ENEE 739C: Advanced Topics in Signal Processing: Coding Theory
Instructor: Alexander Barg

Lecture 8 (**draft;** 10/27/03). Asymptotically good families of codes. (Serially) concatenated codes and their decoding. Generalized minimum distance decoding.

http://www.enee.umd.edu/~abarg/ENEE739C/course.html

In this part we will start our study of constructive code families, i.e., codes that can be constructed and/or decoded in time polynomial in the code length such as $O(n^2)$ or even $O(n)$. With these restrictions it is still possible to construct very good codes, both in the asymptotic and "practical" sense.

PRODUCT CODES. Let $\mathcal{A}[n_0, k_0, d_0]$ and $\mathcal{B}[n_1, k_1, d_1]$ be two binary linear codes. Consider a linear code $\mathcal{C}$ whose codewords are $n_0 \times n_1$ matrices such that every column is a codeword in $\mathcal{A}$ and every row a codeword in $\mathcal{B}$ (in mathematical terms, $\mathcal{C} = \mathcal{A} \otimes \mathcal{B}$, the tensor product of the linear spaces $\mathcal{A}$ and $\mathcal{B}$).

Theorem 1. *$\mathcal{C}$ is an $[N = n_0 n_1, K = k_0 k_1, D = d_0 d_1]$ code. The number of codewords of weight $d_0 d_1$ is $A_{d_0}(\mathcal{A}) A_{d_1}(\mathcal{B})$, where the factors are the corresponding weight coefficients of the component codes.*

Proof : Clearly, $D \geq d_0 d_1$ because a nonzero row must contain at least d_1 ones and each of the d_1 nonzero columns must contain at least d_0 ones. To show the equality consider a codeword $\mathbf{a} \in \mathcal{A}$ of weight d_0 and suppose that $\mathrm{supp}(\mathbf{a}) = \{i_1, \ldots, i_{d_0}\}$. Let $\mathbf{b} \in \mathcal{B}$ be a codeword of weight d_1. Form a codeword of $\mathcal{C}$ as follows: rows $i_1, i_2, \ldots, i_{d_0}$ contain $\mathbf{b}$ and the rest is filled with zeros. Then $\mathrm{wt}(\mathbf{c}) = d_0 d_1$.

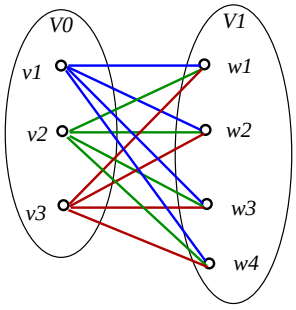
The claims about the dimension of $\mathcal{C}$ and the number of weight D codewords are left for Homework 2. ∎

We will call $\mathcal{A}$ the column code and $\mathcal{B}$ the row code.

Another useful way to think of product codes is as follows. Let $G = (V_0 \cup V_1, E)$ be a bipartite graph with $|V_0| = n_1, |V_1| = n_0$ and $\deg(v) = n_0$ for every $v \in V_0$ and $\deg(w) = n_1$ for every $w \in V_1$. The graph contains $N = n_0 n_1$ edges which are in an one-to-one correspondence with the coordinates of the product code $\mathcal{C}$ (we need some ordering of the edges in E). Let $\mathbf{x}(v)$ be the projection of a binary N-vector $\mathbf{x}$ on the coordinates in $E(v), v \in V_0 \cup V_1$.. By definition $\mathbf{c} \in \mathcal{C}$ if

(1) for every $v \in V_0$ the vector $\mathbf{c}(v) \in \mathcal{A}$,
(2) for every $w \in V_1$ the vector $\mathbf{c}(w) \in \mathcal{B}$.

Example: Consider $\mathcal{C} = \mathcal{A} \otimes \mathcal{B}$, where $\mathcal{A}[4, 2, 2]$ and $\mathcal{B}[3, 1, 3]$ are some linear codes (so $\mathcal{C}$ is a $[12, 2, 6]$ linear code). The graph $G = K_{4,3}$ has the following form:



A codeword $\mathbf{c} \in \mathcal{C}$ satisfies $\mathbf{c}(v_i) \in \mathcal{A}, \mathbf{c}(w_j) \in \mathcal{B}$. In this context we call the column code A the *left code* and the row code B the *right code*.

Let us write out a parity-check matrix $\mathbf{H}$ of the code $\mathcal{C}$. Take parity-check matrices of $\mathcal{A}$ and $\mathcal{B}$ in the form

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix} \qquad \mathbf{F} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

Let h_i, f_j denote the columns of $\mathbf{H}$ and $\mathbf{F}$ respectively. A parity-check matrix of $\mathcal{C}$ has the following form.

$$\begin{bmatrix} h_1 & h_2 & h_3 & h_4 & & & & & & & & \\ & & & & h_1 & h_2 & h_3 & h_4 & & & & \\ & & & & & & & & h_1 & h_2 & h_3 & h_4 \\ f_1 & & & & f_2 & & & & f_3 & & & \\ & f_1 & & & & f_2 & & & & f_3 & & \\ & & f_1 & & & & f_2 & & & & f_3 & \\ & & & f_1 & & & & f_2 & & & & f_3 \end{bmatrix}$$

Caution: this matrix has 14 rows while the code length $n = 12$, so it must have dependent rows.

DECODING of the code $\mathcal{C}$ can be accomplished in many ways. The simplest decoding algorithm is first to decode all the columns with the code $\mathcal{A}$ and then all the rows with the code $\mathcal{B}$ (or first rows and then columns).

Proposition 2. *This decoding corrects every error pattern of weight not exceeding* $(D-1)/4$.

Proof : Assume that $d_1 \geq d_0$. The error pattern of a fixed weight e that corrupts the most columns has weight $(d_0+1)/2$ in every column. If the number of miscorrected columns $\leq (d_1-1)/2$ then the error pattern will be corrected. This means that $e \leq (D-1)/4$.

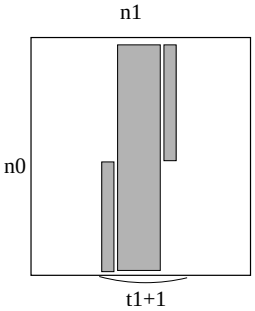
If $d_1 \leq d_0$ then the result is established by considering first row decoding and then column decoding. ∎

Computing the weight distribution of the code $\mathcal{C}$ given the weight distributions of the codes $\mathcal{A}$ and $\mathcal{B}$ is a difficult problem (in general the w.d. of $\mathcal{C}$ is not determined uniquely by the weight distributions of $\mathcal{A}$ and $\mathcal{B}$). However the number of vectors of weight $d_0 d_1$ equals the product $A_{d_0}(\mathcal{A}) A_{d_1}(\mathcal{B})$.

If the codes $\mathcal{A}, \mathcal{B}$ are asymptotically good then so is the code $\mathcal{C}$. However if $\mathcal{A}, \mathcal{B}$ meet the GV bound, the code $\mathcal{C}$ is substantially below it.

The product construction is just one of the forms of *interleaving* the transmission sequence. The purpose of introducing interleaving is fighting error bursts and other effects of memory in the channel. An error burst of length b is an error pattern in which errors are grouped within a window of b consecutive symbols in the transmission.

Proposition 3. *The product code* $\mathcal{C}$ *corrects error bursts of length* $\max(n_0 t_1, n_1 t_0)$. *Here* $t_i = \lfloor (d_i - 1)/2 \rfloor, i = 0, 1$.



Proof : If $n_0 t_1 \geq n_1 t_0$, we transmit by columns and decode first columns then rows. Then upon column decoding no row can continue more than t_1 errors which will be corrected by row decoding. ∎

OTHER OPERATIONS ON CODES. We have seen shortening and puncturing. There are a number of other operations used on linear codes to increase or reduce their length. In this way it is often possible to construct very good codes.

$|\mathbf{u}|\mathbf{u}+\mathbf{v}|$ construction. Let $\mathcal{C}_1[n, k_1, d_1], \mathcal{C}_2[n, k_2, d_2]$ be two linear codes. Consider a code $\mathcal{B}$ formed by vectors $|\mathbf{u}|\mathbf{u}+\mathbf{v}|$, where $\mathbf{u}_1 \in \mathcal{C}_1, \mathbf{v} \in \mathcal{C}_2$.

Proposition 4. *The parameters of* $\mathcal{B}$ *are* $[n, k_1 + k_2, \min(2d_1, d_2)]$.

Proof : Let $\mathbf{c} = |\mathbf{u}|\mathbf{u}+\mathbf{v}|$, where $\mathbf{u} \in \mathcal{C}_1 . \mathbf{v} \in \mathcal{C}_2$. If $\mathbf{v} = 0$ then $\mathrm{wt}(\mathbf{c}) \geq 2d_1$. Otherwise the smallest weight is obtained if $\mathbf{u} \prec \mathbf{v}$ and is at least $d_1 + (d_2 - d_1) = d_2$. ∎

The generator matrix of $\mathcal{B}$ has the form

$$\begin{bmatrix} \mathbf{G}_1 & \mathbf{G}_1 \\ \mathbf{0} & \mathbf{G}_2 \end{bmatrix}$$

In this way a Reed-Muller code of length 2^m is constructed from two RM codes of length 2^{m-1}:

$$\mathscr{R}(m,r) = |\mathbf{u}|\mathbf{u}+\mathbf{v}|, \quad \mathbf{u} \in \mathscr{R}(m-1,r),\ \mathbf{v} \in \mathscr{R}(m-1,r-1)|$$

We shall briefly discuss RM codes, their properties and decoding (see [6, 2]).

Suffix construction (construction X). Let $\mathscr{C}_1 \subset \mathscr{C}_2$ be codes with the parameters (n, M_1, d_1) and $(n, M_2 = bM_2, d_2)$. Let

$$\mathscr{C}_2 = (\mathbf{x}_1 + \mathscr{C}_1) \cup (\mathbf{x}_2 + \mathscr{C}_1) \cup \cdots \cup (\mathbf{x}_b + \mathscr{C}_1)$$

be a partition of $\mathscr{C}_2$ into "cosets" with respect to $\mathscr{C}_1$. Let $\mathscr{C}_3$ be an (m, b, d_3) code. Define a code $\mathcal{C}$ as

$$|\mathbf{x}_1 + \mathscr{C}_1|\mathbf{y}_1 \cup |\mathbf{x}_2 + \mathscr{C}_1|\mathbf{y}_2| \cup \cdots \cup |\mathbf{x}_b + \mathscr{C}_1|\mathbf{y}_b|$$

The parameters of the code $\mathcal{C}$ are $(n+m, M_2, \min(d_1, d_2 + d_3))$.

For instance, BCH codes can be used codes $\mathscr{C}_1, \mathscr{C}_2$ in this construction (we will discuss the BCH codes in more detail after reviewing finite fields in lecture 11). Many good codes can be obtained in this way.

ANOTHER WAY TO SHORTEN A CODE. Let $\mathcal{C}$ be an (n, M, d) code, $t \geq 0$, and let $\mathbf{x} \in \mathscr{H}_q^t$ be a vector. Let $E = \{n - t + 1, \ldots, n\}$. Consider the subset of codewords $(c_1, \ldots, c_n) \in \mathcal{C}$ such that their projection on E falls in the ball of radius r around $\mathbf{x}$:

$$\mathbf{c}(E) := (c_{n-t+1}, \ldots, c_n) \in \mathcal{B}_r(\mathbf{x}).$$

From every such vector delete the last t coordinates and denote the subset obtained by $\mathcal{A}$. The code $\mathcal{A}$ depends on $\mathbf{x}, r, t$.

Lemma 5. *For any* $\mathbf{x}$ *the code* $\mathcal{A}$ *has length* $n - t$ *and distance* $\geq d - 2r$. *There exists a vector* $\mathbf{x}$ *such that*

$$|\mathcal{A}| \geq B_r^t M / q^t.$$

where B_r^t *denotes the volume of the ball of radius* r *in* $\mathscr{H}_q^t$.

Proof : We use the usual trick: compute the cumulative size of the codes $\mathcal{A}_t(\mathbf{x})$ and average. A vector $\mathbf{c}(E)$ is contained in all the balls $\mathcal{B}_r(\mathbf{x})$ such that $\mathbf{x} \in \mathcal{B}_r(\mathbf{c}(E))$. We thus have

$$\sum_{\mathbf{x} \in \mathscr{H}_q^t} \sum_{\mathbf{c}:\mathbf{c}(E) \in \mathcal{B}_r(\mathbf{x})} 1 = \sum_{\mathbf{c} \in \mathcal{C}} \sum_{\mathbf{x} \in \mathcal{B}_e(\mathbf{c}(E))} 1 = B_r^t M.$$

There exists a vector $\mathbf{x}$ such that

$$\sharp\{\mathbf{c} : \mathbf{c}(E) \in \mathcal{B}_r(\mathbf{x})\} \geq q^{-t} B_r^t M.$$

The code $\mathcal{A}$ constructed from this vector $\mathbf{x}$ proves our claim. ∎

This lemma is used for establishing a recurrence relation on the parameters of the code $\mathcal{C}$ using upper bounds on the size of an $(n - t, *, d - t)$ code:

$$A(n,d) \leq \frac{q^t}{\sum_{i=0}^{r} \binom{t}{i} (q-1)^i} A(n-t, d-2t).$$

More operations on codes of the types discussed here appear in [6, 10].

CONCATENATED CODES. Consider a different way of combining two codes, called (serial) concatenation. Let $\mathcal{A}$ be a binary $[n_0, k_0 = R_0 n_0, d_0]$ code and $\mathcal{B}$ an $[n_1, k_1]$ Reed-Solomon (RS) code over $\mathbb{F}_{2^{k_0}}$. Through the rest of this lecture we write $R_1 = (\log_q |\mathcal{B}|)/n_1 = k_1/n_1$ (in deviation from our earlier definition involving base-2 logs).

Define a concatenated code $\mathcal{C}$ via its encoding procedure:

- the k_0k_1 bits of the message are viewed as a k_1-vector over $\mathbb{F}_{2^{k_0}}$ and encoded with the code $\mathcal{B}$ into a codevector $\mathbf{b}$,
- the vector $\mathbf{b}$ is viewed as a binary $k_0 \times n_1$ array C,
- the columns of C are encoded with the code $\mathcal{A}$.

We will denote the "concatenated product" of codes by $\mathcal{C} = \mathcal{A} \boxtimes \mathcal{B}$.

This construction is not as symmetric as product codes in the sense that we cannot switch the roles of $\mathcal{A}$ and $\mathcal{B}$ in encoding and decoding. It is also not clear (and often wrong) that the code meets the product bound, i.e., contains vectors of weight d_0d_1 : it can happen that the symbols in a vector of minimum weight of the RS code $\mathcal{A}$ are mapped by $\mathcal{B}$ on vectors of weight higher than d_0.

Proposition 6. *The parameters of the concatenated code $\mathcal{C}$ are $[N = n_0n_1, k = k_0k_1, D \geq d_0d_1]$.*

The parameter D is called the *designed distance* of the code. Its true distance is often greater than the designed distance.

Concatenated codes were introduced and analyzed by G.D. Forney [5]. They proved to be a very fruitful idea both theoretically and in numerous practical applications. Their properties can be summarized as follows:

(1) There are families of asymptotically good concatenated codes that can be constructed and decoded to correct $D/2$ errors. These codes also attain capacity of a DMC and Gaussian channels with a positive error exponent. These results can be attained by decoding algorithms of complexity $O(N^2)$.
(2) Ensembles of random concatenated codes contain codes that meet the GV bound. Under ML decoding the performance of these codes is the same as that of random linear codes (i.e., they achieve the random coding exponent).
(3) Simple decoding algorithms exist that correct up to $D/2$ errors where D is the designed distance of the code.

Let us quote the ensemble result (2) (from [3, 8, 9]).

Theorem 7. *Consider the ensemble of concatenated codes with an outer $[n_1, n_1R_1]$ RS code $\mathcal{B}$ of rate R_1 and random $[n_0, n_0R_0]$ inner code $\mathcal{A}$. It contains codes with relative distance $\delta(R)(1 - o(1))$, where*

$$\delta(R) = \begin{cases} \delta_{\mathrm{GV}}(R) & R_0 \geq \log_2(2(1 - \delta_{\mathrm{GV}}(R))) \\ \frac{R-R_0}{\log(2^{1-R_0}-1)} & \textit{otherwise.} \end{cases}$$

This result is proved by a rather tedious (and not very insightful) calculation of the average weight distribution in the code ensemble. We will discuss in more detail the other two results. Before doing this, let us also remark that the ensemble weight distribution results enable one to claim that concatenated codes *under max likelihood decoding* achieve the random coding exponent $E_0(R, p)$. This claim is made under a restriction on the inner rate R_0 similar to the one in Theorem 6.

The main emphasis in the study of code concatenation is on decoding. A simple possibility is to first decode all the received columns with the code $\mathcal{A}$, then form a "received vector" for the code $\mathcal{B}$ and decode it. If $\mathcal{A}$ is decoded by a bounded distance decoding algorithm (which can also produce erasures) then decoding of the code $\mathcal{A}$ must be able to handle both errors and erasures. A foremost example is the RS code for which there exist low-complexity algebraic algorithms which correct any combination of e errors and x erasures as long as $2e + x \leq d_1 - 1$.

Proposition 8. *Suppose that the column code is used for decoding up to $t_0 = \lfloor (d_0 - 1)/2 \rfloor$ and the row code decodes e errors and x erasures as long as $2e + x \leq d_1 - 1$. If the true distance of $\mathcal{C}$ equals the designed distance, then the smallest weight of an uncorrectable error is*

$$(t_0 + 1)\Big(\Big\lfloor \frac{d_1 - 1}{2} \Big\rfloor + 1\Big).$$

Proof : Let $t_1 = \lfloor \frac{d_1-1}{2} \rfloor$. Let $\mathbf{c}_1, \mathbf{c}_2$ be two codewords of $\mathcal{C}$ that differ in d_0d_1 bits. This means that the corresponding outer codewords $\mathbf{b}_1, \mathbf{b}_2$ are at distance d_1 and in these columns the corresponding inner codewords are at distance d_0. Consider an error pattern with the following properties

(1) it is located in t_1+1 columns within the support $\mathrm{supp}(\mathbf{b}_1-\mathbf{b}_2)$,

(2) in each of these columns it contains t_0+1 bits which make the inner decoder miscorrect.

Then the overall algorithm decodes wrongly. It is also clear that any other error pattern which leads to a decoding error has weight $(t_0+1)(t_1+1)$ or greater. ∎

Thus decoding described corrects up to $D/4$ (quarter of the product bound $D=d_0d_1$) errors. This falls below our expextations of correcting errors up to half the designed distance. This problem is addressed next.

GMD DECODING. Our exposition follows [4]. Consider a concatenated code $\mathcal{C}=\mathcal{A}\boxtimes\mathcal{B}$. In this context $\mathcal{B}$ can be any code for which there is an algebraic decoding procedure which corrects any combination of e errors and x erasures as long as $2e+x\le d_1-1$. (You may always assume that $\mathcal{B}$ is an RS code.)

Suppose we decode the columns $\mathbf{y}_1,\dots,\mathbf{y}_{n_1}$ of the received vector and obtain vectors of the inner code or erasures. The idea is that we can extract more information from inner decoding. namely, we can also estimate the reliability of our decisions by computing the distance $\mathrm{d}(\mathbf{y}_i,\mathbf{a}_i)$ (if the ith column is not erased by inner decoding). We will then treat the symbols of the outer received vector differently based on their reliabilities.

Let

$$w_i=\begin{cases}1/2 & \text{if inner decoding erases the } i\text{th column}\\ \mathrm{d}(\mathbf{y}_i,\mathbf{a}_i)/d_0 & \text{if it outputs } \mathbf{a}_i\in\mathcal{A}.\end{cases}$$

The numbers w_i should be thought of as reliability weights of columns.

Consider the following decoding algorithm due to G. D. Forney [5] Decode every column with the inner code. Find the weight vector $\mathbf{w}=(w_1,\dots,w_{n_1})$. Let us assume wlog that $1/2\ge w_1\ge\cdots\ge w_{n_1}$. Erased columns receive the worst reliability weight of $1/2$. Form a "received vector" $\mathbf{z}=(z_1,z_2,\dots,z_{n_1})$ of the outer code where some of the left entries may be erasures. In the first iteration the vector $\mathbf{z}=\mathbf{z}_0$ is decoded to correct errors and erasures. In each subsequent iteration we erase the leftmost nonerased symbol and decode the obtained vector $\mathbf{z}_i$.

Stopping condition. Let

$$W(\mathbf{z},\mathbf{b})=\sum_{i:z_i=b_i}w_i+\sum_{i:\ z_i\ne b_i}(1-w_i).$$

The algorithm stops if it finds a vector $\mathbf{b}\in\mathcal{B}$ such that $W(\mathbf{z}_i,\mathbf{b})<d_1/2$ (and outputs $\mathbf{b}$) or if in the next iteration the number of erased symbols in $\mathbf{z}_i$ would reach d_1 (in this case we detect an error).

Remarkably, GMD decoding enables one to correct twice more errors that the basic algorithm described above.

Theorem 9. *The GMD algorithm corrects all errors of weight* $\le(D-1)/2$.

Proof : Let us fix notation:
$\mathbf{c}=(\mathbf{c}_1,\dots,\mathbf{c}_{n_1})$ is the transmitted word of $\mathcal{C}$, where $\mathbf{c}_i\in\mathcal{A}, i=1,\dots,n_1$;
$\mathbf{y}=(\mathbf{y}_1,\dots,\mathbf{y}_{n_1})$ is the received vector;
$\mathbf{b}=(b_1,\dots,b_{n_1})$ is the vector of $\mathcal{B}$ that corresponds to $\mathbf{c}$;
$\mathbf{z}=(z_1,\dots,z_{n_1})$ is a generic notation for the "received vector" of the code $\mathcal{B}$, i.e., an n_1-vector of erasures and q-ary symbols.

Our plan is to show that if the weight of error is below $(D-1)/2$ then the stopping condition $W(\mathbf{z}_j,\mathbf{b})<d_1/2$ will hold for one and only one of the vectors $\mathbf{z}_j$. Next we will show that if $\mathbf{z}$ satisfies $W(\mathbf{z},\mathbf{b})<d_1/2$ then the number of erasures ℓ plus twice the number of distinct nonerased coordinates in $\mathbf{z}$ and $\mathbf{b}$ satisfies

$$W_\ell(\mathbf{z},\mathbf{b})=\frac{\ell}{2}+\mathrm{d}_{\mathrm{nonerased}}(\mathbf{z},\mathbf{b})\le\frac{d_1-1}{2},$$

so $\mathbf{z}$ will be found in one of the GMD decoding iterations.

The first part is easy. For simplicity we write $\mathbf{z}$ instead of $\mathbf{z}_j$. Assume that $\mathrm{d}(\mathbf{y},\mathbf{c}) \le D/2$. We have

$$\begin{aligned} d_0 W(\mathbf{z},\mathbf{b}) &= \sum_{i:z_i=b_i} \mathrm{d}(\mathbf{y}_i,\mathbf{c}_i) + \sum_{i:z_i\neq b_i} (d_0 - \mathrm{d}(\mathbf{y}_i,\mathbf{c}_i)) \\ &\le \mathrm{d}(\mathbf{y},\mathbf{c}) < d_0 d_1/2, \end{aligned}$$

which shows that the stopping condition holds for the vector $\mathbf{z}$. Moreover, there is at most one codeword $\mathbf{b} \in \mathcal{B}$ which satisfies $W(\mathbf{z},\mathbf{b})$. Indeed, let $\mathbf{b}', \mathbf{b}'' \in \mathcal{B}$ and let i be such that $b'_i \neq b''_i$. Then

$$W(z_i, b'_i) + W(z_i, b''_i) \ge 1,$$

(an inequality can occur if $z_i \neq b'_i, z_i \neq b''_i$) so

$$W(\mathbf{z},\mathbf{b}') + W(\mathbf{z},\mathbf{b}'') \ge d_1.$$

Then $W(\mathbf{z},\mathbf{b}') < d_1/2$ implies $W(\mathbf{z},\mathbf{b}'') > d_1/2$.

Let us prove the second part. For that observe that a given vector $W = (w_1, \ldots, w_{n_1})$ of reliabilities can be written as

$$W = \sum_{\ell=0}^{n_1} \omega_\ell W_\ell,$$

where

$$W_\ell = ((1/2)^\ell 0^{n_1-\ell}) \quad (\ell = 0, 1, \ldots, n_1),$$

and $\omega_0 = 1 - 2w_1$, $\omega_\ell = 2(w_{\ell-1} - w_\ell)$ for $\ell = 1, \ldots, n_1 - 1$ and $\omega_{n_1} = 2w_{n_1}$. Note that $\sum \omega_\ell = 1$. Hence $W(\mathbf{z},\mathbf{b})$ can be written as a convex combination of the functions $W_\ell(\mathbf{z},\mathbf{b})$:

$$W(\mathbf{z},\mathbf{b}) = \sum_{\ell=0}^{n_1} \omega_\ell W_\ell(\mathbf{z},\mathbf{b}).$$

Observe that it is not possible that every term on the right exceeds $d_1/2$ since this would contradict the assumption $W(\mathbf{z},\mathbf{b}) < d_1/2$. Hence $W_\ell(\mathbf{z},\mathbf{b}) < d_1/2$ for some ℓ. This just means that the vector $\mathbf{z}_\ell$ obtained by erasing the left ℓ coordinates of $\mathbf{z}$ will decode to $\mathbf{b}$. ∎

Note that GMD decoding in principle can be used for any code if the reliability values are provided by the channel detector rather than the inner decoder. Therefore, GMD decoding has been also analyzed on a Gaussian channel and in general on a discrete-input continuous-output memoryless channel (semicontinuous in Wolfowitz's terminology). While it is possible to prove that GMD decodes up to half the designed distance, exact characterization of the decoding domains remains elusive. We refer to [4] for a comprehensive discussion of these remarks.

Generalized Minimum Distance Decoding is a fundamental idea: after 35 years it is still a cornerstone of our understanding of decoding of multilevel (concatenated) schemes. It is also remarkably resistant to improvement attempts (we will elaborate when we discuss list decoding of RS codes).

In the GMD algorithm above we suggested to erase the least reliable symbols one by one, which leads to $d_1 - 1$ trials of outer decoding. There are ways of accomplishing the same by fewer trials (in particular, it is clear that we can erase those symbols in pairs). Let us consider the other extreme: what should we do if we are allowed just two attempts of outer decoding. In other words, we want to find an optimal threshold τ (which does not depend on the received vector) such that if all the symbols whose reliability is below it are erased we correct the largest number of errors. Roughly the answer is given by $\tau = d_1/3$ which enables us to correct up to $D/3$ errors. This is an improvement over the $D/4$ estimate of the basic algorithm.

Theorem 10. *Let $\tau = (d_1 - 2)/3$. The GMD algorithm with one threshold τ corrects any combination of up to $d_0 d_1/3 - 1$ errors.*

CONSTRUCTIVE, ASYMPTOTICALLY GOOD FAMILIES OF CONCATENATED CODES. In the next theorem we show that there exist families of binary concatenated codes that are asymptotically good and can be constructed with polynomial complexity.

Theorem 11. *(a) Let R be the code rate. There exists a concatenated code $\mathcal{C}[N, NR]$ whose distance meets the bound*

$$\delta_Z(R) = \max_{R \le x \le 1} \delta_{\mathrm{GV}}(x)(1 - R/x) \quad \text{(Zyablov bound)}.$$

The construction complexity of the code is $O(N^2)$. The code can be GMD-decoded to correct a $\frac{1}{2}\delta_Z(R)$ proportion of errors.

(b) There exists a concatenated code of rate R which on a BSC(p) provides the error probability of decoding falling as $P_e(\mathcal{C}, p) \le 2^{-NE_F(R,p)-o(N)}$ where

$$E_F(R, p) = \max_{R \le x \le \mathscr{C}} E_0(x, p)(1 - R/x) \quad \text{(Forney bound)}$$

This error exponent is attained under an $O(N^2)$ GMD decoding algorithm.

Proof : (a) Let $\mathcal{A}$ be a code of rate R_0 and growing length n_0 which meets the GV bound. Let $\mathcal{B}$ be an $[n_1 = q, k_1 = R_1 n_1, d_1]$ q-ary extended Reed-Solomon code ($q = 2^{k_0}$). By the product bound

$$\frac{D}{N} = \frac{n_1 - k_1 + 1}{n_1} \delta_{\mathrm{GV}}(R_0) \ge (1 - R_1)\delta_{\mathrm{GV}}(R_0).$$

Let us analyze the construction complexity of the code $\mathcal{C}$. We have $N = n_0 n_1 = n_0 q = n_0 2^{R_0 n_0}$, so $n_0 \approx (\log N)/R_0$. The construction complexity of the code $\mathcal{A}$ is $2^{n_0 - k_0} \approx N^{\frac{1}{R_0} - 1}$. The power of N is below 2 for rates $R_0 \ge 1/3$. To contain complexity for smaller rates R_0 we have to use other component codes such as Hermitian codes.

(b) (OUTLINE) This time we take $\mathcal{A}$ to be a code of rate R_0 that achieves the random coding exponent $E_0(R, p)$ of the BSC[1]. Suppose that the inner decoder outputs a weight $\alpha_i \in [0, 1]$ which measures the reliability of its decision in the ith column of the received word. It is assumed that the value of α_i is a function of the likelihood $\beta(\mathbf{y}_i)$ computed by the inner decoder based on the received value of the ith column. As in the proof of the GMD theorem (Thm. 8) we can show that the GMD decoder will decode incorrectly if and only if

$$\sum_{i=1}^{n_1} \alpha_i' \le n_1 - d_1,$$

where $\alpha_i' = \alpha_i$ if $\mathbf{y}_i$ is decoded correctly and $\alpha_i' = -\alpha_i$ otherwise. Finally the probability of the event $\sum \alpha_i' \le n_1 - d_1$ is estimated using the Chernov bound, which yields the result. The full argument (which becomes fairly technical) is given in [5, pp.86-87].

It is easier to give a proof of an error probability estimate which has a $\frac{1}{2}E_F$ error exponent:

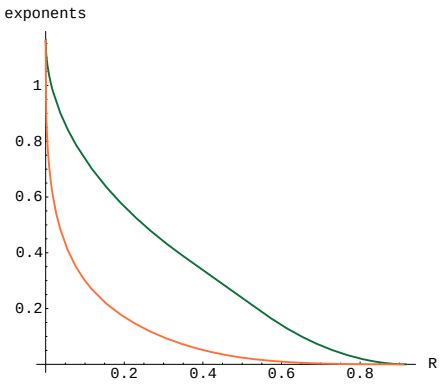
$$P_e \le \sum_{i=t_1+1}^{n_1} \binom{n_1}{i} p_0^i (1 - p_0)^{n_1 - i}, \tag{1}$$

where $p_0 = 2^{-n_0 E_0(R_0, p)}$. For $n_1 \to \infty$ the exponent of this expression is exactly the claimed $\frac{1}{2}E_F$.

The complexity estimate is related to the complexity of GMD decoding. Under a straightforward implementation like the one discussed above, it is $O(n_1^3)$ which yields the the overall $O(N^3)$ complexity estimate. There is a faster GMD procedure which involves only a small amount of additional computations in passing from step to step of GMD decoding [7]. This result gives the $O(N^2)$ complexity estimate. ∎

We see that concatenated codes reach capacity of the BSC under an $O(N^2)$ decoding algorithm. The error exponent is substantially smaller than the random coding exponent (see the plot), but positive for all $R \le \mathscr{C}$.

[1]The result does not depend on the BSC assumption. For another channel just take an inner code that achieves the random coding exponent E_0 of that channel.



The case of $R \to \mathscr{C}$ is the most annoying: put $\epsilon = \mathscr{C} - R$, then $E_0(R,p) = \Omega(\epsilon^2)$ while $E_F(R,p) = \Omega(\epsilon^3)$.

Remarks. 1. We have seen in Part (b) that if instead GMD decoding we perform algebraic decoding to correct $d_1/2$ errors, the error exponent attained is $\frac{1}{2}E_F(R,p)$. Performing errors-and-erasures decoding we can achieve a $\frac{2}{3}E_F(R,p)$ error exponent.

2. It is possible to achieve capacity under an $O(N^2)$ decoding. Is the communication problem solved? Yes and no: the task of inner decoding involves constructing and decoding a random-like inner binary code. The length of the code is only about $\log N$, so theoretically its brute-force decoding is polynomial. Yet there is an uneasy feeling that the result is not 100% clean even though the complexity-theoretic argument looks flawless. I would like to point out one way to quantify this remark: suppose that $R = \mathscr{C} - \epsilon$. Then since $R_0 > R$, the difference $\mathscr{C} - R_0 < \epsilon$. We know that if $\epsilon \to 0$ then $E_0(\mathscr{C} - \epsilon, p) \approx \epsilon^2$. Now consider equation (1): the logarithm of the term for $i = t_1 + 1$ behaves as

$$(const)n_1 - n_0 E(R_0,p)n_1(1 - R_1).$$

We would like to claim that this quantity is negative. We start running into problems if $E(R_0,p) \to 0$: we then need that $n_0 > 1/\epsilon^2$; hence the decoding complexity of inner codes grows as $2^{R_0 n_0} = 2^{O(\epsilon^{-2})}$.

The question of whether this theorem is indeed constructive was discussed in the literature in the 1970s (see [1]). It was reiterated in a plenary talk at the ISIT 2001, where R. J. McEliece suggested to look at codes from the point of view of estimating the number of operations *per message bit* needed to guarantee reliable transmission for rates ϵ away from capacity.

3. The results of this theorem can be improved in a number of ways: by considering multilevel concatenations, by taking algebraic geometry codes as outer codes B instead of RS codes. We will indicate another way of improving this theorem when we discuss bipartite-graph (expander) codes.

OTHER VERSIONS OF SERIAL CONCATENATION. Let us consider a code construction somewhere between product and concatenated codes. Let $\mathcal{A}[n_0, n_1]$ and $\mathcal{B}[n_1, k_1]$ be the inner and the outer codes. Denote by m an integer parameter called the "interleaving depth." We will define a serial concatenation $\mathcal{C}$ of the codes $\mathcal{A}$ and $\mathcal{B}$ by specifying the encoding procedure. Encoding operates on an array of mk_1 message bits which are encoded with the code $\mathcal{B}$. This results into m codewords of $\mathcal{B}$ totaling mn_1 bits. These bits are permuted according to a pre-defined mapping called an *interleaver*. The $m \times n_1$ bit array obtained is then encoded with the inner code $\mathcal{B}$. We obtain m codewords of the inner code. The rate of the code $\mathcal{C}$ equals

$$\frac{mk_1}{mn_0} = \frac{n_1}{n_0}\frac{k_1}{n_1} = R_0 R_1.$$

The main problem addressed for serially concatenated codes is estimating the bit error rate under iterative decoding.

The success of Forney's concatenated codes relies upon employing Reed-Solomon codes in the outer level. These codes possess efficient, low-complexity algebraic decoding algorithms. This makes the overall scheme amenable to theoretical analysis and opens a way for its applications in communication systems. The most prominent of these applications is coding for audio and data compact disks; others include deep-space communication. No comparable results are known for other versions of serially concatenated codes.

References

1. L. A. Bassalygo, *Formalization of the problem of the complexity of code assignment*, Problemy Peredači Informacii **12** (1976), no. 4, 105–106.
2. R. E. Blahut, *Algebraic codes for data transmission*, Cambridge University Press, 2003.
3. E. L. Blokh and V. V. Zyablov, *Existence of linear concatenated binary codes with optimal correcting properties*, Problems of Information Transmission **9** (1973), 3–10.
4. I. Dumer, *Concatenated codes and their multilevel generalizations*, Handbook of Coding Theory (V. Pless and W. C. Huffman, eds.), vol. 2, Elsevier Science, Amsterdam, 1998, pp. 1911–1988.
5. G. D. Forney, Jr., *Concatenated codes*, MIT Press, Cambridge, MA, 1966.
6. F. J. MacWilliams and N. J. A. Sloane, *The theory of error-correcting codes*, 3 ed., North-Holland, Amsterdam, 1991.
7. U. K. Sorger, *A new Reed-Solomon code decoding algorithm based on Newton's interpolation*, IEEE Trans. Inform. Theory **39** (1993), no. 2, 758–765.
8. C. Thommesen, *The existence of binary linear concatenated codes with Reed-Solomon outer codes which asymptotically meet the Gilbert-Varshamov bound*, IEEE Trans. Inform. Theory **29** (1983), 850–853.
9. ———, *Error-correcting capabilities of concatenated codes with MDS outer codes on memoryless channels with maximum-likelihood decoding*, IEEE Trans. Inform. Theory **33** (1987), no. 5, 632–640.
10. M. Tsfasman and S. Vlăduţ, *Algebraic-geometric codes*, Kluwer, Dordrecht, 1991.