3GPP TSG-RAN WG1#49*bis* **R1-073031**
Orlando, USA, June 25—29, 2007

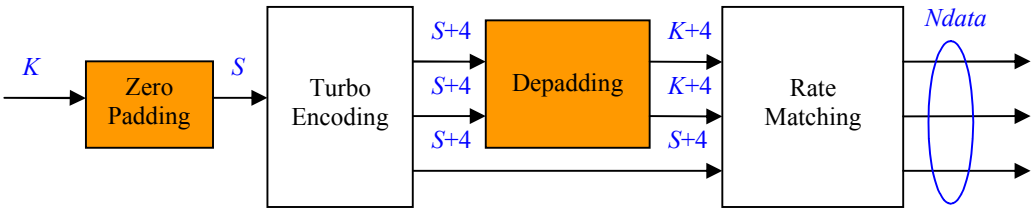
Source: **Ericsson**
Title: **Combined Partial Depadding and Rate Matching Based on Circular Buffers**
Agenda Item: **5.5 Channel Coding**
Document for: **Discussion and Decision**

1. Introduction

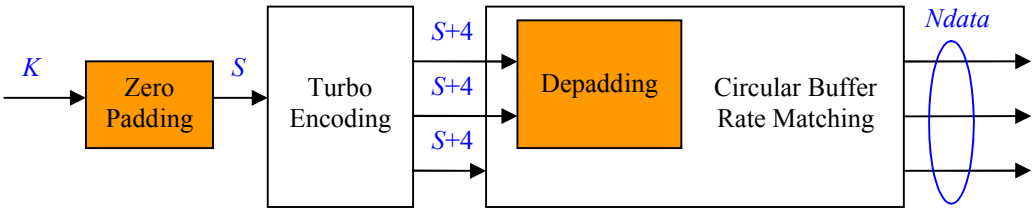
In LTE turbo coding [1], a set of 188 byte-aligned block sizes from 40 to 6144 bits is defined. To support transport blocks of variable sizes, code block segmentation and code shortening techniques are adopted. The later involves padding the first code segment of size K up to the next supported turbo coding interleaver size S prior to turbo encoding if K is not one of the turbo coding interleaver size. Discussion regarding further processing of the padding bits in the channel coding chain was presented in [2—4]. The objective of these depadding algorithms is to provide turbo coding performance consistent with a segment size K without an actual QPP interleaver of size K.

The partial depadding approach, where padding bits in the Systematic and the Parity1 streams are explicitly removed and the Parity2 stream is treated uniformly by the rate matcher, was agreed in [5]. However, as shown in Figure 1 (a), the partial depadding solution as originally proposed would reduce the length of the Parity1 stream to K+4 while leaving the length of the Parity2 stream at S+4. The presents an incompatibility challenge to the recently agreed circular buffer rate matching (CBRM) algorithms [6—10] since the two Parity streams of different lengths cannot be bit-multiplexed into the buffer.

In this paper, we propose a partial depadding method compatible with CBRM algorithms. As illustrated in Figure 1 (b), it is proposed to pass all padding bits into the rate matcher. Minimal changes are then introduced to the CBRM algorithm to exclude these bits from the output. Details of the proposal are presented in Section 2. Extensive performance analysis on worst-case padding scenarios is discussed in Section 3. It is found the proposal achieves performance consistent with no padding baselines, i.e., no loss from the pre-padding and depadding operations. Hence, it is suggested to adopt this combined partial depadding and rate matching approach for the LTE system.



(a) Original proposal for partial depadding [2, 4]



(b) Proposal for combined partial depadding and circular buffer rate matching

Figure 1 Partial depadding encoders for an information segment size *K* smaller than a supported QPP interleaver size *S*.

2. CBRM Computation with Partial Depadding

To retain the integrity of the overall CBRM structure, it is proposed to pass all padding bits into the rate matcher. Since the addresses of the padding bits are pre-determined [0, 1, …, $S-K-1$], they can be excluded from the output of the rate matcher with simple modification to the CBRM algorithm. As shown in Figure 2, this can be accomplished with adding two range checks (lines 7—9 and 18—22) to the simple rate matching algorithm discussed in [11]. This is a straightforward implementation of the virtual circular buffer approach, where

redundancy versions are constructed directly from the outputs of a turbo encoder. By taking account of the structural properties of the sub-block interleavers, the algorithm computes the exact set of addresses for Ndata memory accesses and requires no multiplications for address computation.[1] Note further that a complete buffer (to be used in the physical circular buffer approach) can be constructed by calling this algorithm with RV=0 and Ndata=3*K*+12.

```
Input  RV                              % redundancy version
Input  Ndata                           % number of desired coded bits
Input  Sys, Par1, Par2                 % buffers for sys, Parity1, Parity2
                                       %   length = S+4 >= K+4
Output Out                             % buffer for retained bits
```

```
 1 n = 0;                              % index of the output buffer
 2 c = σ+RV*(96/nRV);                  % get the starting column
 3 while n<Ndata
 4   c = mod(c, 96);                   % go through the columns circularly
 5   if c<32                           % the first 32 cols are Systematic
 6     A = ColPerm(c);                 % get starting address for Sys
 7     while A<S-K                     % avoid padding bits in Systematic
 8       A += 32;
 9     end
10     while A<S+4                     % before the end of column
11       Out(n++) = Sys(A);            % take the systematic bit at A
12       if n==Ndata, return, end;
13       A += 32;                      % next address
14     end
15     c += 1;                         % next column
16   else                              % the next 2*32 cols are parities
17     A = ColPerm(c/2-16);            % get starting address for Parity1
18     while A<S-K                     % avoid padding bits in Parity1
19       Out(n++) = Par2(A+δ);
20       if n==Ndata, return, end;
21       A += 32;
22     end
23     while A<S+4                     % before the end of column
24       Out(n++) = Par1(A);           % take the Parity1 bit at A
25       if n==Ndata, return, end;
26       Out(n++) = Par2(mod(A+δ,S+4)); % take the Parity2 bit at A+δ
27       if n==Ndata, return, end;
28       A += 32;                      % next address
29     end
30     c += 2;                         % next two columns
31   end
32 end
```

Figure 2 Pseudo codes for combined partial depadding and rate matching based on circular buffers.

The algorithm starts by computing the initial column index for the requested RV in line 2. The algorithm then handles the Systematic area in lines 6—15 and the Parity area in lines 17—30. In the Systematic area, the first address in each column is directly given by the column permutation:

ColPerm = [0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30,
1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31],

because bits are written into the rectangular interleaver row by row. Since bits are read out of the interleaver column by column, each successive address in the column can be obtained by an increment of 32 (i.e., the

[1] Since σ and nRV are not yet fixed in the specification, line 2 is expressed with a multiplication for generality purpose. Once these parameters are finalized, line 2 can be implemented with a look-up table of nRV entries. Note also that line 26 requires only a range check instead of a full-blown modulo operation in implementation

number of columns). If the address falls in the padding bit range, then the bit is omitted. Once a column is exhausted, the process continues to the next column. In the Parity area (the last 64 columns of the circular buffer), two columns (one from Parity1 and one from Parity2) are bit-multiplexed into two columns. Hence the initial address of an even-index column is given by ColPerm((c-32)/2). The addresses for the two Parity streams are offset by δ. If the address falls in the padding bit range, then only Parity1 bit is selected. Since two columns are processed jointly, the column index is advanced by 2.

Essentially the same algorithm for rate matching computation at the transmitter side can be used for de-rate-matching and re-padding computation at the receiver side as shown in Figure 3. The buffers for the three streams (Systematic, Parity1 and Parity2) are initialized to all zeros for new data. With this approach, received soft values can be directly de-rate-matched into formats acceptable to a turbo decoder. Note that line 8 of setting soft values at the padded bit positions to sure zero (represented by $-\infty$) is just one possible solution for illustration purpose. Alternatively, one can leave these Systematic SVs to zero and instead sets the corresponding extrinsic information to sure zero.

```
Input  RV                              % redundancy version
Input  Rec                             % buffer for received soft values
Input  Ndata                           % number of coded bits in Rec
Output Sys, Par1, Par2                 % buffers for sys, Parity1, Parity2
                                       %   length = S+4 >= K+4
```

```
1  n = 0;                              % index of the output buffer
2  c = σ+RV*(96/nRV);                  % get the starting column
3  while n<Ndata
4    c = mod(c, 96);                   % go through the columns circularly
5    if c<32                           % the first 32 cols are Systematic
6      A = ColPerm(c);                 % get starting address for Sys
7      while A<S-K                     % set padding bits in Systematic
8        Sys(A) = -∞;                  % to sure zero
9        A += 32;
10     end
11     while A<S+4                     % before the end of column
12       Sys(A) += Rec(n++);           % accumulate Systematic SV at A
13       if n==Ndata, return, end;
14       A += 32;                      % next address
15     end
16     c += 1;                         % next column
17   else                              % the next 2*32 cols are parities
18     A = ColPerm(c/2-16);            % get starting address for Parity1
19     while A<S-K                     % avoid padding bits in Parity1
20       Par2(A+δ) = Rec(n++);
21       if n==Ndata, return, end;
22       A += 32;
23     end
24     while A<S+4                     % before the end of column
25       Par1(A) += Rec(n++);          % accumulate Parity1 SV at A
26       if n==Ndata, return, end;
27       Par2(mod(A+δ,S+4)) += Rec(n++);% accumulate Parity2 SV at A+δ
28       if n==Ndata, return, end;
29       A += 32;                      % next address
30     end
31     c += 2;                         % next two columns
32   end
33 end
```

Figure 3 Pseudo codes for combined re-padding and de-rate-matching computation. The buffers for the three streams (Systematic, Parity1 and Parity2) are initialized to all zeros for new transmissions.

3. Performance Analysis

Since transport blocks in the LTE system will have byte-aligned sizes, code shortening procedure are not expected for code segments shorter than 512 bits. For a segment size K larger than 512, the maximum amount of padding overhead depends on the segment size:

- For $512 < K < 1024$, maximum padding is 8 bits (up to 1.5%).
- For $1024 < K < 2048$, maximum padding is 24 bits (up to 2.3%).
- For $2048 < K < 6144$, maximum padding is 56 bits (up to 2.7%).

We test the worst-case padding for each QPP interleaver size $512 \leq S_{n-1} \leq 6080$ by setting the information segment length to $K=S_{n-1}+8$. For performance comparison, the references are the segment lengths requiring no code shortening: $K=S_n$, where $520 \leq S_n \leq 6144$. Other parameters for the performance tests are listed in Table 1.

The performance of the test and reference cases is presented in Figures 4—8. It is found performance of partial-depadding cases with $K=S_{n-1}+8$ to be similar to that of the reference no-padding cases with $K=S_n$. That is, the objective of providing turbo coding performance consistent with a segment size K without an actual QPP interleaver of size K is achieved with the proposed partial depadding method.

Table 1 Simulation Parameters

Common Code Structure	LTE Turbo Coding [1]
Rate Matching Algorithms	CBRM(σ=4,δ=4) as described in Section 2
Coding Rates	r = 0.4, 0.5, 0.6, 0.7, 0.8
Test Block Lengths	1. Partial depadding cases: $K=S_{n-1}+8$ 2. No padding references: $K=S_n$ for all QPP interleaver sizes $512 \leq S_{n-1} \leq 6080$
Redundancy Version	RV = 0
Decoding Algorithm	Improved Max-Log-MAP (i.e., 0.75 scaling on extrinsic information)
Iterations	8
Modulation	QPSK
Channel	Static AWGN

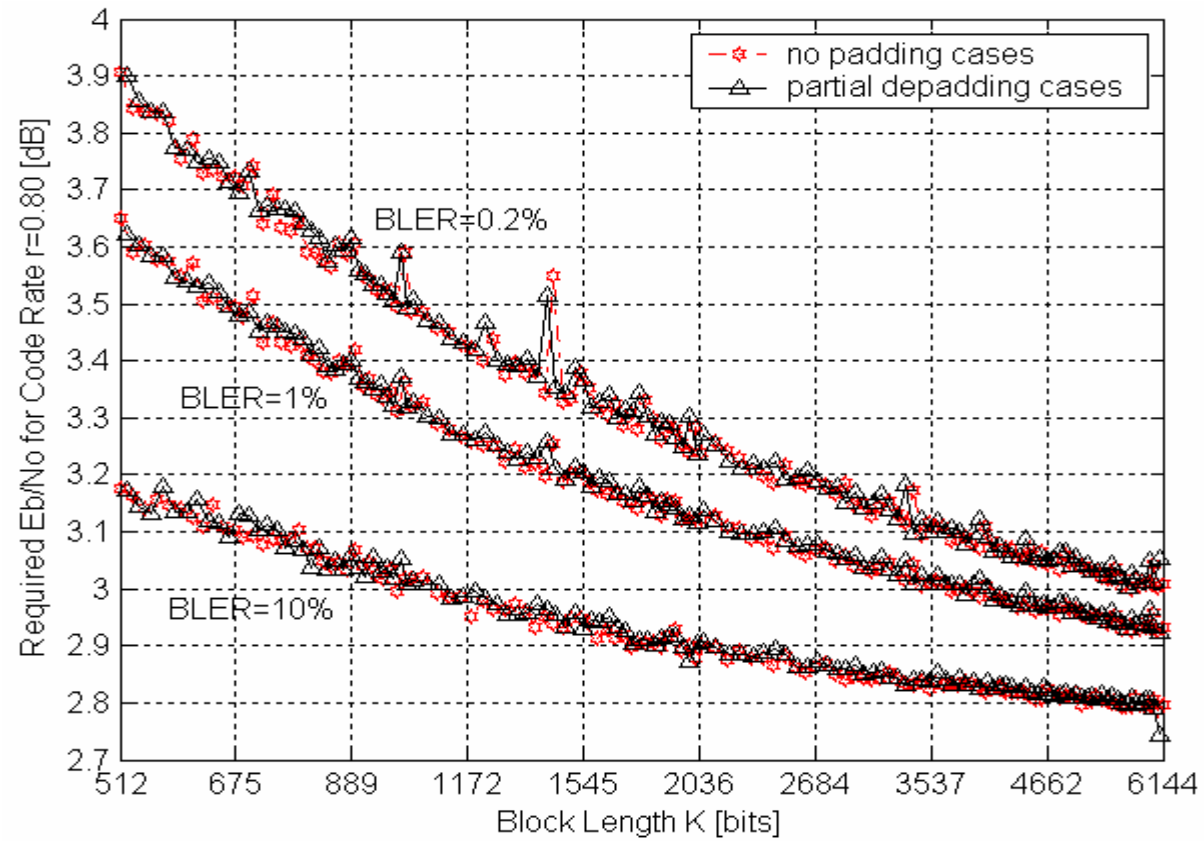


Figure 4 Required E_b/N_0 code rate r=0.8. For each of the partial depadding (K=S_{n-1}+8 bits) and no padding cases (K=S_n bits), three curves corresponding to BLER targets of 10%, 1% and 0.2% are shown.

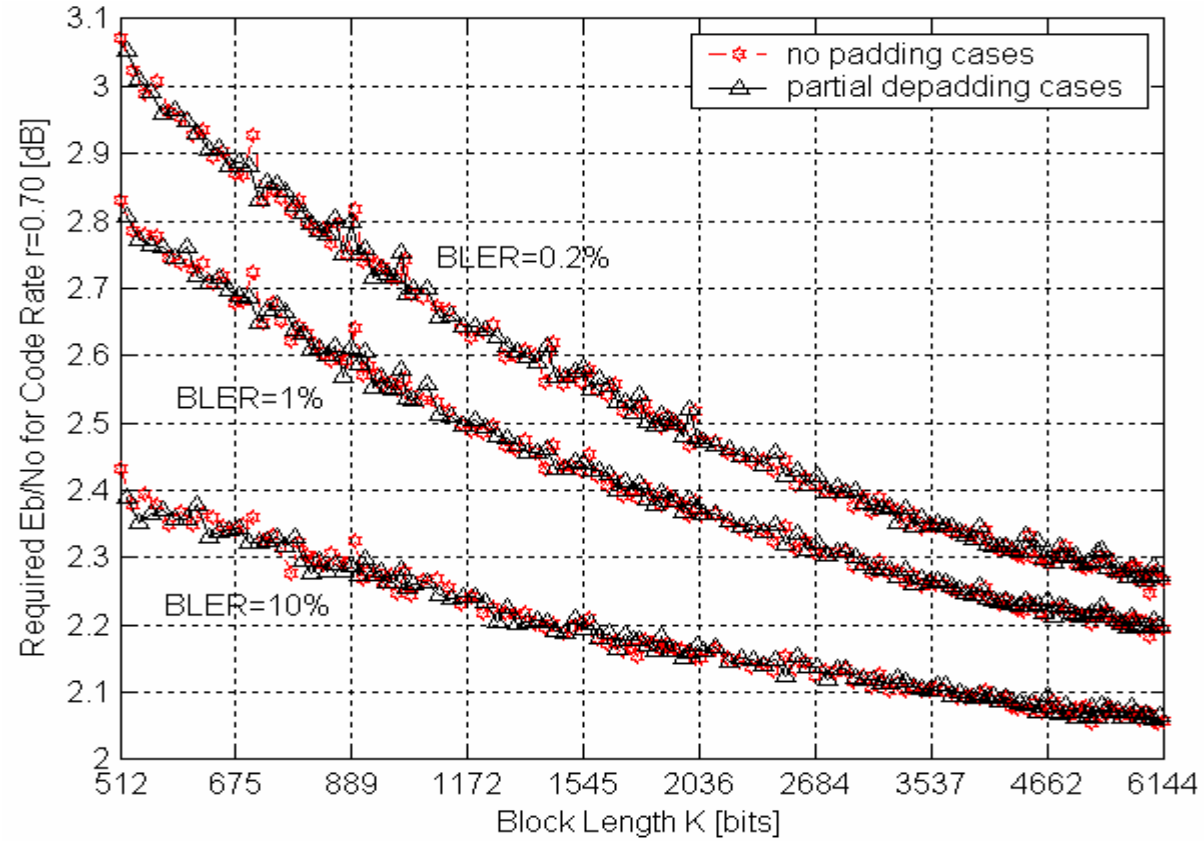


Figure 5 Required E_b/N_0 code rate r=0.7. For each of the partial depadding (K=S_{n-1}+8 bits) and no padding cases (K=S_n bits), three curves corresponding to BLER targets of 10%, 1% and 0.2% are shown.

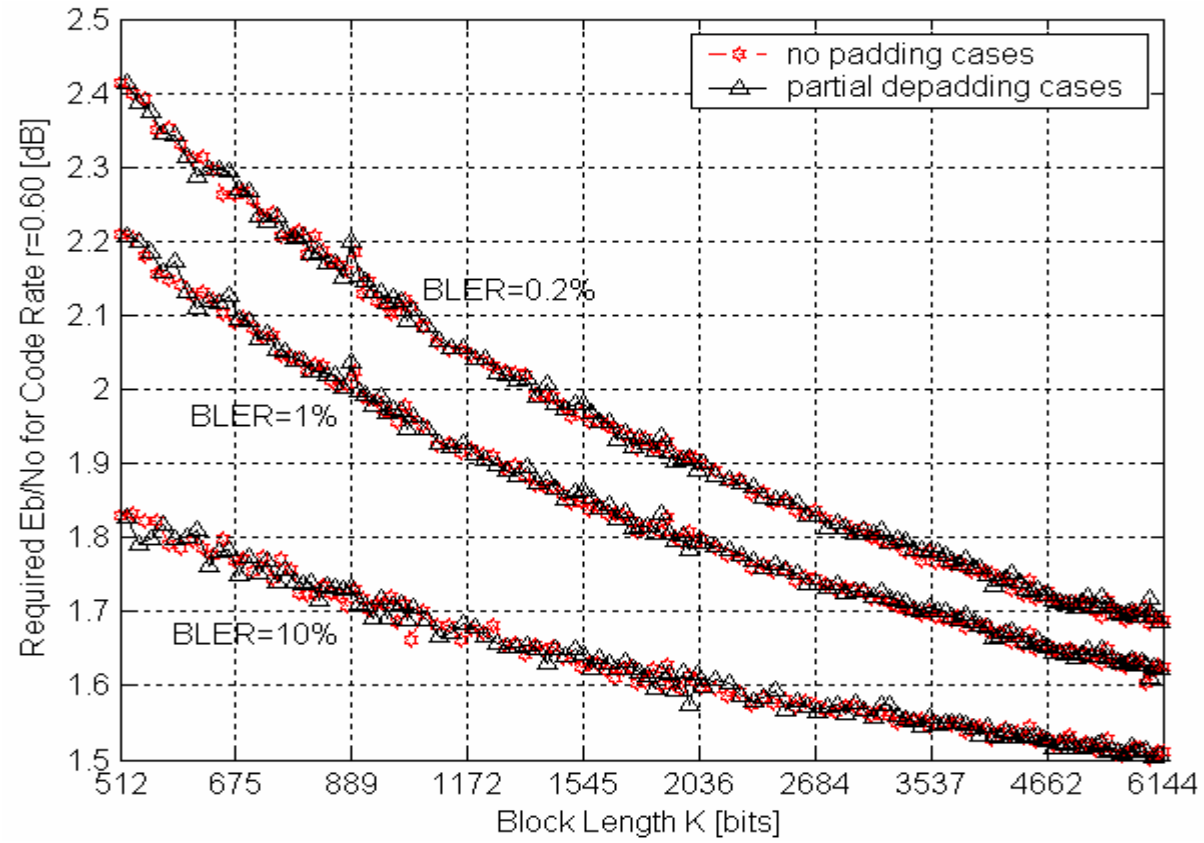


Figure 6 Required E_b/N_0 code rate r=0.6. For each of the partial depadding ($K=S_{n-1}+8$ bits) and no padding cases ($K=S_n$ bits), three curves corresponding to BLER targets of 10%, 1% and 0.2% are shown.

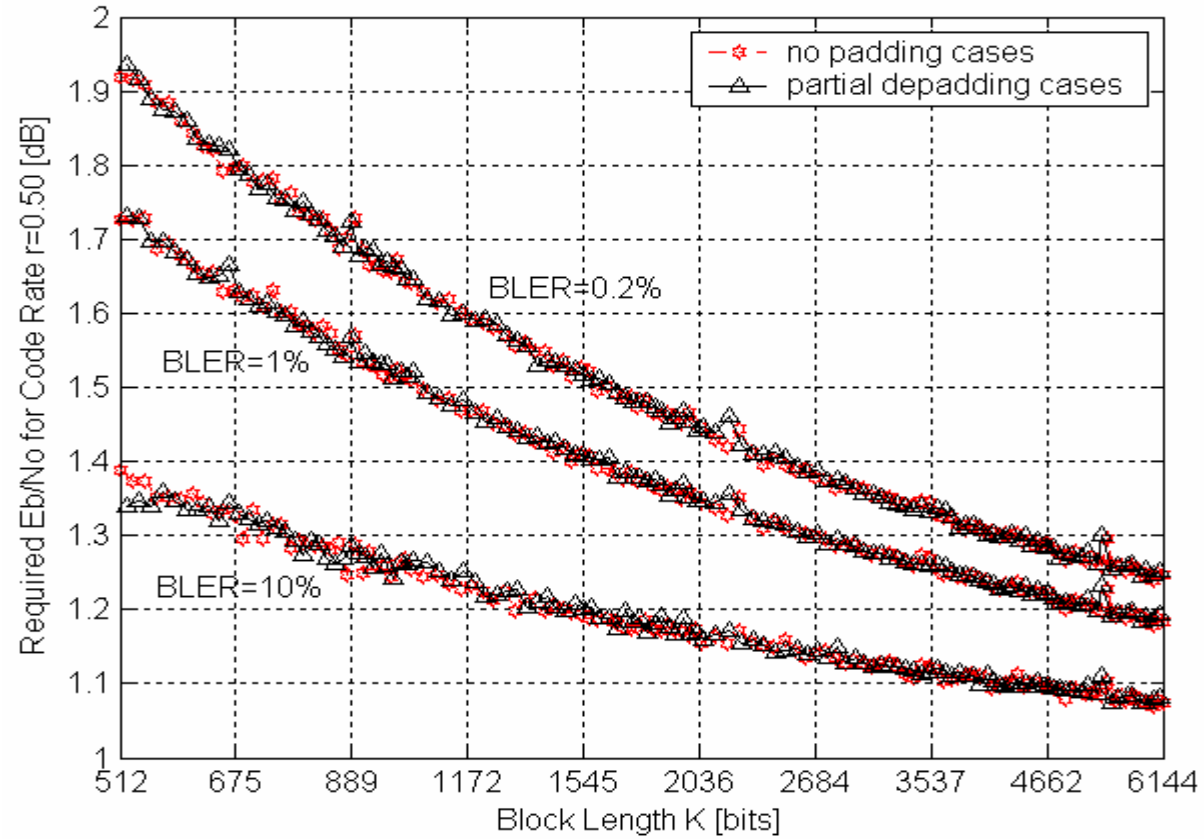


Figure 7 Required E_b/N_0 code rate r=0.5. For each of the partial depadding ($K=S_{n-1}+8$ bits) and no padding cases ($K=S_n$ bits), three curves corresponding to BLER targets of 10%, 1% and 0.2% are shown.

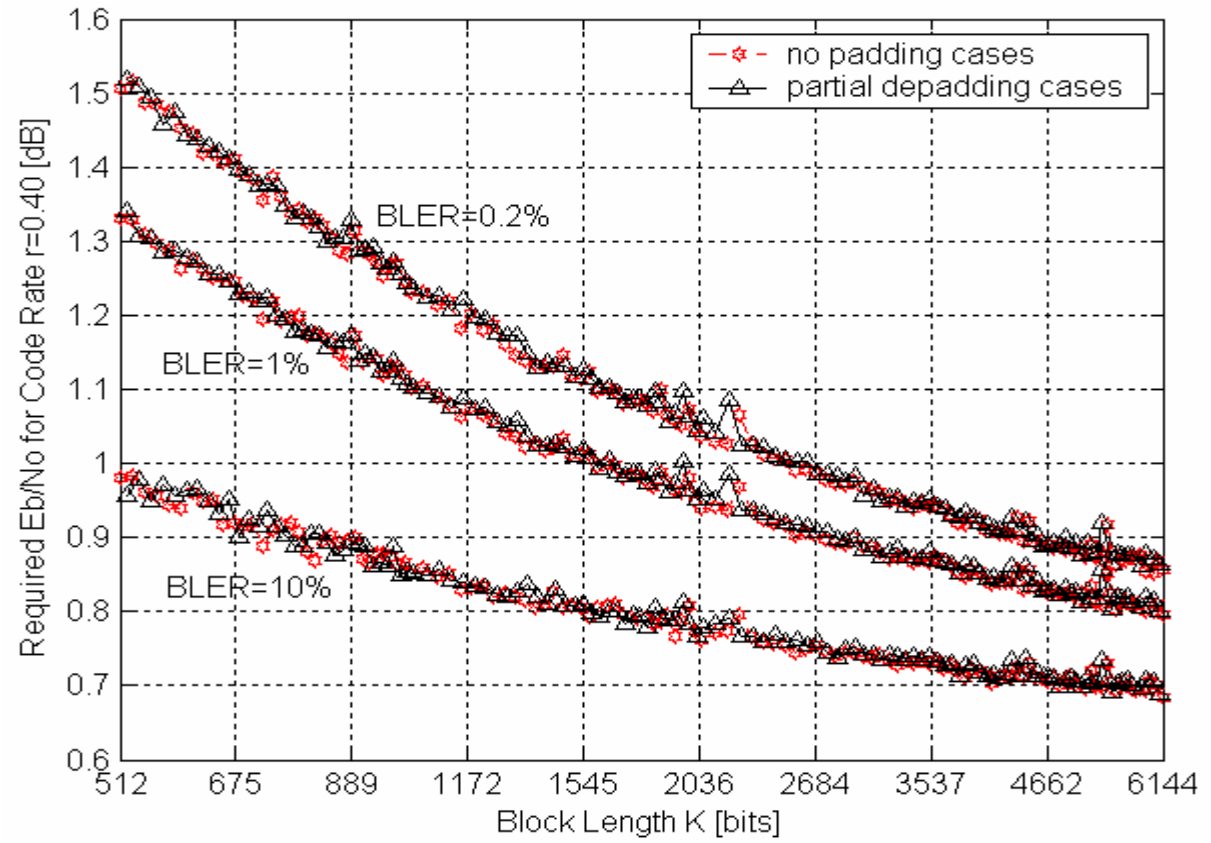


Figure 8 Required E_b/N_0 code rate r=0.4. For each of the partial depadding (K=S_{n-1}+8 bits) and no padding cases (K=S_n bits), three curves corresponding to BLER targets of 10%, 1% and 0.2% are shown.

4. Conclusion

It is proposed to adopt the combined partial depadding and circular buffer rate matching approach for the LTE system.

5. References

[1] 3GPP TS 36.212 v1, "Multiplexing and Channel Coding (Release 8)," 2007.

[2] R1-071514, Broadcom, "Evaluation of possible QPP interleaved turbo code outputs for rate matching," 3GPP TSG RAN WG1#48bis, St. Julians, Malta, Mar. 26– 30, 2007.

[3] R1-071590, Ericsson, "Code shortening procedure for turbo coding," 3GPP TSG RAN WG1#48bis, St. Julians, Malta, Mar. 26– 30, 2007.

[4] R1-071771, Ericsson, "Further Consideration on Code shortening procedure for turbo coding," 3GPP TSG RAN WG1#48bis, St. Julians, Malta, Mar. 26– 30, 2007.

[5] R1-071807, Ericsson, Motorola, Broadcom, Siemens, Samsung, "On Depadding Method for LTE Turbo Coding," 3GPP TSG RAN WG1#48bis, St. Julians, Malta, Mar. 26– 30, 2007.

[6] R1-072604, Ericsson, ETRI, Freescale, ITRI, LGE, Motorola, Qualcomm, Samsung, ZTE, "Way forward on HARQ rate matching for LTE," 3GPP TSG RAN WG1 #49, Kobe, Japan, May 7– 11, 2007.

[7] R1-072621, Nokia Siemens Networks, Nokia, Motorola, Qualcomm, ZTE, Ericsson, LGE, Samsung, TI, ETRI, InterDigital, "LTE rate matching conclusion," 3GPP TSG RAN WG1 #49, Kobe, Japan, May 7– 11, 2007.

[8] R1-072245, Samsung, Qualcomm, LGE, ITRI, "Circular buffer rate matching for LTE," 3GPP TSG RAN WG1 #49, Kobe, Japan, May 7– 11, 2007.

[9] R1-072137, Motorola, "Turbo rate matching in LTE," 3GPP TSG RAN WG1 #49, Kobe, Japan, May 7– 11, 2007.

[10] R1-072138, Motorola, "Redundancy version definition for circular buffer rate matching," 3GPP TSG RAN WG1 #49, Kobe, Japan, May 7– 11, 2007.

[11] R1-073028, Ericsson, "Catastrophic puncturing avoidance," 3GPP TSG RAN WG1#49bis, Orland, USA, June 25– 29, 2007.