A Modified Concatenated Coding Scheme, with Applications to Magnetic Data Storage

John L. Fan*
Dept. of Electrical Engineering
Stanford University
P.O.Box 2586
Stanford, CA 94309-2586
jfan@isl.stanford.edu

A. R. Calderbank
Information Sciences Center
AT&T Labs - Research
Murray Hill, NJ 07974

August 21, 1997

Abstract

When block modulation codes are concatenated with an error-correction code (ECC) in the standard way, the use of long block lengths results in error-propagation. This paper analyzes the performance of modified concatenation, which involves reversing the order of modulation and ECC. This modified scheme reduces error propagation, provides greater flexibility in the choice of parameters, and facilitates soft-decision decoding, with little or no loss in transmission rate. In particular, examples are presented which show how this technique can allow fewer interleaves per sector in hard disk drives, and permit the use of sophisticated block modulation codes which are better suited to the channel.

Index terms: concatenated codes, Reed-Solomon codes, modulation codes, magnetic data storage

1 Introduction

This paper is concerned with the interaction between the modulation code and the error-correcting code (ECC). The idea of modulation is to ensure that the sequence of bits transmitted to the channel satisfies certain properties, which we will refer to as the modulation constraint. The purpose of the error-correcting code is to introduce redundancy so that it is possible to correct random errors in the codeword. Some widely used ECCs are Reed-Solomon codes, where the data is organized into symbols, and there are well-known efficient algorithms to correct symbols in error. By simply concatenating the modulation code with the ECC, the overall performance of a system can often be greatly improved.

*This work was supported in part by AT&T Bell Laboraties

Much of the previous work on combining modulation and ECC has been concerned with (d, k) runlength constraints, along with the correction of a bit shift or a single bit error, which addresses the technology of peak-detection in magnetic recording. A number of these papers ([2][9][13][18][19]) present combined modulation and ECC schemes in which the message is first modulated, and an appropriate set of modulated parity bits is appended using a systematic encoding algorithm.

In this paper, we consider the standard method of concatenation, which we will refer to as *StdConcat*, and compare it with a modification which better suits the use of block modulation codes in conjunction with ECCs. In [3] and [1], comparisons are made between the use of block codes and sliding window (including convolutional) encoders to implement modulation constraints. Block codes perform favorably when the number of bits matches the Reed-Solomon symbol size, but if the block modulation codes have long blocklengths, then error-propagation results. This motivates us to consider a scheme in which the message is first modulated, and then fed into a systematic RS encoder, which generates parity that is then modulated. This idea, which we refer to as the modified concatenated coding (*ModConcat*) scheme, has appeared before in papers by Bliss [4] and Mansuripur [16]. Recently, Immink [11] has proposed the insertion of a lossless compression step, which improves the efficiency of the scheme.

We discuss these concatenation schemes in detail, and then present an analysis of error-propagation and decoder performance using the two schemes. Finally, we will consider an application to modulation codes for magnetic recording.

2 Concatenation Schemes

The modulation code is implemented using a binary block code of rate K/N, where the modulated blocks (of length N) all satisfy the modulation constraint (such as a constraint on the run-length or on the DC content), and in addition maintain the constraints when modulation blocks are arbitrarily placed side by side. The modulation code can be described by a look-up table consisting of 2^K words of N bits.

The Reed-Solomon code uses S-bit symbols, can correct up to t symbol errors, and is encoded with a systematic encoder that takes k message symbols and appends $2t$ parity symbols to obtain a codeword of length $n = k + 2t$.

2.1 Standard concatenation

The standard method of concatenated coding firmly sandwiches the modulation code between the encoder and decoder for the Reed- Solomon code, as we see in Figure 1. Starting with M message symbols of S bits each, we encode to obtain $M + 2t$ symbols, which are modulated into $\frac{N}{K}(M + 2t)S$ bits of data sent through the channel, so the transmission rate is $\frac{K}{N}\frac{M}{M+2t}$. (Note that in the

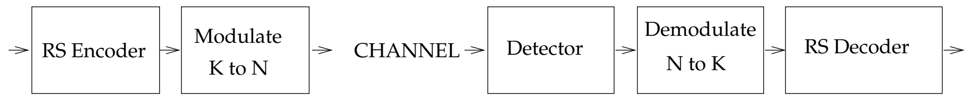

Figure 1: Standard Concatenated Coding

figures, we omit the details of the interleaving which takes place between the inner and outer codes.)

We assume that a single error event in a block of N bits results in the entire block becoming corrupted upon demodulation. Hence a short error on the channel can propagate into $\lceil K/S \rceil$ symbol errors, and a short error burst on the boundary of two blocks can even result in $\lceil 2K/S \rceil$ symbol errors. To reduce this error-propagation, it makes sense to choose K to match the symbol size S, so that we would want K to be an integral multiple of S. To avoid this error propagation due to demodulation, it would be ideal to choose $K = S$, but this severely restricts the type of modulation code we can use.

For the same modulation constraint, let us suppose we have two suitable modulation codes C_1 and C_2: The first modulation code C_1 has rate $\frac{K_1}{N_1}$, where the block length are long so that $K_1 > S$. The second modulation code C_2 has rate $\frac{K_2}{N_2}$, where $K_2 = S$ to avoid error-propagation. We would prefer to use a code like C_1, which has a longer block length and hence a higher rate ($\frac{K_1}{N_1} > \frac{K_2}{N_2}$). But since C_1 magnifies errors by a factor of K_1/S, it is often the case that a code like C_2, with $K_2 = S$, is used in practice.

2.2 Modified concatenation

In standard concatenation, the demodulation step leads to error-propagation so it is desirable to delay demodulation until after the Reed-Solomon decoder, where the data which is almost entirely error-free. By reversing the order of the modulation and error-correcting codes, it will be possible to use a modulation code with long block length.

In the top half of Figure 2, we depict the modified concatenated coding scheme, which may be attributed to Bliss [4]. The number of user symbols M can be freely chosen, but for comparison, let us choose M to be the same as in StdConcat. Starting with M user symbols of S bits each, we modulate these into $\frac{N_1}{K_1}MS$ bits using code C_1, and transmit these over the channel. Meanwhile, we pass these modulated messages into a systematic Reed-Solomon encoder that adds $2t$ parity-check symbols, for a total of $\frac{N_1}{K_1}M + 2t$ symbols in the RS codeword. The parity symbols need to be modulated before being sent to the channel, and to avoid error-propagation, we will use the code C_2, with rate K_2/N_2, giving a total of $(\frac{N_1}{K_1}M + \frac{N_2}{K_2}2t)S$ bits transmitted through the channel. This is fairly close to the $\frac{N_1}{K_1}(M + 2t)S$ channel bits using StdConcat with code C_1, and is much smaller than the $\frac{N_2}{K_2}(M + 2t)S$ bits using StdConcat with code C_2.

No error-propagation has taken place on the message bits, so that the code

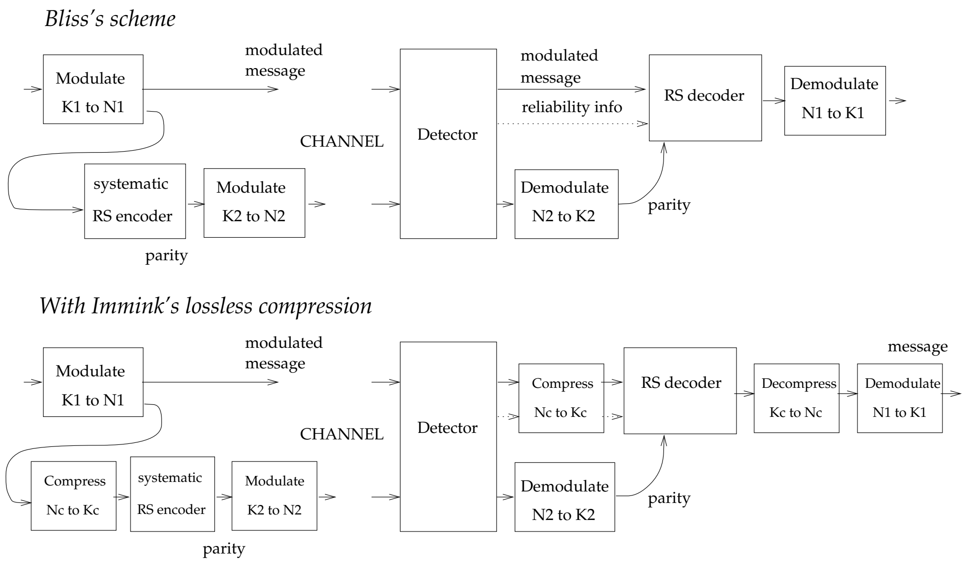

Figure 2: Modified Concatenated Coding

C_1 can have arbitrary block lengths, and in particular, K_1 and S need not match. Also, no error propagation takes place on the parity portion since we are using code C_2 which has $K_2 = S$. Moreover, all the data reaching the channel satisfies the modulation constraint, and the rate is very close to the rate of C_1. Figure 3 illustrates how ModConcat can be successfully used in cases where StdConcat would lead to decoder overflow due to error-propagation with a code like C_1.

On the other hand, this method suffers from an expansion of the input to the Reed-Solomon code. We have $k' = \frac{N_1}{K_1} M$ and $n' = \frac{N_1}{K_1} M + 2t$ for ModConcat, so there is an $\alpha = \frac{N_1}{K_1} > 1$ increase in the input to the RS decoder, and a factor of $\frac{n'}{n} = \frac{\alpha M + 2t}{M + 2t} \approx \alpha$ increase in the codeword length, as compared with StdConcat using code C_1. Some problems are that the complexity of RS decoding roughly increases linearly by α, and the length of Reed-Solomon codewords is limited ($n' < 2^S$). In addition, very long error bursts are not demodulated, so that relative to StdConcat, ModConcat expands long error bursts by α. In other words, if we ignore error-propagation effects, then a bit burst of length LS on the channel affects roughly L symbols in ModConcat, but only $\frac{K_1}{N_1} \frac{L}{S} = \frac{1}{\alpha} L$ using StdConcat.

2.3 Lossless compression

In [11], Immink considers this problem and proposes a method to reduce this expansion factor α. The idea is to introduce a block code of rate K_c/N_c which acts as a lossless compression scheme (taking N_c bits to K_c bits) and is placed before the Reed-Solomon encoder and the decoder, as shown in the bottom half

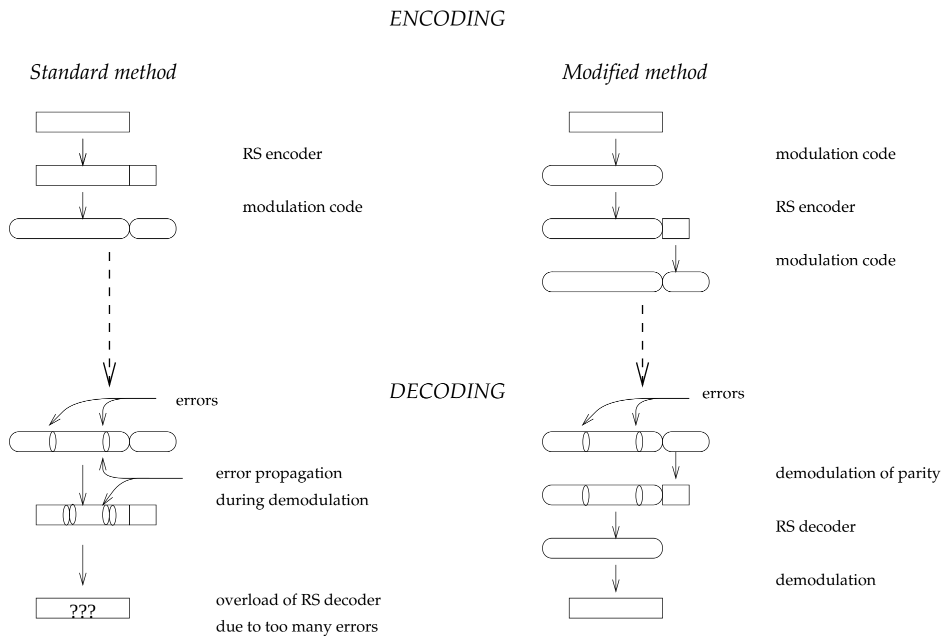

Figure 3: A Comparison of the Concatenation Schemes

of Figure 2. Since the compression step takes place before RS decoding, we need to choose $K_c = S$ to avoid error propagation. The expansion factor α is reduced to $\frac{N_1}{K_1}\frac{K_c}{N_c}$, and Immink's lossless compression alleviates the problems associated with using modified concatenation where $\frac{K_1}{N_1}$ is not close to 1.

The N_c-bit sequences which are valid inputs to the block compression scheme must include all the possible outputs of the modulation scheme. To understand better the requirements of this lossless compression block code, it will help to take a closer look at modulation using block codes. Let the set $Constr$ consist of all bi-infinite sequences of bits which satisfy the modulation constraint. The capacity of the constraint is the maximum number of information bits per channel bit that can be acheived while satisfying this constraint, and is given by $\lim_{n\to\infty} \frac{1}{n}\log_2 W_n$, where W_n is the number of constrained sequences of length n.

The idea of block modulation is to use an encoder that takes K bit to N bits, such that the output bitstream satisfies the constraints. In other words, the desired block code corresponds to a 1-1 map

$$\varphi : \{0,1\}^K \to \{0,1\}^N$$

where any combination of words in the image of φ will satisfy the modulation constraint. Let $\mathrm{Im}\,\varphi$ be the image of the mapping φ, and $(\mathrm{Im}\,\varphi)^\infty = \{(..., w_{-1}, w_0, w_1, ...) \mid w_i \in \mathrm{Im}\,\varphi \subset \{0,1\}^N\}$. Then a modulation block code satisfies:

$$\text{Modulation: } (\operatorname{Im}\varphi)^\infty \subset \mathit{Constr}$$

In other words, any bi-infinite sequence of bits consisting of blocks of length N in the image of φ will satisfy the modulation constraint. The rate K/N will be less than or equal to the capacity of the modulation constraint.

Now, for lossless compression, we need to find a 1-1 map

$$\psi : \{0,1\}^{K_c} \to \{0,1\}^{N_c}$$

such that the image contains all possible sequences that can appear in the output of the modulation code. In other words, we want a map ψ such that:

$$(\operatorname{Im}\psi)^\infty \supset (\operatorname{Im}\varphi)^\infty$$

The map ψ^{-1}, restricted to $\operatorname{Im}\psi$, gives a lossless compression map, and the map ψ gives the decompression. This mapping ψ defines a block code of rate K_c/N_c. Note that it is clear that $K_c/N_c \geq K/N$, since we cannot compress a code smaller than its original size. For a given modulation constraint, it may be appropriate to develop block compression codes in conjunction with the modulation code. For example, it might make sense for N to be an integer multiple of N_c.

But since the modulation code C_1 is assumed to be sufficiently long to make the rate close to the capacity of the constraint, it is often reasonable to assume the compression scheme to handle all possible valid constrained sequences, so we would want

$$\text{Compression: } (\operatorname{Im}\psi)^\infty \supset \mathit{Constr}$$

In this sense, there is a certain duality between the concepts of block modulation and block compression.

3 Analysis of performance

We recall that using standard correction techniques, the Reed-Solomon decoder decodes the received word to a codeword lying within Hamming distance t, if such a codeword exists. In the usual terminology (ref. [25]), when no codeword exists within distance t, this detectable malfunction is called *decoder failure*. If the decoder outputs a wrong decision because the received word lies within distance t of the wrong codeword, this event is called *decoder error*. In this section, we will simply measure performance by the probability of having more than t symbol errors in a received word. This is the sum of decoder failure rate and the decoder error rate, which we will denote by P_{decoder}.

We first consider the case where the bit errors are independent, and interleaving is sufficient to separate bursts of symbol errors. Then we take a closer look at the process of error propagation, and finally consider the case where bursts lengths exceed the interleave depth.

3.1 Decoder performance, with independent errors

First we consider a Reed-Solomon decoder where the symbol errors occur independently, which is also the case if the bursts are sufficiently dispersed by interleaving. With independent errors, the expected number of errors is binomially distributed, with the probability of having a errors given by $\binom{n}{a}_p := \binom{n}{a}p^a(1-p)^{n-a}$, where p is the symbol error rate. The probability of a decoder error or decoder failure is then given by

$$P_{\text{decoder}}(n,t,p) = \sum_{i=t+1}^{n} \binom{n}{i}_p = \sum_{i=t+1}^{n} \binom{n}{i} p^i (1-p)^{n-i} \tag{1}$$

for a t-error-correcting RS decoder. For small p and large n, this summation is dominated by the first term $\binom{n}{t+1}_p$, and the Poisson approximation to the binomial distribution, $\binom{n}{a}_p \approx e^{-np}\frac{(np)^a}{a!} \approx \frac{(np)^a}{a!}$, gives a rough estimate of performance.

We can then compare the performance of the two concatenation methods using code C_1 given that the errors are single bit errors occuring independently with probability b. For StdConcat, an error bit in any of N_1 bits in a block can corrupt the block, and the probability of symbol error is equal to the block error probability $p^{\text{std}}_{\text{symbol}} = p^{\text{std}}_{\text{block}} = N_1 b$. (We assume that interleaving has dispersed the burst of $\frac{K_1}{S}$ symbol errors that occur when a block is in error.) On the other hand, for ModConcat, each of S bits can cause a symbol to be in error, so $p^{\text{mod}}_{\text{symbol}} = Sb$. Hence for independent errors, the symbol error probabilities $p = p^{\text{std}}_{\text{symbol}}$ and $p' = p^{\text{mod}}_{\text{symbol}}$ are related by

$$p' \approx \frac{S}{N_1} p \tag{2}$$

Now if we assume the same amount of user data per codeword, then we need to increase the code length for ModConcat to $n' = \frac{N_1}{K_1}(n-2t) + 2t \approx \frac{N_1}{K_1}n$. We can then use the Poisson approximation to compare the performance of StdConcat and ModConcat as follows:

$$\begin{aligned} P^{\text{mod}}_{\text{decoder}} &\approx \binom{n'}{t+1}_{p'} \approx \frac{1}{(t+1)!}\left((\frac{N_1}{K_1}n)(\frac{S}{N_1}p)\right)^{t+1} \\ &\approx (\frac{S}{K_1})^{(t+1)} \binom{n}{t+1}_p \approx (\frac{S}{K_1})^{(t+1)} P^{\text{std}}_{\text{decoder}} \end{aligned} \tag{3}$$

This rough calculation indicates that the amount which ModConcat can be expected to perform better than StdConcat depends on the ratio S/K_1 and the strength of the RS decoder.

It is interesting to note that if we consider ModConcat with a lossless compression scheme, the result does not change. For a lossless compression block code of rate K_c/N_c and $K_c = S$, we get $p^{\text{mod}}_{\text{symbol}} = \frac{N_c}{K_c} S b = N_c b$. Also, $n' \approx \alpha n = \frac{K_c}{N_c}\frac{N_1}{K_1} n$, so we end up with the same approximate relationship:

$$\begin{aligned} P^{\text{mod}}_{\text{decoder}} &\approx \frac{1}{(t+1)!}\left((\frac{K_c}{N_c}\frac{N_1}{K_1} n)(\frac{N_c}{N_1} p)\right)^{t+1} \\ &\approx (\frac{S}{K_1})^{(t+1)} P^{\text{std}}_{\text{decoder}} \end{aligned}$$

3.2 Error propagation

For standard concatenation, let us examine the number of block errors caused by demodulating a burst of error bits with a block code of rate K/N. A single bit in error affects one block. Two adjacent bits in error have a $\frac{N-1}{N}$ chance of causing one block error and a $\frac{1}{N}$ chance of overlapping two blocks. overlapping two blocks. Given non-negative integers a and b, define $m(a,b)$ by $0 \leq m(a,b) \leq b-1$ and $m(a,b) \equiv a \pmod b$. Then in general, a burst of length L bits has a probability $\frac{m(L-1,N)}{N}$ of causing $\lfloor \frac{L}{N} \rfloor + 1$ block errors and probability $1 - \frac{m(L-1,N)}{N}$ of causing $\lfloor \frac{L}{N} \rfloor + 2$ block errors. (In the middle of a burst there may be bits not in error, but we assume that the burst begins and ends with error bits, and there are enough error bits in between to cause a contiguous burst of block errors.)

Let b_L denote the probability of observing a burst error of length L, which we find by taking the number of bursts of length L and dividing by the total number of bits. The bit error distribution is then described by $\{b_i\}$ and the average bit error rate is $\overline{b} = \sum i b_i$. It is reasonable to assume that the locations of the burst errors are uniformly distributed in the block, and that for a block of N channel bits long, there are N chances that it contains the start of a burst error. As for the number of blocks affected, we argue that a burst of length L has a probability $\frac{m(L-1,N)}{N}$ of causing $\lfloor \frac{L}{N} \rfloor + 1$ block errors and probability $1 - \frac{m(L-1,N)}{N}$ of causing $\lfloor \frac{L}{N} \rfloor + 2$ block errors, and that the effect of bursts of different lengths is additive. (Note that by our convention, if two bursts overlap, then they are counted as a single burst.)

Then we have the following expression for $p_{\text{block}}(m)$, the probability that a particular block is the start of a burst of m consecutive blocks in error:

$$\begin{aligned} p^{\text{std}}_{\text{block}}(1) &= N(b_1 + b_2 \frac{N-1}{N} + b_3 \frac{N-2}{N} + \cdots + b_N \frac{1}{N}) \\ p^{\text{std}}_{\text{block}}(m) &= \sum_{i=2}^{N} b_{(m-2)N+i}(i-1) + \sum_{i=1}^{N} b_{(m-1)N+i}(N-i+1) \end{aligned}$$

If we use a generating function to express the error distribution $b(z) = \sum_{i=1} b_i z^i$, then we have the compact expression:

$$p^{\text{std}}_{\text{block}}(m) = (\Delta_N(z) b(z))_{(1+(m-1)N)} \quad (4)$$

where $\Delta_N(z)$ is a triangular function

$$\Delta_N(z) = z^{-(N-1)} + 2z^{-(N-2)} + \cdots + (N-1)z^{-1} + N + (N-1)z + \cdots + z^{(N-1)}$$

and the subscript indicates we are taking the coefficient of $z^{1+(m-1)N}$ in the product $\Delta_N(z)b(z)$).

3.2.1 Standard Concatenation

For StdConcat with C_1, we have

$$p^{\text{std}}_{\text{block}}(m) = (\Delta_{N_1}(z)b(z))_{(1+(m-1)N_1)}$$

and since a block error causes $\frac{K_1}{S}$ symbol errors (where we will assume for simplicity that K_1 is a multiple of S), the probability that a symbol is the start of a burst of m symbol errors is

$$\begin{aligned} p^{\text{std}}_{\text{symbol}}(m) &= p^{\text{std}}_{\text{block}}\left(\frac{m}{(K_1/S)}\right) \text{ if } m \text{ is a multiple of } K_1/S \qquad (5) \\ p^{\text{std}}_{\text{symbol}}(m) &= 0 \text{ otherwise} \end{aligned}$$

In terms of generating functions, we have

$$p^{\text{std}}_{\text{symbol}}(z) = p^{\text{std}}_{\text{block}}(z^{K_1/S}).$$

3.2.2 Modified Concatenation

For simplicity, we will only look at error propagation in the message, since the fraction of parity bits is small, and there is limited error propogation because of the code C_2 (where $K_2 = S$) that modulates parity bits. If we assume that a lossless compression code of rate K_c/N_c is used (where $K_c = S$), then we obtain the following symbol error distribution in terms of $b(z)$:

$$\begin{aligned} p^{\text{mod}}_{\text{symbol}}(m) &= (\Delta_{N_c}(z)b(z))_{(1+(m-1)N_c)} \qquad (6) \\ &= \sum_{i=2}^{N_c} b_{(m-2)N_c+i}(i-1) + \sum_{i=1}^{N_c} b_{(m-1)N_c+i}(N_c - i + 1) \end{aligned}$$

(In the case that there is no lossless compression step, simply let $N_c = S$.)

3.3 Decoder performance, with burst errors

The Reed-Solomon codewords are usually interleaved to reduce the possibility that a single burst error leads to multiple symbol errors in the same codeword. Let us assume that the interleave depth is ID. Then a burst of L symbols has a probability of $1 - \frac{m(L,ID)}{ID}$ causing $\lfloor \frac{L}{ID} \rfloor$ error(s) in an interleave and probability of $\frac{m(L,ID)}{ID}$ of causing $\lceil \frac{L}{ID} \rceil$ error(s) in an interleave. Hence, if $p(m)$

is the probability, before interleaving, of a symbol burst of length m, then we can evaluate the probability $q(m)$ of a burst of length m after interleaving by

$$q(m) = (\Delta_{ID}(z)p(z))_{mID} \tag{7}$$

where we use the generating function $p(z) = \sum_{m=1} p(m)z^m$ and the same triangular function as before. Note that the average symbol error rate stays the same ($\sum mq(m) = \sum mp(m)$).

In particular, the probabilities of single errors (q_1) and double errors (q_2) after interleaving are

$$\begin{aligned} q_1 &= p(1) + 2p(2) + \cdots + ID \cdot p(ID) + \cdots + p(2ID-1) \qquad (8) \\ q_2 &= p(ID+1) + 2p(ID+2) + \cdots + ID \cdot p(2ID) + \cdots + p(3ID-1) \end{aligned}$$

In most situations, the probability of a triple error post-interleaving is negligible, so the performance of the RS decoder is determined by single and double errors.

The probability that there are more than t symbol errors in an interleave is given in terms of q_1 and q_2 by the following formula (ref. [24]):

$$P_{\text{decoder}}(n,t,q_1,q_2) = \sum_{j=0}^{\lfloor n/2 \rfloor} \sum_{i=t+1-2j}^{n-2j} \binom{n-j}{j} \binom{n-2j}{i} q_1^i q_2^j (1-q_1-q_2)^{n-i-2j} \tag{9}$$

Hence we can estimate the performance of ModConcat and StdConcat from the burst error distribution $\{b_i\}$. The symbol error rates can be found using (5) and (6), from which we can find the post-interleaving symbol error rates using (8), and then use (9) to calculate the decoder error-failure rates for both concatenation schemes.

3.4 Other features

It is often possible to extract some information from the detector (such as a Viterbi algorithm [8]) as to the reliability of the output bits. If we use this information to perform soft-decision or erasure decoding of the Reed-Solomon code, we can achieve improved performance.[6] With StdConcat and long block codes, however, it is usually difficult to associate the reliability information from the detector with individual symbols, since the the demodulation process works on a block by block basis. With modified concatenation, on the other hand, it is still possible to perform soft-decision decoding using reliability information directly from the detector.

Also, since the demodulation takes place after the RS decoder, then it is possible to use the demodulation step as an extra check on the validity of the output from the RS decoder. (For a detailed analysis on the probability of decoder error, we refer to [17].) Suppose that we have a misdecoding by the RS decoder. There are then at least $d = 2t + 1$ errors in this output word. Making some rough approximations, we can find the probability of detecting this miscorrection. If c is the capacity of the modulation constraint, then there are about 2^{Sc} codewords of length S, so the probability that a symbol error still satisifes the modulation constraint is approximately $\frac{2^{Sc}}{2^S} = 2^{S(c-1)}$. Then the probability that at least one error violates the constraint is $\approx 1 - (2^{S(c-1)})^d$.

If we include a lossless compression step of rate K_c/N_c (where $K_c = S$), then we see that there are 2^{K_c} outputs (of length N_c) of the block decompression step, but only about 2^{cN_c} words which satisfy the constraint. hence there are $2^{K_c} - 2^{cN_c}$ "leftover" words which cannot be demodulated, so the probability that a random error lead to a violation of the constraint is $\frac{1}{2^{K_c}}(2^{K_c} - 2^{cN_c}) = 1 - 2^{cN_c - K_c}$. Then the probability that at least one error violates the constraint is $\approx 1 - (2^{N_c c - K_c})^d$. Note that for a long block code C_1, $c \approx \frac{K_1}{N_1}$, so we get $N_c c - K_c \approx K_c(\frac{N_c}{K_c}\frac{K_1}{N_1} - 1) = S(\frac{1}{\alpha} - 1)$, and then the probability of detection becomes $\approx 1 - 2^{S(\frac{1}{\alpha}-1)d}$. Hence, as we lower the expansion factor α down towards 1, the probability of detection decreases to 0.

4 Application to magnetic data storage

An application where modified concatenation may prove useful is hard disk drives. We consider some typical parameters: The data is organized into sectors of 512 user bytes, and the Reed-Solomon code has symbol size $S = 8$, so the number of user bytes per interleave $M \approx 512/ID$ is limited by $M < 2^S - 2t$. Then we must have $ID \geq 3$. Also, ID should be large enough to disperse bursts into different interleaves. On the other hand, the performance of the ECC is determined largely by t, so that the amount of redundancy needed is roughly $2t \cdot ID$, and to minimize the redundancy, ID should be kept as small as possible.

The type of channel equalization which is employed will determine the appropriate types of modulation constraints to use. For read channels which employ peak detection, it is important to prevent transitions from happening too close together, since that can result in intersymbol interference. This is accomplished by using (d, k) RLL codes, with $d \geq 1$. Some well known examples are the $(1, 7)$ code with rate $\frac{2}{3}$, and the $(2, 7)$ code with rate $\frac{1}{2}$. In [11], Immink provides an efficient construction for modulation block codes that are hundreds of bits long, and can increase the modulation rate so that it approaches the capacity of the (d, k) modulation constraint. This provides a practical method to boost the coding rate of systems which require this sort of RLL constraint.

In recent years, there has been a shift of attention from peak detection to more sophisticated signal processing techniques such as decision-feedback equalization (DFE) and partial-response maximal likelihood (PRML) equalization techniques for magnetic data storage. The partial response techniques (which involve the Viterbi algorithm for maximum likelihood sequence detection) show substantial improvement over the previous techniques, and are used in numerous commercial hard disk drives. [5]. In particular, one common equalization is *partial response class 4* (PR4), where the channel is equalized to a $1 - D^2$ response, which can be treated as two interleaved binary dicode channels $(1-D)$. The d constraint can be 0, since partial response takes care of the ISI. In one implementation [22][27], a rate $\frac{8}{9}$ block code is used with RLL constraints $(0, G/I) = (0, 4/4)$, which includes a runlength constraint on each $1-D$ interleave as well as a global constraint, for the purpose of timing recovery.

Here we look at two examples of codes for PR4, which illustrate the flexibility and improved performance allowed by modified concatenation, even for relatively short block lengths. For this and other partial response channels, the ability to use longer block lengths enables us to construct appropriate block modulation codes, including more sophisticated constraints such as higher order spectral nulls or increased minimum distance. [23].

4.1 A rate 16/17 code for PR4

It is possible to construct modulation codes with rate 16/17 that satisfy a slightly weaker $(0, G/I)$ constraint than the rate $\frac{8}{9}(0, 4/4)$ code mentioned above. If we simply alternate uncoded bytes with codewords from the rate 8/9 code, there will be no error propagation upon demodulation. In this case, it has been shown [26] that more sophisticated code constructions can provide rate 16/17 codes with better $(0, G/I)$ constraints and no error propagation. However, when the strength of the $(0, G/I)$ constraint means that error-propagation cannot be avoided, there will be benefits to employing modified concatenation. For simplicity, we will demonstrate these benefits using a rate $\frac{16}{17}$ code where error propagation causes two bytes to be in error.

Let us suppose we have two modulation codes which satisfy a $(0, G/I)$ constraint with rate $K_1/N_1 = 16/17$ and $K_2/N_2 = 8/9$. (Note that the choice of K_1 as a multiple of S is useful for StdConcat, but irrelevant to ModConcat.) Since the rate is so close to 1, it will not be necessary to use a lossless compression step, so $K_c = N_c = S$.

Then for StdConcat with $ID = 3$, we have 170, 170 and 172 user bytes in each interleave (where we choose M even to accommodate the modulation code, where $K_1 = 2S$). Let us suppose the error-correction capability of the ECC is $t = 3$ errors per interleave, so the length of the RS code is $n = k + 2t$ for StdConcat, so the total number of bytes is $178 + 176 + 176 = 530$. Then there are a total of $530 \cdot S \cdot \frac{N_1}{K_1}$ channel bits per sector. For $ID = 4$ interleaves per sector, we have $M = 128$ user bytes per interleave, so for StdConcat, we have $n = 128 + 6 = 134$ and $ID \cdot n \cdot S \cdot \frac{N_1}{K_1}$ channel bits. Note that this requires an additional $2t$ parity symbols per sector, along with an additional Reed-Solomon decoding.

For ModConcat, we can modulate all the user bytes first, and then add parity, to get $k' = \frac{1}{S \cdot ID}\left(\left\lceil \frac{ID \cdot M \cdot S}{K_1} \right\rceil N_1\right)$ bytes per interleave, and a total of $\left\lceil \frac{ID \cdot M \cdot S}{K_1} \right\rceil N_1 + ID \cdot 2t \cdot N_2$ channel bits. For $ID = 3$, we get $k' = 181$ for 2 interleaves and 182 for one interleave, and for $ID = 4$, we get $k' = 136$.

16/17 code	RS input	RS output	Bits/Sector	Rate	% of Max.
Std $ID = 3$	$k = 170, 172$	$n = 176, 178$	4505	0.9092	100%
Mod $ID = 3$	$k' = 181, 182$	$n' = 187, 188$	4514	0.9074	99.8%
Std $ID = 4$	$k = 128$	$n = 134$	4556	0.8990	98.9%
Mod $ID = 4$	$k' = 136$	$n' = 142$	4568	0.8967	98.6%

The rate listed in the table is given by $\frac{\text{\# user bits}}{\text{\# channel bits}}$ and includes both the modulation and ECC. The maximum rate possible under these assumptions is $\frac{16}{17}\frac{512}{512+18} = 0.9092$.

Using our earlier analysis and formulas, we can compare the ModConcat and StdConcat schemes using only the distribution of the bit errors out of the detector. Based on the expected minimum distance error events on the two $1 - D$ channels which make up the PR4 channel, we can take as our bit error distribution, $b(z) = b \cdot (\frac{1}{2}z^3 + \frac{1}{4}z^5 + \frac{1}{8}z^7 + \cdots)$, where b is the probability of an error event. In Figure 4, we see that StdConcat requires 4 interleaves while ModConcat requires only 3 interleaves, and ModConcat always outperforms StdConcat.

In Figure 5, we suppose that the errors on the channel are all exactly of length L, with probability $b_L = \frac{b}{L}$, so that the overall bit error rate is fixed at $b = 10^{-5}$. The decreasing error rate as L increases is due to the smaller probability of error as L increases, and the sudden upward jumps are due to the loss in performance as it becomes possible that a single error event causes two (or more) errors in a single interleave.

4.2 A DC-free code for PR4

For partial response channels, it is known that matched-spectral null (MSN) codes can provide significant coding gain [12]. For the $1 - D$ channel, DC-free codes, which are balanced in the number of ones and zeros, are matched to the spectrum and provide approximately 3 dB coding gain. We can treat the PR4 channel as two completely independent $1 - D$ channels and modulate appropriately.

One method of satisfying these DC-free constraints efficiently is to use block modulation codes with long length. The design and use of DC-constrained block codes is discussed at length in [10]. It should be noted that our analysis of error propagation assumes a simple detector, which does not depend on the modulation code, so our analysis may not accurately describe the situation of detection using a time-varying trellis [20][21] or of post-processing schemes [14], for DC-free block codes, but the general conclusions about modified concatenation should be the same.

Let us consider a hypothetical modulation constraint consisting of 10-bit blocks which are DC-free. There are $\binom{10}{5} = 252$ DC-free words of length 10, and concatenating these words provides a data rate of $\frac{1}{10}\log_2 252 = 0.7977$ bits per symbol. However, encoding one 10-bit DC-free word at a time would

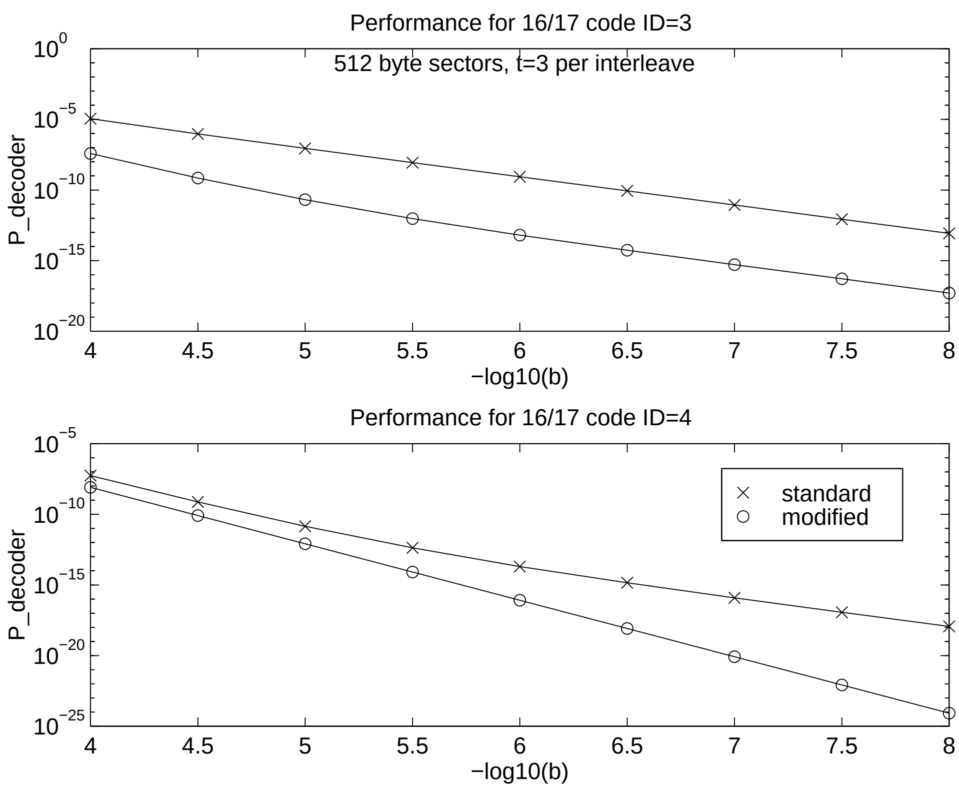

Figure 4: Performance comparison for the 16/17 code

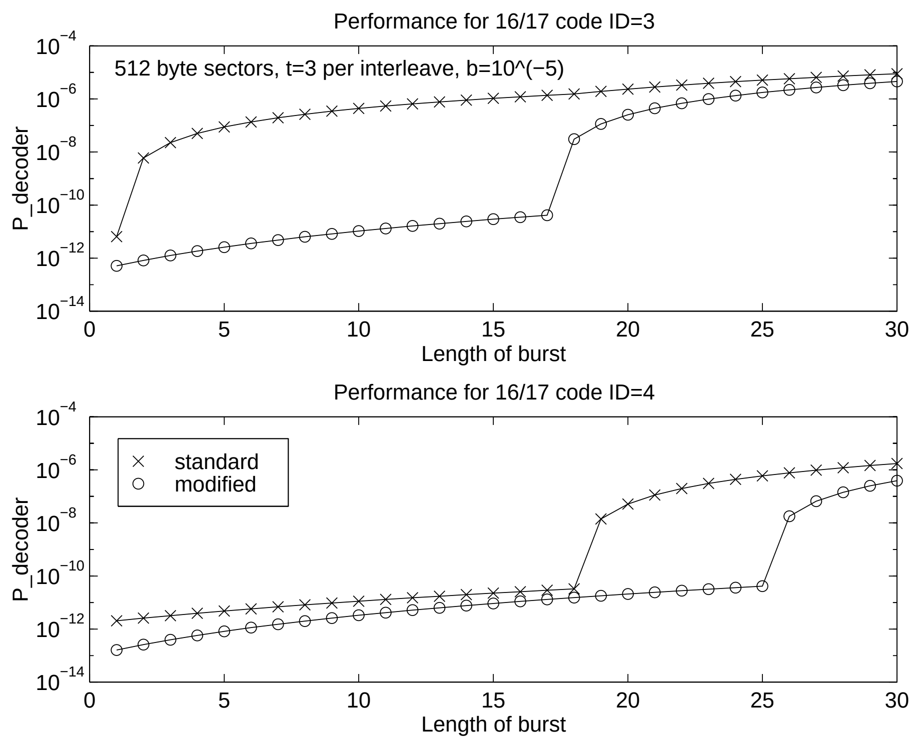

Figure 5: A comparison of the effects of long bursts for the 16/17 code

only provide a data rate of $\frac{1}{10}\lfloor\log_2 252\rfloor = 0.7$. To get closer, Gelblum and Calderbank [7] provide an example of a generalized cross-constellation method, which groups four of these 10-bit words together to obtain a rate $31/40 = 0.775$ code, utilizing 240 of the 252 available codewords. Let us take this code as C_1, with $K_1/N_1 = 31/40$. Notice that this construction allows for a very simple lossless compression scheme, where we simply map each of the 252 possible 10-bit DC-free words to a different 8-bit word (leaving four 8-bit words unassigned), so we have $K_c/N_c = 8/10$.

To complete our use of modified concatenation, we need to find a modulation code for the parity bits, where $K_2 = S = 8$. Since $\log_2\binom{11}{6} > 8$, one possible choice is a code of rate $K_2/N_2 = 8/11$, , where we alternate between a word with 6 ones and 5 zeros, and a word with 5 ones and 6 zeros. For the sake of this example, we will be flexible and say that the performance of this constraint is indistinguishable from being DC free over every block of 10 bits.

We then have the following alternatives:

- StdConcat with code C_2, with rate $K_2/N_2 = 8/11 = 0.7272$.
- StdConcat with code C_1, with rate $K_1/N_1 = 31/40 = \frac{31}{40} = 0.775$. (Error propagation occurs.)
- ModConcat with code C_1 on the message portion, and C_2 on the parity portion, along with a lossless compression code of rate $K_c/N_c = 8/10$. The rate is $K_1/N_1 = 0.775$ on the message portion.

Suppose we use $ID = 4$, so the number of user bytes per interleave is $M = 128$. For StdConcat, the number of channel bits is $\left\lceil\frac{ID\cdot n\cdot S}{K}\right\rceil N$ for a code of rate K/N. Meanwhile, for ModConcat,we have a total of $\left\lceil\frac{ID\cdot M\cdot S}{K_1}\right\rceil = 133$ modulation blocks of four 10-bit words, which get compressed into 4 bytes. Hence, we have $k' = \frac{1}{S\cdot ID}\left(\left\lceil\frac{ID\cdot M\cdot S}{K_1}\right\rceil N_1\right)\frac{K_c}{N_c} = 133$ message bytes per interleave, and the number of channel bits is given by $\left\lceil\frac{ID\cdot M\cdot S}{K_1}\right\rceil N_1 + ID\cdot 2t\cdot N_2 = 133\cdot 40 + 264 = 5584$.

DC-free code, $ID = 4$	RS input	RS output	Bits/sector	Rate	% of Max.
StdConcat with $C_2 = \frac{8}{11}$	$k = 128$	$n = 134$	5896	0.6947	91.1%
StdConcat with $C_1 = \frac{31}{40}$	$k = 128$	$n = 134$	5560	0.7367	96.7%
ModConcat with C_1, C_2	$k' = 133$	$n' = 139$	5584	0.7335	96.3%

For the "maximum" possible rate for this situation, we use $(\frac{1}{10}\log_2 252)\frac{128}{134} = 0.762$.

In Figure 6, we plot the performance of these three schemes, where we model the straightforward Viterbi detection scheme for the $1-D$ channel with an error distribution of $b(z) = b\cdot(\frac{1}{2}z^2 + \frac{1}{4}z^3 + \frac{1}{8}z^4 + \cdots)$, but the error characteristics may be different with other detection schemes. Then we see that ModConcat performs almost the same as StdConcat with C_2, while StdConcat with C_1 suffers greatly due to error propagation. Hence using modified concatenation, we

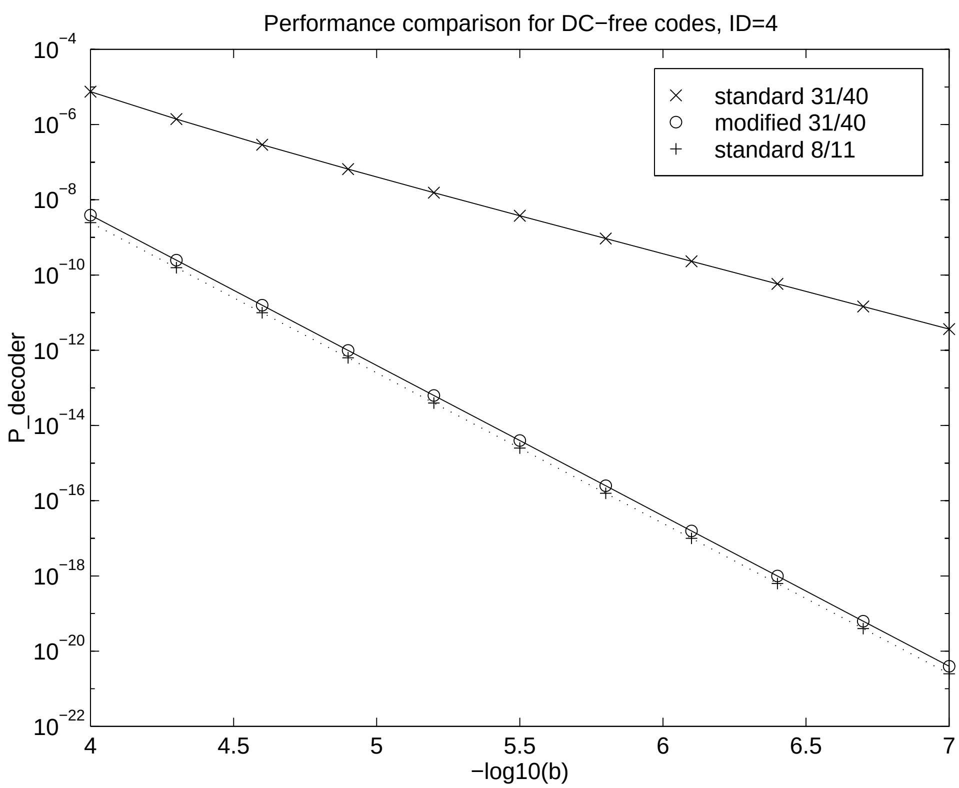

Figure 6: Performance comparison for DC-free codes

can implement the generalized cross constellation code C_1 and achieve the same performance as a StdConcat with C_2, thereby increasing the overall coding rate. This example also demonstrates how a lossless compression step can be effective in keeping down the size of the Reed-Solomon code, so that the expansion factor in this case is only $\alpha = \frac{N_1}{K_1}\frac{K_c}{N_c} = 1.032$.

5 Conclusion

We have considered a system comprising a simple detector and a block code concatenated with a Reed-Solomon code. We have analyzed performance of both standard and modified concatenation in terms of the parameters S, $\frac{K_1}{N_1}$, $\frac{K_2}{N_2}$, $\frac{K_c}{N_c}$, M, and t, and the bit error distribution $\{b_i\}$ out of the detector.

We considered magnetic data storage and showed that for two examples relevant to PRML in hard disk drives, modified concatenation performs better than standard concatenation. For a rate $\frac{16}{17}$ modulation code and 8-bit Reed-Solomon code, modified concatenation permits the use of three interleaves per sector, whereas standard concatenation requires four interleaves for good performance. Also, we showed how modified concatenation allows the practical implementation of a DC-free block code of length 40. In general, this technique allows the use of codes whose rates approach the capacity of the constraint, without a loss in performance.

It should be noted that this modified concatenation scheme also solves the error-propagation problem for sliding-window encoders, as well as long block codes, in concatenated systems. These ability to use sophisticated, high rate modulation codes may prove useful for many applications in data storage, such as hard disk drives, optical discs and digital video tape recorders. [15]. The co-design of modulation codes and block codes for lossless compression poses an interesting challenge. In addition, the benefits of erasure decoding, which is facilitated by modified concatenation, deserve to be explored.

6 Acknowledgements

The authors would like to express thanks to E.A. Gelblum, J.T. Gill, K.A.S. Immink, A. Patapoutian, and E. Soljanin for discussions related to this topic. They also thank J. Sonntag, N. Sayiner and AT&T Microelectronics (now Lucent) for their support and encouragement. The helpful comments from the anonymous reviewers were also greatly appreciated.

References

[1] K.A.S. Abdel-Ghaffar, M. Blaum, and J. Weber, "Analysis of Coding Schemes for Modulation and Error Control," *IEEE Trans. Inform. Theory*, vol. 41, no. 6, Nov. 1995, pp. 1955-1968.

[2] A. Bassalygo, "Correcting Codes with an additional property," *Prob. Inform. Transm.*, vol. 4, no. 1, Spring 1968, pp. 1-5.

[3] M. Blaum, "Combining ECC with Modulation: Performance Comparisons," *IEEE Trans. Inform. Theory*, vol. 37, no. 3, May 1991, pp. 945-949.

[4] W.G. Bliss, "Circuitry for Performing Error Correction Calculations on Baseband Encoded Data to Eliminate Error Propagation," *IBM Techn. Discl. Bul.*, vol. 23, pp. 4633-4634, 1981.

[5] R.D. Cideciyan, F. Dolivo, R. Hermann, W. Hirt, and W. Schott, "A PRML System for Digital Magnetic Recording," *IEEE Journal on Selected Areas in Commun.*, vol. 10, no. 1, Jan. 1992, pp. 38-56.

[6] A.B. Cooper III, "Soft Decision Decoding of Reed-Solomon Codes," in *Reed-Solomon Codes and their Applications*, ed. by S.B. Wicker and V.K. Bhargava. IEEE Press, 1994.

[7] E.A. Gelblum and A.R. Calderbank, "A Forbidden Rate Region for Generalized Cross Constellations," *IEEE Trans. Inform. Theory*, vol. 43, no. 1, Jan. 1997, pp. 335-341.

[8] J. Hagenauer and P. Hoeher, "A Viterbi Algorithm with Soft-Decision Outputs and its Applications," in *Proc. Globecom '89* (Dallas, TX, Nov. 1989), vol. 3, pp. 47.1.1-47.1.7.

[9] H.M. Hilden, D.G. Howe and E.J. Weldon, Jr., "Shift Error Correcting Modulation Codes," IEEE Trans. Magnetics, Vol. 27, No. 6, Nov. 1991, pp. 4600-4605.

[10] K.A.S. Immink, *Coding Techniques for Digital Recorders.* Prentice Hall, 1991.

[11] K.A.S. Immink, "A Practical Method for Approaching the Channel Capacity of Constrained Channels", to appear in *IEEE Trans. Inform. Theory*.

[12] R. Karabed and P.H. Siegel, "Matched Spectral-Null Codes for Partial-Response Channels," *IEEE Trans. Inform. Theory*, vol. 37, no. 3, May 1991, pp. 818-855.

[13] W.H. Kautz, "Fibonacci Codes for Synchronization Control," *IEEE Trans. Inform. Theory*, Apr. 1965, pp. 284-292.

[14] K. Knudsen, J. Wolf, L. Milstein, "A Concatenated Decoding Scheme for (1-D) Partial Response with Matched Spectral-Null Coding," *Proc. Globecom '93* (Houston, TX, Nov. 1993), pp. 1960-1964.

[15] J. C. Mallinson, *The Foundations of Magnetic Recording*, 2nd Ed., Academic Press, Inc., 1993.

[16] M. Mansuripur, "Enumerative Modulation Coding with Arbitrary Constraints and Post-Modulation Error Correction Coding and Data Storage Systems," Proceedings SPIE, vol. 1499, pp. 72-86, 1991.

[17] R.J. McEliece and L. Swanson, "On the Decoder Error Probability for Reed-Solomon Codes," *IEEE Trans. Inform. Theory*, vol. 32, no. 5, Sept. 1986, pp. 701-703.

[18] A. Patapoutian and P.V. Kumar, "The (d,k) Subcode of a Linear Block Code," *IEEE Trans. Inform. Theory*, vol. 38, no. 4, July 1992, pp. 1375-1382.

[19] P.N. Perry, "Runlength-Limited Codes for Single Error Detection in the Magnetic Recording Channel," *IEEE Trans. Inform. Theory*, vol. 41, no. 3, May 1995, pp. 809-814.

[20] E. Soljanin, "A Coding Scheme for Generating Bipolar Dc-Free Sequences," to appear in INTERMAG'97.

[21] E. Soljanin, "Decoding Techniques for Some Specially Constructed Dc-Free Codes," to appear in ICC'97.

[22] J. Sonntag, *et al.*, "A High Speed, Low Power PRML Read Channel Device," *IEEE Trans. Magnetics*, vol. 31, no. 2, March 1995, pp. 1186-1195.

[23] L.M.G.M. Tolhuizen, K.A.S. Immink, H.D.L. Hollmann, "Constructions and Properties of Block Codes for Partial-Response Channels," *IEEE Trans. Inform. Theory*, vol. 41, no. 6, Nov. 1995, pp. 2019-2026.

[24] A.J. van Wijngaarden and K.A.S. Immink, "Combinatorial Construction of High Rate Runlength-limited Codes," *Proc. IEEE Globecom (London)*, 1996, pp. 343-347.

[25] E.J. Weldon, "Coding Gain of Reed-Solomon Codes on Disk Memories," ICC '92, 346.5.1-6.

[26] S.B. Wicker, *Error control systems for digital communications and storage*, Prentice Hall, Inc., 1995.

[27] J.K. Wolf, "A Survey of Codes for Partial Response," *IEEE Trans. Magnetics*, vol. 27, no. 6,, Nov. 1991, pp. 4585-4589.

[28] O. Ytrehus, "Upper Bounds on Error-Correcting Runlength-Limited Block Codes," *IEEE Trans. Inform. Theory*, vol. 37, no. 3, May 1991, pp. 941-945.