ETSI TS 125 212 V3.1.1 (2000-01)

Technical Specification

Universal Mobile Telecommunications System (UMTS); Multiplexing and channel coding (FDD) (3G TS 25.212 version 3.1.1 Release 1999)

Reference

DTS/TSGR-0125212U

Keywords

UMTS

ETSI

Postal address

F-06921 Sophia Antipolis Cedex - FRANCE

Office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16
Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Internet

secretariat@etsi.fr
Individual copies of this ETSI deliverable
can be downloaded from
http://www.etsi.org
If you find errors in the present document, send your
comment to: editor@etsi.fr

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.org/ipr).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by the ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables. The mapping of document identities is as follows:

For 3GPP documents:

3G TS | TR nn.nnn "<title>" (with or without the prefix 3G)

is equivalent to

ETSI TS | TR 1nn nnn "[Digital cellular telecommunications system (Phase 2+) (GSM);] Universal Mobile Telecommunications System; <title>

For GSM document identities of type "GSM xx.yy", e.g. GSM 01.04, the corresponding ETSI document identity may be found in the Cross Reference List on www.etsi.org/key

Contents

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

The present document describes the characteristics of the Layer 1 multiplexing and channel coding in the FDD mode of UTRA.

2 References

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

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1] 3G TS 25.201: "Physical layer – General Description"

[2] 3G TS 25.211: "Physical channels and mapping of transport channels onto physical channels (FDD)"

[3] 3G TS 25.213: "Spreading and modulation (FDD)"

[4] 3G TS 25.214: "Physical layer procedures (FDD)"

[5] 3G TS 25.215: "Measurements (FDD)"

[6] 3G TS 25.221: "Physical channels and mapping of transport channels onto physical channels (TDD)"

[7] 3G TS 25.222: "Multiplexing and channel coding (TDD)"

[8] 3G TS 25.223: "Spreading and modulation (TDD)"

[9] 3G TS 25.224: "Physical layer procedures (TDD)"

[10] 3G TS 25.225: "Measurements (TDD)"

[11] 3G TS 25.302: "Services Provided by the Physical Layer"

[12] 3G TS 25.402: "Synchronisation in UTRAN, Stage 2"

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.

TG: Transmission Gap is consecutive empty slots that have been obtained with a transmission time reduction method. The transmission gap can be contained in one or two consecutive radio frames.

TGL: Transmission Gap Length is the number of consecutive empty slots that have been obtained with a transmission time reduction method. $0 \leq TGL \leq 14$.

TrCH number: Transport channel number represents a TrCH ID assigned to L1 by L2. Transport channels are multiplexed to the CCTrCH in the ascending order of these IDs.

3.2 Symbols

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

$\lceil x \rceil$	round towards ∞, i.e. integer such that $x \le \lceil x \rceil < x+1$
$\lfloor x \rfloor$	round towards -∞, i.e. integer such that $x-1 < \lfloor x \rfloor \le x$
$\lvert x \rvert$	absolute value of x
N_{first}	The first slot in the *TG*.
N_{last}	The last slot in the *TG*. N_{last} is either a slot in the same radio frame as N_{first} or a slot in the radio frame immediately following the slot that contains N_{first}.

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.

Temporary variables, i.e. variables used in several (sub)sections with different meaning.

x, X
y, Y
z, Z

3.3 Abbreviations

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

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
RACH	Random Access Channel
RSC	Recursive Systematic Convolutional Coder
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 sections 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.

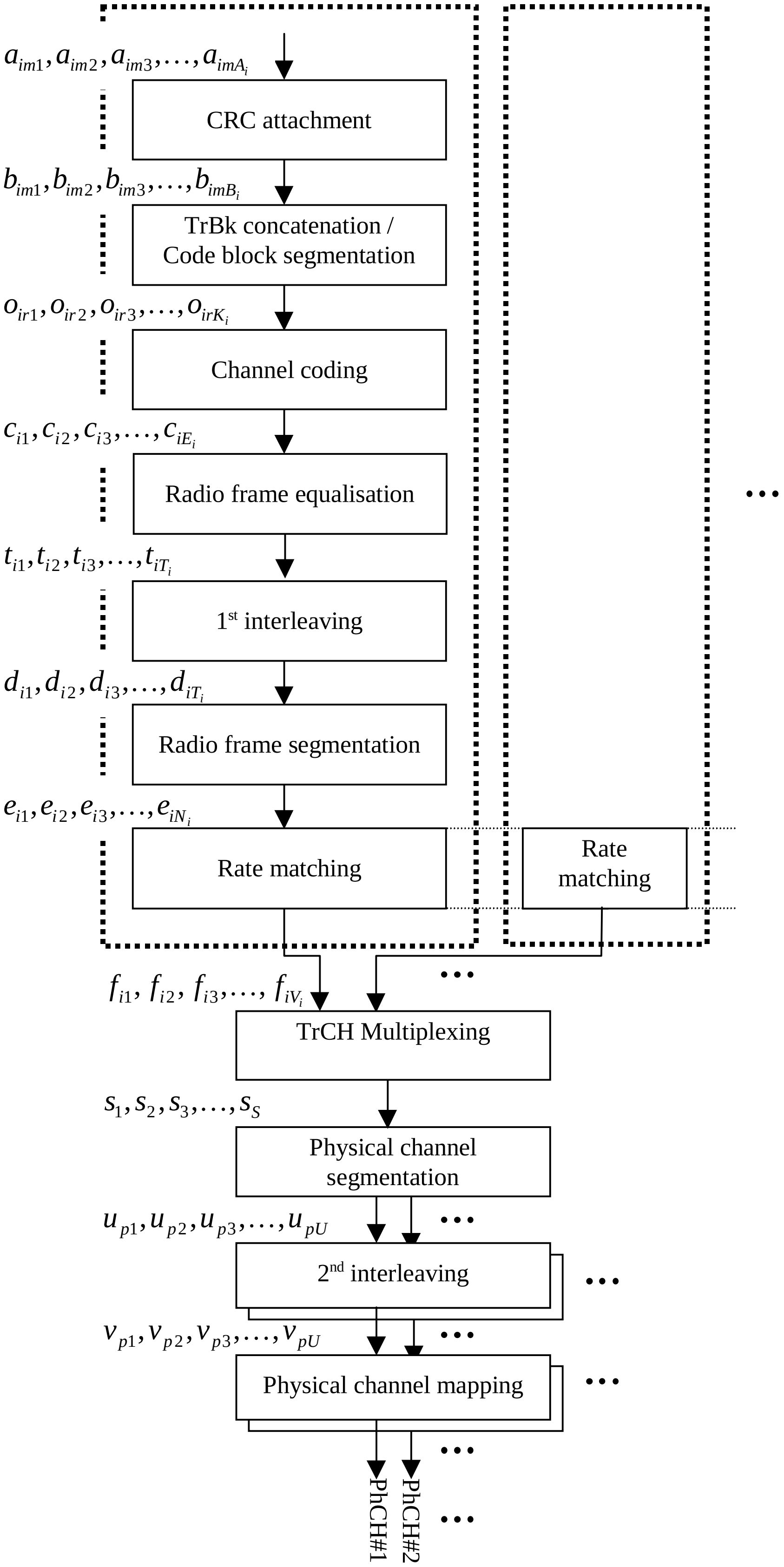


Figure 1: Transport channel multiplexing structure for uplink

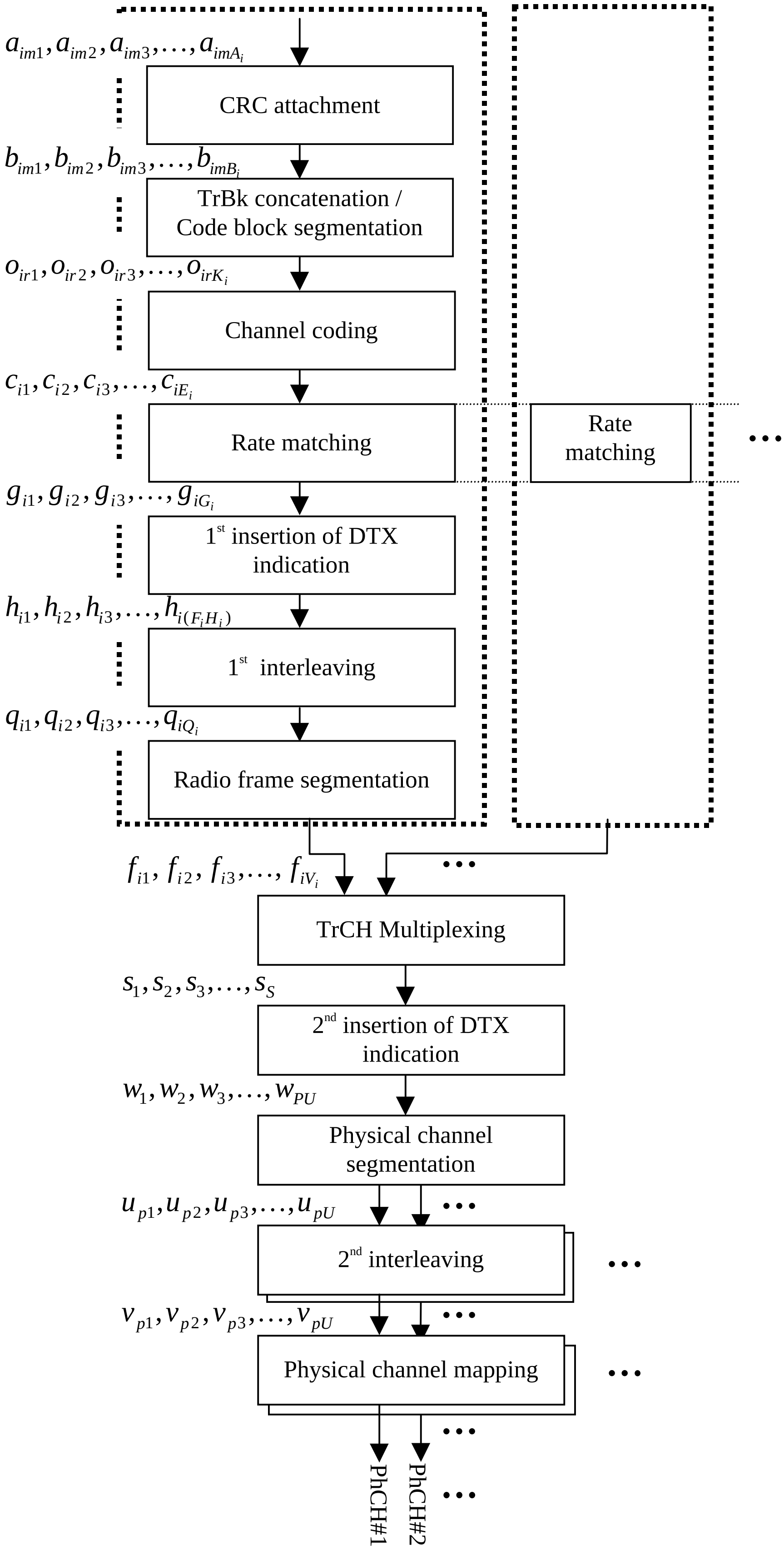


Figure 2: Transport channel multiplexing structure for downlink

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, 12, 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:

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

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

$g_{CRC12}(D) = D^{12} + D^{11} + D^{3} + D^{2} + D + 1$

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

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, 12, 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 $g_{CRC16}(D)$, polynomial

$$a_{im1}D^{A_i+11} + a_{im2}D^{A_i+10} + \ldots + a_{imA_i}D^{12} + p_{im1}D^{11} + p_{im2}D^{10} + \ldots + p_{im11}D^{1} + p_{im12}$$

yields a remainder equal to 0 when divided by $g_{CRC12}(D)$ 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 $g_{CRC8}(D)$.

If no transport blocks are input to the CRC calculation ($M_i = 0$), no CRC attachment shall be done.

4.2.1.1.1 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*, the maximum size of a code block in question, then code block segmentation is performed after the concatenation of the transport blocks. The maximum size of the code blocks depends on whether convolutional coding, turbo coding or no 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:

$x_{ik} = b_{i1k}$ $k = 1, 2, \ldots, B_i$

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

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

…

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

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: $Z = 504$

turbo coding: $Z = 5114$

no channel coding: Z = *unlimited*

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: $C_i = \lceil X_i / Z \rceil$

Number of bits in each code block: $K_i = \lceil X_i / C_i \rceil$

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

$o_{i1k} = x_{ik}$ $k = 1, 2, \ldots, K_i$

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

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

…

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

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

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 $y_{ir1}, y_{ir2}, y_{ir3}, \ldots, y_{irY_i}$. 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:

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

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

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

…

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

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		
RACH		
CPCH, DCH, DSCH, FACH		1/3, 1/2
	Turbo Code	1/3
	No coding	

If no code blocks are input to the channel coding ($C_i = 0$), no bits shall be output from the channel coding, i.e. $E_i = 0$.

4.2.3.1 Convolutional coding

Convolutional codes with constraint length 9 and coding rates 1/3 and 1/2 are defined.

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

Output from the rate 1/3 convolutional coder shall be done in the order output0, output1, output2, output0, output1, output 2, output 0,…,output2. 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.

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.

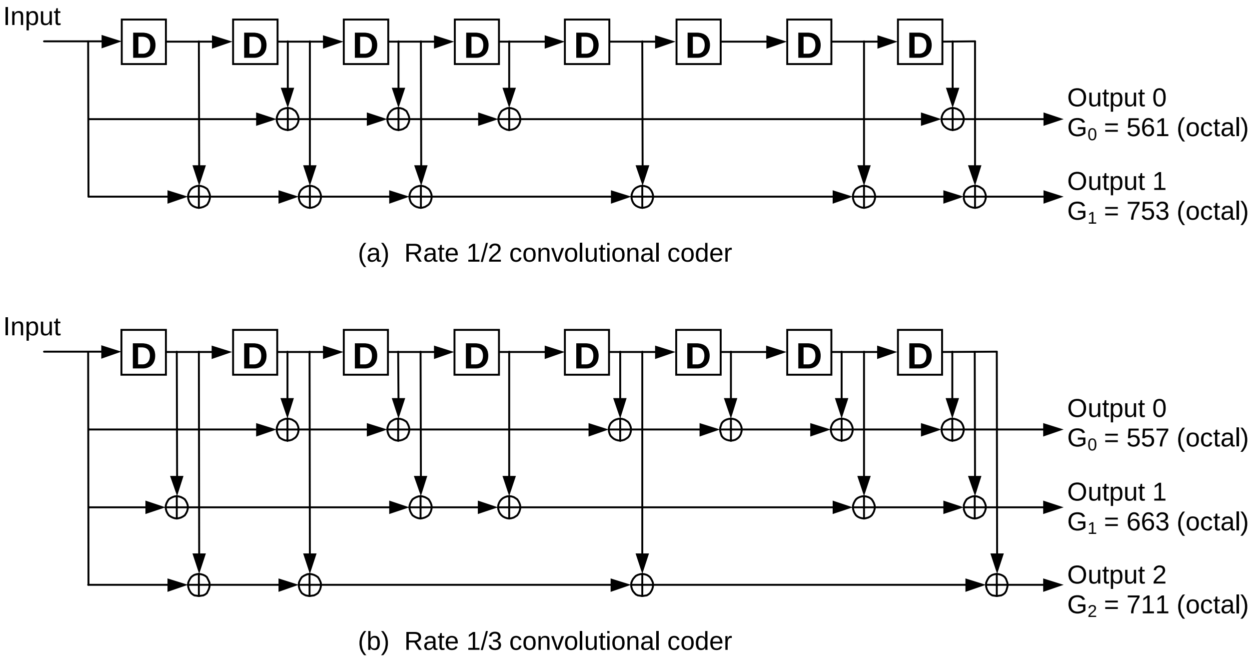


Figure 3: Rate 1/2 and rate 1/3 convolutional coders

4.2.3.2 Turbo coding

4.2.3.2.1 Turbo coder

The turbo coding scheme is a parallel concatenated convolutional code (PCCC) with 8-state constituent encoders.

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

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

where,

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

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

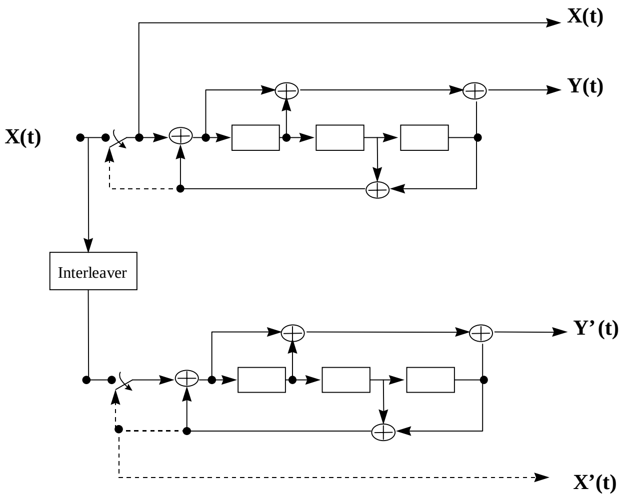


Figure 4: Structure of the 8 state PCCC encoder (dotted lines effective for trellis termination only)

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. For rate 1/3, none of the systematic or parity bits are punctured, and the output sequence is X(0), Y(0), Y'(0), X(1), Y(1), Y'(1), etc.

4.2.3.2.2 Trellis termination for Turbo coding

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

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

The transmitted bits for trellis termination shall then be

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

4.2.3.2.3 Turbo code internal interleaver

Figure 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. Tail bits T_1 and T_2 are added for constituent encoders RSC1 and RSC2, respectively. The definition of *l* is shown in section 4.2.3.2.3.2.

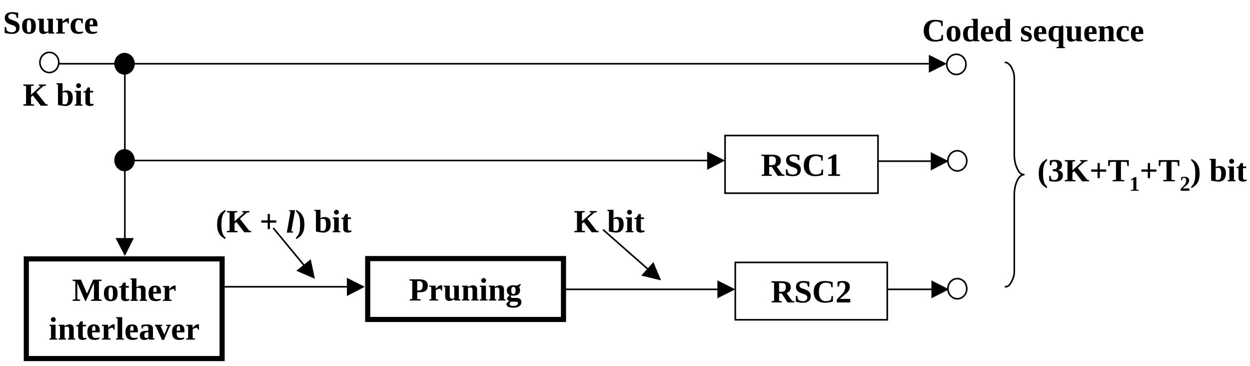


Figure 5: Overall 8 State PCCC Turbo Coding

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 the number of rows 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 the number of columns C such that

Case-1; C = $p = 53$

Case-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 starting from row 0.

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,…, (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 starting from column 0.

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 4.1.6. 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:

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

(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}:

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

(5) Read the output bit sequence $y_{i1}, y_{i2}, y_{i3}, \ldots, y_{i,(C_I R_I)}$ 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 $t_{i1}, t_{i2}, t_{i3}, \ldots, t_{iT_i}$, where i is the TrCH number and T_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 $d_{i1}, d_{i2}, d_{i3}, \ldots, d_{iT_i}$, 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 $h_{i1}, h_{i2}, h_{i3}, \ldots, h_{i(F_i H_i)}$, 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 $g_{i1}, g_{i2}, g_{i3}, \ldots, g_{iG_i}$, 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 $q_{i1}, q_{i2}, q_{i3}, \ldots, q_{iQ_i}$, 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:

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

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

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 TrCH multiplexing is identical to the total channel bit rate of the allocated dedicated physical channels.

If no bits are input to the rate matching for all TrCHs within a CCTrCH, the rate matching shall output no bits for all TrCHs within the CCTrCH and no uplink DPDCH will be selected in the case of uplink rate matching.

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 an 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 an 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 or repetition distance (normalised to only show the remaining rate matching on top of an integer number of repetitions). 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 or repetition 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 4.2.7.5.

e_{plus} Increment of variable *e* in the rate matching pattern determination algorithm of section 4.2.7.5.

e_{minus} Decrement of variable *e* in the rate matching pattern determination algorithm of section 4.2.7.5.

b: Indicates systematic and parity bits

b=1: Systematic bit. *X(t)* in section 4.2.3.2.1.

b=2: 1st parity bit (from the upper Turbo constituent encoder). *Y(t)* in section 4.2.3.2.1.

b=3: 2nd parity bit (from the lower Turbo constituent encoder). *Y'(t)* in section 4.2.3.2.1.

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** x **do** $X_x = Y$ ". In the right wing of an assignment, the meaning is that "$Y = X_*$ " is equivalent to "**take any** x **and do** $Y = X_x$ "

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

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

$$Z_{ij} = \left\lfloor \frac{\sum_{m=1}^{i} RM_m \cdot N_{mj}}{\sum_{m=1}^{I} RM_m \cdot N_{mj}} \cdot N_{data,j} \right\rfloor \quad \text{for all } i = 1 \text{ .. I} \qquad (1)$$

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

4.2.7.1 Determination of rate matching parameters in uplink

4.2.7.1.1 Determination of SF and number of PhCHs needed

In uplink, puncturing can be applied to match the CCTrCH bit rate to the PhCH bit rate. The bit rate of the PhCH(s) is limited by the UE capability and restrictions imposed by UTRAN, through limitations on the PhCH spreading factor. 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 capability and the restrictions from UTRAN, the allowed 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 SET1 is not empty and 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

4.2.7.1.2 Determination of parameters needed for calculating the rate matching pattern

The number of bits to be repeated or punctured, ΔN_{ij}, within one radio frame for each TrCH *i* is calculated with equation 1 for all possible transport format combinations *j* and selected every radio frame. $N_{data,j}$ is given from section 4.2.7.1.1. In compressed mode $N_{data,j}$ is replaced by $N_{data,j}^{cm}$ in Equation 1. $N_{data,j}^{cm}$ is given from the following relation:

$N^{cm}_{data,j} = 2N_{data,j} - 2N_{TGL}$, for compressed mode by spreading factor reduction

$N^{cm}_{data,j} = N_{data,j} - N_{TGL}$, for compressed mode by higher layer scheduling

$$N_{TGL} = \begin{cases} \frac{TGL}{15} N_{data,j}, \text{ if } N_{first} + TGL \le 15 \\ \frac{15 - N_{first}}{15} N_{data,j}, \text{ in first frame if } N_{first} + TGL > 15 \\ \frac{TGL - (15 - N_{first})}{15} N_{data,j}, \text{ in second frame if } N_{first} + TGL > 15 \end{cases}$$

N_{first} and *TGL* are defined in section 4.4.

If $\Delta N_{ij} = 0$ then the output data of the rate matching is the same as the input data and the rate matching algorithm of section 4.2.7.5 does not need to be executed.

If $\Delta N_{ij} \neq 0$ the parameters listed in sections 4.2.7.1.2.1 and 4.2.7.1.2.2 shall be used for determining e_{ini}, e_{plus}, and e_{minus} (regardless if the radio frame is compressed or not).

4.2.7.1.2.1 Uncoded and convolutionally encoded TrCHs

$R = \Delta N_{ij}$ mod N_{ij} -- note: in this context ΔN_{ij} mod N_{ij} is in the range of 0 to N_{ij}-1 i.e. -1 mod 10 = 9.

if $R \neq 0$ and $2R \le N_{ij}$

then $q = \lceil N_{ij} / R \rceil$

else

$q = \lceil N_{ij} / (R - N_{ij}) \rceil$

endif

-- note: *q* is a signed quantity.

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

$S(I_F(|\lfloor x*q' \rfloor| \bmod F_i)) = (|\lfloor x*q' \rfloor| \text{ div } F_i)$

end for

$\Delta N_i = \Delta N_{i,j}$

a = 2

For each radio frame, the rate-matching pattern is calculated with the algorithm in section 4.2.7.5, where :

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

$e_{ini} = (a \cdot S(n_i) \cdot |\Delta N_i| + 1) \bmod (a \cdot N_{ij})$.

$e_{plus} = a \cdot N_{ij}$

$e_{minus} = a \cdot |\Delta N_i|$

puncturing for $\Delta N<0$, repetition otherwise.

4.2.7.1.2.2 Turbo encoded TrCHs

If repetition is to be performed on turbo encoded TrCHs, i.e. $\Delta N_{i,j} >0$, the parameters in section 4.2.7.1.2.1 are used.

If puncturing is to be performed, the parameters below shall be used. Index *b* is used to indicate systematic (*b*=1), 1st parity (*b*=2), and 2nd parity bit (*b*=3).

$a=2$ when $b=2$

$a=1$ when $b=3$

$$\Delta N_i = \begin{cases} \lfloor \Delta N_{i,j}/2 \rfloor, & b=2 \\ \lceil \Delta N_{i,j}/2 \rceil, & b=3 \end{cases}$$

$X_i = \lfloor N_{i,j}/3 \rfloor$,

$q = \lfloor X_i / |\Delta N_i| \rfloor$

```
if(q ≤ 2)
    for x=0 to Fi-1
        S[IF[(3x+b-1) mod Fi]] = x mod 2;
    end for
else
    if q is even
    then  q' = q – gcd(q, Fi)/ Fi   -- where gcd (q, Fi) means greatest common divisor of q and Fi
        -- note that q' is not an integer, but a multiple of 1/8
    else    q' = q
    endif
    for x=0 to Fi -1
        r = ⌈ x*q' ⌉ mod Fi;
        S[IF[(3r+b-1) mod Fi]] = ⌈ x*q' ⌉ div Fi;
    endfor
endif
```

For each radio frame, the rate-matching pattern is calculated with the algorithm in section 4.2.7.5, where:

X_i is as above,

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

$e_{plus} = a \cdot X_i$

$e_{minus} = a \cdot |\Delta N_i|$

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:

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

If $\Delta N_{max} = 0$ then, for TrCH *i*, the output data of the rate matching is the same as the input data and the rate matching algorithm of section 4.2.7.5 does not need to be executed. In this case we have :

$$\forall l \in TFS(i)\, \Delta N_{i,l}^{TTI} = 0$$

If $\Delta N_{max} \neq 0$ the parameters listed in sections 4.2.7.2.1.1 and 4.2.7.2.1.2 shall be used for determining e_{ini}, e_{plus}, and e_{minus}.

4.2.7.2.1.1 Uncoded and convolutionally encoded TrCHs

$$\Delta N_i = \Delta N_{max}$$

a=2

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

For each transmission time interval of TrCH *i* with TF *l*, the rate-matching pattern is calculated with the algorithm in section 4.2.7.5. The following parameters are used as input:

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

$$e_{ini} = 1$$

$$e_{plus} = a \cdot N_{max}$$

$$e_{minus} = a \cdot |\Delta N_i|$$

Puncturing if $\Delta N_i < 0$, repetition otherwise. The values of $\Delta N_{i,l}^{TTI}$ may be computed by counting repetitions or puncturing when the algorithm of section 4.2.7.5 is run.

4.2.7.2.1.2 Turbo encoded TrCHs

If repetition is to be performed on turbo encoded TrCHs, i.e. $\Delta N_{max} > 0$, the parameters in section 4.2.7.2.1.1 are used.

If puncturing is to be performed, the parameters below shall be used. Index *b* is used to indicate systematic (*b*=1), 1st parity (*b*=2), and 2nd parity bit (*b*=3).

$a=2$ when $b=2$

$a=1$ when $b=3$

The bits indicated by b=1 shall not be punctured.

$$\Delta N_i = \begin{cases} \lfloor \Delta N_{max}/2 \rfloor, & b=2 \\ \lceil \Delta N_{max}/2 \rceil, & b=3 \end{cases}$$

$$N_{max} = \max_{l \in TFS(i)} (N_{il}^{TTI}/3)$$

For each transmission time interval of TrCH i with TF l, the rate-matching pattern is calculated with the algorithm in section 4.2.7.5. The following parameters are used as input:

$$X_i = N_{il}^{TTI}/3$$

$$e_{ini} = N_{max}$$

$$e_{plus} = a \cdot N_{max}$$

$$e_{\min us} = a \cdot |\Delta N_i|$$

The values of $\Delta N_{i,l}^{TTI}$ may be computed by counting repetitions or puncturing when the algorithm of section 4.2.7.5 is run.

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:

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

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.

If $\Delta N_{i,l}^{TTI} = 0$ then, for TrCH *i* at TF *l*, the output data of the rate matching is the same as the input data and the rate matching algorithm of section 4.2.7.5 does not need to be executed.

If $\Delta N_{i,l}^{TTI} \neq 0$ the parameters listed in sections 4.2.7.2.2.1 and 4.2.7.2.2.2 shall be used for determining e_{ini}, e_{plus}, and e_{minus}.

4.2.7.2.2.1 Uncoded and convolutionally encoded TrCHs

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

a=2

For each transmission time interval of TrCH *i* with TF *l*, the rate-matching pattern is calculated with the algorithm in section 4.2.7.5. The following parameters are used as input:

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

$e_{ini} = 1$

$e_{plus} = a \cdot N_{il}^{TTI}$

$e_{minus} = a \cdot |\Delta N_i|$

puncturing for $\Delta N_i < 0$, repetition otherwise.

4.2.7.2.2.2 Turbo encoded TrCHs

If repetition is to be performed on turbo encoded TrCHs, i.e. $\Delta N_{il}^{TTI} > 0$, the parameters in section 4.2.7.2.2.1 are used.

If puncturing is to be performed, the parameters below shall be used. Index *b* is used to indicate systematic (*b*=1), 1st parity (*b*=2), and 2nd parity bit (*b*=3).

a=2 when *b*=2

a=1 when *b*=3

The bits indicated by *b*=1 shall not be punctured.

$$\Delta N_i = \begin{cases} \lfloor \Delta N_{il}^{TTI} / 2 \rfloor, & b = 2 \\ \lceil \Delta N_{il}^{TTI} / 2 \rceil, & b = 3 \end{cases}$$

For each transmission time interval of TrCH *i* with TF *l*, the rate-matching pattern is calculated with the algorithm in section 4.2.7.5. The following parameters are used as input:

$X_i = N_{il}^{TTI} / 3$N,

$e_{ini} = X_i$,

$e_{plus} = a \cdot X_i$

$e_{minus} = a \cdot |\Delta N_i|$

4.2.7.3 Bit separation and collection in uplink

The systematic bits (excluding bits for trellis termination) of turbo encoded TrCHs shall not be punctured. The systematic bit, first parity bit, and second parity bit in the bit sequence input to the rate matching block are therefore separated from each other. Puncturing is only applied to the parity bits and systematic bits used for trellis termination.

The bit separation function is transparent for uncoded TrCHs, convolutionally encoded TrCHs, and for turbo encoded TrCHs with repetition. The bit separation and bit collection are illustrated in figures 6 and 7.

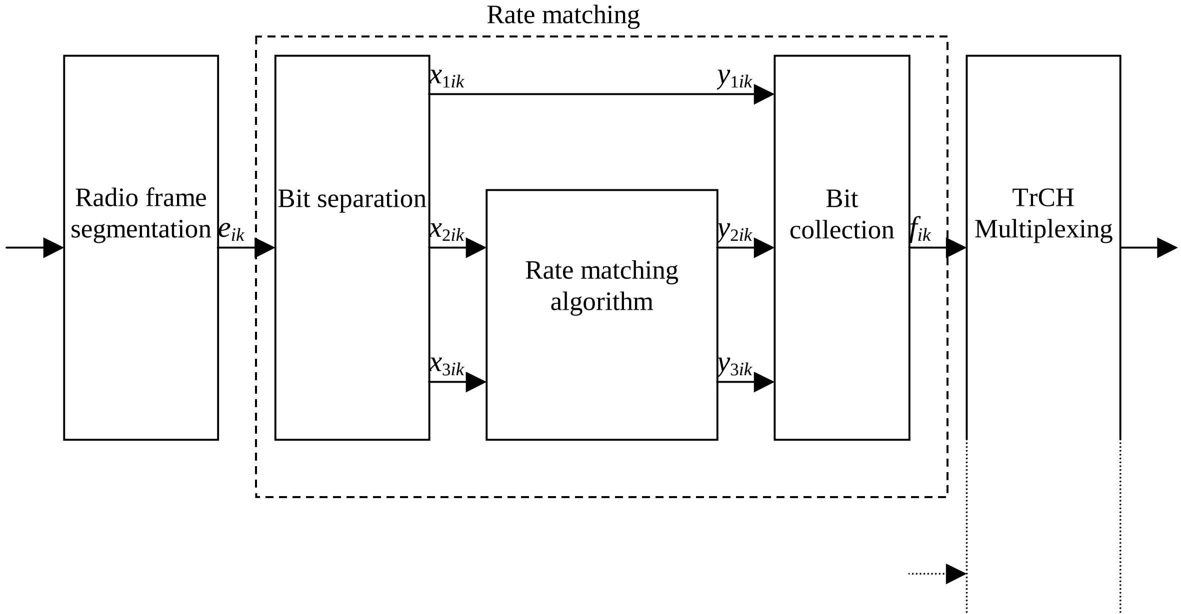


Figure 6: Puncturing of turbo encoded TrCHs

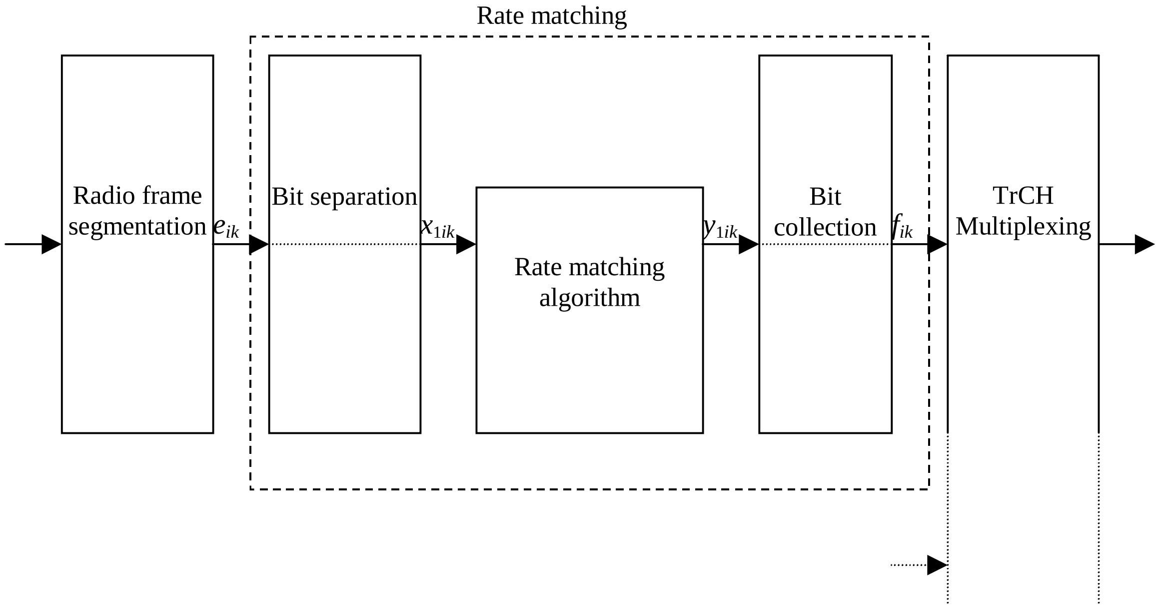


Figure 7: Rate matching for uncoded TrCHs, convolutionally encoded TrCHs, and for turbo encoded TrCHs with repetition.

The bit separation is dependent on the 1st interleaving and offsets are used to define the separation for different TTIs. The offsets α_b for the systematic (b=1) and parity bits ($b\in\{2, 3\}$) are listed in table 4.

Table 4: TTI dependent offset needed for bit separation

TTI (ms)	α_1	α_2	α_3
10, 40	0	1	2
20, 80	0	2	1

The bit separation is different for different radio frames in the TTI. A second offset is therefore needed. The radio frame number for TrCH i is denoted by n_i. and the offset by β_{n_i} .

Table 5: Radio frame dependent offset needed for bit separation

TTI (ms)	β_0	β_1	β_2	β_3	β_4	β_5	β_6	β_7
10	0	NA	NA	NA	NA	NA	NA	NA
20	0	1	NA	NA	NA	NA	NA	NA
40	0	1	2	0	NA	NA	NA	NA
80	0	1	2	0	1	2	0	1

4.2.7.3.1 Bit separation

The bits input to the rate matching are 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 input to the rate matching block. Note that the transport format combination number j for simplicity has been left out in the bit numbering, i.e. N_i=N_{ij}. The bits after separation are denoted by $x_{bi1}, x_{bi2}, x_{bi3}, \ldots, x_{biX_i}$. For turbo encoded TrCHs with puncturing, b indicates systematic, first parity, or second parity bit. For all other cases b is defined to be 1. X_i is the number of bits in each separated bit sequence. The relation between e_{ik} and x_{bik} is given below.

For turbo encoded TrCHs with puncturing:

$x_{1,i,k} = e_{i,3(k-1)+1+(\alpha_1+\beta_{n_i}) \bmod 3}$ k = 1, 2, 3, …, X_i $X_i = \lfloor N_i/3 \rfloor$

$x_{1,i,\lfloor N_i/3 \rfloor + k} = e_{i,3\lfloor N_i/3 \rfloor + k}$ k = 1, …, N_i mod 3 Note: When (N_i mod 3) = 0 this row is not needed.

$x_{2,i,k} = e_{i,3(k-1)+1+(\alpha_2+\beta_{n_i}) \bmod 3}$ $k = 1, 2, 3, \ldots, X_i$ $X_i = \lfloor N_i / 3 \rfloor$

$x_{3,i,k} = e_{i,3(k-1)+1+(\alpha_3+\beta_{n_i}) \bmod 3}$ $k = 1, 2, 3, \ldots, X_i$ $X_i = \lfloor N_i / 3 \rfloor$

For uncoded TrCHs, convolutionally encoded TrCHs, and turbo encoded TrCHs with repetition:

$x_{1,i,k} = e_{i,k}$ $k = 1, 2, 3, \ldots, X_i$ $X_i = N_i$

4.2.7.3.2 Bit collection

The bits x_{bik} are input to the rate matching algorithm described in section 4.2.7.5. The bits output from the rate matching algorithm are denoted $y_{bi1}, y_{bi2}, y_{bi3}, \ldots, y_{biY_i}$.

Bit collection is the inverse function of the separation. The bits after collection are denoted by $z_{bi1}, z_{bi2}, z_{bi3}, \ldots, z_{biY_i}$. After bit collection, the bits indicated as punctured are removed and the bits are then denoted by $f_{i1}, f_{i2}, f_{i3}, \ldots, f_{iV_i}$, where i is the TrCH number and $V_i = N_{ij} + \Delta N_{ij}$. The relations between y_{bik}, z_{bik}, and f_{ik} are given below.

For turbo encoded TrCHs with puncturing ($Y_i = X_i$):

$z_{i,3(k-1)+1+(\alpha_1+\beta_{n_i}) \bmod 3} = y_{1,i,k}$ $k = 1, 2, 3, \ldots, Y_i$

$z_{i,3\lfloor N_i/3 \rfloor + k} = y_{1,i,\lfloor N_i/3 \rfloor + k}$ $k = 1, \ldots, N_i \bmod 3$ Note: When (N_i mod 3) = 0 this row is not needed.

$z_{i,3(k-1)+1+(\alpha_2+\beta_{n_i}) \bmod 3} = y_{2,i,k}$ $k = 1, 2, 3, \ldots, Y_i$

$z_{i,3(k-1)+1+(\alpha_3+\beta_{n_i}) \bmod 3} = y_{3,i,k}$ $k = 1, 2, 3, \ldots, Y_i$

After the bit collection, bits $z_{i,k}$ with value δ, where $\delta \notin \{0, 1\}$, are removed from the bit sequence. Bit $f_{i,1}$ corresponds to the bit $z_{i,k}$ with smallest index k after puncturing, bit $f_{i,2}$ corresponds to the bit $z_{i,k}$ with second smallest index k after puncturing, and so on.

For uncoded TrCHs, convolutionally encoded TrCHs, and turbo encoded TrCHs with repetition:

$z_{i,k} = y_{1,i,k}$ $k = 1, 2, 3, \ldots, Y_i$

When repetition is used, $f_{i,k} = z_{i,k}$ and $Y_i = V_i$.

When puncturing is used, $Y_i = X_i$ and bits $z_{i,k}$ with value δ, where $\delta \notin \{0, 1\}$, are removed from the bit sequence. Bit $f_{i,1}$ corresponds to the bit $z_{i,k}$ with smallest index k after puncturing, bit $f_{i,2}$ corresponds to the bit $z_{i,k}$ with second smallest index k after puncturing, and so on.

4.2.7.4 Bit separation and collection in downlink

The systematic bits (excluding bits for trellis termination) of turbo encoded TrCHs shall not be punctured. The systematic bit, first parity bit, and second parity bit in the bit sequence input to the rate matching block are therefore separated from each other. Puncturing is only applied to the parity bits and systematic bits used for trellis termination.

The bit separation function is transparent for uncoded TrCHs, convolutionally encoded TrCHs, and for turbo encoded TrCHs with repetition. The bit separation and bit collection are illustrated in figures 8 and 9.

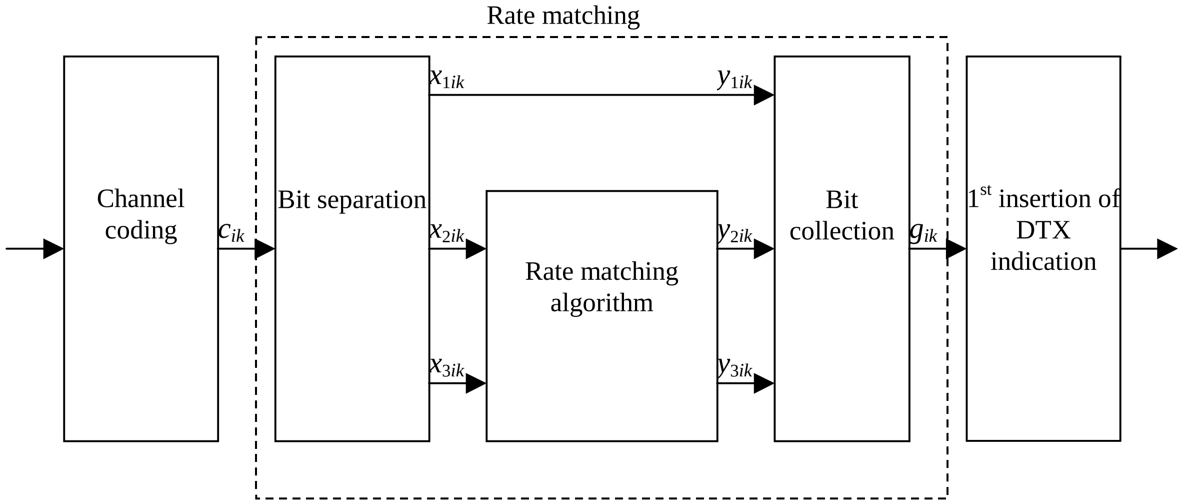


Figure 8: Puncturing of turbo encoded TrCHs

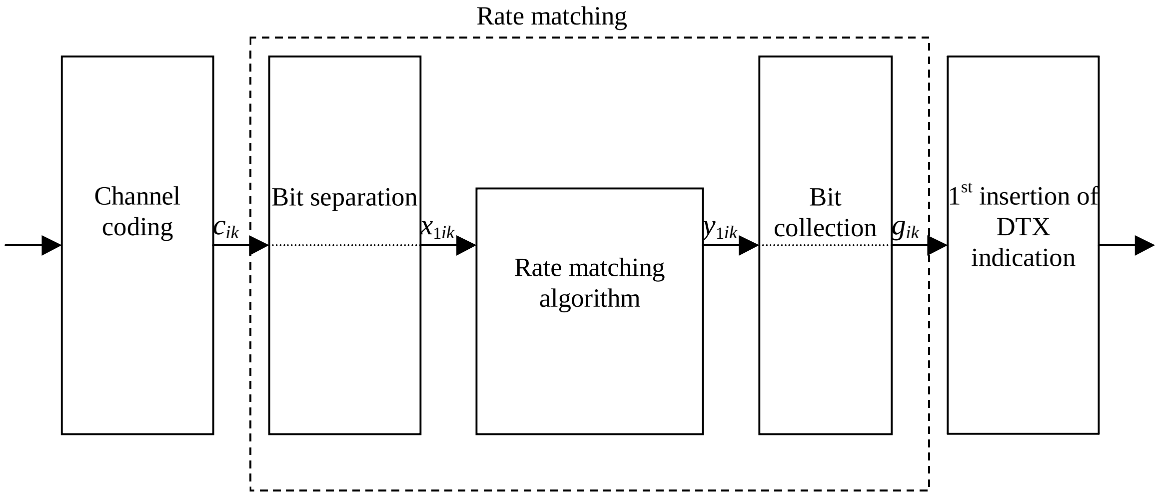


Figure 9: Rate matching for uncoded TrCHs, convolutionally encoded TrCHs, and for turbo encoded TrCHs with repetition.

4.2.7.4.1 Bit separation

The bits input to the rate matching are denoted by $c_{i1}, c_{i2}, c_{i3}, \ldots, c_{iE_i}$, where i is the TrCH number and E_i is the number of bits input to the rate matching block. Note that E_i is a multiple of 3 for turbo encoded TrCHs and that the transport format combination number j for simplicity has been left out in the bit numbering, i.e. $E_i=N_{ij}$. The bits after separation are denoted by $x_{bi1}, x_{bi2}, x_{bi3}, \ldots, x_{biX_i}$. For turbo encoded TrCHs with puncturing, b indicates systematic, first parity, or second parity bit. For all other cases b is defined to be 1. X_i is the number of bits in each separated bit sequence. The relation between c_{ik} and x_{bik} is given below.

For turbo encoded TrCHs with puncturing:

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

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

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

For uncoded TrCHs, convolutionally encoded TrCHs, and turbo encoded TrCHs with repetition:

$x_{1,i,k} = c_{i,k}$ $k = 1, 2, 3, \ldots, X_i$ $X_i = E_i$

4.2.7.4.2 Bit collection

The bits x_{bik} are input to the rate matching algorithm described in section 4.2.7.5. The bits output from the rate matching algorithm are denoted $y_{bi1}, y_{bi2}, y_{bi3}, \ldots, y_{biY_i}$.

Bit collection is the inverse function of the separation. The bits after collection are denoted by $z_{bi1}, z_{bi2}, z_{bi3}, \ldots, z_{biY_i}$. After bit collection, the bits indicated as punctured are removed and the bits are then denoted by $g_{i1}, g_{i2}, g_{i3}, \ldots, g_{iG_i}$, where *i* is the TrCH number and $G_i = N_{ij} + \Delta N_{ij}$. The relations between y_{bik}, z_{bik}, and g_{ik} are given below.

For turbo encoded TrCHs with puncturing ($Y_i=X_i$):

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

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

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

After the bit collection, bits $z_{i,k}$ with value δ, where $\delta \notin \{0, 1\}$, are removed from the bit sequence. Bit $g_{i,1}$ corresponds to the bit $z_{i,k}$ with smallest index *k* after puncturing, bit $g_{i,2}$ corresponds to the bit $z_{i,k}$ with second smallest index *k* after puncturing, and so on.

For uncoded TrCHs, convolutionally encoded TrCHs, and turbo encoded TrCHs with repetition:

$z_{i,k} = y_{1,i,k}$ $\quad k = 1, 2, 3, \ldots, Y_i$

When repetition is used, $g_{i,k}=z_{i,k}$ and $Y_i=G_i$.

When puncturing is used, $Y_i=X_i$ and bits $z_{i,k}$ with value δ, where $\delta \notin \{0, 1\}$, are removed from the bit sequence. Bit $g_{i,1}$ corresponds to the bit $z_{i,k}$ with smallest index *k* after puncturing, bit $g_{i,2}$ corresponds to the bit $z_{i,k}$ with second smallest index *k* after puncturing, and so on.

4.2.7.5 Rate matching pattern determination

Denote the bits before rate matching by:

$x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iX_i}$, where *i* is the TrCH number and X_i is the parameter given in sections 4.2.7.1 and 4.2.7.2.

The rate matching rule is as follows:

```
if puncturing is to be performed
    e = e_ini          -- initial error between current and desired puncturing ratio
    m = 1              -- index of current bit
    do while m <= X_i
        e = e – e_minus          -- update error
        if e <= 0 then           -- check if bit number m should be punctured
            set bit x_i,m to δ where δ∉{0, 1}
            e = e + e_plus   -- update error
        end if
        m = m + 1                --  next bit
    end do
```

```
else
    e = e_ini            -- initial error between current and desired puncturing ratio
    m = 1                -- index of current bit
    do while m <= X_i
        e = e – e_minus      -- update error
        do while e <= 0      -- check if bit number m should be repeated
            repeat bit x_i,m
            e = e + e_plus  -- update error
        end do
        m = m + 1            -- next bit
    end do
end if
```

A repeated bit is placed directly after the original one.

4.2.8 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:

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

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

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

…

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

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 1st insertion of DTX indication bits

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 in one radio frame of TrCH *i* by H_i. In normal or compressed mode by spreading factor reduction, H_i is constant and corresponds to the maximum number of bits from TrCH *i* in one radio frame for any transport format of TrCH *i*. In compressed mode by higher layer scheduling, only a subset of the TFC Set is allowed. From this subset it is possible to derive which TFs on each TrCH that are allowed. The maximum number of bits belonging to one TTI of TrCH *i* for the allowed TFs is denoted by X_i. H_i is then calculated as $H_i = \left\lceil \frac{X_i}{F_i} \right\rceil$, where F_i is the number of radio frames in a TTI of TrCH *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:

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

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

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

4.2.9.2 2nd insertion of DTX indication bits

The DTX indication bits inserted in this step 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 number of available bits on the PhCH is denoted by N_{data} and $N_{data}=15N_{data1}+15N_{data2}$, where N_{data1} and N_{data2} are defined in [25.211]. In normal mode $U=N_{data}$. In compressed mode N_{data} is changed from the value in normal node. The exact value of N_{data} is dependent on the *TGL* and the transmission time reduction method, which are signalled from higher layers. The number of bits that are located within the transmission gap is denoted N_{TGL} and defined as:

$$N_{TGL} = \begin{cases} \frac{TGL}{15} N_{data}, & \text{if } N_{first} + TGL \le 15 \\ \frac{15 - N_{first}}{15} N_{data}, & \text{in first frame if } N_{first} + TGL > 15 \\ \frac{TGL - (15 - N_{first})}{15} N_{data}, & \text{in second frame if } N_{first} + TGL > 15 \end{cases}$$

N_{first} and *TGL* are defined in Section 4.4.

In compressed mode $U=N_{data}-N_{TGL}$..

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, …, S

$w_k = \delta$ k = S+1, S+2, S+3, …, PU

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

4.2.10 Physical channel segmentation

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:

$u_{1k} = x_k$ $k = 1, 2, \ldots, U$

Bits on second PhCH after physical channel segmentation:

$u_{2k} = x_{(k+U)}$ $k = 1, 2, \ldots, U$

…

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

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

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

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 $U \le R_2 C_2$.

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

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

(4) Perform the inter-column permutation based on the pattern $\{P_2(j)\}$ $(j = 0, 1, \ldots, C_2-1)$ 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.

In compressed mode, no bits are mapped to certain slots of the PhCH(s). If $N_{first} + TGL \leq 15$, no bits are mapped to slots N_{first} to N_{last}. If $N_{first} + TGL > 15$, i.e. the transmission gap spans two consecutive radio frames, the mapping is as follows:

- In the first radio frame, no bits are mapped to slots N_{first}, $N_{first}+1$, $N_{first}+2$, …, 14.
- In the second radio frame, no bits are mapped to the slots 0, 1, 2, …, N_{last}.

TGL, N_{first}, and N_{last} are defined in section 4.4.

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. The only exception is when the UE is in compressed mode. The transmission can then be turned off during consecutive slots of the radio frame.

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.

The following rules should be used for the selection of fixed or flexible positions of the TrCHs in the radio frame:

- 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[11]. 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 either 10 ms or 20 ms.
- 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: 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.
- 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 shall have co-ordinated timings. When the TFCS of a CCTrCH is changed because a transport channel *i* is added to the CCTrCH or reconfigured within the CCTrCH, the TTI of transport channel *i* may only start in radio frames with CFN fulfilling the relation

$CFN_i \bmod F_{max} = 0$,

where F_{max} denotes the maximum number of radio frames within the transmission time intervals of all transport channels which are multiplexed into the same CCTrCH, including transport channel *i* which is added or reconfigured, and CFN_i denotes the connection frame number of the first radio frame within the transmission time interval of transport channel *i*.

After addition or reconfiguration of a transport channel *i* within a CCTrCH, the TTI of transport channel *i* may only start in radio frames with CFN fulfilling the relation

$CFN_i \bmod F_i = 0$.

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.

4.3 Transport format detection

Transport format detection can be performed both with and without Transport 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 may be 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.

4.3.1 Blind transport format detection

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

4.3.2 Explicit transport format detection based on 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.

4.3.3 Coding of Transport-Format-Combination Indicator (TFCI)

The TFCI bits are encoded using a (32, 10) sub-code of the second order Reed-Muller code. The coding procedure is as shown in figure 10.

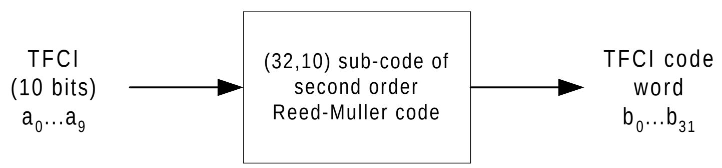


Figure 10: Channel coding of TFCI bits

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 length of the TFCI code word is 32 bits.

The code words of the (32,10) sub-code of second order Reed-Muller code are linear combination of 10 basis sequences. The basis sequences are as in the following table 7.

Table 7: Basis sequences for (32,10) TFCI code

i	$M_{i,0}$	$M_{i,1}$	$M_{i,2}$	$M_{i,3}$	$M_{i,4}$	$M_{i,5}$	$M_{i,6}$	$M_{i,7}$	$M_{i,8}$	$M_{i,9}$
0	1	1	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	1	0	0	0
2	1	1	1	0	0	0	0	0	0	1
3	1	0	0	1	0	0	1	0	1	1
4	1	1	0	1	0	0	0	0	0	1
5	1	0	1	1	0	0	0	0	1	0
6	1	1	1	1	0	0	0	1	0	0
7	1	0	0	0	1	0	0	1	1	0
8	1	1	0	0	1	0	1	1	1	0
9	1	0	1	0	1	0	1	0	1	1
10	1	1	1	0	1	0	0	0	1	1
11	1	0	0	1	1	0	0	1	1	0
12	1	1	0	1	1	0	0	1	0	1
13	1	0	1	1	1	0	1	0	0	1
14	1	1	1	1	1	0	1	1	1	1
15	1	1	0	0	0	1	1	1	0	0
16	1	0	1	0	0	1	1	1	0	1
17	1	1	1	0	0	1	1	0	1	0
18	1	0	0	1	0	1	0	1	1	1
19	1	1	0	1	0	1	0	1	0	1
20	1	0	1	1	0	1	0	0	1	1
21	1	1	1	1	0	1	0	1	1	1
22	1	0	0	0	1	1	0	1	0	0
23	1	1	0	0	1	1	1	1	0	1
24	1	0	1	0	1	1	1	0	1	0
25	1	1	1	0	1	1	1	0	0	1
26	1	0	0	1	1	1	0	0	1	0
27	1	1	0	1	1	1	1	1	0	0
28	1	0	1	1	1	1	1	1	1	0
29	1	1	1	1	1	1	1	1	1	1
30	1	0	0	0	0	0	0	0	0	0
31	1	0	0	0	0	1	1	0	0	0

For TFCI 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 output code word bits b_i are given by:

$$b_i = \sum_{n=0}^{9} (a_n \times M_{i,n}) \bmod 2$$

where i=0…31.

The output bits are denoted by b_k, k = 0, 1, 2, …, 31.

In downlink, when the SF <128 the encoded TFCI code words are repeated yielding 8 encoded TFCI bits per slot in normal mode and 16 encoded TFCI bits per slot in compressed mode. Mapping of repeated bits to slots is explained in section 4.3.5.

4.3.4 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.

The TFCI bits are encoded using a (16, 5) bi-orthogonal (or first order Reed-Muller) code. The coding procedure is as shown in figure 11.

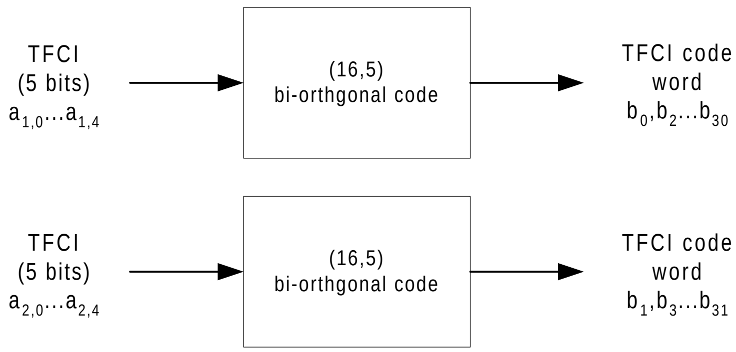


Figure 11: Channel coding of split mode TFCI bits

The code words of the (16,5) bi-orthogonal code are linear combinations of 5 basis sequences as defined in table 8 below.

Table 8: Basis sequences for (16,5) TFCI code

i	$M_{i,0}$	$M_{i,1}$	$M_{i,2}$	$M_{i,3}$	$M_{i,4}$
0	1	1	0	0	0
1	1	0	1	0	0
2	1	1	1	0	0
3	1	0	0	1	0
4	1	1	0	1	0
5	1	0	1	1	0
6	1	1	1	1	0
7	1	0	0	0	1
8	1	1	0	0	1
9	1	0	1	0	1
10	1	1	1	0	1
11	1	0	0	1	1
12	1	1	0	1	1
13	1	0	1	1	1
14	1	1	1	1	1
15	1	0	0	0	0

For TFCI information bits for DCH $a_{1,0}$, $a_{1,1}$, $a_{1,2}$, $a_{1,3}$, $a_{1,4}$ ($a_{1,0}$ is LSB and $a_{1,4}$ is MSB) and for DSCH $a_{2,0}$, $a_{2,1}$, $a_{2,2}$, $a_{2,3}$, $a_{2,4}$ ($a_{2,0}$ is LSB and $a_{2,4}$ is MSB), the output code word bits b_0, b_1, ..., b_{31}, are given by:

$$b_{2i} = \sum_{n=0}^{4} (a_{1,n} \times M_{i,n}) \bmod 2 ; \qquad b_{2i+1} = \sum_{n=0}^{4} (a_{2,n} \times M_{i,n}) \bmod 2$$

where i=0…15, j=0,1.

The output bits are denoted by b_k, k = 0, 1, 2, …, 31.

4.3.5 Mapping of TFCI words

4.3.5.1 Mapping of TFCI word in non compressed mode

The bits of the code word are directly mapped to the slots of the radio frame. Within a slot the bit with lower index is transmitted before the bit with higher index. The coded bits b_k, are mapped to the transmitted TFCI bits d_k, according to the following formula:

$d_k = b_{k \bmod 32}$

For uplink physical channels regardless of the SF and downlink physical channels, if SF≥128, k = 0, 1, 2, …, 29. Note that this means that bits b_{30} and b_{31} are not transmitted.

For downlink physical channels whose SF<128, k = 0, 1, 2, …, 119. Note that this means that bits b_0 to b_{23} are transmitted four times and bits b_{24} to b_{31} are transmitted three times.

4.3.5.2 Mapping of TFCI in compressed mode

The mapping of the TFCI bits in compressed mode is different for uplink, downlink with SF≥128 and downlink with SF<128.

4.3.5.2.1 Uplink compressed mode

For uplink compressed mode, the slot format is changed so that no TFCI bits are lost. The different slot formats in compressed mode do not match the exact number of TFCI bits for all possible TGLs. Repetition of the TFCI bits is therefore used.

Denote the number of bits available in the TFCI fields of one compressed radio frame by D and the number of bits in the TFCI field in a slot by N_{TFCI}. Denote by E the first bit to be repeated, $E=N_{first}N_{TFCI}$. If $N_{last}\neq14$, then E corresponds to the number of the first TFCI bit in the slot directly after the TG. The following relations then define the mapping.

$d_k = b_{k \bmod 32}$

where k = 0, 1, 2, …, min (31, D-1).

If D > 32, the remaining positions are filled by repetition (in reversed order):

$d_{D-k-1} = b_{(E+k) \bmod 32}$

where k = 0, …, D-33.

4.3.5.2.2 Downlink compressed mode

For downlink compressed mode, the slot format is changed so that no TFCI bits are lost. The different slot formats in compressed mode do not match the exact number of TFCI bits for all possible TGLs. DTX is therefore used if the number of TFCI fields exceeds the number of TFCI bits. The block of fields, where DTX is used, starts on the first field after the gap. If there are fewer TFCI fields after the gap than DTX bits, the last fields before of the gap are also filled with DTX.

Denote the number of bits available in the TFCI fields of one compressed radio frame by D and the number of bits in the TFCI field in a slot by N_{TFCI}. Denote by E the first bit to be repeated, $E=N_{first}N_{TFCI}$. If $N_{last}\neq14$, then E corresponds to the number of the first TFCI bit in the slot directly after the TG. Denote the total number of TFCI bits to be transmitted by N_{tot}. If SF ≥ 128 then N_{tot} = 32, else N_{tot} = 128. The following relations then define the mapping:

$d_k = b_{(k \bmod 32)}$

where k = 0, 1, 2, …, min (E, N_{tot})-1 and, if E< N_{tot},

$d_{k+D-Ntot} = b_{(k \bmod 32)}$

where k = E, ..., N_{tot} -1.

DTX bits are sent on d_k where k = min (E, N_{tot}), ..., min (E, N_{tot}) +D - N_{tot} -1.

4.4 Compressed mode

In compressed mode, slots N_{first} to N_{last} are not used for transmission of data. As illustrated in figure 12, which shows the example of fixed transmission gap position with single frame method, 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 4.4.3). What frames are compressed, are decided by the network. When in compressed mode, compressed frames can occur periodically, as illustrated in figure 12, or requested on demand. The rate and type of compressed frames is variable and depends on the environment and the measurement requirements.

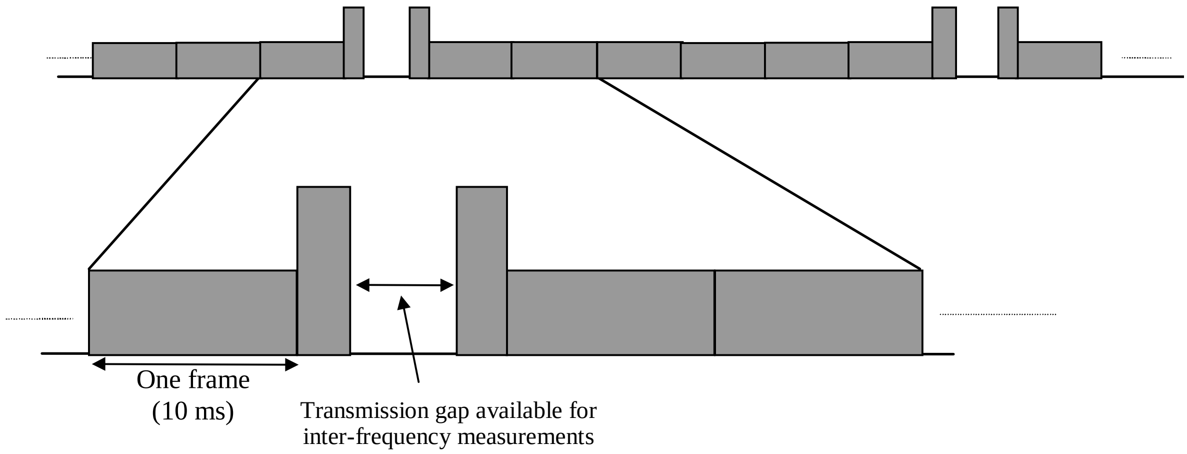


Figure 12: Compressed mode transmission

4.4.1 Frame structure in the uplink

The frame structure for uplink compressed mode is illustrated in figure 13.

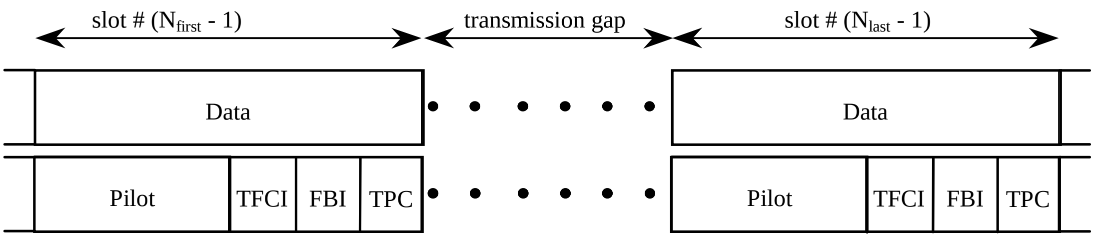


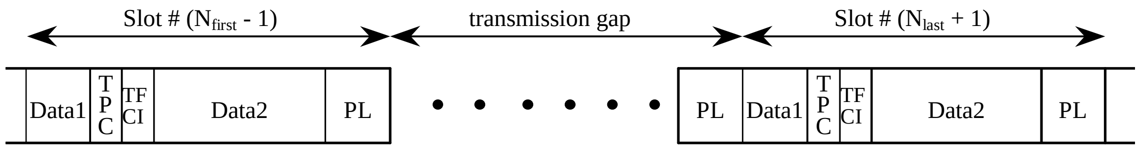
Figure 13: Frame structure in uplink compressed transmission

4.4.2 Frame structure types in the downlink

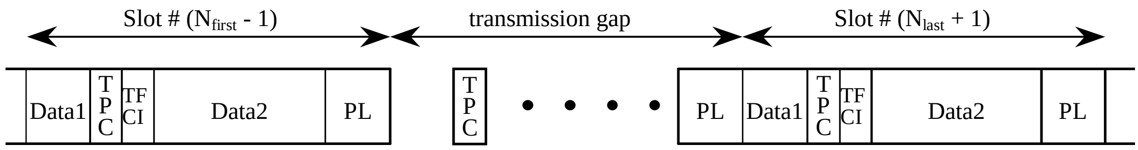
There are two different types of frame structures defined for downlink compressed mode. Type A maximises the transmission gap length and type B is optimised for power control.

- With frame structure of type A, the pilot field of the last slot in the transmission gap is transmitted. Transmission is turned off during the rest of the transmission gap (figure 14(a)).

- With frame structure of type B, the TPC field of the first slot in the transmission gap and the pilot field of the last slot in the transmission gap is transmitted. Transmission is turned off during the rest of the transmission gap (figure 14(b)).



(a) Frame structure type A



(b) Frame structure type B

Figure 14: Frame structure types in downlink compressed transmission

4.4.3 Transmission time reduction method

When in compressed mode, the information normally transmitted during a 10 ms frame is compressed in time. The mechanisms provided for achieving this are puncturing, reduction of the spreading factor by a factor of two , and higher layer scheduling. In the downlink, all methods are supported while compressed mode by puncturing is not used in the uplink. The maximum idle length is defined to be 7 slots per one 10 ms frame. The slot formats that are used in compressed mode are listed in [2].

4.4.3.1 Compressed mode by puncturing

During compressed mode, rate matching (puncturing) is applied for creating transmission gap in one frame. The algorithm for rate matching (puncturing) as described in section 4.2.7 is used.

4.4.3.2 Compressed mode by reducing the spreading factor by 2

During compressed mode, the spreading factor (SF) can be reduced by 2 during one radio frame to enable the transmission of the information bits in the remaining time slots of a compressed frame.

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[3] section 5.2.1.

4.4.3.3 Compressed mode by higher layer scheduling

Compressed mode can be obtained by higher layer scheduling. Higher layers then set restrictions so that only a subset of the allowed TFCs are used in compressed mode. The maximum number of bits that will be delivered to the physical layer during the compressed radio frame is then known and a transmission gap can be generated.

4.4.4 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.

4.4.4.1 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 (TGL) as shown in figure 15 (1). When using double frame method, the fixed transmission gap is located on the center of two connected frames as shown in figure 15 (2). Table 9 shows the parameters for the fixed transmission gap position case.

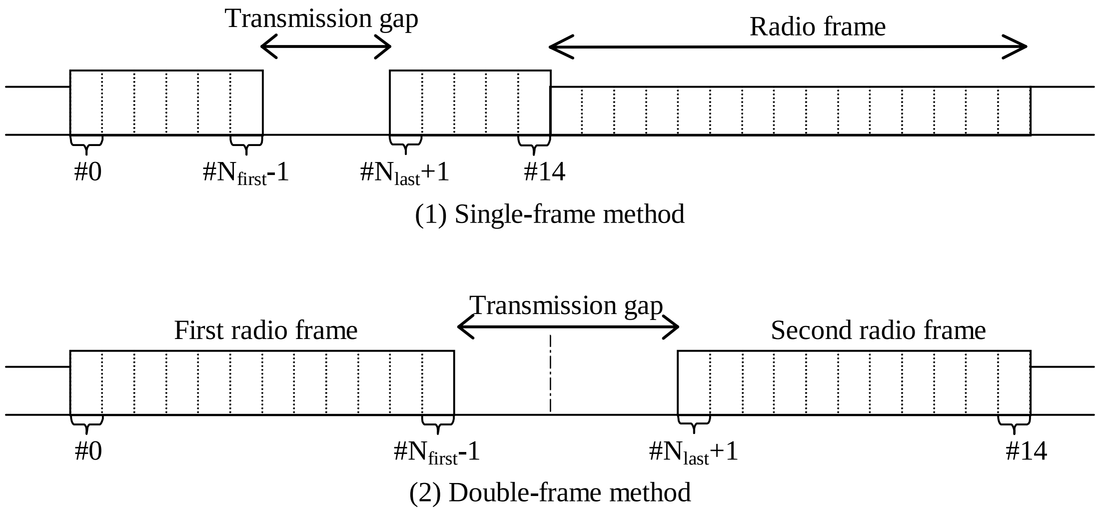


Figure 15: Fixed transmission gap position

Table 9: Parameters for fixed transmission gap position

	Single-frame method		Double-frame method	
TGL (slot)	N_{first}	N_{last}	N_{first}	N_{last}
3	7	9	14 in first frame	1 in second frame
4	6	9	13 in first frame	1 in second frame
7	6	12	12 in first frame	3 in second frame
10	N.A.	N.A.	10 in first frame	4 in second frame
14	N.A.	N.A.	8 in first frame	6 in second frame

4.4.4.2 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 16. Parameters of the adjustable transmission gap positions are calculated as follows:

TGL is the number of consecutive idle slots during compressed mode,

TGL = 3, 4, 7, 10, 14

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

N_{first} = 0,1,2,3,…,14.

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

If $N_{first} + TGL \leq 15$, then $N_{last} = N_{first} + TGL - 1$ (in the same frame),

If $N_{first} + TGL > 15$, then $N_{last} = (N_{first} + TGL - 1) \bmod 15$ (in the next frame).

When the transmission gap spans two consecutive radio frames, N_{first} and TGL must be chosen so that at least 8 slots in each radio frame are transmitted.

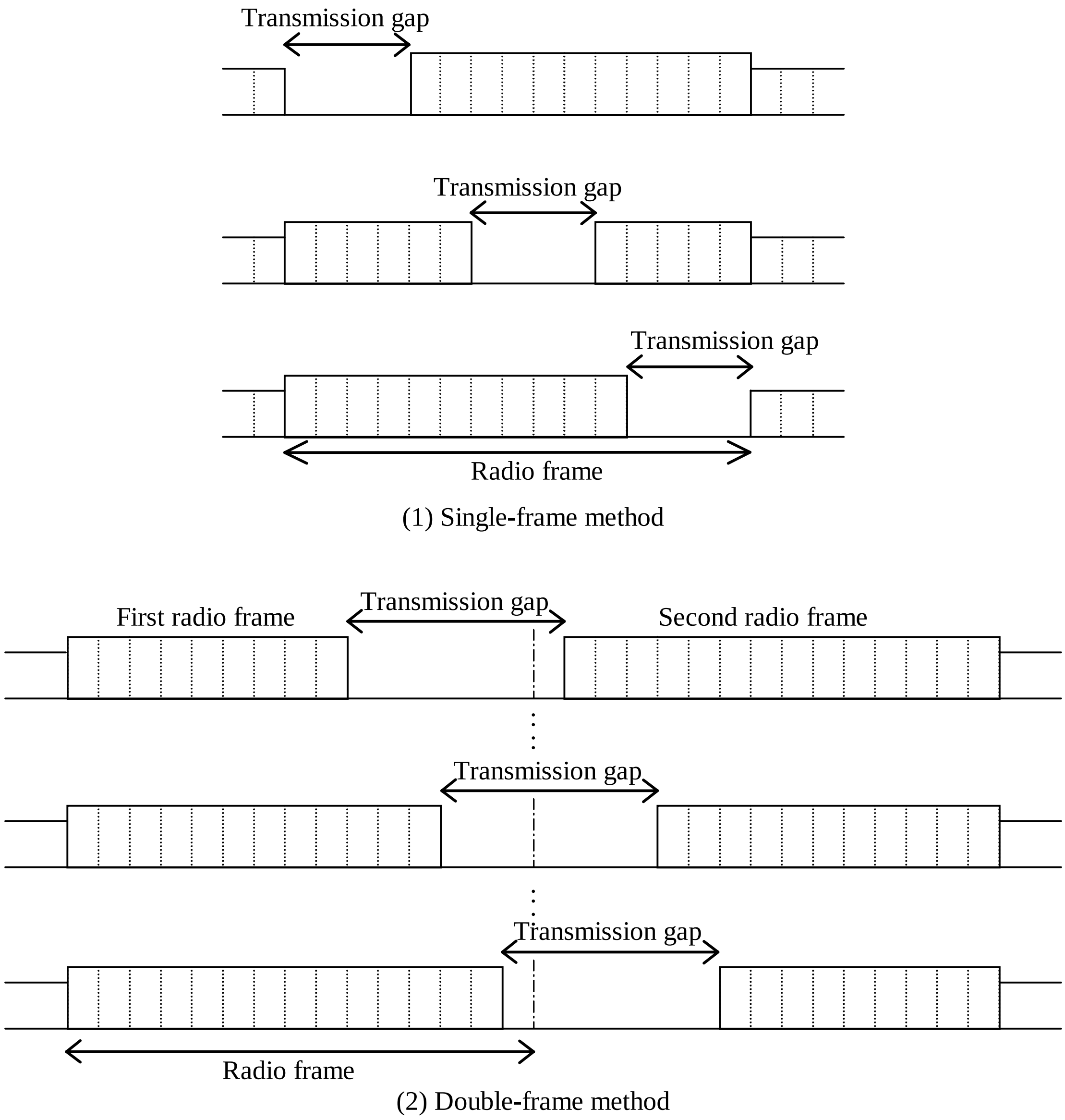


Figure 16: Adjustable transmission gap lengths position

4.4.4.3 Parameters for downlink compressed mode

Table 10 shows the detailed parameters for each transmission gap length for the different transmission time reduction methods.

Table 10: Parameters for compressed mode

TGL	Type	Adjustable /fixed gap position	Spreading Factor	Idle length[ms]	Transmission time Reduction method	Idle frame Combining
3	A	Adjustable Or Fixed	512 – 4	1.73-1.99	Puncturing Spreading factor reduction by 2 Higher layer scheduling	(S) (D) =(1,2),(2,1)
	B		256- 4	1.60-1.86		
4	A		512 - 4	2.40-2.66		(S) (D) =(1,3),(2,2),(3,1)
	B		256- 4	2.27-2.53		
7	A		512 -4	4.40-4.66		(S) (D)=(1,6),(2,5),(3,4),(4,3),(5,2),(6,1)
	B		256- 4	4.27-4.53		
10	A		512 - 4	6.40-6.66		(D)=(3,7),(4,6),(5,5),(6,4),(7,3)
	B		256- 4	6.27-6.53		
14	A	Fixed	512 - 4	9.07-9.33		(D) =(7,7)
	B		256- 4	8.93-9.19		

(S): Single-frame method as shown in figure 15 (1).

(D): Double-frame method as shown in figure 15 (2). (x,y) indicates x: the number of idle slots in the first frame, y: the number of idle slots in the second frame.

NOTE: Compressed mode by spreading factor reduction is not supported when SF=4 is used in normal mode.

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). When this method is used, no one transport format should have the same number of bits as any other transport format does within a TrCH.
- At the transmitter, the data stream with variable number of bits from higher layers is block-encoded using a cyclic redundancy check (CRC) and then convolutionally encoded. CRC parity bits are attached just after the data stream with variable number of bits 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:

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

  where $a_{max}(n_{end})$ and $a_{min}(n_{end})$ are, respectively, the maximum and minimum path-metric values 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

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

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.

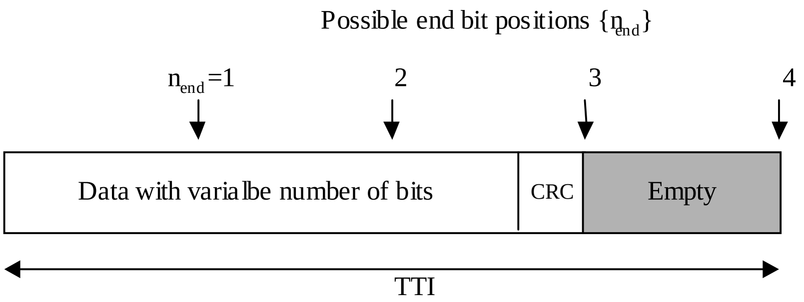


Figure A-1: An example of the data format with variable number of bits

(Number of possible transport formats = 4, transmitted end bit position n_{end} = 3)

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.

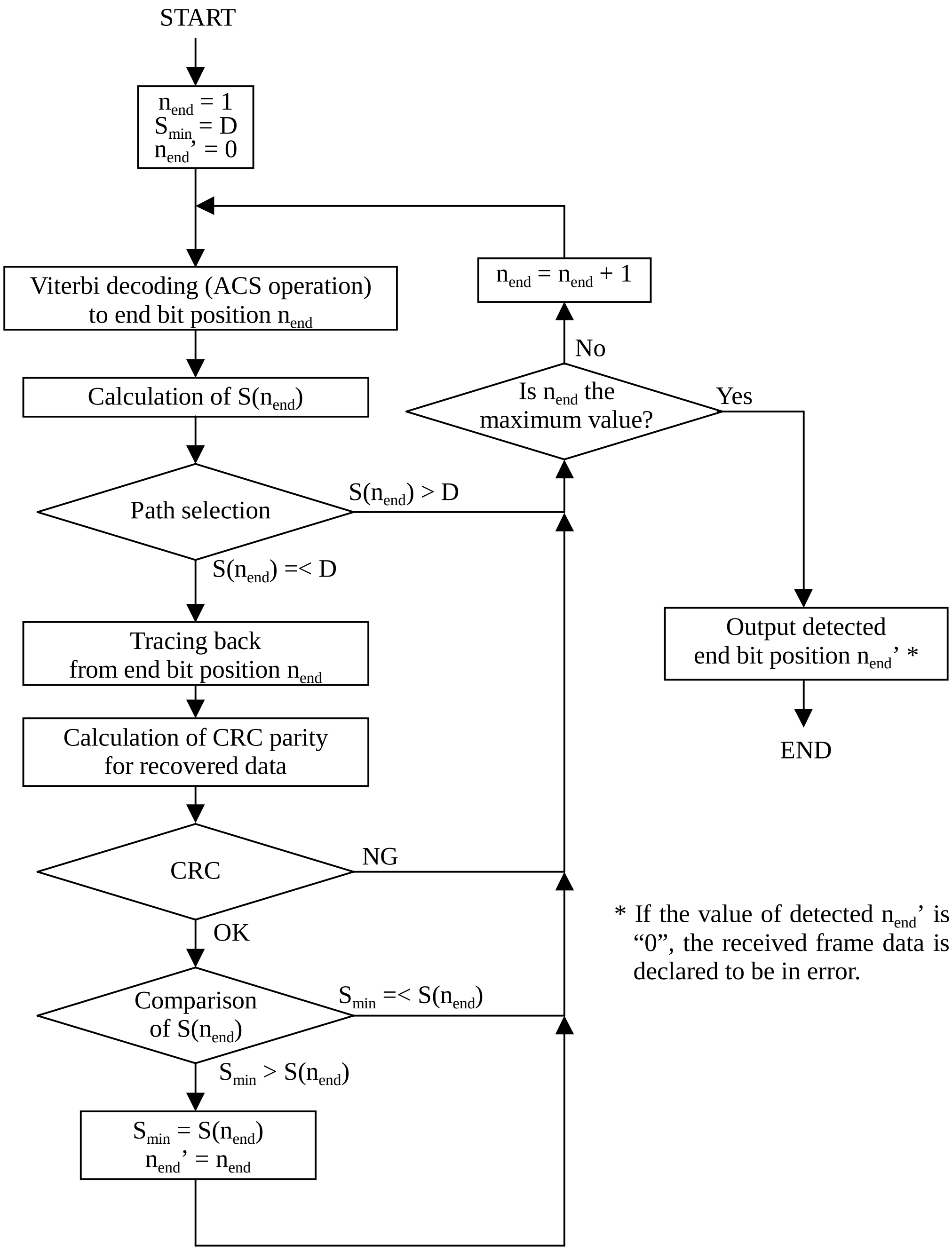


Figure A-2: Basic processing flow of blind transport format detection

Annex B (informative): Change history

Change history					
TSG RAN#	Version	CR	Tdoc RAN	New Version	Subject/Comment
RAN_05	-	-	RP-99588	3.0.0	Approved at TSG RAN #5 and placed under Change Control
RAN_06	3.0.0	001	RP-99680	3.1.0	Correction of rate matching parameters for repetition after 1st unterleaving in 25.212
RAN_06	3.0.0	004	RP-99680	3.1.0	Changing the initial offset value for convolutional code rate matching
RAN_06	3.0.0	005	RP-99681	3.1.0	Introduction of compressed mode by higher layer scheduling
RAN_06	3.0.0	008	RP-99679	3.1.0	Editorial corrections to TS 25.212
RAN_06	3.0.0	009	RP-99680	3.1.0	Removal of SFN multiplexing
RAN_06	3.0.0	010	RP-99680	3.1.0	Clarification of bit separation and collection
RAN_06	3.0.0	011	RP-99680	3.1.0	Connection between TTI and CFN
RAN_06	3.0.0	012	RP-99680	3.1.0	Zero length transport blocks
RAN_06	3.0.0	014	RP-99679	3.1.0	Update of channel coding sections
RAN_06	3.0.0	016	RP-99680	3.1.0	Removal of TrCH restriction in DSCH CCTrCH
RAN_06	3.0.0	017	RP-99681	3.1.0	20 ms RACH message length
RAN_06	3.0.0	018	RP-99680	3.1.0	Minimum SF in UL
RAN_06	3.0.0	024	RP-99680	3.1.0	Rate matching parameter determination in DL and fixed positions
RAN_06	3.0.0	026	RP-99680	3.1.0	Corrections to TS 25.212
RAN_06	3.0.0	027	RP-99679	3.1.0	Modification of BTFD description in 25.212 Annex
RAN_06	3.0.0	028	RP-99681	3.1.0	TFCI coding and mapping including compressed mode
-	3.1.0	-	-	3.1.1	Change history was added by the editor

History

Document history		
V3.1.1	January 2000	Publication