DAE Panel
DOI: https://doi.org/10.52825/dae-p.v3i.2632
Submitted: 06 Mar. 2025
Accepted: 23 Aug. 2025
Published: 03 Sep. 2025

On Computation of Lyapunov Exponents by QR Methods with Error Control for Semi-Linear DAEs

VU HOANG LINH, NGUYEN DUY TRUONG , AND PHAN QUANG TUYEN

Abstract: In this paper, we propose and analyze numerical methods for computing Lyapunov exponents of semi-linear differential-algebraic equations (DAEs), leveraging smooth QR factorizations and Runge-Kutta (RK) methods with error control and automatic step size selection. We demonstrate how both discrete and continuous QR approaches efficiently approximate Lyapunov exponents by simultaneously solving semi-linear DAEs and their linearized counterparts. The paper details the underlying algorithms, error analysis, and numerical integration techniques, focusing on half-explicit RK (HERK) and explicit singly diagonal implicit RK (ESDIRK) methods. We also provide implementation details and present numerical experiments to illustrate the efficiency of these methods.

Keywords: Semi-Linear Differential-Algebraic Equations, Linearization, Lyapunov Exponents, Smooth QR Factorization, Embedded Runge-Kutta Methods, Error Control

AMS Subject Classification (2020): 65L07, 65L80, 34D08, 34D09

1 Introduction

In this paper, we propose and analyze numerical methods for computing Lyapunov exponents for semi-linear differential-algebraic equations (DAEs) of the form

$$E(t)x'(t) = f(t,x(t)),\ t \in \mathbb{I}, \tag{1}$$

where the interval $\mathbb{I} = [0,\infty)$, $x : \mathbb{I} \to \mathbb{R}^m$, $E : \mathbb{I} \to \mathbb{R}^{m\times m}$, and $f = f(t,x) : \mathbb{I} \times \mathbb{R}^m \to \mathbb{R}^m$ are sufficiently smooth functions with bounded (partial) derivatives. The leading term E is supposed to be singular, but has a constant rank for all $t \geq 0$. We also assume that the initial value problem (IVP) for DAE (1) together with a consistent initial condition $x(0) = x_0$ has a unique solution $x(t)$, which is sufficiently smooth on $\mathbb{I}$.

Lyapunov exponents are a powerful tool for analyzing the asymptotic behavior of solutions to ordinary differential equations (ODEs). They are also widely used to study nonlinear systems through linearized analysis. The stability and spectral theories of ODEs were developed by Lyapunov, Perron, Bohl, and others (see [1, 5]). Numerical methods for computing spectral intervals of ODEs have been extensively studied, particularly in a series of works by Dieci and Van Vleck (see [7, 8, 9, 10, 11, 12, 13]).

Differential-algebraic equations (DAEs) arise in various applications, including constrained multibody dynamics, electrical circuit simulation, and chemical engineering [3, 17]. The stability theory for DAEs has been developed more recently compared to ODEs. Spectral concepts such as Lyapunov, Bohl, and Sacker-Sell spectral intervals have been extended to general linear DAEs with variable coefficients [18, 20, 21]. These works also introduced numerical methods for computing spectral intervals using QR and SVD factorizations. Furthermore, the stability analysis of DAEs via Lyapunov exponents has been applied in several real-world scenarios [2, 4, 14, 23, 24].

For numerical integration of DAE systems, implicit Runge-Kutta and BDF methods are commonly used [3, 17]. Later studies demonstrated the efficiency of half-explicit Runge-Kutta (HERK) methods as an alternative to implicit methods [19, 22]. However, early numerical experiments [19, 22] relied solely on uniform meshes. Embedded HERK methods with variable step sizes and error control have also been proposed for solving nonlinear DAEs [25].

In this paper, following the results in [18, 20, 21] for linear DAEs, we present and discuss numerical algorithms for computing Lyapunov exponents associated with a particular solution of semi-linear DAE (1). Without loss of generality, we assume that the semi-linear DAE (1) is already given in the strangeness-free form and consider the linear variational system along a particular solution $x^*(t)$

$$E(t)y'(t) = A(t)y(t),\ t \in \mathbb{I}, \tag{2}$$

where

$$E(t) = \begin{bmatrix} E_1(t) \\ 0 \end{bmatrix},\ A(t) = \frac{\partial f}{\partial x}(t, x^*) = \begin{bmatrix} A_1(t) \\ A_2(t) \end{bmatrix},$$

and $E_1 \in C(\mathbb{I}, \mathbb{R}^{d\times m})$ and $A_2 \in C(\mathbb{I}, \mathbb{R}^{(m-d)\times m})$ are of full row rank. The strangeness-free assumption means that the matrix

$$\bar{E}(t) = \begin{bmatrix} E_1(t) \\ A_2(t) \end{bmatrix} \tag{3}$$

is nonsingular for all $t \in \mathbb{I}$, which - under the smoothness assumption - implies that the DAEs are of differentiation index one [17]. Furthermore, any fully implicit index-one semi-linear DAE of the form (1) can be reduced to the strangeness-free form by multiplying both sides of (1) by an appropriate (and numerically computable) matrix function.

To approximate Lyapunov exponents, we simultaneously integrate the DAE systems (1) and (2) using numerical methods with error control and automatic step-size selection. Previous works [20, 18, 21] focused on linear DAEs and employed low-order integration methods without error control. Thanks to the key reformulation and convergence analysis in [22], we leverage well-known embedded RK methods with error estimation and automatic step-size selection. The motivation for investigating half-explicit methods in [19] and revisiting RK methods in [22] stemmed from a critical observation: applying standard ODE methods directly to strangeness-free DAEs often leads to order reduction [19], with the exception of collocation methods [17, Theorem 5.17]. However, collocation methods are implicit, making their use in continuous QR methods challenging, as they require solving matrix-valued nonlinear DAEs, leading to nonlinear matrix equations (see Section 3).

This work extends the approximation of Lyapunov exponents to semi-linear DAEs by combining established techniques and providing detailed algorithms. Implicit RK methods with error estimation can be efficient for stiff problems but come with increased computational cost. Therefore, in this work, we recommend the use of embedded half-explicit Runge-Kutta (HERK) and explicit singly diagonal implicit Runge-Kutta (ESDIRK) methods [15, 16] due to their cheaper computational cost and convenient implementation. Importantly, following the approach in [22], these integration methods preserve their ODE order. This is a significant novelty of the paper. Another novelty is the extension of the error analysis to semi-linear DAEs. Building on the approach in [21] for linear DAEs, we present an error analysis demonstrating the applicability of our algorithms. In these methods, errors in approximating Lyapunov exponents are controlled by the local integration error. Last but not least, algorithmic aspects are discussed in details. Note that while existing QR methods for ODEs and their implementations can be adapted to semi-explicit DAEs, they are not directly applicable to more general DAEs in the strangeness-free form like (1) and (2).

The paper is organized as follows. In the next section, we review fundamental concepts from the Lyapunov theory of linear DAEs and extend the definition of Lyapunov exponents to semi-linear DAEs. We also provide a brief overview of embedded RK methods with error control for solving semi-linear DAEs in their reformulated form. In Section 3, we introduce two variants of the QR algorithm for computing Lyapunov exponents of semi-linear DAEs and discuss their implementation in detail. Section 4 presents an error analysis of these QR methods. Finally, in Section 5, we showcase numerical experiments demonstrating the efficiency and robustness of our approach. The paper concludes with a summary of key findings.

2 Preliminaries

2.1 Lyapunov exponents for DAEs

In this section, we summarize key concepts from the Lyapunov exponent theory for linear DAEs and extend them to semi-linear DAEs. For a detailed discussion on Lyapunov exponents in the linear case, we refer to [18, 20, 21]. Our goal is to define Lyapunov exponents associated with a particular solution of semi-linear DAEs in the form of (1).

We first recall some notions for linear DAEs of the strangeness-free form

$$E(t)x' = A(t)x,\ t \in \mathbb{I}, \tag{4}$$

where

$$E(t) = \begin{bmatrix} E_1(t) \\ 0 \end{bmatrix},\ A(t) = \begin{bmatrix} A_1(t) \\ A_2(t) \end{bmatrix},$$

and $E_1 \in C(\mathbb{I}, \mathbb{R}^{d\times m})$ and $A_2 \in C(\mathbb{I}, \mathbb{R}^{(m-d)\times m})$ are of full row rank. Furthermore, the matrix

$$\bar{E}(t) = \begin{bmatrix} E_1(t) \\ A_2(t) \end{bmatrix} \tag{5}$$

is nonsingular for all $t \in \mathbb{I}$. Reduction to the strangeness-free form and solvability analysis for DAEs (1) are discussed in [17].

Definition 2.1. *A matrix function $X \in C^1(\mathbb{I}, \mathbb{R}^{m\times d})$ is called a (minimal) fundamental solution matrix of* (4) *if each of its columns is a solution of* (4) *and* $\operatorname{rank} X(t) = d$*, for all* $t \geq 0$.

Definition 2.2. *For a given minimal fundamental matrix solution X of a strangeness-free DAE system of the form* (4)*, we introduce*

$$\lambda_i^u = \limsup_{t\to\infty} \frac{1}{t} \ln \|X(t)e_i\|,\ \lambda_i^l = \liminf_{t\to\infty} \frac{1}{t} \ln \|X(t)e_i\|,\ i = 1, 2, \cdots, d,$$

where e_i denotes the i-th unit vector. The columns of a minimal fundamental solution matrix are said to form a normal basis if $\sum_{i=1}^d \lambda_i^u$ is minimal. The quantities λ_i^u, $i = 1,2,\cdots,d$, belonging to a normal basis are called (upper) Lyapunov exponents. The set of the Lyapunov exponents is called the Lyapunov spectrum of DAE (4).

For the purpose of numerical computation, it is important to study the behavior of Lyapunov exponents under small perturbations. We consider a perturbed system of DAEs

$$[E(t)+\Delta E(t)]x' = [A(t)+\Delta A(t)]x,\ t \in \mathbb{I}, \tag{6}$$

where we restrict the perturbations to have the form

$$\Delta E(t) = \begin{bmatrix} \Delta E_1(t) \\ 0 \end{bmatrix},\ \Delta A(t) = \begin{bmatrix} \Delta A_1(t) \\ \Delta A_2(t) \end{bmatrix}.$$

Here ΔE and ΔA are assumed to be as smooth as E and A, respectively. Perturbations of this structure are called admissible. The DAE (4) is said to be robustly strangeness-free if it is still strangeness-free under all sufficiently small admissible perturbations. Note that it is essential to restrict the perturbations to this structure to prevent from changing the strangeness-index.

Definition 2.3. *The upper Lyapunov exponents $\lambda_1^u \geq \lambda_2^u \geq \cdots \lambda_d^u$ of (4) are said to be stable if for any $\varepsilon > 0$, there exists $\delta > 0$ such that the conditions $\sup_t \|\Delta E(t)\| < \delta$, $\sup_t \|\Delta A(t)\| < \delta$ on the perturbations imply that the perturbed DAE (6) system is strangeness-free and*

$$|\lambda_i^u - \gamma_i^u| < \varepsilon, \text{ for all } i = 1, 2, \cdots, d,$$

where the γ_i^u are the ordered upper Lyapunov exponents of the perturbed system (6).

The stability of distinct Lyapunov exponents can be established by the property of integral separation, see [1].

Definition 2.4. *A minimal fundamental solution matrix X for (4) is called integrally separated if for $i = 1, 2, \cdots, d-1$ there exists constants $\beta > 0$ and $\gamma > 0$ such that*

$$\frac{\|X(t)e_i\|}{\|X(s)e_i\|} \cdot \frac{\|X(s)e_{i+1}\|}{\|X(t)e_{i+1}\|} \geq \gamma e^{\beta(t-s)},$$

for all t, s with $t \geq s \geq 0$. If a DAE system has an integrally separated minimal fundamental solution matrix, then we say it has the integral separation property.

In practice, the integral separation (and so the stability of Lyapunov exponents) can be checked via the computation of Steklov differences, see [9, 11, 20, 21]. Furthermore, if DAE (4) has the integral separation, then the lower Lyapunov exponents associated with the columns of an integrally separated fundamental solution matrix are well defined and the set of intervals $[\lambda_i^l, \lambda_i^u]$, $i = 1, 2, \cdots, d$, are called the spectral intervals of DAE (4).

As described in [7, 8], Lyapunov exponents provide a meaningful way to characterize the asymptotic behavior of solutions for nonlinear ODEs and associated linearized systems. They provide a generalization of the linear stability analysis for perturbations of steady state solutions to time-dependent solutions.

For a given solution trajectory $x(t)$, one considers the linear variational system (2). Then, for a minimal fundamental matrix solution $Y(t)$, the matrix

$$\Lambda = \limsup_{t\to\infty} \Lambda_{x_0}(t) := \limsup_{t\to\infty} (Y^T(t)Y(t))^{\frac{1}{2t}} \tag{7}$$

is well-defined under some additional boundedness assumptions, and it is a symmetric positive definite matrix.

If $\{p_i, \mu_i\}$ denote the eigenvectors and associated eigenvalues of Λ such that $\Lambda p_i = \mu_i p_i$, or $p_i^T \Lambda p_i = \mu_i$, then the Lyapunov exponents with respect to the trajectory $x(t)$ of (1) or the linear DAE system (2) are given by

$$\lambda_i = \ln(\mu_i) = \ln\left(\limsup_{t\to\infty}\langle Y(t)p_i, Y(t)p_i\rangle\right)^{\frac{1}{2t}} = \limsup_{t\to\infty}\frac{1}{t}\ln\|Y(t)p_i\|,\ i = 1,\ldots,d, \tag{8}$$

where $\langle x,y\rangle = x^T y$ and $\|x\| := \langle x, x\rangle^{\frac{1}{2}}$. Thus, λ_i is a measure of the mean logarithmic growth rate of perturbations in the subspace $Eig(\Lambda, \mu_i) = \{p_i \in \mathbb{R}^d : \Lambda p_i = \mu_i p_i\}$, and λ_i describes how nearby trajectories for the DAE system (1) converge or diverge from $x(t)$.

In practice, we solve (1) with a given consistent initial condition to find the solution $x(t)$. In parallel, we evaluate the matrix coefficients $A(t)$ of (2). Then, we approximate the Lyapunov exponents of linear DAE system (2) by the methods proposed in [18, 20, 21]. In the remainder part of the paper, we will describe the algorithms in details and discuss their implementation and error analysis. For numerical integration, we use embedded RK methods with error control and variable stepsizes.

2.2 Runge-Kutta methods with error control and variable stepsize

When using numerical methods for integration, one has to estimate and control the actual errors of numerical solutions. We use embedded Runge-Kutta methods to estimate the errors and choose suitable (and variable) stepsizes accordingly to a given error tolerance. We note that numerical integration of strangeness-free DAEs like (1) and (2) is more complicated than semi-explicit index-1 DAEs since the differential and algebraic variables are not separated. Thus, well-known ODE methods may suffer order reduction, see [19] and references therein. In this paper, we use the approach presented in [22] and with regard to the efficiency, we recommend half-explicit and explicit singly diagonal implicit Runge-Kutta methods, see [15, 16].

First, we rewrite the DAEs (1) into the form

$$(E(t)x(t))' = E'(t)x(t) + f(t, x(t)), \tag{9}$$

or in the strangeness-free formulation

$$\begin{aligned} (E_1(t)x(t))' &= E_1'(t)x(t) + f_1(t, x(t)), \\ 0 &= f_2(t, x(t)), \end{aligned}$$

on a finite interval $\mathbb{I} = [t_0, T]$, where $E(t) = \begin{bmatrix} E_1(t) \\ 0 \end{bmatrix}$, $f(t, x(t)) = \begin{bmatrix} f_1(t, x(t)) \\ f_2(t, x(t)) \end{bmatrix}$. Note that the reformulation (9) plays the key role in avoiding order reduction for the numerical methods, see [22].

For illustration of the use of half-explicit RK methods, let us take an embedded pair of explicit Runge-Kutta methods with order p and $\hat{p} = p+1$, whose coefficients are $c = (c_1 \ldots c_s)^T$, $A = [a_{ij}]_{s\times s}$, $b = (b_1 \ldots b_s)$ and $\hat{b} = (\hat{b}_1 \ldots \hat{b}_s)$, respectively.

Now, we consider a subinterval $[t_{n-1}, t_n]$ and suppose that an approximation x_{n-1} to $x(t_{n-1})$ is given. Let $T_i = t_{n-1} + c_i h$ be the time at stage i and the stage approximations $U_i \approx x(T_i)$, $K_i \approx (E_1 x)'(T_i)$, $i = 1, 2, \ldots, s$. Furthermore, we assume that the values $E_{1,i} = E_1(T_i)$, $E_{1,i}' = E_1'(T_i)$ are available. The half-explicit Runge-Kutta scheme applied to (9) reads

$$\begin{aligned} U_1 &= x_{n-1}, \\ E_{1,i}U_i &= E_1(t_{n-1})U_1 + h\sum_{j=1}^{i-1} a_{ij}K_j,\ i = 2,3,\ldots,s, \\ E_1(t_n)x_n &= E_1(t_{n-1})x_{n-1} + h\sum_{i=1}^{s} b_iK_i. \end{aligned} \tag{10}$$

The approximations U_{i+1}, K_i, $i = 1,2,\ldots,s-1$ can be determined from the systems given by

$$\begin{aligned} K_i &= E_1'(T_i)U_i + f_1(T_i, U_i),\ i = 1, 2, \cdots, s, \qquad (11) \\ E_1(T_{i+1})U_{i+1} &= E_1(t_{n-1})U_1 + h\sum_{j=1}^{i} a_{i+1,j}K_j, \\ f_2(T_{i+1}, U_{i+1}) &= 0,\ i = 1, 2, \cdots, s-1. \end{aligned}$$

Finally, the numerical solution x_n of order p is determined by the system

$$\begin{aligned} E_1(t_n)x_n &= E_1(t_{n-1})U_1 + h\sum_{i=1}^{s} b_iK_i, \\ f_2(t_n, x_n) &= 0. \end{aligned} \tag{12}$$

Due to the strangeness-free assumption, for sufficiently small h, the Jacobian matrices associated with nonlinear systems (11) and (12) are nonsingular. Therefore, nonlinear systems (11), (12) can be solved by the Newton iterative method.

In [22], the convergence of HERK methods applied to (9) was established, namely if the underlying explicit Runge-Kutta method is convergent of order p for ODEs, then the HERK scheme (11)-(12) applied to (9) is convergent of order p, i.e.,

$$\|x_n - x(t_n)\| = \mathscr{O}(h^p) \text{ as } h \to 0,$$

where $t_n = t_0 + nh$ is fixed.

As a consequence, [25, Proposition 1] shows that the numerical solutions of the embedded pair of Runge-Kutta methods of orders p and $p+1$ for solving the DAE system (9) are convergent of order p and $p+1$, respectively.

Let us denote the numerical solutions at t_n by x_n and $\hat{x}_n$, respectively. Then the local error of the numerical solution x_n can be estimated by using the difference between the two numerical solutions, e.g.

$$error = \max_{1\le i\le m} \frac{|(x_n)_i - (\hat{x}_n)_i|}{(1 + |(x_n)_i|)}. \tag{13}$$

Given an error tolerance TOL, if h is the actual stepsize, we recommend a new suitable stepsize as follows

$$h_{new} = \nu h \left[\frac{TOL}{error}\right]^{\frac{1}{p+1}}, \tag{14}$$

where ν is a safety factor, usually $\nu = 0.9$ is used.

Alternatively, one can apply implicit Runge-Kutta methods in a similar manner to solve the IVP for the DAEs (1)/(9), see [22]. However, the computational cost for the IRK methods is significantly increased since we have to solve a large nonlinear system at each step. Moreover, additional difficulties arise with IRK schemes in the continuous QR method presented in Section 3 below. For comparison, we suggest the ESDIRK methods presented in [15, 16] as an alternative family for numerical integration. The implementation and convergence of ESDIRK methods with error control are similar to the above discussions for HERK methods.

For numerical experiments in this paper, we use the HERK method based on the Dormand-Prince embedded pair of orders $p = 5(4)$, see [3], and several ESDIRK methods to solve DAEs (1) /(9).

3 QR methods

In this section, we present numerical methods based on smooth QR factorization to approximate Lyapunov exponents of problem (1). These methods are more complicated than the ones for linear DAEs. We must solve the semi-linear DAE to get a particular solution in order to obtain the linearized system (2). In practice, numerical solutions of the semi-linear system and its variational linearized system are computed using the same mesh.

3.1 Discrete QR method

In the discrete QR method, the fundamental solution matrix $Y(t)$ and its triangular factor R are indirectly evaluated by a re-orthogonalized integration of the DAE system (2).

First, we choose an initial stepsize h_0. Let the initial matrix Y_0 be given at t_0 for the linearized system (2). We perform the QR factorization $Y_0 = Q(t_0)R(t_0)$, where $R(t_0)$ has positive diagonal elements.

For $j = 1, 2, \ldots, N$, let $Y(t, t_{j-1})$ be the numerical solution to the matrix initial value problem

$$\begin{aligned} E(t)Y'(t, t_{j-1}) &= A(t)Y(t, t_{j-1}), \\ Y(t_{j-1}, t_{j-1}) &= Q(t_{j-1}). \end{aligned} \tag{15}$$

Then, we perform the QR factorization

$$Y(t_j, t_{j-1}) = Q(t_j)R(t_j, t_{j-1}). \tag{16}$$

We require the diagonal elements of $R(t_j, t_{j-1})$ be positive. Consequently, we have the unique QR factorization $Y(t_j) = Q(t_j)R(t_j)$, where

$$R(t_j) = R(t_j, t_{j-1})R(t_{j-1}, t_{j-2}) \ldots R(t_2, t_1)R(t_1, t_0)R(t_0). \tag{17}$$

Thus, we have the approximations to the upper Lyapunov exponents

$$\lambda_i = \limsup_{t\to\infty} \frac{1}{t} \ln\left(R(t)_{ii}\right),\ i = 1, 2, \ldots, d, \tag{18}$$

where $R(t)_{ii}$ is the i-th diagonal element of $R(t)$ and approximated by $R(t_j)_{ii} = \sum_{l=1}^{j} \ln(R(t_l, t_{l-1})_{ii})$. The formula for the approximations to the lower Lyapunov exponents is similar.

In the interval from t_j to t_{j+1}, there are four steps to follow:

1. We integrate (9) to obtain a solution $x(t)$, then compute the matrix coefficients $A_1(t), A_2(t)$ corresponding to solution $x(t)$ to obtain a linearized DAE system of the form (15). We note that both systems (9) and (15) are integrated simultaneously by embedded Dormand-Prince pair of order 4(5) in the same stages.

Thus, with a stepsize h, we obtain approximate solutions x_{j+1}, Y_{j+1} of order 5 and $\hat{x}_{j+1}, \hat{Y}_{j+1}$ of order 4.

2. In this step, we perform error control based on the pair of solutions $x_{j+1}, \hat{x}_{j+1}$, or on the pair of solutions $Y_{j+1}, \hat{Y}_{j+1}$ and the solutions $x_{j+1}, \hat{x}_{j+1}$. For example, we can perform error control on the solution pair $x_{j+1}, \hat{x}_{j+1}$. Let h be the current stepsize and h_{new} be the new stepsize. We will restrict the new stepsize as $\alpha_1 h \le h_{new} \le \alpha_2 h$, $\alpha_1 < 1$, $\alpha_2 > 1$, e.g. we set $\alpha_1 = \frac{1}{5}$, $\alpha_2 = 5$ and choose h_{new} as follows:
   - First, we estimate the error by using formula (13).
   - Then, we compute the new stepsize by using formula (14). We also restrict h_{new} to not less than $\alpha_1 h$ and not larger than $\alpha_2 h$.
   - Next, given a tolerance TOL, we check if $error \le TOL$, the step is accepted and the computed new stepsize will be used for the next step. Otherwise, it is rejected, and we repeat the integration with the new stepsize.
3. After a successful step, we perform QR factorizations (16) for Y_{j+1} to obtain the approximate factor R_{j+1}.
4. Finally, we update the approximations for λ_i, $i = 1, \dots, d$ by computing

$$s_i(t_{j+1}) = s_i(t_j) + \ln (R_{j+1})_{ii},\ \lambda_i(t_{j+1}) = \frac{1}{t_{j+1}} s_i(t_{j+1}), \tag{19}$$

   and solve the optimization problems $\min_{\tau \le t \le t_{j+1}} \lambda_i(t)$ and $\max_{\tau \le t \le t_{j+1}} \lambda_i(t)$ with a given (sufficiently large) τ. At the beginning, we set $s_i(t_0) = 0$, $i = 1, \dots, d$.

The discrete QR algorithm is given in the pseudo-code form as follows.

Algorithm 3.1. Input: *Given the DAE (1) on an interval* $[0, T]$*, with the consistent initial condition* $x(t_0) = x_0$, $Y_0 \in \mathbb{R}^{n \times p}$*, an error tolerance* TOL*, and an initial guess of stepsize* h_0, $\tau \in [0, T]$.

Output: *Approximate Lyapunov exponents* λ_i^u, λ_i^l, $i = 1, 2, \dots, p \le d$.

Initialization:

- *Set* $t_0 := 0$*, and perform QR factorization* $Y(t_0) = Q_0 R_0$*, where* R_0 *has positive diagonal elements.*
- *Set* $\lambda_i(t_0) := 0$ *and* $s_i(t_0) := 0$ *for* $i = 1, \dots, p$ *to calculate the sum in (18).*

While $t_j < T$

1. *If* $t_{j+1} = t_j + h_j > T$ *then* $h_j = T - t_j$;
2. *We solve the initial value problem (9) and its linearized system (15) in the same stages of scheme (11)-(12). Denote the numerical solution computed at* $t = t_{j+1}$ *by* $\bar{x}_{j+1}, \bar{Y}_{j+1}$ *of order 5 and* $\tilde{x}_{j+1}, \tilde{Y}_{j+1}$ *of order 4.*
3. *Calculate*

$$error = \max_{1 \le i \le m} \frac{|(\bar{x}_{j+1})_i - (\tilde{x}_{j+1})_i|}{(1 + |(\bar{x}_{j+1})_i|)},$$

   and

$$h_{new} = \nu h_j \left(\frac{TOL}{error} \right)^{\frac{1}{p+1}}.$$

4. *If error $\leq TOL$ (the step is accepted), then we carry out QR factorization $\bar{Y}_{j+1} = Q_{j+1}R_{j+1}$ to find the factors R_{j+1} with positive diagonal elements.*

   We then update the Lyapunov exponents:

   $t_{j+1} = t_j + h_j$; $Y(t_{j+1}) = Q_{j+1}$; $x(t_{j+1}) = \bar{x}_{j+1}$;
   $s_i(t_{j+1}) = s_i(t_j) + \ln(R_{j+1})_{ii}$ *and* $\lambda_i(t_{j+1}) = \frac{1}{t_{j+1}} s_i(t_{j+1})$, *for* $i = 1, 2, \ldots, p$.
   If desired, check the integral separation property by using $\{s_i\}_{i=1}^p$.
   Set $h_{j+1} = \min\{h_{new}, \alpha_2 h_j\}$ *and go to the next interval.*
   Otherwise, set $h_j = \max\{h_{new}, \alpha_1 h_j\}$ *(the stepsize will be adjusted) and go back to 2.*

5. *Compute* $s_i(t_{j+1}) = s_i(t_j) + \ln(R_{j+1})_{ii}$, $\lambda_i(t_{j+1}) = \frac{1}{t_{j+1}} s_i(t_{j+1})$.
   Update $\min_{\tau \leq t \leq t_{j+1}} \lambda_i(t)$ *and* $\max_{\tau \leq t \leq t_{j+1}} \lambda_i(t)$.

3.2 Continuous QR method

For the continuous QR method, we consider the linearized DAE system of (9) and then apply the method in [21]. The unique factorization $Y(t) = Q(t)R(t)$ with positive diagonal elements in R is to be determined for $t \in \mathbb{I}$. Differentiating $Y = QR$ and inserting this into DAE (9) yields

$$EQ' + EQR'R^{-1} = AQ. \tag{20}$$

Equation (20) is a nonlinear strangeness-free DAE system for the matrix function Q. We note that the algebraic equation of (20) satisfies $A_2Q = 0$, its derivative is $A_2'Q + A_2Q' = 0$, and replace it into (20) to obtain

$$\bar{E}(Q' + QR'R^{-1}) = \bar{A}Q, \tag{21}$$

where

$$\bar{E} = \begin{bmatrix} E_1 \\ A_2 \end{bmatrix}, \bar{A} = \begin{bmatrix} A_1 \\ -A_2' \end{bmatrix}, \tag{22}$$

Now, we derive a formula for Q^TQ'. By [21, Lemma 12], there exists a bounded, full-column rank matrix function $P \in C(\mathbb{I}, \mathbb{R}^{n\times p})$, and an upper triangular nonsingular matrix solution $\mathscr{E} \in \mathscr{C}(\mathbb{I}, \mathbb{R}^{p\times p})$ such that

$$P^T\bar{E} = \mathscr{E}Q^T \tag{23}$$

holds. Furthermore, if we require $P^TP = I_p$ and the diagonal elements of $\mathscr{E}$ to be positive, then P and $\mathscr{E}$ are unique. A practical method for computing P and $\mathscr{E}$ is described in [21].

Multiplying (21) from the left by P^T defined as in (23), we obtain

$$\mathscr{E}Q^TQ' + \mathscr{E}R'R^{-1} = P^T\bar{A}Q.$$

Setting $B := R'R^{-1}$, $S(Q) := [s_{i,j}(Q)] = Q^TQ'$, and $K := P^T\bar{A}Q$, it follows that $S(Q) = \mathscr{E}^{-1}K - B$. Since $S(Q)$ is a skew-symmetric matrix and B is upper triangular, the strictly lower part of $S(Q)$ is defined by the strictly lower part of $W := [w_{i,j}] = \mathscr{E}^{-1}K$ (denoted $low(W)$) and its upper triangular part is determined by the skew symmetry. We have $S(Q) = low(W) - [low(W)]^T$, i.e.,

$$s_{i,j} = \begin{cases} w_{i,j}, & i > j, \\ 0, & i = j, \\ -w_{i,j}, & i < j, \end{cases} \quad 1 \leq i, j \leq p. \tag{24}$$

Thus, Q is obtained by solving the initial value problem for the nonlinear strangeness-free matrix-valued DAE

$$EQ' = AQ - EQB,$$

or equivalently

$$EQ' = -EQ(W - S(Q)) + AQ. \tag{25}$$

For the numerical integration, we need to use a DAE solver which preserves the algebraic constraint as well as the orthogonality condition $Q^T Q = I_p$. We also see that $B = W - S(Q) = u\bar{p}p(W) + [low(W)]^T$, where $u\bar{p}p(W)$ denotes the upper triangular part of W (including the diagonal).

We compute W by solving the upper triangular algebraic system $\mathscr{E}W = K$. If we set $\mathscr{A} := K - \mathscr{E}S(Q) = P^T\bar{A}Q - \mathscr{E}Q^TQ'$, then the differential equation for the factor R is given by the upper triangular matrix equation of size $p \times p$

$$\mathscr{E}R' = \mathscr{A}R, \text{ or equivalently } R' = BR. \tag{26}$$

However, we are only interested in the diagonal elements $r_{i,i}$ of R (or more exactly, in their logarithm). Since the system (26) is upper triangular and the diagonal elements of $S(Q)$ are zeros, we obtain the scalar differential equations

$$r'_{i,i} = w_{i,i} r_{i,i}, \tag{27}$$

where $w_{i,i}$, $i = 1, \ldots, p$ is the i-th diagonal element of the matrix W. To determine these quantities, we introduce the auxiliary functions $\phi_i(t)$ defined by the solution of the initial value problems

$$\phi_i'(t) = w_{i,i}(t),\ \phi_i(t) = 0,\ (i = 1, 2, \ldots, p). \tag{28}$$

Finally, the functions $\lambda_i(t)$ are defined by

$$\lambda_i(t) = \frac{1}{t}\phi_i(t),\ i = 1, 2, \ldots, p, \tag{29}$$

and their limits as $t \to \infty$ are to be approximated.

In the step from t_j to t_{j+1}, we use the Dormand-Prince pair to obtain the solutions x_{j+1} of order 5 and $\hat{x}_{j+1}$ of order 4 for the nonlinear DAE system of the form (9). Then, we use these solutions to compute the matrix coefficients $A_1(t), A_2(t), A_2'(t)$ and find the solution Q_{i+1} and $\hat{Q}_{j+1}$, respectively for nonlinear system (25). The DAE systems (9) and (25) are integrated together by the same Runge-Kutta scheme $(11)-(12)$. We note that the numerical solution for (25) needs to be re-orthogonalized at each meshpoint to preserve the orthogonality condition.

At this point, it is possible to perform error control for the solution $x(t)$ in (9), and/or for the orthogonal factor Q in (25). For example, we monitor the error in the solution $x(t)$ and select a suitable stepsize as follows.

Let h be the current stepsize, and h_{new} be the new stepsize.

- Estimate the error of the solution $x(t)$ by using the formula (13).
- We compute the new stepsize by using (14), where $p = 4$ for Dormand-Prince embedded pair.
- We will restrict that h_{new} does not change too much, i.e., $\alpha_1 h \le h_{new} \le \alpha_2 h$; α_1 and α_2 are given as in Section 3.1.
- If $error \le TOL$, the step is accepted and we go to the next step. Otherwise, it is rejected and we repeat the integration.

Next, we use the solution Q_{j+1} of order 5 of nonlinear system (25) to compute $P(t_{j+1}), \mathscr{E}(t_{j+1}), K(t_{j+1})$ as in (23) and their definitions, respectively. Then, we must solve for $W(t_{j+1})$. Finally, we compute $\phi_i(t_{j+1}), \lambda_i(t_{j+1}), i = 1, \ldots, p$, by their formulas in (28) and (29).

Remark 3.2. *In the continuous QR method, we must approximate the derivative of the coefficient $-A_2'(t)$. We can compute by the analytical formula as*

$$A_2'(t) = \frac{dA_2}{dt} = \frac{d}{dt}\left(\frac{\partial f_2(t,x)}{\partial x}\right) = \frac{\partial^2 f_2(t,x)}{\partial x \partial t} + \frac{\partial^2 f_2(t,x)}{\partial x^2}\frac{dx}{dt}.$$

Otherwise, if it is not available, we can use a finite difference formula to approximate the derivative.

In summarizing the above description, we give the continuous QR algorithm as follows.

Algorithm 3.3. Input: *Given the DAE (1) on an interval $[0,T]$, with the consistent initial condition $x(t_0) = x_0$ of (9), and $Q_0 = Q(t_0)$ as the initial value for (25), an error tolerance TOL, and an initial guess of the step size h_0.*

Output: *Approximate Lyapunov exponents λ_i^u, λ_i^l, $i = 1,2,\dots,p$.*

Initialization:

- *Set $j = 0$, $t_0 := 0$. Compute $P(t_0)$, $\mathscr{E}(t_0)$, and $K(t_0)$ as in (23).*
- *Calculate $W(t_0)$ by its formula.*
- *Set $\lambda_i(t_0) := 0$ and $\phi_i(t_0) := 0$ for $i = 1,\dots,p$.*

While $t_j < T$

1. *If $t_{j+1} = t_j + h_j > T$ then $h_j = T - t_j$;*
2. *We solve the initial value problems for DAE (9) and for system (25) in the same stages in schemes (11)-(12) to find $x(t_{j+1})$ and $Q(t_{j+1})$ on $[t_j, t_{j+1}]$. Denote the numerical solution computed at $t = t_{j+1}$ by $\bar{x}_{j+1}, \bar{Q}_{j+1}$ of order 5 and $\tilde{x}_{j+1}, \tilde{Q}_{j+1}$ of order 4.*
3. *Calculate*

$$\begin{aligned} error &= \max_{1\le i\le m} \frac{|(\bar{x}_{j+1})_{(i)} - (\tilde{x}_{j+1})_{(i)}|}{(1+|(\bar{x}_{j+1})_{(i)}|)} \\ h_{new} &= \nu h_j \left(\frac{TOL}{error}\right)^{\frac{1}{p+1}}; \end{aligned}$$

4. *If $error \le TOL$ (the step is accepted) then we reorthogonalize the factor $Q(t_{j+1})$ of (25) as: $\bar{Q}_{j+1} = Q_{j+1}R_{j+1}$ to find R_{j+1} with positive diagonal elements.*

   We then update the Lyapunov exponents:

   $t_{j+1} = t_j + h_j$;
   $x(t_{j+1}) = \bar{x}_{j+1}$ and $Q(t_{j+1}) = Q_{j+1}$;
   We compute $P(t_{j+1}), \mathscr{E}(t_{j+1}), K(t_{j+1})$ and solve $W(t_{j+1})$;
   Compute $\phi_i(t_{j+1})$ and $\lambda_i(t_{j+1})$, for $i = 1,2,\dots,p$ as in (28) and (29).
   If desired, we compute the Steklov differences to check integral separation property;
   Set $h_{j+1} = \min\{h_{new}, \alpha_2 h_j\}$ and go to the next interval.
   Otherwise, set $h_j = \max\{h_{new}, \alpha_1 h_j\}$ (the stepsize will be adjusted) and go back to 2.

5. *Update $\min_{\tau\le t\le t_{j+1}} \lambda_i(t)$ and $\max_{\tau\le t\le t_{j+1}} \lambda_i(t)$.*

4 Error analysis

In this section, we aim to give an error analysis for the QR methods described above. We need to integrate the semi-linear DAEs of the form (1) on the interval $[0, T]$, together with an initial condition $x(0) = x_0$. Linearizing the equation (1) along the obtained solution, we obtain the linear DAE (2). Thus, we solve simultaneously two systems (1) and (2). In order to obtain error analysis, we first recall the error analysis of the QR methods for linear DAE (2), see [21]. Then, we will investigate the perturbation analysis for semi-linear DAE (1) and linearized DAE (2).

4.1 Error analysis of the QR methods for linear DAEs

A careful error analysis of the QR methods for the linear case was given in [21]. Under the integral separation assumption, the absolute errors of computed Lyapunov exponents can be estimated by a bound that has essentially the same magnitude as the local integration errors that can be controlled.

The error analysis for the Lyapunov exponents combines two parts: backward error analysis and forward error analysis. We briefly summarize the idea and the main results in [21] below.

Backward error analysis. The main aim of the backward error analysis is to show that the exact realization of the QR methods can be interpreted as the solution of a piecewise constant and upper triangular differential system, while the numerical realization can be interpreted as the solution of a perturbed system. Theorem 23 and Theorem 25 in [21] state that the perturbation arising in the coefficient matrix has the same magnitude as the local discretization error for the discrete/continuous QR method.

Forward error analysis. In the forward error analysis for the discrete/continuous QR method presented in [21], we consider an implicitly given linear time-varying system of upper triangular form together with a perturbed system, where the perturbations in the coefficients are small and can be estimated. Assuming the integral separation and some further boundedness conditions, Theorem 31 and Corollary 32 in [21] give a bound for the gaps between the Lyapunov exponents of the unperturbed system and those of the perturbed one. This bound can be computed explicitly by using the bounds of the perturbations.

Since the main error source in the QR methods is the error arising from numerical integration, under the assumptions stated in the backward and forward error analysis, we can conclude that the error of the Lyapunov exponents has the same order of magnitude as the local error tolerance. Namely, if we use an integrator of order p and consider only discretization errors arising from numerical integration, then the errors of the Lyapunov exponents in both the QR methods have magnitude $\mathcal{O}(h^p)$, where $h = \max_{j\geq 1} h_j$ is the maximal stepsize.

It is important to note that unfortunately we cannot deal with the error caused by the time termination, i.e., the error arising from truncating the semi-infinite interval to a finite one when approximating the limit as $t \to \infty$.

4.2 Error analysis of the linearization

By extending the error analysis for nonlinear ODEs, see [7], we will consider error analysis for semi-linear DAEs. Our goal is to compute the Lyapunov exponents of the semi-linear problem

$$E(t)x' = f(t,x),\ x(0) = x_0, \tag{30}$$

with the reference solution $x(t) = \phi(t, x_0)$.

To compute the Lyapunov exponents of (30), we would like to obtain the Lyapunov exponents for the linear variational problem

$$E(t)X'(t) = \frac{\partial f}{\partial x}(t, \phi(t, x_0))X(t),\ \text{i.e., } E(t)X'(t) = A(t)X(t), \tag{31}$$

subject to some initial conditions X_0.

By the exact QR factorizations on the problem (31) and the error analysis in [21, Section 4.1], we would find a sequence $\{t_j\}$ such that

$$X(t_j) = Q(t_j)R(t_j, t_{j-1})\dots R(t_2, t_1)R(t_1, t_0)R_0. \tag{32}$$

In practice, we will have a numerical approximation to the reference solution $\psi_h(t, x_0)$ instead of $\phi(t, x_0)$. Thus, we will end up attempting to approximate the Lyapunov exponents of the problem

$$E(\tau)Y'(\tau) = \frac{\partial f}{\partial x}\big(\tau, \psi_h(\tau, x_0)\big)Y(\tau), \text{ i.e., } E(\tau)Y'(\tau) = \tilde{A}(\tau)Y(\tau), \tag{33}$$

and by the exact QR factorizations on the problem (33), we would find a sequence of $\{\tau_j\}$ such that

$$Y(\tau_j) = U(\tau_j)V(\tau_j, \tau_{j-1})\dots V(\tau_2, \tau_1)V(\tau_1, \tau_0)V_0. \tag{34}$$

Assume that there exist smooth monotone functions $\omega_j(t)$ such that

$$\omega_j(t_j) = \tau_j, \;\; \omega_j(t_{j+1}) = \tau_{j+1},$$

and define $\omega(t) = \omega_j(t)$ for all $t \in [t_j, t_{j+1})$. We further assume that for all $t \geq 0$, there exist constants $\varepsilon > 0$ and $\delta > 0$ such that

$$\begin{aligned} &(a) \quad |\omega(t) - t| \leq \delta, \\ &(b) \quad \|\psi_h(\omega(t), x_0) - \phi(t, x_0)\| \leq \varepsilon. \end{aligned} \tag{35}$$

Here, δ and ε are proportional to the absolute local error in approximating the semilinear DAE system.

In addition to the errors arising from the backward and forward error analysis for the linear DAE system, we face the error arising from the difference between two linear problems (31) and (33), that is, we will compare the matrix functions A and $\tilde{A}$.

To address this issue, analogously to the ODE case [7], we have the following theorem.

Theorem 4.1. *Suppose that (35) holds and let the second-order derivatives f_{xt} and f_{xx} exist and be continuous. Then, we have*

$$\|\tilde{A}(\tau) - A(t)\| \leq M\delta + N\varepsilon, \tag{36}$$

where M and N are bounds on the second order derivatives f_{xt} and f_{xx}, respectively.

Proof. By the integral calculus, we have that

$$\begin{aligned} & f_x\big(\omega(t), \psi_h(\omega(t), x_0)\big) - f_x\big(t, \phi(t, x_0)\big) \\ = \;& f_x\big(\omega(t), \psi_h(\omega(t), x_0)\big) - f_x\big(t, \psi_h(\omega(t), x_0)\big) + f_x\big(t, \psi_h(\omega(t), x_0)\big) - f_x\big(t, \phi(t, x_0)\big) \\ = \;& \int_0^1 f_{xt}\big(t + s(\omega(t) - t), \psi_h(\omega(t), x_0)\big)ds\big(\omega(t) - t\big) \\ & + \int_0^1 f_{xx}\Big(t, \phi(t, x_0) + s\big(\psi_h(\omega(t), x_0) - \phi(t, x_0)\big)\Big)ds\big(\psi_h(\omega(t), x_0) - \phi(t, x_0)\big), \end{aligned}$$

for $0 \leq s \leq 1$ and $t \geq 0$.

So, applying the estimates in (35), we obtain a bound

$$\begin{aligned}
& \|f_x(\omega(t),\psi_h(\omega(t),x_0)) - f_x(t,\phi(t,x_0))\| \\
\leq\ & \max_{0\leq s\leq 1} \|f_{xt}(t+s(\omega(t)-t),\psi_h(\omega(t),x_0))\| |\omega(t)-t| \\
& + \max_{0\leq s\leq 1} \left\| f_{xx}\Big(t,\phi(t,x_0)+s\big(\psi_h(\omega(t),x_0)-\phi(t,x_0)\big)\Big)\right\| \|\psi_h(\omega(t),x_0)-\phi(t,x_0)\|.
\end{aligned}$$

That is

$$\|\tilde{A}(\tau)-A(t)\| \leq M\delta + N\varepsilon.$$

□

The bound (36) combined with the stability result for Lyapunov exponents implies that for $M\delta+N\varepsilon$ small enough uniformly in t, the Lyapunov exponents of systems (31) and (33) are close to each other.

5 Numerical experiments

In this section, we carry out some numerical experiments with both continuous and discrete QR algorithms using the Dormand-Prince method and the ESDIRK methods as described in Section 3. The algorithms have been implemented by Matlab version R2022a and the numerical results are obtained on a computer with Intel CPU core I5 processor 2.7 GHz.

To test the performance of the numerical methods, we consider several examples of nonlinear strangeness-free DAEs and apply the QR algorithms to approximate their Lyapunov exponents. Here we present the numerical results for a real-life example and a constructive one. For numerical integration, we use the HERK method based on the Dormand-Prince pair and several ESDIRK methods from [15, 16].

Example 5.1. *Consider the semi-explicit DAE system which models a nonlinear damp-spring system in [6]*

$$\begin{aligned}
x_1' &= x_2, \\
x_2' &= -\frac{k_1}{M}x_1 - \frac{k_2}{M}x_1^3 - \frac{b}{M}x_2 + \frac{x_4}{M}, \\
0 &= x_2 - rx_3, \\
0 &= -\frac{k_2}{M}x_1^3 + \left(\frac{2r^2}{J}-\frac{1}{M}\right)bx_2 - \frac{k_1}{M}x_1 + \left(\frac{r^2}{J}+\frac{1}{M}\right)x_4,
\end{aligned}$$

with initial conditions $x_1(0)=1$, $x_2(0)=1$, $x_3(0)=\frac{1}{2}$, $x_4(0)=0$, *and parameter values taken from [6],* $k_1=1$, $k_2=1$, $b=2$, $r=2$, $M=1$, $J=4$.

We can rewrite the above system as follows

$$E(t)x'(t) = f(x),$$

where $x(t)=(x_1(t),x_2(t),x_3(t),x_4(t))^T$ and

$$E(t)=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix} \text{ and } f(t,x)=\begin{bmatrix} x_2 \\ -\frac{k_1}{M}x_1-\frac{k_2}{M}x_1^3-\frac{b}{M}x_2+\frac{x_4}{M} \\ x_2-rx_3 \\ -\frac{k_2}{M}x_1^3+\left(\frac{2r^2}{J}-\frac{1}{M}\right)bx_2-\frac{k_1}{M}x_1+\left(\frac{r^2}{J}+\frac{1}{M}\right)x_4 \end{bmatrix}.$$

Table 1 and Table 2 display the computed Lyapunov exponents with respect to different values of the error tolerance TOL by using the half-explicit Dormand-Prince method.

Table 1. Lyapunov exponents by the discrete QR algorithm using Dormand-Prince method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17903157	-2.82030775	4580	0.558963
5000	10^{-4}	-0.17750595	-2.82221670	22907	1.888204
10000	10^{-4}	-0.17731527	-2.82245528	45816	3.891060
1000	10^{-5}	-0.17902793	-2.82048083	7029	0.699758
5000	10^{-5}	-0.17750506	-2.82238425	35156	2.899508
10000	10^{-5}	-0.17731470	-2.82262218	70316	6.150663
1000	10^{-6}	-0.17902253	-2.82050542	10890	1.099596
5000	10^{-6}	-0.17750396	-2.82240051	54464	4.749011
10000	10^{-6}	-0.17731415	-2.82263738	108931	9.135048

Table 2. Lyapunov exponents by the continuous QR algorithm using Dormand-Prince method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17980518	-2.82074422	865	0.256192
5000	10^{-4}	-0.17766056	-2.82244931	4277	0.832777
10000	10^{-4}	-0.17739246	-2.82266247	8543	1.395277
1000	10^{-5}	-0.17981842	-2.82072801	874	0.259642
5000	10^{-5}	-0.17766273	-2.82244646	4287	0.906540
10000	10^{-5}	-0.17739351	-2.82266108	8553	1.365671
1000	10^{-6}	-0.17982136	-2.82071778	892	0.274772
5000	10^{-6}	-0.17766358	-2.82244421	4305	0.845633
10000	10^{-6}	-0.17739399	-2.82265991	8570	1.458471

We also implement several ESDIRK methods presented in [15] in the discrete QR algorithm. The numerical results show that the ESDIRK methods require more computational time than the HERK method.

Table 3. Lyapunov exponents by the discrete QR algorithm using ESDIRK34 method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17902421	-2.82038368	10020	0.707572
5000	10^{-4}	-0.17750535	-2.82227923	50113	2.565906
10000	10^{-4}	-0.17731551	-2.82251616	100229	5.098090
1000	10^{-5}	-0.17902248	-2.82049556	18245	1.032116
5000	10^{-5}	-0.17750399	-2.82239049	91256	4.610991
10000	10^{-5}	-0.17731418	-2.82262734	182520	9.095078
1000	10^{-6}	-0.17902231	-2.82050608	32877	1.810772
5000	10^{-6}	-0.17750394	-2.82240085	164448	8.193071
10000	10^{-6}	-0.17731414	-2.82263770	328913	16.396372

Table 4. Lyapunov exponents by the discrete QR algorithm using ESDIRK54b method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17902458	-2.81535892	2202	0.263587
5000	10^{-4}	-0.17750439	-2.81724595	10955	0.982761
10000	10^{-4}	-0.17731437	-2.81748183	21988	1.84532
1000	10^{-5}	-0.17902306	-2.81948801	3177	0.493237
5000	10^{-5}	-0.17750398	-2.82139569	15913	2.015899
10000	10^{-5}	-0.17731416	-2.82163186	31819	3.922165
1000	10^{-6}	-0.17902233	-2.82048569	6973	0.659497
5000	10^{-6}	-0.17750394	-2.82238048	34857	2.800313
10000	10^{-6}	-0.17731414	-2.82261734	69713	5.513546

Example 5.2. *Now, we construct a new strangeness-free system by transforming the previous example as follows. First, we take the system*

$$\begin{bmatrix} E_{11}(t) & 0 \\ 0 & 0 \end{bmatrix} x'(t) = \begin{bmatrix} E_{11}(t)f(t,x) \\ g(t,x) \end{bmatrix},$$

where $E_{11}(t) = \begin{bmatrix} \cos(\gamma_1 t) & \sin(\gamma_1 t) \\ -\sin(\gamma_1 t) & \cos(\gamma_1 t) \end{bmatrix}$ *and*

$$f(t,x) = \begin{bmatrix} x_2 \\ -x_1 - x_1^3 - 2x_2 + x_4 \end{bmatrix}, \; g(t,x) = \begin{bmatrix} x_2 - 2x_3 \\ -x_1 - x_1^3 + 2x_2 + 2x_4 \end{bmatrix}.$$

We then change the variables by setting $x(t) = Q(t)y(t)$*, where* $Q(t)$ *is a Givens rotation*

$$Q(t) = \begin{bmatrix} \cos(\gamma_2 t) & 0 & 0 & \sin(\gamma_2 t) \\ 0 & \cos(\gamma_3 t) & \sin(\gamma_3 t) & 0 \\ 0 & -\sin(\gamma_3 t) & \cos(\gamma_3 t) & 0 \\ -\sin(\gamma_2 t) & 0 & 0 & \cos(\gamma_2 t) \end{bmatrix}.$$

Thus, the above system becomes a semi-linear DAE of the form

$$\begin{bmatrix} \hat{E}_1(t) \\ 0 \end{bmatrix} y'(t) = \begin{bmatrix} \hat{f}(t,y) \\ \hat{g}(t,y) \end{bmatrix},$$

where

$$\begin{aligned} \hat{E}_1(t) &= \begin{bmatrix} E_{11}(t) & 0 \end{bmatrix} Q(t),\ \hat{g}(t,y) = g(t,Qy), \\ \hat{f}(t,y) &= E_{11}(t) f(t,Q(t)y) - [E_{11}(t) \quad 0] Q'(t) y. \end{aligned}$$

To get a linear variational system, we linearize the above system along the given solution y^* to obtain

$$\begin{bmatrix} \hat{E}_1(t) \\ 0 \end{bmatrix} u' = \begin{bmatrix} \hat{A}_1(t) \\ \hat{A}_2(t) \end{bmatrix} u,$$

where

$$\begin{aligned} \hat{A}_1(t) &= \frac{\partial \hat{f}(t,y^*)}{\partial y} = E_{11}(t) \frac{\partial f(t,x^*)}{\partial x} Q(t) - [E_{11}(t) \quad 0] Q'(t), \\ \hat{A}_2(t) &= \frac{\partial \hat{g}(t,y^*)}{\partial y} = \frac{\partial g(t,x^*)}{\partial x} Q(t). \end{aligned}$$

For our numerical tests, we use different values for parameters γ_1, γ_2, γ_3 and the initial conditions

$$y_1(0) = 1,\ y_2(0) = 1,\ y_3(0) = \frac{1}{2},\ y_4(0) = 0.$$

Clearly the Lyapunov exponents of this example are the same as those of the previous example.

In the following tables, we display the interval length T, different error tolerances TOL, the computed Lyapunov exponents, the number of steps, and the CPU-time to approximate Lyapunov exponents by the discrete and the continuous QR algorithms.

In Tables 5-6, we present numerical results by the discrete QR algorithm for this strangeness-free system with two different parameter sets $\gamma_1 = 1$, $\gamma_2 = 2$, $\gamma_3 = 3$ and $\gamma_1 = 20$, $\gamma_2 = 100$, $\gamma_3 = 200$.

Table 5. Lyapunov exponents by the discrete QR algorithm using Dormand-Prince method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17903161	-2.82059790	3985	0.943338
5000	10^{-4}	-0.17750554	-2.82240853	19917	4.506534
10000	10^{-4}	-0.17731478	-2.82274736	39833	9.271679
1000	10^{-5}	-0.17902655	-2.82050775	6087	1.552909
5000	10^{-5}	-0.17750475	-2.82240921	30421	6.987843
10000	10^{-5}	-0.17731453	-2.82264689	60839	14.020309
1000	10^{-6}	-0.17902252	-2.82050758	9374	2.272484
5000	10^{-6}	-0.17750396	-2.82240265	46842	10.616184
10000	10^{-6}	-0.17731415	-2.82263953	93678	21.575532

Table 6. Lyapunov exponents by the discrete QR algorithm using Dormand-Prince method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17976496	-2.82127345	31043	10.012391
5000	10^{-4}	-0.17824721	-2.82317224	155155	48.464997
10000	10^{-4}	-0.17805749	-2.82340959	310295	98.683108
1000	10^{-5}	-0.17911753	-2.82061094	47209	16.454477
5000	10^{-5}	-0.17759861	-2.82250735	235948	86.814337
10000	10^{-5}	-0.17740875	-2.82274440	471871	165.901471
1000	10^{-6}	-0.17903376	-2.82052018	71619	25.547329
5000	10^{-6}	-0.17751530	-2.82241518	357912	122.833276
10000	10^{-6}	-0.17732549	-2.82265206	715778	251.898163

Next, the numerical results by the continuous QR method using Dormand-Prince method are displayed in Tables 7-8.

Table 7. Lyapunov exponents by the continuous QR algorithm using Dormand-Prince method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17937204	-2.82119132	2013	0.744567
5000	10^{-4}	-0.17716789	-2.82294473	10032	3.195832
10000	10^{-4}	-0.17689251	-2.82316380	20055	6.388512
1000	10^{-5}	-0.17985206	-2.82070347	3135	1.237828
5000	10^{-5}	-0.17765247	-2.82245862	15627	5.089826
10000	10^{-5}	-0.17737755	-2.82267799	31242	10.198042
1000	10^{-6}	-0.17986002	-2.82068455	4921	1.791315
5000	10^{-6}	-0.17767114	-2.82243776	24537	7.992353
10000	10^{-6}	-0.17739754	-2.82265691	49058	15.725143

Table 8. Lyapunov exponents by the continuous QR algorithm using Dormand-Prince method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17926189	-2.82126759	29114	14.871598
5000	10^{-4}	-0.17709000	-2.82301589	145501	73.598495
10000	10^{-4}	-0.17681852	-2.82323442	290984	139.676315
1000	10^{-5}	-0.17976240	-2.82076474	47571	23.700261
5000	10^{-5}	-0.17758862	-2.82251680	237794	123.334587
10000	10^{-5}	-0.17731691	-2.82273581	475572	228.881655
1000	10^{-6}	-0.17982984	-2.82069480	75801	37.847378
5000	10^{-6}	-0.17766016	-2.82244476	378931	180.256558
10000	10^{-6}	-0.17738895	-2.82266351	757844	366.559093

We also test the efficiency of ESDIRK34 and ESDIRK54b methods presented in [15]. The numerical results obtained by the discrete QR algorithm for the case $\gamma_1 = 1$, $\gamma_2 = 2$, $\gamma_3 = 3$ are displayed in Tables 9-10.

Table 9. Lyapunov exponents by the discrete QR algorithm using ESDIRK34 method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17901947	-2.82067699	8565	1.639997
5000	10^{-4}	-0.17750054	-2.82257306	42771	8.083445
10000	10^{-4}	-0.17731068	-2.82281006	85528	16.039974
1000	10^{-5}	-0.17902206	-2.82052288	15471	3.155167
5000	10^{-5}	-0.17750359	-2.82241784	77267	14.694354
10000	10^{-5}	-0.17731379	-2.82265469	154511	28.652010
1000	10^{-6}	-0.17902232	-2.82050865	27759	5.586281
5000	10^{-6}	-0.17750390	-2.82240349	138649	25.909193
10000	10^{-6}	-0.17731410	-2.82264033	277261	51.264427

Table 10. Lyapunov exponents by the discrete QR algorithm using ESDIRK54b method

T	TOL	λ_1	λ_2	steps	CPU-time (s)
1000	10^{-4}	-0.17903531	-2.82113820	3143	1.201963
5000	10^{-4}	-0.17751485	-2.82303791	15695	6.077862
10000	10^{-4}	-0.17732486	-2.82327528	31384	11.717520
1000	10^{-5}	-0.17902337	-2.82054659	5566	2.274696
5000	10^{-5}	-0.17750471	-2.82244189	27802	10.467966
10000	10^{-5}	-0.17731489	-2.82267878	55596	20.886199
1000	10^{-6}	-0.17902250	-2.82051025	9297	3.633157
5000	10^{-6}	-0.17750400	-2.82240518	46449	17.483210
10000	10^{-6}	-0.17731420	-2.82264204	92890	34.767049

A comparison of the results in Tables 5-10 reveals the efficiency and robustness of both QR methods. For the case of slowly varying coefficients (small values of the parameters γ_i), the continuous QR method tends to be faster. However, for larger parameter values, both methods require more steps and CPU time, with the discrete method being slightly more efficient. ESDIRK methods require more computational time in these examples, but they are recommended for stiff problems. Additionally, implementing ESDIRK methods within the continuous QR approach is more complex due to the nonlinear matrix equations that arise from their implicit nature.

Remark 5.3. *If a DAE system can be analytically transformed into an equivalent ODE, the QR-based algorithms developed by Dieci and Van Vleck [9, 11] may be applied directly to compute Lyapunov exponents. In such cases, these established ODE methods are expected to outperform the DAE-oriented approaches proposed in this work, as the transformation eliminates the singularity and reduces the system dimension. However, for general DAEs, deriving an analytically equivalent ODE is typically infeasible, and thus the system must be addressed in its original differential–algebraic form.*

6 Conclusions

In this paper, we have proposed QR methods that incorporate half-explicit Runge-Kutta (HERK) methods for numerical integration to compute Lyapunov exponents of semi-linear DAEs. Error control in the simultaneous integration of both the semi-linear DAE and its linearized system is achieved using embedded pairs, such as Dormand-Prince and ESDIRK methods, following the approach in [22]. We have analyzed the errors in Lyapunov exponent approximations using both discrete and continuous QR methods, demonstrating that the Lyapunov exponent error is essentially controlled by the local integration error. Numerical experiments confirm the efficiency of QR methods and validate the error analysis.

These methods can be extended to general nonlinear DAEs and sensitivity analysis. More research is needed to refine the continuous QR algorithm. Furthermore, developing robust software packages and testing them on large-scale DAE problems from real-world applications remains an important future direction.

Acknowledgement

The authors sincerely thank anonymous referees for useful suggestions and comments that led to improvements in the paper.

Funding

This work is supported by the National Foundation for Science and Technology Development (NAFOSTED), Grant No. 101.02-2021.43.

Author contributions

The authors are equally involved, responsible, and listed by name alphabetically.

Competing interests

The authors declare that they have no competing interests.

References

[1] L. Y. Adrianova. *Introduction to linear systems of differential equations*. Vol. 146. Trans. Math. Monographs: AMS, Providence, RI, 1995.

[2] A. Alamodi, K. Sun, Y. Peng. Chaotic attractor with varied parameters. *The European Physical Journal Special Topics*. 2020;229:1095–1108.

[3] U. M. Ascher, L. R. Petzold. *Computer methods for ordinary differential equations and differential - algebraic equations*. SIAM, Philadelphia, 1998.

[4] G. Cassoni, A. Zanoni, A. Tamer, P. Masarati. Stability analysis of nonlinear rotating systems using Lyapunov characteristic exponents estimated from multibody dynamics. *J. Comput. Nonlinear Dynam.* 2023;18(8): 081002.

[5] J. L. Daleckii, M. G. Krein. *Stability of solutions of differential equations in Banach spaces*. American Mathematical Society: Providence, RI, 1974.

[6] P. Di Franco, G. Scarciotti, A. Astolfi. Stability of nonlinear differential-algebraic systems via additive identity. *IEEE/CAA J. Automatica Sinica*. 2020;7:929–941.

[7] L. Dieci, R. D. Russell, E. Van Vleck. On the computation of Lyapunov exponents for continuous dynamical systems. *SIAM J. Appl. Math.* 1997;34:402–423.

[8] L. Dieci, E. Van Vleck. Computation of a few Lyapunov exponents for continuous and discrete dynamical systems. *Appl. Numer. Math.* 1995;17:275–291.

[9] L. Dieci, E. Van Vleck. Lyapunov and other spectra: a survey. *In: Collected Lectures on the Preservation of Stability Under Discretization (Fort Collins, CO, 2001)*. SIAM, Philadelphia. 2002:197–218.

[10] L. Dieci, E. Van Vleck. Lyapunov and Sacker-Sell spectral intervals. *J. Dyn. Diff. Equ.* 2007;19:265–293.

[11] L. Dieci, E. Van Vleck. Lyapunov spectral intervals: theory and computation. *SIAM J. Numer. Anal.* 2002;40:516–542.

[12] L. Dieci, E. Van Vleck. On the error in computing Lyapunov exponents by *QR* methods. *Numer. Math.* 2005;101:619–642.

[13] L. Dieci, E. Van Vleck. Pertubation theory for approximation of Lyapunov exponents by *QR* methods. *J. Dyn. Diff. Equ.* 2006;18:815–842.

[14] A. González-Zumba, P. Fernández De Córdoba, J.-C. Cortés, V. Mehrmann. Stability assessment of stochastic differential-algebraic systems via Lyapunov exponents with an application to power systems. *Mathematics*. 2020;8(9):1–26.

[15] J. Jørgensen, M. Kristensen, P. Thomsen. A family of ESDIRK integration methods. *arXiv:1803.01613v1 [math.NA]*. 2018;5 Mar:22.

[16] M. Kristensen, J. Jørgensen, P. Thomsen, S. Jørgensen. An ESDIRK method with sensitivity analysis capabilities. *Computers and Chemical Engineering*. 2004;28:2695–2707.

[17] P. Kunkel, V. Mehrmann. *Differential-Algebraic Equations Analysis and Numerical Solution*. Zurich: European Mathematical Society, 2006.

[18] V. H. Linh, V. Mehrmann. Approximation of spectral intervals and leading directions for differential-algebraic equations via smooth singular value decompositions. *SIAM J. Numer. Anal.* 2011;49:1810–1835.

[19] V. H. Linh, V. Mehrmann. Efficient integration of strangeness-free non-stiff differential-algebraic equations by half-explicit methods. *J. Comput. Appl. Math.* 2014;262:346–360.

[20] V. H. Linh, V. Mehrmann. Lyapunov, Bohl and Sacker-Sell spectral intervals for differential-algebraic equations. *J. Dyn. Diff. Equ.* 2009;21:153–194.

[21] V. H. Linh, V. Mehrmann, E. Van Vleck. *QR* methods and error analysis for computing Lyapunov and Sacker-Sell spectral intervals for linear differential-algebraic equations. *Adv. Comput. Math.* 2011;35:281–322.

[22] V. H. Linh, N. D. Truong. Runge-Kutta methods revisited for a class of structured strangeness-free differential-algebraic equations. *Electr. Trans. Num. Anal.* 2018;48:131–135.

[23] P. Masarati, A. Tamer. Sensitivity of trajectory stability estimated by Lyapunov characteristic exponents. *Aerospace Science and Technology*. 2015;47:501–510.

[24] P. Masarati. Estimation of Lyapunov exponents from multibody dynamics in differential-algebraic form. *Proceedings Inst. Mech. Eng., Part K: Journal of Multi-body Dynamics*. 2013;227:23–33.

[25] P. Q. Tuyen. Some Runge-Kutta algorithms with error control and variable stepsizes for solving a class of differential-algebraic equations. *VNU Journal of Science: Mathematics-Physics*. 2020;36:104–119.

Vu Hoang Linh
Faculty of Mathematics, Mechanics and Informatics
Vietnam National University
334 Nguyen Trai, Thanh Xuan, Hanoi, Vietnam
linhvh@vnu.edu.vn
ORCID: 0000-0002-8098-9810

Nguyen Duy Truong
Department of Basic Science
Tran Quoc Tuan University
Son Tay, Hanoi, Viet Nam
truong.nguyenduy80@gmail.com

Phan Quang Tuyen
Department of Basic Science
College of Artillery Officer's Training
Son Tay, Hanoi, Viet Nam
quangtuyen.ktqs@gmail.com