in a radio frame is used then the bits input to the 1st interleaving are denoted by $g_{i1}, g_{i2}, g_{i3}, \dots, g_{iG_i}$, where i is the TrCH number. Hence, $x_{ik} = h_{ik}$ and $X_i = G_i$.

The bits output from the 1st interleaving are denoted by $q_{i1}, q_{i2}, q_{i3}, \dots, q_{iQ_i}$, where i is the TrCH number and Q_i is the number of bits. Hence, $q_{ik} = y_{ik}$, $Q_i = F_i H_i$ if fixed positions are used, and $Q_i = G_i$ if flexible positions are used.

4.2.6 Radio frame segmentation

When the transmission time interval is longer than 10 ms, the input bit sequence is segmented and mapped onto consecutive radio frames. Following rate matching in the DL and radio frame size equalisation in the UL the input bit sequence length is guaranteed to be an integer multiple of F_i.

The input bit sequence is denoted by $x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iX_i}$ where i is the TrCH number and X_i is the number bits. The *Fi* output bit sequences per TTI are denoted by $y_{i,n_i1}, y_{i,n_i2}, y_{i,n_i3}, \ldots, y_{i,n_iY_i}$ where n_i is the radio frame number in current TTI and Y_i is the number of bits per radio frame for TrCH i. The output sequences are defined as follows:

$y_{i,n_ik} = x_{i,((n_i-1)\cdot Y_i)+k}$, $n_i = 1\ldots F_i$, $j = 1\ldots Y_i$

where

$Y_i = (X_i / F_i)$ is the number of bits per segment,

x_{ik} is the k^th bit of the input bit sequence and

y_{i,n_ik} is the k^th bit of the output bit sequence corresponding to the n^th radio frame

The n_i -th segment is mapped to the n_i -th radio frame of the transmission time interval.

4.2.6.1 Relation between input and output of the radio frame segmentation block in uplink

The input bit sequence to the radio frame segmentation is denoted by $d_{i1}, d_{i2}, d_{i3}, \ldots, d_{iT_i}$, where i is the TrCH number and T_i the number of bits. Hence, $x_{ik} = d_{ik}$ and $X_i = T_i$.

The output bit sequence corresponding radio frame n_i is denoted by $e_{i1}, e_{i2}, e_{i3}, \ldots, e_{iN_i}$, where i is the TrCH number and N_i is the number of bits. Hence, $e_{i,k} = y_{i,n_ik}$ and $N_i = Y_i$.

4.2.6.2 Relation between input and output of the radio frame segmentation block in downlink

The bits input to the radio frame segmentation are denoted by $q_{i1}, q_{i2}, q_{i3}, \ldots, q_{iQ_i}$, where i is the TrCH number and Q_i the number of bits. Hence, $x_{ik} = q_{ik}$ and $X_i = Q_i$.

The output bit sequence corresponding to radio frame n_i is denoted by $f_{i1}, f_{i2}, f_{i3}, \ldots, f_{iV_i}$, where i is the TrCH number and V_i is the number of bits. Hence, $f_{i,k} = y_{i,n_ik}$ and $V_i = Y_i$.

4.2.7 Rate matching

<Editors' note: Rate matching for Turbo codes is a working assumption.>

Rate matching means that bits on a transport channel are repeated or punctured. Higher layers assign a rate-matching attribute for each transport channel. This attribute is semi-static and can only be changed through higher layer signalling. The rate-matching attribute is used when the number of bits to be repeated or punctured is calculated.

The number of bits on a transport channel can vary between different transmission time intervals. In the downlink the transmission is interrupted if the number of bits is lower than maximum. When the number of bits between different transmission time intervals in uplink is changed, bits are repeated or punctured to ensure that the total bit rate after second multiplexing is identical to the total channel bit rate of the allocated dedicated physical channels.

Notation used in Section 4.2.7 and subsections:

N_{ij}: For uplink: Number of bits in a radio frame before rate matching on TrCH i with transport format combination j .

For downlink : An intermediate calculation variable (not a integer but a multiple of 1/8).

N_{il}^{TTI} : Number of bits in a transmission time interval before rate matching on TrCH i with transport format l. Used in downlink only.

ΔN_{ij} : For uplink: If positive - number of bits that should be repeated in each radio frame on TrCH *i* with transport format combination *j*.

If negative - number of bits that should be punctured in each radio frame on TrCH *i* with transport format combination *j*.

For downlink : An intermediate calculation variable (not integer but a multiple of 1/8).

ΔN_{il}^{TTI} : If positive - number of bits to be repeated in each transmission time interval on TrCH *i* with transport format *j*.
If negative - number of bits to be punctured in each transmission time interval on TrCH *i* with transport format *j*.
Used in downlink only.

RM_i: Semi-static rate matching attribute for transport channel *i*. Signalled from higher layers.

PL: Puncturing limit for uplink. This value limits the amount of puncturing that can be applied in order to avoid multicode or to enable the use of a higher spreading factor. Signalled from higher layers.

$N_{data,j}$: Total number of bits that are available for the CCTrCH in a radio frame with transport format combination *j*.

I: Number of TrCHs in the CCTrCH.

Z_{ij}: Intermediate calculation variable.

F_i: Number of radio frames in the transmission time interval of TrCH *i*.

n_i: Radio frame number in the transmission time interval of TrCH *i* ($0 \le n_i < F_i$).

q: Average puncturing distance. Used in uplink only.

$I_F(n_i)$: The inverse interleaving function of the 1st interleaver (note that the inverse interleaving function is identical to the interleaving function itself for the 1st interleaver). Used in uplink only.

$S(n_i)$: The shift of the puncturing pattern for radio frame n_i. Used in uplink only.

$TF_i(j)$: Transport format of TrCH *i* for the transport format combination *j*.

TFS(i) The set of transport format indexes *l* for TrCH *i*.

TFCS The set of transport format combination indexes *j*.

e_{ini} Initial value of variable *e* in the rate matching pattern determination algorithm of section 4.2.7.3.

X: Systematic bit in section 4.2.3.2.1.

Y: 1st parity bit (from the upper Turbo constituent encoder) in section 4.2.3.2.1.

Y': 2nd parity bit (from the lower Turbo constituent encoder) in section 4.2.3.2.1.

Note: Time index t in section 4.2.3.2.1 is omitted for simplify the rate matching description.

The * (star) notation is used to replace an index *x* when the indexed variable X_x does not depend on the index *x*. In the left wing of an assignment the meaning is that "$X_* = Y$" is equivalent to "**for all** <u>x</u> **do** $X_x = Y$". In the right wing of an assignment, the meaning is that "$Y = X_*$" is equivalent to "**take any** <u>x</u> **and do** $Y = X_x$"

The following relations, defined for all TFC *j*, are used when calculating the rate matching parameters:

$$Z_{0,j} = 0$$

$$Z_{ij} = \left\lfloor \frac{\sum_{m=1}^{i} RM_m \cdot N_{mj}}{\sum_{m=1}^{I} RM_m \cdot N_{mj}} \cdot N_{data,j} \right\rfloor$$ for all *i* = 1 .. *I, where ⌊ ⌋ means round downwards*

$$\Delta N_{ij} = Z_{ij} - Z_{i-1,j} - N_{ij}$$ for all *i* = 1 .. *I*

4.2.7.1 Determination of rate matching parameters in uplink

In uplink puncturing can be used to avoid multicode or to enable the use of a higher spreading factor when this is needed because the UE does not support SF down to 4. The maximum amount of puncturing that can be applied is signalled from higher layers and denoted by PL. The number of available bits in the radio frames for all possible spreading factors is given in [2]. Denote these values by N_{256}, N_{128}, N_{64}, N_{32}, N_{16}, N_{8}, and N_{4}, where the index refers to the spreading factor. The possible values of N_{data} then are $\{ N_{256}, N_{128}, N_{64}, N_{32}, N_{16}, N_{8}, N_{4}, 2N_{4}, 3N_{4}, 4N_{4}, 5N_{4}, 6N_{4}\}$.Depending on the UE capabilities, the supported set of N_{data} , denoted SET0, can be a subset of $\{ N_{256}, N_{128}, N_{64}, N_{32}, N_{16}, N_{8}, N_{4}, 2N_{4}, 3N_{4}, 4N_{4}, 5N_{4}, 6N_{4}\}$. $N_{data,\,j}$ for the transport format combination j is determined by executing the following algorithm:

SET1 = { N_{data} in SET0 such that $N_{data} - \sum_{x=1}^{I} \frac{RM_{x,}}{\min_{1\le y\le I}\{RM_y\}} \cdot N_{x,j}$ is non negative }

If the smallest element of SET1 requires just one PhCH then

$N_{data,j}$ = min SET1

else

SET2 = { N_{data} in SET0 such that $N_{data} - PL \cdot \sum_{x=1}^{I} \frac{RM_{x}}{\min_{1\le y\le I}\{RM_y\}} \cdot N_{x,j}$ is non negative }

Sort SET2 in ascending order

N_{data} = min SET2

While N_{data} is not the max of SET2 and the follower of N_{data} requires no additional PhCH do

N_{data} = follower of N_{data} in SET2

End while

$N_{data,j} = N_{data}$

End if

The number of bits to be repeated or punctured, DN_{ij}, within one radio frame for each TrCH i is calculated with the relations given in Section 4.2.7 for all possible transport format combinations j and selected every radio frame.

Additionally, for determining e_{ini}, the following parameters are needed:

For convolutonal codes,

$a=2$ for the rate matching algorithm in section 4.2.7.3.

$q=$ ëN_{ij}/(ôDN_{ij}ô)û

if q is even

then $q' = q - gcd(q, F_i)/F_i$ -- where $gcd(q, F_i)$ means greatest common divisor of q and F_i

-- note that q' is not an integer, but a multiple of 1/8

else

$q' = q$

endif

for x = 0 to F_i-1

$S(I_F$ (é$x*q'$ù mod F_i)) = (é$x*q'$ù div F_i)

end for

For each radio frame, the rate-matching pattern is calculated with the algorithm in Section 4.2.7.3, where :

$DN = DN_{i,j}$

$N = N_{i,j}$., and

$e_{ini} = (a \cdot S(n_i) \cdot |DN| + N) \bmod a \cdot N$, if $e_{ini} = 0$ then $e_{ini} = a \cdot N$.

For turbo codes, if repetition is to be performed, such as $DN_{i,j} > 0$, parameters for turbo codes are the same as parameter for convolutional codes. If puncturing is to be performed, parameters are as follows.

a=2 for *Y* parity sequence, and

a=1 for *Y'* parity sequence.

For each radio frame, the rate-matching pattern is calculated with the algorithm in section 4.2.7.3, where:

$$DN = \begin{cases} \lfloor DN_{i,j} / 2 \rfloor \text{for Y sequence} \\ \lceil DN_{i,j} / 2 \rceil \text{for Y' sequence} \end{cases}$$

$N = ëN_{i,j}/3û$,

$q = ëN/|DN|\, û$

if($q \leq 2$)

  for x=0 to F_i-1

    if(Y sequence)

      $S[I_F[(3x+1) \bmod F_i]] = x \bmod 2;$

    if(Y' sequence)

        $S[I_F\, [(3x+2) \bmod F_i]] = x \bmod 2;$

  end for

else

  if q is even

    then $q' = q - gcd(q, F_i)/ F_i$ -- where $gcd\,(q, F_i)$ means greatest common divisor of q and F_i
    -- note that q' is not an integer, but a multiple of 1/8

  else $q' = q$

  endif

  for x=0 to F_i -1

    $r = é\, x*q'ù \bmod F_i;$

    if(Y sequence)

      $S[I_F[(3r+1) \bmod F_i]] = é\, x*q'ù \text{ div } F_i;$

    if(Y' sequence)

      $S[I_F[(3r+2) \bmod F_i\,]] = é\, x*q'ù \text{ div } F_i;$

  endfor

endif

$e_{ini} = (a \cdot S(n_i) \cdot |DN| + N) \bmod a \cdot N$, *if* $e_{ini} = 0$ *then* $e_{ini} = a \cdot N$.

4.2.7.2 Determination of rate matching parameters in downlink

For downlink $N_{data,j}$ does not depend on the transport format combination j. $N_{data,*}$ is given by the channelization code(s) assigned by higher layers.

4.2.7.2.1 Determination of rate matching parameters for fixed positions of TrCHs

First an intermediate calculation variable $N_{i,*}$ is calculated for all transport channels i by the following formula :

$$N_{i,*} = \frac{1}{F_i} \cdot \max_{l \in TFS(i)} N_{i,l}^{TTI}$$

The computation of the $\Delta N_{i,l}^{TTI}$ parameters is then performed in for all TrCH *i* and all TF *l* by the following formula, where $\Delta N_{i,*}$ is derived from $N_{i,*}$ by the formula given at section 4.2.7:

$$\Delta N_{i,*}^{TTI} = F_i \cdot \Delta N_{i,*}$$

Note : the order in which the transport format combinations are checked does not change the final result.

For each transmission time interval of TrCH *i* with TF *l*, the rate-matching pattern is calculated with the algorithm in Section 0. The following parameters are used as input:

For convolutional codes,

$$\Delta N = \Delta N_{i,*}^{TTI}$$

$$N = N_{il}^{TTI}$$

$$e_{ini} = \max_{l \in TFS(i)} N_{il}^{TTI}$$

a=2 for the rate matching algorithm.

For turbo codes, if repetition is to be performed, such as *DN>0*, parameters for turbo codes are the same as parameter for convolutional codes. If puncturing is to be performed, parameters are as follows.

a=2 for *Y* sequence,

a=1 for *Y'* sequence.

Systematic bit should not be punctured.

$$DN = \begin{cases} \left\lfloor DN_{i,*}^{TTI} / 2 \right\rfloor \text{for Y sequence} \\ \left\lceil DN_{i,*}^{TTI} / 2 \right\rceil \text{for Y' sequence} \end{cases}$$

$$N = \left\lfloor N_{il}^{TTL} / 3 \right\rfloor$$

$$\max_{l \in TFS(i)} \left\lfloor N_{il}^{TTL} / 3 \right\rfloor$$

$$e_{ini} = \max_{\in \ (\ )} \left\lfloor N^{TTL} / 3 \right\rfloor \bmod a \times \max_{l \in TFS(i)} \left\lfloor N_{il}^{TTL} / 3 \right\rfloor.$$

4.2.7.2.2 Determination of rate matching parameters for flexible positions of TrCHs

First an intermediate calculation variable N_{ij} is calculated for all transport channels *i* and all transport format combinations *j* by the following formula :

$$N_{i,j} = \frac{1}{F_i} \cdot N_{i,TF_i(j)}^{TTI}$$

Then rate matching ratios RF_i are calculated for each the transport channel *i* in order to minimise the number of DTX bits when the bit rate of the CCTrCH is maximum. The RF_i ratios are defined by the following formula :

$$RF_i = \frac{N_{data,*}}{\max_{j \in TFCS} \sum_{i=1}^{i=I} \left(RM_i \cdot N_{i,j} \right)} \cdot RM_i$$

The computation of $\Delta N_{i,l}^{TTI}$ parameters is then performed in two phases. In a first phase, tentative temporary values of $\Delta N_{i,l}^{TTI}$ are computed, and in the second phase they are checked and corrected. The first phase, by use of the RF_i ratios, ensures that the number of DTX indication bits inserted is minimum when the CCTrCH bit rate is maximum, but it does not ensure that the maximum CCTrCH bit rate is not greater than $N_{data,*}$. per 10ms. The latter condition is ensured through the checking and possible corrections carried out in the second phase.

At the end of the second phase, the latest value of $\Delta N_{i,l}^{TTI}$ is the definitive value.

The first phase defines the tentative temporary $\Delta N_{i,l}^{TTI}$ for all transport channel *i* and any of its transport format *l* by use of the following formula :

$$\Delta N_{i,l}^{TTI} = F_i \cdot \left\lceil \frac{RF_i \cdot N_{i,l}^{TTI}}{F_i} \right\rceil - N_{i,l}^{TTI}$$

The second phase is defined by the following algorithm :

for all j in *TFCS* do -- for all TFC

$$D = \sum_{i=1}^{i=I} \frac{N_{i,TF_i(j)}^{TTI} + \Delta N_{i,TF_i(j)}^{TTI}}{F_i}$$ -- CCTrCH bit rate (bits per 10ms) for TFC *l*

if $D > N_{data,*}$ then

for *i* = 1 to *I* do -- for all TrCH

$\Delta N = F_i \cdot \Delta N_{i,j}$ -- $\Delta N_{i,j}$ is derived from $N_{i,j}$ by the formula given at section 4.2.7

if $\Delta N_{i,TF_i(j)}^{TTI} > \Delta N$ then

$\Delta N_{i,TF_i(j)}^{TTI} = \Delta N$

end-if

end-for

end-if

end-for

Note : the order in which the transport format combinations are checked does not change the final result.

For each transmission time interval of TrCH *i* with TF *l*, the rate-matching pattern is calculated with the algorithm in Section 4.2.7.3. The following parameters are used as input:

For convolutional codes,

$\Delta N = \Delta N_{il}^{TTI}$

$N = N_{il}^{TTI}$

$e_{ini} = N_{il}^{TTI}$

a=2 for the rate matching algorithm.

For turbo codes, if repetition is to be performed, such as *DN>0*, parameters for turbo codes are the same as parameter for convolutional codes. If puncturing is to be performed, parameters are as follows.

a=2 for *Y* sequence,

a=1 for *Y'* sequence.

Systematic bit should not be punctured.

$$DN = \begin{cases} \left\lfloor DN_{il}^{TTI} / 2 \right\rfloor \text{for Y sequence} \\ \left\lceil DN_{il}^{TTI} / 2 \right\rceil \text{for Y' sequence} \end{cases}$$

$N = \left\lfloor N_{il}^{TTI} / 3 \right\rfloor$,

$e_{ini} = N_{il}^{TTI} \bmod a \cdot N_{il}^{TTI}$,

4.2.6.1. Bit separation for rate matching

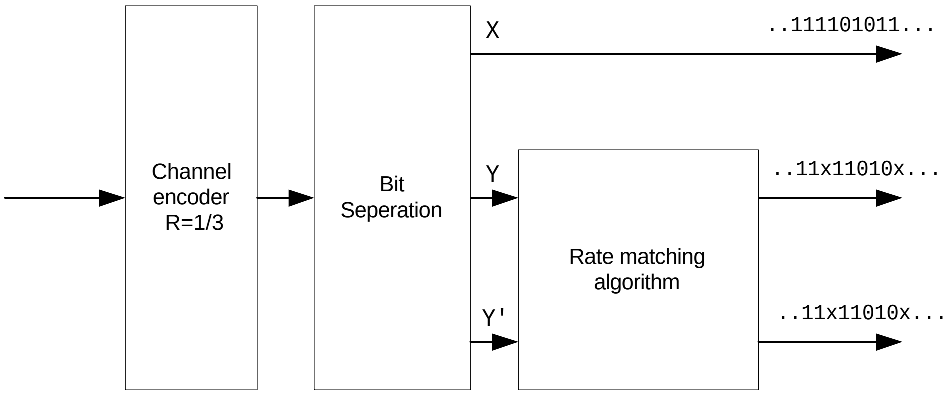


Figure 6: Overall rate matching block diagram before first interleaving where x denotes punctured bit.

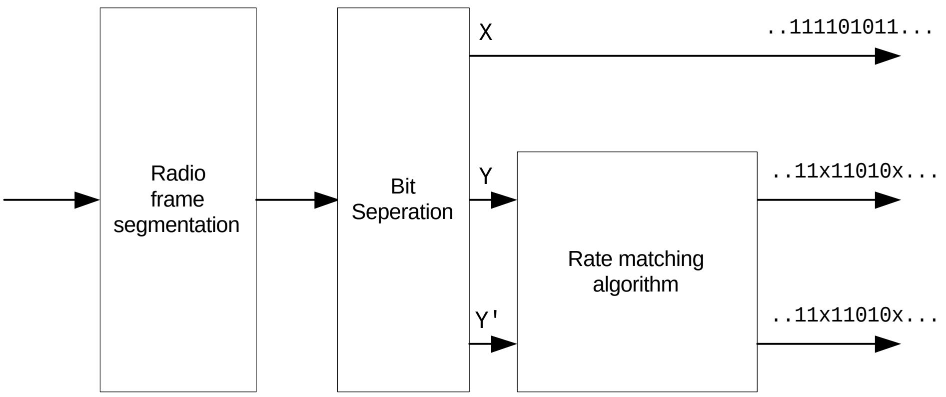


Figure 7: Overall rate matching block diagram after first interleaving where x denotes punctured bit.

Rate matching puncturing for Turbo codes in uplink is applied separately to *Y* and *Y'* sequences. No puncturing is applied to *X* sequence. Therefore, it is necessary to separate *X*, *Y*, and *Y'* sequences before rate matching is applied.

For uplink, there are two different alternation patterns in bit stream from radio frame segmentation according to the TTI of a TrCH as shown in Table 4.

Table 4: Alternation patterns of bits from radio frame segmentation in uplink

TTI (msec)	Alternation patterns
10, 40	… *X*,*Y*,*Y'*,…
20, 80	… *X*,*Y'*,*Y*,…

In addition, each radio frame of a TrCH starts with different initial parity type. Table 5 shows the initial parity type of each radio frame of a TrCH with TTI = {10, 20, 40, 80} msec.

Table 5: Initial parity type of radio frames of TrCH in uplink

TTI (msec)	**Radio frame indexes (n_i)**							
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
10	*X*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
20	*X*	*Y*	*NA*	*NA*	*NA*	*NA*	*NA*	*NA*
40	*X*	*Y'*	*Y*	*X*	*NA*	*NA*	*NA*	*NA*
80	*X*	*Y*	*Y'*	*X*	*Y*	*Y'*	*X*	*Y*

Table 4 and Table 5 defines a complete output bit pattern from radio frame segmentation.

Ex. 1. TTI = 40 msec, n_i = 2

Radio frame pattern: *Y, Y', X, Y, Y', X, Y, Y', X, …*

Ex. 2 TTI = 40 msec, n_i = 3

Radio frame pattern: *X, Y, Y', X, Y, Y', X, Y, Y', X, …*

Therefore, bit separation is achieved with the alternative selection of bits with the initial parity type and alternation pattern specified in Table 4 and Table 5 according to the TTI and n_i of a TrCH.

Rate matching puncturing for Turbo codes in downlink is applied separately to Y and Y's sequences. No puncturing is applied to X sequence. Therefore, it is necessary to separate X, Y, and Y' sequences before rate matching is applied.

For downlink, output bit sequence pattern from Turbo encoder is always X, Y, Y', X, Y, Y', …. Therefore, bit separation is achieved with the alternative selection of bits from Turbo encoder.

4.2.7.3 Rate matching pattern determination

Denote the bits before rate matching by:

$x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iN}$, where *i* is the TrCH number and *N* is the parameter given in section 4.2.7.2.

The rate matching rule is as follows:

```
if puncturing is to be performed
    y = -DN
    e = e_ini        -- initial error between current and desired puncturing ratio
    m = 1            -- index of current bit
    do while m <= N
        e = e – a * y        -- update error
        if e <= 0 then       -- check if bit number m should be punctured
            puncture bit x_i,m
            e = e + a*N  -- update error
        end if
        m = m + 1            -- next bit
    end do
else
    y = DN
    e = e_ini        -- initial error between current and desired puncturing ratio
```

```
        m = 1              -- index of current bit
        do while m <= N
            e = e – a * y          -- update error
            do while e <= 0        -- check if bit number m should be repeated
                repeat bit x_{i,m}
                e = e + a*N -- update error
            end do
            m = m + 1              -- next bit
        end do
    end if
```

A repeated bit is placed directly after the original one.

4.2.7.4 Relation between input and output of the rate matching block in uplink

The bits input to the rate matching are denoted by $e_{i1}, e_{i2}, e_{13}, \ldots, e_{iN_i}$, where *i* is the TrCH.

Hence, $x_{ik} = e_{ik}$ and $N = N_{ij} = N_i$.

The bits output from the rate matching are denoted by $f_{i1}, f_{i2}, f_{13}, \ldots, f_{iV_i}$, where *i* is the TrCH number and $V_i=N+DN = N_{ij}+DN_{ij}$.

Note that the transport format combination number *j* for simplicity has been left out in the bit numbering.

4.2.7.5 Relation between input and output of the rate matching block in downlink

The bits input to the rate matching are denoted by $c_{i1}, c_{i2}, c_{13}, \ldots, c_{iE_i}$, where *i* is the TrCH number and *l* the transport format number. Hence, $x_{ik} = e_{ik}$ and $N = N_{il}^{TTI} = E_i$.

The bits output from the rate matching are denoted by $g_{i1}, g_{i2}, g_{13}, \ldots, g_{iG_i}$, where *i* is the TrCH number and $G_i = N + \Delta N = N_{il}^{TTI} + \Delta N_{il}^{TTI}$.

Note that the transport format number *l* for simplicity has been left out in the bit numbering.

4.2.8 TrCH multiplexing

Every 10 ms, one radio frame from each TrCH is delivered to the TrCH multiplexing. These radio frames are serially multiplexed into a coded composite transport channel (CCTrCH).

The bits input to the TrCH multiplexing are denoted by $f_{i1}, f_{i2}, f_{i3}, \ldots, f_{iV_i}$, where *i* is the TrCH number and V_i is the number of bits in the radio frame of TrCH *i*. The number of TrCHs is denoted by *I*. The bits output from TrCH multiplexing are denoted by $s_1, s_2, s_3, \ldots, s_S$, where *S* is the number of bits, i.e. $S = \sum_i V_i$. The TrCH multiplexing is defined by the following relations:

$s_k = f_{1k}$ $\quad k = 1, 2, \ldots, V_1$

$s_k = f_{2,(k-V_1)}$ $\quad k = V_1+1, V_1+2, \ldots, V_1+V_2$

$s_k = f_{3,(k-(V_1+V_2))}$ $\quad k = (V_1+V_2)+1, (V_1+V_2)+2, \ldots, (V_1+V_2)+V_3$

$\ldots$

$s_k = f_{I,(k-(V_1+V_2+\ldots+V_{I-1}))}$ $\quad k = (V_1+V_2+\ldots+V_{I-1})+1, (V_1+V_2+\ldots+V_{I-1})+2, \ldots, (V_1+V_2+\ldots+V_{I-1})+V_I$

4.2.9 Insertion of discontinuous transmission (DTX) indication bits

In the downlink, DTX is used to fill up the radio frame with bits. The insertion point of DTX indication bits depends on whether fixed or flexible positions of the TrCHs in the radio frame are used. It is up to the UTRAN to decide for each CCTrCH whether fixed or flexible positions are used during the connection. DTX indication bits only indicate when the transmission should be turned off, they are not transmitted.

4.2.9.1 Insertion of DTX indication bits with fixed positions

This step of inserting DTX indication bits is used only if the positions of the TrCHs in the radio frame are fixed. With fixed position scheme a fixed number of bits is reserved for each TrCH in the radio frame.

The bits from rate matching are denoted by $g_{i1}, g_{i2}, g_{i3}, \ldots, g_{iG_i}$, where G_i is the number of bits in one TTI of TrCH i. Denote the number of bits reserved for one radio frame of TrCH i by H_i, i.e. the maximum number of bits in a radio frame for any transport format of TrCH i. The number of radio frames in a TTI of TrCH i is denoted by F_i. The bits output from the DTX insertion are denoted by $h_{i1}, h_{i2}, h_{i3}, \ldots, h_{i(F_i H_i)}$. Note that these bits are three valued. They are defined by the following relations:

$h_{ik} = g_{ik}$ $k = 1, 2, 3, \ldots, G_i$

$h_{ik} = d$ $k = G_i+1, G_i+2, G_i+3, \ldots, F_iH_i$

where DTX indication bits are denoted by d. Here $g_{ik} \in \{0, 1\}$ and $d \notin \{0, 1\}$.

4.2.9.2 Insertion of DTX indication bits with flexible positions

Note: Below, it is assumed that all physical channels belonging to the same CCTrCH use the same SF. Hence, U_p=U=constant.

This step of inserting DTX indication bits is used only if the positions of the TrCHs in the radio frame are flexible. The DTX indication bits shall be placed at the end of the radio frame. Note that the DTX will be distributed over all slots after 2nd interleaving.

The bits input to the DTX insertion block are denoted by $s_1, s_2, s_3, \ldots, s_S$,where S is the number of bits from TrCH multiplexing. The number of PhCHs is denoted by P and the number of bits in one radio frame, including DTX indication bits, for each PhCH by U.
The bits output from the DTX insertion block are denoted by $w_1, w_2, w_3, \ldots, w_{(PU)}$. Note that these bits are threevalued. They are defined by the following relations:

$w_k = s_k$ $k = 1, 2, 3, \ldots, S$

$w_k = d$ $k = S+1, S+2, S+3, \ldots, PU$

where DTX indication bits are denoted by d.. Here $s_k \in \{0,1\}$and $d \notin \{0,1\}$.

4.2.10 Physical channel segmentation

Note: Below, it is assumed that all physical channels belonging to the same CCTrCH use the same SF. Hence, U_p=U=constant.

When more than one PhCH is used, physical channel segmentation divides the bits among the different PhCHs. The bits input to the physical channel segmentation are denoted by $x_1, x_2, x_3, \ldots, x_Y$, where Y is the number of bits input to the physical channel segmentation block. The number of PhCHs is denoted by P.

The bits after physical channel segmentation are denoted $u_{p1}, u_{p2}, u_{p3}, \ldots, u_{pU}$, where p is PhCH number and U is the number of bits in one radio frame for each PhCH, i.e. $U = \frac{Y}{P}$. The relation between x_k and u_{pk} is given below.

Bits on first PhCH after physical channel segmentation:

$u_{1k} = x_k$ k = 1, 2 , …, U

Bits on second PhCH after physical channel segmentation:

$u_{2k} = x_{(k+U)}$ k = 1, 2 , …, U

…

Bits on the P^{th} PhCH after physical channel segmentation:

$u_{Pk} = x_{(k+(P-1)U)}$ k = 1, 2 , …, U

4.2.10.1 Relation between input and output of the physical segmentation block in uplink

The bits input to the physical segmentation are denoted by $s_1, s_2, s_3, \ldots, s_S$. Hence, $x_k = s_k$ and $Y = S$.

4.2.10.2 Relation between input and output of the physical segmentation block in downlink

If fixed positions of the TrCHs in a radio frame are used then the bits input to the physical segmentation are denoted by $s_1, s_2, s_3, \ldots, s_S$. Hence, $x_k = s_k$ and $Y = S$.
If flexible positions of the TrCHs in a radio frame are used then the bits input to the physical segmentation are denoted by $w_1, w_2, w_3, \ldots, w_{(PU)}$. Hence, $x_k = w_k$ and $Y = PU$.

4.2.11 2nd interleaving

The 2nd interleaving is a block interleaver with inter-column permutations. The bits input to the 2nd interleaver are denoted $u_{p1}, u_{p2}, u_{p3}, \ldots, u_{pU}$, where p is PhCH number and U is the number of bits in one radio frame for one PhCH.

(1) Set the number of columns C_2 = 30. The columns are numbered 0, 1, 2, …, C_2-1 from left to right.

(2) Determine the number of rows R_2 by finding minimum integer R_2 such that
U £ R_2C_2.

(3) The bits input to the 2nd interleaving are written into the $R_2 \times C_2$ rectangular matrix row by row.

$$\begin{bmatrix} u_{p1} & u_{p2} & u_{p3} & \cdots & u_{p30} \\ u_{p31} & u_{p32} & u_{p33} & \cdots & u_{p60} \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ u_{p,((R_2-1)30+1)} & u_{p,((R_2-1)30+2)} & u_{p,((R_2-1)30+3)} & \ldots u_{p,(R_2 30)} \end{bmatrix}$$

(4) Perform the inter-column permutation based on the pattern $\{P_2(j)\}$ $(j$ = 0, 1, ..., C_2-1$)$ that is shown in Table 6, where $P_2(j)$ is the original column position of the j-th permuted column. After permutation of the columns, the bits are denoted by y_{pk}.

$$\begin{bmatrix} y_{p1} & y_{p,(R_2+1)} & y_{p,(2R_2+1)} & \cdots y_{p,(29R_2+1)} \\ y_{p2} & y_{p,(R_2+2)} & y_{p,(2R_2+2)} & \cdots y_{p,(29R_2+2)} \\ \vdots & \vdots & \vdots & \ldots \quad \vdots \\ y_{pR_2} & y_{p,(2R_2)} & y_{p,(3R_2)} & \cdots \; y_{p,(30R_2)} \end{bmatrix}$$

(5) The output of the 2nd interleaving is the bit sequence read out column by column from the inter-column permuted $R_2 \times C_2$ matrix. The output is pruned by deleting bits that were not present in the input bit sequence, i.e. bits y_{pk} that corresponds to bits u_{pk} with $k>U$ are removed from the output. The bits after 2nd interleaving are denoted by

$v_{p1}, v_{p2}, \ldots, v_{pU}$, where v_{p1} corresponds to the bit y_{pk} with smallest index k after pruning, v_{p2} to the bit y_{pk} with second smallest index k after pruning, and so on.

Table 6

Number of column C_2	Inter-column permutation pattern
30	{0, 20, 10, 5, 15, 25, 3, 13, 23, 8, 18, 28, 1, 11, 21, 6, 16, 26, 4, 14, 24, 19, 9, 29, 12, 2, 7, 22, 27, 17}

4.2.12 Physical channel mapping

The PhCH for both uplink and downlink is defined in [2]. The bits input to the physical channel mapping are denoted by $v_{p1}, v_{p2}, \ldots, v_{pU}$, where p is the PhCH number and U is the number of bits in one radio frame for one PhCH. The bits v_{pk} are mapped to the PhCHs so that the bits for each PhCH are transmitted over the air in ascending order with respect to k.

4.2.12.1 Uplink

In uplink, the PhCHs used during a radio frame are either completely filled with bits that are transmitted over the air or not used at all.

4.2.12.2 Downlink

In downlink, the PhCHs do not need to be completely filled with bits that are transmitted over the air. Bits $v_{pk} \notin \{0, 1\}$ are not transmitted.

- For TrCHs not relying on TFCI for transport format detection (blind transport format detection), the positions of the transport channels within the radio frame should be fixed. In a limited number of cases, where there are a small number of transport format combinations, it is possible to allow flexible positions.

- For TrCHs relying on TFCI for transport format detection, higher layer signal whether the positions of the transport channels should be fixed or flexible.

4.2.13 Restrictions on different types of CCTrCHs

Restrictions on the different types of CCTrCHs are described in general terms in TS 25.302. In this section those restrictions are given with layer 1 notation.

4.2.13.1 Uplink Dedicated channel (DCH)

The maximum value of the number of TrCHs I in a CCTrCH, the maximum value of the number of transport blocks M_i on each transport channel, and the maximum value of the number of DPDCHs P are given from the UE capability class.

4.2.13.2 Random Access Channel (RACH)

- There can only be one TrCH in each RACH CCTrCH, i.e. I=1, $s_k = f_{1k}$ and $S = V_1$.
- The maximum value of the number of transport blocks M_1 on the transport channel is given from the UE capability class.
- The transmission time interval is always 10 ms, i.e. $e_{1k} = c_{1k}$ and $N_1 = E_1$.
- At initial RACH transmission the rate matching attribute has a predefined value.
- Only one PRACH is used, i.e. P=1, $u_{1k} = s_k$, and $U = S$.

4.2.13.3 Common Packet Channel (CPCH)

- The maximum value of the number of TrCHs I in a CCTrCH, the maximum value of the number of transport blocks M_i on each transport channel, and the maximum value of the number of DPDCHs P are given from the UE capability class.
- Note 1: The need to multiplex several CPCH transport channels is FFS (this note is taken from TS 25.302).
- Note 2: Only the data part of the CPCH can be mapped on multiple physical channels (this note is taken from TS 25.302).

4.2.13.4 Downlink Dedicated Channel (DCH)

The maximum value of the number of TrCHs I in a CCTrCH, the maximum value of the number of transport blocks M_i on each transport channel, and the maximum value of the number of DPDCHs P are given from the UE capability class.

4.2.13.5 Downlink Shared Channel (DSCH) associated with a DCH

- The spreading factor is indicated with the TFCI or with higher layer signalling on DCH.
- There can only be one TrCH in each DSCH CCTrCH, i.e. I=1, $s_k = f_{1k}$ and $S = V_1$.
- The maximum value of the number of transport blocks M_1 on the transport channel and the maximum value of the number of PDSCHs P are given from the UE capability class.

4.2.13.6 Broadcast channel (BCH)

- There can only be one TrCH in the BCH CCTrCH, i.e. I=1, $s_k = f_{1k}$, and $S = V_1$.
- There can only be one transport block in each transmission time interval, i.e. M_1 = 1.
- All transport format attributes have predefined values.
- Only one primary CCPCH is used, i.e. P=1.

4.2.13.7 Forward access and paging channels (FACH and PCH)

- The maximum value of the number of TrCHs I in a CCTrCH and the maximum value of the number of transport blocks M_i on each transport channel are given from the UE capability class.
- The transmission time interval for TrCHs of PCH type is always 10 ms.
- Only one secondary CCPCH is used per CCTrCH, i.e. P=1.

4.2.14 Multiplexing of different transport channels into one CCTrCH, and mapping of one CCTrCH onto physical channels

The following rules shall apply to the different transport channels which are part of the same CCTrCH:

1) Transport channels multiplexed into one CCTrCh should have co-ordinated timings in the sense that transport blocks arriving from higher layers on different transport channels of potentially different transmission time intervals shall have aligned transmission time instants as shown in Figure 8.

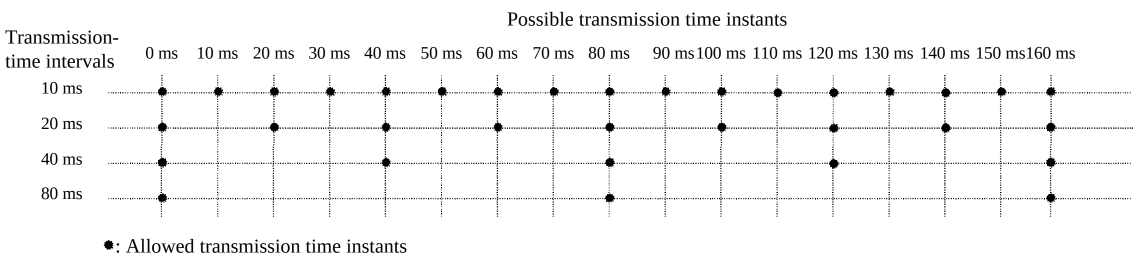


Figure 8: Possible transmission time instants regarding CCTrCH

2) Only transport channels with the same active set can be mapped onto the same CCTrCH.

3) Different CCTrCHs cannot be mapped onto the same PhCH.
4) One CCTrCH shall be mapped onto one or several PhCHs. These physical channels shall all have the same SF.
5) Dedicated Transport channels and common transport channels cannot be multiplexed into the same CCTrCH
6) For the common transport channels, only the FACH and PCH may belong to the same CCTrCH

There are hence two types of CCTrCH

1) CCTrCH of dedicated type, corresponding to the result of coding and multiplexing of one or several DCHs.
2) CCTrCH of common type, corresponding to the result of the coding and multiplexing of a common channel, RACH in the uplink, DSCH ,BCH, or FACH/PCH for the downlink.

4.2.14.1 Allowed CCTrCH combinations for one UE

4.2.14.1.1 Allowed CCTrCH combinations on the uplink

A maximum of one CCTrCH is allowed for one UE on the uplink. It can be either

1) one CCTrCH of dedicated type
2) one CCTrCH of common type

4.2.14.1.2 Allowed CCTrCH combinations on the downlink

The following CCTrCH combinations for one UE are allowed :

x CCTrCH of dedicated type + y CCTrCH of common type

The allowed combination of CCTrCHs of dedicated and common type are FFS.

Note 1 : There is only one DPCCH in the uplink, hence one TPC bits flow on the uplink to control possibly the different DPDCHs on the downlink, part of the same or several CCTrCHs.

Note 2 : There is only one DPCCH in the downlink, even with multiple CCTrCHs. With multiple CCTrCHs, the DPCCH is transmitted on one of the physical channels of that CCTrCH which has the smallest SF among the multiple CCTrCHs. Thus there is only one TPC command flow and only one TFCI word in downlink even with multiple CCTrCHs.

4.2.15 Transport format detection

Transport format detection can be performed both with and without Tansport Format Combination Indicator (TFCI). If a TFCI is transmitted, the receiver detects the transport format combination from the TFCI. When no TFCI is transmitted, so called blind transport format detection is used, i.e. the receiver side detects the transport format combination using some information, e.g. received power ratio of DPDCH to DPCCH, CRC check results.

For uplink, the blind transport format detection is an operator option. For downlink, the blind transport format detection can be applied with convolutional coding, the maximum number of different transport formats and maximum data rates allowed shall be specified.

4.2.15.1 Blind transport format detection

Examples of blind transport format detection methods are given in Annex A.

4.2.15.2 Explicit transport format detection based on TFCI

4.2.15.2.1 Transport format combination indicator (TFCI)

The Transport Format Combination Indicator (TFCI) informs the receiver of the transport format combination of the CCTrCHs. As soon as the TFCI is detected, the transport format combination, and hence the individual transport channels' transport formats are known, and decoding of the transport channels can be performed.

4.2.16 Coding procedure

4.2.16.1 SFN(System Frame Number)

- SFN indicates super frame synchronisation. It is broadcasted in BCH. (See TS 25.211)

<Editor's note: Length of SFN is FFS. It will be determined according to requirement from WG2and WG3>

- SFN is multiplexed with a BCH transport block (see Figure 9).
- SFN is applied CRC calculation and FEC with BCH transport block.

Figure 9: SFN multiplexing

4.3 Coding for layer 1 control

4.3.1 Coding of Transport-format-combination indicator (TFCI)

The number of TFCI bits is variable and is set at the beginning of the call via higher layer signalling. For improved TFCI detection reliability, in downlink, repetition is used by increasing the number of TFCI bits within a slot.

The TFCI bits are encoded using (30, 10) punctured sub-code of the second order Reed-Muller code. The coding procedure is as shown in Figure 10.

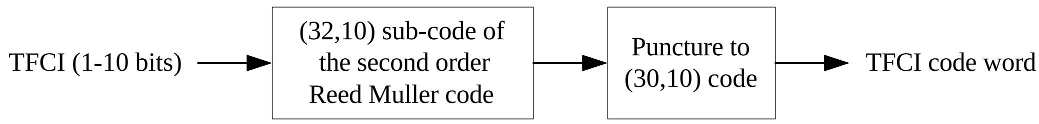


Figure 10: Channel coding of TFCI bits

If the TFCI consist of less than 10 bits, it is padded with zeros to 10 bits, by setting the most significant bits to zero. The receiver can use the information that not all 10 bits are used for the TFCI, thereby reducing the error rate in the TFCI decoder. The length of the TFCI code word is 30 bits. Thus there are 2 bits of (encoded) TFCI in every slot of the radio frame.

Firstly, TFCI is encoded by the (32,10) sub-code of second order Reed-Muller code. The code words of the (32,10) sub-code of second order Reed-Muller code are linear combination of 10 basis sequences: all 1's, 5 OVSF codes ($C_{32,1}$, $C_{32,2}$, $C_{32,4}$, $C_{32,8}$, $C_{32,16}$), and 4 masks (Mask1, Mask2, Mask3, Mask4). The 4 mask sequences are as following Table 7.

Table 7: Mask sequences

Mask 1	00101000011000111111000001110111
Mask 2	00000001110011010110110111000111
Mask 3	00001010111110010001101100101011
Mask 4	00011100001101110010111101010001

For information bits a_0 , a_1 , a_2 , a_3 , a_4 , a_5 , a_6 , a_7 , a_8 , a_9 (a_0 is LSB and a_9 is MSB), the encoder structure is as following Figure 11.

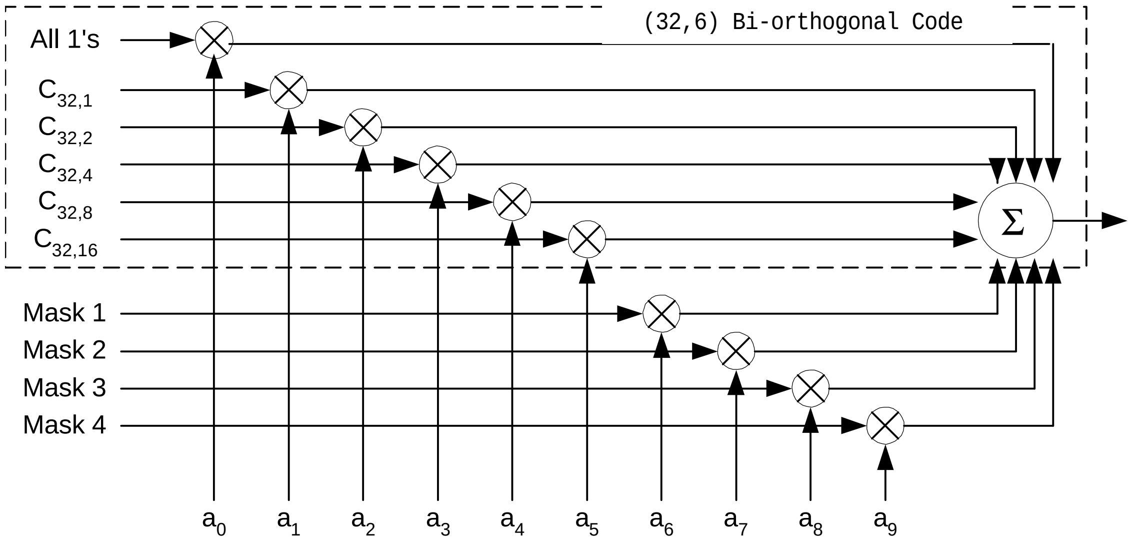


Figure 11: Encoder structure for (32,10) sub-code of second order Reed-Muller code

Then, the code words of the (32,10) sub-code of second order Reed-Muller code are punctured into length 30 by puncturing 1st and 17th bits.

In downlink, when the SF is lower then 128 the encoded and punctured TFCI code words are repeated four times yielding 8 encoded TFCI bits per slot. Mapping of repeated bits to slots is explained in section 4.3.3.

4.3.2 Operation of Transport-format-combination indicator (TFCI) in Split Mode

In the case of DCH in Split Mode, the UTRAN shall operate with as follows:

- If one of the links is associated with a DSCH, the TFCI code word may be split in such a way that the code word relevant for TFCI activity indication is not transmitted from every cell. The use of such a functionality shall be indicated by higher layer signalling.

TFCI information is encoded by biorthogonal (16, 5) block code. The code words of the biorthogonal (16, 5) code are from two mutually biorthogonal sets, $S_{C_{16}} = \{C_{16,0}, C_{16,1}, ..., C_{16,15}\}$ and its binary complement, $\overline{S}_{C_{16}} = \{\overline{C}_{16,0}, \overline{C}_{16,1}, ..., \overline{C}_{16,15}\}$. Code words of set $S_{C_{16}}$ are from the level 16 of the code three of OVSF codes defined in document TS 25.213. The mapping of information bits to code words is shown in the Table 8.

Table 8: Mapping of information bits to code words for biorthogonal (16, 5) code

Information bits	Code word
00000	$C_{16,0}$
00001	$\overline{C}_{16,0}$
00010	$C_{16,1}$
...	...
11101	$\overline{C}_{16,14}$
11110	$C_{16,15}$
11111	$\overline{C}_{16,15}$

Biorthogonal code words, $C_{16,i}$ and $\overline{C}_{16,i}$, are then punctured into length 15 by puncturing the 1st bit.

4.3.3 Mapping of TFCI words

4.3.3.1 Mapping of TFCI word

As only one code word for TFCI is needed no channel interleaving for the encoded bits are done. Instead, the bits of the code word are directly mapped to the slots of the radio frame as depicted in the Figure 12. Within a slot the more significant bit is transmitted before the less significant bit.

In compressed mode all the TFCI bits are reallocated into the remaining slots. However, the same principle of transmitting the most significant bit first is valid.

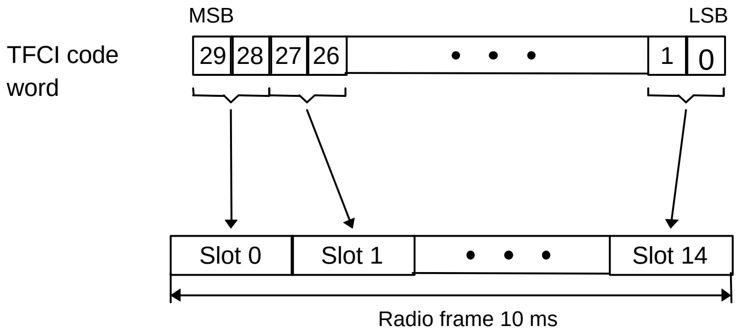


Figure 12: Mapping of TFCI code words to the slots of the radio frame

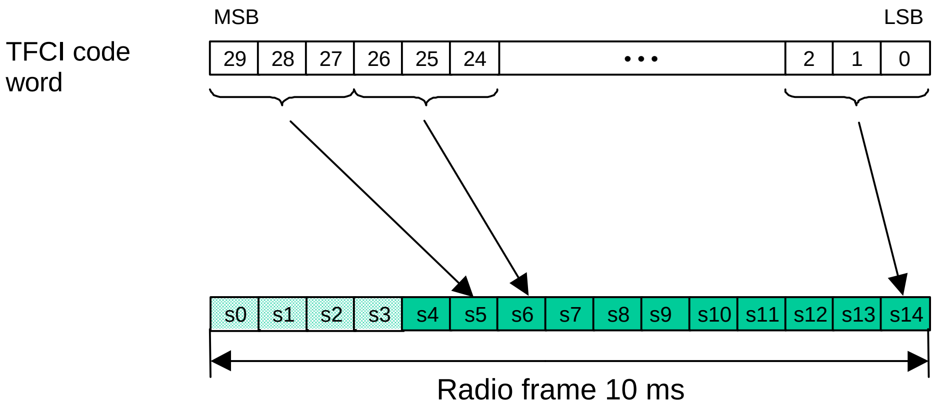


Figure 13: Mapping of TFCI code words to the slots of a compressed radio frame of 11 slots.

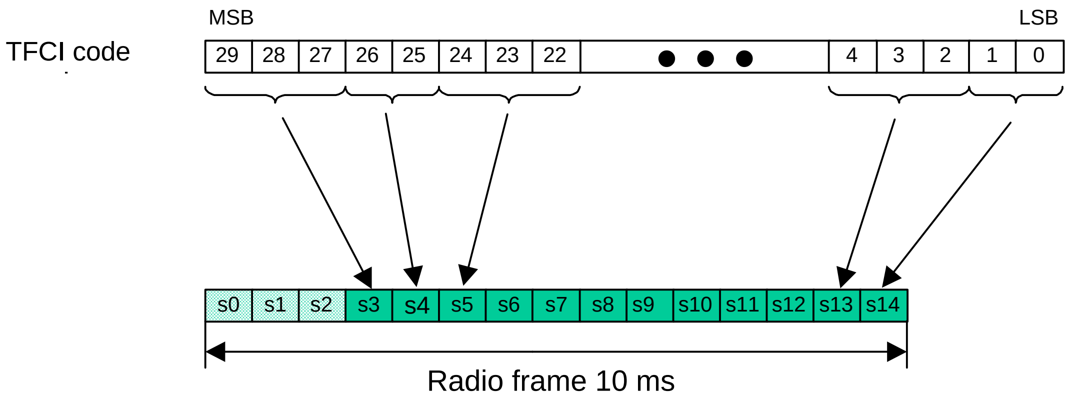


Figure 14: Mapping of TFCI code words to the slots of a compressed radio frame of 12 slots.

For downlink physical channels whose SF is lower than 128, bits of the TFCI code words are repeated and mapped to slots as shown in the Table 9. Code word bits are denoted as $b_{i,j}$, where subscript i, indicates bit position in the code word (i=29 is the MSB bit) and subscript j indicates bit repetition. In each slot transmission order of the bits is from left to right in the Table 9.

Table 9: Mapping order of repetition encoded TFCI code word bits into slots.

Slot	TFCI code word bits							
0	$b_{29,1}$	$b_{29,2}$	$b_{29,3}$	$b_{29,4}$	$b_{28,1}$	$b_{28,2}$	$b_{28,3}$	$b_{28,4}$
1	$b_{27,1}$	$b_{27,2}$	$b_{27,3}$	$b_{27,4}$	$b_{26,1}$	$b_{26,2}$	$b_{26,3}$	$b_{26,4}$
2	$b_{25,1}$	$b_{25,2}$	$b_{25,3}$	$b_{25,4}$	$b_{24,1}$	$b_{24,2}$	$b_{24,3}$	$b_{24,4}$
3	$b_{23,1}$	$b_{23,2}$	$b_{23,3}$	$b_{23,4}$	$b_{22,1}$	$b_{22,2}$	$b_{22,3}$	$b_{22,4}$
4	$b_{21,1}$	$b_{21,2}$	$b_{21,3}$	$b_{21,4}$	$b_{20,1}$	$b_{20,2}$	$b_{20,3}$	$b_{20,4}$
5	$b_{19,1}$	$b_{19,2}$	$b_{19,3}$	$b_{19,4}$	$b_{18,1}$	$b_{18,2}$	$b_{18,3}$	$b_{18,4}$
6	$b_{17,1}$	$b_{17,2}$	$b_{17,3}$	$b_{17,4}$	$b_{16,1}$	$b_{16,2}$	$b_{16,3}$	$b_{16,4}$
7	$b_{15,1}$	$b_{15,2}$	$b_{15,3}$	$b_{15,4}$	$b_{14,1}$	$b_{14,2}$	$b_{14,3}$	$b_{14,4}$
8	$b_{13,1}$	$b_{13,2}$	$b_{13,3}$	$b_{13,4}$	$b_{12,1}$	$b_{12,2}$	$b_{12,3}$	$b_{12,4}$
9	$b_{11,1}$	$b_{11,2}$	$b_{11,3}$	$b_{11,4}$	$b_{10,1}$	$b_{10,2}$	$b_{10,3}$	$b_{10,4}$
10	$b_{9,1}$	$b_{9,2}$	$b_{9,3}$	$b_{9,4}$	$b_{8,1}$	$b_{8,2}$	$b_{8,3}$	$b_{8,4}$
11	$b_{7,1}$	$b_{7,2}$	$b_{7,3}$	$b_{7,4}$	$b_{6,1}$	$b_{6,2}$	$b_{6,3}$	$b_{6,4}$
12	$b_{5,1}$	$b_{5,2}$	$b_{5,3}$	$b_{5,4}$	$b_{4,1}$	$b_{4,2}$	$b_{4,3}$	$b_{4,4}$
13	$b_{3,1}$	$b_{3,2}$	$b_{3,3}$	$b_{3,4}$	$b_{2,1}$	$b_{2,2}$	$b_{2,3}$	$b_{2,4}$
14	$b_{1,1}$	$b_{1,2}$	$b_{1,3}$	$b_{1,4}$	$b_{0,1}$	$b_{0,2}$	$b_{0,3}$	$b_{0,4}$

4.3.3.2 Mapping of TFCI word in Split Mode

After channel encoding of the two 5 bit TFCI words there are two code words of length 15 bits. They are mapped to DPCCH as shown in the Figure 15. Note that $b_{1,i}$ and $b_{2,i}$ denote the bit i of code word 1 and code word 2, respectively.

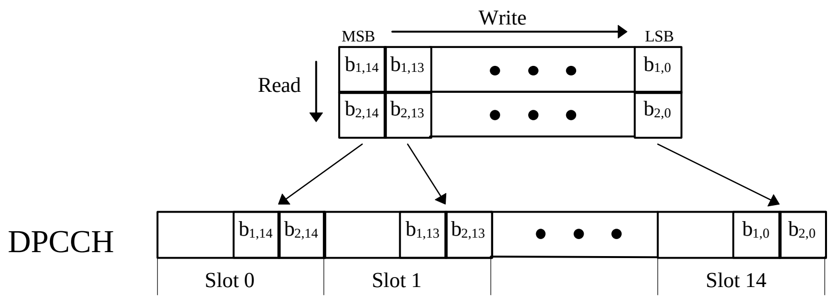


Figure 15: Mapping of TFCI code words to the slots of the radio frame in Split Mode

In compressed mode the mapping of TFCI bits takes place in a similar fashion as for (30,10) word in section 4.3.3.1. The order of bits is $b_{1,15}, b_{2,15}, b_{1,14}, b_{2,14}, \ldots b_{1,0}, b_{2,0}$.

For downlink physical channels whose SF is lower than 128, bits of the extended TFCI code words are repeated and mapped to slots as shown in the Table 10. Code word bits are denoted as $b^k_{i,j}$, where subscript k indicates the code word, subscript i indicates bit position in the code word (i=14 is the MSB bit) and subscript j indicates bit repetition. In each slot transmission order of the bits is from left to right in the Table 10.

Table 10: Mapping order of repetition encoded TFCI code word bits to slots in Split Mode

Slot	TFCI code word bits in split mode							
0	$b^1_{14,1}$	$b^1_{14,2}$	$b^1_{14,3}$	$b^1_{14,4}$	$b^2_{14,1}$	$b^2_{14,2}$	$b^2_{14,3}$	$b^2_{14,4}$
1	$b^1_{13,1}$	$b^1_{13,2}$	$b^1_{13,3}$	$b^1_{13,4}$	$b^2_{13,1}$	$b^2_{13,2}$	$b^2_{13,3}$	$b^2_{13,4}$
2	$b^1_{12,1}$	$b^1_{12,2}$	$b^1_{12,3}$	$b^1_{12,4}$	$b^2_{12,1}$	$b^2_{12,2}$	$b^2_{12,3}$	$b^2_{12,4}$
3	$b^1_{11,1}$	$b^1_{11,2}$	$b^1_{11,3}$	$b^1_{11,4}$	$b^2_{11,1}$	$b^2_{11,2}$	$b^2_{11,3}$	$b^2_{11,4}$
4	$b^1_{10,1}$	$b^1_{10,2}$	$b^1_{10,3}$	$b^1_{10,4}$	$b^2_{10,1}$	$b^2_{10,2}$	$b^2_{10,3}$	$b^2_{10,4}$
5	$b^1_{9,1}$	$b^1_{9,2}$	$b^1_{9,3}$	$b^1_{9,4}$	$b^2_{9,1}$	$b^2_{9,2}$	$b^2_{9,3}$	$b^2_{9,4}$
6	$b^1_{8,1}$	$b^1_{8,2}$	$b^1_{8,3}$	$b^1_{8,4}$	$b^2_{8,1}$	$b^2_{8,2}$	$b^2_{8,3}$	$b^2_{8,4}$
7	$b^1_{7,1}$	$b^1_{7,2}$	$b^1_{7,3}$	$b^1_{7,4}$	$b^2_{7,1}$	$b^2_{7,2}$	$b^2_{7,3}$	$b^2_{7,4}$
8	$b^1_{6,1}$	$b^1_{6,2}$	$b^1_{6,3}$	$b^1_{6,4}$	$b^2_{6,1}$	$b^2_{6,2}$	$b^2_{6,3}$	$b^2_{6,4}$
9	$b^1_{5,1}$	$b^1_{5,2}$	$b^1_{5,3}$	$b^1_{5,4}$	$b^2_{5,1}$	$b^2_{5,2}$	$b^2_{5,3}$	$b^2_{5,4}$
10	$b^1_{4,1}$	$b^1_{4,2}$	$b^1_{4,3}$	$b^1_{4,4}$	$b^2_{4,1}$	$b^2_{4,2}$	$b^2_{4,3}$	$b^2_{4,4}$
11	$b^1_{3,1}$	$b^1_{3,2}$	$b^1_{3,3}$	$b^1_{3,4}$	$b^2_{3,1}$	$b^2_{3,2}$	$b^2_{3,3}$	$b^2_{3,4}$
12	$b^1_{2,1}$	$b^1_{2,2}$	$b^1_{2,3}$	$b^1_{2,4}$	$b^2_{2,1}$	$b^2_{2,2}$	$b^2_{2,3}$	$b^2_{2,4}$
13	$b^1_{1,1}$	$b^1_{1,2}$	$b^1_{1,3}$	$b^1_{1,4}$	$b^2_{1,1}$	$b^2_{1,2}$	$b^2_{1,3}$	$b^2_{1,4}$
14	$b^1_{0,1}$	$b^1_{0,2}$	$b^1_{0,3}$	$b^1_{0,4}$	$b^2_{0,1}$	$b^2_{0,2}$	$b^2_{0,3}$	$b^2_{0,4}$

4.4 Coding of compressed mode

In compressed mode, slots N_{first} to N_{last} are not used for transmission of data. As illustrated in Figure 16, which shows the example of fixed idle length position with single frame method (see section 0), the instantaneous transmit power is increased in the compressed frame in order to keep the quality (BER, FER, etc.) unaffected by the reduced processing gain. The amount of power increase depends on the transmission time reduction method (see section 4.4.2). What frames are compressed, are decided by the network. When in compressed mode, compressed frames can occur periodically, as illustrated in Figure 16, or requested on demand. The rate and type of compressed frames is variable and depends on the environment and the measurement requirements.

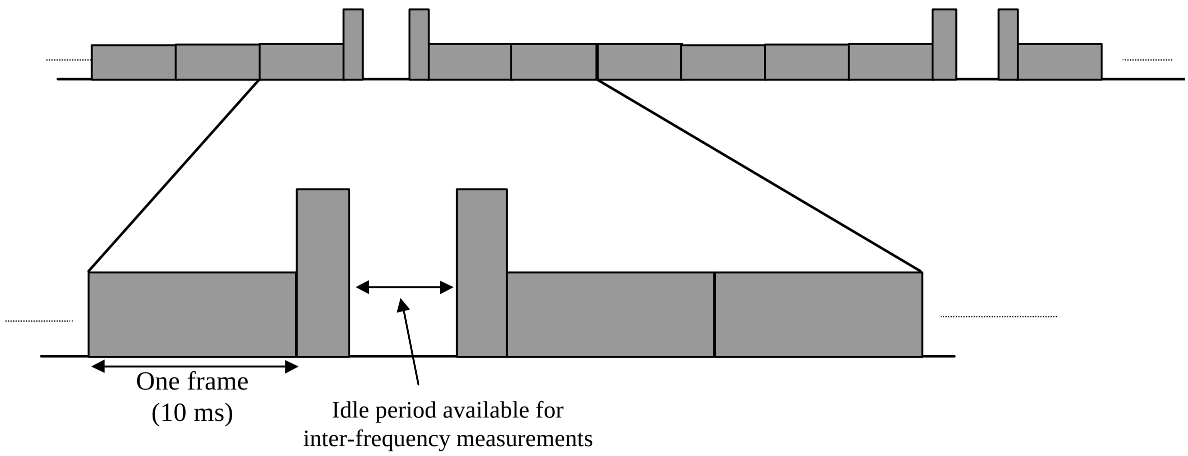
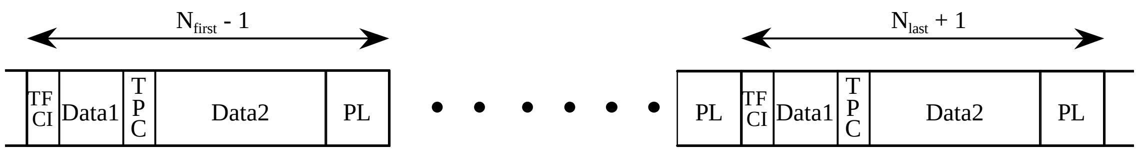


Figure 16: Compressed mode transmission

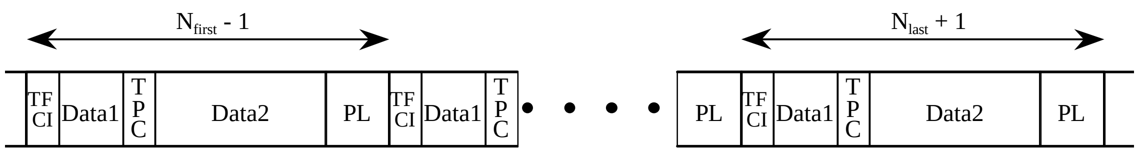
4.4.1 Frame structure types in downlink

There are two different types of frame structures defined for downlink compressed transmission. Type A is the basic case, which maximises the transmission gap length. Type B, which is more optimised for power control, can be used if the requirement of the transmission gap length allows that. Slot structure for uplink compressed mode is for further study.

- With frame structure of type A, BTS transmission is off from the beginning of TFCI field in slot N_{first}, until the end of Data2 field in slot N_{last} (Figure 17(a)).
- With frame structure of type B, BTS transmission is off from the beginning of Data2 field in slot N_{first}, until the end of Data2 field in slot N_{last} (Figure 17(b)) Dummy bits are transmitted in the TFCI and Data1 fields of slot N_{first}, and BTS and MS do not use the dummy bits. Thus BTS and MS utilize only the TPC field of N_{first}.



(a) Frame structure type A



(b) Frame structure type B

Figure 17: Frame structure types in downlink compressed transmission

4.4.2 Transmission time reduction method

When in compressed mode, the information normally transmitted during a 10 ms frame is compressed in time. The mechanism provided for achieving this is either changing the code rate, which means puncturing in practice, or the reduction of the spreading factor by a factor of two. The maximum idle length is defined to be 5 ms per one 10 ms frame.

4.4.2.1 Method A1: By puncturing, basic case

During compressed mode, rate matching (puncturing) is applied for making short transmission gap length in one frame. Algorithm of rate matching (puncturing) described in Section 4.2.7 is used. The maximum transmission gap length allowed to be achieved with this method is the case where the code rate is increased from 1/3 to 1/2 by puncturing, depending on the rate matching conditions that would be used in the non-compressed frame case. The explanation of the rate matching conditions are given below:

Example 1: If rate matching conditions in the non-compressed frame case would be such that maximum puncturing =0.2 would be used, then during compressed mode further puncturing of 1-(2/(3*(1-0.2))) =0.17 is allowed which corresponds to 0.17*15=2.55 => 2 time slots.

Example 2: If rate matching conditions in the non-compressed frame case would be such that no puncturing would be used, then during compressed mode puncturing of 1-(2/3)=0.33 is allowed which corresponds to 0.33*15=4.95 => 4 time slots.

DPDCH and DPCCH fields for compressed mode when puncturing 4 slots and 3 slots, respectively, are shown in Table 11 and Table 12. Because of higher encoding rate, some DPDCH symbols remain unused and shall be indicated as DTX.

Table 11: DPDCH and DPCCH fields in compressed mode when puncturing 4 slots

Channel Bit Rate (kbps)	Channel Symbol Rate (ksps)	SF	Bits/Frame			Bits/ Slot	DPDCH Bits/Slot		DPCCH Bits/Slot			Extra DPDCH symbols for DTX
			DPDCH	DPCCH	TOT		N_{Data1}	N_{Data2}	N_{TFCI}	N_{TPC}	N_{Pilot}	
16	8	512	40	66	110	10	2	2	0	2	4	4
32	16	256	100	110	220	20	2	8	0	2	8	10
32	16	256	80	140	220	20	0	8[1]	2[1]	2	8	0
64	32	128	300	110	440	40	6[1]	24	0	2	8	30
64	32	128	280	140	440	40	4[1]	24	2[1]	2	8	20
128	64	64	600	252	880	80	4[1]	56	8[1,2]	4	8	28
256	128	32	1400	252	1760	160	20[1]	120	8[1,2]	4	8	108
512	256	16	2880	384	3520	320	48[1]	240	8[1,2]	8	16	256
1024	512	8	6080	384	7040	640	112[1]	496	8[1,2]	8	16	576
2048	1024	4	12480	384	14080	1280	240[1]	1008	8[1,2]	8	16	1216

1) This figure does not take into account the extra TFCI bits from deleted slots

2) If no TFCI then the TFCI field is blank

Note: Compressed mode with puncturing cannot be used for SF=512 with TFCI

Table 12: DPDCH and DPCCH fields in compressed mode frame when puncturing 3 slots

Channel Bit Rate (kbps)	Channel Symbol Rate (ksps)	SF	Bits/Frame			Bits/ Slot	DPDCH Bits/Slot		DPCCH Bits/Slot			Extra DPDCH symbols for DTX
			DPDCH	DPCCH	TOT		N_{Data1}	N_{Data2}	N_{TFCI}	N_{TPC}	N_{Pilot}	
16	8	512	40	72	120	10	2	2	0	2	4	8
32	16	256	100	120	240	20	2	8	0	2	8	20
32	16	256	80	150	240	20	0	8[1]	2[1]	2	8	10
64	32	128	300	120	480	40	6	24	0	2	8	60
64	32	128	280	150	480	40	4[1]	24	2[1]	2	8	50
128	64	64	600	264	960	80	4[1]	56	8[1,2]	4	8	96
256	128	32	1400	264	1920	160	20[1]	120	8[1,2]	4	8	256
512	256	16	2880	408	3840	320	48[1]	240	8[1,2]	8	16	552
1024	512	8	6080	408	7680	640	112[1]	496	8[1,2]	8	16	1192
2048	1024	4	12480	408	15360	1280	240[1]	1008	8[1,2]	8	16	2472

1) This figure does not take into account the extra TFCI bits from deleted slots

2) If no TFCI then the TFCI field is blank

Note: Compressed mode with puncturing cannot be used for SF=512 with TFCI

4.4.2.2 Method A2: By puncturing, for services that allow larger delay

Other methods of supporting compressed mode may be considered as options. For example, with services that allows for a larger delay, e.g. data services with interleaving over several frames, multiple frames might be compressed together in order to create a short measurement slot. As an example, for a 2 Mbps service, with interleaving of 5 frames (50 ms), a 5 ms idle slot can be created by puncturing only 10% of 5 frames, as illustrated in Figure 18.

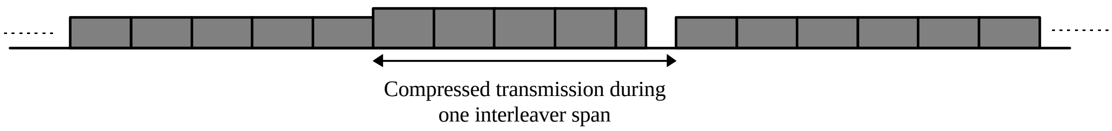


Figure 18: Multi-frame compressed mode for long-delay services

4.4.2.3 Method B: By reducing the spreading factor by 2

During compressed mode, the spreading factor (SF) can be reduced by 2 to enable the transmission of the information bits in the remaining time slots of a compressed frame. This can accommodate up to 50% idle slots per frame which is the maximum compression factor required. Additional rate matching is required if there are less than 50% idle slots. Reducing the spreading factor will normally be used if rate matching alone is not sufficient to transmit all information bits in compressed mode. Use of this method for uplink compressed mode is for further study.

On the downlink, UTRAN can also order the UE to use a different scrambling code in compressed mode than in normal mode. If the UE is ordered to use a different scrambling code in compressed mode, then there is a one-to-one mapping between the scrambling code used in normal mode and the one used in compressed mode, as described in TS 25.213 section 5.2.1.

4.4.3 Transmission gap position

Transmission gaps can be placed at both fixed position and adjustable position for each purpose such as interfrequency power measurement, acquisition of control channel of other system/carrier, and actual handover operation.

4.4.3.1 Fixed transmission gap position

The transmission gaps can be placed onto fixed positions. When using single frame method, the fixed transmission gap is located within the compressed frame depending on the transmission gap length as shown in Figure 19 (1). When using double frame method, the fixed transmission gap is located on the center of two connected frames as shown in Figure 19 (2). Table 13 shows the parameters for the fixed transmission gap position case.

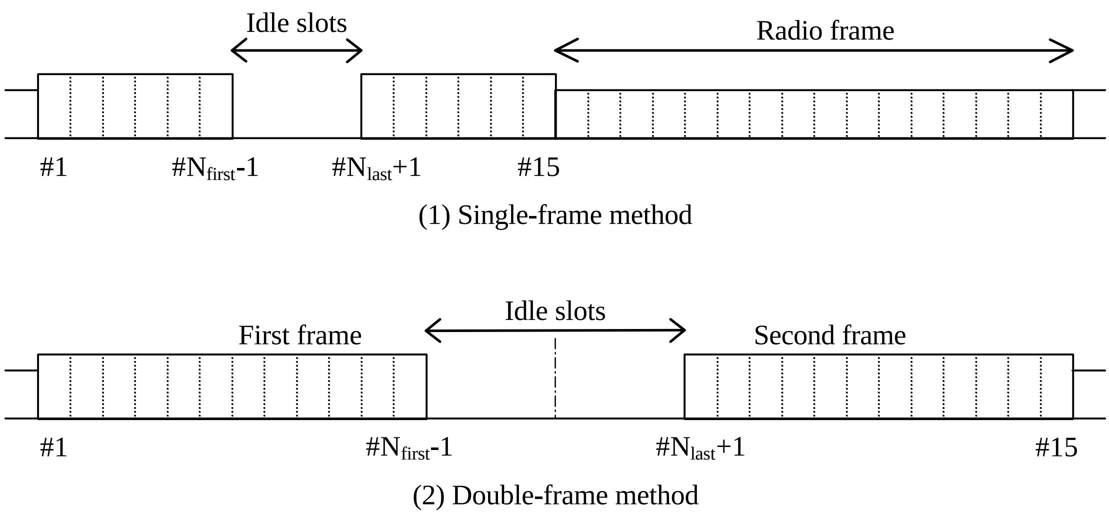


Figure 19: Fixedtransmission gap lengths position

Table 13: Parameters for fixed transmission gap position

	Single-frame method		Double-frame method	
Transmission gap length (slot)	**N_{first}**	**N_{last}**	**N_{first}**	**N_{last}**
3	8	10	15 in first frame	2 in second frame
4	7	10	14 in first frame	2 in second frame
7	7	13	13 in first frame	4 in second frame
10	N.A.	N.A.	11 in first frame	5 in second frame
14	N.A.	N.A.	9 in first frame	7 in second frame

4.4.3.2 Adjustable transmission gap position

Position of transmission gaps can be adjustable/relocatable for some purpose e.g. data acquisition on certain position as shown in Figure 20. Parameters of the adjustable transmission gap lengths positions are calculated as follows:

N_{idle} is the number of consecutive idle slots during compressed mode, as shown in Table 13,

N_{idle} = 3,4,5,6,8,10,16.

N_{first} specifies the starting slot of the consecutive idle slots,

N_{first} = 1,2,3,…,16.

N_{last} shows the number of the final idle slot and is calculated as follows;

If $N_{first} + N_{idle} <= 17$, then $N_{last} = N_{first} + N_{idle} - 1$ (in the same frame),

If $N_{first} + N_{idle} > 17$, then $N_{last} = N_{first} + N_{idle} - 17$ (in the next frame).

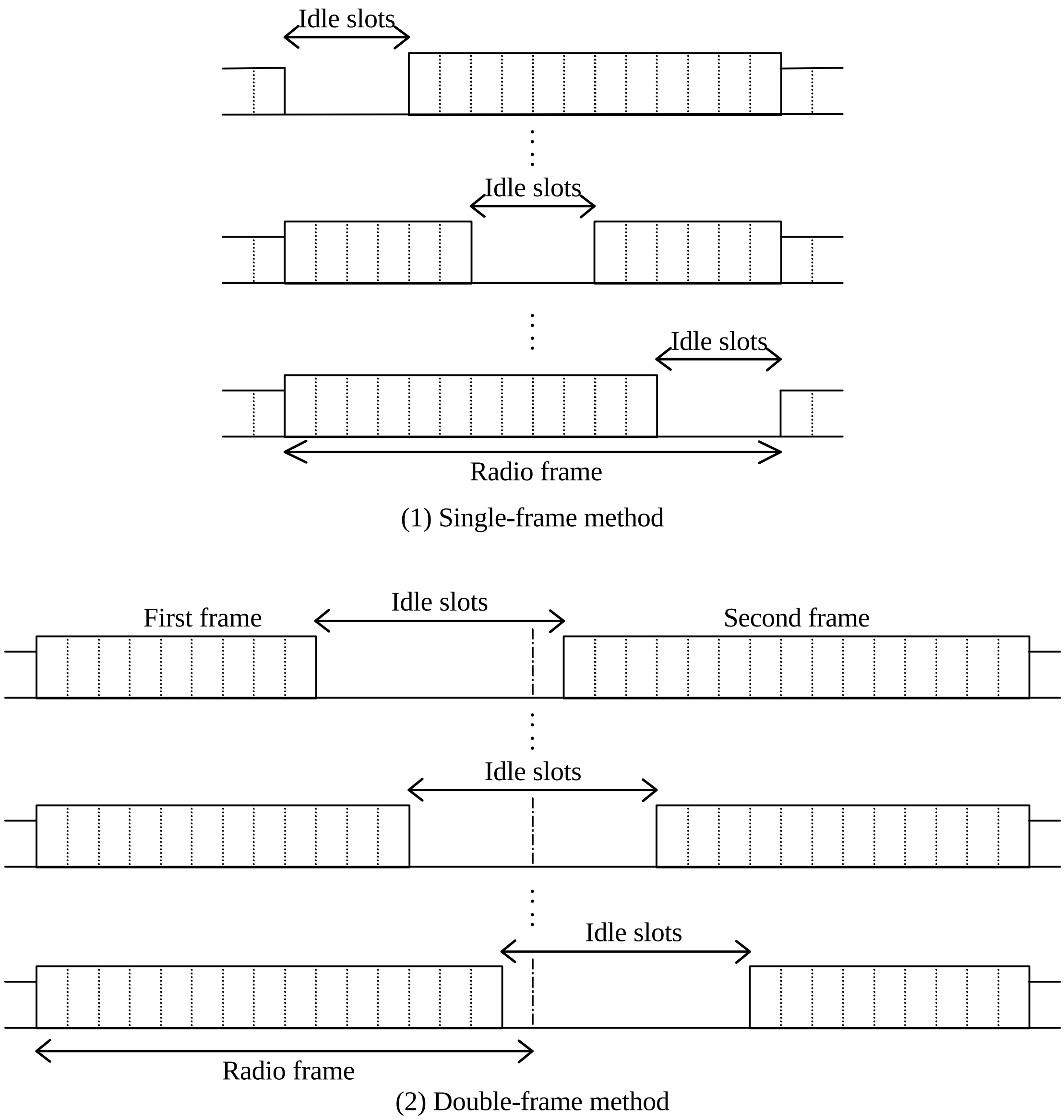


Figure 20: Concept of adjustable transmission gap lengths position

<Editors note: Adjustment needed>

4.4.3.3 Parameters for compressed mode

< Editor's note: WG1 suggestion is that there is need for further clarifications in Table 14 (e.g. rationales between change of coding rate/puncturing/change of spreading factor and idle time size, spreading factor range for different modes, etc.).>

Table 14 shows the detailed parameters for each number of idle slots. This is an example for the 10ms interleaving depth. Application of compressed mode for interleaving depths other than 10ms are for further study. Each number of idle slots are classified for three cases:

Case 1 - Power measurement : Number of idle slots = 3, 4, 5, 6.

Case 2 - Acquisition of control channels : Number of idle slots = 3, 4, 5, 6, 8, 10.

Case 3 - Actual handover operation : Number of idle slots = 10, 16.

Table 14: Parameters for compressed mode

Number of idle slots	Mode	Spreading Factor	Idle length [ms]	Transmission time reduction method	Idle frame combining
3	A	512 - 256	1.63 - 1.63	Puncturing Spreading facter reduction by 2	(S)/(D)
	B	128 - 4	1.63 - 1.75		
4	A	512 - 256	2.25 - 2.25	Puncturing (D) Spreading facter reduction by 2(D)/(S)	
	B	128 - 4	2.25 - 2.37		
5	A	512 - 256	2.87 - 2.87		
	B	128 - 4	2.87 – 2.99		
6	A	512 - 256	3.50 - 3.50	Puncturing (D)/(S) Spreading factor reduction by 2 (S)/(D)	
	B	128 - 4	3.50 - 3.62		
7	A	512 - 256	4.75 - 4.75	Spreading factor reduction by 2 (S)	
	B	128 - 4	4.75 - 4.87		
10	A	512 - 256	6.00 - 6.00	Puncturing Spreading factor reduction by 2	(D)
	B	128 - 4	6.00 - 6.12		
14	A	512 - 256	9.75 - 9.75	Puncturing Spreading factor reduction by 2	
	B	128 - 4	9.75 - 9.87		

(S): Single-frame method as shown in Figure 19 (1).

(D): Double-frame method as shown in Figure 19 (2).

SF="2/1": "2" is for (S) and "1" is for (D).

Annex A (informative): Blind transport format detection

A.1 Blind transport format detection using fixed positions

A.1.1 Blind transport format detection using received power ratio

- This method is used for dual transport format case (the possible data rates, 0 and full rate, and only transmitting CRC for full rate).
- The rate detection is done using average received power ratio of DPDCH to DPCCH.
  Pc: Received Power per bit of DPCCH calculated from all pilot and TPC bits per slot over 10ms frame.
  Pd: Received Power per bit of DPDCH calculated from *X* bits per slot over 10ms frame.
  X: the number of DPDCH bits per slot when transport format corresponds to full rate.
  T: Threshold of average received power ratio of DPDCH to DPCCH for rate detection.

If *Pd/Pc >T* then

"TX_ON"

else

"TX_OFF"

A.1.2 Blind transport format detection using CRC

- This method is used for multiple transport format case (the possible data rates: 0, …, (full rate)/r, …, full rate, and always transmitting CRC for all transport formats).
- At the transmitter, the variable-rate DCH data to be transmitted is block-encoded using a cyclic redundancy check (CRC) and then convolutionally encoded. It is necessary that the CRC parity bits are mapped on the head position (or certain position) in a frame as shown in Figure A-1.
- The receiver knows only the possible transport formats (or the possible end bit position $\{n_{end}\}$ by Layer-3 negotiation (See Figure A-1). The receiver performs Viterbi-decoding on the soft decision sample sequence. The correct trellis path of the Viterbi-decoder ends at the zero state at the correct end bit position.
- Blind rate detection method by using CRC traces back the surviving trellis path ending at the zero state (hypothetical trellis path) at each possible end bit position to recover the data sequence. Each recovered data sequence is then error-detected by CRC and if there is no error, the recovered sequence is declared to be correct.
- The following variable is defined:

$$s(n_{end}) = -10 \log \left((a_0(n_{end}) - a_{min}(n_{end})) / (a_{max}(n_{end}) - a_{min}(n_{end})) \right) \text{ [dB]} \qquad \text{(Eq. 1)}$$

  where $a_{max}(n_{end})$ and $a_{min}(n_{end})$ are, respectively, the maximum and minimum path-metric valuse among all survivors at end bit position n_{end}, and $a_0(n_{end})$ is the path-metric value at zero state.

- In order to reduce the probability of false detection (this happens if the selected path is wrong but the CRC misses the error detection), a path selection threshold D is introduced. D determines whether the hypothetical trellis path connected to the zero state should be traced back or not at each end bit position n_{end}. If the hypothetical trellis path connected to the zero state that satisfies

$$s(n_{end}) =< D \qquad \text{(Eq. 2)}$$

  is found, the path is traced back to recover the frame data, where D is the path selection threshold and a design parameter.

- If more than one end bit positions satisfying Eq. 2 are found, the end bit position which has minimum value of $s(n_{end})$ is declared to be correct.
- If no path satisfying Eq. 2 is found even after all possible end bit positions have been exhausted, the received frame data is declared to be in error.

Figure A-2 shows the procedure of blind transport format detection using CRC.

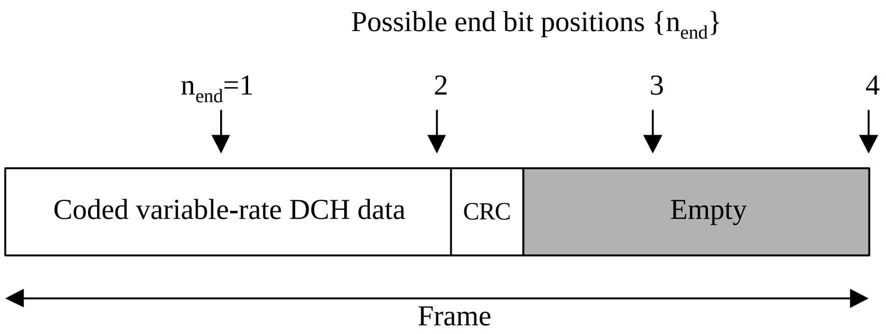


Figure A-1: An example of variable rate data format
(Number of possible transport formats = 4, transmitted end bit position n_{end} = 2)

A.2 Blind transport format detection with flexible positions

In certain cases where the CCtrCH consists of multiple transport channels and a small number of transport format combinations are allowed, it is possible to allow blind transport format detection with flexible positions.

Several examples for how the blind transport format detection with flexible positions might be performed are:

- The blind transport format detection starts at a fixed position and identifies the transport format of the first present transport channel and stops. The position of the other transport channels and their transport formatbeing derived on the basis of the allowed transport format combinations, assuming that there is a one to one relationship between the transport format combination and the transport format of the first present transport channel.
- The blind rate detection evaluates all transport format combinations and picks the most reliable one.

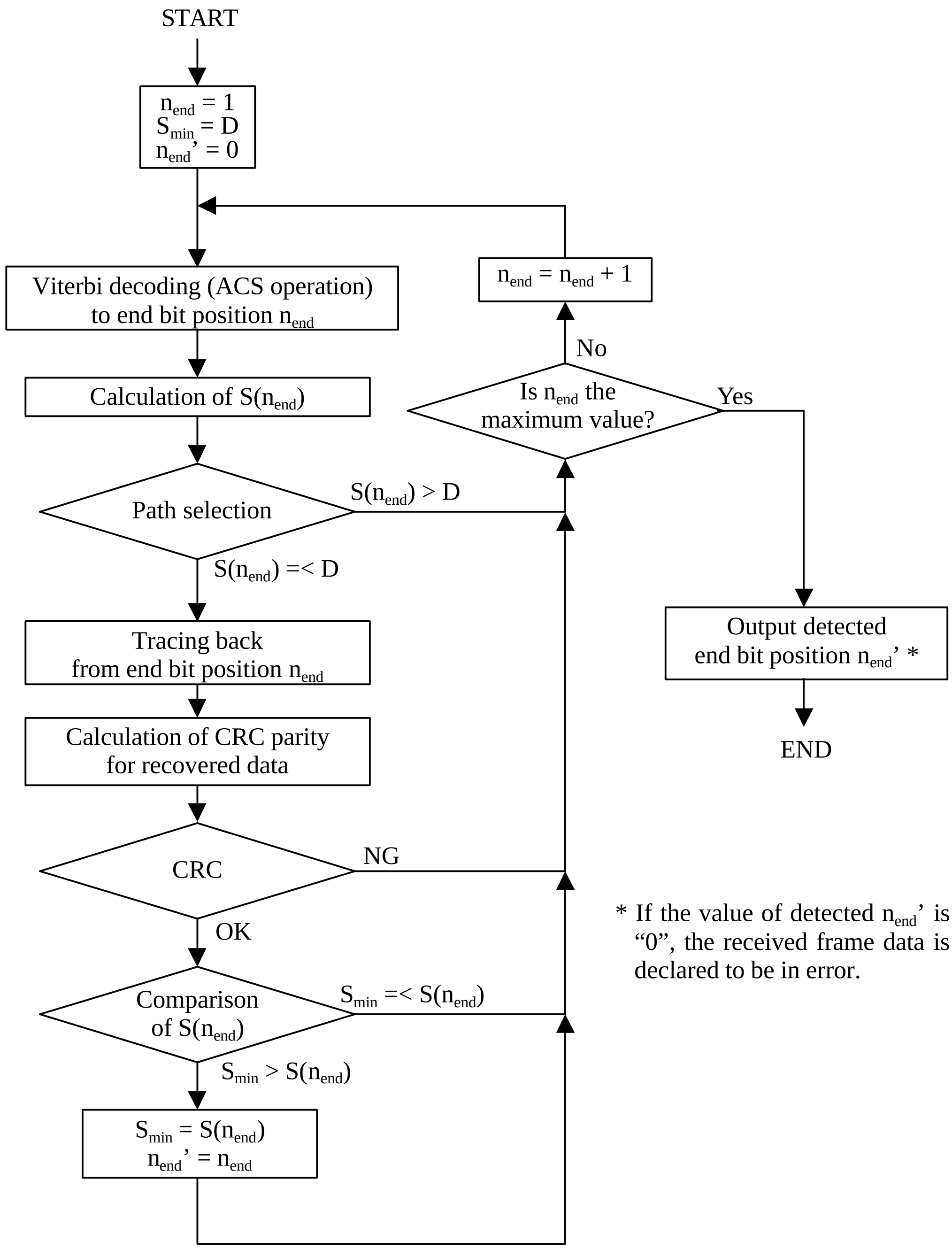


Figure A-2: Basic processing flow of blind transport format detection

5 History

Document history		
V0.0.1	1999-02-12	First version created by the editor on the basis of XX.04 and the Volume 3 of the ARIB specification.
V0.1.0	1999-02-26	Version approved by WG1#2 meeting (Yokohama). The changes agreed at the meeting to incorporate e.g. Ad Hoc conclusions not yet included.
V1.0.0	1999-03-05	Version approved by RAN. Identical to V0.1.0
V1.0.1	1999-03-17	Document updated based on Ad Hoc conclusions and comments at the WG1#2 meeting (Yokohama). Editorial changes also included.
V1.0.2	1999-03-23	Document updated based on Ad Hoc conclusions at the WG1#3 meeting. Editorial changes and corrections of mistake also included.
V1.0.3	1999-03-25	Document updated based on comments at the WG1#3 meeting. Editorial changes also included.
V1.1.0	1999-03-26	Version approved by WG1#3 meeting (Nynashamn). Identical to V1.0.3
V1.1.1	1999-04-19	Document updated based on text proposals approved in WG1#4 Day 1.
V1.1.2	1999-04-20	Document updated based on text proposals approved in WG1#4 Day 2.
V2.0.0	1999-04-20	Version approved by WG1#4 meeting (Yokohama).
V2.0.1	1999-04-22	This is a results from the drafting group in RAN#3 based on the comments received.
TS 25.212 V1.0.0	1999-04-22	Noted by TSG-RAN as TS 25.212 V1.0.0
V1.0.1	1999-06-04	Document updated based on text proposals approved in WG1#5 (Cheju).
V1.1.0	1999-06-04	Version approved by WG1#5 meeting (Cheju).
V2.0.0	1999-06-22	Endorsed by TSG-RAN as version 2.0.0
V2.0.1	1999-08-17	Document updated based on text proposals approved in WG1#6 (Espoo).
V2.1.0	1999-08-31	Version approved at August 30 by WG1#7 meeting (Hannover).
V2.1.1	1999-09-03	Document updated based on text proposals approved in WG1#7 (Hannover).
V2.2.0	1999-09-06	Version approved at WG1#7 (Hannover) for RAN submission.

Editor for TS 25.212, Multiplexing and channel coding (FDD), is:

Yoshinori Tanaka
Fujitsu Laboratories Ltd.
Email: yoshi@flab.fujitsu.co.jp

This document is written in Microsoft Word 97.