Technische Universität Chemnitz

Sonderforschungsbereich 393

Numerische Simulation auf massiv parallelen Rechnern

Thilo Penzl

A Cyclic Low Rank Smith Method for Large Sparse Lyapunov Equations with Applications in Model Reduction and Optimal Control

Preprint SFB393/98-6

Preprint-Reihe des Chemnitzer SFB 393

SFB393/98-6 **March 1998**

Contents

Author's address:

Thilo Penzl
TU Chemnitz
Fakultät für Mathematik
D-09107 Chemnitz
Germany

Email: `tpenzl@mathematik.tu-chemnitz.de`

A Cyclic Low Rank Smith Method for Large Sparse Lyapunov Equations with Applications in Model Reduction and Optimal Control[*]

Thilo Penzl[†]

March 18, 1998

Abstract

We present a new method for the computation of low rank approximations to the solution of large, sparse, stable Lyapunov equations. It is based on a generalization of the classical Smith method and profits by the usual low rank property of the right hand side matrix. The requirements of the method are moderate with respect to both computational cost and memory. Hence, it provides a possibility to tackle large scale control problems. Besides the efficient solution of the matrix equation itself, a thorough integration of the method into several control algorithms can improve their performance to a high degree. This is demonstrated for algorithms for model reduction and optimal control. Furthermore, we propose a heuristic for determining a set of suboptimal ADI shift parameters. This heuristic, which is based on a pair of Arnoldi processes, does not require any a priori knowledge on the spectrum of the coefficient matrix of the Lyapunov equation. Numerical experiments show the efficiency of the iterative scheme combined with the heuristic for the ADI parameters.

Key Words: ADI iteration, Smith method, iterative methods, Lyapunov equation, matrix equation, model reduction, balanced truncation, optimal control, Riccati equation, Newton method.

AMS Subject Classification: 65F30, 65F10, 15A24, 93C05.

1 Introduction

The Lyapunov matrix equation plays an important role in control theory. For example, it arises in stability analysis [LL61], the solution of Riccati matrix equations [Kle68], model

[*] This work was supported by the Deutsche Forschungsgemeinschaft within the Sonderforschungsbereich 393 “Numerische Simulation auf massiv parallelen Rechnern”.

[†] Fakultät für Mathematik, Technische Universität Chemnitz, 09107 Chemnitz, Germany. E-mail: tpenzl@mathematik.tu-chemnitz.de

reduction [Moo81, SC89], and H_∞ optimal control [Fra87]. In this paper, we consider the Lyapunov equation

$$A^T X + XA = -BB^T, \tag{1}$$

where the matrix $A \in \mathbb{R}^{n,n}$ is stable, i.e., its eigenvalues are contained in $\mathbb{C}_-$, which denotes the set of the complex numbers with negative real parts. Under this assumption a unique solution $X \in \mathbb{R}^{n,n}$ exists, which is symmetric and positive semidefinite, e.g., [LT85]. There are a number of direct methods for solving the Lyapunov equation (1) numerically, the most important of which are the Bartels-Stewart method [BS72] and the Hammarling method [Ham82]. Unfortunately, these methods, which are based on the QR algorithm, ignore any sparsity in the equation and are not very attractive for parallelization. A rough estimation of the complexity of the Bartels-Stewart method and the Hammarling method gives about $25n^3$ flops and $3n^2$ words of memory. Note that we count one flop as a single floating point operation according to [GL96]. Although these methods should be considered as standard methods for small, dense Lyapunov equations, their use is very limited when large, sparse equations have to be solved. For example, dynamical systems arising from the discretization of parabolic differential equations lead to large, sparse Lyapunov equations, e.g., [RW95]. It is important to note that the number of columns of the matrix $B \in \mathbb{R}^{n,m}$, which is related to the number of inputs and outputs of the underlying dynamical system, is usually very small and does not depend on the fineness of the discretization. This fact is of importance for the method presented in Section 3.

If large, sparse problems have to be solved, iterative schemes are often the method of choice because they do not destroy sparsity. Mostly, they are much more suitable for parallelization than direct methods. In the sequel, we briefly review two popular iterative methods for Lyapunov equations which do not benefit from the low rank property of the right hand side.

The iterates X_i^A of the *alternating direction implicit iteration* (ADI) [PR55, Wac88b] are usually generated by the solution of two linear systems with multiple right hand sides

$$\begin{aligned} (A^T + p_i I) X_{i-1/2}^A &= -BB^T - X_{i-1}^A (A - p_i I) \\ (A^T + p_i I) X_i^{A\,T} &= -BB^T - {X_{i-1/2}^A}^T (A - p_i I), \end{aligned} \tag{2}$$

where $X_0^A = 0$ and the shift parameters $p_1, p_2, p_3, \ldots$ are elements of $\mathbb{C}_-$. This pair of equations is mathematically equivalent to the iteration step

$$X_i^A = (A^T - p_i I)(A^T + p_i I)^{-1} X_{i-1}^A (A - p_i I)(A + p_i I)^{-1} - 2p_i (A^T + p_i I)^{-1} BB^T (A + p_i I)^{-1}. \tag{3}$$

The error matrices $D_i = X - X_i^A$ obey the recursion

$$D_l = \left(r_l(A) r_l(-A)^{-1} \right)^T D_0 r_l(A) r_l(-A)^{-1}, \tag{4}$$

where r_l is the polynomial $r_l(t) = (t - p_1) \cdot \ldots \cdot (t - p_l)$. The rate of convergence is dominated by the spectral radius of the error transfer operator given by (4). The minimization of this spectral radius with respect to the shift parameters $p_1, \ldots, p_l$ leads to the ADI minimax problem

$$\{p_1, \ldots, p_l\} = \underset{\{p_1,\ldots,p_l\}\subset\mathbb{C}_-}{\operatorname{argmin}} \; \max_{t\in\sigma(A)} \frac{|r_l(t)|}{|r_l(-t)|}, \tag{5}$$

which delivers criteria for the optimal and suboptimal choice of these parameters. Here $\sigma(A)$ denotes the spectrum of A. The minimax problem (5) is solved for equations with symmetric matrices A, e.g., [Wac63]. Unfortunately, there is still a lack in theory for the general case, where the eigenvalues of A are possibly not real. Contributions to the solution of the complex ADI minimax problem can be found in [Bag69, CR96, EW91, LR93, Sta91, Sta92, Sta93, Wac88a, Wac90], for example.

The *Smith method* [Smi68] is derived from the Stein equation

$$X - S^T X S = T \tag{6}$$

with

$$S = (A - pI)(A + pI)^{-1} \qquad \text{and} \qquad T = -2p(A + pI)^{-T} BB^T (A + pI)^{-1}, \tag{7}$$

which is equivalent to (1) for any real $p < 0$. Under this assumption the sequence $\{X_i^S\}_{i=0}^{\infty}$ generated by

$$X_0^S = 0, \qquad X_{i+1}^S = T + S^T X_i^S S \tag{8}$$

converges to the solution X and the iterates can be written as

$$X_i^S = \sum_{j=1}^{i} \left(S^{j-1}\right)^T T S^{j-1}. \tag{9}$$

The Smith method can be shown to be mathematically equivalent to ADI when $p = p_1 = p_2 = \ldots$, but in general it converges much slower than ADI with nonconstant shift parameters. Nevertheless, it has become quite popular since there exists an accelerated version, the so-called *squared Smith method.* This version is based on the recursion

$$X_0^S = 0, \qquad X_{2^0}^S = T, \qquad X_{2^{j+1}}^S = X_{2^j}^S + \left(S^{2^j}\right)^T X_{2^j}^S S^{2^j},$$

which is derived from (9). Thus, the subsequence $\left\{X_{2^j}^S\right\}_{j=0}^{\infty}$ can be obtained with linear computational cost. Despite the quadratic rate of convergence, one should be reluctant to apply the squared method to large, sparse equations. The matrices S^{2^j}, which have to be squared explicitly in each step of the iteration, are dense even if A is sparse.

If the structure of the matrix A enables the efficient solution of linear systems $(A^T + pI)x = y$, e.g., when A is a banded matrix, both ADI and the standard Smith method should be considered as sparse matrix techniques. However, since the iterates are formed explicitly in both methods, neither method can be applied when the dimension n becomes so large that dense n-$\times$-n matrices cannot be stored in memory. There are only a few iterative methods which can really be used to solve very large, sparse Lyapunov equations, e.g., [GL94, HP88, HPT96, HR92, JK94, Saa90]. These methods produce low rank approximations to the solution X. The iterates are stored implicitly in factored form, which decreases the memory requirement significantly. However, these methods often fail to determine approximate solutions of high accuracy. Moreover, the rank of their approximations tends to be relatively large. This is indicated by the numerical experiments reported in the original references as well as in our own experiments with one of the classical low rank methods introduced in the sequel.

The method we refer to as *full orthogonalization method* for Lyapunov equations (FOM-L) [HR92, JK94, Saa90] could be considered as an extension of FOM for systems of linear equations [Saa81] to matrix equations. Note that this method is frequently called Arnoldi method or Galerkin method. FOM-L is based on the Arnoldi process (if $m = 1$) or the block Arnoldi process (if $m > 1$) applied to the matrices A^T and B. The purpose of this process is to establish an orthonormal basis $V_k \in \mathbb{R}^{n,w}$ ($w \leq mk$) in the *Krylov subspace*

$$\mathcal{K}_k(A^T, B) = \text{range}\left(\left[\begin{array}{ccccc} B & A^T B & \left(A^T\right)^2 B & \ldots & \left(A^T\right)^{k-1} B \end{array}\right]\right).$$

For details of the Arnoldi process or its block version see, e.g., [Arn51, GL96, Wil65]. The FOM-L iterates X_k^F defined by

$$X_k^F = V_k \tilde{X}_k V_k^T$$

are required to fulfil the Galerkin condition

$$V_k^T \left(A^T X_k^F + X_k^F A + BB^T\right) V_k = 0.$$

Hence, $\tilde{X}_k \in \mathbb{R}^{w,w}$ is given by the solution of the Lyapunov equation

$$V_k^T A^T V_k \tilde{X}_k + \tilde{X}_k V_k^T A V_k = -V_k^T B B^T V_k. \qquad (10)$$

If the symmetric part of A is negative definite, it can be shown by Bendixon's theorem (e.g., [MM92]) that $V_k^T A V_k$ is stable. Under this assumption the Lyapunov equation (10) has a unique solution and the matrices $\tilde{X}_k$ and X_k^F are symmetric, positive semidefinite. If w is much smaller than n, this equation can be solved by direct standard methods. The problem with FOM-L is that it converges rather slowly in many cases, i.e., relatively large values of k and w are necessary to attain a quite accurate approximate solution. This in turn may cause problems because the dense n-$\times$-w matrix V_k has to be stored in memory.

The remainder of this paper is organized as follows. In Section 2 we introduce the cyclic Smith method, which is a fast converging generalization of the Smith method. It is related to the ADI iteration with cyclic shift parameters and yields a sequence of full rank iterates. Low rank versions of ADI and the cyclic Smith method are proposed in Section 3. For each of these methods a set of suboptimal ADI parameters is needed. In Section 4 we propose a heuristic procedure for determining such parameters, which does not require any a priori knowledge of the spectrum of A. The cyclic low rank Smith method is not only an attractive means for solving large Lyapunov equations. It also enables us to improve the efficiency of some "outer" control algorithms. This is illustrated in Sections 5 and 6 by example algorithms for model reduction and optimal control. Numerical tests in Section 7 demonstrate the efficiency of the cyclic low rank Smith method combined with the heuristic procedure for determining ADI shift parameters. Conclusions are provided in Section 8.

2 Smith(l) – a generalization of the Smith method

In this section we study the special case of the ADI iteration where l different shift parameters are applied in a cyclic manner. In other words, we require $p_{i+jl} = p_i$ for $j = 1, 2, \ldots$ in (2). The practical importance of this special case is illustrated by an experiment with the following medium scale example.

Example 1 [HPT96] This example describes the boundary control of the heat flow in a thin rod. The discretization of the underlying parabolic differential equation by finite differences results in a dynamical system (18). This in turn leads to a Lyapunov equation of order $n = 400$. The matrices $A \in \mathbb{R}^{n,n}$ and $B \in \mathbb{R}^{n,1}$ are defined as

$$A = \begin{bmatrix} -1/h & 1/h & 0 & \cdots & 0 \\ 1/h & -2/h & 1/h & \ddots & \vdots \\ 0 & 1/h & -2/h & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 1/h \\ 0 & \cdots & 0 & 1/h & -2/h \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1/h \end{bmatrix}$$

with $h = 1/(n+1)$.

We investigate the dependence of the convergence speed of ADI on l. To this end we determine the extremal eigenvalues of the symmetric matrix A. After that we compute sets of optimal shift parameters for several values of l by an algorithm due to Wachspress [Wac63, Section 2].

Table 1: ADI applied to Example 1. Numbers of iterations required to attain different relative residual norms depending on the number of different shift parameters.

l	relative residual norm				
	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}
1	851	1368	1903	2448	3001
2	53	85	119	153	187
4	17	29	41	53	65
8	15	23	31	39	49
16	13	21	29	37	41
32	13	21	29	33	41

Table 1 shows the numbers of iterations required to attain different relative residual norms (i.e., $\left\|A^T X_i + X_i A + BB^T\right\|_F / \left\|BB^T\right\|_F \leq tol$ with $tol = 10^{-4}, 10^{-6}, \ldots$). It reveals two facts. First, the convergence is very slow for $l = 1$, which corresponds to the Smith method, but a moderate increase of l (say $l = 4$) accelerates it to a high degree. Second, the speed of convergence is hardly improved by a further increase of l. In fact, this experiment and a number of further tests not reported here indicate that a relatively small number of different ADI parameters is sufficient to achieve rapid convergence of ADI.

Although the speed of convergence of the Smith iteration is often unsatisfactory, this method has two algorithmic advantages over ADI. First, there exists a squared version of the Smith method. Second, the low rank version of the Smith method, which is presented in a generalized form in Section 3, is more efficient with respect to the amount of computation than that of ADI. This leads to the question whether there exists a generalization of the Smith method that is mathematically equivalent to the fast converging ADI with cyclic parameters. Indeed, such a generalization is easily derived from (4). If we extend the original definition of the matrices S and T given in (7) to

$$S = \prod_{j=1}^{l} (A - p_j I)(A + p_j I)^{-1} \qquad \text{and} \qquad T = X_l^A, \tag{11}$$

then (4) is equivalent to the Stein equation $X - S^T X S = T$, which has the same structure as (6). This Stein equation is the base for the generalized version of the Smith iteration we refer to as *cyclic Smith method* (Smith(l)). The only essential difference between the standard and the cyclic version is that the matrix T is given explicitly in the first case, whereas it is the result of l steps of the ADI iteration with shift parameters $p_1, \ldots, p_l$ applied to (1) in the second. Analogous to (8), the Smith(l) iterates are generated by the recursion

$$X_0 = 0, \qquad X_{(i+1)l} = T + S^T X_{il} S. \tag{12}$$

For consistency we label these iterates by multiples of l. Note that, in contrast to ADI, standard Smith method, and FOM-L, the iterates of Smith(l) are not provided with an extra superscript. Using (4) and (11), it is easy to prove that $X_{il} = X_{il}^A$ actually holds for $i = 1, 2, \ldots$, if the ADI iterates X_{il}^A are generated by use of l-cyclic parameters. As a consequence, Smith(1) is identical with the classical Smith method. In fact, the implementation (12) of Smith(l) in not more favorable than that of the ADI iteration (2) if sparse Lyapunov equations are to be solved. It should rather be considered as a preliminary step for deriving the low rank method LR-Smith(l) presented in the next section.

The squared version of Smith(l), which is derived analogously to that of the standard Smith method, is not considered here because it involves dense n-×-n matrices in the computation. However, such a version may be of interest for large, dense Lyapunov equations. For $l = 2$ such a generalization of the squared Smith method has been proposed by Davison and Man [DM68].

3 LR-ADI and LR-Smith(l) – low rank versions of ADI and Smith(l)

For the remaining part of this paper we assume $m \ll n$. This enables us to establish a low rank version of the ADI we will refer to as LR-ADI. This iterative method again is needed to construct the cyclic low rank Smith method. The key idea is to substitute the ADI and Smith(l) iterates by products

$$X_i^A = Z_i^A Z_i^{A^T} \qquad \text{and} \qquad X_{il} = Z_{il} Z_{il}^T, \tag{13}$$

respectively. This is always possible because the iterates X_i^A and X_{il} can be shown recursively to be symmetric and positive semidefinite. Although a similar approach is pursued in several methods for Lyapunov equations, e.g., [AL93, BQO97, Ham82, HR92, JK94, Saa90], this has not been done in combination with ADI or Smith-like methods.

LR-ADI is based on the ADI single sweep (3). Using (13) this formula can be rewritten in terms of the matrices Z_i^A as

$$Z_i^A = \left[\begin{array}{cc} (A^T - p_i I)(A^T + p_i I)^{-1} Z_{i-1}^A & \sqrt{-2p_i}(A^T + p_i I)^{-1} B \end{array}\right] \tag{14}$$

with

$$Z_1^A = \sqrt{-2p_1}(A^T + p_1 I)^{-1} B.$$

The number of columns in the matrices Z_i^A is enlarged by m in each step of the iteration and $\operatorname{rank}(X_i^A) \leq mi$. Although the memory requirement and the computational cost per iteration are linearly raising, LR-ADI poses an efficient iterative scheme for solving large, sparse Lyapunov equations since, in general, the number of ADI iterations is much smaller than the dimension of the problem. In particular, LR-ADI is of interest if a sequence $\{p_i\}_{i=1}^{\infty}$ of different shift parameters (e.g., [Bag69, Sta91, Sta93]) is available.

If the number of different ADI parameters is limited, the *cyclic low rank Smith method* (LR-Smith(l)) poses a more efficient alternative to LR-ADI. The algorithm LR-Smith(l) consists of two stages. First, the l-th iterate Z_l^A of the LR-ADI method with the shift parameters $p_1, \ldots, p_l$ is computed. Analogous to (11), this matrix is used for the initialization

$$\begin{aligned} Z^{(l)} &= Z_l^A \\ Z_l &= Z^{(l)}. \end{aligned}$$

Second, the actual LR-Smith(l) iteration is performed by

$$\begin{aligned} Z^{((i+1)l)} &= S^T Z^{(il)} \\ Z_{(i+1)l} &= \left[\begin{array}{cc} Z_{il} & Z^{((i+1)l)} \end{array}\right], \end{aligned} \tag{15}$$

where S is given by (11). Note that the computational cost per iteration step (15) is constant, which is an important advantage of LR-Smith(l) over LR-ADI. It is straightforward to prove that LR-Smith(l) is linked to Smith(l) by (13). Moreover, LR-Smith(l) and

ADI are mathematically equivalent in the sense of $Z_{il}Z_{il}^T = X_{il}^A$, if the shift parameters $p_1, \ldots, p_l$ are used cyclically in the ADI iteration. However, determining Z_{il} by LR-Smith(l) is generally much more efficient than computing X_{il} by ADI if n is large and m is small.

Concerning the implementation of the low rank methods, a few remarks should be made. Neither the matrices $(A^T + p_i I)^{-1}$ in LR-ADI nor the matrix S in LR-Smith(l) are formed explicitly. Instead, the sparse matrices $A^T + p_i I$ are factorized a priori (e.g., by LU factorizations) and the iterations (14) and (15) involve forward and backward substitutions. Of course, a certain amount of fill-in is generally produced by these factorizations. Nevertheless, this procedure is much more efficient than computing the inverses of $A^T + p_i I$ explicitly when the matrix A is banded. Alternatively, iterative methods can be utilized to solve sparse, linear systems of the type $(A^T + p_i I)x = y$, e.g., [Saa96].

Theoretically, a squared version of LR-Smith(l) can be derived as well. It requires to form the dense matrix S explicitly. Hence, such a method should not be applied to large, sparse Lyapunov equations since its memory requirement is $\mathcal{O}(n^2)$ and the computational cost is $\mathcal{O}(n^3)$ even if A is sparse.

In some algorithms in control theory only the product of X with a matrix V containing a few columns is sought instead of the solution matrix X itself. In this case LR-Smith(l) can be very efficient with respect to the memory requirement because the iterates Z_{il} need not be stored. The product XV can be evaluated by accumulating the sum on the right hand side of

$$XV = \lim_{i \to \infty} Z_{il}Z_{il}^T V = \sum_{i=1}^{\infty} Z^{(il)} \left(Z^{(il)^T} V \right) \tag{16}$$

in the course of the iteration (15). Such a procedure has been proposed in [Saa90, Section 3] in a similar context.

In general, it is not known a priori how many LR-Smith(l) steps are necessary to attain a prescribed accuracy of the approximate solution. Therefore, it is necessary to compute repeatedly the Frobenius norm of the current residual matrix. Unfortunately, this matrix cannot be formed explicitly if the dimension of the problem is large. In this case it is advisable to compute the norm of the residual by

$$\begin{aligned} \left\| A^T Z_{il} Z_{il}^T + Z_{il} Z_{il}^T A + BB^T \right\|_F &= \left\| \begin{bmatrix} A^T Z_{il} & Z_{il} & B \end{bmatrix} \begin{bmatrix} Z_{il} & A^T Z_{il} & B \end{bmatrix}^T \right\|_F \\ &= \left\| R_1 R_2^T \right\|_F, \end{aligned} \tag{17}$$

where R_1 and R_2 are the square, upper, triangular matrices resulting from the "economy size" QR decompositions $Q_1 R_1 = \begin{bmatrix} A^T Z_{il} & Z_{il} & B \end{bmatrix}$ and $Q_2 R_2 = \begin{bmatrix} Z_{il} & A^T Z_{il} & B \end{bmatrix}$.

There are a number of approaches to a parallelization of LR-Smith(l). The factorization of the matrices $A^T + p_i I$ can be realized efficiently in parallel using l processors. Moreover, in (15) the products of S^T with the single columns of $Z^{(il)}$ can be computed simultaneously. The computational cost for realizing one step of (15) is proportional to m. If $m > 1$, the right hand side matrix of the Lyapunov equation (1) can be split up as

$$-BB^T = -\sum_{j=1}^{m} b_j b_j^T$$

with $B = \left[\begin{array}{ccc} b_1 & \ldots & b_m \end{array} \right]$, which has been proposed in [HR92, Section 5] in a similar fashion. For a parallel computer with m processors this offers an ideal parallelization, because the resulting m Lyapunov equations with right hand side matrices of rank 1 can be solved simultaneously.

4 A heuristic procedure for determining suboptimal ADI shift parameters

The performance of the ADI-based methods described in the previous sections depends strongly on the choice of the shift parameters p_i. The conventional approach to the computation of these parameters is to cover the spectrum of A by a domain $\Omega \subset \mathbb{C}_-$ and to solve the ADI minimax problem (5) with respect to Ω instead of $\sigma(A)$. For a few shapes of the domain Ω (e.g., intervals, rectangles, circles, trapezoids) optimal or at least suboptimal shift parameters have been found, e.g., [Sta91, Wac66, Wac90]. Moreover, several procedures for constructing sequences of suboptimal ADI parameters for more general domains have been proposed in [Bag69, Sta91, Sta93, CR96]. However, all these approaches, which are based on approximation theory, require knowledge of certain bounds of the spectrum. In rare cases such bounds can be computed analytically, but mostly one has to determine bounds a priori by numerical methods. If A is a symmetric, banded matrix it is reasonable to compute the spectrum by the QR method. In the unsymmetric case the most simple approach might be to estimate the extreme eigenvalues of the symmetric and the skew-symmetric part of A by power iteration or inverse iteration, e.g., [GL96], which deliver bounds for the spectrum of A by Bendixon's theorem, e.g., [MM92]. Of course, this procedure cannot be applied if the symmetric part of A is indefinite, since the rectangular obtained in this way is not a subset of $\mathbb{C}_-$. Alternatively, estimates for the eigenvalues of A can be obtained by the Arnoldi process. However, this method can fail in the indefinite case, too, because it may deliver estimates with nonnegative real parts.

In this section, we propose a procedure for determining a set $\mathcal{P}$ of l different suboptimal ADI shift parameters without first finding a superset Ω of the spectrum. The resulting algorithm is easy to implement. Although it relies more on heuristics than on approximation theory, the numerical results are quite satisfactory. Our algorithm does not require any a priori knowledge of the spectrum of A. All information about this matrix is obtained by a pair of Arnoldi processes related to the matrix A itself and its inverse. We choose the initial vector r of these processes at random. The integers k_+ and k_- denote the numbers of Arnoldi steps in the processes for the matrices A and A^{-1}, respectively. Writing the result of $k = k_+$ Arnoldi steps w.r.t. the pair (A, r) as a matrix equation, we get (e.g., [GL96])

$$AV_k = V_{k+1}\tilde{H}_k$$

with $V_k \in \mathbb{R}^{n,k}$, $\tilde{H}_k \in \mathbb{R}^{k+1,k}$, $V_1 \in \operatorname{span}\{r\}$, $V_{k+1}^T V_{k+1} = I_{k+1}$, $(V_{k+1})_{(1:n,1:k)} = V_k$. Moreover,

$$H_k := (\tilde{H}_k)_{(1:k,1:k)} = V_k^T A V_k$$

is an upper Hessenberg matrix. This matrix and its eigenvalues are called *Ritz matrix* and *Ritz values*, respectively. It is well known that the set $\mathcal{R}_+ := \sigma(H_k)$ represents an approximation of the spectrum of A [Arn51]. Repeating this procedure with the inverse of A delivers the set $\mathcal{R}_-$, the elements of which approximate the eigenvalues of A^{-1}. Consequently, the set $\mathcal{R} := \mathcal{R}_+ \cup 1/\mathcal{R}_-$ can be considered as a guess of the spectrum of A. The Ritz values obtained by the Arnoldi process tend to be located near the "outer" eigenvalues, i.e., the eigenvalues near the convex hull of the spectrum. In particular, eigenvalues of large magnitude are usually approximated well. In contrast, the elements of $\mathcal{R}_+$ are generally poor approximations to the eigenvalues near the origin. Therefore, we involve the set $1/\mathcal{R}_-$ to approximate these eigenvalues. In fact, this procedure can have a drastic impact on the speed of convergence of the iteration, which is shown by an example at the end of this section.

The key idea of our heuristic procedure is to replace $\sigma(A)$ by $\mathcal{R}$ in (5), provided that $\mathcal{R} \subset \mathbb{C}_-$. Moreover, we choose the suboptimal ADI parameters $\mathcal{P} := \{p_1, \ldots, p_l\}$ among the elements of $\mathcal{R}$ because the function

$$s_{\mathcal{P}}(t) = \frac{|r_l(t)|}{|r_l(-t)|} = \frac{|(t-p_1)\cdot \ldots \cdot (t-p_l)|}{|(t+p_1)\cdot \ldots \cdot (t+p_l)|}$$

becomes small over $\sigma(A)$ if there is one of the shifts p_i in the neighbourhood of each eigenvalue. Since the Lyapunov equations to be solved are real, we require $\mathcal{P} = \bar{\mathcal{P}}$. This ensures the approximations $Z_{il}^A {Z_{il}^A}^T$ and $Z_{il}{Z_{il}}^T$ to be real as well.

Based on these considerations we determine the elements of $\mathcal{P}$ as follows. Firstly, we detect the element $\rho_i \in \mathcal{R}$ which minimizes the function $s_{\{\rho_i\}}$ over $\mathcal{R}$. The set $\mathcal{P}$ is initialized by either $\{\rho_i\}$ or $\{\rho_i, \bar{\rho}_i\}$. Afterwards, we successively augment the set $\mathcal{P}$ by the elements or pairs of elements of $\mathcal{R}$, for which the maximum of $s_{\mathcal{P}}$ is attained. In other words, the maximum of $s_{\mathcal{P}}$ with respect to the current set $\mathcal{P}$ is replaced by a zero in the refined function $s_{\mathcal{P}}$. This strategy is summarized in the following algorithm. Note that the heuristic applied in Step 7 is related to an algorithm by Bagby [Bag69]. The notation card$(\mathcal{P})$ is used for the number of elements contained in the set $\mathcal{P}$.

Algorithm 1 (Suboptimal ADI parameters)

```
INPUT: $A$, $l_0$, $k_+$, $k_-$

OUTPUT: $\mathcal{P}$

1. Choose $r \in \mathbb{R}^n$ at random.

2. Perform $k_+$ steps of the Arnoldi process w.r.t. $(A, r)$ and compute the set of Ritz values $\mathcal{R}_+$.

3. Perform $k_-$ steps of the Arnoldi process w.r.t. $(A^{-1}, r)$ and compute the set of Ritz values $\mathcal{R}_-$.

4. $\mathcal{R} = \{\rho_1, \ldots, \rho_{k_+ + k_-}\} := \mathcal{R}_+ \cup (1/\mathcal{R}_-)$

5. IF $\mathcal{R} \not\subset \mathbb{C}_-$, STOP
```

```
6. Detect $i$ with $\max_{t \in \mathcal{R}} s_{\{\rho_i\}}(t) = \min_{\rho \in \mathcal{R}} \max_{t \in \mathcal{R}} s_{\{\rho\}}(t)$
and initialize $\mathcal{P} := \begin{cases} \{\rho_i\} & : \ \rho_i \text{ real} \\ \{\rho_i, \bar{\rho}_i\} & : \ \text{otherwise} \end{cases}$ .
WHILE card($\mathcal{P}$) $< l_0$
    7. Detect $i$ with $s_{\mathcal{P}}(\rho_i) = \max_{t \in \mathcal{R}} s_{\mathcal{P}}(t)$ and set $\mathcal{P} := \begin{cases} \mathcal{P} \cup \{\rho_i\} & : \ \rho_i \text{ real} \\ \mathcal{P} \cup \{\rho_i, \bar{\rho}_i\} & : \ \text{otherwise} \end{cases}$ .
END WHILE
```

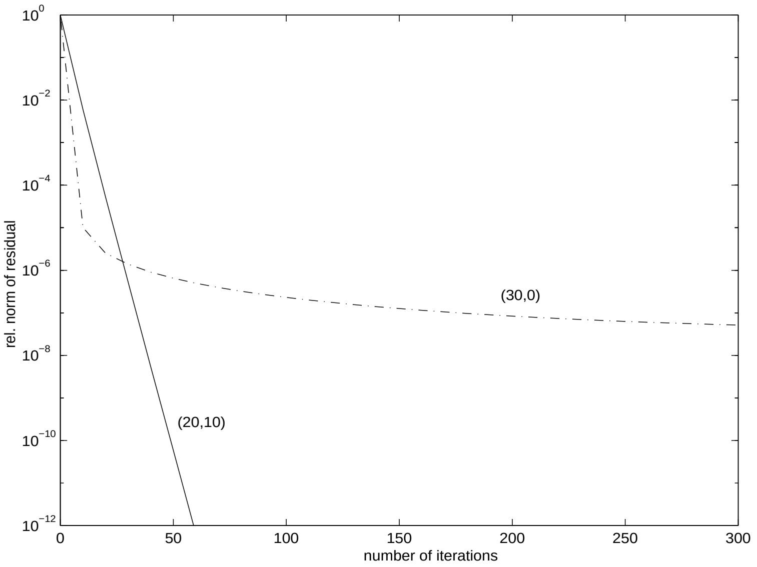
Step 5 can be omitted if $A + A^T$ is negative definite. This enables us to prove that $A^{-1} + A^{-T}$ as well as the symmetric parts of the Ritz matrices w.r.t. A and A^{-1} are negative definite, too. Otherwise, this algorithm can fail, although this has never been observed in our numerical experiments. However, it is possible to construct a starting vector r which makes it fail in the indefinite case.

If Algorithm 1 stops at Step 5, it can be restarted with a new random vector r or the values of k_+ or k_- can be increased. The latter is motivated by the observation that Ritz values obtained by the Arnoldi process tend to approximate the spectrum of a matrix better if the number of Arnoldi steps is enlarged. More sophisticated approaches may involve implicit restart techniques [Sor92] to purge the sets $\mathcal{R}_+$ and $\mathcal{R}_-$ of elements with nonnegative real parts.

In the remainder of this section we show the importance of employing the Ritz values w.r.t. A^{-1} in Algorithm 1. In Figure 1 we compare two runs of LR-Smith(10) applied to Example 1 given in Section 2 with different sets of ADI shift parameters. For the first run we include the Ritz values w.r.t. A^{-1} in the computation of the shift parameters by Algorithm 1. More precisely, we choose $(k_+, k_-) = (20, 10)$. In contrast, we set $(k_+, k_-) = (30, 0)$ for determining the parameter set $\mathcal{P}$ for the second run. In either case $\mathcal{P}$ consists of $l = l_0 = 10$ elements.

Figure 1 shows that the convergence of LR-Smith(10) is fast and linear for the set $\mathcal{P}$ obtained by use of the parameters $(k_+, k_-) = (20, 10)$. In contrast, for $(k_+, k_-) = (30, 0)$ the convergence is very fast in the first stage, but in the second stage it almost stagnates. This phenomenon can be explained as follows. There are a few eigenvalues which are poorly approximated by the set $\mathcal{R}$. As a consequence, the function $s_{\mathcal{P}}(t)$ delivered by Algorithm 1 is almost 1 if t is equal to one of these eigenvalues, but it is relatively small if t belongs to the majority of eigenvalues which are approximated well. Thus, the component of the residual related to the latter sort of eigenvalues is quickly damped in the first stage of the iteration, whereas the iteration is delayed in the second stage by a small number of eigenvalues approximated poorly by $\mathcal{R}_+$. These eigenvalues, which are typically of small magnitute, are usually represented well by elements of $1/\mathcal{R}_-$.

Figure 1: Example 1. Convergence of the LR-Smith(10) iteration with two different sets of shift parameters. These sets are determined by Algorithm 1 with $(k_+, k_-) = (20, 10)$ and $(k_+, k_-) = (30, 0)$.



5 Applications I: model reduction

Lyapunov equations have to be solved in a number of control algorithms for the dynamical system

$$\begin{aligned} \dot{x}(\tau) &= Ax(\tau) + Bu(\tau) \\ x(0) &= x_0 \\ y(\tau) &= Cx(\tau), \end{aligned} \tag{18}$$

which is described by the matrix triplet (A, B, C) with $A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$, and $C \in \mathbb{R}^{q,n}$. The entries of u, x, and y are called *input*, *state*, and *output*, respectively. If m and q are small and A is large, sparse, and stable, a thorough integration of LR-Smith(l) in these control algorithms can improve their overall complexity by a high degree. This is demonstrated for an established model reduction method.

The purpose of model reduction is to replace the dynamical system (A, B, C) by a reduced system $(\hat{A}, \hat{B}, \hat{C})$ with $\hat{A} \in \mathbb{R}^{k,k}$, $\hat{B} \in \mathbb{R}^{k,m}$, $\hat{C} \in \mathbb{R}^{q,k}$, $k < n$, such that the input-output behavior of the reduced system approximates that of the original system in some sense. Here, we consider the implementation of the balanced truncation technique proposed by Safonov and Chiang [SC89], which requires to compute the *reachability Gramian* X_B and the *observability Gramian* X_C. See also [Moo81, Glo84] for details. The matrices X_B and X_C solve the Lyapunov equations

$$AX_B + X_B A^T = -BB^T \tag{19}$$

$$A^T X_C + X_C A \;=\; -C^T C \tag{20}$$

and it can be shown that the eigenvalues of the product $X_B X_C$ are real and nonnegative [Moo81]. The standard implementation of the method is sketched in the following algorithm.

Algorithm 2 (Sketch of the model reduction method by Safonov and Chiang)

INPUT: A, B, C, k

OUTPUT: $\hat{A}$, $\hat{B}$, $\hat{C}$

1. Solve (19) and (20).

2. Determine the k largest eigenvalues of $X_B X_C$ and compute orthonormal bases $V_B, V_C \in \mathbb{R}^{n,k}$ of the corresponding right and left, invariant subspaces, respectively, by means of ordered Schur forms.

3. $U_C \Sigma U_B^T := V_C^T V_B$ (singular value decomposition)

4. $S_C := V_C U_C \Sigma^{-1/2}$, $\quad S_B := V_B U_B \Sigma^{-1/2}$

5. $\hat{A} := S_C^T A S_B$, $\quad \hat{B} := S_C^T B$, $\quad \hat{C} := C S_B$

Assuming $k \ll n$ it is obvious that the major part of the computational cost and the memory requirement are related to the first two steps. If m and q are very small, which is actually the case in many applications, the Lyapunov equations in Step 1 can be solved efficiently by LR-Smith(l). In general, this method produces quite accurate approximate solutions, the rank of which is much less then the dimension n, provided that n is large and the ADI shift parameters are chosen properly. Note for instance the results of the numerical experiments with Example 2 given by Table 2 in Section 7. Here the solution matrix X_B of order $n = 10\,000$ is approximated very accurately by a matrix $Z_B Z_B^T$ which is only of rank $r_B = 100$! What follows is a proposal for an implementation of Step 2 which strongly benefits from the low rank property of the solution matrices.

Suppose that the application of LR-Smith(l) in Step 1 delivers a pair of (approximate) solutions in factored form as $X_B = Z_B Z_B^T$ and $X_C = Z_C Z_C^T$, where $Z_B \in \mathbb{R}^{n,r_B}$, $Z_C \in \mathbb{R}^{n,r_C}$, and $\max\{r_B, r_C\} \ll n$. Taking the low rank structure into account, we determine a singular value decomposition (SVD) of the product $X_B X_C$. For this reason, we first compute "economy size" QR factorizations $Z_B =: Q_{B1} R_B$ and $Z_C =: Q_{C1} R_C$ with $Q_{B1} \in \mathbb{R}^{n,r_B}$ and $Q_{C1} \in \mathbb{R}^{n,r_C}$. After that an "economy size" SVD

$$R_B Z_B^T Z_C R_C^T =: Q_{B2} D Q_{C2}^T$$

with the nonsingular diagonal matrix $D \in \mathbb{R}^{r,r}$ and $r \leq \min\{r_B, r_C\} \ll n$ is computed. Defining $Q_B := Q_{B1} Q_{B2}$ and $Q_C := Q_{C1} Q_{C2}$, we finally get the desired "economy size" SVD of $X_B X_C$ by

$$X_B X_C = Z_B Z_B^T Z_C Z_C^T = Q_{B1} R_B Z_B^T Z_C R_C^T Q_{C1}^T = Q_B D Q_C^T. \tag{21}$$

By means of this equation we will now compute an orthogonal basis for the right, dominant, invariant subspace of $X_B X_C$. Obviously, the right, invariant subspace related to the nonzero eigenvalues of $X_B X_C$ coincides with the range of Q_B. Because of

$$X_B X_C Q_B = Z_B Z_B^T Z_C Z_C^T Q_B = Q_B D Q_C^T Q_B \tag{22}$$

all nonzero eigenvalues of $X_B X_C$ are eigenvalues of the matrix $DQ_C^T Q_B$ as well. If $r \ll n$, the distinct merit of our approach is that we have to determine the dominant eigenvalues of the r-×-r matrix $DQ_C^T Q_B$ instead of those of the n-×-n matrix $X_B X_C$ itself. More precisely, we compute an ordered Schur factorization

$$DQ_C^T Q_B =: PMP^T = \begin{bmatrix} P_1 & P_2 \end{bmatrix} \begin{bmatrix} M_{11} & M_{12} \\ 0 & M_{22} \end{bmatrix} \begin{bmatrix} P_1 & P_2 \end{bmatrix}^T, \tag{23}$$

where the block $M_{11} \in \mathbb{R}^{k,k}$ ($k \le r$) corresponds to the k largest eigenvalues of M. Thus, the desired orthonormal basis in the right, dominant, invariant subspace is formed by the columns of the matrix $V_B := Q_B P_1$ since

$$X_B X_C V_B = Z_B Z_B^T Z_C Z_C^T Q_B P_1 = Q_B D Q_C^T Q_B P_1 = Q_B P_1 M_{11} = V_B M_{11},$$

which is a consequence of (22) and (23). An orthonormal basis in the left, dominant, invariant subspace of $X_B X_C$ is obtained by an analogous procedure.

This realization of Step 2 requires $\mathcal{O}(n \max\{r_B^2, r_C^2\})$ flops and $\mathcal{O}(n \max\{r_B, r_C\})$ words of memory, whereas the conventional implementation has a complexity of $\mathcal{O}(n^3)$ flops and $\mathcal{O}(n^2)$ words of memory. Thus, if $\max\{r_B, r_C\} \ll n$, we expect a considerable gain in efficiency.

6 Applications II: optimal control

In this section we consider the *linear-quadratic optimal control problem* where the cost functional

$$\mathcal{J}(x_0, u) = \frac{1}{2} \int_0^\infty y(\tau)^T Q y(\tau) + u(\tau)^T R u(\tau) d\tau \tag{24}$$

with

$$Q = Q^T \ge 0 \qquad \text{and} \qquad R = R^T > 0 \tag{25}$$

is to be minimized and the dynamics (18) represents the constraints. The solution of this problem is determined by the linear feedback

$$u(\tau) = -R^{-1} B^T P x(\tau) =: -K^T x(\tau), \tag{26}$$

where P is the symmetric, positive semidefinite, stabilizing solution of the *algebraic Riccati equation* (ARE)

$$C^T Q C + A^T P + PA - PBR^{-1} B^T P = 0, \tag{27}$$

e.g., [Meh91, Sim96]. A solution of this ARE is called *stabilizing* iff each eigenvalue of the matrix $A - BR^{-1}B^TP$ has a negative real part. Under moderate assumptions a unique stabilizing solution of (27) exists, e.g., [LR95].

The Newton method proposed by Kleinman [Kle68] is one of the standard methods for computing the stabilizing solution of the ARE (27).

Algorithm 3 (Newton method for the ARE (27))

```
INPUT: $A$, $B$, $C$, $Q$, $R$, $P_0$ (e.g., $P_0 = 0$, if $A$ is stable)
OUTPUT: $P$
FOR $k = 1, 2, \dots$
    1. $K_{k-1} := P_{k-1}^T BR^{-1}$
    2. Solve  $(A - BK_{k-1}^T)^T P_k + P_k(A - BK_{k-1}^T) = -C^TQC - K_{k-1}RK_{k-1}^T$  for $P_k$.
END FOR
3. $P := P_\infty$
```

If the initial guess P_0 is stabilizing, which is for instance the case when $P_0 = 0$, Algorithm 3 yields a sequence of stabilizing iterates. The convergence of the Newton method is quadratic, e.g., [LR95].

Due to (25) the matrices Q and R can be factored, e.g., by a Cholesky decomposition, as $Q = \tilde{Q}\tilde{Q}^T$ and $R = \tilde{R}\tilde{R}^T$, where the matrices $\tilde{Q}$ and $\tilde{R}$ have full rank. Thus the Lyapunov equations to be solved in Step 2 have the structure

$$\tilde{A}_k^T X + X\tilde{A}_k = -\tilde{B}_k\tilde{B}_k^T$$

with the stable matrix $\tilde{A}_k = A - BK_{k-1}^T$ and the matrix $\tilde{B}_k = \begin{bmatrix} C^T\tilde{Q} & K_{k-1}\tilde{R} \end{bmatrix}$. Note that $\tilde{B}_k$ contains at most $m + q$ columns. If m and q are very small, these equations can be solved by LR-Smith(l). The Lyapunov solutions pose approximate solutions of the ARE (27). Therefore, the combination of the Newton method and LR-Smith(l) can be utilized to determine low rank solutions of Riccati equations. In fact, this approach enables the solution of a class of large Riccati equations, where the explicit solution matrix is too large to be stored in memory.

A second merit, which is maybe more important in the context of the optimal control problem (18,24), relies on the fact that mostly the feedback matrix K is desired, instead of the Riccati solution P. In the sequel, we propose a modification of Algorithm 3 which computes K without forming factored or explicit Riccati or Lyapunov iterates at all. The basic idea is to generate the matrix K_k itself in Step 2 instead of solving the Lyapunov equation and computing the product $K_k = P_kBR^{-1}$ afterwards. The matrix K_k is formed in the course of the "inner" iteration, i.e., the LR-Smith(l) iteration, as a series of type (16), where V is replaced by BR^{-1}. Note that the partial sum of the first i terms of this series is denoted by $K_{k,i}$ in Algorithm 4. Eventually, the desired matrix K is the limit of the matrices $K_k = K_{k,\infty}$ for $k \to \infty$. The result of this strategy is stated in the following algorithm, which is best understood as a version of Algorithm 3 with an inner loop consisting of interlaced sequences based on (15) and (16).

Algorithm 4 (Computation of the optimal feedback K)

```
INPUT: $A$, $B$, $C$, $Q$, $R$, $K_0$ (such that $A - BK_0^T$ is stable; e.g. $K_0 = 0$), $k_+$, $k_-$, $l_0$
OUTPUT: $K$
FOR $k = 1, 2, \dots$
    1. Determine ADI shifts $p_{k,1}, \dots, p_{k,l}$ with respect to the matrix $\tilde{A}_k = A - BK_{k-1}^T$ by Algorithm 1.
    2. $\tilde{B}_k := \left[ \begin{array}{cc} C^T\tilde{Q} & K_{k-1}\tilde{R} \end{array} \right]$
    3. $Z_k^{(l)} := Z_{k,l}^A$, which denotes the $l$-th iterate of LR-ADI with shifts $p_{k,1}, \dots, p_{k,l}$ applied to $\tilde{A}_k^T X + X\tilde{A}_k = -\tilde{B}_k\tilde{B}_k^T$.
    4. $K_{k,1} := Z_k^{(l)} \left( {Z_k^{(l)}}^T BR^{-1} \right)$
    FOR $i = 2, 3, \dots$
        5. $Z_k^{(il)} := S_k^T Z_k^{((i-1)l)}$   with   $S_k = \prod_{j=1}^{l} (\tilde{A}_k - p_{k,j}I)(\tilde{A}_k + p_{k,j}I)^{-1}$
        6. $K_{k,i} := K_{k,i-1} + Z_k^{(il)} \left( {Z_k^{(il)}}^T BR^{-1} \right)$
    END FOR
    7. $K_k := K_{k,\infty}$
END FOR
8. $K := K_\infty$
```

For the remaining discussion in this section we assume that computing an LU factorization of the sparse matrix A does not produce a large amount of fill-in. In particular, this is guaranteed if A is a banded matrix, which is the case in many applications. Otherwise, it is often possible to transform a given matrix A into a matrix of such shape by several reordering algorithms, e.g., [Cut72]. Under the assumption above, linear systems with coefficient matrices $A^T + p_{k,j}I$ can be solved efficiently. Dense matrices with the structure $A - BK_{k-1}^T$ or $A - BK_{k-1}^T + p_{k,j}I$, which are involved in Steps 1, 3, and 5, are never formed explicitly. Linear systems with these matrices are solved by application of the Sherman-Morrison formula, e.g., [GL96]. Concerning the complexity of Algorithm 4, it should be noted that except for A all matrices involved in the computation consist of a relatively small number of columns. More precisely, the matrices $K_{k,i}$ and K_k contain m columns whereas the matrix $Z^{(il)}$ has at most $l(m+q)$ columns. We do not state an estimate for the memory requirement since this quantity strongly depends on the amount of fill-in produced by the factorizations of the matrices $A + p_{k,j}I$. Likewise, we do not give an estimate for the computational cost, because it is affected by several factors such as the speed of convergence of the inner and outer iterations as well as the choice of the stopping criteria in both loops. Nevertheless, it is obvious that Algorithm 4 is favorable compared to the conventional implementation if n is large and m, q, l are small.

7 Numerical experiments

In this section, we provide four examples of large scale Lyapunov equations. We display the results of numerical experiments with LR-Smith(l) applied to these examples. For comparison we also show the test results obtained by FOM-L. All experiments were carried out using MATLAB 5.1 and IEEE double precision arithmetic (machine precision $\epsilon \approx 2.22{\cdot}10^{-16}$) on an HP9000/800 workstation at the TU Chemnitz, Germany. We characterize the performance of the iterative methods by both the number of iterations and the number of flops required to attain prescribed tolerances for the accuracy, which is measured by the relative Frobenius norm of the residual

$$\frac{\left\| A^T X_i + X_i A + BB^T \right\|_F}{\left\| BB^T \right\|_F},$$

where X_i denotes the i-th iterate of LR-Smith(l) or FOM-L. Note that for LR-Smith(l) each sweep of (15) is counted as l iterations. The residual norm is determined after each of these sweeps. In our implementation the norms of the residual are computed by (17) since the dimensions of the examples are too large to form the residual matrix explicitly. We consider the following examples in our tests.

Example 2 The structure of this example coincides with that of the medium scale Example 1, but here the Lyapunov equation is of order $n = 10\,000$.

Example 3 [HPT96] This example corresponds to a second order model of dimension n_0 which is equivalent to a dynamical system (18) of dimension $n = 2n_0$. The matrices $A \in \mathbb{R}^{n,n}$ and $B \in \mathbb{R}^{n,1}$ are given as

$$A = \begin{bmatrix} 0_{n_0} & I_{n_0} \\ A_{21} & -dI_{n_0} \end{bmatrix} \qquad \text{and} \qquad B = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1/h^2 \end{bmatrix}$$

with

$$A_{21} = \begin{bmatrix} -k/h^2 & k/h^2 & 0 & \cdots & 0 \\ k/h^2 & -2k/h^2 & k/h^2 & \ddots & \vdots \\ 0 & k/h^2 & -2k/h^2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & k/h^2 \\ 0 & \cdots & 0 & k/h^2 & -2k/h^2 \end{bmatrix} \in \mathbb{R}^{n_0,n_0},$$

where $n = 2n_0 = 3\,000$, $h = 1/(n+1)$, $k = 10$, and $d = 1$. For our numerical experiments this system has been reordered by applying the permutation $(n_0+1, 1, n_0+2, 2, \ldots, 2n_0, n_0)$ to the columns and rows of A, such that it becomes a banded matrix with a very small bandwidth.

Example 4 This example describes a model of heat flow with convection in the domain $\Omega = (0,1)^2$. The underlying parabolic differential equation has the structure

$$\dot{\mathbf{x}} = \Delta \mathbf{x} - \mathbf{f_1}(\xi)\frac{\partial \mathbf{x}}{\partial \xi_1} - \mathbf{f_2}(\xi)\frac{\partial \mathbf{x}}{\partial \xi_2} + \mathbf{b}(\xi)u(\tau)$$

with $\mathbf{x} = \mathbf{x}(\xi,\tau)$, $\xi = \begin{bmatrix} \xi_1 & \xi_2 \end{bmatrix}^T \in \Omega$, $\tau \in [0,\infty)$. The coefficient functions in the convection term are defined as $\mathbf{f_1}(\xi) = 10\xi_1$ and $\mathbf{f_2}(\xi) = 1000\xi_2$. The differential equation is discretized by finite differences using a grid with equidistant spacing and 50×50 grid points. The resulting stiffness matrix $A \in \mathbb{R}^{2500,2500}$ is sparse, stable and its bandwidth is 50. The matrix $B \in \mathbb{R}^{2500,1}$ is chosen at random.

Example 5 This example originates from a nonlinear descriptor system arising in chromatography. For more background information we refer to [KG97]. This descriptor system has been linearized in a working point and transformed into a dynamical system (18). To reduce the bandwidth of the matrix A, we applied the reverse Cuthill-McKee algorithm, which is provided as build-in function in MATLAB. The reordered matrix has the bandwidth $\max_{i,j:(A)_{ij}\neq 0} |i-j| = 31$. It is stable, but its symmetric part is indefinite. The Lyapunov equation is of order $n = 3\,600$. The underlying dynamical system has four inputs ($m = 4$) and two outputs ($q = 2$). In our numerical experiment we used the first column of the matrix B in (18) to create the right hand side of the Lyapunov equation. Thus, the computational cost for solving equation (19) is about four times the number of flops given in Table 3, whereas solving (20) requires about twice the cost displayed in this table.

In the sequel, we investigate the convergence speed of LR-Smith(l) applied to Examples 2–5. It should be noted that these examples pose problems of very large scale. The properties of these examples are not favorable for iterative methods. Whereas the matrix A is symmetric, but ill-conditioned in Example 2, it has an indefinite symmetric part in Examples 3–5. Note that we do not restrict ourself to examples with very small bandwidth for which our method is most effective. For all examples we determine the shift parameters by Algorithm 1, where the input parameters (k_+, k_-, l_0) of this algorithm are chosen as (40,20,10) for Examples 2, 4, 5 and (60,0,20) for Example 3. The latter is one of the few examples where slightly better results are obtained by ignoring the Ritz values w.r.t. A^{-1} in Algorithm 1. It is worth noting that this algorithm failed neither for Examples 2–5 nor in any of our further numerical tests not reported here.

Table 2 displays the numbers of iterations required by LR-Smith(l) to attain different relative residual norms. For each example, LR-Smith(l) delivers, with reasonable convergence speed, a solution of satisfactory accuracy. Consequently, the rank of the approximate solutions and the memory size needed to store them are comparably low. Table 3 shows the computational costs in terms of the number of flops.

Table 2: LR-Smith(l). Number of iterations required to attain different relative residual norms.

Example	relative residual norm				
	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}
2	30	50	60	80	100
3	42	63	63	84	> 315
4	50	60	80	90	100
5	40	50	70	80	> 300

Table 3: LR-Smith(l). Number of flops required to attain different relative residual norms.

Example	relative residual norm				
	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}
2	9.5e+07	1.3e+08	1.5e+08	1.9e+08	2.2e+08
3	2.2e+08	3.4e+08	3.4e+08	4.6e+08	>1.8e+09
4	1.4e+09	1.7e+09	2.1e+09	2.3e+09	2.5e+09
5	7.0e+08	8.6e+08	1.2e+09	1.3e+09	>4.9e+09

For comparison, we provide the results delivered by FOM-L in Table 4. In contrast to LR-Smith(l), FOM-L fails to compute accurate solutions within a reasonable number of iterations in two cases. In Table 5 the corresponding numbers of flops are displayed. Furthermore, the last column of this table shows estimates for the expected computational cost of the Bartels-Stewart method. Note that it was impossible to solve the large scale Lyapunov equations of Examples 2–5 by the Bartels-Stewart method due to extensive memory requirement of this method. However, even if the available memory was not the limiting factor for the application of this method, a comparison of the flop estimate for the Bartels-Stewart method with the number of flops required by LR-Smith(l) shows the superiority of our low rank method.

Table 4: FOM-L. Number of iterations required to attain different relative residual norms.

Example	relative residual norm				
	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}
2	49	> 300	> 300	> 300	> 300
3	61	95	203	> 300	> 300
4	166	207	254	298	> 300
5	> 300	> 300	> 300	> 300	> 300

Table 5: FOM-L and Bartels-Stewart method. Number of flops required by FOM-L to attain different relative residual norms. Estimates for the flop count of the Bartels-Stewart method are shown in the last column.

Example	relative residual norm for FOM-L					Bartels–
	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}	Stewart
2	1.1e+08	>4.2e+09	>4.2e+09	>4.2e+09	>4.2e+09	2.5e+13
3	6.6e+07	1.8e+08	1.1e+09	>2.7e+09	>2.7e+09	6.8e+11
4	6.0e+08	1.0e+09	1.8e+09	2.7e+09	>2.7e+09	3.9e+11
5	>3.1e+09	>3.1e+09	>3.1e+09	>3.1e+09	>3.1e+09	1.2e+12

8 Conclusions

This paper addresses the numerical solution of large, sparse, stable Lyapunov equations with right hand side matrices of low rank. We have presented the iterative methods LR-ADI and LR-Smith(l), which deliver low rank approximations to the solution matrix. LR-ADI and LR-Smith(l) are mathematically equivalent to the ADI iteration with a sequence of arbitrary shift parameters or with a set of l cyclic shift parameters, respectively. In this paper, LR-Smith(l) is of particular interest because the proper choice of l different shift parameters, where l is of moderate size (say $l = 10$), generally ensures a rapid convergence of ADI. The computational cost per iteration is constant for LR-Smith(l), but it is increasing for LR-ADI. Furthermore, we have presented a heuristic algorithm for determining a set of l suboptimal ADI parameters. The heuristic algorithm is easy to implement and does not require any a priori knowledge about the spectrum of the matrix A. All information about this matrix is gained from a pair of Arnoldi processes. Thus, LR-Smith(l) combined with the algorithm for determining the ADI parameters can be considered as a “black box” solver for large, sparse, stable Lyapunov equations. In general, the computational costs of LR-Smith(l) and LR-ADI are much smaller than that of the classical implementation of the ADI iteration. In particular, if A is a banded matrix, the memory requirements of both methods are moderate because the low rank iterates are efficiently stored in factored form. This allows to solve Lyapunov equations the order of which is so large that the explicit solution cannot be stored in computer memory. In general, LR-Smith(l) converges fast compared to other low rank methods, such as FOM-L. As a consequence, it delivers approximate solutions of very low rank. For instance, in Example 2, the largest of our test examples, the solution matrix of order 10 000 is approximated quite accurately by a matrix of rank 100. Considering this example, a comparison of LR-Smith(l) with direct standard methods, such as Bartels-Stewart method or Hammarling method, reveals the efficiency of the low rank method. The estimated computational and memory costs of the direct standard methods exceed those of LR-Smith(l) by factors 100 000 and 300, respectively!

LR-Smith(l) is not only an efficient means for solving an important class of Lyapunov equations, which arise in in a number of algorithms in control theory. Also, the thorough integration of this method into such “outer” algorithms can improve the overall complexity of these algorithms. This has been demonstrated using a model reduction method and an optimal control algorithm as examples. The model reduction method profits from the low rank property of the solution matrix. In contrast, only the product of the Lyapunov solution with a matrix containing few columns is desired in the optimal control algorithm. Such products can be computed efficiently by LR-Smith(l) without even forming approximate solutions implicitly or explicitly. Whereas dense n-$\times$-n matrices are involved in the conventional implementations of both example algorithms, this is avoided in the alternative implementations presented in this paper. Hence, LR-Smith(l) offers an opportunity to apply these control algorithms to several large scale control problems for which the standard implementations fail due to extensive computation or lack of memory. However, a prerequisite for involving LR-Smith(l) in these algorithms is that the numbers of inputs and outputs in the underlying dynamical system are small.

Finally, we should point out two aspects of our low rank methods which can become disadvantageous in some situations. Both LR-ADI and LR-Smith(l) require the solution of systems of linear equations $(A^T + pI)x = y$. If the nonzero pattern of the matrix A is unfavorable, the solution of these systems as well as the algorithms itself may be expensive with respect to both memory and computation. However, most sparse matrices arising in applications are either banded matrices or matrices which can be reordered to achieve this structure. The second drawback of our method is the restriction to Lyapunov equations with right hand side matrices of small rank. Nevertheless, if $m \gg 1$, splitting up the right hand side matrix into a sum of low rank matrices enables an efficient parallelization of our method.

References

[AL93] F.A. Aliev and V.B. Larin. Construction of square root factor for solution of the Lyapunov matrix equation. *Sys. Control Lett.*, 20:109–112, 1993.

[Arn51] W.E. Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quart. Appl. Math.*, 9:17–29, 1951.

[Bag69] T. Bagby. On interpolation by rational functions. *Duke J. Math.*, 36:95–104, 1969.

[BQO97] P. Benner and E. Quintana-Orti. Solving stable generalized Lyapunov equations with the matrix sign function. Preprint SFB97-23, Technische Universität Chemnitz, Chemnitz, Germany, 1997.

[BS72] R.H. Bartels and G.W. Stewart. Solution of the matrix equation $AX + XB = C$: Algorithm 432. *Comm. ACM*, 15:820–826, 1972.

[CR96] D. Calvetti and L. Reichel. Application of ADI iterative methods to the restoration of noisy images. *SIAM J. Matrix Anal. Appl.*, 17:165–186, 1996.

[Cut72] E. Cuthill. Several strategies for reducing the bandwidth of matrices. In D.J. Rose and R.A. Willoughby, editors, *Sparse Matrices and Their Applications.* Plenum Press, New York, 1972.

[DM68] E. Davison and F. Man. The numerical solution of $A^TQ + QA = -C$. *IEEE Trans. Automat. Control*, 13:448–449, 1968.

[EW91] N.S. Ellner and E.L. Wachspress. Alternating direction implicit iteration for systems with complex spectra. *SIAM J. Numer. Anal.*, 28:859–870, 1991.

[Fra87] B. A. Francis. *A Course in H_∞ Control Theory*, volume 88 of *Lecture Notes in Control and Information Sciences.* Springer-Verlag, New York, 1987.

[GL94] T. Gudmundsson and A.J. Laub. Approximate solution of large sparse Lyapunov equations. *IEEE Trans. Automat. Control*, 39:1110–1114, 1994.

[GL96] G.H. Golub and C.F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, Baltimore, 3rd edition, 1996.

[Glo84] K. Glover. All optimal Hankel norm approximations of linear multivariable systems and their L^∞-error bounds. *Internat. J. Control*, 39:1115–1193, 1984.

[Ham82] S.J. Hammarling. Numerical solution of the stable, non–negative definite Lyapunov equation. *IMA J. Numer. Anal.*, 2:303–323, 1982.

[HP88] A.S. Hodel and K.R. Poolla. Heuristic approaches to the solution of very large sparse Lyapunov and algebraic Riccati equations. In *Proc. 27th IEEE Conf. Decis. Cont.*, Austin, TX, December 1988.

[HPT96] A.S. Hodel, K.P. Poolla, and B. Tenison. Numerical solution of the Lyapunov equation by the power iteration. *Linear Algebra Appl.*, 236:205–230, 1996.

[HR92] D.Y. Hu and L. Reichel. Krylov-subspace methods for the Sylvester equation. *Linear Algebra Appl.*, 172:283–313, 1992.

[JK94] I.M. Jaimoukha and E.M. Kasenally. Krylov subspace methods for solving large Lyapunov equations. *SIAM J. Numer. Anal.*, 31:227–251, 1994.

[KG97] E. Kloppenburg and E. D. Gilles. Automatic control of the simulated moving bed process for C_8 aromatics separation using asymptotically exact input/output-linearization. submitted to *Journal of Process Control*, 1997.

[Kle68] D.L. Kleinman. On an iterative technique for Riccati equation computations. *IEEE Trans. Automat. Control*, 13:114–115, 1968.

[LL61] J. LaSalle and S. Lefschetz. *Stability of Liapunov's Direct Method.* Academic Press, New York, NY, 1961.

[LR93] N. Levenberg and L. Reichel. A generalized ADI iterative method. *Numer. Math.*, 66:215–233, 1993.

[LR95] P. Lancaster and L. Rodman. *Algebraic Riccati Equations.* Clarendon Press, Oxford, 1995.

[LT85] P. Lancaster and M. Tismenetsky. *Theory of Matrices.* Academic Press, Orlando, 2nd edition, 1985.

[Meh91] V. Mehrmann. *The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution*, volume 163 of *Lecture Notes in Control and Information Sciences.* Springer-Verlag, Heidelberg, July 1991.

[MM92] M. Marcus and M. Minc. *A Survey of Matrix Theory and Matrix Inequalities.* Dover Publications, New York, 1992.

[Moo81] B. C. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Trans. Automat. Control*, 26:17–31, 1981.

[PR55] D. Peaceman and H. Rachford. The numerical solution of elliptic and parabolic differential equations. *J. Soc. Indust. Appl. Math.*, 3:28–41, 1955.

[RW95] I.G. Rosen and C. Wang. A multi-level technique for the approximate solution of operator Lyapunov and algebraic Riccati equations. *SIAM J. Numer. Anal.*, 32:514–541, 1995.

[Saa81] Y. Saad. Krylov subspace methods for solving large unsymmetric linear systems. *Math. Comp.*, 37:105–126, 1981.

[Saa90] Y. Saad. Numerical solution of large Lyapunov equations. In M.A. Kaashoek, J.H. Van Schuppen, and A.C.M. Ran, editors, *Signal Processing, Scattering, Operator Theory and Numerical Methods*, pages 503–511. Birkhäuser, Boston, MA, 1990.

[Saa96] Y. Saad. *Iterative Methods for Sparse Linear Systems.* PWS Publishing Company, 1996.

[SC89] M.G. Safonov and R.Y. Chiang. A Schur method for balanced-truncation model reduction. *IEEE Trans. Automat. Control*, 34:729–733, 1989.

[Sim96] V. Sima. *Algorithms for Linear-Quadratic Optimization.* Pure and Applied Mathematics. Marcel Dekker, New York, NY, 1996.

[Smi68] R.A. Smith. Matrix equation $XA + BX = C$. *SIAM J. Appl. Math.*, 16, 1968.

[Sor92] D.C. Sorensen. Implicit application of polynomial filters in a k-step Arnoldi method. *SIAM J. Matrix Anal. Appl.*, 13:357–385, 1992.

[Sta91] G. Starke. Optimal alternating direction implicit parameters for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.*, 28:1431–1445, 1991.

[Sta92] G. Starke. Near-circularity for the rational Zolotarev problem in the complex plane. *J. Approximation Theory*, 70:115–130, 1992.

[Sta93] G. Starke. Fejer-Walsh points for rational functions and their use in the ADI iterative method. *J. Comput. Appl. Math.*, 46:129–141, 1993.

[Wac63] E.L. Wachspress. Extended application of alternating direction implicit iteration model problem theory. *J. Soc. Indust. Appl. Math.*, 11:994–1016, 1963.

[Wac66] E. Wachspress. *Iterative Solution of Elliptic Systems.* Prentice-Hall, 1966.

[Wac88a] E. Wachspress. The ADI minimax problem for complex spectra. *Appl. Math. Lett.*, 1:311–314, 1988.

[Wac88b] E. Wachspress. Iterative solution of the Lyapunov matrix equation. *Appl. Math. Lett.*, 1:87–90, 1988.

[Wac90] E.L. Wachspress. The ADI minimax problem for complex spectra. In D. Kincaid and L. Hayes, editors, *Iterative Methods for Large Linear Systems*, pages 251–271. Academic Press, San Diego, 1990.

[Wil65] J.H. Wilkinson. *The Algebraic Eigenvalue Problem.* Clarendon Press, Oxford, 1965.

Other titles in the SFB393 series:

96-01 V. Mehrmann, H. Xu. Chosing poles so that the single-input pole placement problem is well-conditioned. Januar 1996.

96-02 T. Penzl. Numerical solution of generalized Lyapunov equations. January 1996.

96-03 M. Scherzer, A. Meyer. Zur Berechnung von Spannungs- und Deformationsfeldern an Interface-Ecken im nichtlinearen Deformationsbereich auf Parallelrechnern.
March 1996.

96-04 Th. Frank, E. Wassen. Parallel solution algorithms for Lagrangian simulation of disperse multiphase flows. Proc. of 2nd Int. Symposium on Numerical Methods for Multiphase Flows, ASME Fluids Engineering Division Summer Meeting, July 7-11, 1996, San Diego, CA, USA. June 1996.

96-05 P. Benner, V. Mehrmann, H. Xu. A numerically stable, structure preserving method for computing the eigenvalues of real Hamiltonian or symplectic pencils. April 1996.

96-06 P. Benner, R. Byers, E. Barth. HAMEV and SQRED: Fortran 77 Subroutines for Computing the Eigenvalues of Hamiltonian Matrices Using Van Loans's Square Reduced Method. May 1996.

96-07 W. Rehm (Ed.). Portierbare numerische Simulation auf parallelen Architekturen. April 1996.

96-08 J. Weickert. Navier-Stokes equations as a differential-algebraic system. August 1996.

96-09 R. Byers, C. He, V. Mehrmann. Where is the nearest non-regular pencil? August 1996.

96-10 Th. Apel. A note on anisotropic interpolation error estimates for isoparametric quadrilateral finite elements. November 1996.

96-11 Th. Apel, G. Lube. Anisotropic mesh refinement for singularly perturbed reaction diffusion problems. November 1996.

96-12 B. Heise, M. Jung. Scalability, efficiency, and robustness of parallel multilevel solvers for nonlinear equations. September 1996.

96-13 F. Milde, R. A. Römer, M. Schreiber. Multifractal analysis of the metal-insulator transition in anisotropic systems. October 1996.

96-14 R. Schneider, P. L. Levin, M. Spasojević. Multiscale compression of BEM equations for electrostatic systems. October 1996.

96-15 M. Spasojević, R. Schneider, P. L. Levin. On the creation of sparse Boundary Element matrices for two dimensional electrostatics problems using the orthogonal Haar wavelet. October 1996.

96-16 S. Dahlke, W. Dahmen, R. Hochmuth, R. Schneider. Stable multiscale bases and local error estimation for elliptic problems. October 1996.

96-17 B. H. Kleemann, A. Rathsfeld, R. Schneider. Multiscale methods for Boundary Integral Equations and their application to boundary value problems in scattering theory and geodesy. October 1996.

96-18 U. Reichel. Partitionierung von Finite-Elemente-Netzen. November 1996.

96-19 W. Dahmen, R. Schneider. Composite wavelet bases for operator equations. November 1996.

96-20 R. A. Römer, M. Schreiber. No enhancement of the localization length for two interacting particles in a random potential. December 1996. to appear in: Phys. Rev. Lett., March 1997

96-21 G. Windisch. Two-point boundary value problems with piecewise constant coefficients: weak solution and exact discretization. December 1996.

96-22 M. Jung, S. V. Nepomnyaschikh. Variable preconditioning procedures for elliptic problems. December 1996.

97-01 P. Benner, V. Mehrmann, H. Xu. A new method for computing the stable invariant subspace of a real Hamiltonian matrix or Breaking Van Loan's curse? January 1997.

97-02 B. Benhammouda. Rank-revealing 'top-down' ULV factorizations. January 1997.

97-03 U. Schrader. Convergence of Asynchronous Jacobi-Newton-Iterations. January 1997.

97-04 U.-J. Görke, R. Kreißig. Einflußfaktoren bei der Identifikation von Materialparametern elastisch-plastischer Deformationsgesetze aus inhomogenen Verschiebungsfeldern. March 1997.

97-05 U. Groh. FEM auf irregulären hierarchischen Dreiecksnetzen. March 1997.

97-06 Th. Apel. Interpolation of non-smooth functions on anisotropic finite element meshes. March 1997

97-07 Th. Apel, S. Nicaise. The finite element method with anisotropic mesh grading for elliptic problems in domains with corners and edges.

97-08 L. Grabowsky, Th. Ermer, J. Werner. Nutzung von MPI für parallele FEM-Systeme. March 1997.

97-09 T. Wappler, Th. Vojta, M. Schreiber. Monte-Carlo simulations of the dynamical behavior of the Coulomb glass. March 1997.

97-10 M. Pester. Behandlung gekrümmter Oberflächen in einem 3D-FEM-Programm für Parallelrechner. April 1997.

97-11 G. Globisch, S. V. Nepomnyaschikh. The hierarchical preconditioning having unstructured grids. April 1997.

97-12 R. V. Pai, A. Punnoose, R. A. Römer. The Mott-Anderson transition in the disordered one-dimensional Hubbard model. April 1997.

97-13 M. Thess. Parallel Multilevel Preconditioners for Problems of Thin Smooth Shells. May 1997.

97-14 A. Eilmes, R. A. Römer, M. Schreiber. The two-dimensional Anderson model of localization with random hopping. June 1997.

97-15 M. Jung, J. F. Maitre. Some remarks on the constant in the strengthened C.B.S. inequality: Application to h- and p-hierarchical finite element discretizations of elasticity problems. July 1997.

97-16 G. Kunert. Error estimation for anisotropic tetrahedral and triangular finite element meshes. August 1997.

97-17 L. Grabowsky. MPI-basierte Koppelrandkommunikation und Einfluß der Partitionierung im 3D-Fall. August 1997.

97-18 R. A. Römer, M. Schreiber. Weak delocalization dueto long-range interaction for two electrons in a random potential chain. August 1997.

97-19 A. Eilmes, R. A. Römer, M. Schreiber. Critical behavior in the two-dimensional Anderson model of localization with random hopping. August 1997.

97-20 M. Meisel, A. Meyer. Hierarchically preconditioned parallel CG-solvers with and without coarse-matrix-solvers inside FEAP. September 1997.

97-21 J. X. Zhong, U. Grimm, R. A. Römer, M. Schreiber. Level-Spacing Distributions of Planar Quasiperiodic Tight-Binding Models. October 1997.

97-22 W. Rehm (Ed.). Ausgewählte Beiträge zum 1. Workshop Cluster-Computing. TU Chemnitz, 6./7. November 1997.

97-23 P. Benner, Enrique S. Quintana-Ortí. Solving stable generalized Lyapunov equations with the matrix sign function. October 1997

97-24 T. Penzl. A Multi-Grid Method for Generalized Lyapunov Equations. October 1997

97-25 G. Globisch. The hierarchical preconditioning having unstructured threedimensional grids. December 1997

97-26 G. Ammar, C. Mehl, V. Mehrmann. Schur-like forms for matrix Lie groups, Lie algebras and Jordan algebras. November 1997

97-27 U. Elsner. Graph partitioning - a survey. December 1997.

97-28 W. Dahmen, R. Schneider. Composite Wavelet Bases for Operator Equations. December 1997.

97-29 P. L. Levin, M. Spasojević, R. Schneider. Creation of Sparse Boundary Element Matrices for 2-D and Axi-symmetric Electrostatics Problems Using the Bi-orthogonal Haar Wavelet. December 1997.

97-30 W. Dahmen, R. Schneider. Wavelets on Manifolds I: Construction and Domain Decomposition. December 1997.

97-31 U. Elsner, V. Mehrmann, F. Milde, R. A. Römer, M. Schreiber. The Anderson Model of Localization: A Challenge for Modern Eigenvalue Methods. December 1997.

98-01 B. Heinrich, S. Nicaise, B. Weber. Elliptic interface problems in axisymmetric domains. Part II: The Fourier-finite-element approximation of non-tensorial singularities. January 1998.

98-02 T. Vojta, R. A. Römer, M. Schreiber. Two interfacing particles in a random potential: The random model revisited. February 1998.

98-03 B. Mehlig, K. Müller. Non-universal properties of a complex quantum spectrum. February 1998.

98-04 B. Mehlig, K. Müller, B. Eckhardt. Phase-space localization and matrix element distributions in systems with mixed classical phase space. February 1998.

98-05 M. Bollhöfer, V. Mehrmann. Nested divide and conquer concepts for the solution of large sparse linear systems. to app.: April 1998.

98-06 T. Penzl. A cyclic low rank Smith method for large, sparse Lyapunov equations with applications in model reduction and optimal control. March 1998.

98-07 V. Mehrmann, H. Xu. Canonical forms for Hamiltonian and symplectic matrices and pencils. March 1998.

98-08 C. Mehl. Condensed forms for skew-Hamiltonian/Hamiltonian pencils. March 1998.

The complete list of current and former preprints is available via
`http://www.tu-chemnitz.de/sfb393/preprints.html`.