



# Deep Space Network Turbo Decoder Infusion

## Enhanced Performance and Lower Decoder Complexity

*Fabrizio Pollara and Kenneth Andrews*

Communications Systems and Research Section

### Introduction

Since their discovery in 1993, turbo codes have generated a great deal of interest due to their excellent performance and relatively low decoding complexity. In 1999 turbo codes have been included in the recommendations of the Consultative Committee for Space Data Systems (CCSDS) for telemetry channel coding, and turbo encoders are being included in the flight hardware and software of missions in development. This article describes the effort to deploy turbo decoders in the Deep Space Network (DSN) to service missions launching in 2003 and later, and the implications of these new capabilities for the design of future missions.

Turbo codes come very close to the attainment of the "Holy Grail" of information theory, sought for half a century: the ultimate capacity of a communication channel as quantified by Claude Shannon in 1948. History shows that there is typically a substantial time gap between the formulation of a new error-correcting code and the implementation of decoders suitable for operational communication systems. The familiar convolutional codes were invented in the fifties and their decoders became widely available in the seventies, and were used for the Voyager mission. A similar timeline holds for Reed-Solomon codes invented in 1960 and used for Voyager's outer planets flybys. The gap became narrower for turbo codes invented in 1993, since our turbo decoder prototype was first tested in 2000. Commercial applications of turbo codes are also widespread, including their planned deployment in the upcoming third generation cellular phone system (Universal Mobile Telecommunications System).

### Turbo Decoder Prototype

After a first attempt at designing a field programmable gate array (FPGA)-based turbo decoder, to be also used for a modified coding system emulating a turbo code on the Cassini spacecraft, it was decided to reduce the risk

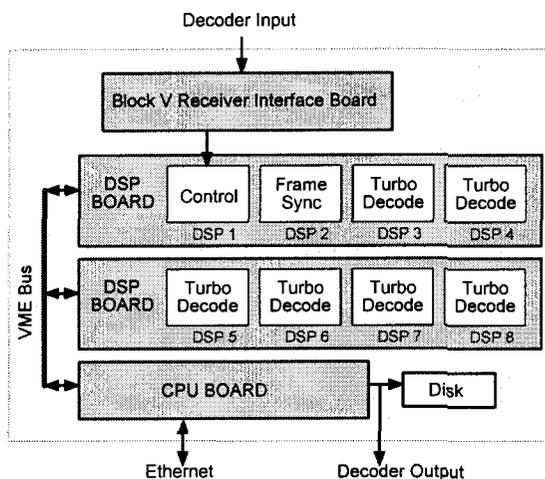
and cost of the first implementation and resort to digital signal processor (DSP) chips—the fastest on the market. Each DSP contains eight internal arithmetic units, so it can simultaneously perform as many as six adds and two multiplies. Collectively, they deliver an impressive 12,800 millions of instructions per second (MIPS).

The use of DSPs avoided the design of complicated and expensive hardware, as was the case with the previous generation of maximum-likelihood convolutional (MCD III) decoders. The first goal was set at decoding the CCSDS turbo codes at 50 kilobits per second (kbps) per DSP chip. It took a lot of ingenuity and perseverance to attain this first goal, due to the intricate structure of the DSP. The DSP provides a certain degree of parallelism, but only for a selected subset of instructions. The fundamental routines for the computation of the forward and reverse passes on the trellis and the evaluation of the extrinsic reliabilities were all carefully hand optimized in assembly language, with painstaking attention to maintaining a high level of occupancy of each arithmetic unit, minimizing idle cycles, and carefully allocating memory. The first encouraging result was to realize a Viterbi decoder (which has a similar but simpler structure) with highly efficient use of the DSP. A lot more work led to the success in decoding a short turbo block. The next hurdle was to accommodate large block sizes without the need for large transfers of data between the DSP and external memory, since the memory available within the DSP was not sufficient to hold a whole decoding block. The concept of a "sliding window" or more precisely of sub-blocks with some overlap "glue" proved to be

essential for obtaining good decoding speed. *The 50 kbps speed barrier was finally broken!* (And it has been more than doubled in the current decoder design).

The decoder prototype (see Figure 1) was implemented on two Versa Module Eurocard (VME) cards with four, 200-MHz DSP chips on each card. Six DSPs were used for the decoder itself, one for the synchronizer, and the last for peripheral services and for control (see Figure 1). The final decoder will use a single VME card with eight, 300-MHz DSPs achieving approximately 700 kbps (116 kbps per DSP) thanks to a modified algorithm using "stopping rules" as described later in this article.

A more detailed description of the turbo decoder can be found in [1] and [2].



**Figure 1. Turbo Decoder Structure**

### Technology Infusion and Interfaces to the DSN

An essential ingredient for the success of this task was the early involvement of designers familiar with DSN implementation issues. Throughout the development of the prototype, and during the current implementation of the final units for the DSN, there has been a close and continuous interaction between researchers, familiar with the turbo decoder principles, and implementers, familiar with interfaces and other DSN requirements. In this respect, the infusion of this technology has been exemplar. This interaction was also successful because of the ability of the different experts to speak each other's

"language" and in particular due to Jeff Berner's breadth of knowledge of DSN systems.

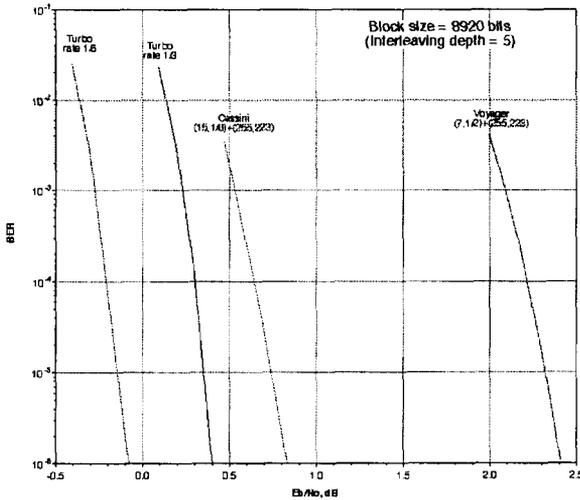
Besides the essential function of decoding the turbo encoded data blocks and outputting the decoded bits in the CCSDS Standard Formatted Data Units (SFDU), the turbo decoder encompasses several peripheral but important functions: time-tagging, frame synchronization, pseudo de-randomization, and cyclic redundancy code (CRC) checking.

Time-tagging is accomplished by counting the cycles of the 10 megahertz (MHz) reference signal and appending an accurate time tag to each symbol. Next, frame synchronization is performed. The frame synchronizer searches for the frame-synchronization marker that was appended to the coded block. The frame synchronizer checks for both normal and inverted polarity in the marker; if it detects the inverted polarity, the encoded block is marked for inversion before being sent to the decoding task. The synchronizer can buffer a minimum of four frames. This allows the system to acquire synchronization and then apply it backwards to the previous frames, reducing loss of data during the lockup period. Also, since the search for the marker is done in the symbol domain, the synchronizer operates at low signal-to-noise (SNR) and it is necessary to utilize multiple frames to boost the SNR for sufficient synchronization reliability. A single DSP can do the job of matching four, 192-bit sync patterns, giving optimum detection within the permitted latency. The synchronized block is passed to the decoder element, along with a flag indicating whether or not the polarity of the block is inverted. Also, if pseudo randomization was applied to the code block, it is removed.

### Performance and Decoding Speed

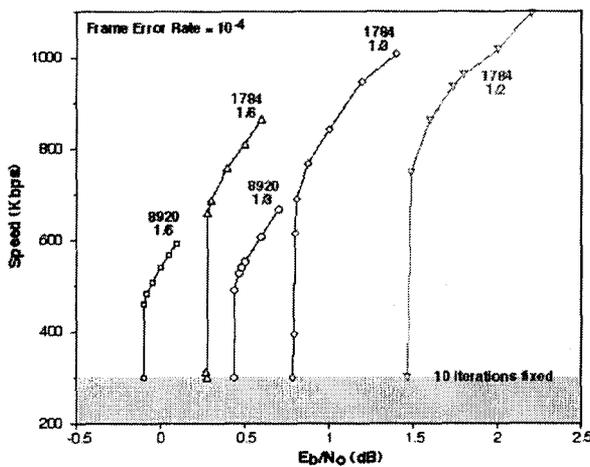
Turbo codes provide up to 0.8 dB improvement in  $E_b/N_o$  over the current best codes used by deep space missions and a substantial reduction in decoding complexity compared to that needed for the Galileo and Cassini codes. It is expected that the lower complexity and the modular design with multiple DSPs will greatly simplify the maintenance of the decoder units, in sharp contrast with the difficulties encountered in the troubleshooting of the ailing MCD III decoders. The performance of turbo codes compares favorably to that of current codes with equal

codeblock size (see Figure 2). Note:  $E_b =$  energy per bit;  $N_o =$  single-sided noise power spectral density



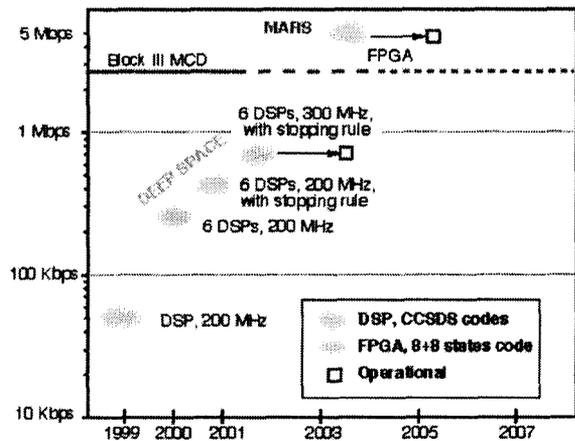
**Figure 2. Performance of two turbo codes, compared with current codes of similar rate and block length**

The initial decoder prototype used a fixed number of iterations. However, we have shown that simple schemes, called “stopping rules”, can be used to decide to stop iterating based on some simple internal reliability measures. This decreases the average number of required iterations for a given data quality requirement and provides a useful tradeoff between power efficiency and decoding speed (see Figure 3).



**Figure 3. Tradeoff between power efficiency and decoder speed permitted by a stopping rule**

The evolution of the speed performance of the turbo decoder is shown in Figure 4 and is compared to the speed of the current MCD III decoders for long-constraint-length convolutional codes used for Galileo, Cassini, Pathfinder and other recent missions. Figure 4 shows the planned deployment of the turbo decoder for missions launching in 2003 and later. More recently, Mars missions such as the Mars Reconnaissance Orbiter (MRO) have an interest in higher data rates up to 4 or 5 Mbps. The trend seems to point to even higher data rate requirements in the near future. We are currently in the midst of negotiations aimed at realizing a next generation turbo decoder based on FPGA technology. This has been recommended as the best path for the evolution of DSN decoding services. This is after careful consideration of other alternatives, such as further increasing the number of DSP cards or just relying on expected technological advances in the speed to be provided by future DSP chips. If the FPGA turbo decoder project can be funded soon, we will be able to service MRO in 2006.



**Figure 4. Expected advances in turbo decoder speed**

For even higher decoding speeds in the tens of Mbps, we will probably have to resort to a different class of codes called low-density parity-check codes (LDPC), which are close relatives of turbo codes and as good as turbo codes in terms of power efficiency. LDPC codes have been predicted to be practically decodable up to 1 Gbps in recent studies.

When data rates are so high it is also important to consider the spectral efficiency of

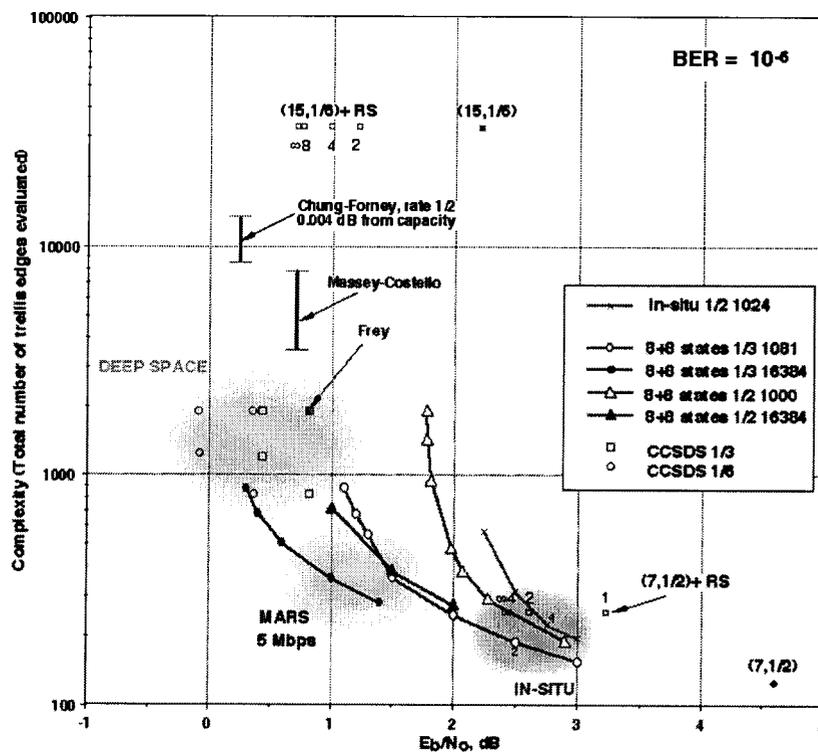
the communications system (modulation and coding, combined). We are currently comparing the most promising bandwidth efficient coded modulation schemes: coded Feher's quadrature-phase-shift keying (FQPSK) with an iterative decoding similar to that used in serial turbo codes, or "pragmatic turbo codes" developed in FY01, using punctured turbo codes. The former scheme can deliver high spectral efficiency thanks to a special form of filtered QPSK signaling, but would require major modifications in the currently-planned DSN decoding system. The second scheme has the advantage of being decodable with the same turbo decoder described in this article, but has limitations due to its nonconstant-envelope signal set and related effects due to the power amplifier nonlinearity.

Throughout the development of new turbo coded systems for the DSN we have considered the best tradeoffs in terms of *power efficiency* of the code (required  $E_b/N_0$  for a given quality of the decoded bits or frames) and *decoder complexity* that translates into sustainable data rate. The tradeoffs in terms of these two main metrics are illustrated in **Figure 5** below for three scenarios: deep space mission (up to 700 kbps using the CCSDS turbo codes), Mars missions requiring high data rates (up to 5 Mbps), and in-situ links. Deep space links can afford larger codeblock sizes and low code rates. Mars missions need to use higher code rates for spectral efficiency and may sacrifice a little performance for lower complexity<sup>1</sup>. In-situ links require even lower complexity (the decoder is in the orbiter) and short codeblocks for bursts

<sup>1</sup> Current negotiations for a future turbo decoder for Mars missions up to 5 Mbps seem to have reached a consensus to implement a faster decoder for the same CCSDS codes originally developed for deep space missions and to avoid the introduction of a new code for Mars.

of communication. On board decoder development is currently not funded, but could substantially benefit from an FPGA-based decoder design since such a design can be easily ported to rad-hard FPGAs or commissioned to a very large scale integration (VLSI) foundry providing rad-hard processes.

**Figure 5** also shows the substantially higher complexity incurred by the decoders for the Cassini/Pathfinder codes shown as (15,1/6)+RS. This figure includes some relevant results published by researchers elsewhere, which we have painstakingly tracked and recorded.



**Figure 5. Comparison of power efficiency and decoder complexity for several code families**

Besides power and bandwidth efficiency future codes will have to contend with ever increasing demands in terms of decoded data quality. Our typical design target of  $10^{-6}$  bit error rate or  $10^{-4}$  frame error rate may not be sufficient for future links closely knit into a complex network. Improving the error-rate performance is difficult for turbo codes due to their characteristic behavior called "error floor". However, we are studying methods for such improvements, that

will be based either on an additional outer code or on LDPC codes.

### **Epilog (Or a New Beginning?)**

Will the discovery of turbo codes and the implementation of their decoders put an end to the saga of error-correcting codes? At first glance, this appears to be the case since their performance is so close to the ultimate limits, but in fact the reverse seems to be true. The "turbo principle" is having a profound influence on related disciplines and has led to something of a renaissance in communication theory research, including equalization, synchronization, and complex estimation problems in general.

### **References**

- [1] K. Andrews, V. Stanton, S. Dolinar, V. Chen, J. Berner, F. Pollara, "Turbo Decoder Implementation for the Deep Space Network". IPN-ISP Progress Report 42-148. Feb. 15, 2002
- [2] J. Berner, K. Andrews, "Deep Space Network Turbo Decoder Implementation", Aerospace Conference, Big Sky, Montana, June 01