3GPP TSG RAN WG1 #48Bis **R1-071514**
St. Julians, Malta, March 26 – 30, 2007

Agenda Item:	**7.5**
Source:	**Broadcom**
Title:	**Evaluation of possible QPP interleaved turbo code outputs for rate matching**
Document for:	**Discussion/Decision**

1 Summary

In RAN1 #48 meeting of 3GPP LTE, QPP (quadratic polynomial permutation) interleave [1] is adopted for turbo code of 3GPP LTE. In the adopted proposal [1] only 188 interleaves are provided. Suppose only multiple of 8 information block sizes are used, there are 575 other block sizes among 40 to 6144 are not listed. To apply turbo codes with those block sizes, 3 possible methods [2-4] are known. In this contribution we provide the following evaluations and suggestion.

1) Study pruning method [2], filling and puncturing method I [3] and filling and puncturing method II [3-4];

2) The simulation shown the performance differences among the three methods are negligible;

3) Based on 2) and requirement of parallel turbo decoding, filling and puncturing method II might be the best solution. Using this method, the number of information bits, the number of check bits from 1st constituent encoder and the number of check bits from 2nd constituent encoder are the same;

4) Since pruning is not used, the output order of turbo encoder shall be different to what given in Rel.6. A straightforward order of the turbo encoder output is suggested in this contribution;

5) Some unexpected simulation performances caused by QPP interleave are given.

2 Modifying methods for information size not in the 188 QPP interleave list

2.1 Pruning

According to the pruning technique in Rel.6 [2], if the information block size L is not in the interleave list, take the smallest size N in the list such that N>L. The dummy bits are padded in positions k= L+1, …, N. Using size N interleave to interleave the information sequence and the padded bits. After the interleaving, dummy bits are pruned away from the output of the interleaved bits, see Fig.1 for details.

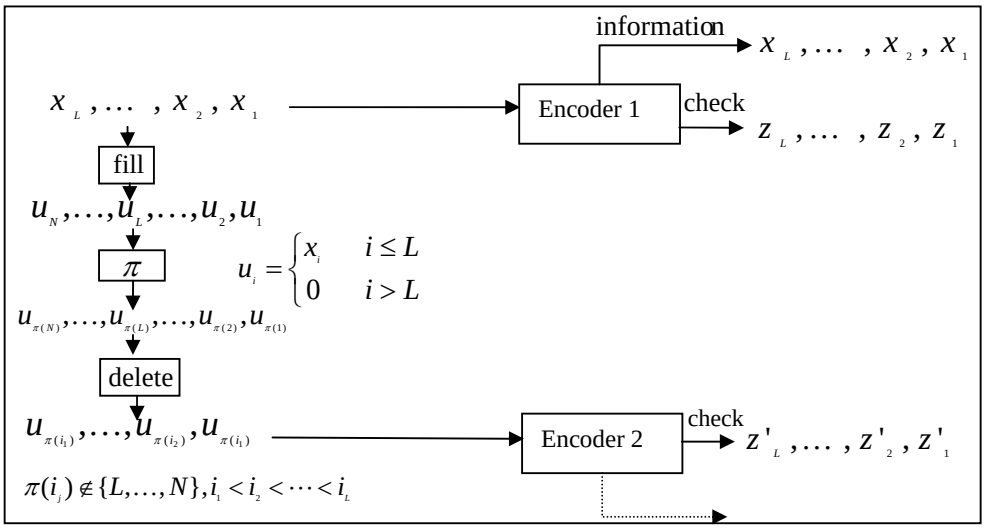

Figure 1 Pruning

The major disadvantage of pruning is that it may add the complexity in parallel turbo decoding. After pruning, the pruned interleave is no longer QPP. Although there exists a memory mapping of parallel decoding for the modified interleave, up to now there is no universal, systematic and algebraic memory mapping method for all possible 575 pruned information block sizes.

2.2 Filling and Puncturing

According to the methods proposed in [3-4], if the information block size L is not in the QPP list, take the smallest size N in the list such that N>L and fill N-L 0 bits at the beginning of the information sequence. Then, turbo encoder will encode the filled information sequence and QPP-interleaved sequence. After that the filling systematic bits will be punctured. If the initial state is 0, then the first N-L check bits of the 1st constituent encoder are all 0. Therefore, these bits can be punctured too. For the 2nd check bits (check bits from 2nd constituent encoder) corresponded to the filling bits, there are two proposed methods, namely

Method I) [3] Keep all 2nd check bits, see Fig 2 for detail. This will cause some rate loss but may gain the performance.

Method II) [3-4] Puncture the 2nd check bits that corresponded to the filling bits, see Fig.3 for details. This will not cause rate loss but it may cause some performance loss.

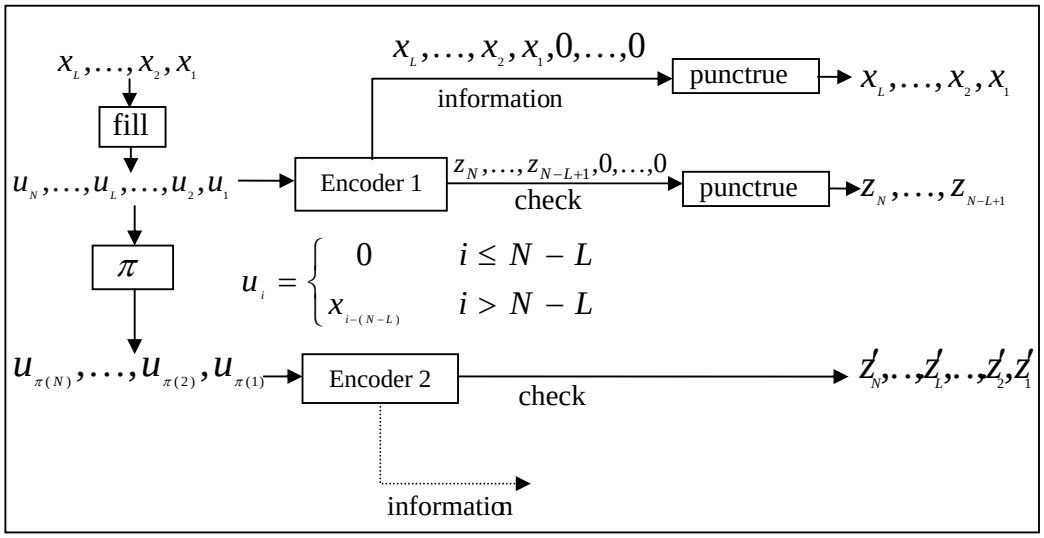

Figure 2 Filling and puncture I

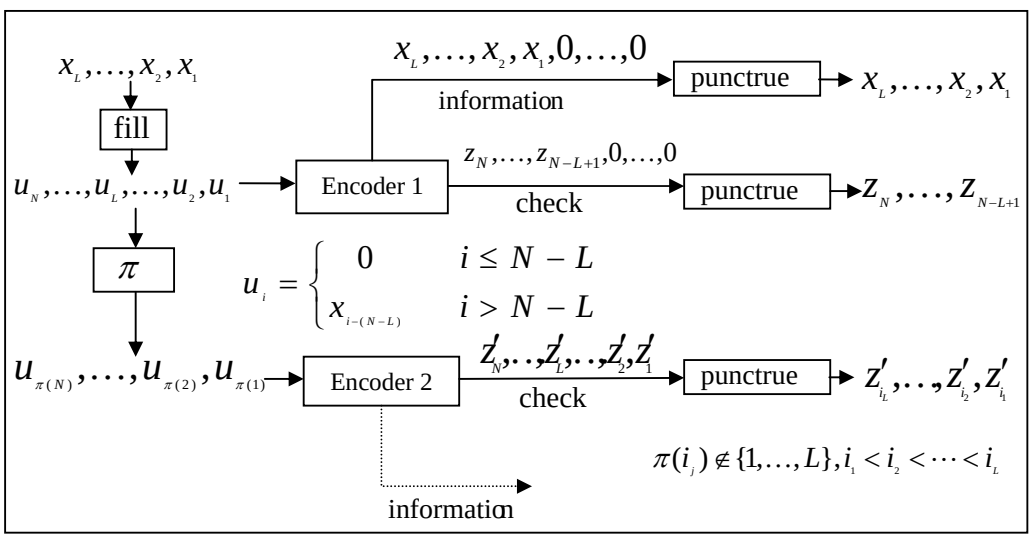

Figure 3 Filling and puncture II

With filling and puncture methods, decoding always uses the listed QPP. Therefore, parallel turbo decoding can use universal, systematic and algebraic memory mapping such as those given in [5] and [6].

2.3 Performance comparisons on AWGN channel

In this subsection, the performances of the three methods on information block size 520 with the interleave size 528, on information block size 616 with the interleave size 624, on information block size 808 with the interleave size 816, on information

block size 1032, 1040 and 1048 with the interleave size 1056, on information block size 2056, 2064, 2072, 2080, 2088, 2096 and 2104 with the interleave size 2112, and on information block size 5096 with the interleave size 5120 are given. The noise channel is BPSK AWGN. In the following the performance curves of filling and puncturing method I is normalized with the rate loss.

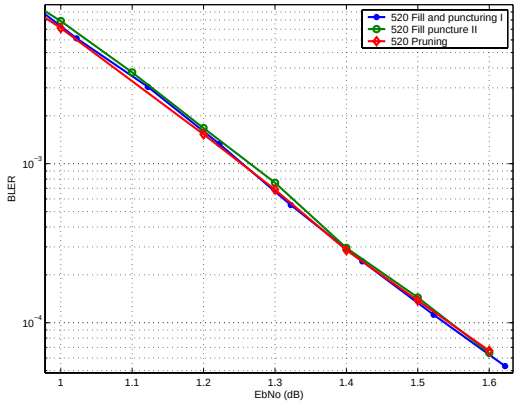

Figure 4 Information size 520 with interleave size 528

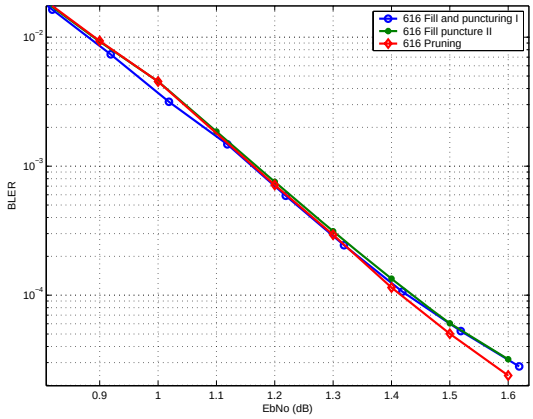

Figure 5 Information block size 616 with interleave size 624

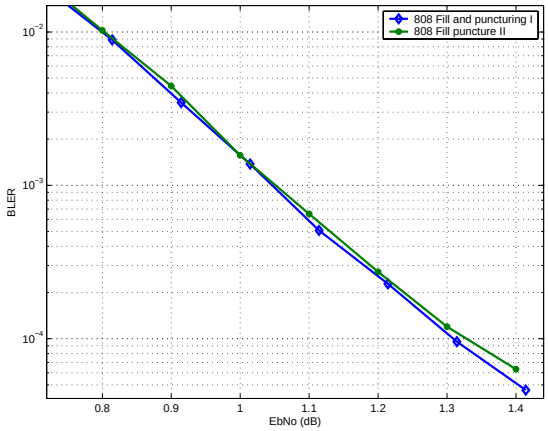

Figure 6 Information block size 808, interleave size 816

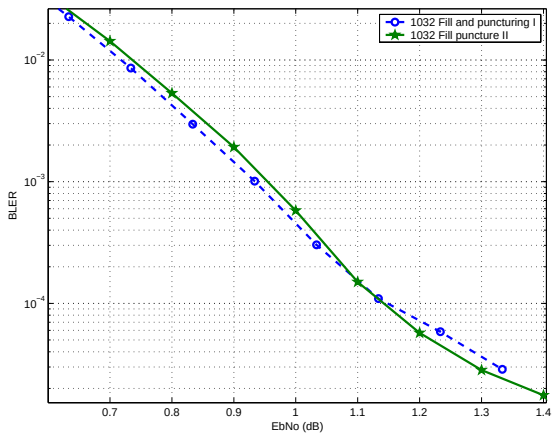

Figure 7 Information block size 1032, interleave size 1056

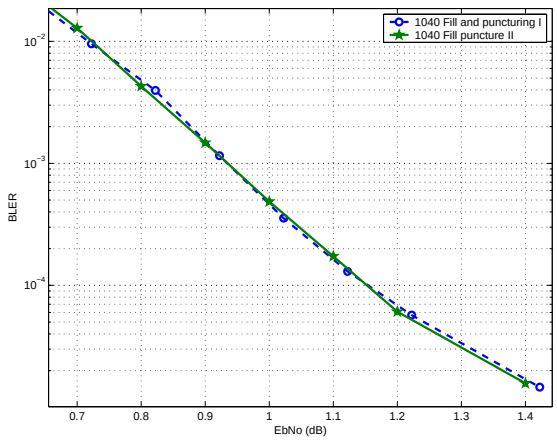

Figure 8 Information block size 1040, interleave size 1056

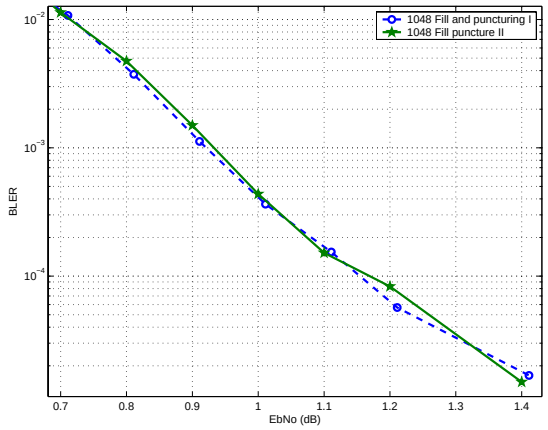

Figure 9 Information block size 1048, interleave size 1056

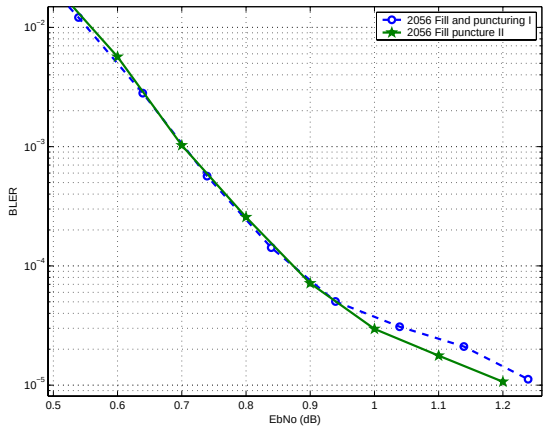

Figure 10 Information block size 2056, Interleave size 2112

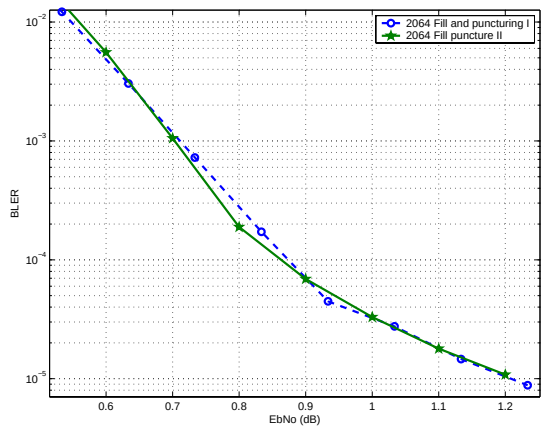

Figure 11 Information block size 2064, Interleave size 2112

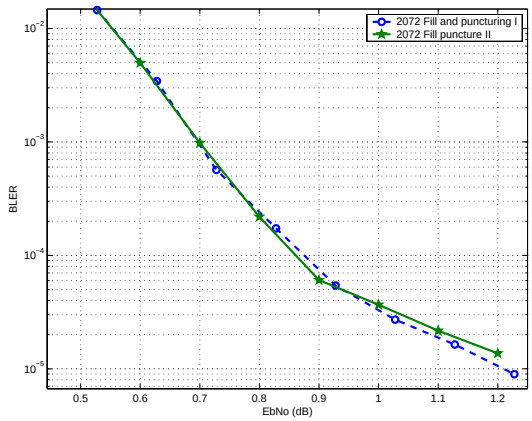

Figure 12 Information block size 2072, Interleave size 2112

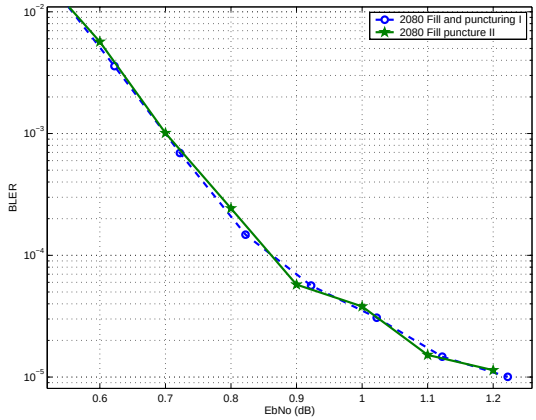

Figure 13 Information block size 2080, Interleave size 2112

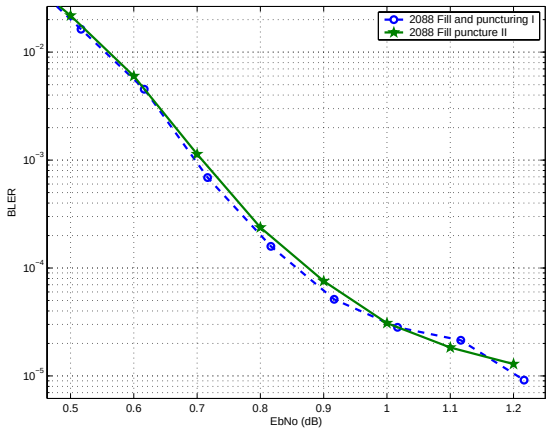

Figure 14 Information block size 2088, Interleave size 2112

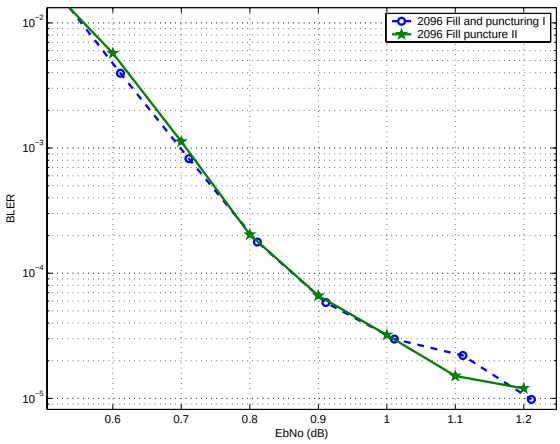

Figure 15 Information block size 2096, Interleave size 2112

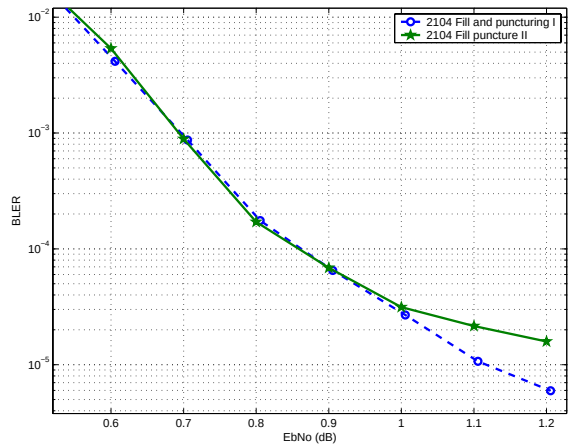

Figure 16 Information block size 2104, Interleave size 2112

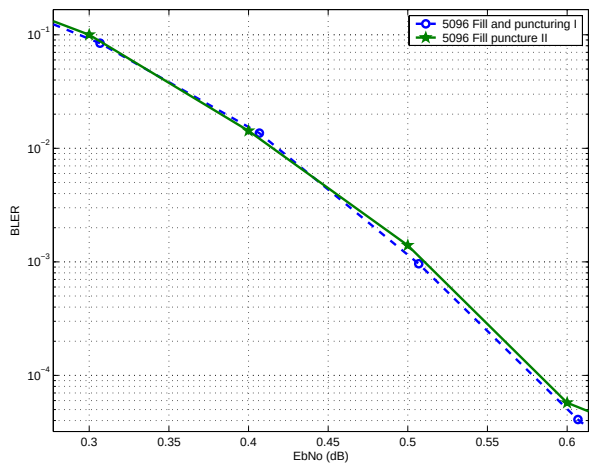

Figure 17 Information block size 5096, Interleave size 5120

The above examples show that the performance differences among three methods are negligible. Therefore, to support the parallel turbo decoding and to simplify the output sequence for rate matching it is better use Filling and Puncture Method II.

3 Turbo output for rate matching based on filling and puncturing method II

One of the good properties of Filling and Puncturing Method II is that the output of the turbo encoder is always three times of information bits, except the 12 flush bits. With this outputs, the bits sequence sent out for rate matching can naturally be ordered as follows.

Let π be a QPP interleave of size N listed. Suppose the information sequence is L. Then the number of fill bits is F=N-L. In this case the first F bits sent to turbo encoder shall be all 0. We denote them by $x_1 = x_2 = \cdots = x_F = 0$. Let the L information bits be

denoted by $x_{F+1}, x_{F+2} \ldots, x_N$. Then the bits sent to turbo encoder are $0,0,0, x_{F+1}, \ldots, x_N$. When the initial state of the 1st constituent encoder is 0, the check bits of this encoder can be denoted by $0, \cdots, 0, z_{F+1}, z_{F+2}, \ldots, z_N$. The interleaved information bits sent to 2nd constituent encoder are $x_{\pi(1)}, x_{\pi(2)}, \ldots, x_{\pi(N)}$. Let us denote the corresponded check bits by $z'_1, z'_2, \ldots, z'_N$. Among them F bits are corresponded to the filling bits. Suppose they are in the positions $j_1, j_2, \ldots, j_F \in \{1,2,\ldots,N\}$ such that $\{\pi(j_1), \pi(j_2), \ldots, \pi(j_F)\} = \{1,2,\ldots,F\}$. Now define integers $i_F < i_{F+1} < \cdots < i_N$ such that $\{i_1, i_2, \ldots, i_L\} = \{1,2,\ldots N\} \setminus \{j_1, \ldots, j_F\}$, i.e. i_1 is the first check bit that is not corresponded to the filling bits, i_2 is the second check bit that is not corresponded to the filling bits, i_3 is the third check bit that is not corresponded to the filling bits, etc. Then the order of the first 3L output bits sent to rate matching shall be

$$x_{F+1}, z_{F+1}, z'_{i_1}, x_{F+2}, z_{F+2}, z'_{i_2}, \ldots . x_N, z_N, z'_{i_L}$$

To keep the consistent with the above order, the output bits for trellis termination can be

$$x_{N+1}, z_{N+1}, x'_{N+1}, z'_{N+1}, x_{N+2}, z_{N+2}, x'_{N+2}, z'_{N+2}, x_{N+3}, z_{N+3}, x'_{N+3}, z'_{N+3}$$

this is different to what is given in Rel.6, i.e.

$$x_{N+1}, z_{N+1}, x_{N+2}, z_{N+2}, x_{N+3}, z_{N+3}, x'_{N+1}, z'_{N+1}, x'_{N+2}, z'_{N+2}, x'_{N+3}, z'_{N+3}$$

where $x_{N+1}, x_{N+2}, x_{N+3}$ are termination bits sent to 1st constituent encoder with parity bits $z_{N+1}, z_{N+2}, z_{N+3}$, and $x'_{N+1}, x'_{N+2}, x'_{N+3}$ are termination bits sent to 2nd constituent encoder with parity bits $z'_{N+1}, z'_{N+2}, z'_{N+3}$.

4 Some observations

Usually with the same code rate the longer the information block size, the better performance the turbo code provided. In fact, when the block sizes have big differences the turbo codes with QPP interleaves performed as expected, see Fig.18.

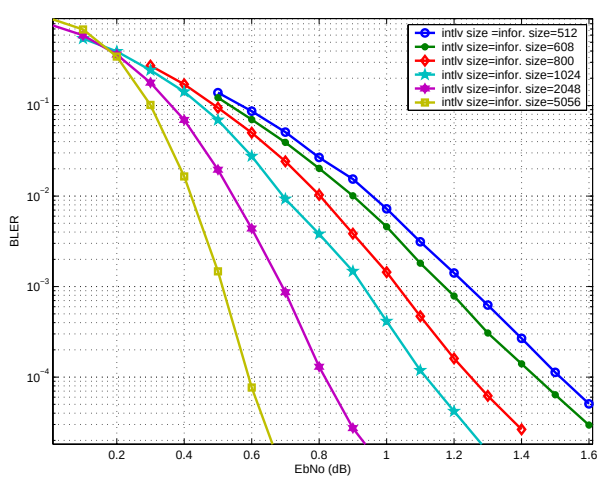

Figure 18 Rate 1/3 sizes 512,608,800,1024,2048,5056

However, when the block sizes difference is smaller, say about 100-200, the longer codes may perform worse. In the following, examples are given by performance curves as well as tables. We number the 1 for the best performance.

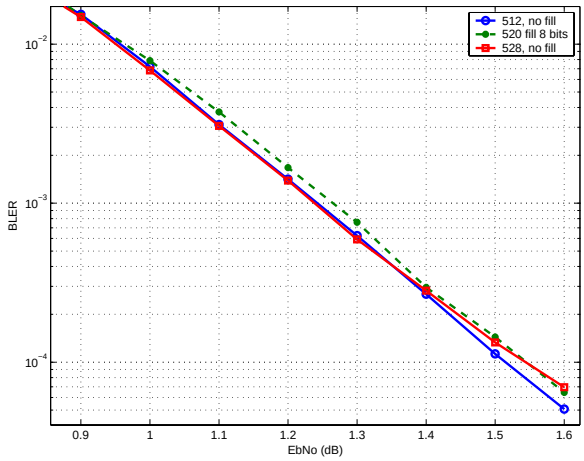

Figure 19 Sizes 512, 520 and 528

Information block size	Expected BLER @ 1.5dB rank	Simulated BLER @ 1.5dB rank
512	3	1
520(fill)	2	3
528	1	2

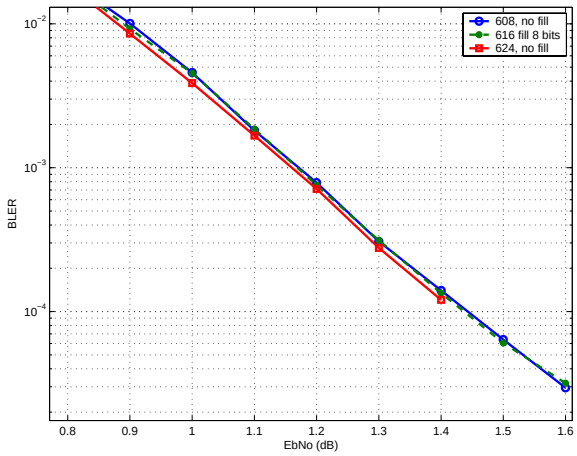

Figure 20 size 608,616 and 624

Information block size	Expected BLER @ 1.4dB rank	Simulated BLER @ 1.4dB rank
608	3	2
616(fill)	2	2
624	1	1

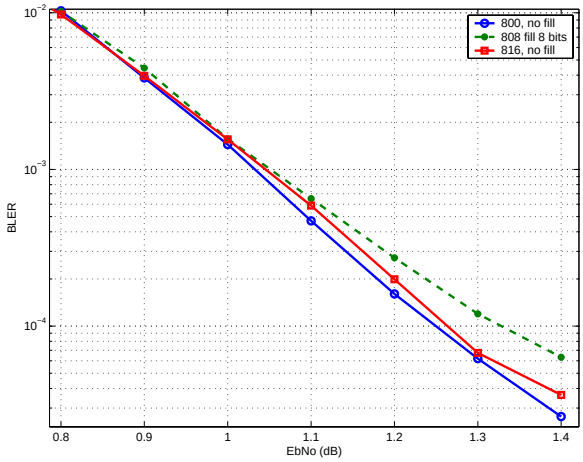

Figure 21 Sizes 800,808 and 816

Information block size	Expected BLER @ 1.3dB rank	Simulated BLER @ 1.3dB rank
800	3	1
808(fill)	2	3
816	1	2

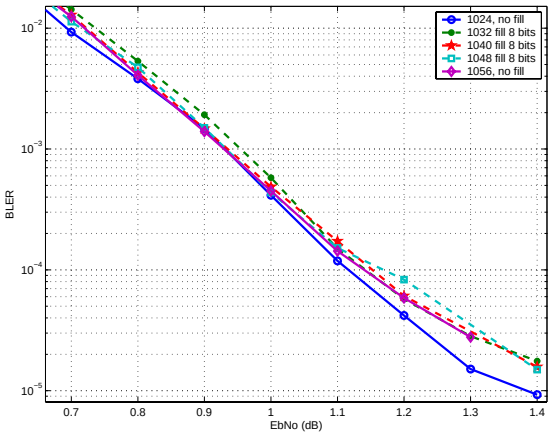

Figure 22 Sizes 1024,1032,1040,1048 and 1056

Information block size	Expected BLER @ 1.2dB rank	Simulated BLER @ 1.2dB rank
1024	5	1
1032(fill)	4	2
1040(fill)	3	4
1048(fill)	2	5
1056	1	3

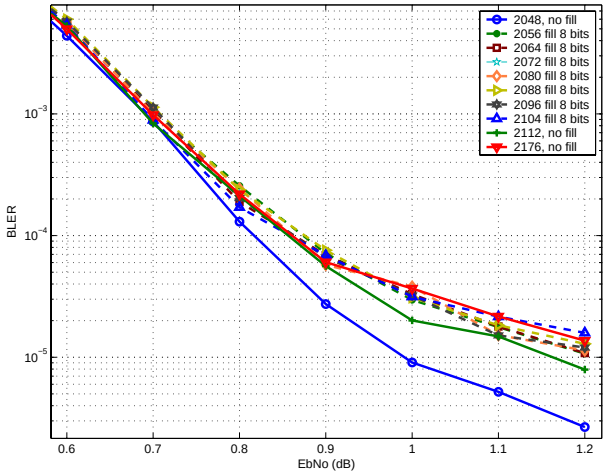

Figure 23 Sizes 2048,2056,2064,2072,2080,2088,2096,2104,2112 and 2176

Information block size	Expected BLER @ 0.9B rank	Simulated BLER @ 0.9dB rank
2048	10	1
2056(fill)	9	9
2064(fill)	8	7
2072(fill)	7	4
2080(fill)	6	3
2088(fill)	5	10
2096(fill)	4	6
2104(fill)	3	7

2112	2	2
2176	1	4

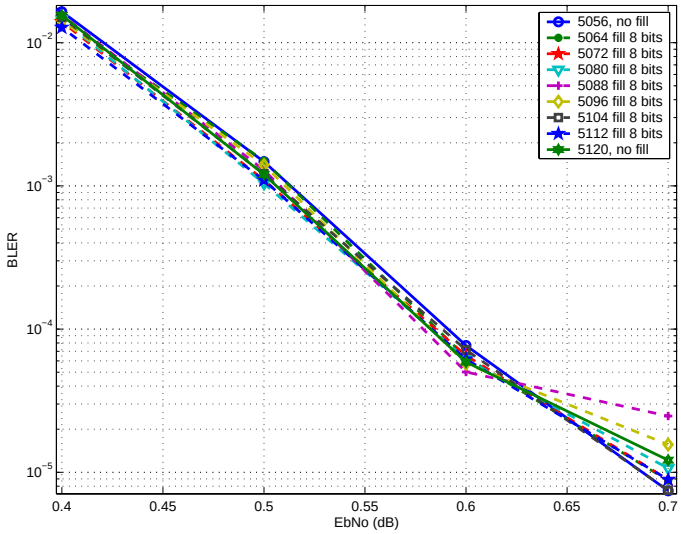

Figure 24 Sizes 5056,5064,5072,5080,5088,5096,5104,5112 and 5120

Information block size	Expected BLER @ 0.8B rank	Simulated BLER @ 0.8dB rank
5056	9	1
5064	8	8
5072	7	6
5080	6	4
5088	5	3
5096	4	9
5104	3	5
5112	2	6
5120	1	2

5 References

[1] Ericsson, Motorola, Broadcom, ZTE, Nokia, TI, Qualcomm Europe, Freescal, CATT, ITRI, NTT DoCoMo, Panasonic, Samsung, Mitsubishi, Nortel, “QPP interleaver parameters,” 3GPP TSG RAN WG1 #48 R1- 071195.

[2] 3GPP TS 25.212 V6.8.0 (2006-06)

[3] Motorola, Contention-free Interleaver designs for LTE Turbo Codes, 3GPP TSG RAN WG1 #47bis R1-070054

[4] Ericsson, Performance Comparison of LTE Turbo Internal Interleaver Proposals, 3GPP TSG-RAN WG1 #47bis R1-070464

[5] A. Nimbalker, T. E. Fuja, D. J. Costello, Jr. T. K. Blankenship and B. Classon, "Contention-Free Interleavers," *IEEE ISIT 2004*, Chicago, USA, June 27 – July 2, 2004

[6] Broadcom, "Flexible and Formulaic Collision-free Memory Accessing for Parallel Turbo decoding with Quadratic polynomial permutation (QPP) Interleave," 3GPP TSG-RAN WG1 #47bis R1-070618