Computation of Lyapunov characteristic exponents for continuous dynamical systems

Firdaus E. Udwadia and Hubertus F. von Bremen

Abstract. A new method for computing all the Lyapunov characteristic exponents (LCEs) of n-dimensional continuous dynamical systems is presented. The method relies on the use of the Cayley Transform and a development of the ability to restart the computations for the variational equation. The method *intrinsically* maintains the orthogonality of the Q matrix in the QR decomposition of the solution of the variational equation. It therefore does not suffer from the type of computational breakdown that occurs with the standard method for computing LCEs. An example of a Lorenz system showing the breakdown of the standard method is presented, and the same example is used to compute the LCEs by the present method. Comparisons of the computational efficiency of the proposed method in relation to the standard method, and the standard-method-with-reorthogonalization are presented. Issues of accuracy are addressed.

Keywords. Continuous n-dimensional dynamical systems, computation of all Lyapunov exponents, Cayley transformation, preservation of orthogonality.

1. Introduction

Lyapunov characteristic exponents were originally introduced by Lyapunov [1] in the context of non-stationary solutions of ordinary differential equations (also see, Sansone and Conti [2]). They provide a way to characterize the asymptotic behavior of nonlinear dynamical systems by giving a measure of the exponential growth (or shrinkage) of perturbations about a nominal trajectory. Since they measure the sensitivity of solutions of dynamical systems to small perturbations, they are often used as indicators of chaotic motion when the dynamics occurs on an invariant set (see Eckmann and Ruelle [3]). Positive LCEs are thus often used to establish chaos; all the LCEs of a non-chaotic system are negative.

The increased interest in exploring the possibility of chaotic motions in all sorts of dynamical systems has spawned a concomitant interest in computational and experimental methods for determining LCEs, since most often, it is the occurrence of positive LCEs that signals the presence of chaotic motions (see for example Froeschle et al. [4], Schmid and Dunkin [5], Udwadia and Raju [6], Efimov et al. [7]).

Benettin et al. [8], in their two-part paper, provide a computational procedure for determining the LCEs of a dynamical system. Using arguments motivated by a geometrical approach to the problem, they propose a Gram-Schmidt orthogonalization type procedure (see Benettin et al. [9]). Their computational procedure relies on the fundamental Multiplicative Ergodic Theorem of Oseledec [10]. Johnson et al. [11] provide an alternative way of proving this theorem, and in the process they indicate how the LCEs can be obtained computationally through a QR decomposition of the fundamental matrix solution of the variational equation that corresponds to a given dynamical system. Geist et al. [12] provide a comparison of different methods used for computing LCEs.

The determination of the LCEs of continuous dynamical systems begins with their variational equation, which is a time-variant linear system of ordinary differential equations. The fundamental matrix solution, $Y(t)$ is expressed uniquely as $Y(t) = Q(t)R(t)$, where $Q(t)$ is an orthogonal matrix, and $R(t)$ is an upper-triangular matrix all of whose diagonal elements are positive. As shown by Bennettin et al. [8] and Johnson et al. [11], the information on the LCEs is extracted using such a decomposition from the diagonal elements of the matrix $R(t)$. It has been known for some time now, though it is not commonly admitted by many, that during the numerical computations, the matrix $Q(t)$ is prone to lose its orthogonal nature (see Geist et al. [12]). The loss of orthogonality when considering continuous dynamical systems can then result in not just the inaccurate determination of LCEs, but also a complete breakdown of the computations (due to overflows and underflows). Though perhaps known to some researchers in the field, such a complete breakdown has, it appears, never been actually reported in the literature to date. In this paper we show an example of this occurring.

While the loss of orthogonality has been recognized as the source of numerical problems in the computation of LCEs, there are no numerical methods available to date that can be implemented in a straightforward manner and generally guarantee the orthogonality of $Q(t)$. Dieci et al. [13, 14[1]] propose two approaches: the use of automatic unitary integrators (mainly as a theoretical possibility), and the use of projected schemes. The automatic unitary integrators are specialized integrators developed to ensure the orthogonality of $Q(t)$. These methods however as stated by Dieci et al. [13] and Dieci and Van Vleck [15] are in general difficult to implement, and they choose the projected schemes instead when dealing with nonlinear systems. The projected scheme basically consists of using an integrator combined with an MGS factorization of $Q(t)$ after each integration step. However, despite the additional computational burden of such a scheme, there is still no guarantee that the computed orthogonal matrix $Q(t)$ is the correct one. Numerical results shown in section 5 also indicate that the projected scheme could consequently lead to errors in the computed LCEs. Rangarajan et al. [16] provide a method which they claim maintains orthogonality of $Q(t)$, but

---

[1] The authors thank one of the reviewers for pointing out this recent reference, unavailable when the paper was submitted.

their method seems to be amenable to low order (3-dimensional) systems only, as does the recent method of Udwadia and von Bremen [17].

In this paper we present a method for computing all the LCEs of a general n-dimensional continuous dynamical system. The method is based on the Cayley transform. It *intrinsically* preserves the orthogonality of $Q(t)$ without the use of ad hoc procedures like reorthogonalization of the columns of $Q(t)$. Since orthogonality is preserved, no breakdown occurs, and the computation of LCEs is, in general, more accurately carried out. The method requires (for large n) the solution of just about half the number of differential equations required by the standard method.

This paper is structured as follows. Section 2 gives the standard method for computing LCEs of continuous dynamical systems, and here we also establish our notation. Section 2.1 gives a numerical example that shows the actual breakdown of such a computational scheme. In Section 3.1 we prove a useful lemma that is crucial to the use of the method developed in this paper. Section 3.2 gives our method for determining the LCEs of general n-dimensional systems. In Section 4 we revisit the example of Section 2.1, which fails when using the standard method. Section 5 provides some insights into the computational efficiency and numerical accuracy of our method. In Section 6 we give our conclusions and remarks.

2. Computation of LCEs

Consider the dynamical system

$$\dot{y}(t) = f\left(y(t)\right), \qquad y(0) = y_0, \tag{1}$$

where $y \in R^n$ and $t \in R^+$. The variational equation associated with this dynamical system is given by

$$\dot{Y}(t) = J(t)Y(t), \qquad Y(0) = I_n, \tag{2}$$

where $Y \in R^{n \times n}$ and $J(t)$ is the n by n Jacobian matrix $\left[\frac{\partial f_i}{\partial y_j}\right]$ evaluated at the point $y(t)$ along the trajectory of (1). We note that since $Y(0)$ is nonsingular, $Y(t)$ is nonsingular for $t \geq 0$.

We assume that the regularity conditions are adequately satisfied [2] and the n Lyapunov exponents λ_i are then defined as the logarithms of the eigenvalues of the matrix

$$\Lambda_y = \lim_{t \to \infty} \left[Y^T(t)Y(t)\right]^{1/(2t)}. \tag{3}$$

The unique QR-factorization $Y(t) = Q(t)R(t)$, where all the diagonal elements of $R(t)$ are positive, when substituted into equation (2) yields

$$\dot{Q}(t)R(t) + Q(t)\dot{R}(t) = J(t)Q(t)R(t), \qquad Q(0)R(0) = I_n. \tag{4}$$

---

[2]   As pointed out by Dieci et al. [13], it is difficult to verify the regularity of a given system.

Premultiplying the differential equation in (4) by $Q^T(t)$ and postmultiplying it by $R^{-1}(t)$ gives

$$Q^T(t)\dot{Q}(t) + \dot{R}(t)R^{-1}(t) = Q^T(t)J(t)Q(t), \qquad Q(0) = I_n, R(0) = I_n. \quad (5)$$

Since $Q^T(t)Q(t) = I_n$, the first member on the left in equation (5) is a skew symmetric matrix. Furthermore, since $R(t)$ is upper triangular, $\dot{R}(t)R^{-1}(t)$ is upper triangular. If we now define the matrix $S(t)$ as having the elements

$$S_{i,j}(t) = \begin{cases} \left(Q^T(t)J(t)Q(t)\right)_{i,j}, & i > j \\ \\ 0, & i = j \quad , \\ \\ -\left(Q^T(t)J(t)Q(t)\right)_{j,i}, & i < j \end{cases} \quad (6)$$

then from equation (5), $S(t) = Q^T(t)\dot{Q}(t)$. This then yields the differential equation governing the matrix $Q(t)$ as

$$\dot{Q}(t) = Q(t)S(t), \qquad Q(0) = I_n, \quad (7)$$

where the elements of the matrix $S(t)$ are defined in (6). Again, since $Q^T(t)\dot{Q}(t)$ is skew symmetric, from equation (5) the differential equations for $\rho_i(t) = \ln(R_{i,i}(t))$ are

$$\dot{\rho}_i(t) = \left(Q^T(t)J(t)Q(t)\right)_{i,i}, \qquad \rho_i(0) = 0, \qquad i = 1, 2, \ldots, n. \quad (8)$$

The time evolution of the LCEs is now given by $\lambda_i(t) = \rho_i(t)/t$, and the LCEs are obtained as (see, Geist et al. [12])

$$\lambda_i = \lim_{t \to \infty} \lambda_i(t). \quad (9)$$

We note that the determination of the LCEs requires the determination of the n^2 elements of the matrix $Q(t)$ using the n^2 differential equations given in (7). In addition to solving these n^2 differential equations, one is required to: (a) obtain the trajectory of the dynamical system by integrating the n equations given by (1), and (b) integrate the n equations given by (8) to obtain $\rho_i(t)$. Therefore the total number of differential equations required to be solved for determining all n LCEs of the system (1) is $n(n+2)$.

Since $Q(t)$ is required to be orthogonal, its columns (obtained by integrating (7)) must always be orthonormal. It is in the maintenance of this orthonormality among the columns of $Q(t)$ that straightforward numerical schemes when used to compute LCEs appear to fare poorly, and could even breakdown completely.

2.1 Breakdown of the standard method for computation of LCEs

For brevity, in this paper we shall refer to the method outlined in Section 2 for computing LCEs as the standard method. As mentioned before, it has been known for some time now that errors caused by the lack of orthogonality of the matrix $Q(t)$ lead to erroneous results in the computation of LCEs when using the standard method. However, what appears to be left unstated in the literature to date is that not only can the standard method yield inaccurate LCE results, but also it can completely breakdown. We illustrate this with the following example system.

Consider the Lorenz system given by

$$
\begin{bmatrix} \dot{y}_1(t) \\ \dot{y}_2(t) \\ \dot{y}_3(t) \end{bmatrix} = \begin{bmatrix} \sigma(y_2 - y_1) \\ \alpha y_1 - y_1 y_3 - y_2 \\ y_1 y_2 - \beta y_3 \end{bmatrix}, \quad \begin{bmatrix} y_1(0) = 0 \\ y_2(0) = 1 \\ y_3(0) = 0 \end{bmatrix}, \tag{10}
$$

with the parameter values, $\alpha = 45.92$, $\beta = 4$, and $\sigma = 16$. Before LCE computations are initiated, the Lorenz system is integrated for 500 time units using the ode45 routine that is available in MATLAB (with a relative error tolerance of 10^{-7}, and a component-wise absolute error tolerance of 10^{-8}).

The LCEs are computed using the standard method with a fixed time-step size (Δt), 4th order Runge-Kutta scheme. The LCEs are computed using different values of Δt. For each value of Δt, the computations eventually breakdown. Figure 1 shows the times, T_B, at which this breakdown occurs for each value of Δt considered. The results from numerical simulations are recorded as experimental points. The curve corresponds to the least-squares fit of these points. The experimental points appear to lie along the curve given by $T_B = 0.0469\Delta t^{-2.61}$ obtained from the curve fit.

Figure 2(a) shows the time evolution of the LCEs, $\lambda_i(t)$, prior to break down when using the step-size $\Delta t = 0.01$. The figure shows that the smallest LCE does not converge to a definite value. Figure 2(b) shows the logarithm of the error in orthogonality of $Q(t)$ defined as $e_0(t) = \left\| Q^T(t)Q(t) - I \right\|_2$; Figure 2(c) shows the error in the sum of the LCEs defined as $e_s(t) = Tr[J(t)] - \sum_{i=1}^{3} \lambda_i(t)$. Ideally, both these errors must be zero. The figures show that these errors can indeed become significant – significant enough to cause the standard method to breakdown, in this case at $t = 8{,}475.8$. Recall that the LCEs are obtained, theoretically speaking, by allowing the duration of integration to go to infinity.

Having illustrated the breakdown of the standard method, we next present a method that guarantees that $Q(t)$ will be orthogonal when computing the LCEs for general n-dimensional dynamical systems.
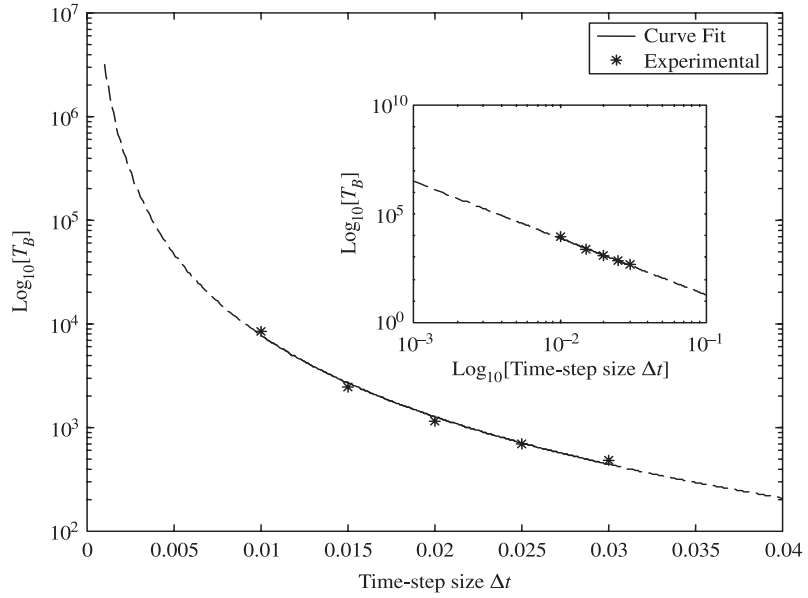
Figure 1. Logarithm of the time when the standard method fails ($\log_{10}[T_B]$) versus the time-step size Δt .
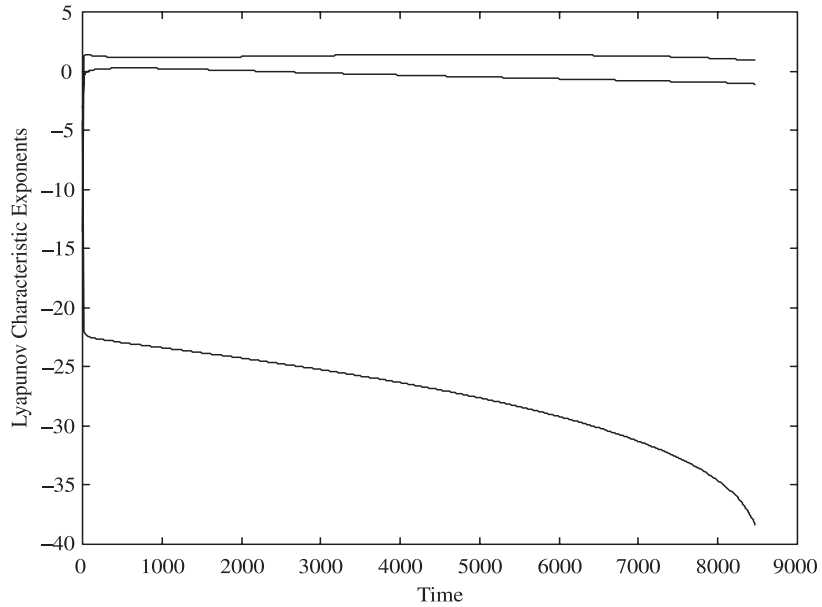

Figure 2. (a) Computed LCEs as a function of time, using the standard method for the Lorenz system with $\Delta t = 0.01$.
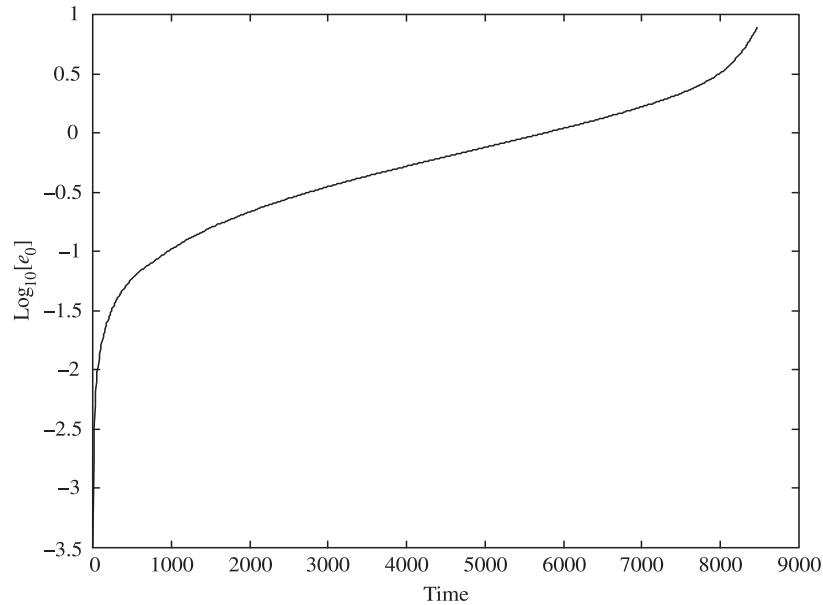
Figure 2. (b) Logarithm of the error $e_0(t) = \left\|Q^T(t)Q(t) - I\right\|_2$ in orthogonality as a function of time, using the standard method for the Lorenz system with $\Delta t = 0.01$.

3. Computation of LCEs of n-dimensional systems using the Cayley method

At present there appears to be no general computational approach available that guarantees that throughout the duration of integration, the matrix $Q(t)$ is maintained orthogonal. In this section we develop an approach that maintains the orthogonality of $Q(t)$. We provide the explicit equations that need to be numerically solved to obtain the LCEs of an n-dimensional dynamical system.

3.1 A useful Lemma

We begin by proving a result that will be heavily used in the development of the computational scheme developed in this section.

Consider the variational equation

$$\dot{Y}(t) = J(t)Y(t), \quad t > t_0, \quad Y(t_0) = Q_0 R_0. \tag{11}$$

We shall take Q_0 to be orthogonal, and R_0 to be upper-triangular with all its diagonal elements positive. Subdivide the real line $t > t_0$ into subintervals

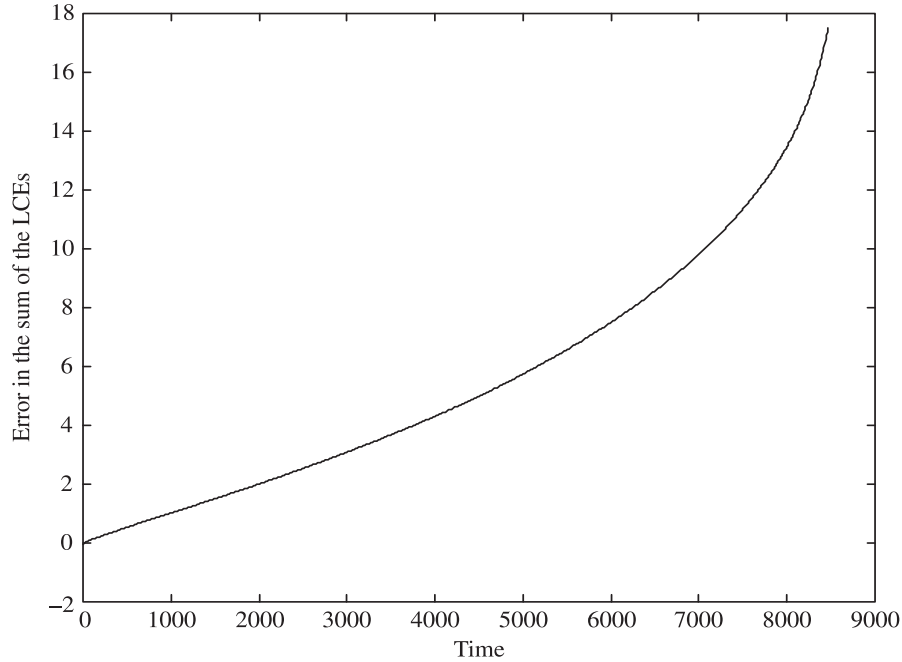$$t_i \le t \le t_{i+1}, \qquad i = 0, 1, 2, \ldots. \tag{12}$$

Figure 2. (c) Error in the sum of the LCEs ($e_s(t)$) as a function of time, using the standard method for the Lorenz system with $\Delta t = 0.01$.

each of length $\Delta t_i = t_{i+1} - t_i$, $i = 0, 1, 2,\dots$. Denote the solution $Y(t)$ of (11) by

$$Y(t) = Q(t)R(t) \tag{13}$$

so that

$$Y(t_i) = Q(t_i)R(t_i) \equiv Q_i R_i, \quad i = 0, 1, 2, \dots . \tag{14}$$

Lemma 1. *At any time* $t = t_i + \tau$, $0 \leq \tau \leq \Delta t_i$, $i = 0, 1, 2,\dots$, *the solution of equation (11) can be expressed as*

$$Y(t) = Y(t_i + \tau) = Q_i \tilde{Y}(\tau)R_i = Q_i \tilde{Q}(\tau)\tilde{R}(\tau)R_i, \quad 0 \leq \tau \leq \Delta t_i, \quad t_i \leq t \leq t_{i+1}. \tag{15}$$

Here $\tilde{Y}(\tau) = \tilde{Q}(\tau)\tilde{R}(\tau)$ *is the solution of the differential equation*

$$\dot{\tilde{Y}}(\tau) = \tilde{J}(\tau)\tilde{Y}(\tau), \quad 0 \leq \tau \leq \Delta t_i, \quad \tilde{Y}(0) = I_n, \qquad i = 0, 1, 2,\dots \tag{16}$$

so that $\tilde{Q}(0) = I_n, \tilde{R}(0) = I_n$, *and*

$$\tilde{J}(\tau) = Q_i^T J(t_i + \tau)Q_i. \tag{17}$$

Proof. Clearly the solution given by equation (15) satisfies (11) at $\tau = 0$ since, by (15)

$$Y(t_i) = Q_i \tilde{Q}(0)\tilde{R}(0)R_i = Q_i \tilde{Y}(0)R_i = Q_i R_i, \qquad (18)$$

as required by (14). The last equality in (18) follows from (16). Also by (15)

$$Y(t) = Y(t_i + \tau) = Q_i \tilde{Q}(\tau)\tilde{R}(\tau)R_i = Q_i \tilde{Y}(\tau)R_i, \quad 0 \le \tau \le \Delta t_i. \qquad (19)$$

Differentiating (19) with respect to τ we get for $0 \le \tau \le \Delta t_i$ and $t_i \le t \le t_{i+1}$,

$$\dot{Y}(t) = Q_i \dot{\tilde{Y}}(\tau)R_i = Q_i Q_i^T J(t_i + \tau)Q_i \tilde{Y}(\tau)R_i = J(t)Q_i \tilde{Y}(\tau)R_i = J(t)Y(t), \quad (20)$$

where we have made use of the fact that $dt/d\tau = 1$, the second equality follows from relations (16) and (17), and the last equality is because of relation (15). Equation (20) is the same as equation (11), and equation (18) is the same as (14); we have thus shown that over each interval of time Δt_i, equation (15) is the solution of the variational equation (11). $\qquad \square$

Remark 1. This lemma provides us, from a geometrical viewpoint, of the capability of changing the local coordinate system over the time interval Δt_i, $i = 0, 1, 2, \ldots$, so that at the beginning of each subinterval, the n-dimensional volume element in phase space has its edges aligned along the local, orthogonal coordinate axes.

3.2 The Cayley method

The Cayley transformation [18] provides a one-to-one relation between an n by n orthogonal matrix $Q(t)$, and an n by n skew symmetric matrix $K(t)$. The relation is

$$Q(t) = (I_n - K(t))(I_n + K(t))^{-1} = 2(I_n + K(t))^{-1} - I_n, \qquad (21)$$

and it is valid as long as none of the eigenvalues of $Q(t)$ equal –1. In fact, when $\|K(t)\| < 1$, the inverse in equation (21) can be expanded in a convergent series in powers of $K(t)$. We note that $K(0) = 0 \Leftrightarrow Q(0) = I_n$.

Using the relation for $Q(t)$ from (21) in equation (5) yields (see Appendix 1)

$$-2H^T(t)\dot{K}(t)H(t) + \dot{R}(t)R^{-1}(t) = H^T(t)\hat{J}(t)H(t), \quad K(0) = 0, R(0) = I_n \quad (22)$$

where we have denoted the n by n matrix

$$H(t) = [h_1 \quad h_2 \quad \ldots \quad h_n] = (I_n + K(t))^{-1} = G^{-1}(t), \qquad (23)$$

and, $\hat{J}(t) = G(t)J(t)G^T(t)$. But the matrix $S = H^T \dot{K} H$ is skew symmetric, and

$\dot{R}R^{-1}$ is upper-triangular, hence

$$S_{i,j}(t) = \begin{cases} -\frac{1}{2}\left(H^T(t)\hat{J}(t)H(t)\right)_{i,j}, & i > j \\[2mm] 0, & i = j \\[2mm] \frac{1}{2}\left(H^T(t)\hat{J}(t)H(t)\right)_{j,i}, & i < j \end{cases} \quad . \tag{24}$$

The differential equation governing the elements of $\dot{K}(t)$ is then given by

$$\dot{K}(t) = H^{-T}(t)S(t)H^{-1}(t) = G^T(t)S(t)G(t). \tag{25}$$

But the matrix $\dot{K}(t)$ is also skew symmetric and hence its elements are determined by computing the elements of the lower triangle of the matrix on the right hand side in equation (25) yielding

$$\dot{K}_{i,j}(t) = \left(G^T(t)S(t)G(t)\right)_{i,j}, i > j \tag{26}$$

Notice that equation (26) constitutes a set of $n_K = n(n-1)/2$ differential equations.

Going back to equation (22), and denoting $\rho_i(t) = \ln(R_{ii}(t))$, $i = 1, 2,...,n$, we obtain

$$\dot{\rho}_i(t) = h_i^T(t)\hat{J}(t)h_i(t), \quad \rho_i(0) = 0, \quad i = 1, 2, \ldots, n \tag{27}$$

from which the time evolution of the LCEs can be obtained as

$$\lambda_i(t) = \rho_i(t)/t, \quad i = 1, 2,...,n. \tag{28}$$

The LCEs are given by $\lambda_i = \lim_{t\to\infty} \lambda_i(t)$, $i = 1, 2,...,n$.

Equations (24)-(28) along with equation (1), summarize the computational procedure for obtaining the LCEs.

We note that these equations are valid as long as the transformation (21) remains valid. Since $Q(0) = I_n$ and $Q(t)$ is a continuous function of time, there is always an interval of time $0 \le t \le t_0$ in which no eigenvalue of $Q(t)$ equals –1. However, as the integration of equations (1), (24)-(28) proceeds, an eigenvalue of $Q(t)$ may well go though –1, and then the method would become inapplicable. In fact the matrix H (see equation (23)) will become ill-conditioned when one of the eigenvalues of $Q(t)$ approaches –1.

It is now that we invoke the result of Lemma 1. We segment the integration as follows. Let us say that we start at time $t = 0$ (note, $K(0) = 0$) and by using equations (1), (24)-(28) we integrate up to time $t = t_0$. Let us say that at time t_0 a suitable norm of $K(t_0)$ becomes equal to some (user-defined) pre-assigned value, η; that is, $\|K\| \le \eta < 1$, for t belonging to the interval $(0, t_0)$. At time t_0 we thus obtain the matrix $K(t_0)$, and from it we construct $Q_0 = Q(t_0)$ by equation (21). We also obtain $y_i(t_0) \equiv y_{t_0}$, $\rho_i(t_0)$, $i = 1, 2,...,n$.

To proceed beyond time t_0, we store $Q(t_0)$, and continue the integration for $t_0 \leq t \leq t_1$ using the following equation to track the trajectory of the dynamical system

$$\dot{y}(t) = f\left(y(t)\right), \quad y(t_0) = y_{t_0}. \tag{29}$$

We can thus compute the Jacobian, $J(t)$, along the trajectory.

For $\tau = t - t_0$, by Lemma 1 (see Equation (16)), we obtain

$$\frac{dY}{d\tau} = Q_0^T J(t) Q_0 Y(\tau) \equiv \tilde{J}(\tau) Y(\tau), \quad Y(\tau = 0) = I_n. \tag{30}$$

Equation (30) now "restarts" the variational equation at $\tau = 0$ (or $t = t_0$). Setting $Y(\tau) = Q(\tau)R(\tau)$ as before, we see that $Q(\tau = 0) = I_n$ and $R(\tau = 0) = I_n$. The matrix $Q(\tau = 0)$ thus satisfies the Cayley Transform requirement! In fact equation (30) is of the same form as equation (2) except with a modified Jacobian, $\tilde{J}(t)$. Hence the scheme given by equations (24)-(28) is again applicable using this modified Jacobian. As seen from Lemma 1, equation (15), the updated LCEs, $\hat{\rho}_i(t)$, can now be obtained from the relations

$$\hat{\rho}_i(t) = \hat{\rho}_i(t_0 + \tau) = \rho_i(t_0) + \rho_i(\tau), \quad t_0 \leq t \leq t_1, \quad 0 \leq \tau \leq \Delta t_0, \quad i = 1, 2, ..., n. \tag{31}$$

and the updated matrix $Q(t) = Q(t_0 + \tau) = Q(t_0)Q(\tau)$, $t_0 \leq t \leq t_1$.

As the integration of equation (31) proceeds, the eigenvalue of $Q(\tau)$ may start approaching –1 again (or alternatively $\|K\|$ may eventually equal η) and once again the procedure may need to be restarted at some time $t_1 > t_0$, and so on.

It should be noted that the number of differential equations that need to be solved using the approach described in this section is $2n + n_K = n(n+3)/2$. The standard method requires $2n + n^2 = n(n+2)$ differential equations to be solved for determining all the LCEs. For large n, we thus require to solve about half the number of differential equations required by the standard method (For details regarding comparisons of computational efficiency and accuracy, see Section 5).

For 3 dimensional systems, we present below the differential equations governing the elements of the matrix $\dot{K}(t)$. The skew symmetric matrix $K(t)$ may be taken as

$$K(t) = \begin{bmatrix} 0 & -a(t) & -b(t) \\ a(t) & 0 & -c(t) \\ b(t) & c(t) & 0 \end{bmatrix}, \tag{32}$$

so that the matrix $Q(t) = [q_1 \ q_2 \ q_3]$, obtained from equation (21), is given explic-

itly as

$$Q(t) = \frac{1}{\alpha} \begin{bmatrix} (1 - a^2 - b^2 + c^2) & 2(a - bc) & 2(ac + b) \\ -2(a + bc) & (1 - a^2 + b^2 - c^2) & 2(c - ab) \\ -2(b - ac) & -2(c + ab) & (1 + a^2 - b^2 - c^2) \end{bmatrix} \qquad (33)$$

where, $\alpha(t) = 1 + a^2(t) + b^2(t) + c^2(t)$.

Equation (25) then gives

$$\begin{bmatrix} \dot{a} \\ \dot{b} \\ \dot{c} \end{bmatrix} = -\frac{1}{2} \begin{bmatrix} (1 + a^2) & (ab + c) & (ac - b) \\ (ab - c) & (1 + b^2) & (a + bc) \\ (b + ac) & (bc - a) & (1 + c^2) \end{bmatrix} \begin{bmatrix} q_2^T J q_1 \\ q_3^T J q_1 \\ q_3^T J q_2 \end{bmatrix}, \qquad (34)$$

where q_i are the columns of the matrix $Q(t)$, which are explicitly given in equation (33).

Also, equations (27) now reduce to

$$\dot{\rho}_i(t) = q_i^T(t) J(t) q_i(t), \quad \rho_i(0) = 0, \quad i = 1, 2, 3, \qquad (35)$$

Equations (1), (34), and (35) now form a set of 9 differential equations for the determination of the 3 LCEs of the 3-dimensional system.

4. Numerical examples and remarks on computational procedure

In this section we provide a numerical example of the computational procedure described in the previous section by determining the LCEs of the same example discussed in Section 2.1 of the Lorenz system (equation (10)) which suffered a breakdown during computation with the standard method.

The system was allowed to evolve for 500 units of time. The succeeding 100 units were then used for computation of the LCEs. Figure 3(a) shows the phase plot of the system. In Figure 3(b) we show the time-evolution of the LCEs. The integration is done using a constant time-step, 4th order Runge-Kutta scheme. The time step used is 0.001. The variational equation is restarted so that during the computations $\|K\|_1$ is maintained to be less than 0.2.

Two measures of the error in the orthogonality of the matrix $Q(t)$ are considered: (1) $e_o = \|Q^T(t)Q(t) - I\|_2$, and (2) $e_D = ||Det(Q(t)| - 1|$. The latter measure seems to give about the same quality of information as the former, but
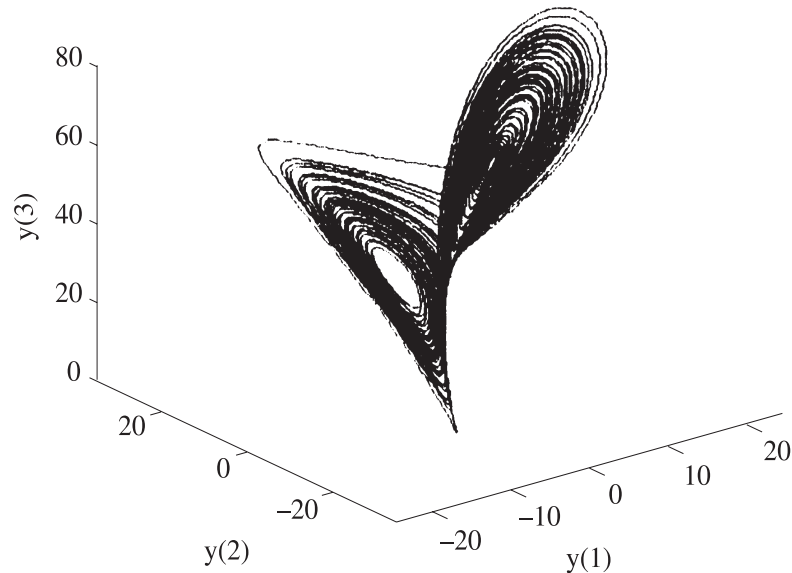
Figure 3. (a) Phase plot of the Lorenz system.

is substantially faster to compute. Logarithms of these measures are shown in Figures 3(c) and 3(d), indicating the accuracy of the proposed method.

For purposes of comparison, the errors in orthogonality that result by using the standard method for computation of the LCEs of the same system under identical conditions as before are shown in Figures 3(e) and 3(f).

A comparison of Figures 3(c, d) and 3(e, f) shows that the error in orthogonality of our proposed method is near the order of machine precision, and it is about 7 orders of magnitude smaller than that for the standard method; it is the build up of this error in orthogonality that causes the standard method to eventually break down.

5. Computational considerations, efficiency and accuracy

1. Orthogonlity of Q: Figures 3(c) and 3(d) point out that the errors in orthogonality in $Q(t)$ are indeed small. However, the method is constructed to maintain $Q(t)$ intrinsically orthogonal, and one might wonder why the error in orthogonality is not of the order of machine precision (10^{-16}). The reason is that when we restart the variational equation at, say, $t = t_i$, we have $Q(t) = Q(t_i)\tilde{Q}(\tau) = Q_i\tilde{Q}(\tau)$, $t \geq t_i$, where $\tilde{Q}(\tau)$ is the Q-matrix in the time segment (t_i, t_{i+1}), and t_{i+1} is the next time at which the variational equation needs to be restarted again. It is the round-off in carrying out this matrix product that seems to cause the error in
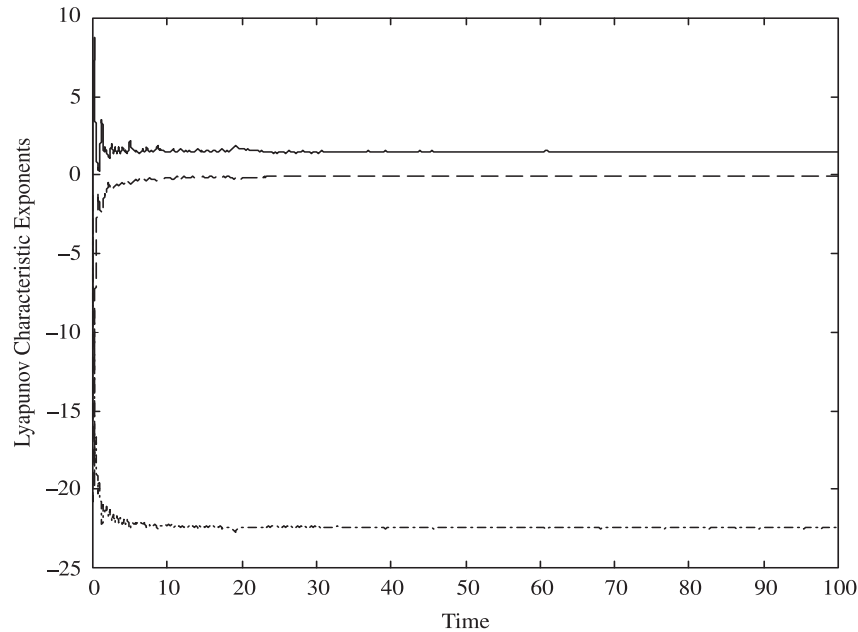
Figure 3. (b) LCEs of the Lorenz system.

the orthogonality of $Q(t)$ to differ from machine precision. This round off error is roughly proportional to the dimension of the matrix Q, to the unit error u, and the number of segmentations of the time axis required to keep $\|K(t)\| \leq \eta$ throughout the integration process. Over each individual time segment over which the integration is done, the error in orthogonality (i.e. error in $\tilde{Q}(\tau)$) is indeed found to be of the order of machine precision. In keeping with this observation, our experiments found that a ten-fold increase in the duration of time over which the LCEs are computed (say from 100 time units to 1000 time units) increases the error in orthogonality by roughly one unit on the Log scale shown in Figures 3.[3] Also, the Jacobian matrix over each segment needs to be pre- and post-multiplied by Q_i^T and Q_i respectively, another source of round-off error, again because of matrix multiplication. Hence, to reduce the roundoff errors caused by matrix multiplication, it is desirable to have as few segments as possible, i.e., have as few 'restarts' of the variational equation as possible. Larger values of $\eta(< 1)$ than the ones chosen for our numerical illustrations may therefore be better. Furthermore, very small values of η may lead to very small segments of time (t_i, t_{i+1})

---

[3] It should be pointed out that round off errors in the computation of LCEs of *discrete systems* caused by matrix multiplication is an issue that has gone largely unnoticed. Experience with continuous systems seems to indicate that this may be an important issue as far as the accuracy of the computed results is concerned, especially because, the LCEs are usually obtained by computations over numerous time steps.
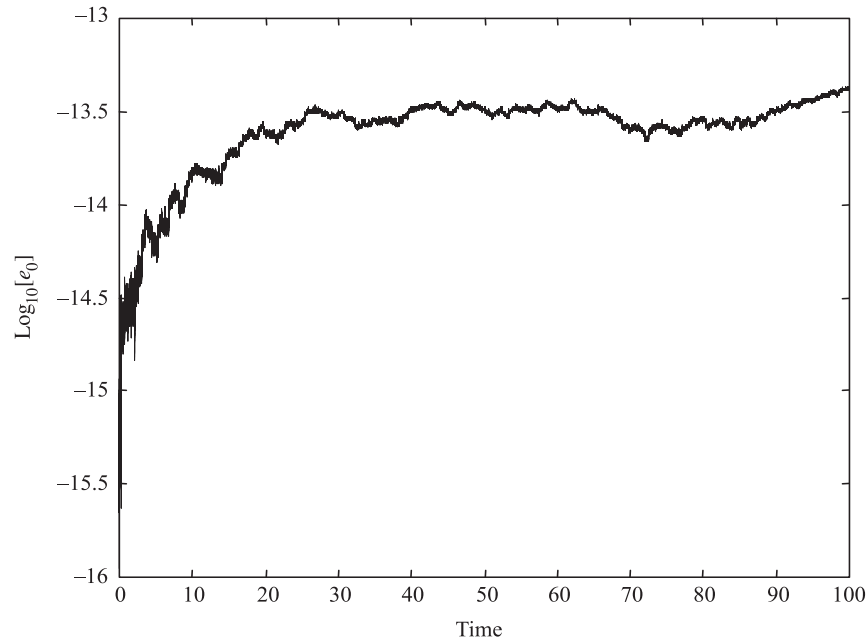
Figure 3. (c): Logarithm of the error $e_0(t) = \|Q^T(t)Q(t) - I\|_2$ in orthogonality as a function of time.

over which $\|K(t)\| \leq \eta$, requiring in turn smaller integration time steps, and thus also a consequent loss in computational efficiency. This problem can be averted to a great extend by using variable time step integration schemes (like MATLAB's ODE45) with prescribed relative and absolute error tolerances.

2. Operations Count for the Cayley Method and the Standard Method: The number of differential equations required to be solved by our method, for large n, is roughly half that needed by the standard method. This is indeed helpful in terms of memory storage for large dimensional systems. One can compare the efficiency of the different methods based on the number of floating point operations needed to compute the derivatives of the state vector for each method. The derivatives of the state vector are evaluated at each step during the integration process and thus they can be used as a measure of efficiency. The state vector basically has three parts, the first n components correspond to the trajectory, the next set of components correspond to the information required for obtaining Q, and the last n components correspond to the LCEs. Regardless of the method to be used, the portion where the trajectory is computed will be the same for all methods. Thus when comparing the numerical efficiencies of different methods, there is no need to include an operation count for this part. Similarly, for the methods that are compared here, the operation count corresponding to the LCEs is the same for each method, and so we do not include the operation count for this part in our
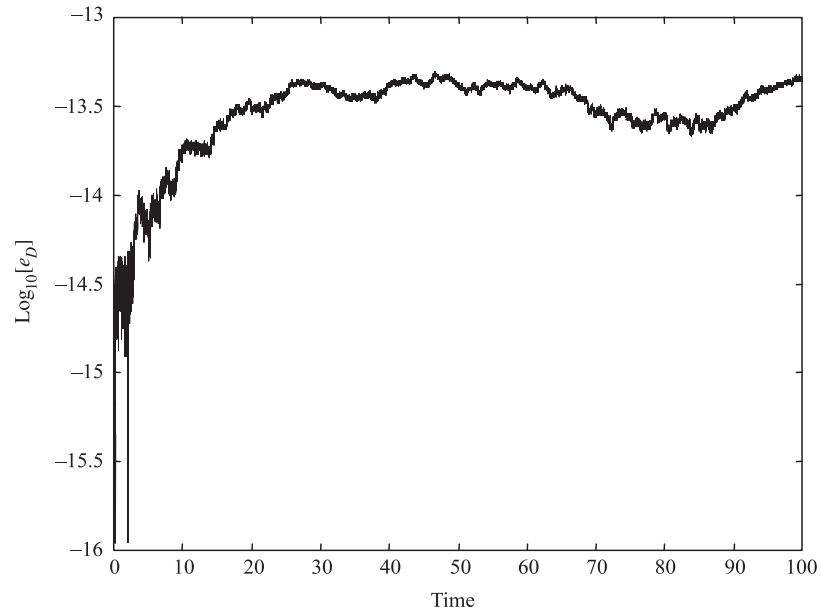
Figure 3. (d) Logarithm of the error $e_D = ||Det(Q(t)| - 1|$ in orthogonality as a function of time.
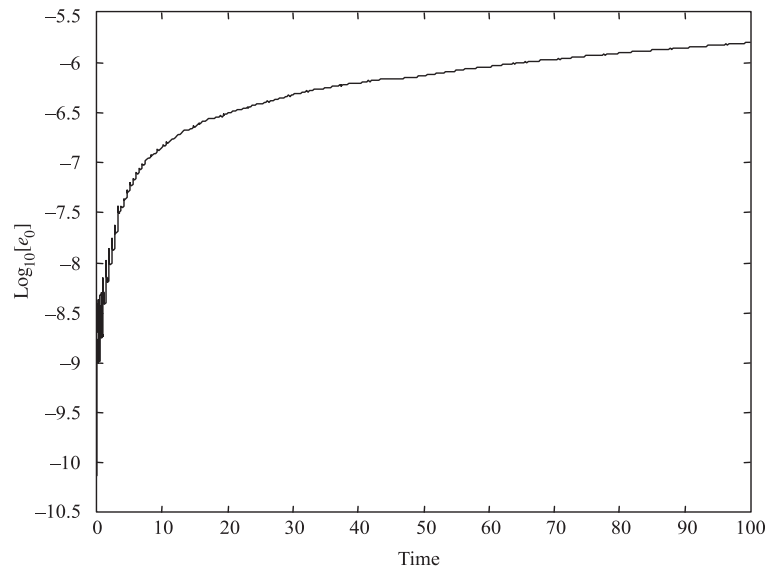

Figure 3. (e) Logarithm of the error $e_0(t) = \|Q^T(t)Q(t) - I\|_2$ in orthogonality as a function of time using the standard method.
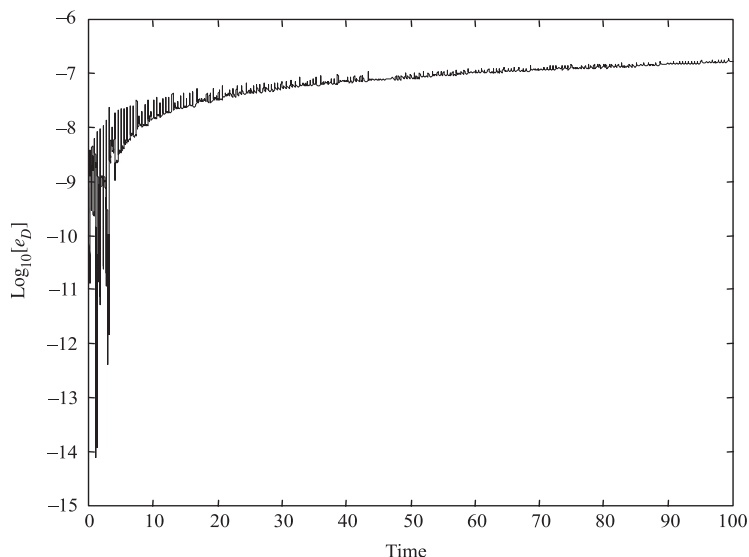
Figure 3. (f) Logarithm of the error $e_D = ||Det(Q(t)| - 1|$ in orthogonality as a function of time using the standard method.

comparison of the efficiencies.

Table 1 shows the number of operations needed to compute the derivative of the state vector for the portion corresponding to the Q matrix. Only higher order terms in the number of operations are shown in the table. Note that in the case of the Cayley method, only the upper triangular elements of K are part of the state vector (only $n(n-1)/2$ elements). By exploiting the structure of the matrices involved in the operations of the Cayley method one can obtain significant savings in the number of operations, see Appendix 2 for ways to save computations. On the otherhand, for the Standard Method, all the n^2 elements of Q are part of the state vector. The number of operations needed to compute the Jacobian of the system are not included in the operation count presented on Table 1, since the number of computations needed to establish the Jacobian of the system are the same for all the methods and depend only on the specific dynamical system.

The method here called the standard-method-with-reorthogonalization is basically the standard method where the Q matrix is reorthogonalized after each integration step. The reorthogonalization can be done using different methods, the values reflected on the table correspond to using the Modified Gram-Schmidt (MGS) method.

The table shows that asymptotically (for large n), the ratio of the operation count of the Cayley, to the Standard, to the Standard-method-with-reorthogonalization, is 9:5:7. Though the Cayley method is computationally more expensive than the other two methods, it has the advantage that the orthogonality of Q is

preserved. The standard-method-with-reorthogonalization preserves the orthogonality of Q, in an ad hoc manner; however the method will, in general, cause errors in the computed LCEs because the reorthogonalization process while enforcing the orthogonality of Q, leaves the R matrix unaltered.

Method	Multiplications/ Division	Additions/ Subtractions	Total
Cayley Method (Section 3.2)	$\frac{9}{2}n^3 - \frac{4}{2}n^2$	$\frac{9}{2}n^3 - \frac{8}{2}n^2$	$9n^3 - 6n^2$
Standard Method (Section 2)	$\frac{5}{2}n^3 - \frac{3}{2}n^2$	$\frac{5}{2}n^3 - \frac{3}{2}n^2$	$5n^3 - 3n^2$
Standard-Method-with-Reorthogonalization	$\frac{7}{2}n^3 - \frac{1}{2}n^2$	$\frac{7}{2}n^3 - \frac{3}{2}n^2$	$7n^3 - 2n^2$

Table 1. Operation count needed to evaluate the part of the derivative of the state vector corresponding to Q for the Cayley, the Standard, and the standard-method- with-reorthogonalization.

3. Accuracy of Computed LCEs: As a comparative study of the accuracy of the different methods we consider a 3-dimensional dynamical system whose variational equation has the matrix solution $Y(t)$ given by:

$$Y(t) = Q_z(\theta_1 t)Q_y(\theta_2 t)Q_x(\theta_3 t)R(t) = Q(t)R(t),\qquad(36)$$

where Q_z is an orthogonal matrix of rigid body rotation about the z-axis, with rotation magnitude $\theta_1 t$; Q_y is an orthogonal matrix of rotation about the y-axis with rotation magnitude $\theta_2 t$; Q_x is an orthogonal matrix of rotation about the x-axis with rotation magnitude $\theta_3 t$; and, the upper triangular matrix $R(t)$ is taken to be

$$R(t) = \begin{bmatrix} e^{l_1 t} & \sin(t) & t \\ 0 & e^{l_2 t} & t^2 \\ 0 & 0 & e^{l_3 t} \end{bmatrix}.\qquad(37)$$

Thus the Jacobian matrix, $J(t)$, can be constructed using equation (4) as

$$J(t) = (\dot{Q}(t)R(t) + Q(t)\dot{R}(t))R^{-1}(t)Q^T(t).\qquad(38)$$

We use this specially constructed Jacobian in the variational equation, to compute

the LCEs by the various methods. Since the exact LCEs for the system are now
known to be l_1, l_2, and l_3, (and at each time t, the exact matrix $Q(t)$ is
also known) one can perform tests on the accuracy of the different methods for
computing LCEs.

In the numerical example, we use the following parameter values: $\theta_1 = 1, \theta_2 = 2, \theta_3 = 3$; and, $l_1 = 0.2, l_2 = 0.05, l_3 = -0.25$. We compare the standard
method, the standard-method-with-reorthogonalization, and the method proposed
in this study. The integration for each of the methods was done with MATLAB's
variable time-step integrator ODE45 using a relative tolerance of 10^{-8}, and an
absolute tolerance of 10^{-9} for each component. Figure 4 shows the logarithm of
$e_c(t) = \|Q^c(t) - Q(t)\|_2$ where $Q^c(t)$ is the computed matrix, and $Q(t)$ is the
exact matrix known from equation (36). The MGS method is used to perform
reorthogonalization; rather than do the reorthogonalization after computing each
point on the trajectory, the reorthogonalization is done every 0.05 units of time.
As seen from the figure, the computed $Q^c(t)$ using the standard method diverges
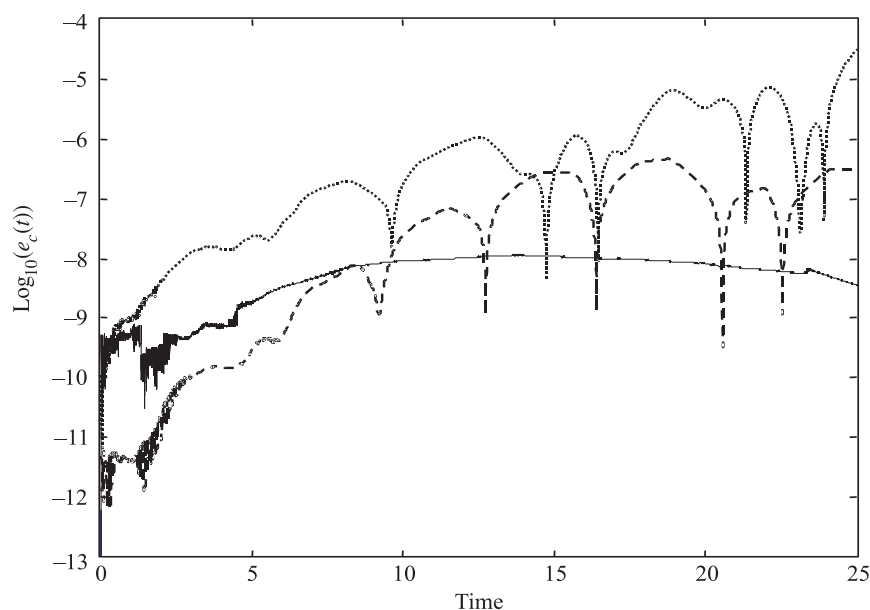with time from the exact $Q(t)$.

Figure 4. The dotted line shows the logarithm of the error $e_c(t)$ with time obtained using the
Standard method; the dashed line shows the same error in Q after reorthogonalization every
0.05 units of time using MGS; and, the solid line shows the error using the Cayley Transform.

Also, though the reorthogonalization enforces the matrix $Q^c(t)$ to be orthog-
onal, this orthogonal matrix *differs* from the exact matrix $Q(t)$; the error again
appears to show a tendency to increase with time. The $Q^c(t)$ computed by the
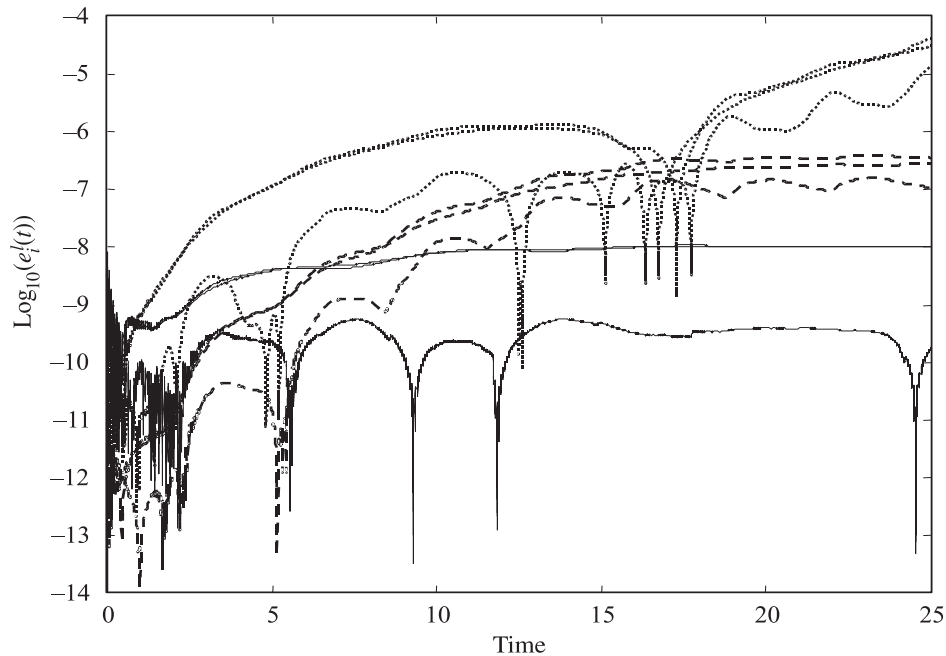
Figure 5. The dotted lines are the logarithms of the errors $e_i^l(t) = |\lambda_i^c(t) - l_i(t)|$ in the 3 LCEs obtained using the Standard Method; the dashed lines show the same errors using reorthogonalization of the Q matrix with MGS at every 0.05 units of time; and, the solid lines show the errors using the Cayley Transform method of Section 3.2. Lines with the sharp 'dips' correspond to the largest LCE whose value is 0.2.

approach presented in Section 3.2 (with $\eta = 0.7$) does not appear to increase, and is consistent with the integration error tolerances.

Figure 5 shows the logarithm of the errors $e_i^l(t) = |\lambda_i^c(t) - l_i(t)|$ where $\lambda_i^c(t)$ is the i-th computed LCE and $l_i(t)$ is the exact value. We observe that the errors in computation of the LCEs using the standard method rapidly rise, errors with additional reorthogonalization are less, though they have the tendency to increase with time, and those using our approach appear to be smaller and non-increasing over the time range considered. As seen from the figure, for computing the largest LCE the method proposed herein is superior in accuracy to the standard-method-with-reorthogonalization by at least 2 orders of magnitude. Our method provides more accurate results for the remaining two LCEs by about 1.5 orders of magnitude.

6. Conclusions and discussion

We have the following conclusions and remarks.

1. In this paper we have presented a method for computing *all* the LCE's

of an n-dimensional dynamical system using QR-decomposition [4]. The method *intrinsically* preserves the orthogonality of the matrix $Q(t)$, and thereby averts the problem of breakdown (which could occur with the standard method) caused by the loss of this orthogonality. Furthermore, no ad hoc reorthogonalization of Q is required in order to maintain its orthogonality. The breakdown of the standard method for computing LCEs is illustrated using the example of a Lorenz system.

2. The computational scheme proposed in this paper relies on two central ideas: (1) the Cayley transformation which allows us to express the orthogonal matrix $Q(t)$ in terms of a skew symmetric matrix $K(t)$, and (2) the method of restarting the computations for the variational equation when an eigenvalue of Q approaches –1. Since the scheme preserves orthogonality, the estimates of the LCEs it produces are more accurate, in general, than those obtained from the standard method.

3. For large n, the method proposed here requires the solution of about half the number of differential equations required by the standard method. The actual operations count for evaluating the derivative of part of the state vector (the part corresponding to the elements needed to obtain Q) is $9n^3 - 6n^2$ for the Cayley Method and $5n^3 - 3n^2$ for the Standard Method. The important difference between the standard method for computing LCEs and our approach is that our computational scheme *intrinsically* preserves orthogonality, a key issue in the accurate computation of LCEs when using the QR-decomposition.

4. To compare the accuracy of the computed LCEs using our proposed method, the standard method, and the standard-method-with-reorthogonalization, we construct an example whose LCEs are exactly known. For the example considered, the Cayley method proposed herein yields LCEs that are more accurate by about 1 to 2 orders of magnitude than those obtained using the standard-method-with-reorthogonalization. Ad hoc reorthogonalization of the Q matrix keeps it orthogonal; but we show that such an ad hoc procedure could cause the computed Q matrix to deviate from the exact Q matrix of the variational system of equations. Thus despite maintaining orthogonality, inaccurate LCE estimates may result.

5. Understanding the dynamics of certain nonlinear mechanical and structural systems often requires models with several 1000's of degrees of freedom. Even with the reduction that our method generates in the number of differential equations to be solved as compared to the standard method, due to computational resource constraints our method may still be difficult to use for finding the LCEs for many large-scale systems that arise in structural dynamics and computational mechanics. To provide qualitative insights into the dynamics of such large order systems, one needs to device methods for computing a *few* LCEs while preserving the orthogonality of the relevant vectors. Work in this direction is currently ongoing and will be reported when completed.

---

[4] Computer code to compute the LCEs is available at http://www.usc.edu/go/DynCon.

Appendix 1

We show here that: (1) $Q^T(t)\dot{Q}(t) = -2H^T(t)\dot{K}(t)H(t)$, and
    (2) $Q^T(t)J(t)Q(t) = H^T(t)\hat{J}(t)H(t)$
    where $H(t) = (I + K(t))^{-1} = G(t)^{-1}$, and $\hat{J}(t) = G(t)J(t)G^T(t)$.

Proof.
(1) Differentiating with respect to t,

$$Q(t) = (I - K(t))(I + K(t))^{-1}, \tag{A1.1}$$

we get

$$\dot{Q}(t) = -\dot{K}(t)(I + K(t))^{-1} + (I - K(t))\frac{d}{dt}\left[(I + K(t))^{-1}\right] \tag{A1.2}$$
$$= -\dot{K}(t)(I + K(t))^{-1} - (I - K(t))(I + K(t))^{-1}\dot{K}(t)(I + K(t))^{-1}.$$

Since $K(t)$ is skew symmetric, using (A1.2) we get

$$Q^T(t)\dot{Q}(t) = -(I - K(t))^{-1}(I + K(t))\left\{I + (I - K(t))(I + K(t))^{-1}\right\} \times$$
$$\dot{K}(t)(I + K(t))^{-1}$$
$$= -\left\{(I - K(t))^{-1}(I + K(t)) + (I - K(t))^{-1}(I - K(t))(I + K(t))(I + K(t))^{-1}\right\} \times$$
$$\dot{K}(t)(I + K(t))^{-1} = -\left\{(I - K(t))^{-1}(I + K(t)) + I\right\}\dot{K}(t)(I + K(t))^{-1}$$
$$= -(I - K(t))^{-1}\left\{I + K(t) + I - K(t)\right\}\dot{K}(t)(I + K(t))^{-1}$$
$$= -2(I - K(t))^{-1}\dot{K}(t)(I + K(t))^{-1} = -2H^T(t)\dot{K}(t)H(t). \tag{A1.3}$$

In the second equality above, we have used the fact that the matrices $(I - K(t))$ and $(I + K(t))$ commute with each other.

(2) Noting that $Q(t) = (I - K(t))(I + K(t))^{-1}$, and $K(t)$ is skew symmetric, the result is obvious.   □

Appendix 2
Observations on how to minimize the number of operations needed in the Cayley method.

In the process of computing the right hand side of the matrix $\dot{K}$ in the computation of the LCEs using the Cayley method one can save computations by: (I) using the form $Q = 2(I_n + K)^{-1} - I_n$ in equation (21) and not the form $Q = (I_n - K)(I_n + K)^{-1}$ (using the first form will save n^3 multiplications and n^3 additions); (II) using the special structure of the matrices in equation (26) will lead to a reduction of $\frac{1}{2}n^3$ multiplications and the same number of additions.

Since the matrix S in equation (26) is skew symmetric, it can be expressed as

$$S = U - U^T, \tag{A2.1}$$

where U is an upper triangular matrix with zeros along its main diagonal. Equation (26) can then be written as

$$\dot{K} = G^T S G = G^T (U - U^T) G = G^T U G - G^T U^T G = A - A^T. \qquad (A2.2)$$

Since the matrix $\dot{K}$ is skew symmetric, we only need to compute the elements above the main diagonal of $\dot{K}$. Basically then one has to compute the elements of $A = G^T U G$ that are above the main diagonal; with this in mind it is clear that the last row of the product $G^T U$ does not enter into the computations as well as the first column of G. The matrix U is upper triangular with zeros along the main diagonal; using this fact, it takes $\frac{n(n-1)^2}{2}$ multiplications and the same number of additions to compute the first n-1 rows of $G^T U$. Additionally, since U is upper triangular with zeros along the main diagonal, the first column of $G^T U$ has all entries equal to zero. Using the last mentioned fact and that only the elements above the main diagonal of A are needed, yields an additional $\frac{n(n-1)^2}{2}$ multiplications and additions when performing the product $[G^T U]G$. The total number of operations needed to compute the elements above the main diagonal of $\dot{K}$ is: $n(n-1)^2$ multiplications, and $n^3 - \frac{3}{2}n^2 - \frac{1}{2}n$ additions/subtractions (note that $\frac{n(n-1)}{2}$ subtractions are needed to get the upper triangular elements of $A - A^T$). In contrast, if one would not exploit the structure of S (and make use of the form $S = U - U^T$), one would need to use $\frac{3n^2(n-1)}{2}$ multiplications (and the same number of additions).

Acknowledgements

References

[1] Lyapunov, A. M., *Problème Général de la Stabilité du Mouvement*. Princeton University Press, 1947 (report of the French translation in 1907 of the Russian memoire dated 1892).
[2] Sansone G. and Conti R., *Non-linear Differential Equations*, Pergamon Press, Oxford/New York, 1964.
[3] Eckmann J. P. and D. Ruelle., Ergodic theory of chaos and strange attractors, *Rev. Mod. Phys.* **57** (1985), 617-656.
[4] Froeschle C., Lega E., and Gonczi R., Fast Lyapunov indicators-application to asteroidal motion, *Celestial Mechanics and Dynamical Astronomy* **67** (1997), 41-62.
[5] Schmid G., and Dunkin R., Indications of nonlinearity, intra-individual specificity and stability of human EEG – The unfolding dimension, *Physica D* **93** (1996), 165-190.
[6] Udwadia F. E. and Raju N., Dynamics of coupled nonlinear maps and its application to ecological modeling, *Appl. Math. Computation* **82** (1997), 137-179.
[7] Efimov V., Prusov A. and Shokurov M., Seasonal instability of pacific sea-surface-temperature anomalies, *Quarterly Journal of the Royal Meteorological Society* **123** suppl. B (1997), 337-356.

[8] Benettin G., Galgani L., Giorgilli A. and Strelcyn J.M., Lyapunov characteristic exponents for smooth dynamical systems and for hamiltonian systems; a method for computing all of them, Part I and Part II, *Meccanica* **15** (1980), 9-30.

[9] Benettin G., Galgani L., Giorgilli A. and Strelcyn J.M., Tous les nombres characteristiques de Lyapunov sont effectivement calculables, *Comptes-Rendues Acad. Sc. Paris* **286A** (1978), 431-433.

[10] Oseledec V., A multiplicative ergodic theorem. Lyapunov characteristic numbers for dynamical systems, *Trudy Mosk. Mat. Obsc.* **19**, 179 [*Moscow Math. Soc.* **19** (1968), 197].

[11] Johnson R., Palmer K., and Sell G., Ergodic properties of linear dynamical systems, *Siam J. Math Anal.* **18** (1987), 1-33.

[12] Geist K., Parlitz U., and Lauterborn W., Comparison of different methods for computing Lyapunov exponents, *Prog. Theor. Phys.* **83** (1990), 875-893.

[13] Dieci L., Russell R. and Van Vleck E., On the computation of Lyapunov exponents for continuous dynamical systems, *SIAM J. Numer. Anal.* **34** (1997), 402-423.

[14] Dieci L., Van Vleck E., Computation of orthonormal factors for fundamental solution matrices, *Dynamical Systems, Numerical Math.* **83** (1999), 599-620.

[15] Dieci L. and Van Vleck E., Computation of a few Lyapunov exponents for continuous and discrete dynamical systems, *Applied Numerical Mathematics* **17** (1995), 275-291.

[16] Rangarajan G., Habib S. and Ryne R., Lyapunov exponents without rescaling and reorthogonalization, *Physical Rev. Lett.* **80** (1998), 3747-3750.

[17] Udwadia F. and von Bremen H., An efficient and stable approach for computation of Lyapunov characteristic exponents of continuous dynamical systems. *Applied Math. and Computation* **121** (2001), 219–259.

[18] Gantmacher F. R., *Theory of Matrices*, Vol. 1, Chelsea , New York, 1959.

Firdaus E. Udwadia
Civil Engineering, Aerospace and Mechanical
Engineering
and Information and Operations Management
430K Olin Hall of Engineering
University of Southern California
Los Angeles, CA 90089-1453
USA
e-mail: fudwadia@usc.edu

Hubertus F. von Bremen
Department of Mechanical Engineering
University of Southern California
Los Angeles, CA 90089
USA

To access this journal online:
http://www.birkhauser.ch