RAN WG1 meeting #7 **TSGR1#7(99) a80**
Place : Hanover
Date : August 30th to September 3rd, 1999
Title : Text proposal for DL rate matching signalling
Source : Mitsubishi Electric (MCRD)
Paper for : Decision

1 Introduction

In [2] a rule has been proposed to compute the rate matching ratios. However this rule raises several problems for the DL direction, and this is why it was accepted in meeting #6 only as a working assumption and with a WG1 note mentioning the problem.
This contribution first lists several of these problems, then proposes a solution, and finally makes a text proposal.

This contribution also handles a problem connected with setting initial error value in the rate matching pattern algorithm.

2 References

[1] 3GPP TSG RAN WG1 R1-99A86 TS 25.212 V2.0.1: 'Multiplexing and channel coding (FDD)', Source: Editor

[2] 3GPP TSG RAN WG1 R(99)612: 'Text proposal for rate matching signalling' , Source: Ericsson, Mitsubishi ELectric, Siemens,

3 Abbreviations :

TFC : Transport Format Combination
TrCH : Transport Channel
TF : Transport Format

4 Problems with current rule

In a nutshell the current rule consists for flexible position of services in :

- finding the TFC *L* with greatest CCTrCH bit rate
- computing the ΔN offsets for this TFC *L*
- scaling the ΔN offsets computed for TFC *L*, in order to get the ΔN offsets for other TFCs.

Most problems in the current rule are related to the case when several TrCHs have not independent TFs, notably when the TrCHs never have simultaneously their respective maximum bit rate. Other problems are related to this that in DL rate matching is operated on a TTI, that is to say for a TF, and not on a radio frame, that is to say for a TFC.
The case of TrCHs with mutually dependant bit rates typically happens if some arbitration is done at the MAC layer so that the TrCH with highest priority can transmit before the other TrCHs : this is useful if the bit rate N_{data} available from the physical channel(s) does not suffice for all the TrCH to transmit simultaneously at maximum bit rate. So when the TrCH with highest priority is at maximum bit rate, the other TrCH(s) cannot be also at maximum bit rate.

4.1 Problem of definition of the TFC with highest CCTrCH bit rate

The first step of the current rule is to find a TFC *L* such that the bit rate of the CCTrCH be maximum for this TFC *L*. Such a step is at best incompletely defined. First of all the bit rate of the CCTrCH is known for each TFC only once that all the values of the $\Delta N_{i,l}^{TTI}$ have been computed, so the bit rate of the CCTrCH being a function of the $\Delta N_{i,l}^{TTI}$ cannot be used in the process of computing the $\Delta N_{i,l}^{TTI}$ (this is like the problem of getting the egg form the hen and vice versa). Second, even if some estimate of the CCTrCH bit rate not using the $\Delta N_{i,l}^{TTI}$ was defined, nevertheless the TFC yielding the highest bit rate might be not unique, notably in the case when, whereas the CCTrCH has a maximum bit rate, not all the TrCHs have simultaneously a maximum bit rate.

4.2 Problems of definition of ΔN_{ij} when several TrCH have mutually dependant bit rates

Now let us consider the following formula in use in the current rule :

$$\Delta N_{ij} = \left\lfloor \frac{\Delta N_{iL}}{N_{iL}} \cdot N_{ij} \right\rfloor$$

In this formula it is possible that both ΔN_{iL} and N_{iL} be simultaneously null, because for instance TrCH *i* has a low priority, and so has a null bit rate when the highest priority TrCH make the CCTrCH reach its highest bit rate in TFC = *L*. So if both ΔN_{iL} and N_{iL} are null for some TrCH *i* the formula becomes :

$$\Delta N_{ij} = \left\lfloor \frac{0}{0} \cdot N_{ij} \right\rfloor$$

and we have a $\frac{0}{0}$ indetermination when computing ΔN_{ij} for the other TFC *j* for the TrCH *i*.

More generally, if in the TFC *L*, the TrCH *i* have a bit rate that is significantly less than in the TFC *j*, we have $N_{iL} << N_{ij}$, so the rounding error on ΔN_{iL} is amplified by a factor $\frac{N_{ij}}{N_{iL}} >> 1$ in the formula

$$\Delta N_{ij} = \left\lfloor \left(\frac{N_{ij}}{N_{iL}} \right) \cdot \Delta N_{iL} \right\rfloor.$$

4.3 Problems related to use of radio frame instead of TTI

Finally, the formula $\Delta N_{ij} = \left\lfloor \frac{\Delta N_{iL}}{N_{iL}} \cdot N_{ij} \right\rfloor$ gives the value of ΔN_{ij}, for a radio frame (TFC), and not for a TTI (TF), so in the current rule the following scaling is done :

$$l = TF_i(j) \quad \text{and} \quad \Delta N_{il}^{TTI} = F_i \Delta N_{ij}$$

This scaling rule is not completely defined, as there can be two distinct TFC j and j' for which the TF of TrCH i is the same TF l. That is to say if $l = TF_i(j) = TF_i(j')$ then we don't know whether

$$\Delta N_{il}^{TTI} = F_i \Delta N_{ij} \text{ or } \Delta N_{il}^{TTI} = F_i \Delta N_{ij'}$$

4.4 Problems related to the use of fixed and flexible positions

Finally the current rule sounds like if either fixed or flexible position were to be used. That is to say as if these two techniques were mutually exclusive. Our understanding is that the use of fixed or flexible position is determined on a per TrCH basis. That is to say the step of "DTX indication insertion for fixed position" is present only for those TrCH using fixed position. Conversely the step "DTX indication insertion for flexible position" is present if there is at least one TrCH using flexible positions.

The new rule stresses this point.

5 Other problems corrected in the text proposal

Another problem, not connected with rate matching parameter determination is handled in this contribution. In case of fixed service position the initial error value must not depend on the size *N* of the block to rate match. Otherwise the puncturing/repeating pattern would depend on the TF. This problem is fixed in the text proposal by using a new parameter e_{ini} .

6 Proposed solution

6.1 General on the proposed new rule

Because the problems of the current rule are lying in the very concept of working with TFC (radio frames) and not with TF (TTI), we propose a new rule.

In a nutshell the new rule consists in :

- computing tentative temporary values of ΔN_{il}^{TTI} that minimise the number of DTX inserted when the CCTrCH bit rate is maximum
- checking for all TFC that the temporary values of ΔN_{il}^{TTI} do not make the bit rate of the CCTrCH exceed the N_{data} limit
- in the case of exceeding N_{data} , correcting the temporary values with the “Z” formula
- after checking & possible correcting is complete, the temporary values become the definitive values

6.2 Notations

It was observed by delegates from several companies that using the same letter *j* for TFC and TF can be misleading. So we propose the following convention :

- *i* for indexing the TrCH
- *l* for indexing the TFs of some TrCH
- *j* for indexing the TFCs

So the text proposal will have some editorial impact also on the UL part, in order to use the convention above. Other notation modifications have been carried out in order to align with the notation harmonisation proposal from Ericsson :

I instead of *T* for number of TrCHs

Also we introduce the following (not completely new) notations :

- *TFS(i)* : is the set of TF indexes *l* for TrCH *i*,
- *TFCS* : is the set of TFC indexes *j*

We go on using the notation $N_{i,j}$ as in the current rule, but some text will clarify that this is in the DL an intermediate variable used in the process of computing the ΔN_{il}^{TTI} . Based on this, the definitions of $Z_{i,j}$ and of $\Delta N_{i,j}$ remain unchanged.

Finally we use the notation * (star) instead of some index *u* when some value X_u that is a function of *u* does not depend of the *u*. So X_* in the left wing of an assignment equation means “for all *u* X_u” and in the right wing is means “ X_u for any *u* ”.

6.3 Algorithm

Here we describe all the algorithm in a Pascal-like pseudo language. In the text proposal English will be used wherever possible.

It is to be noted that the description below can be simplified by defining the algorithm only when all the TrCH are in flexible position, and then by saying that, in case there are some TrCH *i* in fixed position, then the only thing to do is to replace wherever met the expression N_{il}^{TTI} by its maximum value $\max_{l \in TFS(i)} N_{il}^{TTI}$

From line 1 to 9 lies the definition of the intermediate variables $N_{i,j}$. It is to be noted that for the flexible position case, the expression at line 6 is changed compared to the current rule as the ceiling ($\lceil . \rceil$) is no longer taken. This is possible because for DL $N_{i,j}$ is not the positive integer size of a real data block, but just an intermediate variable. Also a new expression is given at line 3 for the fixed position case.

From line 12 to 36 lies the algorithm itself. This algorithm comprises two phases. In a first phase, from line 12 to 21, tentative temporary values for the $\Delta N_{i,l}^{TTI}$ are computed. These temporary values ensure that a minimum number of DTX is inserted when the bit rate of the CCTrCH is maximum, but they don't ensure that the bit rate of the CCTrCH is always within the $N_{data,*}$ limit.
So a second phase of checking and correction is needed. This second phase is lying from line 23 to 36.

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

Finally, it is to be noted that $\Delta N_{i,l}^{TTI}$ determined by the algorithm below are such that $N_{i,l}^{TTI} + \Delta N_{i,l}^{TTI}$ is always a multiple of F_i. So in DL with this new rule, the "size equalisation" step is not needed.

line	instruction	comment
	definition of the $N_{i,j}$ variables for all TrCHs and all TFCs	
1	**for** *i* = 1 **to** *I* **do**	for all TrCH *i*
2	**if** TrCH *i* is in fixed position of services **then**	
3	$N_{i,*} = \frac{1}{F_i} \cdot \max_{l \in TFS(i)} N_{i,l}^{TTI}$	for fixed positions of service case, $N_{i,j}$ does not depend of the TFC *j* and is always maximum.
4	**else**	
5	**for all** *j* **in** *TFCS* **do**	
6	$N_{i,j} = \frac{1}{F_i} \cdot N_{i,TF_i(j)}^{TTI}$	flexible positions of service case
7	**end-for**	
8	**end-if**	
9	**end-for**	
10		
11		
	computation of tentative temporary values for $\Delta N_{i,l}^{TTI}$	
12	$L = \frac{N_{data,*}}{\max_{j \in TFCS} \sum_{i=1}^{i=T} \left(RM_i \cdot N_{i,j}\right)}$	Compute rate matching ratios $RF_i = L \cdot RM_i$ for all transport channels *i* in order to minimise the number of DTX bits inserted for the maximum bit rate of the CCTrCH.
13	**for** *i* = 1 **to** *I* **do**	
14	$RF_i = L \cdot RM_i$	
15	**end-do**	
16		
17	**for** *i* = 1 **to** *I* **do**	
	if TrCH *i* is in fixed position of services **then**	
	$\Delta N_{i,*}^{TTI} = F_i \cdot \left\lceil \frac{RF_i \cdot \max_{l \in TFS(i)} N_{i,l}^{TTI}}{F_i} \right\rceil - \max_{l \in TFS(i)} N_{i,l}^{TTI}$	
	else	
18	**for all** l **in** *TFS*(*i*) **do**	Flexible positions of service case.
19	$\Delta N_{i,l}^{TTI} = F_i \cdot \left\lceil \frac{RF_i \cdot N_{i,l}^{TTI}}{F_i} \right\rceil - N_{i,l}^{TTI}$	
20	**end-do**	
	end-if	
21	**end-do**	

checking and correction of the $\Delta N_{i,l}^{TTI}$	
for all j **in** *TFCS* **do**	
$$D = \sum_{i=1}^{i=I} \frac{N_{i,TF_i(j)}^{TTI} + \Delta N_{i,TF_i(j)}^{TTI}}{F_i}$$	D is the bit rate (in number of bits per radio frames) of the CCTrCH for TFC j computed with the temporary values of $\Delta N_{i,l}^{TTI}$
if $D > N_{data,*}$ **then**	
for i = 1 **to** I **do**	correction of $\Delta N_{i,TF_i(l)}^{TTI}$ if needed.
$\Delta N = F_i \cdot \Delta N_{i,j}$	
if $\Delta N_{i,TF_i(j)}^{TTI} > \Delta N$ **then**	
if TrCH i is in fixed position of services **then**	
$\Delta N_{i,*}^{TTI} = \Delta N$	
else	
$\Delta N_{i,TF_i(j)}^{TTI} = \Delta N$	
end-if	
end-if	
end-for	
end-if	
end-for	

7 Conclusion

We have presented a new rule for rate matching parameter computation.
This rules solves the problems existing with the current one. It also solves the "radio frame size equalisation" problem for the DL direction, as after rate matching the block size $N + \Delta N$ is always a multiple of F that is the TTI duration in 10ms unit.
We propose the described rule for approval.

8 Text proposal

-------- SNIP --------

-------- SNIP --------

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

Notation used in Section 4.2.6 and subsections:

<Mitsubishi note : some notations have already been aligned with Ericsson's proposal of notation harmonisation in order to ease the editor's work>

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

For downlink : An intermediate calculation variable (might be not integer).

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

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

For downlink : An intermediate calculation variable (might be not integer).

~~ΔN_{ij}^{TTI}~~ ΔN_{il}^{TTI} : If positive - number of bits to be repeated in each transmission time interval on transport channel i with transport format j.
If negative - number of bits to be punctured in each transmission time interval on transport channel 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.

~~T~~I: Number of transport channels in the CCTrCH.

$Z_{~~m~~ij}$: Intermediate calculation variable.

F_i: Number of radio frames in the transmission time interval of transport channel i.

~~k~~n_i: Radio frame number in the transmission time interval of transport channel i ($0 \le n_i$ ~~k~~ $< F_i$).

q: Average puncturing distance. Used in uplink only.

$I_F(n_i$~~k~~$)$: 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.

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

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

$TFS(i)$ The set of transport format indexes l for transport channel 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 4.2.6.3.

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

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

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

<Mitsubishi note : in the formula below the change is that T is replaced by I>

~~$$Z_{mj} = \left\lfloor \frac{\sum_{i=1}^{m} RM_i \cdot N_{ij}}{\sum_{i=1}^{T} RM_i \cdot N_{ij}} \cdot N_{data,j} \right\rfloor$$~~

$$Z_{ij} = \left\lfloor \frac{\sum_{m=1}^{i} RM_m \cdot N_{mj}}{\sum_{m=1}^{I} RM_m \cdot N_{mj}} \cdot N_{data,j} \right\rfloor$$

for all ~~m~~ i = 1 .. ~~T~~I, *where ë û means round downwards*

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

4.2.6.1 Determination of rate matching parameters in uplink

In uplink puncturing can be used to avoid multicode or to enable the use of a higher spreading factor when this is needed because the UE does not support SF down to 4. The maximum amount of puncturing that can be applied is signalled at connection setup from higher layers and denoted by *PL*. The number of available bits in the radio frames for all possible spreading factors is given in [2]. Denote these values by N_{256}, N_{128}, N_{64}, N_{32}, N_{16}, N_8, and N_4, where the index refers to the spreading factor. The possible values of N_{data} then are $\{ N_{256}, N_{128}, N_{64}, N_{32}, N_{16}, N_8, N_4, 2N_4, 3N_4, 4N_4, 5N_4, 6N_4\}$.Depending on the UE capabilities, the supported set of N_{data} , denoted SET0, can be a subset of $\{ N_{256}, N_{128}, N_{64}, N_{32}, N_{16}, N_8, N_4, 2N_4, 3N_4, 4N_4, 5N_4, 6N_4\}$. $N_{data,j}$ for the transport format combination j is determined by executing the following algorithm:

SET1 = { N_{data} in SET0 such that ~~$N_{data} - \sum_{i=1}^{T} \frac{RM_i}{\min_l \{RM_l\}} \cdot N_{ij}$~~ $N_{data} - \sum_{m=1}^{I} \frac{RM_m}{\min_{1 \le n \le I} \{RM_n\}} \cdot N_{mj}$ is non negative }

If the smallest element of SET1 requires just one DPDCH then
 $N_{data,j}$ = min SET1
else
 SET2 = { N_{data} in SET0 such that
 ~~$N_{data} - PL \cdot \sum_{i=1}^{T} \frac{RM_i}{\min_l \{RM_l\}} \cdot N_{ij}$~~ $N_{data} - PL \cdot \sum_{m=1}^{I} \frac{RM_m}{\min_{1 \le n \le I} \{RM_n\}} \cdot N_{mj}$ 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 DPDCH do
  N_{data} = follower of N_{data} in SET2
 End while
 $N_{data,j} = N_{data}$
End if

The number of bits to be repeated or punctured, DN_{ij}, within one radio frame for each transport channel i is calculated with the relations given in Section 0 for all possible transport format combinations j and selected every radio frame. ~~For each radio frame, the rate-matching pattern is calculated with the algorithm in Section 4.2.6.3, where $DN = DN_{ij}$ and $N = N_{ij}$.~~
Additionally, for determining e_{ini}, the following parameters are needed:
q= ëN_{ij} /(ôDN_{ij}ô)û , where ë û means round downwards and ô ô means absolute value.

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 ~~l~~m = 0 to F_i-1
 *S(I$_F$ (é~~l~~m*q'ù mod F_i)) = (é~~l~~m*q'ù div F_i) -- where é ù means round upwards.*
end for

For each radio frame, the rate-matching pattern is calculated with the algorithm in Section 4.2.6.3, where :

$DN = DN_{ij}$

$N = N_{ij}$., and

$$e_{ini} = \left(2 \cdot S(k) \cdot |\Delta N| + N\right) \mod 2N$$

4.2.6.2 Determination of rate matching parameters in downlink

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

~~NOTE: The rule to convert the rate matching attributes in downlink to the parameters input to rate matching pattern algorithm are working assumption. So, it remains to be verified that they hold for all possible transport format combinations. It has been identified that the case when the transport format combination with highest rate include a transport format with zero bits need special treatment.~~

~~Radio frame segmentation is performed after 1st interleaving and N_{ij} is therefore calculated as:~~

$$l = TF_i(j) \quad \text{and} \quad N_{ij} = \left\lceil \frac{N_{i,l}^{TTI}}{F_i} \right\rceil$$

~~The number of bits repeated or punctured, DN_{iL}, within one radio frame for each transport channel is calculated for the transport format combination L with highest bitrate with the relations given in Section 4.2.6.~~

~~If fix positions of the transport channels in the radio frame are used then the same DN_{ij} is used for all transport format combinations and the last part of the rate-matching pattern omitted. That is to say for all transport format combinations j we have:~~

$$\Delta N_{ij} = \Delta N_{iL}$$

~~When flexible positions of the transport channels are used, the number of bits DN_{ij} repeated or punctured for all transport format combinations j other than L is calculated as:~~

$$\Delta N_{ij} = \left\lfloor \frac{\Delta N_{iL}}{N_{iL}} \cdot N_{ij} \right\rfloor$$

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

If transport channel i uses flexible positions of service :

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

If transport channel i uses fixed positions of service :

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

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 :

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

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

If transport channel *i* uses flexible positions of service :

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

If transport channel *i* uses fixed positions of service :

$$\Delta N_{i,*}^{TTI} = F_i \cdot \left\lceil \frac{RF_i \cdot \max_{l \in TFS(i)} N_{i,l}^{TTI}}{F_i} \right\rceil - \max_{l \in TFS(i)} N_{i,l}^{TTI}$$

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

$\Delta N = F_i \cdot \Delta N_{i,j}$

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

if TrCH *i* is in fixed positions of services **then**

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

else **--** flexible position case

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

end-if

end-if

end-for

end-if

end-for

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

For each transmission time interval of TrCH *i* with TF *l*, the rate-matching pattern is calculated with the algorithm in Section 4.2.6.3. The following parameters are used as input:

~~$l = TF_i(j)$ and $\Delta N = \Delta N_{il}^{TTI} = F_i \Delta N_{ij}$~~

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

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

If transport channel *i* is in flexible positions :

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

If transport channel *i* is in fixed positions :

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

~~S=0.~~

4.2.6.3 Rate matching algorithm

Denote the bits before rate matching by:

$c_1, c_2, c_3, \ldots c_N$

The rate matching rule is as follows:

if puncturing is to be performed

```
        y = -DN
        e = e_ini(2*S(k) * y +  N) mod 2N  -- initial error between current and desired puncturing ratio
        m = 1                -- index of current bit
        do while m <= N
                e = e – 2 * y              -- update error
                if e <= 0 then             -- check if bit number m should be punctured
                        puncture bit c_m
                        e = e + 2*N   -- update error
                end if
                m = m + 1                      --  next bit
        end do
else
        y = DN

        e = e_ini(2*S(k) * y +  N) mod 2N  -- initial error between current and desired puncturing ratio
        m = 1                    -- index of current bit
        do while m <= N
                e = e – 2 * y              -- update error
                do while e <= 0            -- check if bit number m should be repeated
                        repeat bit c_m
                        e = e +  2*N  -- update error
                enddo
                m = m + 1                      --  next bit
        end do
end if
```

A repeated bit is placed directly after the original one.

------- SNIP -------