

Turbo Codes: Interleavers Types

M.Boulan and M.A.Ambroze

Centre for Security, Communications and Network Research,
Plymouth University, Plymouth, UK
e-mail: info@cscan.org

Abstract

One of the most important and critical component in achieving good performances with iterative decoding of Turbo codes is the choice of the interleaver. Indeed the interleaver is a key component because it allows the decoded extrinsic information to be nearly independent of the observed data in the decoder. This publication describes the structure of different types of interleavers and their behaviour thanks to a turbo encoder simulator developed by the author himself.

Keywords

Error correcting codes, Turbo codes, interleaving, iterative decoding, MAP decoding, BCJR, Recursive Systematic Convolutional codes

1 Turbo codes

Turbo codes were discovered in 1993 by Claude Berrou and his team (Berrou et al, 1993). The typical parallel structure of a Turbo encoder uses two identical Recursive Systematic Convolutional (RSC) codes, separated by one interleaver (Figure 1).

The weight distribution of these codes being close to random codes, their performance is remarkable. However, these codes have a low free distance and lose efficiency for a low Signal to Noise Ratio. Therefore, an optimal interleaver has to improve the weight distribution and the minimal distance of the Turbo code. Thus the interleaving function plays an important role on the performance of Turbo codes.

The purpose of the interleaver is to offer each encoder an uncorrelated (random) version of the information, resulting in parity bits from each independent RSC encoder.

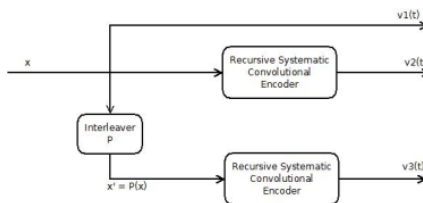


Figure 1: 1/3-rate Turbo encoder

2 Turbo decoding

There are two main algorithm to decode the code message. One is a modification of the well-known Viterbi Algorithm (Viterbi, 1967) and is called Soft Output Viterbi Algorithm (SOVA). It consists in finding the most probable output sequence by drawing the Trellis diagram. The other one is the Maximum A Posteriori Algorithm (MAP), which is often referred to as the BCJR (after Bahl, Cock, Jelenik and Raviv) who proposed it in 1974 (Moon, 2005). MAP is very similar to the Viterbi Algorithm but while Viterbi computes the maximum likelihood codeword (a priori), MAP computes the a posteriori probabilities of symbols.

SOVA and MAP are usually comparable, however the MAP algorithm is used in the most of Turbo decoders because it minimises the bit error probability (Rekh et al, 2005).

3 Interleaver Types

3.1 Block Interleaver

The interleaver the most widely used is the block interleaver, also called rectangular interleaver. It considers the initial positions are written row-wise in a matrix $L1 * L2$. These positions are permuted by reading column-wise the matrix.

I	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Interleaver(i)	0	5	10	1	6	11	2	7	12	3	8	13	4	9	14

Table 1: Block interleaver, $L1=3$ $L2=5$

Ramsey (1970) proves we can construct an optimal block interleaver by choosing $L1$ (number of rows) and $L2$ (number of columns) such as:

- $L1 + 1$ and $L2$ are prime
- $L1 + 1 < L2$.

Therefore the case above (Table 1) is an optimal block interleaver.

3.2 Diagonal Interleaver

The diagonal interleaver is a modification of the block interleaver (Morelos-Zaragoza, 2001). Instead of reading column-wise, the matrix is read diagonally from left to right and top to bottom.

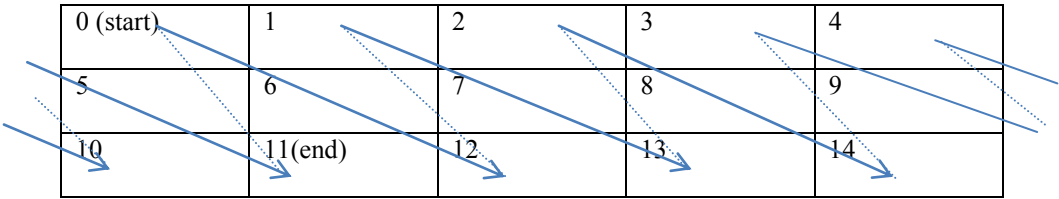


Table 2: Diagonal process

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Interleaver(i)	0	6	12	1	7	13	2	8	14	3	9	10	4	5	11

Table 3: Diagonal interleaver, L1=3 L2=5

3.3 Helical Interleaver

The helical interleaver is also another version of the block interleaver, proposed by Adrian Barbulescu and Silvio Pietrobon (1994).

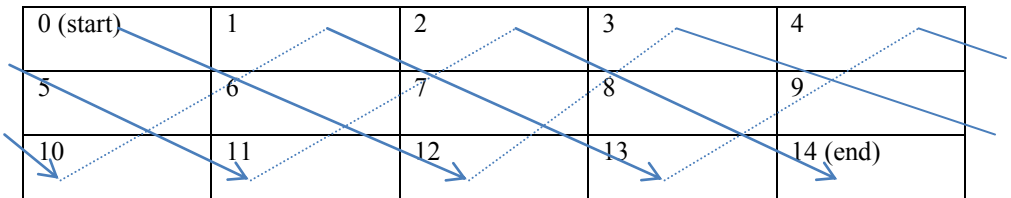


Table 4: Helical process

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Interleaver(i)	0	6	12	3	9	10	1	7	13	4	5	11	2	8	14

Table 5: Helical interleaver, L1=3 L2=5

4 Results

Figure 2. shows the results of the simulation for different types of interleavers (Block, Diagonal & Helical). The chosen interleaver size is 125 ($L1=5, L2=21$) for each of them. The simulation consists in encoding a 125-length message with a 1/3-rate Turbo code, then modulate the encoded message, add noise and finally decode it with a BCJR decoder (8 iterations).

From the simulation we find that the diagonal interleavers give the best performance.

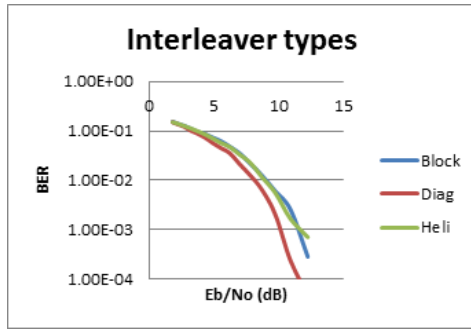


Figure 2: Interleaver types

5 References

- Berrou, C., Glavieux, A., and Thitimajshima, P., (1993) Near Shannon limit error correcting coding and decoding: Turbo codes. *IEEE International Conference on Communications*, pages 1064-1070.
- Barbulescu, S.A. and Pietrobon, S.S., (1994), Interleaver Design for Turbo codes. *Electronic Letters*, 30(25), 2107-2108.
- Moon, T.K. (2005), Error Correction Coding: Mathematical Methods and Algorithms, Wiley.
- Morelos-Zaragoza, R.H., (2001), The Art of Error Correcting Coding. Wiley.
- Ramsey, J., (1970), Realization of Optimum Interleavers. *IEEE Transactions on Information Theory*, 16(3), 338-345.
- Rekh, S., Subha, S. and Shanmugam, A., (2005). Optimal choice for turbo codes. *Academic Open Internet Journal, Vol. 15, 2005*
- Viterbi, A., (1967), Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, EEE, 13, apr 1967.*