Turbo Codes in the CCSDS Standard for Wireless Data

Teodor B. Iliev[1] and Dimitar I. Radev[2]

***Abstract* – A turbo code has been included in the CCSDS channel coding standard for space telemetry. Many future missions with critical link budgets will benefit of its large coding gain. This paper will review the performance of the proposed Turbo Code configurations of the CCSDS telemetry channel coding standard, and we will discuss how this may be implemented in reconfigurable hardware.**

***Keywords* – Turbo code, Interleaver, Code rate, Convolutional code.**

I. INTRODUCTION

Turbo codes, introduced in 1993 [1], represent a great advancement in coding theory. Their excellent performances, especially at low and medium signal-to-noise ratios, have raised up an enormous interest for applications. Currently, even if many research issues are still open, the success of turbo codes is growing, and their introduction in many international standards is in progress (among them, the UMTS standard for third generation personal communications, and the DVB-T ETSI standard for Terrestrial Digital Video Broadcasting). This is also the case of the CCSDS (Consultative Committee for Space Data Systems) standard for space telemetry, which has often represented a benchmark for new technologies. Recently, the CCSDS channel coding standard has been updated to include turbo codes [2]. Turbo codes can now be employed as an alternative to the old standard codes: a (255,223) Reed-Solomon code, a 64-state rate-1/2 convolutional code, and their serial concatenation through an interleaver. A number of missions have already manifested their intend to use turbo codes, such as Rosetta, Lunarsat, and Mars Express. The BER/FER performances of the CCSDS turbo code have been largely studied: it has been pointed out that an additional coding gain of 2.5 dB can be achieved with respect to the serial concatenated scheme of the standard.

II. THE CCSDS TURBO CODE

The CCSDS Telemetry Channel Coding standard [3] uses a turbo code with two component codes with selectable rates and block lengths. The encoder for this code is shown in Figure 1. Two rate R=1/4 recursive convolutional encoders are used to generate R=1/2, 1/3, 1/4, and R=1/6 turbo codes. The two switching matrices combine these rates, where a solid circle means every symbol is taken, and an open circle means every other symbol is taken. The feedback polynomial $h_0 = D^4 + D + 1 = 23$ of both encoders is a primitive polynomial of degree 4, and the feedforward polynomials are $h_1 = 33$, $h_2 = 25$, and $h_3 = 37$. The various rates are achieved by using the connections shown on the right-hand side of Figure 1. For example, to achieve R=1/4 the four outputs are the systematic bit, the second and third parity bits from encoder 1, and the first parity bit from encoder 2.

Trellis termination is accomplished by first terminating code number 1, then code number 2, by equating the input bit to the feedback bit for four symbol clock ticks. This causes the encoders to be loaded with the all-zero state.

The interleaver is a block permutation interleaver that has good spreading factors and, unlike the S-random interleavers, has an algorithmic implementation and is scalable. The interleavers are fashioned according to Berrou's analytical algorithm [1]. The CCSDS standard allows five interleaver lengths $K_1 = 1784$, $K_2 = 3568$, $K_3 = 7136$, $K_4 = 8920$, and $K_5 = 16384$. The first four block lengths are chosen to be compatible with an outer Reed–Solomon code with interleaving depths of 1, 2, 4, and 5, respectively. The largest block length is allowed for those users requiring the most power efficiency and able to tolerate a larger decoder latency.

The corresponding codeblock lengths in bits, n=(k+4)/R, for the specified code rates are shown in Table 1.

TABLE I
CODEBLOCK LENGTHS FOR SUPPORTED CODE RATES

Information block	Codeblock length n			
length k	R=1/2	R=1/3	R=1/4	R=1/6
1784	3576	5364	7152	10728
3568	7144	10716	14288	21432
7136	14280	21420	28560	42840
8920	17848	26772	35696	53544
16384	32776	49164	65552	98328

[1]Teodor B. Iliev is with the Department of Communication Technique and Technologies, 8 Studentska Str., 7017 Rousse, Bulgaria, E-mail: tiliev@ecs.ru.acad.bg

[2] Dimitar I. Radev is with the Department of Communication Technique and Technologies, 8 Studentska Str., 7017 Rousse, Bulgaria, E-mail: dradev@abv.bg

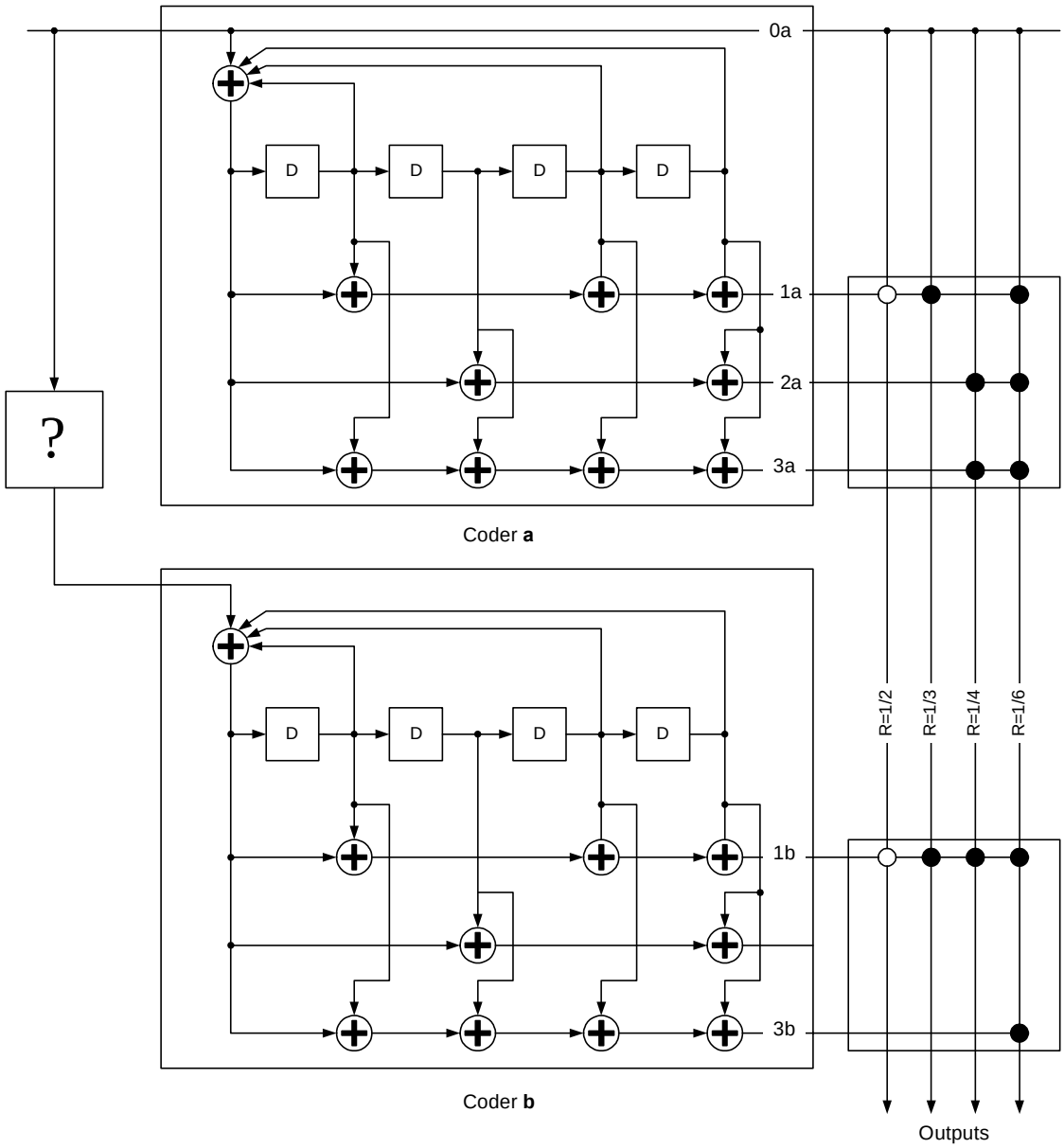

Fig. 1. Turbo Encoder Block Diagram

III. TURBO CODEBLOCK SPECIFICATION

Both component encoders in figure 1 are initialized with 0s in all registers, and both are run for a total of k+4 bit times, producing an output codeblock of (k+4)/r encoded symbols, where r is the nominal code rate. For the first k bit times, the input switches are in the lower position (as indicated in the figure) to receive input data. For the final 4 bit times, these switches move to the upper position to receive feedback from the shift registers. This feedback cancels the same feedback sent (unswitched) to the leftmost adder and causes all four registers to become filled with zeros after the final 4 bit times. Filling the registers with zeros is called terminating the trellis. During trellis termination the encoder continues to output nonzero encoded symbols. In particular, the 'systematic uncoded' output (line 'out 0a' in the figure) includes an extra 4 bits from the feedback line in addition to the k information bits.

In figure 1, the encoded symbols are multiplexed from top-to-bottom along the output line for the selected code rate to form the Turbo Codeblock. For the rate 1/3 code, the output sequence is (out 0a, out 1a, out 1b); for rate 1/4, the sequence is (out 0a, out 2a, out 3a, out 1b); for rate 1/6, the sequence is (out 0a, out 1a, out 2a, out 3a, out 1b, out 3b). These sequences are repeated for (k+4) bit times. For the rate 1/2 code, the output sequence is (out 0a, out 1a, out 0a, out 1b), repeated (k+4)/2 times. Note that this pattern implies that out 1b is the first to be punctured, out 1a is the second, and so forth. The turbo codeblocks constructed from these output

sequences are depicted in figure 2 for the four nominal code rates.[4]

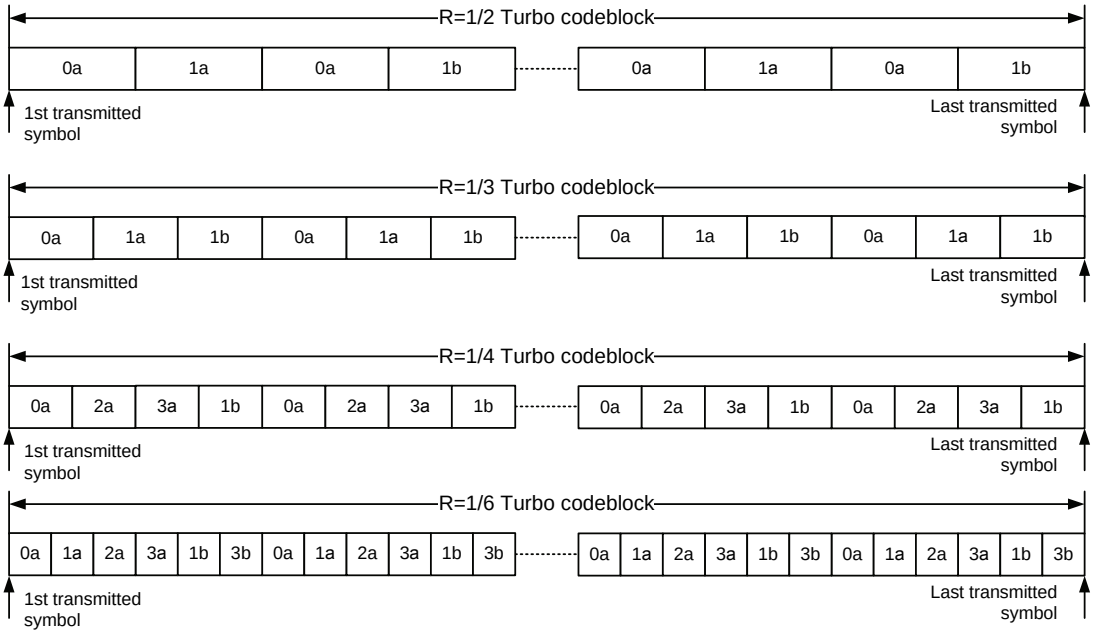

Fig. 2. Turbo Codeblocks for Different Code Rates

IV. Code performance

This section will review the performance of the CCSDS coding standard. Figures 3, and 4 show the performance of the codes in a different light. Here, the block lengths are kept at 1784 bits and 7136 bits respectively, but the rates have been allowed to vary. Common for all these graphs is that the R=1/2 code is substantially worse than the others, and that, as expected, the R=1/6 code is superior. More than 1dB asymptotic coding gain is obtained with the R=1/6 code compared with the R=1/2 code for a block length of 7136 bits, which is also consistent with the other block lengths.

Figure 5 shows the performance of the R=1/2 code of the CCSDS standard, where the block length is 16384 bits long. A varying number of iterations were used to obtain the curves. It is noticeable from this graph that a substantial coding gain is obtained when increasing the number of iterations from one to four. Furthermore, coding gains are consistently obtained when increasing the number of iterations beyond 4, albeit only minor ones after about 8 iterations. It will depend heavily on the application whether or not increasing the number of iterations beyond 4-8 would be of interest.

Figures 6, 7 and 8 show the performance of the rate R=1/2, R=1/3, and R=1/6 codes respectively, for a varying number of block lengths. The number of iterations was kept constant at 10 iterations for all these codes. As is seen from all these curves, the higher the block length, the better and the performance. A considerable coding gain is obtained by going from a block length of 1784 to 16384, particularly at low BER values. As an example, a coding gain of about 0.5dB is obtained when using a block length of 16384 instead of 1784 for the R=1/2 code, when seen at a BER of 10^{-5}. This coding gain reduces for the lower rate codes, albeit it is still around 0.35dB for the 1/6 rate code.

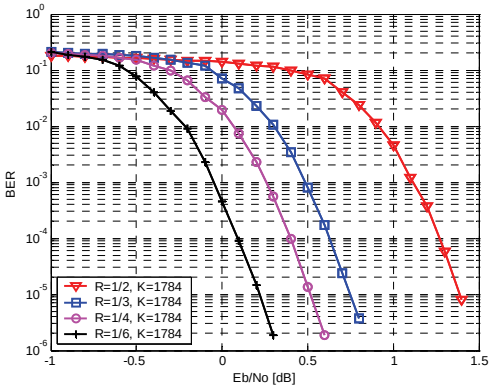

Fig. 3. Performance of different rate CCSDS codes of information block lengths 1784 bits

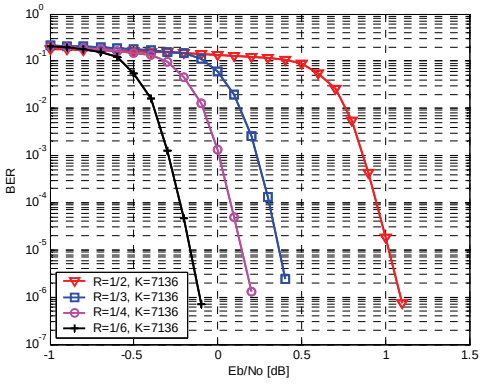

Fig. 4. Performance of different rate CCSDS codes of information block lengths 7136 bits

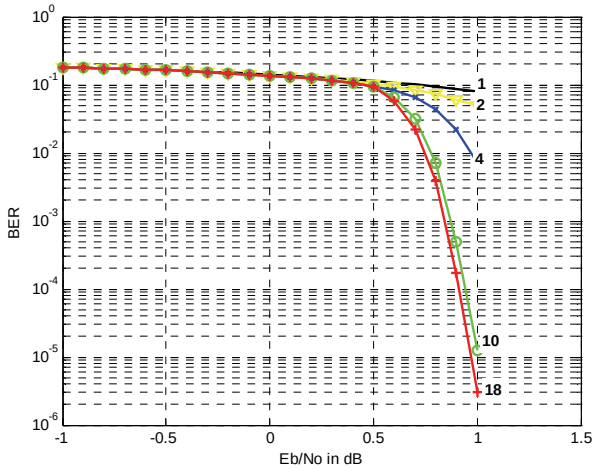

Fig. 5. Performance of CCSDS R=1/2 Turbo code of block length 16284 bits, with varying number of iterations in the decoder

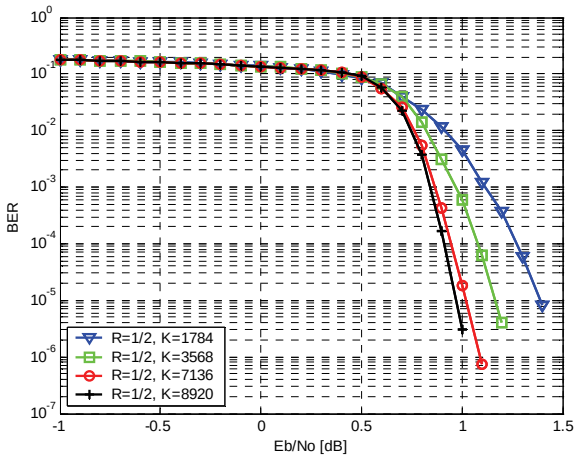

Fig. 6. Performance of CCSDS rate R=1/2 for different block lengths

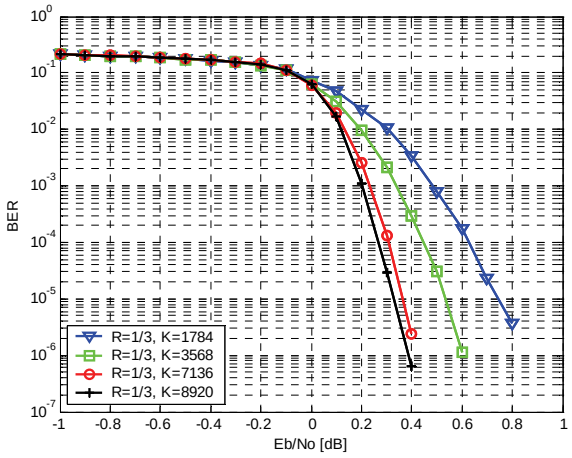

Fig. 7. Performance of CCSDS rate R=1/3 for different block lengths

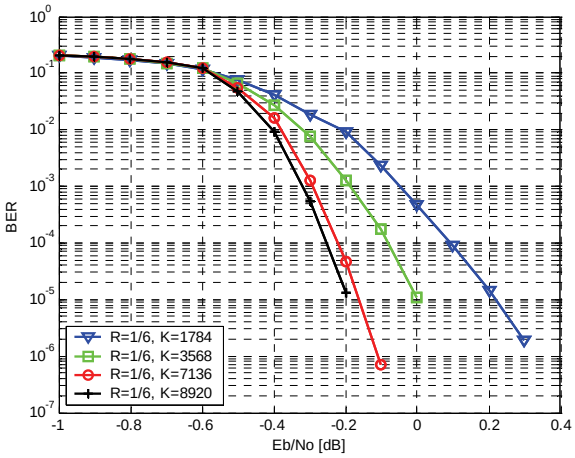

Fig. 8. Performance of CCSDS rate R=1/6 for different block lengths

V. Conclusion

In this paper we have reviewed the performance of the CCSDS turbo coding standard. The codes have been presented in general terms, and it has been argued that this standard may easily be used and adapted by other applications, where UMTS and DVB were explicitly mentioned. Some notes about implementation aspects were given.

A remaining challenge with turbo codes, is to find a suitable form of implementation that will allow for these various standards to the one turbo codec and to be of the sufficiently high speed which is required by modern data communications. Big steps have been made already, e.g. [5], and more will arrive as more applications are found for these codes.

Acknowledgement

The authors are grateful to Georgi Petkov for his helpful comments.

References

[1] C. Berrou, A. Glavieux, P. Thitmajshima, “Near Shannon limit error – correction coding and decoding: turbo–codes”, in Proc. IEEE International Conf. on Commun. (ICC), Geneva, Switzerland, pp.1064–1071, 1993.

[2] C. Schlegel, L. Perez, Trellis and turbo coding, IEEE Press, 2004.

[3] Consultative Committee for Space Data Systems, Recommendation for space data system standards, TM synchronization and channel coding, CCSDS 131.0-B-1,Blue Book, 2003

[4] M. R. Soleymani, Y. Gao, and U. Uilaipornsawai, Turbo Coding for Satellite and Wireless Communications, Kluwer Academic Publishers, Boston, 2002.

[5] F. Chiaraluce and R. Garello, “Highly efficient channel codes for high data rate missions,” presented at CCSDS Sub-Panel 1B Spring 2001 Meeting, Pasedena, CA, May 2001.