CRITICALLY SUBSAMPLED FILTERBANKS IMPLEMENTING REED-SOLOMON CODES: AN ALGEBRAIC POINT OF VIEW

Geert Van Meerbergen [⋆], *Marc Moonen* [⋆]

[⋆] E.E. Dept., ESAT/SISTA, K.U.Leuven
Kasteelpark Arenberg 10, 3001 Leuven, Belgium
gvanmeer,moonen@esat.kuleuven.ac.be

Hugo De Man [⋆†]

[†] IMEC
Kapeldreef 75, 3001 Leuven, Belgium
deman@imec.be

ABSTRACT

The last decade shows a growing interest in soft decoding techniques, motivated by a soft decoding gain of roughly 2dB. Most of the techniques are applied to concatenated codes, in particular to Turbo codes. This is in strong contrast to many existing coding schemes where Reed-Solomon (RS) codes are common. Recently, we unveiled a filterbank structure behind the RS codes. Using this filterbank decomposition, a RS code is broken into many smaller subcodes that can consequently be used to build a Soft-In Soft-Out (SISO) RS decoder. A limitation of this previous work is that it is only applicable to RS codes where the codeword and dataword length are *not* coprime. In this paper, this constraint is eliminated. A purely algebraic method is presented to construct a filterbank decomposition for any RS code, as long as a subfield exists in the Galois field in which the RS code operates. This method gives a lot of insight into the algebraic structure of RS codes and their corresponding filterbanks.

1. INTRODUCTION

The importance of efficient SISO Reed-Solomon decoders [1] nowadays can hardly be underestimated: The application domain of RS codes spans from storage devices (including tape, Compact Disk, DVD, barcodes, etc) over high-speed modems (such as ADSL, VDSL, etc), wireless and mobile communications (including cellular telephones, microwave links, etc) to satellite and deep space communications. Algebraic decoding algorithms are readily available for hard-decision decoding, e.g. Berlekamp-Massey's algorithm and Guruswami-Sudan's list decoding algorithm [2]. Motivated by a potential 2dB soft decoding gain for RS codes [3], Koetter and Vardy developed a soft front-end for hard RS decoding in [4], resulting in a Soft-In Hard-Out algorithm. However, Benedetto, Divsalar and Hagenauer describe in [5] that a Soft-In Soft-Out (SISO) decoder is really necessary since this allows soft information to be exchanged in an iterative fashion between the different blocks in the receiver: This so-called *Turbo Principle* is the basic mechanism behind joined equalisation, demodulation and channel decoding. Thus in light of the ubiquity of RS codes, and taking into account the possible soft decoding gain, SISO decoding of RS codes is currently one of the most important problems in coding theory. Recently, we presented a method to build a SISO RS decoder [6]. This is based on a critically subsampled filterbank structure behind RS codes [7].

Filterbanks have long been known to be a powerfull tool for image and audio processing. Recently, their importance has also been recognized in communication systems. In [8], Scaglione *et al.* show that many modulation schemes including OFDM, DMT, TDMA, and CDMA can actually be viewed as filterbanks that build input diversity (add redundancy) at the transmitter. Filterbanks that add redundancy with the purpose of error correction - and therefore work in finite fields - are addressed by Fekri [9]. In [6], we develop filterbanks with a completely different structure that implement the famous RS codes. This method only works if dataword and codeword length are *not* coprime, which seriously impairs its practical usability. In the first part of this paper, the filterbank structure is studied from a purely algebraic point of view. This insight leads to filterbanks implementing any RS code, as long as a subfield of the Galois Field in which the RS code operates exists. This is an important step, since in many practical applications, the codeword and dataword length are coprime, e.g. the popular RS(255,223).

2. COOK-TOOM'S ALGORITHM AS THE BASIS OF A FILTERBANK IMPLEMENTATION

Since non-systematic RS codes are essentially FIR filters, fast FIR filtering algorithms based on *Cook-Toom's algorithm* [11] are studied in this section. It is seen that *Cook-Toom's algorithm* is also the basic algorithm behind FIR filterbank realizations, as studied by Vetterli [12].

Straightforwardly computing a product $y(z^{-1})$ of two polynomials $g(z^{-1}) = \mathbf{g}_0 + \mathbf{g}_1 z^{-1} + ... + \mathbf{g}_{L-1} z^{-L+1}$ and $u(z^{-1}) = \mathbf{u}_0 + \mathbf{u}_1 z^{-1} + ... + \mathbf{u}_{N-1} z^{-N+1}$ requires NL multiplications. The application of *Cook-Toom*'s algorithm is known to reduce the number of multiplications to $M \geq N + L - 1$. The procedure is as follows: First, choose a set of interpolation points $\{\rho_i\}_{i=0:M-1}$ that are the roots of $r(z^{-1}) = \prod_{i=0}^{M-1} \left(z^{-1} - \rho_i\right)$. Evaluate $y(\rho_i) = g(\rho_i)u(\rho_i)$ and perform *Lagrange* interpolation to restore $y(z^{-1})$ $= \sum_{i=0}^{M-1} y(\rho_i) L_i(z^{-1})$ with $L_i(z^{-1}) = \frac{\prod_{k \neq i} (z^{-1} - \rho_k)}{\prod_{k \neq i} (\rho_i - \rho_k)}$.

If the polynomial multiplication is written as $\mathbf{y} = \mathbf{G}\mathbf{u}$, then *Cook-Toom*'s algorithm can be viewed as a matrix decomposition $\mathbf{G} = \mathbf{C}\mathbf{D}\mathbf{A}$, with $\mathbf{D} = \mathrm{diag}(\mathbf{B}\mathbf{g})$. In this equation, $\mathbf{G}$ is the $(N+L-1) \times N$ Toeplitz matrix defining the filter $g(z^{-1})$. $\mathbf{A}$ is the $M \times N$ Vandermonde matrix with $\mathbf{A}_{m,n} = \rho_m^n (n = 0 : N-1)$, $\mathbf{B}$ is also the $M \times L$ Vandermonde matrix with $\mathbf{B}_{m,l} = \rho_m^l (l = 0 : L-1)$ and $\mathbf{C}$ is the $(N+L-1) \times M$ matrix whose ith column contains the first $N + L - 1$ coefficients of $L_i(z^{-1})$.

Note that *Cook-Toom*'s algorithm is a special case of the *Chinese Remainder Theorem* [11], and that if the degree of $r(z^{-1})$ is too low $(M < L + N - 1)$, the convolution modulo $r(z^{-1})$ is

Geert Van Meerbergen is a Research Assistant with the F.W.O. Vlaanderen. This research work was carried out at the ESAT laboratory of the Katholieke Universiteit Leuven, in the frame of the Interuniversity Poles of Attraction Programme P5/22 and P5/11, the Concerted Research Action GOA-MEFISTO-666, Research Project FWO nr.G.0196.02. The scientific responsibility is assumed by its authors.

0-7803-8874-7/05/$20.00 ©2005 IEEE

calculated. In a coding context, finite fields like the *Galois Field* (GF) are common. Because of the underlying polynomial arithmetic, *Cook-Toom*'s algorithm is also applicable in other fields besides the complex field $\mathbb{C}$, as illustrated by the following example in $GF(2^4)$[1]:

Example 1 *Using Cook-Toom's algorithm with roots* $[\alpha^2, \alpha^5, \alpha^8, \alpha^{11}, \alpha^{14}]$*, a Toeplitz matrix* $\mathbf{G}$ *corresponding to a convolution of* $u(z^{-1})$ *(*$N = 4$*) with* $g(z^{-1}) = \alpha^5 + z^{-1}$ *(*$L = 2$*) can be decomposed as:*

$$\underbrace{\begin{bmatrix} \alpha^5 & & & \\ 1 & \alpha^5 & & \\ & 1 & \alpha^5 & \\ & & 1 & \alpha^5 \\ & & & 1 \end{bmatrix}}_{\mathbf{G}} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ \alpha^{13} & \alpha^{10} & \alpha^7 & \alpha^4 & \alpha^1 \\ \alpha^{11} & \alpha^5 & \alpha^{14} & \alpha^8 & \alpha^2 \\ \alpha^9 & 1 & \alpha^6 & \alpha^{12} & \alpha^3 \\ \alpha^7 & \alpha^{10} & \alpha^{13} & \alpha^1 & \alpha^4 \end{bmatrix}}_{\mathbf{C}} \underbrace{\begin{bmatrix} \alpha^1 & & & & \\ & 0 & & & \\ & & \alpha^4 & & \\ & & & \alpha^3 & \\ & & & & \alpha^{12} \end{bmatrix}}_{\mathbf{D}} \underbrace{\begin{bmatrix} 1 & \alpha^2 & \alpha^4 & \alpha^6 \\ 1 & \alpha^5 & \alpha^{10} & 1 \\ 1 & \alpha^8 & \alpha^1 & \alpha^9 \\ 1 & \alpha^{11} & \alpha^7 & \alpha^3 \\ 1 & \alpha^{14} & \alpha^{13} & \alpha^{12} \end{bmatrix}}_{\mathbf{A}}$$

Note that in $GF(2^4)$, a subfield exists generated by α^3. Using the elements of this subfield $[\alpha^0, \alpha^3, \alpha^6, \alpha^9, \alpha^{12}]$ as the roots in Cook-Toom's algorithm leads to a DFT-like decomposition. In the Galois field, the DFT-ed sequence is better known as the Mattson-Solomon polynomial [13].

Using Cook-Toom's algorithm, it is also possible to decompose a given diagonal matrix $\mathbf{D}$, where one of the factors is a Toeplitz matrix $\mathbf{G}$. This property will be crucial later on. Assume a diagonal matrix $\mathbf{D}$ and M Cook-Toom roots are given. The decomposition can only exist if diag($\mathbf{D}$) lies in the subspace spanned by (the columns of) $\mathbf{B}$. Otherwise stated, only L diagonal elements of $\mathbf{D}$ can freely be chosen non-zero for Cook-Toom's algorithm to exist. Nevertheless, note that $\mathbf{D}$ can at most contain $L-1$ *zero* diagonal elements. This leads to the following theorem:

Theorem 1 *Let* L,N,$M \geq L + N - 1$ *and* M *distinct roots* $\{\rho_i\}_{i=0:M-1}$ *be given. Any* $M \times M$ *diagonal matrix* $\mathbf{D}$ *with* $L-1$ *zeros can be decomposed as a product* $\mathbf{D} = \tilde{\mathbf{A}}\mathbf{G}\mathbf{W}$*, where* $\tilde{\mathbf{A}}$ *is an* $M \times M$ *Vandermonde matrix* $\tilde{\mathbf{A}}(m,n) = \rho_m^n (n = 0 : M-1)$*,* $\mathbf{G}$ *is a Toeplitz matrix implementing a filter of length* L *and* $\mathbf{W}$ *is an* $N \times M$ *matrix.*

Proof 1 *Let* $\mathbf{A}$*,* $\mathbf{B}$ *and* $\mathbf{C}$ *be the Cook-Toom matrices corresponding to the roots* $\{\rho_i\}_{i=0:M-1}$*. With* ν *the non-zero positions of* $\mathbf{D}$*,* $\mathbf{G}$ *can be calculated as follows*[2]*:*

$$\mathbf{D}' = \mathbf{B}\textit{null}(\mathbf{B}(\nu, 1:L)). \tag{1}$$

Now, $\mathbf{G} = \mathbf{C}\mathbf{D}'\mathbf{A}$ *holds. To calculate* $\mathbf{W}$*, let* $\hat{\mathbf{T}}$ *be the submatrix of* $\mathbf{T} = \mathbf{C}^{-1}\mathbf{G}$ *with the zero rows omitted:* $\hat{\mathbf{T}} = \mathbf{T}(\nu, :)$*. Correspondingly, define* $\hat{\mathbf{D}} = \mathbf{D}(\nu,\nu)$*,* $\hat{\mathbf{D}}' = \mathbf{D}'(\nu,\nu)$ *and* $\hat{\mathbf{A}} = \mathbf{A}(\nu, :)$*. The matrix* $\hat{\mathbf{T}}$ *can now be written as:*

$$\begin{aligned} \hat{\mathbf{T}} &= \hat{\mathbf{D}}'\hat{\mathbf{A}} && (2) \\ &= \hat{\mathbf{D}}\underbrace{\hat{\mathbf{D}}^{-1}\hat{\mathbf{D}}'\hat{\mathbf{A}}}_{\hat{\mathbf{W}}^{-1}} && (3) \end{aligned}$$

Inserting the zeros back in $\hat{\mathbf{T}}$*,* $\hat{\mathbf{D}}$ *and* $\hat{\mathbf{W}}$ *(*$\mathbf{W}(:,\nu) = \hat{\mathbf{W}}$*) gives*

$$\mathbf{D} = \mathbf{T}\mathbf{W} = \mathbf{C}^{-1}\mathbf{G}\mathbf{W} \tag{4}$$

Finally, to see this $\mathbf{C}^{-1}$ *is a Vandermonde matrix, consider Cook-Toom's decomposition with* N *and* L *both equal to* M*. In that case,* $\mathbf{C}$ *remains the same but* $\mathbf{A}$ *is extended and denoted as* $\tilde{\mathbf{A}}$*. It is clear that* $\mathbf{C}\tilde{\mathbf{A}} = \mathbf{I}_M$ *(*$\mathbf{C}\tilde{\mathbf{B}} = \mathbf{I}_M$*) and by definition* $\tilde{\mathbf{A}}$ *is a Vandermonde matrix where* $\tilde{\mathbf{A}}(m,n) = \rho_m^n (n = 0 : M-1)$*.*

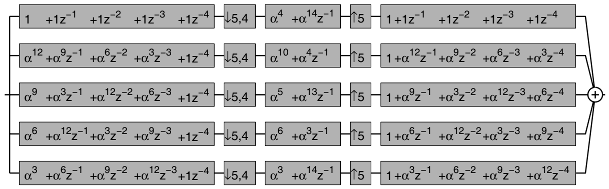
$1 \; +1z^{-1} \; +1z^{-2} \; +1z^{-3} \; +1z^{-4}$
$\downarrow 5,4$
$\alpha^4 \; +\alpha^{14}z^{-1}$
$\uparrow 5$
$1 +1z^{-1} \; +1z^{-2} \; +1z^{-3} \; +1z^{-4}$
$\alpha^{12} +\alpha^9 z^{-1} \; +\alpha^6 z^{-2} \; +\alpha^3 z^{-3} \; +1z^{-4}$
$\downarrow 5,4$
$\alpha^{10} +\alpha^4 z^{-1}$
$\uparrow 5$
$1 +\alpha^{12}z^{-1} +\alpha^9 z^{-2} \; +\alpha^6 z^{-3} \; +\alpha^3 z^{-4}$
$\alpha^9 \; +\alpha^3 z^{-1} \; +\alpha^{12}z^{-2} +\alpha^6 z^{-3} \; +1z^{-4}$
$\downarrow 5,4$
$\alpha^5 \; +\alpha^{13}z^{-1}$
$\uparrow 5$
$1 +\alpha^9 z^{-1} \; +\alpha^3 z^{-2} \; +\alpha^{12}z^{-3} +\alpha^6 z^{-4}$
$\alpha^6 \; +\alpha^{12}z^{-1} +\alpha^3 z^{-2} \; +\alpha^9 z^{-3} \; +1z^{-4}$
$\downarrow 5,4$
$\alpha^6 \; +\alpha^3 z^{-1}$
$\uparrow 5$
$1 +\alpha^6 z^{-1} \; +\alpha^{12}z^{-2} +\alpha^3 z^{-3} \; +\alpha^9 z^{-4}$
$\alpha^3 \; +\alpha^6 z^{-1} \; +\alpha^9 z^{-2} \; +\alpha^{12}z^{-3} +1z^{-4}$
$\downarrow 5,4$
$\alpha^3 \; +\alpha^{14}z^{-1}$
$\uparrow 5$
$1 +\alpha^3 z^{-1} \; +\alpha^6 z^{-2} \; +\alpha^9 z^{-3} \; +\alpha^{12}z^{-4}$
+

Fig. 1. Critically downsampled $(L = M = N = 5)$ filterbank for $\mathcal{R}(15, 10)$ code, found in example 3

Example 2 *As an example, let us take a* 5×5 *(*$M = 5$*) arbitrary diagonal matrix* $\mathbf{D}$ *with a zero in the second position (*$L = 2$*):* $\mathbf{D} = \textit{diag}([\alpha^2, 0, \alpha^2, 1, 1])$*. The roots used in example 1 are used again here. Since the zeros are at the same position compared to example 1, the same Toeplitz matrix* $\mathbf{G}$ *is found. Finally, the following decomposition is found using the construction given in the proof:*

$$\underbrace{\begin{bmatrix} \alpha^2 & & & & \\ & 0 & & & \\ & & \alpha^2 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}}_{\mathbf{D}} = \underbrace{\begin{bmatrix} 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha^8 \\ 1 & \alpha^5 & \alpha^{10} & 1 & \alpha^5 \\ 1 & \alpha^8 & \alpha^1 & \alpha^9 & \alpha^2 \\ 1 & \alpha^{11} & \alpha^7 & \alpha^3 & \alpha^{14} \\ 1 & \alpha^{14} & \alpha^{13} & \alpha^{12} & \alpha^{11} \end{bmatrix}}_{\tilde{\mathbf{A}}} \underbrace{\begin{bmatrix} \alpha^5 & & & \\ 1 & \alpha^5 & & \\ & 1 & \alpha^5 & \\ & & 1 & \alpha^5 \\ & & & 1 \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} \alpha^{12} & 0 & \alpha^{12} & \alpha^{10} & \alpha^{10} \\ \alpha^6 & 0 & \alpha^3 & \alpha^{12} & \alpha^3 \\ \alpha^{10} & 0 & \alpha^4 & \alpha^4 & \alpha^1 \\ \alpha^9 & 0 & 1 & \alpha^1 & \alpha^4 \end{bmatrix}}_{\mathbf{W}}$$

The next section shows how a filterbank can be constructed starting from Cook-Toom's algorithm.

3. CRITICALLY SAMPLED FIR FB IMPLEMENTING RS CODES

In [6], it is explained how a critically subsampled filterbank implementing a RS code is built using the following example:

Example 3 *Let* $\mathcal{R}(15, 10)$ *denote a* 15×10 *RS code [13]. It is a linear code in* $GF(q), q = 2^4$ *which encodes a data word* $u(z^{-1})$ *of* $k = 10$ *symbols into a codeword* $y(z^{-1})$ *of* $n = q - 1 = 15$ *symbols by filtering with* $g_{\mathcal{R}}(z^{-1}) = \prod_{k=3}^{7}(z^{-1} - \alpha^k)$*. More formally,*

$$\mathcal{R}(15, 10) = \{(u(z^{-1}), y(z^{-1}))|y(z^{-1}) = g_{\mathcal{R}}(z^{-1})u(z^{-1})\}$$

The method described in [6] can only be applied if the number of bands M divides $k = 10$ and $n = 15$. As a result, the filterbank obtained has $M = 5$ bands, as shown in **Figure 1**. If n and k are coprime, or e.g. if a filterbank with less bands is desired, the method in [6] fails to produce a suitable decomposition. In this paper, it is shown that critically subsampled filterbanks for RS codes with M bands exist as long as a subfield can be identified with M elements. Since each divisor d of $q - 1$ in $GF(q)$ defines a subfield with d elements (generated by $\alpha^{(q-1)/d}$), the number of bands should only be a divisor of $q - 1$. In the case of RS codes, $n = q - 1$ and thus M should divide n. The example of the $\mathcal{R}(15, 10)$ code is extended in this paper by providing an algebraic method to construct a filterbank with $M = 3$ bands. This filterbank must be *critically subsampled* in order to build a SISO decoder. This can briefly be explained as follows: Essentially, the filterbank is run in reverse order in the SISO decoder. Therefore, the subband samples are found starting from the codeword samples. This can only be done if there are as many subband samples as codeword samples, which holds if the filterbank is critically subsampled. A detailed discussion can be found in [6].

Next, an algebraic method is presented to construct a critically sampled filterbank implementing a (non-systematic) RS code. For

[1] The primitive polynomial is $x^4 + x + 1$, with α a primitive 15th root of unity.

[2] A matlab-like notation is adopted

the sake of an easy exposition, let us continue our simple, yet instructive example of the $\mathcal{R}(15,10)$. Although we limit our attention to this example, the technique presented here is generally applicable.

Example 4 *Let the number of bands* $M = 3$*. Since* M *divides* $n = 15$*, subfields with* $M = 3$ *and* $n' = n/M = 5$ *elements exist, defining the DFT matrices (in* $GF(2^4)$*)* $\mathbf{F}_3$ *and* $\mathbf{F}_5$*. Therefore, the* 15×15 *DFT matrix* $\mathbf{F}_{15}$ *can be decomposed with an FFT-like procedure into*

$$\mathbf{F}_{15} = \mathbf{\Pi}_3(\mathbf{I}_3 \otimes \mathbf{F}_5)\mathbf{\Pi}_5\mathbf{Q}_5(\mathbf{I}_5 \otimes \mathbf{F}_3)\mathbf{\Pi}_3. \tag{5}$$

In this equation, $\mathbf{\Pi}_x$ *is the interleaver matrix with* $\mathbf{\Pi}_x(i, i + (n-1)(i \mod x))/x) = 1$ *and* $\mathbf{I}_x$ *is the* $x \times x$ *identity matrix. Finally, the diagonal matrix*

$$\mathbf{Q_5} = \textit{diag}(\alpha^{[0:n'-1]\otimes[0:M-1]}). \tag{6}$$

contains the so-called "twiddle-factors". First, $\mathbf{G}_\mathcal{R}$ *can be diagonalized using a DFT decomposition (Cook-Toom's algorithm with roots* $\alpha^{0:14}$*):*

$$\mathbf{G}_\mathcal{R} = \mathbf{F}_{15}^{-1}\mathbf{\Delta}\mathbf{F}_{15}(:, 0:9) \tag{7}$$

Note that by the definition of a RS code, the diagonal $\mathbf{\Delta}$ *contains a block of* $n - k = 5$ *consecutive zeros:*

$$\begin{array}{rl} \textit{CT roots} \;\rightarrow & 1\ \alpha^1\alpha^2\alpha^3\alpha^4\alpha^5\alpha^6\alpha^7\alpha^8\ \alpha^9\ \alpha^{10}\alpha^{11}\alpha^{12}\alpha^{13}\alpha^{14} \\ \mathbf{\Delta} = & \textit{diag}([\alpha^2\alpha^6\alpha^2\ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \alpha^2\alpha^{11}\alpha^{12}\ \ 1\ \ \alpha^6\ \ \alpha^1\ 1]). \end{array}$$

For reasons that will become clear later, $\mathbf{\Delta}$ *is cyclically shifted by* $b = 2$ *steps (In general,* $b = n - k + b \mod M$*):*

$$\begin{array}{rl} \textit{CT roots} \;\rightarrow & \alpha^2\,\alpha^3\alpha^4\alpha^5\alpha^6\alpha^7\alpha^8\ \alpha^9\ \alpha^{10}\alpha^{11}\alpha^{12}\alpha^{13}\alpha^{14}\ 1\ \ \alpha^1 \\ \mathbf{\Delta}^{(2,2)} = & \textit{diag}([\alpha^2\ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \alpha^2\alpha^{11}\alpha^{12}\ \ 1\ \ \alpha^6\ \ \alpha^1\ \ 1\ \alpha^2\alpha^6]). \end{array}$$

From a notational point of view, a rotated $x \times x$ *matrix* $\mathbf{Y}^{(a,b)} = \mathbf{R}_x^{-b}\mathbf{Y}\mathbf{R}_x^a$ *with*

$$\mathbf{R}_x = \begin{bmatrix} \mathbf{O}_{1\times x-1} & 1 \\ \mathbf{I}_{x-1} & \mathbf{O}_{x-1,1} \end{bmatrix}. \tag{8}$$

Using this notation, the rotated DFT matrix

$$\begin{aligned} \mathbf{F}_{15}^{(a,b)} &= \mathbf{R}_{15}^{-b}\mathbf{F}_{15}\mathbf{R}_{15}^a & (9) \\ &= \mathbf{\Pi}_3(\mathbf{I}_3 \otimes \mathbf{F}_5^{(a,0)})\mathbf{\Pi}_5\mathbf{Q}_5^{(a,b)}(\mathbf{I}_5 \otimes \mathbf{F}_3^{(0,b)})\mathbf{\Pi}_3. & (10) \end{aligned}$$

In this equation, $\mathbf{Q}_5^{(a,b)}$ *is denoted as follows (slightly abusing our notation):*

$$\mathbf{Q}_5^{(a,b)} = \textit{diag}(\alpha^{(a+[0:n'-1])\otimes(b+[0:M-1])}). \tag{11}$$

Hence,

$$\mathbf{G}_\mathcal{R} = \left(\mathbf{F}_{15}^{(0,2)}\right)^{-1}\mathbf{\Delta}^{(2,2)}\mathbf{F}_{15}^{(0,2)}(:, 0:9) \tag{12}$$

With the filterbank structure in mind, $\mathbf{\Delta}^{(2,2)}$ *should be decomposed as* $\mathbf{S}\mathbf{G}_\mathcal{D}\mathbf{W}$*. Therefore, define* $M = 3$ *submatrices* $\hat{\mathbf{\Delta}}_m$*,* $m = 0 : M - 1$*:*

$$\begin{array}{l} \hat{\mathbf{\Delta}}_0 = \textit{diag}([\alpha^2\,0\ \alpha^2\ \ 1\ \ 1\ ])\ \textit{at CT roots}\ [\alpha^2\alpha^5\ \alpha^8\ \alpha^{11}\alpha^{14}] \\ \hat{\mathbf{\Delta}}_1 = \textit{diag}([\ 0\ \ 0\,\alpha^{11}\alpha^6\alpha^2])\ \textit{at CT roots}\ [\alpha^3\alpha^6\ \alpha^9\ \alpha^{12}\ \ 1\ ] \\ \hat{\mathbf{\Delta}}_2 = \textit{diag}([\ 0\ \ 0\,\alpha^{12}\alpha^1\alpha^6])\ \textit{at CT roots}\ [\alpha^4\alpha^7\alpha^{10}\alpha^{13}\ \alpha^1\ ] \end{array}$$

Note that only $\hat{\mathbf{\Delta}}_0$ *has only one zero diagonal element. Hence, Theorem 1 can be applied with the corresponding roots,* $N = 4$ *and* $L = 2$*, resulting in the following decomposition*

$$\hat{\mathbf{\Delta}}_0 = \hat{\mathbf{S}}_0\hat{\mathbf{G}}_{0,\mathcal{D}}\hat{\mathbf{W}}_0. \tag{13}$$

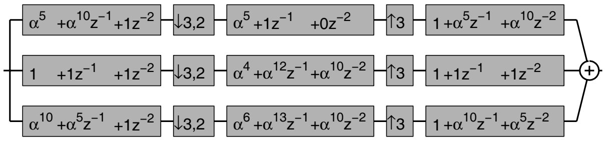

Fig. 2. Critically subsampled filterbank ($M = 3$) for the $\mathcal{R}(15,10)$ code described in example 4.

Note that $\hat{\mathbf{G}}_{0,\mathcal{D}}$ *is the* 5×4 *Toeplitz matrix, which was already given in Example 2. A similar decomposition holds for* $\hat{\mathbf{\Delta}}_1$ *and* $\hat{\mathbf{\Delta}}_2$*, where* $\hat{\mathbf{G}}_{1,\mathcal{D}}$ *and* $\hat{\mathbf{G}}_{2,\mathcal{D}}$ *are* 5×3 *Toeplitz matrices. Now, the reason why* $\mathbf{\Delta}$ *is first rotated to* $\mathbf{\Delta}^{(2,2)}$ *becomes clear: The Toeplitz matrices* $\hat{\mathbf{G}}_{m,\mathcal{D}}$ *are then ordered according to their size, starting with the largest. Substitution of the submatrices yields* $\mathbf{\Delta}^{(2,2)} = \mathbf{S}\mathbf{G}_\mathcal{D}\mathbf{W}$*. Looking at* $\mathbf{G}_\mathcal{D}$ *in more detail, it is seen that this matrix has a very specific structure (See also* ***Figure 3****):*

$$\begin{aligned} \mathbf{G}_\mathcal{D}(0:3:12, 0:3:9) &= \hat{\mathbf{G}}_{0,\mathcal{D}}\ (5 \times 4\ \textit{Black boxes}) & (14) \\ \mathbf{G}_\mathcal{D}(1:3:13, 1:3:7) &= \hat{\mathbf{G}}_{1,\mathcal{D}}\ (5 \times 3\ \textit{White boxes}) & (15) \\ \mathbf{G}_\mathcal{D}(2:3:14, 2:3:8) &= \hat{\mathbf{G}}_{2,\mathcal{D}}\ (5 \times 3\ \textit{Gray boxes}) & (16) \end{aligned}$$

Note that $\hat{\mathbf{S}}_m = \mathbf{F}_5\textit{diag}(\alpha^{(m+b)[0:4]})$*. Essentially,* $\mathbf{S}$ *consists of* M *torn apart* $\mathbf{F}_5$ *matrices, each multiplied by a diagonal matrix. Therefore, it is seen that*

$$\mathbf{S} = \mathbf{\Pi}_3(\mathbf{I}_3 \otimes \mathbf{F}_5)\mathbf{\Pi}_5\mathbf{Q}_5^{(0,2)}. \tag{17}$$

$$\begin{aligned} \mathbf{G}_\mathcal{R} &= \left(\mathbf{F}_{15}^{(0,2)}\right)^{-1}\mathbf{\Delta}^{(2,2)}\mathbf{F}_{15}^{(0,2)}(:, 0:9) & (18) \\ &= \left(\mathbf{F}_{15}^{(0,2)}\right)^{-1}\mathbf{S}\mathbf{G}_\mathcal{D}\mathbf{W}\mathbf{F}_{15}^{(0,2)}(:, 0:9) & (19) \\ &= \left(\mathbf{F}_{15}^{(0,2)}\right)^{-1}\mathbf{\Pi}_3(\mathbf{I}_3 \otimes \mathbf{F}_5)\mathbf{\Pi}_5\mathbf{Q}_5^{(0,2)}\mathbf{G}_\mathcal{D}\mathbf{W}\mathbf{F}_{15}^{(0,2)}(:, 0:9) & \\ &= \mathbf{\Pi}_3^{-1}\left(\mathbf{F}_{5\otimes 3}^{(0,2)}\right)^{-1}\mathbf{G}_\mathcal{D}\mathbf{W}\mathbf{F}_{15}^{(0,2)}(:, 0:9) & (20) \\ &= \mathbf{\Pi}_3^{-1}\underbrace{\left(\mathbf{I}_5 \otimes \mathbf{F}_3^{(0,2)}\right)^{-1}\mathbf{G}_\mathcal{D}(\mathbf{I}_5 \otimes \mathbf{F}_3)(0:9, 0:9)}_{\mathbf{G}_{\mathcal{R}^{\Pi_3,V}}}\mathbf{V} & (21) \end{aligned}$$

In the following, it is first explained that the factors of $\mathbf{G}_{\mathcal{R}^{\Pi_3,V}}$ *indeed represent a filterbank like structure. Secondly, it is shown that* $\mathbf{G}_{\mathcal{R}^{\Pi_3,V}}$ *implements a RS code.*

Figure 3 *shows that the factors of* $\mathbf{G}_{\mathcal{R}^{\Pi_3,V}}$ *indeed represent a filterbank like structure. The filterbank itself is shown in* ***Figure 2****. Consider only the first subband, which corresponds to the top-most row in* ***Figure 2****. The dataword is filtered in the analysis bank by* $\alpha^5 + \alpha^{10}z^{-1} + z^{-2}$ *(multiplied by the Toeplitz matrix* $(\mathbf{I}_5 \otimes \mathbf{I}_3)(0:3:9, 0:9)$*) followed by a 3-fold dowsampling resulting in 4 subband samples. These subband samples are padded with one zero, and fed into the subband filter* $\alpha^5 + z^{-1}$*. This is represented by the matrix* $\hat{\mathbf{G}}_{0,\mathcal{D}}$*, which is a downsampled version of* $\mathbf{G}_\mathcal{D}$ *(See Equation 13). Finally, in the synthesis bank, the samples are upsampled and filtered with* $1 + \alpha^5z^{-1} + \alpha^{10}z^{-2}$ *(represented by* $(\mathbf{I}_5 \otimes \mathbf{F}_3)^{-1}(:, 0:3:15)$*). Adding the corresponding outputs of the other bands gives the codeword. Due to the critical subsampling, there are as many subband samples as codeword samples. For decoding, it is now possible to operate the filterbank of* ***Figure 2*** *in reverse order. A detailed discussion is found in [7].*

Secondly, this filterbank implements an equivalent RS code denoted $\mathcal{R}^{\Pi_3,V}(15,10)$ *since each codeword is an interleaved version (*$\mathbf{\Pi}_3^{-1}$*) of a codeword of the original code* $\mathcal{R}(15,10)$*. Hence,*

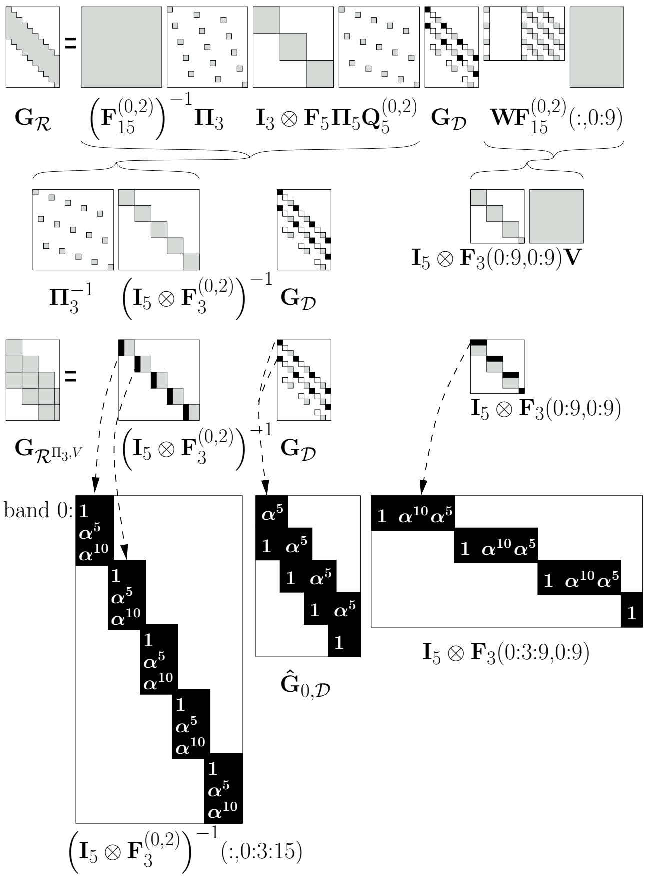
$\mathbf{G}_{\mathcal{R}}$
=
$\left(\mathbf{F}_{15}^{(0,2)}\right)^{-1}$
$\mathbf{\Pi}_3$
$\mathbf{I}_3 \otimes \mathbf{F}_5 \mathbf{\Pi}_5 \mathbf{Q}_5^{(0,2)}$
$\mathbf{G}_{\mathcal{D}}$
$\mathbf{W}\mathbf{F}_{15}^{(0,2)}(:,0{:}9)$
$\mathbf{\Pi}_3^{-1}$
$\left(\mathbf{I}_5 \otimes \mathbf{F}_3^{(0,2)}\right)^{-1}$
$\mathbf{G}_{\mathcal{D}}$
$\mathbf{I}_5 \otimes \mathbf{F}_3(0{:}9,0{:}9)\mathbf{V}$
$\mathbf{G}_{\mathcal{R}^{\Pi_3,V}}$
=
$\left(\mathbf{I}_5 \otimes \mathbf{F}_3^{(0,2)}\right)^{+1}$
$\mathbf{G}_{\mathcal{D}}$
$\mathbf{I}_5 \otimes \mathbf{F}_3(0{:}9,0{:}9)$
band 0:
1 α^5 α^{10}
$\left(\mathbf{I}_5 \otimes \mathbf{F}_3^{(0,2)}\right)^{-1}(:,0{:}3{:}15)$
α^5
1 α^5
1
$\hat{\mathbf{G}}_{0,\mathcal{D}}$
1 $\alpha^{10}\alpha^5$
$\mathbf{I}_5 \otimes \mathbf{F}_3(0{:}3{:}9,0{:}9)$

Fig. 3. Factorization of the generator matrix of $\mathcal{R}(15, 10)$. The 3 middle factors define a critically subsampled filterbank implementing the $\mathcal{R}^{\Pi_3,V}(15, 10)$ code.

the distance properties are preserved. The basis transformation matrix $\mathbf{V}$ *describes how the mapping between a specific dataword-codeword pair in* $\mathcal{R}(15, 10)$ *has changed. More formally,*

$$\mathcal{R}^{\Pi_3,V}(15,10) = \{(\mathbf{u},\mathbf{y})|(\mathbf{V}^{-1}\mathbf{u}, \mathbf{\Pi}_3^{-1}\mathbf{y}) \in \mathcal{R}(15,10)\} \quad (22)$$

Note also that the generator matrix $\mathbf{G}_{\mathcal{R}^{\Pi_3,V}}$ *has a special block Toeplitz structure with circulant* $M \times M$ *blocks. The cyclic behavior modulo 15 of the RS code is arranged using the FFT into a cyclic behavior modulo 3. Since* k *and* M *are coprime, the last column of* $\mathbf{G}_{\mathcal{R}^{\Pi_3,V}}$ *consists of rank 1 (in general* $\mod (k, M)$*) approximations of the corresponding circulant blocks.*

Finally, note that the roots of $\mathcal{R}(15, 10)$ $\alpha^3, \alpha^4, .., \alpha^7$ *are evenly distributed among the subbandfilters (see Equation 4). For the first band, the resulting subbandcode resembles the* $\mathcal{R}(5, 4)$ *code, with a free distance of 2. The codes corresponding to the other bands are similar to* $\mathcal{R}(5, 3)$ *with a free-distance of 3. It can be shown that these codes are projections onto subspaces of* $\mathcal{R}(15, 10)$*. However, a detailed discussion of these codes is out of the scope of this paper.*

4. CONCLUSION

In this paper, an algebraic construction is presented to build a critically subsampled filterbank realization for non-systematic RS codes. The key element is the exploitation of the cyclic properties of the RS code, in order to find a block Toeplitz generator matrix of an equivalent RS code with small circulant blocks. The core element is the FFT operation that transforms the cyclic character modulo the blocklength into a cyclic behavior modulo a divisor of this blocklength. This allows insight to be gained into the algebraic structure of the RS code and the corresponding filterbank realization. This lays the foundation that allows us to extend the methods presented earlier to RS codes where the codeword- and dataword length are coprime. With this algebraic construction, it is possible to build filterbanks (and their corresponding SISO decoders) for many RS codes encountered in standards, as well as for other codes, e.g. BCH codes.

5. REFERENCES

[1] I. Reed and G. Solomon, “Polynomial codes over certain finite fields,” *SIAM J.*, vol. 8, no. 2, pp. 300–304, june 1960.

[2] V. Guruswami and M. Sudan, “Improved decoding of reed-solomon and algebraic-geometric codes,” *IEEE Trans. Info. Theory*, vol. 45, pp. 1757–1767, 1999.

[3] U. Cheng and G.K. Huth, “Bounds on the bit error probability of a linear cyclic code over gf(2l) and its extended code,” *Information Theory, IEEE Transactions on*, vol. 34, no. 4, pp. 776 – 785, July 1988.

[4] R. Koetter and A. Vardy, “Algebraic soft-decision decoding of reed-solomon codes,” *Information Theory, IEEE Transactions on*, vol. 49, no. 11, pp. 2809 – 2825, Nov 2003.

[5] S. Benedetto, D. Divsalar, and J. Hagenauer, “Guest editorial concatenated coding techniques and iterative decoding: Sailing toward channel capacity,” *Selected Areas in Communications, IEEE Journal on*, vol. 16, no. 2, pp. 137 – 139, Feb 1998.

[6] G. Van Meerbergen, M. Moonen, and H. De Man, “Critically subsampled filterbanks implementing Reed-Solomon codes,” in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2004), Montreal, Canada*, May 17-21 2004, vol. 2, pp. 989–992.

[7] G. Van Meerbergen, M. Moonen, and H. De Man, “Turbo-like soft-decision decoding of reed-solomon codes,” in *Accepted for publication in Proc. of the IEEE Global Telecommunications Conference (Globecom 2004), Dallas, USA*, dec 2004.

[8] A. Scaglione, G.B. Giannakis, and S. Barbarossa, “Redundant filterbank precoders and equalizers. i. unification and optimal designs,” *Signal Processing, IEEE Transactions on*, vol. 47, no. 7, pp. 1988 – 2006, July 1999.

[9] F. Fekri, R.M. Mersereau, and R.W. Schafer, “Two-band wavelets and filterbanks over finite fields with connections to error control coding,” *Signal Processing, IEEE Transactions on*, vol. 51, no. 12, pp. 3143 – 3151, Dec 2003.

[10] A. Vardy and Y. Be’ery, “Bit-level soft-decision decoding of reed-solomon codes,” *Communications, IEEE Transactions on*, vol. 39, no. 3, pp. 440 –444, Mar. 1991.

[11] R.E. Blahut, *Fast algorithms for Digital Signal Processing*, Reading, MA: Addison-Wesley, 1984.

[12] M. Vetterli, “Running FIR and IIR filtering using multirate filter banks,” *Acoustics, Speech, and Signal Processing, IEEE Transactions on*, vol. 36, no. 5, pp. 730 –738, May 1988.

[13] N.J.A. Sloane and F.J. MacWilliams, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, 1977.