Calcolo (2019) 56:51
https://doi.org/10.1007/s10092-019-0348-x

Solution formulas for differential Sylvester and Lyapunov equations

Maximilian Behr[1] · **Peter Benner**[1] · **Jan Heiland**[1,2]

Received: 21 December 2018 / Revised: 31 October 2019 / Accepted: 4 November 2019

Abstract
The differential Sylvester equation and its symmetric version, the differential Lyapunov equation, appear in different fields of applied mathematics like control theory, system theory, and model order reduction. The few available straight-forward numerical approaches when applied to large-scale systems come with prohibitively large storage requirements. This shortage motivates us to summarize and explore existing solution formulas for these equations. We develop a unifying approach based on the spectral theorem for normal operators like the Sylvester operator $\mathcal{S}(X) = AX + XB$ and derive a formula for its norm using an induced operator norm based on the spectrum of A and B. In view of numerical approximations, we propose an algorithm that identifies a suitable Krylov subspace using Taylor series and use a projection to approximate the solution. Numerical results for large-scale differential Lyapunov equations are presented in the last sections.

Keywords Differential Lyapunov equations · Differential Sylvester equations · Low-rank methods

Mathematics Subject Classification 15A24 · 65F60 · 65L05

✉ Maximilian Behr
behr@mpi-magdeburg.mpg.de

Peter Benner
benner@mpi-magdeburg.mpg.de

Jan Heiland
heiland@mpi-magdeburg.mpg.de; jan.heiland@ovgu.de

[1] Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany

[2] Faculty of Mathematics, Otto von Guericke University Magdeburg, Universitätsplatz 2, 39106 Magdeburg, Germany

Published online: 16 November 2019

Contents

1 Introduction

For coefficient matrices $A \in \mathbb{C}^{n\times n}$ and $B \in \mathbb{C}^{m\times m}$, an inhomogeneity $C \in \mathbb{C}^{n\times m}$, and an initial value $D \in \mathbb{C}^{n\times m}$, we consider the differential matrix equation

$$\begin{aligned} \dot{X}(t) &= AX(t) + X(t)B + C, \\ X(t_0) &= D, \end{aligned} \tag{1}$$

and provide formulas for the solution X with $X(t) \in \mathbb{C}^{n\times m}$. Equation (1) is commonly known as differential Sylvester equation (as opposed to the algebraic *Sylvester equation* $AX+BX+C = 0$). In the symmetric case when $B = A^T$, Eq. (1) and its algebraic counterpart are called differential and algebraic Lyapunov equations, respectively. In what follows, we will occasionally abbreviate Sylvester or Lyapunov equations by SLE.

In particular the differential Lyapunov equation is a useful tool for stability analysis and controller design for linear time-varying systems [2]. Equilibrium points of the differential Lyapunov equation, namely solutions of the algebraic Lyapunov equation, are used to construct quadratic Lyapunov functions for asymptotically stable linear time-invariant systems [33, Thm. 7.4.7]. The controllability and observability problem for linear time-varying systems is strongly connected to the solution of the differential Lyapunov equation [9, Ch. 13–14], [19, Ch. 3–4]. Other important applications lie in model order reduction [3] or in optimal control of linear time-invariant systems on finite time horizons [30]. Despite its importance, there have been but a few efforts to solve the differential Sylvester / Lyapunov or Riccati equation numerically, see [6,7,16, 21,25,26,31,36]. These algorithms are usually based on applying a time discretization and solving the resulting algebraic equations. Thus, even if the algebraic SLE are solved efficiently, the storage needed for the discrete solution at the time instances makes these direct approaches infeasible for large scale settings. Recently, Krylov subspace based methods were proposed in [12–14,24].

In an attempt to overcome this shortage, we revisit known solution formulas, develop alternative solution representations, and discuss their suitability for numerical approximations. We start with deriving a spectral decomposition for the Sylvester

operator $\mathcal{S}$ which allows functional calculus. We obtain formulas for the operator norm $\|\mathcal{S}\|$ as well as for $\mathcal{S}^{-1}$ and $e^{\mathcal{S}}$. This recovers previously known solution formulas. It will turn out that, in terms of efficiency, this solution representation is not well suited for approximation in general but, in special cases, allows for the construction or computation of exact solutions.

As a step towards efficient solution approximation, we use Taylor series expansions to identify suitable Krylov subspaces. For the differential Lyapunov equation with stable coefficient matrices and symmetric low-rank factored inhomogeneity, it is well-known that the solution of the algebraic Lyapunov equation spans a Krylov subspace under these assumptions. We split the solution of the differential Lyapunov equation in a constant and time dependent part, where the constant part is the solution of an algebraic Lyapunov equation. We approximate the time dependent part using the subspace spanned by the solution of the algebraic Lyapunov equation. The resulting algorithm overcomes the essential problems with storage consumption. Numerical results are presented in Sect. 7 and "Appendices B and C".

2 Preliminaries

In this section, we introduce the considered equations and the Sylvester operator, set the notation, and recall basic results.

The spectrum of a matrix $A \in \mathbb{C}^{n\times n}$ is denoted by $\Lambda(A) \subseteq \mathbb{C}$. A matrix is called stable if its spectrum is contained in the left open half plane $\mathbb{C}^-$. The Frobenius inner product on $\mathbb{C}^{n\times m}$ is given by $\langle A, B\rangle_F := \sum_{i=1}^{n}\sum_{j=1}^{m} A_{i,j}\overline{B_{i,j}}$. The Hadamard product is $A \odot B = \left(A_{i,j} \cdot B_{i,j}\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \in \mathbb{C}^{n\times m}$ for $A, B \in \mathbb{C}^{n\times m}$. The Hermitian transpose, transpose, conjugate are denoted by A^H, A^T, $\overline{A}$, respectively. Further, we will refer to the Kronecker delta: $\delta_{ij} = \begin{cases} 1 & \text{if } i = j,\\ 0 & \text{if } i \neq j \end{cases}$ and the Kronecker product: $A \otimes B = \left(A_{i,j} \cdot B\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}}$. The identity matrix in $\mathbb{C}^{d\times d}$ is denoted by $E_{d,d}$. For an interval $I \subseteq \mathbb{R}$ the set of continuous matrix-valued functions with complex entries is denoted by $\mathcal{C}(I, \mathbb{C}^{n\times m})$.

We start with a well known result on the solutions of differential Sylvester equations which concerns a more general case of the SLE (where the coefficient matrices may depend on time).

Theorem 1 (Existence and Uniqueness of Solutions, [1, Thm. 1.1.1., Thm. 1.1.3., Thm. 1.1.5]) *Let $I \subseteq \mathbb{R}$ be an open interval with $t_0 \in I$, $A \in \mathcal{C}(I, \mathbb{C}^{n\times n})$, $B \in \mathcal{C}(I, \mathbb{C}^{m\times m})$, $C \in \mathcal{C}(I, \mathbb{C}^{n\times m})$ and $D \in \mathbb{C}^{n\times m}$. The differential Sylvester equation*

$$\begin{aligned} \dot{X}(t) &= A(t)X(t) + X(t)B(t) + C(t),\\ X(t_0) &= D, \end{aligned}$$

has the unique solution $X(t) = \Phi_A(t,t_0)D\Phi_{B^H}(t,t_0)^H + \int_{t_0}^{t} \Phi_A(t,s)C(s)\Phi_{B^H}(t,s)^H \mathrm{d}s$.
Here $\Phi_A(t,t_0)$ *and* $\Phi_{B^H}(t,t_0)$ *are the unique state-transition matrices with respect to* $t_0 \in I$ *defined by*

$$\begin{aligned}
\dot{\Phi}_A(t,t_0) &:= \frac{\partial}{\partial t}\Phi_A(t,t_0) = A(t)\Phi_A(t,t_0),\\
\Phi_A(t_0,t_0) &= E_{n,n}.\\
\dot{\Phi}_{B^H}(t,t_0) &:= \frac{\partial}{\partial t}\Phi_{B^H}(t,t_0) = B(t)^H\Phi_{B^H}(t,t_0),\\
\Phi_{B^H}(t_0,t_0) &= E_{m,m}.
\end{aligned}$$

An application of Theorem 1 specifically to the autonomous case with constant coefficients by simply replacing the state transition matrices with the matrix exponentials yields the next result.

Theorem 2 *Let* $I \subseteq \mathbb{R}$ *be an open interval with* $t_0 \in I$, $A \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{m\times m}$, $C \in \mathcal{C}(I,\mathbb{C}^{n\times m})$ *and* $D \in \mathbb{C}^{m\times n}$. *The differential Sylvester equation*

$$\begin{aligned}
\dot{X}(t) &= AX(t) + X(t)B + C(t),\\
X(t_0) &= D,
\end{aligned}$$

has the unique solution

$$X(t) = e^{A(t-t_0)}De^{B(t-t_0)} + \int_{t_0}^{t} e^{A(t-s)}C(s)e^{B(t-s)}\mathrm{d}s. \tag{2}$$

Next, we restate basic properties of the Sylvester operator $\mathcal{S} : \mathbb{C}^{n\times m} \to \mathbb{C}^{n\times m}$, which, for given $A \in \mathbb{C}^{n\times n}$ and $B \in \mathbb{C}^{m\times m}$, is defined by

$$\mathcal{S}(X) = AX + XB. \tag{3}$$

The Sylvester operator $\mathcal{S}$ has been thoroughly studied in [10,22,23,35]. Among others, it has been shown that the eigenvalues and eigenvectors of the Sylvester operator $\mathcal{S}$ can be expressed in terms of the eigenvalues and eigenvectors of A and B, cf. [1, Rem. 1.1.2.], [3, Ch. 6.1.1], [20].

By exploiting the lemma presented next, it is possible to express the solution in (2) in terms of the Sylvester operator.

Lemma 1 (Sylvester Operator $\mathcal{S}$) *For the Sylvester operator* $\mathcal{S}\colon \mathbb{C}^{n\times m} \to \mathbb{C}^{n\times m}$ *and its partial realizations* $\mathcal{H}, \mathcal{V}\colon \mathbb{C}^{n\times m} \to \mathbb{C}^{n\times m}$, $\mathcal{H}(X) = AX$ *and* $\mathcal{V}(X) = XB$, *it holds that:*

– $\mathcal{S} = \mathcal{H} + \mathcal{V}$ *and* $\mathcal{H}\mathcal{V} = \mathcal{V}\mathcal{H}$,

– $e^{t\mathcal{S}} = e^{t\mathcal{H}} e^{t\mathcal{V}}$ *for all* $t \in \mathbb{R}$,

for any $A \in \mathbb{C}^{n\times n}$ *and* $B \in \mathbb{C}^{m\times m}$.

Proof The first claim can be confirmed by direct computations. The second claim is a standard result for commuting linear operators. □

By Lemma 1, formula (2) can be rewritten as

$$\begin{aligned} X(t) &= e^{(t-t_0)A} D e^{(t-t_0)B} + \int_{t_0}^{t} e^{(t-s)A} C(s) e^{(t-s)B} \mathrm{d}s = e^{(t-t_0)\mathcal{H}} e^{(t-t_0)\mathcal{V}}(D) \\ &\quad + \int_{t_0}^{t} e^{(t-s)\mathcal{H}} e^{(t-s)\mathcal{V}}(C(s)) \mathrm{d}s \\ &= e^{(t-t_0)\mathcal{S}}(D) + \int_{t_0}^{t} e^{(t-s)\mathcal{S}}(C(s)) \mathrm{d}s. \end{aligned} \tag{4}$$

3 Spectral decomposition of the Sylvester operator

In this section we show that the Sylvester operator $\mathcal{S}$, as defined in (3), is a normal operator if A and B are diagonalizable and if a suitably chosen inner product on a Hilbert space is considered. The inner product depends on the decomposition of A and B. Nevertheless, this approach will enable us to apply the spectral theorem and to derive a solution formula for the differential and algebraic SLE. This resembles the formulas of [11, Ch. 4.1.1], [18,34]. Those results were obtained by inserting the spectral decomposition into the SLE and by applying suitable algebraic manipulations and using the unrolled Kronecker representation of the SLE. Our strategy is to decompose the operator $\mathcal{S}$ first and then using functional calculus to obtain formulas for $e^{\mathcal{S}}$ and $\mathcal{S}^{-1}$. The eigenspaces of $\mathcal{S}$ can be constructed from the eigenspaces of A and B. The choice of the inner product ensures that the eigenvectors are orthonormal and $\mathcal{S}$ becomes a normal operator.

Lemma 2 (Inner product for $\mathcal{S}$) *Let* $A \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{m\times m}$ *be diagonalizable and* $C, D \in \mathbb{C}^{n\times m}$. *Furthermore let* $A = U D_A U^{-1}$ *and* $B^H = V D_{B^H} V^{-1}$ *be the spectral decompositions of* A *and* B^H *with* $U, D_A \in \mathbb{C}^{n\times n}$, $V, D_{B^H} \in \mathbb{C}^{m\times m}$ *and* D_A *and* D_{B^H} *be diagonal matrices.*

The following holds:

(i) $\langle X, Y\rangle_{U,V} := \langle U^{-1} X V^{-H}, U^{-1} Y V^{-H}\rangle_F$ *is an inner product on* $\mathbb{C}^{n\times m}$.
(ii) $(u_i v_j{}^H)_{\substack{i=1,\dots,n \\ j=1,\dots,m}}$ *is an orthonormal basis of* $\mathbb{C}^{n\times m}$ *with respect to* $\langle\cdot,\cdot\rangle_{U,V}$.
(iii) *The adjoint operator* $\mathcal{S}^* : \mathbb{C}^{n\times m} \to \mathbb{C}^{n\times m}$ *with respect to* $\langle\cdot,\cdot\rangle_{U,V}$ *is* $\mathcal{S}^*(X) = U\overline{D_A} U^{-1} X + X V^{-H} D_{B^H} V^H$.
(iv) $\mathcal{S}$ *is a normal operator with respect to* $\langle\cdot,\cdot\rangle_{U,V}$.

Proof It is straightforward to see that $\langle \cdot, \cdot \rangle_{U,V}$ is an inner product on $\mathbb{C}^{n\times m}$. Because of the identity

$$\langle u_i v_j^H, u_k v_l^H \rangle_{U,V} = \langle U^{-1} u_i v_j^H V^{-H}, U^{-1} u_k v_l^H V^{-H} \rangle_F = \langle e_i e_j^H, e_k e_l^H \rangle_F = \delta_{i,k}\delta_{j,l},$$

the matrices $u_i v_j^H \in \mathbb{C}^{n\times m}$ are orthogonal with respect to $\langle \cdot, \cdot \rangle_{U,V}$ and therefore linearly independent. As $\dim(\mathbb{C}^{n\times m}) = n \cdot m$, the tuple $(u_i v_j^H)_{\substack{i=1,\dots,n \\ j=1,\dots,m}}$ is an orthonormal basis of $\mathbb{C}^{n\times m}$.

Finally we show that, in this inner product, the adjoint of $\mathcal{S}$ is defined as $\mathcal{S}^*(X) = U\overline{D_A}U^{-1}X + XV^{-H}D_{B^H}V^H$ and that $\mathcal{S}\mathcal{S}^* = \mathcal{S}^*\mathcal{S}$ meaning that $\mathcal{S}$ is normal. In fact, for $X, Y \in \mathbb{C}^{n\times m}$,

$$\begin{aligned}
\langle S(X), Y\rangle_{U,V} &= \langle AX + XB, Y\rangle_{U,V} = \langle AX, Y\rangle_{U,V} + \langle XB, Y\rangle_{U,V} \\
&= \langle U^{-1}AXV^{-H}, U^{-1}YV^{-H}\rangle_F + \langle U^{-1}XBV^{-H}, U^{-1}YV^{-H}\rangle_F \\
&= \langle D_A U^{-1}XV^{-H}, U^{-1}YV^{-H}\rangle_F + \langle U^{-1}XV^{-H}\overline{D_{B^H}}, U^{-1}YV^{-H}\rangle_F \\
&= \langle U^{-1}XV^{-H}, \overline{D_A}U^{-1}YV^{-H} + U^{-1}YV^{-H}D_{B^H}\rangle_F \\
&= \langle U^{-1}XV^{-H}, U^{-1}\left(U\overline{D_A}U^{-1}Y + YV^{-H}D_{B^H}V^H\right)V^{-H}\rangle_F \\
&= \langle X, \left(U\overline{D_A}U^{-1}Y + YV^{-H}D_{B^H}V^H\right)\rangle_{U,V} = \langle X, \mathcal{S}^*(Y)\rangle_{U,V}
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{S}\mathcal{S}^*(X) &= \mathcal{S}(U\overline{D_A}U^{-1}X + XV^{-H}D_{B^H}V^H) = \mathcal{S}(U\overline{D_A}U^{-1}X) + \mathcal{S}(XV^{-H}D_{B^H}V^H) \\
&= UD_A\overline{D_A}U^{-1}X + U\overline{D_A}U^{-1}XV^{-H}\overline{D_{B^H}}V^H \\
&\quad + UD_AU^{-1}XV^{-H}D_{B^H}V^H + XV^{-H}D_{B^H}\overline{D_{B^H}}V^H \\
&= U\overline{D_A}D_AU^{-1}X + U\overline{D_A}U^{-1}XV^{-H}\overline{D_{B^H}}V^H \\
&\quad + UD_AU^{-1}XV^{-H}D_{B^H}V^H + XV^{-H}\overline{D_{B^H}}D_{B^H}V^H \\
&= \mathcal{S}^*(UD_AU^{-1}X) + \mathcal{S}^*(XV^{-H}\overline{D_{B^H}}V^H) = \mathcal{S}^*\mathcal{S}(X).
\end{aligned}$$

This means $\mathcal{S}$ and $\mathcal{S}^*$ commute and, therefore, by definition, $\mathcal{S}$ is normal. □

Now that we have an inner product on a Hilbert space for which $\mathcal{S}$ is normal, the second step is to compute the spectral decomposition of $\mathcal{S}$. The spectral decomposition allows functional calculus and we get a formula for $\mathcal{S}^{-1}$ and $e^{t\mathcal{S}}$. Since for normal operators, the operator norm is its spectral radius, we directly get a formula for the induced operator norm of $\mathcal{S}$. We remark that in the case that A and B are unitarily diagonalizable Lemma 2 also holds when $\langle \cdot, \cdot \rangle_{U,V}$ is replaced by the standard Frobenius inner product $\langle \cdot, \cdot \rangle_F$.

Lemma 3 (Spectral Decomposition of $\mathcal{S}$) *Let the assumptions of Lemma 2 hold. Moreover let $\alpha_1, \dots, \alpha_n \in \mathbb{C}$ and $\overline{\beta_1}, \dots, \overline{\beta_m} \in \mathbb{C}$ be the diagonal entries of D_A and D_{B^H}, respectively. Then we have*

(i) $\mathcal{S}(X) = \sum_{i=1}^{n}\sum_{j=1}^{m}(\alpha_i+\beta_j)\langle X, u_i v_j^H\rangle_{U,V} u_i v_j^H = U\left(\left(\alpha_i+\beta_j\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1} X V^{-H}\right) V^H,$

(ii) $\|\mathcal{S}\| = \max_{X\in\mathbb{C}^{n\times m}\setminus\{0\}} \frac{\|\mathcal{S}(X)\|_{U,V}}{\|X\|_{U,V}} = \max_{i,j}|\alpha_i+\beta_j|$, *where* $\|X\|_{U,V} = \sqrt{\langle X, X\rangle_{U,V}}$,

(iii) $e^{t\mathcal{S}}(X) = U\left(\left(e^{t(\alpha_i+\beta_j)}\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1} X V^{-H}\right) V^H.$

If in addition $\alpha_i+\beta_j \neq 0$ *for* $i = 1,\dots,n$ *and* $j = 1,\dots,m$, *then* $\mathcal{S}^{-1}(X) = U\left(\left(\frac{1}{\alpha_i+\beta_j}\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1} X V^{-H}\right) V^H.$

Proof

(i) From $AU = UD_A$ and $B^H V = V D_{B^H}$, we deduce $\mathcal{S}(u_k v_l^H) = A u_k v_l^H + u_k v_l^H B = (\alpha_k+\beta_l) u_k v_l^H$. Representing $\mathcal{S}(X) \in \mathbb{C}^{n\times m}$ as well as $X \in \mathbb{C}^{n\times m}$ with respect to the orthonormal basis $\left(u_i v_j^H\right)$ and exploiting linearity of $\mathcal{S}$ and $\langle\cdot,\cdot\rangle_{U,V}$ yield

$$
\begin{aligned}
\mathcal{S}(X) &= \sum_{i=1}^{n}\sum_{j=1}^{m}\langle \mathcal{S}(X), u_i v_j^H\rangle_{U,V} u_i v_j^H \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m}\langle \mathcal{S}(\sum_{k=1}^{n}\sum_{l=1}^{m}\langle X, u_k v_l^H\rangle_{U,V} u_k v_l^H), u_i u_j^H\rangle_{U,V} u_i v_j^H \\
&= \sum_{i,k=1}^{n}\sum_{j,l=1}^{m}\langle X, u_k v_l^H\rangle_{U,V}\langle \mathcal{S}(u_k v_l^H), u_i v_j^H\rangle_{U,V} u_i v_j^H \\
&= \sum_{i,k=1}^{n}\sum_{j,l=1}^{m}(\alpha_k+\beta_l)\langle X, u_k v_l^H\rangle_{U,V}\langle u_k v_l^H, u_i v_j^H\rangle_{U,V} u_i v_j^H \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m}(\alpha_i+\beta_j)\langle X, u_i v_j^H\rangle_{U,V} u_i v_j^H \\
&= U\left((\alpha_i+\beta_j)\langle X, u_i v_j^H\rangle_{U,V}\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} V^H \\
&= U\left(\left(\alpha_i+\beta_j\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot \left(\langle U^{-1} X V^{-H}, e_i e_j^H\rangle_F\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}}\right) V^H \\
&= U\left(\left(\alpha_i+\beta_j\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1} X V^{-H}\right) V^H.
\end{aligned}
$$

(ii) The claim about the norm follows from a direct application of the fundamental functional analytical result on compact normal operators, see, e.g., [37, Thm. VI.3.2].

(iii) With the spectral decomposition of $\mathcal{S}$ one can resort to functional calculus, cf. [32, Cor. 9.3.38], [37, Kor. IX.3.8], and obtain the formula for $\mathcal{S}^{-1}$ under the additional assumption that $\alpha_i + \beta_j \neq 0$.

□

Using the spectral decomposition and functional calculus we find that, under the assumptions of Lemma 2, the solution of the differential Sylvester equation

$$\begin{aligned}\dot{X}(t) &= AX(t) + X(t)B + C = \mathcal{S}(X(t)) + C,\\ X(0) &= D,\end{aligned}$$

has the form

$$\begin{aligned}X(t) &= e^{t\mathcal{S}}(D) + \int_0^t e^{(t-s)\mathcal{S}}(C)\mathrm{d}s\\ &= U\left(\left(e^{t(\alpha_i+\beta_j)}\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1}DV^{-H}\right)V^H\\ &\quad + \int_0^t U\left(\left(e^{(t-s)(\alpha_i+\beta_j)}\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1}CV^{-H}\right)V^H\mathrm{d}s\\ &= U\left(\left(e^{t(\alpha_i+\beta_j)}\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1}DV^{-H}\right.\\ &\quad \left. + \left(\int_0^t e^{(t-s)(\alpha_i+\beta_j)}\mathrm{d}s\right)_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \odot U^{-1}CV^{-H}\right)V^H,\end{aligned} \tag{5}$$

with the involved scalar integrals given explicitly as:

$$\int_0^t e^{(t-s)(\alpha_i+\beta_j)}\mathrm{d}s = \begin{cases}\frac{e^{t(\alpha_i+\beta_j)}-1}{\alpha_i+\beta_j} & \text{if } \alpha_i+\beta_j \neq 0,\\ t & \text{if } \alpha_i+\beta_j = 0\end{cases}.$$

4 Variation of constants

The application of the variation of constants formula leads to yet another solution formula for the SLE (1).

Lemma 4 (Variations of Constants, [9, Ch. 13]) *Let $A \in \mathbb{C}^{n\times n}$, $B \in \mathbb{C}^{m\times m}$, $C \in \mathbb{C}^{m\times n}$, $D \in \mathbb{C}^{m\times n}$ with $\Lambda(A) \cap \Lambda(-B) = \emptyset$. The differential Sylvester equation*

$$\begin{aligned}\dot{X}(t) &= AX(t) + X(t)B + C = \mathcal{S}(X(t)) + C,\\ X(0) &= D,\end{aligned}$$

has the solution

$$X(t) = e^{t\mathcal{S}}(D) + \mathcal{S}^{-1}(-C) - e^{t\mathcal{S}}\mathcal{S}^{-1}(-C). \tag{6}$$

Proof Because of $\Lambda(A) \cap \Lambda(-B) = \emptyset$, the inverse $\mathcal{S}^{-1}$ exists and we can rewrite the solution formula (4) as

$$\begin{aligned} X(t) &= e^{t\mathcal{S}}(D) + \int_0^t e^{(t-s)\mathcal{S}}(C)\mathrm{d}s = e^{t\mathcal{S}}(D) + \left[-\mathcal{S}^{-1}e^{(t-s)\mathcal{S}}(C)\right]_{s=0}^{s=t} \\ &= e^{t\mathcal{S}}(D) + \mathcal{S}^{-1}(-C) - \mathcal{S}^{-1}e^{t\mathcal{S}}(-C) \end{aligned}$$

and confirm that $X(0) = D + \mathcal{S}^{-1}(-C) - \mathcal{S}^{-1}(-C) = D$. □

From formula (6), we find that the solution can be written as the solution of the algebraic Sylvester equation and a time dependent part. We will make use of this fact in the numerical scheme that we propose in Sect. 6.

5 Solution as Taylor series

In this section we use Taylor series to derive a solution formula. From this we can read off suitable Krylov subspaces for our projection approach in the next section.

Lemma 5 (Taylor series solution formula) *Let* $A \in \mathbb{C}^{n\times n}, B \in \mathbb{C}^{m\times m}, C \in \mathbb{C}^{m\times n}, D \in \mathbb{C}^{m\times n}$. *The differential Sylvester equation*

$$\begin{aligned} \dot{X}(t) &= AX(t) + X(t)B + C = \mathcal{S}(X(t)) + C, \\ X(0) &= D, \end{aligned}$$

has the unique solution

$$X(t) = D + \sum_{k=1}^{\infty} \frac{t^k}{k!}(\mathcal{S}^k(D) + \mathcal{S}^{k-1}(C)). \tag{7}$$

Proof First observe that

$$\begin{aligned} ||D|| + \sum_{k=1}^{\infty} ||\frac{t^k}{k!}(\mathcal{S}^k(D) + \mathcal{S}^{k-1}(C))|| &\leq ||D|| + \sum_{k=1}^{\infty} \frac{|t|^k}{k!}(||\mathcal{S}^k(D)|| + ||\mathcal{S}^{k-1}(C)||) \\ &\leq ||D|| + \sum_{k=1}^{\infty} \frac{|t|^k}{k!}(||\mathcal{S}||^k||D|| + ||\mathcal{S}||^{k-1}||C||) \end{aligned}$$

$$
\begin{aligned}
&= ||D||e^{|t|||\mathcal{S}||} + ||C|| \sum_{k=0}^{\infty} \frac{|t|^{k+1}}{(k+1)!} ||\mathcal{S}||^k \\
&\leq ||D||e^{|t|||\mathcal{S}||} + |t|||C||e^{|t|||\mathcal{S}||}.
\end{aligned}
$$

The series converges absolutely and since $(\mathbb{C}^{n\times m}, ||\cdot||)$ is a Banach space, the series converges for every $t \in \mathbb{R}$. Therefore its radius of convergence is infinity, X is continuously differentiable and can be differentiated term-wise. Since, furthermore,

$$X(0) = D$$

and

$$
\begin{aligned}
\dot{X}(t) &= \sum_{k=1}^{\infty} \frac{t^{k-1}}{(k-1)!} (\mathcal{S}^k(D) + \mathcal{S}^{k-1}(C)) \\
&= \sum_{k=0}^{\infty} \frac{t^k}{k!} (\mathcal{S}^{k+1}(D) + \mathcal{S}^k(C)) = \mathcal{S}(D) + \sum_{k=1}^{\infty} \frac{t^k}{k!} (\mathcal{S}^{k+1}(D) + \mathcal{S}^k(C)) + C \\
&= \mathcal{S}(D + \sum_{k=1}^{\infty} \frac{t^k}{k!} (\mathcal{S}^k(D) + \mathcal{S}^{k-1}(C))) + C = \mathcal{S}(X(t)) + C,
\end{aligned}
$$

$X(t)$ is the unique solution. □

If we assume that D and C are given in factored form, then we can exploit this to rewrite the truncated series in a closed form of a matrix product.

Remark 1 Let $D = D_1 D_2^H$ and $C = C_1 C_2^H$ with $D_1 \in \mathbb{C}^{n\times d}$, $D_2 \in \mathbb{C}^{m\times d}$, $C_1 \in \mathbb{C}^{n\times c}$ and $C_2 \in \mathbb{C}^{m\times c}$. Then, having truncated the two parts of the series (7) after m_1 and m_2 summands, respectively, we can rewrite the solution approximation as

$$
\begin{aligned}
X_{m_1,m_2}(t) &= \sum_{k=0}^{m_1} \frac{t^k}{k!} \mathcal{S}^k(D) + \sum_{k=1}^{m_2} \frac{t^k}{k!} \mathcal{S}^{k-1}(C) \\
&= \sum_{k=0}^{m_1} \frac{t^k}{k!} \mathcal{S}^k(D_1 D_2^H) + \sum_{k=1}^{m_2} \frac{t^k}{k!} \mathcal{S}^{k-1}(C_1 C_2^H) \\
&= \sum_{k=0}^{m_1} \frac{t^k}{k!} (\mathcal{H} + \mathcal{V})^k (D_1 D_2^H) + \sum_{k=1}^{m_2} \frac{t^k}{k!} (\mathcal{H} + \mathcal{V})^{k-1} (C_1 C_2^H) \\
&= \sum_{k=0}^{m_1} \sum_{i=0}^{k} \frac{t^k}{k!} \binom{k}{i} \mathcal{H}^{k-i} \mathcal{V}^i (D_1 D_2^H) + \sum_{k=1}^{m_2} \sum_{i=0}^{k-1} \frac{t^k}{k!} \binom{k-1}{i} \mathcal{H}^{k-1-i} \mathcal{V}^i (C_1 C_2^H) \\
&= \sum_{k=0}^{m_1} \sum_{i=0}^{k} \frac{t^k}{k!} \binom{k}{i} A^{k-i} D_1 D_2^H B^i + \sum_{k=1}^{m_2} \sum_{i=0}^{k-1} \frac{t^k}{k!} \binom{k-1}{i} A^{k-1-i} C_1 C_2^H B^i,
\end{aligned}
$$

with the explicit representation of the sums

$$\sum_{k=0}^{m_1}\sum_{i=0}^{k}\frac{t^k}{k!}\binom{k}{i}A^{k-i}D_1D_2^HB^i$$
$$=\begin{bmatrix}D_1, AD_1, \dots, A^{m_1}D_1\end{bmatrix}\left(\begin{bmatrix}\frac{t^0}{0!}\binom{0}{0} & \frac{t^1}{1!}\binom{1}{1} & \frac{t^2}{2!}\binom{2}{2} & \cdots & \frac{t^m}{m_1!}\binom{m_1}{m_1}\\ \frac{t^1}{1!}\binom{1}{0} & \ddots & \ddots & \ddots & 0\\ \frac{t^2}{2!}\binom{2}{0} & \ddots & \ddots & \ddots & \vdots\\ \vdots & \ddots & \ddots & \ddots & 0\\ \frac{t^{m_1}}{m_1!}\binom{m_1}{0} & 0 & \cdots & 0 & 0\end{bmatrix}\otimes E_{d,d}\right)\begin{bmatrix}D_2^H\\ D_2^HB\\ \vdots\\ D_2^HB^{m_1}\end{bmatrix}$$

and

$$\sum_{k=1}^{m_2}\sum_{i=0}^{k-1}\frac{t^k}{k!}\binom{k-1}{i}A^{k-1-i}C_1C_2^HB^i$$
$$=\begin{bmatrix}C_1, AC_1, \dots, A^{m_2-1}C_1\end{bmatrix}\left(\begin{bmatrix}\frac{t^1}{1!}\binom{0}{0} & \frac{t^2}{2!}\binom{1}{1} & \frac{t^3}{3!}\binom{2}{2} & \cdots & \frac{t^{m_2}}{m_2!}\binom{m_2-1}{m_2-1}\\ \frac{t^2}{2!}\binom{1}{0} & \ddots & \ddots & \ddots & 0\\ \frac{t^3}{3!}\binom{2}{0} & \ddots & \ddots & \ddots & \vdots\\ \vdots & \ddots & \ddots & \ddots & 0\\ \frac{t^{m_2}}{m_2!}\binom{m_2-1}{0} & 0 & \cdots & 0 & 0\end{bmatrix}\otimes E_{c,c}\right)\begin{bmatrix}C_2^H\\ C_2^HB\\ \vdots\\ C_2^HB^{m_2-1}\end{bmatrix}. \tag{8}$$

6 Feasible numerical solution approaches

In this section, we briefly note that, for various reasons, all presented solution representations are not feasible for a straight-forward numerical approximation, in particular in a large-scale sparse setting.

A common reason is that none of the formulas supports a sparse representation of the solutions such that exorbitant amounts of memory will be required.

Limitations in memory will doubly affect the solution representation through the spectral decomposition (5) since also the basis matrices U and V are generically dense matrices. Apart from that, the computation of a spectral decomposition is typically computationally expensive and can be ill conditioned. Nonetheless, the spectral decomposition formula is useful to construct exact solutions for given coefficients with known spectral decompositions.

Another issue is the unfeasible computation of the full matrix exponential in all variants (2), (4), and (6) of the *variation of constants* formula. A possible remedy is the approximation of the action of the matrix exponential on a low-rank matrix in a Krylov subspace.

The approach to the solution via a Taylor series (see Sect. 5) seems best suited for the large-scale case since, at least in the symmetric case, the formulas provided in Remark 1 allow for a solution representation in factored form with the original coefficients. One problem here is that the truncated Taylor series only leads to good approximations locally around the point of expansion.

We will, however, exploit and combine certain parts of the solution representations to propose an algorithm for fast and memory efficient solution approximations.

We consider the stable, linear time-invariant case, i.e. we assume that $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, and $\Lambda(A) \subseteq \mathbb{C}^-$. We consider the differential Lyapunov equation

$$\dot{X}(t) = A^T X(t) + X(t)A + BB^T, \tag{9a}$$
$$X(0) = 0. \tag{9b}$$

By Lemma 4, we have that the solution splits into a constant part and a time dependent part. From Remark 1, we infer that the solution is contained in a Krylov subspace. We combine both observations in the following numerical solution approach:

1. *Factors of the time constant part as Krylov space basis*
With A stable, the associated algebraic Lyapunov $A^T X + XA + BB^T = 0$ has a unique symmetric positive semi-definite solution X_∞ that can be written in factored form $X_\infty = Z_\infty Z_\infty^T$, with $Z_\infty \in \mathbb{R}^{n\times q}$ and $\operatorname{rank}(Z_\infty) = q \leq n$. Moreover, since A is stable, it holds (see [9, Ch. 13]) that

$$\operatorname{range}(Z_\infty) = \operatorname{range}(X_\infty) = \operatorname{range}\left(\left[B, A^T B, \ldots, (A^T)^{n-1} B\right]\right).$$

2. *Factors of the time dependent part evolve in the same Krylov space*
With $X(t) = X_\infty + \tilde{X}(t)$, we obtain that

$$\dot{\tilde{X}}(t) = A^T \tilde{X}(t) + \tilde{X}(t)A,$$
$$\tilde{X}(0) = -X_\infty,$$

where $\tilde{X}(t)$ is given by $\tilde{X}(t) = -e^{tA^T} X_\infty e^{tA} = -e^{tA^T} Z_\infty Z_\infty^T e^{tA} =: -\tilde{Z}(t)\tilde{Z}(t)^T$. And since by (8) also $X(t)$ evolves in the same Krylov space[1], as does the difference $\tilde{X}(t)$ and, thus, $\tilde{Z}(t)$.

3. *Orthogonalize the basis and compute the time dependent factors*
By means of a (reduced) *Singular Value Decomposition* of Z_∞, we obtain orthogonal matrices $Q_\infty \in \mathbb{R}^{n\times q}$ and $V_\infty \in \mathbb{R}^{q\times q}$ with $\operatorname{range}(Q_\infty) = \operatorname{range}(Z_\infty)$ and $Z_\infty = Q_\infty S_\infty V_\infty^T$, where $S_\infty \in \mathbb{R}^{q\times q}$. Like Z_∞, the columns of Q_∞ span an A^T invariant subspace and with $Q_\infty^T Q_\infty = E_{q,q}$ it holds that

$$A^T Q_\infty = Q_\infty Q_\infty^T A^T Q_\infty,$$
$$e^{tA^T} Q_\infty = Q_\infty e^{tQ_\infty^T A^T Q_\infty},$$

from which we deduce that

$$\begin{aligned}\tilde{Z}(t)\tilde{Z}(t)^T &= \left(e^{tA^T} Z_\infty\right)\left(e^{tA^T} Z_\infty\right)^T = \left(e^{tA^T} Q_\infty S_\infty\right)\left(e^{tA^T} Q_\infty S_\infty\right)^T \\ &= \left(Q_\infty e^{tQ_\infty^T A^T Q_\infty} S_\infty\right)\left(Q_\infty e^{tQ_\infty^T A^T Q_\infty} S_\infty\right)^T.\end{aligned}$$

[1] In (8), replace $A \leftarrow A^T$, $B \leftarrow A$ and $C \leftarrow B$ to account for the Lyapunov equation setting.

We define $z(t) = e^{t\,Q_\infty^T A^T Q_\infty} S_\infty \in \mathbb{R}^{q\times q}$ and find that z can be obtained by solving

$$\dot{z}(t) = Q_\infty^T A^T Q_\infty z(t) \tag{10a}$$
$$z(0) = S_\infty, \tag{10b}$$

which is a matrix valued ODE that can be solved column-wise or by computing the matrix exponential $e^{t\,Q_\infty^T A^T Q_\infty}$.

The solution of the differential Lyapunov equation is, thus, given by $X(t) = Z_\infty Z_\infty^T - Q_\infty z(t) z(t)^T Q_\infty^T$.

Remark 2 The differential equation for z is of size $q \times q$, which can be much smaller than n, if the solution of the algebraic Lyapunov equation $X_\infty = Z_\infty Z_\infty^T$ has low-rank. Moreover, the orthogonalization of the basis allows for the detection of the numerical rank and for a compression of Z_∞ through truncating singular values that are smaller than a certain threshold.

We further note that, with minor adjustments, all arguments also hold for the generalized differential Lyapunov equation

$$M^T \dot{X}(t) M = A^T X(t) M + M^T X(t) A + B B^T,$$
$$X(0) = 0,$$

with $M \in \mathbb{R}^{n\times n}$ nonsingular and AM^{-1} stable that can accommodate, e.g., a mass matrix from a finite element discretization. We have included the derivation of the relevant formulas, and an example that illustrates a fundamental difference when M is symmetric positive definite in "Appendix A".

In summary, the proposed approach reads as written down in Algorithm 1.

7 Numerical results

7.1 Setup

To quantify and illustrate the performance of Algorithm 1, we consider differential Lyapunov equations that are used to define optimal controls for a finite element discretization of a heat equation; see [8] for the model description. Namely, we solve the differential Lyapunov equations:

$$M\dot{X}(t)M^T = AX(t)M^T + MX(t)A^T + BB^T, \quad X(0) = 0. \tag{DLE-1}$$

$$M^T\dot{X}(t)M = A^TX(t)M + M^TX(t)A + C^TC, \quad X(0) = 0. \tag{DLE-2}$$

that are defined through matrices $M, A \in \mathbb{R}^{n\times n}$ that are symmetric, M is positive definite and A stable, $B \in \mathbb{R}^{n\times 7}$ and $C \in \mathbb{R}^{6\times n}$. For computing the error, we precomputed the spectral decomposition of (A, M) and constructed the exact solution

Algorithm 1 Projection approach for generalized differential Lyapunov equations

Input: $M, A \in \mathbb{R}^{n\times n}$ with $\Lambda(AM^{-1}) \subseteq \mathbb{C}_-$ and $B \in \mathbb{R}^{n\times p}$.
Output: $X(t) = Z_\infty Z_\infty^T - Q_\infty z(t) z(t)^T Q_\infty^T$ that approximates the solution to
$M^T \dot{X}(t) M = A^T X(t) M + M^T X(t) A + BB^T,\ X(0) = 0.$

1: % Solve Lyapunov equation:
2: $A^T X_\infty M + M^T X_\infty A = -BB^T$ for $X_\infty \approx Z_\infty Z_\infty^T$ and $Z_\infty \in \mathbb{R}^{n\times q}$.

3: % Compute singular value decomposition:
4: $[Q_\infty, S_\infty, \sim] = \text{svd}(Z_\infty, 0)$.

5: % Set tolerance to largest singular value times machine epsilon:
6: $tol = \varepsilon_{\text{machine}} \cdot S_\infty(1,1)$.

7: % Truncate all singular values smaller than tolerance and get truncated low-rank factor:
8: $idx = \text{diag}(S_\infty) \geq tol$.
9: $S_\infty \leftarrow S_\infty(idx, idx)$.
10: $Q_\infty \leftarrow Q_\infty(:, idx)$.
11: $Z_\infty \leftarrow Q_\infty S_\infty$.

12: % Compute projected system and solve:
13: **if** M is symmetric positive definite **then**
14: $\quad M_F = Q_\infty^T M^T Q_\infty$.
15: $\quad A_F = Q_\infty^T A^T Q_\infty$.
16: **else**
17: $\quad M_F = E$.
18: $\quad A_F = Q_\infty^T M^{-T} A^T Q_\infty$.
19: **end if**

20: **for** k=1,..., cols (S_∞) **do**
21: $\quad$ Solve: $M_F \dot{z}(:,k)(t) = A_F z(:,k)(t),\ z(:,k)(0) = S_\infty(:,k)$.
22: **end for**

by means of the formula from Eq. (5). The memory consuming computation of the spectral decomposition was done on a compute server with 4 × Xeon® CPU E7-8837 @ 2.67 GHz with 8 cores and 1 TB Ram and MATLAB® 2015b. All other computations were carried out on a machine with 2 × Xeon® CPU E5-2640 v3 @ 2.60 GHz with 8 Cores and 64 GB Ram and MATLAB 2017a. We have used the low-rank ADI iteration implemented in MEX-M.E.S.S. [4] to solve the algebraic Lyapunov equations; as required for Algorithm 1 (Step 2).

We solve the resulting projected ODE system column-wise using MATLAB ODE solvers `ode45`, `ode23`, `ode113`, `ode15s`, `ode23s`, `ode23t` and `ode23tb`. The solvers `ode45`, `ode23`, `ode113` are nonstiff solvers, whereas the rest are stiff solvers. We have used the following parameters for the `odeset` function:

- `RelTol`: $1 \cdot 10^{-9}$
- `AbsTol`: $1 \cdot 10^{-10}$
- `Stats`: `off`
- `NormControl`: `off`
- `BDF`: `on`
- `Jacobian`: A_F
- `JPattern`: `logical` (A_F)
- `Mass`: M_F

– `MassSingular: no`
– `MStateDependence: none`

As the time interval, we consider [0, 4500].

7.2 Projection approach

The initial step of Algorithm 1 requires the solutions to the associated algebraic Lyapunov equations. For this task we call MEX-M.E.S.S. that iteratively computes the solutions up to the following absolute and relative residuals

$$||AZ_\infty Z_\infty^T M^T + MZ_\infty Z_\infty^T A^T + BB^T||_2 \text{ or } ||A^T Z_\infty Z_\infty^T M + M^T Z_\infty Z_\infty^T A + C^T C||_2.$$

and

$$\frac{||AZ_\infty Z_\infty^T M^T + MZ_\infty Z_\infty^T A^T + BB^T||_2}{||BB^T||_2} \text{ or } \frac{||A^T Z_\infty Z_\infty^T M + M^T Z_\infty Z_\infty^T A + C^T C||_2}{||C^T C||_2}.$$

The achieved values for the different test setups as well as the number of columns of the corresponding Z_∞ after truncation (see Step 7 of Algorithm 1), that define the dimension of the reduced model for the time dependent part, are listed in Tables 1 and 2.

We report the absolute and relative errors

$$||X(t) - X_{ref}(t)||_2 \quad \text{and} \quad \frac{||X(t) - X_{ref}(t)||_2}{||X_{ref}(t)||_2},$$

where X is the numerical solution obtained from Algorithm 1 with various ODE solvers and where the reference solution X_{ref} was obtained from the spectral decomposition

Table 1 Residuals for $AXM^T + MXA^T + BB^T = 0$

Size	Size of $z(t)$	Absolute residual	Relative residual
1357	261 × 261	3.413488×10^{-25}	7.748357×10^{-12}
5177	302 × 302	1.037846×10^{-25}	4.728703×10^{-12}
20209	376 × 376	6.053185×10^{-26}	5.525974×10^{-12}

Table 2 Residuals for $A^T XM + M^T XA + C^T C = 0$

Size	Size of $z(t)$	Absolute residual	Relative residual
1357	230 × 230	1.011843×10^{-10}	8.432027×10^{-12}
5177	259 × 259	5.595100×10^{-11}	4.662583×10^{-12}
20209	310 × 310	4.382439×10^{-11}	4.382439×10^{-12}

of (A, M). The spectral decomposition of (A, M) is $AV = MVD$, where V contains the eigenvectors and D has the corresponding eigenvalues of $M^{-1}A$ on the diagonal. According to Eq. (5) the reference solution for (DLE-1) can be formed as

$$X_{ref}(t) = V \left(\left(\int_0^t e^{(t-s)\left(D_{i,i}+\overline{D_{j,j}}\right)} \mathrm{d}s \right)_{\substack{i=1,\dots,n \\ j=1,\dots,n}} \odot \left(V^{-1}M^{-1}B\right)\left(V^{-1}M^{-1}B\right)^H \right) V^H.$$

As the matrices A and M are real symmetric and M is positive definite, the eigenproblem for (A, M) is a generalized symmetric definite one. Therefore all eigenvalues are real and the corresponding system of eigenvectors can be chosen real and orthogonal with respect to M, that is $V^T MV = E_{n,n}$. The representation of the reference solution simplifies as follows

$$X_{ref}(t) = V \left(\left(\int_0^t e^{(t-s)\left(D_{i,i}+D_{j,j}\right)} \mathrm{d}s \right)_{\substack{i=1,\dots,n \\ j=1,\dots,n}} \odot \left(V^T B\right)\left(V^T B\right)^T \right) V^T.$$

We plot the numerical errors and $||X_{ref}(t)||_2$ on the initial short time interval [0, 10], where most of the evolution is happening, and on the full time interval [0, 4500] in "Appendix B", Figs. 4a–d, 5a–d, 6a–d, 7a–d, 8a–d and 9a–d.

In view of the performance of the different ODE solvers, we can interpret the presented numbers and plots as follows: the solver `ode15s`, which is a stiff solver of variable order, performs best in time and accuracy. Due to the stiffness, the error is oscillating for the solvers `ode45`, `ode23`, `ode113`. Note that the discrete Laplacian that is encoded in the coefficient matrix A becomes stiffer with a finer space discretization, i.e. for larger n. Accordingly, the computational times for the non-stiff solvers grow with n at a higher rate than the stiff solvers; see Fig. 2.

The solution of (DLE-2) itself is large in norm which makes the relative error stagnate around the prescribed tolerance and the absolute error comparatively large; see the plots in "Appendix B.2". Particularly, the non-stiff solvers (and `ode15s`) achieve this error level and the oscillations as stiffness is dominated by the approximation error. This might be the reason, why for (DLE-2), despite the fact that the coefficient matrices are still stiff, the non-stiff solvers perform better than the stiff solvers (except `ode15s`). Nonetheless, again the computation times for the non-stiff solver grow at a higher rate with the increasing stiffness that comes with increasing n; see Fig. 2.

Because $X(t) \to 0$ for $t \searrow 0$, the plots for the relative error spread out for small times.

Except `ode15s`, there is no general rule which solver performs better in terms of computational time; see Fig. 2. The timings may change, when different relative and absolute error tolerances are used in the MATLAB `odeset` function.

Finally, we want to make the following remark: instead of integrating the projected ODE (10) which is linear with constant coefficients of moderate dimension, one may consider using the *Schur–Parlett* [17, Ch. 10] algorithm to compute the matrix expo-

Code and Data Availability
The source code of the implementations used to compute the presented results is available from:

`doi:10.5281/zenodo.1484327`
`https://gitlab.mpi-magdeburg.mpg.de/behr/diff_lyap_eqn_solution_formulas_release`

under the GPLv2+ license and is authored by Maximilian Behr.

Fig. 1 Link to code and data

nential. The initialization efforts of the *Schur–Parlett* algorithm will pay off, if the matrix exponential has to be evaluated for many different values of t. Also, asymptotically, the storage requirements for $z(t)$ will be lifted, since the matrix exponential for a given t can be computed on demand. Nonetheless, we used the ODE approach, which we think is more efficient because of the sophisticated MATLAB ODE solver implementations that come, e.g, with step size selection methods integrated.

The code of the implementation with the precomputed spectral decompositions needed to construct the exact solution is available as mentioned in Fig. 1.

7.3 Computational time

7.4 Comparison with backward differentiation formulas

To benchmark our method, we have run comparison tests with the MATLAB implementation of the Backward Differentiation Formulas / Alternating Direction Implicit (BDF/ADI) scheme as developed in [29] (see also the "Appendix C" for a short summary of the algorithm). The numerical experiments were conducted on the same machine, with the same MATLAB version and the same model as described in Sect. 7.1.

In contrast to the Algorithm 1 that needs to solve only a single algebraic Lyapunov equation for the initialization, the BDF/ADI approach solves a Lyapunov equation in every time step. Moreover, the numerical solution is stored in LDL^T-format, i.e., in terms of the factors of $X(t_k) \approx L_k D_k L_k^T$ and $L_k \in \mathbb{R}^{n \times l_k}$, which grow at least linearly with the size of n and the number of time steps. For this reason, we had to restrict our numerical experiments with BDF/ADI to the interval $[0, 100]$ and consider only the model of the smallest size $n = 1357$. As for the test of our Algorithm 1 in "Appendix B", for computing the error, we used an exact solution X_{ref} based on the spectral decomposition of (A, M).

We have compared various BDF methods that we abbreviated as `BDF1`, `BDF2`, `BDF3`, `BDF4`, `BDF5` and `BDF6`, where the number denotes the order s of the method. We used the constant time step sizes $\tau_k := h \in \{2^{-4}, 2^{-6}, 2^{-8}\}$ for our computations. In "Appendices C.1 and C.2", we plot the relative and absolute errors compared to the solution obtained by the spectral decomposition, c.f. Figs. 10a–f and 11a–f. For comparison, we also plot the error of the numerical solution obtained by Algorithm 1 and the MATLAB ODE solver `ode15s`. We list the computational times for performing the BDF/ADI method based computations in Sect. 7.3.

As the actual solution X converges towards the solution of the algebraic Lyapunov equation, also the numerical errors show a decay towards a certain error level. Decreasing the step size lowers this level accordingly. This effect is visible only for methods of lower order ($s \leq 2$). For higher orders, the error level stagnates and the error rather

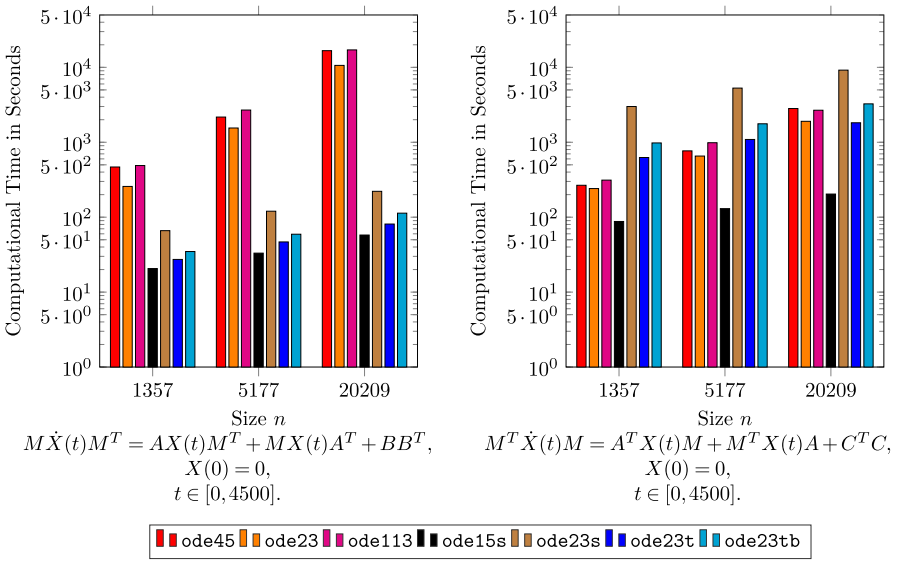

Fig. 2 Timings of the different MATLAB ODE solvers for the solution of the projected time-dependent factors as defined in Eq. (10)

shows oscillations, which are likely due to errors in the solution of the Lyapunov equations. Since all BDF methods are stiff solvers, there are no oscillations of higher frequency to observe. The error levels are comparable to the level reached by our approach with `ode15s`.

The computational timings for the BDF/ADI solvers are nearly the same for same step sizes; cf. Fig. 3. Accordingly, the higher order methods clearly outperform the low-order methods.

As for the comparison to our approach, we note that the reported timings were for the longer time-interval [0, 4500]. However, since the transient behavior of the solution is confined to a short initial phase and since the ODE solvers have a step size control, a restriction to a shorter interval would not change the timings by much.

For the test case with (DLE-1), the BDF/ADI schemes achieved the same accuracy as our approach already with the coarsest step size; see Fig. 10a–f. Thus, we compare the timings in the left most columns in Fig. 2 ($n = 1357$) and Fig. 3 ($h = 2^{-4}$) to conclude that our approach with the best performing ODE solver is about 25 times faster.

As for the test case with (DLE-2), the BDF/ADI schemes reached the same accuracy only for the finest chosen step size; see Fig. 11a–f. Thus, comparing the right most column in Fig. 3 ($h = 2^{-8}$) to the left column in the right plot of Fig. 2 ($n = 1357$), we find that our Algorithm 1, again, is faster by a factor of 30.

To conclude the comparison, we add the following remarks. With the costs of solving Lyapunov equations, apart from the increased memory requirements, the BDF/ADI scheme will also become significantly more costly for larger system sizes. As opposed to our Algorithm 1 however, the BDF/ADI scheme also applies for the time varying case as well as for nonzero initial conditions. Moreover, as `ode15s` in fact uses the

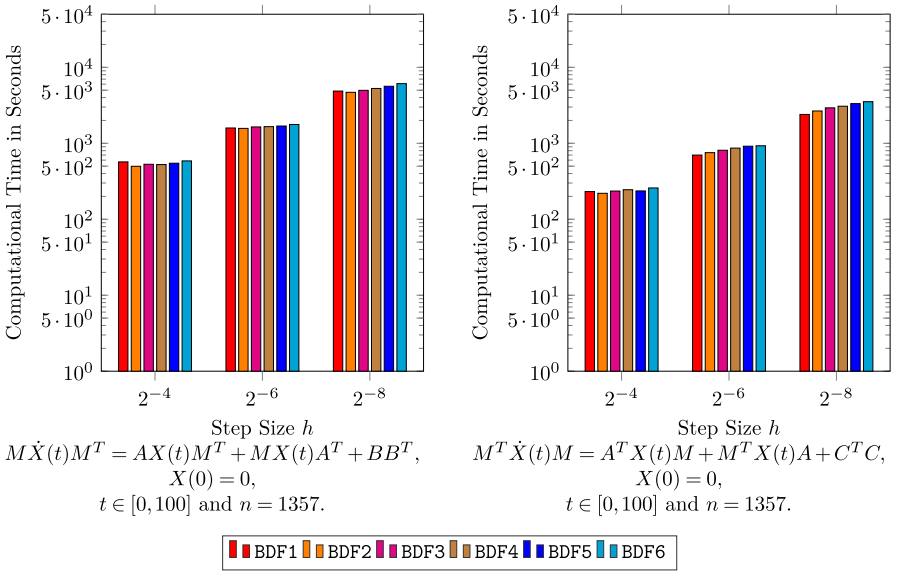

Fig. 3 Timings of the different BDF/ADI solvers

BDF formulas but with variable order s and variable time step, one may consider integrating similar error control mechanisms in the BDF/ADI approach to improve performance.

8 Conclusions

We presented several solution formulas for the differential Sylvester and Lyapunov equations. For the autonomous stable differential Lyapunov equation, we proposed a numerical algorithm that combined certain aspects derived from the solution representations. The main feature of the algorithm is the projection of the time dependent part onto a suitable subspace of lower dimension. Only this makes the numerical solution feasible in terms of memory requirements. As for the computational time, the projected system can be directly solved with optimized ODE solvers like the `ode-suite` in MATLAB. Moreover, its structure allows for column-wise computation of the factors and, thus, for straight-forward parallelization.

We illustrated the performance of the algorithm in an example that was derived from a finite-element discretization of a heat equation. The achieved accuracy is fully satisfactory. The greatest benefit, also in contrast to existing numerical schemes, is the low memory requirement.

The possible extension to the unstable or unsymmetric as well as to the non-autonomous case is not straightforward since the space in which the solution evolves has not been found to span a suitable invariant Krylov subspace and is possibly of high dimension. Moreover, the solution of $AX = C$ (which is the special case of $AX + XB = C$ with $B = 0$) is unlikely to span an A-invariant subspace at all,

meaning that the span of the solution of the algebraic Sylvester equation is not suited for the differential equation. Thus, for the general case, a different ansatz is needed. The same is true for time-dependent coefficients or non-zero initial conditions. Here, a remedy might be structured approaches for the case that time dependence comes, e.g., from a low-rank update.

In the unstable case, apart from the fact that the algebraic Lyapunov equation may not have a unique solution, there can be modes that grow exponentially in time. For this case it may be worth investigating whether a projector that identifies the stable part of the underlying mathematical model can be efficiently incorporated in the proposed solution approach.

Acknowledgements Open access funding provided by Max Planck Society.

A Derivation of algorithm 1 for generalized differential Lyapunov equations

We consider the generalized differential Lyapunov equation

$$M^T \dot{X}(t) M = A^T X(t) M + M^T X(t) A + B B^T, \tag{12a}$$

$$X(0) = 0. \tag{12b}$$

We assume that $A, M \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, M is nonsingular and AM^{-1} is stable. Let $X_\infty = Z_\infty Z_\infty^T$ be the unique symmetric positive semidefinite solution of the generalized algebraic Lyapunov equation

$$0 = A^T X M + M^T X A + B B^T. \tag{13}$$

We insert the ansatz $X(t) = X_\infty + \tilde{X}(t)$ into Eq. (12) and deduce that $\tilde{X}$ fulfills

$$\dot{\tilde{X}}(t) = (AM^{-1})^T \tilde{X}(t) + \tilde{X}(t)(AM^{-1}),$$
$$\tilde{X}(0) = -X_\infty.$$

Therefore we have that $\tilde{X}(t) = -e^{(AM^{-1})^T} X_\infty e^{(AM^{-1})}$. The solution X_∞ of the generalized algebraic Lyapunov equation (13) satisfies the algebraic Lyapunov equation

$$0 = (AM^{-1})^T X + X(AM^{-1}) + \left(M^{-T} B\right)\left(M^{-T} B\right)^T.$$

By assumption AM^{-1} is stable and it follows (see [9, Ch. 13]) that

$$\begin{aligned}\operatorname{range}(Z_\infty) &= \operatorname{range}(X_\infty)\\ &= \operatorname{range}\left(\left[M^{-T}B, (AM^{-1})^T M^{-T}B, \ldots, \left((AM^{-1})^T\right)^{n-1} M^{-T}B\right]\right),\end{aligned}$$

and $\operatorname{range}(Z_\infty)$ is a $\left(AM^{-1}\right)^T$ invariant subspace. By using the reduced SVD we obtain $Z_\infty = Q_\infty S_\infty V_\infty^T$ and as well as $\operatorname{range}(Z_\infty) = \operatorname{range}(Q_\infty)$. Now we see that

$$\begin{aligned}\tilde{X}(t) &= -e^{(AM^{-1})^T} X_\infty e^{(AM^{-1})} = -\left(e^{(AM^{-1})^T} Z_\infty\right)\left(e^{(AM^{-1})^T} Z_\infty\right)^T\\ &= -\left(e^{(AM^{-1})^T} Q_\infty S_\infty\right)\left(e^{(AM^{-1})^T} Q_\infty S_\infty\right)^T =: \tilde{Z}(t)\tilde{Z}(t)^T.\end{aligned}$$

We use that $\operatorname{range}(Q_\infty)$ is an $\left(AM^{-1}\right)^T$ invariant subspace and get

$$\tilde{Z}(t) = e^{(AM^{-1})^T} Q_\infty S_\infty = Q_\infty e^{Q_\infty^T (AM^{-1})^T Q_\infty} S_\infty =: Q_\infty z(t). \tag{14}$$

The function $\tilde{Z}$ satisfies the ordinary differential equations

$$M^T \dot{\tilde{Z}}(t) = A^T \tilde{Z}(t), \tag{15a}$$
$$\tilde{Z}(0) = Z_\infty = Q_\infty S_\infty. \tag{15b}$$

and

$$\dot{\tilde{Z}}(t) = M^{-T} A^T \tilde{Z}(t), \tag{16a}$$
$$\tilde{Z}(0) = Z_\infty = Q_\infty S_\infty. \tag{16b}$$

If M is symmetric positive definite we make use of the Eqs. (14) and (15) and obtain the ordinary differential equation

$$Q_\infty^T M^T Q_\infty \dot{z}(t) = Q_\infty^T A^T Q_\infty z(t), \tag{17a}$$
$$z(0) = S_\infty. \tag{17b}$$

If M is not symmetric positive definite, Eqs. (14) and (16) leads to

$$\dot{z}(t) = Q_\infty^T M^{-T} A^T Q_\infty z(t), \tag{18a}$$
$$z(0) = S_\infty. \tag{18b}$$

It is necessary to distinguish between the cases, that M is symmetric positive definite and M is nonsingular. If M is nonsingular, but not symmetric positive definite, the matrix $Q_\infty^T M^T Q_\infty$ might be singular and (17) may not have a unique solution. We illustrate the situation by an example.

Example 1 Let $A = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$, $M = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ and $B = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ be. The unique solution of the generalized Lyapunov equation (13) is given by

$$X_\infty = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} = \left(\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix}\right)\left(\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix}\right)^T = Z_\infty Z_\infty^T.$$

A reduced SVD of Z_∞ is given by

$$Z_\infty = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix} = \left(\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix}\right)[1][1]^T = Q_\infty S_\infty V_\infty^T.$$

System (17) reads now as

$$\begin{aligned} 0 \cdot \dot{z}(t) = Q_\infty^T M^T Q_\infty \dot{z}(t) = Q_\infty^T A Q_\infty z(t) = 0 \cdot z(t), \\ z(0) = 1, \end{aligned}$$

which is no ordinary differential equation and has multiple solutions. In contrast system (18) is given by

$$\begin{aligned} \dot{z}(t) &= Q_\infty^T M^{-T} A^T Q_\infty z(t) = -1 \cdot z(t), \\ z(0) &= 1. \end{aligned}$$

The unique solution of the generalized differential Lyapunov equation (12) is given by

$$\begin{aligned} X(t) &= \left(\frac{1}{2} - \frac{e^{-2t}}{2}\right)\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} - \frac{e^{-2t}}{2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \\ &= Z_\infty Z_\infty^T - (Q_\infty z(t))\,(Q_\infty z(t))^T. \end{aligned}$$

B Numerical results for the projection approach

B.1 Results for the differential Lyapunov equation

See Figs. 4, 5 and 6.

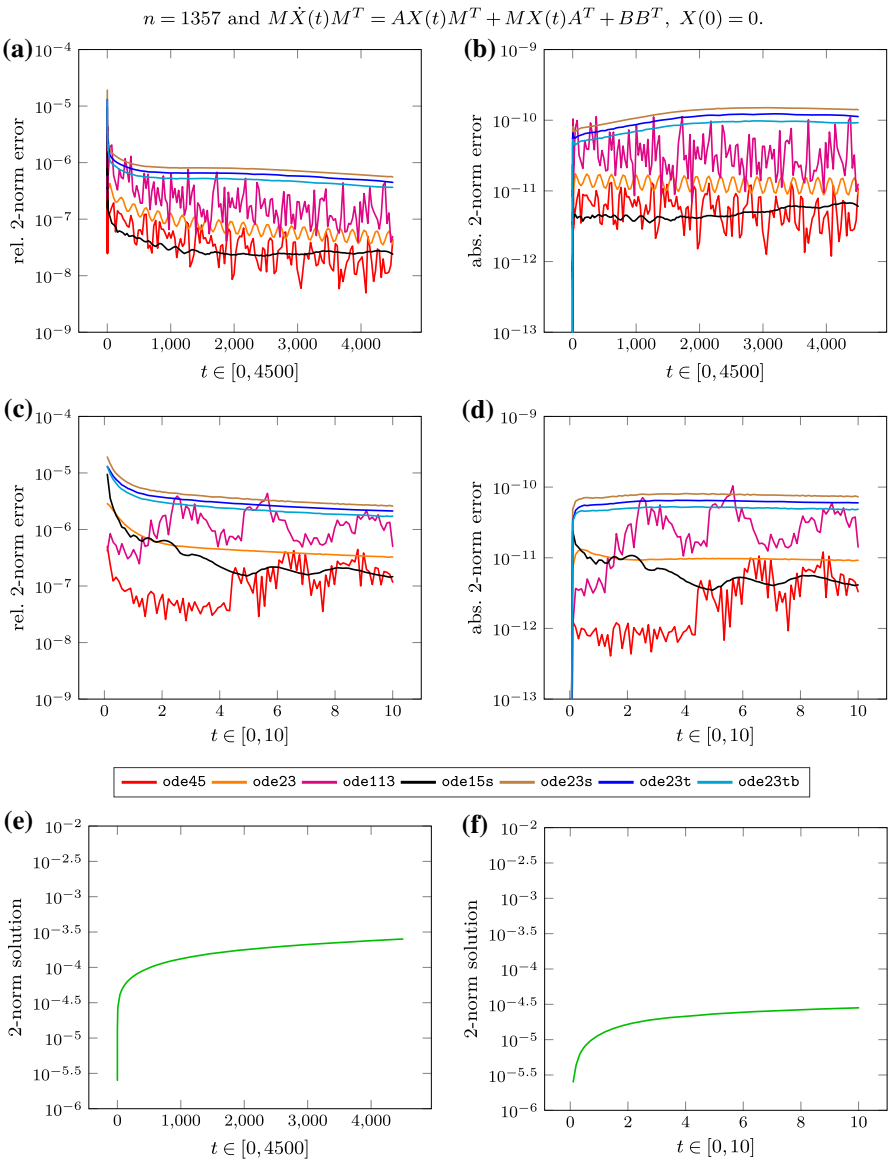

Fig. 4 **a**, **c** Relative 2-norm error of the approximation. **b**, **d** Absolute 2-norm error of the approximation. **e**, **f** 2-norm of the reference solution

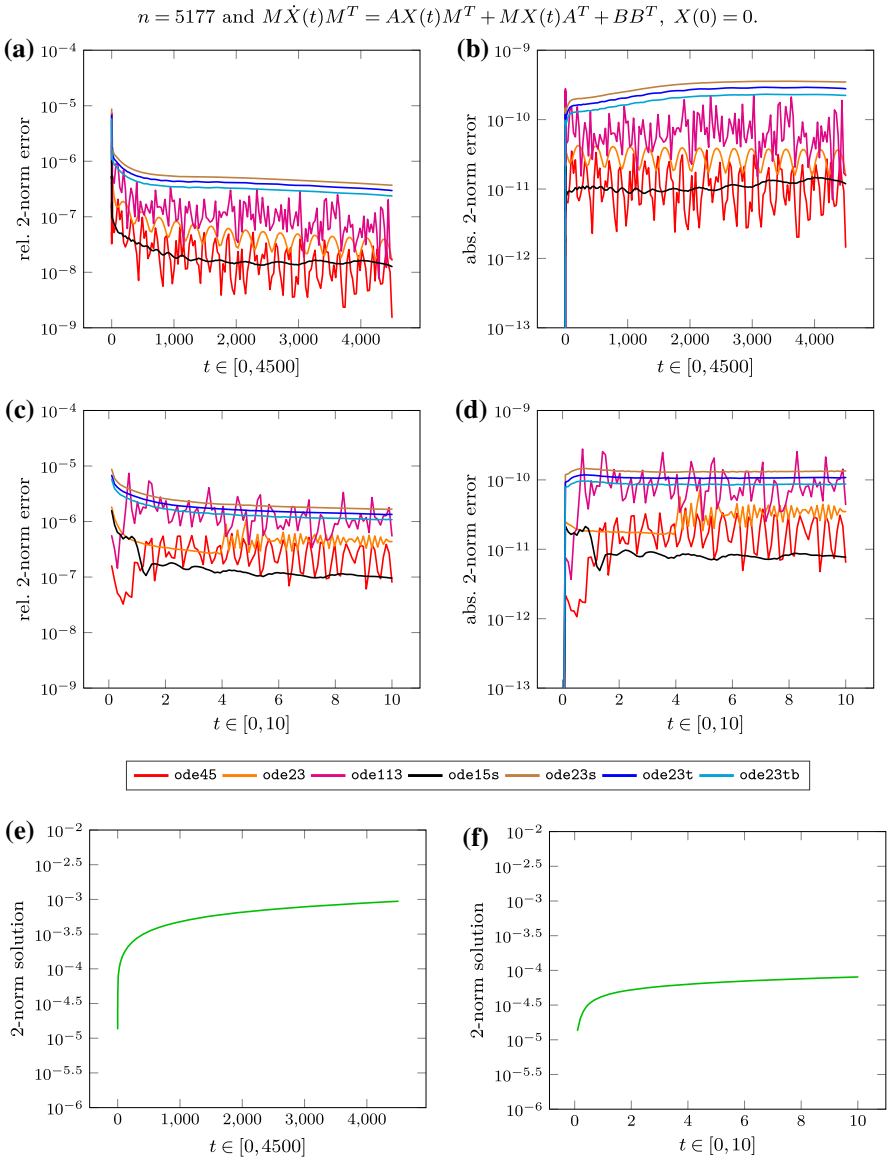

Fig. 5 **a**, **c** Relative 2-norm error of the approximation. **b**, **d** Absolute 2-norm error of the approximation. **e**, **f** 2-norm of the reference solution

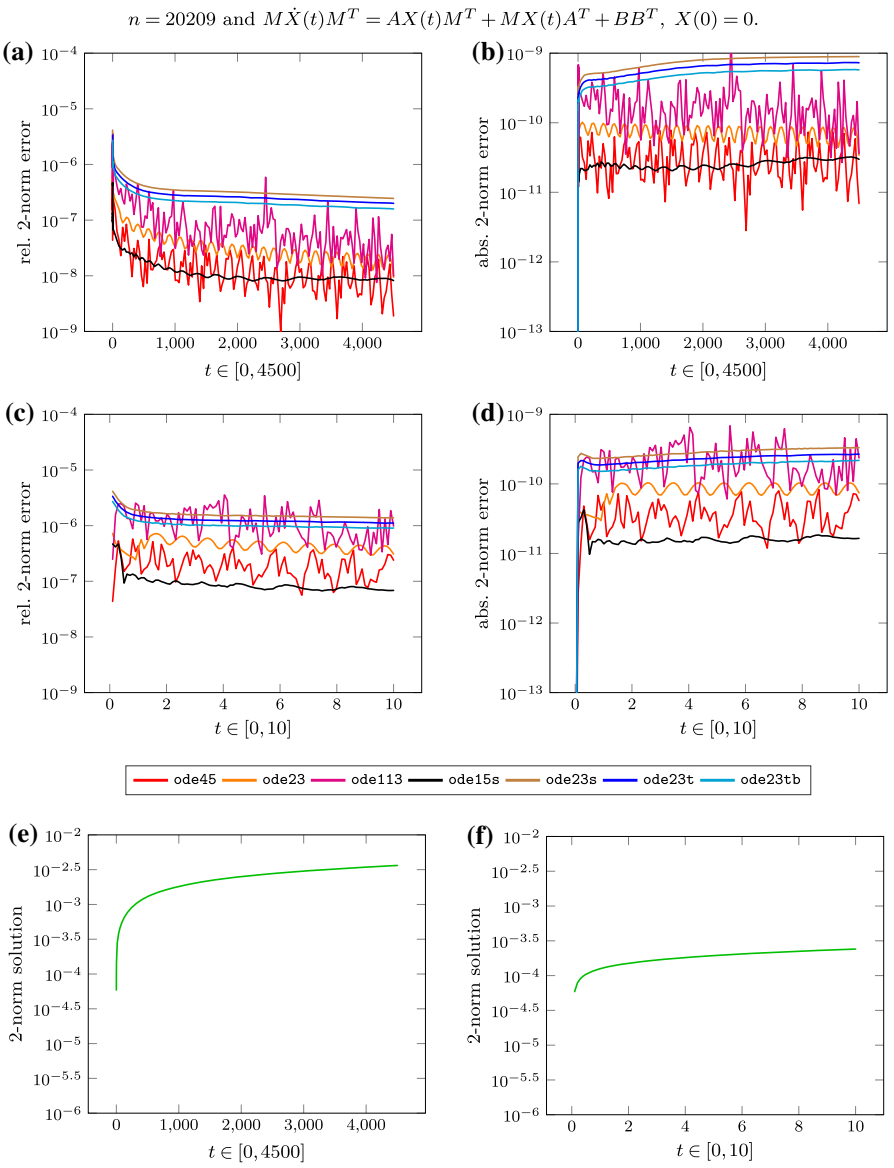

Fig. 6 **a**, **c** Relative 2-norm error of the approximation. **b**, **d** Absolute 2-norm error of the approximation. **e**, **f** 2-norm of the reference solution

B.2 Results for the transposed differential Lyapunov equation

See Figs. 7, 8 and 9.

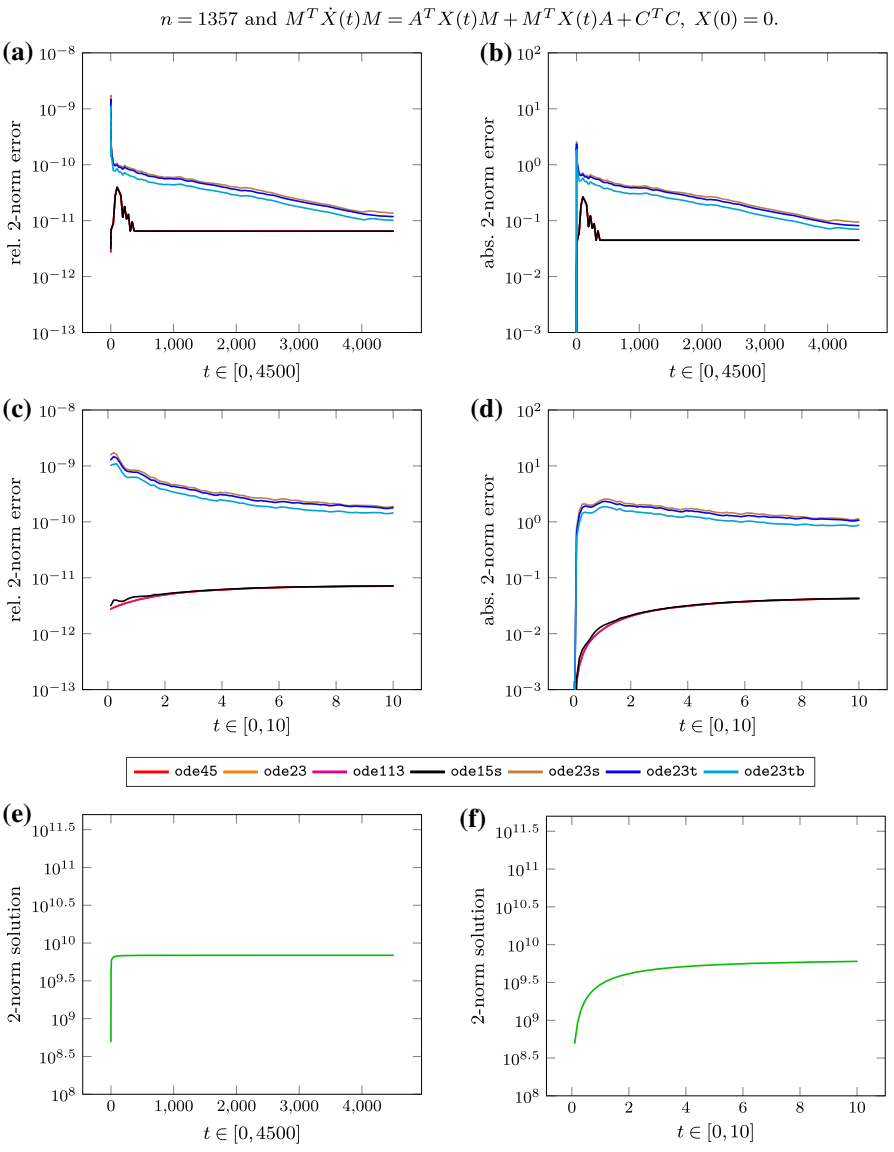

Fig. 7 **a**, **c** Relative 2-norm error of the approximation. **b**, **d** Absolute 2-norm error of the approximation. **e**, **f** 2-norm of the reference solution

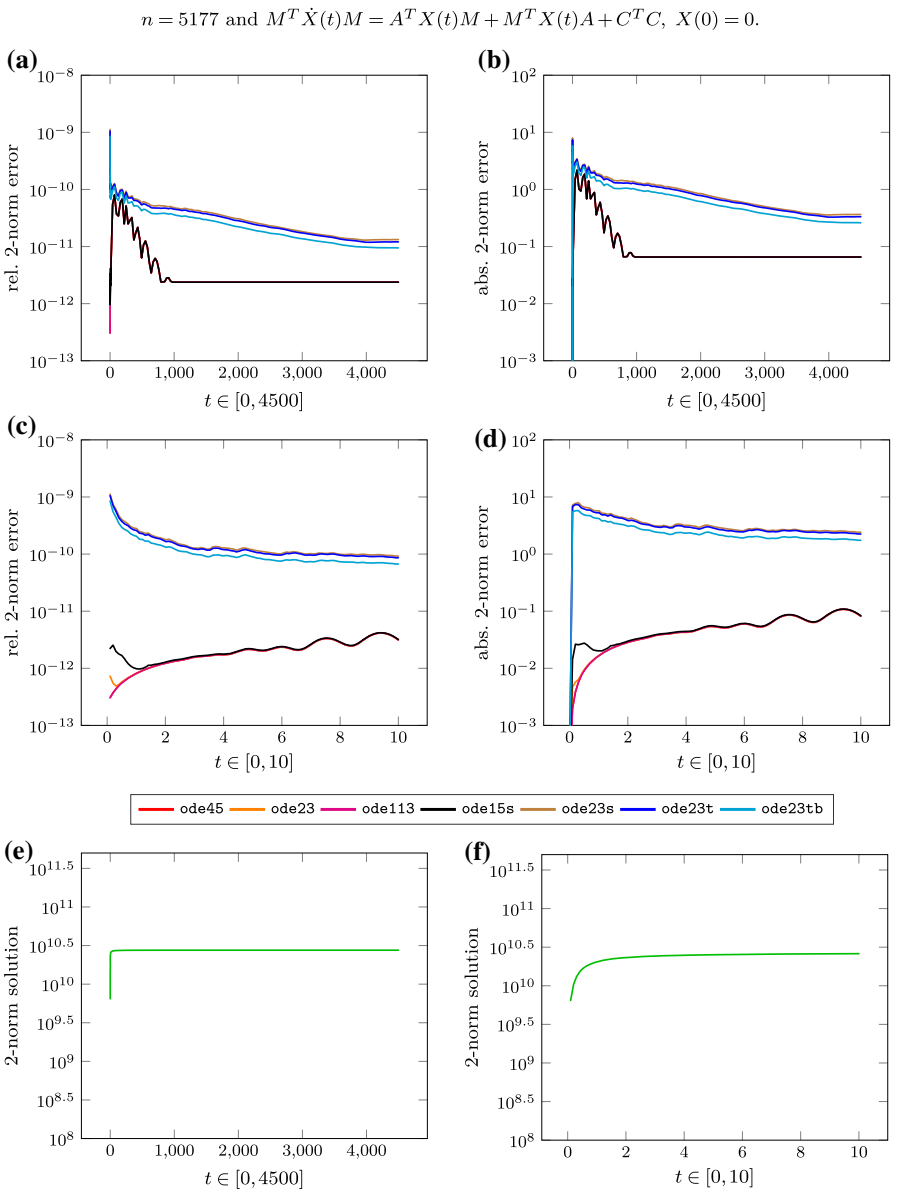

Fig. 8 **a**, **c** Relative 2-norm error of the approximation. **b**, **d** Absolute 2-norm error of the approximation. **e**, **f** 2-norm of the reference solution

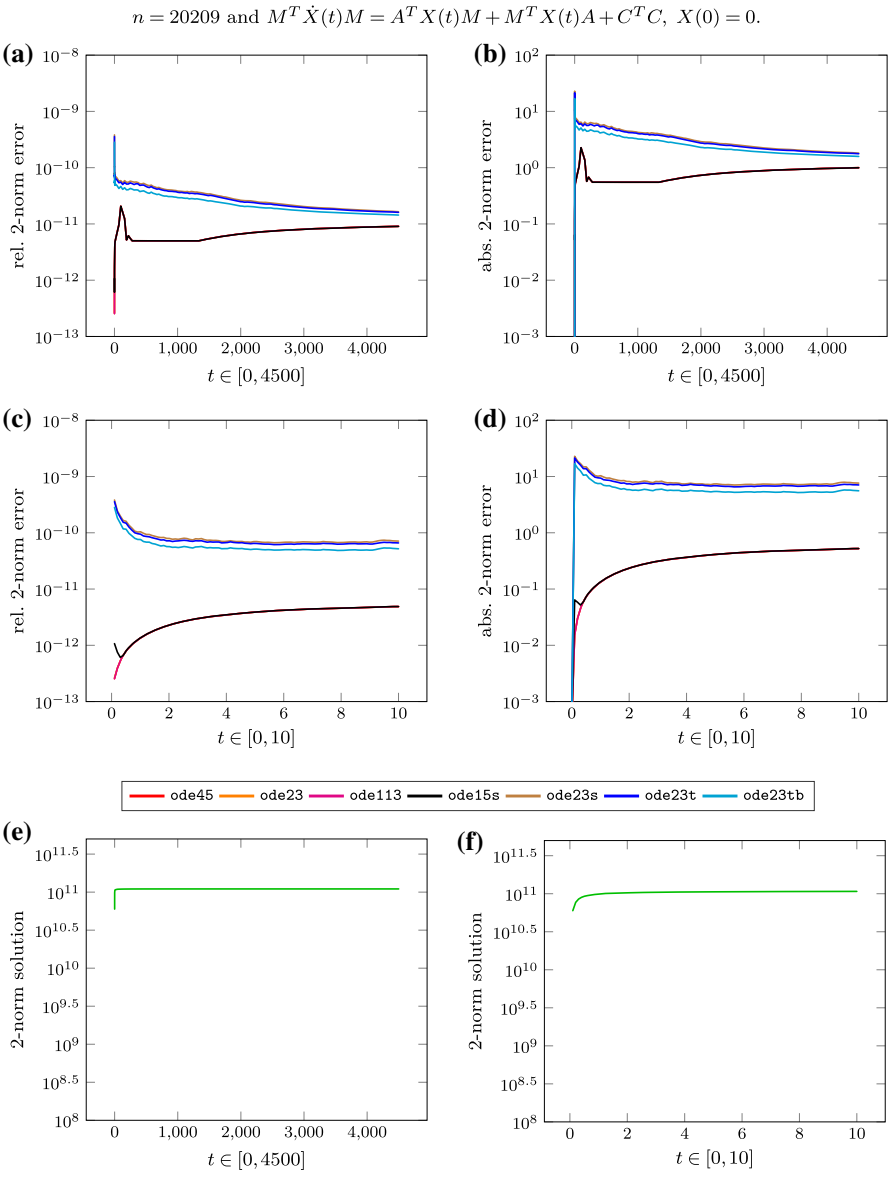

Fig. 9 **a**, **c** Relative 2-norm error of the approximation. **b**, **d** Absolute 2-norm error of the approximation. **e**, **f** 2-norm of the reference solution

C Backward differentiation formulas

We consider Backward Differentiation Formulas (BDF) for differential Lyapunov equations [27,28]. Let $0 = t_0 < t_1 < \cdots < t_N = T$ be a decomposition of the interval $[0, T]$. We define the step size $\tau_k = t_k - t_{k-1}$ for $k = 1, \ldots, N$.

The s-step BDF method applied to the DLE 9 is given by

$$\sum_{j=0}^{s} \alpha_j X_{k-j} = \tau_k \beta \left(A^T X_k + X_k A + B B^T \right),$$

where α_j and β are coefficients of the BDF method and can be found in [15]. The parameter s is the order of the BDF method. We recall that for $s > 6$, the method is not numerical stable, and for $s = 1$, the BDF method coincides with the implicit Euler method. A minor rearrangement shows that the current iterate X_k can be obtained as the solution of the algebraic Lyapunov equation

$$\left(\tau_k \beta A - \frac{\alpha_0}{2} E_{n,n} \right)^T X_k + X_k \left(\tau_k \beta A - \frac{\alpha_0}{2} E_{n,n} \right) = -\tau_k \beta B B^T + \sum_{j=1}^{s} \alpha_j X_{k-j}. \tag{19}$$

Since for $s \geq 2$, certain coefficients α_j, $j \geq 1$ are positive, the algebraic Lyapunov equation (19) has a symmetric but possibly indefinite right-hand side, which makes the standard ADI method infeasible. For this reason a LDL^T-decomposition for the numerical solution is proposed and suitable modifications of the ADI method have been developed; [5,26]: Assume that $X_i \approx L_i D_i L_i^T$ for $i = 0, \ldots, k-1$, $L_i \in \mathbb{R}^{n \times l_i}$, $D_i \in \mathbb{R}^{l_i \times l_i}$ and $l_i \ll n$, then the right hand side can be factored as

$$\begin{aligned} -\tau_k \beta B B^T + \sum_{j=1}^{s} \alpha_j X_{k-j} &\approx -G_k S_k G_k^T, \\ G_k &= \left[B, L_{k-1}, \ldots, L_{k-s} \right], \\ S_k &= \begin{bmatrix} \tau_k \beta E_{p,p} & & & \\ & -\alpha_1 D_{k-1} & & \\ & & \ddots & \\ & & & -\alpha_s D_{k-s} \end{bmatrix}. \end{aligned}$$

Now the LDL^T-type ADI method can be used to determine $X_k \approx L_k D_k L_k^T$. The BDF/ADI methods can also be extended to generalized differential Lyapunov equations in a similar way [28].

C.1 Results for the differential Lyapunov equation

See Fig. 10.

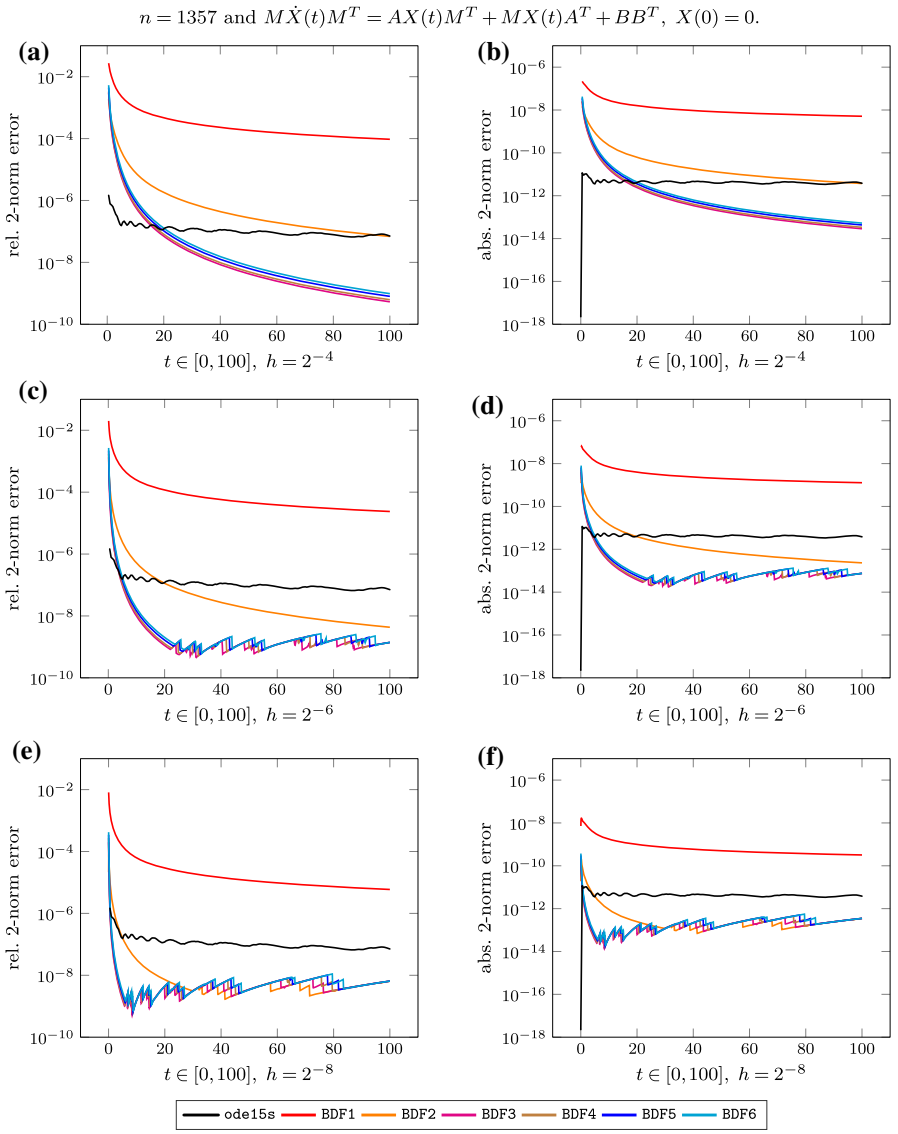

Fig. 10 **a**, **c**, **e** Relative 2-norm error of the BDF/ADI approximation. **b**, **d**, **f** Absolute 2-norm error of the BDF/ADI approximation

C.2 Results for the transposed differential Lyapunov equation

See Fig. 11.

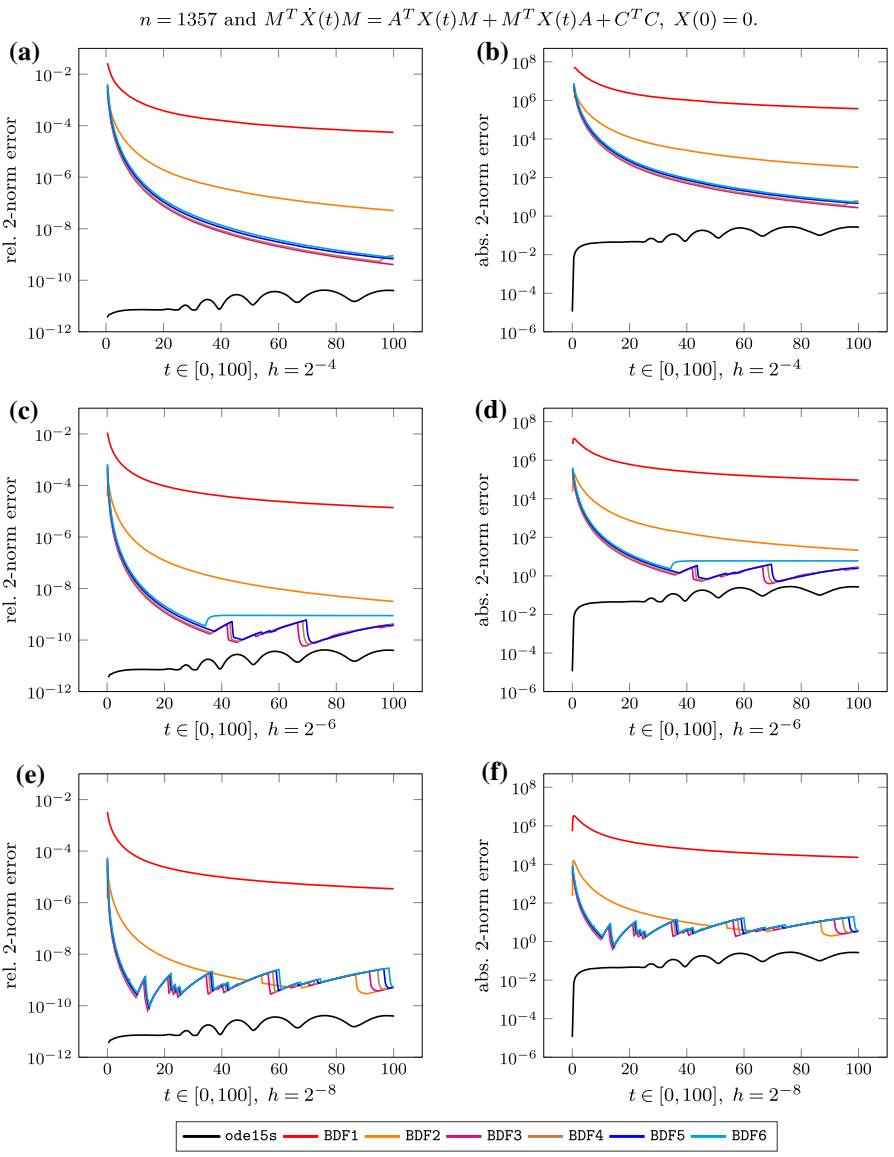

Fig. 11 **a**, **c**, **e** Relative 2-norm error of the BDF/ADI approximation. **b**, **d**, **f** Absolute 2-norm error of the BDF/ADI approximation

References

1. Abou-Kandil, H., Freiling, G., Ionescu, V., Jank, G.: Matrix Riccati Equations in Control and Systems Theory. Birkhäuser, Basel (2003)
2. Amato, F., Ambrosino, R., Ariola, M., Cosentino, C., De Tommasi, G.: Finite-Time Stability and Control. Lecture Notes in Control and Information Sciences, vol. 453. Springer, London (2014)

3. Antoulas, A.C.: Approximation of Large-Scale Dynamical Systems. Advances in Design Control, vol. 6. SIAM Publications, Philadelphia, PA (2005)
4. Benner, P., Köhler, M., Saak, J.: M.E.S.S.—the matrix equations sparse solvers library. https://www.mpi-magdeburg.mpg.de/projects/mess. Accessed 14 Nov 2019
5. Benner, P., Li, R.C., Truhar, N.: On the ADI method for Sylvester equations. J. Comput. Appl. Math. **233**(4), 1035–1045 (2009)
6. Benner, P., Mena, H.: BDF methods for large-scale differential Riccati equations. In: De Moor, B., Motmans, B., Willems, J., Van Dooren, P., Blondel, V. (eds.) Proceedings of the 16th International Symposium on Mathematical Theory of Network and Systems, MTNS 2004 (2004)
7. Benner, P., Mena, H.: Rosenbrock methods for solving Riccati differential equations. IEEE Trans. Autom. Control **58**(11), 2950–2957 (2013)
8. Benner, P., Saak, J.: A semi-discretized heat transfer model for optimal cooling of steel profiles. In: Benner, P., Mehrmann, V., Sorensen, D. (eds.) Dimension Reduction of Large-Scale Systems. Lecture Notes of Computer Science and Engineering, vol. 45, pp. 353–356. Springer, Berlin (2005)
9. Brocket, R.A.: Finite Dimensional Linear Systems. Wiley, New York (1970)
10. Byers, R., Nash, S.: On the singular "vectors" of the Lyapunov operator. SIAM J. Algebraic Discrete Methods **8**(1), 59–66 (1987)
11. Gajić, Z., Qureshi, M.: Lyapunov Matrix Equation in System Stability and Control. Mathematics in Science and Engineering. Academic Press, San Diego, CA (1995)
12. Güldoğan, Y., Hached, M., Jbilou, K., Kurulay, M.: Low rank approximate solutions to large-scale differential matrix Riccati equations. Technical report. arXiv:1612.00499v2, arXiv (2017). Math.NA
13. Hached, M., Jbilou, K.: Numerical solutions to large-scale differential Lyapunov matrix equations. Numer. Algorithms **79**, 741–757 (2017)
14. Hached, M., Jbilou, K.: Approximate solutions to large nonsymmetric differential Riccati problems. Technical report. arXiv:1801.01291v1, arXiv (2018). Math.NA
15. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations. I. Springer Series in Computational Mathematics. Springer, Berlin (1987)
16. Heiland, J.: Decoupling and optimization of differential-algebraic equations with application in flow control. Dissertation, TU Berlin (2014)
17. Higham, N.J.: Functions of Matrices: Theory and Computation. Applied Mathematics. SIAM Publications, Philadelphia, PA (2008)
18. Jameson, A.: Solution of the equation $AX + XB = C$ by inversion of an $M \times M$ or $N \times N$ matrix. SIAM J. Appl. Math. **16**, 1020–1023 (1968)
19. Knobloch, H.W., Kwakernaak, H.: Lineare Kontrolltheorie. Springer, Berlin (1985). In German
20. Kohaupt, L.: Solution of the matrix eigenvalue problem $VA + A^*V = \mu V$ with applications to the study of free linear dynamical systems. J. Comput. Appl. Math. **213**(1), 142–165 (2008)
21. Köhler, M., Lang, N., Saak, J.: Solving differential matrix equations using parareal. Proc. Appl. Math. Mech. **16**(1), 847–848 (2016)
22. Konstantinov, M., Mehrmann, V., Petkov, P.: On properties of Sylvester and Lyapunov operators. Linear Algebra Appl. **312**(1–3), 35–71 (2000)
23. Konstantinov, M.M., Gu, D.W., Mehrmann, V., Petkov, P.H.: Perturbation Theory for Matrix Equations. Studies in Computational Mathematics, vol. 9, 1st edn. Elsevier, Amsterdam (2003)
24. Koskela, A., Mena, H.: A structure preserving Krylov subspace method for large scale differential Riccati equations. e-print arXiv:1705.07507, arXiv (2017). Math.NA
25. Lang, N.: Numerical methods for large-scale linear time-varying control systems and related differential matrix equations. Dissertation, Technische Universität Chemnitz, Germany (2017)
26. Lang, N., Mena, H., Saak, J.: On the benefits of the LDL^T factorization for large-scale differential matrix equation solvers. Linear Algebra Appl. **480**, 44–71 (2015)
27. Lang, N., Saak, J., Stykel, T.: Towards practical implementations of balanced truncation for LTV systems. IFAC-PapersOnLine **48**(1), 7–8 (2015)
28. Lang, N., Saak, J., Stykel, T.: Balanced truncation model reduction for linear time-varying systems. Math. Comput. Model. Dyn. Syst. **22**(4), 267–281 (2016)
29. Lang, N., Saak, J., Stykel, T.: LTV-BT for MATLAB (2017)
30. Locatelli, A.: Optimal Control: An Introduction. Birkhäuser, Basel (2001)
31. Mena, H.: Numerical solution of differential Riccati equations arising in optimal control of partial differential equations. Dissertation, Escuela Politécnica Nacional, Ecuador (2007)

32. Palmer, T.W.: Banach Algebras and the General Theory of $*$-Algebras. Vol. 2, Encyclopedia of Mathematics and Its Applications, vol. 79. Cambridge University Press, Cambridge (2001)
33. Polderman, J.W., Willems, J.C.: Introduction to Mathematical Systems Theory. Texts in Applied Mathematics, vol. 26. Springer, New York (1998)
34. Rome, H.: A direct solution to the linear variance equation of a time-invariant linear system. IEEE Trans. Autom. Control **14**(5), 592–593 (1969)
35. Stewart, G.W.: Matrix Algorithms. Eigensystems, vol. II. SIAM Publications, Philadelphia, PA (2001)
36. Stillfjord, T.: Low-rank second-order splitting of large-scale differential Riccati equations. IEEE Trans. Autom. Control **60**(10), 2791–2796 (2015)
37. Werner, D.: Funktionalanalysis, extended edn. Springer, Berlin (2000)