PROCEEDINGS

THIRTY-SEVENTH ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

Bruce Hajek
R.S. Sreenivas
Conference Co-Chairs

Conference held
September 22, September 23, and September 24, 1999
Allerton House
Monticello, Illinois

Sponsored by
The Coordinated Science Laboratory
and
The Department of Electrical and Computer Engineering
of the
UNIVERSITY OF ILLINOIS
at
Urbana-Champaign

The Serial Concatenation of Rate-1 Codes Through Uniform Random Interleavers*

Henry D. Pfister and Paul H. Siegel
Signal Transmission and Recording Group
Department of Electrical and Computer Engineering
University of California, San Diego
La Jolla, California 92093-0407
{hpfister, psiegel} @ucsd.edu

Abstract

Until the Repeat Accumulate codes of Divsalar, *et al.* [4], few people would have guessed that simple rate-1 codes could play a crucial role in the construction of "good" codes. In this paper, we will construct "good" linear block codes at any rate $r < 1$ by serially concatenating an arbitrary outer code of rate r with a large number of rate-1 inner codes through uniform random interleavers. We derive the average output weight enumerator for this ensemble in the limit as the number of inner codes goes to infinity. Using a probabilistic upper bound on the minimum distance, we prove that long codes from this ensemble will achieve the Gilbert-Varshamov bound with high probability. Finally, by numerically evaluating the probabilistic upper bound, we observe that it is typically achieved with a small number of inner codes.

1 Introduction

The introduction of turbo codes by Berrou, Glavieux, and Thitimajshima [3] is remarkable because it combined simple components together to set a new standard for error-correcting codes. Since then, iterative "turbo" decoding has made it practical to consider a whole new world of concatenated codes while the use of "random" interleavers and recursive convolutional encoders has given us a starting point for choosing new code structures. Many of these concatenated code structures fit into a class that Divsalar, Jin, and McEliece call "turbo-like" codes [4]. This class includes their Repeat Accumulate (RA) codes which consist only of a repetition code, an interleaver, and an accumulator. Still they prove that, for sufficiently low rates and any fixed E_b/N_0 greater than a threshold, these codes have vanishing word error probability as the block length goes to infinity. This shows that powerful error-correcting codes may be constructed from extremely simple components.

In this paper we consider the serial concatenation of an arbitrary outer code of rate $r < 1$ with m identical rate-1 inner codes where, following the convention of turbo coding literature, we use the term serial concatenation to mean serial concatenation through a "random" interleaver. Any real system must, of course, choose a particular interleaver. Our analysis, however, will make use of the *uniform random interleaver* (URI) [2] which is equivalent to averaging over all possible interleavers. We analyze this system using a probabilistic bound on the minimum distance and show that, in the limit as the number of inner codes m goes to infinity, the minimum distance is bounded by an expression that resembles the Gilbert Bound (GB) [5].

*This work was supported in part by the National Science Foundation (NSF) under grant NCR-9612802 and by the National Storage Industry Consortium (NSIC).

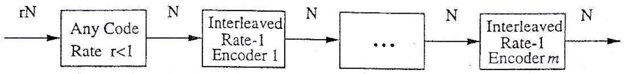

Figure 1: Our system consists of any rate $r < 1$ code followed by m rate-1 codes.

Our work is largely motivated by [4] and by the results of Öberg and Siegel [10]. Both papers consider the effect of a simple rate-1 "Accumulate" code in a serially concatenated system. In [4] a coding theorem is proved for RA codes, while in [10] the "Accumulate" code is analyzed as a precoder for the dicode magnetic recording channel. Benedetto, *et al.* also investigated the design and performance of Double Serially Concatenated Codes in [1].

If the outer code consists of multiple independent copies of a short block code and the inner code is a cascade of m interleaved "Accumulate" codes, we will refer to these codes as Generalized Repeated Accumulated (GRA^{m}) codes. McEliece has analyzed the maximum likelihood decoding performance of these codes for $m = 1$ [9], and we focus on the minimum distance of these codes for $m \geq 1$.

The outline of the paper is as follows. In Section 2 we review the *weight enumerator* (WE) of linear block codes and the union bound on the probability of error for maximum likelihood decoding. We also review the average weight enumerator for the serial concatenation of two linear block codes through a URI, and relate serial concatenation to matrix multiplication using a normalized form of each code's *input output weight enumerator* (IOWE). In Section 3 we introduce our system, shown in Figure 1, and we compute its average output WE. In Section 4 we derive a probabilistic bound on the minimum distance of any code, taken from a random ensemble, in terms of the ensemble's average WE. Applying this bound to the WE from Section 3 gives an expression very similar to the GB, and examining the bound as the block length goes to infinity produces the Gilbert-Varshamov Bound (GVB). In Section 5 we numerically evaluate our bound on minimum distance for various GRA^{m} codes and observe that 3 or 4 "Accumulate" codes seem to be sufficient to achieve the bound corresponding to asymptotically large m. Finally, in Section 6 we discuss some conclusions and directions for future work.

2 Weight Enumerators and Serial Concatenation

2.1 The Union Bound

In this section, we review the weight enumerator of a linear block code and the union bound on error probability for maximum likelihood decoding. The IOWE $A_{w,h}$ of an (n, k) block code is the number of codewords with input weight w and output weight h, and the WE A_h is the number of codewords with output weight h and any input weight. Using these definitions, the probability of word error is upper bounded by

$$P_w \leq \sum_{h=1}^{n} \sum_{w=1}^{k} A_{w,h} z^h,$$

and the probability of bit error is upper bounded by

$$P_b \leq \sum_{h=1}^{n} \sum_{w=1}^{k} \frac{w}{k} A_{w,h} z^h.$$

The term z^h represents an upper bound on the pairwise error probability, between any two codewords differing in h positions, for the channel of interest. The constant z is defined for many memoryless channels [7, Section 5.3], and for the AWGN channel it is $z = e^{-(k/n)(E_b/N_0)}$.

2.2 Serial Concatenation through a Uniform Interleaver

In this section, we review the serial concatenation of codes through a uniform random interleaver. The introduction of URI in the analysis of turbo codes by Benedetto and Montorsi [2] has made the analysis of complex concatenated coding systems relatively straightforward, and using the URI for analysis is equivalent to averaging over all possible interleavers. The important property of the URI is that the distribution of output sequences is a function only of the weight distribution of input sequences. More precisely, an input sequence of weight w produces all possible output sequences of weight w, each with equal probability.

Consider any (n, k) block code with IOWE $A_{w,h}$ preceded by a URI. We will refer to such a code as a *uniformly interleaved code* (UIC). The probability of the combined system mapping an input sequence of weight w to an output sequence of weight h is

$$Pr\,(w \to h) = \frac{A_{w,h}}{\binom{k}{w}}. \tag{1}$$

We can now consider an (n, k) block code formed by first encoding with an (n_1, k) outer code with IOWE $A^{(o)}_{w,h}$, then permuting the output bits with a URI, and finally encoding again with an (n, n_1) inner code with IOWE $A^{(i)}_{w,h}$. The average number of codewords with input weight w and output weight h is then given by

$$\begin{aligned}\overline{A}_{w,h} &= \sum_{h_1=0}^{n_1} A^{(o)}_{w,h_1}\,Pr\,(h_1 \to h) \\ &= \sum_{h_1=0}^{n_1} A^{(o)}_{w,h_1}\,\frac{A^{(i)}_{h_1,h}}{\binom{n_1}{h_1}}.\end{aligned} \tag{2}$$

The average IOWE for the serial concatenation of two codes may also be written as the matrix product of the IOWE for the outer code and a normalized version of the IOWE for the inner code. Let us define, for any code, the *input output weight transition probability* (IOWTP) $P_{w,h}$ as the probability that a uniform random input sequence of weight w is mapped to an output sequence of weight h. From (1), we can see that

$$P^{(i)}_{w,h} = \frac{A^{(i)}_{w,h}}{\binom{k}{w}}. \tag{3}$$

Substituting (3) into (2), we have

$$\overline{A}_{w,h} = \sum_{h_1=0}^{n_1} A^{(o)}_{w,h_1}\,P^{(i)}_{h_1,h} = \mathbf{A}^{(o)}\mathbf{P}^{(i)}.$$

where $\mathbf{A}^{(o)}$ is the matrix representation of the outer code IOWE and $\mathbf{P}^{(i)}$ is the matrix representation of the inner code IOWTP. By inductively applying this to multiple inner code IOWTP matrices, one can see that matrix multiplication computes the overall $\overline{A}_{w,h}$ for an arbitrary number of serial concatenations. It is also clear from (3) that IOWTP matrices are stochastic (i.e. all rows sum to 1).

2.3 A Simple Example

In this section, we will compute the IOWE and IOWTP of the rate-1 "Accumulate" code [4]. The "Accumulate" code is a block code formed by truncating the simplest recursive convolutional code possible, having generator matrix $G(D) = 1/(1 \oplus D)$, after n symbols. The

Input Sequence	000	001	010	100	011	101	110	111
Input Weight	0	1	1	1	2	2	2	3
Output Sequence	000	001	011	111	010	110	100	101
Output Weight	0	1	2	3	1	2	1	2

Table 1: Input-output sequences and weight mappings for $n = 3$ "Accumulate" code.

generator matrix for this block code is an $n \times n$ matrix with all 1's in the upper triangle and all 0's elsewhere. In the example, we will look at the case $n = 3$. The generator matrix is

$$G = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Using Table 1, we see that a uniform random input of weight 1 maps to output weights 1, 2, and 3 with equal probability, and cannot be mapped to output weight 0. So the $w = 1$ row of the IOWTP matrix is $\begin{bmatrix} 0 & 1/3 & 1/3 & 1/3 \end{bmatrix}$. Filling in the rest of the entries, we give both the IOWE $A_{w,h}$ and the associated IOWTP $P_{w,h}$ in matrix form:

$$A_{w,h} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}_{w,h}, \quad P_{w,h} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/3 & 1/3 & 1/3 \\ 0 & 2/3 & 1/3 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}_{w,h}.$$

3 Multiple Rate-1 Serial Concatenations

3.1 The Input Output Weight Enumerator

In this section, we will consider a code formed by encoding $m + 1$ times. The first (outer) encoder is for an (n, k) block code with IOWE $A^{(o)}_{w,h}$. The next m (inner) encoders are for identical rate-1 UICs of block length n with IOWE $A^{(i)}_{w,h}$. If we let $\mathbf{P}$ be the IOWTP matrix associated with $A^{(i)}_{w,h}$, then we can write the average IOWE $\overline{A}_{w,h}$ for this code as

$$\overline{A}_{w,h} = \sum_{h_1=0}^{n} A^{(o)}_{w,h_1} \left[\mathbf{P}^m\right]_{h_1 h}. \tag{4}$$

The linearity of the code guarantees that the matrix $\mathbf{P}$ will be block diagonal with at least two blocks because inputs of weight 0 will always be mapped to outputs of weight 0 and inputs of weight greater than 0 will always be mapped to outputs of weight greater than 0. So let the first block be the 1×1 submatrix associated with $w = h = 0$, and let the second block $\mathbf{Q}$ be the $n \times n$ submatrix formed by deleting the first row and column of $\mathbf{P}$. Writing $\mathbf{P}^m$as the product of block diagonal matrices, we see that

$$\mathbf{P}^m = \begin{bmatrix} 1 & 0 \\ 0 & \mathbf{Q}^m \end{bmatrix}.$$

3.2 Stationary Distributions and Markov Chains

In this section, we will discuss the stationary distributions of a Markov Chain (MC) and how they relate to the stationary weight distributions of a rate-1 UIC. This discussion is based on

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.