



MAX-PLANCK-GESELLSCHAFT

Peter Benner

Tatjana Stykel

**Numerical Solution of Projected Algebraic
Riccati Equations**



MAX-PLANCK-INSTITUT
FÜR DYNAMIK KOMPLEXER
TECHNISCHER SYSTEME
MAGDEBURG

**Max Planck Institute Magdeburg
Preprints**

MPIMD/13-06

May 24, 2013

Impressum:

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg

Publisher:

Max Planck Institute for Dynamics of Complex
Technical Systems

Address:

Max Planck Institute for Dynamics of
Complex Technical Systems
Sandtorstr. 1
39106 Magdeburg

www.mpi-magdeburg.mpg.de/preprints

Numerical Solution of Projected Algebraic Riccati Equations

Peter Benner* Tatjana Stykel†

May 24, 2013

Abstract

We consider the numerical solution of projected algebraic Riccati equations using Newton's method. Such equations arise, for instance, in model reduction of descriptor systems based on positive real and bounded real balanced truncation. We also discuss the computation of low-rank Cholesky factors of the solutions of projected Riccati equations. Numerical examples are given that demonstrate the properties of the proposed algorithms.

Keywords. Projected Riccati equation, projected Lyapunov equation, Newton's method, alternating direction implicit (ADI) method.

AMS subject classifications. 15A22, 15A24, 65F99, 65H10, 93B40.

1 Introduction

Consider generalized algebraic Riccati equations of the form

$$H + FXE^T + EXF^T + EXGXE^T = 0, \quad (1)$$

where $E, F, G, H \in \mathbb{R}^{n,n}$ are given matrices and $X \in \mathbb{R}^{n,n}$ is unknown. If E is nonsingular, then (1) can be transformed into a standard Riccati equation with $E = I_n$. Such equations arise in many control problems for dynamical systems including the linear-quadratic optimal regulator problem, $\mathbb{H}_2/\mathbb{H}_\infty$ controller design, spectral factorization and balancing-related model reduction, e.g., [19, 21, 28, 31]. The generalized Riccati equation (1) with singular E occurs in control problems for differential-algebraic equations or descriptor systems [3, 31, 45]. Unfortunately, the analysis of such an equation is more complicated compared to the standard case. In the literature, different types of generalized Riccati equations have been introduced for descriptor systems, e.g., [25, 26, 36, 50]. However, most of them are restricted to index one problems.

In this paper, we study the projected algebraic Riccati equation (PARE) of the form

$$P_l H P_l^T + F X E^T + E X F^T + E X G X E^T = 0, \quad X = P_r X P_r^T, \quad (2)$$

where the pencil $\lambda E - F$ is assumed to be regular, i.e., $\det(\lambda E - F) \neq 0$ for some $\lambda \in \mathbb{C}$, and P_r and P_l are the spectral projectors onto the right and left deflating subspaces of $\lambda E - F$ corresponding to the finite eigenvalues along the right and left deflating subspaces corresponding to the eigenvalue at infinity. We will also assume that G and H are both symmetric and positive semidefinite. Such equations play a fundamental role in positive real and bounded real balanced truncation model reduction of descriptor systems [35, 36].

*Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany.
benner@mpi-magdeburg.mpg.de

†Institut für Mathematik, Universität Augsburg, Universitätsstraße 14, 86159 Augsburg, Germany.
stykel@math.uni-augsburg.de

For standard Riccati equations many different numerical methods have been proposed over the last thirty years. These are the Schur vector method [29], the sign function method [13], the structured doubling algorithm [14], Krylov subspace methods [22, 23] and symplectic methods [12]. All these methods rely on a Hamiltonian eigenvalue problem. Another approach is based on considering the Riccati equation (1) with $E = I_n$ as a system of nonlinear equations. Such a system can be solved by Newton's method [4, 9, 27, 44]. In this paper, we present an extension of this method to the PARE (2).

Throughout the paper, the open left half-plane is denoted by \mathbb{C}_- . The matrices A^T and A^* stand for the transpose and conjugate transpose of A , respectively, and $A^{-T} = (A^{-1})^T$. An identity matrix of order n is denoted by I_n or simply by I . For symmetric matrices $A, B \in \mathbb{R}^{n,n}$, we write $A > B$ ($A \geq B$) if $A - B$ is positive definite (semidefinite). We denote by $\|A\|_2$ and $\|A\|_F$ the spectral and Frobenius matrix norms of $A \in \mathbb{R}^{n,m}$.

This paper is organized as follows. In Section 2, we discuss the solvability of the PARE (2). In Section 3, we present Newton and Newton-Kleinman iterations for solving this equation. We also study the convergence of these methods and discuss the computation of a stabilizing initial guess. Section 4 contains low-rank versions of Newton-type methods. Finally, some results of numerical experiments for the presented algorithms are reported in Section 5.

2 Preliminaries

In this section, we give basic definitions and some notations from matrix analysis and control theory that will be used in the following. We also study the solvability of the PARE (2).

Any regular pencil $\lambda E - F$ can be transformed into its Weierstrass canonical form

$$E = T_l \begin{bmatrix} I_{n_f} & 0 \\ 0 & E_\infty \end{bmatrix} T_r, \quad F = T_l \begin{bmatrix} F_f & 0 \\ 0 & I_{n_\infty} \end{bmatrix} T_r, \quad (3)$$

where T_l and T_r are the left and right nonsingular transformation matrices, $F_f \in \mathbb{R}^{n_f, n_f}$ and $E_\infty \in \mathbb{R}^{n_\infty, n_\infty}$ are matrices in Jordan canonical form, and E_∞ is nilpotent with index of nilpotency ν , see [17]. The eigenvalues of F_f are the finite eigenvalues of $\lambda E - F$, and E_∞ corresponds to an eigenvalue at infinity. The number ν is called the *index* of $\lambda E - F$. The pencil $\lambda E - F$ is called *stable* if all its finite eigenvalues have negative real part. Using the Weierstrass canonical form (3), the spectral projectors P_r and P_l onto the right and left deflating subspaces of the pencil $\lambda E - F$ corresponding to the finite eigenvalues can be represented as

$$P_r = T_r^{-1} \begin{bmatrix} I_{n_f} & 0 \\ 0 & 0 \end{bmatrix} T_r, \quad P_l = T_l \begin{bmatrix} I_{n_f} & 0 \\ 0 & 0 \end{bmatrix} T_l^{-1}. \quad (4)$$

A triple (E, F, G) is called *stabilizable* if $\text{rank}[\lambda E - F, G] = n$ for all $\lambda \in \mathbb{C} \setminus \mathbb{C}_-$. A triple (E, F, H) is called *detectable* if $\text{rank}[\lambda E^T - F^T, H^T] = n$ for all $\lambda \in \mathbb{C} \setminus \mathbb{C}_-$. A solution X_* of the PARE (2) is called *stabilizing* if X_* is symmetric and the pencil $\lambda E - F - EX_*GP_r$ is stable. If this pencil has all finite eigenvalues in the closed left-half plane, then the symmetric solution X_* of (2) is called *semi-stabilizing*. The following theorem gives sufficient conditions for the existence of a unique stabilizing solution of (2).

Theorem 2.1 *Consider the PARE (2) with $G = G^T \geq 0$ and $H = H^T \geq 0$. If (E, F, G) is stabilizable and (E, F, H) is detectable, then (2) has a unique stabilizing solution X_* .*

Proof. Let $\lambda E - F$ be in Weierstrass canonical form (3) and let the matrices

$$G = T_r^T \begin{bmatrix} G_{11} & G_{12} \\ G_{12}^T & G_{22} \end{bmatrix} T_r, \quad H = T_l \begin{bmatrix} H_{11} & H_{12} \\ H_{12}^T & H_{22} \end{bmatrix} T_l^T, \quad (5)$$

$$X = T_r^{-1} \begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} T_r^{-T}$$

be partitioned into blocks accordingly to E and F . Since G and H are both symmetric and positive semidefinite, the matrices G_{11} and H_{11} are also symmetric and positive semidefinite. Substituting (3), (4) and (5) into the PARE (2), we obtain that any solution of (2) has the form

$$X = T_r^{-1} \begin{bmatrix} X_{11} & 0 \\ 0 & 0 \end{bmatrix} T_r^{-T}, \quad (6)$$

where X_{11} satisfies the standard Riccati equation

$$H_{11} + F_f X_{11} + X_{11} F_f^T + X_{11} G_{11} X_{11} = 0. \quad (7)$$

Since (E, F, G) is stabilizable, we have

$$\begin{aligned} n &= \text{rank}[\lambda E - F, G] = \text{rank} \begin{bmatrix} \lambda I - F_f & 0 & G_{11} & G_{12} \\ 0 & \lambda E_\infty - I & G_{12}^T & G_{22} \end{bmatrix} \\ &= n_\infty + \text{rank}[\lambda I - F_f, G_{11}, G_{12}] \end{aligned}$$

for all $\lambda \in \mathbb{C} \setminus \mathbb{C}_-$. Then

$$\text{rank}[\lambda I - F_f, G_{11}, G_{12}] = n_f \quad \text{for all } \lambda \in \mathbb{C} \setminus \mathbb{C}_-. \quad (8)$$

We now show that $\text{rank}[\lambda I - F_f, G_{11}] = n_f$ for all $\lambda \in \mathbb{C} \setminus \mathbb{C}_-$, i.e., the pair (F_f, G_{11}) is stabilizable in the classical sense. For this purpose, consider the Cholesky factorization $G = G_1 G_1^T$. Then for $T_r^{-T} G_1 = [T_1^T, T_2^T]^T$, we have

$$\begin{bmatrix} G_{11} & G_{12} \\ G_{12}^T & G_{22} \end{bmatrix} = T_r^{-T} G T_r^{-1} = \begin{bmatrix} T_1 T_1^T & T_1 T_2^T \\ T_2 T_1^T & T_2 T_2^T \end{bmatrix}.$$

Assume that $\text{rank}[\lambda_0 I - F_f, G_{11}] < n_f$ for some $\lambda_0 \in \mathbb{C} \setminus \mathbb{C}_-$. In this case, there exists $v \neq 0$ such that $v^*[\lambda_0 I - F_f, G_{11}] = 0$. Then $0 = v^* G_{11} v = v^* T_1 T_1^T v$ implies $v^* T_1 = 0$ and, hence,

$$v^*[\lambda_0 I - F_f, G_{11}, G_{12}] = v^*[\lambda_0 I - F_f, T_1 T_1^T, T_1 T_2^T] = 0.$$

This contradicts (8).

Analogously, we can show that the detectability of (E, F, H) implies the detectability of the pair (F_f, H_{11}) in the classical sense. In this case, the Riccati equation (7) has a unique symmetric solution X_{11} such that all the eigenvalues of $F_f + X_{11} G_{11}$ have negative real part, see [28, Chapter 8]. For this X_{11} , the matrix X in (6) is symmetric and the pencil $\lambda E - F - EXGP_r$ is stable. Thus, the PARE (2) has a unique stabilizing solution. \square

Note that in Theorem 2.1 we make no assumptions about the index of the pencil $\lambda E - F$. Similarly to the standard case with $E = I$, see [28, Section 8.5], the detectability condition can be weakened. If only a semi-stabilizable solution of (2) is required, then also stabilizability can be relaxed. Moreover, for projected Riccati equations arising in passivity-preserving balanced truncation of structured passive circuit equations, these conditions can be removed at all [35].

3 Newton's method

Observing that the first equation in (2) is a system of nonlinear equations, it suggests itself to solve it using Newton's method.

Let P be a projector and let $\mathbb{S}_P = \{X \in \mathbb{R}^{n,n} : X = X^T \text{ and } X = PXP^T\}$. Consider a *Riccati operator* $\mathcal{R} : \mathbb{S}_P \rightarrow \mathbb{S}_P$ given by

$$\mathcal{R}(X) = P_1 H P_1^T + F X E^T + E X F^T + E X G X E^T.$$

The Frechét derivative of \mathcal{R} at $X \in \mathbb{S}_P$ is a linear operator $\mathcal{R}'_X : \mathbb{S}_P \rightarrow \mathbb{S}_P$ defined as

$$\mathcal{R}'_X(N) = \lim_{\delta \rightarrow 0} \frac{1}{\delta} (\mathcal{R}(X + \delta N) - \mathcal{R}(X))$$

for $N \in \mathbb{S}_{P_r}$. Taking into account that $N = P_r N = N P_r^T$, we have

$$\mathcal{R}'_X(N) = (F + EXGP_r)NE^T + EN(F + EXGP_r)^T.$$

Then Newton's method for the PARE (2) can be written as

$$\begin{aligned} N_j &= -(\mathcal{R}'_{X_j})^{-1}(\mathcal{R}(X_j)), \\ X_{j+1} &= X_j + N_j. \end{aligned}$$

The standard formulation of this method is given in Algorithm 1.

Algorithm 1 Newton's method

INPUT: $E, F, G, H \in \mathbb{R}^{n,n}$, projectors P_r and P_l , a stabilizing initial guess $X_0 \in \mathbb{S}_{P_r}$.

OUTPUT: An approximate solution of the PARE (2).

FOR $j = 0, 1, 2, \dots$

1. Compute $F_j = F + EX_jGP_r$.
2. Solve the projected algebraic Lyapunov equation (PALE)

$$F_j N_j E^T + EN_j F_j^T = -P_l \mathcal{R}(X_j) P_l^T, \quad N_j = P_r N_j P_r^T. \quad (9)$$

3. Compute $X_{j+1} = X_j + N_j$.

END FOR

As in the standard case [27], we can combine the second and third steps in Algorithm 1 and compute the new iterate X_{j+1} directly from the projected Lyapunov equation as presented in Algorithm 2.

Algorithm 2 Newton-Kleinman method

INPUT: $E, F, G, H \in \mathbb{R}^{n,n}$, projectors P_r and P_l , a stabilizing initial guess $X_0 \in \mathbb{S}_{P_r}$.

OUTPUT: An approximate solution of the PARE (2).

FOR $j = 0, 1, 2, \dots$

1. Compute $F_j = F + EX_jGP_r$.
2. Solve the PALE

$$\begin{aligned} F_j X_{j+1} E^T + EX_{j+1} F_j^T &= -P_l (H - EX_j G X_j E^T) P_l^T, \\ X_{j+1} &= P_r X_{j+1} P_r^T. \end{aligned} \quad (10)$$

END FOR

Although Algorithms 1 and 2 are mathematically equivalent, they behave differently in finite precision arithmetic and there are significant differences in their implementation especially for large-scale problems. We will compare the Newton and Newton-Kleinman methods in Section 4.5.

Remark 3.1 Due $X_j = P_r X_j$ and $P_l E = EP_r$, the matrices F_j in Algorithms 1 and 2 satisfy $F_j = F + EX_jGP_r = F + P_l EX_jGP_r$. Then using the Weierstrass canonical form (3) and representations (4), we obtain that P_l and P_r are the spectral projectors onto the left and right deflating subspaces corresponding to the finite eigenvalues not only of $\lambda E - F$ but also of $\lambda E - F_j$.

3.1 Convergence

First, we investigate the convergence of Algorithms 1 and 2. The following theorem establishes that the PALEs (9) and (10) are solvable and the iterate X_j converges in both algorithms to a stabilizing solution of the PARE (2) for every stabilizing $X_0 \in \mathbb{S}_{P_r}$.

Theorem 3.2 Let $X_0 \in \mathbb{S}_{P_r}$ be chosen such that the pencil $\lambda E - F - EX_0GP_r$ is stable. Assume that the PARE (2) with $G = G^T \geq 0$ has a unique stabilizing solution X_* .

Then for the iterate X_j in Algorithm 1 or 2, we have:

- (i) the pencil $\lambda E - F_j$ with $F_j = F + EX_jGP_r$ is stable for all $j \geq 0$;
- (ii) $X_1 \leq X_2 \leq \dots \leq X_j \leq X_{j+1} \leq \dots \leq X_*$;
- (iii) $\lim_{j \rightarrow \infty} X_j = X_*$ and $\lim_{j \rightarrow \infty} \mathcal{R}(X_j) = 0$;
- (iv) there exists a constant $\gamma > 0$ such that $\|X_* - X_{j+1}\| \leq \gamma \|X_* - X_j\|^2$ for $j \geq 1$, i.e., the iterate X_j converges globally and quadratic to X_* .

Proof. This theorem can be proved in two different ways. The first approach is based on transforming the pencil $\lambda E - F$ into the Weierstrass canonical form (3) and applying the classical convergence results [5, 44] to the standard Riccati equation (7). On the other hand, these results can be reformulated in terms of the original data. We choose the second approach.

(i) The stability of $\lambda E - F_j$ with $F_j = F + EX_jGP_r$, $j = 0, 1, \dots$, can be proved by induction. The pencil $\lambda E - F_0$ is stable by the choice of X_0 . Assume now that $\lambda E - F_j$ is stable. Then the PARE (9) has a unique symmetric solution N_j [40]. Since X_0 is symmetric, the iterate $X_{j+1} = X_j + N_j$ is also symmetric. Subtracting equation (10) for X_{j+1} from the PARE (2) with X replaced by X_* , we obtain that the difference $D_{j+1} = X_* - X_{j+1}$ satisfies the PARE

$$F_j D_{j+1} E^T + E D_{j+1} F_j^T = -P_l E D_j G D_j E^T P_l^T, \quad D_{j+1} = P_r D_{j+1} P_r^T. \quad (11)$$

Since $\lambda E - F_j$ is stable and the right-hand side in the first equation in (11) is symmetric and negative semidefinite, then (11) has a unique symmetric, positive semidefinite solution D_{j+1} , see [40]. Thus, $X_{j+1} \leq X_*$.

We now show that $\lambda E - F_{j+1}$ is stable. The first equation in (11) can be written as

$$F_{j+1} D_{j+1} E^T + E D_{j+1} F_{j+1}^T = -E D_{j+1} G D_{j+1} E^T - E N_j G N_j E^T. \quad (12)$$

Assume that $\lambda E - F_{j+1}$ has an eigenvalue λ_0 with $\text{Re}(\lambda_0) \geq 0$. Let $v = P_l v$ be a left eigenvector of $\lambda E - F_{j+1}$ corresponding to λ_0 , i.e., $\lambda_0 v^* E = v^* F_{j+1}$. Pre-multiplying (12) by v^* and post-multiplying by v , we obtain that

$$2\text{Re}(\lambda_0) v^* E D_{j+1} E^T v = -v^* E D_{j+1} G D_{j+1} E^T v - v^* E N_j G N_j E^T v.$$

Since D_{j+1} and G are both symmetric, positive semidefinite and $\text{Re}(\lambda_0) \geq 0$, we have $G D_{j+1} E^T v = 0$ and $G N_j E^T v = 0$. Then

$$v^* F_j = v^* (F + EX_jGP_r) = v^* (F + EX_{j+1}GP_r) = v^* F_{j+1} = \lambda_0 v^* E.$$

This contradicts the stability of $\lambda E - F_j$. Thus, $\lambda E - F_{j+1}$ is stable.

(ii) It follows from the equations (2) with $X = X_*$ and (9) that

$$\mathcal{R}(X_j) = E N_{j-1} G N_{j-1} E^T, \quad j \geq 1. \quad (13)$$

Since $\lambda E - F_j$ is stable and $\mathcal{R}(X_j)$ is symmetric, positive semidefinite, the PARE (9) has a unique symmetric, positive semidefinite solution $N_j = X_{j+1} - X_j$. Hence, $X_j \leq X_{j+1}$ for $j \geq 1$.

(iii) Since the sequence $\{X_j\}_{j \geq 1}$ is non-decreasing and bounded above by X_* , this sequence is convergent, i.e., $\lim_{j \rightarrow \infty} X_j = X_\infty$ with symmetric $X_\infty \leq X_*$. Hence the relation (13) implies $\lim_{j \rightarrow \infty} \mathcal{R}(X_j) = 0$. Passing to the limit in (10), we obtain that X_∞ solves the PARE (2). Replacing X by X_∞ and X_* in (2) and considering the difference of the resulting equations, we find that

$$\begin{aligned} F_*(X_\infty - X_*)E^T + E(X_\infty - X_*)F_*^T &= -P_l E(X_\infty - X_*)G(X_\infty - X_*)E^T P_l^T, \\ X_\infty - X_* &= P_r(X_\infty - X_*)P_r^T \end{aligned} \quad (14)$$

with $F_* = F + EX_*GP_r$. Since $\lambda E - F_*$ is stable and G is symmetric, positive semidefinite, the PALE (14) has a unique symmetric, positive semidefinite solution $X_\infty - X_*$, i.e., $X_\infty \geq X_*$. Thus, $X_\infty = X_*$.

(iv) Subtracting equation (10) for X_{j+1} from the PARE (2) with X replaced by X_* , we obtain that the difference $D_{j+1} = X_* - X_{j+1}$ satisfies the PALE

$$\begin{aligned} F_*D_{j+1}E^T + ED_{j+1}F_*^T &= -P_lE(N_jGN_j - D_{j+1}GD_{j+1})E^TP_l^T, \\ D_{j+1} &= P_rD_{j+1}P_r^T. \end{aligned} \quad (15)$$

Since $\lambda E - F_*$ is stable, this equation has a unique solution given by

$$0 \leq D_{j+1} = \frac{1}{2\pi} \int_{-\infty}^{\infty} (i\omega E - F_*)^{-1} P_l E (N_j G N_j - D_{j+1} G D_{j+1}) E^T P_l^T (-i\omega E - F_*)^{-T} d\omega,$$

see [40]. Taking norms, we get

$$\begin{aligned} \|X_* - X_{j+1}\| &\leq \left\| \frac{1}{2\pi} \int_{-\infty}^{\infty} (i\omega E - F_*)^{-1} P_l E N_j G N_j E^T P_l^T (-i\omega E - F_*)^{-T} d\omega \right\| \\ &\leq \|N_j\|^2 \|E\|^2 \|G\| \left\| \frac{1}{2\pi} \int_{-\infty}^{\infty} (i\omega E - F_*)^{-1} P_l P_l^T (-i\omega E - F_*)^{-T} d\omega \right\| \\ &= \gamma \|X_{j+1} - X_j\|^2. \end{aligned}$$

Since $0 \leq X_{j+1} - X_j \leq X_* - X_j$, we obtain that $\|X_* - X_{j+1}\| \leq \gamma \|X_* - X_j\|^2$. \square

It follows from the proof of Theorem 3.2 that the stabilizing solution X_* of the PARE (2) is minimal in the sense that $X_* \leq X$ for all symmetric solutions X of (2). Theorem 3.2 also shows that in each step of Algorithms 1 and 2, all iterates X_j are stabilizing once the initial guess X_0 is chosen to be stabilizing. Such a matrix exists, if, for example, (E, F, G) is stabilizable. If this condition is violated, it is still possible, similarly to the standard case [7], to construct a convergent Newton-Kleinman iteration for computing a minimal solution of (2).

Note that the existence of a stabilizing solution of the PARE (2) guarantees a quadratic convergence of the Newton iteration. However, if (2) has only a semi-stabilizing solution X_* , then the quadratic convergence may be lost. For the standard Riccati equation with negative (semi)definite quadratic term ($G \leq 0$), a modification of the Newton iteration has been proposed in [20] which has a linear rate of convergence under assumption that the purely imaginary eigenvalues of $\lambda E - F - EX_*GP_r$ are semi-simple. This result can also be extended to the PARE (2) with positive semidefinite quadratic term.

At each iteration step of Algorithms 1 and 2, we have to solve the PALEs (9) and (10), respectively. For small and medium size problems, such equations can be solved using the generalized Schur-Bartels-Stewart method or the generalized Schur-Hammarling method [39]. For large dense problems, we can use the modified sign function method [41], whereas projected Lyapunov equations with large-scale sparse matrix coefficients can be solved using the generalized alternating direction implicit method [42] or Krylov subspace methods [43]. We will discuss a combination of Lyapunov solvers with Newton iteration in Section 4 in more detail.

3.2 Computing the stabilizing initial guess

The convergence of Algorithms 1 and 2 relies on a stabilizing initial guess X_0 that also satisfies $X_0 \in \mathbb{S}_{P_r}$. If $\lambda E - F$ is stable, then $X_0 = 0$ trivially satisfies these demands. This is often the case in applications, but certainly not always. Thus, computing such X_0 is required in the unstable situations. The stabilization of descriptor systems using partial stabilization, i.e., computing X_0 such that the stable and infinite eigenvalues of $\lambda E - F$ remain unchanged and the unstable ones are moved to the open left half-plane, is considered in [6]. The suggested procedures basically use a numerically robust variant to compute a block form as in (3), where the nonzero blocks are not required to have a special structure. Then using the decomposition as in (5), the stabilization problem can be solved using the Bass algorithm or an algebraic Bernoulli equation as in the

standard case. Though the symmetry condition $X_0 \in \mathbb{S}_{P_r}$ is not considered in [6], using the same decomposition as in (5) with $G = 0$ and the resulting form of the solutions X as in (6) of the Lyapunov or Bernoulli equations, it is easy to show that the associated X_0 matrices satisfy this symmetry equation. The solution of these Lyapunov or Bernoulli equations can also be obtained in factored forms as in [1, 2, 41, 42].

4 Stabilizing solution in factored form and its low-rank approximation

4.1 Computing the Cholesky factor of the stabilizing solution

In many control applications including positive real and bounded real balanced truncation model reduction, the matrices G and H in the PARE (2) are given in factored form $G = \tilde{G}^T \tilde{G}$ and $H = \tilde{H} \tilde{H}^T$, where $\tilde{G} \in \mathbb{R}^{p,n}$ and $\tilde{H} \in \mathbb{R}^{n,m}$. In this case, the stabilizing positive semidefinite solution of (2) can also be determined in factored form $X_* = \tilde{X}_* \tilde{X}_*^T$. Note that the computation of the Cholesky factorization $X_* = \tilde{X}_* \tilde{X}_*^T$ should be avoided because the computed Cholesky factor \tilde{X}_* has usually lower accuracy than X_* . Fortunately, the factor \tilde{X}_* of X_* can be computed directly without calculating X_* itself. This can be done in two different ways based on Algorithms 1 and 2, respectively. We consider first Newton's method in Algorithm 1. It follows from $EP_r = P_l E$, $FP_r = P_l F$ and $X_j = P_r X_j P_r^T$ that $\mathcal{R}(X_j) = P_l \mathcal{R}(X_j) P_l^T$. Then taking into account (9), we obtain that

$$\begin{aligned} \mathcal{R}(X_{j+1}) &= \mathcal{R}(X_j + N_j) \\ &= \mathcal{R}(X_j) + F_j N_j E^T + EN_j F_j^T + EN_j \tilde{G}^T \tilde{G} N_j E^T \\ &= EN_j \tilde{G}^T \tilde{G} N_j E^T. \end{aligned} \quad (16)$$

In this case, the PALE (9) can be replaced by

$$F_j N_j E^T + EN_j F_j^T = -P_l K_j K_j^T P_l^T, \quad N_j = P_r N_j P_r^T \quad (17)$$

with $K_j = EN_{j-1} \tilde{G}^T$ for $j > 0$. Since $\lambda E - F_j$ is stable, (17) has a unique symmetric, positive semidefinite solution N_j that can be factorized as $N_j = \tilde{N}_j \tilde{N}_j^T$.

For stable $\lambda E - F$, we can start with $X_0 = 0$ and solve the PALE

$$FN_0 E^T + EN_0 F^T = -P_l \tilde{H} \tilde{H}^T P_l^T, \quad N_0 = P_r N_0 P_r^T \quad (18)$$

for a Cholesky factor \tilde{N}_0 of $N_0 = \tilde{N}_0 \tilde{N}_0^T$. Otherwise, we first compute the Cholesky factorization $\mathcal{R}(X_0) = K_0 K_0^T$ for some stabilizing starting guess X_0 and determine the Cholesky factor of the solution of the PALE

$$F_0 N_0 E^T + EN_0 F_0^T = -P_l K_0 K_0^T P_l^T, \quad N_0 = P_r N_0 P_r^T \quad (19)$$

with $F_0 = F + EX_0 \tilde{G}^T \tilde{G} P_r$. If $\mathcal{R}(X_0)$ is indefinite, one can compute the solution of (9) for $j = 0$, and then employ (17) starting with $j = 1$. Once we have $X_j = \tilde{X}_j \tilde{X}_j^T$ and $N_j = \tilde{N}_j \tilde{N}_j^T$, then the next iterate can be obtained in factored form as

$$X_{j+1} = X_j + N_j = \tilde{X}_{j+1} \tilde{X}_{j+1}^T,$$

where $\tilde{X}_{j+1} \in \mathbb{R}^{n,n}$ is computed from the LQ factorization $[\tilde{X}_j, \tilde{N}_j] = [\tilde{X}_{j+1}, 0] \tilde{Q}_j$ with orthogonal \tilde{Q}_j . Note that the Cholesky factors \tilde{N}_j of the solutions of the PALEs (17) – (19) can be determined directly without computing N_j itself by using the generalized Schur-Hammarling method [39]. This method does not require the preliminary computation of the projectors P_r and P_l . It is based on reducing the pencil $\lambda E - F_j$ to the generalized Schur form [18] and solving the generalized Sylvester and Lyapunov equations. Using the fact that the pencils $\lambda E - F$ and $\lambda E - F_j$ have the same deflating subspaces corresponding to the finite eigenvalues, we do not need to compute the generalized Schur form of $\lambda E - F_j$ at every Newton iteration. It is enough to reduce $\lambda E - F$ into the

Algorithm 3 Newton-Schur-Hammarling method

INPUT: $E, F \in \mathbb{R}^{n,n}$, $\tilde{G} \in \mathbb{R}^{p,n}$, $\tilde{H} \in \mathbb{R}^{n,m}$ and a stabilizing initial guess $X_0 \in \mathbb{S}_{P_r}$.
OUTPUT: An approximate solution $\tilde{X}\tilde{X}^T$ of the PARE (2).

1. Compute the generalized Schur form

$$V^T E U = \begin{bmatrix} E_{11} & E_{12} \\ 0 & E_{22} \end{bmatrix}, \quad V^T F U = \begin{bmatrix} F_{11} & F_{12} \\ 0 & F_{22} \end{bmatrix}, \quad (20)$$

where V and U are orthogonal, E_{11} is upper triangular, nonsingular and E_{22} is upper triangular with zeros on the diagonal, F_{11} is upper quasi-triangular and F_{22} is upper triangular, nonsingular.

2. Solve the generalized Sylvester equation

$$\begin{aligned} E_{11}Y - ZE_{22} &= -E_{12}, \\ F_{11}Y - ZF_{22} &= -F_{12}. \end{aligned} \quad (21)$$

3. Compute the matrices

$$\tilde{G}U = [\tilde{G}_1, \tilde{G}_2], \quad V^T \tilde{H} = \begin{bmatrix} \tilde{H}_1 \\ \tilde{H}_2 \end{bmatrix}, \quad U^T X_0 U = \begin{bmatrix} X_{11,0} & 0 \\ 0 & 0 \end{bmatrix}.$$

4. Compute the Cholesky factorizations

$$\begin{aligned} X_{11,0} &= \tilde{X}_{11,0} \tilde{X}_{11,0}^T, \\ \mathcal{R}_1(X_{11,0}) &= F_{11} X_{11,0} E_{11}^T + E_{11} X_{11,0} F_{11}^T + E_{11} X_{11,0} \tilde{G}_1^T \tilde{G}_1 X_{11,0} E_{11}^T \\ &\quad + (\tilde{H}_1 - Z\tilde{H}_2)(\tilde{H}_1 - Z\tilde{H}_2)^T = K_{1,0} K_{1,0}^T \end{aligned}$$

and the matrix $F_{11,0} = F_{11} + E_{11} X_{11,0} \tilde{G}_1^T \tilde{G}_1$.

5. Solve the generalized Lyapunov equation

$$F_{11,0} N_{11,0} E_{11}^T + E_{11} N_{11,0} F_{11,0}^T = -K_{1,0} K_{1,0}^T \quad (22)$$

for the Cholesky factor $\tilde{N}_{11,0}$ of $N_{11,0} = \tilde{N}_{11,0} \tilde{N}_{11,0}^T$.

6. Compute the LQ factorization $[\tilde{X}_{11,0}, \tilde{N}_{11,0}] = [\tilde{X}_{11,1}, 0] Q_{1,j}$.

7. FOR $j = 1, 2, \dots, j_{\max} - 1$

a) Compute $K_{1,j} = E_{11} \tilde{N}_{11,j-1} \tilde{N}_{11,j-1}^T \tilde{G}_1^T$ and $F_{11,j} = F_{11,j-1} + K_{1,j} \tilde{G}_1$.

b) Solve the generalized Lyapunov equation

$$F_{11,j} N_{11,j} E_{11}^T + E_{11} N_{11,j} F_{11,j}^T = -K_{1,j} K_{1,j}^T \quad (23)$$

for the Cholesky factor $\tilde{N}_{11,j}$ of $N_{11,j} = \tilde{N}_{11,j} \tilde{N}_{11,j}^T$.

c) Compute the LQ factorization $[\tilde{X}_{11,j}, \tilde{N}_{11,j}] = [\tilde{X}_{11,j+1}, 0] Q_{1,j}$.

END FOR

8. Compute

$$\tilde{X} = U \begin{bmatrix} \tilde{X}_{11,j_{\max}} \\ 0 \end{bmatrix}.$$

generalized Schur form and solve the projected Lyapunov equations on the subspace corresponding to the finite eigenvalues. We summarize the resulting Newton's method in Algorithm 3.

For computing the generalized Schur form (20), we can use the QZ algorithm [18, 48] or the GUPTRI algorithm [15, 16]. To solve the generalized Sylvester equation (21) one can use the generalized Schur method [24]. The Cholesky factor of the solutions of the generalized Lyapunov equations (22) and (23) can be determined using the generalized Hammarling method [33]. The Newton-Schur-Hammarling method for the PARE (2) costs $O(n^3)$ flops and has the memory complexity $O(n^2)$.

4.2 Low-rank approximation to the stabilizing solution

If the eigenvalues of the stabilizing solution X_* decay to zero very rapidly, then X_* can be well approximated by a matrix of low rank. Such a low-rank approximation can be computed in factored form $X_* \approx \tilde{X}\tilde{X}^T$ with $\tilde{X} \in \mathbb{R}^{n,k}$, $k \ll n$, by solving the PALEs (17) and (18) for low-rank approximate solutions. For this purpose, we can use an extension of the low-rank sign function method, the low-rank alternating direction implicit method or Krylov subspace methods to projected Lyapunov equations as presented in [41, 42, 43].

Algorithm 4 Low-rank Newton method

INPUT: $E, F \in \mathbb{R}^{n,n}$ such that $\lambda E - F$ is stable, $\tilde{G} \in \mathbb{R}^{p,n}$, $\tilde{H} \in \mathbb{R}^{n,m}$, projectors P_r, P_l .

OUTPUT: A low-rank Cholesky factor of the stabilizing solution of the PARE (2).

1. Solve the PALE (18) for the low-rank Cholesky factor \tilde{N}_0 such that $N_0 \approx \tilde{N}_0\tilde{N}_0^T$.
 2. Set $\tilde{X}_1 = \tilde{N}_0$, $K_1 = E\tilde{N}_0\tilde{N}_0^T\tilde{G}^T$ and $F_1 = F + K_1\tilde{G}P_r$.
 3. FOR $j = 1, 2, \dots$
 - a) Solve the PALE (17) for the low-rank Cholesky factor \tilde{N}_j such that $N_j \approx \tilde{N}_j\tilde{N}_j^T$.
 - b) Compute $\tilde{X}_{j+1} = [\tilde{X}_j, \tilde{N}_j]$.
 - c) Compute $K_{j+1} = E\tilde{N}_j\tilde{N}_j^T\tilde{G}^T$ and $F_{j+1} = F_j + K_{j+1}\tilde{G}P_r$.
- END FOR
-

The computation of the low-rank Cholesky factor of the stabilizing solution of the PARE (2) with stable $\lambda E - F$ is summarized in Algorithm 4. Note that in each iterative step in this algorithm the number of columns of the approximate Cholesky factor \tilde{X}_j of the solution of (2) increases by the number k_j of columns of the approximate Cholesky factor \tilde{N}_j of the solution of the PALE (18). In case of large k_j or slow convergence of the Newton iteration, a large workspace is required to store \tilde{X}_{j+1} . In order to keep low-rank structure in \tilde{X}_{j+1} , one can replace this iterate by its low-rank approximation computed via a rank-revealing QR decomposition

$$[\tilde{X}_j, \tilde{N}_j]^T = Q_j \begin{bmatrix} R_{j,1} & R_{j,2} \\ 0 & R_{j,3} \end{bmatrix} \Pi_j^T, \quad (24)$$

where Q_j has orthogonal columns, Π_j is a permutation matrix, $R_{j,1}$ has full row rank and $\|R_{j,3}\|_F \leq \tau \|[\tilde{X}_j, \tilde{N}_j]\|_F$ for some small tolerance τ . Setting $R_{j,3} = 0$, we can proceed with the new iterate $\tilde{X}_{j+1} = \Pi_j[R_{j,1}, R_{j,2}]^T$. Note that in (24) we do not need to accumulate the matrix Q_j .

Low-rank Cholesky factors of the stabilizing solution of the PARE (2) can also be computed using the Newton-Kleinman method in Algorithm 2. Such an approach has been considered previously for the case $E = I$ in [44]. An extension of this approach is given in Algorithm 5.

If $\lambda E - F$ is stable, in Step 3 of Algorithm 5 we can solve the PALE

$$FXE^T + EXF^T = -P_l\hat{H}\hat{H}^T P_l^T, \quad X = P_r X P_r^T \quad (27)$$

Algorithm 5 Low-rank Newton-Kleinman method

INPUT: $E, F \in \mathbb{R}^{n,n}$, $\tilde{G} \in \mathbb{R}^{p,n}$, $\tilde{H} \in \mathbb{R}^{n,m}$, projectors P_r and P_l , a low-rank matrix \tilde{X}_0 such that $\tilde{X}_0 = P_r \tilde{X}_0$ and $\lambda E - F - E \tilde{X}_0 \tilde{X}_0^T \tilde{G}^T P_r$ is stable.

OUTPUT: A low-rank Cholesky factor of the stabilizing solution of the PARE (2).

1. Set $\tilde{X}_{1,0} = \tilde{X}_0$ and $\tilde{X}_{2,0} = 0$.
2. FOR $j = 0, 1, 2, \dots, j_{\max} - 1$
 - a) Compute $K_j = E(\tilde{X}_{1,j} \tilde{X}_{1,j}^T - \tilde{X}_{2,j} \tilde{X}_{2,j}^T) \tilde{G}^T$ and $F_j = F + K_j \tilde{G} P_r$.
 - b) Solve the PALEs

$$F_j X_{1,j+1} E^T + E X_{1,j+1} F_j^T = -P_l \tilde{H} \tilde{H}^T P_l^T, \quad X_{1,j+1} = P_r X_{1,j+1} P_r^T, \quad (25)$$

$$F_j X_{2,j+1} E^T + E X_{2,j+1} F_j^T = -P_l K_j K_j^T P_l^T, \quad X_{2,j+1} = P_r X_{2,j+1} P_r^T \quad (26)$$

for the low-rank Cholesky factors $\tilde{X}_{1,j+1}$ and $\tilde{X}_{2,j+1}$ such that $X_{1,j+1} \approx \tilde{X}_{1,j+1} \tilde{X}_{1,j+1}^T$ and $X_{2,j+1} \approx \tilde{X}_{2,j+1} \tilde{X}_{2,j+1}^T$.

END FOR

3. Solve the PALE $\hat{F} X E^T + E X \hat{F}^T = -P_l \tilde{H} \tilde{H}^T P_l^T$, $X = P_r X P_r^T$ with $\hat{F} = F + \frac{1}{2} E(\tilde{X}_{1,j_{\max}} \tilde{X}_{1,j_{\max}}^T - \tilde{X}_{2,j_{\max}} \tilde{X}_{2,j_{\max}}^T) \tilde{G}^T \tilde{G} P_r$ for the low-rank Cholesky factor \tilde{X} such that $X \approx \tilde{X} \tilde{X}^T$.
-

with $\hat{H} = [\tilde{H}, E(\tilde{X}_{1,j_{\max}} \tilde{X}_{1,j_{\max}}^T - \tilde{X}_{2,j_{\max}} \tilde{X}_{2,j_{\max}}^T) \tilde{G}^T]$.

Note that in Algorithms 4 and 5, the projectors P_l and P_r are required in explicit form. The computation of these projectors is, in general, very expensive. Fortunately, in many applications including control of fluid flow, electrical circuit simulation and constrained multibody systems, the matrices E and F have some special block structure. This structure can be exploited to construct the projections P_l and P_r explicitly and cheaply, see [35, 42].

4.3 Low-rank ADI iteration for projected Lyapunov equations

In this section, we briefly discuss the computation of approximate solutions to the PALE

$$\tilde{F} Z E^T + E Z \tilde{F}^T = -P_l \tilde{K} \tilde{K}^T P_l^T, \quad Z = P_r Z P_r^T \quad (28)$$

with given $E, \tilde{F} \in \mathbb{R}^{n,n}$, $\tilde{K} \in \mathbb{R}^{n,g}$ and unknown $Z \in \mathbb{R}^{n,n}$. According to Remark 3.1 we can assume that P_l and P_r are the spectral projectors onto the left and right deflating subspaces of $\lambda E - \tilde{F}$ corresponding to the finite eigenvalues. For solving the PALE (28), we use the *alternating direction implicit* (ADI) *method*. This method has been first proposed for standard Lyapunov equations [9, 30, 34, 46] and then extended in [42] to projected Lyapunov equations. Recently, a more efficient version of the ADI iteration for standard Lyapunov equations was proposed in [8] which allows a cheap computation of the Lyapunov residuals. Here, we extend this result to projected Lyapunov equations.

The generalized ADI iteration for the PALE (28) is given by

$$\begin{aligned} (E + \tau_k \tilde{F}) Z_{k-1/2} \tilde{F}^T + \tilde{F} Z_{k-1} (E - \tau_k \tilde{F})^T &= -P_l \tilde{K} \tilde{K}^T P_l^T, \\ (E + \bar{\tau}_k \tilde{F}) Z_k^T \tilde{F}^T + \tilde{F} Z_{k-1/2}^T (E - \bar{\tau}_k \tilde{F})^T &= -P_l \tilde{K} \tilde{K}^T P_l^T \end{aligned} \quad (29)$$

with an initial matrix $Z_0 = 0$ and shift parameters $\tau_1, \dots, \tau_k \in \mathbb{C}_-$. Solving the first equation for $Z_{k-1/2}$ and the second equation for Z_k , we obtain

$$\begin{aligned} Z_k &= (E + \tau_k \tilde{F})^{-1} (E - \bar{\tau}_k \tilde{F}) Z_{k-1} (E - \tau_k \tilde{F})^T (E + \bar{\tau}_k \tilde{F})^{-T} \\ &\quad - 2\text{Re}(\tau_k) (E + \tau_k \tilde{F})^{-1} P_l \tilde{K} \tilde{K}^T P_l^T (E + \bar{\tau}_k \tilde{F})^{-T}. \end{aligned}$$

Since the solution of the PALE (28) also satisfies this equation, we get the following expression for the error

$$\begin{aligned} Z_k - Z &= (E + \tau_k \tilde{F})^{-1} (E - \bar{\tau}_k \tilde{F}) (Z_{k-1} - Z) (E - \tau_k \tilde{F})^T (E + \bar{\tau}_k \tilde{F})^{-T} \\ &= \dots = -\mathcal{A}_k Z \mathcal{A}_k^*, \end{aligned} \quad (30)$$

where $\mathcal{A}_k = (E + \tau_k \tilde{F})^{-1} (E - \bar{\tau}_k \tilde{F}) \dots (E + \tau_1 \tilde{F})^{-1} (E - \bar{\tau}_1 \tilde{F})$. If the pencil $\lambda E - \tilde{F}$ is stable, then Z_k converges toward the solution of the PALE (28). The rate of convergence depends strongly on the choice of the shift parameters. The optimal shift parameters providing the superlinear convergence satisfy the generalized ADI minimax problem

$$\{\tau_1, \dots, \tau_q\} = \arg \min_{\{\tau_1, \dots, \tau_q\} \in \mathbb{C}_-} \max_{t \in \text{Sp}_f(E, \tilde{F})} \frac{|(1 - \bar{\tau}_1 t) \dots (1 - \bar{\tau}_q t)|}{|(1 + \tau_1 t) \dots (1 + \tau_q t)|},$$

where $\text{Sp}_f(E, \tilde{F})$ denotes the finite spectrum of the pencil $\lambda E - \tilde{F}$. Similarly to [34], the suboptimal ADI parameters can be obtained from a set of largest and smallest in modulus approximate finite eigenvalues of $\lambda E - \tilde{F}$ computed by an Arnoldi procedure. Other parameter selection techniques developed for standard Lyapunov equations [10, 38, 47] can also be used for the PALE (28).

The ADI iteration is terminated if the normalized residual satisfies the condition

$$\varrho_L(Z_k) = \frac{\|\mathcal{L}(Z_k)\|_F}{\|P_l \tilde{K} \tilde{K}^T P_l^T\|_F} \leq \text{tol}, \quad (31)$$

where tol is a user defined tolerance, and

$$\mathcal{L}(Z_k) = \tilde{F} Z_k E^T + E Z_k \tilde{F}^T + P_l \tilde{K} \tilde{K}^T P_l^T$$

is the Lyapunov residual. The following theorem shows that even though $\mathcal{L}(Z_k)$ is a large and dense matrix, it has a low rank.

Theorem 4.1 *The Lyapunov residual at step k of the ADI iteration has the form*

$$\mathcal{L}(Z_k) = \tilde{\mathcal{A}}_k P_l \tilde{K} \tilde{K}^T P_l^T \tilde{\mathcal{A}}_k^*, \quad (32)$$

where $\tilde{\mathcal{A}}_k = (E - \bar{\tau}_k \tilde{F}) (E + \tau_k \tilde{F})^{-1} \dots (E - \bar{\tau}_1 \tilde{F}) (E + \tau_1 \tilde{F})^{-1}$.

Proof. It follows from (28) and (30) that

$$\begin{aligned} \mathcal{L}(Z_k) &= \tilde{F} Z_k E^T + E Z_k \tilde{F}^T + P_l \tilde{K} \tilde{K}^T P_l^T = \tilde{F} (Z_k - Z) E^T + E (Z_k - Z) \tilde{F}^T \\ &= -\tilde{F} \mathcal{A}_k Z \mathcal{A}_k^* E^T - E \mathcal{A}_k Z \mathcal{A}_k^* \tilde{F}^T. \end{aligned}$$

Since P_r and P_l are the spectral projectors onto the right and left deflating subspaces of the pencil $\lambda E - \tilde{F}$ corresponding to the finite eigenvalues, one can show using the Weierstrass canonical form of $\lambda E - \tilde{F}$ that

$$\begin{aligned} E(E + \tau_k \tilde{F})^{-1} (E - \bar{\tau}_k \tilde{F}) &= (E - \bar{\tau}_k \tilde{F}) (E + \tau_k \tilde{F})^{-1} E, \\ \tilde{F}(E + \tau_k \tilde{F})^{-1} (E - \bar{\tau}_k \tilde{F}) &= (E - \bar{\tau}_k \tilde{F}) (E + \tau_k \tilde{F})^{-1} \tilde{F}. \end{aligned}$$

Then we have $\tilde{F} \mathcal{A}_k Z \mathcal{A}_k^* E^T = \tilde{\mathcal{A}}_k \tilde{F} Z E^T \tilde{\mathcal{A}}_k^*$ and, hence,

$$\mathcal{L}(Z_k) = -\tilde{\mathcal{A}}_k (\tilde{F} Z E^T + E Z \tilde{F}^T) \tilde{\mathcal{A}}_k^* = \tilde{\mathcal{A}}_k P_l \tilde{K} \tilde{K}^T P_l^T \tilde{\mathcal{A}}_k^*. \quad \square$$

It follows from (32) that $\mathcal{L}(Z_k)$ is of rank at most g and its Frobenius norm can be computed as $\|\mathcal{L}(Z_k)\|_F = \|W_k W_k^*\|_F = \|W_k^* W_k\|_F$ with $W_k = \tilde{\mathcal{A}}_k P_l \tilde{K} \in \mathbb{R}^{n, g}$. We now show that the matrix W_k can be obtained iteratively at low cost.

It has been shown in [42] that the iterate Z_k can be determined in factored form $Z_k = \tilde{Z}_k \tilde{Z}_k^T$, where

$$\begin{aligned} V_1 &= (E + \tau_1 \tilde{F})^{-1} P_l \tilde{K}, & \tilde{Z}_1 &= \sqrt{-2\text{Re}(\tau_1)} V_1, \\ V_k &= V_{k-1} - (\bar{\tau}_{k-1} + \tau_k) (E + \tau_k \tilde{F})^{-1} \tilde{F} V_{k-1}, & \tilde{Z}_k &= [\tilde{Z}_{k-1}, \sqrt{-2\text{Re}(\tau_k)} V_k]. \end{aligned}$$

The iterate V_k can also be written as

$$\begin{aligned} V_k &= (E + \tau_k \tilde{F})^{-1} (E - \bar{\tau}_{k-1} \tilde{F}) V_{k-1} \\ &= (E + \tau_k \tilde{F})^{-1} (E - \bar{\tau}_{k-1} \tilde{F}) (E + \tau_{k-1} \tilde{F})^{-1} (E - \bar{\tau}_{k-2} \tilde{F}) V_{k-2} \\ &= \dots = (E + \tau_k \tilde{F})^{-1} \tilde{A}_{k-1} P_l \tilde{K} = (E + \tau_k \tilde{F})^{-1} W_{k-1} \end{aligned}$$

for $k \geq 2$. Then we have

$$\begin{aligned} W_k &= \tilde{A}_k P_l \tilde{K} = (E - \bar{\tau}_k \tilde{F}) (E + \tau_k \tilde{F})^{-1} \tilde{A}_{k-1} P_l \tilde{K} \\ &= (E - \bar{\tau}_k \tilde{F}) V_k = (E - \bar{\tau}_k \tilde{F}) (E + \tau_k \tilde{F})^{-1} W_{k-1} \\ &= (I - 2\text{Re}(\tau_k) \tilde{F} (E + \tau_k \tilde{F})^{-1}) W_{k-1} = W_{k-1} - 2\text{Re}(\tau_k) \tilde{F} V_k. \end{aligned}$$

Summarizing, we obtain the following algorithm for computing a low-rank approximate solution of the PALE (28).

Algorithm 6 The generalized LR-ADI for the projected Lyapunov equation

INPUT: $E, \tilde{F} \in \mathbb{R}^{n,n}$, $\tilde{K} \in \mathbb{R}^{n,g}$, projector P_l , shift parameters $\tau_1, \dots, \tau_q \in \mathbb{C}_-$, and tolerance tol .

OUTPUT: A low-rank approximation $Z \approx \tilde{Z}_k \tilde{Z}_k^T$ to the solution of the PALE (28).

1. $V_1 = (E + \tau_1 \tilde{F})^{-1} P_l \tilde{K}$,
2. $\tilde{Z}_1 = \sqrt{-2\text{Re}(\tau_1)} V_1$,
3. $W_1 = P_l \tilde{K} - 2\text{Re}(\tau_1) \tilde{F} V_1$, $k = 2$.
4. WHILE ($\|W_{k-1}^* W_{k-1}\| / \|\tilde{K}^T P_l^T P_l \tilde{K}\| \geq tol$ AND $k \leq k_{\max}$)
 - (a) $V_k = (E + \tau_k \tilde{F})^{-1} W_{k-1}$,
 - (b) $\tilde{Z}_k = [\tilde{Z}_{k-1}, \sqrt{-2\text{Re}(\tau_k)} V_k]$,
 - (c) $W_k = W_{k-1} - 2\text{Re}(\tau_k) \tilde{F} V_k$,
 - (d) $k \leftarrow k + 1$.

END

At each iteration we have $\tilde{Z}_k = [\sqrt{-2\text{Re}(\tau_1)} V_1, \dots, \sqrt{-2\text{Re}(\tau_k)} V_k] \in \mathbb{R}^{n,gk}$. To keep the low-rank structure in \tilde{Z}_k for large gk , we can compress the columns of \tilde{Z}_k using the rank-revealing QR factorization as described in [11]. Furthermore, in order to guarantee for the factors \tilde{Z}_k to be real in case of complex shift parameters, we take these parameters in complex conjugate pairs $\{\tau_k, \tau_{k+1} = \bar{\tau}_k\}$ and compute \tilde{Z}_k as in (33) if τ_k is real and

$$\begin{cases} \tilde{Z}_k = [\tilde{Z}_{k-1}, 2|\tau_k| \sqrt{-\text{Re}(\tau_k)} \hat{V}] \\ \tilde{Z}_{k+1} = [\tilde{Z}_k, 2\sqrt{-\text{Re}(\tau_k)} (V_k - \tau_{k+1} \hat{V})] \end{cases} \quad \text{if} \quad \begin{cases} \tau_k \text{ is complex} \\ \tau_{k+1} = \bar{\tau}_k \end{cases}, \quad (34)$$

where $\hat{V} = (E + \tau_{k+1} \tilde{F})^{-1} \tilde{F} V_k$, see [42] for details.

Finally, note that if the LR-ADI method is used in the inner Newton iteration, then we need to compute the products $(E + \tau_k \tilde{F})^{-1} w$ with some vector $w \in \mathbb{R}^n$ and $E + \tau_k \tilde{F} = (E + \tau_k F) + \tau_k K_j \tilde{G} P_r$ with low-rank matrices $\tilde{G} P_r \in \mathbb{R}^{p,n}$ and $K_j \in \mathbb{R}^{n,p}$ depending on the Lyapunov equation to be solved. For this purpose, we can use the Sherman-Morrison-Woodbury formula [18, Section 2.1.3]

$$(E + \tau \tilde{F})^{-1} w = w_1 - M_{K_j} \left((I_p + \tilde{G} P_r M_{K_j})^{-1} \tilde{G} P_r w_1 \right),$$

where $w_1 = (E + \tau_k F)^{-1} w$ and $M_{K_j} = \tau_k (E + \tau_k F)^{-1} K_j$ can be determined either by computing (sparse) LU factorizations and forward/backward substitutions or by using iterative Krylov subspace methods [37].

4.4 Stopping criteria for Newton's method

The iterations in Algorithms 4 and 5 can be stopped as soon as the normalized residual norm satisfies the condition

$$\varrho_R(\tilde{X}_j \tilde{X}_j^T) = \frac{\|\mathcal{R}(\tilde{X}_j \tilde{X}_j^T)\|}{\|P_l \tilde{H} \tilde{H}^T P_l^T\|} \leq \text{tol}, \quad (35)$$

where $\|\cdot\|$ is the spectral or Frobenius matrix norm and tol is a user-defined tolerance, or a stagnation of the residual norms is observed. Unfortunately, the computation of $\|\mathcal{R}(\tilde{X}_j \tilde{X}_j^T)\|$ by forming the residual matrix $\mathcal{R}(\tilde{X}_j \tilde{X}_j^T)$ is memory-intensive for large-scale problems. If \tilde{X}_j has a small number of columns n_j , then the residual norm can be determined efficiently using a factorization based approach proposed in [9] for standard Riccati equations. In Algorithm 4, we have

$$\mathcal{R}(\tilde{X}_j \tilde{X}_j^T) = \begin{bmatrix} P_l \tilde{H}, & F \tilde{X}_j, & E \tilde{X}_j \end{bmatrix} \begin{bmatrix} I_m & 0 & 0 \\ 0 & 0 & I_{n_j} \\ 0 & I_{n_j} & \tilde{X}_j^T \tilde{G}^T \tilde{G} \tilde{X}_j \end{bmatrix} \begin{bmatrix} \tilde{H}^T P_l^T \\ \tilde{X}_j^T F^T \\ \tilde{X}_j^T E^T \end{bmatrix}.$$

Then computing an "economy size" QR decomposition

$$\begin{bmatrix} P_l \tilde{H}, & E \tilde{X}_j, & F \tilde{X}_j \end{bmatrix} = Q^{(j)} R^{(j)}, \quad (36)$$

where $Q^{(j)}$ has orthonormal columns and $R^{(j)}$ has full row rank, we find

$$\|\mathcal{R}(\tilde{X}_j \tilde{X}_j^T)\| = \left\| R^{(j)} \begin{bmatrix} I_m & 0 & 0 \\ 0 & 0 & I_{n_j} \\ 0 & I_{n_j} & \tilde{X}_j^T \tilde{G}^T \tilde{G} \tilde{X}_j \end{bmatrix} (R^{(j)})^T \right\|.$$

A similar procedure can also be applied to determine $\|\mathcal{R}(\tilde{X}_{1,j} \tilde{X}_{1,j}^T - \tilde{X}_{2,j} \tilde{X}_{2,j}^T)\|$ in Algorithm 5. Thus, the evaluation of the residual norms reduces to the computation of the norm of much smaller matrices. Though this is much cheaper than computing the residual matrix explicitly, the verification of the stopping criterion (35) can still be much more expensive than computing \tilde{X}_j itself.

Another approach for computing the Riccati residuals in Algorithm 4 is based on the relation (16). If the PALE (17) is solved for N_j exactly, then (16) leads to

$$\|\mathcal{R}(X_j)\| = \|EN_{j-1} \tilde{G}^T \tilde{G} N_{j-1} E^T\| = \|K_j K_j^T\| = \|K_j^T K_j\|$$

with a small matrix $K_j^T K_j \in \mathbb{R}^{p \times p}$. However, if we solve the PALE (17) approximately using the LR-ADI method, then

$$\begin{aligned} \mathcal{R}(\tilde{X}_j \tilde{X}_j^T) &= K_j K_j^T + F_{j-1} \tilde{N}_{j-1} \tilde{N}_{j-1}^T E^T + E \tilde{N}_{j-1} \tilde{N}_{j-1}^T F_{j-1}^T + \mathcal{R}(\tilde{X}_{j-1} \tilde{X}_{j-1}^T) \\ &= \dots = K_j K_j^T + \mathcal{L}_{j-1}(\tilde{N}_{j-1} \tilde{N}_{j-1}^T) + \dots + \mathcal{L}_0(\tilde{N}_0 \tilde{N}_0^T), \end{aligned} \quad (37)$$

where $\mathcal{L}_k(\tilde{N}_k \tilde{N}_k^T) = F_k \tilde{N}_k \tilde{N}_k^T E^T + E \tilde{N}_k \tilde{N}_k^T F_k^T + P_l K_k K_k^T P_l^T$ is the Lyapunov residual with $K_0 = \tilde{H}$, $F_0 = F$, $K_k = E \tilde{N}_{k-1} \tilde{N}_{k-1}^T \tilde{G}^T$, and $F_k = F_{k-1} + K_k \tilde{G} P_r$ for $k = 1, \dots, j-1$. Thus, the Riccati residual can be estimated as

$$\|\mathcal{R}(\tilde{X}_j \tilde{X}_j^T)\| \leq \|K_j^T K_j\| + \|\mathcal{L}_{j-1}(\tilde{N}_{j-1} \tilde{N}_{j-1}^T)\| + \dots + \|\mathcal{L}_0(\tilde{N}_0 \tilde{N}_0^T)\|.$$

Note that (37) implies that the Lyapunov residuals accumulate during the Newton iteration. This means that the tolerance for the Lyapunov residuals should be taken smaller than the tolerance for the Riccati residuals.

In Algorithm 4, the Newton iteration can also be stopped as soon as the changes in \tilde{X}_j become small, i.e., $\|\tilde{N}_j\|_F / \|\tilde{X}_j\|_F \leq \text{tol}$.

4.5 Comparison of the Newton and Newton-Kleinman methods

We now compare the low-rank formulations of the Newton and Newton-Kleinman iterations with respect to complexity and numerical robustness.

Consider first the case when $\lambda E - F$ is stable. While in each Newton iteration in Algorithm 4, only one PALE (17) has to be solved, the Newton-Kleinman iteration in Algorithm 5 involves solving two PALEs (25) and (26) in each iteration plus one PALE (27) at the end. Since equations (25) and (26) differ in the right-hand side only, they can be solved simultaneously. If we solve the PALE

$$F_j Z_{j+1} E^T + E Z_{j+1} F_j^T = -P_l[\tilde{H}, K_j][\tilde{H}, K_j]^T P_l^T, \quad Z_{j+1} = P_r Z_{j+1} P_r^T \quad (38)$$

for the low-rank Cholesky factor $\tilde{Z}_{j+1}^{(k)}$ such that $Z_{j+1} \approx \tilde{Z}_{j+1}^{(k)} (\tilde{Z}_{j+1}^{(k)})^T$ using the generalized LR-ADI method in Algorithm 6, then the low-rank Cholesky factors $\tilde{X}_{1,j+1}$ and $\tilde{X}_{2,j+1}$ can be extracted from

$$\tilde{Z}_{j+1}^{(k)} = \left[\underbrace{Z_{11}}_m, \underbrace{Z_{12}}_p, \dots, \underbrace{Z_{k1}}_m, \underbrace{Z_{k2}}_p \right]$$

as $\tilde{X}_{1,j+1} = [Z_{11}, \dots, Z_{k1}]$ and $\tilde{X}_{2,j+1} = [Z_{12}, \dots, Z_{k2}]$. Since the right hand-side in the PALE (17) has smaller rank than that in the PALE (38), Algorithm 4 is less expensive than Algorithm 5. Moreover, the computation of the normalized residual in Algorithm 4 is much cheaper than that in Algorithm 5. Finally, as numerical experiments show, each Newton (outer) iteration requires usually less (inner) ADI iterations compared with the Newton-Kleinman step.

It should, however, be noted that Algorithm 4 can be used only if the pencil $\lambda E - F$ is stable, whereas Algorithm 5 can also be applied to unstable problems provided a stabilizing initial guess is available. Furthermore, the inexact version of the Newton method may be unstable. As shown in Section 4.4, due to the approximate solution of the PALE (17) in Algorithm 4, the residuals accumulate over the iterations that may even cause the iteration to diverge.

5 Numerical examples

In this section, we present some results of numerical experiments to demonstrate the properties of the presented methods for solving the PARE (2). As mentioned earlier, projected Riccati equations arise in balancing-related model reduction of the descriptor system

$$E\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t), \quad (39)$$

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. A transfer function of (39) is given by $\mathbf{G}(s) = C(sE - A)^{-1}B$. It can be additively decomposed as $\mathbf{G}(s) = \mathbf{G}_{\text{sp}}(s) + \mathbf{G}_0(s)$, where $\mathbf{G}_{\text{sp}}(s)$ is a strictly proper part of $\mathbf{G}(s)$ satisfying $\lim_{s \rightarrow \infty} \mathbf{G}_{\text{sp}}(s) = 0$ and $\mathbf{G}_0(s) = M_0 + sM_1 + \dots + s^d M_d$ is a polynomial part. In order to compute a reduced-order model for (39), we have to solve the PARE (2) (and also its dual), where the matrix coefficients have one of the following form depending on the applied balanced truncation approach [36]:

- positive real balanced truncation

$$\begin{aligned} E &= E, \quad F = A - P_l B (M_0 + M_0^T)^{-1} C P_r, \\ G &= C^T (M_0 + M_0^T)^{-1} C, \quad H = 0; \end{aligned} \quad (40)$$

- bounded real balanced truncation

$$\begin{aligned} E &= E, \quad F = A + P_l B (I - M_0^T M_0)^{-1} M_0^T C P_r, \\ G &= C^T (I - M_0 M_0^T)^{-1} C, \quad H = B (I - M_0^T M_0)^{-1} B^T; \end{aligned} \quad (41)$$

- bounded real balanced truncation via a Moebius transformation

$$\begin{aligned} E &= E, & F &= A - BC - 2\hat{P}_l B(I - \hat{M}_0^T \hat{M}_0)^{-1} \hat{M}_0^T C \hat{P}_r, \\ G &= 2C^T(I - \hat{M}_0 \hat{M}_0^T)^{-1} C, & H &= 2B(I - \hat{M}_0^T \hat{M}_0)^{-1} B^T, \\ P_r &= \hat{P}_r, & P_l &= \hat{P}_l, \end{aligned} \quad (42)$$

where \hat{P}_l and \hat{P}_r are the spectral projectors onto the left and right deflating subspaces of $\lambda E - A + BC$ corresponding to the finite eigenvalues and $\hat{M}_0 = I - 2 \lim_{s \rightarrow \infty} C(sE - A + BC)^{-1} B$.

Note that the PARE (2) with (42) is just the bounded real PARE of the Moebius-transformed system $\hat{G}(s) = (I - G(s))(I + G(s))^{-1} = \hat{C}(s\hat{E} - \hat{A})^{-1} \hat{B} + I$ with $\hat{E} = E$, $\hat{A} = A - BC$, $\hat{B} = -\sqrt{2}B$ and $\hat{C} = \sqrt{2}C$.

Example 5.1 The first example is a three-port RC circuit. This circuit is modelled by a descriptor system of index 1 in modified nodal analysis (MNA) form. It has $n = 2007$ state variables and $m = p = 3$ inputs and outputs. Under some conditions on circuit topology guaranteeing that the transfer function G is positive real, i.e., G is analytic in \mathbb{C}_+ and $G(s) + G(s)^* \geq 0$ for all $s \in \mathbb{C}_+$, one can show that the positive real PARE (2), (40) (the PR-PARE for short) and also the PARE (2), (42) (the BR(M)-PARE for short) are solvable. We compute the semi-stabilizing solutions of these equations using the Newton-Schur-Hammarling method as in Algorithm 3.

Figure 1(a) shows the normalized residual $\varrho_R(\tilde{X}\tilde{X}^T) = \|\mathcal{R}(\tilde{X}\tilde{X}^T)\|_F / \|P_l H P_l^T\|_F$. One can see that for both Riccati equations, the Newton iteration has a linear convergence only. This can be explained by the fact that \mathcal{R}'_{X_*} is singular. In Figure 1(b), we present the condition number $\kappa_2(E, F_j)$ of the operator \mathcal{R}'_{X_j} or, equivalently, of the PALE (9), which is defined as $\kappa_2(E, F_j) = 2\|E\|_2 \|F_j\|_2 \|Z_j\|_2$, where Z_j solves the PALE

$$E Z_j F_j^T + F_j Z_j E^T = -P_l P_l^T, \quad Z_j = P_r Z_j P_r^T,$$

see [39]. As expected, in both cases, the condition number increases, as X_j approaches X_* .

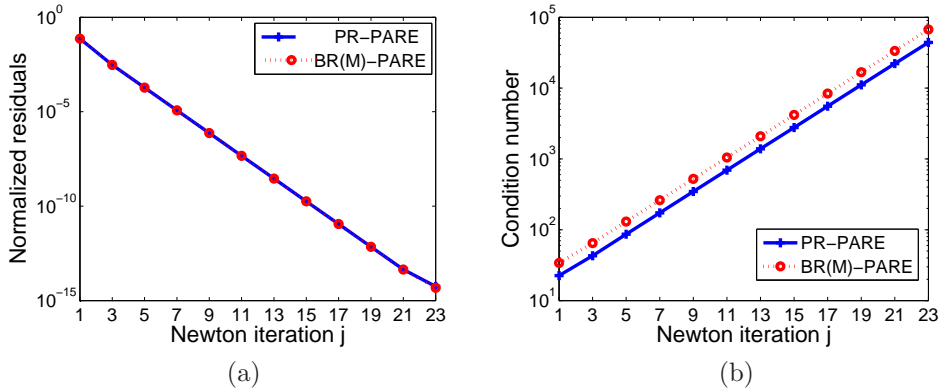


Figure 1: RC circuit: (a) the convergence history of the Newton-Schur-Hammarling method; (b) the condition number $\kappa_2(E, F_j)$ of the operator \mathcal{R}'_{X_j} .

Example 5.2 Consider the 2D instationary Stokes equation that describes the flow of an incompressible fluid in a domain. The spatial discretization of this equation by the finite volume method on a uniform staggered grid leads to the descriptor system (39) of index 2. The transfer function of this system is bounded real, i.e., G is analytic in \mathbb{C}_+ and $I - G(s)G(s)^* \geq 0$ for all $s \in \mathbb{C}_+$, that guarantees the solvability of the bounded real PARE (2), (41). We compute the low-rank approximations to the solution of this equation using the low-rank Newton (LR-N) and low-rank Newton-Kleinman (LR-NK) methods as in Algorithms 4 and 5, respectively, combined with the LR-ADI iteration. The inner ADI iterations have been stopped as soon as the normalized residuals

for the PALE (28) satisfy (31) with $tol = 10^{-13}$. In Table 1, we present the number j of outer Newton iterations (# outer), the normalized Riccati residuals $\varrho_R(X_j)$ given in (35), the number of the inner ADI iterations (# inner) and the reached normalized Lyapunov residuals $\varrho_L(\tilde{Z}_k \tilde{Z}_k^T)$ as in (31) for the LR-N and LR-NK methods. The problem dimensions are $n = 10679$ and $m = p = 5$. Figure 2(a) shows the normalized residuals $\varrho_R(X_j)$, whereas in Figure 2(b), we present the number of ADI iterations for both methods.

Table 1: Stokes equation: comparison of the Newton and Newton-Kleinman methods

Low-rank Newton method				Low-rank Newton-Kleinman method			
# outer	$\varrho_R(X_j)$	# inner	$\varrho_L(\tilde{Z}_k \tilde{Z}_k^T)$	# outer	$\varrho_R(X_j)$	# inner	$\varrho_L(\tilde{Z}_k \tilde{Z}_k^T)$
1	$5.363e-02$	27	$9.920e-13$	1	$5.363e-02$	27	$9.920e-13$
2	$3.912e-04$	23	$7.289e-13$	2	$3.912e-04$	25	$4.472e-13$
3	$3.487e-08$	20	$8.043e-13$	3	$3.487e-08$	25	$4.457e-13$
4	$1.032e-12$	20	$4.238e-13$	4	$4.910e-13$	25	$4.457e-13$

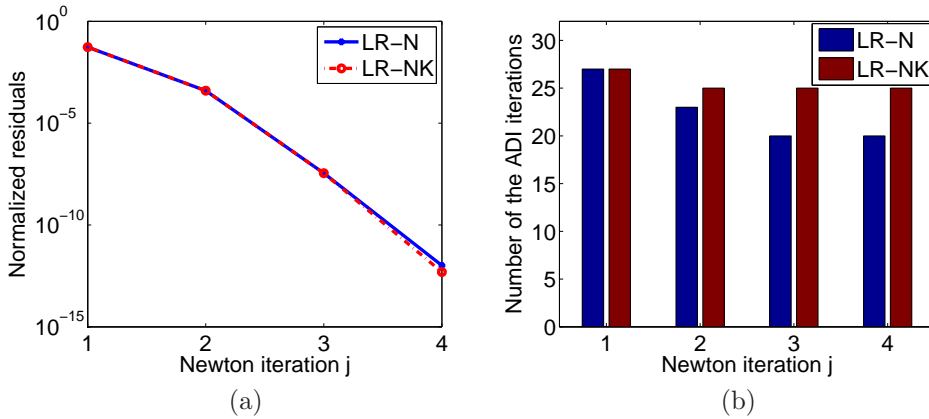


Figure 2: Stokes equation: (a) the convergence history of the low-rank Newton and low-rank Newton-Kleinman methods; (b) the number of ADI iterations required for solving the projected Lyapunov equations at each Newton iteration.

Example 5.3 Consider a constrained damped mass-spring system from [32]. The vibration of this system is described by the single-input single-output descriptor system (39) which is of index 3 and has a bounded real transfer function. We compute the low-rank approximations to the solution of the bounded real PARE (2), (41) using the low-rank Newton (LR-N) and low-rank Newton-Kleinman (LR-NK) methods as in Algorithms 4 and 5, respectively, combined with the LR-ADI iteration. For both methods, we present in Figure 3(a) a comparison of CPU time in seconds for problems of different state space dimension ranging from 101 to 10001. This figure confirms that the Newton-Kleinmann iteration is more expensive than the Newton iteration. Figure 3(b) shows the convergence history (the normalized residual norms) for the problem of state space dimension $n = 10001$.

6 Conclusions

In this paper, we have presented efficient and reliable numerical methods for solving projected Riccati equations as they arise in positive real and bounded real balanced truncation of descriptor systems. These methods are based on the Newton and Newton-Kleinman iterations. We have also considered the computation of the Cholesky factors and low-rank Cholesky factors of the stabilizing, positive semidefinite solutions of projected Riccati equations. The convergence analysis has been presented for Newton's iteration. The numerical experiments for different types of

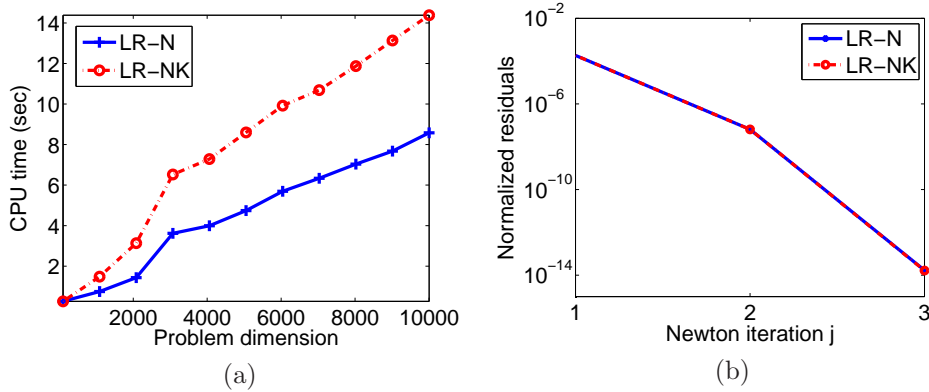


Figure 3: Mechanical system: (a) CPU time for the low-rank Newton and low-rank Newton-Kleinman methods; (b) the convergence history for the low-rank Newton method.

descriptor systems and different forms of projected Riccati equations illustrate the properties of the presented numerical algorithms.

In [49], a quadratic ADI method has been proposed for standard Riccati equations. An extension of this method to projected Riccati equations remains for future work.

Acknowledgement

The work described in this paper was supported by the Research Network MORESIM4NANO — *Model Reduction for Fast Simulation of New Semiconductor Structures for Nanotechnology and Microsystems Technology*, grants 05M10EVA and 05M10WAC, funded by the German Federal Ministry of Education and Science. Responsibility for the contents of this publication rests with the authors.

The authors would like to thank Jens Saak for helpful discussions.

References

- [1] L. AMODEI AND J.-M. BUCHOT, *A stabilization algorithm of the Navier-Stokes equations based on algebraic Bernoulli equation*, Numer. Linear Algebra Appl., 19 (2012), pp. 700–727.
- [2] S. BARRACHINA, P. BENNER, AND E. S. QUINTANA-ORTÍ, *Efficient algorithms for generalized algebraic Bernoulli equations based on the matrix sign function*, Numer. Algorithms, 46 (2007), pp. 351–368.
- [3] D. BENDER AND A. LAUB, *The linear-quadratic optimal regulator for descriptor systems*, IEEE Trans. Automat. Control, 32 (1987), pp. 672–688.
- [4] P. BENNER, *Contributions to the Numerical Solution of Algebraic Riccati Equations and Related Eigenvalue Problems*, Logos Verlag, Berlin, 1997.
- [5] P. BENNER, *Numerical solution of special algebraic Riccati equations via exact line search method*, in Proceedings of the European Control Conference (ECC97), Paper 786, Waterloo, Belgium, 1997, BELWARE Information Technology.
- [6] P. BENNER, *Partial stabilization of descriptor systems using spectral projection methods*, in Numerical Linear Algebra in Signals, Systems, and Control, S. Bhattacharyya, R. Chan, V. Olshevsky, A. Routray, and P. Van Dooren, eds., vol. 80 of Lecture Notes in Electrical Engineering, Springer-Verlag, Dordrecht, NL, 2011, pp. 55–76.

- [7] P. BENNER, V. HERNÁNDEZ, AND A. PASTOR, *On the Kleinman iteration for nonstabilizable systems*, Math. Control, Signals, Syst., 16 (2003), pp. 76–93.
- [8] P. BENNER, P. KÜRSCHNER, AND J. SAAK, *An improved numerical method for balanced truncation for symmetric second order systems*, Math. Comput. Model. Dyn. Syst., (to appear).
- [9] 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.
- [10] P. BENNER, H. MENA, AND J. SAAK, *On the parameter selection problem in the Newton-ADI iteration for large-scale Riccati equations*, Electron. Trans. Numer. Anal., 29 (2008), pp. 136–149.
- [11] P. BENNER AND E. QUINTANA-ORTÍ, *Solving stable generalized Lyapunov equations with the matrix sign function*, Numerical Algorithms, 20 (1999), pp. 75–100.
- [12] A. BUNSE-GERSTNER AND V. MEHRMANN, *A symplectic QR-like algorithm for the solution of the real algebraic Riccati equation*, IEEE Trans. Automat. Contr., 31 (1986), pp. 1104–1113.
- [13] R. BYERS, *Solving the algebraic Riccati equation with the matrix sign function*, Linear Algebra Appl., 85 (1987), pp. 267–279.
- [14] E.-W. CHU, H.-Y. FAN, AND W.-W. LIN, *A structure-preserving doubling algorithm for continuous-time algebraic Riccati equations*, Linear Algebra Appl., 396 (2005), pp. 55–80.
- [15] J. DEMMEL AND B. KÅGSTRÖM, *The generalized Schur decomposition of an arbitrary pencil $A - \lambda B$: Robust software with error bounds and applications. Part I: Theory and algorithms*, ACM Trans. Math. Software, 19 (1993), pp. 160–174.
- [16] ———, *The generalized Schur decomposition of an arbitrary pencil $A - \lambda B$: Robust software with error bounds and applications. Part II: Software and applications*, ACM Trans. Math. Software, 19 (1993), pp. 175–201.
- [17] F. GANTMACHER, *Theory of Matrices*, Chelsea Publishing Company, New York, 1959.
- [18] G. GOLUB AND C. V. LOAN, *Matrix Computations. 3rd ed*, The Johns Hopkins University Press, Baltimore, London, 1996.
- [19] S. GUGERCIN AND A. ANTOULAS, *A survey of model reduction by balanced truncation and some new results*, Internat. J. Control, 77 (2004), pp. 748–766.
- [20] C.-H. GUO AND P. LANCASTER, *Analysis and modification of Newton’s method for algebraic Riccati equations*, Math. Comp., 67 (1998), pp. 1089–1105.
- [21] V. IONESCU, C. OARĂ, AND M. WEISS, *Generalized Riccati Theory and Robust Control: A Popov Function Approach*, John Wiley and Sons, Chichester, UK, 1999.
- [22] I. JAIMOUKHA AND E. KASENALLY, *Krylov subspace methods for solving large Lyapunov equations*, SIAM J. Numer. Anal., 31 (1994), pp. 227–251.
- [23] K. JBILOU, *An Arnoldi based algorithm for large algebraic Riccati equations*, Appl. Math. Lett., 19 (2006), pp. 437–444.
- [24] B. KÅGSTRÖM AND L. WESTIN, *Generalized Schur methods with condition estimators for solving the generalized Sylvester equation*, IEEE Trans. Automat. Control, 34 (1989), pp. 745–751.

- [25] A. KAWAMOTO AND T. KATAYAMA, *The semi-stabilizing solution of generalized algebraic Riccati equation for descriptor systems*, Automatica, 38 (2002), pp. 1651–1662.
- [26] A. KAWAMOTO, K. TAKABA, AND T. KATAYAMA, *On the generalized algebraic Riccati equation for continuous-time descriptor systems*, Linear Algebra Appl., 296 (1999), pp. 1–14.
- [27] D. KLEINMAN, *On an iterative technique for Riccati equation computations*, IEEE Trans. Automat. Control, 13 (1968), pp. 114–115.
- [28] P. LANCASTER AND L. RODMAN, *The Algebraic Riccati Equation*, Oxford University Press, Oxford, 1995.
- [29] A. LAUB, *Schur method for solving algebraic Riccati equations*, IEEE Trans. Automat. Control, 24 (1979), pp. 913–921.
- [30] J.-R. LI AND J. WHITE, *Low rank solution of Lyapunov equations*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 260–280.
- [31] V. MEHRMANN, *The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution*, Lecture Notes in Control and Information Sciences, 163, Springer-Verlag, Heidelberg, 1991.
- [32] 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.
- [33] T. PENZL, *Numerical solution of generalized Lyapunov equations*, Adv. Comput. Math., 8 (1998), pp. 33–48.
- [34] ———, *A cyclic low-rank Smith method for large sparse Lyapunov equations*, SIAM J. Sci. Comput., 21 (1999/2000), pp. 1401–1418.
- [35] 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.
- [36] ———, *Positive real and bounded real balancing for model reduction of descriptor systems*, Internat. J. Control, 83 (2010), pp. 74–88.
- [37] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, MA, 1996.
- [38] J. SABINO, *Solution of Large-Scale Lyapunov Equations via the Block Modified Smith Method*, ph.D. thesis, Rice University, Houston, 2006.
- [39] T. STYKEL, *Numerical solution and perturbation theory for generalized Lyapunov equations*, Linear Algebra Appl., 349 (2002), pp. 155–185.
- [40] ———, *Stability and inertia theorems for generalized Lyapunov equations*, Linear Algebra Appl., 355 (2002), pp. 297–314.
- [41] ———, *A modified matrix sign function method for projected Lyapunov equations*, Systems Control 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] T. STYKEL AND V. SIMONCINI, *Krylov subspace methods for projected Lyapunov equations*, Appl. Numer. Math., 62 (2012), pp. 35–50.

- [44] A. VARGA, *On computing high accuracy solutions of a class of Riccati equations*, Control Theory Adv. Techn., 10 (1995), pp. 2005–2016.
- [45] ———, *Computation of normalized coprime factorizations of rational matrices*, Systems Control Lett., 33 (1998), pp. 37–45.
- [46] E. WACHSPRESS, *Iterative solution of the Lyapunov matrix equation*, Appl. Math. Lett., 1 (1988), pp. 87–90.
- [47] ———, *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.
- [48] D. WATKINS, *Performance of the QZ algorithm in the presence of infinite eigenvalues*, SIAM J. Matrix Anal. Appl., 22 (2000), pp. 364–375.
- [49] N. WONG AND V. BALAKRISHNAN, *Fast positive-real balanced truncation via quadratic alternating direction implicit iteration*, IEEE Trans. Computer-Aided Design Integr. Circuits Syst., 26 (2007), pp. 1725–1731.
- [50] L. ZHANG, J. LAM, AND S. XU, *On positive realness of descriptor systems*, IEEE Trans. Circuits Syst., 49 (2002), pp. 401–407.