

# A Review on Punctured Analysis of Turbo codes

Shridhr.B.Devamane<sup>1</sup>, Dr.R.L.Itagi<sup>2</sup>

Assistant Professor, Electronics & Communication, APS College of Engineering, Bangalore, India.<sup>1</sup>

Professor, K.L.E’s Society’s Dr.M.S.Sheshgiri College of Engineering, Belgaum, India.<sup>2</sup>

**Abstract:** Turbo codes have performance superior than all other coding techniques. Main factors that make turbo codes so efficient include parallel concatenation structure of the encoding system recursive convolutional encoder, interleave, puncturing pattern and iterative decoding. Puncturing is the tradeoff between rate and performance. Puncturing increases code rate without increasing complexity for code rate from 1/3 to 1/2 or more and decreases free distance of code. The redundant bits in coding decrease the bandwidth efficiency. In this paper the desired turbo coding rates providing methods using the puncturing process and also the effect of this process on BER.

**Keywords:** Turbo coding, Puncturing, Interleaver, Iterative decoding, Convolutional code.

## I. INTRODUCTION

The turbo codes constitute a family of error correcting codes which make possible to achieve the theoretical limit of correction predicted by Shannon there is more than 50 years. These codes, invented at the beginning of the years 1990, are obtained by the concatenation of two or several convolutional codes of low complexity, separated by a interleaving function introducing the diversity. Their decoding calls an iterative process (or turbo) using two or several elementary decoders which exchange information of reliability, called (extrinsic information), in order to improve the correction with the iterations. the effect of puncturing on the turbo-codes according to the following criterion: the power of coding is uniformly distributed to all the bits of information . Recalling that puncturing is used to increase the rate of code, and it increases also the groups of error at output of the elementary decoder.

its powerful error correcting capability, reasonable complexity, and flexibility in terms of different block lengths and code rates, number of memory elements as shown in Figure turbo codes are widely used in different communication system like satellite communication, CDMA, mobile communication etc.

A turbo encoder actually consists of two systematic convolution encoders [6], which consist of an interleaver unit with a specified encoding structure. We send n number of input data bits to the encoder. This data goes to the first convolution encoder where an interleaved data is passed through the second convolution encoder. From above Fig. it is clear that systematic bit is same as input data bits and with the help of two RSC encoders; we get the two ‘n’ bits parity sequences. These three sequences create rate 1/3.

## II. ELEMENTS OF TURBO CODE

### TURBO ENCODER:

The turbo encoder consists of two recursive systematic convolutional (RSC) encoders operating in parallel manner with the input data bits. We can design either a 4-state or 8- state turbo encoder with different rate. An interleaver is used between two systematic convolution encoder as shown in figure 1.

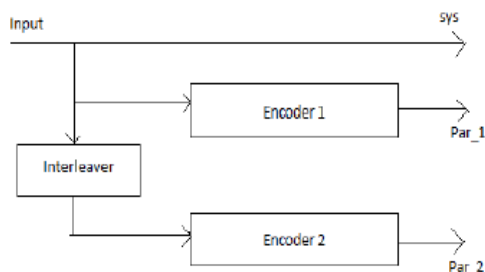


Fig 1. Turbo Encoder

Here, we can achieve the rate 1/3, without puncturing and 1/2, with puncturing method. Other code rates are also obtained by puncturing mechanism. When turbo encoded data is applied to turbo decoder through Base-Band transmission over AWGN channel, Log-Map decoding structure provides performance nearer to Shannon’s limit with less complexity with respect MAX-Log-MAP. Due to

## III. TURBO DECODER

Turbo codes offer the ability to build powerful codes that utilize a relative simple soft-decision decoding algorithm. This is possible because the turbo decoder is constructed from two identical serial decoding blocks that share information. The turbo code decoders often work iteratively (loop wise) by sharing the a priori information obtained from the log-likelihood ratio (a posteriori information) of the previously cascaded decoder. The constituent decoders are the optimal decoders for the component codes used by the turbo encoder.

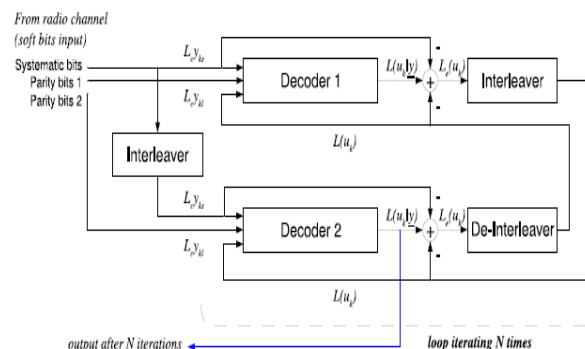


Fig 2. Turbo Decoder

In the turbo decoder, each component decoder provides an output containing extrinsic information. These outputs are feed as a-priori information to the complementary decoder. The extrinsic output indicates the likelihood of a bit to be 0 or 1 excluding the value of this bit received from the channel, and instead based on the information of the surrounding bits and the constraints imposed by the code being used.

The turbo decoder shown in above fig operates iteratively. In the first iteration the first component decoder takes as input the stream of bits received from the channel corresponding to the bits encoded by the first encoder in the turbo encoder. Since it is the first iteration, the values of the a-priori information, not known at the beginning, are ignored or alternatively set to zero. The soft output provided by the first encoder is then used to calculate the extrinsic information, and it is then feed as a-priori information to the second component decoder. Then the second decoder estimates the bits relying now in the information received from the channel, associated to the stream encoded by the second encoder in the turbo encoder, and the a-priori information supplied by the first decoder. Next the second iteration begins; the first decoder takes again as input the bits from the channel and in addition the a-priori information supplied by the second decoder in the previous iteration. As before, the second decoder takes as input the a-priori information from the first decoder and the received bits from the channel.

This cycle is repeated until a given criteria is met, like for example after a given number of iterations. In each iteration the BER and BLER tends to decrease, nevertheless the gain in performance decreases as the number of iterations increases. When the iterations are completed the estimated sequence is output from the second component decoder.

#### IV. PUNCTURING

Using the technique of puncturing, it is possible to provide a different turbo-coders of various rate, starting from only one standard code which rate is generally equal to 1/3. The codes obtained present performance very close to those obtained with optimal codes (not punctured) of the same rate. However, in condition that the matrix of puncturing are well selected. This puncturing technique makes it possible to simplify the trellis since in each state converges, only, two branches and not  $2^M$  branches as for an optimal code of rate  $M/N$ . This decreases the complexity of decoding using the soft output Viterbi algorithm (SOVA). Moreover, the same turbo-decoder can then be used for a whole of compatible punctured codes. A simple method consists to puncture only the parity bits which differs from the information bits, in this case we should choose a puncturing matrix with information bits coefficient equal to 1. Such a matrix is known as “invariant” and this technique of puncturing is called “parity data puncturing”, noted PDP will be adopted during the simulation of the transmission system described previously.

Puncturing is usually introduced to increase the rate of the turbo code. Equivalently, once may use higher rate

component codes. A puncturing matrix  $P$  of period  $p$  is applied to a turbo code having  $N$  output branches can be represented by

$$P = \begin{bmatrix} g_{11} & \dots & g_{1p} \\ \dots & g_{k} & \dots \\ g_{N1} & \dots & g_{Np} \end{bmatrix}$$

where even' row corresponds to one output branch, i.e. the first row corresponds to the systematic trench, the second row corresponds to the first parity branch and so on. Note that  $q, e(0, 1)$  where 0 implies that the corresponding bit is punctured.

#### V. RESULTS AND DISCUSSIONS

The simulation results for the puncture and Un punctured turbo code are as shown below

Un Punctured Turbo code having code rate 1/3:

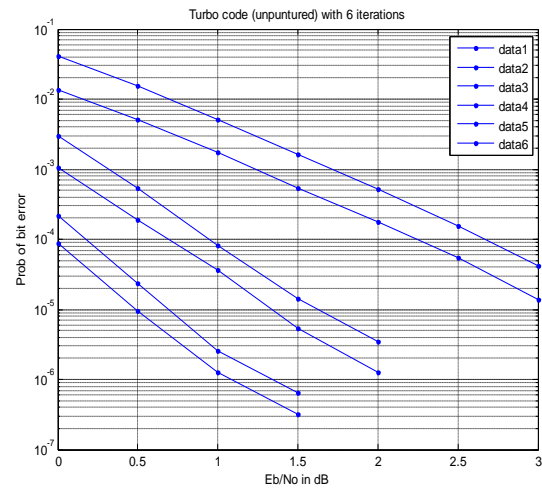


Fig 3. Turbo code with Unpunctured with data rate 1/3

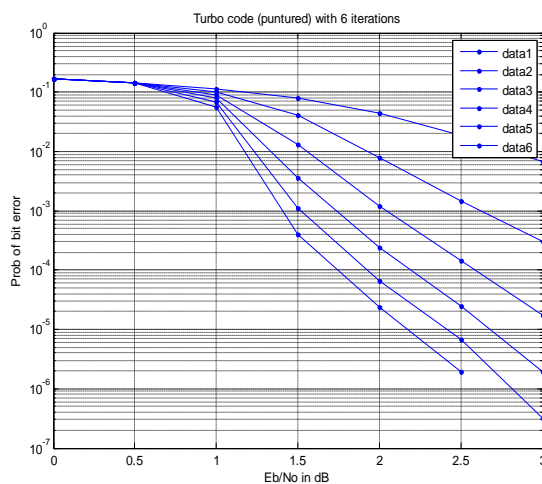


Fig 4. Turbo code with Punctured with data rate 1/3

#### VI. CONCLUSION

In this paper it was concluded that the performance of the turbo codes can be improved by studying the puncturing methods. The simulation results are obtained for both the punctured and un punctured turbo code having the code rate of 1/3.

**REFERENCES**

- [1] CAIRE, G., and BIGLIERI, E.: 'Parallel concatenated codes with unequal error protection', *IEEE Trans. Commun.* 1998, 46,(5), pp.565-567.
- [2] . KIM, B., and KWON, S.: 'A new soft handover scheme using punctured turbo codes in the wideband CDMA system'. Proceedings of the IEEE Conference on VTC, 2001, pp. 1420-1424
- [3] Berrou, A. Glavieux, P. Thitimajshima, "Near Shannon limit error correcting coding and decoding: Turbo-codes", Proc.ICC'93, Geneva,Switzerland,May 1993,pp 1064-1070.
- [4] C. Berrou, A. Glavieux, "Near Optimum Error Correcting Coding and Decoding: Turbo-Codes", *IEEE Transactions on Communications*, Vol. 44, No. 10, October 1996.
- [5] D. Haccoun and G. B' egin, "High-rate punctured convolutional codes for Viterbi and sequential decoding," *IEEE Trans. Commun.*, vol. 37, no. 11, pp. 1113–1125, Nov. 1989.
- [6] J.Hagenauer, "Rate compatible punctured convolutional codes and their applications,"*IEEE Trans. Commun.*, vol. 36, no. 4, pp. 389–400, Apr.1988.
- [7] J.Hagenauer, E. Offer, L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes", *IEEE Trans. Inform.Theory*, vol. 42, no. 2, pp. 429-445, March 1996.
- [8] Narushan Pillay and HongJun Xu, " Dual-Repeated- Punctured Turbo Codes on AWGN channels" in *IEEE AFRICON 2009*.
- [9] D. Haccoun and G. B' egin, "High-rate punctured convolutional codes for Viterbi and sequential decoding," *IEEE Trans. Commun.*, vol. 37, no. 11, pp. 1113–1125, Nov. 1989.
- [10] "O. Ac ikel and W. E. Ryan, "Punctured turbo-codes for BPSK/QPSK channels," *IEEE Trans. Commun.*, vol. 47, pp. 1315–1323, Sept. 1999.