# Upcoming Numerical Linear Algebra Issues in Systems and Control Theory

Paul Van Dooren

University of Illinois at Urbana-Champaign
Coordinated Science Laboratory
1308 W. Main Str., Urbana, IL 61801
Phone: (217)3330656, Fax: (217)2441642
E-mail: vdooren@uicsl.csl.uiuc.edu

Abstract. We discuss a number of novel issues in the interdisciplinary area of numerical linear algebra and systems and control theory. Although we do not claim to be exhaustive we give a number of "upcoming issues" which we believe will play an important role in the near future. These are : sparse matrices, structured matrices, novel matrix decompositions and numerical shortcuts. Each of those is presented in relation to a particular (class of) control problems. These are respectively : large scale control systems, polynomial system models, control of periodic systems, and normalized coprime factorizations in robust control.

**Key words.** Numerical algorithms, linear algebra, sparse matrices, polynomial systems, periodic systems, robust control

## Introduction

During the last few decades linear algebra has played an important role in advances being made in the area of systems and control [30]. The most profound impact has been in the computational and implementational aspects, where numerical linear algebraic algorithms have strongly influenced the ways in which problems are being solved. This paper discusses a number of novel numerical linear algebra issues in this interdisciplinary area. Although we do not claim to be exhaustive we give a number of "upcoming issues" which we believe will play an important role in the near future. These numerical linear algebra issues are :

- **Sparse matrices**. Large plants in control typically arise from discretizations of continuum problems (such as those that appear in mechanics or chemistry). The models obtained from that are then automatically sparse, such as finite element methods used for mechanical problems. It is also commonly observed that large scale problems in other application areas typically involve matrices with some kind of sparsity or structure. A recent report on large matrix problems [9] shows that indeed there are *no large general dense matrix problems* to be tackled in most application areas.

- **Structured matrices**. A compact way of representing the transfer function of a system is by using polynomial system models. The number of parameters in these models is often much lower than for their equivalent state space model. Numerical linear algebra problems occurring there typically involve matrix problems with some kind of structure (Hankel, Toeplitz, companion, ...) and the solution of the underlying control problem usually boils down to some matrix decomposition of such a structured matrix.

- **New decompositions**. The most reliable numerical linear algebra methods proposed for particular control problems are related to particular eigenvalue and singular value decompositions of "special" matrices, such as the Schur decomposition of a Hamiltonian matrix for solving Riccati equations. Several new decomposition have been proposed recently in numerical linear algebra and their application to control theoretic problems still has to be explored.

- **Numerical shortcuts**. A typical approach for solving matrix problems for which there is no "direct" algorithm, is to break it down into a sequence of "intermediate" problems for which one can apply known techniques. A typical early example of this was the construction of the normal equations for solving least squares problems. This is nowadays replaced by the "shortcut" given by the QR decomposition, which is known to be more reliable in general. Similar "detours" are still present in the solution of certain control problems and ought to be avoided if possible.

In the sections below we elaborate on these topics and give for each a particular control theoretic example where it applies. These control examples illustrate well the issues being raised.

## Sparse matrices and large scale systems

Large plants in control typically arise from discretizations of continuum problems such as finite element methods used for mechanical problems. The models used are then almost always represented in state space (or generalized state space) form. The plant matrices $\{A, B, C, D\}$ typically have a special structure such as sparsity (due to modeling techniques using e.g. finite element methods) or diagonal dominance (inherited from the physical properties of the system being modeled [21], [25]).

For large scale systems, the design of the controller is often faced with the practical constraint that it has to be of relatively low order. The main reasons for this are that the controller has to run in real time at a specific clock rate or has to be implemented on existing hardware of limited capacity (a typical example of this is the laser beam tracking device of the compact disc player which has to be implemented on an existing digital signal processor with relatively small memory and processing speed [26]). But even when no such constraint is imposed, model reduction is a useful approach for yielding approximate solutions to particular control problems. Instead of solving the control problem for the large scale system one solves the corresponding problem for its lower order approximation and then tries to derive from that an *approximate solution* for the original problem.

A justification for this follows from the following observation. A large class of model reduction methods of a system given in state space form $\{A, B, C, D\}$ can be interpreted as performing a similarity transformation $T$ yielding $\{A^{(t)}, B^{(t)}, C^{(t)}, D\} \doteq \{T^{-1}AT, T^{-1}B, CT, D\}$, and then extracting from that the leading diagonal system $\{\hat{A}, \hat{B}, \hat{C}, D\}$, i.e.

$$\left[\begin{array}{c|c} T^{-1}AT & T^{-1}B \\ \hline CT & D \end{array}\right] \doteq \left[\begin{array}{cc|c} A_{1,1}^{(t)} & A_{1,2}^{(t)} & B_1^{(t)} \\ A_{2,1}^{(t)} & A_{2,1}^{(t)} & B_2^{(t)} \\ \hline C_1^{(t)} & C_2^{(t)} & D \end{array}\right] \tag{1}$$

$$\left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & \hat{D} \end{array}\right] \doteq \left[\begin{array}{c|c} A_{1,1}^{(t)} & B_1^{(t)} \\ \hline C_1^{(t)} & D \end{array}\right].$$

When writing $T \doteq [X_1 \mid X_2]$, $T^{-1} \doteq [Y_1^t \mid Y_2^t]^t$, then $\Pi \doteq X_1 Y_1$ is a projector on $X_1$ along $Y_1$, and the subsystem $\{\hat{A}, \hat{B}, \hat{C}, D\}$ is in fact the

original system restricted to that projector : $\{\Pi A\Pi, \Pi B, C\Pi, D\}$. Such model approximations thus only differ in the choice of projector. Each choice typically tries to achieve some kind of decomposition. Modal approximation e.g. selects "dominant" frequencies in $A$ and performs thus a block diagonal decomposition (1), i.e. with $A_{1,2}^{(t)}$ and $A_{2,1}^{(t)}$ being 0 and $A_{1,1}^{(t)}$ containing the dominant eigenfrequencies. When large scale systems $\{A, B, C, D\}$ are involved one can only afford approximate decompositions, i.e. with $A_{1,2}^{(t)}$ and $A_{2,1}^{(t)}$ small compared to $A^{(t)}$. Iterative techniques for large systems are indeed only applied a limited number of steps for reasons of complexity. Techniques for nearly decomposed systems [25], [21] can then be used to derive bounds on the approximation criterion or an iterative refinement to improve the decomposition.

The above ideas thus suggest the following approach for solving particular control problems for large scale systems :

- Choose a projector $\Pi$ yielding a lower order system $\{\Pi A\Pi, \Pi B, C\Pi, D\}$
- Solve the given problem for this lower order system using dense matrix techniques
- "Lift" back the solution to the original coordinate system.

This sounds of course very simple but the crux is to find the projector $\Pi$ that achieves two important goals. It should be *easy to construct*, i.e. one should be able to exploit sparsity of the model $\{A, B, C, D\}$ for the construction of $\Pi$. And the lifted solution should provide a *good approximation* for the true solution of the control problem, i.e. certain "performance bounds" ought to be satisfied. For the first goal there exist a number of iterative techniques such as Krylov type schemes (Lanczos, Arnoldi, GMRES, QMR [24], [10]). The second goal is of course problem dependent and is probably the most challenging one. One may have to assume certain system properties here (such as diagonal dominance) in order to derive sufficiently powerful results. Also criteria have to be selected to obtain an appropriate system performance and how far iteration techniques have to be applied in order to satisfy corresponding bounds. The complexity of this approach typically ought to be an order of magnitude less (in terms of the system order $n$) than the complexity of dense matrix techniques applied to the original problem.

## Structured matrices and polynomial system models

Polynomial system models have been proposed for most control problems as an interesting alternative to state space techniques [2], [22]. The key advantage of this approach is the reduced complexity of the algorithms for solving particular problems. In turn, the main reason for that is the reduced number of parameters needed to represent a system of the same order. As an example, a $n$-th order SISO (single input single output) system requires $\mathrm{O}(n^2)$ parameters when represented by an arbitrary state space model $\{A, B, C, D\}$, while it requires only $\mathrm{O}(n)$ parameters when represented by a polynomial model $\{p(z), q(z)\}$.

Consider then the problem of checking minimality of the given realization. In both cases this amounts to a rank test. For the state space realization one has to check the rank of the controllability and observability matrices, which requires $\mathrm{O}(n^3)$ operations. For the polynomial model one has to check coprimeness of $p(z)$ and $q(z)$, which amounts to checking the rank of the Sylvester matrix [2], [11] :

$$S_{2n} \doteq \begin{bmatrix} p_0 & p_1 & \dots & p_n & 0 & \dots & 0 \\ 0 & p_0 & p_1 & \dots & p_n & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & & \ddots & 0 \\ 0 & \dots & 0 & p_1 & p_2 & \dots & p_n \\ 0 & \dots & 0 & q_1 & q_2 & \dots & q_n \\ \vdots & \iddots & \iddots & \iddots & & \iddots & 0 \\ 0 & q_0 & q_1 & \dots & q_n & & \vdots \\ q_0 & q_1 & \dots & q_n & 0 & \dots & 0 \end{bmatrix} \tag{2}$$

Although this is a $2n \times 2n$ matrix its rank can be computed by Euclid's algorithm and this requires only $\mathrm{O}(n^2)$ operations !

This difference of an order $n$ in operation count is typical in the SISO case and thus speaks strongly in favor of polynomial models. Yet, the algorithms for polynomial models are typically known to be numerically unstable (see e.g. [27]). Indeed, Euclid's algorithm can be shown to be equivalent to Gaussian elimination *without pivoting* on the Sylvester matrix, and this is known to be numerically unreliable. Moreover Euclid's algorithm is also at the basis of most polynomial matrix decompositions such as the Smith form or Hermite form of a polynomial matrix. The polynomial approach thus seems to sacrifice accuracy for speed.

We strongly believe that iterative refinement could be a way to overcome the above drawback of the polynomial approach. It is known indeed that

a few steps of iterative refinement can often turn an unstable method into a stable one [39] and this ought to be tried out on the matrix agorithm for polynomial models. If the number of refinement steps is still significantly lower than $n$ then one still has an efficient algorithm compared to the corresponding state space algorithms. We now illustrate this with two examples. The first one is the computation of poles of a system. For a state space system this boils down to computing the eigenvalues of the matrix $A$, and it requires $\mathrm{O}(n^3)$ operations using the (numerically stable) $QR$ algorithm. For a polynomial model this amounts to computing the zeros of $q(z)$, but this can be done in $\mathrm{O}(n^2)$ operations using a stable algorithm [20] ! The basic iterative scheme underlying this is the Newton-Raphson algorithm which is a kind of iterative refinement.

The second example is a more important one in control applications and is known as *spectral factorization*. It is the corner stone in problems of optimal control and robust control [8], [36]. For a state space model one typically solves this via a Riccati equation which is again an eigenvalue problem (with particular symmetry) and requires $\mathrm{O}(n^3)$ operations [23], [28]. For a (discrete time) SISO polynomial model one can state the problem as follows. Given the finite power series

$$\Phi(z) = \phi_n z^{-n} + \cdots + \phi_1 z^{-1} + \phi_0 + \phi_1 z + \cdots + \phi_n z^n \tag{3}$$

find a *stable* polynomial $p(z)$ (i.e. zeros inside $\|z\| < 1$) such that :

$$\Phi(z) = p(z^{-1})\, p(z). \tag{4}$$

It is known that this always has a solution if $\Phi(e^{j\omega}) \geq 0$ for all $\omega$.

Vostry's [37] proposed an algorithm for this which starts with some stable polynomial $x^{(0)}(z)$ and then iterates with :

$$p^{(i)}(z^{-1})x^{(i)} + p^{(i)}(z).x^{(i)}(z^{-1}) = 2\Phi(z)$$

$$p^{(i+1)}(z) = \frac{1}{2}[p^{(i)}(z) + x^{(i)}(z)] \tag{5}$$

One iteration step requires the solution of a system of equations with a matrix close to the Sylvester matrix and again this can be done in $O(n^2)$ operations. The stability of a single step is *not* guaranteed, but since it is a method of iterative refinement using the original data $\Phi(z)$ for computing the correction, the overall stability of Vostry's method is quite good ! The convergence of $p^{(i)}(z)$ to $p(z)$ is quadratic when no roots of $p(z)$ are close

to the unit circle and the overall process is thus also $O(n^2)$. It is in fact a Newton correction scheme applied to equation (3).

## Novel decompositions and periodic Riccati equations

The most reliable numerical linear algebra methods proposed for particular control problems are related to particular eigenvalue and singular value decompositions of "special" matrices, such as the Schur decomposition of a Hamiltonian matrix for solving Riccati equations [23], [28]. Several new decompositions have been proposed recently in numerical linear algebra and their application to control theoretic problems still has to be fully exploited. As an example the generalized Schur form has been shown to have several applications in geometric systems theory [27], and this decomposition is an extension of the standard Schur form to singular pencils of matrices.

In [3][1] a new decomposition for a sequence of matrices $A_i$, $B_i$, $i = 1, \ldots, K$ is proposed. Consider the set of (homogenous) difference equations

$$B_i \cdot x_{i+1} = A_i \cdot x_i, \quad i = 1, \ldots \tag{6}$$

with *periodic coefficients* $A_i = A_{i+K}$, $B_i = B_{i+K}$. For period $K = 1$ one has the constant coefficient case $A_i = A$, $B_i = B$ and it is well-known that the generalized eigenvalues of the pair $A, B$ yields important information of the system (6). When $K > 1$ one can derive from (6) a set of $K$ time invariant systems which describe completely the behavior of (6). For simplicity we assume all $B_i$ to be invertible. Then define the matrices $S_i = B_i^{-1} A_i$ yielding the explicit system :

$$x_{i+1} = B_i^{-1} A_i \cdot x_i = S_i \cdot x_i, \quad i = 1, \ldots \tag{7}$$

and using $S^{(k)} = S_{k+K-1} \cdot \ldots S_{k+1} \cdot S_k$, $k = 1, ..., K$ the set of $K$ *time invariant* but *subsampled systems* :

$$\begin{array}{ll} x_{1+(i+1)K} & = S^{(1)} \cdot x_{1+iK}, \quad i = 1, 2, \ldots \\ x_{2+(i+1)K} & = S^{(2)} \cdot x_{2+iK}, \quad i = 1, 2, \ldots \\ & \vdots \\ x_{K+(i+1)K} & = S^{(K)} \cdot x_{K+iK}, \quad i = 1, 2, \ldots \end{array} \tag{8}$$

[1] Similar unpublished ideas are being pursued by John Hench, UC Santa Barbara (personal communication)

The behaviour of these systems will thus require the eigenvalues and eigenvectors of the *periodic matrix products* $S^{(k)}$. An implicit decomposition of these matrices is now obtained as follows. One proves that there exist unitary matrices $Q_i,\ Z_i,\ i = 1, ..., K$ such that :

$$\begin{array}{llll} \hat{B}_1 & = Z_1^* \cdot B_1 \cdot Q_2 & \hat{A}_1 & = Z_1^* \cdot A_1 \cdot Q_1 \\ \hat{B}_2 & = Z_2^* \cdot B_2 \cdot Q_3 & \hat{A}_2 & = Z_2^* \cdot A_2 \cdot Q_2 \\ & \vdots & & \\ \hat{B}_K & = Z_K^* \cdot B_K \cdot Q_{K+1} & \hat{A}_K & = Z_K^* \cdot A_K \cdot Q_K \end{array} \tag{9}$$

where now all matrices $\hat{B}_i,\ \hat{A}_i$ are upper triangular. It is easy to see that in fact the matrices $Q_i$ transform the vectors $x_i$ to $\hat{x}_i = Q_i^* \cdot x_i$ and the matrices $S^{(i)}$ to $\hat{S}^{(i)} = Q_i^* \cdot S^{(i)} \cdot Q_i$. The latter are in upper Schur form, such that the "hat" versions of the systems (8) are now all simultaneously in upper triangular form. Notice that the diagonal elements of the $\hat{S}^{(i)}$ matrices are all equal since they are the products of the diagonal elements of the upper triangular matrices $\hat{B}_i^{-1}\hat{A}_i$. So, if one matrix $\hat{S}^{(i)}$ has a particular ordering of eigenvalues then all other matrices $\hat{S}^{(j)}$ have the same ordering of eigenvalues. In [3] we give an algorithm to compute the above decomposition *implicitly*, i.e. without ever forming the products $S^{(i)}$. Moreover we show how to reorder the eigenvalues of these Schur forms. We called this the *periodic Schur factorization* [3] because of its relation to periodic systems (6).

The application of this decomposition to control theory is apparent. In optimal control of a periodic system one considers the problem :

$$\begin{array}{c} \text{Minimize } J = \sum_{k=1}^{\infty} z_k^T Q_k z_k + u_k^T R_k u_k \\ \text{subject to } z_{k+1} = F_k z_k + G_k u_k \end{array} \tag{10}$$

where the matrices $Q_k, R_k, F_k, G_k$ are periodic with period $K$. The Hamiltonian equations are periodic homogenous systems of difference equations (6) in the state $z_k$ and co-state $\lambda_k$ of the system. The correspondences with (6) are :

$$x_k \doteq \begin{bmatrix} \lambda_k \\ z_k \end{bmatrix}, B_k \doteq \begin{bmatrix} -G_k R_k^{-1} G_k^T & I \\ F_k^T & 0 \end{bmatrix}, A_k \doteq \begin{bmatrix} 0 & F_k \\ I & Q_k \end{bmatrix}. \tag{11}$$

For finding the periodic solutions to the underlying periodic Riccati equation one has to find the stable invariant subspaces of matrices $S^{(k)}$ as above,

which happen to be simplectic in the discrete time case. Clearly the Schur form is useful here as well as the reordering of eigenvalues [23], [28]. In pole placement of periodic systems, again the Schur form and reordering is useful when one wants to extend Varga's Schur algorithm [35] for pole placement [15].

Another decomposition recently proposed for a sequence of matrices is the generalized $QR$-decomposition [7], which is connected to the computation of singular values of such sequences. Applications of this in control has not fully been explored yet, but some indications are given in [7]. They inlude the computation of geometric concepts for time varying discrete time systems.

## Numerical shortcuts and normalized coprime factorizations

A typical approach for solving matrix problems for which there is no "direct" algorithm, is to break it down into a sequence of "intermediate" problems for which one can apply known techniques. We first give a few typical example of this in linear algebra and then extend these ideas to control theory. The particular example of normalized coprime factorizations is then worked out.

An early example of numerical detour was the construction of the normal equations $A^TA \cdot x = A^Tb$ for solving the least squares problem :

$$min\|Ax - b\|_2. \tag{12}$$

This can be "shortcut" by the $QR$ decomposition [14], which is known to be more reliable in general. The sensitivity of the normal equations can only be worse than that of the least squares problem and the numerical stability of the $QR$ decomposition is superior to that of the Choleski decomposition applied to $A^TA \cdot x = A^Tb$. Another typical "detour" is the construction of the matrix $B^{-1}A$ when computing the generalized eigenvalues or generalized singular values of the matrix pair $A, B$. Nowadays there are implicit decompositions of such matrix pairs which directly yield their generalized eigenvalues and singular values without constructing $B^{-1}A$ [14].

Early examples of numerical shortcuts in control theory include the Schur and generalized Schur methods for Riccati equations [23], [28] and staircase forms for computing various concepts in geometric systems theory [27]. But there are still many other "detours" around that require direct approaches

with better numerical properties. We give here an example from robust control. Modern control theory has been largely influenced by the recent development of $H_\infty$ techniques. The underlying theory is now rather well understood but the computational techniques are lagging behind. The design of the robust controller consists in solving factorization problems of transfer matrices and/or related state-space matrix equations [1]. Examples of transfer function factorizations needed here are all-pass factor extraction, inner outer factorization and normalized coprime factorization. Numerical algorithms for such factorizations have been analyzed from a state space and generalized state space point of view [23] [31]. It was found that simple recursive solutions can be obtained for most of them when starting from state space models in so-called condensed forms (such as Schur or generalized Schur forms). The advantage of this approach is a reduced computational complexity and good numerical properties in the corresponding algorithms (see [23], [31] and references therein).

One example where we can expect to shortcut currently proposed algorithms is the construction of normalized coprime factorizations of a given transfer function $R(s) = N(s) \cdot D^{-1}(s)$ with

$$D^*(-\overline{s}) \cdot D(s) + N^*(-\overline{s}) \cdot N(s) = I. \tag{13}$$

Presently, one constructs first a stable coprime factorization (with respect to some region $\Gamma$), and then one normalizes it via a spectral factorization problem. A more direct approach consists in noticing that $D(s)$ and $N(s)$ must be submatrices of a $\Gamma$-stable all-pass $U(s)$ that displays the kernel of $[R(s) \mid -I]$ as follows :

$$\begin{bmatrix} R(s) & -I \end{bmatrix} \cdot U(s) \doteq$$

$$\begin{bmatrix} R(s) & -I \end{bmatrix} \cdot \begin{bmatrix} D(s) & \tilde{N}(s) \\ N(s) & \tilde{D}(s) \end{bmatrix} = \begin{bmatrix} \tilde{R}(s) & 0 \end{bmatrix}. \tag{14}$$

This problem now becomes one of rank factorization with all-pass factors and ideas of [31] could be used to tackle this. Notice that this approach avoids "squaring up" the original problem and then performing a spectral factorization. The squared version requires the solution of a Riccati equations where all eigenvalues occur in pairs, whereas this direct decomposition requires the solution of an eigenvalue problem of only half that size. Moreover, just as for the normal equation example, one should expect better numerical sensitivity properties for the direct approach. Similar remarks were already

made in [31] for the problems of inner-outer factorizations and other coprime factorizations. The approach proposed there involved eigenvalue problems of typically lower order by using particular state space decompositions.

## Concluding remarks

The topics discussed in previous sections all point to the significant role linear algebra problems play in systems and control theory. Over the years, numerous algorithms have been developed in that area. Because of the increasing complexity of the problems being tackled, some of them have become challenging from a numerical point of view as well. The interdisciplinary field of numerical linear algebra and linear system theory has lead to some significant developments in the last decade and several of these results are nowadays being implemented in software for CACSD [34], [38]. However, numerical methods in this area are still far from complete. Above we indicated some of these :

- Most of the techniques available today are specifically aimed at dense systems. Only few methods are available that deal with sparse matrices. These are typically based on Krylov type techniques [19], [4], [24]. More work ought to be done in that area using new results as e.g. the QMR method [10].

- There are very few numerically reliable methods for polynomial system models, although typically algorithms in that area are fast. The complexity of these agorithms is usually lower because the underlying matrix problems are highly structured. Analyzing stability of algorithms for structured matrices needs special care as indicated in [32], but use of iterative refinement ought to be useful (see also [18]).

- New decompositions are being found in linear algebra that are particularly relevant to various control problems. Schur forms and generalized Schur forms have already extensively been applied [23], [27], [28]. Also condensed forms have been used for various control problems [29]. But similar extensions for generalized singular values [16], and decompositions involving sequences of matrices have not fully been explored yet [7], [3].

- Many detours are still present in control algorithms because of the absence of direct methods or appropriate decompositions. This is particularly true for problems in robust control, since there one has only

recently been able to reduce the problems to known ones involving Riccati equations [13], [8]. Only little attention has been paid yet to the development of appropriate numerical techniques for these specialized problems.

The above list is by far not exhaustive. Other recent developments include e.g. :

- Control algorithms developed for special architectures [5], [12], [6] exploiting parallelism.
- Software developments such as interactive packages [38] and software libraries [34]
- Combinations of the above issues, such as in the problem of model reduction of large sparse systems. This clearly involves sparse matrix techniques but all three other issues come up as well as e.g. indicated in [33].

**Acknowledgement**

Part of this research was performed while visiting the Institute of Mathematics and Applications of the University of Minnesota, Minneapolis, during the summer quarter of the Applied Linear Algebra Year organized there. We greatly appreciated the hospitality and the productive atmosphere of that institute. We also acknowledge the support of the Research Board of the University of Illinois at Urbana-Champaign (Grant P 1-2-68114).

## References

[1] B.D.O. Anderson, Y. Liu, Controller reduction: concepts and approaches, *IEEE Trans. Aut. Contr.* **AC-34** (1989) 802-812.

[2] S. Barnett, *Polynomials and linear control systems*, Marcel Dekker, New York, 1983.

[3] A. Bojanczyk, G. Golub, P. Van Dooren, *The perodic Schur decomposition. Algorithms and applications*, Int. Rept. Stanford Univ., Stanford CA 94305, 1992.

[4] D. Boley, G. Golub, The Lanczos-Arnoldi algorithm and controllability, *Systems and Control Letters* **4** (1984) 317-324.

[5] J.-P. Charlier, P. Van Dooren, A systolic algorithm for Riccati and Lyapunov equations, *Mathematics of Control, Signals and Systems* **2** (1989) 109-136.

[6] B.N. Datta, Parallel algorithms in control theory, *Proceedings 30th IEEE Conf. Dec. & Contr.* (1991) 1700-1704.

[7] B. De Moor, P. Van Dooren, Generalizations of the singular value and QR decomposition, *SIAM Matr. Anal. & Applic.* **13** (1992) to appear.

[8] J. Doyle, B. Francis and A. Tannenbaum, *Feedback Control Theory*, McMillan, 1992.

[9] A. Edelman, *The first annual large dense linear system survey*, Int. Rept. Univ. California, Berkeley CA 94720, 1991.

[10] R. Freund, N. Nachtigal, QMR: a quasi-minimal residual method for non-Hermitian linear systems, *Numer. Math.* **60** (1991) 315-339.

[11] F.R. Gantmacher, *The Theory of Matrices I and II*, Chelsea, New York, 1959.

[12] J. Gardiner, *Iterative and parallel algorithms for the solution of algebraic Riccati equations*, Ph. D. Thesis, Univ. California, Santa Barbara, 1988.

[13] K. Glover, D. Mc Farlane, Robust stabilization of normalized coprime factor plant descriptions with $H_\infty$-bounded uncertainty, *IEEE Trans. Aut. Contr.* **AC-34** (1989) 821-830.

[14] G. Golub, C. Van Loan, *Matrix Computations* 2nd edition, The Johns Hopkins University Press, Baltimore, Maryland, 1989.

[15] O. Grasselli, S. Longhi, The geometric approach for linear periodic discrete-time systems, *Lin. Algebra & Applic.* **158** (1991) 27-60.

[16] M. Heath, A. Laub, C. Paige, R. Ward, Computing the singular value decomposition of a product of two matrices, *SIAM J. Sci. Stat. Comput.* **7** (1986) 1147-1159.

[17] A. Heemink, Kalman filtering of two-dimensional shallow water flow on a vector processor, *Supercomputer* **20/21** (1987) 32-44.

[18] D. Higham, N. Higham, Backward error and condition of structured linear systems, *SIAM Matrix Anal. & Appl.* **13** (1992) 162-176.

[19] S. Hodel, K. Poolla, Numerical solution of very large sparse Lyapunov equations through approximate power iteration, *Proceedings 29th IEEE Conf. Dec. & Contr.* (1990) 291-296.

[20] M. Jenkins, J. Traub, A three-stage algorithm for real polynomials using quadratic iteration, *SIAM Numer. Anal.* **7** (1970) 545-566.

[21] P. Kokotovic, W. Perkins, J. Cruz, G. D'Ans, $\epsilon$-coupling method for near-optimum design of large-scale linear systems, *Proceedings of the IEE* **5** (1969) 889-892.

[22] V. Kucera, *Discrete linear control. The polynomial equation approach*, Chichester : Wiley & Sons, 1979.

[23] A. Laub, Invariant subspace methods for the numerical solution of Riccati equations, in *The Riccati equation*, Eds. Bittanti, Laub, Willems, Springer Verlag, 1990.

[24] Y. Saad, Overview of Krylov subspace methods with applications to control problems, in *Signal Processing, Scattering and Operator Theory, and Numerical Methods*, Birkhäuser Boston Inc. 1990,

[25] R. Srikant, T. Basar, Sequential decomposition and policy iteration schemes for M-player games with partial weak coupling, *Automatica* (1992) to appear.

[26] M. Steinbuch, J. Terlouw, O. Bosgra, S. Smit, Multivariable application of $\mu$ analysis of servo system design in the consumer electronics industry, *IEE Colloquium on Successful Industrial Applications of Multivariable Analysis*, London, 1990.

[27] P. Van Dooren, The generalized eigenstructure problem in linear system theory, *IEEE Trans. Aut. Contr.* **AC-26** (1981) 111-129.

[28] P. Van Dooren, A generalized eigenvalue approach for solving Riccati equations, *SIAM Sci. & Stat. Comp.* **2** (1981) 121-135.

[29] P. Van Dooren, M. Verhaegen, On the use of unitary state-space transformations, in *Special Issue of Contemporary Mathematics on Linear Algebra and its Role in Linear System Theory*, AMS, 1985.

[30] P. Van Dooren, Numerical aspects of system and control algorithms, *Journal A* **30** (1989) 25-32.

[31] P. Van Dooren, Rational and polynomial matrix factorizations via recursive pole-zero cancellation, *Linear Algebra & Applications* **137/138** (1990) 663-697.

[32] P. Van Dooren, Structured linear algebra problems in digital signal processing, in *Numerical linear algebra, digital signal processing and parallel algorithms*, Nato Series F, Springer Verlag (1990) 361-384.

[33] P. Van Dooren, Numerical linear algebra techniques for large scale matrix problems in systems and control, *Proceedings 31st IEEE Conf. Dec. & Contr.* (1992) to appear.

[34] A. van den Boom, A. Brown, F. Dumortier, A. Geurts, S. Hammarling, R. Kool, M. Vanbegin, P. Van Dooren, S. Van Huffel, SLICOT, a subroutine library for control and system theory, *IFAC Symp. on CADCS*, Swansea UK, July 1991.

[35] A. Varga, A Schur method for pole assignment, *IEEE Trans. Aut. Contr.* **AC-26** (1981) 517-519.

[36] M. Vidyasagar, *Control System Synthesis : a Factorization Approach*, MIT Press, Cambridge, MA, 1985.

[37] Z. Vostry, New algorithm for polynomial spectral factorization with quadratic convergence I. *Kybernetika* **11** (1975) 415-422.

[38] R. Walker, C. Gregory, S. Shah, MATRIX$_x$ : A data analysis, system identification, control design and simulation package, *Control Systems Magazine* **2:4** (1982) 30-37.

[39] J. H. Wilkinson, *The algebraic eigenvalue problem*, Clarendon press, Oxford, 1965.