Zigzag Form over Families of Parametric Matrices

Robert M. Corless, Marc Moreno Maza and Steven E. Thornton
Ontario Research Centre for Computer Algebra, Western University, London, Canada

Overview

We present a method for computing the Zigzag form of parametric matrices under algebraic and semi-algebraic constraints. The Zigzag form was first introduced by Storjohann in [3], we adapt the same algorithm to parametric matrices.

This work was motivated by a desire to compute both the Frobenius and Jordan canonical forms of parametric matrices with applications in control systems. The Zigzag form itself is not directly useful for applications but provides a matrix from which the Frobenius form can easily be obtained.

Background Material

Let $\mathbb{K}$ be an algebraically closed field or a real closed field and $\mathbb{K}[\alpha] = \mathbb{K}[\alpha_1, \ldots, \alpha_m]$ the ring of polynomials in the variables $\alpha = \alpha_1, \ldots, \alpha_m$ with coefficients in $\mathbb{K}$. We denote by $\mathbb{K}(\alpha)$ the quotient field of $\mathbb{K}[\alpha]$, that is, the field of multivariate rational functions in α with coefficients in $\mathbb{K}$.

- **Proposition ([2, 1])**: For two constructible (resp. semi-algebraic) sets $S_1, S_2 \subset \mathbb{K}^m$, one can compute a triangular decomposition of their
  - intersection $S_1 \cap S_2$
  - union $S_1 \cup S_2$
  - set theoretical difference $S_1 \setminus S_2$.

- **Remark**: Let $S \subseteq \mathbb{K}^m$ be a constructible (resp. semi-algebraic) set and $f(\alpha) \in \mathbb{K}[\alpha]$. A useful tool for later computations is the partition $(S_{\text{eq}}, S_{\text{neq}})$ of S defined by

$$S_{\text{eq}} = S \cap V(f(\alpha)) \quad \text{and} \quad S_{\text{neq}} = S \setminus V(f(\alpha)) = S \setminus S_{\text{eq}}$$

by means of the algorithms of [2, 1].

Zigzag Matrix

- **Parametric Polynomial**: Let $f(x;\alpha)$ be a monic polynomial of degree r w.r.t. x. We write:

$$f(x;\alpha) = f_0(\alpha) + f_1(\alpha)x + \cdots + f_{r-1}(\alpha)x^{r-1} + x^r \qquad (1)$$

with coefficients $f_0(\alpha), \ldots, f_{r-1}(\alpha) \in \mathbb{K}(\alpha)$. The α values are constrained to belong to a constructible (resp. semi-algebraic) set S such that the denominator of every coefficient $f_0(\alpha), \ldots, f_{r-1}(\alpha)$ is nonzero everywhere on S.

- **Zigzag Matrix**: A parametric *Zigzag matrix* takes the form

$$\begin{bmatrix}
C_{c_1(x;\alpha)} & B_1 & & & & & & & \\
 & C^T_{c_2(x;\alpha)} & & & & & & & \\
 & B_2 & C_{c_3(x;\alpha)} & B_3 & & & & & \\
 & & & C^T_{c_4(x;\alpha)} & & & & & \\
 & & & & \ddots & & & & \\
 & & & & & C^T_{c_{k-2}(x;\alpha)} & & & \\
 & & & & & B_{k-2} & C_{c_{k-1}(x;\alpha)} & B_{k-1} & \\
 & & & & & & & C^T_{c_k(x;\alpha)}
\end{bmatrix}$$

for k even.

Each polynomial $c_1(x;\alpha), \ldots, c_k(x;\alpha)$ takes the same form as Equation (1) and each $C_{c_i(x;\alpha)}$ is a companion matrix of $c_i(x;\alpha)$. The B_i blocks have all entries zero except those in the upper left corner which are either 0 or 1; each B_i block has its size determined by its neighbouring companion blocks.

If there is an odd number of diagonal blocks we allow $\deg c_k = 0$ while $\deg c_i \geq 1$ for $1 \leq i < k$. This allows the kth diagonal block to have dimension zero and hence the block B_{k-1} above $C^T_{c_k(x;\alpha)}$ will also have dimension zero.

Computation

- **Theorem**: For every matrix $A(\alpha) \in \mathbb{K}^{n\times n}[\alpha]$, there exists a partition $(S_1, \ldots, S_N)$ of input constructible (resp. semi-algebraic) set$S$ such that for each S_i, there exists a matrix $Z_i(\alpha) \sim A(\alpha)$ in Zigzag form where the denominators of the coefficients of the entries of $Z_i(\alpha)$ are all non-zero everywhere on S_i.

We follow the same algorithm presented in Section 2 of [3]. Stages 1 and 3 must be modified for finding pivots vanishing nowhere on the underlying constructible (resp. semi-algebraic) set.

Once computation has split into two branches, one where a pivot has been found and the set S has been replaced with S_{neq}, and another where the pivot has not yet been found and the search for a pivot continues with S replaced by S_{eq}, the computations proceed in parallel (or by stack execution sequentially).

Implementation

A sequential implementation has been written in Maple to compute the set of Zigzag forms similar to an input parametric matrix under algebraic and semi-algebraic constraints.
This implementation extensively uses the `RegularChains` library, in particular the `ConstructibleSetTools` and `SemiAlgebraicSetTools` sub-packages are used for manipulating constructible sets and semi-algebraic sets respectively. See `http://regularchains.org` for details.

- **Example**: Consider the 3×3 matrix with a single parameter α over the complex numbers $\mathbb{C}$

$$A(\alpha) = \begin{bmatrix} -1 & -\alpha - 1 & 0 \\ -1/2 & \alpha - 2 & 1/2 \\ -2 & 3\alpha + 1 & -1 \end{bmatrix}$$

with no input conditions on α. The Zigzag forms similar to this matrix are

$$Z_1(\alpha) = \begin{bmatrix} 0 & 0 & 4\alpha \\ 1 & 0 & 4(\alpha - 1) \\ 0 & 1 & \alpha - 4 \end{bmatrix} \quad \alpha + 3 \neq 0$$

$$Z_2(\alpha) = \left[\begin{array}{cc|c} 0 & -4 & 1 \\ 1 & -4 & 0 \\ \hline 0 & 0 & -3 \end{array}\right] \quad \alpha + 3 = 0\,.$$

Clearly, $Z_1(\alpha)$ is already in Frobenius form whereas $Z_2(\alpha)$ requires additional work to obtain the Frobenius form. This example turns out to have a continuous Frobenius form in the parameter, hence the Frobenius form is $Z_1(\alpha)$ for all values of α.

The implementation code together with a variety of test examples of various levels of difficulty can be downloaded from `http://steventhornton.ca/coderepository/`. In particular, some examples are taken from the literature on control system design where parametric matrices arise.

References:

[1] Changbo Chen, James H. Davenport, John P. May, Marc Moreno Maza, Bican Xia, and Rong Xiao. Triangular decomposition of semi-algebraic systems. *J. Symb. Comput.*, 49:3–26, 2013.

[2] Victor G. Ganzha, Ernst W. Mayr, and Evgenii V. Vorozhtsov, editors. *Computer Algebra in Scientific Computing, 10th International Workshop, CASC 2007, Bonn, Germany, September 16-20, 2007, Proceedings*, volume 4770 of *Lecture Notes in Computer Science*. Springer, 2007.

[3] Arne Storjohann. An $\mathcal{O}(n^3)$ algorithm for the Frobenius normal form. In *Proceedings of the 1998 international symposium on Symbolic and algebraic computation*, pages 101–105. ACM, 1998.