The Computation of the Inverse of a Square Polynomial Matrix *

Ky M. Vu, PhD. AuLac Technologies Inc. ©2008
Email: kymvu@aulactechnologies.com

Abstract

An approach to calculate the inverse of a square polynomial matrix is suggested. The approach consists of two similar algorithms: One calculates the determinant polynomial and the other calculates the adjoint polynomial. The algorithm to calculate the determinant polynomial gives the coefficients of the polynomial in a recursive manner from a recurrence formula. Similarly, the algorithm to calculate the adjoint polynomial also gives the coefficient matrices recursively from a similar recurrence formula. Together, they give the inverse as a ratio of the adjoint polynomial and the determinant polynomial.

Keywords: Adjoint, Cayley-Hamilton theorem, determinant, Faddeev's algorithm, inverse matrix.

1 Introduction

Modern control theory consists of a large part of matrix theory. A control theorist is usually an expert in matrix theory, a field of applied algebra. There are books written entirely on matrix theory for control engineers. We can cite two books: one by S. Barnett (1960) and the other by T. Kaczorek (2007). The reason for this fact is that matrices are used to represent models of a control system. Since matrices are used in control theory, their manipulation and calculation are of paramount importance for control engineers. The most significant task is the calculation of the inverse of a polynomial matrix. There are algorithms, discussed in textbooks, for this task. We can cite the algorithms in V. Kučera (1979) and T. Kaczorek (2007). These algorithms are poor and inefficient in the sense that they cannot be used for similar types of polynomial matrix. In this paper, we will present algorithms to calculate the determinant and adjoint polynomials of an inverse polynomial matrix. The algorithms are an extension of a known algorithm by K. Vu (1999) to calculate the characteristic coefficients of a pencil. The paper is organized as follows. Section one is the introduction section. The algorithms are discussed in section two. In section three, some examples are given. Section four concludes the discussion of the paper.

*A new-and-improved version of this paper can be found in the Proceedings of the IEEE Multi-conference on Systems and Control on September 3-5th, 2008 in San Antonio, TX under the title *An Extension of the Faddeev's Algorithms*.

2 The Inverse Polynomial

The inverse of a square nonsingular matrix is a square matrix, which by premultiplying or postmultiplying with the matrix gives an identity matrix. An inverse matrix can be expressed as a ratio of the adjoint and determinant of the matrix. A singular matrix has no inverse because its determinant is zero; we cannot calculate its inverse. We, however, can always calculate its adjoint and determinant. It is, therefore, always better to calculate the inverse of a square polynomial matrix by calculating its adjoint and determinant polynomials separately and obtain the inverse as a ratio of these polynomials. In the following discussion, we will present algorithms to calculate the adjoint and determinant polynomials.

2.1 The Determinant Polynomial

To begin our discussion, we consider the following equation that is the determinant of a square matrix of dimension n, viz

$$\begin{aligned}
|\lambda\mathbf{I}-\mathbf{C}| &= \lambda^n - p_1(\mathbf{C})\lambda^{n-1} + p_2(\mathbf{C})\lambda^{n-2}\cdots(-1)^n p_n(\mathbf{C}),\\
&= \sum_{i=0}^{n}(-1)^i p_i(\mathbf{C})\lambda^{n-i}, \qquad p_0(\mathbf{C})=1,\\
&= (\lambda-\lambda_1)(\lambda-\lambda_2)\cdots(\lambda-\lambda_n),\\
&= f(\lambda).
\end{aligned}$$

The coefficients $p_i(\mathbf{C})$'s are called the characteristic coefficients of the matrix $\mathbf{C}$. They are related to the zeros λ_i's of the polynomial $f(\lambda)$ as

$$\begin{aligned}
p_1(\mathbf{C}) &= \lambda_1+\lambda_2+\cdots+\lambda_n,\\
&= tr\mathbf{C},\\
p_2(\mathbf{C}) &= \lambda_1\lambda_2\cdots+\lambda_1\lambda_n+\lambda_2\lambda_3\cdots+\lambda_2\lambda_n\cdots+\lambda_{n-1}\lambda_n,\\
&\vdots\\
p_n(\mathbf{C}) &= \lambda_1\lambda_2\cdots\lambda_n,\\
&= |\mathbf{C}|.
\end{aligned}$$

The characteristic coefficients are given recursively as

$$p_m(\mathbf{C}) = \frac{1}{m}\sum_{i=1}^{m}(-1)^{i-1}p_{m-i}(\mathbf{C})p_1(\mathbf{C}^i).$$

The coefficients $p_1(\mathbf{C}^i)$'s are the Newton sums of the function $f(\lambda)$. Now, consider the case

$$\mathbf{C} = \mathbf{A}+\mu\mathbf{B}_1+\mu^2\mathbf{B}_2+\cdots+\mu^r\mathbf{B}_r. \qquad (1)$$

In this case, the function $p_n(\mathbf{C})$ is a polynomial of degree $n \times r$ in the parameter μ. Similarly, the characteristic coefficient $p_m(\mathbf{C})$ is a polynomial of degree $m \times r$ in the parameter μ. Therefore, we can write

$$p_m(\mathbf{C}) \quad = \quad \sum_{k=0}^{m\times r} q_{m,k}(\mathbf{C})\mu^k, \qquad m = 1, \cdots n.$$

From the last equation, we can obtain

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

Some special values for these coefficients are given below

$$\begin{aligned} q_{m,0}(\mathbf{C}) &= p_m(\mathbf{A}), \\ q_{0,0}(\mathbf{C}) &= 1. \end{aligned}$$

The coefficient $q_{m,k}(\mathbf{C})$ can be calculated recursively, and the last equation serves as the initial condition for the calculation. In the following discussion, we will prove this fact.

From the relation of the characteristic coefficients

$$p_m(\mathbf{C}) \quad = \quad \frac{1}{m}\sum_{i=1}^{m}(-1)^{i-1}p_{m-i}(\mathbf{C})p_1(\mathbf{C}^i)$$

and the derivative of a product of two scalars

$$\frac{d^k p_{m-i}(\mathbf{C})p_1(\mathbf{C}^i)}{d\mu^k} \quad = \quad \sum_{j=0}^{k}\begin{pmatrix} k \\ j \end{pmatrix}\frac{d^{k-j}p_{m-i}(\mathbf{C})}{d\mu^{k-j}}\frac{d^j p_1(\mathbf{C}^i)}{d\mu^j},$$

we can write

$$q_{m,k}(\mathbf{C}) =$$
$$\frac{1}{k!m}\sum_{i=1}^{m}(-1)^{i-1}\sum_{j=0}^{k}\begin{pmatrix} k \\ j \end{pmatrix}\frac{d^{k-j}p_{m-i}(\mathbf{C})}{d\mu^{k-j}}\left.\frac{d^j p_1(\mathbf{C}^i)}{d\mu^j}\right|_{\mu=0}.$$

By canceling and redistributing the factorials, we get

$$\begin{aligned} q_{m,k}(\mathbf{C}) &= \frac{1}{m}\sum_{i=1}^{m}\sum_{j=0}^{k}(-1)^{i-1}\frac{d^{k-j}p_{m-i}(\mathbf{C})}{(k-j)!d\mu^{k-j}}\left.\frac{d^j p_1(\mathbf{C}^i)}{j!d\mu^j}\right|_{\mu=0}, \\ &= \frac{1}{m}\sum_{i=1}^{m}\sum_{j=0}^{k}(-1)^{i-1}q_{m-i,k-j}(\mathbf{C})\left.\frac{d^j p_1(\mathbf{C}^i)}{j!d\mu^j}\right|_{\mu=0}. \end{aligned}$$

The derivatives of the characteristic coefficient $p_1(\mathbf{C}^i)$ are not difficult to obtain, but it is complicated to obtain a general formula for them. We, therefore, will solve the problem as follows. By noting that we can interchange the order of the operators: trace and derivative, we write

$$\frac{d^j p_1(\mathbf{C}^i)}{j!d\mu^j} \quad = \quad p_1\left(\frac{d^j\mathbf{C}^i}{j!d\mu^j}\right),$$

then obtain

$$q_{m,k}(\mathbf{C}) \quad = \quad \frac{1}{m}\sum_{i=1}^{m}\sum_{j=0}^{k}(-1)^{i-1}q_{m-i,k-j}(\mathbf{C})\; p_1\left.\left(\frac{d^j\mathbf{C}^i}{j!d\mu^j}\right)\right|_{\mu=0} \quad (2)$$

The derivatives of the matrix $\mathbf{C}^i$ can be calculated recursively. In the following discussion, we will present an approach to calculate them numerically. We write

$$\mathbf{C}^i \quad = \quad \mathbf{C}^{i-1}\mathbf{C},$$

then differentiate both sides with respect to μ to obtain

$$\begin{aligned} \frac{d^j\mathbf{C}^i}{d\mu^j} &= \sum_{l=0}^{j}\begin{pmatrix} j \\ l \end{pmatrix}\frac{d^{j-l}\mathbf{C}^{i-1}}{d\mu^{j-l}}\frac{d^l\mathbf{C}}{d\mu^l}, \\ &= \frac{d^j\mathbf{C}^{i-1}}{d\mu^j}\mathbf{C} + \sum_{l=1}^{j}\begin{pmatrix} j \\ l \end{pmatrix}\frac{d^{j-l}\mathbf{C}^{i-1}}{d\mu^{j-l}}\frac{d^l\mathbf{C}}{d\mu^l}, \\ &= \frac{d^j\mathbf{C}^{i-2}}{d\mu^j}\mathbf{C}^2 + \sum_{k=1}^{2}\sum_{l=1}^{j}\begin{pmatrix} j \\ l \end{pmatrix}\frac{d^{j-l}\mathbf{C}^{i-k}}{d\mu^{j-l}}\frac{d^l\mathbf{C}}{d\mu^l}\mathbf{C}^{k-1}. \end{aligned}$$

To be able to calculate the derivatives numerically in a recursive manner, we have to eliminate the jth derivative of the matrix $\mathbf{C}^{i-2}$ on the right hand side of the above equation. We, therefore, will make repeated substitutions to reduce the power of the matrix $\mathbf{C}^{i-2}$ with matrices of lower powers until we know the jth derivative of this matrix is zero. This matrix can be an identity matrix, so we have eventually

$$\frac{d^j\mathbf{C}^i}{d\mu^j} \quad = \quad \sum_{k=1}^{i}\sum_{l=1}^{j}\begin{pmatrix} j \\ l \end{pmatrix}\frac{d^{j-l}\mathbf{C}^{i-k}}{d\mu^{j-l}}\frac{d^l\mathbf{C}}{d\mu^l}\mathbf{C}^{k-1}.$$

The lth derivative of the matrix $\mathbf{C}$ can be evaluated as

$$\begin{aligned} \frac{d^l\mathbf{C}}{d\mu^l} &= \frac{d^l}{d\mu^l}[\mathbf{A} + \sum_{s=1}^{r}\mathbf{B}_s\mu^s], \\ &= \sum_{s=l}^{r}s(s-1)\cdots(s-l+1)\mathbf{B}_s\mu^{s-l}. \quad (3) \end{aligned}$$

By putting this derivative into the previous equation, we get

$$\begin{aligned} \frac{d^j\mathbf{C}^i}{d\mu^j} &= \sum_{k=1}^{i}\sum_{l=1}^{j}\begin{pmatrix} j \\ l \end{pmatrix}\frac{d^{j-l}\mathbf{C}^{i-k}}{d\mu^{j-l}}\sum_{s=l}^{r}s(s-1)\cdots(s-l+1) \\ &\qquad \mathbf{B}_s\mu^{s-l}\mathbf{C}^{k-1}, \\ &= \sum_{k=1}^{i}\sum_{l=1}^{j}\frac{j!}{l!(j-l)!}\frac{d^{j-l}\mathbf{C}^{i-k}}{d\mu^{j-l}}\sum_{s=l}^{r}s(s-1)\cdots(s-l+1) \\ &\qquad \mathbf{B}_s\mu^{s-l}\mathbf{C}^{k-1}. \quad (4) \end{aligned}$$

Now, by setting $\mu = 0$, Eq. (3) becomes

$$\begin{aligned} \left.\frac{d^l\mathbf{C}}{d\mu^l}\right|_{\mu=0} &= \left.\sum_{s=l}^{r}s(s-1)\cdots(s-l+1)\mathbf{B}_s\mu^{s-l}\right|_{\mu=0}, \\ &= s!\mathbf{B}_l, \\ &= l!\mathbf{B}_l. \end{aligned}$$

Therefore, by setting $\mu = 0$ and dividing both sides of Eq. (4) by $j!$, we can calculate the derivatives of the matrix $\mathbf{C}^i$ recursively as

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

with the initial condition

$$\left.\frac{d^0\mathbf{C}^i}{d\mu^0}\right|_{\mu=0} = \mathbf{A}^i.$$

With the derivatives obtained, we can take their traces and calculate, using Eq. (2), all the coefficients $q_{m,k}(\mathbf{C})$'s of the characteristic coefficient $p_m(\mathbf{C})$ recursively. The determinant corresponds to the coefficient $p_n(\mathbf{C})$.

2.2 The Adjoint Polynomial

An algorithm, called the Faddeev's algorithm described in D. Faddeev and I. Sominskii (1954), has been used as a standard algorithm to calculate the determinant and adjoint polynomials of a square polynomial matrix for control engineers. The algorithm is discussed in W. Wolovich (1974) and V. Faddeeva (1959), among others. The algorithm, however, can only be used for the resolvent, $[\lambda\mathbf{I} - \mathbf{C}]^{-1}$, of a constant matrix $\mathbf{C}$. We, therefore, need an algorithm, similar to the algorithm to calculate the determinant polynomial, to calculate the adjoint polynomial of the pencil given by (1). To derive the equations for the algorithm, we use the relation between the characteristic coefficients of a matrix and the matrix in the Cayley-Hamilton theorem. From this theorem, we have

$$\mathbf{C}^n - p_1(\mathbf{C})\mathbf{C}^{n-1} + \cdots + (-1)^n p_n(\mathbf{C})\mathbf{I} = \mathbf{0}.$$

From the above equation, we can write

$$\mathbf{C}^{-1} = \sum_{i=0}^{n-1}(-1)^i \frac{p_{n-1-i}(\mathbf{C})}{p_n(\mathbf{C})}\mathbf{C}^i, \qquad p_0(\mathbf{C}) = 1,$$

provided that $p_n(\mathbf{C})$ is not zero, which means that the matrix $\mathbf{C}$ has an inverse. We can write the last equation as

$$\begin{aligned}
[\mathbf{A} + \mu\mathbf{B}_1 + \cdots + \mu^r\mathbf{B}_r]^{-1} &= \frac{1}{p_n(\mathbf{C})}\sum_{i=0}^{n-1}(-1)^i p_{n-1-i}(\mathbf{C})\mathbf{C}^i, \\
&= \frac{1}{p_n(\mathbf{C})} adj\ \mathbf{C}, \\
&= \frac{1}{p_n(\mathbf{C})}\sum_{k=0}^{r(n-1)} \mathbf{D}_k\mu^k.
\end{aligned}$$

This means that we have

$$\begin{aligned}
adj\ \mathbf{C} &= \mathbf{C}^+, \\
&= \sum_{k=0}^{r(n-1)} \mathbf{D}_k\mu^k.
\end{aligned}$$

To calculate the coefficient matrices $\mathbf{D}_k$'s for the adjoint polynomial, we proceed as follows. We write

$$\sum_{j=0}^{r(n-1)} \mathbf{D}_k\mu^k = \sum_{i=0}^{n-1}(-1)^i p_{n-1-i}(\mathbf{C})\mathbf{C}^i.$$

By differentiating both sides k times, dividing both sides by $k!$ and setting $\mu = 0$, we get

$$\begin{aligned}
\mathbf{D}_k &= \left.\sum_{i=0}^{n-1}(-1)^i \frac{1}{k!}\frac{d^k}{d\mu^k}p_{n-1-i}(\mathbf{C})\mathbf{C}^i\right|_{\mu=0}, \\
&= \left.\sum_{i=0}^{n-1}\sum_{j=0}^{k}(-1)^i \frac{1}{k!}\begin{pmatrix} k \\ j \end{pmatrix}\frac{d^{k-j}p_{n-1-i}(\mathbf{C})}{d\mu^{k-j}}\frac{d^j\mathbf{C}^i}{d\mu^j}\right|_{\mu=0}, \\
&= \left.\sum_{i=0}^{n-1}\sum_{j=0}^{k}(-1)^i\ q_{n-1-i,k-j}(\mathbf{C})\frac{d^j\mathbf{C}^i}{j!d\mu^j}\right|_{\mu=0}, \\
&= \left.\sum_{i=0}^{n-1}\sum_{j=0}^{k}(-1)^i\ q_{n-1-i,k-j}(\mathbf{C})\frac{d^j\mathbf{C}^i}{j!d\mu^j}\right|_{\mu=0}.
\end{aligned}$$

In this case, we do not have to take the traces of the derivative matrices, and there is a slight difference in the coefficients $q_{m,k}(\mathbf{C})$'s of the two cases. Nonetheless, once we have calculated all these coefficients and the derivative matrices, a single program can calculate both the adjoint and the determinant polynomials of the inverse of a polynomial matrix.

3 Some Examples

We will consider a verification example first. Suppose that we have the following matrices:

$$\mathbf{A} = \begin{bmatrix} 0.7812 & -1.1878 & 0.3274 & -0.9471 \\ 0.5690 & -2.2023 & 0.2341 & -0.3744 \\ -0.8217 & 0.9863 & 0.0215 & -1.1859 \\ -0.2656 & -0.5186 & -1.0039 & -1.0559 \end{bmatrix},$$

$$\mathbf{B}_1 = \begin{bmatrix} 1.4725 & -1.1283 & 0.1286 & -0.2624 \\ 0.0557 & -1.3493 & 0.6565 & -1.2132 \\ -1.2173 & -0.2611 & -1.1678 & -1.3194 \\ -0.0412 & 0.9535 & -0.4606 & 0.9312 \end{bmatrix},$$

$$\mathbf{B}_2 = \begin{bmatrix} 0.0112 & -0.9898 & 1.1380 & -0.3306 \\ -0.6451 & 1.3396 & -0.6841 & -0.8436 \\ 0.8057 & 0.2895 & -1.2919 & 0.4978 \\ 0.2316 & 1.4789 & -0.0729 & 1.4885 \end{bmatrix}.$$

With the parameter $\mu = 0.9$, we have

$$\begin{aligned}
\mathbf{C} &= \mathbf{A} + \mu\mathbf{B}_1 + \mu^2\mathbf{B}_2, \\
&= \begin{bmatrix} 2.1155 & -3.0050 & 1.3650 & -1.4511 \\ 0.0966 & -2.3316 & 0.2707 & -2.1496 \\ -1.2647 & 0.9858 & -2.0760 & -1.9702 \\ -0.1151 & 1.5374 & -1.4776 & 0.9879 \end{bmatrix}.
\end{aligned}$$

The adjoint matrix of the matrix $\mathbf{C}$ can be calculated directly from the above matrix, and it is obtained as

$$\mathbf{C}^+ = \begin{bmatrix} 6.7558 & -6.9290 & 2.9784 & 0.7866 \\ 3.5827 & -10.8471 & 5.7842 & -6.8049 \\ 0.8805 & -6.6893 & 1.8359 & -9.6012 \\ -3.4716 & 6.0687 & -5.9089 & 6.2071 \end{bmatrix}.$$

The adjoint algorithm gives the following matrices for the adjoint polynomial, viz

$$\mathbf{D}_0 = \begin{bmatrix} 3.4262 & -2.9107 & -2.1668 & 0.3923 \\ 1.1307 & -1.9154 & -0.2246 & -0.0827 \\ 0.0170 & 1.2894 & 1.6667 & -2.3444 \\ -1.4334 & 0.4471 & -0.9293 & -0.3830 \end{bmatrix},$$

$$\mathbf{D}_1 = \begin{bmatrix} 5.1952 & -2.8108 & -0.0282 & 1.9144 \\ 2.3014 & -3.8208 & 2.0678 & -0.9044 \\ 1.0858 & 0.3998 & 1.9747 & -8.9693 \\ -3.8539 & 0.4194 & -4.5510 & 0.1092 \end{bmatrix},$$

$$\mathbf{D}_2 = \begin{bmatrix} -2.4785 & -2.4430 & 2.0131 & 5.2747 \\ 0.2682 & -0.6039 & 2.3843 & -1.3142 \\ -2.5755 & -5.5819 & -5.4978 & -4.1973 \\ 1.2567 & 0.2934 & -1.6719 & 4.7609 \end{bmatrix},$$

$$\mathbf{D}_3 = \begin{bmatrix} 2.6775 & 1.5003 & 3.2174 & 2.3165 \\ -0.5803 & -1.7740 & 0.1633 & -0.5132 \\ -1.5882 & -5.0169 & -4.4797 & 6.0840 \\ 2.5476 & 2.1033 & 2.7061 & 2.6865 \end{bmatrix},$$

$$\mathbf{D}_4 = \begin{bmatrix} 5.8422 & 2.1378 & 5.1934 & -5.0853 \\ 2.5456 & -2.6364 & 2.6891 & -3.1731 \\ 5.6557 & 0.3962 & 5.9156 & 2.0221 \\ -4.0048 & 2.1808 & -2.0852 & -0.4903 \end{bmatrix},$$

$$\mathbf{D}_5 = \begin{bmatrix} -4.7791 & -1.9663 & -1.1173 & -5.8981 \\ -0.2550 & -2.1538 & 1.2754 & -3.2498 \\ 1.0004 & -0.0831 & 4.0484 & -1.9139 \\ -0.0766 & 2.0738 & -1.3298 & 1.7704 \end{bmatrix},$$

$$\mathbf{D}_6 = \begin{bmatrix} -4.3303 & -1.5901 & -2.9224 & -0.8857 \\ -1.7560 & -1.3344 & -0.7904 & -0.8820 \\ -2.2039 & -0.6977 & -0.3026 & -0.7837 \\ 2.3105 & 1.5390 & 1.2252 & -0.0876 \end{bmatrix}.$$

It can be verified that

$$\mathbf{C}^{+} = \mathbf{D}_0 + \mathbf{D}_1\mu + \mathbf{D}_2\mu^2 + \cdots + \mathbf{D}_6\mu^6.$$

Similarly, we can calculate the determinant from the matrix $\mathbf{C}$. We have

$$|\mathbf{C}| = 9.7658.$$

The determinant algorithm gives the following coefficients:

$$q_{4,0}(\mathbf{C}) = 2.6967, q_{4,1}(\mathbf{C}) = 9.4781, q_{4,2}(\mathbf{C}) = 1.3285,$$
$$q_{4,3}(\mathbf{C}) = -4.4752, q_{4,4}(\mathbf{C}) = 7.2693, q_{4,5}(\mathbf{C}) = 2.4330,$$
$$q_{4,6}(\mathbf{C}) = -5.5012, q_{4,7}(\mathbf{C}) = -3.9225, q_{4,8}(\mathbf{C}) = -1.5827.$$

It can also be verified that

$$|\mathbf{C}| = q_{4,0}(\mathbf{C}) + q_{4,1}(\mathbf{C})\mu + \cdots + q_{4,8}(\mathbf{C})\mu^8.$$

Now, we will consider a control theory example. We will obtain the unconstrained controller for a multivariable stochastic regulating control system. The system is described by a state space model as

$$\hat{\mathbf{y}}_{t+f+1} = \mathbf{c}[\mathbf{I} - \mathbf{A}z^{-1}]^{-1}\mathbf{b}\mathbf{u}_t.$$

The triple $(\mathbf{c}, \mathbf{A}, \mathbf{b})$ is called a realization of the system. The system has a pure dead time of f intervals. The input or manipulated variable vector is $\mathbf{u}_t$, and the output variable vector is $\hat{\mathbf{y}}_t$. This variable is the undisturbed output variable. It is disturbed by a VARIMA time series given by the following equation

$$\boldsymbol{\varphi}(z^{-1})\mathbf{n}_{t+f+1} = \boldsymbol{\vartheta}(z^{-1})\mathbf{a}_{t+f+1}.$$

Then, we can write the disturbed output variable $\mathbf{y}_t$ of the control system at the time $(t+f+1)$ as

$$\begin{aligned} \mathbf{y}_{t+f+1} &= \hat{\mathbf{y}}_{t+f+1} + \mathbf{n}_{t+f+1}, \\ &= \mathbf{c}[\mathbf{I} - \mathbf{A}z^{-1}]^{-1}\mathbf{b}\mathbf{u}_t + \boldsymbol{\varphi}(z^{-1})^{-1}\boldsymbol{\vartheta}(z^{-1})\mathbf{a}_{t+f+1}. \end{aligned}$$

The purpose of the controller is to produce minimum variance for $\mathbf{y}_t$.

To obtain the controller, we will reason as follows. At the time t when $\mathbf{y}_t$ is available, we want to move the control variable $\mathbf{u}_t$ such that the output variable $\mathbf{y}_{t+f+1} = \mathbf{0}$. This requires the knowledge of $\mathbf{n}_{t+f+1}$. At the time t, however, the variable $\mathbf{n}_{t+f+1}$ is not available, so we must replace it by its minimum variance predictor. We must obtain the $(f+1)$ steps ahead forecast for $\mathbf{n}_t$. This can be done as follows.

From the equation of the VARIMA time series, we have

$$\begin{aligned} \mathbf{n}_{t+f+1} &= \boldsymbol{\varphi}(z^{-1})^{-1}\boldsymbol{\vartheta}(z^{-1})\mathbf{a}_{t+f+1}, \\ &= \frac{\boldsymbol{\varphi}(z^{-1})^{+}}{|\boldsymbol{\varphi}(z^{-1})|}\boldsymbol{\vartheta}(z^{-1})\mathbf{a}_{t+f+1}, \\ &= [\boldsymbol{\psi}(z^{-1}) + \frac{\boldsymbol{\gamma}(z^{-1})}{|\boldsymbol{\varphi}(z^{-1})|}z^{-f-1}]\mathbf{a}_{t+f+1}, \\ &= \boldsymbol{\psi}(z^{-1})\mathbf{a}_{t+f+1} + \frac{\boldsymbol{\gamma}(z^{-1})}{|\boldsymbol{\varphi}(z^{-1})|}\mathbf{a}_t. \end{aligned}$$

By taking the conditional expectation at the time t, we obtain

$$\hat{\mathbf{n}}_{t+f+1|t} = \frac{\boldsymbol{\gamma}(z^{-1})}{|\boldsymbol{\varphi}(z^{-1})|}\mathbf{a}_t.$$

The controller can be obtained by setting

$$\begin{aligned} \mathbf{c}[\mathbf{I} - \mathbf{A}z^{-1}]^{-1}\mathbf{b}\mathbf{u}_t &= -\hat{\mathbf{n}}_{t+f+1|t}, \\ &= -\frac{\boldsymbol{\gamma}(z^{-1})}{|\boldsymbol{\varphi}(z^{-1})|}\mathbf{a}_t. \end{aligned}$$

To obtain the controller in terms of the input and output variables, we have to replace the white noise $\mathbf{a}_t$ by the output variable $\mathbf{y}_t$. This can be done by observing that under the minimum variance control law, the output variable equals the prediction error, i.e. we have

$$\mathbf{y}_t = \boldsymbol{\psi}(z^{-1})\mathbf{a}_t.$$

Therefore, we have

$$\begin{aligned}\mathbf{a}_t &= \boldsymbol{\psi}(z^{-1})^{-1}\mathbf{y}_t, \\ &= \frac{\boldsymbol{\psi}(z^{-1})^{+}}{|\boldsymbol{\psi}(z^{-1})|}\mathbf{y}_t.\end{aligned}$$

The desired controller is

$$\begin{aligned}&|\boldsymbol{\varphi}(z^{-1})|\;|\boldsymbol{\psi}(z^{-1})|\mathbf{c}[\mathbf{I}-\mathbf{A}z^{-1}]^{+}\mathbf{b}\mathbf{u}_t = \\ &\qquad -|\mathbf{I}-\mathbf{A}z^{-1}|\boldsymbol{\gamma}(z^{-1})\boldsymbol{\psi}(z^{-1})^{+}\mathbf{y}_t.\end{aligned}$$

In the above equation, the polynomials $[\mathbf{I}-\mathbf{A}z^{-1}]^{+}$ and $|\mathbf{I}-\mathbf{A}z^{-1}|$ can be obtained by the Faddeev's algorithm; the other polynomials, however, must be obtained by the algorithms in this paper.

4 Conclusion

In this paper, we have presented algorithms to calculate the adjoint and determinant polynomials of a square polynomial matrix. The algorithms are sufficient and elegant, and they can be extended further for matrices with polynomials in two or more parameters. With such a strength, the algorithms should be standard algorithms, used in control applications, for the calculation of the adjoint and determinant polynomials.

References

S. Barnett (1960). *Matrices in Control Theory*. Van Nostrand Reinhold Company, London.

V.N. Faddeeva (1959). *Computational Methods of Linear Algebra*. Dover Publications Inc., New York.

D.K. Faddeev and I.S. Sominskii (1954). *Problems in Higher Algebra*. Gostekhizdat, Moscow, 2nd ed. 1949, 5th ed..

T. Kaczorek (2007). *Polynomial and Rational Matrices: Applications in Dynamical Systems Theory*. Springer-Verlag London Limited, London.

V. Kučera (1979). *Discrete Linear Control: The Polynomial Equation Approach*. John Wiley and Son Ltd., New York.

Ky M. Vu (1999). Pencil Characteristic Coefficients and Their Applications in Control. *IEE Proceedings: Control Theory and Applications*, 146:450–456.

W.A. Wolovich (1974). *Linear Multivariable Systems*. Springer-Verlag New York Inc., New York.