TSG-RAN Working Group1 meeting #7 **TSGR1#7(99)B05**
30 August – 3 September 1999
Hanover, Germany

Agenda item: **AH4**

Source: **QUALCOMM Europe**

Title: **Simplified transport block equalization and segmentation**

Document for: **Decision**

Introduction

During the meeting TSG RAN WG1 #6, a proposal for radio frame equalization and radio segmentation described in Tdoc R1-99892 was accepted and TR 25.212 was updated accordingly. We believe that an equivalent solution can be described in much simpler terms while minimizing the memory management task associated with those operations.

Current solution

Let's consider the operations associated with 1st IL, radio frame equalization and radio frame segmentation as currently described in 25.212 v2.0.1:

1) Write the input data block of size K row by row in a matrix of size R x C, where C, the number of columns, equals the number of radio frames spanned by the data block and R the minimum integer such that K <= (R x C). Note that this operation virtually adds N = (R x C) – K dummy bits.
2) Permute the columns according to a specific pattern pre-defined for each value of C.
3) Prune the N dummy bits.
4) Add N filler bits in positions defined by the set of equations in section 4.2.5.1
5) Map the result sequentially to C consecutive radio frames

A graphic representation of this set of operations is shown in figure 1.

Proposed solution

We believe that step 3 and step 4 can be omitted, together with the set of equations described in section 4.2.5.1 and 4.2.5.2. Considering that the 1st interleaver is by definition (inter-frame interleaving) dealing with radio frame segmentation and that step 1 is virtually equalizing the size of the C columns, by virtually adding N dummy bits, we conclude that with a very minor editorial modification the 1st IL can perform the interleaving and the frame equalization together as follows:

1) Write the input data block of size K row by row in a matrix of size R x C, where C, the number of column, equals the number of radio frames spanned by the data block and R the minimum integer such that K <= (R x C). **Complete the last row with N = (R x C) – K filler bits**.
2) Permute the columns according to a specific pattern pre-defined for each value of C
3) At this stage, each column of the IL matrix corresponds to a radio frame

Note that if the polarity of the filler bits is left unspecified, the frame size equalization does not involve any processing. A graphic representation of this set of operations is shown in figure 2.

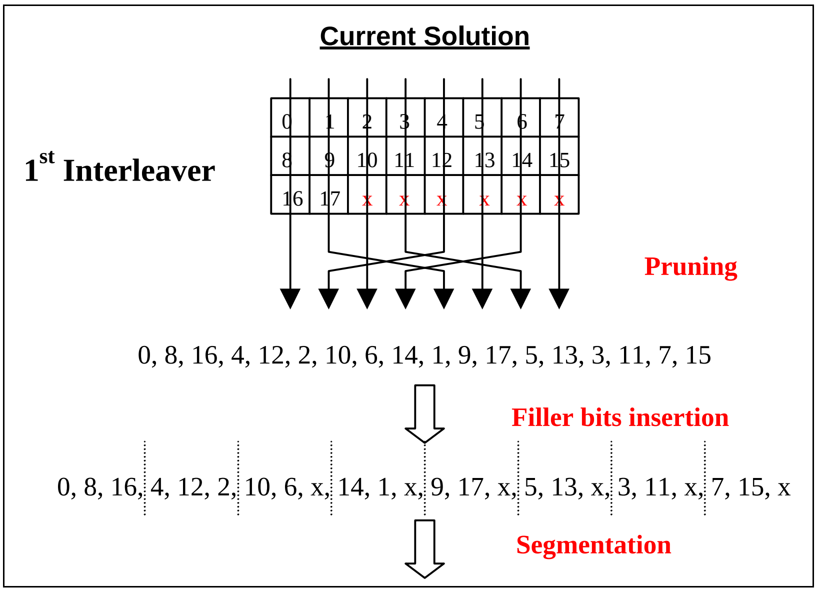

Figure 1

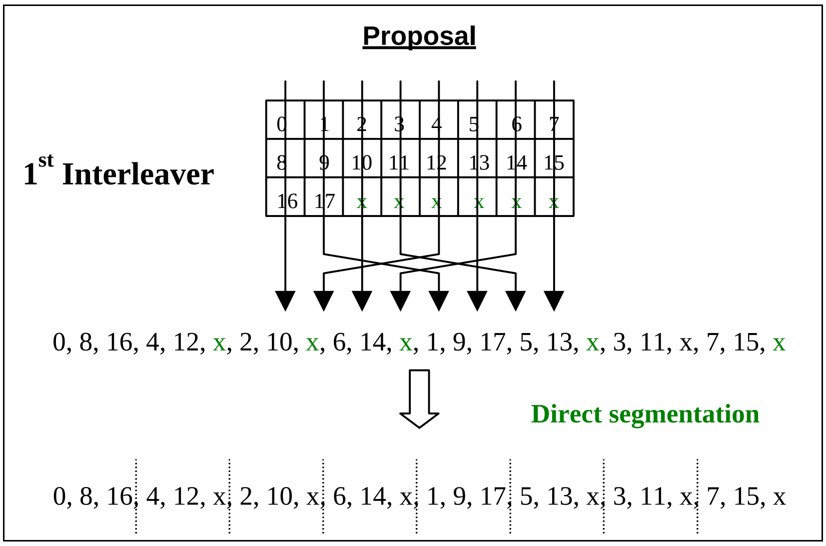

Figure 2

Discussion

With a much simplified procedure and description the proposed solution achieves the same result, which is that the last bit of N radio frame is a filler bit. The only difference with the current solution is that the set of radio frames which are terminated with a filler bit is different. This has absolutely no impact on performance.

Conclusion

With a very minor editorial change the set of operations associated with 1st interleaving, frame equalization and segmentation can be reduced and the associated description simplified. We therefore propose that the text modification presented below is incorporated into the next version of TS 25.212.

Text Proposal

4.2 Transport-channel coding/multiplexing

Figure 1: addition of a radio frame size equalisation block before 1st interleaving
Figure 2: addition of a radio frame size equalisation block before 1st interleaving

4.2.4 Radio frame size equalisation

Radio frame size equalisation ensures that the radio frame can be segmented in C data segments of equivalent size as described in section 4.2.6. Radio frame size equalisation shall be performed as follows.

Given the input sequence $a_1, \ldots, a_K$ of length K (corresponding to the output of channel coding in UL and the output of DTX insertion with fixed position in DL), the output sequence $b_1, \ldots, b_{(C*R)}$ is defined as:

$b_i = a_i$ for i = 1…K
$b_i = \{0 \mid 1\}$ for i=(K+1)…(C*R), if K<(C*R)

where

C = (TTI in ms / 10 ms) is the number of segments per transmission interval and
R = ((K –1) div C) + 1 is the number of bits per segment after size equalisation

~~4.2.4~~4.2.5 1st interleaving

The 1st interleaving of channel interleaving consists of two stage operations. In first stage, the input sequence is written into rectangular matrix row by row. The second stage is inter-column permutation. The input sequence length (output of radio frame size equalisation) is assumed to be K_1. The two-stage interleaving~~operations are described~~ shall be performed as follows:~~, the input block length is assumed to be K_1.~~

First Stage:

(1) Select ~~a~~ the column number C_1 from Table 1~~Table 1~~.
(2) Determine ~~a~~ the row number R_1 defined as ~~by finding minimum integer R_1 such that,~~
$R_1 = K_1/C_1$~~$K1 \le R_1 \times C_1$.~~
(3) Write ~~T~~the input sequence $a_1, \ldots, a_{K1}$ ~~of the 1st interleaving is written~~ into the $R_1 \times C_1$ rectangular matrix row by row starting with a_1 in the first column of the first row and ending with a_{K1} in column C_1 of row R_1.

Second Stage:

(1) Perform the inter-column permutation based on the pattern $\{P_1(j)\}$ (j=0,1, ..., C-1) ~~that is~~ shown in Table 1~~Table 1~~, where $P_1(j)$ is the original column position of the j-th permuted column.
(2) Read ~~T~~the output sequence $b_1, \ldots, b_{K1}$ of the 1st interleaving ~~is the sequence read out~~ column by column from the inter-column permuted $R_1 \times C_1$ matrix. b_1 corresponds to the first row of the first column and b_{K1} corresponds to row R_1 of column C_1. ~~and the output is pruned by deleting the non-existence bits in the input sequence, where the deleting bits number l_1 is defined as:~~

~~$l_1 = R_1 \times C_1 - K_1$.~~

Table 1

Interleaving span	Column number C_1	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}

4.2.54.2.6 Radio frame segmentation

The input data sequence (output of 1st interleaving) $a_1, \ldots, a_{K1}$ of length K_1 is divided into C_1 segments of length R_1. Each segment is then mapped onto C1 consecutive radio frames corresponding to the transmission time interval of the transport channel. The segmentation shall be performed as follows:

$b_{i,j} = a_{((i-1)*R1)+j}$, $i=1\ldots C_1$, $j=1\ldots R_1$

where

C_1 = (associated TTI in ms) / 10 ms is the number for segments,
$R_1 = (K_1 / C_1)$ is the number of bits per segment,
$b_{i,j}$ is the j^{th} bit of the i^{th} output sequence (or segment) and
a_k is the k^{th} bit of the input sequence

The i^{th} segment shall be mapped to the i^{th} radio frame of the transmission time interval as defined by the rate matching and 2nd multiplexing in the UL and by the 2nd multiplexing in the DL.

Each transport channel with transmission time interval 10, 20, 40, or 80 msec is segmented into 10 msec equi-sized data blocks. Those segmented 1, 2, 4, or 8 blocks, depending on transmission time interval, are output to rate matching for uplink and 2nd multiplexing for downlink in block-wise order at every 10 msec. Figure B-1 and B-2 illustrate data flow from 1st interleaver down to 2nd interleaver in both uplink and downlink channel coding and multiplexing chains. In the figures, it is assumed that there are *N* different channel coding and multiplexing chains. The following subsections describe input-output relationship of radio frame segmentation in bit-wise manner, referring to the notations in Figure B-1 and B-2, where the notations in each data block, for examples L_1, R_1, K_1, P/M, etc., indicate number of bits of the data block. Define some notations:

L_i = Size of i^{th} transport channel data in bits to radio frame segmentation

T_i = Transmission Time Interval of i^{th} channel coding and multiplexing chain (msec) / 10 (msec)

So, T_i Î $\{1, 2, 4, 8\}$ for $i = 0, 1, 2, \ldots, N$

4.2.5.1 Radio frame size equalization

i^{th} transport channel data of size L_i is segmented into radio frames of size L_i/T_i. Since the size of radio frame, L_i/T_i is not necessarily an integer, some of T_i the radio frames will contain one bit less than others. For systematic process of the proceeding functional blocks, the radio frame sizes are equalized to be one finite size by considering the number of proper filler bits. Note that maximum possible filler bits are 7 for transmission time interval of 80 msec. These filler bits are evenly distributed over the one-bit short radio frames. Following is the algorithm of radio frame size equalization.

t = radio frame index (1, 2, 3, …, T_i) for a given i^{th} channel coding and multiplexing chain

$r_i = T_i - (L_i \bmod T_i)$ Î $\{0, 1, 2, \ldots, T_i-1\}$ *// number of filler bits*

$(L_i + r_i)/ T_i = R_i$ *// Target radio frame size for uplink*

$(L_i + r_i)/ T_i = K_i$ *// Target radio frame size for downlink*

~~*If ri ¹ 0 then*~~
~~*For each t (³ T_i-r_i+1)*~~
~~*Add one filler bit to the end of t^{th} radio frame*~~
~~*End*~~
~~*End If*~~

~~4.2.5.2 Radio frame segmentation rule~~

~~Parameter r_i for segmentation are determined in radio frame size equalization.~~

~~The bits before radio frame segmentation for i^{th} channel coding and multiplexing chain are denoted by: $b_{i1}, b_{i2}, \ldots b_{iL_i}$~~

~~Bits after radio frame segmentation block are 10 msec-based and denoted by: $c_{i1}, \ldots c_{i,(L_i+r_i)/T_i}$~~
~~and related to the input bits to radio frame segmentation as follows.~~

~~Bits after radio frame segmentation in the first 10 msec time interval: $(t=1)$~~
~~$c_{ij} = b_{ij}$ $j=1,2,\ldots, (L_i+r_i)/T_i$~~
~~($(L_i+ r_i)/T_i$ equals to R_i and K_i for uplink and downlink, respectively.)~~
~~Bits after radio frame segmentation in the second 10 msec time interval: $(t=2)$~~
~~$c_{ij} = b_{i,(j+(L_i+r_i)/T_i)}$ $j=1,2, \ldots, (L_i+r_i)/T_i$~~
~~…~~

~~Bits after radio frame segmentation in the $(T_i-r_i)^{th}$ 10 msec time interval: $(t=T_i-r_i)$~~
~~$c_{ij} = b_{i,(j+(T_i-r_i-1)(L_i+r_i)/T_i)}$ $j=1,2, \ldots, (L_i+r_i)/T_i$~~

~~Bits after radio frame segmentation in the $(T_i-r_i+1)^{th}$ 10 msec time interval: $(t=T_i-r_i+1)$~~
~~$c_{ij} = b_{i,(j+(T_i-r_i)(L_i+r_i)/T_i)}$ $j=1,2, \ldots, (L_i+r_i)/T_i-1$~~
~~c_{ij} = filler_bit(0/1) $j= (L_i+r_i)/T_i$ (filler bit)~~
~~…~~

~~Bits after radio frame segmentation in the T_i^{th} 10 msec time interval: $(t=T_i)$~~
~~$c_{ij} = b_{i,(j+(T_i-1)(L_i+r_i)/T_i)}$ $j=1,2, \ldots, (L_i+r_i)/T_i-1$~~
~~c_{ij} = filler_bit(0/1) $j=(L_i+r_i)/T_i$ (filler bit)~~

Annex B (informative): Data flow from radio frame segmentation to physical channel segmentation

Remove Annex B