New Schedules for Information Processing in Turbo Decoding

Peyman Meshkat
Electrical Engineering Department
Univ. of California, Los Angeles
Los Angeles, CA 90095-1594
Email peyman@icsl.ucla.edu

John D. Villasenor
Dept. of Electrical Engineering
Univ. of California, Los Angeles
Los Angeles, CA 90095-1594
Email villa@icsl.ucla.edu

Abstract — **The framework of Bayesian networks has been used to introduce new ways for decoding parallel concatenated convolutional (turbo) codes. Simulation results show that a noisy received block which does not converge using traditional turbo decoding can converge to the correct value with one or more of the methods introduced here.**

I. Introduction

The close connections between turbo decoding algorithm and Pearl's belief propagation algorithm [1] have been described by MacKay, McEliece and Cheng in [2]. Pearl's algorithm calculates the exact *a-posteriori* probabilities (or "belief" in the AI literature) of the variables for a loop-free Bayesian network. Although the Bayesian network corresponding to turbo codes is not loop-free, MacKay *et al.* have shown in [2] that turbo decoding can nonetheless be viewed as an implementation of Pearl's belief propagation algorithm in which the presence of loops is ignored.The order in which nodes are processed in a network is referred to as the "node activation schedule" or, equivalently, the "timing schedule." In a loop-free network all different timing schedules result in a unique and exact solution. On the graph representing turbo decoding, however, an exact solution is no longer guaranteed, and we show here that the outcome depends on the "node activation schedule". We show that alternative timing schedules can in fact allow performance approximately equal to that achieved by traditional turbo decoding decoding based on the forward/backward algorithm. While the activation schedules we present yield equivalent performance in the average sense, their behavior on individual blocks can be quite different.

II. Generalized Turbo Decoding

Given the Bayesian network corresponding to a turbo code, there are clearly many different node activation schedules that can be used. Below we introduce four such schedules and explore their performance via simulation.

Method 1: This method corresponds to traditional turbo decoding. The nodes corresponding to the trellis edges of each code are activated first in forward direction and then in the backward direction. This is equivalent to the forward/backward algorithm.

Method 2: One can also activate the nodes corresponding to the two constituent decoders in alternation, while preserving the forward/backward structure of the activation. The key difference between this method and traditional turbo decoding (method 1) is that here the two constituent coders are considered together.

Method 3: This method is similar to method 1 but with the difference that immediately after activation of each node in the first constituent decoder, the node in the second constituent decoder corresponding to the same input bit is activated.

Method 4: This method combines the approach of activating nodes in both decoders in alternation (from method 2) with the explicit use of the interleaver structure (from method 3). More details about the above activation methods can be found in [3]

	Node activation schedule		
	Method 2	Method 3	Method 4
Number of correctly decoded blocks (out of total of 269)	15	10	17
convergence percentage	5.6	3.7	6.3

Table 1: **Convergence performance for methods 2-4, conditioned on blocks which method 1 (forward backward algorithm) fails to decode.**

III. Simulation Results and Conclusions

We have implemented the activation schedules above using a rate 1/3 encoder that included a systematic bit and two identical recursive 8-state convolutional encoders with generator matrix $G(D) = \frac{1+D+D^2+D^3}{1+D^2+D^3}$ and an interleaver with length 1024 with a signal to noise ratio of $0.3dB$. Table 1 shows the number of blocks which were correctly decoded using methods 2-4, given that they did not converge to the correct solution in a turbo decoder based on the forward backward algorithm (method 1). These data are based on a total of 269 blocks which failed to correctly converge for method 1. They show that while most blocks that are non-convergent for method 1 remain non-convergent for methods 2-4, some of the blocks that are non-convergent for method 1 can be correctly decoded when other node activation schedules are used.

We have shown that the traditional turbo decoding algorithm using the forward/backward algorithm is not the only iterative way of decoding parallel concatenated convolutional codes. The framework of Bayesian networks, with the aid of of Pearl's belief propagation algorithm, allows other possible orderings for activation of the nodes.

References

[1] J. Pearl, "Probabilistic Reasoning in Intelligent Systems", San Mateo, CA: Morgan Kaufmann, 1988.

[2] D.J.C. MacKay, R.J. McEliece and J.F. Cheng, "Turbo Decoding as an Instance of Pearl's Belief Propagation Algorithm", IEEE *Journal on Selected Areas in Communication*, Jan. 1998.

[3] P. Meshkat and J.D. Villasenor, "Generalized Versions Turbo Decoding in the Framework of Bayesian Networks and Pearl's Belief Propagation Algorithm", Proceedings of ICC '98, Atlanta, GA, June 7-11, 1998.