A state space approach to periodic convolutional codes

Diego Napp, Ricardo Pereira and Paula Rocha

CIDMA CENTRO DE I&D EM MATEMÁTICA E APLICAÇÕES
CENTER FOR R&D IN MATHEMATICS AND APPLICATIONS

SYSTEC RESEARCH CENTER FOR SYSTEMS & TECHNOLOGIES

Universidade de Aveiro

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Castle Meeting 2017 - Estonia

Block codes vs convolutional codes

Block codes vs convolutional codes

$$\dots u_2,\ u_1,\ u_0 \xrightarrow{G} \dots v_2 = u_2 G,\ v_1 = u_1 G,\ v_0 = u_0 G$$

represented in a polynomial fashion

$$\dots + u_2 D^2 + u_1 D + u_0 \xrightarrow{G} \dots + \underbrace{u_2 G}_{v_2} D^2 + \underbrace{u_1 G}_{v_1} D + \underbrace{u_0 G}_{v_0}$$

substitute G **by** $G(D) = G_0 + G_1 D + \dots + G_s D^s$**?**

Block codes vs convolutional codes

$$\dots u_2,\ u_1,\ u_0 \xrightarrow{G} \dots v_2 = u_2G,\ v_1 = u_1G,\ v_0 = u_0G$$

represented in a polynomial fashion

$$\dots + u_2D^2 + u_1D + u_0 \xrightarrow{G} \dots + \underbrace{u_2G}_{v_2}D^2 + \underbrace{u_1G}_{v_1}D + \underbrace{u_0G}_{v_0}$$

substitute G by $G(D) = G_0 + G_1D + \dots + G_sD^s$?

$$\dots u_2D^2 + u_1D + u_0 \xrightarrow{G(D)} \dots \underbrace{(u_2G_0 + u_1G_1 + u_0G_2)}_{v_2}D^2 + \underbrace{(u_1G_0 + u_0G_1)}_{v_1}D + \underbrace{u_0G_0}_{v_0}$$

Block codes vs convolutional codes

$$\dots u_2,\ u_1,\ u_0 \xrightarrow{G} \dots v_2 = u_2 G,\ v_1 = u_1 G,\ v_0 = u_0 G$$

represented in a polynomial fashion

$$\cdots + u_2 D^2 + u_1 D + u_0 \xrightarrow{G} \cdots + \underbrace{u_2 G}_{v_2} D^2 + \underbrace{u_1 G}_{v_1} D + \underbrace{u_0 G}_{v_0}$$

substitute G by $G(D) = G_0 + G_1 D + \cdots + G_s D^s$?

$$\dots u_2 D^2 + u_1 D + u_0 \xrightarrow{G(D)} \dots \underbrace{(u_2 G_0 + u_1 G_1 + u_0 G_2)}_{v_2} D^2 + \underbrace{(u_1 G_0 + u_0 G_1)}_{v_1} D + \underbrace{u_0 G_0}_{v_0}$$

Block codes: $\mathcal{C} = \{uG\} = \operatorname{Im}_{\mathbb{F}} G \sim \{u(D)G\} = \operatorname{Im}_{\mathbb{F}} G(D)$

Convolutional codes: $\mathcal{C} = \{u(D)G(D)\} = \operatorname{Im}_{\mathbb{F}[D]} G(D)$

They are $\mathbb{F}[D]$-modules of $\mathbb{F}^n[D]$

Definition: **A (time-invariant) convolutional code $\mathcal{C}$ of rate k/n is a set of finite support sequences described as**

$$\mathcal{C} = \left\{ v : v(\ell) = \left(G\left(\sigma^{-1}\right) u \right)(\ell);\ \ell \in \mathbb{N}_0, u \in \left[\left(\mathbb{F}^k\right)^{\mathbb{N}_0} \right]_{FS} \right\},$$

Convolutional codes as a Linear System

Definition: A (time-invariant) convolutional code $\mathcal{C}$ of rate k/n is a set of finite support sequences described as

$$\mathcal{C} = \left\{ v : v(\ell) = \left(G\left(\sigma^{-1}\right) u \right)(\ell);\ \ell \in \mathbb{N}_0, u \in \left[\left(\mathbb{F}^k\right)^{\mathbb{N}_0} \right]_{FS} \right\},$$

- With σ, shift operator $\left(\sigma^{-1} u\right)(\ell) = u(\ell - 1)$
- $G(z) \in \mathbb{F}^{n\times k}[z]$: encoder (full column rank $n \times k$ polynomial matrix over $\mathbb{F}$)

Convolutional codes as a Linear System

Definition: A (time-invariant) convolutional code $\mathcal{C}$ of rate k/n is a set of finite support sequences described as

$$\mathcal{C} = \left\{ v : v(\ell) = \left(G\left(\sigma^{-1}\right) u \right)(\ell);\ \ell \in \mathbb{N}_0, u \in \left[\left(\mathbb{F}^k\right)^{\mathbb{N}_0} \right]_{FS} \right\},$$

- With σ, shift operator $\left(\sigma^{-1} u\right)(\ell) = u(\ell - 1)$
- $G(z) \in \mathbb{F}^{n \times k}[z]$: encoder (full column rank $n \times k$ polynomial matrix over $\mathbb{F}$)
- u: information sequence in $\mathbb{F}^k$
- v: codeword

Convolutional codes as a Linear System

Definition: A (time-invariant) convolutional code $\mathcal{C}$ of rate k/n is a set of finite support sequences described as

$$\mathcal{C} = \left\{ v : v(\ell) = \left(G\left(\sigma^{-1}\right) u \right)(\ell); \, \ell \in \mathbb{N}_0, u \in \left[\left(\mathbb{F}^k\right)^{\mathbb{N}_0} \right]_{FS} \right\},$$

- With σ, shift operator $\left(\sigma^{-1} u\right)(\ell) = u(\ell - 1)$
- $G(z) \in \mathbb{F}^{n \times k}[z]$: encoder (full column rank $n \times k$ polynomial matrix over $\mathbb{F}$)
- u: information sequence in $\mathbb{F}^k$
- v: codeword
- The subindex FS affecting a set of sequences indicates that only its finite support elements are considered

An encoder G is called

basic $\rightsquigarrow$ **has a polynomial right inverse**

minimal $\rightsquigarrow$ **the sum of its column degrees attains the minimal possible value**

Convolutional codes

An encoder G is called

basic ⇝ has a polynomial right inverse

minimal ⇝ the sum of its column degrees attains the minimal possible value

Let G be a minimal encoder of the c.c. $\mathcal{C}$. Then $\mathcal{C}$ has:

degree δ ⇝ sum of the column degrees of G

memory m ⇝ maximum of the column degrees of G

Convolutional codes

An encoder G is called

basic $\rightsquigarrow$ has a polynomial right inverse

minimal $\rightsquigarrow$ the sum of its column degrees attains the minimal possible value

Let G be a minimal encoder of the c.c. $\mathcal{C}$. Then $\mathcal{C}$ has:

degree δ $\rightsquigarrow$ sum of the column degrees of G

memory m $\rightsquigarrow$ maximum of the column degrees of G

(n, k, δ, m) code $\rightsquigarrow$ a code of rate k/n, degree δ and memory m

Free distance

Definition: The **free distance** of a convolutional code $\mathcal{C}$ is given by

$$d_{free}(\mathcal{C}) = \min\left\{\sum_{\ell=0}^{\infty} \mathrm{wt}\big(v(\ell)\big) : v \in \mathcal{C} \setminus \{0\}\right\}$$

where wt denotes the **Hamming weight** (number of nonzero components)

Free distance

Definition: The **free distance** of a convolutional code $\mathcal{C}$ is given by

$$d_{free}(\mathcal{C}) = \min\left\{\sum_{\ell=0}^{\infty} \mathrm{wt}\big(v(\ell)\big) : v \in \mathcal{C} \setminus \{0\}\right\}$$

where wt denotes the **Hamming weight** (number of nonzero components)

Griesmer bound

If $\mathcal{C}$ is an (n, k, δ, m) c.c. over $\mathbb{F}_q$ and $\hat{\mathbb{N}} = \begin{cases} \mathbb{N} & \text{if } km = \delta \\ \mathbb{N}_0 & \text{if } km > \delta \end{cases}$, then

$$d_{free}(\mathcal{C}) \leq \max\left\{d' : \sum_{j=0}^{k(m+i)-\delta-1} \left\lceil \frac{d'}{q^j} \right\rceil \leq n(m+i),\ \forall i \in \hat{\mathbb{N}}\right\}$$

State space representations

A convolutional encoder is also a linear device which maps

$$u(0), u(1), \cdots \longrightarrow v(0), v(1), \ldots$$

In this sense it is the same as block encoders. The difference is that the convolutional encoder has an internal "storage vector" or "memory".

State space representations

A convolutional encoder is also a linear device which maps

$$u(0), u(1), \cdots \longrightarrow v(0), v(1), \ldots$$

In this sense it is the same as block encoders. The difference is that the convolutional encoder has an internal "storage vector" or "memory".

$v(i)$ does not depend only on $u(i)$ but also on the storage vector $x(i)$

$$\begin{aligned} x(i+1) &= Ax(i) + Bu(i) \\ v(i) &= Cx(i) + Eu(i) \end{aligned}$$

$A \in \mathbb{F}^{\delta \times \delta}, B \in \mathbb{F}^{\delta \times k}, C \in \mathbb{F}^{n \times \delta}, E \in \mathbb{F}^{\delta \times k}$.

Example

The encoder

$$G(\sigma^{-1}) = \begin{pmatrix} \sigma^{-2} + 1 \\ \sigma^{-2} + \sigma^{-1} + 1 \end{pmatrix}$$

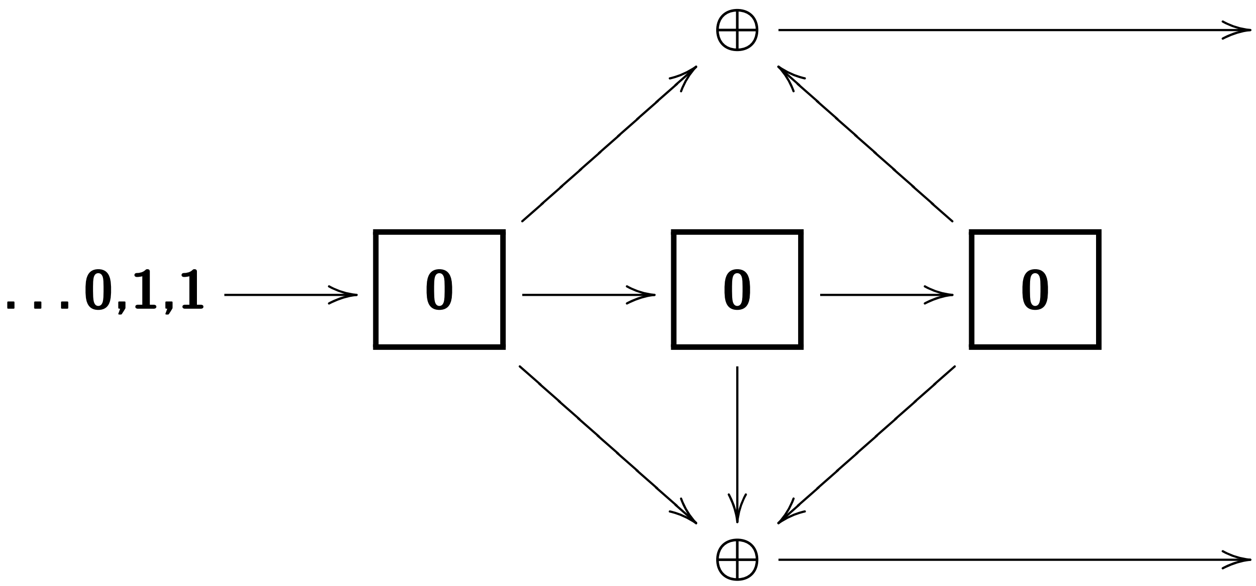
has the following implementation

Example

The encoder

$$G(\sigma^{-1}) = \begin{pmatrix} \sigma^{-2} + 1 \\ \sigma^{-2} + \sigma^{-1} + 1 \end{pmatrix}$$

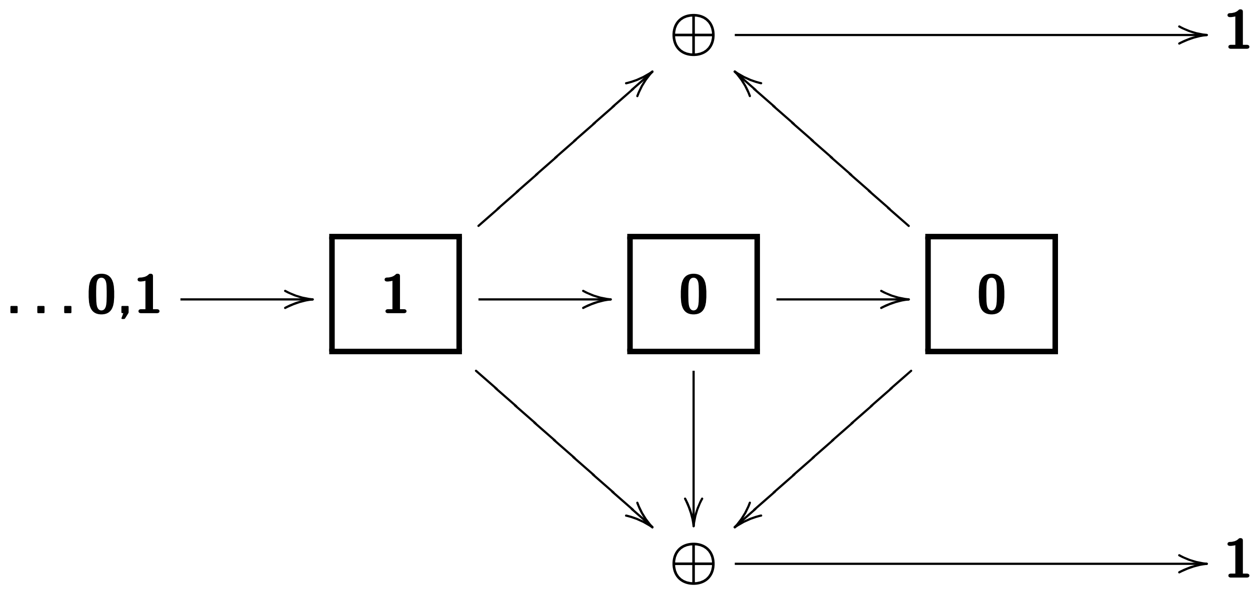
has the following implementation



The encoder

$$G(\sigma^{-1}) = \begin{pmatrix} \sigma^{-2}+1 \\ \sigma^{-2}+\sigma^{-1}+1 \end{pmatrix}$$

has the following implementation

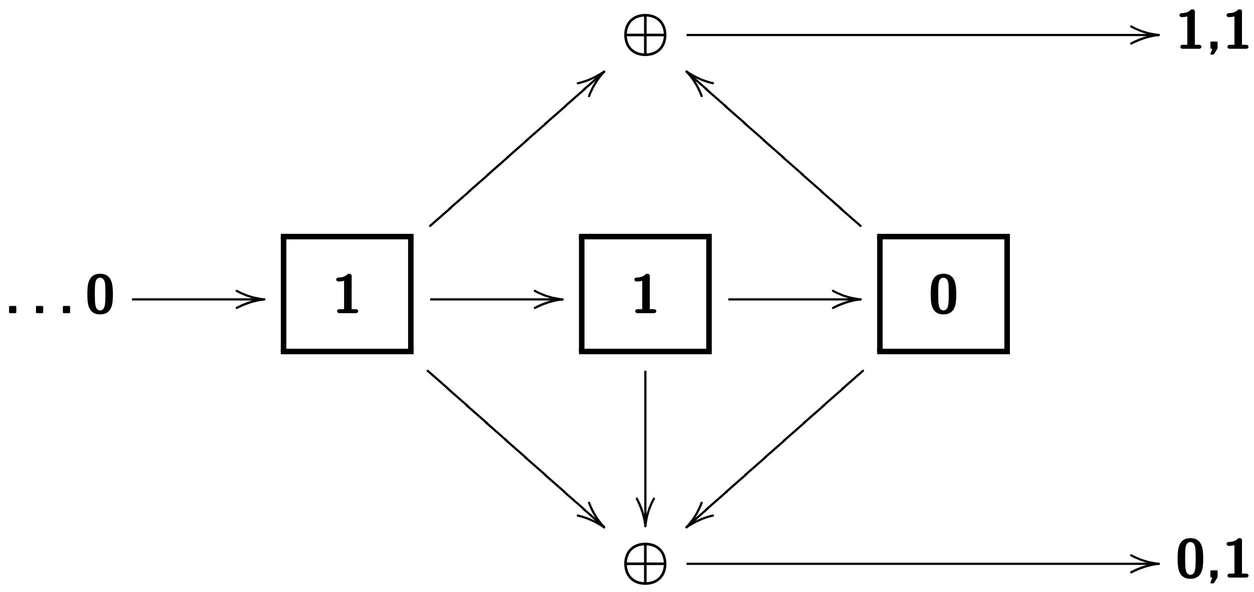


Example

The encoder

$$G(\sigma^{-1}) = \begin{pmatrix} \sigma^{-2} + 1 \\ \sigma^{-2} + \sigma^{-1} + 1 \end{pmatrix}$$

has the following implementation

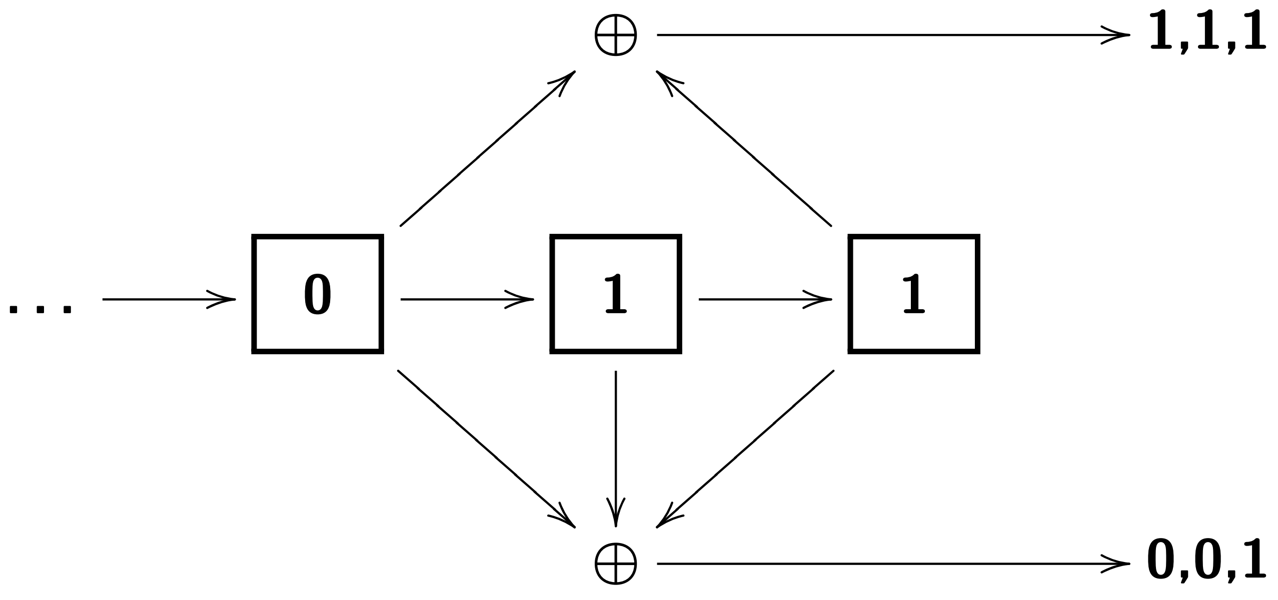


Example

The encoder

$$G(\sigma^{-1}) = \begin{pmatrix} \sigma^{-2} + 1 \\ \sigma^{-2} + \sigma^{-1} + 1 \end{pmatrix}$$

has the following implementation



Example (continuation)

Let the convolutional code be given by matrices

$$\begin{bmatrix} x_1(i+1) \\ x_2(i+1) \end{bmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{bmatrix} x_1(i) \\ x_2(i) \end{bmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u(i)$$

$$\begin{bmatrix} v_1(i) \\ v_2(i) \end{bmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{bmatrix} x_1(i) \\ x_2(i) \end{bmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u(i)$$

Let the convolutional code be given by matrices

$$\begin{bmatrix} x_1(i+1) \\ x_2(i+1) \end{bmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{bmatrix} x_1(i) \\ x_2(i) \end{bmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u(i)$$

$$\begin{bmatrix} v_1(i) \\ v_2(i) \end{bmatrix} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{bmatrix} x_1(i) \\ x_2(i) \end{bmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u(i)$$

We can compute an encoder

$$G(\sigma^{-1}) = E + B(\sigma I_m - A)^{-1}C = \begin{pmatrix} 1+\sigma^{-1}+\sigma^{-2} & 1+\sigma^{-2} \end{pmatrix}^T$$

(A, B, C, D) **is called a realization of** $\mathcal{C}$

Definition: A conv. code $\mathcal{C}$ with P-periodic encoders is described as

$$\mathcal{C} = \Big\{ v : v(P\ell + t) = \left(G^t \left(\sigma^{-1} \right) u \right) (P\ell + t); \quad t = 0, \ldots, P-1;\ \ell \in \mathbb{N}_0;\ u \in \left[\left(\mathbb{F}^k \right)^{\mathbb{N}_0} \right]_{FS} \Big\}$$

where each $G^t(z)$ is an $n \times k$ time-invariant (basic) encoder

Definition: A conv. code $\mathcal{C}$ with P-periodic encoders is described as

$$\mathcal{C} = \{ v : v(P\ell + t) = \left(G^t\left(\sigma^{-1}\right) u\right)(P\ell + t); \quad t = 0, \ldots, P-1; \ \ell \in \mathbb{N}_0; \ u \in \left[\left(\mathbb{F}^k\right)^{\mathbb{N}_0}\right]_{FS} \}$$

where each $G^t(z)$ is an $n \times k$ time-invariant (basic) encoder

• Since Costello showed that they can have larger distance than time-invariant ones, some particular constructions were found...but no general theory or construction is known.

In this talk we only consider conv. codes with 2-periodic encoders

Example of 2-periodic convolutional code

$$\begin{cases} v(2l) = (G^0(\sigma^{-1})\,u)(2\ell) \\ v(2l+1) = (G^1(\sigma^{-1})\,u)(2\ell+1) \end{cases}$$

with

$$G^0(z) = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \end{bmatrix} \text{ and } G^1(z) = \begin{bmatrix} 1-z & z^3 - z \\ z^2 & z - z^2 \\ z^4 + z & 1 \end{bmatrix}$$

Example of 2-periodic convolutional code

$$\begin{cases} v(2l) = \left(G^0\left(\sigma^{-1}\right)u\right)(2\ell) \\ v(2l+1) = \left(G^1\left(\sigma^{-1}\right)u\right)(2\ell+1) \end{cases}$$

with

$$G^0(z) = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \end{bmatrix} \quad \text{and} \quad G^1(z) = \begin{bmatrix} 1 - z & z^3 - z \\ z^2 & z - z^2 \\ z^4 + z & 1 \end{bmatrix}$$

$$\begin{cases} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}(2\ell) = \begin{bmatrix} u_1(2\ell-2) - u_1(2\ell-1) + u_2(2\ell) \\ u_1(2\ell-3) + u_2(2\ell-1) \\ u_1(2\ell-1) + u_1(2\ell) + u_2(2\ell-2) \end{bmatrix} \\ \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}(2\ell+1) = \begin{bmatrix} u_1(2\ell+1) - u_1(2\ell) + u_2(2\ell-2) - u_2(2\ell) \\ u_1(2\ell-1) + u_2(2\ell) - u_2(2\ell-1) \\ u_1(2\ell-3) + u_1(2\ell) + u_2(2\ell+1) \end{bmatrix} \end{cases}$$

Lifted code

Let $\mathcal{C} = \left\{ v : v(2\ell + t) = (G^t \left(\sigma^{-1}\right) u)(2\ell + t);\ t = 0, 1;\ \ell = 0, 1, \ldots \right\}$

Lifted code

Let $\mathcal{C} = \left\{v : v(2\ell + t) = (G^t \left(\sigma^{-1}\right) u)(2\ell + t);\; t = 0, 1;\; \ell = 0, 1, \ldots\right\}$

Consider the linear map

$$L_2 : (\mathbb{F}^n)^{\mathbb{N}_0} \to \left(\mathbb{F}^{2n}\right)^{\mathbb{N}_0}$$

defined by

$$(L_2 v)(\ell) = \begin{bmatrix} v(2\ell) \\ v(2\ell + 1) \end{bmatrix}$$

Lifted code

Let $\mathcal{C} = \{v : v(2\ell + t) = (G^t(\sigma^{-1})\, u)(2\ell + t);\ t = 0, 1;\ \ell = 0, 1, \ldots\}$

Consider the linear map

$$L_2 : (\mathbb{F}^n)^{\mathbb{N}_0} \to (\mathbb{F}^{2n})^{\mathbb{N}_0}$$

defined by

$$(L_2 v)(\ell) = \begin{bmatrix} v(2\ell) \\ v(2\ell + 1) \end{bmatrix}$$

The **"lifted" version of $\mathcal{C}$** is defined as

$$\mathcal{C}^L = \left\{\widetilde{v} \in (\mathbb{F}^{2n})^{\mathbb{N}_0} : \widetilde{v} = L_2 v,\ v \in \mathcal{C}\right\}$$

Lifted code

Since $$(G^t\left(\sigma^{-1}\right)u)(2\ell+t)=\left(\left(\sigma^t G^t\left(\sigma^{-1}\right)\right)u\right)(2\ell),$$

Lifted code

Since $$\left(G^t\left(\sigma^{-1}\right)u\right)(2\ell+t)=\left(\left(\sigma^t G^t\left(\sigma^{-1}\right)\right)u\right)(2\ell),$$

then

$$v(2\ell+t)=\left(G^t\left(\sigma^{-1}\right)u\right)(2\ell+t)$$

$$\Leftrightarrow \quad \left(\begin{bmatrix} I_2 \\ \sigma I_2 \end{bmatrix} v\right)(2\ell)=\left(\begin{bmatrix} G^0\left(\sigma^{-1}\right) \\ \sigma G^1\left(\sigma^{-1}\right) \end{bmatrix} u\right)(2\ell)$$

Lifted code

Since

$$(G^t(\sigma^{-1})u)(2\ell+t) = ((\sigma^t G^t(\sigma^{-1}))u)(2\ell),$$

then

$$v(2\ell+t) = (G^t(\sigma^{-1})u)(2\ell+t)$$

$$\Leftrightarrow \quad \left(\begin{bmatrix} I_2 \\ \sigma I_2 \end{bmatrix} v\right)(2\ell) = \left(\begin{bmatrix} G^0(\sigma^{-1}) \\ \sigma G^1(\sigma^{-1}) \end{bmatrix} u\right)(2\ell)$$

Decomposing

$$\begin{bmatrix} G^0(\sigma^{-1}) \\ \sigma G^1(\sigma^{-1}) \end{bmatrix} = G^L(\sigma^{-2}) \begin{bmatrix} I_2 \\ \sigma I_2 \end{bmatrix}$$

Lifted code

Since $$(G^t(\sigma^{-1})u)(2\ell+t) = ((\sigma^t G^t(\sigma^{-1}))u)(2\ell),$$

then

$$v(2\ell+t) = (G^t(\sigma^{-1})u)(2\ell+t)$$

$$\Leftrightarrow \quad \left(\begin{bmatrix} I_2 \\ \sigma I_2 \end{bmatrix} v\right)(2\ell) = \left(\begin{bmatrix} G^0(\sigma^{-1}) \\ \sigma G^1(\sigma^{-1}) \end{bmatrix} u\right)(2\ell)$$

Decomposing

$$\begin{bmatrix} G^0(\sigma^{-1}) \\ \sigma G^1(\sigma^{-1}) \end{bmatrix} = G^L(\sigma^{-2}) \begin{bmatrix} I_2 \\ \sigma I_2 \end{bmatrix}$$

the lifted code can be represented as

$$\mathcal{C}^L = \left\{ \widetilde{v} : \widetilde{v}(\ell) = (G^L(\sigma^{-1})\widetilde{u})(\ell),\ \ell \in \mathbb{N}_0,\ \widetilde{u} \in \left[\left(\mathbb{F}^{2k}\right)^{\mathbb{N}_0}\right]_{FS} \right\},$$

where $\widetilde{v} = L_2 v$ **and** $\widetilde{u} = L_2 u$

Example

Let $G^0(z) = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z + 1 & z^2 \end{bmatrix}$ and $G^1(z) = \begin{bmatrix} 1 - z & z^3 - z \\ z^2 & z - z^2 \\ z^4 + z & 1 \end{bmatrix}$

Example

Let $G^0(z) = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \end{bmatrix}$ and $G^1(z) = \begin{bmatrix} 1 - z & z^3 - z \\ z^2 & z - z^2 \\ z^4 + z & 1 \end{bmatrix}$

$$\begin{bmatrix} G^0(z) \\ z^{-1}G^1(z) \end{bmatrix} = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \\ z^{-1} - 1 & z^2 - 1 \\ z & 1 - z \\ z^3 + 1 & z^{-1} \end{bmatrix}$$

Example

Let $G^0(z) = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \end{bmatrix}$ and $G^1(z) = \begin{bmatrix} 1-z & z^3 - z \\ z^2 & z - z^2 \\ z^4 + z & 1 \end{bmatrix}$

$$\begin{bmatrix} G^0(z) \\ z^{-1}G^1(z) \end{bmatrix} = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \\ z^{-1} - 1 & z^2 - 1 \\ z & 1 - z \\ z^3 + 1 & z^{-1} \end{bmatrix} = \begin{bmatrix} z^2 & 1 \\ 0 & 0 \\ 1 & z^2 \\ -1 & z^2 - 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} -z^2 & 0 \\ z^4 & z^2 \\ z^2 & 0 \\ 1 & 0 \\ z^2 & -z^2 \\ z^4 & 1 \end{bmatrix} z^{-1}$$

Example

Let $G^0(z) = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \end{bmatrix}$ and $G^1(z) = \begin{bmatrix} 1-z & z^3 - z \\ z^2 & z - z^2 \\ z^4 + z & 1 \end{bmatrix}$

$$\begin{bmatrix} G^0(z) \\ z^{-1}G^1(z) \end{bmatrix} = \begin{bmatrix} z^2 - z & 1 \\ z^3 & z \\ z+1 & z^2 \\ z^{-1} - 1 & z^2 - 1 \\ z & 1-z \\ z^3+1 & z^{-1} \end{bmatrix} = \begin{bmatrix} z^2 & 1 \\ 0 & 0 \\ 1 & z^2 \\ -1 & z^2-1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} + \begin{bmatrix} -z^2 & 0 \\ z^4 & z^2 \\ z^2 & 0 \\ 1 & 0 \\ z^2 & -z^2 \\ z^4 & 1 \end{bmatrix} z^{-1}$$

Then

$$G^L(z) = \begin{bmatrix} z & 1 & -z & 0 \\ 0 & 0 & z^2 & z \\ 1 & z & z & 0 \\ -1 & z-1 & 1 & 0 \\ 0 & 1 & z & -z \\ 1 & 0 & z^2 & 1 \end{bmatrix}$$

State space representations

Definition: Let $A \in \mathbb{F}^{\delta \times \delta}, B \in \mathbb{F}^{\delta \times k}, C \in \mathbb{F}^{n \times \delta}$ and $D \in \mathbb{F}^{n \times k}$.

A state space system (A, B, C, D) described by

$$\left\{ \begin{array}{rcl} x(\ell + 1) & = & Ax(\ell) + Bu(\ell) \\ v(\ell) & = & Cx(\ell) + Du(\ell) \end{array} \right., \; l \in \mathbb{N}_0,$$

is said to be a state space realization of the time-invariant (n, k, δ) convolutional code $\mathcal{C}$ if $\mathcal{C}$ is the set of finite support output sequences v corresponding to finite support input sequences u and zero inicial conditions, i.e., $x(0) = 0$

State space representations

Definition: Let $A \in \mathbb{F}^{\delta \times \delta}, B \in \mathbb{F}^{\delta \times k}, C \in \mathbb{F}^{n \times \delta}$ and $D \in \mathbb{F}^{n \times k}$.

A state space system (A, B, C, D) described by

$$\left\{ \begin{array}{rcl} x(\ell+1) & = & Ax(\ell) + Bu(\ell) \\ v(\ell) & = & Cx(\ell) + Du(\ell) \end{array} \right. , \; l \in \mathbb{N}_0,$$

is said to be a state space realization of the time-invariant (n, k, δ) convolutional code $\mathcal{C}$ if $\mathcal{C}$ is the set of finite support output sequences v corresponding to finite support input sequences u and zero inicial conditions, i.e., $x(0) = 0$

Remark: Let (A, B, C, D) be a minimal realization of $\mathcal{C}$, i.e., A has the minimal possible dimension. Then A is nilpotent, (A, B) is controllable and (A, C) is observable

Periodic state space representation

Consider the following δ-dimensional s.s. representation of a periodic code $\mathcal{C}$

$$\left\{\begin{array}{rcl} x(\ell+1) & = & A(\ell)x(\ell)+B(\ell)u(\ell) \\ v(\ell) & = & C(\ell)x(\ell)+D(\ell)u(\ell) \end{array}\right. , \; l \in \mathbb{N}_0$$

where $(A(\cdot), B(\cdot), C(\cdot), D(\cdot)) \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{n\times\delta} \times \mathbb{F}^{n\times k}$ are periodic functions with period 2

Consider the following δ-dimensional s.s. representation of a periodic code $\mathcal{C}$

$$\left\{ \begin{array}{rcl} x(\ell+1) & = & A(\ell)x(\ell) + B(\ell)u(\ell) \\ v(\ell) & = & C(\ell)x(\ell) + D(\ell)u(\ell) \end{array} \right. , \; l \in \mathbb{N}_0$$

where $(A(\cdot), B(\cdot), C(\cdot), D(\cdot)) \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{n\times\delta} \times \mathbb{F}^{n\times k}$ are periodic functions with period 2

Let

$$w(\ell) = x(2\ell) \quad , \quad u^L(\ell) = \begin{bmatrix} u(2\ell) \\ u(2\ell+1) \end{bmatrix} \quad , \quad v^L(\ell) = \begin{bmatrix} v(2\ell) \\ v(2\ell+1) \end{bmatrix}$$

Induced representation

Then

$$\begin{cases} w(\ell+1) &= Ew(\ell) + Fu^L(\ell) \\ v^L(\ell) &= Hw(\ell) + Ju^L(\ell) \end{cases}$$

with

$$E = A(1)A(0) \qquad F = \begin{bmatrix} A(1)B(0) & B(1) \end{bmatrix}$$

$$H = \begin{bmatrix} C(0) \\ C(1)A(0) \end{bmatrix} \qquad J = \begin{bmatrix} D(0) & 0 \\ C(1)B(0) & D(1) \end{bmatrix}$$

Induced representation

Then

$$\begin{cases} w(\ell+1) & = & Ew(\ell) + Fu^L(\ell) \\ v^L(\ell) & = & Hw(\ell) + Ju^L(\ell) \end{cases}$$

with

$$E = A(1)A(0) \qquad F = \begin{bmatrix} A(1)B(0) & B(1) \end{bmatrix}$$

$$H = \begin{bmatrix} C(0) \\ C(1)A(0) \end{bmatrix} \qquad J = \begin{bmatrix} D(0) & 0 \\ C(1)B(0) & D(1) \end{bmatrix}$$

is a time-invariant δ-dimensional s.s. representation for the lifted code $\mathcal{C}^L$

Induced representation

Then

$$\begin{cases} w(\ell+1) = Ew(\ell) + Fu^L(\ell) \\ v^L(\ell) = Hw(\ell) + Ju^L(\ell) \end{cases}$$

with

$$E = A(1)A(0) \qquad F = \begin{bmatrix} A(1)B(0) & B(1) \end{bmatrix}$$

$$H = \begin{bmatrix} C(0) \\ C(1)A(0) \end{bmatrix} \qquad J = \begin{bmatrix} D(0) & 0 \\ C(1)B(0) & D(1) \end{bmatrix}$$

is a time-invariant δ-dimensional s.s. representation for the lifted code $\mathcal{C}^L$

The representation $\Sigma^L = (E, F, H, J)$ of $\mathcal{C}^L$ is said to be induced by the representation $\Sigma(\cdot) = (A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ of $\mathcal{C}$.

Induced representation

Then

$$\begin{cases} w(\ell+1) &= Ew(\ell) + Fu^L(\ell) \\ v^L(\ell) &= Hw(\ell) + Ju^L(\ell) \end{cases}$$

with

$$E = A(1)A(0) \qquad F = \begin{bmatrix} A(1)B(0) & B(1) \end{bmatrix}$$

$$H = \begin{bmatrix} C(0) \\ C(1)A(0) \end{bmatrix} \qquad J = \begin{bmatrix} D(0) & 0 \\ C(1)B(0) & D(1) \end{bmatrix}$$

is a time-invariant δ-dimensional s.s. representation for the lifted code $\mathcal{C}^L$

The representation $\Sigma^L = (E, F, H, J)$ of $\mathcal{C}^L$ is said to be induced by the representation $\Sigma(\cdot) = (A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ of $\mathcal{C}$.

Moreover, a time-invariant representation Σ^L of $\mathcal{C}^L$ is called induced whenever it is induced by some periodic representation $\Sigma(\cdot)$ of $\mathcal{C}$

Problem

Given a time-invariant state space realization of $\mathcal{C}$...

how can we know if it is induced?

Induced representation - characterization

Proposition: Let $\mathcal{C}$ be a 2-periodic code and $\mathcal{C}^L$ the lifted code associated to $\mathcal{C}$.

Then a δ-dimensional s.s. representation $\Sigma^L = (E, F, H, J)$ of $\mathcal{C}^L$, with

$$E \in \mathbb{F}^{\delta \times \delta}, \; F = \begin{bmatrix} F_1 & F_2 \end{bmatrix} \in \mathbb{F}^{\delta \times 2k}, \; H = \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} \in \mathbb{F}^{2n \times \delta}, \; J = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} \in \mathbb{F}^{2n \times 2k}$$

is induced if and only if

$$\operatorname{rank} \mathcal{M} = \begin{bmatrix} E & F_1 \\ H_2 & J_{21} \end{bmatrix} \leq \delta$$

Proposition: Let $\mathcal{C}$ be a 2-periodic code and $\mathcal{C}^L$ the lifted code associated to $\mathcal{C}$. Then a δ-dimensional s.s. representation $\Sigma^L = (E, F, H, J)$ of $\mathcal{C}^L$, with

$$E \in \mathbb{F}^{\delta\times\delta}, \; F = \begin{bmatrix} F_1 & F_2 \end{bmatrix} \in \mathbb{F}^{\delta\times 2k}, \; H = \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} \in \mathbb{F}^{2n\times\delta}, \; J = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} \in \mathbb{F}^{2n\times 2k}$$

is induced if and only if

$$\operatorname{rank} \mathcal{M} = \begin{bmatrix} E & F_1 \\ H_2 & J_{21} \end{bmatrix} \leq \delta$$

Moreover, in this case, decomposing the matrix $\mathcal{M}$ as $\mathcal{M} = \overbrace{\begin{bmatrix} N_1 \\ N_2 \end{bmatrix}}^{\delta} \left.\begin{bmatrix} Q_1 & Q_2 \end{bmatrix}\right\}\delta$

the 2-periodic δ-dimensional s.s. representation of $\mathcal{C}$ that induces Σ^L is $\Sigma(\cdot) = (A(\cdot), B(\cdot), C(\cdot), D(\cdot))$, where

$$A(0) = Q_1 \quad A(1) = N_1 \quad B(0) = Q_2 \quad B(1) = F_2$$
$$C(0) = H_1 \quad C(1) = N_2 \quad D(0) = J_{11} \quad D(1) = J_{22}$$

Increasing free distance

- $(3, 2, 2, 1)$ time-invariant conv. codes over the binary field have at most free distance 3
- The Griesmer bound for this kind of codes is 4

Increasing free distance

- $(3,2,2,1)$ time-invariant conv. codes over the binary field have at most free distance 3

- The Griesmer bound for this kind of codes is 4

- We will construct a 2-periodic $(3,2,2,1)$ conv. code with free distance 4 based on the construction of a time-invariant code whose s.s. realization is induced by a 2-periodic realization

Increasing free distance

- $(3, 2, 2, 1)$ time-invariant conv. codes over the binary field have at most free distance 3

- The Griesmer bound for this kind of codes is 4

- We will construct a 2-periodic $(3, 2, 2, 1)$ conv. code with free distance 4 based on the construction of a time-invariant code whose s.s. realization is induced by a 2-periodic realization

This shows that time-varying conv. codes can attain larger free distance than time-invariant ones

2-periodic $(3, 2, 2, 1)$ convolutional code

Consider the $(6, 4, 2, 1)$ time-invariant convolutional code, $\mathcal{C}^L$, over $\mathbb{F}_2$ with generator matrix $G = G_1 z + G_0$, where

$$G_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \quad \text{and} \quad G_1 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

2-periodic $(3, 2, 2, 1)$ convolutional code

Consider the $(6, 4, 2, 1)$ time-invariant convolutional code, $\mathcal{C}^L$, over $\mathbb{F}_2$ with generator matrix $G = G_1 z + G_0$, where

$$G_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \quad \text{and} \quad G_1 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The free distance of $\mathcal{C}^L$ is 4

2-periodic $(3, 2, 2, 1)$ convolutional code

A s.s. realization of G is given by $(E, F, H, J) \in \mathbb{F}^{2\times 2} \times \mathbb{F}^{2\times 4} \times \mathbb{F}^{6\times 2} \times \mathbb{F}^{6\times 4}$ where

$$E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad J = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

2-periodic $(3, 2, 2, 1)$ convolutional code

A s.s. realization of G is given by $(E, F, H, J) \in \mathbb{F}^{2\times 2} \times \mathbb{F}^{2\times 4} \times \mathbb{F}^{6\times 2} \times \mathbb{F}^{6\times 4}$ where

$$E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad J = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Then $\mathcal{M} = \begin{bmatrix} E & F_1 \\ H_2 & J_{21} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

A s.s. realization of G is given by $(E, F, H, J) \in \mathbb{F}^{2\times 2} \times \mathbb{F}^{2\times 4} \times \mathbb{F}^{6\times 2} \times \mathbb{F}^{6\times 4}$ where

$$E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad J = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Then $\mathcal{M} = \begin{bmatrix} E & F_1 \\ H_2 & J_{21} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

$\operatorname{rank} \mathcal{M} = 2 \leq \delta$, so this realization is induced by a 2-periodic $(3,2,2,1)$ code

2-periodic $(3, 2, 2, 1)$ convolutional code

The 2-periodic $(3, 2, 2, 1)$ code has a realization $\Sigma(\cdot) = (A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ where

$$A(0) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad A(1) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad B(0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad B(1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$C(0) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad C(1) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \quad D(0) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \quad D(1) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{bmatrix}$$

The 2-periodic $(3,2,2,1)$ code has a realization $\Sigma(\cdot) = (A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ where

$$A(0) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad A(1) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad B(0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad B(1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$C(0) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \quad C(1) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \quad D(0) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \quad D(1) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{bmatrix}$$

This 2-periodic code can also be described as

$$\mathcal{C} = \{v : v(2\ell + t) = (G^t(\sigma^{-1})u(2\ell + t); t = 0, 1; \ \ell \in \mathbb{N}_0\}$$

where $G^t(z) = C(t)\left(z^{-1}I - A(t)\right)^{-1} B(t) + D(t)$, i.e.,

$$G^0(z) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} z + \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \text{and} \quad G^1(z) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} z + \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{bmatrix}$$

Conclusions

- **We have studied the relation between time-invariant and time-varying conv. codes by means of s.s. representations**

Conclusions

- We have studied the relation between time-invariant and time-varying conv. codes by means of s.s. representations

- Using a lifting technique we provided conditions to transform a given periodic code into a time-invariant one

Conclusions

- We have studied the relation between time-invariant and time-varying conv. codes by means of s.s. representations

- Using a lifting technique we provided conditions to transform a given periodic code into a time-invariant one

- We showed conditions for state-space representation of a time-invariant convolutional code to be induced (i.e. being the lifting of a 2-periodic one)

Conclusions

- We have studied the relation between time-invariant and time-varying conv. codes by means of s.s. representations

- Using a lifting technique we provided conditions to transform a given periodic code into a time-invariant one

- We showed conditions for state-space representation of a time-invariant convolutional code to be induced (i.e. being the lifting of a 2-periodic one)

- This provides a technique to build 2-periodic convolutional codes from time-invariant ones.

Conclusions

- We have studied the relation between time-invariant and time-varying conv. codes by means of s.s. representations

- Using a lifting technique we provided conditions to transform a given periodic code into a time-invariant one

- We showed conditions for state-space representation of a time-invariant convolutional code to be induced (i.e. being the lifting of a 2-periodic one)

- This provides a technique to build 2-periodic convolutional codes from time-invariant ones.

- Constructions of periodic convolutional codes of higher periods and with other parameters are currently under investigation.

Open problems

- **Which are the conditions for a time-invariant convolutional code $\mathcal{C} = \operatorname{Im} G$, to admit a minimal state space realization that is induced by a 2-periodic realization?**

Open problems

- **Which are the conditions for a time-invariant convolutional code $\mathcal{C} = \mathrm{Im}\, G$, to admit a minimal state space realization that is induced by a 2-periodic realization?**

- **Given a periodic convolutional code (a set of G_i), does there always exist a periodic realization that realizes that convolutional code?**

Open problems

- **Which are the conditions for a time-invariant convolutional code $\mathcal{C} = \mathrm{Im}\, G$, to admit a minimal state space realization that is induced by a 2-periodic realization?**

- **Given a periodic convolutional code (a set of G_i), does there always exist a periodic realization that realizes that convolutional code?**

- **If so, How to characterize the state space periodic realizations associated to periodic kernel/image representations?**

Open problems

- Which are the conditions for a time-invariant convolutional code $\mathcal{C} = \mathrm{Im}\, G$, to admit a minimal state space realization that is induced by a 2-periodic realization?

- Given a periodic convolutional code (a set of G_i), does there always exist a periodic realization that realizes that convolutional code?

- If so, How to characterize the state space periodic realizations associated to periodic kernel/image representations?

- Given an encoder $G(D)$, find $G_0(D)$ and $G_1(D)$ s.t. $G(D)$ corresponds to the lifted of the periodic system associated to $G_0(D)$ and $G_1(D)$.

Open problems

- Which are the conditions for a time-invariant convolutional code $\mathcal{C} = \operatorname{Im} G$, to admit a minimal state space realization that is induced by a 2-periodic realization?

- Given a periodic convolutional code (a set of G_i), does there always exist a periodic realization that realizes that convolutional code?

- If so, How to characterize the state space periodic realizations associated to periodic kernel/image representations?

- Given an encoder $G(D)$, find $G_0(D)$ and $G_1(D)$ s.t. $G(D)$ corresponds to the lifted of the periodic system associated to $G_0(D)$ and $G_1(D)$.

Thank you

Acknowledgements

This work was supported in part by the Portuguese Foundation for Science and Technology (FCT-Fundação para a Ciência e a Tecnologia), through CIDMA - Center for Research and Development in Mathematics and Applications, within project UID/MAT/04106/2013 and also by Project POCI-01-0145-FEDER-006933 - SYSTEC - Research Center for Systems and Technologies - funded by FEDER funds through COMPETE2020 - Programa Operacional Competitividade e Internacionalização (POCI) - and by national funds through FCT.

CIDMA CENTRO DE I&D EM MATEMÁTICA E APLICAÇÕES CENTER FOR R&D IN MATHEMATICS AND APPLICATIONS

SYSTEC RESEARCH CENTER FOR SYSTEMS & TECHNOLOGIES

Universidade de Aveiro

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO