Projection methods for large-scale T-Sylvester equations

Froilán M. Dopico* Javier González† Daniel Kressner‡ Valeria Simoncini§

May 29, 2015

Abstract

The matrix Sylvester equation for congruence, or T-Sylvester equation, has recently attracted considerable attention as a consequence of its close relation to palindromic eigenvalue problems. The theory concerning T-Sylvester equations is rather well understood and there are stable and efficient numerical algorithms which solve these equations for small- to medium-sized matrices. However, developing numerical algorithms for solving large-scale T-Sylvester equations still remains an open problem. In this paper, we present several projection algorithms based on different Krylov spaces for solving this problem when the right-hand side of the T-Sylvester equation is a low-rank matrix. The new algorithms have been extensively tested, and the reported numerical results show that they work very well in practice, offering a clear guidance on which algorithm is the most convenient in each situation.

Key words. matrix equations, Krylov subspace, iterative methods, large-scale equations, Sylvester equation, Sylvester equation for congruence

AMS subject classification. 65F10, 65F30, 15A06

1 Introduction

This paper is concerned with the numerical solution of the real square *T-Sylvester equation*

$$AX + X^T B = C, \tag{1.1}$$

where $A, B, C \in \mathbb{R}^{n\times n}$ are given and $X \in \mathbb{R}^{n\times n}$ is the unknown. The study of theoretical properties for this equation goes back to at least 1962, when Taussky and Wielandt [38] analyzed the linear map $X \mapsto AX + X^T B$ for the special case $B = A^T$. Conditions for the existence of solutions in the general case were established by Wimmer in the early '90s [42]. Recently, there has been renewed interest in studying (1.1) for the general case, see, e.g., [11, 13, 14, 15, 18]. To some extent, this has been sparked by the close relation of (1.1) to palindromic eigenvalue problems of the form $G + \lambda G^T$. For example, the solution of (1.1) is needed to determine the first-order perturbation expansion for a deflating subspace of $G + \lambda G^T$ [8]. In turn, a Newton method for computing such a deflating subspace would require the repeated solution of possibly large-scale T-Sylvester equations, similar to the methods presented in [9, 16] for standard eigenvalue problems. Equations

*Departamento de Matemáticas, Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés, Spain (dopico@math.uc3m.es). Supported by Ministerio de Economía y Competitividad of Spain through grant MTM2012-32542.

†Departamento de Matemáticas, Universidad Carlos III de Madrid, Avda. Universidad 30, 28911 Leganés, Spain (jagpizar@math.uc3m.es). Supported by Ministerio de Economía y Competitividad of Spain through grant MTM2012-32542.

‡ANCHP, MATHICSE, EPF Lausanne, Station 8, 1015 Lausanne, Switzerland (daniel.kressner@epfl.ch).

§Dipartimento di Matematica, Università di Bologna, Piazza di Porta San Donato 5, I-40127 Bologna, Italy (valeria.simoncini@unibo.it).

of the form (1.1) also arise as auxiliary problems in a structure-preserving QR algorithm [29] for solving palindromic eigenvalue problems. Applications that involve (1.1) with $B = \pm A^T$ arise from Hamiltonian systems [7], time-varying singular value decompositions [3], and quadratic inverse eigenvalue problems [43].

Finding solutions of (1.1) becomes rather simple for the special case $B = \pm A^T$, $C = \pm C^T$ [7]. For example, if A is invertible then $X = \frac{1}{2}A^{-1}C$ is trivially a solution. In the general case, however, solutions of (1.1) do not admit such a simple expression. For small- to medium-sized matrices, extensions of the Bartels-Stewart algorithm [2] have been proposed in [13, 11, 39]. A whole class of iterative methods can be derived by viewing (1.1) as an $n^2 \times n^2$ linear system in the entries of X and applying an existing iterative solver for linear systems, see [41] for an example. Still, the need for storing all entries of the approximate solution limits these methods to $n \lesssim 10^4$.

In this work, we develop novel projection methods that iteratively construct low-rank approximations to the solution of T-Sylvester equations with a low-rank right-hand side matrix C, a situation that arises in some applications discussed in detail in Section 2. Based on Krylov subspaces, our methods only require matrix-vector multiplications and the solution of linear systems with A, B, which makes them applicable to equations with large and sparse coefficient matrices. Similar projection methods are routinely used for approximating the solution of large-scale Sylvester and Lyapunov equations, see [35] for a recent survey. As we will see in the course of this paper, the extension of these existing projection methods to (1.1) is by no means straightforward.

Throughout this paper, we restrict our attention to T-Sylvester equations with real coefficient matrices. However, our results and numerical methods can be easily adapted to complex coefficients, for which the transpose in (1.1) is replaced either by the complex transpose or by the conjugate transpose, see [13] and the references therein.

Notation. We follow standard notation found in, e.g., [19]. Given a matrix G, the subspace spanned by the columns of G is denoted by range(G). The block Krylov subspace of order m associated with the matrices $A \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{n\times r}$ is denoted by

$$\mathcal{K}_m(A, V) = \text{range}([V, AV, A^2V, \ldots, A^{m-1}V]). \tag{1.2}$$

For any subspace $\mathbb{V} \subseteq \mathbb{R}^n$, we set $A\,\mathbb{V} := \{Ax : x \in \mathbb{V}\} \subseteq \mathbb{R}^n$. The field of values of a matrix A is given by $\mathcal{F}(A) := \{x^*Ax : x \in \mathbb{C}^n, x^*x = 1\}$, where x^* is the conjugate-transpose vector. The spectral and Frobenius norm of matrices are denoted by $\|\cdot\|_2$ and $\|\cdot\|_F$, respectively.

2 Preliminaries, relation to Sylvester equations, and applications

In this section, we recall existing results on the real square T-Sylvester equation (1.1), show a new relation to standard Sylvester equations, and discuss some applications where the right-hand side C of (1.1) has low-rank.

To state necessary and sufficient conditions for the existence and uniqueness of a solution X of (1.1), we need to introduce the following notion.

Definition 2.1. *A set* $\{\lambda_1, \lambda_2, \ldots, \lambda_n\} \subset \mathbb{C} \cup \{\infty\}$ *is called* reciprocal free *if* $\lambda_i \neq 1/\lambda_j$ *for* $1 \leq i, j \leq n$.

By convention, $1/0 = \infty$ and $1/\infty = 0$ in Definition 2.1. We let spec(A, B^T) denote the set of eigenvalues of the matrix pencil $A - \lambda B^T$. If B is singular then $A - \lambda B^T$ has infinite eigenvalues, that is, $\infty \in \text{spec}(A, B^T)$. The matrix pencil is called *regular* if $\det(A - \lambda B^T)$ does not vanish for all λ.

Theorem 2.2 ([8, Lemma 5.10])**.** *The T-Sylvester equation $AX + X^TB = C$ with $A, B \in \mathbb{R}^{n\times n}$ has a unique solution for every right-hand side $C \in \mathbb{R}^{n\times n}$ if and only if the following two conditions hold:*

(a) *The matrix pencil* $A - \lambda B^T$ *is regular; and*

(b) $\operatorname{spec}(A, B^T)\backslash\{1\}$ *is reciprocal free and if* $1 \in \operatorname{spec}(A, B^T)$*, then it has algebraic multiplicity* 1.

Some references erroneously replace condition (b) in Theorem 2.2 simply by "$\operatorname{spec}(A, B^T)$ is reciprocal free", which requires $1 \notin \operatorname{spec}(A, B^T)$ (see, e.g., [25, Theorem 3]). It is easy to construct examples showing that this requirement is not needed. Consider, for instance, $n = 1$ and $A = B = 1$. In this case, the only eigenvalue of $A - \lambda B^T$ is 1, but $AX + X^T B = C$ has the unique solution $X = C/2$.

An extension of the Bartels-Stewart algorithm [2] to T-Sylvester equations admitting a unique solution was presented in [13, Algorithm 3.1], see also [11] and [39]. This algorithm proceeds by first computing a generalized real Schur decomposition of the pencil $A - \lambda B^T$ via the QZ algorithm [19, Section 7.7]. This yields orthogonal matrices $V, W \in \mathbb{R}^{n\times n}$ such that

$$A = W T_A V^T \quad \text{and} \quad B^T = W T_B V^T, \tag{2.1}$$

where $T_A \in \mathbb{R}^{n\times n}$ is upper quasi-triangular (that is, block upper triangular with 1×1 or 2×2 diagonal blocks), and T_B is upper triangular. By defining $Y = V^T X W$, the factorizations (2.1) allow us to transform (1.1) into the T-Sylvester equation

$$T_A Y + Y^T T_B^T = W^T C W.$$

The (block) triangular structure of T_A and T_B allows us to construct the solution Y of this equation by a simple substitution procedure, see [13, Algorithm 3.1] for details. The algorithm is completed by performing the back transformation $X = VYW^T$. Requiring $O(n^3)$ operations and $O(n^2)$ memory, the scope of this algorithm is limited to moderately sized equations.

The following theorem reveals a relationship between (1.1) and a standard Sylvester equation under certain assumptions, and also between (1.1) and a generalized Sylvester equation. To the best of our knowledge, these relations are new.

Theorem 2.3. *Let* $A, B, C \in \mathbb{R}^{n\times n}$ *and assume that* A *and* B *are nonsingular. Consider the matrix equations*

$$AX + X^T B = C, \tag{2.2}$$

$$\left(B^{-T} A\right) X - X \left(A^{-T} B\right) = B^{-T} C - B^{-T} C^T A^{-T} B, \tag{2.3}$$

$$AXA^T - B^T X B = C - C^T A^{-T} B, \tag{2.4}$$

for the unknown $X \in \mathbb{R}^{n\times n}$*. Then the following statements hold.*

(a) *If* X_0 *is a solution of the T-Sylvester equation* (2.2) *then* X_0 *is also a solution of the Sylvester equation* (2.3)*.*

(b) *If the Sylvester equation* (2.3) *has a unique solution* X_0 *then the T-Sylvester equation* (2.2) *has also a unique solution, which is equal to* X_0*.*

(c) X_0 *is a solution of the generalized Sylvester equation* (2.4) *if and only if* $(X_0 A^T)$ *is a solution of the Sylvester equation* (2.3)*, i.e., there is a bijection between the sets of solutions of* (2.4) *and* (2.3)*.*

Proof. (a) If X_0 satisfies (2.2), then

$$AX_0 = C - X_0^T B \Longrightarrow X_0 = A^{-1} C - A^{-1} X_0^T B \Longrightarrow X_0^T = C^T A^{-T} - B^T X_0 A^{-T}.$$

Inserting this expression for X_0^T into $AX_0 + X_0^T B = C$, we obtain

$$AX_0 + C^T A^{-T} B - B^T X_0 A^{-T} B = C \Longrightarrow AX_0 - B^T X_0 A^{-T} B = C - C^T A^{-T} B.$$

Multiplying the latter equation with B^{-T} on the left yields that X_0 satisfies (2.3).
(b) Assume that (2.3) has a unique solution, which is equivalent to assuming that

$$\operatorname{spec}\left(B^{-T}A\right) \cap \operatorname{spec}\left(A^{-T}B\right) = \emptyset, \tag{2.5}$$

see [23, p. 270], where $\operatorname{spec}(Z)$ denotes the spectrum of Z. Note that

$$\operatorname{spec}\left(\left(B^{-T}A\right)^{-1}\right) = \operatorname{spec}\left(\left(A^{-1}B^{T}\right)^{T}\right) = \operatorname{spec}\left(BA^{-T}\right) = \operatorname{spec}\left(A^{-T}B\right).$$

This shows that (2.5) is equivalent to

$$\operatorname{spec}\left(B^{-T}A\right) \cap \operatorname{spec}\left(\left(B^{-T}A\right)^{-1}\right) = \emptyset.$$

In other words, $\operatorname{spec}\left(B^{-T}A\right) = \operatorname{spec}\left(A, B^{T}\right)$ is reciprocal free. Thus, Theorem 2.2 shows that (2.2) has a unique solution.

So far, we have established that the unique solvability of (2.3) implies the unique solvability of (2.2), but not yet that the solutions of both equations are the same. This, however, follows directly from part (a), which states that the solution set of (2.2) is included in the solution set of (2.3). Therefore both sets must be identical when they only have one element.
(c) If we multiply equation (2.3) on the left by B^T, then we get the equivalent equation $A(XA^{-T})A^T - B^T(XA^{-T})B = C - C^TA^{-T}B$ and the result follows. □

Remark 2.4. Observe that elementary results on ranks [22, p. 13] yield

$$\operatorname{rank}\left(B^{-T}C - B^{-T}C^{T}A^{-T}B\right) \leq \operatorname{rank}\left(B^{-T}C\right) + \operatorname{rank}\left(B^{-T}C^{T}A^{-T}B\right) \leq 2\operatorname{rank}(C).$$

So, if the right-hand side of (2.2) has low rank then the right-hand side of (2.3) has also low rank.

Before discussing the use of Theorem 2.3 for computations, let us remark that the converse of Theorem 2.3(b) does not hold. The T-Sylvester equation (2.2) may have a unique solution when the Sylvester equation (2.3) does not. To see this, consider again the case $n = 1$ with $A = B = 1$ and arbitrary $C \in \mathbb{R}$. Then (2.2) has the unique solution $X = C/2$, while (2.3) reads $X - X = 0$ and thus every $X \in \mathbb{R}$ is a solution of (2.3). In addition, note that also every $X \in \mathbb{R}$ is a solution of the generalized Sylvester equation (2.4) in this example. It follows from the proof of Theorem 2.3 that such a situation can only occur when (2.2) has a unique solution and 1 is a simple eigenvalue of $A - \lambda B^T$.

Theorem 2.3(b) suggests that we can compute the solution of (1.1) by simply computing the solution of the standard Sylvester equation (2.3), provided that (2.3) has a unique solution. In view of the availability of reliable numerical algorithms for solving both small- and large-scale Sylvester equations [19, 35], such an approach sounds quite attractive and will be investigated in the large-scale setting in Section 7. However, we advance that the different unique solvability conditions for (2.2) and (2.3) are not merely a theoretical issue and cause numerical difficulties, as we have observed in the numerical tests presented in Section 7. The generalized Sylvester equation (2.4) can also be used, in the case it has a unique solution, for computing the solution of (1.1): simply compute the solution of (2.4) and multiply it by A^T on the right. However, in the case of generalized Sylvester equations the available numerical algorithms and software are somewhat less developed compared to standard Sylvester equations [35].

In a situation where the standard Sylvester equation (2.3) is uniquely solvable but $A - \lambda B^T$ has an eigenvalue close to one, we can expect (2.3) to be much more sensitive to approximation and roundoff errors than the original T-Sylvester equation (2.2). In particular, this may lead to approximate solutions $\tilde{X}$ that yield a tiny residual for (2.3) but not for (2.2). To illustrate this

seemingly subtle but practically important point, it suffices to consider a *scalar* example; examples of larger sizes can be easily constructed. Let $A = 1 + \delta$, with a small parameter $\delta \neq 0$, and $B = 1$. Then the T-Sylvester equation (2.2) has the unique solution $X = C/(2 + \delta)$. For this example, it is straightforward to see that tiny relative perturbations of A and B always lead to tiny relative variations of the solution for (2.2). On the other hand, the Sylvester equation (2.3) for $A = 1 + \delta$ and $B = 1$ is given by

$$(1 + \delta)\, X - \frac{X}{1 + \delta} = C\frac{\delta}{1 + \delta}. \tag{2.6}$$

Its solution is unique and, of course, equal to $X = C/(2 + \delta)$. In contrast to the T-Sylvester equation, tiny relative perturbations of the coefficients of (2.6) may change the solution drastically. Suppose that the first coefficient $(1 + \delta)$ in (2.6) is replaced by 1, yielding the perturbed solution $\tilde{X} = C$. Clearly, this is quite different from X, although the residual of $\tilde{X}$ for (2.6) is small:

$$R_S = (1 + \delta)\, \tilde{X} - \frac{\tilde{X}}{1 + \delta} - C\frac{\delta}{1 + \delta} = \delta C.$$

In contrast, the residual of $\tilde{X}$ for $AX + X^T B = C$ is large:

$$R_{TS} = \tilde{X}(2 + \delta) - C = (1 + \delta)C\,.$$

We invite the reader to check that the sensitivity to tiny relative perturbations in A and B of the solution of the generalized Sylvester equation (2.4) is, in this particular example, exactly the same as the sensitivity of the T-Sylvester equation (2.2), as long as the solution of (2.4) remains unique. From this perspective, equation (2.4) seems to be superior to (2.3), as it is suggested also by the absence of inverses in the left hand side of (2.4). Nonetheless, note that if the coefficients $A = 1+\delta$ and $B = 1$ suffer the tiny change $\hat{A} = 1$ and $\hat{B} = 1$, then the solution of (2.2) simply change to $C/2$, while (2.4) has now *any* number as solution.

A rigorous perturbation analysis of the equations (2.2), (2.3), and (2.4) is out of the scope of this paper and the only purpose of the discussion above is to warn the reader that in transforming an equation into another the sensitivity of the solution may be an issue for numerical computations.

Since the goal of this paper is to solve the T-Sylvester equation (1.1) when the rank of C is low, we finish this section by discussing some applications where this situation arises. It is well-known that standard Sylvester equations with low-rank right-hand sides appear very often in linear system theory and control theory, more precisely in the context of model order reduction and as intermediate steps in the solution of continuous-time algebraic Riccati equations (CARE) by iterative methods in linear-quadratic optimal control problems [1, 5]. Analogously, certain non-standard problems in optimal Hankel-norm model reduction and H_2/H_∞ controls related to non-standard J-spectral factorization problems lead to T-CARE whose nonlinear and constant terms have both low rank, under the natural assumption that the numbers of inputs and outputs are much smaller than the number of internal states of the system (see [27, 28] and the references therein). For instance, the T-CARE appearing in [27, 28] is

$$A^T X + X^T A + X^T R X + Q = 0, \tag{2.7}$$

$$E^T X - X^T E = 0, \tag{2.8}$$

where $E, A, Q, R \in \mathbb{R}^{n\times n}$, E may be singular, $Q = Q^T$, $R = R^T$, Q and R are indefinite, and the ranks of Q and R are much smaller than n under the conditions mentioned above. The problem of interest in applications is to compute a *stabilizing solution* X of (2.7)-(2.8), which roughly speaking is a solution such that the corresponding close-loop matrix has its eigenvalues in the left-hand plane including the extended imaginary axis. Conditions for the existence of such solutions have been established in [28] and their numerical computation is a non-trivial problem considered in [27] and solved satisfactorily in [12] for small- to medium- size matrices. However, the solution of (2.7)-(2.8)

for large-scale matrices remains an open problem. The efficient solution of this problem will require iterative methods that will need efficient solvers of large-scale T-Sylvester equations with low-rank right-hand sides, similarly to the solution of large-scale standard CAREs which requires solvers of large-scale Sylvester equations with low-rank right-hand sides [5]. To realize this point, note that the two equations (2.7)-(2.8) are equivalent to

$$(A^T + E^T)X + X^T(A - E) + X^T RX + Q = 0, \tag{2.9}$$

because the sum of (2.7) and (2.8) yields (2.9), while (2.9) plus and minus its transpose yield (2.7) and (2.8), respectively [11, Section 2.3]. Clearly, any fixed point iteration or any Newton-based method for solving (2.9) would need the solution of T-Sylvester equations with low-rank right-hand sides. Variants of equations (2.7)-(2.8) where Q and R are low-rank positive semidefinite matrices have also appeared in applications [40] and they can be connected to T-Sylvester equations exactly in the same way.

3 A general projection framework for the T-Sylvester equation

As explained in the introduction, the goal of this paper is to solve T-Sylvester equations with large and sparse coefficient matrices A, B, and low-rank right-hand side C. So, from now on, we write the T-Sylvester equation in the form

$$AX + X^T B = C_1 C_2^T, \tag{3.1}$$

where $C_1, C_2 \in \mathbb{R}^{n\times r}$ are full-rank matrices with $r \ll n$. As our methods involve the inversion of A and/or B, we additionally assume the generic condition that these matrices are invertible. Moreover, (3.1) is assumed to admit a unique solution, that is, the conditions of Theorem 2.2 hold. Although A and B may be sparse, the solution matrix X is full, in general, so the storage of X requires excessive memory allocations for large problems. We thus look for low-rank approximations to X. This strategy is strongly supported by the link established in Theorem 2.3(b) between a T-Sylvester equation and a standard Sylvester equation both with low-rank right-hand sides (see Remark 2.4), together with existing results on the singular value decay of solutions to Sylvester equations with low-rank right-hand side [20, 31], which suggest that the solution X of (3.1) can often be well approximated by a low-rank matrix.

To construct low-rank approximations to the solution, existing approaches for large-scale Sylvester equations with low-rank right-hand side often proceed by imposing a Galerkin condition on a tensor product of low-dimensional subspaces, see [6, 26, 32] and [35, Section 4.4.1] for examples. We follow a similar strategy for T-Sylvester equations. Our starting point is therefore to consider approximations of the form

$$X \approx X_m = \mathcal{V}_m Y_m \mathcal{W}_m^T \in \mathbb{R}^{n\times n}, \tag{3.2}$$

where the columns of $\mathcal{V}_m, \mathcal{W}_m \in \mathbb{R}^{n\times p_m}$ form orthonormal bases of subspaces $\mathbb{V}_m, \mathbb{W}_m \subset \mathbb{R}^n$, respectively:

$$\mathbb{V}_m = \operatorname{range}(\mathcal{V}_m) \quad \text{and} \quad \mathbb{W}_m = \operatorname{range}(\mathcal{W}_m). \tag{3.3}$$

In terms of these subspaces, the relation (3.2) states that $X_m \in \mathbb{V}_m \otimes \mathbb{W}_m$, where the tensor product of these subspaces is defined as

$$\mathbb{V}_m \otimes \mathbb{W}_m := \operatorname{span}\{v \otimes w : v \in \mathbb{V}_m, w \in \mathbb{W}_m\}.$$

When identifying $\mathbb{R}^{n\times n} \cong \mathbb{R}^{n^2}$, which will be assumed in the following, the set $\mathbb{V}_m \otimes \mathbb{W}_m$ is the set of all matrices of the form (3.2). The matrix $Y_m \in \mathbb{R}^{p_m\times p_m}$, which contains the coefficients of

X_m in the chosen bases, is determined by imposing a Petrov-Galerkin condition: We require the residual

$$R_m = AX_m + X_m^T B - C_1 C_2^T = A(\mathcal{V}_m Y_m \mathcal{W}_m^T) + (\mathcal{V}_m Y_m \mathcal{W}_m^T)^T B - C_1 C_2^T \tag{3.4}$$

to be orthogonal to $\mathbb{W}_m \otimes \mathbb{W}_m$ in the matrix inner product, $R_m \perp \mathbb{W}_m \otimes \mathbb{W}_m$. Using the orthonormal basis $\mathcal{W}_m$, this becomes equivalent to requiring

$$\mathcal{W}_m^T R_m \mathcal{W}_m = 0. \tag{3.5}$$

Inserting (3.4) into (3.5) yields the small-scale T-Sylvester equation

$$(\mathcal{W}_m^T A \mathcal{V}_m) Y_m + Y_m^T (\mathcal{V}_m^T B \mathcal{W}_m) = (\mathcal{W}_m^T C_1)(\mathcal{W}_m^T C_2)^T. \tag{3.6}$$

Assuming that this equation admits a unique solution, it can be solved within $O(p_m^3)$ operations using the Bartels-Stewart-like algorithm discussed in Section 2.

It is important to emphasize that most existing projection methods for the standard Sylvester equation $FX + XG = C_1 C_2^T$ use a Galerkin approach instead of a Petrov-Galerkin approach. This means that the same tensorized subspace $\mathbb{V}_m \otimes \mathbb{W}_m$ is used for searching the approximate solution and testing the residual:

$$X_m \in \mathbb{V}_m \otimes \mathbb{W}_m \quad \text{and} \quad (FX_m + X_m G - C_1 C_2^T) \perp \mathbb{V}_m \otimes \mathbb{W}_m \,. \tag{3.7}$$

In contrast, our projection framework for T-Sylvester equations involves two different tensorized subspaces $\mathbb{V}_m \otimes \mathbb{W}_m$ and $\mathbb{W}_m \otimes \mathbb{W}_m$ as search and test spaces, respectively. This is needed to ensure that the compressed equation (3.6) is again a T-Sylvester equation, which allows for its inexpensive solution. In the language of projection methods [33, 34], our framework for T-Sylvester equations yields *oblique projection methods*, while the one for Sylvester equations yields *orthogonal projection methods*.

The choice of the subspaces $\mathbb{V}_m$ and $\mathbb{W}_m$ decisively determines the quality of the approximation X_m obtained from (3.2) and (3.6). Two different choices adapted to the structure of the T-Sylvester equation will be studied in detail in the subsequent sections.

4 Block Krylov subspaces for the T-Sylvester equation

In order to motivate the choice of adequate subspaces $\mathbb{V}_m$ and $\mathbb{W}_m$ for the projection framework from Section 3, we will first consider the generalized Schur decomposition (2.1) of the pencil $A - \lambda B^T$, which implies

$$B^{-T} A = V \, T_B^{-1} T_A \, V^T \quad \text{and} \quad B^T V = W T_B \,. \tag{4.1}$$

Letting $V(:, 1:p)$ denote the first $p < n$ columns of V, this means that $V(:, 1:p)$ represents an orthonormal basis for an invariant subspace of $B^{-T}A$, provided that the subdiagonal entry $(p+1, p)$ of the upper quasi-triangular matrix T_A is zero. Moreover,

$$\operatorname{range}(W(:, 1:p)) = \operatorname{range}(B^T V(:, 1:p)) = B^T \operatorname{range}(V(:, 1:p)). \tag{4.2}$$

This suggests to choose a subspace $\mathbb{V}_m$ that contains good approximations to invariant subspaces of $B^{-T}A$ and set $\mathbb{W}_m = B^T \mathbb{V}_m$. Krylov subspaces are known to often contain excellent approximations to invariant subspaces [34]; it is therefore natural to choose $\mathbb{V}_m$ as a Krylov subspace for the matrix $B^{-T}A$. On the other hand, in view of (3.6), it is important to ensure $\operatorname{range}(C_1) \cup \operatorname{range}(C_2) \subset \mathbb{W}_m$, in order to fully preserve the information from the right-hand side during the projection. Taking both considerations into account leads us to choose the block Krylov subspaces

$$\mathbb{V}_m = \mathcal{K}_m(B^{-T}A, B^{-T}\,[C_1, C_2]) \,, \tag{4.3}$$

$$\mathbb{W}_m = B^T \mathbb{V}_m \,. \tag{4.4}$$

Note that the definition (1.2) and the algebraic identity $B^T(B^{-T}A)^kB^{-T} = (AB^{-T})^k$ imply

$$\mathbb{W}_m = \mathcal{K}_m(AB^{-T}, [C_1, C_2]), \tag{4.5}$$

which shows that $\mathbb{W}_m$ indeed contains the columns of C_1, C_2. The computational cost can be reduced if range(C_1)=range(C_2). Indeed, in this case it holds that

$$\mathbb{V}_m = \mathcal{K}_m(B^{-T}A\,, B^{-T}\,[C_1, C_2]) = \mathcal{K}_m(B^{-T}A\,, B^{-T}\,C_1),$$

yielding a block Krylov subspace of half the dimension. Algorithms for constructing orthonormal bases for $\mathbb{V}_m, \mathbb{W}_m$ will be discussed in Section 4.2.

Finally, we remark that the roles of A and B can be reversed. By transposing (3.1), we obtain

$$B^TX + X^TA^T = C_2C_1^T. \tag{4.6}$$

This is again a T-Sylvester equation, with A replaced by B^T, B by A^T, C_1 by C_2, and C_2 by C_1. Assuming that A is invertible, we thus arrive at the block Krylov subspaces

$$\mathbb{V}'_m = \mathcal{K}_m(A^{-1}B^T\,, A^{-1}\,[C_1, C_2])\,, \tag{4.7}$$

$$\mathbb{W}'_m = A\mathbb{V}'_m\,. \tag{4.8}$$

Following the arguments in Section 6, we can thus expect fast convergence of the projection method applied to (4.6) with $\mathbb{V}'_m, \mathbb{W}'_m$ if $\rho(A^{-1}B^T) < 1$ and if this quantity is not too close to one. Equivalently, this means that all eigenvalues of $B^{-T}A$ should be located well outside the unit circle.

We can thus choose between two methods depending on the location of the eigenvalues of $B^{-T}A$:

(1) when all eigenvalues are well inside the unit circle, use the projection method with the block Krylov subspaces $\mathbb{V}_m, \mathbb{W}_m$ defined in (4.3)–(4.4);

(2) when all eigenvalues are well outside the unit circle, use the projection method with the block Krylov subspaces $\mathbb{V}'_m, \mathbb{W}'_m$ defined in (4.7)–(4.8).

The power method can be used to estimate $\rho(B^{-T}A)$, $\rho(A^{-1}B^T)$, and guide the decision between (1) and (2). In Section 5, we will discuss a variant that combines both approaches and does not require such a decision.

In the rest of this section, we will focus on the equation (3.1) and the block Krylov subspaces (4.3)–(4.4), but it should be kept in mind that our developments apply likewise to the transposed equation.

4.1 Solvability of the compressed equation

When performing the projection method from Section 3, we require the compressed T-Sylvester equation (3.6) to admit a unique solution. The unique solvability of the original equation (3.1) is not sufficient to guarantee this property. Similar difficulties arise in projection methods for the Sylvester equation $FX + XG = C_1C_2^T$, where this issue is addressed by imposing the more restrictive condition that the fields of values (instead of the spectra) of F and $-G$ are disjoint, see [35, Section 4.4.1] and the references therein. In fact, the assumption $\mathcal{F}(F) \cap \mathcal{F}(-G) = \emptyset$ is one of the keys to prove rigorous error bounds for rational Galerkin projection methods for the Sylvester equation [4]. Theorem 4.1 below suggests an analogous condition for T-Sylvester equations. Note that the result of the theorem does not only hold for block Krylov subspaces, but for any pair of subspaces $\mathbb{V}_m, \mathbb{W}_m$ satisfying $\mathbb{W}_m = B^T\mathbb{V}_m$.

Theorem 4.1. *Let $A, B \in \mathbb{R}^{n\times n}$ and assume that B is nonsingular. Let the columns of $\mathcal{V}_m, \mathcal{W}_m \in \mathbb{R}^{n\times p_m}$ form orthonormal bases of two subspaces $\mathbb{V}_m, \mathbb{W}_m \subseteq \mathbb{R}^n$, respectively, which satisfy $\mathbb{W}_m = B^T\mathbb{V}_m$. If $\mathcal{F}(AB^{-T})$ is inside the open unit disk then*

(a) *$AX + X^TB = C$ has a unique solution for every right-hand side C, and*

(b) *$(\mathcal{W}_m^TA\mathcal{V}_m)Y_m + Y_m^T(\mathcal{V}_m^TB\mathcal{W}_m) = \tilde{C}$ has a unique solution for every right-hand side $\tilde{C}$.*

Proof. (a) Since $\mathcal{F}(AB^{-T})$ is inside the open unit disk, all eigenvalues of AB^{-T} are inside the open unit disk and hence the spectrum of $A - \lambda B^T$ is reciprocal free. Therefore, part (a) follows from Theorem 2.2.

(b) The assumption $\mathbb{W}_m = B^T\mathbb{V}_m$ implies that there exists a nonsingular matrix $Q_m \in \mathbb{R}^{p_m\times p_m}$ such that $B^T\mathcal{V}_m = \mathcal{W}_mQ_m$. This implies

$$\begin{aligned}\mathcal{W}_m^TA\mathcal{V}_m &= \mathcal{W}_m^TAB^{-T}\mathcal{W}_mQ_m,\\ (\mathcal{V}_m^TB\mathcal{W}_m)^T &= \mathcal{W}_m^TB^T\mathcal{V}_m = \mathcal{W}_m^T\mathcal{W}_mQ_m = Q_m,\end{aligned}$$

showing that the pencil $(\mathcal{W}_m^TA\mathcal{V}_m) - \lambda(\mathcal{V}_m^TB\mathcal{W}_m)^T$ is strictly equivalent to the regular pencil $\mathcal{W}_m^TAB^{-T}\mathcal{W}_m - \lambda I$, and both pencils have the same eigenvalues. Since, $\mathcal{F}(\mathcal{W}_m^TAB^{-T}\mathcal{W}_m) \subseteq \mathcal{F}(AB^{-T})$, all eigenvalues of $\mathcal{W}_m^TAB^{-T}\mathcal{W}_m$ are inside the open unit disk. Therefore, part (b) also follows from Theorem 2.2. □

The assumption that $\mathcal{F}(AB^{-T})$ is inside the open unit disk is sufficient but by no means necessary for $\rho(AB^{-T}) = \rho(B^{-T}A) < 1$ and the unique solvability of the compressed equation. In our numerical experience, we have regularly observed fast convergence of the projection method in situations where $\rho(B^{-T}A) < 1$, but $\mathcal{F}(AB^{-T})$ is much larger than the open unit disk.

4.2 Algorithmic details of the block Krylov subspace method

In this section we describe the algorithm associated with the Petrov-Galerkin projection onto the block Krylov subspaces (4.3)-(4.4). One can choose between two implementations: (i) use the block Arnoldi method [34] to compute an orthonormal basis $\mathcal{V}_m$ of $\mathbb{V}_m$ and, then, an orthonormal basis $\mathcal{W}_m$ of $\mathbb{W}_m$ by orthonormalizing $B^T\mathcal{V}_m$; or (ii) take into account (4.5) to first compute an orthonormal basis $\mathcal{W}_m$ of $\mathbb{W}_m$ with the block Arnoldi method, and, then, an orthonormal basis $\mathcal{V}_m$ of $\mathbb{V}_m$ by orthonormalizing $B^{-T}\mathcal{W}_m$. Observe that the implementation (ii) does not require more linear system solves than (i), since the linear solves needed for computing $B^{-T}\mathcal{W}_m$ can be re-used by the Arnoldi method in computing $\mathcal{W}_{m+1}$. In fact, (ii) is a bit cheaper than (i), since the matrix-vector products $B^T\mathcal{V}_m$ in (i) cannot be re-used by block Arnoldi to compute $\mathcal{V}_{m+1}$. We will describe in detail only implementation (i), which is similar to the one in Section 5, and leave to interested readers the details of (ii). Observe that applying block Arnoldi twice for constructing $\mathcal{V}_m$ and $\mathcal{W}_m$ independently requires more work and does not provide us with the connection between both bases, which is needed to cheaply compute the coefficients of the compressed equations (see Proposition 4.2).

After m steps of the block Arnoldi method the following block Arnoldi relation holds (see, e.g., [34, Section 6.5]):

$$B^{-T}A\,\mathcal{V}_m = \mathcal{V}_mH_m + V_{m+1}H_{m+1,m}E_m^T = \mathcal{V}_{m+1}\underline{H}_m. \tag{4.9}$$

Here, $\mathcal{V}_m = [V_1,\ V_2,\ \ldots,\ V_m] \in \mathbb{R}^{n\times 2mr}$ is an orthonormal basis of $\mathbb{V}_m = \mathcal{K}_m(B^{-T}A\,, B^{-T}\,[C_1, C_2])$, E_m denotes the last $2r$ columns of the $2mr \times 2mr$ identity matrix, and $H_m \in \mathbb{R}^{2mr\times 2mr}$, $\underline{H}_m \in \mathbb{R}^{2(m+1)r\times 2mr}$ are block Hessenberg matrices with

$$\underline{H}_m = \begin{bmatrix} H_m \\ H_{m+1,m}E_m^T \end{bmatrix}.$$

The matrix $\mathcal{W}_m$ can be obtained from $B^T\mathcal{V}_m$ by means of the "skinny" QR decomposition,

$$\mathcal{W}_m Z_m = B^T\mathcal{V}_m, \qquad Z_m \in \mathbb{R}^{2mr\times 2mr}, \tag{4.10}$$

where $\mathcal{W}_m = [W_1,\ W_2,\ \ldots,\ W_m] \in \mathbb{R}^{n\times 2mr}$ is an orthonormal basis of $\mathbb{W}_m = B^T\mathbb{V}_m$ and Z_m is upper triangular. With these expressions, we can show that the matrices

$$H_{A,m} := \mathcal{W}_m^T A\mathcal{V}_m, \qquad H_{B,m} := \mathcal{V}_m^T B\mathcal{W}_m$$

can be cheaply obtained during the generation of the bases $\mathcal{V}_m, \mathcal{W}_m$. In particular, the two matrices $H_{A,m}, H_{B,m}$ can be expanded as the iteration proceeds.

Proposition 4.2. *With the notation introduced above, the following relations hold:*

$$H_{A,m} = [I_{2mr}\ ,\ 0_{2mr\times 2r}]\, Z_{m+1}\underline{H}_m \qquad \textit{and} \qquad H_{B,m} = Z_m^T\,. \tag{4.11}$$

Proof. Using (4.9) and (4.10) for $m+1$, we obtain

$$H_{A,m} = \mathcal{W}_m^T A\mathcal{V}_m = \mathcal{W}_m^T B^T\mathcal{V}_{m+1}\underline{H}_m = \mathcal{W}_m^T\mathcal{W}_{m+1}Z_{m+1}\underline{H}_m = [I_{2mr},\ 0_{2mr\times 2r}]\, Z_{m+1}\underline{H}_m.$$

Using (4.10) we have $H_{B,m} = \mathcal{V}_m^T B\mathcal{W}_m = Z_m^T\mathcal{W}_m^T\mathcal{W}_m = Z_m^T$. □

Algorithm 1 gives an overview of our proposed block Krylov subspace method for solving large-scale T-Sylvester equations, which amounts to the projection method from Section 3 with the subspaces $\mathbb{V}_m$, $\mathbb{W}_m$ defined in (4.3)–(4.4).

Algorithm 1 Block Krylov method for solving T-Sylvester equation

Input: $A, B \in \mathbb{R}^{n\times n}$ and $C_1, C_2 \in \mathbb{R}^{n\times r}$.
Output: Factors of approximate solution X_m of $AX + X^TB = C_1C_2^T$.

Compute V_1 orthonormal basis of range$(B^{-T}[C_1,\ C_2])$ and set $\mathcal{V}_0 = \mathcal{W}_0 = \emptyset$.
for $m = 1, 2, \ldots$ **do**
1. $\mathcal{V}_m = [\mathcal{V}_{m-1},\ V_m]$.
2. $W_m \leftarrow$ orthonormalize columns of B^TV_m with respect to $\mathcal{W}_{m-1}$.
3. $\mathcal{W}_m = [\mathcal{W}_{m-1},\ W_m]$.
4. Expand $H_{A,m} = \mathcal{W}_m^T A\mathcal{V}_m$, $H_{B,m} = \mathcal{V}_m^T B\mathcal{W}_m$, and $\tilde{C}_m = (\mathcal{W}_m^T C_1)(\mathcal{W}_m^T C_2)^T$.
5. Compute Y_m solution of $H_{A,m}\,Y_m + Y_m^T H_{B,m} = \tilde{C}_m$ via [13, Algorithm 3.1].
if converged **then**
Return $\mathcal{V}_m, Y_m, \mathcal{W}_m$ such that $X_m = \mathcal{V}_m Y_m \mathcal{W}_m^T$ and stop.
end if
6. $V_{m+1} \leftarrow$ orthonormalize columns of $B^{-T}AV_m$ with respect to $\mathcal{V}_m$.
end for

Several comments concerning the implementation of Algorithm 1 are in order:

Step 2. The orthogonalization of B^TV_m with respect to the columns of $\mathcal{W}_{m-1}$ in Step 2 is performed by applying the classical Gram-Schmidt process twice.

Step 6. One step of the standard block Arnoldi method presented in [34, Algorithm 6.8] is used in Step 6: it requires $2r$ matrix-vector multiplies with A, followed by $2r$ solves with B^T. For a sparse matrix B, the latter can be implemented efficiently by computing and storing a sparse LU factorization of B^T beforehand. Solves with B^T then only require forward/backward substitutions with the sparse LU factors. For simplicity, it is assumed that breakdowns do not occur in the block Arnoldi method and hence the matrices $\mathcal{V}_m \in \mathbb{R}^{n\times(2mr)}$ and $\mathcal{W}_m \in \mathbb{R}^{n\times(2mr)}$ are of full rank and have orthonormal columns.

Step 4. The matrices $H_{A,m} = \mathcal{W}_m^T A \mathcal{V}_m$ and $H_{B,m} = \mathcal{V}_m^T B \mathcal{W}_m$ in Step 4 are cheaply updated using the expressions presented in Proposition 4.2. Observe, in addition, that (4.5) implies $\mathrm{range}[C_1, C_2] = \mathbb{W}_1 = \mathrm{range}(W_1)$, and hence $\mathcal{W}_m^T C_i = [(W_1^T C_i)^T, 0, \ldots, 0]^T$, for $i = 1, 2$.

Stopping criterion. To check convergence, we use the stopping criterion

$$\frac{\|AX_m + X_m^T B - C_1 C_2^T\|_F}{(\|A\|_F + \|B\|_F)\|Y_m\|_F + \|C_1 C_2^T\|_F} < \texttt{tol}, \tag{4.12}$$

where `tol` is a fixed user-specified tolerance, $X_m = \mathcal{V}_m Y_m \mathcal{W}_m^T$. Analogous stopping criteria are used in other algorithms for matrix equations [37, p. 1275], see also [21, Chapter 16].

The residual norm $\|AX_m + X_m^T B - C_1 C_2^T\|_F$ is calculated inexpensively via the relation in Proposition 4.3 below, which again uses information obtained in Steps 2 and 6. The calculation of the quantities $\|A\|_F$ and $\|B\|_F$ in (4.12) needs to be performed only once and is inexpensive if A and B are sparse. To compute $\|C_1 C_2^T\|_F$, we use the expression

$$\|C_1 C_2^T\|_F^2 = \mathrm{trace}\left((C_1 C_2^T)^T C_1 C_2^T\right) = \mathrm{trace}\left((C_1^T C_1)(C_2^T C_2)\right),$$

which involves the small $r \times r$ matrices $C_1^T C_1$ and $C_2^T C_2$ only.

The expressions (4.11) allow us to construct $H_{A,m}$ at the cost of a small matrix multiplication and $H_{B,m}$ at no additional cost. The expressions (4.11) also show that $H_{A,m}$ is a block Hessenberg matrix and $H_{B,m}^T$ is upper triangular. To a certain extent, this structure can be exploited to reduce the cost of the QZ algorithm [19, Section 7.7] for computing the generalized Schur decomposition needed when solving the compressed equation. A minor complication in the expression (4.11) for $H_{A,m}$ is that it requires Z_{m+1}, which only becomes available after the orthonormalization of $B^T \mathcal{V}_{m+1}$ has been performed. This issue, however, can be easily addressed by slightly reorganizing Algorithm 1.

Finally, the following proposition gives an expression for the residual norm that requires the computation of the Frobenius norm for a small $2r \times 2mr$ matrix only, in the spirit of a corresponding result for Lyapunov equations [26, Theorem 2.1].

Proposition 4.3. *With the notation introduced in Proposition 4.2, the following relation holds for the residual norm:*

$$\|AX_m + X_m^T B - C_1 C_2^T\|_F = \|Z_{m+1,m+1} H_{m+1,m} E_m^T Y_m\|_F,$$

where $Z_{m+1,m+1}$ is the trailing $2r \times 2r$ principal submatrix of Z_{m+1}.

Proof. Set $R_m := AX_m + X_m^T B - C_1 C_2^T = A\mathcal{V}_m Y_m \mathcal{W}_m^T + \mathcal{W}_m Y_m^T \mathcal{V}_m^T B - C_1 C_2^T$ and let the columns of $\mathcal{W}_{m,\perp}$ form an orthonormal basis for $\mathbb{W}_m^\perp$, such that the first $2r$ columns of $\mathcal{W}_{m,\perp}$ coincide with W_{m+1}. Then $\mathcal{W}_{m,\perp}^T \mathcal{W}_m = 0$, $\mathcal{W}_{m,\perp}^T C_1 = \mathcal{W}_{m,\perp}^T C_2 = 0$ and the relations (4.9)–(4.10) imply

$$\begin{aligned}
\|\mathcal{W}_{m,\perp}^T R_m \mathcal{W}_m\|_F &= \|\mathcal{W}_{m,\perp}^T A \mathcal{V}_m Y_m\|_F = \|\mathcal{W}_{m,\perp}^T B^T \mathcal{V}_{m+1} \underline{H}_m Y_m\|_F \\
&= \|\mathcal{W}_{m,\perp}^T \mathcal{W}_{m+1} Z_{m+1} \underline{H}_m Y_m\|_F = \|E_{m+1}^T Z_{m+1} \underline{H}_m Y_m\|_F \\
&= \|Z_{m+1,m+1} H_{m+1,m} E_m^T Y_m\|_F.
\end{aligned}$$

Similarly,

$$\|\mathcal{W}_m^T R_m \mathcal{W}_{m,\perp}\|_F = \|Y_m^T \mathcal{V}_m^T B \mathcal{W}_{m,\perp}\|_F = \|Y_m^T Z_m^T \mathcal{W}_m^T \mathcal{W}_{m,\perp}\|_F = 0$$

and $\|\mathcal{W}_{m,\perp}^T R_m \mathcal{W}_{m,\perp}\|_F = 0$. Using that the Petrov-Galerkin condition (3.5) implies $\|\mathcal{W}_m^T R_m \mathcal{W}_m\|_F = 0$, we thus obtain

$$\begin{aligned}
\|R_m\|_F^2 &= \|\mathcal{W}_m^T R_m \mathcal{W}_m\|_F^2 + \|\mathcal{W}_{m,\perp}^T R_m \mathcal{W}_m\|_F^2 + \|\mathcal{W}_m^T R_m \mathcal{W}_{m,\perp}\|_F^2 + \|\mathcal{W}_{m,\perp}^T R_m \mathcal{W}_{m,\perp}\|_F^2 \\
&= \|\mathcal{W}_{m,\perp}^T R_m \mathcal{W}_m\|_F^2 = \|Z_{m+1,m+1} H_{m+1,m} E_m^T Y_m\|_F^2,
\end{aligned}$$

which completes the proof. □

5 Extended Krylov subspaces for the T-Sylvester equation

Extended Krylov subspaces are often used in modern algorithms for solving large-scale Sylvester equations (see [37], [35, Section 4.4.1] and the references therein). One motivation for their use is to attain spaces that have smaller dimension compared to standard Krylov spaces but contain equally good approximations of the solution. This results in decreased storage requirements for the dense matrices $\mathcal{V}_m$ and $\mathcal{W}_m$ needed to get the approximate solution (3.2). In addition, in the case of standard Sylvester equations, it has been observed that these enriched spaces exhibit an impressive performance despite the fact that multiplication by inverses, i.e., solution of large linear systems, are involved in their construction, in contrast to standard Krylov spaces. In the case of the T-Sylvester equation $AX + X^TB = C_1C_2^T$ the use of extended Krylov subspaces is even more natural, since the standard block Krylov spaces (4.3) and (4.4) adapted to this equation already involve inverses. Other methods using enriched Krylov subspaces suggest themselves, such as Rational Krylov space and ADI methods (see, e.g., [35] and also the discussion in section 6); a complete overview of all possible variants is clearly beyond this paper, although it may be interesting to explore these alternatives further.

Following [37, Section 3], in this section we propose the use of *extended block Krylov subspaces* to implement the projection method in Section 3:

$$\mathbb{V}_m = \mathcal{K}_m(B^{-T}A\,, B^{-T}\,[C_1, C_2]) + \mathcal{K}_{m+1}((B^{-T}A)^{-1}\,, B^{-T}\,[C_1, C_2])\,, \tag{5.1}$$

$$\mathbb{W}_m = B^T\mathbb{V}_m\,. \tag{5.2}$$

Let $\mathcal{C}_B = B^{-T}[C_1, C_2] \in \mathbb{R}^{n\times 2r}$. The subspace $\mathbb{V}_m$ is iteratively expanded with two blocks of vectors at the time by an Arnoldi-type process as

$$[\mathcal{C}_B, (B^{-T}A)^{-1}\mathcal{C}_B],\ [(B^{-T}A)\mathcal{C}_B, (B^{-T}A)^{-2}\mathcal{C}_B],\ [(B^{-T}A)^2\mathcal{C}_B, (B^{-T}A)^{-3}\mathcal{C}_B], \dots$$

where for each pair, the first block of vectors expands the space in $B^{-T}A$, while the second block expands the space in $(B^{-T}A)^{-1} = A^{-1}B^T$. The actual basis is computed by orthogonalizing the newly built vectors by means of the Gram-Schmidt process [37]. Note that $2r + 2r$ vectors are added to the basis during each iteration. In summary, after m iterations the following matrix whose columns form an orthonormal basis of $\mathbb{V}_m$ is iteratively generated,

$$\mathcal{V}_m = [V_1, V_2, \dots, V_m] \in \mathbb{R}^{n\times 4mr}, \qquad V_j = [V_j^{(1)}, V_j^{(2)}] \in \mathbb{R}^{n\times 4r}, \tag{5.3}$$

where the "odd" $n \times 2r$ matrices $V_j^{(1)}$ originally stem from multiplications with $B^{-T}A$, while the "even" $n \times 2r$ matrices $V_j^{(2)}$ stem from multiplications with $A^{-1}B^T$. In the following we shall heavily rely on this type of partitioning to build key recurrences.

From the definition (1.2) and the identity $B^T(B^{-T}A)^kB^{-T} = (AB^{-T})^k$, $k \in \mathbb{Z}$, we obtain

$$\mathbb{W}_m = \mathcal{K}_m(AB^{-T}\,, [C_1, C_2]) + \mathcal{K}_{m+1}((AB^{-T})^{-1}\,, [C_1, C_2]). \tag{5.4}$$

From this equation, it is immediate to prove that

$$AB^{-T}\,\mathbb{W}_m \subseteq \mathbb{W}_{m+1}. \tag{5.5}$$

These relations show that there is nothing special about the matrix equation (5.2), since the same properties could be obtained by explicitly building the two extended spaces $\mathbb{V}_m$, $\mathbb{W}_m$ as in (5.1) and (5.4), respectively. On the other hand, as already mentioned, using (5.2) allows one to avoid extra system solves, and provides us with the connection between orthonormal bases of both spaces to cheaply compute the coefficients of the compressed equations. Observe also that (5.4) allows one to construct first an orthonormal basis $\mathcal{W}_m$ of $\mathbb{W}_m$ and then use $\mathbb{V}_m = B^{-T}\mathbb{W}_m$ for computing an

orthonormal basis of $\mathbb{V}_m$. However, this approach requires more system solves than the approach we follow below, since only half of the solves in $B^{-T}\mathcal{W}_m$ can be re-used for expanding $\mathbb{W}_m$ to $\mathbb{W}_{m+1}$.

Denoting with $\mathcal{V}_m \in \mathbb{R}^{n\times 4mr}$ the orthonormal basis of $\mathbb{V}_m$ in (5.3), we construct an orthonormal basis $\mathcal{W}_m \in \mathbb{R}^{n\times 4mr}$ of $\mathbb{W}_m$ by using again the "skinny" QR factorization

$$\mathcal{W}_m Z_m = B^T \mathcal{V}_m \,, \tag{5.6}$$

from which it readily follows that

$$H_{B,m} := \mathcal{V}_m^T B \mathcal{W}_m = Z_m^T.$$

The derivation of an updating expression for $H_{A,m} := \mathcal{W}_m^T A \mathcal{V}_m$ which does not involve multiplications with A may be obtained as follows. From (5.1), we get that $B^{-T}A\mathbb{V}_m \subseteq \mathbb{V}_{m+1}$ and this implies that there exists a $4(m+1)r \times 4mr$ block Hessenberg matrix $\underline{K}_m$ such that $B^{-T}A\mathcal{V}_m = \mathcal{V}_{m+1}\underline{K}_m$. The matrix $\underline{K}_m$ is not obtained directly from the construction of the extended Krylov space $\mathbb{V}_m$, in contrast with the situation for standard Krylov spaces, but $\underline{K}_m$ can be computed without extra large matrix-vector multiplications and system solves via the recursion presented in [37, Proposition 3.2]. Therefore, with (5.6), we get:

$$\begin{aligned} H_{A,m} &= \mathcal{W}_m^T A \mathcal{V}_m = \mathcal{W}_m^T B^T \left(B^{-T} A \mathcal{V}_m\right) \\ &= \mathcal{W}_m^T B^T \, \mathcal{V}_{m+1} \underline{K}_m = \mathcal{W}_m^T \mathcal{W}_{m+1} Z_{m+1} \underline{K}_m \\ &= [I_{4mr}, 0] \, Z_{m+1} \underline{K}_m \,. \end{aligned}$$

This method for computing $H_{A,m}$ is very efficient since it only involves a small matrix product and allows to easily expand $H_{A,m}$ as the iteration proceeds. Observe also that the expression above shows that $H_{A,m}$ is block upper Hessenberg. In the appendix at the end of the paper, we include a detailed description of another method for computing $H_{A,m}$ which is slightly more efficient, since it adapts the recursion in [37, Proposition 3.2] to computing directly $H_{A,m}$ instead of $\underline{K}_m$. The method in the appendix is the one we have used in all the numerical tests presented in Section 7.

Algorithm 2 outlines the projection algorithm in Section 3 for the subspaces defined in (5.1) and (5.2). It is based on the algorithm introduced in [37, Section 3]. Technical comments similar to those made after Algorithm 1 apply likewise to Algorithm 2. In particular, the same stopping criterion (4.12) is used here. Note also that the product $(B^{-T}A)^{-1}V_m^{(2)} = A^{-1}B^T V_m^{(2)}$ in step 6 is computed by first multiplying the columns of $V_m^{(2)}$ with B^T and then solving $2r$ linear systems with A.

In the next proposition we show that, in the same manner as for the methods in Section 4, the residual norm can be computed without explicitly storing the residual matrix.

Proposition 5.1. *Let $H_{A,m} := \mathcal{W}_m^T A \mathcal{V}_m \in \mathbb{R}^{4mr\times 4mr}$ be partitioned in blocks as $H_{A,m} = (\mathtt{h}_{ij})_{1\le i,j\le m}$, where $\mathtt{h}_{ij} \in \mathbb{R}^{4r\times 4r}$, and let E_m be the matrix of the last $4r$ columns of I_{4mr}. Then, with the notation in Algorithm 2, the residual matrix $R_m = AX_m + X_m^T B - C_1 C_2^T$ satisfies*

$$\|R_m\|_F = \|\mathtt{h}_{m+1,m} \, E_m^T \, Y_m\|_F.$$

Proof. We have $R_m = A\mathcal{V}_m Y_m \mathcal{W}_m^T + \mathcal{W}_m Y_m^T \mathcal{V}_m^T B - C_1 C_2^T$. Combining (5.5) and (5.6) it follows that $\mathrm{range}(A\mathcal{V}_m) \subseteq \mathbb{W}_{m+1}$. In addition, $\mathrm{range}(\mathcal{W}_m) \subseteq \mathbb{W}_{m+1}$ and $\mathrm{range}(C_1) \subseteq \mathbb{W}_{m+1}$ by (5.4). Therefore, $\mathrm{range}(R_m) \subseteq \mathbb{W}_{m+1}$. An analogous argument shows that $\mathrm{range}(R_m^T) \subseteq \mathbb{W}_m$, by using again (5.6) and (5.4). As a consequence we can write $R_m = \mathcal{W}_{m+1}\widehat{R}_m\mathcal{W}_m^T$. Thus, $\widehat{R}_m = \mathcal{W}_{m+1}^T R_m \mathcal{W}_m$ and by (3.5)

$$\widehat{R}_m = \begin{bmatrix} \mathcal{W}_m^T R_m \mathcal{W}_m \\ W_{m+1}^T R_m \mathcal{W}_m \end{bmatrix} = \begin{bmatrix} 0 \\ W_{m+1}^T R_m \mathcal{W}_m \end{bmatrix}.$$

Algorithm 2 Extended block Krylov method for solving T-Sylvester equation

Input: $A, B \in \mathbb{R}^{n\times n}$ and $C_1, C_2 \in \mathbb{R}^{n\times r}$.

Output: Factors of approximate solution X_m of $AX + X^TB = C_1C_2^T$.

Compute V_1 orthonormal basis of range($[B^{-T}[C_1, C_2]\,,\, A^{-1}[C_1, C_2]]$), and set $\mathcal{V}_0 = \mathcal{W}_0 = \emptyset$.

for m=1,2,..., **do**

1. $\mathcal{V}_m = [\mathcal{V}_{m-1},\ V_m]$.
2. $W_m \leftarrow$ orthonormalize columns of B^TV_m with respect to $\mathcal{W}_{m-1}$.
3. $\mathcal{W}_m = [\mathcal{W}_{m-1},\ W_m]$.
4. Set $H_{A,m} = \mathcal{W}_m^TA\mathcal{V}_m$, $H_{B,m} = \mathcal{V}_m^TB\mathcal{W}_m$, and $\tilde{C}_m = (\mathcal{W}_m^TC_1)(\mathcal{W}_m^TC_2)^T$.
5. Compute Y_m solution of $H_{A,m}Y_m + Y_m^TH_{B,m} = \tilde{C}_m$ via [13, Algorithm 3.1].

if converged **then**

Return $\mathcal{V}_m, Y_m, \mathcal{W}_m$ such that $X_m = \mathcal{V}_mY_m\mathcal{W}_m^T$ and stop.

end if

6. $V'_{m+1} = [B^{-T}A\,V_m^{(1)}\,,\, (B^{-T}A)^{-1}V_m^{(2)}]$ where $V_m = [V_m^{(1)}, V_m^{(2)}]$.
7. $V_{m+1} \leftarrow$ orthonormalize columns of V'_{m+1} with respect to $\mathcal{V}_m$.

end for

Since $W_{m+1}^T\mathcal{W}_m = 0$ and $W_{m+1}^TC_1 = 0$, because range$(C_1) \subseteq \mathbb{W}_1 =$ range(W_1) by (5.4), we have $W_{m+1}^TR_m\mathcal{W}_m = W_{m+1}^TA\mathcal{V}_mY_m = \mathtt{h}_{m+1,1:m}Y_m$. Besides, $H_{A,m+1}$ is block upper Hessenberg, which implies that $\mathtt{h}_{m+1,1:m} = \mathtt{h}_{m+1,m}\,E_m^T$, which proves the result, since $\|R_m\|_F = \|\widehat{R}_m\|_F$. □

6 Relation to a T-Stein equation and a fixed point iteration

In this section, we provide another motivation for choosing the Krylov subspaces (4.3) and (4.4), which offers some intuition on the expected convergence of Algorithm 1. This intuition is fully confirmed by the numerical tests presented in Section 7.

Since B is assumed to be nonsingular, the T-Sylvester equation (3.1) is equivalent to $X^T = C_1C_2^TB^{-1} - AXB^{-1}$. Transposing this equation yields the fixed point equation

$$X = B^{-T}C_2C_1^T - B^{-T}X^TA^T. \tag{6.1}$$

Matrix equations of this form are sometimes called T-Stein equations [10]. The following theorem shows that the iterates produced by the fixed point iteration applied to (6.1) are contained in the tensor product $\mathbb{V}_m \otimes \mathbb{W}_m$ of the block Krylov subspaces in (4.3) and (4.4).

Theorem 6.1. *Let $A, B \in \mathbb{R}^{n\times n}$, $C_1, C_2 \in \mathbb{R}^{n\times r}$, and assume that B is nonsingular. Consider the sequence of matrices $\{\tilde{X}_m\}_{m=0}^{\infty}$ defined by the fixed point iteration*

$$\tilde{X}_0 = 0, \qquad \tilde{X}_{m+1} = B^{-T}C_2C_1^T - B^{-T}\tilde{X}_m^TA^T, \quad \text{for } m = 0, 1, 2, \ldots \tag{6.2}$$

Then it holds that $\tilde{X}_{m+1} \in \mathbb{V}_m \otimes \mathbb{W}_m$ for $m = 2, 3, \ldots$, where $\mathbb{V}_m = \mathcal{K}_m(B^{-T}A, B^{-T}\,[C_1, C_2])$ and $\mathbb{W}_m = \mathcal{K}_m(AB^{-T}, [C_1, C_2])$.

Proof. We will show by induction that range$(\tilde{X}_{m+1}) \subseteq \mathbb{V}_m$ and range$(\tilde{X}_{m+1}^T) \subseteq \mathbb{W}_m$; these two conditions taken together are equivalent to $\tilde{X}_{m+1} \in \mathbb{V}_m \otimes \mathbb{W}_m$.

The first three iterates of (6.2) are given by

$$\begin{aligned}
\tilde{X}_1 &= B^{-T}C_2C_1^T\,,\\
\tilde{X}_2 &= B^{-T}C_2C_1^T - B^{-T}C_1C_2^T(AB^{-T})^T\,,\\
\tilde{X}_3 &= B^{-T}C_2C_1^T - B^{-T}C_1C_2^T(AB^{-T})^T - (B^{-T}A)\,B^{-T}C_2C_1^T(AB^{-T})^T\,.
\end{aligned}$$

By inspection, $\operatorname{range}(\tilde{X}_3) \subseteq \mathbb{V}_2$ and $\operatorname{range}(\tilde{X}_3^T) \subseteq \mathbb{W}_2$.

Suppose now that $\operatorname{range}(\tilde{X}_m) \subseteq \mathbb{V}_{m-1}$ and $\operatorname{range}(\tilde{X}_m^T) \subseteq \mathbb{W}_{m-1}$ hold for some $m \geq 3$. Together with (6.2), this implies

$$\operatorname{range}(\tilde{X}_{m+1}^T) \subseteq \operatorname{range}(C_1) + \operatorname{range}(A\tilde{X}_m) \subseteq \operatorname{range}(C_1) + A\,\mathbb{V}_{m-1}.$$

Using

$$A\,\mathbb{V}_{m-1} = \operatorname{range}\big(\big[\, AB^{-T}[C_1, C_2]\,,\, (AB^{-T})^2[C_1, C_2], \ldots, (AB^{-T})^{m-1}[C_1, C_2]\,\big]\big) \subseteq \mathbb{W}_m,$$

we therefore obtain

$$\operatorname{range}(\tilde{X}_{m+1}^T) \subseteq \mathbb{W}_1 + \mathbb{W}_m = \mathbb{W}_m.$$

Analogously,

$$\begin{aligned}\operatorname{range}(\tilde{X}_{m+1}) &\subseteq \operatorname{range}(B^{-T}C_2) + \operatorname{range}(B^{-T}\tilde{X}_m^T)\\ &\subseteq \mathbb{V}_1 + B^{-T}\mathbb{W}_m = \mathbb{V}_1 + \mathbb{V}_m = \mathbb{V}_m,\end{aligned}$$

where we have used (4.4). We have thus shown that the statement of the theorem holds for $m+1$, which completes the induction proof. □

Theorem 6.1 shows that there is a $2mr \times 2mr$ matrix $\tilde{Y}_m$ such that

$$\tilde{X}_{m+1} = \mathcal{V}_m \tilde{Y}_m \mathcal{W}_m^T,$$

for orthonormal bases $\mathcal{V}_m, \mathcal{W}_m$ of $\mathbb{V}_m, \mathbb{W}_m$, respectively. In particular, $\tilde{X}_{m+1}$ has rank at most $2mr$. Although the iteration (6.2) itself operates with full $n \times n$ matrices, it is certainly possible to develop a low-rank variant, similar to the low-rank Smith method [30] for Lyapunov equations. More importantly, $\tilde{X}_{m+1}$ is in general different from the approximation X_m obtained when applying the projection method from Section 3 with the same block Krylov subspaces $\mathbb{V}_m, \mathbb{W}_m$. Although, $\tilde{X}_{m+1}$ and X_m are both contained in $\mathbb{V}_m \otimes \mathbb{W}_m$, the Petrov-Galerkin condition (3.5) will usually pick an approximation different from $\tilde{X}_{m+1}$. Still we believe that the established link to the fixed point iteration (6.2) also offers some intuition on the expected convergence of the projection method.

Establishing sufficient conditions that guarantee fast convergence of the projection method is likely a difficult problem, which is not considered in this work. In contrast, the convergence analysis of fixed point iterations like (6.2) is very simple. Note that although we use in Theorem 6.2 the spectral norm, any other submultiplicative matrix norm may be also used.

Theorem 6.2. *Let $A, B \in \mathbb{R}^{n\times n}$, $C_1, C_2 \in \mathbb{R}^{n\times r}$, and assume that B is nonsingular and $\rho(B^{-T}A) < 1$. Then the following statements hold:*

(a) *the T-Sylvester equation* (3.1) (*or, equivalently, the T-Stein equation* (6.1)) *has a unique solution X;*

(b) *the sequence of matrices $\{\tilde{X}_m\}_{m=0}^{\infty}$ defined by the fixed point iteration* (6.2) *converges to X;*

(c) *the error matrix $E_m = X - \tilde{X}_m$ satisfies $\|E_{2\ell}\|_2 \leq \left\|\left(B^{-T}A\right)^{\ell}\right\|_2 \left\|\left(B^{-1}A^T\right)^{\ell}\right\|_2 \|X\|_2$ for $\ell = 0, 1, 2, \ldots$.*

Proof. Part (a) follows immediately from Theorem 2.2; the condition $\rho(B^{-T}A) < 1$ implies that the spectrum of $A - \lambda B^T$ is reciprocal free.

By the definition (6.2) of $\tilde{X}_m$, we get $E_0 = X$ and $E_{m+1} = -B^{-T}E_m^T A^T$. Then a simple induction argument shows

$$E_{2\ell} = (B^{-T}A)^{\ell} X (B^{-1}A^T)^{\ell} \quad \text{and} \quad E_{2\ell+1} = -B^{-T}\,(AB^{-T})^{\ell} X^T (A^T B^{-1})^{\ell}\, A^T, \tag{6.3}$$

for $\ell = 0, 1, 2, \ldots$. Observe that $\rho(B^{-T}A) = \rho(AB^{-T}) = \rho(B^{-1}A^T) = \rho(A^TB^{-1})$. Hence, $\rho(B^{-T}A) < 1$ implies

$$\lim_{\ell\to\infty}(B^{-T}A)^\ell = \lim_{\ell\to\infty}(AB^{-T})^\ell = \lim_{\ell\to\infty}(B^{-1}A^T)^\ell = \lim_{\ell\to\infty}(A^TB^{-1})^\ell = 0,$$

see [22, Theorem 5.6.12], and $\lim_{m\to\infty} E_m = 0$. This proves (b). Part (c) follows from the first equation in (6.3). □

Standard properties of the spectral radius [22, Theorem 5.6.14] and Theorem 6.2(c) show that asymptotically

$$\|E_{2\ell}\|_2 \lesssim \big(\rho(B^{-T}A)\big)^{2\ell}\,\|X\|_2, \qquad \ell \to \infty\,. \tag{6.4}$$

Therefore fast convergence of the fixed point iteration (6.2) for the T-Stein equation is expected if $\rho(B^{-T}A) < 1$ and this spectral radius is not too close to one. By Theorem 6.1, this implies that $\mathbb{V}_m \otimes \mathbb{W}_m$ contains an approximation to X that rapidly becomes more accurate as m increases. Numerically, we have observed this behavior as well for the approximation X_m picked by the Petrov-Galerkin condition; the experiments in Section 7 demonstrate that the projection method with the block Krylov subspaces $\mathbb{V}_m, \mathbb{W}_m$ in (4.3)-(4.4) converges always quickly when $\rho(B^{-T}A) < 1$, although, sometimes, it converges also in other situations.

7 Numerical experiments

In this section we report on numerical experiments with the new Algorithms 1 and 2. All reported experiments were performed using Matlab R2012a on a PC with processor Intel (R) Core (TM) 2 Quad CPU Q9400 @2.66GHz (4CPUs), with 4096 MB of RAM, and with operating system Windows 7 Enterprise 64 bits. Both CPU time (in seconds) and the dimension of the approximation space $\mathbb{V}_m$ (equal to the dimension of $\mathbb{W}_m$) are used to measure the cost of the different methods. We also report on the number of iterations for completeness.

We compare our algorithms with the extended block Krylov subspace method applied to the standard Sylvester equation (2.3) (see [35, Section 4.4.1] and the references therein for details). Some remarks are in order concerning this comparison. In view of the eventual goal, we use the relative residual defined in (4.12) for the T-Sylvester equation (instead of the Sylvester equation) for monitoring the convergence of the approximate solution $\widehat{X}_m \approx \widehat{\mathcal{V}}_m\widehat{Y}_m\widehat{\mathcal{W}}_m^T$ provided by the extended block Krylov subspace method. To compute this residual norm, we note that

$$\widehat{R}_m = A\widehat{X}_m + \widehat{X}_m^TB - C_1C_2^T = \begin{bmatrix} A\widehat{\mathcal{V}}_m & \widehat{\mathcal{W}}_m & C_1 \end{bmatrix} \begin{bmatrix} \widehat{Y}_m & & \\ & \widehat{Y}_m^T & \\ & & I_r \end{bmatrix} \begin{bmatrix} \widehat{\mathcal{W}}_m^T \\ \widehat{\mathcal{V}}_m^TB \\ -C_2^T \end{bmatrix}.$$

Therefore, if $[A\widehat{\mathcal{V}}_m, \widehat{\mathcal{W}}_m, C_1] = Q_mS_m$ and $[\widehat{\mathcal{W}}_m, B^T\widehat{\mathcal{V}}_m, -C_2] = U_mG_m$ are two "skinny" QR factorizations then $\|\widehat{R}_m\|_F = \|S_m\,\mathrm{diag}(\widehat{Y}_m, \widehat{Y}_m^T, I_r)\,G_m^T\|_F$. Although this computation is cheap, it is far more expensive than the methods for computing residual norms presented in Propositions 4.3 and 5.1 for Algorithms 1 and 2, respectively. To take this into account, we will report CPU times with and *without* the computation of the relative residual in every iteration. One motivation for reporting both times is that it usually suffices to verify the stopping criterion (4.12) only every few (say 5 or 10) iterations and hence the actual time will be in between.

The following table summarizes the algorithms that will be compared.

BK	Algorithm 1
BK-TR	Algorithm 1 applied to the transposed equation (4.6)
EK	Algorithm 2
EK-woR	EK without time for residual computation
EK-SYLV	extended block Krylov subspace method applied to Sylvester equation (2.3)
EK-SYLV-woR	EK-SYLV without time for residual computation

Before presenting the numerical tests in detail, let us summarize the main conclusions we have obtained, since they offer a clear guidance on the selection of algorithm for solving a given problem. EK is the most reliable method as it succeeds to converge more often than any other method; indeed, we have not found any example where EK does not converge but one of the other methods does. However, for problems satisfying $\rho(B^{-T}A) < 1$ we recommend the use of BK, which we have observed to be much faster than the other methods in this situation. On the other hand, if $\rho(A^{-1}B^T) < 1$ then BK-TR is the fastest method. This is in accordance with the discussions in Section 6 and in the paragraphs after (4.6). The method EK-SYLV based on the standard Sylvester equation (2.3) is not recommended since it fails to converge in situations where other methods succeed and even when it works, it is almost always slower than EK.

We report only on a small sample of the many numerical tests we have performed. In particular, we only show results for right-hand sides $C_1C_2^T$ with vectors $C_1, C_2 \in \mathbb{R}^{n\times 1}$ generated by the command `randn` in Matlab, although we have also performed tests with random matrices $C_1, C_2 \in \mathbb{R}^{n\times r}$ with $r = 2:7$.

We present three types of numerical tests. In the first type of tests (7.1–7.3) the coefficient matrices A and B are finite difference discretizations of certain differential operators. In the second type of tests (7.4–7.9), we construct structured matrices A and B with prescribed eigenvalues for $B^{-T}A$. Finally, we discuss tests with block diagonal matrices built from the previous tests; see (7.1) and the numerical test 7.10.

Numerical test 7.1. A and B are $10^4 \times 10^4$ matrices obtained by finite-difference discretizations in $[0,1]\times[0,1]$ of the differential operators

$$a(u) = -u_{xx} - u_{yy} + y(1-x)u_x + \gamma u\,,$$
$$b(u) = -u_{xx} - u_{yy}\,,$$

respectively, where $\gamma = 10^4$. The vectors C_1, C_2 have been multiplied by 10^4 to match the magnitude of the entries of the matrices A and B. The following table displays the obtained results.

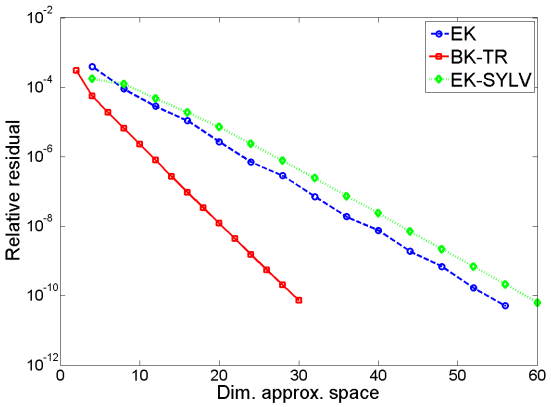
`tol` $= 10^{-10}$	EK	BK	BK-TR	EK-SYLV	EK-SYLV-woR	EK-woR
iterations	14	70	15	15	15	14
dim. approx. space	56	140	30	60	60	56
time (seconds)	4.277	50.856	1.684	6.255	2.901	3.666

The convergence history is shown in Figure 7.1(a). In this test, all eigenvalues of $B^{-T}A$ are well outside the unit circle; the smallest absolute value of the eigenvalues is approximately equal to 1.1226. All methods converge, with BK-TR being the fastest by far and BK the slowest. In this test, the use of extended Krylov spaces in EK does not add any essential information compared to BK-TR and therefore EK wastes half of the space.

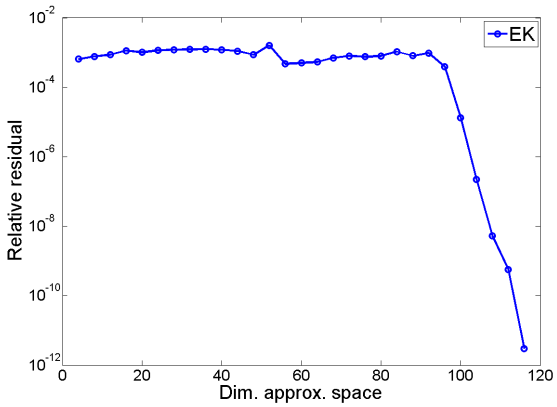
Numerical test 7.2. A and B are $10^4 \times 10^4$ matrices obtained by finite-difference discretizations in $[0,1]\times[0,1]$ of the differential operators

$$a(u) = (-\exp(-xy)\,u_x)_x + (-\exp(xy)\,u_y)_y + 100\,x\,u_x + \gamma\,u$$
$$b(u) = -u_{xx} - u_{yy},$$

where $\gamma = 5\cdot 10^4$. The vectors C_1, C_2 have been multiplied by 10^4 to match the magnitude of the entries of the matrices A and B.



(a) Numerical test 7.1.



(b) Numerical test 7.3.

Figure 7.1: Convergence histories for numerical tests 7.1 and 7.3.

$\texttt{tol} = 10^{-10}$	EK	BK	BK-TR	EK-SYLV	EK-SYLV-woR	EK-woR
iterations	8	83	8	8	8	8
dim. approx. space	32	166	16	32	32	32
time (seconds)	1.716	58.175	0.764	2.496	1.784	1.6160

The results are similar to the numerical test 7.1, only that the convergence (except for BK) is faster due to the fact that the eigenvalues of $B^{-T}A$ are even further outside the unit circle. The eigenvalue of smallest absolute value is approximately equal to 1.6159.

Numerical test 7.3. A and B are $10^4 \times 10^4$ matrices obtained via finite-difference discretizations in $[0,1] \times [0,1]$ of the differential operators

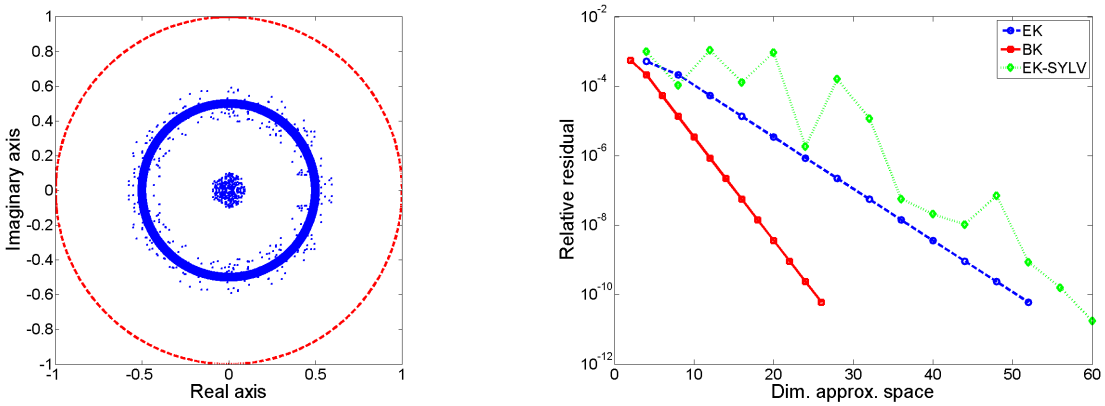
$$a(u) = (-\exp(-xy)\, u_x)_x + (-\exp(xy)\, u_y)_y + 100\, x\, u_x + \gamma u\,,$$
$$b(u) = -u_{xx} - u_{yy} + 100\, x\, u_x\,,$$

where $\gamma = 5 \cdot 10^4$. The vectors C_1, C_2 have been multiplied by 10^4 to match the magnitude. It turns out that only EK converges for this example. BK, BK-TR, and EK-SYLV do not converge within 100 iterations and their relative residuals remained essentially constant around 10^{-3}. This lack of convergence is marked with a star in the following table.

$\texttt{tol} = 10^{-10}$	EK	BK*	BK-TR*	EK-SYLV*	EK-SYLV-woR*	EK-woR
iterations	29	100	100	100	100	29
dim. approx. space	116	200	200	400	400	116
time (seconds)	10.920	70.715	63.835	521.214	71.807	9.158

The convergence history of EK is plotted in Figure 7.1(b), which shows stagnation until the dimension of the approximation subspace is 90, after which quick convergence sets in. A key difference to the previous tests 7.1 and 7.2 is that the eigenvalues of $B^{-T}A$ are now located inside *and* outside the unit circle, with the magnitudes of the eigenvalues varying between 0.8679 and 1.4563. This fact has drastic effects on the convergence of the methods making EK the only valid method.

Tests 7.1, 7.2, and 7.3 clearly show that the eigenvalue distribution of $B^{-T}A$ plays an important role in the behavior of the algorithms. To investigate this further, we prescribe the eigenvalues of



(a) Eigenvalues of $B^{-T}A$ and unit circle (dashed line).

(b) Convergence history.

Figure 7.2: Distribution of eigenvalues of $B^{-T}A$ and convergence history for numerical test 7.4.

$B^{-T}A$ in the numerical tests 7.4-7.9. We construct these matrices as follows: let A_1 be a block diagonal real matrix with 1×1 or 2×2 blocks, whose eigenvalues are, respectively, the real and complex eigenvalues that are prescribed for $B^{-T}A$. We consider in addition the following tridiagonal matrices P and Q

$$P = \begin{bmatrix} 1 & \frac{1}{3} & & \\ \frac{1}{2} & 1 & \ddots & \\ & \ddots & \ddots & \frac{1}{3} \\ & & \frac{1}{2} & 1 \end{bmatrix}, \qquad Q = \begin{bmatrix} 1 & \frac{1}{6} & & \\ \frac{1}{4} & 1 & \ddots & \\ & \ddots & \ddots & \frac{1}{6} \\ & & \frac{1}{4} & 1 \end{bmatrix}.$$

Next, we define the matrices $A = PA_1Q$ and $B = Q^TP^T$. The eigenvalues of A_1 and $B^{-T}A$ are equal because

$$B^{-T}A = (Q^TP^T)^{-T}PA_1Q = (PQ)^{-1}PA_1Q = Q^{-1}P^{-1}PA_1Q = Q^{-1}A_1Q.$$

Note that B is pentadiagonal and that A has at most 7 nonzero diagonals.

Numerical test 7.4. A and B are $10^5 \times 10^5$ matrices such that the eigenvalues of $B^{-T}A$ are distributed as in Figure 7.2(a). Observe that all eigenvalues of $B^{-T}A$ are well inside the unit circle. BK-TR did not converge in 70 iterations; it was not even close to.

$\texttt{tol} = 10^{-10}$	EK	BK	BK-TR*	EK-SYLV	EK-SYLV-woR	EK-woR
iterations	13	13	70	15	15	13
dim. approx. space	52	26	140	60	60	52
time (seconds)	14.227	6.474	179.416	107.796	15.568	11.776

The convergence history is shown in Figure 7.2(b). BK is the fastest method by far, which is in full agreement with the discussion in Section 6. The behaviors of BK and BK-TR are opposite to what has been observed in the tests 7.1 and 7.2, while EK and EK-SYLV behave the same.

Numerical test 7.5. A and B are $10^5 \times 10^5$ matrices such that the eigenvalues of $B^{-T}A$ are distributed as in Figure 7.3(a). Observe that all eigenvalues of $B^{-T}A$ are well outside the unit circle. BK did not converge in 70 iterations; it was not even close to.

tol $= 10^{-10}$	EK	BK*	BK-TR	EK-SYLV	EK-SYLV-woR	EK-woR
iterations	13	70	13	13	13	13
dim. approx. space	52	140	26	52	52	52
time (seconds)	14.024	168.041	6.489	73.492	12.448	11.965

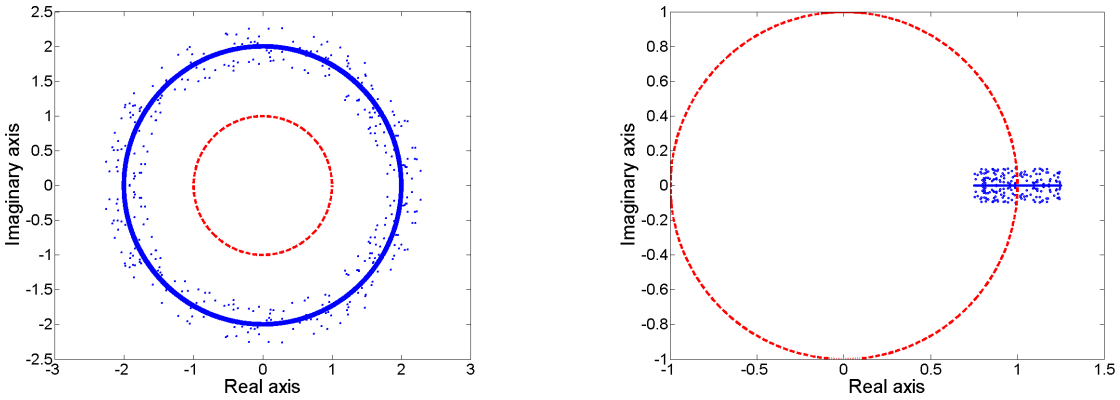
All results are very similar to the ones in test 7.1, except that the CPU times increase as a consequence of the larger size. However, we emphasize that the distributions of eigenvalues of $B^{-T}A$ are very different in both cases, even though the eigenvalues are well outside the unit circle in both cases.

Numerical test 7.6. A and B are $10^5 \times 10^5$ matrices such that the eigenvalues of $B^{-T}A$ are distributed as in Figure 7.3(b). Observe that $B^{-T}A$ has eigenvalue inside and outside the unit circle and near 1. The behaviors of the methods are as in test 7.3, with EK being the only converging method. All other methods were not close to convergence in 70 iterations.

tol $= 10^{-10}$	EK	BK*	BK-TR*	EK-SYLV*	EK-SYLV-woR*	EK-woR
iterations	21	70	70	70	70	21
dim. approx. space	84	140	140	280	280	84
time (seconds)	35.427	178.722	192.536	3041.208	296.339	32.569

Numerical test 7.7. A and B are $10^5 \times 10^5$ matrices such that the eigenvalues of $B^{-T}A$ are distributed as in Figure 7.4(a). In this test, the eigenvalues of $B^{-T}A$ are outside the unit circle and close to the real interval $[3, 5]$. This test illustrates a situation that we have not seen before: both BK-TR and BK exhibit fast convergence, which is surprising in the light of the discussion in Section 6 and the results of the previous tests. As expected, BK-TR is the fastest method, but the difference is not as large as in the other tests.

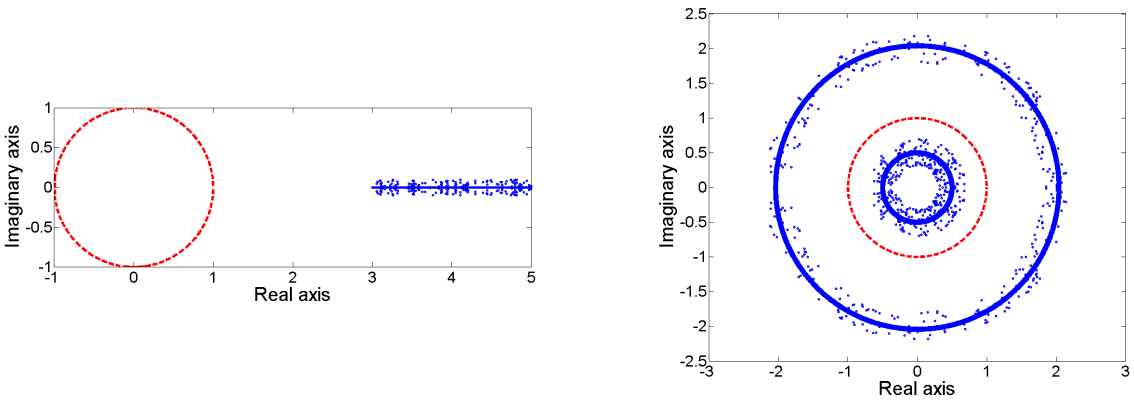
tol $= 10^{-10}$	EK	BK	BK-TR	EK-SYLV	EK-SYLV-woR	EK-woR
iterations	3	9	4	3	3	3
dim. approx. space	12	18	8	12	12	12
time (seconds)	1.435	8.190	1.092	3.135	1.544	1.382



(a) Numerical test 7.5 (unit circle-dashed line).

(b) Numerical test 7.6 (unit circle-dashed line).

Figure 7.3: Distribution of eigenvalues of $B^{-T}A$ for numerical tests 7.5 and 7.6.



(a) Numerical test 7.7 (unit circle-dashed line).

(b) Numerical test 7.8 (unit circle-dashed line).

Figure 7.4: Distribution of eigenvalues of $B^{-T}A$ for numerical tests 7.7 and 7.8.

Numerical test 7.8. The tests presented so far may give the impression that EK converges well for any real T-Sylvester equation; however, this is not true. In fact, when $B^{-T}A$ has eigenvalues inside and outside the unit circle, often none of the methods presented in this paper can be expected to work well. This test is an example of such a situation. A and B are $10^5 \times 10^5$ matrices such that the eigenvalues of $B^{-T}A$ are distributed as in Figure 7.4(b). With $\texttt{tol} = 10^{-10}$ none of the methods converged in 200 iterations, at which they reached relative residuals in the order of $10^{-2} - 10^{-3}$. We have performed other tests with eigenvalues of $B^{-T}A$ distributed as in Figure 7.4(b) but with matrices of size 1000×1000. These small sized cases can be solved directly with the dense solver [13, Algorithm 3.1] and we have observed that the solution is very far from having low-rank. As a consequence, the relative residuals for all methods remain constant around 10^{-3} until the dimensions of the approximation spaces are equal to 1000, i.e., they are the whole space.

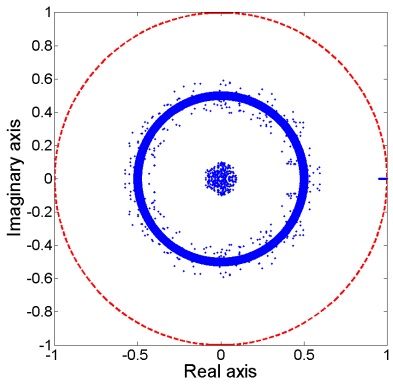
Numerical test 7.9. In this example, the eigenvalues of $B^{-T}A$ are prescribed to lie all inside the unit circle, but some of them are close to 1, see Figure 7.5(a). This test illustrates a situation where EK-SYLV does not converge in 100 iterations, while EK and BK do converge. The failure of BK-TR is expected from the discussion in Section 6 and the failure of EK-SYLV from the discussion after Theorem 2.3. After 100 iterations BK-TR and EK-SYLV were far from satisfying the stopping criterion. All methods have difficulties with this example and the convergence history for EK and BK differs from the other examples, see Figure 7.5(b).

$\texttt{tol} = 10^{-7}$	EK	BK	BK-TR*	EK-SYLV*	EK-SYLV-woR*	EK-woR
iterations	57	57	100	100	100	57
dim. approx. space	228	114	200	400	400	228
time (seconds)	261.566	112.351	336.915	6785.715	567.110	217.339

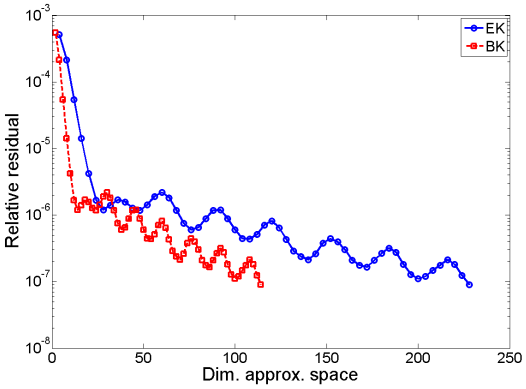
Finally, we have considered matrices A and B from the previous tests such that BK or BK-TR performs well but the other methods do not, and then we construct the new pair of matrices

$$\tilde{A} = \mathrm{diag}(A, B) \qquad \text{and} \qquad \tilde{B} = \mathrm{diag}(B, A)\,. \tag{7.1}$$

When proceeding in this way for the matrices from tests 7.1, 7.2, and 7.4, only EK is observed to perform well. For brevity we report only the case coming from test 7.4.



(a) Eigenvalues of $B^{-T}A$ and unit circle (dashed line).

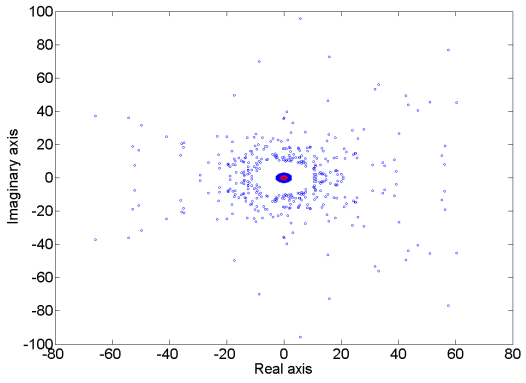


(b) Convergence history.

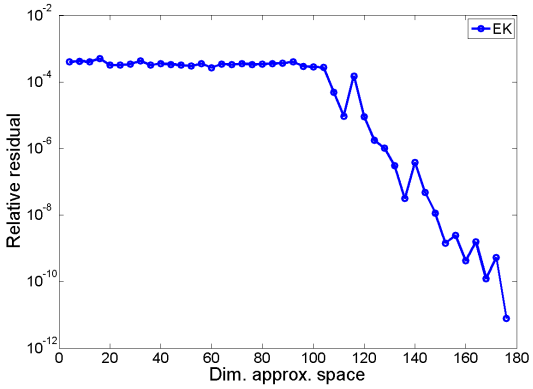
Figure 7.5: Distribution of eigenvalues of $B^{-T}A$ and convergence history for numerical test 7.9.

Numerical test 7.10. We construct matrices A and B of size $(2 \cdot 10^5) \times (2 \cdot 10^5)$ applying (7.1) to the matrices from test 7.4. BK, BK-TR and EK-SYLV do not converge in 70 iterations and they are by no means close to satisfying the stopping criterion. In contrast, EK converges in 44 iterations, see also Figure 7.6.

$\texttt{tol} = 10^{-10}$	EK	BK*	BK-TR*	EK-SYLV*	EK-SYLV-woR*	EK-woR
iteration	44	70	70	70	70	44
dim. approx. space	176	140	140	280	280	176
time (seconds)	305.621	352.733	372.358	6750.553	675.375	289.663



(a) Eigenvalues of $B^{-T}A$.



(b) Convergence history.

Figure 7.6: Distribution of eigenvalues of $B^{-T}A$ and convergence history for numerical test 7.10.

8 Conclusions and future work

We have introduced two new projection algorithms based on Krylov spaces for solving large-scale T-Sylvester equations and we have tested them extensively in many different situations. In these tests, we have compared the new methods with an extended Krylov method applied to the standard Sylvester equation (2.3). As far as we know, the connection we have established between the T-Sylvester equation and the standard Sylvester equation (2.3) is also new. Our experiments show

that the new extended Krylov method we have introduced in Algorithm 2 works very well in many situations and that if it fails then all the other methods also fail. However, if the eigenvalues of $B^{-T}A$ are inside the unit circle then the new non-extended Krylov method introduced in Algorithm 1 is much faster and it is the most recommended method. Similar comments hold if the eigenvalues of $B^{-T}A$ are outside the unit circle and Algorithm 1 is applied to the transposed equation (4.6). The extended Krylov method applied to the standard Sylvester equation (2.3) fails more often than the new methods and it is never faster, therefore, we do not recommended its use. With this work, we solve the last relevant open problem concerning the T-Sylvester equation, namely, its numerical solution in the large-scale setting. The convergence analysis of the proposed methods remains an open problem.

Appendix: Efficient computation of $H_{A,m}$ in Algorithm 2

In this appendix, we include a new method for computing the compressed matrix $H_{A,m}$ in Algorithm 2. This method adapts the recursion in [37, Proposition 3.2] for computing $\underline{K}_m$ to yield directly a recursion for $H_{A,m}$. To this end, we need to introduce some additional notation, which completes the one used in Section 5. First, we partition the upper triangular matrix $Z_m = (Z_{ij})_{1\le i,j\le m} \in \mathbb{R}^{4mr\times 4mr}$ with $Z_{ij} \in \mathbb{R}^{4r\times 4r}$ as $Z_{ij} = [Z_{ij}^{(1)},\ Z_{ij}^{(2)}]$, with $Z_{ij}^{(1)}, Z_{ij}^{(2)} \in \mathbb{R}^{4r\times 2r}$, and define the two block upper triangular matrices

$$Z_m^{(1)} = (Z_{ij}^{(1)})_{1\le i,j\le m},\ \ Z_m^{(2)} = (Z_{ij}^{(2)})_{1\le i,j\le m} \in \mathbb{R}^{4mr\times 2mr}. \tag{8.1}$$

Second, we observe that the orthonormalization process associated with the construction of $\mathcal{V}_m$ generates the block upper Hessenberg matrix

$$\underline{H}_m = (H_{ij}) \in \mathbb{R}^{4(m+1)r\times 4mr}, \qquad \text{where } H_{ij} \in \mathbb{R}^{4r\times 4r}, \tag{8.2}$$

which satisfies

$$[B^{-T}AV_1^{(1)}\ ,\ (B^{-T}A)^{-1}V_1^{(2)}\ ,\ \ldots,\ B^{-T}AV_m^{(1)}\ ,\ (B^{-T}A)^{-1}V_m^{(2)}] = \mathcal{V}_{m+1}\,\underline{H}_m. \tag{8.3}$$

The subdiagonal blocks $H_{j+1,j}$ are upper triangular and we consider them partitioned as follows

$$H_{j+1,j} = \begin{bmatrix} \chi_{11}^{(j)} & \chi_{12}^{(j)} \\ 0 & \chi_{22}^{(j)} \end{bmatrix}, \qquad \text{with } \chi_{11}^{(j)},\ \chi_{12}^{(j)},\ \chi_{22}^{(j)} \in \mathbb{R}^{2r\times 2r}. \tag{8.4}$$

Submatrices $\underline{H}_m^{(1)} = (H_{ij}^{(1)})$ and $\underline{H}_m^{(2)} = (H_{ij}^{(2)})$ of $\underline{H}_m$ analogous to those in (8.1) will also be used.

Proposition 8.1. *Let $H_{A,m} := \mathcal{W}_m^T A\mathcal{V}_m \in \mathbb{R}^{4mr\times 4mr}$ be partitioned in blocks as $H_{A,m} = (\mathtt{h}_{ij})_{1\le i,j\le m}$, where $\mathtt{h}_{ij} \in \mathbb{R}^{4r\times 4r}$. Let each block $\mathtt{h}_{ij}$ be partitioned as $\mathtt{h}_{ij} = \left[\mathtt{h}_{ij}^{(1)},\ \mathtt{h}_{ij}^{(2)}\right]$, where $\mathtt{h}_{ij}^{(1)},\ \mathtt{h}_{ij}^{(2)} \in \mathbb{R}^{4r\times 2r}$, and define the following submatrices of $H_{A,m}$ (separating the odd and even block columns of $H_{A,m}$)*

$$\mathtt{h}^{(1)} = \left(\mathtt{h}_{ij}^{(1)}\right)_{1\le i,j\le m} \in \mathbb{R}^{4mr\times 2mr} \qquad \textit{and} \qquad \mathtt{h}^{(2)} = \left(\mathtt{h}_{ij}^{(2)}\right)_{1\le i,j\le m} \in \mathbb{R}^{4mr\times 2mr}.$$

By using the matrices defined in (5.3)-(5.6)-(8.2)*, the following results hold.*

(a) $\mathtt{h}^{(1)} = Z_{1:m,1:(m+1)}\,\underline{H}_m^{(1)}$.

(b) *With $E_1 = [I_{2r}; 0]^T \in \mathbb{R}^{4mr\times 2r}$, let $w_1 = \mathcal{W}_m E_1$ be the matrix of the first $2r$ columns of $\mathcal{W}_m$, and let*

$$[B^{-T}[C_1, C_2]\,,\ A^{-1}[C_1, C_2]] = V_1 \begin{bmatrix} \chi_{11}^{(0)} & \chi_{12}^{(0)} \\ 0 & \chi_{22}^{(0)} \end{bmatrix}, \qquad \chi_{11}^{(0)},\ \chi_{12}^{(0)},\ \chi_{22}^{(0)} \in \mathbb{R}^{2r\times 2r} \tag{8.5}$$

be the QR decomposition of the matrix in the left-hand side. Then

$$\mathtt{h}^{(2)}_{1:m,1} = \left(E_1\, w_1^T\,[C_1,\, C_2] \;-\; \mathtt{h}^{(1)}_{1:m,1}\,\chi^{(0)}_{12}\right)\left(\chi^{(0)}_{22}\right)^{-1}, \tag{8.6}$$

$$\mathtt{h}^{(2)}_{1:m,k+1} = \left(Z^{(2)}_{1:m,k} \;-\; \mathtt{h}_{1:m,1:k}\,H^{(2)}_{1:k,k} \;-\; \mathtt{h}^{(1)}_{1:m,k+1}\,\chi^{(k)}_{12}\right)\left(\chi^{(k)}_{22}\right)^{-1}, \quad k = 1, \ldots, m-1. \tag{8.7}$$

Proof. To prove (a), let $\mathcal{V}_m^{(1)} := [V_1^{(1)}, \ldots, V_m^{(1)}]$ and notice that (8.3) implies $B^{-T} A\, \mathcal{V}_m^{(1)} = \mathcal{V}_{m+1}\, \underline{H}_m^{(1)}$ and $A\, \mathcal{V}_m^{(1)} = B^T \mathcal{V}_{m+1}\, \underline{H}_m^{(1)}$. Therefore, $\mathtt{h}^{(1)} = \mathcal{W}_m^T A \mathcal{V}_m^{(1)} = \mathcal{W}_m^T B^T \mathcal{V}_{m+1} \underline{H}_m^{(1)} = \mathcal{W}_m^T \mathcal{W}_{m+1} Z_{m+1} \underline{H}_m^{(1)}$, by using (5.6) for $m+1$. So $\mathtt{h}^{(1)} = [I_{4mr}\ 0_{4mr\times 4r}] Z_{m+1} \underline{H}_m^{(1)}$, and the result follows.

Let us prove (b). To obtain (8.6), we start by equating the second block columns in (8.5) to get $A^{-1}[C_1, C_2] = V_1^{(1)}\chi_{12}^{(0)} + V_1^{(2)}\chi_{22}^{(0)}$, which implies $AV_1^{(2)} = \left([C_1, C_2] - AV_1^{(1)}\chi_{12}^{(0)}\right)\left(\chi_{22}^{(0)}\right)^{-1}$. Therefore

$$(\mathtt{h}^{(2)})_{1:m,1} = \mathcal{W}_m^T A V_1^{(2)} = (\mathcal{W}_m^T [C_1, C_2] - \mathcal{W}_m^T A V_1^{(1)} \chi_{12}^{(0)})\,(\chi_{22}^{(0)})^{-1}.$$

Equation (8.6) follows from observing that $\mathcal{W}_m^T A V_1^{(1)} = (\mathtt{h}^{(1)})_{1:m,1}$ and that, from (5.6), $\mathrm{range}(w_1) = B^T \mathrm{range}(V_1^{(1)}) = \mathrm{range}([C_1, C_2])$. Next, we focus on (8.7). From (8.3), we obtain

$$(B^{-T}A)^{-1} V_k^{(2)} = \mathcal{V}_{m+1} H^{(2)}_{1:(m+1),k} = \mathcal{V}_k H^{(2)}_{1:k,k} + V_{k+1} H^{(2)}_{k+1,k} = \mathcal{V}_k H^{(2)}_{1:k,k} + [V_{k+1}^{(1)}\ V_{k+1}^{(2)}] \begin{bmatrix} \chi_{12}^{(k)} \\ \chi_{22}^{(k)} \end{bmatrix},$$

where (8.4) has been used. This equation implies, after some manipulations,

$$\begin{aligned} A\, V_{k+1}^{(2)} &= (B^T\, V_k^{(2)} - A\, \mathcal{V}_k H^{(2)}_{1:k,k} - A\, V_{k+1}^{(1)}\, \chi_{12}^{(k)})\,(\chi_{22}^{(k)})^{-1} \\ &= (\mathcal{W}_m\, Z^{(2)}_{1:m,k} - A\, \mathcal{V}_k H^{(2)}_{1:k,k} - A\, V_{k+1}^{(1)}\, \chi_{12}^{(k)})\,(\chi_{22}^{(k)})^{-1}, \end{aligned} \tag{8.8}$$

where we used again (5.6). Equation (8.7) follows from combining (8.8) with $\mathtt{h}^{(2)}_{1:m,k+1} = \mathcal{W}_m^T A V_{k+1}^{(2)}$, and the definition of $H_{A,m}$. □

References

[1] A. C. Antoulas. *Approximation of Large-scale Dynamical Systems*, volume 6 of *Advances in Design and Control*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2005.

[2] R. H. Bartels and G. W. Stewart. Algorithm 432: The solution of the matrix equation $AX - BX = C$. *Communications of the ACM*, 8:820–826, 1972.

[3] M. Baumann and U. Helmke. Singular value decomposition of time-varying matrices. *Future Generation Computer Systems*, 19(3):353–361, 2003.

[4] B. Beckermann. An error analysis for rational Galerkin projection applied to the Sylvester equation. *SIAM J. Numer. Anal.*, 49(6):2430–2450, 2011.

[5] P. Benner, J.-R. Li, and T. Penzl. Numerical solution of large-scale Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems. *Numer. Linear Algebra Appl.*, 15(9):755–777, 2008.

[6] P. Benner, R.-C. Li, and N. Truhar. On the ADI method for Sylvester equations. *J. Comput. Appl. Math.*, 233(4):1035–1045, 2009.

[7] H. W. Braden. The equations $A^T X \pm X^T A = B$. *SIAM J. Matrix Anal. Appl.*, 20(2):295–302 (electronic), 1999.

[8] R. Byers and D. Kressner. Structured condition numbers for invariant subspaces. *SIAM J. Matrix Anal. Appl.*, 28(2):326–347, 2006.

[9] F. Chatelin. Simultaneous Newton's iteration for the eigenproblem. In *Defect correction methods (Oberwolfach, 1983)*, volume 5 of *Comput. Suppl.*, pages 67–74. Springer, Vienna, 1984.

[10] C.-Y. Chiang. A note on the T-Stein matrix equation. *Abstr. Appl. Anal.*, pages Art. ID 824641, 8, 2013.

[11] C.-Y. Chiang, E. K.-W. Chu, and W.-W. Lin. On the $\bigstar$-Sylvester equation $AX \pm X^{\bigstar} B^{\bigstar} = C$. *Appl. Math. Comput.*, 218(17):8393–8407, 2012.

[12] D. Chu, W.-W. Lin, and R. C. E. Tan. A numerical method for a generalized algebraic Riccati equation. *SIAM J. Control Optim.*, 45(4):1222–1250, 2006.

[13] F. De Terán and F. M. Dopico. Consistency and efficient solution of the Sylvester equation for $\star$-congruence. *Electron. J. Linear Algebra*, 22:849–863, 2011.

[14] F. De Terán and F. M. Dopico. The solution of the equation $XA+AX^T = 0$ and its application to the theory of orbits. *Linear Algebra Appl.*, 434(1):44–67, 2011.

[15] F. De Terán, F. M. Dopico, N. Guillery, D. Montealegre, and N. Reyes. The solution of the equation $AX + X^{\star}B = 0$. *Linear Algebra Appl.*, 438(7):2817–2860, 2013.

[16] J. W. Demmel. Three methods for refining estimates of invariant subspaces. *Computing*, 38:43–57, 1987.

[17] A. El Guennouni, K. Jbilou, and A. J. Riquet. Block Krylov subspace methods for solving large Sylvester equations. *Numer. Algorithms*, 29(1-3):75–96, 2002. Matrix iterative analysis and biorthogonality (Luminy, 2000).

[18] S. R. Garcia and A. L. Shoemaker. On the matrix equation $XA + AX^T = 0$. *Linear Algebra Appl.*, 438(6):2740–2746, 2013.

[19] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[20] L. Grasedyck. Existence of a low rank or $\mathcal{H}$-matrix approximant to the solution of a Sylvester equation. *Numer. Linear Algebra Appl.*, 11(4):371–389, 2004.

[21] N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2002.

[22] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 1985.

[23] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, Cambridge, 1994. Corrected reprint of the 1991 original.

[24] D. Y. Hu and L. Reichel. Krylov-subspace methods for the Sylvester equation. *Linear Algebra Appl.*, 172:283–313, 1992. Second NIU Conference on Linear Algebra, Numerical Linear Algebra and Applications (DeKalb, IL, 1991).

[25] Kh. D. Ikramov. On conditions for the unique solvability of the matrix equation $AX + X^T B = C$. *Dokl. Akad. Nauk*, 430(4):444–447, 2010.

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

[27] A. Kawamoto and T. Katayama. The semi-stabilizing solution of generalized algebraic Riccati equation for descriptor systems. *Automatica J. IFAC*, 38(10):1651–1662, 2002.

[28] A. Kawamoto, K. Takaba, and T. Katayama. On the generalized algebraic Riccati equation for continuous-time descriptor systems. *Linear Algebra Appl.*, 296(1-3):1–14, 1999.

[29] D. Kressner, C. Schröder, and D. S. Watkins. Implicit QR algorithms for palindromic and even eigenvalue problems. *Numer. Algorithms*, 51(2):209–238, 2009.

[30] T. Penzl. A cyclic low-rank Smith method for large sparse Lyapunov equations. *SIAM J. Sci. Comput.*, 21(4):1401–1418, 1999.

[31] T. Penzl. Eigenvalue decay bounds for solutions of Lyapunov equations: the symmetric case. *Systems Control Lett.*, 40(2):139–144, 2000.

[32] Y. Saad. Numerical solution of large Lyapunov equations. In *Signal processing, scattering and operator theory, and numerical methods (Amsterdam, 1989)*, volume 5 of *Progr. Systems Control Theory*, pages 503–511. Birkhäuser Boston, Boston, MA, 1990.

[33] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.

[34] Y. Saad. *Numerical Methods for Large Eigenvalue Problems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, revised edition, 2011.

[35] V. Simoncini. Computational methods for linear matrix equations. Submitted (available at `http://www.dm.unibo.it/ simoncin/list.html`).

[36] V. Simoncini. On the numerical solution of $AX - XB = C$. *BIT*, 36(4):814–830, 1996.

[37] V. Simoncini. A new iterative method for solving large-scale Lyapunov matrix equations. *SIAM J. Sci. Comput.*, 29(3):1268–1288, 2007.

[38] O. Taussky and H. Wielandt. On the matrix function $AX + X'A'$. *Arch. Rational Mech. Anal.*, 9:93–96, 1962.

[39] Yu. O. Vorontsov and Kh. D. Ikramov. A numerical algorithm for solving the matrix equation $AX + X^{\mathrm{T}}B = C$. *Zh. Vychisl. Mat. Mat. Fiz.*, 51(5):739–747, 2011.

[40] H.-S. Wang, C.-F. Yung, and F.-R. Chang. Bounded real lemma and H_∞ control for descriptor systems. *Proceedings of the American Control Conference, Albuquerque, New Mexico, June 1997*, pages 2115–2119, 1997.

[41] L. Wang, M. T. Chu, and Y. Bo. A computational framework of gradient flows for general linear matrix equations. *Numer. Algorithms*, 68(1):121–141, 2015.

[42] H. K. Wimmer. Roth's theorems for matrix equations with symmetry constraints. *Linear Algebra Appl.*, 199:357–362, 1994.

[43] Y. Yuan and H. Dai. The direct updating of damping and gyroscopic matrices. *J. Comput. Appl. Math.*, 231(1):255–261, 2009.