Iterative Min-Sum Decoding of Tail-biting Codes[1]

Srinivas Aji, Gavin Horn, Robert McEliece, and Meina Xu
Department of Electrical Engineering, California Institute of Technology,
Pasadena, California 91125, USA
e-mail: (mas, gavinh, rjm, mxu)@systems.caltech.edu

Abstract — **By invoking a form of the Perron-Frobenius theorem for the "min-sum" semiring, we obtain a union bound on the performance of iterative decoding of tail-biting codes. This bound shows that for the Gaussian channel, iterative decoding will be optimum, at least for high SNRs, if and only if the minimum "pseudodistance" of the code is larger than the ordinary minimum distance.**

I. Introduction

Because of the remarkable success of the iterative turbo-decoding algorithm [4], many coding researchers have been focussing on the study of other, more easily analyzed, suboptimal iterative decoding algorithms. Perhaps the simplest such algorithm is the iterative decoding of tail-biting codes. In this paper we announce the result that iterative min-sum decoding of a tail-biting code will be effective if and only if the minimum "pseudoweight" of the code is strictly greater than its ordinary minimum weight. We are, however, only able to define the pseudoweight for the AWGN channel.

II. Perron-Frobenius for the min-sum semiring

In this section we will state without proof a "Perron-Frobenius" theorem for the min-sum semiring. It is not the most general such theorem, but it will suffice for our purposes. (Cf. the usual "sum-product" P.-F. theorem, e.g. [7, Theorem 4.5.12]).

Thus let A be an $s \times s$ irreducible matrix with entries from $R \cup \{\infty\}$, with rows and columns indexed by $\{1, 2, \ldots, s\}$, and let G be the corresponding weighted digraph. Assume that among all simple closed paths in G, there is a unique one with minimum average edge weight, and that this "critical cycle" is in fact a self-loop of weight ρ at vertex 1. (We summarize this condition by saying that A has a "simple eigenvalue.") Then for n sufficiently large, with min-sum arithmetic,

$$A^n = \rho^n E.$$

Here E is a fixed $s \times s$ "rank one" matrix, i.e., $E_{i,j} = x_i y_j$, where $\mathbf{x} = (x_1, \ldots, x_s)$ and $\mathbf{y} = (y_1, \ldots, y_s)$ are right and left "eigenvectors" for A with corresponding "eigenvalue" ρ.

III. Tail-Biting Codes and Pseudo-Codewords

In this section we give a brief introduction to tail-biting codes. For further details, we refer the reader to [6].

A tail-biting trellis is a finite, labeled, digraph in which the vertices are partitioned into n classes $\Sigma_0, \ldots \Sigma_{n-1}$, each class being indexed by an element of $Z_n = \{0, 1, \ldots, n-1\}$, the cyclic group of order n. (All index arithmetic is done modulo n.) If E is an edge, we denote the initial vertex of E by init(E) and the final vertex of E by fin(E). An edge E must have init(E) $\in \Sigma_k$ and fin(E) $\in \Sigma_{k+1}$, for some $k \in Z_n$. The label of such an edge, denoted out(E) (for "output"), belongs to a finite alphabet A_k. If P is a path, the label of P, denoted out(P), is the concatenation of the labels of the edges comprising P. We call out(P) the output of the path P.

An *L-segment tail-biting path* P is a trellis path of length Ln for which init(P) = fin(P) (which makes it tail-biting), and init(P) $\in \Sigma_0$, fin(P) $\in \Sigma_{Ln}$ (which makes it have L segments). The *code* generated by the tail-biting trellis is the set of outputs of the the one-segment tail-biting paths. A *pseudocodeword* is the output of any tail-biting path, whether the number of segments is 1 or more than 1.

IV. Iterative Decoding of Tail-biting Codes

The iterative min-sum decoding algorithm for tail-biting codes is discussed explicitly in [3, 6, 8]. Our view is that it is an application of the Generalized Distributive Law [2], as applied to a junction graph with just one cycle [1].

In any case, if $\mathbf{y}$ is the received noisy codeword, after a finite number of iterations, the decoder will "lock on" to the pseudocodeword nearest to $\mathbf{y}$, which is called the *dominant pseudocodeword* in [6]. This follows from the min-sum Perron-Frobenius theorem (alternatively see [8] or [6]). Here the appropriate matrix A has entry $a_{i,j}$ given by $a_{i,j} = \min\{p(\mathbf{y}|\mathbf{x}) : \text{init}(\mathbf{x}) = i, \text{fin}(\mathbf{x}) = j\}$, A ML decoder will compute $\min_i\{a_{i,i}\}$, since that corresponds to the most likely tail-biting codeword. On the other hand, a two-way iterative min-sum decoding algorithm will converge after a finite number of iterations, to the same result, *provided A has a simple eigenvalue.* In coding terms, this condition amounts to saying that there is a unique nearest pseudocodeword to $\mathbf{y}$, which is in fact a codeword. This fact allows us to bound the probability of decoder error, using the familiar union bound argument. This we do in the next section.

V. The Union Bound for Iterative Decoding on the AWGN Channel

In this section we restrict attention to binary linear (tail-biting) codes, being used with BPSK modulation on an additive white Gaussian channel. We will use the insights gained in the previous section (the decoder converges to the nearest pseudocodeword) to obtain a "union bound" on the decoder word error probability.

If $\mathbf{x} = (x_1, \ldots, x_L)$ is an L-segment $(0, 1)$ pseudocodeword, and if c_j (the jth column sum) is defined to be $c_j = \sum_{i=1}^{L} x_{i,j}$, its pseudoweight is defined to be

$$w(\mathbf{x}) = \frac{(\sum_j c_j)^2}{\sum_j c_j^2}.$$

[1] This work was partially supported by NSF grant no. NCR-9505975, AFOSR grant no. 5F49620-97-1-0313, and a grant from Qualcomm, Inc. Gavin Horn's contribution was also supported by an NSERC scholarship.

0-7803-4408-1/98/$10.00 ©1998 IEEE

Thus for example the three-segment pseudocodeword (0000) (0101) (0011) has $c_1 = 0$, $c_2 = c_3 = 1$, and $c_4 = 2$, so that its pseudoweight is $(0+1+1+2)^2/(0^2+1^2+1^2+2^2) = 8/3$. Note that the pseudoweight of an ordinary codeword is the same as its weight as usually defined.

If C denotes the set of all codewords, and if P denotes the set of all *simple* pseudocodewords (a simple pseudocodeword is one that does not pass through the same vertex twice), we can prove a union bound on the (iterative min-sum) decoder word error probability P_E^{IT} (here for completeness we have included the ordinary union bound on P_E^{ML}, the maximum-likelihood word error probability):

$$P_E^{\mathrm{ML}} \leq \sum_{\mathbf{x} \in C} Q(\sqrt{2Rw(\mathbf{x})E_b/N_0}), \tag{1}$$

$$P_E^{\mathrm{IT}} \leq \sum_{\mathbf{x} \in P} Q(\sqrt{2Rw(\mathbf{x})E_b/N_0}) \tag{2}$$

where $Q(t) = (1/\sqrt{2\pi}) \int_t^\infty e^{-s^2/2} ds$. It is in general not easy to compute the pseudoweight-enumerator for a given code. However, in the next section we will do so for the (8,4,4) Hamming code.

VI. The (8,4,4) Hamming Code

In [5, section 5.2], an optimal tail-biting trellis for the extended (8,4,4) binary Hamming code is constructed, with state-complexity profile (2,4,4,4,2,4,4,4). We have used this trellis to experiment with the iterative min-sum decoding algorithm.

In Figure 1, we have plotted the actual performance (bit error probability) of an ML decoder and an iterative min-sum decoder (five iterations) for the (8,4,4) Hamming code for values of E_b/N_0 ranging from 0 dB to 9 dB in increments of 0.5 dB. We see no measurable difference, although we know that theoretically the iterative is not as good as ML because of the presence of pseudocodewords.

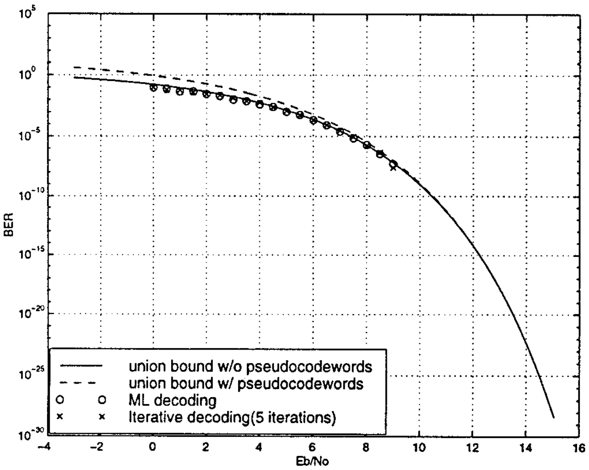

Figure 1: (8,4,4) tail biting Hamming code union bound

The pseudoweight enumerator for the (8,4,4) Hamming code, as represented by the minimal tail-biting trellis from [5] is given in the table below. In the first column is the ordinary weight enumerator, i.e., a list of the weights of the codewords (the one-segment pseudocodewords). In the second column is the pseudoweight enumerator for the 64 two-segment pseudocodewords. (There can be no simple pseudocodewords with more than two segments, because the trellis has state complexity 2 at two indices.)

	W.E	P.W.E
0	1	
1		
2		
4	15	
$4\frac{1}{2}$		32
5		
6		
$6\frac{1}{4}$		30
7		
$7\frac{2}{17}$		2
8	1	

In Figure 1, we have plotted the bounds in eqs. (1) and (2), using the data from the table, modified to give bounds on bit error probability. These bounds are asymptotically equal. This is because the leading term in both bounds is the same, because the minimum pseudoweight (in this case 4.5) is strictly larger than the minimum weight (in this case 4).

VII. Conclusions

The results in this paper strongly suggest that the excellent experimental performance reported for iterative min-sum decoding of tail-biting codes is due to the fact that the minimum pseudoweight of most tail-biting codes is strictly larger than the ordinary minimum weight. It remains a challenging problem to produce a practical algorithm for computing the minimum pseudoweight. In any case, because of the union bound, we can say with confidence that if, for a given code, the minimum pseudoweight is indeed greater than the minimum weight, then the performance of iterative min-sum decoding will be asymptotically the same as ML decoding.

References

[1] S. M. Aji, G. B. Horn, and R. J. McEliece, "On the convergence of iterative decoding on graphs with a single cycle," Proc. CISS 1998 (Princeton, N.J., March 1998).

[2] S. M. Aji and R. J. McEliece, "The generalized distributive law," preliminary versions presented at ISIT97 and ISCTA97. Current version available at www.systems.caltech.edu/EE/Faculty/rjm/.

[3] J. B. Anderson and S. M. Hladik, "Tail-biting MAP decoders," *IEEE J. Select. Areas Comm.*, vol. 16, no. 2 (Feb. 1998).

[4] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes." *Proc. IEEE Int. Comm. Conf.*, (Geneva, Switzerland, 1993), pp. 1064–1070.

[5] A. R. Calderbank, G. D. Forney, Jr., and A. Vardy, "Minimal tail-biting trellises: the Golay code and more." submitted to *IEEE Trans. Inform. Theory.*

[6] G. D. Forney, Jr., F. R. Kschischang, and B. Marcus, "Iterative decoding of tail-biting trellises," presented at 1998 Information Theory Workshop. San Diego: Feb. 9–11, 1998.

[7] D. Lind and B. Marcus, *Symbolic Dynamics and Coding.* Cambridge, England: Cambridge University Press, 1995.

[8] Y. Weiss, "Belief propagation and revision in networks with loops." M.I.T. A.I. Memo no. 1616, November 1997. (Can be retrieved by anonymous ftp from publications.ai.mit.edu.)