MATHEMATICAL MODELING AND COMPUTING, Vol. 8, No. 3, pp. 526–536 (2021)

The extended nonsymmetric block Lanczos methods for solving large-scale differential Lyapunov equations

Sadek L., Talibi Alaoui H.

Department of Mathematics, Faculty of Science, Chouaib Doukkali University, El Jadida, Morocco

(Received 24 December 2020; Accepted 15 August 2021)

In this paper, we present a new approach for solving large-scale differential Lyapunov equations. The proposed approach is based on projection of the initial problem onto an extended block Krylov subspace by using extended nonsymmetric block Lanczos algorithm then, we get a low-dimensional differential Lyapunov matrix equation. The latter differential matrix equation is solved by the Backward Differentiation Formula method (BDF) or Rosenbrock method (ROS), the obtained solution allows to build a low-rank approximate solution of the original problem. Moreover, we also give some theoretical results. The numerical results demonstrate the performance of our approach.

Keywords: *extended block Krylov subspace, extended nonsymmetric block Lanczos algorithm, low-rank approximation, differential Lyapunov equations.*

2010 MSC: 65F50, 65F50, 15A24 **DOI:** 10.23939/mmc2021.03.526

1. Introduction

In this paper, we consider the differential Lyapunov equation (DLE in short) on the time in the interval $[t_0, t_f]$ of the form

$$\begin{cases} X'(t) = AX(t) + X(t)A^T + BB^T, \\ X(t_0) = X_0, \end{cases} \tag{1}$$

where the matrix $A \in \mathbb{R}^{n\times n}$ is assumed to be large, sparse and nonsingular, and $B \in \mathbb{R}^{n\times s}$ is matrix full rank, with $s \ll n$. We assume that the initial condition is given in a factored form as $X_0 = Z_0 Z_0^T$.

Differential Lyapunov matrix equations play a fundamental role in many problems in control, filter design theory, model reduction problems and robust control problems; see, e.g. [1] and the references therein.

The DLE (1) is equivalent to the following linear ordinary differential equation:

$$x'(t) = \mathfrak{A}x(t) + b, \quad x(t_0) = \mathrm{vec}(X_0), \tag{2}$$

where $\mathfrak{A} = I_n \otimes A + A \otimes I_n$, $b = \mathrm{vec}(BB^T)$ and $\mathrm{vec}(X)$ is the vector of $\mathbb{R}^{n\times s}$ defined by $\mathrm{vec}(X) = [X_{11}, X_{21}, \ldots, X_{n1}, \ldots, X_{1s}, X_{2s}, \ldots, X_{ns}]^T \in \mathbb{R}^{ns}$. Reasonable size problems, which is given by (2), is solved by using an integration method. The Kronecker product $A \otimes B = [a_{ij}B]$, where $A = [a_{ij}]$. This product satisfies the properties: $(A \otimes B)(C \otimes D) = (AC \otimes BD)$, $(A \otimes B)^T = A^T \otimes B^T$, and $\mathrm{vec}(AXB) = (B^T \otimes A)\mathrm{vec}(X)$.

The exact solution of the differential Lyapunov equation (1) is given by the following result.

Theorem 1 (Ref. [1]). *The unique solution of the differential Lyapunov equations (1) is defined by*

$$X(t) = e^{(t-t_0)A} X_0 e^{(t-t_0)A^T} + \int_{t_0}^{t} e^{(t-\tau)A} BB^T e^{(t-\tau)A^T} d\tau. \tag{3}$$

There are several methods for solving small or medium-sized differential Lyapunov matrix equations, for example BDF method and ROS method [2–4]. For large problems, we propose a new method based on projection onto extended block Krylov subspaces with an orthogonality Petrov-Galerkin condition; see, e.g. [3, 5–8].

During the last years, there is a large variety of methods to compute the solution of large scale matrix differential equations such as differential Lyapunov equation. For more details see [3,5,6,9]. For large-scale problems, the effective methods are based on Krylov subspaces. Some methods have been proposed for solving large matrix equation, see, e.g. [10]. The main idea employed in these methods is to use an extended Krylov subspace and then apply the Galerkin-type orthogonality condition. The main idea in this work is using the extended nonsymmetric block Lanczos to solve (1).

Behr et al. [11]. They developed a unifying approach based on the spectral theorem for normal operators like the Sylvester operator and derived a formula for its norm using an induced operator norm. In view of numerical approximations, they proposed an algorithm that identifies a suitable Krylov subspace using Taylor series and use a projection to approximate the solution.

Lang et al. [12]. They proposed efficient algorithms for solving large-scale differential Lyapunov equations. They focused on methods, based on standard versions of ordinary differential equations, in the matrix setting. The application of these methods yields algebraic Lyapunov equations with a certain structure to be solved in every step. The alternating direction implicit algorithm and Krylov subspace based methods allow to exploit this special structure.

The rest of the paper is organized as follows. In the next section 2, we summarize the steps of the extended nonsymmetric block Lanczos algorithm to generate the biorthonormal bases and some of the characteristics of the theory. In section 3, we present a low-rank approximation of the solution of the differential Lyapunov equation using projection and low-rank approximation (ENBL-ROS and ENBL-BDF). Finally, Section 4 is devoted to numerical experiments showing the effectiveness of proposed methods.

Throughout the paper, we use the following notations. The Frobenius inner product of the matrices X and Y is defined by $\langle X, Y\rangle_F = \mathrm{tr}(X^T Y)$, where $\mathrm{tr}(Z)$ denotes the trace of a square matrix Z. The associated norm is the Frobenius norm denoted by $\|\cdot\|_F$.

Let $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times s}$, the extended block Krylov subspace $\mathcal{K}^e_m(A, B)$ can be considered as the subspace of $\mathbb{R}^n$ spanned by the columns of the matrices $A^k B$, $k = -m, \dots, m-1$, i.e.,

$$\mathcal{K}^e_m(A, B) = \mathrm{range}\{B, A^{-1}B, AB, A^{-2}B, A^2B, \dots, A^{m-1}B, A^{-m}B\}.$$

Recall extended block Arnoldi (EBA) [13] algorithm, when it applied to the pair (A, B) (m steps). EBA is described in Algorithm 1 as follows

Algorithm 1 The extended block Arnoldi algorithm (EBA).

Inputs: A an $n \times n$ matrix, B an $n \times s$ matrix and m an integer.
Compute the QR decomposition of $[B, A^{-1}B]$, i.e, $[B, A^{-1}B] = V_1\Lambda$.
Set $\mathcal{V}_0 = []$.
For $j = 1, 2, 3, \dots, m$

1. Set $V_j^{(1)} = V_j(:, 1 : s)$ and $V_j^{(2)} = V_j(:, s+1 : 2s)$
2. $\mathcal{V}_j = [\mathcal{V}_{j-1}, V_j]; \widehat{V}_{j+1} = [AV_j^{(1)}, A^{-1}V_j^{(2)}]$.
3. **For** $i = 1, ..., j$
4. $H_{i,j} = V_i^T \widehat{V}_{j+1}$.
5. $\widehat{V}_{j+1} = \widehat{V}_{j+1} - V_i H_{i,j}$.
6. **End For** i
7. Compute the QR decomposition of U i.e., $\widehat{V}_{j+1} = V_{j+1}H_{j+1,j}$.

End For j.

The matrix $\mathcal{V}_m = [V_1, \ldots, V_m] \in \mathbb{R}^{n\times 2ms}$ such that their columns form an orthonormal basis of the extended block subspace $\mathcal{K}_m^e(A, B)$. The $2ms \times 2ms$ Hessenberg matrix

$$\mathcal{H}_m := \mathcal{V}_m^T A \mathcal{V}_m = [H_{i,j}],$$

with $H_{i,j} \in \mathbb{R}^{2s\times 2s}$. Through our algorithm 1 we have the following relations

$$A\mathcal{V}_m = \mathcal{V}_{m+1} \begin{bmatrix} \mathcal{H}_m \\ H_{m+1,m} E_m^T \end{bmatrix}, \mathcal{V}_m^T \mathcal{V}_m = I_{2ms},$$

$$\mathcal{V}_m = \mathcal{V}_{m+1} \begin{bmatrix} I_{2sm} \\ 0_{2s\times 2s} \end{bmatrix} \quad \text{and} \quad \mathcal{V}_m^T B = \mathcal{E}_1 \Lambda_{11},$$

with

$$\Lambda = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ 0 & \Lambda_{22} \end{bmatrix},$$

where $E_m^T = [0_{2s\times 2s(m-1)}, I_{2s}] \in \mathbb{R}^{2s\times 2ms}$ is the matrix formed with the last $2s$ columns of the $2ms\times 2ms$ identity matrix I_{2ms}, and Λ_{11} is the $s \times s$ matrix, $\mathcal{E}_1 = [I_s, 0_{s\times(2m-1)s}]^T$ is the matrix of the first s columns of the $2ms \times 2ms$ identity matrix I_{2ms}.

2. The extended nonsymmetric block Lanczos process

Before describing the extended nonsymmetric block Lanczos process, we have to describe the following procedure which, if applied to $V = [v_1, \ldots, v_s]$, $W = [w_1, \ldots, w_s] \in \mathbb{R}^{n\times s}$ (that's as a two-sided Gram-Schmidt process applied to the two sequences V and W) allows to obtain biorthogonal blocks (BIORTHB) $\mathbb{V}, \mathbb{W} \in \mathbb{R}^{n\times s}$. For more details see [14, 15]

Algorithm 2 The BIORTHB algorithm to the pair (V, W).

Inputs: V, W an $n \times s$ matrix.
Set $\alpha = w_1^T v_1, r_{11} = \sqrt{\alpha}, z_{11} = \frac{\alpha}{r_{11}}, \widetilde{v}_1 = \frac{v_1}{r_{11}}, \widetilde{w}_1 = \frac{w_1}{z_{11}}$.
For $i = 2, 3, \ldots, s$

1. $v = v_i$ et $w = w_i$
2. **For** $j = 1, \ldots, i-1,$
3. $r_{ji} = \widetilde{w}_j^T v$ and $z_{ji} = \widetilde{v}_j^T w$
4. $v = v - r_{ji}\widetilde{v}_j$ and $w = w - z_{ji}\widetilde{w}_j$;
5. **End For** j
6. $\beta = w^T v, r_{ii} = \sqrt{\beta}, z_{ii} = \frac{\beta}{r_{ii}}, \widetilde{v}_i = \frac{v}{r_{ii}}$ and $\widetilde{w}_i = \frac{w}{z_{ii}}$

End For i.
Outputs: $\mathbb{V} = [\widetilde{v}_1, \ldots, \widetilde{v}_s]$ and $\mathbb{W} = [\widetilde{w}_1, \ldots, \widetilde{w}_s]$.

If s iterations are performed, the above algorithm produces $\mathbb{V}, \mathbb{W}$ and $\mathbb{R} = [r_{ji}], \mathbb{Z} = [z_{ji}] \in \mathbb{R}^{s\times s}$ upper triangular matrices such that

$$V = \mathbb{V}\mathbb{R}, \quad W = \mathbb{W}\mathbb{Z}, \quad \mathbb{V}^T\mathbb{W} = I_s. \tag{4}$$

We mention the extended nonsymmetric block Lanczos (ENBL) [14] algorithm when applied to the triple (A, B, C) for constructing two biorthogonal bases $\mathcal{V}_m$ and $\mathcal{W}_m$ of the extended block Krylov subspaces $\mathcal{K}_m^e(A, B)$ and $\mathcal{K}_m^e(A^T, C)$, respectively. ENBL is described in algorithm 3 as follows

Algorithm 3 The ENBL algorithm to triple (A, B, C).

Inputs: A an $n \times n$ matrix, B an $n \times s$ matrix, C an $n \times s$ matrix and m an integer.
Initialize: $W_0 = W_0 = 0_{2s}$ and $N_0 = \widetilde{N}_0 = 0_{2s}$.
Set $U_1 = [B, A^{-1}B]$, $S_1 = [C, A^{-T}C]$.
Apply Algorithm 2 to (U_1, S_1) to get $V_1 = [v_1^{(1)}, v_2^{(1)}]$, $W_1 = [v_1^{(1)}, v_2^{(1)}]$ and Λ, Ω such that $U_1 = V_1\Lambda$ and $S_1 = W_1\Omega$.
Set $\mathcal{V}_2 = [V_1]$ and $\mathcal{W}_2 = [W_1]$.
For $j = 1, 2, \ldots, m$

1. Set $U_{j+1} = [Av_1^{(j)}, A^{-1}v_2^{(j)}]$ and $S_{j+1} = [A^T w_1^{(j)}, A^{-T} w_2^{(j)}]$
2. Set $N_j = W_{j-1}^T U_{j+1}$, $C_j = W_j^T U_{j+1}$ and $\widetilde{N}_j = V_{j-1}^T S_{j+1}$, $\widetilde{C}_j = V_j^T S_{j+1}$.
3. $U_{j+1} = U_{j+1} - V_j C_j - V_{j-1} N_j$ and $S_{j+1} = S_{j+1} - W_j \widetilde{C}_j - W_{j-1}\widetilde{N}_j$
4. Apply Algorithm 2 to (U_{j+1}, S_{j+1}) to get V_{j+1} ,W_{j+1}, A_{j+1}, and $\widetilde{A}_{j+1}$ such that $U_{j+1} = V_{j+1}A_{j+1}$ and $S_{j+1} = W_{j+1}\widetilde{A}_{j+1}$.
5. $\mathcal{V}_{j+1} = [\mathcal{V}_j, V_{j+1}]$ and $\mathcal{W}_{j+1} = [\mathcal{W}_j, W_{j+1}]$

End For m.

Let the matrices biorthonormal basis $\mathcal{V}_m$ and $\mathcal{W}_m$ ($\mathcal{V}_m^T \mathcal{W}_m = I_{2ms}$), and the $2ms \times 2ms$ block triangular matrices $\mathcal{T}_m = \mathcal{W}_m^T A \mathcal{V}_m$ and $\mathcal{L}_m = \mathcal{W}_m^T A^{-1}\mathcal{V}_m$.

Theorem 2 (Ref. [14]). *Suppose that m steps of Algorithm 3 have been carried out. Then we have the following relations*

$$A\mathcal{V}_m = \mathcal{V}_m\mathcal{T}_m + V_{m+1}T_{m+1,m}E_m^T = \mathcal{V}_{m+1}\begin{bmatrix} \mathcal{T}_m \\ T_{m+1,m}E_m^T \end{bmatrix}, \tag{5}$$

$$A^{-T}\mathcal{W}_m = \mathcal{W}_m\mathcal{L}_m^T + W_{m+1}L_{j+1,j}^T E_m^T = \mathcal{W}_{m+1}\begin{bmatrix} \mathcal{L}_m^T \\ L_{m+1,j}^T E_m^T \end{bmatrix}, \tag{6}$$

where $L_{j+1,j} = \mathcal{W}_m^T A^{-1} V_{m+1}$ and $T_{j+1,j} = \mathcal{W}_m^T A V_{m+1}$.

From Algorithm 2, we obtain $\mathcal{W}_m^T C = \mathcal{E}_1\omega_{11}$, where $\Omega = \begin{bmatrix} \omega_{11} & \omega_{12} \\ 0 & \omega_{22} \end{bmatrix}$ and ω_{lm} is the $s \times s$ matrix.

3. Low-rank approximate solution

This approach is based on extended block Krylov projection of the differential Lyapunov equation (1). For more details on extended block Krylov projection method for solving large matrix equations see [3–5, 10]. When we apply the ENBL algorithm 2 on the triple $\left(A, B, \frac{B}{\|B\|_F}\right)$, we get two biorthonormal matrices

$$\mathcal{V}_{m+1} = [V_1, \ldots, V_m, V_{m+1}] \in \mathbb{R}^{n\times 2(m+1)s} \quad \text{and} \quad \mathcal{W}_{m+1} = [W_1, \ldots, W_m, W_{m+1}] \in \mathbb{R}^{n\times 2(m+1)s}.$$

We then consider low-rank approximate solution of the large differential Lyapunov equation (1) that have the form

$$X_m(t) := \mathcal{V}_m \mathbb{X}_m(t) \mathcal{V}_m^T. \tag{7}$$

Let the residual of an approximation $X_m(t)$ of the exact solution $X(t)$ of problem (1) given by

$$R_m(t) = X_m'(t) - AX_m(t) - X_m(t)A^T - BB^T, \tag{8}$$

be the residual associated with the approximation $X_m(t)$ satisfying the Petrov-Galerkin orthogonality condition
$$\mathcal{W}_m^T R_m(t) \mathcal{W}_m = 0_{2ms\times 2ms}. \tag{9}$$

Theorem 3. *Let $\mathbb{X}_m(t)$ be the matrix function defined by (7), then it satisfies the following low-order differential Lyapunov equation*

$$\mathbb{X}_m'(t) = \mathcal{T}_m \mathbb{X}_m(t) + \mathbb{X}_m(t)\mathcal{T}_m^T + B_m B_m^T, \tag{10}$$

where $B_m = \mathcal{E}_1 \omega_{11}$.

Proof. From the equations (7), (9) and

$$R_m(t) = \mathcal{V}_m\big(\mathbb{X}_m'(t) - \mathcal{T}_m\mathbb{X}_m(t) - \mathbb{X}_m(t)\mathcal{T}_m^T - B_m B_m^T\big)\mathcal{V}_m^T,$$

we obtain the low-dimensional differential Lyapunov equation

$$\mathbb{X}_m'(t) = \mathcal{T}_m \mathbb{X}_m(t) + \mathbb{X}_m(t)\mathcal{T}_m^T + B_m B_m^T. \tag{11}$$

■

Now we have to solve the last differential Lyapunov equation (11) by ROS method or BDF method, see [3, 4, 6].

Next, we give a result that allows us to compute the norm of the residual without forming the approximation $X_m(t)$ at each step m. The approximation $X_m(t)$ is computed in a factored form only when convergence is achieved.

Theorem 4. *The residual $R_m(t)$ associated with the approximation $X_m(t)$ satisfies the relation*

$$R_m(t) = -V_{m+1} T_{m+1,m} E_m^T \mathbb{X}_m(t) \mathcal{V}_m^T - \mathcal{V}_m \mathbb{X}_m(t) E_m T_{m+1,m}^T V_{m+1}^T. \tag{12}$$

Proof. We have

$$\begin{aligned} R_m(t) &= X_m'(t) - AX_m(t) - X_m(t)A^T - BB^T \\ &= \mathcal{V}_m \mathbb{X}_m'(t)\mathcal{V}_m^T - A\mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T - \mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T A^T - BB^T, \end{aligned}$$

since $\mathbb{X}_m'(t) = \mathcal{T}_m \mathbb{X}_m(t) + \mathbb{X}_m(t)\mathcal{T}_m^T + B_m B_m^T$, so

$$\begin{aligned} R_m(t) &= \mathcal{V}_m\big(\mathcal{T}_m\mathbb{X}_m(t) + \mathbb{X}_m(t)\mathcal{T}_m^T + B_m B_m^T\big)\mathcal{V}_m^T - A\mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T - \mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T A^T - BB^T \\ &= \mathcal{V}_m\mathcal{T}_m\mathbb{X}_m(t)\mathcal{V}_m^T + \mathcal{V}_m\mathbb{X}_m(t)\mathcal{T}_m^T\mathcal{V}_m^T - A\mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T - \mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T A^T, \end{aligned}$$

since $\mathcal{V}_m\mathcal{T}_m = A\mathcal{V}_m - V_{m+1}T_{m+1,m}E_m^T$, we obtain

$$R_m(t) = -V_{m+1} T_{m+1,m} E_m^T \mathbb{X}_m(t) \mathcal{V}_m^T - \mathcal{V}_m \mathbb{X}_m(t) E_m T_{m+1,m}^T V_{m+1}^T,$$

so the proof is complete. ■

Theorem 5. *The Frobenius norm of the residual $R_m(t)$ associated with the approximation $X_m(t)$ satisfies the relation*
$$\|R_m(t)\|_F = \|T_{m+1,m}E_m^T\mathbb{X}_m(t)\|_F. \tag{13}$$

Proof. We have

$$R_m(t) = \mathcal{V}_m \mathbb{X}_m'(t)\mathcal{V}_m^T - A\mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T - \mathcal{V}_m\mathbb{X}_m(t)\mathcal{V}_m^T A^T - BB^T,$$

since $W_{m+1}^T V_m = 0$, $W_{m+1}^T \mathcal{V}_m = 0$ and $W_{m+1}^T B = 0$ we obtain

$$W_{m+1}^T R_m(t) \mathcal{W}_m = -T_{m+1,m} E_m^T \mathbb{X}_m(t),$$

since

$$\mathcal{W}_{m+1}^T R_m(t) \mathcal{W}_m = \begin{bmatrix} \mathcal{W}_m^T R_m(t) \mathcal{W}_m \\ W_{m+1}^T R_m(t) \mathcal{W}_m \end{bmatrix} = \begin{bmatrix} 0 \\ W_{m+1}^T R_m(t) \mathcal{W}_m \end{bmatrix},$$

which proves the result, since $\|R_m(t)\|_F = \|\mathcal{W}_{m+1}^T R_m(t) \mathcal{W}_m\|_F$. ∎

To save memory, the solution $X_m(t) = \mathcal{V}_m \mathbb{X}_m(t) \mathcal{V}_m^T$ can be given as a product of two matrices of low-rank. For that, we consider the singular value decomposition of the $2ms \times 2ms$ matrix $\mathbb{X}_m = UDU^T$, where D is the diagonal matrix of the singular values of $\mathbb{X}_m$ sorted in decreasing order. Let U_l be the $2ms \times l$ matrix of the first l columns of U, corresponding to the l singular values of magnitude greater than some tolerance d_{tol}. We obtain the truncated singular value decomposition $\mathbb{X}_m \approx U_l D_l U_l^T$, where $D_l = \text{diag}[\lambda_1, \ldots, \lambda_l]$. Setting $Z_m = \mathcal{V}_m U_l D_l^{1/2}$ it follows that

$$X_m = Z_m Z_m^T. \tag{14}$$

This result is important for large-scale problems to decrease central processing unit (CPU) time and memory requirements; the approximate solution could be given as a product of low-rank matrices.

We summarize the above method for solving large differential Lyapunov equation (ENBL-BDF or ENBL-ROS) in following algorithm.

Algorithm 4 The ENBL-... method for solving differential Lyapunov equation

Inputs: $A \in \mathbb{R}^{n,n}$ and $B \in \mathbb{R}^{n,s}$ an matrix, t_0, t_f .
Choose a tolerance $tol > 0$ and an integer m_{max}.
Initialize: $W_0 = W_0 = 0_{2s}$ and $N_0 = \widetilde{N}_0 = 0_{2s}$.
Set $U_1 = [B, A^{-1}B]$, $S_1 = [B, A^{-T}B]$.
Apply Algorithm 2 to U_1 and S_1 to get $V_1 = [v_1^{(1)}, v_2^{(1)}], W_1 = [v_1^{(1)}, v_2^{(1)}]$ and Λ, Ω such that $U_1 = V_1\Lambda$ and $S_1 = W_1\Omega$.
Set $\mathcal{V}_2 = [V_1]$ and $\mathcal{W}_2 = [W_1]$.
For $m = 1, 2, \ldots, m_{max}$

1. Set $U_{m+1} = \left[Av_1^{(m)}, A^{-1}v_2^{(m)}\right]$ and $S_{m+1} = \left[A^T w_1^{(m)}, A^{-T} w_2^{(m)}\right]$
2. Set $N_m = W_{m-1}^T U_{m+1}$, $C_m = W_m^T U_{m+1}$ and $\widetilde{N}_m = V_{m-1}^T S_{m+1}$, $\widetilde{C}_m = V_m^T S_{m+1}$.
3. $U_{m+1} = U_{m+1} - V_m C_m - V_{m-1} N_m$ and $S_{m+1} = S_{m+1} - W_m \widetilde{C}_m - W_{m-1} \widetilde{N}_m$
4. Apply Algorithm 2 to U_{m+1} and S_{m+1} to compute V_{m+1} ,W_{m+1}, A_{m+1}, and $\widetilde{A}_{m+1}$ such that $U_{m+1} = V_{m+1} A_{m+1}$ and $S_{m+1} = W_{m+1} \widetilde{A}_{m+1}$.
5. $\mathcal{V}_{m+1} = [\mathcal{V}_m, V_{m+1}]$ and $\mathcal{W}_{m+1} = [\mathcal{W}_m, W_{m+1}]$
6. Compute the $\mathcal{T}_m$ and B_m.
7. Compute $\mathbb{X}_m(t)$ solution of low-dimensional differential Lyapunov equation (10).
8. If $\|R_m(t)\|_F < tol$.

End For m.
Compute the approximate solution X_m in the factored form given by the relation (14).

The following result shows that the approximation $X_m(t)$ is an exact solution of a perturbed differential Lyapunov equation.

Theorem 6. *Let $X_m(t) = \mathcal{V}_m \mathbb{X}_m(t) \mathcal{V}_m^T$ be the approximate solution obtained after running m steps of the ENBL. Then we have*

$$X'_m(t) = (A - F_m) X_m(t) + X_m(t)(A - F_m)^T + BB^T, \tag{15}$$

where $F_m = V_{m+1} T_{m+1,m} W_m^T$.

Proof. Multiply the left the equation (10) by $\mathcal{V}_m$ and from the right by $\mathcal{V}_m^T$, we obtain

$$X'_m(t) = \big(A\mathcal{V}_m - V_{m+1} T_{m+1,m} E_m^T\big) \mathbb{X}_m(t) \mathcal{V}_m^T + \mathcal{V}_m \mathbb{X}_m(t) \big(A\mathcal{V}_m - V_{m+1} T_{m+1,m} E_m^T\big)^T - BB^T.$$

On the other hand, since $\mathcal{W}_m^T \mathcal{V}_m = I_{2ms}$, and $E_m^T \mathcal{W}_m^T = W_m^T$, we have

$$X'_m(t) = (A - F_m) X_m(t) + X_m(t)(A - F_m)^T + BB^T.$$

■

Through this result and expression F_m, shows F_m to zero, so the approximate solution $X_m(t)$ is an exact solution for large-scale differential Lyapunov equation.

The following result indicates that the error matrix $\mathbb{E}_m(t) = X(t) - X_m(t)$ satisfies a differential Lyapunov equation.

Theorem 7. *Let the matrix function $\mathbb{E}_m(t)$ verify the following differential Lyapunov equation*

$$\mathbb{E}'_m(t) = A\mathbb{E}_m(t) + \mathbb{E}_m(t) A^T - R_m(t). \tag{16}$$

Proof.

$$\begin{aligned}
\mathbb{E}'_m(t) &= X'(t) - X'_m(t) \\
&= AX(t) + X(t)A^T + BB^T - AX_m(t) - X_m(t)A^T - BB^T - R_m(t) \\
&= A(X(t) - X_m(t)) + (X(t) - X_m(t))A^T - R_m(t) \\
&= A\mathbb{E}_m(t) + \mathbb{E}_m(t)A^T - R_m(t).
\end{aligned}$$

■

Notice that from theorem 1, the error $\mathbb{E}_m(t)$ can be expressed in the integral form as follows

$$\mathbb{E}_m(t) = e^{(t-t_0)A} \mathbb{E}_m(t_0) e^{(t-t_0)A^T} + \int_{t_0}^{t} e^{(t-\tau)A} R_m(\tau) e^{(t-\tau)A^T} d\tau, \quad t \in [t_0, t_f]. \tag{17}$$

Next, we give an upper bound for the norm of the error by using the 2-logarithmic norm.

Theorem 8. *Assume that the matrix A is such that $\mu_2(A) \neq 0$. Then at step m of the ENBL, we have the following upper bound for the norm of the error $\mathbb{E}_m(t)$,*

$$\|\mathbb{E}_m(t)\|_2 \leqslant \|\mathbb{E}_m(t_0)\|_2 e^{2(t-t_0)\mu_2(A)} + \alpha_m \frac{e^{2(t-t_0)\mu_2(A)} - 1}{2\mu_2(A)}, \tag{18}$$

where $\alpha_m = \max_{\xi \in [t_0, t]} \|R_m(\xi)\|_2$.

Proof. Using the expression (17) of $\mathbb{E}_m(t)$, we obtain the following relation

$$\|\mathbb{E}_m(t)\|_2 = \|e^{(t-t_0)A} \mathbb{E}_{m,0} e^{(t-t_0)A^T}\|_2 + \int_{t_0}^{t} \|e^{(t-\tau)A} R_m(\tau) e^{(t-\tau)A^T}\|_2 d\tau.$$

Therefore, using equation (17) and the fact that $\|e^{(t-\tau)A}\|_2 \leqslant e^{(t-\tau)\mu_2(A)}$, we get

$$\|\mathbb{E}_m(t)\|_2 \leqslant \|\mathbb{E}_{m,0}\|_2\, e^{(t-t_0)(\mu_2(A)+\mu_2(A^T))} + \max_{\xi\in[t_0,t]} \|R_m(\xi)\|_2 \int_{t_0}^{t} e^{(t-\tau)\mu_2(A)} e^{(t-\tau)\mu_2(A^T)} d\tau$$
$$\leqslant \|\mathbb{E}_{m,0}\|_2\, e^{2(t-t_0)\mu_2(A)} + \max_{\xi\in[t_0,t]} \|R_m(\xi)\|_2 e^{t2\mu_2(A)} \int_{t_0}^{t} e^{-2\tau\mu_2(A)} d\tau,$$

so the proof is complete. ■

In Figure 1, we computed the upper bound error norm $\|\mathbb{E}_m(t)\|_2$ vs number m of iterations, we use the matrix A obtained from the discretization of the from the `Lyapack` package [16] using the command `fdm_2d_matrix` $A = \mathrm{fdm}(n0, e^{xy}, \sin(xy), y^2 - x^2)$, and $B = \mathrm{rand}(n, s)$.

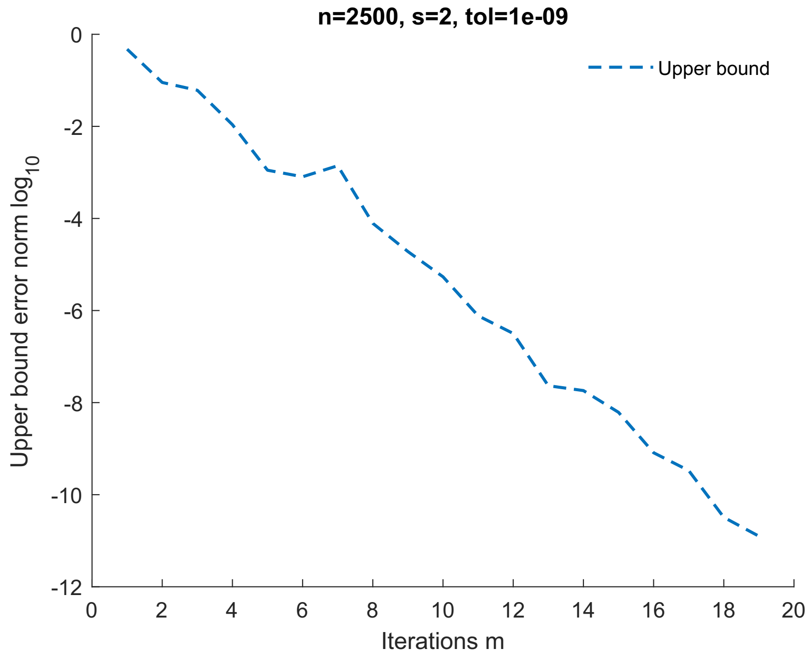

Fig. 1. Norm of the upper bound errors vs number m of iterations.

4. Numerical experiments

In this section, we report some experimental results. All the numerical experiments have been coded in MATLAB 2018b, PC-Intel(R) Core(TM) i3, 4 GB of RAM. We compare the performance of the extended block Arlondi and new methods with equal-sized approximation spaces. In our experiments, we used 4 methods listed in Table 1. The time interval considered was $[0, 1]$ and the initial condition $X_0 = 0_{n\times n}$. The results are shown in Table 2, we give the number of iterations (Iter), the residual

Table 1. The methods.

EBA-BDF	Extended block Arlondi and BDF method
ENBL-BDF	Extended nonsymmetric block Lanczos and BDF method
EBA-ROS	Extended block Arlondi and ROS method
ENBL-ROS	Extended nonsymmetric block Lanczos and ROS method

norm (Res.norm), and the CPU time in seconds (CPU time) required for convergence, we use $s = 2$, and $tol = 10^{-15}$. In this example [6], we set $A = (M - dtK)^{-1}M$, and $B = dt(M - dtK)^{-1}F$, where the matrices M and K are given by:

$$M = \frac{1}{6n}\begin{pmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ & & & 1 & 4 \end{pmatrix}, \quad K = -\alpha n \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}.$$

The entries of the $n \times s$ matrix F and the $s \times n$ matrix C were random values uniformly distributed on $[0, 1]$. In our experiments we used $dt = 0.1, \alpha = 0.5$ and $s = 2$ for different values of n.

Table 2. Runtimes in seconds, the residual norms and iterations for each method.

Test Problem	Method	CPU time	Iter	Res.norm
$n = 2500$	EBA-ROS	2.80	30	5.28977×10^{-15}
	ENBL-ROS	1.28	9	6.87600×10^{-17}
	EBA-BDF	2.51	30	8.51644×10^{-14}
	ENBL-BDF	1.25	9	8.5535×10^{-16}
$n = 4600$	EBA-ROS	12.83	30	6.4430×10^{-14}
	ENBL-ROS	7.42	9	1.37058×10^{-16}
	EBA-BDF	12.28	30	3.56598×10^{-13}
	ENBL-BDF	7.38	9	9.65655×10^{-16}
$n = 8100$	EBA-ROS	19.12	30	1.03065×10^{-13}
	ENBL-ROS	11.67	9	3.03288×10^{-16}
	EBA-BDF	18.77	30	3.40216×10^{-12}
	ENBL-BDF	15.24	11	2.39502×10^{-16}
$n = 10000$	EBA-ROS	28.68	30	7.97113×10^{-14}
	ENBL-ROS	17.60	9	4.4548×10^{-16}
	EBA-BDF	28.25	30	3.3338×10^{-12}
	ENBL-BDF	21.33	11	7.83282×10^{-16}

We used a constant timestep $h = 0.1$. In Figure 2, we chose a size of 5600×5600, for the matrices A, we plotted the Frobenius norms of the residuals at final time t_f versus the number of iterations.

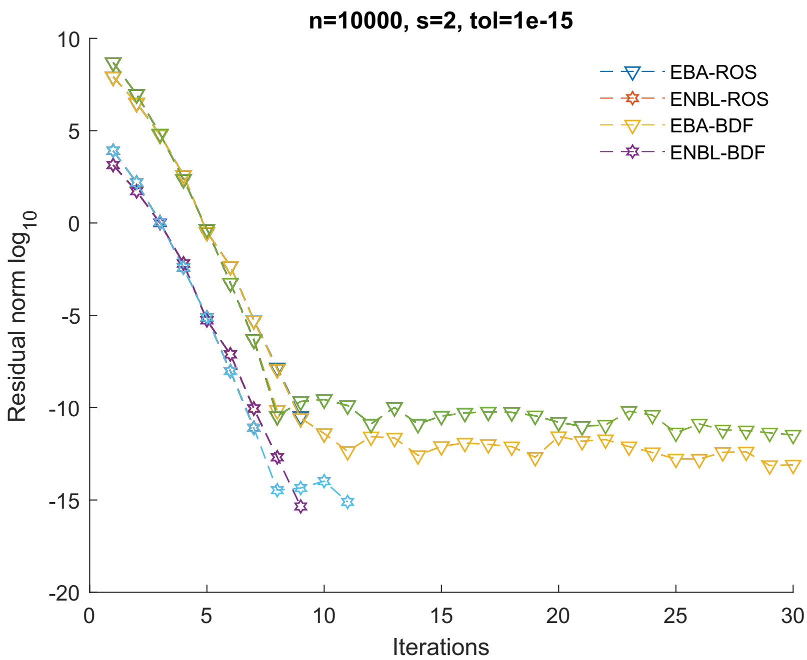

Fig. 2. Residual norm vs number m iterations.

5. Conclusion

In this paper, we presented new iterative method for solving large-scale differential Lyapunov matrix equations. The proposed method is based on the extended nonsymmetric block Lanczos algorithm and the Backward Differentiation Formula method (BDF) or Rosenbrock method (ROS). The numerical experiments show that the proposed new approach is effective for large and sparse problems.

[1] Abou-Kandil H., Freiling G., Ionescu V., Jank G. Matrix Riccati equations in control and systems theory. Birkhäuser (2012). .

[2] Datta B. Numerical methods for linear control systems (Vol. 1). Academic Press (2004).

[3] Sadek L., Talibi Alaoui H. The extended block Arnoldi method for solving generalized differential Sylvester equations. Journal of Mathematical Modeling. **8** (2), 189–206 (2020).

[4] Sadek E. M., Bentbib A. H., Sadek L., Talibi Alaoui H. Global extended Krylov subspace methods for large-scale differential Sylvester matrix equations. J. Appl. Math. Comput. **62**, 157–177 (2020).

[5] Agoujil S., Bentbib A. H., Jbilou K., Sadek E. M. A minimal residual norm method for large-scale Sylvester matrix equations. Electronic Transactions on Numerical Analysis. **43**, 45–59 (2014).

[6] Hached M., Jbilou K. Numerical solutions to large-scale differential Lyapunov matrix equations. Numer Algor. **79**, 741–757 (2018).

[7] Sadek L., Talibi Alaoui H. Numerical methods for solving large-scale systems of differential equations. Ricerche di Matematica. (2021).

[8] Bentbib A., Jbilou K., Sadek E. M. On some Krylov subspace based methods for large-scale nonsymmetric algebraic Riccati problems. Computers & Mathematics with Applications. **70** (10), 2555–2565 (2015).

[9] Mena H., Ostermann A., Pfurtscheller L. M., Piazzola C. Numerical low-rank approximation of matrix differential equations. Journal of Computational and Applied Mathematics. **340**, 602–614 (2018).

[10] Bentbib A. H., Jbilou K., Sadek E. L. On some extended block Krylov based methods for large scale nonsymmetric Stein matrix equations. Mathematics. **5** (2), 21 (2017).

[11] Behr M., Benner P., Heiland J. Solution formulas for differential Sylvester and Lyapunov equations. Calcolo. **56**, 51 (2019).

[12] Lang N., Mena H., Saak J. On the benefits of the LDLT factorization for large-scale differential matrix equation solvers. Linear Algebra and its Applications. **480**, 44–71 (2015).

[13] Druskin V., Knizhnerman L. Extended Krylov subspaces: approximation of the matrix square root and related functions. SIAM Journal on Matrix Analysis and Applications. **19** (3), 755–771 (1998).

[14] 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 (2018).

[15] Parlett B. N., Taylor D. R., Li Z. A. A look-ahead Lanczos algorithm for unsymmetric matrices. Math. Comput. **44** (169), 105–124 (1985).

[16] Penzl T. L YAPACK A MATLAB Toolbox for Large Lyapunov and Riccati Equations, Model Reduction Problems, and Linear-Quadratic Optimal Control Problems. Available online: `http://www.tuchemintz.de/sfb393/lyapack`.

Розширений несиметричний блок методів Ланцоша для розв'язування великомасштабних диференціальних рівнянь Ляпунова

Садек Л., Талібі Алауї Г.

Кафедра математики, Факультет наук, Чуайб Дуккалі університет, Ель Джадіда, Марокко

У статті представено новий підхід до розв'язання великомасштабних диференціальних рівнянь Ляпунова. Запропонований підхід базується на проектуванні початкової задачі на розширеному блоці підпростору Крилова, використовуючи розширений несиметричний алгоритм Ланцоша. У результаті отримується низькорозмірне диференціальне матричне рівняння Ляпунова. Це диференціальне матричне рівняння розв'язується методом диференціацювання назад або методом Розенброка. Отриманий розв'язок дозволяє створювати наближений розв'язок початкової задачі. Крім того, дано деякі теоретичні результати. Чисельні результати демонструють продуктивність запропонованого підходу.

Ключові слова: *розширений блок підпростору Крилова, розширений несиметричний блок алгоритма Ланцоша, наближення низького рангу, диференціальні рівняння Ляпунова.*