DFG-Forschungszentrum MATHEON
Mathematik für Schlüsseltechnologien

Krylov subspace methods for projected Lyapunov equations

Tatjana Stykel and Valeria Simoncini

Technical Report 735-2010

DFG Research Center MATHEON "*Mathematics for key technologies*"
http://www.matheon.de

KRYLOV SUBSPACE METHODS FOR PROJECTED LYAPUNOV EQUATIONS

TATJANA STYKEL[†] AND VALERIA SIMONCINI[‡]

Abstract. We consider the numerical solution of projected Lyapunov equations using Krylov subspace iterative methods. Such equations play a fundamental role in balanced truncation model reduction of descriptor systems. We present generalizations of the extended block and global Arnoldi methods to projected Lyapunov equations and compare these methods with the alternating direction implicit method with respect to performance on different examples. A deflation strategy is also proposed to overcome possible breakdown in the recurrence.

Key words. Projected Lyapunov equation, Krylov subspace method, Arnoldi process, alternating direction implicit method, deflation

AMS subject classifications. 15A22, 15A24, 65F10

1. Introduction. Consider the projected continuous-time algebraic Lyapunov equation (PCALE)

$$EXA^T + AXE^T = -P_l BB^T P_l^T, \quad X = P_r X P_r^T, \tag{1.1}$$

where E, $A \in \mathbb{R}^{n,n}$ and $B \in \mathbb{R}^{n,m}$ are given matrices and $X \in \mathbb{R}^{n,n}$ is an unknown matrix. Furthermore, P_r and P_l are the spectral projectors onto the right and left deflating subspaces of a regular pencil $\lambda E - A$ corresponding to the finite eigenvalues along the right and left deflating subspaces associated with the eigenvalue at infinity. We assume that the pencil $\lambda E - A$ is *stable*, i.e., all its finite eigenvalues have negative real part. In this case, the PCALE (1.1) has a unique symmetric, positive semidefinite solution X, see [38]. Projected Lyapunov equations arise in stability analysis and many control problems for differential-algebraic equations or descriptor systems including the characterization of the controllability and observability properties, computing the $\mathbb{H}_2$ and Hankel system norms and balancing-related model reduction [3, 38, 39, 40].

If $E = I$ is an identity matrix, then $P_r = P_l = I$ and (1.1) is a standard Lyapunov equation. Such equations of small to medium size can be numerically solved using the Bartels-Stewart method [1] or the Hammarling method [15] that are based on the reduction of A to Schur form. For solving Lyapunov equations, one can also employ the sign function method [7, 23, 29], especially, when the hierarchical matrix format can be exploited [2, 12] or when dense large Lyapunov equations are solved on parallel computers [4]. Finally, several iterative methods such as the alternating direction implicit (ADI) method [5, 24, 25, 27], the cyclic Smith method [13, 27] and Krylov subspace type methods [16, 18, 19, 21, 30, 34] have been developed for large-scale Lyapunov equations over the last twenty years. All these methods except for Krylov subspace methods have been extended to projected Lyapunov equations [37, 41, 42].

Krylov subspace methods are attractive when dealing with sparse and large coefficient matrices A and E. The general framework consists in projecting the original problem onto a much smaller subspace, giving rise to a reduced-order linear equation

[†]Institut für Mathematik, MA 4-5, Technische Universität Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany (stykel@math.tu-berlin.de).

[‡]Dipartimento di Matematica, Università di Bologna, Piazza di Porta S. Donato 5, 40127 Bologna, Italy (valeria@dm.unibo.it).

of the same type, then solving this equation and projecting back the result. The small size equation is obtained by imposing an extra condition on the required approximate solution. All methods proposed for the standard Lyapunov equation differ either for the choice of approximation/projection space, or for the extra condition. We will discuss the most common choices in Section 2. Dealing with the projected equation provides an additional challenge, as E is in general singular. In this case, the approximation space needs to remain in the correct deflating spaces of the pencil $\lambda E - A$, and, thus, the algorithms have to be modified accordingly. We will investigate extensions of several variants of known projection-type methods to the PCALE (1.1). In particular, we consider the block and global Krylov subspace methods [18, 21], the extended Krylov subspace method [34], a combination of them [16], and also recently developed preconditioning with the classical ADI method [19]. Note that when B is a column vector, then the first two methods are equivalent. We will use computational costs and time evaluations to compare these methods, highlighting the most characteristic features. We note that such a detailed comparison among these methods appears to be new also for the standard Lyapunov equation. Finally, the singularity of E may significantly affect the robustness of some of the methods, as the basis vectors may become almost linearly dependent. We devise a new deflation strategy that makes the extended block Krylov subspace method more reliable by curing near breakdown due to quasi-linear dependence.

The paper is organized as follows. In Section 2, we consider Galerkin-type approximation to the solution of the PCALE (1.1). In Section 3, we extend the block and global Krylov subspace methods based on the Arnoldi process to solve this equation. A combination of these methods with an extended Krylov subspace approach is presented in Sections 4 and 5. A deflation technique and stopping criteria for new methods are also discussed there. In Section 6, we review the low-rank ADI method and extend an ADI-preconditioned global Arnoldi algorithm to the PCALE (1.1). In Section 7, we report some results of numerical experiments to demonstrate the properties of the presented iterative methods and to compare them.

Notation and definitions. Throughout the paper the open left half-plane is denoted by $\mathbb{C}_-$. The matrix A^T stands for the transpose of A, and $A^{-T} = (A^{-1})^T$. An identity matrix of order n is denoted by I_n or simply by I. We denote by $A(i:k, j:l)$ a submatrix of $A \in \mathbb{R}^{n,m}$ that contains its rows i to k and columns j to l. The image and the trace of a matrix A are denoted by $\operatorname{im} A$ and $\operatorname{trace}(A)$, respectively. A Frobenius inner product of the matrices $A, B \in \mathbb{R}^{n,m}$ is defined as $\langle A, B\rangle_F = \operatorname{trace}(B^T A)$ and $\|A\|_F = \sqrt{\langle A, A\rangle_F}$ is the Frobenius matrix norm. The spectral matrix norm of $A \in \mathbb{R}^{n,m}$ is denoted by $\|A\|_2$. The Kronecker product of the matrices $A = [a_{ij}] \in \mathbb{R}^{n,m}$ and $B \in \mathbb{R}^{p,q}$ is defined as $A \otimes B = [a_{ij}B] \in \mathbb{R}^{np,mq}$. Finally, the $\diamond$-product of $A = [A_1, \ldots, A_k] \in \mathbb{R}^{n,km}$ and $B = [B_1, \ldots, B_l] \in \mathbb{R}^{n,lm}$ is defined as $A^T \diamond B = [\operatorname{trace}(A_i^T B_j)]_{i=1,\ldots,k}^{j=1,\ldots,l} \in \mathbb{R}^{k,l}$. The matrix $A = [A_1, \ldots, A_k]$ is called F-orthonormal if $A^T \diamond A = I_k$.

In the sequel, we will need some generalized inverses. A *Drazin inverse* A^D of a matrix $A \in \mathbb{R}^{n,n}$ is defined as a unique solution of the following system of matrix equations

$$A^D A A^D = A^D, \qquad A A^D = A^D A, \qquad A^{\nu+1} A^D = A^\nu.$$

Furthermore, a matrix A^- is called a *reflexive generalized inverse* of $A \in \mathbb{R}^{n,n}$ with

respect to projectors P_1 and P_2, if it is a unique solution of the matrix equations

$$A^- A A^- = A^-, \qquad A A^- = P_1, \qquad A^- A = P_2.$$

We refer to [9] for detailed properties of these generalized inverses.

2. Order reduction by projection. This section serves as introduction to the methods we will describe in detail in the next sections. All these approaches determine an approximate solution by an iterative process. The approximation space is a subspace of possibly much smaller dimension than that of the original problem, and the space dimension grows at each iteration, that is, the approximation spaces are nested. To avoid notational complications, here and in the following we will avoid explicitly labeling the involved matrices when not strictly necessary. Therefore, quantities such as the basis matrices will be defined within the described context.

If the pencil $\lambda E - A$ is stable, then A is nonsingular and the PCALE (1.1) can be written as the projected standard Lyapunov equation

$$F X + X F^T = -G G^T, \quad X = P_r X P_r^T, \tag{2.1}$$

where $F = A^{-1}E$ and $G = P_r A^{-1} B$. Here we utilized the fact that $A P_r = P_l A$. Now we can use a standard projection technique to determine the solution of (2.1). Let $\mathcal{V}$ be a subspace of $\operatorname{im} P_r$ and let $V \in \mathbb{R}^{n,k}$ be a matrix with orthonormal columns that span $\mathcal{V}$, i.e., $\operatorname{im} V = \mathcal{V}$. We seek an approximate solution in the form $X \approx V Y V^T$, where $Y \in \mathbb{R}^{k,k}$ has to be determined. Let

$$R(V Y V^T) \;:=\; F(V Y V^T) + (V Y V^T) F^T + G G^T$$

be the residual associated with $V Y V^T$. Then the Galerkin orthogonality condition $V^T R(V Y V^T) V = 0$ yields the following reduced Lyapunov equation

$$(V^T F V) Y + Y (V^T F V)^T = -V^T G G^T V.$$

If $V^T F V$ is stable, i.e., all its eigenvalues have negative real part, then this equation has a unique symmetric, positive semidefinite solution Y, see [11]. For $k \ll n$, such a solution can be computed using direct methods (cf., e.g., [1, 15]) based on the Schur form of the small matrix $V^T F V$. An alternative approach for computing Y is the minimization of the residual norm $\|R(V Y V^T)\|_F$. In this case, Y will satisfy another matrix equation, see [18, 30] for details. Since the latter is more computationally involved, we will consider in the sequel the Galerkin projection only.

The accuracy of the approximate solution $V Y V^T \approx X$ highly depends on the choice of $\mathcal{V}$. If $\mathcal{V} = \operatorname{im} V$ approximates well the solution space, then we can expect that $V Y V^T$ is a good approximation to the solution of (2.1) and also of the PCALE (1.1). It should be noted that the matrix $V Y V^T$ satisfies the second equation in (1.1) and also in (2.1) exactly since $\operatorname{im} V \subset \operatorname{im} P_r$.

3. Krylov subspace methods. For standard Lyapunov equations with $E = I$ and $P_r = P_l = I$, the subspace $\mathcal{V}$ was chosen in [17, 18, 30] as a *block Krylov subspace*

$$\begin{aligned} \mathcal{K}_k^{\square}(F, G) &= \operatorname{blockspan}\{\, G,\, F G,\, \ldots,\, F^{k-1} G \,\} \\ &= \left\{ \sum_{i=0}^{k} F^i G\, \Theta_i,\;\; \Theta_i \in \mathbb{R}^{m,m} \right\}, \end{aligned}$$

where $G \in \mathbb{R}^{n,m}$; see, e.g., [14] for a more detailed description of block Krylov subspaces. This choice was originally justified by the fact that under certain hypotheses and for sufficiently large k, such a space contains the exact solution. A complete convergence theory for Galerkin-type methods based on this space was recently developed in [35]. In the framework of Section 2, we note that $\mathcal{K}_k^{\square}(F,G) \subseteq \mathcal{K}_{k+1}^{\square}(F,G)$. The block Arnoldi procedure [31] can be used to build and expand a basis for the subspace: starting with V such that $V^TV = I$ and $\operatorname{im} V = \operatorname{im} G$, at step $k+1$ the basis matrix V is updated from that at the previous step by adding vectors after orthogonalization with respect to the basis vectors already computed. For this choice of space, an approximation to the solution of the PCALE (1.1) is obtained as $X^{BA} = VYV^T$, where Y solves the reduced Lyapunov equation

$$HY + YH^T = -\hat{B}\hat{B}^T \tag{3.1}$$

with a block upper Hessenberg matrix $H = V^TA^{-1}EV$ and $\hat{B} = V^TP_rA^{-1}B$.

In the global Arnoldi method [20], the approximation space is given by

$$\widetilde{\mathcal{K}}_k(F,G) = \left\{ \sum_{i=0}^{k} F^i G\,\theta_i, \quad \theta_i \in \mathbb{R} \right\}.$$

Note that $\widetilde{\mathcal{K}}_k(F,G)$ can be obtained from $\mathcal{K}_k^{\square}(F,G)$ by requiring that each Θ_i is a multiple of the identity, i.e., $\Theta_i = \theta_i I$.

A basis matrix whose columns span $\widetilde{\mathcal{K}}_k(F,G)$ is obtained by requiring that the block columns be F-orthonormal. As in the standard block Krylov space, this requirement can be imposed during the iterative construction of the basis vectors. Note that for problems where G is a single column vector, the block Arnoldi and the global Arnoldi methods coincide. Due to the special orthogonality condition, the matrices involved in the projection phase take a special structure. Let V be the matrix whose F-orthonormal columns span $\widetilde{\mathcal{K}}_k(F,G)$. An approximate solution of the PCALE (1.1) can be determined as

$$X^{GA} = V(Y \otimes I_m)V^T,$$

where Y is a symmetric solution of the Lyapunov equation

$$TY + YT^T = -\|P_rA^{-1}B\|_F^2 e_1e_1^T \tag{3.2}$$

with upper Hessenberg $T = V^T \diamond (A^{-1}EV) \in \mathbb{R}^{k,k}$ and $e_1 = [\,1,\,0,\ldots,0\,]^T \in \mathbb{R}^k$. The matrix T has lower dimension than the projected matrix in the block Krylov subspace (k instead of km), so that solving the reduced Lyapunov equation (3.2) may be significantly cheaper than (3.1). This is possibly the main advantage of the global approach, compared to that based on the much richer block Krylov subspace.

A drawback of Krylov subspace methods is that they often converge slowly and relatively many iterations need be performed to determine an approximate solution with high accuracy [35]. For increasing k, the storage requirements for the dense basis matrix V is excessive and the computation of $X \approx VYV^T$ becomes expensive. In the next sections, we describe state-of-the-art and very recent methods that tend to overcome these difficulties, and that have now superseded these basic approaches for solving the standard Lyapunov equation.

4. Extended Krylov subspace methods. In [34], it was proposed to use the *extended block Krylov subspace*

$$\mathcal{K}_k^{\square}(F, G) + \mathcal{K}_k^{\square}(F^{-1}, G)$$

as approximation space, noticing that powers in F and F^{-1} may have a beneficial effect on the speed of convergence, and thus on the performed number of iterations. We observe that F^{-1} is not explicitly required, but only its action to a vector, so that either direct or iterative solves can be employed. A complete account on the convergence properties of the solver based on this extended Krylov subspace was very recently developed in [22].

In the context of the PCALE (1.1) with singular E, the matrix $F = A^{-1}E$ is not invertible so that it seems that the space $\mathcal{K}_k^{\square}(F^{-1}, G)$ could not be constructed. Fortunately, this problem can be circumvented and we present a generalization of the extended Krylov subspace method to the PCALE (1.1).

For this purpose, we transform the pencil $\lambda E - A$ into the Weierstrass canonical form

$$E = T_l \begin{bmatrix} I_{n_f} & 0 \\ 0 & E_\infty \end{bmatrix} T_r, \qquad A = T_l \begin{bmatrix} A_f & 0 \\ 0 & I_{n_\infty} \end{bmatrix} T_r, \tag{4.1}$$

where T_l and T_r are the left and right nonsingular transformation matrices [11]. The eigenvalues of A_f are the finite eigenvalues of $\lambda E - A$, and a nilpotent matrix E_∞ corresponds to an eigenvalue at infinity. Using (4.1), the projectors P_r and P_l can be represented as

$$P_r = T_r^{-1} \begin{bmatrix} I_{n_f} & 0 \\ 0 & 0 \end{bmatrix} T_r, \qquad P_l = T_l \begin{bmatrix} I_{n_f} & 0 \\ 0 & 0 \end{bmatrix} T_l^{-1}, \tag{4.2}$$

and the matrix $F = A^{-1}E$ takes the form

$$F = T_r^{-1} \begin{bmatrix} A_f^{-1} & 0 \\ 0 & E_\infty \end{bmatrix} T_r. \tag{4.3}$$

Since the inverse of F does not exist, we consider the Drazin inverse F^D of F that can be written due to (4.3) as

$$F^D = T_r^{-1} \begin{bmatrix} A_f & 0 \\ 0 & 0 \end{bmatrix} T_r$$

see, e.g., [9]. On the other hand, we have $F^D = E^- A$, where

$$E^- = T_r^{-1} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} T_l^{-1}$$

is a reflexive generalized inverse of E with respect to the projectors P_l and P_r. This representation can be obtained from the definition of E^- and relations (4.1) and (4.2). Note that the matrix E^- has already been employed in [8, 36] for computing a partial realization for descriptor systems and also the largest eigenvalues of a matrix pencil using Krylov subspace methods.

Thus, similar to the case $E = I$ in [34], we propose to compute an approximate solution $V_k Y V_k^T$ to the PCALE (1.1), where a Galerkin orthogonality is enforced, and V_k has orthonormal columns that span the extended block Krylov subspace

$$\mathcal{K}_k^{\square}(A^{-1}E, P_r A^{-1}B) + \mathcal{K}_k^{\square}(E^- A, E^- B).$$

An outline of the extended block Arnoldi method for (1.1) is given in Algorithm 4.1.

ALGORITHM 4.1. *The extended block Arnoldi method for the PCALE.*

Given $E, A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$ and the spectral projectors P_l and P_r, compute an approximate solution X_k of the PCALE (1.1).

1. *Compute the QR decomposition $[E^-B,\ P_rA^{-1}B\,] = \hat{V}_1 H_{1,0}$, where the matrix $\hat{V}_1 \in \mathbb{R}^{n,2m}$ has orthonormal columns and $H_{1,0} \in \mathbb{R}^{2m,2m}$ is upper triangular.*
2. *Set $V_1 := \hat{V}_1$, $V_{1,1} := \hat{V}_1[\,I_m,\,0\,]^T$ and $V_{1,2} := \hat{V}_1[\,0,\,I_m\,]^T$.*
3. *FOR $j = 1, 2, \ldots, k$*
   (a) *Compute $V^{(j)} = [\,E^-AV_{j,1},\ A^{-1}EV_{j,2}\,]$.*
   (b) *FOR $i = 1, 2, \ldots, j$ % Modified block Gram-Schmidt orthogonal.*
   $H_{i,j} := \hat{V}_i^T V^{(j)}$;
   $V^{(j)} := V^{(j)} - \hat{V}_i H_{i,j}$;
   END FOR
   (c) *Compute the QR decomposition $V^{(j)} = \hat{V}_{j+1} H_{j+1,j}$, where the matrix $\hat{V}_{j+1} \in \mathbb{R}^{n,2m}$ has orthonormal columns and $H_{j+1,j} \in \mathbb{R}^{2m,2m}$ is upper triangular.*
   (d) *Set $V_{j+1} := [V_j, \hat{V}_{j+1}]$, $V_{j+1,1} := \hat{V}_{j+1}[I_m, 0]^T$, $V_{j+1,2} := \hat{V}_{j+1}[0, I_m]^T$.*
   (e) *Solve the Lyapunov equation*

$$\Phi_j Y_j + Y_j \Phi_j^T = -B_j B_j^T, \tag{4.4}$$

   where $\Phi_j = V_j^T E^- A V_j$ and $B_j = V_j^T E^- B$.
   END FOR
4. *Compute $X_k = V_k Y_k V_k^T$.*

We now derive some recursive relations that can be used to significantly reduce the computational cost of this algorithm. First of all we show that the matrices B_j and Φ_j at Step 3(e) can be obtained as a by-product of the iteration with no additional matrix-vector products with E^- and A and inner products with long vectors.

Let $J_{2mk,j} = [\,0, \ldots, 0, I_m, 0, \ldots, 0\,]^T \in \mathbb{R}^{2mk,m}$ be formed from the identity matrix I_{2mk} by removing the first $m(j-1)$ and the last $m(2k-j)$ columns and let $J_{2mk,j:j+i} = [\,J_{2mk,j} \ldots, J_{2mk,j+i}\,] \in \mathbb{R}^{2mk,m(i+1)}$. Then the relation $[\,E^-B,\ P_rA^{-1}B\,] = \hat{V}_1 H_{1,0}$ at Step 1 of Algorithm 4.1 implies that

$$E^-B = \hat{V}_1 H_{1,0} J_{2m,1} = V_j J_{2mj,1:2} H_{1,0} J_{2m,1}.$$

Hence, $B_j = V_j^T E^- B = J_{2mj,1:2} H_{1,0} J_{2m,1} \in \mathbb{R}^{2mj,m}$. Steps 3(a)-(c) can be shortly written as

$$[\,E^-AV_{j,1},\ A^{-1}EV_{j,2}\,] - \sum_{i=1}^{j} \hat{V}_j H_{i,j} = \hat{V}_{j+1} H_{j+1,j}. \tag{4.5}$$

Then for the matrices $V^{(i)} = [\,E^-AV_{i,1},\ A^{-1}EV_{i,2}\,]$, $i = 1, \ldots, j$, we have

$$[\,V^{(1)}, \ldots, V^{(j)}\,] = V_j H_j + \hat{V}_{j+1} H_{j+1,j} J_{2mj,2j-1:2j}^T = V_{j+1} \underline{H}_j,$$

where

$$\underline{H}_j = \begin{bmatrix} H_{1,1} & \cdots & H_{1,j-1} & H_{1,j} \\ H_{2,1} & \ddots & H_{2,j-1} & H_{2,j} \\ & \ddots & \ddots & \vdots \\ & & H_{j,j-1} & H_{j,j} \\ & & & H_{j+1,j} \end{bmatrix} = \begin{bmatrix} H_j \\ H_{j+1,j} J_{2mj,2j-1:2j}^T \end{bmatrix} \in \mathbb{R}^{2m(j+1),2mj} \tag{4.6}$$

and $H_j \in \mathbb{R}^{2mj,2mj}$ is block upper Hessenberg. The following theorem shows that the matrix

$$\underline{\Phi}_j = V_{j+1}^T E^- A V_j = \begin{bmatrix} \Phi_j \\ \Phi_{j+1,j} J_{2mj,2j-1:2j}^T \end{bmatrix} \in \mathbb{R}^{2m(j+1),2mj}$$

and, hence, also Φ_j can be obtained from $\underline{H}_j$ in an inexpensive way.

THEOREM 4.2. *Let $\underline{H}_j$ be as in (4.6) such that $H_{i+1,i}$, $i = 0, \ldots, j$, are upper triangular and nonsingular and let*

$$H_{i+1,i}^{-1} = \begin{bmatrix} S_{11}^{(i)} & S_{12}^{(i)} \\ 0 & S_{22}^{(i)} \end{bmatrix}, \quad S_{11}^{(i)}, S_{22}^{(i)} \in \mathbb{R}^{m,m}.$$

Then the odd column blocks of the matrix $\underline{\Phi}_j = V_{j+1}^T E^- A V_j = [\,\hat{\Phi}_1, \ldots, \hat{\Phi}_{2j}\,]$ with $\hat{\Phi}_i \in \mathbb{R}^{2m(j+1),m}$ satisfy

$$\hat{\Phi}_{2i-1} = \underline{\Phi}_j J_{2mj,2i-1} = \underline{H}_j J_{2mj,2i-1}, \quad i = 1, \ldots, j,$$

while the even column blocks of $\underline{\Phi}_j$ have the form

$$\begin{aligned} \hat{\Phi}_2 &= \underline{H}_j J_{2mj,1} (S_{11}^{(0)})^{-1} S_{12}^{(0)} + J_{2m(j+1),1} (S_{11}^{(0)})^{-1} S_{22}^{(0)}, \\ \hat{\Phi}_{2i} &= \hat{\Phi}_{2i-1} (S_{11}^{(i-1)})^{-1} S_{12}^{(i-1)} + \Big(J_{2m(j+1),2i-2} - \begin{bmatrix} \underline{\Phi}_{i-1} \\ 0 \end{bmatrix} H_{i-1} J_{2m(i-1),2i-2} \Big) S_{22}^{(i-1)}, \end{aligned}$$

for $i = 2, \ldots, j$.

Proof. The result can be proved analogously to [34, Proposition 3.2]. ∎

Having Φ_j and B_j, the small dimensional Lyapunov equation (4.4) at Step 3(e) of Algorithm 4.1 can be solved by the Bartels-Stewart method [1]. Note that Φ_j has a block upper Hessenberg form that can be exploited to further reduce the computational effort. If all eigenvalues of Φ_j have negative real part, then the solution of (4.4) is a symmetric, positive semidefinite matrix. In this case, we can use the Hammarling method [15] to compute the Cholesky factor Z_j of $Y_j = Z_j Z_j^T$ rather than Y_j. Then the approximate solution of the PCALE (1.1) can be obtained also in factored form $X_k = (V_k Z_k)(V_k Z_k)^T$. Let $Z_k = [\,U_{1k}, U_{2k}\,]\mathrm{diag}(\Sigma_{1k}, \Sigma_{2k})[\,W_{1k}, W_{2k}\,]^T$ be a singular value decomposition (SVD), where $[\,U_{1k}, U_{2k}\,]$ and $[\,W_{1k}, W_{2k}\,]$ are orthogonal, Σ_{1k} and Σ_{2k} are diagonal with $\|\Sigma_{2k}\|_2 \le \epsilon \|\Sigma_{1k}\|_2$ for some small threshold ϵ. Then we can determine the approximate solution of (1.1) in the compressed form $X \approx Z Z^T$ with the low-rank Cholesky factor $Z = V_k U_{1k} \Sigma_{1k}$.

REMARK 4.3. For $E = I$, the solvability of the Lyapunov equation (4.4) is guaranteed if the matrix A is dissipative, i.e., $A + A^T < 0$. An extension of this

condition to the matrix pencil case is that $E = E^T \geq 0$ and $v^T(A + A^T)v < 0$ for all vectors v such that $Ev \neq 0$. Unfortunately, we were unable to show that these dissipativity conditions together imply the existence of the solution of (4.4). The conditions above imply that if $\lambda E - A$ is stable, then $\lambda V^T EV - V^T AV$ is also stable for all V such that $\operatorname{im} V \subseteq \operatorname{im} P_r$. In order to guarantee the solvability of (4.4), we need, however, that $V^T E^- AV$ is stable. Of course, one could assume that $E^- A$ is dissipative. But this condition is difficult to verify in practice.

A major difficulty in the iterative solution of large-scale projected Lyapunov equations is that the projectors P_r and P_l are required. Fortunately, in many applications such as semi-explicit problems of index one, Stokes-like problems, constrained multibody dynamics and circuit equations, the matrices E and A have some special block structure. This structure can be exploited to construct these projectors also in block form [28, 42]. Of course, we will never store these matrices. Instead, we compute projector-vector products block-wise. Furthermore, we can use the structure of E to determine the matrix-vector products $z = E^- b$, see [8]. Generally, these products can be computed by solving the linear systems

$$Ez = P_l b, \qquad z = P_r z$$

using iterative methods. Analogously, the matrix A^{-1} does not have to be determined explicitly. Instead, we can employ a sparse LU factorization or solve the linear systems $Az = b$ iteratively.

REMARK 4.4. In exact arithmetic, the basis matrix V_j satisfies $V_j = P_r V_j$ and, hence, the second equation in (1.1) is fulfilled for the low-rank approximation $V_j Y_j V_j^T$. However, in finite precision arithmetic, a drift-off effect may occur. In this case, we need to project V_j onto the image of P_r by pre-multiplication with P_r. In order to limit the additional computation cost we can do this, for example, at every second or third iteration step.

4.1. Stopping criterion. The iteration in Algorithm 4.1 can be stopped as soon as a normalized residual norm $\eta(X_k) = \|R_k\|_F / \|P_l BB^T P_l^T\|_F$ with the residual

$$R_k = AX_k E^T + EX_k A^T + P_l BB^T P_l^T \tag{4.7}$$

satisfies the condition $\eta(X_k) \leq tol$ with a user-defined tolerance tol. The following theorem shows how to verify this condition without computing the approximation X_k.

THEOREM 4.5. *Let $X_k = V_k Y_k V_k^T$ be an approximate solution of the PCALE (1.1) computed by Algorithm 4.1 and let R_k be as in (4.7). Then*

$$\|E^- R_k (E^-)^T\|_F = \sqrt{2}\, \|J_{2m(k+1),2k+1}^T \underline{\Phi}_k Y_k\|_F =: \delta_{EBA}, \tag{4.8}$$

$$\|R_k\|_F \leq \sqrt{2}\, \|E\|_F^2 \|J_{2m(k+1),2k+1}^T \underline{\Phi}_k Y_k\|_F. \tag{4.9}$$

Proof. We have

$$E^- AV_k = V_k \Phi_k + \hat{V}_{k+1} \Phi_{k+1,k} J_{2mk,2k-1:2k}^T = V_{k+1} \underline{\Phi}_k. \tag{4.10}$$

Taking into account that $E^- P_l = E^-$, $E^- E = P_r$ and $X_k = P_r X_k P_r^T$, we obtain

$$E^- R_k (E^-)^T = E^- AX_k + X_k (E^- A)^T + E^- BB^T (E^-)^T.$$

Substituting $X_k = V_k Y_k X_k^T$ and $E^- B = V_k B_k$ in this equation and using (4.10), we have

$$\begin{aligned} E^- R_k (E^-)^T &= V_{k+1} \underline{\Phi}_k Y_k V_k^T + V_k Y_k \underline{\Phi}_k^T V_{k+1}^T + V_k B_k B_k^T V_k^T \\ &= V_{k+1} \begin{bmatrix} 0 & Y_k J_{2mk,2k-1:2k} \Phi_{k+1,k}^T \\ \Phi_{k+1,k} J_{2mk,2k-1:2k}^T Y_k & 0 \end{bmatrix} V_{k+1}^T. \end{aligned}$$

Thus, $\|E^- R_k (E^-)^T\|_F = \sqrt{2}\, \|\Phi_{k+1,k} J_{2mk,2k-1:2k}^T Y_k\|_F$. Since

$$\Phi_{k+1,k} J_{2mk,2k-1:2k}^T = J_{2m(k+1),2k+1:2k+2}^T \underline{\Phi}_k, \qquad J_{2m(k+1),2k+2}^T \underline{\Phi}_k = 0,$$

we have (4.8). Estimate (4.9) follows then from $R_k = E E^- R_k (E^-)^T E^T$. □

Theorem 4.5 suggests that the following stopping criterion could be used in Algorithm 4.1:

$$\text{Stop if } \|E^- R_k (E^-)^T\|_F / \|E^- B B^T (E^-)^T\|_F \le tol.$$

Here $E^- R_k (E^-)^T$ is the residual for the projected Lyapunov equation

$$E^- A X + X (E^- A)^T = -E^- B B^T (E^-)^T, \quad X = P_r X P_r^T,$$

which is equivalent to (1.1). The verification of this condition does not require the approximate solution X_k, but only its projection Y_k. Therefore, X_k need only be computed after convergence is achieved.

As a final remark, we note that instead of the Lyapunov equation (4.4), in Algorithm 4.1 we could solve the Lyapunov equation

$$\tilde{\Phi}_j Y_j + Y_j \tilde{\Phi}_j^T = -\tilde{B}_j \tilde{B}_j^T, \tag{4.11}$$

where $\tilde{\Phi}_j = V_j^T A^{-1} E V_j$, $\tilde{B}_j = V_j^T P_r A^{-1} B$. The matrices $\underline{\tilde{\Phi}}_j = V_{j+1}^T A^{-1} E V_j$ could again be determined recursively from $\underline{H}_j$. In this case, the iteration could be terminated once the condition

$$\frac{\|A^{-1} R_k A^{-T}\|_F}{\|P_r A^{-1} B B^T A^{-T} P_r^T\|_F} = \frac{\sqrt{2} \|J_{2m(k+1),2k+1}^T \underline{\tilde{\Phi}}_k Y_k\|_F}{\|P_r A^{-1} B B^T A^{-T} P_r^T\|_F} \le tol$$

were fulfilled for a prescribed tolerance *tol*. Note that $A^{-1} R_k A^{-T}$ is the residual matrix for the projected Lyapunov equation (2.1). To limit the number of explored variants, we do not pursue this alternative in the following.

4.2. Deflation procedure for (quasi-)loss of rank in the basis. A well-known problem of block Krylov subspace methods is that loss of rank in the basis may occur. Near loss of rank is also problematic, since this influences the final accuracy attainable by the iteration [14].

In the extended block Arnoldi method this drawback is even more severe, since two block recurrences need to make progress. To improve the robustness of the method, we have implemented a deflation strategy that eliminates almost linearly dependent blocks in each of the two sequences, and also ensures that the two blocks are sufficiently linearly independent, so that really new information is added to the space.

Step 3(c) of Algorithm 4.1 is thus replaced by a modified rank revealing type procedure as follows (here ϵ_0 is a fixed threshold chosen a priori):

(i) Set $V_1^{(j)} = V^{(j)}[\, I_m,\, 0\,]^T$ and $V_2^{(j)} = V^{(j)}[\, 0,\, I_m\,]^T$.
(ii) Compute the economical SVD $V_1^{(j)} = U_1 \Sigma_1 W_1^T$, where U_1 and W_1 have orthonormal columns and $\Sigma_1 = \mathrm{diag}(\sigma_{11}, \ldots, \sigma_{m1})$.
(iii) Determine the first j_1 singular values such that $\sigma_{i1} > \epsilon_0 \sigma_{11}$, $i = 1, \ldots, j_1$, and set $\hat{U}_1 = U_1(:, 1 : j_1)$, $\hat{S}_1 = \Sigma_1(1 : j_1, :)W_1^T$.
(iv) Compute the economical SVD $V_2^{(j)} = U_2 \Sigma_2 W_2^T$, where U_2 and W_2 have orthonormal columns and $\Sigma_2 = \mathrm{diag}(\sigma_{12}, \ldots, \sigma_{m2})$.
(v) Determine the first j_2 singular values such that $\sigma_{i2} > \epsilon_0 \sigma_{12}$, $i = 1, \ldots, j_2$, and set $\hat{U}_2 = U_2(:, 1 : j_2)$, $\hat{S}_2 = \Sigma_2(1 : j_2, :)W_2^T$.
(vi) Compute the Gram-Schmidt orthogonalization $[\hat{U}_1, \hat{U}_2]\hat{H} = \hat{V}$.
(vii) Retain only the set $\mathcal{I}$ of columns of $\hat{V}$ whose diagonal elements in the matrix $\hat{H}$ are greater than ϵ_0 and get the final $\hat{V}_{j+1} = \hat{V}(:, \mathcal{I})$ and $H_{j+1,j} = \hat{H}(\mathcal{I}, :)\mathrm{diag}(\hat{S}_1, \hat{S}_2)$.

This procedure shows that the dimensions of $\hat{V}_{j+1}$ and $H_{j+1,j}$ may vary (decrease) as the iteration proceeds. This will be the same for all blocks in $\underline{H}_j$, whose dimensions vary with j. In particular, the number of deflated columns in the even and odd recurrences, i.e., value of j_1 and j_2 at steps (iii) and (v) above, may be different. This structure needs to be taken into account when updating the matrix $\underline{\Phi}_j$. A MATLAB implementation of this technical step is reported in the Appendix.

5. Extended global Arnoldi method. In the extended global Arnoldi method, an F-orthonormal basis matrix $V_k \in \mathbb{R}^{n,2km}$ is constructed whose columns span subspaces

$$\widetilde{\mathcal{K}}_k(A^{-1}E, P_r A^{-1}B) + \widetilde{\mathcal{K}}_k(E^- A, E^- B).$$

First, the matrix V_1 is determined from the global QR decomposition

$$[\, E^- B,\, P_r A^{-1} B\,] = \tilde{V}_1 (S \otimes I_m),$$

where $\tilde{V}_1 = [V_{1,1}, V_{1,2}\,] \in \mathbb{R}^{n,2m}$ is F-orthonormal, i.e., $\tilde{V}_1^T \diamond \tilde{V}_1 = I_2$ and $S = [s_{pq}] \in \mathbb{R}^{2,2}$ is upper triangular. The global QR decomposition can be computed as

$$\begin{array}{lcll} s_{11} & = & \|E^- B\|_F, & V_{1,1} = E^- B / s_{11}, \\ s_{12} & = & \mathrm{trace}(V_{1,1}^T P_r A^{-1} B), & \tilde{V}_{1,2} = P_r A^{-1} B - s_{12} V_{1,1}, \\ s_{22} & = & \|\tilde{V}_{1,2}\|_F, & V_{1,2} = \tilde{V}_{1,2} / s_{22}. \end{array} \tag{5.1}$$

Then in the j-th global Arnoldi step, the block matrix $[\, E^- A V_{j,1},\, A^{-1} E V_{j,2}\,]$ is F-orthonormalized with respect to $V_j = [\tilde{V}_1, \ldots, \tilde{V}_j]$. Similarly to the standard case [16], we get the Arnoldi-like recurrence

$$[\, E^- A V_{j,1},\, A^{-1} E V_{j,2}\,] - \sum_{i=1}^{j} \tilde{V}_i (T_{i,j} \otimes I_m) = \tilde{V}_{j+1} (T_{j+1,j} \otimes I_m)$$

with $T_{i,j} = \tilde{V}_i^T \diamond [\, E^- A V_{j,1},\, A^{-1} E V_{j,2}\,] \in \mathbb{R}^{2,2}$ for $i = 1, \ldots, j+1$. We summarize the extended global Arnoldi method for the PCALE (1.1) in Algorithm 5.1.

ALGORITHM 5.1. *The extended global Arnoldi method for the PCALE.*

Given $E, A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$ and the spectral projectors P_l and P_r, compute an approximate solution X_k of the PCALE (1.1).

1. *Compute the global QR decomposition* $[E^-B,\, P_rA^{-1}B\,] = \tilde{V}_1(S \otimes I_m)$, *where* $\tilde{V}_1 \in \mathbb{R}^{n,2m}$ *is F-orthonormal and* $S = [s_{pq}] \in \mathbb{R}^{2,2}$ *is upper triangular.*
2. *Set* $V_1 := \tilde{V}_1$, $V_{1,1} := \tilde{V}_1[I_m, 0]^T$ *and* $V_{1,2} := \tilde{V}_1[0, I_m]^T$.
3. *FOR* $j = 1, \ldots, k$
   (a) *Compute* $V^{(j)} = [\,E^-AV_{j,1},\, A^{-1}EV_{j,2}\,]$.
   (b) *FOR* $i = 1, \ldots, j$ % *Modified global Gram-Schmidt orthogonal.*
   $T_{i,j} := \tilde{V}_i^T \diamond V^{(j)}$;
   $V^{(j)} := V^{(j)} - \tilde{V}_i(T_{i,j} \otimes I_m)$;
   END FOR
   (c) *Compute the global QR decomposition* $V^{(j)} = \tilde{V}_{j+1}(T_{j+1,j} \otimes I_m)$, *where* $\tilde{V}_{j+1}$ *is F-orthonormal and* $T_{j+1,j} \in \mathbb{R}^{2,2}$ *is upper triangular.*
   (d) *Set* $V_{j+1} := [V_j, \tilde{V}_{j+1}]$, $V_{j+1,1} := \tilde{V}_{j+1}[I_m, 0]^T$, $V_{j+1,1} := \tilde{V}_{j+1}[0, I_m]^T$.
   (e) *Solve the Lyapunov equation*

$$\Psi_j Y_j + Y_j \Psi_j^T = -s_{11}^2 e_1^{2j} (e_1^{2j})^T, \tag{5.2}$$

   where $\Psi_j = V_j^T \diamond (E^-AV_j) \in \mathbb{R}^{2j,2j}$ *and* $e_1^{2j} = [1, 0, \ldots, 0]^T \in \mathbb{R}^{2j}$.
   END FOR
4. *Compute* $X_k = V_k(Y_k \otimes I_m)V_k^T$.

One can show that $\underline{T}_j = [T_{i,l}]_{i=1,\ldots,j+1}^{l=1,\ldots,j}$ and

$$\underline{\Psi}_j = V_{j+1}^T E^- A V_j = \begin{bmatrix} \Psi_j \\ \Psi_{j+1,j} J_{2j,2j-1:2j}^T \end{bmatrix}$$

are both the $2(j+1) \times 2j$ upper block Hessenberg matrices. The following theorem shows that the columns of $\underline{\Psi}_j$ can be determined from $\underline{T}_j$ without additional multiplication with E^-A.

THEOREM 5.2. *Let the matrix* $\underline{T}_j = [T_{i,l}]_{i=1,\ldots,j+1}^{l=1,\ldots,j}$ *with* $T_{i,l} = [t_{pq}]_{p=2i-1,2i}^{q=2l-1,2l}$ *be as in Algorithm* 5.1. *Then the odd columns of* $\underline{\Psi}_j = V_{j+1}^T E^- A V_j = [\,\psi_1, \ldots, \psi_{2j}\,] \in \mathbb{R}^{2(j+1),2j}$ *satisfy*

$$\psi_{2i-1} = \underline{T}_j e_{2i-1}^{2j}, \quad i = 1, \ldots, j,$$

while the even columns of $\underline{\Psi}_j$ *have the form*

$$\psi_2 = \big(s_{11} e_1^{2(j+1)} - s_{12} \underline{T}_j e_1^{2j}\big)/s_{22},$$

$$\psi_{2i} = \big(e_{2(i-1)}^{2(j+1)} - [\psi_1, \ldots \psi_{2i-1}]\underline{T}(1:2i-1, 2(i-1))\big)/t_{2i,2(i-1)}, \quad i = 2, \ldots, j,$$

where e_i^k *denotes the* i*-th column of* I_k *and* s_{il} *are as in* (5.1).

Proof. The result can be proved analogously to [16, Proposition 1]. □

The following theorem shows that the stopping criterion in Algorithm 5.1 can be based on an inexpensive estimation of the residual $E^-R_k(E^-)^T$.

THEOREM 5.3. *Let* $X_k = V_k(Y_k \otimes I_m)V_k^T$ *be an approximate solution of the PCALE* (1.1) *computed by Algorithm* 5.1 *and let*

$$R_k = AX_kE^T + EX_kE^T + P_lBB^TP_l^T$$

be an associated residual. Then

$$\|E^-R_k(E^-)^T\|_F \le \sqrt{2}\,\|(e_{2k+1}^{2(k+1)})^T \underline{\Psi}_k Y_k\|_F =: \delta_{EGA}, \tag{5.3}$$

$$\|R_k\|_F \le \sqrt{2}\,\|E\|_F^2 \|(e_{2k+1}^{2(k+1)})^T \underline{\Psi}_k Y_k\|_F. \tag{5.4}$$

TABLE 5.1
Computational costs for Algorithms 4.1 and 5.1

	Algorithm 4.1	Algorithm 5.1
Step 1	$4nm^2 + 6nm$	$10nm$
Step 3(b)	$(16nm^2 + 2nm)j$	$18nmj$
Step 3(c)	$4nm^2 + 6nm$	$10nm$
Step 3(d)	$200m^3j^3$	$200j^3$

Proof. Estimate (5.3) can be proved analogously to [16, Proposition 3]. Estimate (5.4) follows from $R_k = EE^- R_k (E^-)^T E^T$. □

By solving the Lyapunov equation (5.2) for the Cholesky factor Z_j of $Y_j = Z_j Z_j^T$, $j = 1, \ldots, k$, and computing the SVD

$$Z_k = [\, U_{1k},\, U_{2k}\,]\mathrm{diag}(\Sigma_{1k}, \Sigma_{2k})[\, W_{1k},\, W_{2k}\,]^T$$

with $\|\Sigma_{2k}\|_2 \le \epsilon \|\Sigma_{1k}\|_2$, the approximate solution of the PCALE (1.1) can be determined in the factored form $X \approx ZZ^T$, where $Z = V_k\big((U_{1k}\Sigma_{1k}) \otimes I_m\big)$.

We now compare the computational complexity of the extended block and global Arnoldi methods. The computation of the QR decomposition via the Gram-Schmidt orthogonalization in Steps 1 and 3(c) of Algorithm 4.1 costs $4nm^2+6nm$ flops, whereas the global QR decomposition in Algorithm 5.1 costs $10nm$ flops disregarding the computation of E^-B and $P_r A^{-1}B$ in both algorithms. The modified block Gram-Schmidt orthogonalization in Algorithm 4.1 requires $(16nm^2 + 2nm)j$ flops at the j-th iteration, while the computation of the modified global Gram-Schmidt orthogonalization in Algorithm 5.1 costs $18nmj$ flops. The computation of $V^{(j)}$ at Step 3(a) in both algorithms costs the same amount of flops. Finally, solving the Lyapunov equations (4.4) and (5.2) using the Bartels-Stewart or Hammarling method costs at most $200m^3j^3$ and $200j^3$, respectively. Here, we do not take into account the (minor) computational cost for generating the matrices Φ_j and Ψ_j. We collect the counting of costs for the extended block and global Arnoldi algorithms in Table 5.1.

The values in the table show that for the same number of iterations, Step 3(d) may be significantly more expensive for Algorithm 4.1 than for Algorithm 5.1, if the number of columns in B is large. In fact, it was suggested already in [34] to perform Step 3(d) periodically, and not at every iteration, especially when convergence is slow, i.e., j is large. We should notice, however, that Algorithm 4.1 may converge significantly earlier than Algorithm 5.1, that is fewer iterations need to be performed, thus overcoming the higher cost of solving the reduced Lyapunov equation at Step 3(d).

6. ADI-preconditioned Krylov subspace methods. In order to accelerate the convergence of the global Arnoldi method, a preconditioning technique based on the ADI method was proposed in [19] for standard Lyapunov equations. This technique can also be extended to the PCALE (1.1).

First, we briefly describe the ADI method that was previously proposed for standard Lyapunov equations in [5, 24, 25, 27] and then generalized to projected Lyapunov equations in [42]. The generalized ADI iteration for the PCALE (1.1) is given by

$$\begin{aligned} (E + \tau_k A)X_{k-1/2}A^T + AX_{k-1}(E - \tau_k A)^T &= -P_l BB^T P_l^T, \\ (E + \overline{\tau}_k A)X_k^T A^T + AX_{k-1/2}^T(E - \overline{\tau}_k A)^T &= -P_l BB^T P_l^T \end{aligned} \tag{6.1}$$

with an initial matrix $X_0 = 0$ and shift parameters $\tau_1, \ldots, \tau_k \in \mathbb{C}_-$. If the pencil $\lambda E - A$ is stable, then X_k converges towards the solution of the PCALE (1.1). The rate of convergence depends strongly on the choice of the shift parameters. The optimal shift parameters can be determined by solving the generalized ADI minimax problem

$$\{\tau_1, \ldots, \tau_q\} = \underset{\{\tau_1,\ldots,\tau_q\}\in\mathbb{C}_-}{\arg\min} \max_{t\in \mathrm{Sp}_f(E,A)} \frac{|(1-\overline{\tau}_1 t)\cdot \ldots \cdot (1-\overline{\tau}_q t)|}{|(1+\tau_1 t)\cdot \ldots \cdot (1+\tau_q t)|},$$

where $\mathrm{Sp}_f(E, A)$ denotes the finite spectrum of the pencil $\lambda E - A$. Similarly to [27], the suboptimal ADI parameters can be obtained from a set of largest and smallest in modulus approximate finite eigenvalues of $\lambda E - A$ computed by an Arnoldi procedure. Other parameter selection techniques developed for standard Lyapunov equations [32, 43] can also be used for the PCALE (1.1).

A low-rank approximation to the solution of the PCALE (1.1) can be computed in factored form $X \approx Z_k Z_k^T$ using a low-rank version of the ADI method (LR-ADI) as presented in Algorithm 6.1.

ALGORITHM 6.1. *The generalized LR-ADI for the PCALE.*

Given $E, A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$, projector P_l, parameters $\tau_1, \ldots, \tau_q \in \mathbb{C}_-$, compute a low-rank approximation $X \approx Z_k Z_k^T$ to the solution of the PCALE (1.1).

1. *Compute $Z^{(1)} = \sqrt{-2Re(\tau_1)}\,(E + \tau_1 A)^{-1} P_l B$ and set $Z_1 := Z^{(1)}$.*
2. *FOR $j = 2, 3, \ldots, k$*

$$\begin{aligned} Z^{(j)} &= \sqrt{\frac{Re(\tau_j)}{Re(\tau_{j-1})}}\left(I - (\overline{\tau}_{j-1} + \tau_j)(E + \tau_j A)^{-1} A\right) Z^{(j-1)}; \\ Z_j &= [\, Z_{j-1},\ Z^{(j)}\,]; \end{aligned}$$

END FOR

The ADI iteration can be stopped as soon as $\|R_k\|_F / \|P_l B B^T P_l^T\|_F \le tol$ with $R_k = E Z_k Z_k^T A^T + A Z_k Z_k^T E^T + P_l B B^T P_l^T$. If the number of shift parameters is smaller than the number of iterations required to attain a prescribed tolerance, then we reuse these parameters in a cyclic manner. Note that computing the normalized residuals $\eta(Z_k)$ even via the efficient methods proposed in [27, 32] can still be quite expensive for large-scale problems. The stopping criterion in Algorithm 6.1 can also be based on the condition $\|Z^{(k)}\|_F / \|Z_k\|_F \le tol$ which is much cheaper to verify than that based on the residual. However, in this case, more iterations are usually required to achieve the tolerance *tol*.

Equation (1.1) can be written in an equivalent form as a projected discrete-time Lyapunov equation

$$A_\ell X A_\ell^T - X = -Z_\ell Z_\ell^T, \qquad X = P_r X P_r^T, \tag{6.2}$$

where $A_\ell = P_r (E + \tau_\ell A)^{-1}(E - \overline{\tau}_\ell A) \cdot \ldots \cdot (E + \tau_1 A)^{-1}(E - \overline{\tau}_1 A)$ and $Z_\ell \in \mathbb{R}^{n,\ell m}$ is the ℓ-th ADI iterate from Algorithm 6.1. Then we can solve this equation using the block or global Arnoldi method. In the block Arnoldi method, the solution of (6.2) is approximated by $X_k = V_k \tilde{Y}_k V_k^T$, where $\tilde{Y}_k$ satisfies the discrete-time Lyapunov equation

$$\tilde{H}_k \tilde{Y}_k \tilde{H}_k^T - \tilde{Y}_k = -\tilde{B}_k \tilde{B}_k^T$$

with a block upper Hessenberg matrix $\tilde{H} = V_k^T A_\ell V_k = [\tilde{H}_{ij}] \in \mathbb{R}^{k\ell m, k\ell m}$ and $\tilde{B}_k = V_k^T Z_\ell \in \mathbb{R}^{k\ell m, \ell m}$. The norm of the residual $R_{k,\ell} = A_\ell X_k A_\ell^T - X_k + Z_\ell Z_\ell^T$ of the Lyapunov equation (6.2) can be determined as

$$\|R_{k,\ell}\|_F = \|\tilde{H}_{k+1,k} J_k^T \tilde{Y} [\, \sqrt{2}\tilde{H}_k^T,\ J_k \tilde{H}_{k+1,k}^T \,]\|_F =: \delta_{ADI-BA}, \tag{6.3}$$

where J_k denotes a matrix of the last ℓm columns of the identity matrix $I_{k\ell m}$, see [18] for details.

In the global Arnoldi method, the solution of the Lyapunov equation (6.2) is approximated by $X_k = V_k(\tilde{Y}_k \otimes I_{\ell m}) V_k^T$, where $\tilde{Y}_k$ satisfies the discrete-time Lyapunov equation

$$\tilde{T}_k \tilde{Y}_k \tilde{T}_k^T - \tilde{Y}_k = -\|Z_\ell\|_F^2 e_1 e_1^T$$

with an upper Hessenberg matrix $\tilde{T}_k = V_k^T \diamond (A_\ell V_k) = [\tilde{t}_{ij}] \in \mathbb{R}^{k,k}$. Similarly to [19], we obtain the following estimate of the residual

$$\|R_{k,\ell}\|_F \le \tilde{t}_{k+1,k} \sqrt{2\|\tilde{T}_k \tilde{Y}_k e_k\|_2^2 + \tilde{t}_{k+1,k}^2 (e_k^T \tilde{Y}_k e_k)^2} =: \delta_{ADI-GA}, \tag{6.4}$$

where $e_k = [0, \ldots, 0, 1]^T \in \mathbb{R}^k$. Thus, the iteration can be stopped as soon as $\delta_{ADI-GA}/\|Z_\ell Z_\ell^T\|_F$ exceeds a prescribed tolerance tol.

The Lyapunov equation (6.2) can also be solved using the extended block (or global) Arnoldi method. It should, however, be noted that if the ADI shift parameters τ_j are close to finite eigenvalues of $\lambda E - A$, then the matrix A_ℓ is ill-conditioned. In this case, small roundoff errors may have a dramatic effect on the convergence of the algorithm.

7. Numerical examples. In this section we present some results of numerical experiments to demonstrate the properties of the presented iterative methods for projected Lyapunov equations. Comparisons with respect to the computational complexity and accuracy were performed for the generalized LR-ADI method, the extended block Arnoldi method (EBA), the extended global Arnoldi method (EGA), the ADI-preconditioned block Arnoldi method (ADI(ℓ)-BA) and the ADI-preconditioned global Arnoldi method (ADI(ℓ)-GA) with ℓ ADI-preconditioning steps. The PCALE to be solved arises in balanced truncation model reduction of the descriptor system

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \tag{7.1}$$

where E, $A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$, $C \in \mathbb{R}^{p,n}$, $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^m$ is the input and $y \in \mathbb{R}^p$ is the output. All computations were done on Intel Core 2 CPU 6400 @ 2.13GHz RAM 2GB using MATLAB 7.9 with machine precision $\varepsilon \approx 2.22 \times 10^{-16}$.

For the LR-ADI iteration, the ADI parameters were determined using the heuristic procedure [27] with the search parameters (q, q_l, q_s), where q is the number of the ADI shifts, q_l is the number of the largest in modulus finite eigenvalues and q_s is the number of the smallest in modulus finite eigenvalues of $\lambda E - A$. These eigenvalues were computed by an Arnoldi process applied to $E^- A$ and $A^{-1}E$, respectively. In all cases we only report the best obtained results among the many choices of (q, q_l, q_s) we tested. In fact, this step required significant tuning as the performance of these methods varied wildly with these values.

As stopping criteria in the different methods, unless stated otherwise, we used the following conditions

$$\begin{aligned}
&\text{LR-ADI:} && \frac{\|R_k\|_F}{\|P_l B B^T P_l^T\|_F} \le tol,\\
&\text{EBA:} && \frac{\|E^- R_k (E^-)^T\|_F}{\|E^- B B^T (E^-)^T\|_F} = \frac{\delta_{EBA}}{\|E^- B B^T (E^-)^T\|_F} \le tol,\\
&\text{EGA:} && \frac{\|E^- R_k (E^-)^T\|_F}{\|E^- B B^T (E^-)^T\|_F} \le \frac{\delta_{EGA}}{\|E^- B B^T (E^-)^T\|_F} \le tol,\\
&\text{ADI}(\ell)\text{-BA:} && \frac{\|R_{k,\ell}\|_F}{\|Z_\ell Z_\ell^T\|_F} = \frac{\delta_{ADI-BA}}{\|Z_\ell Z_\ell^T\|_F} \le tol,\\
&\text{ADI}(\ell)\text{-GA:} && \frac{\|R_{k,\ell}\|_F}{\|Z_\ell Z_\ell^T\|_F} \le \frac{\delta_{ADI-GA}}{\|Z_\ell Z_\ell^T\|_F} \le tol,
\end{aligned}$$

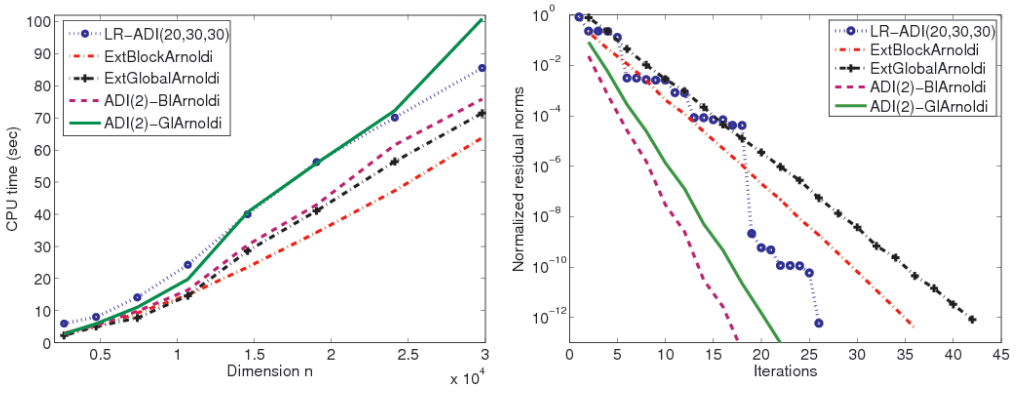
where δ_{EBA}, δ_{EGA}, δ_{ADI-BA} and δ_{ADI-GA} are defined in (4.8), (5.3), (6.3) and (6.4), respectively. In the Krylov subspace methods, the reduced Lyapunov equation was solved at every other iteration.

EXAMPLE 7.1. Consider the 2D instationary Stokes equation describing the flow of an incompressible fluid in a square domain, see [33, Section 3.7.1] for details. The spatial discretization of this equation by the finite volume method on a uniform staggered grid leads to the descriptor system (7.1) with a matrix pencil $\lambda E - A$ which has real finite eigenvalues only. The matrix $B \in \mathbb{R}^{n,5}$ results from the distributed control.

In Figure 7.1(a), we compare the computational time of all five methods for problems of different dimensions obtained by varying the mesh resolution. Note that the cost for computing the ADI parameters is included in the overall cost of the LR-ADI method. The iterations were stopped when either the normalized residual norm or its upper bound (in case of the global Arnoldi iteration) exceeded the tolerance $tol = 10^{-12}$. Figure 7.1(a) shows that the EBA method outperforms all other methods in terms of CPU time, while the ADI(2)-GA method is the most expensive for larger problems.

We investigate further this example by presenting the convergence history for the different methods for the problem dimension $n = 29799$ in Figure 7.1(b). One can see that although the ADI(2)-BA and ADI(2)-GA methods require fewer iterations than other methods, it does not imply that they are less expensive in terms of execution time. This can be explained by the fact that after ADI preconditioning with $\ell = 2$ the global Arnoldi algorithm is applied to the Lyapunov equation (6.2) with $Z_2 \in \mathbb{R}^{n,10}$ and the large number of columns in Z_2 makes the ADI(2)-GA method expensive. In addition to timings, Table 7.1 reports the final normalized residuals, the number of iterations, the dimension of the solution space and the rank of the approximate solution after column compression with the threshold $\epsilon = 10^{-12}$ for the different methods. Comparing the EBA and EGA methods, we can conclude that although the EBA method requires more operations at every iteration step, it converges in fewer iterations than the EGA method, resulting in less computation time. Moreover, the rank of the approximate solution is also in favor of the EBA method.

EXAMPLE 7.2. Consider a constrained damped mass-spring system with g masses, see [26]. The i-th mass of weight m_i is connected to the $(i+1)$-st mass by a spring and



(a) (b)

FIG. 7.1. *Stokes equation. (a) CPU time for problems of different dimensions; (b) Convergence history for problem of dimension* $n = 29799$.

TABLE 7.1
Stokes equation: performance of the different iterative methods

	LR-ADI	EBA	EGA	ADI(2)-BA	ADI(2)-GA
CPU time (sec)	85.53	**63.82**	71.48	75.83	100.82
Residuals/Estimates	$5.93 \cdot 10^{-13}$	$3.82 \cdot 10^{-13}$	$8.25 \cdot 10^{-13}$	$\mathbf{4.67 \cdot 10^{-14}}$	$9.77 \cdot 10^{-14}$
Number of iterations	26	36	42	**18**	22
Dim. of solution space	**130**	360	420	180	220
Rank of solution	130	**85**	105	**85**	130

a damper with constants k_i and d_i, respectively, and also to the ground by another spring and damper with constants δ_i and κ_i, respectively. Additionally, we assume that the first mass is connected to the last one by a rigid bar and it can be influenced by a control. The vibration of this system is described by the descriptor system (7.1) with a column vector B and a matrix pencil $\lambda E - A$ which has complex finite eigenvalues. The system parameters are $g = 20862$, $m_1 = \ldots = m_g = 100$ and

$$\begin{aligned} k_1 = \ldots = k_{g-1} = k = 2, \qquad & \kappa_1 = \ldots = \kappa_g = \kappa = 4, \\ d_1 = \ldots = d_{g-1} = d = 5, \qquad & \delta_1 = \ldots = \delta_g = \delta = 10. \end{aligned}$$

Problem (1.1) is thus of dimension $n = 41725$. Figure 7.2(a) shows the normalized residuals $\|R_k\|_F / \|P_l B B^T P_l^T\|_F$ for the LR-ADI method as well as

$$\eta_1(X_k) = \frac{\|E^- R_k (E^-)^T\|_F}{\|E^- B B^T (E^-)^T\|_F} \qquad \text{and} \qquad \eta_2(X_k) = \frac{\|A^{-1} R_k A^{-T}\|_F}{\|P_r A^{-1} B B^T A^{-T} P_r^T\|_F}$$

for the EBA method based on the reduced Lyapunov equations (4.4) and (4.11), respectively (cf. the final remark in Section 4.1). We do not consider here the EGA method since it coincides with the EBA method. One can see that both methods converge very quickly. However, the LR-ADI method delivers the approximate solution in 13.47 sec, while the computation of the solution using the EBA method takes in both variants 2.19 sec only. Note that in this example, the second version of the EBA method is preferable, since it requires fewer iterations to meet the chosen stopping criterion.

As a second test with these data, we consider changing the damping constants to $d = 0.5$ and $\delta = 1$. The convergence of the LR-ADI method becomes much slower and

the EBA method does not provide a satisfactory solution for a tolerance $tol = 10^{-10}$, see Figure 7.2(b). We should add, however, that this behavior is highly expected, since the matrix $E^- A$ appears not to be dissipative. The residual curves for the EBA method are broken after 24 and 26 iterations, respectively, since in both cases the subsequent reduced Lyapunov equations were unstable.

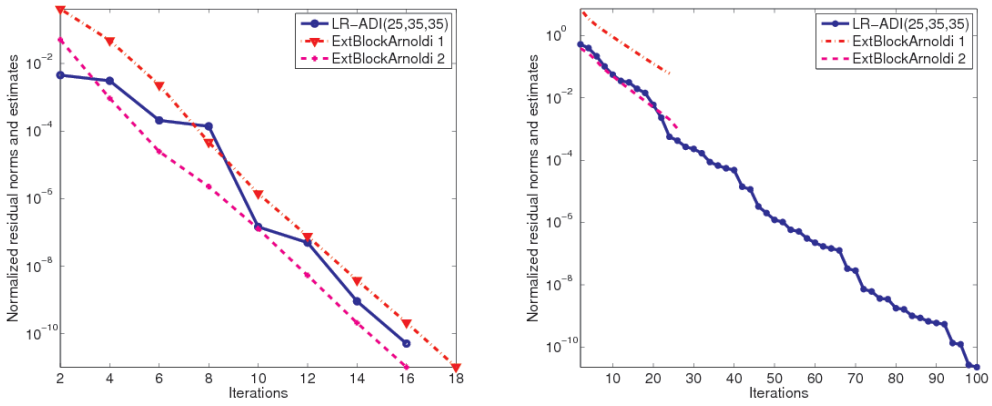


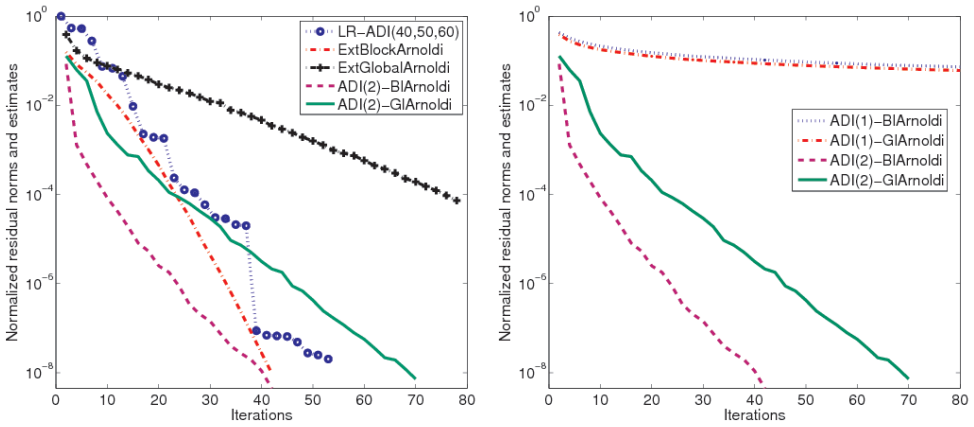
FIG. 7.2. *Mechanical systems. Convergence history for the problems with different damping parameters: (a) $d = 5$, $\delta = 10$ and (b) $d = 0.5$, $\delta = 1$.*

EXAMPLE 7.3. We consider now RC circuits that can be modeled via modified nodal analysis by the descriptor system (7.1). First, we investigate the importance of the deflation for systems with many inputs. For an RC circuit with 109 capacitors, 2910 resistors and 4 voltage sources, we have the problem of dimension $n = 540$ and $m = 4$. For this problem, the EBA method without deflation does not converge. However, incorporating the deflation procedure as described in Section 4.2 with a threshold $\epsilon_0 = 10^{-7}$ completely cures the problem: not only the EBA method converges, but it outperforms all other methods, cf. Table 7.2.

In this table, we also present the results for another RC circuit with 12005 capacitors, 12016 resistors and 8 voltage sources. The corresponding system has dimension $n = 12021$ and $m = 8$. Figure 7.3(a) shows the residual curves for the different methods. We see that although the EBA and ADI(2)-BA methods have both the same number of iterations, the first method is less expensive with respect to time than the second one.

Finally, we compare in this example the ADI(ℓ)-BA and ADI(ℓ)-GA methods for different ℓ. Figure 7.3(b) shows the normalized residuals for $\ell = 1$ and 2. One can see that the preconditioning with two ADI steps improves the performance of the block and global Arnoldi methods considerably. A further increasing of ℓ results, usually, in more expensive computations. We also observed that both methods are very sensitive to the choice of the ADI parameters in the preconditioning step.

8. Conclusion. In this paper we adapted known methods for solving large-scale Lyapunov equations to the case of the projected Lyapunov equation, when a low-rank solution is sought. These methods are based on the block and global Arnoldi processes combined with the extended Krylov subspace approach and the ADI iteration. For the extended block Arnoldi method, a new deflation procedure was devised to overcome



(a) (b)

FIG. 7.3. *Electrical circuit, $n = 12021$. (a) Convergence history for different methods. (b) Convergence history for the ADI(ℓ)-BA and ADI(ℓ)-GA methods with $\ell = 1$ and 2.*

TABLE 7.2
Electrical circuits: performance of the different iterative methods

	LR-ADI	EBA+Defl	EGA	ADI(2)-BA	ADI(2)-GA
$n = 540, \quad m = 4$					
CPU time (sec)	1.17	**0.48**	0.84	0.96	1.66
Residuals/Estimates	$7.34 \cdot 10^{-11}$	$\mathbf{1.77 \cdot 10^{-13}}$	$5.28 \cdot 10^{-11}$	$3.35 \cdot 10^{-13}$	$4.01 \cdot 10^{-11}$
Number of iterations	31	18	26	**14**	26
Dim. of solution space	124	**80**	208	112	208
Rank of the solution	124	**28**	68	**28**	80
$n = 12021, \quad m = 8$					
CPU time (sec)	144.89	**61.28**	147.20	79.76	108.75
Residuals/Estimates	$\mathbf{3.40 \cdot 10^{-9}}$	$9.80 \cdot 10^{-9}$	$7.31 \cdot 10^{-5}$	$4.52 \cdot 10^{-9}$	$7.30 \cdot 10^{-9}$
Number of iterations	54	**42**	78	**42**	70
Dim. of solution space	432	**672**	1248	**672**	1120
Rank of the solution	432	**154**	240	**154**	288

possible (near)breakdown problems during the iteration, significantly increasing the robustness of the method. Note that the presence of singular coefficient matrices highly increases the occurrence of near breakdown, making the implementation of a deflation strategy mandatory.

The reported numerical experiments clearly show the competitiveness of the developed parameter-free Krylov subspace methods with respect to the classical LR-ADI method in terms of computation time, accuracy and memory requirements, when applied to several problems of distinct structure. We would like to stress that the parameter tuning in the preprocessing of ADI may be extremely tricky, as the ADI performance is dramatically affected by non-optimal parameters.

Within the extended family, our experiments showed that the extended block Arnoldi method usually requires fewer iterations than the extended global Arnoldi method to achieve a prescribed residual accuracy. This results in savings in both memory and computation time.

We did not consider here a combination of the ADI method and Galerkin projection as it was proposed in [6]. In all our experiments, this method was more expensive than Krylov subspace methods. A venue in a similar direction we would like to explore

is the use of general rational Krylov subspace methods, which combine the strength of rational functions with that of projection type schemes, see [10].

Further investigations would include an extension of the convergence results for standard Lyapunov equations [22, 35] to projected Lyapunov equations as well as computational issues such as restarting, memory-reducing strategies and preconditioning techniques. We should mention that there are some difficult problems for which none of the existing iterative methods provides an approximate solution with an acceptable accuracy although the exact solution has low numerical rank. This implies the necessity to continue research to develop even more robust solvers for large-scale projected Lyapunov equations.

Acknowledgments. This work was started while the second author was visiting the TU Berlin in the winter semester 2009. The first author was supported by the DFG Research Center MATHEON in Berlin.

Appendix A. In this appendix, we reproduce the code for performing the deflation step in the extended block Arnoldi method. The procedure replaces Step 3(c) of Algorithm 4.1, and it starts with: `Up` $= V^{(j)}$, `nu1`, `nu2` the current number of blocks in the two sequences, `defltol` the chosen deflation tolerance, `js` the current basis size, and `AUp2` $= E^{-}AV_{1,2}$. Note that certain indices are updated at each iteration using the new dimensions of the two blocks. In particular, these are the initial settings for some of them: `j=1; jms=1; js=nu1+nu2; jm=nu1; jxm=nu2; jx1m=1`.

```
[uu1,ss1,vv1]=svd(Up(:,1:nu1),0);
[uu2,ss2,vv2]=svd(Up(:,nu1+1:nu1+nu2),0);
if (nu1>1),
  k01=find(diag(ss1/ss1(1))>=defltol);
elseif (nu1==0)
  k01=[];
else
  k01=1;
end
if (nu2>1),
  k02=find(diag(ss2/ss2(1))>=defltol);
elseif (nu2==0)
  k02=[];
else
  k02=1;
end
P=[uu1(:,k01),uu2(:,k02)];
rr=blkdiag(ss1(k01,:)*vv1',ss2(k02,:)*vv2');
[S,Up]=gram_sh(P);
Si=inv(S);

id=find(diag(Si)<defltol); Si(id,:)=[];
rr=Si*rr; Up(:,id)=[];
id1=find(id<=length(k01));    id2=find(id>length(k01));
if (~isempty(id1)),k01(id(id1))=[];end
if (~isempty(id2)),k02(id(id2)-length(k01))=[];end

nu1old=nu1;      nu2old=nu2;
nu1=length(k01); nu2=length(k02);
nu =nu1+nu2;
rrn=rr(:,[k01;nu1old+k02]);
```

```
H(js+1:js+nu,jms:js)=rr;
U(:,js+1:js+nu)=Up;
I=speye(js+nu);

if (nu>0)
   if (j==1),
     kk2=nu1old+nu2old+nu1;
     l(1:kk2,1:nu2old)=U(:,1:kk2)'*AUp2;
   else
    l(1:js+nu,jxm-nu2old+1:jxm)= ...
     l(1:js+nu,jxm-nu2old+1:jxm)+H(1:js+nu,jms:jms-1+nu1old)*rho;
   end
end

% even columns
Phi(1:js+nu1,jms+nu1old:jms-1+nu1old+nu2old)= ...
   l(1:js+nu1,jxm-nu2old+1:jxm);
% odd columns
Phi(1:js+nu,jms:jms-1+nu1old)=H(1:js+nu,jms:jms-1+nu1old);

if (nu>0)
    g1=I(1:js+nu,js-nu2old+1:js)- ...
        Phi(1:js+nu,1:js)*H(1:js,js-nu2old+1:js);
    g1=g1(:,k02);
    l(1:js+nu,jxm+1:jxm+nu2)=g1*pinv(rrn(nu1+1:nu,nu1+1:nu));
    rho=-rrn(1:nu1,nu1+1:nu)*pinv(rrn(nu1+1:nu,nu1+1:nu));
end

j  =j+1;  jxm=jxm+nu2; jx1m=jx1m+nu2;
jms=js+1; jm =js+nu1;  js  =js+nu1+nu2;
```

REFERENCES

[1] R. Bartels and G. Stewart, *Solution of the equation $AX + XB = C$*, Comm. ACM, 15 (1972), pp. 820–826.

[2] U. Baur and P. Benner, *Factorized solution of Lyapunov equations based on hierarchical matrix arithmetic*, Computing, 78 (2006), pp. 211–234.

[3] D. Bender, *Lyapunov-like equations and reachability/observability Gramians for descriptor systems*, IEEE Trans. Automat. Control, 32 (1987), pp. 343–348.

[4] P. Benner, J. Claver, and E. Quintana-Ortí, *Parallel distributed solvers for large stable generalized Lyapunov equations*, Parallel Process. Lett., 9 (1999), pp. 147–158.

[5] P. Benner, J.-R. Li, and T. Penzl, *Numerical solution of large Lyapunov equations, Riccati equations, and linear-quadratic control problems*, Numer. Linear Algebra Appl., 15 (2008), pp. 755–777.

[6] P. Benner, R.-C. Li, and N. Truhar, *On the ADI method for Sylvester equations*, J. Comput. Appl. Math., 233 (2009), pp. 1035–1045.

[7] P. Benner and E. Quintana-Ortí, *Solving stable generalized Lyapunov equations with the matrix sign function*, Numerical Algorithms, 20 (1999), pp. 75–100.

[8] P. Benner and V. Sokolov, *Partial realization of descriptor systems*, Systems Control Lett., 55 (2006), pp. 929–938.

[9] S. Campbell and C. Meyer, *Generalized Inverses of Linear Transformations*, Dover Publications, New York, 1979.

[10] V. Druskin and V. Simoncini, *daptive rational Krylov subspaces for large-scale dynamical systems*, Systems Control Lett., 60 (2011), pp. 546–560.

[11] F. Gantmacher, *Theory of Matrices*, Chelsea Publishing Company, New York, 1959.

[12] L. Grasedyck, W. Hackbusch, and B. N. Khoromskij, *Solution of large scale algebraic matrix Riccati equations by use of hierarchical matrices*, Computing, 70 (2003), pp. 121–165. 10.1007/s00607-002-1470-0.

[13] S. Gugercin, D. Sorensen, and A. Antoulas, *A modified low-rank Smith method for large-scale Lyapunov equations*, Numerical Algorithms, 32 (2003), pp. 27–55.

[14] M. Gutknecht, *Block Krylov space methods for linear systems with multiple right-hand sides: An introduction*, in Modern Mathematical Models, Methods and Algorithms for Real World Systems, A. Siddiqi, I. Duff, and O. Christensen, eds., Anamaya Publishers, New Delhi, India, 2007, pp. 420–447.

[15] S. Hammarling, *Numerical solution of the stable non-negative definite Lyapunov equation*, IMA J. Numer. Anal., 2 (1982), pp. 303–323.

[16] M. Heyouni, *Extended Arnoldi methods for large low-rank Sylvester matrix equations*, Appl. Numer. Math., 60 (2010), pp. 1171–1182.

[17] D. Hu and L. Reichel, *Krylov-subspace methods for the Sylvester equation*, Linear Algebra Appl., 172 (1992), pp. 283–313.

[18] I. Jaimoukha and E. Kasenally, *Krylov subspace methods for solving large Lyapunov equations*, SIAM J. Numer. Anal., 31 (1994), pp. 227–251.

[19] K. Jbilou, *ADI preconditioned Krylov methods for large Lyapunov matrix equations*, Linear Algebra Appl., 432 (2010), pp. 2473–2485.

[20] K. Jbilou, A. Messaoudi, and H. Sadok, *Global FOM and GMRES algorithms for matrix equations*, Appl. Numer. Math., 31 (1999), pp. 49–63.

[21] K. Jbilou and A. Riquet, *Projection methods for large Lyapunov matrix equations*, Linear Algebra Appl., 415 (2006), pp. 344 – 358.

[22] L. Knizhnerman and V. Simoncini, *Convergence analysis of the extended Krylov subspace method for the Lyapunov equation*, Numer. Math., 118 (2011), pp. 567–586.

[23] V. Larin and F. Aliev, *Construction of square root factor for solution of the Lyapunov matrix equation*, Systems Control Lett., 20 (1993), pp. 109–112.

[24] J.-R. Li and J. White, *Low rank solution of Lyapunov equations*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 260–280.

[25] A. Lu and E. Wachspress, *Solution of Lyapunov equations by alternating direction implicit iteration*, Comput. Math. Appl., 21 (1991), pp. 43–58.

[26] V. Mehrmann and T. Stykel, *Balanced truncation model reduction for large-scale systems in descriptor form*, in Dimension Reduction of Large-Scale Systems, P. Benner, V. Mehrmann, and D. Sorensen, eds., vol. 45 of Lecture Notes in Computational Science and Engineering, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 83–115.

[27] T. Penzl, *A cyclic low-rank Smith method for large sparse Lyapunov equations*, SIAM J. Sci. Comput., 21 (1999/2000), pp. 1401–1418.

[28] T. Reis and T. Stykel, *PABTEC: Passivity-preserving balanced truncation for electrical circuits*, IEEE Trans. Computer-Aided Design Integr. Circuits Syst., 29 (2010), pp. 1354–1367.

[29] J. Roberts, *Linear model reduction and solution of the algebraic Riccati equation by use of the sign function*, Internat. J. Control, 32 (1980), pp. 677–687. (Reprint of Technical Report TR-13, CUED/B-Control, Engineering Department, Cambridge University, 1971).

[30] Y. Saad, *Numerical solution of large Lyapunov equations*, in Signal Processing, Scattering, Operator Theory, and Numerical Methods (Amsterdam, 1989), M. Kaashoek, J. V. Schuppen, and A. Ran, eds., Birkhäuser, Boston, MA, 1990, pp. 503–511.

[31] ———, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, MA, 1996.

[32] J. Sabino, *Solution of Large-Scale Lyapunov Equations via the Block Modified Smith Method*, ph.D. thesis, Rice University, Houston, 2006.

[33] M. Schmidt, *Systematic Discretization of Input/Output Maps and other Contributions to the Control of Distributed Parameter Systems*, Ph.D. thesis, Institut für Mathematik, Technische Universität Berlin, Berlin, Germany, 2007.

[34] V. Simoncini, *A new iterative method for solving large-scale Lyapunov matrix equations*, SIAM J. Sci. Comput., 29 (2007), pp. 1268–1288.

[35] V. Simoncini and V. Druskin, *Convergence analysis of projection methods for the numerical solution of large Lyapunov equations*, SIAM J. Numer. Anal., 47 (2009), pp. 828–843.

[36] V. Sokolov, *Contributions to the Minimal Realization Problem for Descriptor Systems*, Ph.D. thesis, Fakultät für Mathematik, Technische Universität Chemnitz, Germany, 2006.

[37] T. Stykel, *Numerical solution and perturbation theory for generalized Lyapunov equations*, Linear Algebra Appl., 349 (2002), pp. 155–185.

[38] ———, *Stability and inertia theorems for generalized Lyapunov equations*, Linear Algebra Appl., 355 (2002), pp. 297–314.

[39] ———, *Gramian-based model reduction for descriptor systems*, Math. Control Signals Systems, 16 (2004), pp. 297–319.

[40] ———, *On some norms for descriptor systems*, IEEE Trans. Automat. Control, 51 (2006), pp. 842–847.

[41] ———, *A modified matrix sign function method for projected Lyapunov equations*, Syst. Contr. Lett., 56 (2007), pp. 695–701.

[42] ———, *Low-rank iterative methods for projected generalized Lyapunov equations*, Electron. Trans. Numer. Anal., 30 (2008), pp. 187–202.

[43] E. Wachspress, *The ADI minimax problem for complex spectra*, in Iterative Methods for Large Linear Systems, D. Kincaid and L. Hayes, eds., Academic Press, 1990, pp. 251–271.