Numerical Algorithms (2020) 84:1459–1479
https://doi.org/10.1007/s11075-020-00896-8

ORIGINAL PAPER

Extended nonsymmetric global Lanczos method for matrix function approximation

A. H. Bentbib[1] · M. El Ghomari[1] · K. Jbilou[2,3]

Received: 18 July 2019 / Accepted: 23 January 2020 / Published online: 4 March 2020

Abstract
Extended Krylov subspace methods are attractive methods for computing approximations of matrix functions and other problems producing large-scale matrices. In this work, we propose the extended nonsymmetric global Lanczos method for solving some matrix approximation problems. The derived algorithm uses short recursive relations to generate bi-orthonormal bases, with respect to the Frobenius inner product, of the corresponding extended Krylov subspaces $K_m^e(A, V)$ and $K_m^e(A^T, W)$. Here, A is a large nonsymmetric matrix; V and $W \in \mathbb{R}^{n\times s}$ are two blocks. New algebraic properties of the proposed method are developed and applications to approximation of both $W^T f(A)V$ and $\mathrm{trace}(W^T f(A)V)$ are given. Numerical examples are presented to show the performance of the extended nonsymmetric global Lanczos for these problems.

Keywords Extended Krylov subspace · Extended moment matching · Laurent polynomial · Nonsymmetric global Lanczos method · Matrix function

Mathematics Subject Classification (2010) 15A16 · 15A63 · 65F60

✉ M. El Ghomari
m.elghomari10@gmail.com

A. H. Bentbib
a.bentbib@uca.ac.ma

K. Jbilou
khalide.jbilou@univ-littoral.fr

1 Laboratory LAMAI, Faculty of Sciences and Technologies, Cadi Ayyad University, Marrakech, Morocco

2 Laboratory LMPA, University ULCO, 50 Rue F. Buisson, Calais cedex, France

3 Laboratory CSEHS, University UM6P, Ben Guérir, Morocco

1 Introduction

Let $A \in \mathbb{R}^{n\times n}$ be a large nonsymmetric and nonsingular matrix, and let V and $W \in \mathbb{R}^{n\times s}$ be two block vectors with $1 < s \ll n$. We are interested in approximating expressions of the form

$$\mathcal{I}_{tr} := \text{trace}(W^T f(A)V) \quad \text{and} \quad \mathcal{I} := W^T f(A)V, \tag{1}$$

where f is a function such that $f(A)$ is well defined. Evaluation of expressions (1) arises in various applications such as in network analysis ($f(t) = \exp(t)$ or $f(t) = t^3$) [10], machine learning ($f(t) = \log(t)$) [13, 20], theory of Quantum Chromodynamics ($f(t) = t^{1/2}$), electronic structure computation [3, 7, 22], and the solution of ill-posed problems [11, 14]. The matrix function $f(A)$ can be defined by the spectral factorization of A; see, e.g., [14, 16] for discussions on several possible definitions of matrix functions. In the present paper, we assume that the matrix A is so large that it is impractical to evaluate its spectral factorization.

Approximation of $\mathcal{I}_{tr}$ in the case when A is a symmetric matrix and $W = V$ and using global methods is well studied in [4]. Global Krylov subspace techniques were first proposed in [17] for solving linear equations with multiple right hand sides and Lyapunov equations. Consider the global Krylov subspace

$$\mathbb{K}_m^{gl}(A, V) = \text{span}\{V, AV, \ldots, A^{m-1}V\} = \{p(A)V \ : \ p \in \Pi_{m-1}\}, \tag{2}$$

where Π_{m-1} denotes the set of polynomials of degree at most $m-1$. Using m steps of the global Lanczos method [18] to A with the initial block vector V leads to the decomposition

$$A\mathbb{V}_m = \mathbb{V}_m(T_m \otimes I_s) + \beta_m V_{m+1} E_m^T, \tag{3}$$

where $\otimes$ stands for the Kronecker product. The block columns $V_1, \ldots, V_m$ of the matrix $\mathbb{V}_m = [V_1, \ldots, V_m] \in \mathbb{R}^{n\times ms}$ form an F-orthonormal (with respect to the Frobenius inner product) basis of the global Krylov subspace (2) which means that

$$\langle V_j, V_k\rangle_F := \text{trace}(V_k^T V_j) = \begin{cases} 1 & j = k, \\ 0 & j \neq k. \end{cases}$$

Moreover, the matrix $T_m \in \mathbb{R}^{m\times m}$ in (3) is symmetric and tridiagonal, $I_s \in \mathbb{R}^{s\times s}$ denotes the identity matrix, $\beta_m \geq 0$ and $E_m \in \mathbb{R}^{ms\times s}$ is the mth block axis vector, i.e., E_m corresponds to the $(m-1)s+1, \ldots, ms$ columns of the identity matrix I_{ms}.

As for the classical Krylov subspace methods and in the symmetric case, it is attractive to approximate the expression $\mathcal{I}_{tr}$ in (1) by

$$\mathcal{G}_m(f) := \|V\|_F^2 \widetilde{e}_1^T f(T_m)\widetilde{e}_1, \tag{4}$$

where $\widetilde{e}_1$ is the first vector of the canonical basis of $\mathbb{R}^m$. One can show that the approximation (4) is exact for polynomials of degree at most $2m-1$, i.e.,

$$\mathcal{I}_{tr}(p) = \mathcal{G}_m(p), \quad \forall p \in \Pi_{2m-1},$$

see, Bellalij et al. [4] for more details. When A is nonsymmetric, the F-orthonormal basis $\mathbb{V}_m$ of $\mathbb{K}_m^{gl}(A, V)$ cannot be generated with short recurrence relations. The nonsymmetric global Lanczos method [18] generates a pair of F-biorthonormal bases $\mathbb{V}_m = [V_1, \ldots, V_m]$ and $\mathbb{W}_m = [W_1, \ldots, W_m]$ with short recurrence relations, for the Krylov subspaces $\mathbb{K}_m(A, V)$ and $\mathbb{K}_m(A^T, W)$, respectively. Application of m steps of global Lanczos nonsymmetric algorithm yields the following relations:

$$\begin{aligned} A\mathbb{V}_m &= \mathbb{V}_m(T_m^b \otimes I_s) + \beta_m V_{m+1} E_m^T, \\ A^T\mathbb{W}_m &= \mathbb{W}_m(T_m^{b,T} \otimes I_s) + \gamma_m W_{m+1} E_m^T. \end{aligned}$$

The block vectors V_is and W_is of $\mathbb{V}_m = [V_1, \ldots, V_m]$ and $\mathbb{W}_m = [W_1, \ldots, W_m]$ elements $\mathbb{R}^{n\times s}$ are said to be F-biorthonormal, if

$$\langle V_j, W_k\rangle_F := \text{trace}(W_k^T V_j) = \begin{cases} 1 & j = k, \\ 0 & j \neq k. \end{cases}$$

The matrix $T_m^b \in \mathbb{R}^{m\times m}$ is nonsymmetric and tridiagonal. The coefficients $\beta_m, \gamma_m \in \mathbb{R}$, and E_m is the matrix defined in (3). The expression $\mathcal{I}_{tr}$ in (1) can be approximated by

$$\mathcal{G}_m^b(f) := \langle V, W\rangle_F \widetilde{e}_1^T f(T_m^b)\widetilde{e}_1.$$

In the same way as in [12], one can show that the approximation $\mathcal{G}_m^b$ is exact for polynomials of degree at most $2m - 1$, i.e.,

$$\mathcal{I}_{tr}(p) = \mathcal{G}_m^b(p) \quad \forall p \in \Pi_{2m-1}.$$

Approximation of $W^T f(A)V$ is well studied in the literature; see, e.g., [12, 21] and the references therein. For the nonsymmetric matrices, Reichel et al. [21], approximate $\mathcal{I}$ by performing m steps with the nonsymmetric block Lanczos method, applied to A with initial block vectors W and V and this leads to the following algebraic relations:

$$\begin{aligned} A[V_1, \ldots, V_m] &= [V_1, \ldots, V_m]J_m + V_{m+1}\Gamma_m E_m^T, \\ A^T[W_1, \ldots, W_m] &= [W_1, \ldots, W_m]J_m^T + W_{m+1}\Delta_m E_m^T, \end{aligned}$$

where $W_j, V_j \in \mathbb{R}^{n\times s}$ are bi-orthonormal, i.e.,

$$W_k^T V_j := \begin{cases} I_s & j = k, \\ O_s & j \neq k. \end{cases}$$

O_s denotes the zero matrix of order s. The matrix $J_m \in \mathbb{R}^{ms\times ms}$ is a block nonsymmetric tridiagonal matrix. The size of the block entries is $s \times s$. $\Gamma_m, \Delta_m \in \mathbb{R}^{s\times s}$, and E_m are the matrix defined in (3). In [21], it was shown that $\mathcal{G}_m^{\text{block}}(f)$ defined by

$$\mathcal{G}_m^{\text{block}}(f) := \widetilde{E}_1^T f(J_m)\widetilde{E}_1,$$

where $\widetilde{E}_1 \in \mathbb{R}^{m\times s}$ corresponds to the first s columns of the identity matrix I_m, can be used to approximate $\mathcal{I}$. Also $\mathcal{G}_m^{\text{block}}$ is exact for polynomials of degree at most $2m - 1$, i.e.,

$$\mathcal{I}(p) = \mathcal{G}_m^{\text{block}}(p), \quad \forall\, p \in \Pi_{2m-1}.$$

In the context of approximating expressions of the form $f(A)b$ for some vector $b \in \mathbb{R}^n$, Druskin et al. [9, 19] computed approximation using the standard extended Krylov subspace generated by the vectors $A^{-m}b, \ldots, A^{-1}b, b, Ab, \ldots, A^{m-1}b$.

We are interested in exploring approximations of (1) by using a pair of F-biorthonormal bases from the two extended global Krylov subspaces

$$\begin{aligned} \mathbb{K}_m^e(A, V) &= \text{span}\{V, A^{-1}V, AV, \ldots, A^{m-1}V, A^{-m}V\}, \\ \mathbb{K}_m^e(A^T, W) &= \text{span}\{W, A^{-T}W, A^TW, \ldots, A^{m-1,T}W, A^{-m,T}W\}. \end{aligned} \tag{5}$$

This paper is organized as follows. Section 2 introduces the extended nonsymmetric global Lanczos process, for generating a pair of F-biorthonormal bases for the two extended global Krylov subspaces defined by (5). Section 3 describes the application of this process to the approximation of the two matrix functions given in (1) and give some properties. Section 4 presents some numerical experiments that illustrate the quality of the computed approximations.

2 Some algebraic properties of the extended nonsymmetric global Lanczos process

2.1 Preliminaries and notations

We begin by recalling some notations and definitions that will be used throughout this paper. The Kronecker product satisfies the following properties:

1. $(A \otimes B)(C \otimes D) = AC \otimes BD$.
2. $(A \otimes B)^T = A^T \otimes B^T$.

Definition 1 Partition the matrices $M = [M_1, \ldots, M_p] \in \mathbb{R}^{n\times ps}$ and $N = [N_1, \ldots, N_l] \in \mathbb{R}^{n\times ls}$ into block columns $M_i, N_j \in \mathbb{R}^{n\times s}$, and define the $\diamond$-product of the matrices M and N as

$$M^T \diamond N = [\text{trace}(M_i^T N_j)]_{i=1,\ldots,p}^{j=1,\ldots,l} \in \mathbb{R}^{p\times l}. \tag{6}$$

The following proposition gives some properties satisfied by the above product.

Proposition 1 [5, 6] *Let $A, B, C \in \mathbb{R}^{n\times ps}$, $D \in \mathbb{R}^{n\times n}$, $L \in \mathbb{R}^{p\times p}$, and $\alpha \in \mathbb{R}^{n\times n}$. Then, we have*

1. $(A + B)^T \diamond C = A^T \diamond C + B^T \diamond C$
2. $A^T \diamond (B + C) = A^T \diamond B + A^T \diamond C$
3. $(\alpha A)^T \diamond C = \alpha(A^T \diamond C)$
4. $(A^T \diamond B)^T = B^T \diamond A$
5. $(DA)^T \diamond B = A^T \diamond (D^T B)$
6. $A^T \diamond (B(L \otimes I_s)) = (A^T \diamond B)L$

Definition 2 Let X be an $n \times s$ matrix. The vectorization of the matrix X, denoted vec(X), is the $ns \times 1$ column vector obtained by stacking the columns of the matrix X, i.e.,

$$\text{vec}(X) = [X_{1,1}, \ldots, X_{n,1}, X_{1,2}, \ldots, X_{n,2}, \ldots, X_{1,s}, \ldots, X_{n,s}]^T.$$

vec satisfies the following properties:

1. $\text{vec}(MXN) = (N^T \otimes M)\text{vec}(X)$, $\forall A \in \mathbb{R}^{p\times q}$, $N \in \mathbb{R}^{r\times s}$, $X \in \mathbb{R}^{q\times r}$.
2. $\text{vec}(A)^T \text{vec}(B) = \text{trace}(A^T B)$, $\forall A, B \in \mathbb{R}^{n\times n}$.

Let $M = [M_1, \ldots, M_m]$ and $N = [N_1, \ldots, N_m] \in \mathbb{R}^{n\times ms}$, with $M_i, N_i \in \mathbb{R}^{n\times s}$. The following algorithm applied to the pair (M, N) allows us to obtain two F-biorthonormal matrices $\mathbb{V}, \mathbb{W} \in \mathbb{R}^{n\times ms}$.

Algorithm 1 The global bi-orthonormalization decomposition.

Input: $M = [M_1, \ldots, M_m]$ and $N = [N_1, \ldots, N_m] \in \mathbb{R}^{n\times ms}$.
for $j = 1$ **to** m
  $\widehat{V}_j = M_j$; $\widehat{W}_j = N_j$;
  for $i = 1$ **to** $j-1$
    $h_{i,j} = \text{trace}(W_i^T \widehat{V}_j)$;
    $g_{i,j} = \text{trace}(V_i^T \widehat{W}_j)$;
    $\widehat{V}_j = \widehat{V}_j - h_{i,j} V_i$;
    $\widehat{W}_j = \widehat{W}_j - g_{i,j} W_i$;
  end for
  $h_{j,j} = \sqrt{\left|\text{trace}(\widehat{W}_j^T \widehat{V}_j)\right|}$; $g_{j,j} = \text{trace}(\widehat{W}_j^T \widehat{V}_j)/h_{j,j}$;
  $V_j = \widehat{V}_j / h_{j,j}$; $W_j = \widehat{W}_j / g_{j,j}$;
end for
Output: F-biorthonormal matrices $\mathbb{V}, \mathbb{W} \in \mathbb{R}^{n\times ms}$, and $H = [h_{i,j}], G = [g_{i,j}]$ are two $m \times m$ upper triangular matrices.

Proposition 2 *Let M and N be the matrices defined above and let $\mathbb{V}$ and $\mathbb{W} \in \mathbb{R}^{n\times ms}$ be the corresponding matrices obtained from Algorithm 1. Then we have the following decompositions:*

$$M = \mathbb{V}(H \otimes I_s), \quad N = \mathbb{W}(G \otimes I_s),$$

where $H = [h_{i,j}]$ and $G = [g_{i,j}]$ are the two $m \times m$ upper triangular matrices determined by Algorithm 1 and satisfying

$$H = \mathbb{W}^T \diamond M \quad \text{and} \quad G = \mathbb{V}^T \diamond N.$$

Proof From Algorithm 1, the block vectors V_j and W_j are written as follows:

$$M_j = \sum_{i=1}^{j} h_{i,j} V_i, \text{ and } N_j = \sum_{i=1}^{j} g_{i,j} W_i,$$

and because $h_{i,j}$ and $g_{i,j}$ vanish for $i > j$, we obtain

$$M_j = \sum_{i=1}^{m} V_i (h_{i,j} \otimes I_s) \text{ and } N_j = \sum_{i=1}^{m} W_i (g_{i,j} \otimes I_s).$$

Then

$$M = \mathbb{V}(H \otimes I_s) \text{ and } N = \mathbb{W}(G \otimes I_s).$$

Using the properties of $\diamond$-product, it follows that

$$\begin{aligned} H &= (\mathbb{W}^T \diamond \mathbb{V})H = \mathbb{W}^T \diamond (\mathbb{V}(H \otimes I_s)) = \mathbb{W}^T \diamond M. \\ G &= (\mathbb{V}^T \diamond \mathbb{W})G = \mathbb{V}^T \diamond (\mathbb{W}(G \otimes I_s)) = \mathbb{V}^T \diamond N. \end{aligned}$$

□

2.2 Description of the process

In this section, we recall the standard extended nonsymmetric Lanczos method given in [25]. Proofs in this section are formulated from results given in [25]. Let v and w be two vectors in $\mathbb{R}^n$. The standard extended nonsymmetric Lanczos algorithm applied to the pairs (A, v) and (A^T, w) determines two bi-orthonormal basis $\mathbb{V}_{2m+2} = [v_1, \ldots, v_{2m+2}]$ and $\mathbb{W}_{2m+2} = [w_1, \ldots, w_{2m+2}]$ of dimensions $n \times 2m$, with $v_i, w_i \in \mathbb{R}^n$ (see [25, Algorithm 3]). Then the odd vectors v_{2j+1}, w_{2j+1} verify the following equations:

$$\begin{aligned} \alpha = \sqrt{|w^T v|}, v_1 &= v/\alpha, \beta = w^T v/\alpha, w_1 = w/\beta, \\ h_{2j+1,2j-1} v_{2j+1} &= A v_{2j-1} - \sum_{i=2j-3}^{2j} h_{i,2j-1} v_i = \widehat{v}_{2j+1}, \\ g_{2j+1,2j-1} w_{2j+1} &= A^T w_{2j-1} - \sum_{i=2j-3}^{2j} g_{i,2j-1} w_i = \widehat{w}_{2j+1}, \end{aligned} \qquad j = 1, \ldots, m. \quad (7)$$

where

$$h_{i,2j-1} = w_i^T A v_{2j-1} \text{ and } g_{i,2j-1} = v_i^T A^T w_{2j-1}, \text{ for } i = 1, \ldots, 2j.$$

The $h_{2j+1,2j-1}, g_{2j+1,2j-1}$ are such that $w_{2j+1}^T v_{2j+1} = 1$. Hence,

$$h_{2j+1,2j-1} = \sqrt{|\widehat{w}_{2j+1}^T \widehat{v}_{2j+1}|}, g_{2j+1,2j-1} = (\widehat{w}_{2j+1}^T \widehat{v}_{2j+1})/h_{2j+1,2j-1}.$$

Similarly, the even vectors v_{2j+2}, w_{2j+2} are computed by the following relations:

$$\begin{aligned} \widehat{v}_2 &= A^{-1} v - (w_1^T A^{-1} v) v_1, \ \widehat{w}_2 = A^{-T} w - (v_1^T A^{-T} w) w_1 \\ \gamma &= \sqrt{|\widehat{w}_2^T \widehat{v}_2|}, v_2 = \widehat{v}_2/\gamma, \quad \Gamma = \widehat{w}_2^T \widehat{v}_2/\gamma, w_2 = \widehat{w}_2/\Gamma, \end{aligned}$$

$$\begin{aligned} h_{2j+2,2j} v_{2j+2} &= A^{-1} v_{2j} - \sum_{i=2j-2}^{2j+1} h_{i,2j} v_i = \widehat{v}_{2j+2}, \\ g_{2j+2,2j} w_{2j+2} &= A^{-T} w_{2j} - \sum_{i=2j-2}^{2j+1} g_{i,2j} w_i = \widehat{w}_{2j+2}, \end{aligned} \qquad j = 1, \ldots, m. \quad (8)$$

where

$$h_{i,2j} = w_i^T A^{-1} v_{2j} \text{ and } g_{i,2j} = v_i^T A^{-T} w_{2j}, \text{ for } i = 1, \ldots, 2j+1.$$

The $h_{2j+2,2j}, g_{2j+2,2j}$ are such that $w_{2j+2}^T v_{2j+2} = 1$. Hence,

$$h_{2j+2,2j} = \sqrt{|\widehat{w}_{2j+2}^T \widehat{v}_{2j+2}|}, g_{2j+2,2j} = \widehat{w}_{2j+2}^T \widehat{v}_{2j+2}/h_{2j+2,2j}.$$

In the case where V and W are two blocks of size $n \times s$, the implementation of the extended nonsymmetric global Lanczos algorithm applied to (A, V) and (A^T, W) will be similar to the standard algorithm [25, Algorithm 3] except that the standard inner product will be replaced by the Frobenius inner product $\langle \cdot, \cdot \rangle_F$. This algorithm provides two F-biorthonormal bases $\mathbb{V}_{2m+2} = [v_1, \ldots, v_{2m+2}]$ and $\mathbb{W}_{2m+2} = [w_1, \ldots, w_{2m+2}]$. The dimension of these bases is $n \times 2(m+1)s$, where $v_i, w_i \in \mathbb{R}^{n \times s}$ are the ith block vector of $\mathbb{V}_{2m+2}$ and $\mathbb{W}_{2m+2}$, respectively. In addition, the block vectors v_i and w_i satisfy the relations (7) and (8). Following the idea in [24], the column vectors of the bases $\mathbb{V}_{2m+2}$ and $\mathbb{W}_{2m+2}$ can be computed two by two blocks as follows. Introduce

$$V_j = [v_{2j-1}\,,\, v_{2j}], \quad W_j = [w_{2j-1}\,,\, w_{2j}],$$

and

$$H_{i,j} = \begin{bmatrix} h_{2i-1,2j-1} & h_{2i-1,2j} \\ h_{2i,2j-1} & h_{2i,2j} \end{bmatrix}, \quad G_{i,j} = \begin{bmatrix} g_{2i-1,2j-1} & g_{2i-1,2j} \\ g_{2i,2j-1} & g_{2i,2j} \end{bmatrix}, \tag{9}$$

where V_j and W_j are the jth $n \times 2s$ block vectors of the matrices $\mathbb{V}_{2m+2} = [V_1, \ldots, V_{m+1}]$ and $\mathbb{W}_{2m+2} = [W_1, \ldots, W_{m+1}]$, respectively; $H_{i,j}$ and $G_{i,j}$ are 2×2 matrices. The block vectors V_j and W_j are such that

$$[V\,,\, A^{-1}V] = V_1(H_0 \otimes I_s), \quad [W\,,\, A^{-T}W] = W_1(G_0 \otimes I_s),$$

where H_0 and $G_0 \in \mathbb{R}^{2\times 2}$ are computed by applying Algorithm 1 to $[V, A^{-1}V]$ and $[W, A^{-T}W] \in \mathbb{R}^{n\times 2s}$. And

$$V_{j+1}(H_{j+1,j} \otimes I_s) = [Av_{2j-1}, A^{-1}v_{2j}] - V_{j-1}(H_{j-1,j} \otimes I_s) - V_j(H_{j,j} \otimes I_s),$$
$$W_{j+1}(G_{j+1,j} \otimes I_s) = [A^T w_{2j-1}, A^{-T} w_{2j}] - W_{j-1}(G_{j-1,j} \otimes I_s) - W_j(G_{j,j} \otimes I_s).$$

Set

$$H_0 = \begin{pmatrix} \alpha_{1,1} & \alpha_{1,2} \\ 0 & \alpha_{2,2}, \end{pmatrix} \quad G_0 = \begin{pmatrix} \beta_{1,1} & \beta_{1,2} \\ 0 & \beta_{2,2} \end{pmatrix}. \tag{10}$$

As the $n \times 2s$ blocks $V_1, \ldots, V_m$ and $W_1, \ldots, W_m$ are F-biorthonormal (with respect to the Frobenius-product) and by using the properties of the $\diamond$-product, it follows that

$$H_{i,j} = W_i^T \diamond [Av_{2j-1}\,,\, A^{-1}v_{2j}] \text{ and } G_{i,j} = V_i^T \diamond [A^T w_{2j-1}\,,\, A^{-T} w_{2j}].$$

The extended global nonsymmetric Lanczos is summarized in the following algorithm (Algorithm 2).

Proposition 3 *Let $v_1, v_2, \ldots, v_{2m+2}$ and $w_1, w_2, \ldots, w_{2m+2}$ be the F-biorthon ormal block vectors determined by Algorithm 2. Then*

$$\begin{aligned} v_{2j-1} &= p_{j-1}^V(A)V + q_{j-1}^V(A^{-1})V, \\ v_{2j} &= r_{j-1}^V(A^{-1})A^{-1}V + s_{j-1}^V(A)V, \end{aligned}$$

where $deg(p_{j-1}) = deg(r_{j-1}) = j-1$, $deg(q_{j-1}) \leq j-1$ and $deg(s_{j-1}) \leq j-1$. These relations also hold when V and A are replaced by W and A^T, respectively.

Proof The result follows easily from Algorithm 2. □

We notice here that in the previous proposition, the block vectors $[V_1, \ldots, V_{m+1}]$ and $[W_1, \ldots, W_{m+1}]$ form a F-biorthonormal basis of $\mathbb{K}_{m+1}^e(A, V)$ and $\mathbb{K}_{m+1}^e(A^T, W)$, respectively. When inverting A or solving linear systems via the LU decomposition is not easy, then we can compute matrix products of the form $A^{-1}V$ and $A^{-T}W$ by using an efficient iterative linear solver such as the well-known GMRES method.

Algorithm 2 The extended nonsymmetric global Lanczos algorithm (ENGL)

Input: Matrix A, initial $n \times s$ blocks V and W.
Set $\widehat{V}_1 = [V, A^{-1}V]$ and $\widehat{W}_1 = [W, A^{-T}W]$.
Apply Algorithm 1 onto $\widehat{V}_1, \widehat{W}_1$ to obtain V_1, W_1.
Set $\alpha = \sqrt{|\langle V, W\rangle_F|}$ and $\beta = \langle V, W\rangle_F/\alpha$.
for $j = 1$ **to** m
  Set $V_j^{(1)}$: first s columns of V_j; and $V_j^{(2)}$: second s columns of V_j;
  And $W_j^{(1)}$: first s columns of W_j; and $W_j^{(2)}$: second s columns of W_j;
  $\widehat{V}_{j+1} = [AV_j^{(1)}, A^{-1}V_j^{(2)}]$;
  $\widehat{W}_{j+1} = [A^T W_j^{(1)}, A^{-T}W_j^{(2)}]$;
  for $i = j-1$ **to** j
    $H_{i,j} = W_i^T \diamond \widehat{V}_{j+1}$;
    $G_{i,j} = V_i^T \diamond \widehat{W}_{j+1}$;
    $\widehat{V}_{j+1} = \widehat{V}_{j+1} - V_i(H_{i,j} \otimes I_s)$;
    $\widehat{W}_{j+1} = \widehat{W}_{j+1} - W_i(G_{i,j} \otimes I_s)$;
  end for
  Apply Algorithm 1 to $\widehat{V}_{j+1}, \widehat{W}_{j+1}$ to compute V_{j+1}, W_{j+1}, $H_{j+1,j}$ and $G_{j+1,j}$.
end for
Output: F-biorthonormal basis $\mathbb{V}_{2m+2}$ and $\mathbb{W}_{2m+2}$.

We now introduce the $2m \times 2m$ matrices T_{2m} and S_{2m}, given by

$$T_{2m} = \mathbb{W}_{2m}^T \diamond A\mathbb{V}_{2m}, \quad S_{2m} = \mathbb{W}_{2m}^T \diamond A^{-1}\mathbb{V}_{2m}, \tag{11}$$

where $\mathbb{V}_{2m} = [V_1, \ldots, V_m]$ and $\mathbb{W}_{2m} = [W_1, \ldots, W_m]$.
An important result concerning the connection between the proposed method and the standard extended nonsymmetric Lanczos method is given by the following theorem.

Theorem 1 *Let V, W be the initial given block vectors of $\mathbb{R}^{n\times s}$. Set $\mathcal{A} = I_s \otimes A$, $v = vec(V)$, and $w = vec(W) \in \mathbb{R}^{ns}$ and perform m steps of the standard extended nonsymmetric Lanczos applied to the pairs $(\mathcal{A}, v)$ and $(\mathcal{A}^T, w)$. Then we obtain the same F- biorthonormal basis $\mathbb{V}_{2m+2}$, $\mathbb{W}_{2m+2}$ as the one determined by the extended nonsymmetric global Lanczos algorithm. Moreover, the basis $\mathbb{V}_{2m+2}$, $\mathbb{W}_{2m+2}$ satisfy the following relations:*

$$A\mathbb{V}_{2m} = \mathbb{V}_{2m}(T_{2m} \otimes I_s) + v_{2m+1}\left(\begin{bmatrix} t_{2m+1,2m-1}, & t_{2m+1,2m}\end{bmatrix} E_m^T \otimes I_s\right), \tag{12}$$

$$A^{-1}\mathbb{V}_{2m} = \mathbb{V}_{2m}(S_{2m} \otimes I_s) + V_{m+1}\left(\begin{bmatrix} 0 & s_{2m+1,2m} \\ 0 & s_{2m+2,2m}\end{bmatrix} E_m^T \otimes I_s\right),$$

$$A^T\mathbb{W}_{2m} = \mathbb{W}_{2m}(T_{2m}^T \otimes I_s) + w_{2m+1}\left(\begin{bmatrix} \widetilde{t}_{2m+1,2m-1}, & \widetilde{t}_{2m+1,2m}\end{bmatrix} E_m^T \otimes I_s\right), \tag{13}$$

$$A^{-T}\mathbb{W}_{2m} = \mathbb{W}_{2m}(S_{2m}^T \otimes I_s) + W_{m+1}\left(\begin{bmatrix} 0 & \widetilde{s}_{2m+1,2m} \\ 0 & \widetilde{s}_{2m+2,2m}\end{bmatrix} E_m^T \otimes I_s\right),$$

where $[\widetilde{t}_{2m+1,2m-1}, \widetilde{t}_{2m+1,2m}] = [\langle A^T w_{2m-1}, v_{2m+1}\rangle_F, \langle A^T w_{2m}, v_{2m+1}\rangle_F]$, and $[\widetilde{s}_{2m+1,2m}, \widetilde{s}_{2m+2,2m}] = [\langle A^{-T} w_{2m}, v_{2m+1}\rangle_F, \langle A^{-T} w_{2m}, v_{2m+2}\rangle_F]$.

$E_m = [e_{2m-1}, e_{2m}] \in \mathbb{R}^{2m\times 2}$ corresponds to the last two columns of the identity matrix I_{2m}.

Proof Perform m steps of the extended Lanczos algorithm applied to the pairs $(\mathcal{A}, v)$ and $(\mathcal{A}^T, w)$ yield two $ns \times (2m+2)$ bi-orthonormal bases $\mathcal{V}_{2m+2} = [x_1, \ldots, x_{2m+2}]$, $\mathcal{W}_{2m+2} = [y_1, \ldots, y_{2m+2}]$. These bases satisfy the relations (7) and (8) for the matrix $\mathcal{A}$. By setting, $X_j, Y_j \in \mathbb{R}^{n\times s}$ such that $x_j = \text{vec}(X_j)$ and $y_j = vec(Y_j)$, we obtain

$$\text{trace}(Y_i^T X_j) = vec(Y_i)^T \text{vec}(X_j) = y_i^T x_j = \delta_{i,j},$$

which means that X_j, Y_j are F-biorthonormal. In addition, it follows that $X_j = V_j$ and $Y_j = W_j$ since

$$\begin{aligned} h_{i,2j-1} &= y_i^T \mathcal{A} x_{2j} = \text{vec}(Y_i)^T (I_s \otimes A)\text{vec}(X_{2j}). \\ &= \text{vec}(Y_i)^T \text{vec}(AX_{2j}) = \langle AX_{2j-1}, Y_i\rangle_F. \\ g_{i,2j-1} &= v_i^T \mathcal{A}^T w_{2j-1} = \langle A^T Y_{2j-1}, X_i\rangle_F. \\ h_{i,2j} &= w_i^T \mathcal{A}^{-1} v_{2j} = \langle A^{-1} X_{2j}, Y_i\rangle_F. \\ g_{i,2j} &= v_i^T \mathcal{A}^{-T} w_{2j} = \langle A^{-T} Y_{2j}, X_i\rangle_F. \end{aligned}$$

Using the same techniques as above, we obtain that $T_{2m} = \mathcal{W}_{2m}^T \mathcal{A}\mathcal{V}_{2m}$ and $S_{2m} = \mathcal{W}_{2m}^T \mathcal{A}^{-1}\mathcal{V}_{2m}$. Let us prove the relation (12). Using [25, Lemma 3.1], it holds that

$$\mathcal{A}\mathcal{V}_{2m} = \mathcal{V}_{2m}\mathcal{T}_{2m} + x_{2m+1}[t_{2m+1,2m-1}, t_{2m+1,2m}]E_m^T,$$

where $\mathcal{T}_{2m} = \mathcal{W}_{2m}^T \mathcal{A}\mathcal{V}_{2m} = T_{2m}$.

In other hand,

$$\begin{aligned} \mathcal{A}\mathcal{V}_{2m} &= (I_s \otimes A)[vec(v_1), \ldots, vec(v_{2m})], \\ &= [vec(Av_1), \ldots, vec(Av_{2m})], \end{aligned}$$

while

$$\begin{aligned} \mathcal{V}_{2m} T_{2m} &= \left[\sum_{i=1}^{2m} [T_{2m} e_1]_i \text{vec}(v_i), \ldots, \sum_{i=1}^{2m} [T_{2m} e_{2m}]_i \text{vec}(v_i) \right] \\ &= [vec(\mathbb{V}_{2m}(T_{2m}e_1 \otimes I_s)), \ldots, vec(\mathbb{V}_{2m}(T_{2m}e_{2m} \otimes I_s))]. \end{aligned}$$

Define $\tau = [t_{2m+1,2m-1}, t_{2m+1,2m}]E_m^T$, then

$$x_{2m+1}\tau = [vec(t_{2m+1,2m-1}v_{2m+1}), vec(t_{2m+1,2m}v_{2m+1})]E_m^T.$$

For $i = 1, \ldots, 2m-2$, it is shown that

$$\text{vec}(Av_i) = \text{vec}(\mathbb{V}_{2m}(T_{2m}e_i \otimes I_s))$$

and

$$\begin{aligned} \text{vec}(Av_{2m-1}) &= \text{vec}(\mathbb{V}_{2m}(T_{2m}e_{2m-1} \otimes I_s)) + \text{vec}(v_{2m+1}(t_{2m+1,2m-1} \otimes I_s)), \\ \text{vec}(Av_{2m}) &= \text{vec}(\mathbb{V}_{2m}(T_{2m}e_{2m} \otimes I_s)) + \text{vec}(v_{2m+1}(t_{2m+1,2m} \otimes I_s)). \end{aligned}$$

Which completes the proof of (12). The proof of the other relations is similar. □

The next two propositions express the entries of T_{2m} and S_{2m} in terms of the recursion coefficients. This will allow us to compute the entries quite efficiently.

Proposition 4 *Let the coefficients $h_{i,j}$ and $\alpha_{i,j}$ as defined in (9) and (10), respectively. The matrix $T_{2m} = [t_{i,j}]$ in (11) is pentadiagonal with the nontrivial entries,*

$$\begin{aligned} t_{i,2j-1} &= h_{i,2j-1} \quad \textit{for } i \in \{2j-3, \dots, 2j+1\}, \\ t_{1,2} &= \frac{1}{\alpha_{2,2}}(\alpha_{1,1} - \alpha_{1,2}h_{1,1}), \\ t_{2,2} &= \frac{-1}{\alpha_{2,2}}\alpha_{1,2}h_{2,1}, \\ t_{3,2} &= \frac{-1}{\alpha_{2,2}}\alpha_{1,2}h_{3,1}. \end{aligned}$$

For $j = 1, 2, \dots, m-1$,

$$\begin{aligned} t_{2j+1,2j+2} &= \frac{-1}{h_{2j+2,2j}} t_{2j+1,2j-1:2j+1} h_{2j-1:2j+1,2j}, \\ t_{2j+2,2j+2} &= \frac{-1}{h_{2j+2,2j}} t_{2j+2,2j+1} h_{2j+1,2j}, \\ t_{2j+3,2j+2} &= \frac{-1}{h_{2j+2,2j}} t_{2j+3,2j+1} h_{2j+1,2j}. \end{aligned}$$

Proof The proof is similar to the one given [25, Lemma 3.3]. □

Proposition 5 *Let the coefficients $h_{i,j}$ as defined in (9). The matrix $S_{2m} = [s_{i,j}]$ in (11) is also pentadiagonal with the nontrivial entries, for $j = 1, \dots, m-1$,*

$$\begin{aligned} s_{i,2j} &= h_{i,2j} \quad \textit{for } i \in \{2j-2, \dots, 2j+2\}, \\ s_{2j,2j+1} &= \frac{-1}{h_{2j+1,2j-1}} s_{2j,2j-2:2j+1} h_{2j-2:2j+1,2j-1}, \\ s_{2j+1,2j+1} &= \frac{-1}{h_{2j+1,2j-1}} [s_{2j+1,2j} h_{2j,2j-1} + s_{2j+1,2j-1} h_{2j+1,2j-1}], \\ s_{2j+2,2j+1} &= \frac{-1}{h_{2j+1,2j-1}} [s_{2j+2,2j} h_{2j,2j-1} + s_{2j+2,2j+1} h_{2j+1,2j+1}]. \end{aligned}$$

Proof The proof is analogous to the proof of Proposition 4. □

The following result relates positive powers of S_{2m} to negative powers of T_{2m}.

Lemma 1 *Let T_{2m} and S_{2m} as given by (11) and let $e_1 = [1, 0, \dots, 0]^T \in \mathbb{R}^{2m}$. Then*

$$S_{2m}^{j} e_1 = T_{2m}^{-j} e_1 \quad \textit{for} \ \ j = 1, 2, \dots, m.$$

Lemma 2 *Let the matrices T_{2m} and S_{2m} as defined by (11). Let $\mathbb{V}_{2m} = [v_1, v_2, \ldots, v_{2m}]$ be the matrix given in (12). Then for $j = 1 \ldots m-1$, we have*

$$A^j v_1 = \mathbb{V}_{2m}(T_{2m}^j e_1 \otimes I_s), \quad j = 0, 1, \ldots, m-1, \tag{14}$$

$$A^{-j} v_1 = \mathbb{V}_{2m}(S_{2m}^j e_1 \otimes I_s), \quad j = 0, 1, \ldots, m, \tag{15}$$

$$A^{-j} v_1 = \mathbb{V}_{2m}(T_{2m}^{-j} e_1 \otimes I_s), \quad j = 0, 1, \ldots, m. \tag{16}$$

Proof It was shown in [25] that when performing m steps of standard extended nonsymmetric Lanczos method to the pairs $(\mathcal{A}, v)$ and $(\mathcal{A}^T, w)$, it holds that

$$\mathcal{A}^j x_1 = \mathcal{V}_{2m}\mathcal{T}_{2m} e_1, \quad j = 0, \ldots, m-1,$$

by using the properties of the $\otimes$-product and the "vec" operation, we obtain

$$\mathcal{A}^j x_1 = \text{vec}(A^j v_1) \text{ and } \mathcal{V}_{2m}\mathcal{T}_{2m} e_1 = \sum_{i=1}^{2m} [T_{2m}^j e_1]_i \text{vec}(v_i) = \text{vec}(\mathbb{V}_{2m}(T_{2m}^j e_1 \otimes I_s)),$$

which prove (14). Using the same techniques as above, we can prove (15). Finally, (16) follows from (15) and Lemma 1. □

Lemma 3 *Let the matrices T_{2m} and S_{2m} as defined by (11). Let $\mathbb{W}_{2m} = [w_1, w_2, \ldots, w_{2m}]$ be the matrix given in (13). Then for $j = 1 \ldots m-1$, we have*

$$A^{j,T} w_1 = \mathbb{W}_{2m}(T_{2m}^{j,T} e_1 \otimes I_s), \quad j = 0, 1, \ldots, m-1,$$

$$A^{-j,T} w_1 = \mathbb{W}_{2m}(S_{2m}^{j,T} e_1 \otimes I_s), \quad j = 0, 1, \ldots, m,$$

$$A^{-j,T} w_1 = \mathbb{W}_{2m}(T_{2m}^{-j,T} e_1 \otimes I_s), \quad j = 0, 1, \ldots, m.$$

Proof The proof is similar to the ones given in Lemma 2. □

3 Application to the approximation of matrix functions

3.1 Computing the matrix function trace($W^T f(A) V$)

The expression $\mathcal{I}_{tr}$ defined in (1) can be approximated by

$$\mathcal{G}_{2m}^e(f) := \langle W, V \rangle_F e_1^T f(T_{2m}) e_1 \tag{17}$$

Lemma 4 *Let the matrices T_{2m} and S_{2m} as defined by (11). Let v_1 and w_1 be the initial block vectors computed by Algorithm 2. Then the following equalities hold*

$$trace(w_1^T A^j v_1) = e_1^T T_{2m}^j e_1 \quad for\ j = 0, 1, \ldots, 2m-1, \tag{18}$$

$$trace(w_1^T A^{-j} v_1) = e_1^T T_{2m}^{-j} e_1 \quad for\ j = 0, 1, \ldots, 2m.$$

Proof We have $\mathrm{trace}(w_1^T A^j v_1) = \mathrm{vec}(w_1)^T \mathrm{vec}(A^j v_1)$. From Theorem 1, we get $y_1 = \mathrm{vec}(w_1)$ and $x_1 = \mathrm{vec}(v_1)$. By using the properties of the "vec" operation, it follows that $\mathrm{trace}(w_1^T A^j v_1) = y_1^T \mathcal{A}^j x_1$. Then using the result given in [25, Lemma 4.1], we obtain $y_1^T \mathcal{A}^j x_1 = e_1^T \mathcal{T}_{2m} e_1$. This shows (18) since, $\mathcal{T}_{2m} = T_{2m}$. The second equation can be established by using the same techniques as above. □

Theorem 2 *After m steps of Algorithm 2 with initial block vectors $V, W \in \mathbb{R}^{n\times s}$, we get*

$$trace(W^T p(A) V) = \langle V, W\rangle_F \, e_1^T p(T_{2m}) e_1 \quad \forall p \in \Delta_{-2m,2m-1},$$

where

$$\Delta_{-2m,2m-1} = span\{x^{-2m}, \dots, x^{-1}, 1, x, \dots, x^{2m-1}\}.$$

Proof From Algorithm 2, V and W are collinear with V_1 and W_1, respectively, i.e., $V = \alpha V_1$ and $W = \beta W_1$ with $\alpha\beta = \langle V, W\rangle_F$. Which implies that $\mathrm{trace}(W^T p(A) V) = \langle V, W\rangle_F \mathrm{trace}(W_1^T p(A) V_1)$. Using the results of Lemma 4, we get $\mathrm{trace}(W_1^T p(A) V_1) = e_1^T p(T_{2m}) e_1$. This completes the proof. □

Algorithm 3 Approximation of $\mathrm{trace}(W^T f(A) V)$ by the extended global Lanczos method.

1: **Input:** Matrix $A \in \mathbb{R}^{n\times n}$, initial block vectors $V, W \in \mathbb{R}^{n\times s}$, and function f
2: Compute $\mathbb{V}_{2m}$ and $\mathbb{W}_{2m}$ by Algorithm 2.
3: Compute matrix $T_{2m} = \mathbb{W}_{2m}^T \diamond A\mathbb{V}_{2m}$ using Proposition 4.
4: Compute $\mathcal{G}_{2m}^e(f) = \langle V, W\rangle_F e_1^T f(T_{2m}) e_1$ given by (17).
5: **Output:** Approximation $\mathcal{G}_{2m}^e(f)$ of $\mathrm{trace}(W^T f(A) V)$

3.2 Computing the matrix function $W^T f(A) V$

The aim of this subsection is to show how to use the extended nonsymmetric global Lanczos algorithm to approximate $\mathcal{I}$ in (1) (see Algorithm 2). Before describing the application of the proposed method, we notice that the extended nonsymmetric block Lanczos method (ENBL) given in [2] can also be used to approximate $\mathcal{I}$. After m steps of this algorithm applied to the pairs (A, V) and (A^T, W), we obtain two $n \times 2ms$ bi-orthonormal bases $\mathscr{V}_{2m}$ and $\mathscr{W}_{2m}$, i.e., $\mathscr{W}_{2m}^T \mathscr{V}_{2m} = I_{2ms}$. Then the expression $\mathcal{I}$ can be approximated as follows: $W^T(\mathscr{V}_{2m} f(\mathscr{T}_{2m}) \mathscr{W}_{2m}^T) V$ where $\mathscr{T}_{2m}$ is $2ms \times 2ms$ block tridiagonal matrix with $2s \times 2s$ blocks. The matrix $\mathscr{T}_{2m} = \mathscr{W}_{2m}^T A \mathscr{V}_{2m}$ is computed recursively without requiring additional matrix-vector products with A (see [2, Proposition 3]). The ENBL algorithm is expensive as the number m of iterations increases and also for large values of s.

Now, let us come back to our proposed method. Applying Algorithm 2 to the pairs (A, V) and (A^T, W) allows us to obtain two F-biorthonormal bases $\mathbb{V}_{2m}$ and $\mathbb{W}_{2m}$

such that $V = \alpha v_1$ and $W = \beta w_1$. It is clear that $\mathbb{W}_{2m}^{+}\mathbb{V}_{2m} = I_{2ms}$; and then, we can consider the projector $\mathcal{P}_{2m}$ defined as

$$\begin{array}{rcl}\mathcal{P}_{2m} : \mathbb{R}^{n\times s} & \longrightarrow & \mathbb{K}_m^e(A, V) \\ X & \longmapsto & \mathbb{V}_{2m}\mathbb{W}_{2m}^{+}X.\end{array}$$

Applying the projector $\mathcal{P}_{2m}$ to $\mathcal{I}$ gives the reduced matrix function

$$\mathcal{B}_{2m}^{e}(f) = C_{2m}^{T} f(A_{2m})B_{2m}, \tag{19}$$

where $A_{2m} = \mathbb{W}_{2m}^{+}A\mathbb{V}_{2m} \in \mathbb{R}^{2ms\times 2ms}$, $B_{2m} = \mathbb{W}_{2m}^{+}V \in \mathbb{R}^{2ms\times s}$ and $C_{2m} = \mathbb{V}_{2m}^{T}W \in \mathbb{R}^{2ms\times s}$.

The next proposition will allow us to compute A_{2m} and B_{2m} from the recursion matrix T_{2m} without requiring the computation of the $\mathbb{W}_{2m}^{+}$ and $A\mathbb{V}_{2m}$.

Proposition 6 *Let V, W be the initial block vectors where $V = \alpha V_1$. Then the matrices A_{2m} and B_{2m} defined above are computed as follows:*

$$\begin{aligned} A_{2m} &= (T_{2m}\otimes I_s) + \mathbb{W}_{2m}^{+}v_{2m+1}(\left[\, t_{2m+1,2m-1},\ t_{2m+1,2m}\,\right] E_m^T \otimes I_s), \\ B_{2m} &= \alpha\, \mathscr{E}_1, \end{aligned}$$

where $\left[\, t_{2m+1,2m-1},\ t_{2m+1,2m}\,\right]$ is defined by (12), and $\mathscr{E}_1$ corresponds to the first s columns of the identity matrix I_n.

Proof Let $E_j = [e_{2j-1}, e_{2j}] \in \mathbb{R}^{2m\times 2}$, for $j = 1, \ldots, m$. Multiplying the matrix A_{2m} from the right by $E_j \otimes I_s$ gives

$$A_{2m}(E_j\otimes I_s) = \mathbb{W}_{2m}^{+}A\mathbb{V}_{2m}(E_j\otimes I_s).$$

Using (12) , we get

$$A_{2m}(E_j\otimes I_s) = (T_{2m}E_j\otimes I_s) + \mathbb{W}_{2m}^{+}v_{2m+1}\left(\left[\, t_{2m+1,2m-1},\ t_{2m+1,2m}\,\right] E_m^T E_j \otimes I_s\right).$$

It follows that $j = 1, \ldots, m-1$,

$$A_{2m}(E_j\otimes I_s) = T_{2m}E_j\otimes I_s,$$

while for $j = m$, it results that

$$A_{2m}(E_m\otimes I_s) = (T_{2m}E_m\otimes I_s) + \mathbb{W}_{2m}^{+}v_{2m+1}\left(\left[\, t_{2m+1,2m-1},\ t_{2m+1,2m}\,\right]\otimes I_s\right).$$

Therefore,

$$A_{2m} = (T_{2m}\otimes I_s) + \mathbb{W}_{2m}^{+}v_{2m+1}\left(\left[\, t_{2m+1,2m-1},\ t_{2m+1,2m}\,\right] E_m^T\otimes I_s\right).$$

To show the expression of B_{2m}, we use the fact that $V = \alpha\, v_1$, which implies that

$$\begin{aligned} B_{2m} &= \mathbb{W}_{2m}^{+} V = \alpha\, \mathbb{W}_{2m}^{+} v_1 \\ &= \alpha\, \mathbb{W}_{2m}^{+} \mathbb{V}_{2m} \mathscr{E}_1 = \alpha\, \mathscr{E}_1 . \end{aligned}$$

□

Lemma 5 *Let A_{2m} be the matrix be defined in (19). Let $\mathbb{V}_{2m} = [v_1, \ldots, v_{2m}]$ defined by (12). Then*

$$\begin{aligned} A^j v_1 &= \mathbb{V}_{2m} A_{2m}^{j} \mathcal{E}_1 \quad \text{for } j = 0, \ldots, m-1, \\ A^{-j} v_1 &= \mathbb{V}_{2m} A_{2m}^{-j} \mathcal{E}_1 \quad \text{for } j = 0, \ldots, m. \end{aligned} \tag{20}$$

Proof The first equation is shown by induction. Av_1 and v_1 are two elements of $\mathbb{K}_m^e(A, v_1)$ and then

$$Av_1 = \mathbb{V}_{2m} \mathbb{W}_{2m}^{+} A v_1 = \mathbb{V}_{2m} \mathbb{W}_{2m}^{+} A \mathbb{V}_{2m} \mathbb{W}_{2m}^{+} v_1 = \mathbb{V}_{2m} A_{2m} \mathcal{E}_1 .$$

The equality is true for $j = 1$. Let $j = 2, \ldots, m-1$, and assume that

$$A^k v_1 = \mathbb{V}_{2m} A_{2m}^{k} \mathcal{E}_1 \quad k = 1, \ldots, j-1.$$

Since $A^j v_1 \in \mathbb{K}_m^e(A, v_1)$, it follows that $A^j v_1 = \mathbb{V}_{2m} \mathbb{W}_{2m}^{+} A^j v_1$. By induction, we have

$$A^j v_1 = \mathbb{V}_{2m} \mathbb{W}_{2m}^{+} A \mathbb{V}_{2m} A_{2m}^{j-1} \mathcal{E}_1 ,$$

which completes the proof of (20).

The second equation follows from the fact that $\mathbb{V}_{2m} \mathbb{W}_{2m}^{+} A^{-j} v_1 = A^{-j} v_1 \ \forall\, j = 1, \ldots, m$ and by using same technique induction as above. □

Lemma 6 *Let A_{2m} be the matrix defined by (19) and let $\mathbb{W}_{2m} = [w_1, \ldots, w_{2m}]$ be the matrix in (13). Then*

$$\begin{aligned} A^{j,T} w_1 &= \mathbb{W}_{2m}^{+,T} A_{2m}^{j,T} \mathbb{V}_{2m}^{T} w_1 \quad \text{for } j = 0, \ldots, m-1, \\ A^{-j,T} w_1 &= \mathbb{W}_{2m}^{+,T} A_{2m}^{-j,T} \mathbb{V}_{2m}^{T} w_1 \quad \text{for } j = 0, \ldots, m. \end{aligned}$$

Proof We have $A^{j,T} w_1 \in \mathbb{K}_m^e(A^T, W)$, which means that $\mathbb{W}_{2m}^{+,T} \mathbb{V}_{2m}^{T} A^{j,T} w_1 = A^{j,T} w_1$ for any integer j such that $-m \le j \le m-1$. According to the previous equality, and by the same techniques as the proof of Lemma 5, then both equations are shown. □

Proposition 7 *After m steps of the process, we have*

$$\begin{aligned} W^T A^{-j} V &= C_{2m}^{T} A_{2m}^{-j} B_{2m}, \quad \text{for } j = 0, \ldots, 2m, \\ W^T A^{j} V &= C_{2m}^{T} A_{2m}^{j} B_{2m}, \quad \text{for } j = 0, \ldots, 2m-1. \end{aligned}$$

Proof The proof is based on an application of results of Lemmas 5 and 6. Let $j = j_1 + j_2$ with $j_1, j_2 \in \{0, \ldots, m\}$. We have $V = \alpha v_1$ and $W = \beta w_1$, then

$$\begin{aligned} W^T A^{-j} V &= \alpha\beta w_1^T A^{-j_1} A^{-j_2} v_1 \\ &= \alpha\beta (A^{-j_1,T} w_1)^T (A^{-j_2} v_1) \end{aligned}$$

Using equations of Lemmas 5 and 6, we obtain

$$\begin{aligned} &= \alpha\beta (\mathbb{W}_{2m}^{+,T} A_{2m}^{-j_1,T} \mathbb{V}_{2m}^T w_1)^T (\mathbb{V}_{2m} A_{2m}^{-j_2} \mathcal{E}_1) \\ &= W^T \mathbb{V}_{2m} A_{2m}^{-j} \mathbb{W}_{2m}^{+} V \\ &= C_{2m}^T A_{2m}^{-j} B_{2m}, \end{aligned}$$

which completes the proof of the first relation. The proof of the second relation is similar. □

Now, we compare the operations requirements for the extended block Lanczos method ENBL and the extended global Lanczos method ENGL. Both methods require the same cost of computing matrix-matrix products AV for some $V \in \mathbb{R}^{n\times s}$; they also require the same cost of solving linear systems. The ENBL method requires $4ns^2$ operations to compute the $n \times 2s$ matrix VH and $4ns^2$ for computing the $2s \times 2s$ matrix $W^T V$, while the ENGL method only needs $4ns$ operations to compute $W^T \diamond V$ and $4ns$ to compute $V(H \otimes I_s)$. To update the new bases V_{j+1} and W_{j+1}, the ENBL method has to perform the bi-orthonormalization decomposition of $\widehat{V}_{j+1}$ and $\widehat{W}_{j+1}$ that costs $16ns^2$ in every step, while the global bi-orthonormalization decomposition costs only $16ns$. To compute $\mathscr{T}_{2m}$ using [2, Proposition 3], we need to $2s(4ns + n + 1)(m + 1)$ operations, while for ENGL method, computing T_{2m} costs only $(10ns + 2)(m + 1)$. Moreover, the computation of $\mathscr{T}_{2m}$ requires solving linear systems of size $s \times s$ for every step, while ENGL method only needs the division by a scalar (see Proposition 4). In Table 1, we summarize the number of operations after m iterations of the ENBL algorithm and the ENGL algorithm. As we observed, the ENGL algorithm is less expensive than the ENBL algorithm, when the number of iteration m increases and the block size s is not small.

Algorithm 4 Approximation of $W^T f(A)V$ by the extended global Lanczos method.

1: **Input:** Matrix $A \in \mathbb{R}^{n\times n}$, initial block vectors $V, W \in \mathbb{R}^{n\times s}$, and a function f
2: Compute $\mathbb{V}_{2m}$, $\mathbb{W}_{2m}$, v_{2m+1} and α by Algorithm 2.
3: Compute $T_{2m} = \mathbb{W}_{2m}^T \diamond A\mathbb{V}_{2m}$ using Proposition 4.
4: $U_{2ms}\Sigma_{2ms}Z_{2ms}^T = \mathbb{W}_{2m}^T \mathbb{V}_{2m}$ a singular value decomposition of $\mathbb{W}_{2m}^T \mathbb{V}_{2m}$.
5: $r = \text{rank}(\Sigma_{2ms})$.
6: Set $U_r = [u_1, \ldots, u_r] \in \mathbb{R}^{2ms\times r}$, $S_r = diag(\Sigma(1,1), \ldots, \Sigma(r,r)) \in \mathbb{R}^{r\times r}$, and $Z_r = [z_1, \ldots, z_r] \in \mathbb{R}^{2ms\times r}$.
7: $A_{2m} = (T_{2m} \otimes I_s) + Z_r[S_r^{-1}(U_r^T \mathbb{W}_{2m}^T v_{2m+1})](\widetilde{T}_{m+1,m}\mathcal{E}_m^T \otimes I_s)$.
8: $B_{2m} = \alpha\mathscr{E}_1$ and $C_{2m} = \mathbb{V}_{2m}^T W$.
9: Compute $\mathcal{B}_{2m}^e(f) = C_{2m}^T f(A_{2m})B_{2m}$.
10: **Output:** Approximation $\mathcal{B}_{2m}^{e,L}(f)$ of $W^T f(A)V$.

Table 1 Comparison of the extended nonsymmetric block Lanczos (ENBL) and the extended nonsymmetric global Lanczos (ENGL) algorithms

Flops	ENBL	ENGL
Matrix matrix multiplication $n \times n$ with $n \times 2s$ matrices AV and A^TW	$4n^2ms$ Line 6	$4n^2ms$ Line 8, 9
Solution of linear systems $A^{-1}V$ and $A^{-T}W$	$4n^2(m+1)s$ Line 3, 6	$4n^2(m+1)s$ Line 2, 8, 9
Matrix matrix multiplication, resp. the product diamond of $2s \times n$ and $n \times 2s$ matrices W^TV resp. $W^T \diamond V$	$16nms^2$ Line 7	$16nms$ Line 11, 12
Matrix matrix multiplication $n \times 2s$ with $2s \times 2s$ matrices resp. the Kroncker product VH resp. $V(H \otimes I_s)$	$16nms^2$ Line 8	$16nms$ Line 13, 14
Bi-orthogonalization decomp- osition $n \times 2s$ of with $n \times 2s$ matrices resp. global bi-ortho- gonalization decomposition	$16n(m+1)s^2$ Line 3,9	$16n(m+1)s$ Line 3, 16
Computation of $T^b_{2m} = \mathbb{W}^T_{2m}A\mathbb{V}_{2m}$ resp. $T_{2m} = \mathbb{W}^T_{2m} \diamond A\mathbb{V}_{2m}$	$2s(4ns+n+1)$ $(m+1)$ [2, Proposition 3]	$(10ns+2)$ $(m+1)$ Proposition 4
Computation of $A_{2m} = \mathbb{W}^+_{2m}A\mathbb{V}_{2m}$		$4(ms)^2(n+1)+2ms^2n$ Proposition 6

4 Numerical experiments

In this section, we give some numerical examples to show the performance of the extended nonsymmetric global Lanczos (ENGL) method . In the selected examples, the proposed method is applied to the approximation of expressions of the form $\mathcal{I}_{tr}$ and $\mathcal{I}$ given in (1). All experiments were carried out in MATLAB $R2015a$ on a computer with an Intel Core i-3 processor and 3.89 GBytes of RAM. The computations were done with about 15 significant decimal digits.

As mentioned in the second section, we did not have to compute explicitly A^{-1}. In all examples, the matrix products with A^{-1} and A^{-T} in lines $2-8-9$ of Algorithm 4 are computed via an LU factorization or by using an iterative solver. We used a preconditioned block biconjugate gradient (PBBiCG) method as described in Algorithm 5.

Algorithm 5 Preconditioned block biconjugate gradient algorithm (PBBiCG).

Input: Matrix $A \in \mathbb{R}^{n\times n}$, initial block vectors $B, C \in \mathbb{R}^{n\times s}$, and inital solutions X_0, X_0^*
Compute $R_0 = B - AX_0$ and $R_0^* = C - A^T X_0^*$;
Orthogonalize $R_0 = Q_1\mathcal{R}_1$ and $R_0^* = Q_2\mathcal{R}_2$;
Set $R_0 = M^{-1}Q_1$, $P_0 = R_0$; and $P_0^* = R_0^*$;
for $k = 0, 1, \ldots,$ until convergence **do**
  $\Lambda_k = (R_k^{*,T} M^{-1} A P_k)^{-1}(R^{*T} R_k)$;
  $\Lambda_k^* = (R_k^T (M^{-1}A)^T P_k^*)^{-1}(R^{*T} R_k)^T$;
  $X_{k+1} = X_k + P_k \Lambda_k$;
  $X_{k+1}^* = X_k^* + P_k^* \Lambda_k^*$;
  $R_{k+1} = R_k - M^{-1}AP_k\Lambda_k$;
  $R_{k+1}^* = R_k^* - (M^{-1}A)^T P_k^* \Lambda_k^*$;
  if $\max\{\|R_{k+1}\|, \|R_{k+1}^*\|\} < \epsilon$ **then**
    Break ;
  end if
  $\beta_{k+1} = (R_k^{*T} R_k)^{-1}(R_{k+1}^{*T} R_{k+1})$;
  $\beta_{k+1}^* = (R_k^{*T} R_k)^{-T}(R_{k+1}^{*T} R_{k+1})^T$;
  $P_{k+1} = R_{k+1} + P_k \beta_{k+1}$;
  $P_{k+1}^* = R_{k+1}^* + P_k^* \beta_{k+1}^*$;
  $k = k + 1$;
end for
$\mathcal{X} = X_0 + X_{k+1}\mathcal{R}_1$;
$\mathcal{X}^* = M^{-T}[X_0^* + X_{k+1}^*\mathcal{R}_2]$;
Output: $\mathcal{X}$ and $\mathcal{X}^*$;

4.1 Examples for approximations of trace($W^T f(A) V$)

Example 1 In this experiment, we compared the performance of the ENGL algorithm with the performance of extended global Arnoldi algorithm (EGA) described in [1, 15]. We approximate trace($E_1^T \exp(A) E_1$) where A is the nonsymmetric adjacency matrix pesa of order $n = 11738$. This matrix was obtained from the Suite Sparse Matrix Collection[8]. $E_1 \in \mathbb{R}^{n\times s}$ corresponds to the first s columns of identity matrix I_n. Results for several choices of the block size s and number of iterations m are reported in Table 2. We notice that in the ENGL and EGA algorithms, we used the PBBiCG algorithm defined by Algorithm 5 with an ILU preconditionner. As observed from this table, the approximate errors determined by ENGL have higher accuracy as compared with the approximations obtained by the EGA method.

Example 2 In this example, we used a diagonalizable matrix of order $n = 1000$ whose eigenvalues are log-uniformly distributed in the interval $[10^{-1}, 10^4]$ and random eigenvectors. We computed approximations of trace($E_1^T f(A) E_1$) given by ENGL and by the standard nonsymmetric global Lanczos method (SNGL). Here we used $s = 6$. In Table 3, we reported the number of iterations, the relative errors, and the required CPU times obtained with different functions. Here we used the LU factorization to compute products of the form $A^{-1}V$ and $A^{-T}W$. The results show that the ENGL algorithm is faster and give better relative errors, while SNGL algorithm is unable to determine an accurate approximation for all functions f used in this example.

Table 2 Example 1: $A \in \mathbb{R}^{n\times n}$ is the nonsymmetric adjacency pesa matrix with $n = 11738$

	Relative error of ENGL	Relative error of EGA
$s = 2$		
$m = 10$	2.2×10^{-4}	0.81
$m = 15$	1.38×10^{-9}	5.3×10^{-4}
$m = 20$	1.63×10^{-14}	3.69×10^{-5}
$m = 25$	1.3×10^{-16}	3.42×10^{-8}
$s = 6$		
$m = 10$	2.84×10^{-4}	0.33
$m = 15$	6.45×10^{-11}	0.07
$m = 20$	2.16×10^{-15}	2.21×10^{-5}
$m = 25$	1.08×10^{-15}	7.31×10^{-9}

4.2 Examples for the approximation of $W^T f(A)V$

In this subsection, we present some results to approximate $W^T f(A)V$ using the ENGL algorithm. In the following experiments, we used the functions: $f(x) = \sqrt{x}$ and $f(x) = x^{-1/3}$. The blocks W and V were generated randomly with entries uniformly distributed on [0, 1]. The matrix A was obtained from the centered finite difference discretization (CFDD) of the elliptic operators given by (21) on the unit square $[0, 1] \times [0, 1]$ with Dirichlet homogeneous boundary conditions. The number of inner grid points in each direction was n_0 and the dimension of matrices is $n = n_0^2$.

$$\begin{aligned} \mathcal{L}_1(u) &= -100u_{xx} - u_{yy} + 10xu_x, \\ \mathcal{L}_2(u) &= -e^{-xy}u_{xx} - e^{xy}u_{yy} + 1/(x+y)u_x. \end{aligned} \tag{21}$$

Example 3 We consider the approximation of $W^T A^{1/2} V$ and $W^T B^{1/2} V$ where the matrices $A, B \in \mathbb{R}^{4900\times 4900}$ are nonsymmetric matrices coming from CFDD of the operators $\mathcal{L}_1(u)$ and $\mathcal{L}_2(u)$, respectively, and given by (21). The block size s was $s = 4$. In Fig. 1, we reported the relative errors of ENBL, ENGL, and SNBL algorithms versus the dimension of the projected subspace using the matrix A on the left and on the right part of this figure, we give the results corresponding to the matrix

Table 3 Example 2: $A \in \mathbb{R}^{n\times n}$ has eigenvalues distributed in the interval $[10^{-1}, 10^4]$ and a random eigenvector matrix. The block size $s = 6$

$f(x)$	ENGL			SNBL		
	Dim	Relative error	Time(s)	*Dim*	Relative error	Time(s)
e^{-x}	59	5.02×10^{-6}	4.59	250	6.84×10^{-5}	26
$\sqrt{x}$	55	1.11×10^{-6}	5.05	250	5.74×10^{-5}	78.5
$x^{-1/4}$	57	1.08×10^{-6}	6.12	250	3.5×10^{-3}	161
$\log(x)$	53	3.33×10^{-7}	6.18	250	8.66×10^{-4}	212

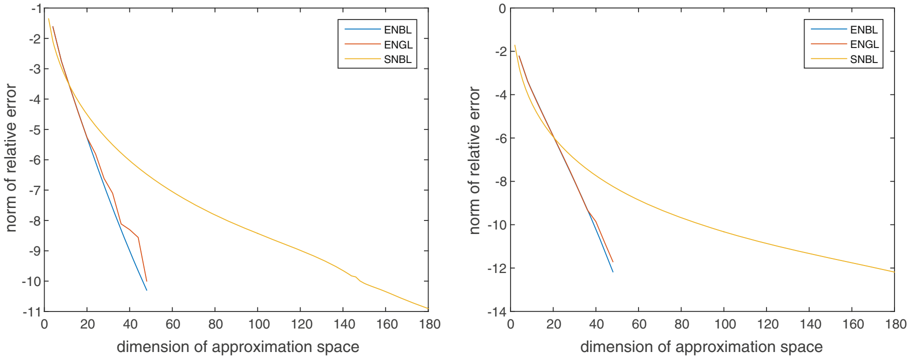

Fig. 1 Approximation of $E_1^T A^{1/2} E_1$(left plot) and $E_1^T B^{1/2} E_1$(right plot)

B. Both plots show that ENGL and ENBL algorithms yield significantly smaller errors than the SNBL algorithm.

Example 4 In this example, we consider nonsymmetric matrices coming from CFDD of the same operators as in Example 3. In Tables 4 and 5, we reported results for the ENGL and ENBL algorithms when approximating $\mathcal{I}$. We used different values of the dimension n ({2500, 4900, 7225, 10000}) and two different block sizes $s = 10, 20$

Table 4 Example 4: Approximation of $W^T f(A) V$ for two functions and different matrix dimensions for the operators given by (21)

Oper.	$f(x)$	n	m	Relative error of ENGL	Relative error of ENBL
$\mathcal{L}_1(x)$	$x^{-1/3}$	2500	15	$6.11 \cdot 10^{-11}$	$1.61 \cdot 10^{-8}$
	$s = 10$	4900	15	$1.88 \cdot 10^{-10}$	$2.48 \cdot 10^{-8}$
		7225	20	$5.57 \cdot 10^{-10}$	$1.37 \cdot 10^{-8}$
		10000	20	$3.44 \cdot 10^{-10}$	$1.7 \cdot 10^{-8}$
	$\sqrt{x}$	2500	15	$1.88 \cdot 10^{-10}$	$2.23 \cdot 10^{-7}$
	$s = 20$	4900	15	$2.48 \cdot 10^{-9}$	$1.99 \cdot 10^{-7}$
		7225	20	$4.27 \cdot 10^{-10}$	$1.83 \cdot 10^{-7}$
		10000	20	$8.47 \cdot 10^{-11}$	$7.78 \cdot 10^{-7}$
$\mathcal{L}_2(x)$	$x^{-1/3}$	2500	15	$5.10 \cdot 10^{-10}$	$5.98 \cdot 10^{-10}$
	$s = 10$	4900	15	$2.91 \cdot 10^{-11}$	$1.11 \cdot 10^{-8}$
		7225	20	$8.51 \cdot 10^{-11}$	$5.59 \cdot 10^{-8}$
		10000	20	$1.87 \cdot 10^{-11}$	$4.50 \cdot 10^{-8}$
	$\sqrt{x}$	2500	15	$2.64 \cdot 10^{-12}$	$2.47 \cdot 10^{-10}$
	$s = 20$	4900	15	$3.45 \cdot 10^{-11}$	$4.17 \cdot 10^{-11}$
		7225	20	$5.98 \cdot 10^{-12}$	$3.26 \cdot 10^{-10}$
		10000	20	$4.18 \cdot 10^{-11}$	$2.15 \cdot 10^{-9}$

Table 5 Example 4: Approximation of $W^T f(A)V$ for two functions and different matrix dimensions for the operators given by (21)

Oper.	$f(x)$	n	m	Relative error of ENGL	Relative error of ENBL
$\mathcal{L}_1(x)$	$x^{-1/3}$	2500	15	$1.67 \cdot 10^{-10}$	$3.39 \cdot 10^{-6}$
	$s = 30$	4900	15	$1.22 \cdot 10^{-10}$	$1.06 \cdot 10^{-6}$
		7225	20	$1.55 \cdot 10^{-10}$	$5.45 \cdot 10^{-8}$
		10000	20	$1.20 \cdot 10^{-10}$	$4.37 \cdot 10^{-6}$
	$\sqrt{x}$	2500	15	$1.50 \cdot 10^{-9}$	$1.45 \cdot 10^{-4}$
	$s = 40$	4900	15	$1.97 \cdot 10^{-10}$	$1.34 \cdot 10^{-4}$
		7225	20	$1.63 \cdot 10^{-10}$	$6.28 \cdot 10^{-4}$
		10000	20	$2.20 \cdot 10^{-10}$	$4.16 \cdot 10^{-4}$
$\mathcal{L}_2(x)$	$x^{-1/3}$	2500	15	$1.28 \cdot 10^{-10}$	$5.30 \cdot 10^{-7}$
	$s = 40$	4900	15	$7.44 \cdot 10^{-11}$	$1.66 \cdot 10^{-8}$
		7225	20	$1.50 \cdot 10^{-11}$	$4.55 \cdot 10^{-6}$
		10000	20	$2.64 \cdot 10^{-11}$	$2.37 \cdot 10^{-6}$
	$\sqrt{x}$	2500	15	$3.10 \cdot 10^{-12}$	$1.50 \cdot 10^{-6}$
	$s = 30$	4900	15	$1.02 \cdot 10^{-11}$	$1.17 \cdot 10^{-6}$
		7225	20	$3.35 \cdot 10^{-12}$	$3.31 \cdot 10^{-7}$
		10000	20	$2.25 \cdot 10^{-9}$	$3.40 \cdot 10^{-5}$

in Table 4, and $s = 30, 40$ in Table 5. For the last two values of n, we used PBBiCG preconditioned by the block ILU preconditioner (see [23]). As shown in Tables 4 and 5, when the block size s increases, the approximations of $\mathcal{I}$ computed with ENGL are more accurate than the approximations produced by the ENBL algorithm.

5 Conclusion

This paper describes an extended nonsymmetric global Lanczos method for the approximation of trace$(W^T f(A)V)$ and $W^T f(A)V$. Two F-biorthonormal bases of the extended Krylov subspaces given by (5) are computed by short recurrence formulas. We gave some suitable algebraic relations. The numerical results show that the nonsymmetric extended global Lanczos method requires fewer iterations and CPU time as compared with the standard nonsymmetric global Lanczos method and to the extended global Arnoldi method when approximating trace$(W^T f(A)V)$ and $W^T f(A)V$.

Acknowledgments We would like to thank the two anonymous referees for their valuable remarks and suggestions allowing us to improve the quality of the paper.

References

1. Abidi, O., Heyouni, M., Jbilou, K.: On some properties of the extended block and global Arnoldi methods with applications to model reduction. Numer. Algorithms **75**, 285–304 (2017)
2. Barkouki, H., Bentbib, A.H., Heyouni, M., Jbilou, K.: An extended nonsymmetric block Lanczos method for model reduction in large scale dynamical systems. Calcolo **55**, 13–36 (2018)
3. Baroni, S., Gebauer, R., Malcioglu, O.B., Saad, Y., Umari, P., Xian, J.: Harnessing Molecular Excited States with Lanczos Chains, vol. 22. Art. Id. 074204 (2010)
4. Bellalij, M., Reichel, L., Rodriguez, G., Sadok, H.: Bounding matrix functionals via partial global block Lanczos decomposition. Appl. Numer. Math. **94**, 127–139 (2015)
5. Bellalij, M., Jbilou, K., Sadok, H.: New convergence results on the global GMRES method for diagonalizable matrices. J. Comput. Appl. Math. **219**, 350–358 (2008)
6. Bouyouli, R., Jbilou, K., Sadaka, R., Sadok, H.: Convergence properties of some block Krylov subspace methods for multiple linear systems. J. Comput. Appl. Math. **196**, 498–511 (2006)
7. Bowler, D.R., Miyazaki, T.: O(n) methods in electronic structure calculations. Rep. Prog. Phys. **75**, 43 (2012). Art. Id. 036503
8. Davis, T., Hu, Y.: The SuiteSparse Matrix Collection, https://sparse.tamu.edu
9. Druskin, V., Knizhnerman, L.: Extended Krylov subspaces: approximation of the matrix square root and related functions. SIAM J. Matrix Anal. Appl. **19**, 755–771 (1998)
10. Estrada, E.: The Structure of Complex Networks: Theory and Applications. Oxford University Press, Oxford (2011)
11. Fenu, C., Reichel, L., Rodriguez, G., Sadok, H.: GCV for Tikhonov regularization by partial SVD. BIT **57**, 1019–1039 (2017)
12. Golub, G.H., Meurant, G.: Matrices, Moments and Quadrature with Applications. Princeton University Press, Princeton (2010)
13. Han, I., Malioutov, D., Shin, J.: Large-scale log-determinant computation through stochastic chebyshev expansions. Proceedings of The 32nd International Conference on Machine Learning **37**, 908–917 (2015)
14. Hansen, P.C.: Rank-deficient and Discrete Ill-Posed Problems. SIAM, Philadelphia (1998)
15. Heyouni, M.: Extended Arnoldi methods for large low-rank Sylvester matrix equations. Appl. Numer. Math. **60**, 1171–1182 (2010)
16. HIGHAM, N.J.: Functions of Matrices: Theory and Computation. SIAM, Philadelphia (2008)
17. Jbilou, K., Sadok, A., Messaoudi, H.: Global FOM and GMRES algorithms for matrix equations. Appl. Numer Math. **31**, 49–63 (1999)
18. Jbilou, K., Sadok, H., Tinzeft, A.: Oblique projection methods for multiple linear systems. Elect. Trans. Num. Anal. **20**, 119–138 (2005)
19. Knizhnerman, VL., Simoncini, A.: New investigation of the extended Krylov subspace method for matrix function evaluations. Numer. Linear Algebra Appl. **17**, 615–638 (2010)
20. Ngo, T.T., Bellalij, M., Saad, Y.: The trace ratio optimization problem. SIAM Rev. **54**, 545–569 (2012)
21. Reichel, L., Rodriguez, G., Tang, T.: New block quadrature rules for the approximation of matrix functions. Linear Algebra Appl. **502**, 299–326 (2016)
22. Saad, Y., Chelikowsky, J., Shontz, S.: Numerical methods for electronic structure calculations of materials. SIAM Rev. **52**, 3–54 (2010)
23. Saad, Y. Iterative Methods for Sparse Linear Systems, 2nd edn. SIAM, Philadelphia (2003)
24. Simoncini, V.: A new iterative method for solving large-scale Lyapunov matrix equations. SIAM J. Sci. Comput. **29**, 1268–1288 (2007)
25. Schweitzer, M.: A two-sided short-recurrence extended Krylov subspace method for nonsymmetric matrices and its relation to rational moment matching. Numer. Algorithms **76**, 1–31 (2016)