Matrix Fraction Description of a Multivariable Square Control System by the Vu Algorithm

Ky M. Vu, President
AuLac Technologies Inc.
kymvu@aulactechnologies.com

Abstract

A new method to obtain the two matrix polynomials of a matrix fraction description of a square multivariable system is presented. The method gives an analytical solution in a closed form. This result is obtained by setting one matrix parameter to an identity matrix. For the matrix calculations leading to this result, the method uses an existing matrix algorithm to calculate the adjoint and determinant polynomials of a square matrix polynomial.

Keywords: *Markov Parameters, Matrix Fraction Description, State Space Model, Transfer Function Matrix, Vu Algorithm*

1. Introduction

Matrix computation constitutes a part of control theory practice. Of all the matrix computations, the computation of the inverse of a matrix polynomial is an essential computation. The author of this paper has a fair research on this topic. The initial research was on the determinant polynomial, which gives the characteristic coefficients of a characteristic equation [Vu99]. Then the computation of the adjoint polynomial was derived and published in [Vu7a], [Vu08], [STV09]. These two parallel algorithms constitute an algorithm to compute the inverse of a matrix polynomial. For the lack of a name to distinguish itself from other algorithms, let us call it the Vu algorithm. It is similar to the Faddeev-Leverrier algorithm that control engineers know of. The Faddeev-Leverrier algorithm is, however, only applicable to a resolvent, not a general polynomial matrix. The Vu algorithm would be complete and its impact will be much stronger if it can have an algorithm to obtain a square polynomial matrix or its adjoint polynomial when its determinant polynomial is given. It is the reverse algorithm of the existing algorithm. Usually, we cannot obtain a matrix when its determinant is given. This is true for both types of matrix: numeric or polynomial. If we, however, impose some conditions on the matrix, then we can obtain this matrix for our purpose. The matrix fraction description (MFD) of a square multivariable linear control system provides the framework for the reverse algorithm. It gives us the matrix polynomial when its determinant polynomial is given.

The topic of MFD has only been scantily researched and not adequately solved. This is probably the reason why the theory has not been clearly discussed in textbooks in control engineering. A good textbook, like that of [Kai80], devotes one whole chapter discussing the state-space and MFD models, but it does not provide a good algorithm to procure MFD models from a state space model. Another book, [Kac07], discusses the topic with special kinds of the denominator matrix in the sense that it is a scalar polynomial but scaled up to a matrix polynomial or a diagonal matrix polynomial. The structure theorem mentioned in [Wol74] also presents a special MFD for a controllable state space realization model. The author of [Wol74] has further research and coauthors an algorithm to obtain a left MFD in [WF69]. A result of [WF69] was compared in a newer research by [Pat81] with his own algorithm. His algorithm consists of three algorithms - one transforms a state space realization to a block Hessenberg form, one reduces a Hessenberg matrix to the Frobenius form and one extracts the MFD matrices - but claims similar result accuracy with the method of [WF69] and the transformation to the Luenberger canonical form of this method can be quite numerically sensitive. Technically, these two methods are common in a way: They are numerical methods operating on a state space realization. The orthogonal version of [WF69] was used with the Householder transformation by [DG92] on a minimal realization of a transfer function matrix and claimed that their proposed method is numerically less expensive than the algorithm given by [Pat81]. These researchers also used the Householder transformation on a Sylvester resultant matrix, [DG93], to improve numerical robustness for the approaches that use this matrix. A different approach was suggested by [ZDL09] and a Stein matrix equation has to be solved. But the solution of the Stein matrix equation can be numerical as well. Any numerical method that claims that it can obtain all the matrix parameters of the two matrix polynomials in a matrix fraction description without imposing any condition on these matrix polynomials only flatters itself with a non-unique solution. Perhaps the most recent research on MFD is given by [MCWY14], but this work is about uniqueness of the model and its identification from input data rather than conversion from a state space model or transfer function matrix. These topics have been addressed in [Vu7b].

The MFD problem is actually not a very difficult problem. Its solution can be obtained analytically in closed form if a preferred unique solution is determined first. This paper is written to present this fact. The paper is organized as follows. Section I is the introduction section. In Section II, we present the existing algorithms to compute the adjoint and determinant polynomials of a square matrix polynomial of the Vu algorithm. In Section III, we present algorithms to obtain the matrix polynomials of MFDs of different linear feedback systems. In Section IV, we consider an example, and section V concludes the discussion of the paper.

2. The Vu Algorithm

In this section, we review an existing matrix algorithm of the author. This algorithm consists of two algorithms. These two algorithms are used to calculate some quantities in the procurement of the two MFD matrix polynomials. Suppose that we have a square polynomial matrix

$$\mathbf{C}(\mu) \quad = \quad \sum_{i=0}^{r} \mathbf{C}_i \mu^i$$

with n as the dimension of the matrices $\mathbf{C}_i$'s, and we want to calculate its inverse.

In [Vu7a], the author gives two parallel algorithms to calculate the adjoint and determinant polynomials of the matrix $\mathbf{C}(\mu)$. The inverse matrix is obtained as the ratio of the adjoint and determinant polynomials.

The adjoint polynomial is

$$\mathbf{C}^+(\mu) \quad = \quad \sum_{i=0}^{(n-1)r} \mathbf{C}_i^+ \mu^i,$$

and the determinant polynomial is

$$d(\mu) \quad = \quad \sum_{i=0}^{n\times r} d_i \mu^i.$$

The matrix coefficient $\mathbf{C}_k^+$ of the adjoint polynomial is obtained as

$$\mathbf{C}_k^+ \quad = \quad \sum_{i=0}^{n-1}\sum_{j=0}^{k} (-1)^i \, q_{n-1-i,k-j}(\mathbf{C}_i's) \left.\frac{d^j \mathbf{C}^i(\mu)}{j! d\mu^j}\right|_{\mu=0}.$$

The derivative on the right hand side of the last equation can be calculated recursively as follows.

$$\left.\frac{d^j \mathbf{C}^i(\mu)}{j! d\mu^j}\right|_{\mu=0} \quad = \quad \sum_{k=1}^{i}\sum_{l=1}^{j} \left.\frac{d^{j-l}\mathbf{C}^{i-k}(\mu)}{(j-l)! d\mu^{j-l}}\right|_{\mu=0} \mathbf{C}_l \mathbf{C}_0^{k-1}$$

with the initial condition

$$\left.\frac{d^0 \mathbf{C}^i(\mu)}{d\mu^0}\right|_{\mu=0} \quad = \quad \mathbf{C}_0^i.$$

The coefficient $q_{m,k}(\mathbf{C}_i's)$ is calculated recursively as

$$q_{m,k}(\mathbf{C}_i's) = \frac{1}{k!}\left.\frac{d^k p_m(\mathbf{C}(\mu))}{d\mu^k}\right|_{\mu=0}, \quad k = 0, \cdots m\times r$$

$$= \frac{1}{m}\sum_{i=1}^{m}\sum_{j=0}^{k} (-1)^{i-1} q_{m-i,k-j}(\mathbf{C}_i's) \; p_1\left.\left(\frac{d^j \mathbf{C}^i(\mu)}{j! d\mu^j}\right)\right|_{\mu=0}$$

with the initial condition

$$q_{0,0}(\mathbf{C}_i's) = 1.$$

The coefficients $p_i(\mathbf{C})$'s of the constant matrix $\mathbf{C}$ are the characteristic coefficients in the characteristic equation

$$|\lambda \mathbf{I} - \mathbf{C}| = \lambda^n - p_1(\mathbf{C})\lambda^{n-1} + p_2(\mathbf{C})\lambda^{n-2}\cdots(-1)^n p_n(\mathbf{C}).$$

They are related as follows

$$\begin{aligned} p_m(\mathbf{C}) &= \frac{1}{m}\sum_{i=1}^{m}(-1)^{i-1} p_{m-i}(\mathbf{C}) p_1(\mathbf{C}^i), \quad m = 1, \cdots n, \\ p_0(\mathbf{C}) &= 1, \\ p_1(\mathbf{C}) &= trace\,(\mathbf{C}), \\ p_n(\mathbf{C}) &= |\mathbf{C}|. \end{aligned}$$

For the case of our polynomial matrix $\mathbf{C}(\mu)$, we have

$$p_m(\mathbf{C}(\mu)) \quad = \quad \sum_{i=0}^{m\times r} q_{m,i}(\mathbf{C}_i's)\mu^i.$$

This implies that the determinant polynomial is given as below.

$$
\begin{aligned}
d(\mu) &= p_n(\mathbf{C}(\mu)), \\
&= \sum_{i=0}^{n\times r} q_{n,i}(\mathbf{C}_i' s)\mu^i, \\
&= \sum_{i=0}^{n\times r} d_i \mu^i.
\end{aligned}
$$

And therefore we have

$$
d_i = q_{n,i}(\mathbf{C}_i' s).
$$

The coefficients $q_{n,i}(\mathbf{C}_i' s)$'s are calculated by the recurrence formula given above.

3. Matrix Fraction Descriptions

We will present matrix fraction description algorithms for both discrete and continuous control systems.

3.1. Discrete Control Systems

We discuss the discrete control system model first because it is the less complicated of the two control systems. Consider the discrete control system

$$
\begin{aligned}
\mathbf{x}_{t+1} &= \mathbf{A}_d \mathbf{x}_t + \mathbf{B}_d \mathbf{u}_{t-f}, \\
\mathbf{y}_t &= \mathbf{C}_d \mathbf{x}_t.
\end{aligned}
$$

By assuming that $\mathbf{x}_0 = \mathbf{0}$ and by using the z-transform, we can obtain the following equation

$$
\begin{aligned}
\mathbf{y}(z^{-1}) &= \mathbf{C}_d[\mathbf{I} - \mathbf{A}_d z^{-1}]^{-1}\mathbf{B}_d z^{-f-1}\mathbf{u}(z^{-1}), \\
&= \mathbf{C}_d[\mathbf{I} - \mathbf{A}_d z^{-1}]^{-1}\mathbf{B}_d \mathbf{u}^*(z^{-1}).
\end{aligned}
$$

When the system is square, ie. the dimensions of the vectors $\mathbf{y}_t$ and $\mathbf{u}_t$ are the same, we wish to write the model of the linear system as follows:

$$
\begin{aligned}
\mathbf{y}(z^{-1}) &= \mathbf{C}_d[\mathbf{I} - \mathbf{A}_d z^{-1}]^{-1}\mathbf{B}_d \mathbf{u}^*(z^{-1}), \\
&= \mathbf{D}_L(z^{-1})^{-1}\mathbf{N}_L(z^{-1})\mathbf{u}^*(z^{-1}), \\
&= \mathbf{N}_R(z^{-1})\mathbf{D}_R(z^{-1})^{-1}\mathbf{u}^*(z^{-1}).
\end{aligned}
$$

The model of the system is written like a fraction of two polynomials. Since they are matrices, the denominator must be written as an inverse; and the description of the model is named the matrix fraction description (MFD). It is a more parsimonious model, ie. it uses less parameters, compared to the state space model. There are two representations for a system depending on the position of the inverse polynomial matrix: We have a left MFD if the inverse is the premultiplying matrix and a right MFD if the inverse is the postmultiplying matrix.

For a left MFD of the system, we write

$$
\mathbf{D}_L(z^{-1})^{-1}\mathbf{N}_L(z^{-1}) = \frac{\mathbf{C}_d[\mathbf{I} - \mathbf{A}_d z^{-1}]^{+}\mathbf{B}_d}{|\mathbf{I} - \mathbf{A}_d z^{-1}|}.
$$

From the last equation, we can obtain

$$
\frac{\mathbf{D}_L(z^{-1})^{+}\mathbf{N}_L(z^{-1})}{|\mathbf{D}_L(z^{-1})|} = \frac{\mathbf{C}_d[\mathbf{I} - \mathbf{A}_d z^{-1}]^{+}\mathbf{B}_d}{|\mathbf{I} - \mathbf{A}_d z^{-1}|}. \quad (1)
$$

By multiplying both sides of the last equation with the scalar polynomial $|\mathbf{I} - \mathbf{A}_d z^{-1}|$, the right hand side ceases to be a fraction. This must also be true for the left hand side of the equation, which means that we must have

$$
|\mathbf{D}_L(z^{-1})| = |\mathbf{I} - \mathbf{A}_d z^{-1}|.
$$

The polynomials on the right hand side of Eq. (1) can be obtained by the Vu algorithm. So the adjoint polynomial $\mathbf{D}_L(z^{-1})^{+}$ on the left hand side can also be calculated. This means that we can calculate the adjoint polynomial when the determinant polynomial is given. Our purpose is, however, the calculation of the polynomial $\mathbf{D}_L(z^{-1})$, not its adjoint. Therefore, instead of using Eq. (1), we write

$$
|\mathbf{I} - \mathbf{A}_d z^{-1}|\mathbf{N}_L(z^{-1}) = \mathbf{D}_L(z^{-1})\mathbf{C}_d[\mathbf{I} - \mathbf{A}_d z^{-1}]^{+}\mathbf{B}_d
$$

or

$$
|\mathbf{D}_L(z^{-1})|\mathbf{N}_L(z^{-1}) = \mathbf{D}_L(z^{-1})\mathbf{E}(z^{-1}). \quad (2)
$$

Assuming that the dimension of the state vector is n and the dimensions of the control ($\mathbf{u}_t$) and controlled variables ($\mathbf{y}_t$) vectors are $m < n$, then we can state the following facts:

1. The degree of the polynomial $\mathbf{E}(z^{-1})$ is $-(n-1)$, and the degree of the polynomial $|\mathbf{D}_L(z^{-1})|$ is $-n$.
2. The degree of the matrix polynomial $\mathbf{D}_L(z^{-1})$ is $-r = -[n/m]$ where the square brackets symbol means the upward-rounded integer of.
3. The degree of the matrix polynomial $\mathbf{N}_L(z^{-1})$ is one less than that of $\mathbf{D}_L(z^{-1})$, ie. $-(r-1)$.

The known parameters, calculated by the Vu algorithm, in the last equation are as follows:

$$
\begin{aligned}
|\mathbf{D}_L(z^{-1})| &= d_0 + d_1 z^{-1} + \cdots + d_n z^{-n}, \\
\mathbf{E}(z^{-1}) &= \mathbf{E}_0 + \mathbf{E}_1 z^{-1} + \cdots + \mathbf{E}_{n-1} z^{-n+1}.
\end{aligned}
$$

The unknown parameters are

$$
\begin{aligned}
\mathbf{D}_L(z^{-1}) &= \mathbf{D}_0 + \mathbf{D}_1 z^{-1} + \cdots + \mathbf{D}_r z^{-r}, \\
\mathbf{N}_L(z^{-1}) &= \mathbf{N}_0 + \mathbf{N}_1 z^{-1} + \cdots + \mathbf{N}_{r-1} z^{-r+1}.
\end{aligned}
$$

We have said that any numerical method claiming that it can obtain all the unknown matrix parameters of the two matrix polynomials $\mathbf{D}_L(z^{-1})$ and $\mathbf{N}_L(z^{-1})$ is false. Now we will prove it. Consider the relation of the matrix constants on both sides of Eq. (2), we have

$$d_0\mathbf{N}_0 \;=\; \mathbf{D}_0\mathbf{E}_0.$$

Only d_0 and $\mathbf{E}_0$ are given, so there are many values of $\mathbf{N}_0$ and $\mathbf{D}_0$ satisfying the last equation. To have a unique solution, we need to impose some conditions on the unknown parameters. We can find that the constant of the determinant polynomial $|\mathbf{I}-\mathbf{A}_d z^{-1}|$ is always equal to one, ie. we have

$$d_0 \;=\; 1.$$

If we want to have

$$|\mathbf{D}_L(z^{-1})| \;=\; |\mathbf{I}-\mathbf{A}_d z^{-1}|,$$

then we have the condition $|\mathbf{D}_0| = 1$. The easiest choice for us to achieve this condition, with a good physical meaning of the result, is to set

$$\mathbf{D}_0 \;=\; \mathbf{I}.$$

Now we are ready to solve for the unknown parameters. We write Eq. (2) as

$$(1+\sum_{i=1}^{n} d_i z^{-i})\sum_{j=0}^{r-1}\mathbf{N}_j z^{-j} \;=\; (\mathbf{I}+\sum_{i=1}^{r}\mathbf{D}_i z^{-i})\sum_{j=0}^{n-1}\mathbf{E}_j z^{-j}.$$

By taking the transpose of the above equation, we obtain

$$(1+\sum_{i=1}^{n} d_i z^{-i})\sum_{j=0}^{r-1}\mathbf{N}_j^T z^{-j} \;=\; \sum_{j=0}^{n-1}\mathbf{E}_j^T z^{-j}(\mathbf{I}+\sum_{i=1}^{r}\mathbf{D}_i^T z^{-i}).$$

By equating the coefficients with the same powers of z on both sides of the last equation and collect them in a matrix equation, we can write

$$\begin{bmatrix} \mathbf{I} & & & \\ d_1\mathbf{I} & \mathbf{I} & & \\ \vdots & \vdots & \ddots & \\ \vdots & \vdots & \ddots & \mathbf{I} \\ & & & d_1\mathbf{I} \\ & & \ddots & \vdots \\ & & & d_n\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{N}_0^T \\ \mathbf{N}_1^T \\ \vdots \\ \mathbf{N}_{r-1}^T \end{bmatrix} =$$

$$\begin{bmatrix} \mathbf{E}_0^T & & & \\ \mathbf{E}_1^T & \mathbf{E}_0^T & & \\ \vdots & \vdots & \ddots & \\ \vdots & \vdots & \ddots & \mathbf{E}_0^T \\ & & & \mathbf{E}_1^T \\ & & \ddots & \vdots \\ & & & \mathbf{E}_{n-1}^T \end{bmatrix} \begin{bmatrix} \mathbf{I} \\ \mathbf{D}_1^T \\ \vdots \\ \mathbf{D}_r^T \end{bmatrix}.$$

By moving all the unknown parameters to the left hand side, we can write the last equation as

$$\mathbf{G}_L \begin{bmatrix} \mathbf{N}_0^T \\ \mathbf{N}_1^T \\ \vdots \\ \mathbf{N}_{r-1}^T \\ \mathbf{D}_1^T \\ \vdots \\ \mathbf{D}_r^T \end{bmatrix} = \begin{bmatrix} \mathbf{E}_0^T \\ \mathbf{E}_1^T \\ \vdots \\ \mathbf{E}_{n-1}^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}.$$

with

$$\mathbf{G}_L = \begin{bmatrix} \mathbf{I} & & & & \mathbf{0} & & \\ d_1\mathbf{I} & \mathbf{I} & & & -\mathbf{E}_0^T & & \\ \vdots & \vdots & \ddots & & \vdots & & \\ \vdots & \vdots & \ddots & \mathbf{I} & & \vdots & \mathbf{0} \\ & & & d_1\mathbf{I} & & & -\mathbf{E}_0^T \\ & & \ddots & \vdots & & & \vdots \\ & & & d_n\mathbf{I} & & & -\mathbf{E}_{n-1}^T \end{bmatrix}.$$

Then the values of the unknown matrix parameters are given by

$$\begin{bmatrix} \mathbf{N}_0^T \\ \mathbf{N}_1^T \\ \vdots \\ \mathbf{N}_{r-1}^T \\ \mathbf{D}_1^T \\ \vdots \\ \mathbf{D}_r^T \end{bmatrix} = [\mathbf{G}_L^T\mathbf{G}_L]^{-1}\mathbf{G}_L^T \begin{bmatrix} \mathbf{E}_0^T \\ \mathbf{E}_1^T \\ \vdots \\ \mathbf{E}_{n-1}^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \tag{3}$$

Remark:

Note that if we do not choose $\mathbf{D}_0 = \mathbf{I}$ and we obtain a set of correct parameter matrices for the system, then we can obtain the following relation for this set of parameters:

$$\begin{bmatrix} \hat{\mathbf{N}}_0^T \\ \hat{\mathbf{N}}_1^T \\ \vdots \\ \hat{\mathbf{N}}_{r-1}^T \\ \hat{\mathbf{D}}_1^T \\ \vdots \\ \hat{\mathbf{D}}_r^T \end{bmatrix} = [\mathbf{G}_L^T\mathbf{G}_L]^{-1}\mathbf{G}_L^T \begin{bmatrix} \mathbf{E}_0^T \\ \mathbf{E}_1^T \\ \vdots \\ \mathbf{E}_{n-1}^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \hat{\mathbf{D}}_0^T.$$

This means that we have

$$\begin{aligned} \hat{\mathbf{D}}_i &= \hat{\mathbf{D}}_0\mathbf{D}_i, \\ \hat{\mathbf{N}}_i &= \hat{\mathbf{D}}_0\mathbf{N}_i. \end{aligned}$$

There are many sets of correct parameter matrices for a system, the parameter matrices of each set are the parameter matrices given by Eq. (3) premultiplied by the constant matrix of the denominator polynomial.

3.2. Continuous Control Systems

For continuous systems, we consider two separate cases corresponding to two types of system: strictly proper and proper.

3.2.1. Strictly Proper Systems. We consider the continuous state space model of a linear system

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t). \end{aligned}$$

By taking the Laplace transforms of both sides of the above equations and assuming that $\mathbf{x}(0) = \mathbf{0}$, we get

$$\begin{aligned} s\mathbf{x}(s) &= \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s), \\ \mathbf{y}(s) &= \mathbf{C}\mathbf{x}(s). \end{aligned}$$

Then we can write

$$\begin{aligned} [s\mathbf{I} - \mathbf{A}]\mathbf{x}(s) &= \mathbf{B}\mathbf{u}(s), \\ \mathbf{x}(s) &= [s\mathbf{I} - \mathbf{A}]^{-1}\mathbf{B}\mathbf{u}(s), \\ \mathbf{y}(s) &= \mathbf{C}[s\mathbf{I} - \mathbf{A}]^{-1}\mathbf{B}\mathbf{u}(s). \end{aligned}$$

For a left MFD model, we have

$$\begin{aligned} \mathbf{D}_L(s)^{-1}\mathbf{N}_L(s) &= \mathbf{C}[s\mathbf{I} - \mathbf{A}]^{-1}\mathbf{B}, \\ &= s^{-1}\mathbf{C}[\mathbf{I} - \mathbf{A}s^{-1}]^{-1}\mathbf{B}, \\ &= s^{-1}\frac{\mathbf{C}[\mathbf{I} - \mathbf{A}s^{-1}]^{+}\mathbf{B}}{|\mathbf{I} - \mathbf{A}s^{-1}|}, \\ &= s^{-1}\frac{\mathbf{E}(s^{-1})}{|\mathbf{I} - \mathbf{A}s^{-1}|} \end{aligned}$$

with

$$\mathbf{E}(s^{-1}) = \mathbf{E}_0 + \mathbf{E}_1 s^{-1} + \cdots + \mathbf{E}_{n-1}s^{-n+1}.$$

The existence of the factor s^{-1} in the second last equation is an indication of a strictly proper system. We can see that we can obtain a left MFD for the term $\mathbf{C}[\mathbf{I} - \mathbf{A}s^{-1}]^{-1}\mathbf{B}$ on the right hand side of the last equation with the algorithm used for a discrete control system. The obtained solution, from Eq. (3), is given below

$$\begin{aligned} \mathbf{D}_L(s^{-1}) &= \mathbf{I} + \mathbf{D}_1 s^{-1} + \cdots + \mathbf{D}_r s^{-r}, \\ \mathbf{N}_L(s^{-1}) &= \mathbf{N}_0 + \mathbf{N}_1 s^{-1} + \cdots + \mathbf{N}_{r-1}s^{-r+1}. \end{aligned}$$

With this result, we can write

$$\mathbf{D}_L(s)^{-1}\mathbf{N}_L(s) = s^{-1}\mathbf{D}_L(s^{-1})^{-1}\mathbf{N}_L(s^{-1})$$

or

$$\begin{aligned} \mathbf{D}_L(s)^{-1}\mathbf{N}_L(s) &= s^{-1}s^{-r+1}\mathbf{D}_L(s^{-1})^{-1}s^{r-1}\mathbf{N}_L(s^{-1}), \\ &= [s^r\mathbf{D}_L(s^{-1})]^{-1}[s^{r-1}\mathbf{N}_L(s^{-1})], \\ &= \mathbf{D}_L(s)^{-1}\mathbf{N}_L(s). \end{aligned}$$

The left MFD for a strictly proper continuous model consists of the following matrix polynomials:

$$\begin{aligned} \mathbf{D}_L(s) &= s^r\mathbf{D}_L(s^{-1}), \\ &= \mathbf{I}s^r + \mathbf{D}_1 s^{r-1} + \cdots + \mathbf{D}_r, \\ \mathbf{N}_L(s) &= s^{r-1}\mathbf{N}_L(s^{-1}), \\ &= \mathbf{N}_0 s^{r-1} + \mathbf{N}_1 s^{r-2} + \cdots + \mathbf{N}_{r-1}. \end{aligned}$$

The parameter matrices are obtained from Eq. (3), with the appropriate matrices $\mathbf{G}_L$ and $\mathbf{E}_i$'s calculated by the Vu algorithm from the state-space model matrices.

3.2.2. Proper Systems. For the proper system

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t), \end{aligned}$$

the left MFD can be written as

$$\begin{aligned} \mathbf{D}_L(s)^{-1}\mathbf{N}_L(s) &= s^{-1}\mathbf{C}[\mathbf{I} - \mathbf{A}s^{-1}]^{-1}\mathbf{B} + \mathbf{D}, \\ &= \frac{s^{-1}\mathbf{C}[\mathbf{I} - \mathbf{A}s^{-1}]^{+}\mathbf{B}}{|\mathbf{I} - \mathbf{A}s^{-1}|} + \mathbf{D}, \\ &= \frac{s^{-1}\mathbf{C}[\mathbf{I} - \mathbf{A}s^{-1}]^{+}\mathbf{B}}{|\mathbf{I} - \mathbf{A}s^{-1}|} + \frac{|\mathbf{I} - \mathbf{A}s^{-1}|\mathbf{D}}{|\mathbf{I} - \mathbf{A}s^{-1}|}, \\ &= \frac{\mathbf{E}(s^{-1})}{|\mathbf{I} - \mathbf{A}s^{-1}|}. \end{aligned}$$

Since the system is only proper, the factor s^{-1} disappears in the final expression for $\mathbf{E}(s^{-1})$. The matrix polynomial $\mathbf{E}(s^{-1})$ has degree $-n$ and is given by

$$\begin{aligned} \mathbf{E}(s^{-1}) &= \mathbf{E}_0 + \mathbf{E}_1 s^{-1} + \cdots + \mathbf{E}_n s^{-n}, \\ &= s^{-1}\mathbf{C}[\mathbf{I} - \mathbf{A}s^{-1}]^{+}\mathbf{B} + |\mathbf{I} - \mathbf{A}s^{-1}|\,\mathbf{D}. \end{aligned}$$

Now consider the left MFD

$$\frac{\mathbf{E}(s^{-1})}{|\mathbf{I} - \mathbf{A}s^{-1}|} = \mathbf{D}_L(s^{-1})^{-1}\mathbf{N}_L(s^{-1}).$$

In this case, we can obtain the matrix polynomials of the MFD as follows:

$$\begin{aligned} \mathbf{D}_L(s^{-1}) &= \mathbf{I} + \mathbf{D}_1 s^{-1} + \cdots + \mathbf{D}_r s^{-r}, \\ \mathbf{N}_L(s^{-1}) &= \mathbf{N}_0 + \mathbf{N}_1 s^{-1} + \cdots + \mathbf{N}_r s^{-r}. \end{aligned}$$

with

$$\begin{bmatrix} \mathbf{N}_0^T \\ \mathbf{N}_1^T \\ \vdots \\ \mathbf{N}_r^T \\ \mathbf{D}_1^T \\ \vdots \\ \mathbf{D}_r^T \end{bmatrix} = [\mathbf{G}_L^T\mathbf{G}_L]^{-1}\mathbf{G}_L^T \begin{bmatrix} \mathbf{E}_0^T \\ \mathbf{E}_1^T \\ \vdots \\ \mathbf{E}_n^T \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \qquad (4)$$

and

$$\mathbf{G}_L = \begin{bmatrix} \mathbf{I} & & & & \mathbf{0} & & \\ d_1\mathbf{I} & \mathbf{I} & & & -\mathbf{E}_0^T & & \\ \vdots & \vdots & \ddots & & \vdots & & \\ \vdots & \vdots & \ddots & \mathbf{I} & & \vdots & -\mathbf{E}_0^T \\ & & & d_1\mathbf{I} & & & -\mathbf{E}_1^T \\ & & \ddots & \vdots & & & \vdots \\ & & & d_n\mathbf{I} & & & -\mathbf{E}_n^T \end{bmatrix}.$$

The left MFD for a proper continuous model consists of the following matrix polynomials:

$$\begin{aligned} \mathbf{D}_L(s) &= s^r\mathbf{D}_L(s^{-1}), \\ &= \mathbf{I}s^r + \mathbf{D}_1 s^{r-1} + \cdots + \mathbf{D}_r, \\ \mathbf{N}_L(s) &= s^r\mathbf{N}_L(s^{-1}), \\ &= \mathbf{N}_0 s^r + \mathbf{N}_1 s^{r-2} + \cdots + \mathbf{N}_r. \end{aligned}$$

The parameter matrices are obtained from Eq. (4), with the appropriate matrices $\mathbf{G}_L$ and $\mathbf{E}_i$'s calculated by the Vu algorithm from the state-space model matrices.

3.3. Right MFD

We discuss only the case of a discrete control system because it can be used for the continuous systems as well.

For a right MFD of the system, we write

$$\begin{aligned} \mathbf{N}_R(z^{-1})\mathbf{D}_R(z^{-1})^{-1} &= \frac{\mathbf{C}_d[\mathbf{I}-\mathbf{A}_d z^{-1}]^{+}\mathbf{B}_d}{|\mathbf{I}-\mathbf{A}_d z^{-1}|}, \\ &= \frac{\mathbf{E}(z^{-1})}{|\mathbf{I}-\mathbf{A}_d z^{-1}|}. \end{aligned}$$

By post-multiplying both sides of the last equation by $\mathbf{D}_R(z^{-1})$, we obtain

$$|\mathbf{I}-\mathbf{A}_d z^{-1}|\mathbf{N}_R(z^{-1}) = \mathbf{E}(z^{-1})\mathbf{D}_R(z^{-1}). \qquad (5)$$

Then from the last equation, we can say that the two matrix polynomials of a right MFD of a discrete control system are:

$$\begin{aligned} \mathbf{D}_R(z^{-1}) &= \mathbf{I} + \mathbf{D}_1 z^{-1} + \cdots + \mathbf{D}_r z^{-r}, \\ \mathbf{N}_R(z^{-1}) &= \mathbf{N}_0 + \mathbf{N}_1 z^{-1} + \cdots + \mathbf{N}_{r-1} z^{-r+1}, \end{aligned}$$

with the parameter matrices given by

$$\begin{bmatrix} \mathbf{N}_0 \\ \mathbf{N}_1 \\ \vdots \\ \mathbf{N}_{r-1} \\ \mathbf{D}_1 \\ \vdots \\ \mathbf{D}_r \end{bmatrix} = [\mathbf{G}_R^T\mathbf{G}_R]^{-1}\mathbf{G}_R^T \begin{bmatrix} \mathbf{E}_0 \\ \mathbf{E}_1 \\ \vdots \\ \mathbf{E}_{n-1} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}.$$

and

$$\mathbf{G}_R = \begin{bmatrix} \mathbf{I} & & & & \mathbf{0} & & \\ d_1\mathbf{I} & \mathbf{I} & & & -\mathbf{E}_0 & & \\ \vdots & \vdots & \ddots & & \vdots & & \\ \vdots & \vdots & \ddots & \mathbf{I} & & \vdots & \mathbf{0} \\ & & & d_1\mathbf{I} & & & -\mathbf{E}_0 \\ & & \ddots & \vdots & & & \vdots \\ & & & d_n\mathbf{I} & & & -\mathbf{E}_{n-1} \end{bmatrix}.$$

The coefficients d_i's and $\mathbf{E}_i$'s are calculated by the Vu algorithm from the state-space model matrices.

Since the determinant of the transpose of a matrix is the same as that of the matrix, we can say, from Eq. (5), the following fact. The right MFD of a state space model is the transpose of the left MFD of a state space model of which the state transition matrix is the transpose of the former model and the control matrix of one is the transpose of the output matrix of the other and vice versa.

A procedure to obtain a right MFD model from a left MFD model of a same linear system is discussed in [BK97].

3.4. Transfer Function Matrix

Sometimes the information one receives is not a state space model but a transfer function matrix, ie. each element of this matrix is a rational transfer function, not necessarily all strictly proper. In this case, it is not necessarily to look for a state space model, and we do not need the Vu algorithm to calculate the adjoint and determinant polynomials. They are embeded with other matrices to form a term, and this term is obtained by a different way. The procedure is as follows.

1. Separate the transfer function matrix into two parts: one with strictly proper transfer functions and one is a constant matrix, ie.

$$\mathbf{H}(s) = \mathbf{H}_s(s) + \mathbf{D}.$$

2. Amend the transfer function $\mathbf{H}_s(s)$ so that all the denominators of the elements in this matrix are the same. This is called the least common divisor.

$$|s\mathbf{I}-\mathbf{A}| \quad = \quad (s+s_1)(s+s_2)\cdots(s+s_n).$$

3. Multiply the numerator and denominator of each transfer function in the transfer function matrix $\mathbf{H}_s(s)$ by s^{-n}.

4. Bring the common denominator out of the transfer function matrix. The common denominator is the polynomial $|\mathbf{I}-\mathbf{A}s^{-1}|$. The remaining matrix is the term $s^{-1}\mathbf{C}[\mathbf{I}-\mathbf{A}s^{-1}]^{+}\mathbf{B}$.

5. Separate the remaining matrix into a sum of matrices multiplied by different powers of s^{-1}.

6. Then follow the aforementioned steps in the discussion of the strictly proper systems or proper systems depending on if the matrix $\mathbf{D}$ is a zero matrix or not.

3.5. The Markov Parameters

The Markov parameter matrices are the matrix version of the impulse response weights of scalar systems. From the Markov parameters, we can obtain a state space realization model and then an MFD model of a linear system. The methods to obtain a state space model from the Markov parameters are embodied in the realization theory. There is a good number of methods in the control literature known by control engineers. To obtain an MFD model, however, we can skip the realization step and obtain an MFD model directly. The procedure is as follows. We write the following equation of a discrete control system as

$$\frac{\mathbf{C}_d[\mathbf{I}-\mathbf{A}_d z^{-1}]^{+}\mathbf{B}_d}{|\mathbf{I}-\mathbf{A}_d z^{-1}|} = \frac{\mathbf{E}(z^{-1})}{|\mathbf{I}-\mathbf{A}_d z^{-1}|} = \sum_{i=0}\mathbf{H}_i z^{-i}$$

or

$$\frac{\mathbf{E}_0+\mathbf{E}_1 z^{-1}+\cdots+\mathbf{E}_{n-1}z^{-n+1}}{1+d_1 z^{-1}+\cdots+d_n z^{-n}} = \sum_{i=0}\mathbf{H}_i z^{-i}.$$

The parameters on the right hand side of the last equation are known: They are the Markov parameters of the system. Therefore, the parameters on the left hand side can also be known. With this knowledge, we can obtain a left MFD or a right MFD model with the theory presented earlier.

3.6. Coprime, Poles and Zeros

If the given state space model is a minimal realization, then the two matrix polynomials $\mathbf{D}_L(z^{-1})$ and $\mathbf{N}_L(z^{-1})$ will be coprime. The zeros of the polynomials $|\mathbf{D}_L(z^{-1})|$ and $|\mathbf{N}_L(z^{-1})|$ are the poles and transmission zeros of the multivariable control system. If the two polynomials have common zeros, the number of poles and transmission zeros might be reduced. The current control literature suggests the use of the concept of Smith-McMillian form for reduction. The procedure reduces the rational matrix $\mathbf{E}(z^{-1})/|\mathbf{I}-\mathbf{A}_d z^{-1}|$ to the Smith normal form. The resultant is a diagonal matrix premultiplied and postmultiplied by two unimodular polynomial matrices. However as mentioned in [Kai80] page 444, this suggestion is only a valuable conceptual and theoretical tool. It is not convenient for actual numerical computation. Even if it can be computed with powerful modern digital computers, it should not be done this way. Too much work for a small chore! This is the area the Vu algorithm contributes significantly to the matrix fraction description theory: It provides an easy answer to the question of minimal realization and poles and zeros of a multivariable system. If the number of common zeros of the polynomials $|\mathbf{D}_L(z^{-1})|$ and $|\mathbf{N}_L(z^{-1})|$, computed by the determinant algorithm of the Vu algorithm, is less than the dimension of the input and output vectors, the state space realization is minimal and the zeros of the polynomials $|\mathbf{D}_L(z^{-1})|$ and $|\mathbf{N}_L(z^{-1})|$ are the poles and transmission zeros of the system. In the opposite case, we can reduce the number of the poles and transmission zeros and the dimension of the state vector of the state space realization. Since the polynomials $|\mathbf{D}_L(z^{-1})|$ and $|\mathbf{N}_L(z^{-1})|$ are scalar polynomials, we know how to find their greatest common divisor. Then we can extract this left common matrix polynomial divisor out of the two matrix polynomials $\mathbf{D}_L(z^{-1})$ and $\mathbf{N}_L(z^{-1})$ by the method described in [Vu7b], page 253.

4. An Example

In this section, we consider an example to verify our theories. We consider a strictly proper continuous control system with the system matrices:

$$\mathbf{A} = \begin{bmatrix} -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 1 \\ -0.5 & 0 & 0.5 & 0 \end{bmatrix}$$

and

$$\mathbf{B} = \begin{bmatrix} 2 & 3 \\ 1 & 4 \\ 1 & 2 \\ 0 & 1 \end{bmatrix}, \qquad \mathbf{C} = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}.$$

The adjoint algorithm from the Vu algorithm gives us matrices, from which we can calculate the coefficients of the polynomial $\mathbf{E}(s^{-1})$ as follows:

$$\mathbf{E}_0 = \begin{bmatrix} 4 & 11 \\ 2 & 6 \end{bmatrix}, \quad \mathbf{E}_1 = \begin{bmatrix} 9 & 33 \\ 5 & 18 \end{bmatrix},$$

$$\mathbf{E}_2 = \begin{bmatrix} 2 & 20.5 \\ 0 & 9 \end{bmatrix}, \quad \mathbf{E}_3 = \begin{bmatrix} -2 & -3 \\ -1 & -2.5 \end{bmatrix}.$$

From the determinant algorithm, we can calculate the parameters of the determinant polynomial $|\mathbf{I} - \mathbf{A}s^{-1}|$ as below.

$$d_0 = 1, \quad d_1 = 3, \quad d_2 = 1.5, \quad d_3 = -0.5, \quad d_4 = 0.5.$$

For a left MFD, we can obtain the matrix

$$\mathbf{G}_L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 1 & 0 & -4 & -2 & 0 & 0 \\ 0 & 3 & 0 & 1 & -11 & -6 & 0 & 0 \\ 1.5 & 0 & 3 & 0 & -9 & -5 & -4 & -2 \\ 0 & 1.5 & 0 & 3 & -33 & -18 & -11 & -6 \\ -0.5 & 0 & 1.5 & 0 & -2 & 0 & -9 & -5 \\ 0 & -0.5 & 0 & 1.5 & -20.5 & -9 & -33 & -18 \\ 0.5 & 0 & -0.5 & 0 & 2 & 1 & -2 & 0 \\ 0 & 0.5 & 0 & -0.5 & 3 & 2.5 & -20.5 & -9 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0.5 & 0 & 0 & 3 & 2.5 \end{bmatrix}$$

and the following matrices of the fraction polynomials

$$\mathbf{D}_1 = \begin{bmatrix} 2.375 & -2.875 \\ -0.125 & 0.625 \end{bmatrix}, \mathbf{D}_2 = \begin{bmatrix} 1.75 & -3.875 \\ 0.75 & -1.375 \end{bmatrix}$$

with $\mathbf{D}_0 = \mathbf{I}$ and

$$\mathbf{N}_0 = \begin{bmatrix} 4 & 11 \\ 2 & 6 \end{bmatrix}, \quad \mathbf{N}_1 = \begin{bmatrix} 0.75 & 8.875 \\ -0.25 & 2.375 \end{bmatrix}.$$

For a right MFD, we can obtain the matrix

$$\mathbf{G}_R = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 1 & 0 & -4 & -11 & 0 & 0 \\ 0 & 3 & 0 & 1 & -2 & -6 & 0 & 0 \\ 1.5 & 0 & 3 & 0 & -9 & -33 & -4 & -11 \\ 0 & 1.5 & 0 & 3 & -5 & -18 & -2 & -6 \\ -0.5 & 0 & 1.5 & 0 & -2 & -20.5 & -9 & -33 \\ 0 & -0.5 & 0 & 1.5 & 0 & -9 & -5 & -18 \\ 0.5 & 0 & -0.5 & 0 & 2 & 3 & -2 & -20.5 \\ 0 & 0.5 & 0 & -0.5 & 1 & 2.5 & 0 & -9 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0.5 & 0 & 0 & 1 & 2.5 \end{bmatrix}$$

and the following matrices of the fraction polynomials

$$\mathbf{D}_1 = \begin{bmatrix} 4.7143 & 3.5714 \\ -2.1429 & -1.7143 \end{bmatrix}, \mathbf{D}_2 = \begin{bmatrix} 1.5 & 0.5 \\ 0.2857 & 0.4286 \end{bmatrix}$$

with $\mathbf{D}_0 = \mathbf{I}$ and

$$\mathbf{N}_0 = \begin{bmatrix} 4 & 11 \\ 2 & 6 \end{bmatrix}, \quad \mathbf{N}_1 = \begin{bmatrix} -7.7143 & -4.5714 \\ -4.4286 & -3.1429 \end{bmatrix}.$$

To verify the models, we can check the determinants of the denominator polynomials. By using the determinant algorithm of the Vu algorithm, we can calculate the determinants of the polynomials $\mathbf{D}_L(s^{-1})$ and $\mathbf{D}_R(s^{-1})$ as follows:

$$\begin{aligned} |\mathbf{D}_L(s^{-1})| &= \left| \mathbf{I} + \begin{bmatrix} 2.375 & -2.875 \\ -0.125 & 0.625 \end{bmatrix} s^{-1} \right. \\ &\quad \left. + \begin{bmatrix} 1.75 & -3.875 \\ 0.75 & -1.375 \end{bmatrix} s^{-2} \right|, \\ &= 1 + 3s^{-1} + 1.5s^{-2} - 0.5s^{-3} + 0.5s^{-4} \end{aligned}$$

and

$$\begin{aligned} |\mathbf{D}_R(s^{-1})| &= \left| \mathbf{I} + \begin{bmatrix} 4.7143 & 3.5714 \\ -2.1429 & -1.7143 \end{bmatrix} s^{-1} \right. \\ &\quad \left. + \begin{bmatrix} 1.5 & 0.5 \\ 0.2857 & 0.4286 \end{bmatrix} s^{-2} \right|, \\ &= 1 + 3s^{-1} + 1.5s^{-2} - 0.5s^{-3} + 0.5s^{-4}. \end{aligned}$$

The excellent agreement of the determinants $|\mathbf{D}_L(s^{-1})|$, $|\mathbf{D}_R(s^{-1})|$ and $|\mathbf{I} - \mathbf{A}s^{-1}|$ is the proof of the correct denominator polynomials. To ascertain that the numerator polynomials are also correct, we need to prove

$$\mathbf{D}_L(s^{-1})^{+}\mathbf{N}_L(s^{-1}) = \mathbf{E}(s^{-1}) = \mathbf{N}_R(s^{-1})\mathbf{D}_R(s^{-1})^{+}.$$

We only check it for the left MFD. We have

$$\begin{aligned} \mathbf{D}_L(s^{-1})^{+}\mathbf{N}_L(s^{-1}) &= \left[\mathbf{I} + \begin{bmatrix} 0.625 & 2.875 \\ 0.125 & 2.375 \end{bmatrix} s^{-1} + \right. \\ &\left. \begin{bmatrix} -1.375 & 3.875 \\ -0.75 & 1.75 \end{bmatrix} s^{-2} \right] \left[\begin{bmatrix} 4 & 11 \\ 2 & 6 \end{bmatrix} + \begin{bmatrix} 0.75 & 8.875 \\ -0.25 & 2.375 \end{bmatrix} s^{-1} \right] \\ &= \hat{\mathbf{E}}_0 + \hat{\mathbf{E}}_1 s^{-1} + \hat{\mathbf{E}}_2 s^{-2} + \hat{\mathbf{E}}_3 s^{-3}. \end{aligned}$$

Since the numbers of the polynomials are exact, readers can verify that the matrix coefficients $\hat{\mathbf{E}}_i$'s are the same as their correct values $\mathbf{E}_i$'s. The Vu algorithm helps us to verify the obtained fraction polynomials $\mathbf{N}_L(s^{-1})$ and $\mathbf{D}_L(s^{-1})$. In the case of unobtained exactness, it tells us how close the given model and the obtained model are. This is what the Faddeev-Leverrier algorithm cannot do.

The polynomial $|\mathbf{N}_L(s^{-1})|$ can be calculated by the determinant algorithm of the Vu algorithm as given below

$$\begin{aligned} |\mathbf{N}_L(s^{-1})| &= \left| \begin{bmatrix} 4 & 11 \\ 2 & 6 \end{bmatrix} + \begin{bmatrix} 0.75 & 8.875 \\ -0.25 & 2.375 \end{bmatrix} s^{-1} \right|, \\ &= 2 - s^{-1} + 4s^{-2}. \end{aligned}$$

The zeros of the polynomial $|\mathbf{D}_L(s^{-1})|$ are: $s_1 = -2.1377$, $s_2 = -1.2753$, $s_3 = 0.2065 + 0.3752i$ and $s_4 = 0.2065 - 0.3752i$. These are the same as the eigenvalues of the state matrix $\mathbf{A}$. The zeros of the polynomial $|\mathbf{N}_L(s^{-1})|$ are: $s_1 = 0.2500 + 1.3919i$ and $s_2 = 0.2500 - 1.3919i$. There are no common zeros of the two polynomials $|\mathbf{D}_L(s^{-1})|$ and $|\mathbf{N}_L(s^{-1})|$, so the two matrix polynomials $\mathbf{D}_L(s^{-1})$ and $\mathbf{N}_L(s^{-1})$ are coprime. The state space model is a minimal realization. The poles of the system are: $s_1 = -2.1377$, $s_2 = -1.2753$, $s_3 = 0.2065 + 0.3752i$ and $s_4 = 0.2065 - 0.3752i$. Similarly, the transmission zeros of the system are: $s_1 = 0.2500 + 1.3919i$ and $s_2 = 0.2500 - 1.3919i$.

5. Conclusion

Matrix fraction description is the parsimonious model of a square linear multivariable feedback control system. This paper has presented an algorithm to obtain the two matrix polynomials of this representation. The algorithm can be used for discrete systems and proper or strictly proper continuous systems. The procured models can be obtained from a number of given pieces of information: a state space model realization, a transfer function matrix or the Markov parameters of a multivariable square system. The approach makes use of the Vu algorithm, which gives the adjoint and determinant polynomials of a square matrix polynomial. The Vu algorithm is indispensable for matrix fraction description theory. It gives algorithms not only to obtain the necessary parameters to procure the fraction polynomials but also to verify the obtained fraction polynomials for their coprime characteristic. With it, one can calculate the poles and transmission zeros of a square multivariable control system. While the paper discusses only square systems, rectangular systems pose no more difficulty than square systems. In this case, only the denominator polynomials have to be square. Another strength of the approach of this paper is that the Vu algorithm can be extended to two-parameter polynomial matrices easily. Then a similar approach can be used to obtain a matrix fraction description from a state space model for two-dimensional linear systems, like those of the two-dimenional digital filters.

References

[BK97] J.C. Basilio and B. Kouvaritakis. An algorithm for coprime matrix fraction description using sylvester matrices. *Linear Algebra and Its Applications*, Vol. 266(1):107–125, 1997.

[DG92] K.B. Datta and S. Gangopadhyay. Coprime matrix fraction description via orthogonal structure theorem. *IEEE Transaction on Automatic Control*, 37(10):1517–1520, 1992.

[DG93] K.B. Datta and S. Gangopadhyay. Coprime matrix fraction description via sylvester resultant matrix. *Control Theory and Advanced Technology*, 4(1):971–981, 1993.

[Kac07] T. Kaczorek. *Polynomial and Rational Matrices*. Springer-Verlag London, Ltd., London WC1X 8HB, United Kingdom, 1st edition, 2007.

[Kai80] T. Kailath. *Linear Systems*. Prentice-Hall, Inc., Upper Saddle River, NJ 07458, USA, 15th edition, 1980.

[MCWY14] B.Q. Mu, H.F. Chen, L.Y. Wang, and G. Yin. Characterization and identification of matrix fraction descriptions for lti systems. *SIAM J. on Control and Optimization*, 52(6):3694–3721, 2014.

[Pat81] R.V. Patel. Computation of matrix fraction descriptions of linear time-invariant systems. *IEEE Transactions on Automatic Control*, AC-26(1):148–161, 1981.

[STV09] P.S. Stanimirovic, M.B. Tasic, and K.M. Vu. An extension of faddeev algorithms to polynomial matrices. *Applied Mathematics and Computation*, 214(1):246–258, 2009.

[Vu7a] K. Vu. *Optimal Discrete Control Theory: The Rational Function Structure Model*. AuLac Technologies Inc., Ottawa, ON, Canada, 1st edition, 2007a.

[Vu7b] K. Vu. *The ARIMA and VARIMA Time Series: Their Modelings, Analyses and Applications*. AuLac Technologies Inc., Ottawa, ON, Canada, 1st edition, 2007b.

[Vu99] K. Vu. Pencil characteristic coefficients and their applications in control. *IEE Proceedings-Control Theory and Applications*, 146(5):450–456, 1999.

[Vu08] K. Vu. An extension of the faddeev algorithms. pages 150–155, San Antonio, TX, USA, 2008. IEEE MSC Conference.

[WF69] W.A. Wolovich and P.L. Falb. On the structure of multivariable systems. *SIAM Journal of Control*, 7(3):437–451, 1969.

[Wol74] W.A. Wolovich. *Linear Multivariable Systems*. Springer-Verlag New York Inc., New York, NY, USA, 1st edition, 1974.

[ZDL09] B. Zhou, G.R. Duan, and Z.Y. Li. A stein matrix equation approach for computing coprime matrix fraction description. *IET Control Theory and Applications*, 3(6):691–700, 2009.