Technical University
of Munich (Germany)

University of Groningen
(Netherlands)

Carl von Ossietzky
University of Oldenburg
(Germany)

A Short Introduction to Cyclic Convolutional Codes

Barbara Langfeld (joint work with Heide Gluesing-Luerssen and Wiland Schmale)

langfeld@ma.tum.de gluesing@math.rug.nl wiland.schmale@uni-oldenburg.de

❶ Preliminaries: Block Codes, Submodules, and Convolutional Codes (CCs)

Throughout this text, $\mathbb{F}$ denotes a finite field.

Encoding a message word $u \in \mathbb{F}^k$ can be done by applying an injective linear mapping $u \mapsto uG$ (see the box on the right). The set $\mathrm{im}_{\mathbb{F}}\, G$ is called an (n,k)-block code. Encoding a sequence of message words $\sum_{i=0}^{t} z^i u_i$ can be done similarly by applying an arbitrary injective $\mathbb{F}[z]$-module homomorphism $\sum_{i=0}^{t} z^i u_i \mapsto (\sum_{i=0}^{t} z^i u_i)G$ (see the box on the right). If G is a matrix with $\mathbb{F}$-entries, then the z^j-term of $(\sum_{i=0}^{t} z^i u_i)G$ only depends of u_j. In this sense, the encoder $(\cdot)G$ has “no memory” and we could apply G to the single message words u_i as well. But if G has polynomial entries, then in general the z^j-term of $(\sum_{i=0}^{t} z^i u_i)G$ will not only depend on u_j (but of $u_j, u_{j-1}, \ldots$). So $(\cdot)G$ has some kind of “memory”. In this sense the (n,k)-submodule $\mathrm{im}_{\mathbb{F}[z]}\, G$ has some advantage by comparison with block codes. If G is right invertible, then $\mathrm{im}_{\mathbb{F}[z]}\, G$ is called a (n,k)-convolutional code (CC). Right invertibility implies desirable properties such as the existence of a parity check matrix. (For more information see e.g. [McE98].)

	message words	→	code words
(n,k)-block code	$\mathbb{F}^k$	$\rightarrow$	$\mathbb{F}^n$
$\mathrm{im}_{\mathbb{F}}\, G$	u	$\mapsto$	uG
	where $k \le n$, $G \in \mathbb{F}^{k\times n}$, $\mathrm{rank}\, G = k$		
(n,k)-submodule	$\mathbb{F}[z]^k$	$\rightarrow$	$\mathbb{F}[z]^n$
$\mathrm{im}_{\mathbb{F}[z]}\, G$	$\sum_{i=0}^{t} z^i u_i$	$\mapsto$	$\sum_{i=0}^{t} z^i u_i G$
	where $k \le n$, $G \in \mathbb{F}[z]^{k\times n}$, $\mathrm{rank}\, G = k$		
(n,k)-CC	An (n,k)-CC is an (n,k)-submodule $\mathrm{im}_{\mathbb{F}[z]}\, G$ with G right invertible.		

Generator matrices G of an (n,k)-submodule $\mathrm{im}_{\mathbb{F}[z]}\, G$ are not unique. We can try to find ‘nice’ ones, where ‘nice’ can be measured in terms of the degree of G:

$$\deg G := \begin{array}{l}\text{sum of the } z\text{-degrees of the rows of } G\\ \text{(viewed as polynomials in } \mathbb{F}^n[z])\end{array} .$$

For example

$$\mathrm{im}_{\mathbb{F}[z]} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} = \mathrm{im}_{\mathbb{F}[z]} \begin{bmatrix} 1 & z^2 & z^2 \\ z & z^3+1 & z^3+1 \end{bmatrix} ;$$

the left matrix has degree 0, the right one has degree 5. We see that this submodule has a generator matrix with constant entries. It is not better than a block code!

We put this down more formally: Let $\mathcal{C}$ be an (n,k)-submodule (or -CC). Then

$$\delta := \min\{\deg G \mid \mathrm{im}_{\mathbb{F}[z]}\, G = \mathcal{C}\}$$

is called complexity of $\mathcal{C}$. The set $\mathcal{C}$ is also called (n,k,δ)-submodule (or -CC).

Note: Submodules of complexity 0 are in a sense block codes (because they have a constant generator matrix). A nonzero complexity is desirable!

❷ How to Define Cyclic Convolutional Codes?

Standard assumption: $\gcd(n, \mathrm{char}\,\mathbb{F}) = 1$

Cyclic Block Codes

The following diagram briefly recalls the well known definition of cyclic block codes and the characterization of cyclicity in the ‘vector world’ and in the ‘polynomial world’.

$$\begin{tikzcd} \mathbb{F}^n \arrow[r, shift right, "\mathfrak{p}"', "\text{‘polynomialize’}"'] & \mathbb{F}[x]/\langle x^n - 1\rangle =: A \arrow[l, shift right, "\mathfrak{v}"', "\text{‘vectorize’}"'] \end{tikzcd}$$

$v = (v_0, \ldots, v_{n-1})$	$\leftarrow\!\!\dashv$ $\mapsto$	$\mathfrak{p}(v) = \sum_{i=0}^{n-1} v_i x^i$
cyclic shift: $v \mapsto$ $(v_{n-1}, v_0, \ldots, v_{n-2})$	$\longleftrightarrow$	multiplication with x: $\mathfrak{p}(v) \mapsto x \cdot \mathfrak{p}(v)$
$\mathcal{C}$ cyclic, i.e., invariant under cyclic shift	$\Longleftrightarrow$	$\mathfrak{p}(\mathcal{C})$ ideal in A

If g in $\mathfrak{p}(\mathcal{C}) = \langle g \rangle$ is chosen suitably (generator polynomial), then g contains all information about $\mathcal{C}$ (dimension, generator matrix, ...).

Cyclic Submodules — First Idea

Use exactly the same definition as in the block code case, i.e. “invariance under the cyclic shift”. Here, “cyclic shift” means shifting all coefficient vectors *simultaneously once*.

$\mathcal{C}$ cyclic $\overset{\text{reasonable?}}{:\Longleftrightarrow}$ $\mathcal{C}$ invariant under cyclic shift
i.e.
$g \in \mathfrak{p}(\mathcal{C}) \Rightarrow x \cdot g \in \mathfrak{p}(\mathcal{C})$,

Here, the ‘polynomialize’-function $\mathfrak{p}$ from the previous box is extended canonically: $\mathfrak{p}(\sum_{i=0}^{t} z^i u_i) = \sum_{i=0}^{t} z^i \mathfrak{p}(u_i)$

where we have (of course)
$x \cdot (\sum_{\nu \ge 0} z^\nu g_\nu) = \sum_{\nu \ge 0} z^\nu x g_\nu$

This definition is *not* fruitful, because it provides no new structure:

Theorem [Pir76], [Roo79]
A convolutional code that is invariant under the cyclic shift has complexity 0. In particular, it is a cyclic block code (in the sense discussed in ❶).

Can we find a generalized concept of cyclicity that admits nontrivial cyclic submodules?

Cyclic Submodules — Idea of Piret

Use again “invariance under the cyclic shift”, but generalize “shift”: Do not shift all coefficient vectors simultaneously once, but different coefficient vectors differently often (according to rules, that yield a nice structure). See [Pir76] for details.

Let $\mathcal{C}$ be an (n,k)-submodule.
Choose $m \in \mathbb{N}$ so that $\gcd(n,m) = 1$.
Define m-cyclicity via the condition

$$g = \sum_{\nu \ge 0} z^\nu g_\nu \in \mathfrak{p}(\mathcal{C})$$
$$\Rightarrow\ x *_m g := \sum_{\nu \ge 0} z^\nu x^{(m^\nu)} g_\nu \in \mathfrak{p}(\mathcal{C}) .$$

The operation “$*_m$” can be canonically extended to a multiplication on $A[z]$.
$(A[z], +, *_m)$ is an $\mathbb{F}$-Algebra, which is in general non-commutative. Pirets idea yields a first non-trivial *and* reasonable definition because:

$$\mathcal{C}\ m\text{-cyclic} \iff \mathfrak{p}(\mathcal{C}) \text{ left ideal in } (A, +, *) .$$

Indeed, this notion of cyclicity can still be generalized.

Generalisation of Roos

The Definition

Generalize “shift” once more: Let a “shifted” coefficient vector be a linear combination of (multiply) shifted coefficient vectors (according to rules, that yield a nice structure). See [Roo79] and [GLuS04] for details.

Let $\mathcal{C}$ be an (n,k)-submodule.
Let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$, where $\mathrm{Aut}_{\mathbb{F}}(A)$ denotes the group of all $\mathbb{F}$-algebra automorphisms on A.
Define σ-cyclicity via the condition

$$g = \sum_{\nu \ge 0} z^\nu g_\nu \in \mathfrak{p}(\mathcal{C})$$
$$\Rightarrow\ x *_\sigma g := \sum_{\nu \ge 0} z^\nu \sigma^\nu(x) g_\nu \in \mathfrak{p}(\mathcal{C}) .$$

The operation “$*_\sigma$” can be canonically extended to a multiplication on $A[z]$.
$A[z;\sigma] := (A[z], +, *_\sigma)$ is an $\mathbb{F}$-Algebra, the Piret-Algebra, which is in general non-commutative; we put the coefficients always on the right hand side.
Roos’ concept of cyclicity contains that of Piret, it is more general, and it is also reasonable because:

$$\mathcal{C}\ \sigma\text{-cyclic} \iff \mathfrak{p}(\mathcal{C}) \text{ left ideal in } A[z;\sigma] .$$

We adopt the definition of Roos and give **a small Example:**
Let $n = 3$, $\mathbb{F} = \mathbb{F}_4 = \{0, 1, \alpha, \alpha^2\}$, $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ defined via $\sigma(x) = \alpha^2 x$.
Consider

$$\begin{aligned} G &:= [1+z+z^2, \alpha+z+\alpha^2 z^2, \alpha^2+z+\alpha z^2] \\ &= [1, \alpha, \alpha^2] + z[1,1,1] + z^2[1, \alpha^2, \alpha] . \end{aligned}$$

(We will use G both as matrix and as code word.) ≫

The submodule $\mathcal{C} := \mathrm{im}_{\mathbb{F}[z]}\, G$ has dimension $k = 1$ and complexity $\delta = 2$. Moreover, G is right invertible. Thus $\mathcal{C}$ is a $(3,1,2)$-CC.
The submodule $\mathcal{C}$ is even a σ-cyclic CC. To prove this, believe that it is sufficent to show “$x *_\sigma \mathfrak{p}(G)$” and “$x^2 *_\sigma \mathfrak{p}(G) \in \mathfrak{p}(\mathcal{C})$”. This can be done by hand:

$$g := \mathfrak{p}(G) = 1 + \alpha x + \alpha^2 x^2 + z(1 + x + x^2) + z^2(1 + \alpha^2 x + \alpha x^2) ,$$

≫

$$\begin{aligned} x *_\sigma g &= \alpha^2 + x + \alpha x^2 + z\alpha^2(1+x+x^2) + z^2(\alpha^2 + \alpha x + x^2) \\ &= \alpha^2 g \in \mathfrak{p}(\mathcal{C}) , \\ x^2 *_\sigma g &= \alpha g \in \mathfrak{p}(\mathcal{C}) . \end{aligned}$$

Therefore $\mathcal{C}$ is σ-cyclic. One can show, that $\mathcal{C}$ is an MDS convolutional code (it has free distance 9; for the definitions of free distance and “MDS” cf. [McE98]).

❸ Analyzing Cyclic CCs with the Aid of Gröbner-type Theory

Representing $A = \mathbb{F}[x]/\langle x^n - 1\rangle$ as a Product of Fields

Let $x^n - 1 = \pi_1 \cdot \cdots \cdot \pi_r$ be the decomposition of $x^n - 1$ into (pairwise different) normalized prime factors. (The decomposition is unique up to permutation of the π_i.) Due to the Chinesise Remainder Theorem we get the following isomorphism of rings:

$$\begin{array}{rclcl} \rho & : & A & \rightarrow & \mathbb{F}[x]/\langle \pi_1 \rangle \times \cdots \times \mathbb{F}[x]/\langle \pi_r \rangle \\ & & a & \mapsto & [a \mod \pi_1, \ldots, a \mod \pi_r] \end{array}$$

☞ The elements $\varepsilon^{(\ell)} := \rho^{-1}([\ldots 0, \overset{\ell}{\overset{\downarrow}{1}}, 0 \ldots])$ are called primitive idempotent elements of A. Note: $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ permutes $\varepsilon^{(1)}, \ldots, \varepsilon^{(r)}$.

☞ For $h \in A[z;\sigma]$ we call $h^{(\ell)} := \varepsilon^{(\ell)} *_\sigma h$ the ℓ-th component of h.

The box on the left displays how A can be represented as a product of fields, and how a polynomial h can be splitted into components. The order of the fields and thus the order of the components induces a ‘term order’ on the elements of $A[z;\sigma]$.

Since $\mathcal{C}$ is a σ-cyclic CC if and only if $\mathfrak{p}(\mathcal{C})$ is a left ideal in $A[z;\sigma]$, we can ask ourselves, how to find a ‘nice’ generator set of a left ideal in $A[z;\sigma]$. Some basic algebra shows that each such ideal has a finite generator set. The ‘term order’ together with a Buchberger-type algorithm can be used to generate a ‘reduced’ set of generators with nice properties. For σ-cyclic CCs these results can be strengthened. (For details see [GluS04]).

The following list gives several results that can be derived by evaluating this approach.

Results

cf. [GluS04], [GluL06], [GluS], [GluL]

☞ If $\mathcal{C}$ is a σ-cyclic CC, then $\mathfrak{p}(\mathcal{C})$ left *principal* ideal. (The converse is only true under additional assumptions.)

☞ If $\mathcal{C}$ is a σ-cyclic CC, then there exists a ‘reduced’ generator polynomial $g \in A[z;\sigma]$ for $\mathfrak{p}(\mathcal{C}) = {}^{\bullet}\langle g \rangle$. As in the block code case, g contains all information about $\mathcal{C}$ (dimension, complexity, generator matrix, ...).

☞ A minimal σ-cyclic CC is a σ-cyclic CC that has no non-trivial sub-σ-cyclic CCs. Each σ-cyclic CC $\mathcal{C}$ can be decomposed into a direct sum of $\mathbb{F}[z]$-left modules $\mathcal{C} = \bigoplus_{i=1}^{s} \mathcal{C}_i$, where the $\mathcal{C}_i$ are minimal σ-cyclic CCs.

☞ $\mathcal{C}$ is a minimal σ-cyclic CC if and only if its generator polynomial g satisfies $g = g^{(\ell)}$ for some $1 \le \ell \le r$.

☞ For a minimal σ-cyclic CC $\mathcal{C}$ with $g = g^{(\ell)}$ and $k = \deg_x \pi_\ell$, the complexity of $\mathcal{C}$ is $\delta = k \cdot d$ for some $d \in \mathbb{N}$. Moreover, we have the following equivalence:

$$\sigma(\varepsilon^{(\ell)}) \neq \varepsilon^{(\ell)} \quad \Leftrightarrow \quad \text{For any } d \in \mathbb{N}_0 \text{ there exists a minimal } (n,k,kd)\text{-}\sigma\text{-cyclic CC with } g = g^{(\ell)} .$$

☞ Within the large class of cyclic convolutional codes, Reed-Solomon and BCH convolutional codes can be defined. They contain optimal or near optimal with respect to distance and performance (cf. [GluS]).
For example, one can construct a class of cyclic one-dimensional MDS convolutional codes with a Reed-Solomon structure (cf. [GluL]).

... and many more ...

References

[GluL06] Heide Gluesing-Luerssen, Barbara Langfeld: “On the Algebraic Parameters of Convolutional Codes with Cyclic Structure.” *Journal of Algebra and Its Applications*, 5(1):53–76, 2006.

[GluL] Heide Gluesing-Luerssen, Langfeld, Barbara: “A Class of One-Dimensional MDS Convolutional Codes.” To appear in *Journal of Algebra and its Applications*. [Preprint cs.IT/0412085]

[GluS04] Heide Gluesing-Luerssen, Wiland Schmale: “On Cyclic Convolutional Codes.” *Acta Applicandae Mathematicae* 82:183–237, 2004.

[GluS] Heide Gluesing-Luerssen, Wiland Schmale: “On doubly-cyclic convolutional codes.” To appear in *Applicable Algebra in Engineering, Communication and Computing.* [Preprint math.RA/0410317]

[McE98] Robert J. McEliece: “The algebraic theory of convolutional codes.” In: *Handbook of Coding Theory*, volume 1:1065–1138, Elsevier, Amsterdam, 1998.

[Pir76] Philippe Piret: “Structure and constructions of cyclic convolutional codes.” *IEEE Trans. Inform. Theory*, 22:147–155, 1976.

[Roo79] Cornelis Roos: “On the structure of convolutional and cyclic convolutional codes.” *IEEE Trans. Inform. Theory*, 25:676–683, 1979.

Barbara Langfeld, May 2006