Non-binary Convolutional Codes with Good Distance Properties

Roxana Smarandache

University of Notre Dame
Notre Dame, IN 46566

Basen on joint work with:
Joachim Rosenthal, Heide Gluesing-Luerssen,
Ryan Hutchinson, Virtu Tomás, Diego Napp

Banff, October 12, 2015

Convolutional Codes

- Convolutional codes were introduced (in 1955) as alternatives to block codes.
- The encoder contains memory: the n-output depends on m previous k-input blocks.
- n, k are typically small but m is typically large to achieve low error probability.
- $[n, k, m]$ convolutional codes versus $[N, K]$ block codes.
- $[n, k, 0]$ convolutional codes = $[n, k]$ block codes.
- It can be implemented with an k-input, n-output linear sequential circuit, with input memory m.

Convolutional Codes

An (n, k) convolutional code $\mathcal{C}$ is defined either

- as a k-dimensional subspace of $\mathbb{F}(D)^n$ or
- as a rank k-submodules of $\mathbb{F}[D]^n$.

Definition

$$\mathcal{C} = \left\{ \mathbf{u}(D)G(D) \in \mathbb{F}[D]^n \text{ with } \mathbf{u}(D) \in \mathbb{F}[D]^k \right\},$$

Definition (The memory and degrees)

$$m = \max_{\substack{1 \le i \le k \\ 1 \le j \le n}} \{\deg(g_{ij}(D))\} : \text{the memory of } G(D).$$

$$\delta_e = \sum_{i=1}^{k} \max_{1 \le j \le n} \{\deg(g_{ij}(D))\} : \text{ the external degree or total memory of } G(D).$$

$$\delta_i = \max\{\deg(k \times k \text{ minors of any } G(D) \text{ of } \mathcal{C})\} : \text{the internal degree of } G(D).$$

Convolutional Codes

An (n, k) convolutional code $\mathcal{C}$ is defined either

- as a k-dimensional subspace of $\mathbb{F}(D)^n$ or
- as a rank k-submodules of $\mathbb{F}[D]^n$.

Definition

$$\mathcal{C} = \left\{ \mathbf{u}(D)G(D) \in \mathbb{F}[D]^n \text{ with } \mathbf{u}(D) \in \mathbb{F}[D]^k \right\},$$

Definition (The memory and degrees)

$m = \max\limits_{\substack{1 \le i \le k \\ 1 \le j \le n}} \{\deg(g_{ij}(D))\}$: the memory of $G(D)$.

$\delta_e = \sum\limits_{i=1}^{k} \max\limits_{1 \le j \le n} \{\deg(g_{ij}(D))\}$: the external degree or total memory of $G(D)$.

$\delta_i = \max\{\deg(k \times k \text{ minors of any } G(D) \text{ of } \mathcal{C})\}$: the internal degree of $G(D)$.

Definitions

$$\mathcal{C} = \left\{ \mathbf{v}(D) \in \mathbb{F}[D]^n \ : \mathbf{v}(D) = \mathbf{u}(D)G(D) \text{ with } \mathbf{u}(D) \in \mathbb{F}[D]^k \right\},$$

Example

$$G(D) = \begin{bmatrix} 1+D & 1+D & 1 \\ D & 1 & 1+D \end{bmatrix} \begin{matrix} \longrightarrow m_1 = 1 \\ \longrightarrow m_2 = 1 \end{matrix} \longrightarrow \delta_e = 2, \delta_i = 2, m = 1.$$

$m = \max\limits_{\substack{1 \le i \le k \\ 1 \le j \le n}} \{\deg(g_{ij}(D))\}$: the memory of $G(D)$.

$\delta_e = \sum\limits_{i=1}^{k} \max\limits_{1 \le j \le n} \{\deg(g_{ij}(D))\}$: the external degree or total memory of $G(D)$.

$\delta_i = \max\{\deg(k \times k \text{ minors of any } G(D) \text{ of } \mathcal{C})\}$: the internal degree of $G(D)$.

Assumptions

We assume that:

- An infinite sequence cannot generate a finite sequence.
  - $G(D)$ with this property is called non-catastrophic.
  - Equivalently, the gcd of the full size minors of $G(D)$ is D^{ℓ}, for some ℓ.
- $G(D)$ has minimum internal degree δ_i among all polynomial generator matrices for $\mathcal{C}$.
  - $G(D)$ with this property is called *basic*.
  - Equivalently, the gcd of the full size minors of $G(D)$ is 1.

Polynomial generator matrix

A code can be described using:

- the $k \times n$ polynomial matrix:

$$G(D) = (g_{ij}(D))_{k \times n} = G_0 + G_1 D + G_2 D^2 + \ldots + G_m D^m,$$

where the message is a polynomial vector

$$(u_1(D), u_2(D), \ldots, u_k(D)) \in \mathbb{F}^k[D].$$

The sliding generator matrix

A code can be described using:

- The sliding matrix

$$G = \begin{bmatrix} G_0 & G_1 & G_2 & \dots & G_m & & & \\ & G_0 & G_1 & G_2 & \dots & G_m & & \\ & & G_0 & G_1 & G_2 & \dots & G_m & \\ & & & \ddots & \ddots & \ddots & & \ddots \end{bmatrix},$$

where the message is a scalar vector is

$$(u_0, u_1, \dots), \quad \text{with} \quad u_i \in \mathbb{F}^k.$$

Multiplexed polynomial generator matrix

A code can be described using:

- the $k \times 1$ multiplexed polynomial generating matrix:

$$G(D)_{k\times 1} = \begin{bmatrix} g_1(D) \\ g_2(D) \\ \vdots \\ g_k(D) \end{bmatrix}, \quad \text{where} \quad g_i(D) = \sum_{j=1}^{n} g_{ij}(D^n) D^{j-1},$$

and the message is a polynomial vector

$$\left(u_1(D^n), u_2(D^n), \ldots, u_k(D^n)\right) \in \mathbb{F}^k[D^n].$$

Parity-check matrix representation

A code with $G(D)$ basic can be described using:

- a rank $(n-k)$ polynomial parity-check matrix:

$$\mathcal{C} = \left\{ \mathbf{v}(D) \in \mathbb{F}[D]^n \mid H(D)\mathbf{v}^T(D) = \mathbf{0}^T \in \mathbb{F}[D]^{n-k} \right\}.$$

Let

$$\mathbf{v}(D) = \mathbf{v}_0 + \mathbf{v}_1 D + \ldots + \mathbf{v}_l D^l$$
$$H(D) = H_0 + H_1 D + \ldots + H_\nu D^\nu,$$

then $v(D) \in \mathcal{C}$ if and only if

$$\begin{bmatrix} H_0 & & & & \\ \vdots & \ddots & & & \\ H_\nu & \ldots & H_0 & & \\ & \ddots & & \ddots & \\ & & H_\nu & \ldots & H_0 \\ & & & \ddots & \vdots \\ & & & & H_\nu \end{bmatrix} \begin{bmatrix} \mathbf{v}_0 \\ \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_l \end{bmatrix} = \mathbf{0}.$$

Example

- Let
$$G(D) \triangleq \begin{bmatrix} 1+D & 1+D & 1 \\ D & 1 & 1+D \end{bmatrix}$$
be a polynomial generator matrix for $\mathcal{C}$, i.e.,
$$\mathcal{C} = \{v(D) = [u_1(D), u_2(D)]\, G(D) \mid u_1(D), u_2(D) \in \mathbb{F}_2[D]\}.$$
- Let
$$G(D) = \underbrace{\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}}_{G_0} + \underbrace{\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}}_{G_1} D,$$
$$[u_1(D), u_2(D)] = \underbrace{[u_{10}, u_{12}]}_{\mathbf{u}_0} + \underbrace{[u_{11}, u_{12}]D}_{\mathbf{u}_1 D} + \ldots$$
Then
$$\mathcal{C} = [\mathbf{u}_0, \mathbf{u}_1, \ldots] \begin{bmatrix} G_0 & G_1 & & \\ & G_0 & G_1 & \\ & & \ddots & \ddots \end{bmatrix}.$$

Example

- Let
$$G(D) \triangleq \begin{bmatrix} 1+D & 1+D & 1 \\ D & 1 & 1+D \end{bmatrix}$$
be a polynomial generator matrix for $\mathcal{C}$, i.e.,
$$\mathcal{C} = \{v(D) = [u_1(D), u_2(D)]\, G(D) \mid u_1(D), u_2(D) \in \mathbb{F}_2[D]\}.$$
- Let
$$\begin{aligned} g_1(D) = g_{11}(D^3) + g_{12}(D^3)D + g_{13}(D^3)D^2 &= 1 + D^3 + D(1+D^3) + D^2(1) \\ &= 1 + D + D^2 + D^3 + D^4 \\ g_2(D) = g_{21}(D^3) + g_{22}(D^3)D + g_{23}(D^3)D^2 &= D^3 + D(1) + D^2(1+D^3) \\ &= D + D^2 + D^3 + D^5. \end{aligned}$$
Then
$$\mathcal{C} = [u_1(D^3), u_2(D^3)] \begin{bmatrix} g_1(D) \\ g_2(D) \end{bmatrix} = [u_1(D^3), u_2(D^3)] \begin{bmatrix} 1 + D + D^2 + D^3 + D^4 \\ D + D^2 + D^3 + D^5 \end{bmatrix}.$$

Example

- Let
$$G(D) \triangleq \begin{bmatrix} 1+D & 1+D & 1 \\ D & 1 & 1+D \end{bmatrix}$$
be a polynomial generator matrix for $\mathcal{C}$, i.e.,
$$\mathcal{C} = \{v(D) = [u_1(D), u_2(D)]\, G(D) \mid u_1(D), u_2(D) \in \mathbb{F}_2[D]\}.$$
- Let
$$H(D) = \begin{bmatrix} D^2 & 1+D+D^2 & 1+D^2 \end{bmatrix}$$
Then $H(D)G(D)^T = 0$ and $H(D)$ is a parity-check matrix of $\mathcal{C}$.

Let
$$H(D) = \underbrace{\begin{bmatrix} 0 & 1 & 1 \end{bmatrix}}_{H_0} + \underbrace{\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} D}_{H_1 D} + \underbrace{\begin{bmatrix} 1 & 1 & 1 \end{bmatrix} D}_{H_2 D^2},$$
$$[v_1(D), v_2(D), v_3(D)] = \underbrace{[v_{10}, v_{12}, v_{13}]}_{\mathbf{v}_0} + \underbrace{[v_{11}, v_{12}, v_{13}]D}_{\mathbf{v}_1 D} + \underbrace{[v_{21}, v_{22}, v_{23}]D}_{\mathbf{v}_2 D^2} \ldots$$

Then
$$\begin{bmatrix} H_0 & & & \\ H_1 & H_0 & & \\ H_2 & H_1 & H_0 & \\ & \ddots & \ddots & \ddots \end{bmatrix} \cdot \begin{bmatrix} \mathbf{v}_0^T \\ \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \end{bmatrix} = \mathbf{0}^T.$$

The free distance

$$d_{\text{free}}(\mathcal{C}) = \min\{\text{wt}(\mathbf{v}(D)) \mid \mathbf{v}(D) \in \mathcal{C} \quad \text{and} \quad \mathbf{v}(D) \neq 0\}.$$

where *the Hamming weight of a polynomial* $v(D)$:

$$\text{wt}(v(D)) = \sum \text{wt}\,[\text{coefficients}].$$

To give an upper bound on the free distance of a convolutional code we can:

- Fix n, k, δ_e to obtain:

$$d_{\text{free}} \leq n(\mu + 1) - \ell + 1, \ \mu = \min\{m_1, \ldots, m_k\}$$

with ℓ the number of rows equal to μ.

Example

$$G(D) = \begin{bmatrix} 1+D & 1+D & 1 \\ D & 1 & 1+D \end{bmatrix} \begin{matrix} \longrightarrow m_1 = 1 \\ \longrightarrow m_2 = 1 \end{matrix} \longrightarrow \delta_e = 2, \delta_i = 2, m = 1.$$

$$d_{\text{free}} \leq 3(1+1) - 2 + 1 = 5.$$

- If $G(D)$ is *reduced*, i.e.,

$$\delta_e = \delta_i = \delta,$$

then

$$\mu \triangleq \min\{m_1, \ldots, m_k\} = \left\lfloor \frac{\delta}{k} \right\rfloor$$

and the number ℓ of rows equal to μ is

$$\ell = k - (\delta - \left\lfloor \frac{\delta}{k} \right\rfloor k).$$

The generalized Singleton bound

The inequality

$$d_{\text{free}} \leq n(\mu + 1) - \ell + 1,$$

becomes

$$d_{\text{free}}(\mathcal{C}) \leq (n - k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1.$$

It is called the generalized Singleton bound for an (n, k, δ) convolutional code.

- Fix n, k, m to obtain

$$d_{\text{free}} \leq n(m+1) - k + 1.$$

Example

The code generated by $G(D)$ below has $d_{\text{free}} = 8$.

$$G(D) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1+D & 1 & 1 & 1 & D & D & D & 0 \\ 1+D & 1+D & 0 & D & 1+D & 1 & 0 & 0 \\ 1+D & 0 & 1+D & D & 1 & D & 1 & 0 \end{bmatrix} /\mathbb{F}_2.$$

- This is maximum for $(8, 4)$ convolutional codes with encoders of parameter $\delta_e = 3$ for which the upper bound is

$$(8-4)\left(\left\lfloor \frac{3}{4} \right\rfloor + 1\right) + 3 + 1 = 8 = d_{\text{free}}.$$

- It is not maximal for $(8, 4)$ convolutional codes with encoders of memory $m = 1$, for which the upper bound is

$$16 - 4 + 1 = 13.$$

MDS-convolutional codes

Definition

A code $\mathcal{C}$ with $d_{\text{free}}(\mathcal{C})$ maximum among all codes of parameters (n, k, δ) is called a maximum distance separable convolutional code or MDS-convolutional code.

Convolutional Codes versus Block Codes

BLOCK CODES	CONVOLUTIONAL CODES
$[N, K]$, $\delta = 0$	(n, k, δ)
$\mathbf{u} \in \mathbb{F}^K$, $\mathbf{v} \in \mathbb{F}^N$	$\mathbf{u}(D) \in \mathbb{F}[D]^k$, $\mathbf{v}(D) \in \mathbb{F}[D]^n$
$G \in \mathbb{F}^{K \times N}$, $\mathbf{v} = \mathbf{u}G$	$G(D) \in \mathbb{F}[D]^{k \times n}$, $\mathbf{v}(D) = \mathbf{u}(D)G(D)$
$H \in \mathbb{F}^{(N-K) \times N}$, $H\mathbf{v}^T = \mathbf{0}^T$	$H(D) \in \mathbb{F}[D]^{(n-k) \times n}$, $H(D)\mathbf{v}(D)^T = \mathbf{0}^T$
$d_{min} \leq N - K + 1$	$d_{\text{free}}(\mathcal{C}) \leq (n-k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1$
MDS block codes	MDS convolutional codes

MDS-convolutional codes

How can we construct MDS-convolutional codes?

Row distances

For $j \geq 0$, let

$$G_j^r \triangleq \begin{bmatrix} G_0 & G_1 & \dots & G_m & & \\ & G_0 & G_1 & \dots & G_m & \\ & & \ddots & \ddots & & \ddots \\ & & & G_0 & G_1 & \dots & G_m \end{bmatrix}.$$

The jth row distance d_j^r is defined as: $d_j^r \triangleq \min_{u_{[0,j]} \neq 0} \mathrm{wt}\left(u_{[0,j]} \cdot G_j^r\right)$.

Then $d_{\text{free}} = \lim_{j \to \infty} d_j^r \leq \ldots \leq d_1^r \leq d_0^r$

A code attains the generalized Singleton bound if

$$d_{\text{free}} = \ldots = d_1^r = d_0^r = n(\mu + 1) - k + 1.$$

Equivalently, the block code generated by $\begin{bmatrix} G_0 & G_1 & \dots & G_m \end{bmatrix}$ must be MDS and any addition of a row $\begin{bmatrix} 0 & 0 & \dots 0 & G_0 & G_1 & \dots & G_m \end{bmatrix}$ does not decrease this distance.

Row distances

For $j \geq 0$, let

$$G_j^r \triangleq \begin{bmatrix} G_0 & G_1 & \dots & G_m & & \\ & G_0 & G_1 & \dots & G_m & \\ & & \ddots & \ddots & & \ddots \\ & & & G_0 & G_1 & \dots & G_m \end{bmatrix}.$$

The jth row distance d_j^r is defined as: $d_j^r \triangleq \min_{u_{[0,j]} \neq 0} \mathrm{wt}\left(u_{[0,j]} \cdot G_j^r\right)$.

Then $d_{\text{free}} = \lim_{j \to \infty} d_j^r \leq \dots \leq d_1^r \leq d_0^r$

A code attains the generalized Singleton bound if

$$d_{\text{free}} = \dots = d_1^r = d_0^r = n(\mu + 1) - k + 1.$$

Equivalently, the block code generated by $\begin{bmatrix} G_0 & G_1 & \dots & G_m \end{bmatrix}$ must be MDS and any addition of a row $\begin{bmatrix} 0 & 0 & \dots 0 & G_0 & G_1 & \dots & G_m \end{bmatrix}$ does not decrease this distance.

Row distances

For $j \geq 0$, let

$$G_j^r \triangleq \begin{bmatrix} G_0 & G_1 & \dots & G_m & & \\ & G_0 & G_1 & \dots & G_m & \\ & & \ddots & \ddots & & \ddots \\ & & & G_0 & G_1 & \dots & G_m \end{bmatrix}.$$

The jth row distance d_j^r is defined as: $d_j^r \triangleq \min_{u_{[0,j]} \neq 0} \mathrm{wt}\left(u_{[0,j]} \cdot G_j^r\right)$.

Then

$$d_{\text{free}} = \lim_{j \to \infty} d_j^r \leq \dots \leq d_1^r \leq d_0^r$$

A code attains the generalized Singleton bound if

$$d_{\text{free}} = \dots = d_1^r = d_0^r = n(\mu + 1) - k + 1.$$

Equivalently, the block code generated by $\begin{bmatrix} G_0 & G_1 & \dots & G_m \end{bmatrix}$ must be MDS and any addition of a row $\begin{bmatrix} 0 & 0 & \dots 0 & G_0 & G_1 & \dots & G_m \end{bmatrix}$ does not decrease this distance.

Row distances

For $j \geq 0$, let

$$G_j^r \triangleq \begin{bmatrix} G_0 & G_1 & \dots & G_m & & \\ & G_0 & G_1 & \dots & G_m & \\ & & \ddots & \ddots & & \ddots \\ & & & G_0 & G_1 & \dots & G_m \end{bmatrix}.$$

The jth row distance d_j^r is defined as: $d_j^r \triangleq \min_{u_{[0,j]} \neq 0} \mathrm{wt}\left(u_{[0,j]} \cdot G_j^r\right)$.

Then

$$d_{\text{free}} = \lim_{j \to \infty} d_j^r \leq \ldots \leq d_1^r \leq d_0^r$$

A code attains the generalized Singleton bound if

$$d_{\text{free}} = \ldots = d_1^r = d_0^r = n(\mu + 1) - k + 1.$$

Equivalently, the block code generated by $\begin{bmatrix} G_0 & G_1 & \dots & G_m \end{bmatrix}$ must be MDS and any addition of a row $\begin{bmatrix} 0 & 0 & \dots 0 & G_0 & G_1 & \dots & G_m \end{bmatrix}$ does not decrease this distance.

Constructions of MDS convolutional codes

Theorem (Rate 1/n, (Justesen))

Let

$$G(D) = \begin{bmatrix} g_1(D) & g_2(D) & \dots & g_n(D) \end{bmatrix},$$

such that

- $\gcd(g_1(D), \dots, g_n(D)) = 1$ *and*
- *each* $g_i(x)$ *generates a Reed-Solomon code* $C_i \subset \mathbb{F}[x]/(x^N - 1)$ *of common length* $N = ns$*, and minimum distance* $d_{\min} = \delta + 1$.

Then $G(D)$ *generates an MDS convolutional code with* $d_{\text{free}} = n(\delta + 1)$.

Example

Let

$$g_1(x) = (x-1)(x-\alpha), \quad g_2(x) = (x-\alpha^2)(x-\alpha^3).$$

They generate two $[7, 5, 3]$ Reed–Solomon (RS) codes over $GF(8)$. Then

$$G(D) = \begin{bmatrix} g_1(D) & g_2(D) \end{bmatrix}$$

generates an MDS convolutional code of maximum free distance $d_{\text{free}} = 6$.

Constructions of MDS convolutional codes

Theorem (Rate 1/n, (Justesen))

Let

$$G(D) = \begin{bmatrix} g_1(D) & g_2(D) & \dots & g_n(D) \end{bmatrix},$$

such that

- $\gcd(g_1(D), \dots, g_n(D)) = 1$ *and*
- *each* $g_i(x)$ *generates a Reed-Solomon code* $C_i \subset \mathbb{F}[x]/(x^N - 1)$ *of common length* $N = ns$*, and minimum distance* $d_{\min} = \delta + 1$.

Then $G(D)$ *generates an MDS convolutional code with* $d_{\text{free}} = n(\delta + 1)$.

Example

Let

$$g_1(x) = (x - 1)(x - \alpha), \quad g_2(x) = (x - \alpha^2)(x - \alpha^3).$$

They generate two $[7, 5, 3]$ Reed–Solomon (RS) codes over $GF(8)$. Then

$$G(D) = \begin{bmatrix} g_1(D) & g_2(D) \end{bmatrix}$$

generates an MDS convolutional code of maximum free distance $d_{\text{free}} = 6$.

Constructions of convolutional codes from block codes

Theorem (Rate 1/n, n= 2m, (Massey–Costello–Justesen))

Let

$$G(D) = \begin{bmatrix} g_1(D) & g_2(D) & \dots & g_n(D) \end{bmatrix},$$

where

$$g(x) \triangleq g_1(x^n) + x g_2(x^n) + \dots x^{n-1} g_n(x^n)$$

is a generator polynomial of a cyclic code over $GF(2^r)$ of odd length N and $g(x)h(x) = x^N + 1$. Then, the 2^r-ary convolutional code generated by $G(D)$ is non-catastrophic and has

$$d_{\text{free}} \geq \min\{d_g, 2d_h\}.$$

Construction of MDS convolutional codes

Taking $g(x)$ and $h(x)$ to generate dual RS codes with $2d_h = d_g = n(\delta + 1)$ yields $[n, 1, n(\delta + 1)]$ MDS convolutional codes.

Constructions of convolutional codes from block codes

Theorem (Rate 1/n, n= 2m, (Massey–Costello–Justesen))

Let

$$G(D) = \begin{bmatrix} g_1(D) & g_2(D) & \dots & g_n(D) \end{bmatrix},$$

where

$$g(x) \triangleq g_1(x^n) + x g_2(x^n) + \dots x^{n-1} g_n(x^n)$$

is a generator polynomial of a cyclic code over $GF(2^r)$ *of odd length* N *and* $g(x)h(x) = x^N + 1$. *Then, the* 2^r*-ary convolutional code generated by* $G(D)$ *is non-catastrophic and has*

$$d_{\text{free}} \geq \min\{d_g, 2d_h\}.$$

Construction of MDS convolutional codes

Taking $g(x)$ and $h(x)$ to generate dual RS codes with $2d_h = d_g = n(\delta + 1)$ yields $[n, 1, n(\delta + 1)]$ MDS convolutional codes.

Constructions of convolutional codes from block codes

Theorem (Rate 1/2n, n=2m, (Massey–Costello–Justesen))

Let

$$G(D) = \begin{bmatrix} g_1(D) & h_1(D) & g_2(D) & h_2(D) & \dots & g_n(D) & h_n(D) \end{bmatrix},$$

where

$$g(x) \triangleq g_1(x^n) + xg_2(x^n) + \dots x^{n-1} g_n(x^n)$$
$$h(x) \triangleq h_1(x^n) + xh_2(x^n) + \dots x^{n-1} h_n(x^n)$$

generate dual cyclic codes over $GF(2^r)$ of odd length N, $g(x)h(x) = x^N + 1$. Then, the 2^r-ary convolutional code generated by $G(D) = g(D^2) + Dh(D^2)$ is a rate-$1/4m$ non-catastrophic convolutional code with

$$d_{\text{free}} \geq \min\{d_g + d_h, 3d_g, 3d_h\}.$$

Construction of MDS convolutional codes

Taking $g(x)$ and $h(x)$ to generate dual RS codes with designed distances d_g, d_h satisfying $\min\{d_g + d_h, 3d_g, 3d_h\} \geq 2n(\delta + 1)$ yields $[2n, 1, 2n(\delta + 1)]$ MDS convolutional codes.

Constructions of convolutional codes from block codes

Theorem (Rate k/n, (Justesen))

More generally, if $g(x)$ generates a cyclic code having certain properties on the roots, then the code with the $k \times n$ matrix $G(D) = (g_{ij}(D))_{k\times n}$ satisfying

$$\sum_{j=1}^{n} g_{ij}(D^n)D^{j-1} = D^i g(D), \quad \text{for all } i \in [k] \iff G(D)_{k\times 1} = \begin{bmatrix} g(D) \\ Dg(D) \\ \vdots \\ D^{k-1}g(D) \end{bmatrix},$$

generates a convolutional code with

$$d_{\text{free}} \geq d_g.$$

Construction of MDS convolutional codes

Taking $g(x)$ to generate RS codes of designed distances $d_g \geq \text{GSB}$ yields MDS convolutional codes.

Strongly-MDS convolutional codes

We can also look at the truncation of the codewords and ask that the truncated codewords have large weights, no matter where they are truncated. If we truncate the sliding matrix G we get:

$$G_j^c = \begin{bmatrix} G_0 & G_1 & G_2 & \dots & G_j \\ & G_0 & G_1 & \dots & G_{j-1} \\ & & G_0 & \dots & G_{j-2} \\ & & & \ddots & \vdots \\ & & & & G_0 \end{bmatrix}.$$

We define the jth column distances d_j^c, for all $j \geq 0$, as

$$d_j^c = \min_{u_0 \neq 0} \mathrm{wt}\left(u_{[0,j]} \cdot G_j^c\right).$$

The column distances satisfy

$$d_0^c \leq d_1^c \leq \ldots \leq d_{\text{free}} = \lim_{j\to\infty} d_j^c = \lim_{l\to\infty} d_l^r \leq \ldots \leq d_1^r \leq d_0^r$$
$$\leq (n-k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1.$$

Strongly-MDS convolutional codes

Assume that

$$d_0^c \leq d_1^c \leq \ldots \leq d_{\text{free}} = \lim_{j\to\infty} d_j^c = \lim_{l\to\infty} d_l^r = \ldots = d_1^r = d_0^r$$
$$= (n-k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1.$$

How quickly can the sequence of column distances (d_j^c) reach d_{free}?

We have the following upper bound on the column distances:

$$d_j^c \leq (n-k)(j+1) + 1, \quad \forall j \geq 0.$$

Strongly-MDS convolutional codes

We have the following upper bound on the column distances:

$$d_j^c \leq (n-k)(j+1)+1, \quad \forall j \geq 0.$$

Let

$$t \triangleq \min\{j \geq 0 \mid d_j^c = d_{\text{free}}\}.$$

Then

$$t \geq \lfloor \tfrac{\delta}{k} \rfloor + \lceil \tfrac{\delta}{n-k} \rceil.$$

Strongly-MDS convolutional codes

Definition

An (n, k, δ)-code with column distances d_j^c, $j \geq 0$, is called strongly-MDS, if it is MDS and the free distance is attained as early as possible, i.e.,

$$d_t^c = (n-k)\left(\lfloor \tfrac{\delta}{k} \rfloor + 1\right) + \delta + 1 \quad \text{for} \quad t = \lfloor \tfrac{\delta}{k} \rfloor + \lceil \tfrac{\delta}{n-k} \rceil.$$

In some cases including rate $1/2$, strongly-MDS means that the column distances, and hence the truncated weights, are all maximal, i.e.,

$$d_j^c = (n-k)(j+1) + 1, \quad j \leq t.$$

Constructions of strongly-MDS Codes: rate 1/2

Theorem (Gluesing-Luerssen–Rosenthal–Smarandache)

Let

$$G(D) = \begin{bmatrix} a(D) & b(D) \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{\delta} a_i D^i & \sum_{i=0}^{\delta} b_i D^i \end{bmatrix}$$

and let

$$h(D) \triangleq \frac{a(D)}{b(D)} = \sum_{i=0}^{\infty} h_i D^i$$

be the Laurent expansion of $\frac{a(D)}{b(D)}$*. Then* $G(D)$ *generates a strongly-MDS convolutional code if the Toeplitz matrix*

$$T \triangleq \begin{bmatrix} h_0 & 0 & \dots & 0 \\ h_1 & h_0 & & \vdots \\ \vdots & & \ddots & \vdots \\ h_{2\delta} & \dots & \dots & h_0 \end{bmatrix}$$

is super-regular*, i.e., its submatrices with no zero row and no zero column are all nonsingular.*

Constructions of strongly-MDS Codes: rate 1/2

Example

The matrices

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 6 & 1 & 2 & 1 & 0 \\ 4 & 6 & 1 & 2 & 1 \end{bmatrix}$$

over $\mathbb{F}_3$, respectively $\mathbb{F}_7$ are super-regular.

If T is super-regular then the polynomials $a(D), b(D)$ of degree m are uniquely determined by the first $2\delta + 1$ coefficients

$$\frac{a(D)}{b(D)} = h_0 + \cdots + h_{2\delta} D^{2\delta} + \cdots$$

and the code $\mathcal{C}$ generated by $G(D)$ is strongly-MDS.

Example

Let

$$T = \begin{bmatrix} 1 & & & & \\ \beta & 1 & & & \\ \beta^3 & \beta & 1 & & \\ \beta & \beta^3 & \beta & 1 & \\ 1 & \beta & \beta^3 & \beta & 1 \end{bmatrix}$$

be a matrix over $\mathbb{F}_8$, with $\beta^3 + \beta + 1 = 0$. T is super-regular.
Then $1 + \beta D + \beta^3 D^2 + \beta D^3 + D^4$ defines uniquely the quotient

$$\frac{1 + \beta^4 D + \beta^5 D^2}{1 + \beta^2 D + \beta^5 D^2} = 1 + \beta D + \beta^3 D^2 + \beta D^3 + D^4 + \text{ higher powers } .$$

This gives a unique strongly-MDS $(2, 1, 2)$ code with $d_{\text{free}} = 6$:

$$G(D) = \begin{bmatrix} 1 + \beta^2 D + \beta^5 D^2 & 1 + \beta^4 D + \beta^5 D^2 \end{bmatrix}.$$

In fact, finding a class of strongly-MDS codes is equivalent to finding Toepliz super-regular matrices.

Super-regular matrices exist

Example

Let

$$X = \begin{bmatrix} 1 & & & & & \\ 1 & 1 & & & & \\ & 1 & 1 & & & \\ & & \ddots & \ddots & & \\ & & & 1 & 1 & \\ & & & & 1 & 1 \end{bmatrix}.$$

Then

$$A = X^{n-1} = \begin{bmatrix} 1 & & & & & \\ n & 1 & & & & \\ \binom{n}{2} & n & 1 & & & \\ \vdots & \vdots & \ddots & \ddots & & \\ \binom{n}{n-1} & \binom{n}{n-2} & \cdots & n & 1 & \\ 1 & \binom{n}{n-1} & \cdots & & n & 1 \end{bmatrix}$$

is *totally positive* over the reals and super-regular for prime fields, with the prime p_n not dividing any of the minors.

Maximum Distance Profile (MDP) convolutional codes

Definition

An (n, k, δ) convolutional code is a maximum distance profile (MDP) convolutional code if all its column distances are maximal. Equivalently,

$$d_L^c(\mathcal{C}) = (n-k)(L+1)+1,$$

where

$$L = \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lfloor \frac{\delta}{n-k} \right\rfloor = \begin{cases} t & \text{if} \quad (n-k) \mid \delta \\ t-1 & \text{otherwise} \end{cases}.$$

Recall that a strongly-MDS convolutional code has

$$t \triangleq \min\{j \geq 0 \mid d_j^c = d_{\text{free}}\} = \left\lfloor \tfrac{\delta}{k} \right\rfloor + \left\lceil \tfrac{\delta}{n-k} \right\rceil$$

Characterizing MDP codes

Truncated parity-check matrices

$$j = 0: \quad \mathcal{H}_0^c = \begin{bmatrix} H_0 \end{bmatrix}$$

$$j = 1: \quad \mathcal{H}_1^c = \begin{bmatrix} H_0 & \\ H_1 & H_0 \end{bmatrix}$$

$$\vdots$$

$$j = L: \quad \mathcal{H}_L^c = \begin{bmatrix} H_0 & & & \\ \vdots & \ddots & & \\ \vdots & & \ddots & \\ H_L & \dots & \dots & H_0 \end{bmatrix}$$

Characterization of MDP Codes

Theorem

The j-th column distance attains the maximum value

$$d_j^c = (n-k)(j+1)+1,$$

if and only if, every $(j+1)(n-k) \times (j+1)(n-k)$ full-size minor of $\mathcal{H}_j$ formed from the columns with indices $1 \leq i_1 < \cdots < i_{(j+1)(n-k)}$, where $i_{s(n-k)} \leq sn$ for $s = 1, \ldots, j$, is nonzero.

Equivalently, $H(D)$ represents an MDP code, if and only if, every $(L+1)(n-k) \times (L+1)(n-k)$ full-size minor of $\mathcal{H}_L$ formed from the columns with indices $1 \leq i_1 < \cdots < i_{(L+1)(n-k)}$, where $i_{s(n-k)} \leq sn$ for $s = 1, \ldots, L$, is nonzero.

Decoding over an erasure channel

Theorem

Let $\mathcal{C}$ be an (n, k, δ) convolutional code, and let j_0 be some fixed index. If in any sliding window of length $(j_0 + 1)n$ at most $d_{j_0}^c - 1$ erasures occur, then we can recover the whole sequence.

Consequently, for an (n, k, δ) MDP convolutional code, if in any sliding window of length $(L+1)n$ at most $(L+1)(n-k)$ erasures occur, then we can recover the whole sequence.

Decoding over an erasure channel with an MDP code

- Assume we have correctly received or recovered a sequence until instant t.
- We solve a full rank system, where $\star$ represent blocks where some erasures have occurred.

Idea of the process

$$\begin{bmatrix} H_\nu & H_{\nu-1} & \dots & H_{\nu-L} & \dots & H_0 & & & \\ & H_\nu & \dots & H_{\nu-L+1} & \dots & H_1 & H_0 & & \\ & & \ddots & & & & & \ddots & \\ & & & H_\nu & \dots & H_L & H_{L-1} & \dots & H_0 \end{bmatrix} \begin{bmatrix} \mathbf{v}_{t-\nu} \\ \vdots \\ \mathbf{v}_{t-1} \\ \star \\ \star \\ \vdots \\ \star \end{bmatrix} = 0$$

Remarks

The algorithm requires linear algebra; it has polynomial time complexity of order Cn^3, where C is a constant depending on the field size.

The bound is optimal: there exist patterns of $(L+2)(n-k)$ erasures in windows of length $(L+2)n$ that cannot be uniquely decoded.

Adaptative process: we do not need to take the largest L window we are allowed, we can choose smaller window sizes depending on the distribution of the erasures, since the MDP property holds as well for smaller sizes.

Example

Assume we have and MDS block code and an MDP convolutional code of comparable window length. Both are able to recover 100 erasures in windows of 200 symbols, so 50%.

[200,100] MDS block code

(2,1,50) MDP convolutional code

} 100 erasures / 200 symbols

Suppose we receive correctly until instant t, followed by:

$$\overbrace{\star\star\ldots\star\star}^{60} v_{61} v_{62} \ldots v_{140} \overbrace{\star\star\ldots\star\star}^{60} v_{201} v_{202} \ldots$$

In a 200-symbol window the block code cannot decode: $120 > 100$ erasures $\Rightarrow$ skip (and discard) the block.
The MDP convolutional code, can decode in a 120- symbol window to recover 60 erasures, then frame to the next window to recover the rest$\Rightarrow$ the MDP code can adapt and do better that the MDS block code.

The case of too many errors in a window frame

If there are too many erasures in any window:

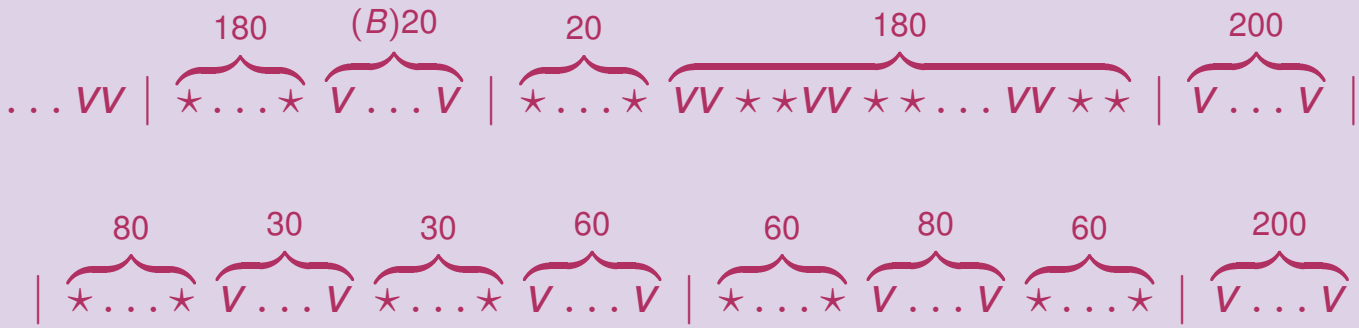
Suppose we receive correctly until instant t and then receive:

$$\begin{bmatrix} H_\nu & H_{\nu-1} & \dots & H_{\nu-j} & \dots & H_0 & & & \\ & H_\nu & \dots & H_{\nu-j+1} & \dots & H_1 & H_0 & & \\ & & \ddots & & & & & \ddots & \\ & & & H_\nu & \dots & H_j & H_{j-1} & \dots & H_0 \end{bmatrix} \begin{bmatrix} \mathbf{v}_{t-\nu} \\ \vdots \\ \mathbf{v}_t \\ \star \\ \star \\ \vdots \\ \star \end{bmatrix} = 0.$$

- Move forward until a clean space is found to clean the memory.
- Feed back decoding without error propagation (symbols we cannot correctly and uniquely recover are not used).
- We would like to be able to use a smaller amount of clean symbols.

The case of too many errors in a window frame

How about the situation below where there exist spaces clean enough but the erasures are too concentrated in the beginning of the block to uniquely solve the system?



How can we do better?

- If we could move in the other direction along the sequence, the erasures are less concentrated and it would be easier to recover.
- We would like MDP behavior in the backwards process as well: to control the column distances of the reverse code.

Backwards process

Proposition:

Let $\mathcal{C}$ be an (n, k, δ)-code with minimal generator matrix $G(D)$. Let $\overline{G}(D)$ be the matrix obtained by replacing each entry $g_{ij}(D)$ of $G(D)$ by $\overline{g_{ij}}(D) := D^{\delta_j} g_{ij}(D^{-1})$, where δ_j is the j-th column degree of $G(D)$. Then, $\overline{G}(D)$ is a minimal generator matrix of an (n, k, δ)-code $\overline{\mathcal{C}}$, and

$$\mathbf{v}_0 + \mathbf{v}_1 D + \cdots + \mathbf{v}_{s-1} D^{s-1} + \mathbf{v}_s D^s \in \mathcal{C}$$

if and only if

$$\mathbf{v}_s + \mathbf{v}_{s-1} D^+ \cdots + \mathbf{v}_1 D^{s-1} + \mathbf{v}_0 D^s \in \overline{\mathcal{C}}.$$

$\overline{\mathcal{C}}$ is the *reverse code* of $\mathcal{C}$.

Backwards process

$$d_{\text{free}}(\mathcal{C}) = d_{\text{free}}(\overline{\mathcal{C}})$$

$$\mathbf{v}(D) = \sum_{i=0}^{s} \mathbf{v}_i D^i \qquad \overline{\mathbf{v}}(D) = \sum_{i=0}^{s} \mathbf{v}_{s-i} D^i$$

$$d_j^c(\mathcal{C}) = \min\left\{ \sum_{i=0}^{j} \text{wt}(\mathbf{v}_i) \mid \mathbf{v}(D) \in \mathcal{C} \quad \text{and} \quad \mathbf{v}_0 \neq 0 \right\}$$

$$d_j^c(\overline{\mathcal{C}}) = \min\left\{ \sum_{i=0}^{s-j+1} \text{wt}(\mathbf{v}_{s-i}) \mid \mathbf{v}(D) \in \mathcal{C} \quad \text{and} \quad \mathbf{v}_s \neq 0 \right\}.$$

Reverse MDP convolutional codes

Definition

Let $\mathcal{C}$ be an MDP (n, k, δ) convolutional code. We say that $\mathcal{C}$ is a reverse-MDP convolutional code if the reverse code $\overline{\mathcal{C}}$ is an MDP code as well.

Theorem

Let k, n and δ be positive integers. An (n, k, δ) reverse-MDP convolutional code exists over a sufficiently large field.

Reverse MDP convolutional codes

Example

Let $\mathcal{C}$ be the $(2, 1, 1)$ convolutional code over $\mathbb{F}_{2^2}$ given by the parity-check matrix

$$H(D) = \begin{bmatrix} 1 + \alpha^2 D & 1 + D \end{bmatrix}$$

where α satisfies $\alpha^2 + \alpha + 1 = 0$.
$\mathcal{C}$ is a reverse-MDP code.
The reverse code $\overline{\mathcal{C}}$ is defined by the matrix

$$\overline{H}(D) = \begin{bmatrix} D + \alpha^2 & D + 1 \end{bmatrix}.$$

Reverse MDP construction

Definition

Recall that a lower triangular Toepliz matrix A

$$A = \begin{bmatrix} a_1 & 0 & \dots & 0 \\ a_2 & a_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ a_n & \dots & a_2 & a_1 \end{bmatrix} \in \mathbb{F}^{n \times n}$$

is superregular if $A^{i_1,\dots,i_r}_{j_1,\dots,j_r}$ is nonsingular for all $1 \leq r \leq n$ and all indices $1 \leq i_1 < \dots < i_r \leq n$, $1 \leq j_1 < \dots < j_r \leq n$ which satisfy $j_s \leq i_s$ for $s = 1, \dots, r$.

Reverse MDP construction

Definition

We say a superregular matrix A is reverse-superregular if the matrix

$$A_{rev} = \begin{bmatrix} a_n & 0 & \dots & 0 \\ a_{n-1} & a_n & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ a_0 & \dots & a_{n-1} & a_n \end{bmatrix}$$

is as well superregular.

Reverse MDP construction

Example

Let $\mathbb{F} = \mathbb{F}_8$ with $\alpha^3 + \alpha^2 + 1 = 0$. The matrices

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha^4 & 1 & 0 & 0 \\ \alpha^6 & \alpha^4 & 1 & 0 \\ \alpha^3 & \alpha^6 & \alpha^4 & 1 \end{bmatrix} \text{ and } C_{rev} = \begin{bmatrix} \alpha^3 & 0 & 0 & 0 \\ \alpha^6 & \alpha^3 & 0 & 0 \\ \alpha^4 & \alpha^6 & \alpha^3 & 0 \\ 1 & \alpha^4 & \alpha^6 & \alpha^3 \end{bmatrix}$$

are superregular. Therefore, C is a reverse-superregular matrix.

Reverse MDP construction

Example

We construct a $(3, 2, 1)$ reverse-MDP convolutional code $\mathbb{F}_{32}$ with $\mu^5 + \mu^2 + 1 = 0$.

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ \mu^{15} & 1 & 0 & 0 & 0 & 0 \\ \mu^{21} & \mu^{15} & 1 & 0 & 0 & 0 \\ \mu^{23} & \mu^{21} & \mu^{15} & 1 & 0 & 0 \\ \mu^{21} & \mu^{23} & \mu^{21} & \mu^{15} & 1 & 0 \\ \mu^{10} & \mu^{21} & \mu^{23} & \mu^{21} & \mu^{15} & 1 \end{bmatrix}.$$

Applying the theorem we obtain the matrix

$$\mathcal{H}_L = \begin{bmatrix} H_0 & O \\ H_1 & H_0 \end{bmatrix} = \begin{bmatrix} \mu^{21} & \mu^{15} & 1 & 0 & 0 & 0 \\ \mu^{10} & \mu^{21} & \mu^{23} & \mu^{21} & \mu^{15} & 1 \end{bmatrix}$$

Reverse MDP construction

Example

The parity-check matrix of $\mathcal{C}$ is

$$H(D) = \left[\begin{array}{ccc} \mu^{21} + \mu^{10} D & \mu^{15} + \mu^{21} D & 1 + \mu^{23} D \end{array} \right].$$

The parity-check matrix of $\overline{\mathcal{C}}$ is

$$\overline{H}(D) = \left[\begin{array}{ccc} \mu^{10} + \mu^{21} D & \mu^{21} + \mu^{15} D & \mu^{23} + D \end{array} \right].$$

Reverse MDP construction

Example

We construct the generator matrix of a $(3, 1, 1)$ over $\mathbb{F}_{32}$ where $\gamma^5 + \gamma^4 + \gamma^3 + \gamma^2 + 1 = 0$.

$$S = \begin{bmatrix} 1 & \gamma^{19} & \gamma^{16} & \gamma^{20} & \gamma^5 & \gamma^{16} \\ 0 & 1 & \gamma^{19} & \gamma^{16} & \gamma^{20} & \gamma^5 \\ 0 & 0 & 1 & \gamma^{19} & \gamma^{16} & \gamma^{20} \\ 0 & 0 & 0 & 1 & \gamma^{19} & \gamma^{16} \\ 0 & 0 & 0 & 0 & 1 & \gamma^{19} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

we obtain the matrix

$$\mathcal{G}_L = \begin{bmatrix} G_0 & G_1 \\ O & G_0 \end{bmatrix} = \begin{bmatrix} \gamma^{16} & \gamma^{19} & 1 & 0 & 0 & 0 \\ \gamma^{16} & \gamma^5 & \gamma^{20} & \gamma^{16} & \gamma^{19} & 1 \end{bmatrix}$$

Reverse MDP construction

Example

The generator matrices of $\mathcal{C}$ and $\overline{\mathcal{C}}$ are

$$G(D) = \begin{bmatrix} \gamma^{16} + \gamma^{16} D & \gamma^{19} + \gamma^{5} D & 1 + \gamma^{20} D \end{bmatrix}$$
$$\overline{G}(D) = \begin{bmatrix} \gamma^{16} + \gamma^{16} D & \gamma^{5} + \gamma^{19} D & \gamma^{20} + D \end{bmatrix}.$$

Conclusions

- We presented a few classes of convolutional codes that can outperform the MDS block codes on the erasure channel.
- We showed how to construct MDS convolutional codes and how to improve the performances of these codes by imposing extra properties. An optimal distance profile gave strongly-MDS and MDP codes.
- We also introduced reversed-MDS for dealing with window frames in which the errors are concentrated at the beginning of the frame.

Non-binary Convolutional Codes with Good Distance Properties

Roxana Smarandache

University of Notre Dame
Notre Dame, IN 46566

Basen on joint work with:
Joachim Rosenthal, Heide Gluesing-Luerssen,
Ryan Hutchinson, Virtu Tomás, Diego Napp

Banff, October 12, 2015