ELSEVIER

Systems & Control Letters 27 (1996) 47–53

SYSTEMS & CONTROL LETTERS

An algorithm for inverting rational matrices ☆

Ching-An Lin *, Chien-Wen Yang, Tung-Fu Hsieh

Department of Control Engineering, National Chiao-Tung University, 1001 Ta Hsueh Road, Hsinchu, Taiwan

Received 27 June 1994; revised 6 July 1995

Abstract

We propose an algorithm for computing the inverses of rational matrices and in particular the inverses of polynomial matrices. The algorithm is based on minimal state space realizations of proper rational matrices and the matrix inverse lemma and is implemented as a MATLAB[1] function. Experiments show that the algorithm gives accurate results for typical rational matrices that arise in analysis and design of linear multivariable control systems. Illustrative examples are given.

Keywords: Rational matrix inverse; Polynomial matrices; Linear multivariable systems

1. Introduction

Computation of inverses of rational matrices is needed in many linear multivariable feedback systems analysis and design problems. For example, it is needed in analysis and design using the inverse Nyquist array method [11, 14], in parametrization and design of decoupling controllers [5, 10], and in design using the QFT methods [9, 11].

When a rational matrix is expressed as a ratio of a numerator polynomial matrix and a denominator scalar polynomial, computation of its inverse essentially reduces to computation of the inverse of a polynomial matrix. Many algorithms for computing the inverses of polynomial matrices has been proposed. The methods proposed in [2, 6, 12, 15] compute the inverse by Cramer's rule in the sense that both the determinant and the adjoint matrix are explicitly computed. In [2, 6] the determinant and adjoint matrix are computed using a generalized Faddeev's recursive formula, while in [12, 15] they are computed by solving a set of interpolation equations. As noted in [15] a careful choice of the base points (for interpolation) is necessary to avoid ill-conditioned equations. In [12] the base points are chosen to be equally spaced points on the unit-circle to take advantage of the FFT algorithm. This then requires complex computations which may not be desirable for matrices with real coefficients. Another problem with the interpolation method is that the degrees of the determinant and the adjoint are usually not available, only upper bounds are. The interpolation thus involves redundant equations and polynomials with unnecessarily high degrees. In [3, 16], a division algorithm for

☆ Research sponsored by National Science Council grant NSC-81-0404-E009-544.

* Corresponding author. Fax: 886-35-715998; e-mail: calin@cc.nctu.edu.tw.

[1] MATLAB is a trademark of The MathWorks, Inc.

SSDI 0167-6911(95)00043-7

polynomial matrices is used to compute the inverse in *irreducible form* at the cost of increased computational complexity.

In this paper, we propose an algorithm for computing the inverses of rational matrices and in particular inverses of polynomial matrices. The algorithm computes neither the determinant nor the adjoint matrix. The algorithm is based on minimal state space realizations of proper rational matrices and the matrix inverse lemma [7, p. 656]. The algorithm is implemented as a MATLAB function that can be used with the associated Control System Toolbox [4]. Experiments show that the algorithm gives accurate results for typical rational matrices that arise in analysis and design of linear multivariable control systems.

The paper is organized as follows. In Section 2 we consider inversion of nonsingular proper rational matrices whose numerator polynomial matrices are either row-reduced or column-reduced to show the main idea of the approach and propose an algorithm. The result is then extended to general rational matrices including polynomial matrices in Section 3. Simple examples are given in Section 4 followed by a brief conclusion.

1.1. Notations and definitions

$\mathbb{R}[s]$($\mathbb{R}(s)$, $\mathbb{R}_p(s)$, resp.) denotes the set of polynomials (rational functions, proper rational functions, resp.) in s with real coefficients. For $a, b \in \mathbb{R}[s]$, $a|b$ denotes that a is a factor of b. The relative degree of $h \in \mathbb{R}(s)$ is defined as the degree of its denominator polynomial minus the degree of its numerator polynomial. The relative degree of $H(s) = [h_1(s) \cdots h_m(s)]^{\mathrm{T}} \in \mathbb{R}(s)^m$ is defined as the lowest relative degree of $h_i(s)$, $1 \leqslant i \leqslant m$. The ith column relative degree of an $m \times m$ rational matrix is the relative degree of its ith column.

2. Inverses of proper rational matrices

Suppose $P(s) \in \mathbb{R}_p(s)^{m\times m}$ is nonsingular. Write $P(s) = N(s)/d(s)$ where $d(s)$ is the monic least common denominator of the entries of $P(s)$ and $N(s) \in \mathbb{R}[s]^{m\times m}$ is the numerator polynomial matrix. Assume that $N(s)$ is column-reduced [7, p. 382]. Let r_i be the ith column relative degree of $P(s)$. Since $P(s)$ is proper, $r_i \geqslant 0$. Let $\alpha_i(s)$ be monic polynomials of degrees r_i which have no common factor with $d(s)$. Define

$$Q(s) := P(s)\,\mathrm{diag}(\alpha_1(s), \ldots, \alpha_m(s)) = \frac{1}{d(s)} N(s)\,\mathrm{diag}(\alpha_1(s), \ldots, \alpha_m(s)). \tag{2.1}$$

Note that in (2.1) we multiply the ith column of $P(s)$ by the polynomial $\alpha_i(s)$ so that $Q(s)$ is proper and each column of $Q(s)$ contains at least one entry that is not strictly proper.

Let $\{A, B, C, D\}$ be a minimal realization of $Q(s)$ where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{m\times n}$, and $D \in \mathbb{R}^{m\times m}$. Since $N(s)$ is column-reduced, the matrix $D = \lim_{s\to\infty} Q(s)$ is nonsingular. A minimal realization of $Q(s)^{-1}$ can be obtained by the following lemma.

Lemma 2.1. *Let $\{A, B, C, D\}$ be a minimal realization of $Q(s) \in \mathbb{R}_p(s)^{m\times m}$. If D is nonsingular, then $\{A - BD^{-1}C, BD^{-1}, -D^{-1}C, D^{-1}\}$ is a minimal realization of $Q(s)^{-1}$.*

Remark. The proof can be found in [1]; an alternative proof is given here as it is very simple.

Proof of Lemma 2.1. Since $Q(s) = C(sI - A)^{-1}B + D$, it follows from the matrix inverse lemma [7, p. 656] that $\{A - BD^{-1}C, BD^{-1}, -D^{-1}C, D^{-1}\}$ is a realization of $Q(s)^{-1}$. We show that this realization is controllable and observable. Since

$$[sI - A + BD^{-1}C \quad BD^{-1}] = [sI - A \quad B]\begin{bmatrix} I & 0 \\ D^{-1}C & D^{-1} \end{bmatrix}$$

and D^{-1} is nonsingular, $\text{rank}[sI - A + BD^{-1}C \;\; BD^{-1}] = \text{rank}[sI - A \;\; B]$ for all $s \in \mathbb{C}$. Since $\{A,B\}$ is controllable, it follows from PBH rank test [7, p. 366] that the realization is controllable. Observability can be similarly shown. Thus $\{A - BD^{-1}C, BD^{-1}, -D^{-1}C, D^{-1}\}$ is a minimal realization of $Q(s)^{-1}$. □

For simplicity let $A_1 = A - BD^{-1}C$, $B_1 = BD^{-1}$, $C_1 = -D^{-1}C$, and $D_1 = D^{-1}$, then we have $Q(s)^{-1} = D_1 + C_1(sI - A_1)^{-1}B_1$. Let us write

$$Q(s)^{-1} = \frac{W(s)}{\det(sI - A_1)} \tag{2.2}$$

where $W(s) = D_1 \det(sI - A_1) + C_1 \,\text{adj}(sI - A_1)B_1$. Note that expression (2.2) is the form we get when we use `ss2tf` in MATLAB to compute $Q(s)^{-1}$ from A_1, B_1, C_1, and D_1. The following theorem is used to obtain $P(s)^{-1}$ from (2.2).

Theorem 2.2. *For the $Q(s)$ defined in* (2.1) *and $Q(s)^{-1}$ given in* (2.2), *we have*
(a) $\prod_{i=1}^{m} \alpha_i(s) \mid \det(sI - A_1)$,
(b) $\prod_{k=1,k\neq i}^{m} \alpha_k(s) \mid W_{ij}(s)$, *for* $i,j = 1,\ldots,m$.

Proof. Let $L(s)$ and $R(s) \in \mathbb{R}[s]^{m\times m}$ be unimodular matrices such that

$$L(s)Q(s)R(s) =: \text{diag}(\varepsilon_1(s)/\psi_1(s), \ldots, \varepsilon_m(s)/\psi_m(s)) \tag{2.3}$$

is the Smith–McMillan form of $Q(s)$, where for $i = 1,\ldots,m-1$, $\varepsilon_i(s)$ and $\psi_i(s)$ are monic and coprime and

$$\varepsilon_i(s) \mid \varepsilon_{i+1}(s), \qquad \psi_{i+1}(s) \mid \psi_i(s).$$

Taking determinants of (2.1) and (2.3) we have

$$d(s)^m \prod_{i=1}^{m} \varepsilon_i(s) = k \det N(s) \prod_{i=1}^{m} \alpha_i(s) \prod_{i=1}^{m} \psi_i(s) \tag{2.4}$$

where the constant $k = \det L(s) \det R(s)$. Since $\prod_{i=1}^{m} \alpha_i(s)$ and $d(s)$ have no common factor, it follows that

$$\prod_{i=1}^{m} \alpha_i(s) \,\Bigg|\, \prod_{i=1}^{m} \varepsilon_i(s). \tag{2.5}$$

Now since $\text{diag}(\psi_m(s)/\varepsilon_m(s), \ldots, \psi_1(s)/\varepsilon_1(s))$ is the Smith–McMillan form of $Q(s)^{-1}$ and, by Lemma 2.1, $\{A_1, B_1, C_1, D_1\}$ is a minimal realization of $Q(s)^{-1}$, we have [7, p. 513]

$$\det(sI - A_1) = \prod_{i=1}^{m} \varepsilon_i(s). \tag{2.6}$$

Assertion (a) follows from (2.5) and (2.6). To prove (b), let $g(s)$ be the greatest common divisor of $\det N(s)$ and $d(s)$. Write

$$\det N(s) = \gamma(s) g(s) \tag{2.7}$$

so that $\gamma(s)$ and $d(s)$ are coprime, then (2.4) becomes

$$d(s)^m \prod_{i=1}^{m} \varepsilon_i(s) = k g(s) \gamma(s) \prod_{i=1}^{m} \alpha_i(s) \prod_{i=1}^{m} \psi_i(s). \tag{2.8}$$

Since $\gamma(s) \prod_{i=1}^{m} \alpha_i(s)$ and $d(s)$ are coprime, it follows from (2.8) and (2.6) that

$$\gamma(s) \prod_{i=1}^{m} \alpha_i(s) \,\Bigg|\, \det(sI - A_1). \tag{2.9}$$

Now from (2.1) and (2.7)

$$Q(s)^{-1} = \frac{d(s)}{g(s)\gamma(s)\prod_{i=1}^{m}\alpha_i(s)}\operatorname{diag}\left(\prod_{i=2}^{m}\alpha_i(s),\ldots,\prod_{i=1}^{m-1}\alpha_i(s)\right)\operatorname{adj}N(s). \tag{2.10}$$

Thus from (2.2) and (2.10) it follows that

$$W(s) = \frac{d(s)\det(sI - A_1)}{g(s)\gamma(s)\prod_{i=1}^{m}\alpha_i(s)}\operatorname{diag}\left(\prod_{i=2}^{m}\alpha_i(s),\ldots,\prod_{i=1}^{m-1}\alpha_i(s)\right)\operatorname{adj}N(s). \tag{2.11}$$

Since $\gamma(s)\prod_{i=1}^{m}\alpha_i(s)\,|\det(sI - A_1)$ and that $g(s)$ (being a divisor of $d(s)$) and $\alpha_i(s)$ are coprime, assertion (b) follows. □

Let us consider the computation of the inverse of $P(s)$. From (2.1) and (2.2)

$$P(s)^{-1} = \operatorname{diag}(\alpha_1(s),\ldots,\alpha_m(s))\frac{W(s)}{\det(sI - A_1)}. \tag{2.12}$$

From Theorem 2.2 and (2.12), it follows that

$$P(s)^{-1} = \bar{W}(s)/v(s) \tag{2.13}$$

where $\bar{W}(s) \in \mathbb{R}[s]^{m\times m}$, $v(s) \in \mathbb{R}[s]$ are given by

$$\bar{W}(s) = \left(\operatorname{diag}\left(\prod_{i=2}^{m}\alpha_i(s),\ldots,\prod_{i=1}^{m-1}\alpha_i(s)\right)\right)^{-1} W(s) \quad \text{and} \quad v(s) = \det(sI - A_1)\Big/\prod_{i=1}^{m}\alpha_i(s). \tag{2.14}$$

We note that the divisions of polynomials in (2.14) are actually removals of polynomial factors. We summarize the procedure developed so far into the following algorithm.

Algorithm 2.3.

data: $P(s) = N(s)/d(s)$, where $d(s)$ is the monic least common denominator of the entries of $P(s)$ and $N(s)$ is column-reduced.

step1: Determine the column relative degrees, r_i, of $P(s)$ and choose monic polynomials α_i so that $\alpha_i(s)$ and $d(s)$ are coprime and compute $Q(s)$ by (2.1).

step2: Compute a minimal realization $\{A, B, C, D\}$ of $Q(s)$.

step3: Compute A_1, B_1, C_1 and D_1 as defined and compute $Q(s)^{-1}$ from $\{A_1, B_1, C_1, D_1\}$ to get $Q(s)^{-1} = W(s)/\det(sI - A_1)$.

step4: Compute (2.14) by extracting polynomial factors from $\det(sI - A_1)$ and each row of $W(s)$ to get $P(s)^{-1} = \bar{W}(s)/v(s)$.

step5: Remove common factors that still remain in $v(s)$ and $\bar{W}(s)$ to obtain an irreducible form expression for $P(s)^{-1}$.

Comments

- All the computations involved in this algorithm can be carried out by using functions in MATLAB and the associated Control System Toolbox. Minimal realizations of $Q(s)$ can be obtained by using `tf2ss` and `minreal`; the rational matrix $Q(s)^{-1}$ can be computed by `ss2tf`; `deconv` can be used to remove factors from a polynomial.
- To reduce numerical problems that may occur in computations (especially for high-order case), the polynomials $\alpha_i(s)$ should be chosen so that its zeros are not close to any of the zeros of $d(s)$.
- The accuracy of solutions computed by this algorithm is limited mainly by the accuracy that can be achieved in computing a minimal realization of $Q(s)$. Computing minimal realizations by `tf2ss` and `minreal` usually gives reliable answer for McMillan degree $\leqslant 12$. If the zeros of $d(s)$ are distinct and known, accurate minimal realizations of much higher dimensions can be obtained by diagonal realization [7, p. 137] (modified

to account for complex-conjugate poles) in which the rank of the residue matrices are determined through singular value decomposition.
- If $N(s)$ is not column-reduced but is row-reduced, then $N(s)^T$ is column-reduced and the algorithm can be used to compute $(P(s)^T)^{-1} = (P(s)^{-1})^T$.

If $N(s)$ is neither column-reduced nor row-reduced, then there exists a unimodular matrix $U(s)$ such that $\tilde{N}(s) := N(s)U(s)$ is column-reduced [7]. The inverse of $\tilde{P}(s) := \tilde{N}(s)/d(s)$ can then be computed and the inverse of $P(s)$ is obtained by $P(s)^{-1} = U(s)\tilde{P}(s)^{-1}$. An algorithm for computing $U(s)$ is proposed in [8]. We note, however, that the algorithm proposed in [8] requires modifications and that it may suffer from large numerical error due to very small pivot elements. An alternative algorithm which is free from this small pivot induced problem is proposed in [13].

3. Extension to general rational matrices

We now consider the computation of inverses of general rational matrices, in particular, inverses of polynomial matrices. Let us consider first the polynomial matrices.

3.1. Inversion of polynomial matrices

Let $P(s) \in \mathbb{R}[s]^{m\times m}$ be nonsingular and without loss of generality assume that $P(s)$ is column-reduced. Let $\gamma_i \leqslant 0$ be the ith column relative degree of $P(s)$. Let $\beta_i(s)$ polynomials of degree $-\gamma_i$ such that $\beta_i(s)$ and $\det(P(s))$ is coprime. Let

$$Q(s) = P(s)\mathrm{diag}(1/\beta_1(s), \ldots, 1/\beta_m(s)). \tag{3.1}$$

Now $Q(s)$ is a proper rational matrix whose numerator polynomial matrix is column-reduced. Thus $Q(s)^{-1}$ can be computed by Algorithm 2.3 with $\alpha_i(s) = 1$. So let $Q(s)^{-1} = T(s)/v(s)$, where $T(s) \in \mathbb{R}[s]^{m\times m}$ and $v(s) \in \mathbb{R}[s]$. From (3.1) we have

$$Q(s)^{-1} = \mathrm{diag}(\beta_1(s), \ldots, \beta_m(s))P(s)^{-1} = \mathrm{diag}(\beta_1(s), \ldots, \beta_m(s))\frac{\mathrm{adj}(P(s))}{\det P(s)}. \tag{3.2}$$

Since $\beta_i(s)$ and $\det P(s)$ are coprime, the ith row of $Q(s)^{-1}$ contains the factor $\beta_i(s)$. Thus from (3.2)

$$P(s)^{-1} = T_1(s)/v(s) \tag{3.3}$$

where $T_1(s)$ is the polynomial matrix obtained from $T(s)$ by removing the polynomial factor $\beta_i(s)$ from its ith row. In computations, the polynomials $\beta_i(s)$ are chosen to have distinct zeros so that an algorithm for computing diagonal minimal realizations of $Q(s)$ can be used.

3.2. Inversion of nonproper rational matrices

Let $P(s) \in \mathbb{R}(s)^{m\times m}$ be nonsingular. Write $P(s) = N(s)/d(s)$, where $d(s)$ is the least common denominator of the entries of $P(s)$ and $N(s) \in \mathbb{R}[s]^{m\times m}$. Again, without loss of generality, assume that $N(s)$ is column-reduced. Let c_i be the ith column relative degree of $P(s)$. Let $\gamma_i := \max(0, -c_i)$. Choose $\beta_i(s) \in \mathbb{R}[s]$, monic and of degree γ_i such that $\det N(s)$ and $\beta_i(s)$ are coprime. Again $\beta_i(s)$ are chosen to have distinct zeros for computational reasons.

Let

$$Q(s) = P(s)\mathrm{diag}(1/\beta_1(s), \ldots, 1/\beta_m(s)). \tag{3.4}$$

Now $Q(s)$ is a proper rational matrix, thus $Q(s)^{-1}$ can be computed by Algorithm 2.3. Let

$$Q(s)^{-1} = T(s)/v(s),$$

where $T(s) \in \mathbb{R}[s]^{m\times m}$ and $v(s) \in \mathbb{R}[s]$. Then from (3.4)

$$P(s)^{-1} = T_1(s)/v(s),$$

where $T_1(s) \in \mathbb{R}[s]^{m\times m}$ is obtained from $T(s)$ by removing the polynomial factor $\beta_i(s)$ from its ith row.

4. Examples

An algorithm based on Algorithm 2.3 and the method discussed in the previous section for computing inverses of general rational matrices is implemented as a MATLAB function, `invratm`, for use with MATLAB/Control System Toolbox. Diagonal minimal realization is computed whenever it is reliable. The computation of a unimodular matrix for transforming a polynomial matrix into a column-reduced one is based on [8] with modifications and checks the pivot elements. The following examples are simple results obtained by using `invratm`. The examples are chosen simple so that comparisons with the exact solutions obtained by hand calculations can be made without too much effort.

Example 1. Let $P(s) = N(s)/d(s)$ where $d(s) = (s + 1.2)(s - 2)(s + 3.5)(s + 4)(s + 0.5)$, $N_{11}(s) = 2s + 8$, $N_{21}(s) = -4$, $N_{12}(s) = 3s + 1.5$, and $N_{22}(s) = s + 1.2$. This is a proper rational matrix with McMillan degree 10. The inverse of $P(s)$ computed by `invratm` is

$$\frac{1}{s^2 - 0.8s + 1.8}\begin{bmatrix} 0.5s^6 + 4.2s^5 + 8.795s^4 - 7.83s^3 - 39.94s^2 - 37.32s - 10.08 & -1.5s^6 - 11.55s^5 + 18.825s^4 + 32.888s^3 + 92.10s^2 + 61.35s + 12.60 \\ -2s^5 - 14.4s^4 - 17.9s^3 + 52.8s^2 + 96.4s + 33.6 & s^6 + 11.2s^5 + 37.75s^4 + 9.4s^3 - 153.8s^2 - 209.6s - 67.2 \end{bmatrix}$$

Compared with the exact solution, the largest coefficient error in the denominator is 1.2×10^{-12} and the largest coefficient error in the entries of numerator is 1.5×10^{-10}. For practical purposes the algorithm gives the exact solution.

Example 2. Let $P(s) = \text{diag}((s+2)^5,\ (s+4)^5,\ (s+5)^5)$. Let the inverse computed by `invratm` be denoted by $M(s)/a(s)$. The polynomial $a(s)$ is monic and of degree 15; the diagonal entries of $M(s)$ are polynomials of degree 10. The off-diagonal entries are:

$$M_{21} = 3.0994 \times 10^{-6}s^2 - 6.2227 \times 10^{-5}s + 8.8477 \times 10^{-4},$$

$$M_{31} = -1.2368 \times 10^{-6}s^3 + 2.3469 \times 10^{-5}s^2 - 3.3104 \times 10^{-4}s + 3.9210 \times 10^{-3},$$

$$M_{12} = -1.2666 \times 10^{-6}s^4 + 2.2270 \times 10^{-5}s^3 - 3.1271 \times 10^{-4}s^2, +3.6679 \times 10^{-3}s - 3.9086 \times 10^{-2},$$

$$M_{32} = -1.7136 \times 10^{-6}s^3 + 3.3543 \times 10^{-5}s^2 - 4.7636 \times 10^{-4}s + 5.6687 \times 10^{-3},$$

$$M_{13} = 1.2442 \times 10^{-6}s^3 - 2.4330 \times 10^{-5}s^2 + 3.4744 \times 10^{-4}s - 4.1378 \times 10^{-3},$$

$$M_{23} = 8.3447 \times 10^{-6}.$$

Compared with the exact solution, the largest relative error (i.e. error/true value) in the coefficients of $a(s)$ is 3.9×10^{-12}; the largest relative error in the coefficients of the diagonal entries of $M(s)$ is 1.8×10^{-9}.

5. Concluding remarks

We have proposed and implemented an algorithm for computing inverses of general nonsingular rational matrices and in particular the inverses of polynomial matrices. The algorithm is based on minimal state space realizations of proper rational matrices. It is interesting to note that the inverse of a polynomial matrix is obtained by first computing the inverse of a proper rational matrix not the other way around as is usually done.

References

[1] J.A. Ball, I. Gohberg and L. Rodman, *Interpolation of Rational Matrix Functions* (Birkhäuser, Basel, 1990).
[2] M. Buslowicz, Inversion of polynomial matrices, *Internat. J. Control* **33** (1980) 977–984.
[3] F.R. Chang, L.S. Shieh and B.C. Mcinnis, Inverse of polynomial matrices in the irreducible form, *IEEE Trans. Automat. Control* **32** (1987) 507–509.
[4] A. Grace, A.J. Laub, J.N. Little and C.M. Thompson, *Control System Toolbox User's Guide* (The MathWorks, Inc., Natick, MA, 1992).
[5] A.N. Gundes and C.A. Desoer, *Algebraic Theory of Linear Feedback Systems with Full and Decentralized Compensators*, Lecture Notes on Control and Information Sciences, Vol. 142 (Springer, Berlin, 1990).
[6] Y. Inouye, An algorithm for inverting polynomial matrices, *Internat. J. Control* **30** (1979) 989–999.
[7] T. Kailath, *Linear Systems* (Prentice-Hall, Englewood Cliffs, NJ, 1980).
[8] I.S. Krishnarao and C.T. Chen, Two polynomial matrix operations, *IEEE Trans. Automat. Control* **29** (1984) 346–348.
[9] G.B. Lamont, M.B. Leahy, Jr., D.E. Bossert and I.M. Horowitz, Design of discrete robust controller by QFT using PCT-approach, *Proc. ACC* (1991) 1989–1994.
[10] C.A. Lin and T.F. Hsieh, Decoupling controller design for linear multivariable plants, *IEEE Trans.* **36** (1991) 485–489.
[11] J.M. Maciejowski, *Multivariable Feedback Design* (Addison-Wesley, Reading, MA, 1989).
[12] O.T. Nam, Y. Ohta and T. Matsumoto, Inversion of rational matrices by using FFT algorithm, *Trans. IECE Japan* **E-61** (1978) 732–733.
[13] W.H.L. Neven and C. Praagman, Column reduction of polynomial matrices, *Linear Algebra Appl.* **188–189** (1993) 569–589.
[14] H.H. Rosenbrock, *Computer-Aided Control System Design* (Academic Press, New York, 1974).
[15] A. Schuster and P. Hippe, Inversion of polynomial matrices by interpolation, *IEEE Trans. Automat. Control* **37** (1992) 363–365.
[16] S.Y. Zhang, Inversion of polynomial matrices, *Internat. J. Control* **46** (1987) 33–37.