On Stabilization Methods of Descriptor Systems

A. Varga
DLR - Oberpfaffenhofen, German Aerospace Research Establishment, Institute for Robotics and System Dynamics
P.O.B. 1116, D-82230 Wessling, Germany
E-mail: df03@master.df.op.dlr.de

Abstract: Numerically reliable computational methods are proposed for the stabilization of a linear descriptor system with or without simultaneous elimination of its impulsive behavior. Two basic stabilization approaches are discussed. The first approach relies on methods which represent generalizations of direct stabilization techniques for standard state-space systems. The second approach is based on a recursive generalized Schur algorithm for pole assignment. Both approaches are based exclusively on numerically reliable procedures and can serve for robust software implementations.

Keywords: Descriptor systems; stabilization methods; pole assignment; numerical algorithms.

1. Introduction

Consider the descriptor system (E, A, B) described by the equation

$$E\lambda x(t) = Ax(t) + Bu(t) \tag{1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, E, $A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$, and $\lambda x(t) = \dot{x}(t)$ for a continuous-time system or $\lambda x(t) = x(t+1)$ for a discrete-time system. The system (1) is called *regular* if $\det(\lambda E - A) \not\equiv 0$. The set of poles of the system (1) (finite and infinite) is denoted with $\lambda(A, E)$ and represents the set of generalized eigenvalues of the pair (A, E). If the number of finite poles is less than $r = \text{rank}(E)$ then the system (1) has impulsive modes. The system (1) is called *impulse-free* if it has exactly r finite poles. Let $\mathbb{C}^-$ denote the stability region of the complex plane $\mathbb{C}$ and let $\mathbb{C}^+$ be its complement. We say that the system (1) is *stable* if all its finite poles are in $\mathbb{C}^-$. The common interpretation of $\mathbb{C}^-$ 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 following stabilization problems are considered:

1. *S-Stabilization Problem* (SSP): For the system (1) determine a state feedback matrix $F \in \mathbb{R}^{m,n}$ such that the closed-loop system $(E, A + BF, B)$ is regular, impulse-free and stable.
2. *R-Stabilization Problem* (RSP): For a *regular* system (1) determine a state feedback matrix $F \in \mathbb{R}^{m,n}$ such that the closed-loop system $(E, A + BF, B)$ is regular and stable.

The following theorems give necessary and sufficient conditions for the existence of solutions of the formulated stabilization problems.

Theorem 1. [7] *A solution F of the SSP exists if and only if the system (E, A, B) is strongly stabilizable, that is: 1)* $\operatorname{rank}([\,\lambda E - A\ B\,]) = n$ *for all finite* $\lambda \in \mathbb{C}^+$*; and 2)* $\operatorname{rank}([\,E\ AS_\infty\ B\,]) = n$*, where the columns of S_∞ span the null space of E.*

Theorem 2. [5] *A solution F of the RSP exists if and only if the system (E, A, B) is stabilizable, that is* $\operatorname{rank}([\,\lambda E - A\ B\,]) = n$ *for all finite* $\lambda \in \mathbb{C}^+$.

The solution of the stabilization problem has several applications. For example, the solution of the SSP can be used for the initialization of Newton's method to solve descriptor Riccati equations [7]. The S- or R-stabilization of a descriptor system can be also considered as simple synthesis methods to design controllers or observers for descriptor systems [5]. Further, the R-stabilization procedures can be easily reformulated to compute stable coprime factorizations of arbitrary rational matrices, while the S-stabilization procedure can produce coprime factorizations with proper factors for arbitrary rational matrices. In spite of these important applications, the solution of stabilization problems received little attention in the literature. Seemingly there are no descriptions of specially devised algorithms to solve the stabilization problems of descriptor systems, as for example, those existing for stabilizing of standard state-space systems (see [10] and the references therein).

In this paper we propose implementable computational techniques for solving the stabilization problems of linear descriptor system. A general procedure in two steps for solving the SSP is described in the next section. At the first step the impulsive behavior of the system is eliminated and the regularity is assured by a special choice of a preliminary state feedback matrix. At the second step a stabilizing state-feedback matrix is determined which moves the unstable poles of the modified system resulted after the first step in the stable region. This step solves basically a RSP and two numerical approaches for this purpose are further presented in the subsequent sections. The first approach is based on direct stabilization techniques which represent generalizations of the corresponding stabilization methods for standard continuous- and discrete-time state-space systems. The second approach is based on a generalized Schur algorithm for pole assignment. Both approaches are based exclusively on numerically reliable procedures and are thought to serve as bases for robust software implementations.

2. A procedure for S-stabilization

Provided the system (E, A, B) is strongly stabilizable, the solution of the SSP can be performed by using the following procedure in two steps:

1. For the system (E, A, B) determine a state feedback matrix F_1 such that the system $(E, A+BF_1, B)$ is regular and impulse free (that is it has exactly r finite poles).
2. For the system $(E, A + BF_1, B)$ determine a stabilizing state-feedback F_2 such that the system $(E, A + BF, B)$ with $F = F_1 + F_2$ has exactly r stable finite poles.

A detailed realization of these steps is the following implementable algorithm.

Algorithm SSDS. S-Stabilization of Descriptor Systems.

1. Find orthogonal matrices U_E and V_E such that $U_E E V_E = \begin{bmatrix} 0 & 0 \\ 0 & E_{22} \end{bmatrix}$, where $E_{22} \in \mathbb{R}^{r,r}$ is non-singular, and set
$$E \leftarrow U_E E V_E := \begin{bmatrix} 0 & 0 \\ 0 & E_{22} \end{bmatrix},\ A \leftarrow U_E A V_E,\ B \leftarrow U_E B := \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}.$$

2. Find an orthogonal matrix U_B such that $U_B B_1 = \begin{bmatrix} 0 \\ B_{21} \end{bmatrix}$, where B_{21} has full row rank, and partition the transformed matrices conformally
$$B \leftarrow \begin{bmatrix} U_B & 0 \\ 0 & I_r \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} := \begin{bmatrix} 0 \\ B_{21} \\ B_{31} \end{bmatrix},$$
$$E := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & E_{33} \end{bmatrix}, \quad A \leftarrow \begin{bmatrix} U_B & 0 \\ 0 & I_r \end{bmatrix} A := \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}.$$

3. Find an orthogonal matrix V_A such that $[A_{11}\ A_{12}]V_A = [\,R\ \ 0\,]$, where R is non-singular and set
$$A \leftarrow A \begin{bmatrix} V_A & 0 \\ 0 & I_r \end{bmatrix} = \begin{bmatrix} A_{11} & 0 & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$
where $A_{11} := R$.

4. Determine $F_1 = [\,0\ F_{12}\ 0\,]$ such that $A_{22} + B_{21}F_{12}$ is non-singular and set
$$A \leftarrow A + BF_1 = \begin{bmatrix} A_{11} & 0 & A_{13} \\ A_{21} & A_{22} + B_{21}F_{12} & A_{23} \\ A_{31} & A_{32} + B_{31}F_{12} & A_{33} \end{bmatrix}$$

5. Find an orthogonal matrix U_F such that
$$U_F \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \\ A_{31} & A_{32} \end{bmatrix} = \begin{bmatrix} S \\ 0 \end{bmatrix},$$
where S is non-singular of order $n - r$ and define the new partitioning of the transformed matrices as
$$E \leftarrow U_F E := \begin{bmatrix} 0 & E_{12} \\ 0 & E_{22} \end{bmatrix},\ \ A \leftarrow U_F A := \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix},\ \ B \leftarrow U_F B := \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$
where $A_{11} := S$.
Comment. E_{22} is further non-singular because the pencil $\lambda E - A$ is regular.

6. Compute F_{22} such that $\lambda(A_{22} + B_2F_{22}, E_{22}) \in \mathbb{C}^-$.

7. Put $F = [\,0\ F_{12}\ F_{22}\,] \begin{bmatrix} V_A^T & 0 \\ 0 & I_r \end{bmatrix} V_E^T$.

The stabilization problem to be solved at step 6 is a particular RSP with E_{22} non-singular. Algorithms to solve the general RSP are presented in the next sections. The reduction of system matrices at steps 1–5 is similar to the reduction used in the regularization procedure proposed in [3]. The column and row compressions can be performed by using QR- or RQ-type decompositions which can also efficiently exploit and preserve particular shapes of submatrices. For rank revealing compressions, as for instance the complete orthogonal decomposition of E performed at step 1, either the singular value decomposition or the rank revealing QR-decomposition [4] can be used. The second alternative is generally preferable because is substantially cheaper and possesses almost the same reliability for rank determinations. Fast and reliable tests permit to verify the conditions of Theorem 1. An easy verification of the condition $\mathrm{rank}([\, E\ AS_\infty\ B\,]) = n$ can be done by testing at step 3 the invertibility of the upper triangular matrix R, for example by estimating its condition number. The verification of the stabilizability of the system is performed in the procedures devised to solve the RSP at step 6. The proposed procedure is well suited for an efficient and robust computer implementation.

3. Direct R-stabilization methods

At step 6 of the SSDS Algorithm we have to solve the following particular RSP: For a system (E, A, B) with E *non-singular*, compute F such that $\lambda(A+BF, E) \in \mathbb{C}^-$. An obvious way to solve this problem is to transform it into a standard stabilization problem for the pair $(E^{-1}A, E^{-1}B)$ and to compute a stabilizing F for this pair by using one of various existing methods [10]. In this paper we propose 1 algorithms to solve the RSP which are generally applicable regardless E is invertible or not. Even when E is invertible, the explicit inversion of E is to be avoided in order to prevent possible accuracy losses if E is ill-conditioned.

In this section we describe a numerical approach which represents a generalization of the direct stabilization methods of standard state-space continuous- and discrete-time systems [10]. An alternative approach based on a pole assignment technique is presented in the next section. We assume in what follows that the system (E, A, B) is regular and stabilizable, but we do not assume the invertibility of E. As it will be apparent, the proposed algorithms involves practically no computational overhead for the particular case of an invertible E.

Algorithm RSDS. R-Stabilization of Descriptor Systems.

1. Reduce the pair (A, E) by an orthogonal similarity transformation, to the ordered *generalized real Schur form* (GRSF) [8], [9]
$$QEZ = \begin{bmatrix} E_{11} & E_{12} \\ 0 & E_{22} \end{bmatrix}, \quad QAZ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \tag{2}$$
where 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 $QB = [\, B_1^T\ B_2^T]^T$ partitioned conformally with the matrices in (2).
2. Compute F_2 such that $\lambda(A_{22} + B_2F_2, E_{22}) \subset \mathbb{C}^-$.
3. Set $F = [\, 0\ F_2\,]Z^T$.

It easy to see that the system (E_{22}, A_{22}, B_2) is controllable (because the original system is stabilizable) and E_{22} is invertible. Thus the stabilization problem to be actually solved at step

2 is the following simpler problem: For a *controllable* system (E, A, B) with E *non-singular* and $\lambda(A, E) \subset \mathbb{C}^+$, compute F such that $\lambda(A + BF, E) \subset \mathbb{C}^-$. Notice that we can even assume that the pair (A, E) is already in a GRSF.

The following theorems provide direct formulas for the solution of the particular stabilization problem at step 2 for continuous-time and discrete-time systems, respectively. They represent straightforward generalizations of similar results for standard systems [1], [2].

Theorem 3. *Let (E, A, B) be a controllable continuous-time system with E invertible and let $\beta \geq 0$ be such that $\mathcal{R}e\lambda_i(A, E) > -\beta$. Then a stabilizing F is given by*

$$F = -B^T(XE^T)^{-1}, \tag{3}$$

where X satisfies the continuous generalized Lyapunov equation

$$(A + \beta E)XE^T + EX(A + \beta E)^T = 2BB^T. \tag{4}$$

Remark. It can be additionally shown that the stabilizing feedback F moves the all generalized eigenvalues of the pair (A, E) on the vertical line $\mathcal{R}e(z) = -\beta$, that is we have $\mathcal{R}e\lambda_i(A + BF, E) = -\beta$. □

Theorem 4. *Let (E, A, B) be a controllable discrete-time system with E invertible and let* $0 < \beta \leq 1$ *be such that $|\lambda_i(A, E)| \geq \beta$. Then a stabilizing F is given by*

$$F = -B^T(EXE^T + BB^T)^{-1}A \tag{5}$$

where X satisfies the discrete generalized Lyapunov equation

$$AXA^T = \beta^2 EXE^T + 2BB^T \tag{6}$$

Remark. The proofs of theorems are straightforward applications of the results derived in [1] and [2] for standard systems ($E = I$). In both theorems the controllability condition can be relaxed to stabilizability. In this case the only necessary modifications occur in the expressions of the stabilizing F and consist in replacing the inverses by pseudo-inverses. □

For the solution of the generalized Lyapunov equations (4) and (6) algorithms and software provided in [6] can be used. Note that the pairs $(A + \beta E, E)$ or $(A, \beta E)$ are already in a GRSF and thus the cost of solving the generalized Lyapunov equations is negligible in comparison with the cost of computing the ordered GRSF of the pair (A, E). The testing of controllability of the system (E, A, B) can be done by simply checking the invertibility of the symmetric matrix X. The smallest singular value (eigenvalue) of this positive definite matrix measures the least distance to singularity and can be viewed as a measure of the degree of controllability of the system (E, A, B), and thus of the stabilizability of the original descriptor system.

4. R-stabilization by pole assignment

A numerically more satisfactory approach for solving the RSP can be based on pole assignment techniques. Although any pole assignment algorithm for descriptor systems can be used as a stabilization algorithm, we feel that the generalized Schur algorithm presented in this section is particularly well suited to solve the formulated RSP. An earlier version of this algorithm was described in [13]. Its main idea is to move sequentially all unstable poles of the system in the stable region by using partial state feedback matrices which do not perturb the locations of already stable poles. The procedure is based on an efficient technique for updating and reordering of the GRSF of the pair (A, E). The main steps of this algorithm are presented below.

Algorithm RSPA. R-stabilization by Pole Assignment.

1. Reduce the pair (A, E) by an orthogonal similarity transformation, to the ordered GRSF
$$QEZ = \begin{bmatrix} E_{11} & E_{12} \\ 0 & E_{22} \end{bmatrix}, \; QAZ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$$
where $E_{11} \in \mathbb{R}^{q,q}$ and $A_{11} \in \mathbb{R}^{q,q}$, 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 $E \leftarrow QEZ, A \leftarrow QAZ, B \leftarrow QB$ and set $F = 0$.
2. If $q = n$, Stop.
3. Let α and δ be the $k \times k$ last diagonal blocks of A and E respectively, and let β the $k \times m$ matrix formed from the last k rows of B. Choose a $k \times k$ matrix γ such that $\lambda(\gamma) \subset \mathbb{C}^-$ and compute $\varphi = \beta^+(\delta\gamma - \alpha)$. (The superscript $^+$ stays for pseudo-inverse.)
4. Compute $A \leftarrow A + B[\,0\ \varphi\,]$, $F \leftarrow F + [\,0\ \varphi]Z^T$.
5. Move the last $k \times k$ blocks of A and E to positions $(q+1, q+1)$ by using an orthogonal similarity transformation $E \leftarrow \widetilde{Q}E\widetilde{Z}$, $A \leftarrow \widetilde{Q}A\widetilde{Z}$ and compute $B \leftarrow \widetilde{Q}B$, $Z \leftarrow Z\widetilde{Z}$, $Q \leftarrow \widetilde{Q}Q$; put $q \leftarrow q + k$ and go to step 2.

This algorithm determines in places of matrices E, A and B, the matrices QEZ, $Q(A+BF)Z$ and QB respectively, which are orthogonally similar with the matrices of the stabilized closed-loop system $(E, A + BF, B)$. The resulting pair $(Q(A + BF)Z, QEZ)$ is in a GRSF and the orthogonal matrices used for the reduction are also available (the accumulation of Q is however optional).

The reduction of the pair (A, E) to the GRSF at step 1 and the reordering of diagonal block pairs of the GRSF are performed by standard numerically stable algorithms [8], [9]. The RSPA algorithm moves at each iteration (steps 2-5) at most two unstable poles of the open-loop system into the stable region by using the partial feedback $K = [\,0\ \varphi]$ at step 4. Only the last at most two columns of A are altered at this step and therefore the GRSF is preserved. The orthogonal transformation at step 5 brings a new pair of blocks of A and E in their last diagonal positions and the computations are resumed if still there remain unstable poles (step 2). The norm of the partial feedback matrix K can be minimized by a suitable choice of γ. This aspect is important for improving the numerical properties of the algorithm.

The RSPA algorithm is a generalization of a pole assignment algorithm for standard systems [11]. The roundoff error analysis of that algorithm [12] revealed that if each partial feedback matrix of the form $K = [\,0\ \varphi]$ satisfies the condition $\|K\| \leq \|A\|/\|B\|$, 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 the E matrix. 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.

5. Conclusions

Numerically reliable computational algorithms for the stabilization of descriptor systems, well suited for robust software implementations, have been proposed. These algorithms could serve as simple design procedures of controllers and observers for descriptor systems or as a preliminary computational step for the initialization of the Newton's method for solving descriptor Riccati equations. All steps of the proposed algorithms can be performed by using exclusively numerically stable procedures. The pole assignment algorithm is numerically more reliable than the direct stabilization methods and in certain conditions, can be considered even as a numerically stable procedure. It can perform the R-stabilization of a descriptor system with respect to an arbitrary stability region of the complex plane.

References

[1] E. S. Armstrong. An extension of Bass algorithm for stabilizing linear continuous constant systems. *IEEE Trans. Autom. Control*, AC-20 (1975) 153–154.

[2] E. S. Armstrong and G. T. Rublein. A stabilization algorithm for linear discrete constant systems. *IEEE Trans. Autom. Control*, AC-21 (1976) 629–631.

[3] A. Bunse-Gerstner, V. Mehrmann, and N. K. Nichols. Regularization of descriptor systems by derivative and proportional state feedback. *SIAM J. Matrix Anal. Appl.*, 13 (1992) 46–67.

[4] T. Chan. Rank revealing QR factorizations. *Lin. Alg. & Appl.*, 88/89 (1987) 67–82.

[5] L. Dai. *Singular Control Systems*, volume 118 of *Lect. Notes Contr. Inf. Scie.* Springer Verlag, New York, (1989).

[6] J. D. Gardiner, A. J. Laub, J. J. Amato, and C. B. Moler. Solution of the Sylvester matrix equation AXB+CXD = E. *ACM Trans. Math. Software*, 18 (1992) 223–231.

[7] V. L. Mehrmann. *The Autonomous Linear Quadratic Control Problem*, volume 163 of *Lect. Notes Contr. Inf. Scie.* Springer Verlag, Berlin, (1991).

[8] C. B. Moler and G. W. Stewart. An algorithm for generalized matrix eigenvalue problem. *SIAM J. Numer. Anal.*, 10 (1973) 241–256.

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

[10] A. Varga. On stabilization algorithms for linear time-invariant systems. *Rev. Roum. Scie. Techn. – Electrotech. et Energ.*, 26 (1981) 115–124.

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

[12] A. Varga. The numerical stability of an algorithm for pole assignment. In G. Leininger, editor, *Proc. 2-nd IFAC CADCS Symp., West Lafayette, Indiana.* Pergamon Press, Oxford, 1982.

[13] A. Varga. 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*, 1983.