GROEBNER BASIS TECHNIQUES IN MULTIDIMENSIONAL MULTIRATE SYSTEMS

Ton Kalker, Hyungju Park and Martin Vetterli

University of California
231 Cory Hall
Berkeley, CA 94720
(kalker,apark,martin)@eecs.berkeley.edu

ABSTRACT

The Euclidean algorithm is a frequently used tool in the analysis of one-dimensional (1D) multirate systems. This tool is however not available for multidimensional (MD) multirate systems. In this paper we discus how Groebner basis techniques can fill this gap. After presenting the relevant facts about Groebner bases, we will show in a few examples how this technique can contribute to MD multirate systems theory.

1. Introduction

Recently there has been much research work done on multidimensional (MD) multirate systems. An overview can be found in [1]. However, some basic questions have been left untouched. In essence most of these questions boil down to the following question: given a matrix of polyphase components[1], can we effectively decide whether or not that matrix has a left inverse. The answer to this question is positive, and relies on Groebner bases techniques. Groebner bases (GB) were first introduced by Buchberger in his Ph.D. thesis and popularized by [2]. By now the importance of Buchberger's contribution has been appreciated in many fields, and many good books on the subject have appeared, notably [3]. Moreover, there are now many algebraic systems incorporating Groebner bases techniques: MATHEMATICA and MAPLE are two well known examples. In this paper we want to highlight the relevance of Groebner bases for MD multirate signal processing.

In many applications, the design and analysis of invertible MD multirate schemes amounts to a quest for the number "1". The following three questions will be used as examples of this paradigm.

1. Given an FIR low-pass filter $G(z)$, can we effectively decide whether or not $G(z)$ can occur as an analysis filter in a critically downsampled, 2-channel, perfect reconstructing (PR) FIR filter bank?
2. Given a sample rate conversion scheme consisting of upsampling by p, filtering with an FIR filter $U(z)$ and downsampling by q, can we effectively decide whether or not this scheme is FIR invertible?
3. Given an oversampled, MD FIR analysis filter bank, can we effectively find an FIR synthesis filter bank, such that the overall system is PR?

These questions give only a small sample of the set of MD problems for which Groebner bases are an essential tool. At this point we would like to mention the work of Park and Woodburn [4]. Using Groebner bases techniques Park and Woodburn have given a complete parametrization (in terms of ladder structures) of MD bi-orthogonal filter banks with 3 or more channels. Another example of an application of Groebner bases can be found in [5]. In this work the Groebner bases technique is an essential tool for the practical implementation of the Quillen-Suslin theorem.

In Section 2. we will derive a mathematical formulation for the three above problems, show how the number "1" is involved, and how to solve them in the 1D case. In Section 3. we introduce Groebner bases, and show how they can be used to solve the MD versions of these questions. Some toy examples will be included.

2. The One-dimensional Case

To answer the first question we decompose $G(z)$ into its polyphase components

$$G(z) = G_0(z^2) + z^{-1}G_1(z^2).$$

As noted in [6], the filter $G(z)$ occurs as the low-pass filter in a 2-channel PR filter bank if and only if there exists Laurent polynomials $H_0(z)$ and $H_1(z)$ such that

$$G_0(z)H_0(z) + G_1(z)H_1(z) = 1. \qquad (1)$$

Choosing an integer L such that $\hat{G}_0(z) = z^L G_0(z)$ and $\hat{G}_1(z) = z^L G_1(z)$ are anti-causal, and by multiplying the left and right hand side of Eq. (1) with a

[1] We will always assume that all our filters are FIR.

0-7803-2431-5/95 $4.00 © 1995 IEEE

sufficiently high power of z, the condition of Eq. (1) can be translated to the existence of *polynomials* $\alpha(z)$ and $\beta(z)$ and a integer M such that

$$\alpha(z)\hat{G}_0(z) + \beta(z)\hat{G}_1(z) = z^M. \qquad (2)$$

The condition and the construction of the polynomials $\alpha(z)$ and $\beta(z)$ are simultaneously solved by the Euclidean algorithm. In a sequence of division with remainder steps, the Euclidean algorithm computes a linear combination of the polynomials $\hat{G}_i(z)$ which equals the greatest common divisor of $\hat{G}_0(z)$ and $\hat{G}_1(z)$. If the greatest common divisor is anything else but a power of z, $G(z)$ can not act as the low-pass filter in a 2-channel filter bank. If the greatest common divisor is indeed a power of z, then the coefficient of the linear combination immediately allow the construction of a filter bank with $G(z)$ as low-pass filter.

To answer the second question we may assume that the numbers p and q are coprime, $p > q$. Let $U(z) = \sum_{i=0}^{pq-1} z^{-i}U_i(z^{pq})$ be its polyphase decomposition with respect to pq. Let the expression $U_{k,l}(z)$, $0 \le k < p$, $0 \le l < q$, be the polyphase component $U_i(z)$ such that $i \equiv k \pmod p$ and $i \equiv l \pmod q$. Invertibility of the sample rate conversion scheme can then conveniently be formulated as the existence of a Laurent matrix $\mathbf{D} = (D_{m,k}(z))$ which is a left inverse of $\mathbf{U} = (U_{k,l}(z))$. As one easily checks, the condition that $\mathbf{U}$ has a left inverse implies that there exist $\binom{p}{q}$ Laurent polynomials $D_i(z)$ such that

$$\sum_i D_i(z)M_i(z) = 1, \qquad (3)$$

where $M_i(z)$ ranges over the determinants of the maximal minors of $\mathbf{U}$. Moreover, Eq. (3) is also sufficient: in [7] it is shown how $\mathbf{D}$ can be derived from the data $D_i(z)$. Similar to the approach in the first question, the condition of Eq. (3) can be translated into a condition on polynomials. The solution can then be found using the Euclidean algorithm.

The third question is mathematically indistinguishable from the second. An oversampled filter bank corresponds to a non-square polyphase matrix, and the question is whether or not this question has a left inverse.

If we consider the three problems in their MD versions, the conditions which need to satisfied resemble their 1D counter parts Eq. (1) and Eq. (3). The only difference is that we have to interpret the variable z as a multivariable $(z_1, \cdots, z_n)$. However, the Euclidean algorithm is no longer valid, and thus the coefficients $\alpha(z)$, $\beta(z)$ and $D_i(z)$ and their existence have to be found in another way. At this point Groebner bases techniques enter the scene.

3. The Multidimensional Case

The Groebner Bases Algorithm (GBA) is a very powerful tool for solving a great variety of problems in multivariable polynomial theory. The general GBA determines for any set of multivariable polynomials $\{f_i(\boldsymbol{x})\}$ and any polynomial $h(\boldsymbol{x})$ whether or not $h(\boldsymbol{x})$ can be written as a linear combination

$$\sum_i \lambda_i(\boldsymbol{x})f_i(\boldsymbol{x}) = h(\boldsymbol{x}). \qquad (4)$$

Moreover, it finds an explicit set of polynomials $\{\lambda_i(\boldsymbol{x}\}$, if such a set exists.

In order to define Groebner basis, we first have to introduce the notion of *monomial order*. A monomial in $\mathbb{C}[\boldsymbol{x}] = \mathbb{C}[x_1, \ldots, x_m]$ is a power product of the form $x_1^{e_1}\cdots x_m^{e_m}$, and we denote by $T(x_1, \ldots, x_m)$, or simply by T, the set of all monomials in these variables. In the univariate case, there is a natural monomial order, that is,

$$1 < x < x^2 < x^3 < \cdots.$$

In the multivariate case, we define a monomial order $\le$ to be a linear order on T satisfying the following two conditions.

1. $1 \le t$ for all $t \in T$.
2. $t_1 \le t_2$ implies $t_1 \cdot s \le t_2 \cdot s$ for all $s, t_1, t_2 \in T$.

Once a monomial order is given, we can talk about the leading monomial, $\mathrm{lt}(f(\boldsymbol{x}))$, of $f(\boldsymbol{x}) \in \mathbb{C}[\boldsymbol{x}]$. It should be noted that, if we change the monomial order, then we may have a different $\mathrm{lt}(f(\boldsymbol{x}))$ for the same $f(\boldsymbol{x})$. Now, fix a monomial order on T, and let $I \subset \mathbb{C}[\boldsymbol{x}]$ be an ideal (i.e a set which includes all the elements which it can generate by taking linear combinations). Define $\mathrm{lt}(I)$ by

$$\mathrm{lt}(I) \;=\; \{\mathrm{lt}(f(\boldsymbol{x})) \mid f \in I\}.$$

Definition 1 $\{f_1(\boldsymbol{x}), \ldots, f_t(\boldsymbol{x}\} \subset I$ *is called a Groebner basis of* I *if*

$$(\mathrm{lt}(f_1(\boldsymbol{x})), \ldots, \mathrm{lt}(f_t(\boldsymbol{x}))) \;=\; \mathrm{lt}(I)$$

i.e. if the ideal generated by $\mathrm{lt}(f_1(\boldsymbol{x})), \ldots, \mathrm{lt}(f_t(\boldsymbol{x}))$ *coincides with* $\mathrm{lt}(I)$.

Example 1 Fix the degree lexicographic order on $\mathbb{C}[x, y]$, and let $I = (f(\boldsymbol{x}), g(\boldsymbol{x}))$, with $f(\boldsymbol{x}) = 1 - xy$ and $g(\boldsymbol{x}) = x^2$. Then the relation

$$(1 + xy)f(\boldsymbol{x}) + y^2 g(\boldsymbol{x}) = 1$$

implies that $I = \mathbb{C}[x, y]$, and therefore $(\mathrm{lt}(I) = \mathbb{C}[x, y]$ But $(\mathrm{lt}(f(\boldsymbol{x})), \mathrm{lt}(g(\boldsymbol{x}))) = (-xy, x^2) \subset (x)$. Therefore, $\{f(\boldsymbol{x}), g(\boldsymbol{x})\}$ is not a Groebner basis of the ideal I. □

The main reason that Groebner basis is useful for us comes from the following analogue of the Euclidean division algorithm.

Theorem 1 *(Division Algorithm)*
Let $\{\mathrm{lt}(f_1(\boldsymbol{x})), \ldots, \mathrm{lt}(f_t(\boldsymbol{x}))\} \subset \mathbb{C}[\boldsymbol{x}]$ *be a Groebner basis w.r.t. a fixed monomial order, and let* $h(\boldsymbol{x}) \in \mathbb{C}[\boldsymbol{x}]$*. Then there is an algorithm for writing* $h(\boldsymbol{x})$ *in the form*

$$h(\boldsymbol{x}) \;=\; \lambda_1(\boldsymbol{x})f_1(\boldsymbol{x}) + \cdots + \lambda_t(\boldsymbol{x})f_t(\boldsymbol{x}) + r(\boldsymbol{x})$$

such that $h(\boldsymbol{x}) \in I$ *if and only if* $r(\boldsymbol{x}) = 0$.

The polynomial $r(\boldsymbol{x})$ in the above is called the normal form of $f(\boldsymbol{x})$ w.r.t. $\{\mathrm{lt}(f_1(\boldsymbol{x})), \ldots, \mathrm{lt}(f_t(\boldsymbol{x}))\}$. Now, in order to solve our problem, we just compute the normal form of $h(\boldsymbol{x})$ w.r.t. the given set of polynomials (assuming that this set is a Groebner basis. Otherwise, we first have to transform it to another set of polynomials which is a Groebner basis. There is a standard algorithm for this transformation). If it is 0, then $h(\boldsymbol{x})$ can be written as a linear combination of the polynomials $f_i(\boldsymbol{x})$ and we have at the same time found the polynomials $\lambda_i(\boldsymbol{x})$.

For the purpose of this paper we are only interested in the case that the polynomial $h(\boldsymbol{x})$ equals the constant 1:

$$\sum_i \lambda_i(\boldsymbol{x})f_i(\boldsymbol{x}) = 1. \tag{5}$$

In that case, the theory also provides an easy parametrization of all solutions: $\{\lambda_i(\boldsymbol{x})\}$ is a solution of Eq. (5), then $\{\mu_i(\boldsymbol{x})\}$ is also a solution if and only if there exist polynomials $\{\alpha_i(\boldsymbol{x})\}$ such that

$$\mu_i(\boldsymbol{x}) - \lambda_i(\boldsymbol{x}) = \sum_j \alpha_j(\boldsymbol{x})(\delta_{i,j} - f_j(\boldsymbol{x})\lambda_i(\boldsymbol{x})). \tag{6}$$

Returning to our three problems, we see that we should be able to apply Groebner bases computations to solve them. There is however one catch: our questions involve *Laurent* polynomials, and not regular polynomials. This can however easily be overcome by a simple trick.

For every variable z_i we introduce two new variables x_i and y_i. Substituting x_i^m for every positive power z_i^m and y_i^k for every negative power z_i^{-k}, we transform the original set of Laurent polynomials into a set of regular polynomials. We then enlarge this set by adding the polynomials $x_iy_i - 1$. One verifies that the constant 1 is a linear combination of the original set of Laurent polynomials if and only if the same is true for the constructed set of regular polynomials. Moreover, given a linear combination of polynomials , we find a linear combination of Laurent polynomials by back substitution: x_i and y_i are replaced by z_i and z_i^{-1} respectively.

We now consider the first question. Applying the method outlined above, we can check whether or not a given MD low-pass $G(\boldsymbol{z})$ can act as an analysis filter in a 2-channel filter bank. Moreover, if the answer is yes, we explicitly find a particular low-pass synthesis filter $H_{part}(\boldsymbol{z})$. From Eq. (6) we also derive that any other solution $H_{gen}(\boldsymbol{z})$ is of the form

$$H_{gen}(\boldsymbol{z}) = (1 - \langle A(\boldsymbol{z})F(\boldsymbol{z})\rangle_0)H_{part}(\boldsymbol{z}) + A(\boldsymbol{z}). \tag{7}$$

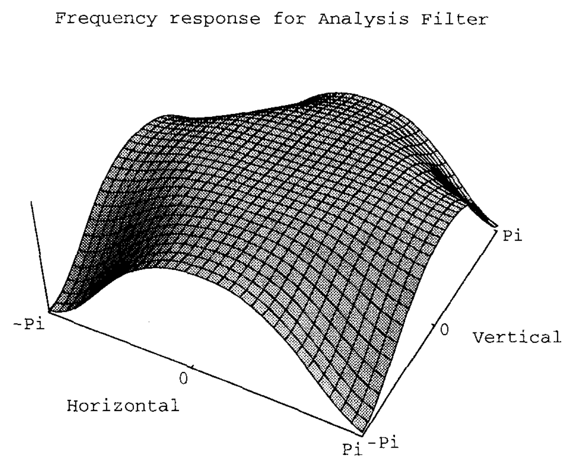

Figure 1: The frequency response of $G(z_1, z_2)$.

In this formula $A(\boldsymbol{z})$ is an arbitrary Laurent polynomial, and the expression $\langle . \rangle_0$ denotes the result of consecutive down- and upsampling with respect to the sampling lattice involved. Note that in the one-dimensional case, Eq. (7) reduces to a formula which can directly be derived from the Bezout identity (see [6] for more details).

Example 2 Consider the filter $G(z_1, z_2)$ with impulse response

$$\frac{1}{4096}\begin{pmatrix} 0 & 0 & 0 & 8 & 0 & 0 & 0 \\ 0 & 0 & 24 & -96 & 24 & 0 & 0 \\ 0 & 24 & -192 & 456 & -192 & 24 & 0 \\ 8 & -96 & 456 & 3200 & 456 & -96 & 8 \\ 0 & 24 & -192 & 456 & -192 & 24 & 0 \\ 0 & 0 & 24 & -96 & 24 & 0 & 0 \\ 0 & 0 & 0 & 8 & 0 & 0 & 0 \end{pmatrix}.$$

The filter $G(z_1, z_2)$ is designed to have a diamond-shaped low-pass frequency response. It is flat of order 2 at DC, and vanishing at the aliasing frequencies of the quincunx sampling lattice (see Fig. 3.). These properties make it a likely candidate for the low-pass analysis filter of a 2-channel, PR filter bank (downsampling on the quincunx lattice). Applying the GBA[2] we indeed

[2] We used MATHEMATICA and the Groebner package written by Garry Helzer, gah@math.umd.edu, for the actual computations.

find that this is the case, and the filter $H_{part}(z_1, z_2)$ with impulse response

$$\frac{1}{3585}\begin{pmatrix} 0 & 0 & 48 & 0 & 0 \\ 0 & 96 & 576 & 96 & 0 \\ 48 & 576 & 4288 & 576 & 48 \\ 0 & 96 & 576 & 96 & 0 \\ 0 & 0 & 48 & 0 & 0 \end{pmatrix}.$$

is found as a particular solution for the synthesis filter. By choosing an appropriate $A(z_1, z_2)$ as in Eq. (7) we can modify $H_{part}(z_1, z_2)$ in order to meet or approximate extra conditions. □

We will now consider the second question and show how Groebner bases can be used in 2D sample rate conversion schemes. As the purpose is to convey the method, and not to show our computational skills, only toy examples will be used.

Example 3 Consider the 2D sample rate conversion scheme which consists of vertical upsampling by a factor 3, filtering with a filter $H(\boldsymbol{z}) = H(z_1, z_2)$ and horizontal downsampling with a factor 2. We assume that H is FIR, and we would like to know if this scheme has an FIR inverse. To be more precise, we are looking for an FIR filter $G(\boldsymbol{z})$, such that horizontal upsampling by a factor 2, filtering with $G(\boldsymbol{z})$ and vertical downsampling with a factor 3, cancels the effect of the first sample rate conversion scheme.

Let the filter $H(\boldsymbol{z})$ be given by $H(\boldsymbol{z}) = \sum h_{i,j} z_1^{-i} z_2^{-j}$. Following the method outlined in Section 2., but now for this 2D case, we construct the 3×2 polynomial matrix $H_{k,l}(\boldsymbol{z}) = \sum h_{3i+k,2j+l} z_1^{-1} z_2^{-1}$, where $0 \leq k \leq 2$ and $0 \leq l \leq 1$.

Now assume momentarily that $H(\boldsymbol{z})$ is a separable filter $H^h(z_1)H^v(z_2)$. It is easily seen that in this case the filters $H_{k,l}(\boldsymbol{z})$ are products of 1D polyphase components, i.e. $H_{k,l}(\boldsymbol{z}) = H_k^h(z_1)H_l^v(z_2)$. Consequently, all the maximal minors of $H_{k,l}(\boldsymbol{z})$ have determinants equal to 0. Therefore the 2D analogue of Eq. 3 cannot be satisfied, and inversion is impossible.

Finally we consider an, admittedly slightly contrived, non-separable example, where the filter $H(\boldsymbol{z})$ is given by the 4×6 (*horizontal* $\times$ *vertical*) impulse response

$$\begin{pmatrix} 2 & 3 & 2 & 1 & 3 & 2 \\ 3 & 5 & 3 & 1 & 3 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 1 \end{pmatrix}.$$

The polyphase component matrix $H_{k,l}$ is then given by

$$(H_{k,l}) = \begin{pmatrix} 2+z_1^{-1}+z_2^{-1}+z_1^{-1}z_2^{-1} & 3+2z_1^{-1}+z_2^{-1}+z_1^{-1}z_2^{-1} \\ 3+z_1^{-1}+3z_2^{-1}+z_1^{-1}z_2^{-1} & 5+2z_1^{-1}+3z_2^{-1}+z_1^{-1}z_2^{-1} \\ 2+z_1^{-1}+2z_2^{-1}+z_1^{-1}z_2^{-1} & 3+2z_1^{-1}+2z_2^{-1}+z_1^{-1}z_2^{-1} \end{pmatrix}.$$

Computing the determinants of the maximal minors we find $D_0(\boldsymbol{z}) = -1 - z_2^{-1}$, $D_1(\boldsymbol{z}) = -z_2^{-1} - z_1^{-1}z_2^{-1}$ and $D_2(\boldsymbol{z}) = 1 - z_2^{-1} - z_1^{-1}z_2^{-1}$. These determinants are proper multivariable expressions and the Euclidean algorithm will therefore not work. This example is simple enough to be solved by ad hoc computations, but the GBA provides the generic method. In this case one easily verifies that $D_2 - D_1 = 1$ and therefore there exist an inverse FIR filter $G(\boldsymbol{z})$. To find $G(\boldsymbol{z})$ we first need to find a left inverse $G_{k,l}$ to $H_{k,l}$. Following the method in [7], the i^{th} row of this left inverse can be found by rewriting $\sum_i \lambda_i(\boldsymbol{z})D_i(\boldsymbol{z}) = 1$ in the form $\sum_j \beta_j H_{j,i} = 1$. The vector $(\beta_0, \beta_1, \beta_2)$ will then make up the i^{th} row of $G_{k,l}$. Finally, the matrix $G_{k,l}$ is related to the inverse filter $G(\boldsymbol{z})$ as the set of *backward* polyphase components. To be precise, $G(\boldsymbol{z})$ is given by $G(\boldsymbol{z}) = \sum z_1^k z_2^l G_{k,l}(z_1^2, z_2^3)$. Working out these formulas one finds the following impulse response for the filter $G(\boldsymbol{z})$:

$$\begin{pmatrix} -2 & 2 & -1 & -1 & 1 & -1 \\ 3 & -3 & 2 & 1 & -1 & 1 \\ -1 & 1 & 0 & -1 & 1 & 0 \\ 2 & -2 & 0 & 1 & -1 & 0 \end{pmatrix}$$

□

References

[1] T. Chen and P. Vaidyanathan, "Recent Developments in Multidimensional Multirate Systems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, pp. 116–137, April 1993.

[2] B. Buchberger, "Gröbner bases – An algorithm method in polynomial ideal theory," in *Recent Trends in Multidimensional Systems Theory* (N. Bose, ed.), ch. 6, pp. 184–232, Dordrecht: D. Reidel, 1985.

[3] D. Cox, J. Little, and D. O'Shea, *Ideals, Varieties and Algorithms.* Undergraduate Texts in Mathematics, Springer-Verlag, 1992. ISBN 0-387-97847-X.

[4] H. Park and C. Woodburn, "An Algorithmic Proof of Suslin's Stability Theorem for Polynomial Rings." To appear in the Journal of Algebra., 1994.

[5] S. Basu and H. M. Choi, "Multidimensional causal, stable, perfect reconstruction filter banks," in *Proceedings of the ICIP-94*, pp. 805–809, Nov. 1994.

[6] M. Vetterli and C. Herley, "Wavelets and Filter Banks: theory and design," *IEEE Transactions on Signal Processing*, vol. 40, pp. 2207–2232, Sept. 1992.

[7] A. Logar and B. Sturmfels, "Algorithms for the Quillen-Suslin Theorem," *Journal of Algebra*, vol. 145, pp. 231–239, 1992.