GENERALIZED SCHUR METHODS TO COMPUTE COPRIME FACTORIZATIONS OF RATIONAL MATRICES

A. Varga

DLR - Oberpfaffenhofen, Institute for Robotics and System Dynamics
P.O.B. 1116, D-82230 Wessling, GERMANY

Abstract. Numerically reliable state space algorithms are proposed for computing the following stable coprime factorizations of rational matrices: 1) factorizations with least order denominators; 2) factorizations with inner denominators; and 3) factorizations with proper stable factors. The new algorithms are based on a recursive generalized Schur algorithm for pole assignment. They are generally applicable regardless the original descriptor state space representation is minimal or not, or is stabilizable/detectable or not. The proposed algorithms are useful in solving various computational problems for both standard and descriptor system representations.

Keywords. Coprime factorization; descriptor systems; pole assignment; numerical algorithms.

1. INTRODUCTION

Let $G(s)$ or $G(z)$ be a given $p \times m$ rational *transfer-function matrix* (TFM) of a linear time-invariant continuous-time or discrete-time descriptor system, respectively, and let $G = (E, A, B, C, D)$ denote an equivalent nth order *regular* ($\det(\lambda E - A) \not\equiv 0$) descriptor representation satisfying $G(\lambda) = C(\lambda E - A)^{-1}B + D$, where λ is either s or z, depending on the type of the system. If G is not proper then E is singular and let $r = \operatorname{rank}(E)$. We say that G is *stable* if all its finite poles are in the stability region $\mathbb{C}^-$ of the complex plane $\mathbb{C}$. $\mathbb{C}^-$ is either the left open complex half-plane for a continuous-time system or the interior of the unit circle for a discrete-time system. The instability region $\mathbb{C}^+$ is the complement of $\mathbb{C}^-$ with respect to $\mathbb{C}$. We denote with $\Lambda(A, E)$ the generalized eigenvalues of the pair (E, A).

A proper and stable TFM G is *inner* if $G^T(-s)G(s) = I$ in continuous-time or $G^T(1/z)G(z) = I$ in discrete-time. A fractional representation of G in the form $G = NM^{-1}$ with N and M stable rational matrices, is called a *right coprime factorization* (RCF) if there exist stable rational matrices U and V such that $UN + VM = I$. Analogously, a fractional representation of G in the form $G = M^{-1}N$ with N and M stable rational matrices, is called a *left coprime factorization* (LCF) if there exist stable rational matrices U and V such that $NU + MV = I$. Several special factorizations could be of interest in particular applications.

The simplest factorization to obtain is when M is proper and N is proper or improper depending on if the original G is proper or not. This factorization with M having possibly least order, is useful as a preliminary or as a final step in computing some other factorizations. A particular case of this factorization is when M is *inner*. This factorization has several important applications in evaluating norms of TFMs or in computing spectral factors of TFMs. Provided G is square ($p = m$), coprime factorizations of its inverse G^{-1} are useful to compute alternative factorizations of rational TFMs, as for instance factorizations with minimum-phase factors or inner-outer factorizations. Factorizations in which both N and M are proper rational matrices can be viewed as alternative representations of rational matrices. This factorization is potentially useful in performing order reduction of descriptor systems by using the coprime factors reduction approach analogously as in the case of standard systems (Liu and Anderson, 1986; Varga, 1993*a*).

In this paper we propose numerically reliable state space algorithms for computing three of the above mentioned RCFs, namely the factorizations with: 1) least order proper M, 2) with M inner, and 3) with both M and N proper. The same algorithms can be also used to compute LCFs by applying them to the dual TFM G^T. The new procedures are generally applicable regardless the original descriptor state space representation of G is minimal or not, or is stabilizable/detectable or not. They are well suited for robust software implementations. The proposed algorithms represent generalizations of similar algorithms for standard systems (Varga, 1993*a*; Varga, 1993*b*) and are based on a recursive generalized Schur technique for pole assignment of descriptor systems (Varga, 1995). The presented techniques can be also seen as extensions of the general recursive factorization approach introduced by Van Dooren (1990).

2. UPDATING FRACTIONAL REPRESENTATIONS

The factorization algorithms proposed in this paper rely on simple facts concerning fractional representations.

Fact 1. *Any rational matrix G with a stabilizable state-space realization (E, A, B, C, D) has a RCF given by the following choice of the factors* (Wang and Balas, 1989)

$$\begin{array}{rcl} N & = & (E,\, A+BF,\, BW,\, C+DF,\, DW) \\ M & = & (E,\, A+BF,\, BW,\, F,\, W) \end{array} \qquad (1)$$

where F is chosen such that all finite eigenvalues of the pair $(E, A+BF)$ (at most r) are stable, the pencil $A+BF-\lambda E$ is regular and W is an arbitrary invertible matrix.

Particular factorizations with special properties, as for instance with inner denominator or with proper factors, can be determined by suitably choosing F and W.

The algorithms proposed in this paper use implicitly the more general expressions for the factors

$$\begin{aligned} N &= (UEV,\, U(A+BF)V,\, UBW,\, (C+DF)V,\, DW) \\ M &= (UEV,\, U(A+BF)V,\, UBW,\, FV,\, W) \end{aligned} \tag{2}$$

where U and V are orthogonal transformation matrices (usually not accumulated), which are used to obtain the resulting matrices in particular condensed forms. ■

Fact 2. *If $G = N_1 M_1^{-1}$ and $N_1 = N_2 M_2^{-1}$, then G has the fractional representation $G = NM^{-1}$, where $N = N_2$ and $M = M_1 M_2$.*

This simple fact allows us to obtain explicit formulas to update partial factorizations by using simple state space formulas. Let N_1 and M_1 be the factors computed as

$$\begin{aligned} N_1 &= (E,\, A+BF_1,\, BW_1,\, C+DF_1,\, DW_1) \\ M_1 &= (E,\, A+BF_1,\, BW_1,\, F_1,\, W_1) \end{aligned} \tag{3}$$

and let N_2 and M_2 be the factors of N_1 computed as

$$\begin{aligned} N_2 &= (E,\, A+BF,\, BW,\, C+DF,\, DW) \\ M_2 &= (E,\, A+BF,\, BW,\, F_2,\, W_2) \end{aligned} \tag{4}$$

where

$$\begin{aligned} F &= F_1 + W_1 F_2 \\ W &= W_1 W_2 \end{aligned} \tag{5}$$

It easy to verify that the product $M_1 M_2$ is given by

$$M_1 M_2 = (E,\, A+BF,\, BW,\, F,\, W) \tag{6}$$

and thus equations (5) serve as explicit updating formulas of fractional representations. These formulas can be extended in a straightforward way to include arbitrary coordinate transformation matrices. All factorization algorithms presented in the paper rely on the use of such updating formulas. If $W_1 = I$ and $W_2 = I$, then the updating formulas reduce to a very simple form

$$F = F_1 + F_2, \tag{7}$$

which is used in some of proposed algorithms. ■

Fact 3. *An implicit updating technique of fractional representations is based on the following evident identities:*

$$\begin{bmatrix} G \\ I \end{bmatrix} = \begin{bmatrix} N_1 \\ M_1 \end{bmatrix} M_1^{-1} = \begin{bmatrix} N_2 \\ M_1 M_2 \end{bmatrix} (M_1 M_2)^{-1} \tag{8}$$

It can be easily seen that the two successive numerator factors $\begin{bmatrix} N_1 \\ M_1 \end{bmatrix}$ and $\begin{bmatrix} N_2 \\ M_1 M_2 \end{bmatrix}$ of the extended TFM $\begin{bmatrix} G \\ I \end{bmatrix}$ contain the elements of the successive factorizations $G = N_1 M_1^{-1} = N_2 (M_1 M_2)^{-1}$. This implicit updating procedure is especially useful when combining different factorization algorithms because it is applicable even if the factors computed by different algorithms have different orders or if coordinate transformations are present in the representations of factors. Notice that the use of the updating formulas (5) requires that the two successive state space representations (3) and (4) have the same order. Otherwise it is not possible to obtain explicit updating formula as in (5) for the state feedback matrix. ■

Remark. If the TFM G is square, any algorithm to compute RCFs can be used to compute a LCF $G = M^{-1}N$ in which both factors are minimum-phase. This can be done by applying the algorithm to the inverse system

$$G^{-1} = \left(\begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \begin{bmatrix} 0 \\ I \end{bmatrix}, [\,0\ -I\,],\, 0 \right) \tag{9}$$

to compute the RCF $G^{-1} = \widetilde{N}\widetilde{M}^{-1}$ in the form (1) by using a feedback matrix partitioned as $F = [\,F_1\ \ F_2\,]$. It easy to verify that the factors of the minimum-phase LCF of G are

$$\begin{aligned} N &= (\,E,\ A,\ B,\ W^{-1}(C+F_1),\ W^{-1}(D+F_2)\,) \\ M &= \left(\begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \begin{bmatrix} 0 \\ I \end{bmatrix}, -\left[\, W^{-1}F_1\ W^{-1}F_2 \,\right],\, W^{-1} \right) \end{aligned}$$

with both $N = \widetilde{N}^{-1}$ and $M = \widetilde{M}^{-1}$ having zeros in $\mathbb{C}^-$. Note that in some factorizations, G should be expressed as $G = \widetilde{M}N$. In this case we use directly the expression of $\widetilde{M}$ resulted from (1)

$$\widetilde{M} = \left(\begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} A & B \\ C+F_1 & D+F_2 \end{bmatrix}, \begin{bmatrix} 0 \\ W \end{bmatrix}, [\,F_1\ F_2\,], W \right)$$

■

3. RCF WITH LEAST ORDER DENOMINATOR

In this section we propose an algorithm to compute a RCF of G with a least order M. The new algorithm can handle even the case when the original descriptor system representation is not stabilizable. The basis for our algorithm is a pole assignment procedure described in (Varga, 1983; Varga, 1995). This algorithm has the ability to keep unaltered the stable eigenvalues of the pair (E, A) and to move only the unstable ones to stable locations by choosing an appropriate feedback matrix F. An additional useful feature of this algorithm is that simultaneously with the stabilizing F, it determines the *generalized real Schur form* (GRSF) of the pair $(E, A+BF)$. This makes possible to extract easily a minimal realization for the denominator factor M. The following implementable state space algorithm can be used to compute a RCF of a rational TFM G.

GRCF Algorithm.

1. Find orthogonal matrices Q and Z to reduce the pair (E, A) to the ordered GRSF (Moler and Stewart, 1973; Van Dooren, 1981)

$$\widetilde{E} = QEZ = \begin{bmatrix} E_{11} & E_{12} \\ 0 & E_{22} \end{bmatrix}, \ \widetilde{A} = QAZ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \tag{10}$$

   where $E_{11}, A_{11} \in \mathbb{R}^{q\times q}$, Q and Z are orthogonal matrices, $\Lambda(A_{11}, E_{11}) \subset \mathbb{C}^- \cup \{\infty\}$ and $\Lambda(A_{22}, E_{22}) \subset \mathbb{C}^+$. Compute $\widetilde{B} = QB$, $\widetilde{C} = CZ$ and set $\widetilde{F} = 0$.

2. If $q = n$, go to 7.
3. Let (δ, α) be the last diagonal blocks of $(\widetilde{E}, \widetilde{A})$ of order k and let β be the $k \times m$ matrix formed from the last k rows of $\widetilde{B}$. If $\|\beta\| \le \epsilon$ (a given tolerance), then $n \leftarrow n - k$ and go to 2.
4. Choose a $k \times k$ matrix γ such that $\Lambda(\gamma) \subset \mathbb{C}^-$ and compute $\varphi = \beta^+(\delta\gamma - \alpha)$. Set $K = [\,0\ \ \varphi\,]$.
5. Compute $\widetilde{A} \leftarrow \widetilde{A} + \widetilde{B}K$, $\widetilde{F} \leftarrow \widetilde{F} + K$.
6. Compute the orthogonal matrices $\widetilde{Q}$ and $\widetilde{Z}$ to move the last blocks of $(\widetilde{E}, \widetilde{A})$ to positions $(q+1, q+1)$ by interchanging the diagonal blocks of the GRSF. Compute $\widetilde{E} \leftarrow \widetilde{Q}\widetilde{E}\widetilde{Z}$, $\widetilde{A} \leftarrow \widetilde{Q}\widetilde{A}\widetilde{Z}$, $\widetilde{B} \leftarrow \widetilde{Q}\widetilde{B}$, $\widetilde{C} \leftarrow \widetilde{C}\widetilde{Z}$, $\widetilde{F} \leftarrow \widetilde{F}\widetilde{Z}$. Put $q \leftarrow q + k$ and go to 2.
7. Put $N = (\widetilde{E}, \widetilde{A}, \widetilde{B}, \widetilde{C} + D\widetilde{F}, D)$, $M = (\widetilde{E}, \widetilde{A}, \widetilde{B}, \widetilde{F}, I)$.

This algorithm can be viewed as a recursive updating procedure of an initial fractional representation $G = N_0 M_0^{-1}$ with $N_0 = G$ and $M_0 = I$, by using the simple updating formula (7) combined with orthogonal coordinate transformations performed on the matrices of partial factorizations. The matrix pair $(\widetilde{E}, \widetilde{A})$ in the initial factorization of G is in a GRSF (computed at step 1) and this form is preserved at subsequent steps. The resulting final pair $(\widetilde{E}, \widetilde{A})$ is therefore in a GRSF and if the original system is stabilizable, then $\widetilde{E}$ and $\widetilde{A}$ contain the matrices UEV and $U(A+BF)V$, respectively, where U and V are the accumulated orthogonal transformations performed at steps 1 and 6 of the algorithm, and F is the stabilizing feedback matrix $\widetilde{F}V^T$. If the original system is not stabilizable, then the unstabilizable blocks are detected at step 3 and the corresponding unstabilizable parts are deflated by simply decreasing the order of system with k. If unstabilizable blocks are detected by the algorithm then the resulting factors have order less than n.

One of the advantages of the resulting form of matrices of the computed factors is that a minimal realization of M can be easily determined. The resulting $\widetilde{F}$ has always the form

$$\widetilde{F} = [\,0\ \widetilde{F}_2\,], \tag{11}$$

where the number of columns of $\widetilde{F}_2$ equals the number of unstable controllable generalized eigenvalues of the pair (E, A). By partitioning accordingly the resulting $\widetilde{E}$, $\widetilde{A}$ and $\widetilde{B}$

$$\widetilde{E} = \begin{bmatrix} \widetilde{E}_{11} & \widetilde{E}_{12} \\ 0 & \widetilde{E}_{22} \end{bmatrix},\ \widetilde{A} = \begin{bmatrix} \widetilde{A}_{11} & \widetilde{A}_{12} \\ 0 & \widetilde{A}_{22} \end{bmatrix},\ \widetilde{B} = \begin{bmatrix} \widetilde{B}_1 \\ \widetilde{B}_2 \end{bmatrix}, \tag{12}$$

then $(\widetilde{E}_{22}, \widetilde{A}_{22}, \widetilde{B}_2, \widetilde{F}_2, I)$ is a minimal realization of M. Because $\widetilde{E}_{22}$ is invertible, the TFM M is always proper (in fact biproper). However generally the factor N is not proper if the original descriptor representation has impulsive modes.

If in the descriptor representation of G all unstable controllable eigenvalues of the pair (E, A) are observable, then M has the least order of all possible proper denominators in a RCF of G. Notice however that the order of the minimal realization of M can be higher then the least possible order if some unstable eigenvalues of (E, A) are controllable but not observable.

The GRCF Algorithm is based on a generalization of a pole assignment algorithm for standard systems (Varga, 1981). The roundoff error analysis of that algorithm (Varga, 1982) revealed that if each partial feedback K computed at step 4 satisfies $\|K\| \leq \|A\|/\|B\|$, then the pole assignment algorithm is numerically backward stable. This condition is also applicable in our case, because it is independent of the presence of E. We note however that unfortunately this condition can not be always fulfilled if large gains are necessary to stabilize the system. This can arise either if the unstable poles are too "far" from the stable region or if these poles are weekly controllable.

4. RCF WITH INNER DENOMINATOR

We assume in this section that G has no poles on the imaginary axis in continuous-time case or on the unit circle in the discrete-time case. The algorithm to compute the *right coprime factorization with inner denominator* (RCFID) of a rational TFM G use recursively the following formulas to compute the RCFID of a particular class of systems.

Fact 4. *Let $G = (E, A, B, C, D)$ a controllable descriptor representation with E non-singular and $\Lambda(E, A) \in \mathbb{C}^+$. Then $M = (E, A+BF, BW, F, W)$ is* inner *by choosing F and W as:*

$$\left.\begin{array}{l} F = -B^T(YE^T)^{-1}, \quad W = I \\ AYE^T + EYA^T - BB^T = 0 \end{array}\right\} \textit{(continuous-time)}$$

$$\left.\begin{array}{l} F = -B^T(EYE^T + BB^T)^{-1}A \\ W = (I + B^T(EYE^T)^{-1}B)^{-1/2} \\ AYA^T - BB^T = EYE^T \end{array}\right\} \textit{(discrete-time)}$$

The above expressions represent straightforward transcriptions of analogous formulas for standard systems (Varga, 1993*b*). In the following algorithm, we use these formulas (at step 4) to compute inner denominators for simple systems of orders at most two.

GRCFID Algorithm.

1. Find orthogonal matrices Q and Z to reduce the pair (E, A) to the ordered GRSF (10), where $E_{11}, A_{11} \in \mathbb{R}^{q\times q}$, Q and Z are orthogonal matrices, $\Lambda(A_{11}, E_{11}) \subset \mathbb{C}^- \cup \{\infty\}$ and $\Lambda(A_{22}, E_{22}) \subset \mathbb{C}^+$. Compute $\widetilde{B} = QB$, $\widetilde{C} = CZ$. Set $\widetilde{F} = 0$, $\widetilde{W} = I$.
2. If $q = n$, go to 7.
3. Let (δ, α) be the last diagonal blocks of $(\widetilde{E}, \widetilde{A})$ of order k and let β be the $k \times m$ matrix formed from the last k rows of $\widetilde{B}$. If $\|\beta\| \leq \epsilon$ (a given tolerance), then $n \leftarrow n - k$ and go to 2.
4. For the system $(\delta, \alpha, \beta, *, *)$ compute φ and V such that $(\delta, \alpha + \beta\varphi,\ \beta V, \varphi, V)$ is inner. Set $K = [\,0\ \varphi\,]$.
5. Compute $\widetilde{A} \leftarrow \widetilde{A} + \widetilde{B}K$, $\widetilde{F} \leftarrow \widetilde{F} + \widetilde{W}K$, $\widetilde{W} \leftarrow \widetilde{W}V$.
6. Compute the orthogonal matrices $\widetilde{Q}$ and $\widetilde{Z}$ to move the last blocks of $(\widetilde{E}, \widetilde{A})$ to positions $(q+1, q+1)$ by interchanging the diagonal blocks of the GRSF. Compute $\widetilde{E} \leftarrow \widetilde{Q}\widetilde{E}\widetilde{Z}$, $\widetilde{A} \leftarrow \widetilde{Q}\widetilde{A}\widetilde{Z}$, $\widetilde{B} \leftarrow \widetilde{Q}\widetilde{B}$, $\widetilde{C} \leftarrow \widetilde{C}\widetilde{Z}$, $\widetilde{F} \leftarrow \widetilde{F}\widetilde{Z}$. Put $q \leftarrow q + k$ and go to 2.
7. $N = (\widetilde{E}, \widetilde{A}, \widetilde{B}\widetilde{W}, \widetilde{C} + D\widetilde{F}, D\widetilde{W})$, $M = (\widetilde{E}, \widetilde{A}, \widetilde{B}\widetilde{W}, \widetilde{F}, \widetilde{W})$.

A minimal realization for the inner factor M is given by $(\widetilde{E}_{22}, \widetilde{A}_{22}, \widetilde{B}_2\widetilde{W}, \widetilde{F}_2, \widetilde{W})$ and can be determined from the partitioning (11) and (12) of the resulting $\widetilde{F}$, $\widetilde{E}$, $\widetilde{A}$ and $\widetilde{B}$.

The above algorithm relies exclusively on reliable numerical techniques. It can be viewed as a pole assignment algorithm which assigns the unstable poles in symmetrical positions with respect to the imaginary axis in the continuous-time case or the unit circle in the discrete-time case. Because practically there is no freedom in assigning the poles, it is to be expected that the algorithm perform in a numerically stable way only if the norms of the elementary feedback matrices K computed at step 4 are not too high.

Remark. If the TFM G is square, the GRCFID Algorithm can be used to compute an *inner-outer* factorization $G = MN$, where M is inner and N is outer (minimum-phase and stable). This can be done by applying the algorithm to the inverse system (9). ■

5. RCF WITH PROPER FACTORS

If the given system (E, A, B, C, D) has impulsive modes, that is, the finite generalized eigenvalues are fewer than $r = \mathrm{rank}(E)$, then the N factor resulting from the GRCF Algorithm has impulsive modes too and therefore is not proper. A trivial example is when G is a polynomial matrix in which case the factors are simply $N = G$ and $M = I$. However if G is proper, then the numerator factor N computed by the GRCF Algorithm results also proper. These observations lead to the following conceptually simple approach to compute a *proper right coprime factorization* (PRCF), in which both factors are proper and stable:

1. Compute a factorization of G in the form $G = N_1 M_1^{-1}$, where both factors are proper but possibly unstable.
2. Compute the RCF $\begin{bmatrix} N_1 \\ M_1 \end{bmatrix} = \begin{bmatrix} N \\ M \end{bmatrix} \widetilde{M}_2^{-1}$ by using the GRCF Algorithm.

It is easy to see that $G = NM^{-1}$ is the desired PRCF.

The above two steps can be related to the two main steps of an S-stabilization (*strong-stabilization*) algorithm proposed recently in (Varga, 1995). To simplify the algorithm presentation, we assume in what follows that the given descriptor representation of G is *stabilizable* and *impulse-controllable*, that is, $\operatorname{rank}([\, E \;\; B \,]) = n$. In the mentioned S-stabilization algorithm a preliminary state feedback F_1 is determined to move all impulsive modes to finite locations. Then a second partial feedback F_2 is used to stabilize the modified impulse-free system. These partial feedback matrices can be used then to define the PRCF of G according to (1) by using the updating formula (7).

We present below a procedure to compute the PRCF of G based on the above approach. For a detailed description of all computational steps see (Varga, 1994), where the most general case is considered (both the stabilizability and impulse-controllability assumptions are removed).

PGRCF Algorithm.

1. Find orthogonal matrices U and V such that

$$E \leftarrow UEV := \begin{bmatrix} E_{11} & 0 \\ 0 & 0 \end{bmatrix},$$

$$A \leftarrow UAV := \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad B \leftarrow UB := \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

where $E_{11} \in \mathbb{R}^{r,r}$ is non-singular and upper-triangular and B_2 has full row rank. Compute $C \leftarrow CV$.

2. Determine $F_1 = [\, 0 \;\; F_{12} \,]$ such that $A_{22} + B_2 F_{12}$ is non-singular and set $A \leftarrow A + BF_1$.

3. Apply the GRCF Algorithm to

$$\begin{bmatrix} N_1 \\ M_1 \end{bmatrix} := \left(E,\; A,\; B,\; \begin{bmatrix} C + DF_1 \\ F_1 \end{bmatrix},\; \begin{bmatrix} D \\ I \end{bmatrix} \right)$$

to compute the factors

$$\begin{bmatrix} N \\ M \end{bmatrix} = \left(\widetilde{E},\; \widetilde{A},\; \widetilde{B},\; \begin{bmatrix} \widetilde{C} \\ \widetilde{F} \end{bmatrix},\; \begin{bmatrix} D \\ I \end{bmatrix} \right)$$

$$\widetilde{M}_2 = (\widetilde{E},\; \widetilde{A},\; \widetilde{B},\; F_2,\; I)$$

The algorithm uses exclusively numerically reliable techniques to perform each computational step. The details of steps 1 and 2 are discussed in (Varga, 1994). Before performing step 3 it is possible to reduce first the pencil $\lambda E - A$ to a block upper triangular form by annihilating the submatrix A_{21} of A and preserving simultaneously the upper triangular shape of E. This form allows to extract immediately the non-dynamic part of the system by including it into an appropriate feedthrough matrix. Step 3 can be then performed on a descriptor representation of smaller order. This approach can be particularly useful when the computed PRCF is intended to be used for coprime factors model reduction (Liu and Anderson, 1986).

Remark. From the matrices computed by the PGRCF Algorithm it is not possible to extract immediately a least order minimal realization for M. If such a realization for M is of interest, then an alternative procedure, described also in (Varga, 1994), is more appropriate. ∎

6. CONCLUSIONS

Efficient numerically reliable algorithms for computing several RCFs have been proposed. They are well suited for robust and modular software implementations. The algorithms are based on a recursive generalized Schur technique for pole assignment by using proportional state-feedback. This technique can be extended in a straightforward way to use derivative state-feedback too, leading to alternative algorithms for computing RCFs. The derivative feedback also allows to compute other useful factorizations as for instance RCFs with polynomial factors. It is still an open question the existence of general recursive algorithms for computing other factorizations as for example the inner–outer factorization for non-square systems, the normalized coprime factorization, the spectral and J–lossless factorizations.

7. REFERENCES

Liu, Y. and Anderson, B. D. O. (1986). ‘Controller reduction via stable factorization and balancing’. *Int. J. Control* **44**, 507–531.

Moler, C. B. and Stewart, G. W. (1973). ‘An algorithm for generalized matrix eigenvalue problem’. *SIAM J. Control Optim.* **10**, 241–256.

Van Dooren, P. (1981). ‘A generalized eigenvalue approach for solving Riccati equations’. *SIAM J. Sci. Stat. Comput.* **2**, 121–135.

Van Dooren, P. (1990). ‘Rational and polynomial matrix factorizations via recursive pole-zero cancellation’. *Lin. Alg. & Appl.* **137/138**, 663–697.

Varga, A. (1981). ‘A Schur method for pole assignment’. *IEEE Trans. Autom. Control* **AC-26**, 517–519.

Varga, A. (1982). The numerical stability of an algorithm for pole assignment. In G. Leininger (Ed.). ‘Proc. 2-nd IFAC CADCS Symp., West Lafayette, Indiana’. Pergamon Press, Oxford.

Varga, A. (1983). A pole assignment algorithm for systems in generalized state-space form. In ‘Proc. of 5-th Int. Conf. Control and Informational Systems in Industry, Bucharest, Romania’.

Varga, A. (1993*a*). ‘Coprime factors model reduction based on accuracy enhancing techniques’. *Systems Analysis Modelling and Simulation* **11**, 303–311.

Varga, A. (1993*b*). A Schur method for computing coprime factorizations with inner denominators and applications in model reduction. In ‘Proc. 1993 American Control Conference, San Francisco, CA’. pp. 2130–2131.

Varga, A. (1994). Generalized Schur methods to compute coprime factorizations of rational matrices. In ‘Proc. 1st Asian Control Conference, Tokyo, Japan’. Vol. 3. pp. 89–92.

Varga, A. (1995). ‘On stabilization of descriptor systems’. *Systems & Control Letters* **24**, 133–138.

Wang, F. Y. and Balas, M. J. (1989). ‘Doubly coprime fractional representations of generalized dynamical systems’. *IEEE Trans. Autom. Control* **AC-34**, 733–744.