Andràs Varga

Computation of irreducible generalized state-space realizations

Kybernetika, Vol. 26 (1990), No. 2, 89--106

Persistent URL: http://dml.cz/dmlcz/124686

KYBERNETIKA — VOLUME 26 (1990), NUMBER 2

COMPUTATION OF IRREDUCIBLE GENERALIZED STATE-SPACE REALIZATIONS

ANDRAS VARGA

In this paper, an efficient, numerically stable procedure is presented for the computation of irreducible generalized state-space realizations from non-minimal ones. The order reduction is performed by removing successively the uncontrollable and the unobservable parts of the system. Each reduction is accomplished by the same basic algorithm which deflates the uncontrollable part of the system using orthogonal similarity transformations. Applications of the proposed procedure are also presented.

1. INTRODUCTION

Consider the linear time-invariant *generalized state-space model* (GSSM)

$$\begin{aligned} \lambda E\, x(t) &= A\, x(t) + B\, u(t) \\ y(t) &= C\, x(t) \end{aligned} \tag{1}$$

where x, u, and y are the n-dimensional state vector, the m-dimensional input vector, and the p-dimensional output vector, respectively, and where λ is the differential operator $\mathrm{d}/\mathrm{d}t$ for a continuous system or the advance operator z for a discrete system. The matrices E, A, B and C have appropriate dimensions, E and A being square. The system (1) will be referred to alternatively as the triple $\{\lambda E - A, B, C\}$. If the matrix E is singular, the system (1) is also called *singular* or *descriptor* system.

Linear time-invariant systems can also be represented by *differential* or *difference state-space models* (DSSM) of the form

$$\begin{aligned} T(\lambda)\, z(t) &= U(\lambda)\, u(t) \\ y(t) &= V(\lambda)\, z(t) + W(\lambda)\, u(t) \end{aligned} \tag{2}$$

where z is a q-dimensional "internal" state vector, u and y are as above, and $T(\lambda)$, $U(\lambda)$, $V(\lambda)$, $W(\lambda)$ are polynomial matrices having appropriate dimensions with $T(\lambda)$ square. The system (2) will be alternatively denoted by $\{T(\lambda), U(\lambda), V(\lambda), W(\lambda)\}$.

A third frequently used representation of linear constant systems is by the *transfer-*

function matrix model (TFMM)

$$\overline{Y}(\lambda) = G(\lambda)\,\overline{U}(\lambda) \tag{3}$$

where $\overline{U}(\lambda)$ and $\overline{Y}(\lambda)$ are the transforms of the input and output vectors, respectively (the Laplace transform for continuous systems or the Z-transform for discrete systems), and where $G(\lambda)$ is a $p \times m$ rational matrix.

If (1), (2) and (3) correspond to the same system, we have the following basic relations

$$G(\lambda) = C(\lambda E - A)^{-1} B \tag{4}$$

$$G(\lambda) = V(\lambda)\, T^{-1}(\lambda)\, U(\lambda) + W(\lambda) \tag{5}$$

For a given DSSM or TFMM, the determination of a corresponding minimal order (or irreducible) GSSM is known as the *minimal realization problem* (MRP). The MRP has no unique solution. If $\{\lambda E - A, B, C\}$ and $\{\lambda \hat{E} - \hat{A}, \hat{B}, \hat{C}\}$ have the same order and correspond to the same TFMM, then there exist invertible matrices Q and Z such that

$$\lambda \hat{E} - \hat{A} = Q(\lambda E - A)\, Z\,, \quad \hat{B} = QB\,, \quad \hat{C} = CZ \tag{6}$$

Two GSSM will be called *similar* if their matrices are related as in (6) and therefore the transformation (6) will be called *system similarity transformation*. If Q and Z are orthogonal matrices, the transformation will be called *orthogonal system similarity transformation*.

In this paper we describe an efficient, numerically stable procedure for the computation of irreducible GSSM from non-minimal ones. The order reduction is performed by removing successively the uncontrollable and then the unobservable parts of the system. Each reduction step is accomplished by using a new numerically stable algorithm which separates the uncontrollable part of a GSSM. This basic algorithm uses exclusively orthogonal system similarity transformations and is an efficient alternative to existing procedures [1], [2], [3]. The proposed algorithms are presented in Section 2.

The main applications of the new algorithms are: 1) the solution of the MRP; 2) the computation of minimal order inverses of linear systems; and 3) the evaluation of the transfer-function matrices of GSSM. These applications are presented in Section 3. Numerical examples are given in Section 4.

Notations and definitions

Throughout the paper $A \in \mathbb{R}^{m \times n}$ denotes an $m \times n$ matrix with real elements. We use A^{T} for the transpose of A. I or I_n denote identity matrices of known order or of order n, respectively. 0_{mn} denotes an $m \times n$ null matrix. A square matrix Q is orthogonal if $Q^{\mathrm{T}}Q = I$. im A and ker A denote, respectively, the image and the kernel of A. For two subspaces $\mathscr{X}$ and $\mathscr{Y}$, $A\mathscr{X}$ is the image of $\mathscr{X}$ under A, and $\mathscr{X} + \mathscr{Y}$ is the sum of subspaces $\mathscr{X}$ and $\mathscr{Y}$. A polynomial matrix is called regular when it is

square and has a nonzero determinant. A deflating subspace $\mathscr{X}$ of a regular pencil $\lambda E - A$ satisfies $\dim(A\mathscr{X} + E\mathscr{X}) = \dim \mathscr{X}$, where dim stands for "dimension of". $O(\varepsilon)$ means a quantity of the order of ε.

2. THE IRREDUCIBLE REALIZATION PROCEDURE

In this section we present a numerically stable procedure to compute an irreducible GSSM from a non-minimal one. The order reduction is performed by removing successively the uncontrollable and then the unobservable parts of the system. Each reduction step is accomplished by the same basic procedure which deflates the uncontrollable part of a GSSM using orthogonal system similarity transformations.

The definitions used in this section closely follow the work of Van Dooren [1]. The controllable and unobservable subspaces of the n-dimensional state-space $\mathscr{X}$ of GSSM $\{\lambda E - A, B, C\}$ can be defined as the deflating subspaces $\mathscr{C}$ and $\bar{\mathscr{O}}$, respectively, which satisfy

$$\mathscr{C} = \inf\{\mathscr{S} \mid \dim(E\mathscr{S} + A\mathscr{S}) = \dim \mathscr{S};\ \operatorname{im} B \subset E\mathscr{S} + A\mathscr{S}\} \tag{7}$$

$$\bar{\mathscr{O}} = \sup\{\mathscr{S} \mid \dim(E\mathscr{S} + A\mathscr{S}) = \dim \mathscr{S};\ \mathscr{S} \in \ker C\}\,. \tag{8}$$

The system is said *controllable* when its controllable subspace $\mathscr{C}$ has dimension n, and *observable* when its unobservable subspace $\bar{\mathscr{O}}$ has zero dimension. We shall assume that the pencil $\lambda E - A$ is regular.

Let r be the dimension of $\mathscr{C}$ defined by (7) and let Z and Q be orthogonal transformation matrices whose first r columns span $\mathscr{C}$ and $E\mathscr{C} + A\mathscr{C}$, respectively. Then we can transform the system $\{\lambda E - A, B, C\}$ as

$$Q^{\mathrm{T}}(\lambda E - A)Z = \left[\begin{array}{c|c} \underbrace{\begin{matrix}\lambda E_c - A_c \\ \hline 0\end{matrix}}_{r} & \underbrace{\begin{matrix}* \\ \hline \lambda E_{\bar{c}} - A_{\bar{c}}\end{matrix}}_{n-r} \end{array}\right]\begin{matrix}\}r \\ \}n-r\end{matrix}, \quad Q^{\mathrm{T}}B = \left[\begin{matrix}B_c \\ \hline 0\end{matrix}\right]\begin{matrix}\}r \\ \}n-r\end{matrix} \tag{9}$$

$$CZ = [\underbrace{C_c}_{r} \mid \underbrace{C_{\bar{c}}}_{n-r}]$$

The reduced order system $\{\lambda E_c - A_c, B_c, C_c\}$ is controllable and has the same TFMM as $\{\lambda E - A, B, C\}$. The eigenvalues of the regular pencils $\lambda E_c - A_c$ and $\lambda E_{\bar{c}} - A_{\bar{c}}$ are called, respectively, the *controllable* and *uncontrollable poles* of the system.

Analogously, let q be the dimension of the unobservable subspace $\bar{\mathscr{O}}$ and let Z and Q be orthogonal transformation matrices whose last q columns span $\bar{\mathscr{O}}$ and $E\bar{\mathscr{O}} + A\bar{\mathscr{O}}$, respectively. Then the system $\{\lambda E - A, B, C\}$ can be transformed to

$$Q^{\mathrm{T}}(\lambda E - A)Z = \left[\begin{array}{c|c} \underbrace{\begin{matrix}\lambda E_o - A_o \\ \hline *\end{matrix}}_{n-q} & \underbrace{\begin{matrix}0 \\ \hline \lambda E_{\bar{o}} - A_{\bar{o}}\end{matrix}}_{q} \end{array}\right]\begin{matrix}\}n-q \\ \}q\end{matrix}, \quad Q^{\mathrm{T}}B = \left[\begin{matrix}B_o \\ \hline B_{\bar{o}}\end{matrix}\right]\begin{matrix}\}n-q \\ \}q\end{matrix} \tag{10}$$

$$CZ = [\underbrace{C_o}_{n-q} \mid \underbrace{0}_{q}]$$

where $\{\lambda E_o - A_o, B_o, C_o\}$ is observable, having the same TFMM as $\{\lambda E - A, B, C\}$.

As can be observed immediately from the form of matrices in (9) and (10), if we have a procedure for computing the controllability form (9) of the system, then the same procedure can be used to determine also the observability form (10) applying it to the dual GSSM $\{\lambda E^T - A^T, C^T, B^T\}$. Therefore, the computation of an irreducible GSSM from a non-minimal one can be performed in two steps: first, determine the controllable part of the system, $\{\lambda E_c - A_c, B_c, C_c\}$ and then determine the observable part $\{\lambda E_{co} - A_{co}, B_{co}, C_{co}\}$ of the resulted controllable part. This system is both controllable and observable, and therefore it is irreducible. It has the same TFMM as the initial GSSM.

2.1 The reduction algorithm

The reduction of the initial system (1) to the form (9) can be accomplished in a numerically stable way using the pencil algorithm of Van Dooren [1]. However, this algorithm, applied as it is stated, is computationally expensive. For example, for a controllable single-input system with E non-singular, the algorithm uses $O(n^4)$ floating-point operations (*flops*). (One flop is roughly equivalent to compute $a + b \times c$, where a, b, c are floating-point numbers.)

Paige [2] outlined a numerically stable algorithm, applicable when E is non-singular, which reduces the system (1) to form (9). This algorithm as well as its modification proposed by Chu [3], performs also $O(n^4)$ flops, the former being a variant of the pencil algorithm for invertible E.

In this section we propose a new numerically stable procedure for the computation of the controllability form (9), which requires only $O(n^3)$ flops. The procedure is applicable regardless E is singular or not. The procedure is based on the following algorithm:

Algorithm 1.

1. Reduce E to *upper-triangular* (U−T) form by using a suitable orthogonal transformation matrix Z_0

$$E \leftarrow EZ_0\,, \quad A \leftarrow AZ_0\,, \quad C \leftarrow CZ_0\,.$$

2. Set $j = 1$, $r = 0$, $n_0 = m$; $E_0 = E$, $A_0 = A$, $B_0 = B$, $Q = I_n$, $Z = Z_0$.
3. Determine the orthogonal transformation matrices Q_j and Z_j to compress the $(n - r) \times n_{j-1}$ matrix B_{j-1} to full row rank while keeping the U-T form of E_{j-1}; perform the transformations and partition the matrices $Q_j^T B_{j-1}$, $Q_j^T E_{j-1} Z_j$ and $Q_j^T A_{j-1} Z_j$ analogously:

$$Q_j^T B_{j-1} \triangleq \begin{bmatrix} A_{j,j-1} \\ \hline 0 \end{bmatrix} \begin{matrix} \}n_j \\ \}\varrho_j \end{matrix}; \quad Q_j^T E_{j-1} Z_j \triangleq \underbrace{\begin{bmatrix} E_{j,j} & E_{j,j+1} \\ \hline 0 & E_j \end{bmatrix}}_{n_j \quad \varrho_j} \begin{matrix} \}n_j \\ \}\varrho_j \end{matrix}$$

$$Q_j^T A_{j-1} Z_j \triangleq \underbrace{\begin{bmatrix} A_{j,j} & A_{j,j+1} \\ \hline B_j & A_j \end{bmatrix}}_{n_j \quad \varrho_j} \begin{matrix} \}n_j \\ \}\varrho_j \end{matrix}$$

4. For $i = 1, \ldots, j-1$ transform and partition $A_{i,j}Z_j$ and $E_{i,j}Z_j$ analogously:

$$A_{i,j}Z_j \triangleq [A_{i,j} \mid A_{i,j+1}]\,; \quad E_{i,j}Z_j \triangleq [E_{i,j} \mid E_{i,j+1}]$$

5. Accumulate the transformations and transform C

$$Q \leftarrow Q\left[\begin{array}{c|c} I_r & 0 \\ \hline 0 & Q_j \end{array}\right]; \quad Z \leftarrow Z\left[\begin{array}{c|c} I_r & 0 \\ \hline 0 & Z_j \end{array}\right]; \quad C \leftarrow C\left[\begin{array}{c|c} I_r & 0 \\ \hline 0 & Z_j \end{array}\right].$$

Comment. Exit if the system has controllable finite poles.

6. $r \leftarrow r + n_j$; If $\varrho_j = 0$, then $k = j$ and *stop*; else, go to 7.

Comment. Exit if the system is uncontrollable.

7. If $n_j = 0$, then $k = j - 1$ and *stop*; else, $j \leftarrow j + 1$ and go to 3.

At the end of this algorithm, we obtain the system matrices in the form (9), where

$$E_c = \begin{bmatrix} E_{11} & E_{12} & \cdot & \cdot \\ 0 & E_{22} & \cdot & \cdot \\ \cdots & \cdots & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot \\ 0 & 0 & 0 & E_{kk} \end{bmatrix} \quad A_c = \begin{bmatrix} A_{11} & A_{12} & \cdot & \cdot \\ A_{21} & A_{22} & \cdot & \cdot \\ \cdots & \cdots & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot \\ 0 & 0 & A_{k,k-1} & A_{k,k} \end{bmatrix} \quad B_c = \begin{bmatrix} A_{10} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \tag{11}$$

In (11), $E_{ii}, A_{ii} \in \mathbb{R}^{n_i \times n_i}$, $i = 1, \ldots, k$; $A_{i,i-1} \in \mathbb{R}^{n_i \times n_{i-1}}$, $i = 1, \ldots, k$ have ranks n_i, E_c is U-T and A_c is in an *upper block-Hessenberg* (U-BH) form. For single-input systems, A_c is obtained in an *upper-Hessenberg* (U-H) form.

The reduced system $\{\lambda E_c - A_c, B_c, C_c\}$ has no *finite* uncontrollable poles. This can be verified easily by inspecting the forms of matrices E_c, A_c and B_c and observing that

$$\text{rank}\,[\lambda E_c - A_c, B_c] = r \text{ for all finite } \lambda \in \mathbb{C}$$

[4]. Therefore, the uncontrollable part

$$\lambda E_{\bar{c}} - A_{\bar{c}} \triangleq \lambda E_k - A_k$$

contains *all* uncontrollable finite poles of the system. However, the pair $(\lambda E_c - A_c, B_c)$ resulted from Algorithm 1 may have uncontrollable infinite poles.

In order to remove the uncotrollable part of the system corresponding to the infinite poles, we apply the *same* algorithm to the system $\{\lambda A - E, B, C\}$. This is equivalent to replacing λ by $1/\lambda$ and it is the basic technique for the study of the infinite pole structure of GSSM. We shall use the label "∞" for the resulted matrices in (9). As can be seen again from the form of the resulted matrices E_c^∞, A_c^∞ and B_c^∞ (E_c^∞ is U-BH, A_c^∞ is U-T), we have

$$\text{rank}\,[E_c^\infty - \lambda A_c^\infty, B_c^\infty] = r \text{ for all finite } \lambda \in \mathbb{C}$$

In particular, for $\lambda = 0$, we have

$$\text{rank}\,[E_c^\infty, B_c^\infty] = r\,,$$

that is, according to Theorem 2 of Cobb [4], the system $\{\lambda E_c^\infty - A_c^\infty, B_c^\infty, C_c^\infty\}$ has

no *infinite* uncontrollable poles and the only uncontrollable finite poles can be those in origine. Thus, the uncontrollable part $\lambda E_{\bar{c}}^{\infty} - A_{\bar{c}}^{\infty}$ contains *all* uncontrollable infinite poles of the system as well as the finite uncontrollable poles excepting those which are zero.

For reducing a given system $\{\lambda E - A, B, C\}$ to the form (9), in which $\lambda E_c - A_c$ has no uncontrollable (finite or infinite) poles, the following procedure can be used:

Controllability form procedure (CFP)

Comment. Separate uncontrollable infinite poles.

1. Perform Algorithm 1 on $\{\lambda A - E, B, C\}$, accumulate the transformations in Q_1 and Z_1 and partition the transformed matrices as follows:

$$Q_1^{\mathrm{T}}(\lambda E - A) Z_1 = \left[\begin{array}{c|c} \lambda E_c^{\infty} - A_c^{\infty} & * \\ \hline 0 & \lambda E_{\bar{c}}^{\infty} - A_{\bar{c}}^{\infty} \end{array}\right], \quad Q_1^{\mathrm{T}} B = \left[\begin{array}{c} B_c^{\infty} \\ \hline 0 \end{array}\right]$$

$$CZ_1 = [C_c^{\infty} \mid C_{\bar{c}}^{\infty}]$$

Comment. Separate uncontrollable finite (zero) poles.

2. Perform Algorithm 1 on $\{\lambda E_c^{\infty} - A_c^{\infty}, B_c^{\infty}, C_c^{\infty}\}$, accumulate transformations in Q_2 and Z_2 and partition the transformed matrices as follows:

$$Q_2^{\mathrm{T}}(\lambda E_c^{\infty} - A_c^{\infty}) Z_2 = \left[\begin{array}{c|c} \lambda E_c - A_c & * \\ \hline 0 & \lambda E_{\bar{c}}^{f} - A_{\bar{c}}^{f} \end{array}\right], \quad Q_2^{\mathrm{T}} B_c^{\infty} = \left[\begin{array}{c} B_c \\ \hline 0 \end{array}\right]$$

$$C_c^{\infty} Z_2 = [C_c \mid C_{\bar{c}}^{f}]$$

3. Compute the transformation matrices

$$Q = Q_1 \begin{bmatrix} Q_2 & 0 \\ 0 & I \end{bmatrix}, \quad Z_1 = Z_1 \begin{bmatrix} Z_2 & 0 \\ 0 & I \end{bmatrix}$$

At the end of this algorithm we obtain

$$Q^{\mathrm{T}}(\lambda E - A) Z = \left[\begin{array}{c|cc} \lambda E_c - A_c & * & * \\ \hline 0 & \lambda E_{\bar{c}}^{f} - A_{\bar{c}}^{f} & * \\ 0 & 0 & \lambda E_{\bar{c}}^{\infty} - A_{\bar{c}}^{\infty} \end{array}\right], \quad Q^{\mathrm{T}} B = \begin{bmatrix} B_c \\ 0 \\ 0 \end{bmatrix}$$

$$CZ = [C_c \mid C_{\bar{c}}^{f} \; C_{\bar{c}}^{\infty}]$$

which is in the controllability form (9). We note that the resulted GSSM matrices E_c, A_c and B_c have the forms (11). For the computation of the observability form (10), the CFP should be applied to the dual GSSM $\{\lambda E^{\mathrm{T}} - A^{\mathrm{T}}, C^{\mathrm{T}}, B^{\mathrm{T}}\}$.

The computation of an irreducible GSSM from a non-minimal GSSM $\{\lambda E - A, B, C\}$ can performed by removing successively the uncontrollable and then the unobservable parts of the system using the CFP. However, in the following procedure we use for convenience Algorithm 1. All computations will be performed without accumulating the transformations.

Irreducible realization procedure (IRP)

1. Perform Algorithm 1 on the GSSM $\{\lambda A - E, B, C\}$ and determine the reduced GSSM $\{\lambda E_c^\infty - A_c^\infty, B_c^\infty, C_c^\infty\}$.
2. Perform Algorithm 1 on the GSSM $\{\lambda E_c^\infty - A_c^\infty, B_c^\infty, C_c^\infty\}$ and determine the controllable GSSM $\{\lambda E_c - A_c, B_c, C_c\}$.
3. $E_c \leftarrow PE_cP$, $A_c \leftarrow PA_cP$, $B_c \leftarrow PB_c$, $C_c \leftarrow C_cP$, where

$$P = \begin{bmatrix} & & & 1 \\ & 0 & \cdot & \cdot \\ & & 1 & 0 \\ 1 & & & \end{bmatrix} \tag{12}$$

4. Perform Algorithm 1 on the dual GSSM $\{\lambda A_c^T - E_c^T, C_c^T, B_c^T\}$ and determine the reduced GSSM $\{\lambda E_{co}^\infty - A_{co}^\infty, B_{co}^\infty, C_{co}^\infty\}$.
5. Perform Algorithm 1 on the dual GSSM $\{\lambda(E_{co}^\infty)^T - (A_{co}^\infty)^T, (C_{co}^\infty)^T, (B_{co}^\infty)^T\}$ and determine the irreducible GSSM $\{\lambda E_{co} - A_{co}, B_{co}, C_{co}\}$.

Remarks.

(a) The reason for permuting the rows and columns of the system matrices at Step 3 using the permutation transformation matrix (12) is to obtain the dual pair (PE_c^TP, PA_c^TP) in (U-T, U-BH) form. As we shall see in the following subsection, Algorithm 1 can be implemented in such way that it can exploit efficiently the null elements structure of E matrix.

(b) The CFP and the IRP can gain in efficiency if the given system $\{\lambda E - A, B, C\}$ has some particular features. For example, if the matrix E is non-singular, that is the corresponding transfer matrix (4) is a strictly proper rational matrix, only Step 2 of CFP or Steps 2, 3 and 5 of IRP must be performed. If the matrix E is nilpotent, that is the corresponding transfer matrix is a polynomial matrix or if the matrix A is non-singular, that is the system has no poles in the origine, then only Step 1 of CFP or Steps 1, 3 and 4 of IRP must be performed. Furthermore, if the system is controllable (observable) only Steps 4 and 5 (1 and 2) of IRP should be performed.

2.2 Algorithmic details

In order to increase the numerical accuracy and speed of the IRP, Algorithm 1 should be implemented with special care so that the number of flops required for performing Steps 1 and 3–5 to be minimized. We recommend to use for annihilating elements of vectors, the class of symmetric orthogonal (Householder) matrices $\mathscr{H}_k(i, j)$, $(k < i \leqq j$ or $i \leqq j < k)$ of the form $I + vu^T$, where u and v are vectors, $v^Tu = -2$, v is a scalar multiple of u, only components $i, i+1, \ldots, j$ and k of u are nonzero and $u_k = 1$. Given a vector x, it is easy to choose a member $Q \in \mathscr{H}_k(i, j)$

so that $Qx = x + (u^T x)\, v$ has its $i, i+1, \ldots, j$ components equal to zero, its kth component changed and all other components unchanged [5]. Since, $u_k = 1$, the computation of Qy for any y requires $2(j - i + 1) + 1$ flops. If $Q \in \mathscr{H}_k(i, i)$, the multiplication Qy requires 3 flops. We note that standard plane rotations (Givens) require 4 flops instead.

The reduction of E to U-T form at Step 1 of Algorithm 1 can be performed as follows:

Step 1.

1.1. Set $Z_0 = I$.

1.2. For $i = n, n-1, \ldots, 2$

1.2.1. Find $k = \min \{j \mid e_{ij} \neq 0, j = 1, 2, \ldots, i\}$.

1.2.2. If $k = i$, next i.

1.2.3. Choose $Z \in \mathscr{H}_i(k, i-1)$ to annihilate $e_{i,k}, e_{i,k+1}, \ldots, e_{i,i-1}$.

1.2.4. $E \leftarrow EZ$, $A \leftarrow AZ$, $C \leftarrow CZ$, $Z_0 \leftarrow Z_0 Z$.

This algorithm performs an RQ decomposition of the matrix E, without row pivoting. It exploits efficiently any particular form of the matrix E (U-T, U-H or U-BH). For this reason, if Algorithm 1 is performed repeteadly, as for example in the IRP, the number of operations performed at second and subsequent executions is usually negligible.

We consider now the efficient implementation of Step 3 of Algorithm 1. This step performs the compression of the $(n - r) \times n_{j-1}$ matrix B_{j-1} to a full row rank matrix, while keeping simultaneously the U-T form of E_{j-1} unaltered. For standard state-space systems with $E = I$, at each step $Z_j = Q_j$ and therefore $E_{j-1} = I_{n-r}$. In this case, row compressions can be computed using either the QR or the singular value decompositions of B_{j-1} [5]. Algorithms based on these techniques have been proposed in [1], [6]. For a general (possibly singular) E matrix, we propose a modified QR-type decomposition based on the use of the elementary orthogonal transformations from $\mathscr{H}_i(j, j)$. Row transformations are used to annihilate single elements in the columns of B_{j-1}. Each row transformation is followed by a column transformation which annihilates the non-zero element generated by the previous transformation under the diagonal of E_{j-1}. For $n = 5$, the reduction can be described diagrammatically as

$$B_{j-1}: \quad \begin{matrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & {\scriptstyle 3}\uparrow & \\ & & \times & \times \\ & & {\scriptstyle 1}\uparrow & \\ & & \times & \times \end{matrix} \qquad E_{j-1}: \quad \begin{matrix} \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & & \\ & & x_3 \underset{4}{\rightarrow} & \times & \times \\ & & & & \\ & & & x_1 \underset{2}{\rightarrow} & \times \end{matrix}$$

Here the row transformations 1 and 3 are applied to B_{j-1} and E_{j-1} to eliminate elements in B_{j-1} under the diagonal and produce non-zeros x_1 and x_3 in E_{j-1}. These are eliminated by column transformations 2 and 4 applied to E_{j-1}. The technique is similar to that proposed by Miminis and Paige [7] for single-input GSSM. In the algorithm given below, we also included a column pivoting strategy in order to make the rank decisions more robust. For ease of notation we define $G \triangleq B_{j-1}$ and $F \triangleq E_{j-1}$.

Step 3.

3.1. Set $Q_j = Z_j = I_{n-r}$, $n_j = 0$, $\varrho_j = n - r$, $i_{\max} = \min(n - r, n_{j-1})$.

3.2. For $i = 1, 2, \ldots, i_{\max}$

3.2.1. Compute $s_t = \sum_{k=i}^{n-r} |g_{kt}|$, for $t = i, i+1, \ldots, n_{j-1}$.

3.2.2. Find q such that $s_q = \max\{s_t \mid t = i, i+1, \ldots, n_{j-1}\}$.

3.2.3. If $s_q \leqq$ TOL, then go to 4; else, if $q \neq i$, permute columns q and i of G.

3.2.4. $n_j \leftarrow n_j + 1$, $\varrho_j \leftarrow \varrho_j - 1$; If $\varrho_j = 0$, then go to 4; else, go to 3.2.5.

3.2.5. For $k = n - r, n - r - 1, \ldots, i + 1$

3.2.5.1. Choose $Q^{\mathrm{T}} \in \mathscr{H}_{k-1}(k, k)$ to annihilate $g_{k,i}$.

3.2.5.2. $G \leftarrow Q^{\mathrm{T}}G$, $F \leftarrow Q^{\mathrm{T}}F$, $Q_j \leftarrow Q_jQ$.

3.2.5.3. Choose $Z \in \mathscr{H}_k(k-1, k-1)$ to annihilate $f_{k,k-1}$.

3.2.5.4. $F \leftarrow FZ$, $Z_j \leftarrow Z_jZ$.

Algorithm 1 has been implemented in double precision as a FORTRAN 77 subroutine TGSCO. The implementation of Steps 3–5 has been made such that all elementary operations at Step 3.2.5 are applied to all system matrices E, A, B, C as well as on the transformation matrices. Moreover, an option in TGSCO allows to apply the transformations performed on submatrices of the model matrices to the whole matrices. This option is useful when TGSCO is used to implement Step 2 of the CFP, where transformations are applied to the whole system matrices and not only on E_c^{∞}, etc. on which actually one works. An efficient version of Algorithm 1 for single-input system has been implemented as a separate subroutine TGSCO1 in order to be used in the evaluation of TFMM of GSSM [8] (see Section 3).

The IRP has been implemented in subroutine GSRMIN. This subroutine offers several useful options for increasing the efficiency and the accuracy of the results. Shorter algorithmic paths are provided for performing the algorithm on strictly proper systems, on systems having no poles in the origine, on controllable or on observable systems.

2.3 Properties of algorithms

Operation counts

The reduction of a full E matrix to the U-T form at Step 1 of Algorithm 1 requires $(\frac{5}{3})n^3 + pn^2$ flops. Additional n^3 flops are necessary when the transformations

are accumulated. It was shown in Subsection 2.2 that at the second and subsequent executions of the algorithm (as for example in the CFP or in the IRP), the null elements structure of E is exploited at Step 1 for improving the efficiency. If we assume that E has q subdiagonals, the reduction of E in this case requires at most

$$M(n, q) = q^2(n + 2q/3) + q(n - q)(2n + q) + qp(2n - q) \tag{13}$$

flops and at most $nq(2n - q)$ additional flops for the accumulation of transformations. We note that if E is U-T, then $q = 0$ and if E is U-BH then $q = \min\{2m - 1, n - 1\}$. If $q \ll n$, then the second and subsequent executions require only $O(n^2)$ flops to be performed.

In evaluating the number of flops required at Steps 3–7 of Algorithm 1 we took into account that the reduction of B_{j-1} is performed using only transformations of type $\mathscr{H}_i(j, j)$. Therefore, each multiplication Qy, where $Q \in \mathscr{H}_i(j, j)$ and y is a vector, requires 3 flops. Let r be the order of the reduced system computed by Algorithm 1. The reduction, performed by repeated execution of Steps 3 and 4, requires approximately $N(n, r)$ flops, where

$$N(n, r) = 4r^3 + 3r(n - r)(3n - r) + \tfrac{3}{2}(m + p)\, r(2n - r) \tag{14}$$

and additional $3nr(2n - r)$ flops for the accumulation of transformations by executing Step 5 repeatedly. In the worst case, when the system is controllable, Algorithm 1 requires about $\frac{17}{3}n^3 + \frac{3}{2}(m + 3p)$ flops for the reduction of matrices and additional $4n^3$ flops for the evaluation of the transformation matrices.

The CFP requires for the reduction of system matrices to the controllability form (9)

$$\tfrac{5}{3}n^3 + M(n, q) + N(n, r_1) + N(n, r_2) \quad \text{flops},$$

where r_1 and r_2, $r_1 \geqq r_2$, are the orders of reduced systems computed at Steps 1 and 2, respectively. The accumulation of transformations can be performed using about $n^3 + nq(2n - q) + 3n[r_1(2n - r_1) + r_2(2n - r_2)]$ flops. In the worst case ($r_1 = r_2 = n$), the CFP requires about $\frac{29}{3}n^3 + 3(n + 3p)\, n^2 + M(n, q)$ flops for the reduction and $7n^3 + nq(2n - q)$ flops for the accumulation of the transformations.

If r_1, r_2, r_3 and r_4 are the orders of the reduced systems determined by the IRP at Steps 1, 2, 4 and 5, respectively, then the total number of flops required is

$$N = \tfrac{5}{3}n^3 + \sum_{i=1}^{3} M(r_i, q) + \sum_{i=1}^{4} N(r_{i-1}, r_i)$$

where $r_0 = n$, and $M(\cdot, \cdot)$ and $N(\cdot, \cdot)$ are given by (13) and (14), respectively. In the worst case of an already irreducible system, that is $r_1 = r_2 = r_3 = r_4 = n$, the number of required flops is less than

$$N_{\max} = 18n^3 + (6m + 7p)\, n^2 + 3M(n, q).$$

In the best case we have $r_1 = r_2 = r_3 = r_4 = r$ and the number of required flops

is about

$$N_{\min} = \tfrac{5}{3}n^3 + pn^2 + 3M(r, q) + N(n, r) + 3N(r, r) =$$

$$= \tfrac{5}{3}n^3 + pn^2 + 3M(r, q) + 16r^3 + 3(m + p)\, r(n + r) + 3r(n - r)\,(3n - r)$$

In practice, as we shall see from the examples presented in Section 4, the number of required operations is usually half or even the quarter of the above evaluations. This happens because frequently only one or two computational steps of the IRP are necessary to be performed.

Storage requirement

All computations involved in Algorithm 1 as well as in the CFP can be performed in the same place, thus $2n^2 + (m + p)\, n$ storage locations are only necessary to hold the system matrices. Additional $2n^2$ storage locations are needed for storing the transformation matrices if they are accumulated. For the IRP, the computations can be performed in $2n^2 + 2 \max(m, p)\, n$ storage locations.

Numerical stability

Numerical stability of Algorithm 1 and therefore also of procedures CFP and IRP can be easily proved because of the use of orthogonal transformations. For details we refer to [1]. It can be shown that the resulted systems computed by Algorithm 1 as well as by the CFP or IRP, are exact for a slightly perturbed GSSM $\{\lambda\bar{E} - \bar{A}, \bar{B}, \bar{C}\}$ with

$$\|\bar{X} - X\| \leqq \varepsilon_X \|X\|\,, \quad X = E, A, B, C$$

where, in each case, ε_X is a small multiple of the machine relative precision.

3. APPLICATIONS

In this section we present shortly the main applications of the IRP.

3.1 Computation of minimal order GSSM of DSSM and TFMM

For a given DSSM of the form (2) or a TFMM of the form (3), a standard numerically reliable method to obtain a minimal order GSSM having the same TFMM, has the following two steps. First, construct a non-minimal "ad-hoc" generalized state-space realization and then, at second step, reduce it to a minimal one using the IRP. We give below two straightforward methods for constructing non-minimal GSSM for DSSM and TFMM.

For the differential model (2), let d be the highest power of λ occurring in the polynomial matrices $T(\lambda)$, $U(\lambda)$, $V(\lambda)$ and $W(\lambda)$, and let T_i, U_i, V_i, W_i be the coeffi-

cients matrices of λ^i. Let us define

$$A_0 = \begin{bmatrix} T_0 & 0 \\ \underbrace{-V_0}_{q} & \underbrace{-I_p}_{p} \end{bmatrix} \begin{matrix} \} q \\ \} p \end{matrix}; \quad A_i = \begin{bmatrix} T_i & 0 \\ \underbrace{-V_i}_{q} & \underbrace{0}_{p} \end{bmatrix} \begin{matrix} \} q \\ \} p \end{matrix}, \quad i = 1, \ldots, d \tag{15}$$

$$C_0 = [\underbrace{0}_{q} \ \underbrace{-I_p}_{p}] \} p; \qquad B_i = \begin{bmatrix} U_i \\ \underbrace{W_i}_{m} \end{bmatrix} \begin{matrix} \} p \\ \} q \end{matrix}, \quad i = 0, 1, \ldots, d$$

Then a GSSM $\{\lambda E - A, B, C\}$ of the DSSM $\{T(\lambda), U(\lambda), V(\lambda), W(\lambda)\}$ is given by the matrices [1]

$$\lambda E - A = \begin{bmatrix} -I & & & A_d \\ \lambda I & \ddots & & \vdots \\ & \ddots & -I & A_1 \\ & & \lambda I & A_0 \end{bmatrix} \quad B = \begin{bmatrix} B_d \\ \vdots \\ B_1 \\ B_0 \end{bmatrix} \tag{16}$$

$$C = [0 \ \ldots \ 0 \ \ C_0]$$

The order of this realization is $n = (d + 1)(q + p)$. If $m < p$, a lower order realization results by constructing the dual GSSM corresponding to the dual DSSM $\{T^T(\lambda), V^T(\lambda), U^T(\lambda), W^T(\lambda)\}$. The order of this realization is $n = (d + 1)(q + m)$. For particular differential models, other, usually smaller order, state-space realizations can be constructed.

For the TFMM (3) we suggest a method based on the following separation of $G(\lambda)$

$$G(\lambda) = H(\lambda) + D(\lambda) \tag{17}$$

where $H(\lambda)$ is the strictly proper part of $G(\lambda)$ and $D(\lambda)$ is its polynomial part. This separation can be easily computed using the standard polynomial division algorithm.

For $H(\lambda)$ we determine a minimal order standard state-space model $\{\lambda I - A_1, B_1, C_1\}$ such that

$$H(\lambda) = C_1(\lambda I - A_1)^{-1} B_1 .$$

A common approach for this problem is to construct a non-minimal state-space realization and then to remove its uncontrollable and unobservable parts. Methods for constructing non-minimal state-space realization of transfer matrices are given for example in [9]. The order reduction of the non-minimal state-space model can be performed using numerically stable algorithms as, for example, those proposed by Van Dooren [1] or Varga [6].

For the second term of (17), let s be the highest power of λ occurring in $D(\lambda)$ and, let D_i be the coefficient matrix of λ^i. Then, an observable GSSM for $D(\lambda)$ is $\{\lambda \bar{E}_2 - \bar{A}_2, \bar{B}_2, \bar{C}_2\}$, where

$$\lambda \bar{E}_2 - \bar{A}_2 = \begin{bmatrix} -I & & 0 \\ \lambda I & \ddots & \\ & \ddots & -I \\ 0 & & \lambda I \ -I \end{bmatrix}, \quad \bar{B}_2 = \begin{bmatrix} D_s \\ \vdots \\ D_1 \\ D_0 \end{bmatrix} \tag{18}$$

$$\bar{C}_2 = [0 \ \ldots \ 0 \ -I]$$

This GSSM is generally non-minimal (uncontrollable). For removing its uncontrollable part, we need to perform only the first step of the IRP. Let $\{\lambda E_2 - A_2, B_2, C_2\}$ the resulted irreducible GSSM. Then, an irreducible GSSM for $G(\lambda)$ is given by the GSSM $\{\lambda E - A, B, C\}$, where

$$\lambda E - A = \begin{bmatrix} \lambda I - A_1 & 0 \\ 0 & \lambda E_2 - A_2 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \quad C = [C_1 \; C_2] .$$

3.2 Computation of minimal order inverses

Consider a "square" GSSM of the form

$$\lambda E \, x(t) = A \, x(t) + B \, u(t)$$
$$y(t) = C \, x(t) + D \, u(t)$$

with $p = m$. Its transfer function matrix is given by

$$G(\lambda) = C(\lambda E - A)^{-1} B + D .$$

The inverse system, having $G^{-1}(\lambda)$ as transfer function matrix, is given by the system $\{\lambda \tilde{E} - \tilde{A}, \tilde{B}, \tilde{C}\}$ [9], where

$$\lambda \tilde{E} - \tilde{A} = \begin{bmatrix} \lambda E - A & B \\ -C & D \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} 0 \\ I_m \end{bmatrix}, \quad \tilde{C} = [0 \; I_m] . \tag{19}$$

The extraction of the minimal order inverse system can be done using the IRP.

3.3 Computation of the TFMM of GSSM

The transfer function matrix $G(\lambda)$ of a given GSSM $\{\lambda E - A, B, C\}$ can be computed by the following method [8]. An element $g_{ij}(\lambda)$ of $G(\lambda)$ is computed by evaluating

$$g_{ij}(\lambda) = c_i(\lambda E - A)^{-1} b_j ,$$

where c_i and b_j are the ith row of C and the jth column of B, respectively. The transfer function $g_{ij}(\lambda)$ is determined in the form

$$\frac{k \prod_i (\lambda - \mu_i)}{\prod_j (\lambda - \varrho_j)}$$

where k is the gain, ϱ_j are the poles and μ_i are the zeros of an irreducible GSSM of the system $\{\lambda E - A, b_j, c_i\}$. For the computation of such an irreducible GSSM we used in [9] the IRP with Algorithm 1 specialized to single-input systems. The poles and zeros computations are performed via the QZ algorithm [10].

4. EXAMPLES

We present several numerical examples to illustrate the performance of the IRP. The computations were carried out on an IBM PC/AT with floating point co-processor. All computations were performed in double precision. The computation of poles has been performed using the QZ method [10]. For the computation of zeros, the algorithm proposed in [11] was used.

Example 1.

This example shows the applicability of the IRP for determining a minimal order GSSM of a DSSM. Consider the DSSM with

$$T(\lambda) = \begin{bmatrix} \lambda+1 & 0 \\ 0 & \lambda+1 \end{bmatrix}; \quad U(\lambda) = \begin{bmatrix} -\lambda+1 & \lambda-2 \\ -2 & 3 \end{bmatrix}; \quad V = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}; \quad W = \begin{bmatrix} 0 & 0 \\ 3 & -3 \end{bmatrix}$$

A 8th order GSSM $\{\lambda E - A, B, C\}$ is constructed using (15) and (16). The resulted system matrices are:

$$E = \begin{bmatrix} 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \end{bmatrix} \quad A = \begin{bmatrix} 1&0&0&0&-1&0&0&0 \\ 0&1&0&0&0&-1&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&-1&0&0&0 \\ 0&0&0&0&0&-1&0&0 \\ 0&0&0&0&3&0&1&0 \\ 0&0&0&0&0&2&0&1 \end{bmatrix} \quad B = \begin{bmatrix} -1&1 \\ 0&0 \\ 0&0 \\ 0&0 \\ 1&-2 \\ -2&3 \\ 0&0 \\ 3&-3 \end{bmatrix}$$

$$C = \begin{bmatrix} 0&0&0&0&0&0&-1&0 \\ 0&0&0&0&0&0&0&-1 \end{bmatrix}.$$

The original system has a pole at $\varrho = -1$ and has no zeros.

The non-minimal system has no poles in the origine, so that, for the computation of the irreducible GSSM given below, only Steps 1, 3 and 4 of the IRP were performed. The resulted system matrices given with 7 exact decimal digits are:

$$E_{co} = \begin{bmatrix} \cdot 2603253 & -\cdot 6392563 \\ \cdot 1136153 & -\cdot 2789943 \end{bmatrix} \quad A_{co} = \begin{bmatrix} \cdot 4816018 & 0 \\ -1\cdot 1389931 & 1\cdot 1624764 \end{bmatrix}$$

$$B_{co} = \begin{bmatrix} \cdot 6480740 & -1\cdot 9442221 \\ 3\cdot 8183766 & -4\cdot 3840620 \end{bmatrix} \quad C_{co} = \begin{bmatrix} -\cdot 5422824 & -\cdot 4931969 \\ \cdot 7431277 & 0 \end{bmatrix}$$

The computed finite pole of the resulted minimal order system is

$$\bar{\varrho} = -\cdot 9999999999999984$$

Example 2.

This example shows the usage of the IRP to compute an irreducible GSSM for a TFMM having only polynomial part. Let us consider the transfer-function matrix

$$D(\lambda) = D_0 + D_1\lambda + D_2\lambda^2$$

where

$$D_0 = \begin{bmatrix} 1 & 2 & -2 \\ 0 & -1 & -2 \\ 0 & 0 & 0 \end{bmatrix} \quad D_1 = \begin{bmatrix} 1 & 3 & 0 \\ 1 & 4 & 2 \\ 0 & -1 & -2 \end{bmatrix} \quad D_2 = \begin{bmatrix} 1 & 4 & 2 \\ 0 & 0 & 0 \\ 1 & 4 & 2 \end{bmatrix}$$

This system has a finite zero at $\mu = 1$.

An observable 9th order realization of $D(\lambda)$ is given by (18), where

$$E = \begin{bmatrix} 0 & 0 & 0 \\ I & 0 & 0 \\ 0 & I & 0 \end{bmatrix} \quad A = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \quad B = \begin{bmatrix} D_2 \\ D_1 \\ D_0 \end{bmatrix} \quad C = [0\ 0\ -I]$$

Each submatrix of the above matrices has order 3.

An irreducible 4th order GSSM has been computed using Algorithm 1 on the system $\{\lambda A - E, B, C\}$. The resulted minimal order GSSM has the following matrices (given with 7 exact decimal digits):

$$E_{co} = \begin{bmatrix} -\cdot1587301 & -\cdot0591461 & -\cdot0815578 & \cdot0968908 \\ \cdot8436106 & \cdot0720390 & \cdot2615623 & -\cdot3107361 \\ \cdot3602503 & -\cdot5490416 & -\cdot2996273 & \cdot3559573 \\ \cdot2323367 & . \quad 0 & \cdot0579613 & -\cdot0688580 \end{bmatrix}$$

$$A_{co} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \quad B_{co} = \begin{bmatrix} -2\cdot1417986 & -7\cdot9372539 & -3\cdot0237158 \\ \cdot6424161 & 0 & -3\cdot8544964 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$C_{co} = \begin{bmatrix} -\cdot2519763 & -\cdot7165410 & \cdot4188871 & -\cdot4976185 \\ \cdot1259881 & -\cdot4200413 & -\cdot2665541 & \cdot3166664 \\ 0 & 0 & \cdot7650448 & \cdot6439769 \end{bmatrix}$$

The computed finite zero of the reduced system is accurate to full machine precision.

Example 3.

This example shows the applicability of the IRP for the computation of minimal order inverses. We consider the transfer matrix

$$G(s) = \frac{3}{s+1}\begin{bmatrix} -s+1 & s-2 \\ s-\frac{1}{3} & -s+1 \end{bmatrix}$$

This is the same system as in Example 1. It has a pole in $\varrho = -1$ and no zeros.

A non-minimal state-space realization of $G(s)$ is for example

$$sE - A = \begin{bmatrix} s+1 & 0 \\ 0 & s+1 \end{bmatrix}; \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; \quad C = \begin{bmatrix} 6 & -9 \\ -4 & 6 \end{bmatrix}; \quad D = \begin{bmatrix} -3 & 3 \\ 3 & -3 \end{bmatrix}$$

The inverse system is computed as the minimal order GSSM of the system having the matrices in (19). This system has no poles in the origine, so that only Steps 1, 3 and 4 were performed in the IRP. The resulted irreducible inverse system has the following matrices (up to 7 exact decimal digits):

$$\tilde{E}_{co} = \begin{bmatrix} 0 & 0 & \cdot 8498366 \\ 0 & 0 & \cdot 5214359 \\ 0 & 0 & \cdot 0766965 \end{bmatrix} \quad \tilde{B}_{co} = \begin{bmatrix} \cdot 2357022 & \cdot 4714045 \\ -\cdot 5061710 & -\cdot 6869464 \\ \cdot 8295994 & -\cdot 5530663 \end{bmatrix}$$

$$\tilde{A}_{co} = \begin{bmatrix} -\cdot 2357022 & 0 & 0 \\ \cdot 8315667 & -\cdot 9761870 & 0 \\ 4\cdot 1905409 & -4\cdot 2118127 & -13\cdot 0384048 \end{bmatrix} \quad \tilde{C}_{co} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

The computed zero of the inverse system is

$$\bar{\mu} = -1\cdot 000000000000003$$

Example 4.

This example illustrates the usage of the IRP in an algorithm for computing the transfer function of a GSSM [8]. Consider the 15th order GSSM $\{\lambda E - A, b, c\}$ with

$$E = \begin{bmatrix} I_{11} & 0 \\ 0 & E_2 \end{bmatrix}, \quad A = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \quad c = [c_1 \; c_2]$$

where

$$A_1 = \left[\begin{array}{cccc|cccc|cccc} 0 & 0 & 0 & -24 & & & & & & & & \\ 1 & 0 & 0 & -50 & & & & & & & & \\ 0 & 1 & 0 & -35 & & & \mathbf{0} & & & & \mathbf{0} & \\ 0 & 0 & 1 & -10 & & & & & & & & \\ \hline & & & & 0 & 0 & 0 & -30 & & & & \\ & & & & 1 & 0 & 0 & -61 & & & & \\ & & \mathbf{0} & & 0 & 1 & 0 & -41 & & & \mathbf{0} & \\ & & & & 0 & 0 & 1 & -11 & & & & \\ \hline & & & & & & & & 0 & 0 & -15 & \\ & & \mathbf{0} & & & & \mathbf{0} & & 1 & 0 & -23 & \\ & & & & & & & & 0 & 1 & -9 & \end{array}\right] \quad b_1 = \left[\begin{array}{c} 18 \\ 42 \\ 30 \\ 6 \\ \hline 10 \\ 17 \\ 8 \\ 1 \\ \hline 0 \\ -10 \\ -2 \end{array}\right]$$

$$c_1 = [0\,0\,0\,0\,0\,0\,0\,1\,0\,0\,0]$$

$$E_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad A_2 = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad b_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$c_2 = [0\ 0\ 0\ -1]$$

The system is both uncontrollable and unobservable, has a pole in the origine as well as infinite poles. The minimal order GSSM was computed by performing all steps in the IRP. The resulted minimal order GSSM has the following matrices

$$E_{co} = \begin{bmatrix} 0 & 0 \\ -\cdot 061038403342540 & -\cdot 998135418326288 \end{bmatrix}$$

$$A_{co} = \begin{bmatrix} \cdot 705788322840989 & -\cdot 043160668916314 \\ \cdot 183115210029681 & 2\cdot 994406254985376 \end{bmatrix}$$

$$b_{co} = \begin{bmatrix} \cdot 999999999999258 \\ 23\cdot 126041449126810 \end{bmatrix} \quad c_{co} = [-\cdot 708427702497790 \quad 0]$$

The transfer function corresponding to this GSSM is

$$g(\lambda) = \frac{\lambda + 4}{\lambda + 3}.$$

The computed values of the zero μ and pole ϱ, are

$$\bar{\mu} = -4\cdot 000000000011713\,, \quad \bar{\varrho} = -3\cdot 000000000006627$$

5. CONCLUSIONS

In this paper we have presented a numerically stable and computationally efficient algorithm for the computation of irreducible generalized state-space realizations. The algorithm uses exclusively orthogonal similarity transformations. The main application of the proposed algorithm is the reliable computation of minimal order GSSM of TFMM or DSSM. Examples have been given to illustrate the numerical performance of the algorithm. It has been successfully implemented on the computer.

(Received November 8, 1988.)

REFERENCES

[1] P. M. Van Dooren: The generalized eigenstructure problem in linear system theory. IEEE Trans. Automat. Control *AC-26* (1981), 111—129.

[2] C. C. Paige: Properties of numerical algorithms related to computing controllability. IEEE Trans. Autom. Control *AC-26* (1981), 130—138.

[3] K.-W. E. Chu: A controllability condensed form and a state feedback pole assignment algorithm for descriptor systems. IEEE Trans. Automat. Control *AC-33* (1988), 366—370.

[4] J. D. Cobb: Feedback and pole-placement in descriptor variable systems. Internat. J. Control *33* (1981), 1135—1146.

[5] G. W. Stewart: Introduction to Matrix Computations. Academic Press, New York 1973.

[6] A. Varga: Numerically stable algorithm for standard controllability form determination. Electron. Lett. *17* (1981), 74—75.
[7] G. S. Miminis and C. C. Paige: An algorithm for pole assignment of time invariant linear systems. Internat. J. Control *35* (1982), 341—354.
[8] A. Varga: Computation of Transfer Function Matrices of Generalized State-Space Models. Technical Report, RR-270.1/1988, Inst. Computers & Informatics, Bucharest.
[9] H. H. Rosenbrock: State-Space and Multivariable Theory. J. Wiley and Sons, New York 1970.
[10] C. B. Moler and G. W. Stewart: An algorithm for generalized matrix eigenvalue problems. SIAM J. Numer. Anal. *10* (1973), 241—256.
[11] A. Varga: Computation of Zeros of Descriptor Systems (report in preparation), 1988.

Dr. Andras Varga, Research Institute for Computers and Informatics, Bd. Miciurin, No. 8—10, 71316 Bucharest. Romania.