



Approximate solutions to large nonsymmetric differential Riccati problems with applications to transport theory

V. Angelova¹ | M. Hached² | K. Jbilou³

¹Department of Intelligent Systems,
Institute of Information and
Communication Technologies, Bulgarian
Academy of Sciences, Sofia, Bulgaria

²Laboratoire P. Painlevé UMR, UFR
de Mathématiques, Université des
Sciences et Technologies de Lille,
Villeneuve d'Ascq, France

³Laboratoire de Mathématiques Pures et
Appliquées, Université du Littoral Côte
d'Opale, Calais, France

Correspondence

K. Jbilou, Laboratoire de Mathématiques
Pures et Appliquées, Université du Littoral
Côte d'Opale, 50 Rue F. Buisson BP699,
62228 Calais, France.
Email: jbilou@univ-littoral.fr

Summary

In this paper, we consider large-scale nonsymmetric differential matrix Riccati equations with low-rank right-hand sides. These matrix equations appear in many applications such as control theory, transport theory, applied probability, and others. We show how to apply Krylov-type methods such as the extended block Arnoldi algorithm to get low-rank approximate solutions. The initial problem is projected onto small subspaces to get low dimensional nonsymmetric differential equations that are solved using the exponential approximation or via other integration schemes such as backward differentiation formula (BDF) or Rosenbrock method. We also show how these techniques can be easily used to solve some problems from the well-known transport equation. Some numerical examples are given to illustrate the application of the proposed methods to large-scale problems.

KEY WORDS

differential Riccati equation, extended block Arnoldi, low-rank approximation, transport theory

AMS CLASSIFICATION

65F10; 65F30

1 | INTRODUCTION

Consider the nonsymmetric differential Riccati equation

$$\begin{cases} \dot{X}(t) = -AX(t) - X(t)D + X(t)SX(t) + Q, & (\text{NDRE}) \\ X(0) = X_0, \end{cases} \quad (1)$$

where $A \in \mathbb{R}^{n \times n}$, $D \in \mathbb{R}^{p \times p}$, $Q \in \mathbb{R}^{n \times p}$, $S \in \mathbb{R}^{p \times n}$, and $X(t) \in \mathbb{R}^{n \times p}$ with $t \in [t_0, t_f]$.

The equilibrium solutions of (1) are the solutions of the corresponding nonsymmetric algebraic Riccati equation (NARE).

$$-AX - XD + XSX + Q = 0. \quad (2)$$

Differential nonsymmetric Riccati equations (NDREs) play a fundamental role in many areas such as transport theory, fluid queues models, variational theory, optimal control and filtering, H_1 -control, invariant embedding and scattering processes, dynamic programming, and differential games.^{1–5}

For NAREs, many numerical methods have been studied for finding the minimal nonnegative solution X^* . The Newton method has been studied in other works^{2,6,7}; however, because it requires at each step the solution of a Sylvester equation, the method could be expensive when direct solvers are used. Generally, fixed point iteration methods^{1,2,7} are less expensive

than the Newton or the Schur method. Some acceleration techniques based on vector extrapolation methods⁸ have been proposed in the work of El-Moallem et al.⁹ to speed up the convergence of some of these fixed point iterative methods such as those introduced in the works of Lu.^{10,11} For large problems, some Krylov-based methods have been studied in the work of Bentbib et al.¹²

For NDREs and to our knowledge, there is no existing method in the large-scale case. In this paper, we consider large-scale NDREs with low-rank right-hand sides. We will show how to apply the extended block Arnoldi (EBA) algorithm^{13,14} to get low-rank approximate solutions. We will treat the special case corresponding to NDREs from transport theory.

This paper is organized as follows: In Section 2, we will be interested in the existence of exact solutions to Equation (1). In Section 3, we will see how to apply the EBA process to get low-rank approximate solutions to NDREs with low-rank right-hand sides. We give different ways for solving the obtained projected low dimensional NDREs. Some convergence and perturbation results are developed in this section. In Section 4, we investigate the BDF–Newton method for solving the problem (1). Section 5 is devoted to the special case where Equation (1) comes from transport theory. In the last section, we give some numerical examples.

Throughout this paper, we use the following notations: The matrix I_n will denote the identity matrix of size $n \times n$. The 2-norm is denoted by $\|\cdot\|_2$.

2 | EXACT SOLUTIONS TO NDREs

We first need to recall some relevant definitions:

Definition 1. For any real matrices $L = [l_{ij}]$ and $N = [n_{ij}]$ with the same size, we write $L \geq N$ if $l_{ij} \geq n_{ij}$.

Definition 2. A real square matrix A is said M-matrix if $A = sI - H$ with $H \geq 0$ and $s \geq \rho(H)$, where $\rho(\cdot)$ denotes the spectral radius. An M-matrix A is nonsingular if $s > \rho(H)$.

Let \mathcal{L} be the following matrix:

$$\mathcal{L} = \begin{pmatrix} D & -S \\ -Q & A \end{pmatrix}. \quad (3)$$

In this paper, we assume that the matrix \mathcal{L} is a nonsingular M-matrix. It follows that the matrices A and D are both nonsingular M-matrices; see the work of Fital et al.¹⁵

We notice that the special structure of the matrix \mathcal{L} ensures the existence of the minimal nonnegative solution X^* such that $X^* \geq 0$ and $X \geq X^*$ for any solution X of the NARE (1); see other works^{2,7,16} for more details.

A solution of (2) can be expressed in the following form:

$$X(t) = e^{-tA}X_0e^{-tD} + \int_0^t e^{-(t-\tau)A}Qe^{-(t-\tau)D}d\tau + \int_0^t e^{-(t-\tau)A}X(\tau)SX(\tau)e^{-(t-\tau)D} d\tau. \quad (4)$$

The proof is easily done by differentiation. Now, as the matrices A and D are also nonsingular M-matrices, they can be expressed as $A = A_1 - A_2$ and $D = D_1 - D_2$, where A_2, D_2 are positive matrices and A_1 and D_1 are nonsingular M-matrices. Therefore, a solution of (1) can be expressed as follows: (See the work of Juang¹⁷)

$$X(t) = e^{-tA_1}X_0e^{-tD_1} + \int_0^t e^{-(t-\tau)A_1}(X(\tau)SX(\tau) + A_2X(\tau) + X(\tau)D_2 + Q)e^{-(t-\tau)D_1} d\tau. \quad (5)$$

Because \mathcal{L} is assumed to be a nonsingular M-matrix, then it has been proved in the work of Fital et al,¹⁵ by using a Picard iteration, that if $0 \leq X_0 \leq X^*$, where X^* is a nonnegative solution of (2), then there exists a global solution $X(t)$ of (1).

It is also well known¹ that the NDRE (1) is related to the initial value problem

$$\begin{pmatrix} \dot{Y}(t) \\ \dot{Z}(t) \end{pmatrix} = \begin{pmatrix} D & -S \\ Q & -A \end{pmatrix} \begin{pmatrix} Y(t) \\ Z(t) \end{pmatrix}, \quad Y(0) = I, Z(0) = X_0, \quad (6)$$

where $Y(t) \in \mathbb{R}^{p \times p}$ and $Z(t) \in \mathbb{R}^{n \times p}$. The solution of the differential linear system (6) is given by

$$\begin{pmatrix} Y(t) \\ Z(t) \end{pmatrix} = e^{t\mathcal{H}} \begin{pmatrix} I \\ X_0 \end{pmatrix}, \quad (7)$$

where

$$\mathcal{H} = \begin{pmatrix} D & -S \\ Q & -A \end{pmatrix}.$$

Therefore, using the Radon's lemma (see the work of Abou-Kandil et al.¹), we can state the following result¹⁵:

Theorem 1. *The problem (1) is equivalent to solving the linear system of differential equations 6. If the solution $X(t)$ exists on $[0, \infty[$, then the solution $Y(t)$ obtained from the problem (6) is nonsingular and in this case,*

$$X(t) = Z(t)Y^{-1}(t).$$

Using this theorem, we obtain the following result¹⁵:

Theorem 2. *Assume that \mathcal{L} is a nonsingular M matrix. If $0 \leq X_0 \leq X^*$, where X^* is the minimal nonnegative solution of (2), then the solution $X(t)$ of (1) converges to X^* as $t \rightarrow \infty$.*

3 | LOW-RANK APPROXIMATE SOLUTIONS TO LARGE NDREs VIA PROJECTION

3.1 | The approximate solutions

From now on, we assume that the constant matrix term Q in (1) has a low-rank approximate solution and is decomposed as $Q = FG^T$ and $X_0 = Z_{0,1}Z_{0,2}^T$, where $F, Z_{0,1} \in \mathbb{R}^{n \times s}$ and $G, Z_{0,2} \in \mathbb{R}^{p \times s}$ with $s \ll n$. The approach that we will consider in this section, consists in projecting the problem (1) onto a suitable subspace, solves the obtained low-order problem and then get an approximate solution to the original problem.

We first recall the EBA process applied to the pair (A, V) , where $A \in \mathbb{R}^{n \times n}$ is assumed to be nonsingular and $V \in \mathbb{R}^{n \times s}$ with $s \ll n$. The projection subspace $\mathcal{K}_m(A, V) \subset \mathbb{R}^n$ that we will consider was introduced in other works^{14,18} and applied for solving large-scale symmetric differential and algebraic matrix Riccati equations in other works^{13,19} and for solving large-scale Lyapunov matrix equations in the work of Simoncini.¹⁴ This extended block Krylov subspace is given as

$$\mathcal{K}_m(A, V) = \text{Range}([A^{-m}V, \dots, A^{-2}V, A^{-1}V, V, AV, A^2V, \dots, A^{m-1}V]).$$

The EBA algorithm allows the computation of an orthonormal basis of the extended Krylov subspace $\mathcal{K}_m(A, V)$. This basis contains information on both A and A^{-1} . Let m be some fixed integer, which limits the dimension of the constructed basis. The obtained blocks V_1, V_2, \dots, V_m , ($V_i \in \mathbb{R}^{n \times 2s}$) have their columns mutually orthogonal provided no breakdown occurs. After m steps, the EBA algorithm builds an orthonormal basis $\mathcal{V}_m = [V_1, \dots, V_m]$ of the extended block Krylov subspace $\mathcal{K}_m(A, V)$.

Let the matrix $\mathcal{T}_m^A \in \mathbb{R}^{2ms \times 2ms}$ denotes the restriction of the matrix A to the extended Krylov subspace $\mathcal{K}_m(A, V)$, that is, $\mathcal{T}_m^A = \mathcal{V}_m^T A \mathcal{V}_m$. It is shown in the work of Simoncini¹⁴ that \mathcal{T}_m^A is a block upper Hessenberg matrix with $2s \times 2s$ blocks and whose elements could be obtained recursively from EBA. Let $\bar{\mathcal{T}}_m^A = \mathcal{V}_{m+1}^T A \mathcal{V}_m$, and suppose that m steps of EBA have been run, then we have¹³:

$$A \mathcal{V}_m = \mathcal{V}_{m+1} \bar{\mathcal{T}}_m^A = \mathcal{V}_m \mathcal{T}_m^A + V_{m+1} T_{m+1,m}^A E_m^T, \quad (8)$$

and

$$A^{-1} \mathcal{V}_m = \mathcal{V}_{m+1} \bar{\mathcal{L}}_m^A = \mathcal{V}_m \mathcal{L}_m^A + V_{m+1} L_{m+1,m}^A E_m^T,$$

with $\bar{\mathcal{L}}_m^A = \mathcal{V}_{m+1}^T A^{-1} \mathcal{V}_m$ and $\mathcal{L}_m^A = \mathcal{V}_m^T A^{-1} \mathcal{V}_m$, where $T_{m+1,m}^A$ and $L_{m+1,m}^A$ are the $(m+1, m)$ -block (of size $2s \times 2s$) of $\bar{\mathcal{T}}_m^A$ and $\bar{\mathcal{L}}_m^A$, respectively, and $E_m = [O_{2s \times 2(m-1)s}, I_{2s}]^T$ is the matrix of the last $2s$ columns of the $2ms \times 2ms$ identity matrix I_{2ms} .

We notice that as EBA requires mat–vec products with the matrices A and A^{-1} , so if the matrix A is singular or when solving linear systems with A is expensive, then one should use the block Arnoldi algorithm that requires only mat–vec

products with the matrix A . In that case, the obtained blocks V_i 's are of dimension $n \times s$ and form an orthonormal basis of the block Krylov subspace $\mathbb{K}(A, V) = \text{Range}([V, AV, \dots, A^{m-1}V])$. However, the block Arnoldi process requires generally more execution time to get good approximate solutions as compared to EBA.

In what follows, we will use the EBA algorithm, but all the results are valid when using the block Arnoldi process. To get low-rank approximate solutions to (1), we first apply the EBA algorithm (or the block Arnoldi algorithm) to the pairs (A, F) and (D, G) to generate two orthonormal bases $\{V_1, \dots, V_m\}$ and $\{W_1, \dots, W_m\}$ of the extended Krylov subspaces $\mathcal{K}_m(A, F)$ and $\mathcal{K}_m(D, G)$, respectively. We obtain two orthonormal matrices $\mathcal{V}_m = [V_1, \dots, V_m]$ and $\mathcal{W}_m = [W_1, \dots, W_m]$ and two block Hessenberg matrices $\tilde{\mathcal{T}}_m^A = \mathcal{V}_m^T A \mathcal{V}_m$ and $\tilde{\mathcal{T}}_m^D = \mathcal{W}_m^T D \mathcal{W}_m$.

Let $X_m(t)$ be the proposed approximate solution to (1) given in the low-rank form

$$X_m(t) = \mathcal{V}_m Y_m(t) \mathcal{W}_m^T, \quad (9)$$

satisfying the Galerkin orthogonality condition

$$\mathcal{V}_m^T R_m(t) \mathcal{W}_m = 0, \quad (10)$$

where $R_m(t)$ is the residual $R_m(t) = \dot{X}_m(t) + AX_m(t) + X_m(t)D - X_m(t)SX_m(t) - FG^T$ associated to the approximation $X_m(t)$. Then, from (9) and (10), we obtain the low-dimensional differential Riccati equation

$$\begin{cases} \dot{Y}_m(t) = -\mathcal{T}_m^A Y_m(t) - Y_m(t) \mathcal{T}_m^D + Y_m(t) S_m Y_m(t) + F_m G_m^T \\ Y_m(0) = Y_0 = \mathcal{V}_m^T X_0 \mathcal{W}_m \end{cases} \quad (11)$$

with $S_m = \mathcal{W}_m^T S \mathcal{V}_m$, $F_m = \mathcal{V}_m^T F$ and $G_m = \mathcal{W}_m^T G$. As $X_0 = Z_{0,1} Z_{0,2}^T$, the initial guess Y_0 can be expressed as $Y_0 = \tilde{Y}_{0,1} \tilde{Y}_{0,2}^T$, where $\tilde{Y}_{0,1} = \mathcal{V}_m^T Z_{0,1}$ and $\tilde{Y}_{0,2} = \mathcal{W}_m^T Z_{0,2}$.

Therefore, the obtained low-dimensional nonsymmetric differential Riccati Equation (11) will be solved by some classical integration method that we will see in Sections 3.2–3.4.

In order to stop the EBA iterations, it is desirable to be able to test if $\|R_m\| < \epsilon$, where ϵ is some chosen tolerance, without having to compute extra matrix products involving the matrices A and D and their inverses. The next result gives an expression of the residual norm of $R_m(t)$, which does not require the explicit calculation of the approximate $X_m(t)$. A factored form will be computed only when the desired accuracy is achieved.

Theorem 3. Let $X_m(t) = \mathcal{V}_m Y_m(t) \mathcal{W}_m^T$ be the approximation obtained at step m by the EBA method, where Y_m solves the low-dimensional differential Riccati Equation (11). Then,

$$\|R_m(t)\| = \max \left\{ \|T_{m+1,m}^A E_m^T Y_m(t)\|, \|Y_m(t) E_m T_{m+1,m}^D\| \right\}, \quad (12)$$

where Y_m is solution of (11).

Proof. Using the fact that Y_m is a solution of the low-order Riccati Equation (11), we get

$$R_m(t) = \mathcal{V}_{m+1} \begin{pmatrix} 0 & Y_m(t) E_m T_{m+1,m}^D \\ \mathcal{T}_{m+1,m}^A E_m^T Y_m(t) & 0 \end{pmatrix} \mathcal{W}_{m+1}^T. \quad (13)$$

Then, because \mathcal{V}_{m+1} and \mathcal{W}_{m+1} are orthonormal matrices, the result follows. \square

Let us see now how the obtained approximation can be expressed in a factored form. As for the algebraic case,^{13,19} using the singular value decomposition of $Y_m(t)$, and neglecting the singular values that are close to zero, the approximate solution $X_m(t) = \mathcal{V}_m Y_m(t) \mathcal{W}_m^T$ can be given in the following factored form

$$X_m(t) \approx Z_{m,1}(t) Z_{m,2}^T(t),$$

where $Z_{m,1}(t)$ and $Z_{m,2}(t)$ are small rank matrices.

The following result shows that the approximation X_m is an exact solution of a perturbed differential Riccati equation and that the error $\mathcal{E}_m(t) = X(t) - X_m(t)$ solves another nonsymmetric differential Riccati equation.

Theorem 4. Let X_m be the approximate solution given by (9). Then, we have

$$\begin{aligned}\dot{X}_m(t) &= -(A - \Delta_m^A) X_m(t) - X_m(t) (D - \Delta_m^D) + X_m(t) S X_m(t) + FG^T, \\ R_m(t) &= \Delta_m^A X_m + X_m \Delta_m^D, \text{ and} \\ \dot{\mathcal{E}}_m(t) &= -(A - X_m S) \mathcal{E}_m(t) - \mathcal{E}_m(t) (D - S X_m) + \mathcal{E}_m(t) S \mathcal{E}_m(t) - \Delta_m^A X_m - X_m \Delta_m^D,\end{aligned}$$

where $\Delta_m^A = V_{m+1} T_{m+1,m}^A V_m^T$, $\Delta_m^D = W_m T_{m+1,m}^D W_m^T$, $\mathcal{E}_m(t) = X(t) - X_m(t)$, and X is an exact solution of (1).

Proof. The proof can be easily obtained from the relation (8) and the expressions of the residual $R_m(t)$ and the initial Equation (1). \square

Remark that $\|\Delta_m^A\| = \|T_{m+1,m}^A\|$ and $\|\Delta_m^D\| = \|T_{m+1,m}^D\|$, which shows that these two quantities tend to 0 as m increases because $\|T_{m+1,m}\|$ goes to zero as m increases.

The matrix associated to the first nonsymmetric differential equation in Theorem 4 is given by

$$\mathcal{L}_m = \begin{pmatrix} D - \Delta_m^D & -S \\ -FG^T & A - \Delta_m^A \end{pmatrix}, \quad (14)$$

also expressed as

$$\mathcal{L}_m = \begin{pmatrix} D & -S \\ -FG^T & A \end{pmatrix} - \begin{pmatrix} \Delta_m^D & 0 \\ 0 & \Delta_m^A \end{pmatrix}.$$

This shows that the matrix \mathcal{L}_m could be considered as a perturbation of the matrix \mathcal{L} associated to the initial problem (1). Notice that when $X_m(t)$ converges to $X(t)$ as m increases, $R_m(t) = \Delta_m^A X_m + X_m \Delta_m^D$ goes to zero and then $\|\Delta_m^A\|$ and $\|\Delta_m^D\|$ tend to zero, which shows that the matrix \mathcal{L}_m converges to the matrix \mathcal{L} .

Let us come back to the NDRE equation of the error $\mathcal{E}_m(t)$ from Theorem 4

$$\dot{\mathcal{E}}_m(t) = -A_c \mathcal{E}_m(t) - \mathcal{E}_m(t) D_c + \mathcal{M}(t, \mathcal{E}_m(t)), \quad (15)$$

where for some matrix P the operator $\mathcal{M}(t, P)$ is defined by

$$\mathcal{M}(t, P) := P(t) S P(t) - \Delta_m^A X_m - X_m \Delta_m^D, \quad (16)$$

and $A_c = A - X_m S$, $D_c = D - S X_m$, $\Delta_m^A = V_{m+1} T_{m+1,m}^A V_m^T$, $\Delta_m^D = W_m T_{m+1,m}^D W_m^T$.

For the error \mathcal{E}_m from Equation (15), the following nonlocal bound is valid:

Theorem 5. Let $\Phi_P(t, t_0)$ be the fundamental matrix for the equation $\dot{\eta}(t) = P \eta(t)$ for some real matrix P .

Denote

$$v = \max \left\{ \int_0^t \|\Phi_{A_c}(t, \tau)\| \|\Phi_{D_c}(\tau, t)\| d\tau, t \in T \right\}, \quad (17)$$

$$\kappa = \max \{ \|\Phi_{A_c}(t, 0)\| \|\Phi_{D_c}(0, t)\| : t \in T \}, \quad (18)$$

and

$$a_0 = v \|S\|; \quad a_1 = v \|X_m\| (\|\Delta_m^A\| + \|\Delta_m^D\|) + \kappa \|\mathcal{E}_m(0)\|. \quad (19)$$

Then, for the spectral norm $\|\mathcal{E}_m\|$ of the error $\mathcal{E}_m = X - X_m$, the nonlocal bound

$$\|\mathcal{E}_m\| \leq \rho = \frac{2a_1}{1 + \sqrt{1 - 4a_0 a_1}} \quad (20)$$

is valid whenever

$$\delta := \{\|\Delta_m^A\|, \|\Delta_m^D\|\} \in \Omega := \{a_0 a_1 \leq 0.25\}. \quad (21)$$

Proof. Define the operator $\mathcal{L}(P)$

$$\mathcal{L}(P) := \int_0^t \Phi_{A_c}(t)\Phi_{A_c}^{-1}(\tau)P\Phi_{D_c}^{-1}(\tau)\Phi_{D_c}(t)d\tau \quad (22)$$

with matrix

$$Mat(\mathcal{L}) := L := \int_0^t \left[\Phi_{D_c}^{-1}(\tau)\Phi_{D_c}(t) \right]^\top \otimes \left[\Phi_{A_c}(t)\Phi_{A_c}^{-1}(\tau) \right] d\tau,$$

and rewrite expression (15) in operator form

$$\dot{\mathcal{E}}_m(t) = \Pi(\mathcal{E}_m)(t) \quad (23)$$

with

$$\Pi(\mathcal{E}_m)(t) := \Phi_{A_c}(t, 0)\mathcal{E}_m(0)\Phi_{D_c}(t, 0) - \int_0^t \Phi_{A_c}(t, \tau)\mathcal{M}(\tau, \mathcal{E}_m(\tau))\Phi_{D_c}(\tau, t)d\tau \quad (24)$$

$$= \Phi_{A_c}(t, 0)\mathcal{E}_m(0)\Phi_{D_c}(t, 0) + \mathcal{L}(-\Delta_m^A X_m - X_m \Delta_m^D) + \mathcal{L}(\mathcal{E}_m S \mathcal{E}_m). \quad (25)$$

Using (16), we get

$$\|\mathcal{M}(t, P)\| \leq \|P\|^2 \|S\| + \|X_m\| (\|\Delta_m^A\| + \|\Delta_m^D\|).$$

The Lyapunov majorant for the operator $\Pi(\cdot)$ (24) such that $\|\Pi(\mathcal{E}_m)(t)\| < h(\|\mathcal{E}_m\|)$ is

$$\|\Pi(\mathcal{E}_m)(t)\| \leq h(\|\mathcal{E}_m\|) := a_1 + a_0 \|\mathcal{E}_m\|^2 \quad (26)$$

with a_0, a_1 given in (19).

In similar way, for some P and Y , we get

$$\|\Pi(P)(t) - \Pi(Y)(t)\| \leq h'(r)\|P - Y\| = 2a_0 r \|P - Y\|, \quad (27)$$

where $r = \max\{\|P\|, \|Y\|\}$.

Assume that there exists a number $\rho > 0$, such that

$$h(\rho) \leq \rho, \text{ and } h'(\rho) < 1. \quad (28)$$

Denote by M_ρ , the set of continuous matrix valued functions $P : T \rightarrow \mathbb{R}^{n \times p}$ and $\|P\| \leq \rho$. Then, from (26)–(28), it follows that the operator $\Pi(\cdot)$ is a contraction on M_ρ and maps this set into itself. Hence, there is a solution $\mathcal{E}_m(t)$ of the operator Equation (23) such that for

$$\begin{aligned} \delta &:= \{\|\Delta_m^A\|, \|\Delta_m^D\|\} \in \Omega := \{a_0 a_1 \leq 0.25\} \\ \|\mathcal{E}_m\| \leq \rho &:= \frac{2a_1}{1 + \sqrt{1 - 4a_0 a_1}}. \end{aligned}$$

In what follows, the theorem is proven. \square

Using the property of the logarithmic norm, the estimates (17), (18) of the numbers v and κ take the form

$$\|\Phi_{A_c}(\tau, 0)\| \leq \exp \left[\int_0^\tau \lambda(A_c(r))dr \right] \leq \exp \left[\int_0^\tau \lambda_+(A_c(r))dr \right] \quad (29)$$

$$\|\Phi_{D_c}(\tau, 0)\| \leq \exp \left[\int_0^\tau \xi(D_c(r))dr \right] \leq \exp \left[\int_0^\tau \xi_+(D_c(r))dr \right], \quad (30)$$

where

$$\begin{aligned} \lambda(t) &= 0, 5\lambda_{\max} [A_c(t) + A_c(t)^\top], \\ \xi(t) &= 0, 5\xi_{\max} [D_c(t) + D_c(t)^\top], \end{aligned}$$

Power series	Log norm	Jordan (1)	Jordan (2)	Schur
c_0	1	1	$\text{cond}(Y)$	1
ϱ	$\ P(t)\ $	$\mu(P(t))$	$\alpha(P(t))$	$\alpha(P(t)) + d_\varsigma$
ϖ	0	0	1	0
p	-	-	m	l

TABLE 1 Bounds for the matrix exponential $e^{P(t)}$

are the logarithmic norms of the matrices $A_c = A - X_m S$ and $D_c = D - SX_m$, respectively. And

$$\nu \leq \nu_1 \leq \nu_2$$

$$\kappa \leq \kappa_1 \leq \kappa_2$$

with

$$\begin{aligned} \nu_1 &= \max \left\{ \int_0^t \exp \left[\int_0^r (\lambda(\tau) + \xi(\tau)) d\tau \right] dr : t \in T \right\} \\ \nu_2 &= \int_0^t \exp \left[\int_0^r (\lambda_+(\tau) + \xi_+(\tau)) d\tau \right] dr, \\ \kappa_1 &= \exp \left[\max \left\{ \int_0^t (\lambda(\tau) + \xi(\tau)) d\tau : t \in T \right\} \right]. \\ \kappa_2 &= \exp \left[\int_0^t (\lambda_+(\tau) + \xi_+(\tau)) d\tau \right], \\ \lambda_+(t) &= \begin{cases} \lambda(t), & \lambda(t) > 0 \\ 0, & \lambda(t) \leq 0 \end{cases} \quad \xi_+(t) = \begin{cases} \xi(t), & \xi(t) > 0 \\ 0, & \xi(t) \leq 0 \end{cases}. \end{aligned}$$

In order to obtain an explicit bound for the norm of the fundamental matrix $\|\Phi_P(t)\|$ for $P(t) = A_c(t)$ or $D_c(t)$, we can use also the known bounds for the matrix exponential $e^{P(t)}$ based on power series, logarithmic norm, and matrix decomposition. Some bounds for the matrix exponential $e^{P(t)}$ are summarized in the work of Petkov²⁰

$$\|e^{P(t)}\| \leq g(t) = c_0 e^{\varrho t} \sum_{k=0}^{p-1} (\varpi t)^k / k!. \quad (31)$$

with constants c_0 , ϱ , ϖ , and p , listed in Table 1.

Here, $\mu(P(t))$ is the maximum eigenvalue of the matrix $(P(t) + P(t)^\top)/2$, $J = Y^{-1}P(t)Y$ is the Jordan canonical form of $P(t)$, and $\varsigma \geq 1$ is the dimension of the maximum block in J (the matrix Y is chosen so that the condition number $\text{cond}(Y) = \|Y\| \|Y^{-1}\|$ is minimized), $d_\varsigma = \cos\left(\frac{\pi}{\varsigma+1}\right)$, $\alpha(P(t))$ is the spectral abscissa of $P(t)$, that is, the maximum real part of the eigenvalues of $P(t)$, and $T = U^H P(t) U = \Lambda + \mathcal{N}$ is the Schur decomposition of $P(t)$, where U is unitary, Λ is diagonal, and \mathcal{N} is strictly upper triangular matrix (the matrix U is chosen so that the norm of the matrix \mathcal{N} is minimized), $l = \min\{\varphi : \mathcal{N}^\varphi = 0\}$ is the index of nilpotency of \mathcal{N} , and $\varpi = \|\mathcal{N}\|$.

3.2 | Solving the projected problem using the exponential-matrix of the low-dimensional problem

Let us see now how to solve the projected low-dimensional nonsymmetric differential Riccati Equation (11), which is related to the initial value problem

$$\begin{pmatrix} \dot{Y}_{1,m}(t) \\ \dot{Y}_{2,m}(t) \end{pmatrix} = \begin{pmatrix} \mathcal{T}_m^D & -S_m \\ F_m G_m^T & -\mathcal{T}_m^A \end{pmatrix} \begin{pmatrix} Y_{1,m}(t) \\ Y_{2,m}(t) \end{pmatrix}, \quad Y_{1,m}(0) = I \text{ and } Y_{2,m}(0) = Y_0. \quad (32)$$

Notice that if we set

$$\mathcal{H}_m = \begin{pmatrix} \mathcal{T}_m^D & -S_m \\ F_m G_m^T & -\mathcal{T}_m^A \end{pmatrix}, \quad \mathcal{H} = \begin{pmatrix} D & -S \\ FG^T & -A \end{pmatrix} \text{ and } \mathcal{U}_m = \begin{pmatrix} \mathcal{W}_m & 0 \\ 0 & \mathcal{V}_m \end{pmatrix}, \quad (33)$$

we get the following relation

$$\mathcal{H}_m = \mathcal{U}_m^T \mathcal{H} \mathcal{U}_m \text{ with } \mathcal{U}_m^T \mathcal{U}_m = I.$$

The solution of the projected linear differential system (32) is given as

$$\begin{pmatrix} Y_{1,m}(t) \\ Y_{2,m}(t) \end{pmatrix} = e^{t\mathcal{H}_m} Z_0 \text{ with } Z_0 = \begin{pmatrix} I \\ Y_0 \end{pmatrix}. \quad (34)$$

As in general m is small, the solution given by (34) can be obtained from Padé approximants implemented in MATLAB as `expm`. The solution Y_m of the projected nonsymmetric differential Riccati Equation (32) is then given as

$$Y_m(t) = Y_{1,m}(t) Y_{2,m}^{-1}(t), \quad (35)$$

provided that $Y_{2,m}(t)$ is nonsingular and then the approximate solution to the initial problem (1) is defined by $X_m = \mathcal{V}_m Y_m \mathcal{W}_m^T$.

Another way of getting approximate solutions is to use directly an approximation of $e^{t\mathcal{H}} Z_0$ as it appears in (7). Using the matrices \mathcal{U}_m and \mathcal{H}_m given in (33), we propose the following approximation:

$$e^{t\mathcal{H}} Z_0 \approx \mathcal{U}_m e^{t\mathcal{H}_m} \Gamma_m, \text{ with } \Gamma_m = \mathcal{U}_m^T Z_0. \quad (36)$$

Therefore, setting

$$\begin{pmatrix} X_{1,m}(t) \\ X_{2,m}(t) \end{pmatrix} = \mathcal{U}_m e^{t\mathcal{H}_m} \Gamma_m,$$

the approximate solution of the solution X of (1) is given as

$$\tilde{X}_m = X_{1,m}(t) X_{2,m}^{-1}(t).$$

Instead of solving the low-dimensional nonsymmetric differential Riccati Equation (11) by using the exponential scheme (34), we can use an integration scheme for solving ordinary differential equations such as Rosenbrock²¹ or BDF methods.^{22,23} That is the subject of the following two sections.

3.3 | Using the BDF integration scheme

At each timestep t_k , the approximate $Y_{m,k}$ of the $Y_m(t_k)$, where Y_m is the solution to (11) is then computed solving a NARE. We consider the problem (11) and apply the s -step BDF method. At each iteration $k+1$ of the BDF method, the approximation $Y_{m,k+1}$ of $Y_m(t_{k+1})$ is given by the implicit relation

$$Y_{m,k+1} = \sum_{i=0}^{s-1} \alpha_i Y_{m,k-i} + h\beta \mathcal{F}_m(Y_{m,k+1}), \quad (37)$$

where $h = t_{k+1} - t_k$ is the step size, α_i , and β are the coefficients of the BDF method as listed in Table 2 and $\mathcal{F}_m(X)$ is given by

$$\mathcal{F}_m(Y) = -\mathcal{T}_m^A Y - Y \mathcal{T}_m^D + Y S_m Y + F_m G_m^T.$$

TABLE 2 Coefficients of the s -step backward differentiation formula (BDF) method with $q \leq 3$

s	β	α_0	α_1	α_2
1	1	1		
2	2/3	4/3	-1/3	
3	6/11	18/11	-9/11	2/11

The approximate X_{k+1} solves the following matrix equation:

$$-Y_{m,k+1} + h\beta (F_m G_m^T - \mathcal{T}_m^A Y_{m,k+1} - Y_{k+1} \mathcal{T}_m^D + Y_{m,k+1} S_m Y_{m,k+1}) + \sum_{i=0}^{p-1} \alpha_i Y_{m,k-i} = 0,$$

which can be written as the following continuous-time NARE

$$\mathcal{A}_m Y_{m,k+1} + Y_{m,k+1} \mathcal{D}_m - Y_{m,k+1} \mathcal{S}_m Y_{m,k+1} - \mathcal{L}_{k+1} \mathcal{G}_{k+1}^T = 0, \quad (38)$$

where, assuming that at each timestep, $Y_{m,k}$ can be approximated as a product of low-rank factors $Y_{m,k} \approx Z_{m,k} \tilde{Z}_{m,k}^T$. The coefficient matrices are given by

$$\begin{aligned} \mathcal{A}_m &= \frac{1}{2}I + h\beta \mathcal{T}_m^A, \quad \mathcal{D}_m = \frac{1}{2}I + h\beta \mathcal{T}_m^D, \quad \mathcal{S}_m = h\beta S_m, \\ \mathcal{L}_{k+1,m} &= [h\beta F_m, \alpha_0 Z_{m,k}, \alpha_1 Z_{m,k-1}, \dots, \alpha_{q-1} Z_{m,k-p+1}], \end{aligned}$$

and

$$\mathcal{G}_{k+1,m} = [G_m, \tilde{Z}_{m,k} \tilde{Z}_{m,k-1}^T, \dots, \tilde{Z}_{m,k-p+1}].$$

We assume that at each step $k+1$, Equation (38) has a solution.

3.4 | Solving the low-dimensional problem with the Rosenbrock method

Applying the two-stage Rosenbrock method^{21,24} to the low-dimensional nonsymmetric differential Riccati Equation (11), the new approximation $Y_{m,k+1}$ of $Y_m(t_{k+1})$ obtained at step $k+1$ is defined by the relations, (see the work of Benner et al.²⁵ for more details)

$$Y_{m,k+1} = Y_{m,k} + \frac{3}{2}H_1 + \frac{1}{2}H_2, \quad (39)$$

where H_1 and H_2 solve the following Sylvester equations:

$$\tilde{\mathbb{T}}_m^A H_1 + H_1 \tilde{\mathbb{T}}_m^D = -\mathcal{F}(Y_{m,k}), \quad (40)$$

$$\tilde{\mathbb{T}}_m^A H_2 + H_2 \tilde{\mathbb{T}}_m^D = -\mathcal{F}(Y_{m,k} + H_1) + \frac{2}{h}H_1, \quad (41)$$

where

$$\tilde{\mathbb{T}}_m^A = \mathcal{T}_m^D - \frac{1}{2h}I \quad \text{and} \quad \tilde{\mathbb{T}}_m^D = \mathcal{T}_m^D - \frac{1}{2h}I,$$

and

$$\mathcal{F}(Y) = -\mathcal{T}_m^A Y - Y \mathcal{T}_m^D + Y S_m Y + F_m G_m^T.$$

The Sylvester matrix Equations (40) and (41) can be solved, for small to medium problems, by direct methods such as the Bartels–Stewart algorithm.²⁶

The different steps of the EBA algorithm for solving NDREs are summarized in the following algorithm:

Algorithm 1 [The extended block Arnoldi algorithm for NDREs (EBA–NDRE)]

- **Inputs.** Matrices A, D, S, F, G , and an integer m .
 - **Outputs:** The approximate solution in a factored form: $X_m(t) \approx Z_{m,1}(t) Z_{m,2}^T(t)$.
 - Compute the QR decompositions of $[F, A^{-1}F] = V_1 \Lambda_1$ and $[G, D^{-1}G] = W_1 \Lambda_2$.
 - Apply the extended block Arnoldi to the pair (A, F) :
 - For $j = 1, \dots, m$
 - Set $V_j^{(1)}$: first s columns of V_j ; $V_j^{(2)}$: second s columns of V_j
 - $\mathcal{V}_j = [\mathcal{V}_{j-1}, V_j]$; $\hat{V}_{j+1} = \begin{bmatrix} A V_j^{(1)}, A^{-1} V_j^{(2)} \end{bmatrix}$.
 - Orthogonalize \hat{V}_{j+1} w.r. to \mathcal{V}_j to get V_{j+1} , that is,
 - * for $i = 1, 2, \dots, j$
 - * $H_{i,j}^A = V_i^T \hat{V}_{j+1}$,
 - * $\hat{V}_{j+1} = \hat{V}_{j+1} - V_i H_{i,j}^A$,
 - * end for
 - Compute the QR decomposition of \hat{V}_{j+1} , i.e., $\hat{V}_{j+1} = V_{j+1} H_{j+1,j}^A$.
 - end for.
 - Apply also the extended Arnoldi process to the pair (D, G) to get the blocks W_1, \dots, W_{m+1} and the upper Hessenberg matrix whose elements are $H_{i,j}^D$.
 - Solve the projected NDRE (11) to get $Y_m(t)$ using the exponential technique, BDF or Rosenbrock method.
 - The approximate solution $X_m(t)$ is given by the expression (3.1).
-

4 | THE BDF-NEWTON METHOD

In this section, we apply directly the BDF integration scheme to the initial problem (1). Then, each timestep t_k , the approximate X_k of the $X(t_k)$, is then computed solving a NARE. Applying the s -step BDF method, the approximation X_{k+1} of $X(t_{k+1})$ is given by the implicit relation

$$X_{k+1} = \sum_{i=0}^{s-1} \alpha_i X_{k-i} + h\beta \mathcal{F}(X_{k+1}), \quad (42)$$

where $h = t_{k+1} - t_k$ is the step size, α_i and β are the coefficients of the BDF method as listed in Table 2 and $\mathcal{F}_m(X)$ is given by

$$\mathcal{F}(X) = -AX - XD + XSX + FG^T.$$

The approximate X_{k+1} solves the following matrix equation

$$-X_{k+1} + h\beta(FG^T - AX_{k+1} - X_{k+1}D + X_{k+1}SX_{k+1}) + \sum_{i=0}^{s-1} \alpha_i X_{k-i} = 0,$$

which can be written as the following continuous-time algebraic Riccati equation

$$\mathcal{G}(X_{k+1}) = -AX_{k+1} - X_{k+1}D + X_{k+1}SX_{k+1} + \tilde{F}_{k+1}^T \tilde{G}_{k+1} = 0, \quad (43)$$

where, assuming that at each timestep, X_k can be approximated as a product of low-rank factors $X_k \approx Z_{k,1}Z_{k,2}^T$, $Z_{k,i} \in \mathbb{R}^{n \times m_k}$, with $m_k \ll n, p$. The coefficients matrices are given by

$$\begin{aligned} \mathcal{A} &= h\beta A + \frac{1}{2}I, \quad \mathcal{D} = h\beta D + \frac{1}{2}I, \quad \mathcal{S} = h\beta S \\ \tilde{G}_{k+1} &= \left[\sqrt{h\beta}G, \sqrt{\alpha_0}Z_{k,1}^T, \dots, \sqrt{\alpha_{s-1}}Z_{k+1-s,1}^T \right], \end{aligned}$$

and

$$\tilde{F}_{k+1} = \left[\sqrt{h\beta} F, \sqrt{\alpha_0} Z_{k,2}^T, \dots, \sqrt{\alpha_{p-1}} Z_{k+1-s,2}^T \right]^T.$$

For large-scale problems, a common strategy of solving the NARE (43) consists in applying the Newton method combined with an iterative method for the numerical solution of the large-scale Sylvester equations arising at each internal iteration of the Newton's algorithm. In that case, we define a sequence of approximations to X_{k+1} as follows:

- Set $X_{k+1}^0 = X_k$
- Build the sequence $(X_{k+1}^l)_{l \in \mathbb{N}}$ defined by

$$X_{k+1}^{l+1} = X_{k+1}^l - D\mathcal{G}_{X_{k+1}^l}(\mathcal{G}(X_{k+1}^l)), \quad (44)$$

where the Fréchet derivative $D\mathcal{G}$ of \mathcal{G} at X_{k+1}^l is given by

$$D\mathcal{G}_{X_{k+1}^l}(H) = (\mathcal{A} - X_{k+1}^l S) H + H (\mathcal{D} - S X_{k+1}^l). \quad (45)$$

A straightforward calculation proves that X_{k+1}^{l+1} is the solution to the Sylvester equation

$$(\mathcal{A} - X_{k+1}^l S) X + X (\mathcal{D} - S X_{k+1}^l) + X_{k+1}^l S X_{k+1}^l + \tilde{F}_{k+1} \tilde{G}_{k+1}^T = 0. \quad (46)$$

The main part in each Newton iteration is to solve a large Sylvester matrix equation with a low-rank right-hand side. For small to medium problems, one can use direct methods such as the Bartels–Stewart algorithm.²⁶ For large problems, many numerical methods have been proposed; see other works.^{14,27–30} In our computations, we used the EBA algorithm for solving the large Sylvester

matrix Equation (46). The method is defined as follows: We first apply the EBA (or the block Arnoldi) to the pairs $(\mathcal{A}_k, \tilde{F}_{k+1})$ and $(\mathcal{D}_k^T, \tilde{G}_{k+1})$, where

$$\mathcal{A}_k = \mathcal{A} - X_{k+1}^l S, \text{ and } \mathcal{D}_k = \mathcal{D} - S X_{k+1}^l$$

and obtain a low-rank approximate solution to the exact solution X_{k+1}^{l+1} .

Because \mathcal{A} and \mathcal{D} are sparse, the matrices \mathcal{A}_k and \mathcal{D}_k are no longer sparse and then the computation of the products $\mathcal{A}_k^{-1} Y$ and $\mathcal{D}_k^{-T} Y$ becomes very expensive. A way to overcome this drawback is to use the Sherman–Morrison–Woodbury formula given by

$$(L + UV^T)^{-1} Y = L^{-1} Y - L^{-1} U (I + V^T L^{-1} U) V^T L^{-1} Y, \quad (47)$$

where L , U , and V are matrices of adequate sizes.

Notice that, if we use the block Arnoldi method²⁷ to solve the Sylvester matrix Equation (46), then only matrix-block vectors products are needed.

5 | APPLICATIONS TO NDRES FROM TRANSPORT THEORY

Nonsymmetric differential Riccati equations (1) associated with M-matrices appear for example in neutron transport theory; see other works.^{1,31,32} The problem to be solved is given as follows:

$$\dot{X}(t) = -(\Delta - eq^T)X - X(\Gamma - qe^T) + Xqq^TX + ee^T. \quad (48)$$

The matrices Δ and Γ involved in the NDRE (48), obtained by a discretization of a integro-differential equation describing neutron transport during a collision, see the work of Huang et al.³³ for more details on the physics, have the same dimension and are given by

$$\Delta = \text{diag}(\delta_1, \dots, \delta_n), \quad \Gamma = \text{diag}(\gamma_1, \dots, \gamma_n), \quad (49)$$

with

$$\delta_i = \frac{1}{c\omega_i(1+\alpha)}, \text{ and } \gamma_i = \frac{1}{c\omega_i(1-\alpha)}, i = 1, \dots, n. \quad (50)$$

Vectors e and q are given as follows:

$$e = (1, \dots, 1)^T, \quad q = (q_1, \dots, q_n)^T \text{ with } q_i = \frac{c_i}{2\omega_i}, i = 1, \dots, n. \quad (51)$$

The matrices and vectors above depend on the two parameters $0 \leq c \leq 1$, which denote the average ratio of the total number of particles emerging from a collision, the angular shift $0 \leq \alpha < 1$ and on the sequences (ω_i) and (c_i) , $i = 1, \dots, n$, which are the nodes and weights of the Gaussian–Legendre quadrature on $[0, 1]$, respectively. They are such that

$$0 < \omega_n < \dots < \omega_1 < 1, \text{ and } \sum_{i=1}^n c_i = 1, \quad c_i > 0, \quad i = 1, \dots, n.$$

The steady-state solutions of (48) satisfy the following NARE:

$$-(\Delta - eq^T)X - X(\Gamma - qe^T) + Xqq^TX + ee^T = 0. \quad (52)$$

For existence of solutions for NAREs (52), we have the following result:

Theorem 6 (See the work of Juang and Lin³⁴). *If $c = 1$ and $\alpha = 0$, Equation (52) has unique nonnegative solution. Otherwise, it has two nonnegative minimal and maximal solutions, say X_{\min} and X_{\max} with $X_{\max} > X_{\min} > 0$. The minimal solution X_{\min} is strictly increasing in c for a fixed α and decreasing in α for fixed c .*

Equation (48) can be expressed as follows:

$$\dot{X}(t) + \Delta X + X\Gamma = eq^T X + qe^T + Xqq^TX + ee^T. \quad (53)$$

Therefore, integrating (53), we get the following expression of a solution of (48).

$$X(t) = e^{-t\Delta}X_0e^{-t\Gamma} + \int_0^t e^{-(t-\tau)\Delta} [ee^T + eq^T X(\tau) + X(\tau)qe^T + X(\tau)qq^TX(\tau)] e^{-(t-\tau)\Delta} d\tau.$$

The global existence of a solution of Equation (48) was investigated in other works,^{4,17} and this is stated in the following theorem:

Theorem 7 (See the work of Juang¹⁷). *Let $0 < c \leq 1$, $0 \leq \alpha < 1$. Assume that $0 \leq X_0 \leq X_{\min}$ and $ee^T - \Delta X_0 - X_0\Gamma \geq 0$. Then a global solution $X(t)$ of (48) exists and is nondecreasing in t on $[0, \infty[$. Furthermore,*

$$\lim_{t \rightarrow \infty} X(t) = X_{\min},$$

where X_{\min} is the minimal solution of the NARE (52).

To obtain low-rank approximate solutions to (48), we first apply the extended Arnoldi process to the pairs (A, e) and (D, e) , where $A = \Delta - eq^T$ and $D = \Gamma - qe^T$ to get orthonormal bases that will be used to construct the desired low-rank approximation $X_m(t) = \mathcal{V}_m Y_m(t) \mathcal{W}_m^T$, where Y_m solves the low dimensional differential Riccati Equation (11). We notice that when applying the above method, we use matrix vector operations of the form $A^{-1}v$ and $D^{-1}v$. As the matrices A and D are the sum of diagonal matrices and rank one matrices, then to reduce the costs, we can compute easily these quantities by using the Sherman–Morrison–Woodbury formula given by

$$A^{-1}v = (\Delta - eq^T)^{-1}v = \Delta^{-1}v + \frac{\Delta^{-1}eq^T \Delta^{-1}v}{1 - q^T \Delta^{-1}e},$$

and a similar relation for $D^{-1}v$.

6 | NUMERICAL EXAMPLES

The experimental tests reported in this section illustrate the methods introduced in this work. We considered the differential nonsymmetric Riccati equation applied to transport theory (48) on a time interval $[t_0, t_f]$, for different values of the parameters α and c and for several sizes. The initial condition was chosen as $X_0 = Z_{0,1}Z_{0,2}^T$, where $Z_{0,1} = Z_{0,2} = O_{n \times 1}$. All the tests were performed on an Intel Core i7 processor laptop equipped with 8GB of RAM. The algorithms were coded in MATLAB R2014b. The three considered methods in this work are

- The BDF-BA-Newton method, which is based on the application of a BDF(s) integration scheme to the original equation that implies, at each timestep, the resolution of the algebraic nonsymmetric Riccati Equation (38). The latter equation is then solved by the Newton method. The numerical resolution of the Sylvester equations that need to be solved at each iteration of the Newton method is done by a block Arnoldi method, as the coefficient matrices can be singular or ill-conditioned, impeding the use of the EBA algorithm.
- The EBA-BDF(s) and EBA-exp methods, which consist in projecting the differential problem onto an extended Arnoldi subspace and then solve the projected nonsymmetric differential Riccati equation by a BDF method (EBA-BDF(s) method) or using the exponential method by a quadrature method as described in Section 3.2 (EBA-exp). The alternative consisting in using a Rosenbrock method instead of the BDF scheme was not useful in our examples as it did not perform better than the BDF1. The Frobenius norm of the residual at final time is then computed and while the tolerance is not met, we repeat the process increasing the dimension of the projection subspace. The computation of the exponential form of the solution is known for being regarding the EBA-exp method, the Davison Maki algorithm is known to be numerically unstable and we had to use the modified Davison–Maki method to overcome this drawback, see the work of Davison et al.³⁵ for more details.

For the EBA algorithm, the stopping criterion was

$$\|R(X_m)\|_F / \|FG^T\|_F < 10^{-10},$$

where the norm of the residual $\|R_m(t_f)\|$ was computed by using Theorem 3. For the Newton-block Arnoldi, the iterations were stopped when

$$\|X_{k+1} - X_k\|_F / \|X_k\|_F < 10^{-10}.$$

Example 1. In order to confirm that the numerical methods presented in this work produce reliable approximations, we compared their outputs to the solution $X_{1,1}^{direct}(t)$ computed by the direct exponential method as described in Section 2, (6). As this direct approach is not suitable for large-sized problems, we set the dimension of the problem to $n = 40$. The choice of the parameter values was $c = 0.5$ and $\alpha = 0.5$. In Figure 1, we plotted the curves of the first component $X_{11}(t)$ for EBA-BDF1 and for the direct exponential method on the time interval $[0, 10]$.

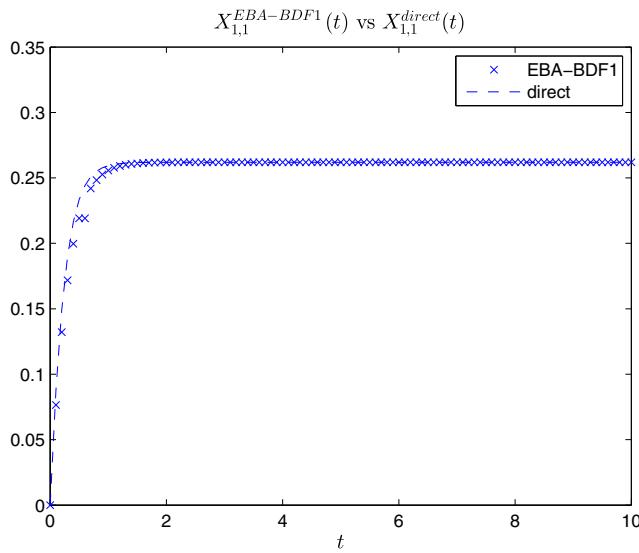


FIGURE 1 First components $X_{11}(t)$, $t \in [0, 10]$

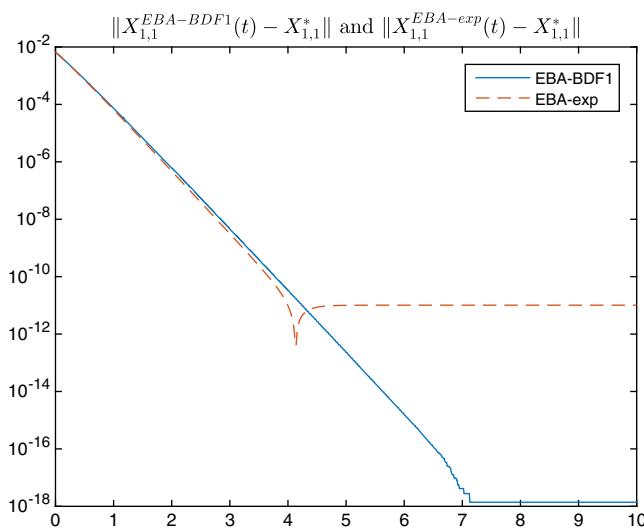


FIGURE 2 Errors, corresponding to the first coefficient

Figure 2 shows that the solution of the DNRE tends to the minimal nonnegative solution X^* of the algebraic nonsymmetric Equation (2) associated to (1) when t tends to infinity. In this figure, we plotted the errors $\|X_{11}^{EBA-BDF1} - X_{11}^*\|$ and $\|X_{11}^{EBA-exp} - X_{11}^*\|$ corresponding the first coefficients.

Example 2. For this example, we set $c = 0.5$ and $\alpha = 0.5$. We first computed the approximations $X_{EBA-BDF1}(t)$, $X_{EBA-exp}(t)$ and $X_{BDF1-BA-n}(t)$ given by the EBA-BDF1, EBA-exp and BDF1-Newton-BA methods for the size $n = 1000$, on the time interval $[0, 1]$, for a timestep $dt = 0.01$ for the BDF1 integration scheme. The relative Frobenius error norms at final time $t_f = 1$ were of order 10^{-10} between the results of EBA-BDF1 and BDF1-BA-Newton methods whereas the EBA-exp did not perform as well with a relative error of order 10^{-4} when compared to both EBA-BDF1 and BDF1-BA-Newton methods. This problem was expected as the modified Davison-Maki requires a large number of steps in order to converge, leading to some loss of accuracy.

We considered problems with the following sizes $n = 4,000$; $n = 10,000$; $n = 20,000$; and $n = 40,000$. In Table 3, we listed the obtained relative residual norms (Res.) at final time for each method and the corresponding CPU time (in seconds). For all the tests, the outer iterations in the Newton method did not exceed 10 iterations. The maximum number of inner iterations was $itermax = 50$ and were stopped when the corresponding residual was less than $tol = 10^{-12}$. In order to spare some computation time, the BDF1 or exponential method was performed every 5 Arnoldi iterations.

The results in Table 3 show that the EBA-BDF1 method performs better than the other approaches, although all achieved satisfactory accuracies even though the EBA-exp method was not as interesting from a practical point of view. This is probably caused by the fact that the modified Davison-Maki algorithm needed a large number of substeps in order to converge (1,000 substeps for the $n = 4,000$ case). As the number of substeps increases with the size of the problem, the EBA-exp could not handle the largest cases of this example.

Example 3. In this example, we repeated the tests of Example 2, for $c = 0.9999$ and $\alpha = 10^{-8}$. As in the previous example, the results showed a clear advantage for the methods based on the EBA algorithm, which are well designed for this problem. Indeed, the computations of the inverses of the matrices A and D (and the forms derived from the application of the BDF integration scheme) do not require important computational efforts.

TABLE 3 Results for the transport case $c = 0.5$ and $\alpha = 0.5$

<i>n</i>	EBA-BDF1		EBA-Exp		BDF1-Newton-BA	
	Res.	time	Res.	time	Res.	time
4,000	$3.9 \cdot 10^{-9}$	2.9s	$4.7 \cdot 10^{-8}$	186s	$3.9 \cdot 10^{-9}$	1293.4s
10,000	$1.1 \cdot 10^{-8}$	4.4s	$1.1 \cdot 10^{-8}$	330s	--	-- s
20,000	$2.4 \cdot 10^{-8}$	7.6s	--	-- s	--	-- s
40,000	$2.3 \cdot 10^{-8}$	12.8s	--	-- s	--	-- s

Note. EBA = extended block Arnoldi; BDF = backward differentiation formula; BA = block Arnoldi; Res. = residual.

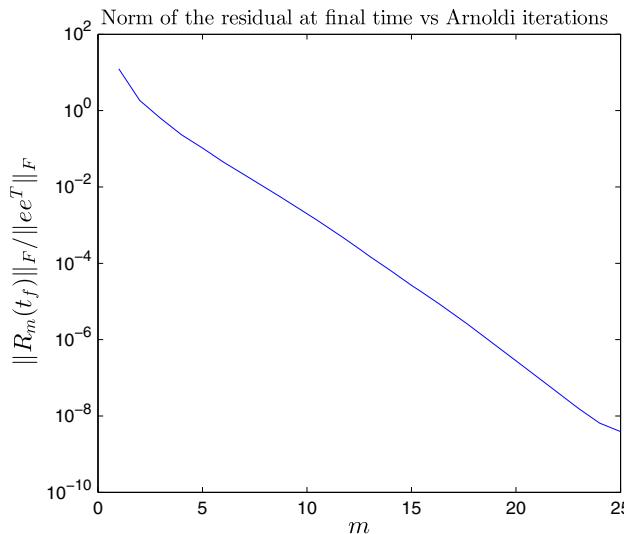


FIGURE 3 Relatives Frobenius residual norms vs the number of extended block Arnoldi (EBA) iterations m

n	EBA-BDF1		EBA-Exp		BDF1-Newton-BA	
	Res.	time	Res.	time	Res.	time
4,000	$3.6 \cdot 10^{-9}$	3.4s	$5.7 \cdot 10^{-8}$	183s	$3.9 \cdot 10^{-9}$	1204.1s
10,000	$8.1 \cdot 10^{-9}$	5.5s	$4.1 \cdot 10^{-8}$	341s	--	-- s
20,000	$2.2 \cdot 10^{-9}$	8.9s	--	-- s	--	-- s
40,000	$2.3 \cdot 10^{-9}$	14.9s	--	-- s	--	-- s

Note. EBA = extended block Arnoldi; BDF = backward differentiation formula; BA = block Arnoldi; Res. = residual.

TABLE 4 Results for the transport case $c = 0.9999$ and $\alpha = 10^{-8}$

n, p	EBA-BDF1		EBA-Exp	
	Res.	time	Res.	time
$n = p = 500$	$7.2 \cdot 10^{-10}$	0.18s	$8.5 \cdot 10^{-10}$	0.08s
$n = p = 5,000$	$3.4 \cdot 10^{-9}$	4.2s	$3.6 \cdot 10^{-9}$	3.9s
$n = p = 10,000$	$8.6 \cdot 10^{-9}$	20.0s	$3.9 \cdot 10^{-9}$	18.5s

Note. EBA = extended block Arnoldi; BDF = backward differentiation formula; Res. = residual.

TABLE 5 Results for Example 4

$$A = D = \begin{pmatrix} 2 & -1 & & \\ & 2 & \ddots & \\ & & \ddots & -1 \\ -1 & & & 2 \end{pmatrix} \text{ and } S = \text{diag}(1, 1, 0, \dots, 0) \in \mathbb{R}^{n \times n}$$

The coefficients of matrices $F \in \mathbb{R}^{n \times 2}$ and $G \in \mathbb{R}^{n \times 2}$ were randomly generated. In Table 5, we reported the obtained residual norms and the CPU times for the EBA-BDF1 and EBA-exp methods for various values of n , as the BDF-BA-Newton method is too slow to be an interesting choice in this case. In this special case, the EBA-exp method could be handled by using the direct Davison-Maki algorithm. Both presented approaches produced equally satisfactory performances.

7 | CONCLUSION

In this paper, we considered large-scale nonsymmetric differential Riccati equations, especially in the case arising from transport theory. We considered two approaches based on the projection of the differential equation onto an extend block Arnoldi subspace, followed by an integration scheme (BDF or exponential form via the Davison–Maki method, or its modified version). Both methods produce low-rank approximates to the solution of the initial problem. We also presented an approach based on the application of the BDF scheme to the initial problem, leading to the resolution of algebraic Riccati equations, which are solved by a Newton-block Arnoldi method. All three methods were able to achieve an approximate solution although the EBA–BDF1 performed better in terms of computational time. The EBA-exp method suffered from some numerical instability, which could be handled to the detriment of computational time. We reported some numerical examples comparing those approaches for large-scale problems.

ORCID

K. Jbilou  <https://orcid.org/0000-0001-7557-5907>

REFERENCES

1. Abou-Kandil H, Freiling G, Ionescu V, Jank G. Matrix Riccati equations in control and systems theory, in systems & control foundations & applications. Basel, Switzerland: Birkhäuser; 2003.
2. Guo C-H. Nonsymmetric algebraic Riccati equations and Wiener–Hopf factorization for M -matrices. SIAM J Matrix Anal Appl. 2001;23(1):225–242.
3. Juang J. Existence of algebraic matrix Riccati equations arising in transport theory. Linear Algebra Appl. 1995;230:89–100.
4. Reid WT. Riccati differential equations. New York, NY: Academic Press; 1992.
5. Rogers LCG. Fluid models in queueing theory and Wiener–Hopf factorization of Markov chains. Ann Appl Probab. 1994;4(2):390–413.
6. Bini DA, Iannazzo B, Poloni F. A fast Newton's method for a nonsymmetric algebraic Riccati equation. SIAM J Matrix Anal Appl. 2008;30:276–290.
7. Guo C-H, Higham NJ. Iterative solution of a nonsymmetric algebraic Riccati equation. SIAM J Matrix Anal Appl. 2007;29(2):396–412.
8. Jbilou K, Sadok H. Vector extrapolation methods. Applications and numerical comparison. J Comput Appl Math. 2000;122:149–165.
9. El-Moallem R, Sadok H. Vector extrapolation methods applied to algebraic Riccati equations arising in transport theory. Electron Trans Numer Anal. 2013;40:489–506.
10. Lu L-Z. Newton iterations for a non-symmetric algebraic Riccati equation. Numer Linear Algebra Appl. 2005;12:191–200.
11. Lu L-Z. Solution form and simple iteration of a nonsymmetric algebraic Riccati equation arising in transport theory. SIAM J Matrix Anal Appl. 2005;26:679–685.
12. Bentbib A, Jbilou K, Sadek EM. On some Krylov subspace based methods for large-scale nonsymmetric algebraic Riccati problems. Comput Math Appl. 2015;70(10):2555–2565.
13. Heyouni M, Jbilou K. An extended block Arnoldi algorithm for large-scale solutions of the continuous-time algebraic Riccati equation. Electron Trans Numer Anal. 2009;33:53–62.
14. Simoncini V. A new iterative method for solving large-scale Lyapunov matrix equations. SIAM J Sci Comput. 2007;29(3):1268–1288.
15. Fital S, Guo C-H. Convergence of the solution of a nonsymmetric matrix Riccati differential equation to its stable equilibrium solution. J Math Anal Appl. 2006;318:648–657.
16. Bini DA, Iannazzo B, Meini B. Numerical solution of algebraic Riccati equations. Philadelphia, PA: SIAM; 2012.
17. Juang J. Global existence and stability of solutions of matrix Riccati equations. J Math Anal Appl. 2001;258:1–12.
18. Druskin V, Knizhnerman L. Extended Krylov subspaces: approximation of the matrix square root and related functions. SIAM J Matrix Anal Appl. 1998;19(3):755–771.
19. Guldogan Y, Hached M, Jbilou K, Kurulay M. Low rank approximate solutions to large-scale differential matrix Riccati equations. Appl Math. 2018;45:233–254.
20. Petkov PH, Christov ND, Konstantinov MM. Computational methods for linear control systems. Hemel Hempstead, UK: Prentice-Hall; 1991. ISBN 0-13-161803-2.
21. Rosenbrock HH. Some general implicit processes for the numerical solution of differential equations. Comput J. 1963;5:329–330.
22. Ascher UM, Petzold LR. Computer methods for ordinary differential equations and differential-algebraic equations. Philadelphia, PA: SIAM; 1998.
23. Dieci L. Numerical integration of the differential Riccati equation and some related issues. SIAM J Numer Anal. 1992;29(3):781–815.
24. Butcher JC. Numerical methods for ordinary differential equations. Chichester, UK: John Wiley & Sons; 2008.
25. Benner P, Mena H. Rosenbrock methods for solving Riccati differential equations. IEEE Trans Autom Control. 2013;58:2950–2957.
26. Bartels RH, Stewart GW. Solution of the matrix equation $AX + XB = c$ algorithm 432. Commun ACM. 1972;15:820–826.
27. El Guennouni A, Jbilou K, Riquet AJ. Block Krylov subspace methods for solving large Sylvester equations. Numerical Algorithms. 2002;29:75–96.

28. Heyouni M. Extended Arnoldi methods for large low-rank Sylvester matrix equations. *Appl Numer Math.* 2010;60(11):1171–1182.
29. Jbilou K. Block Krylov subspace methods for large continuous-time algebraic Riccati equations. *Numerical Algorithms.* 2003;34:339–353.
30. Jbilou K. Low rank approximate solutions to large Sylvester matrix equations. *Appl Math and Comput.* 2006;177:365–376.
31. Bellman R, Wing GM. An introduction to invariant imbedding. New York, NY: Wiley; 1975.
32. Chandrasekhar S. Radiative transfer. New York, NY: Dover; 1960.
33. Huang T-M, Li R-C, Lin W-W. Structure-preserving doubling algorithms for nonlinear matrix equations. Philadelphia, PA: SIAM. 2018.
34. Juang J, Lin W-W. Nonsymmetric algebraic Riccati equations and Hamiltonian-like matrices. *SIAM J Matrix Anal Appl.* 1998;20:228–243.
35. Davison EJ, Maki MC. The numerical solution of the matrix Riccati differential equation. *IEEE Trans Autom Control.* 1973;18:71–73.

How to cite this article: Angelova V, Hached M, Jbilou K. Approximate solutions to large nonsymmetric differential Riccati problems with applications to transport theory. *Numer Linear Algebra Appl.* 2019;e2272.
<https://doi.org/10.1002/nla.2272>