Electronic Transactions on Numerical Analysis.
Volume 12, pp. 234-251, 2001.

ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

ERROR ANALYSIS OF QR ALGORITHMS FOR COMPUTING LYAPUNOV EXPONENTS*

EDWARD J. MCDONALD† AND DESMOND J. HIGHAM‡

Abstract. Lyapunov exponents give valuable information about long term dynamics. The discrete and continuous QR algorithms are widely used numerical techniques for computing approximate Lyapunov exponents, although they are not yet supported by a general error analysis. Here, a rigorous convergence theory is developed for both the discrete and continuous QR algorithm applied to a constant coefficient linear system with real distinct eigenvalues. For the discrete QR algorithm, the problem essentially reduces to one of linear algebra for which the timestepping and linear algebra errors uncouple and precise convergence rates are obtained. For the continuous QR algorithm, the stability, rather than the local accuracy, of the timestepping algorithm is relevant, and hence the overall convergence rate is independent of the stepsize. In this case it is vital to use a timestepping method that preserves orthogonality in the ODE system. We give numerical results to illustrate the analysis. Further numerical experiments and a heuristic argument suggest that the convergence properties carry through to the case of complex conjugate eigenvalue pairs.

Key words. dynamics, eigenvalues, orthogonal iteration, timestepping.

AMS subject classifications. 65L05, 65F15.

1. Introduction.

Several authors have derived numerical algorithms for the computation of Lyapunov exponents of ordinary differential equations (ODEs); see [4, 5] for an overview. However, there is little error analysis to justify the use of these algorithms. In this work, we consider the discrete and continuous QR algorithms, and in order to establish a rigorous convergence result we restrict attention to a simple class of test problems—linear, constant coefficient systems. In this case the Lyapunov exponents reduce to the real parts of the eigenvalues of the Jacobian matrix and the discrete and continuous QR algorithms become closely related to the orthogonal iteration process in numerical linear algebra. In order to develop a convergence theory for the discrete QR algorithm, it is necessary to deal simultaneously with the limits $\Delta t \to 0$ and $T \to \infty$, where Δt is the timestep used for the ODE solver and $[0, T]$ is the truncation of $[0, \infty)$. We resolve this by allowing T to behave like a negative power of Δt. For the continuous QR algorithm, the use of a timestepping method that preserves orthogonality is vital for convergence.

In §2 we define the QR algorithms, state our assumptions and give the main convergence theorems. These theorems are proved in §3. Section 4 presents some numerical results that back up the theory. Although the convergence results apply only for the case of real distinct eigenvalues, we also perform numerical tests involving complex conjugate pairs. It appears that the algorithms remain convergent in this case, despite the fact that only diagonal entries are used (rather than 2×2 blocks). This effect is illustrated in detail. Section 5 discusses the results.

With the notable exception of [4], there has been relatively little attention paid to Lyapunov exponent algorithms in the numerical analysis literature, and hence a general convergence theory is lacking. In [4], a number of fundamental results are given that quantify the error under various simplifying assumptions. In particular, a semi-heuristic discussion of the convergence of the QR algorithms on constant coefficient ODEs is given; [4, pages 412–413]. However, the discussion does not mention how to resolve the $\Delta t \to 0$, $T \to \infty$ issue and

*Received October 25, 2000. Accepted for publication October 25, 2001. Recommended by W.B. Gragg.

†Department of Mathematics, University of Strathclyde, Glasgow, G1 1XH, Scotland. Email: ta.emcd@maths.strath.ac.uk.

‡Department of Mathematics, University of Strathclyde, Glasgow, G1 1XH, Scotland. Email: djh@maths.strath.ac.uk. This research was supported by the Engineering and Physical Sciences Research Council of the UK under grant GR/M42206.

ETNA
Kent State University
etna@mcs.kent.edu

rates of convergence are not given. Our work can be regarded as an attempt to make rigorous that discussion in [4].

Overall, we aim to provide a *rigorous* analysis of the *rate* of convergence of the QR algorithms on a tractable class of test problems. The analysis makes use of convergence theory from numerical linear algebra, but also relies on results from the classical numerical ODE literature and more recent ideas from geometric integration.

2. Motivation and Convergence Results. We begin by describing the algorithms on time-dependent linear systems. The algorithms may also be applied to nonlinear systems after linearizing along a solution trajectory [4].

For the n-dimensional linear system

$$\dot{y}(t) = A(t)y(t), \tag{2.1}$$

we let $Y(t) \in \mathbb{R}^{n\times n}$ denote the fundamental solution matrix for (2.1), so that $\dot{Y}(t) = A(t)Y(t)$ and $Y(0) = I$. A continuous QR factorization of $Y(t)$ then gives

$$Y(t) = Q(t)R(t), \tag{2.2}$$

where $Q(t) \in \mathbb{R}^{n\times n}$ is orthogonal and $R(t) \in \mathbb{R}^{n\times n}$ is upper triangular with positive diagonal entries. (Throughout this work, we will ask for positive diagonal entries in a triangular QR factor—this makes the QR factorization of a nonsingular matrix unique [6].) Under appropriate regularity assumptions, it may be shown [4] that the Lyapunov exponents for the system (2.1) satisfy

$$\lambda^{[k]} = \lim_{t\to\infty} \frac{1}{t} \log R_{kk}(t), \quad 1 \le k \le n. \tag{2.3}$$

2.1. Discrete QR Algorithm. The discrete QR algorithm for (2.1) is based on the following process. Choose a sequence $0 = t_0 < t_1 < t_2 < \cdots$, with $\lim_{j\to\infty} t_j = \infty$. Set $Q_0 = I$ and for $j = 0, 1, 2, \ldots$ let

$$\dot{Z}_j(t) = A(t)Z_j(t), \quad Z_j(t_j) = Q_j, \quad t_j \le t \le t_{j+1}, \tag{2.4}$$

and take the QR factorization

$$Z_j(t_{j+1}) = Q_{j+1}R_{j+1}. \tag{2.5}$$

To see why (2.4) and (2.5) are useful, let $F_j(t) \in \mathbb{R}^{n\times n}$ be such that $\dot{F}_j(t) = A(t)F_j(t)$ and $F_j(t_j) = I$. Then

$$Z_j(t_{j+1}) = F_j(t_{j+1})Q_j \quad \text{and} \quad Y(t_{j+1}) = F_j(t_{j+1})Y(t_j). \tag{2.6}$$

It follows from (2.5) and (2.6) that

$$Y(t_{j+1}) = Z_j(t_{j+1}){Q_j}^T Y(t_j) = Q_{j+1}R_{j+1}{Q_j}^T Y(t_j).$$

Continuing this argument we find that

$$\begin{aligned} Y(t_{j+1}) &= Q_{j+1}R_{j+1}{Q_j}^T Q_j R_j {Q_{j-1}}^T Y(t_{j-2}) \\ &\ \ \vdots \\ &= Q_{j+1}R_{j+1}R_j \cdots R_1. \end{aligned}$$

ETNA
Kent State University
etna@mcs.kent.edu

Hence, (2.4) and (2.5) contain the information needed to construct the QR factorization of $Y(t)$ at each point t_j, and so, from (2.3),

$$\lambda^{[k]} = \lim_{N\to\infty} \frac{1}{t_N} \log \left(\prod_{j=1}^{N} (R_j)_{kk} \right), \quad 1 \le k \le n. \tag{2.7}$$

To convert (2.4) and (2.5) into a numerical algorithm, two types of approximation are introduced.

1. The ODE system (2.4) is solved numerically. We will suppose that a constant spacing $\Delta t := t_{j+1} - t_j$ is used and a one–step numerical method with stepsize Δt is applied for each iteration.
2. The infinite time interval is truncated, so that a finite number of iteration steps is used.

2.2. Continuous QR Algorithm. The continuous QR algorithm proceeds as follows. Differentiating (2.2) we have

$$\dot{Y} = \dot{Q}R + Q\dot{R} = AQR, \tag{2.8}$$

from which we obtain

$$Q^T\dot{Q} - Q^TAQ = -\dot{R}R^{-1}. \tag{2.9}$$

Note that $\dot{R}R^{-1}$ is upper triangular and, since $Q^TQ = I$, $Q^T\dot{Q}$ is skew–symmetric. It thus follows from (2.9) that $Q^T\dot{Q} = H(t, Q)$, where

$$H_{ij} = \begin{cases} (Q^TAQ)_{ij}, & i > j, \\ 0, & i = j, \\ -(Q^TAQ)_{ji}, & i < j. \end{cases} \tag{2.10}$$

Thus, the matrix system

$$\dot{Q}(t) = Q(t)H(t, Q(t)), \tag{2.11}$$

can be solved to obtain $Q(t)$. From (2.9)

$$\dot{R} = (Q^TAQ - Q^T\dot{Q})R,$$

so using (2.10) and the skew-symmetry of $Q^T\dot{Q}$ this gives

$$\dot{R}_{ii} = (Q^TAQ)_{ii}R_{ii}, \quad i = 1, \ldots, n. \tag{2.12}$$

Therefore,

$$\lambda^{[k]} = \lim_{t\to\infty} \frac{1}{t} \log R_{kk}(t) = \lim_{t\to\infty} \frac{1}{t} \int_0^t (Q^T(s)A(s)Q(s))_{kk}\, ds. \tag{2.13}$$

In order to implement the continuous QR algorithm numerically two types of approximation are required.

1. The nonlinear ODE system (2.11) is solved numerically. We assume that a constant stepsize $\Delta t := t_j - t_{j-1}$ is used.
2. The integral in (2.13) is approximated numerically over a finite range $[0, T]$. Following [4] we use the composite trapezoidal rule.

ETNA
Kent State University
etna@mcs.kent.edu

Note also that solutions of the ODE system (2.11) preserve orthogonality—if $Q(0)^T Q(0) = I$, then $Q(t)^T Q(t) = I$, for all $t > 0$. It is natural to ask for this property to be maintained by the numerical method. (Indeed, we will show that this is vital for convergence.) We consider two classes of numerical method that preserve orthogonality.

(i) **Projected Runge–Kutta (PRK) methods**. Here, a Runge–Kutta method is applied over each timestep, and the (generally non-orthogonal) solution is perturbed to an orthogonal one. This can be done by replacing the matrix by its orthogonal polar factor, which corresponds to a projection in the Frobenius norm. Alternatively, the matrix can be replaced by its orthogonal QR factor, a process that is closely related to the Frobenius norm projection [7].

(ii) **Gauss–Legendre–Runge–Kutta (GLRK) methods**. These are one–step methods that automatically preserve orthogonality of the numerical solution.

Both types of integrator have been examined in [3]. We note that orthogonal integration can be viewed within the much more general framework of Lie group methods; see [8].

2.3. Convergence Results. In order to prove sharp convergence results for the algorithms, we restrict attention to the case where $A(t)$ is constant, $A(t) \equiv A$, where A has real, distinct eigenvalues $\{\lambda^{[k]}\}_{k=1}^n$, ordered so that

$$\exp(\lambda^{[1]}) > \exp(\lambda^{[2]}) > \cdots > \exp(\lambda^{[n]}). \tag{2.14}$$

In this case $\{\lambda^{[k]}\}_{k=1}^n$ are also the Lyapunov exponents of the ODE. We also assume that for each $1 \le k \le n-1$ no vector in the space spanned by the first k eigenvectors of A is orthogonal to the space spanned by the first k columns of the identity matrix. This is an extremely mild assumption that generalizes the traditional assumption made about the starting vector in the power method; see, for example, [2, page 158].

2.3.1. Convergence of Discrete QR Algorithm. We let Z_j denote the approximation to $Z_j(t_j)$ produced by the one-step numerical method on (2.4), and we suppose that

$$Z_{j+1} = S(\Delta t A) Z_j,$$

where $S(z)$ is a rational function such that

$$S(z) = \exp(z)\left(1 + \mathcal{O}\left(\Delta t^{p+1}\right)\right), \quad \text{for some integer } p \ge 1. \tag{2.15}$$

This covers the case where the numerical method is a (explicit or implicit) Runge–Kutta formula of order p. The discrete QR algorithm for computing an approximation $l^{[k]}$ to $\lambda^{[k]}$ then has the following form, with $Q_0 = I$.

Discrete QR algorithm

for $j = 0, 1, \ldots, N-1$
    $S(\Delta t A) Q_j =: Q_{j+1} R_{j+1}$ (QR factorization)
end

$$\ell^{[k]} := \frac{1}{T} \log \prod_{j=1}^{N} (R_j)_{kk}, \quad \text{where } T := N\Delta t. \tag{2.16}$$

In analysing the error $|\lambda^{[k]} - l^{[k]}|$ there are two limits to be considered. We must allow $\Delta t \to 0$ in order to reduce the error of the ODE solver, but we must also allow $T \to \infty$ to reduce the error from truncating the time interval. Hence, in contrast to standard finite-time

ETNA
Kent State University
etna@mcs.kent.edu

convergence theory [9], we require $N \to \infty$ faster than $\Delta t \to 0$. We can accomplish this by setting

$$T = K\Delta t^{-\alpha}, \tag{2.17}$$

where $K, \alpha > 0$ are constants. (So the number of timesteps $N = K\Delta t^{-(\alpha+1)}$.) In this framework we consider the single limit $\Delta t \to 0$. In practice this corresponds to repeating the discrete QR algorithm with a smaller Δt *and* a larger time interval T. The result that we prove is stated below.

THEOREM 2.1. *With the notation and assumptions above, there exists a constant C such that, for all sufficiently small Δt,*

$$|\ell^{[k]} - \lambda^{[k]}| \le C\left(\Delta t^{\alpha} + \Delta t^{p}\right), \quad 1 \le k \le n. \tag{2.18}$$

Proof. See §3.2. □

Our proof of Theorem 2.1 relies on the underlying convergence theory for orthogonal iteration [2, 11]—this is also equivalent to the analysis for the QR algorithm [1, 15]. However, the application of that theory is not entirely straightforward since we must study (a variant of) orthogonal iteration on a matrix that is parametrized by Δt. In particular, the naturally arising linear contraction factor $r^{[k]}_{\Delta t}$, which is defined in §3.2, has the property that $r^{[k]}_{\Delta t} \to 1$ as $\Delta t \to 0$. This, however, is balanced by the fact that the number N of iterations increases rapidly as $\Delta t \to 0$, and, as shown in (3.13) below, $(r^{[k]}_{\Delta t})^N \to 0$. (This also emphasizes that both limits $\Delta t \to 0$ and $T \to \infty$ must be addressed in a convergence theory.)

2.3.2. Convergence of Continuous QR Algorithm. The continuous QR algorithm for computing an approximation $\ell^{[k]}$ to $\lambda^{[k]}$ can be summarized as follows, with $Q_0 = I$.

Continuous QR algorithm

Solve (2.11) numerically to obtain $\{Q_j \approx Q(t_j)\}_{j=0}^{N}$.

$$\ell^{[k]} = \frac{1}{T}\frac{\Delta t}{2}\sum_{j=1}^{N}\left[(Q_{j-1}^T A Q_{j-1})_{kk} + (Q_j^T A Q_j)_{kk}\right], \quad \text{where } T = N\Delta t. \tag{2.19}$$

The following convergence theorem holds.

THEOREM 2.2. *Suppose the* ODE *(2.11) is solved using a PRK or GLRK method of classical order $p \ge 1$. Then with the notation and assumptions above there exists a constant C such that, for sufficiently small Δt,*

$$|\ell^{[k]} - \lambda^{[k]}| \le \frac{C}{T}, \quad 1 \le k \le n. \tag{2.20}$$

Proof. See §3.3. □

Note that Δt does not appear in the error bound (2.20). This emphasizes that the structural properties of the ODE method (orthogonality preservation and stability) are relevant, but not the precise classical order of convergence.

3. Convergence Proofs.

3.1. Schur Matrix and Orthogonal Iteration. We begin by reviewing some relevant concepts from numerical linear algebra. See [2, 6, 10] for more details.

DEFINITION 3.1. *Given* $B \in \mathbb{R}^{n\times n}$ *with distinct, real eigenvalues* $\{\mu^{[k]}\}_{k=1}^{n}$ *ordered so that* $|\mu^{[1]}| > |\mu^{[2]}| > \cdots > |\mu^{[n]}|$, *there exists an orthogonal matrix* $Q_\star$, *referred to as a* Schur matrix, *such that*

$$Q_\star^T B Q_\star = \Upsilon,$$

where Υ *is upper triangular with main diagonal given by* $\mu^{[1]}, \dots, \mu^{[n]}$. *The Schur matrix is unique up to a factor of* ± 1 *multiplying each column. The columns of* $Q_\star$ *(denoted by* $q_\star^{[k]}$*) are called* Schur vectors, *and it follows that the subspace spanned by* $\{q_\star^{[1]}, q_\star^{[2]}, \dots, q_\star^{[k]}\}$ *is identical to the subspace spanned by the eigenvectors of* B *that correspond to the eigenvalues* $\mu^{[1]}, \mu^{[2]}, \dots, \mu^{[k]}$.

Orthogonal iteration may be regarded as a technique for computing an approximate Schur decomposition. Given $B \in \mathbb{R}^{n\times n}$, orthogonal iteration proceeds as follows, with $Q_0 = I$.

Orthogonal Iteration

for $j = 0, 1, \dots$
  $BQ_j =: Q_{j+1}R_{j+1}$ (QR factorization)
end

Under the mild assumption that for each $1 \le k \le n-1$ no vector contained in $\mathrm{span}\{q_\star^{[1]}, q_\star^{[2]}, \dots, q_\star^{[k]}\}$ is orthogonal to the space spanned by the first k columns of the identity matrix, this iteration converges linearly, in the manner outlined in Lemma 3.2 below. We let $q_j^{[k]}$ denote the kth column of Q_j, and let $\mu_j^{[k]}$ denote $\left(Q_j^T B Q_j\right)_{kk}$. To be definite, we regard $\|\cdot\|$ as the Euclidean norm. We also write $\|v \pm w\|$ to mean $\min\{\|v+w\|, \|v-w\|\}$.

LEMMA 3.2. *With the assumptions and notation above, there exist constants* C *and* D *such that*

$$\|q_j^{[k]} \pm q_\star^{[k]}\| \le C(r^{[k]})^j \quad \text{and} \quad |\mu_j^{[k]} - \mu^{[k]}| \le D(r^{[k]})^j, \quad 1 \le k \le n, \tag{3.1}$$

where

$$\begin{aligned} r^{[1]} &= |\mu^{[2]}/\mu^{[1]}|, \\ r^{[k]} &= \max(|\mu^{[k+1]}/\mu^{[k]}|, |\mu^{[k]}/\mu^{[k-1]}|), \quad 1 < k < n, \\ r^{[n]} &= |\mu^{[n]}/\mu^{[n-1]}|. \end{aligned}$$

Proof. This result is stated without proof in [11]. Convergence analysis for orthogonal iteration is usually performed in terms of subspaces: generally, the subspace spanned by the first k columns of Q_j converges to the subspace spanned by the first k columns of $Q_\star$ at a linear rate determined by $|\mu^{[k+1]}/\mu^{[k]}|$ [2, 6, 16]. By considering subspaces of dimensions k and $k-1$, the result (3.1) follows. □

3.2. Discrete QR Convergence Analysis.

3.2.1. Orthogonal Iteration Error. In this subsection and the next, we use κ_i to denote generic constants.

Comparing the two algorithms, we see that the matrices Q_j in the discrete QR algorithm are precisely the matrices Q_j that arise when orthogonal iteration is applied to $B = S(\Delta t A)$. Hence, we may appeal to the convergence theory in Lemma 3.2. However, it is vital to exploit the fact that Δt is a small parameter and $S(z)$ approximates $\exp(z)$. Using a second subscript to emphasize Δt-dependence, we let $S(\Delta t A) = Q_\star \Upsilon_{\Delta t} Q_\star^T$ denote a Schur decomposition of $S(\Delta tA)$ with $(\Upsilon_{\Delta t})_{kk} = \mu_{\Delta t}^{[k]}$. Similarly, we let $Q_{j,\Delta t}$ have kth column $q_{j,\Delta t}^{[k]}$ and write $\mu_{j,\Delta t}^{[k]}$ for ${q_{j,\Delta t}^{[k]}}^T S(\Delta tA) q_{j,\Delta t}^{[k]}$.

First we note that with the ordering (2.14) on the eigenvalues of A, for sufficiently small Δt we have, from (2.15),

$$\mu_{\Delta t}^{[1]} > \mu_{\Delta t}^{[2]} > \cdots > \mu_{\Delta t}^{[n]} > 0. \tag{3.2}$$

Following the proof of Lemma 3.2 for this parameterized matrix, we find that the Schur vector convergence bound holds with a constant independent of Δt; that is,

$$\|q_{j,\Delta t}^{[k]} \pm q_\star^{[k]}\| \le \kappa_1 (r_{\Delta t}^{[k]})^j, \tag{3.3}$$

where $r_{\Delta t}^{[k]}$ is defined as in Lemma 3.2 with each $\mu^{[k]}$ replaced by $\mu_{\Delta t}^{[k]}$. We then have

$$\begin{aligned}
|\mu_{j,\Delta t}^{[k]} - \mu_{\Delta t}^{[k]}| &= |{q_{j,\Delta t}^{[k]}}^T S(\Delta tA) q_{j,\Delta t}^{[k]} - {q_\star^{[k]}}^T S(\Delta tA) q_\star^{[k]}| \\
&= |{q_{j,\Delta t}^{[k]}}^T \left(I + (S(\Delta tA) - I)\right) q_{j,\Delta t}^{[k]} - {q_\star^{[k]}}^T \left(I + (S(\Delta tA) - I\right) q_\star^{[k]}| \\
&= |{q_{j,\Delta t}^{[k]}}^T q_{j,\Delta t}^{[k]} - {q_\star^{[k]}}^T q_\star^{[k]} + (q_{j,\Delta t}^{[k]} \pm q_\star^{[k]})^T \left(S(\Delta tA) - I\right) q_{j,\Delta t}^{[k]} \\
&\quad - {q_\star^{[k]}}^T \left(S(\Delta tA) - I\right) (q_\star^{[k]} \pm q_{j,\Delta t}^{[k]})| \\
&\le 2\|q_{j,\Delta t}^{[k]} \pm q_\star^{[k]}\| \|S(\Delta tA) - I\| \\
&\le \kappa_2 \Delta t \|A\| \|q_{j,\Delta t}^{[k]} \pm q_\star^{[k]}\| \\
&\le \kappa_3 \Delta t (r_{\Delta t}^{[k]})^j,
\end{aligned} \tag{3.4}$$

for sufficiently small Δt, where we have used the property (2.15) and the bound (3.3). The inequality (3.4) shows that when orthogonal iteration is applied to a matrix of the form $S(\Delta tA) = I + \Delta tA + \mathcal{O}\left(\Delta t^2\right)$ then the "constant" in the eigenvalue convergence bound is $\mathcal{O}\left(\Delta t\right)$.

We note from (2.16) that the discrete QR algorithm does not use $Q_{j,\Delta t}^T S(\Delta tA) Q_{j,\Delta t}$, but rather the shifted version $R_j := Q_{j,\Delta t}^T S(\Delta tA) Q_{j-1,\Delta t}$. However, it is readily shown that $\|q_{j,\Delta t}^{[k]} - q_{j-1,\Delta t}^{[k]}\| \le \kappa_4 \Delta t (r_{\Delta t}^{[k]})^j$, and hence the bound (3.4) also implies

$$|{q_{j,\Delta t}^{[k]}}^T S(\Delta tA) q_{j-1,\Delta t}^{[k]} - {q_\star^{[k]}}^T S(\Delta tA) q_\star^{[k]}| \le \kappa_5 \Delta t (r_{\Delta t}^{[k]})^j. \tag{3.5}$$

In summary, (3.5) shows that the computed diagonal entries $(R_{j,\Delta t})_{kk}$ in (2.16) approximate the corresponding eigenvalues $\mu_{\Delta t}^{[k]}$ of $S(A\Delta t)$ according to

$$(R_{j,\Delta t})_{kk} = \mu_{\Delta t}^{[k]} (1 + \gamma_{j,\Delta t}^{[k]}), \quad \text{where } |\gamma_{j,\Delta t}^{[k]}| \le \kappa_6 \Delta t (r_{\Delta t}^{[k]})^j. \tag{3.6}$$

3.2.2. ODE Error. We now incorporate the ODE solving error in order to obtain the overall error bound.

Since A is diagonalizable, it is straightforward to show from (2.15) that the eigenvalue $\mu_{\Delta t}^{[k]}$ of $S(\Delta tA)$ is related to the eigenvalue $\exp(\Delta t \lambda^{[k]})$ of $\exp(\Delta tA)$ by

$$\mu_{\Delta t}^{[k]} = \exp(\Delta t \lambda^{[k]})(1 + \delta_{\Delta t}^{[k]}), \quad \text{where } |\delta_{\Delta t}^{[k]}| \le \kappa_7 \Delta t^{p+1}. \tag{3.7}$$

ETNA
Kent State University
etna@mcs.kent.edu

Now, using (3.6) and (3.7), the computed Lyapunov exponent $\ell^{[k]}$ in (2.16) satisfies

$$\ell^{[k]} = \frac{1}{T}\log\prod_{j=1}^{N}\left(\exp(\Delta t\lambda^{[k]})(1+\delta_{\Delta t}^{[k]})(1+\gamma_{j,\Delta t}^{[k]})\right),$$

$$= \lambda^{[k]} + \frac{1}{T}\left(\sum_{j=1}^{N}\log(1+\delta_{\Delta t}^{[k]}) + \sum_{j=1}^{N}\log(1+\gamma_{j,\Delta t}^{[k]})\right). \tag{3.8}$$

We note from (3.6) and (3.7) that both $|\gamma_{j,\Delta t}^{[k]}|$ and $|\delta_{\Delta t}^{[k]}|$ can be made arbitrarily small by reducing Δt. Hence, for sufficiently small Δt,

$$0 < |\log(1+\delta_{\Delta t}^{[k]})| \le 2|\delta_{\Delta t}^{[k]}| \quad \text{and} \quad 0 < |\log(1+\gamma_{j,\Delta t}^{[k]})| \le 2|\gamma_{j,\Delta t}^{[k]}|.$$

In (3.8), using (3.6) and (3.7) and recalling that $T = N\Delta t$, this gives

$$|\ell^{[k]} - \lambda^{[k]}| \le \kappa_8\left(\frac{1}{T}\sum_{j=1}^{N}\gamma_{j,\Delta t}^{[k]} + \frac{\delta_{\Delta t}^{[k]}}{\Delta t}\right) \le \kappa_9\left(\frac{\Delta t}{T}\sum_{j=1}^{N}(r_{\Delta t}^{[k]})^j + \Delta t^p\right).$$

Summing the geometric series gives

$$|\ell^{[k]} - \lambda^{[k]}| \le \kappa_9\left(\frac{\Delta t}{T}\frac{r_{\Delta t}^{[k]}(1-(r_{\Delta t}^{[k]})^N)}{1-r_{\Delta t}^{[k]}} + \Delta t^p\right). \tag{3.9}$$

Now, it follows from (2.15) that

$$\frac{\mu_{\Delta t}^{[k+1]}}{\mu_{\Delta t}^{[k]}} = \exp\left(\Delta t(\lambda^{[k+1]} - \lambda^{[k]})\right)\left(1+\mathcal{O}\left(\Delta t^{p+1}\right)\right),$$

and hence,

$$0 < r_{\Delta t}^{[k]} \le \exp\left(-\Delta t\epsilon^{[k]}\right)\left(1+\mathcal{O}\left(\Delta t^{p+1}\right)\right), \tag{3.10}$$

where

$$\epsilon^{[1]} := \lambda^{[1]} - \lambda^{[2]} > 0, \quad \epsilon^{[n]} := \lambda^{[n-1]} - \lambda^{[n]} > 0 \tag{3.11}$$

and

$$\epsilon^{[k]} := \min\{\lambda^{[k]} - \lambda^{[k+1]}, \lambda^{[k-1]} - \lambda^{[k]}\} > 0, \quad \text{for } 1 < k < n. \tag{3.12}$$

So, for small Δt,

$$0 < r_{\Delta t}^{[k]} < \exp(-\Delta t\epsilon^{[k]}/2)$$

and, using (2.17),

$$0 < \left(r_{\Delta t}^{[k]}\right)^N < \exp(-N\Delta t\epsilon^{[k]}/2) = \exp(-K\Delta t^{-\alpha}\epsilon^{[k]}/2) \to 0 \text{ as } \Delta t \to 0. \tag{3.13}$$

It also follows from (3.10) that

$$0 < r_{\Delta t}^{[k]} \le 1 + \kappa_{10}\Delta t \quad \text{and} \quad 1 - r_{\Delta t}^{[k]} \ge \kappa_{11}\Delta t. \tag{3.14}$$

Using (3.13) and (3.14) in (3.9) leads to the bound

$$|\ell^{[k]} - \lambda^{[k]}| \le \kappa_{12}\left(\frac{1}{N\Delta t} + \Delta t^p\right) \le \kappa_{13}\left(\Delta t^{\alpha} + \Delta t^p\right), \tag{3.15}$$

which establishes Theorem 2.1.

3.3. Continuous QR Convergence Analysis.

3.3.1. Convergence of $q_j^{[k]}$ to $q_\star^{[k]}$. It follows from the theory of QR flows that any solution of the system (2.11) approaches a fixed point as $t \to \infty$; see, for example, [16]. This fixed point must be a Schur matrix $Q_\star$ of A. Our analysis below is aimed at showing that the ODE solver applied to (2.11) also asymptotes to $Q_\star$. This is not a trivial task because, regarding (2.11) as an ODE in $\mathbb{R}^{n\times n}$, if the problem is linearized about $Q(t) = Q_\star$ then no conclusion can be drawn about stability—eigenvalues of zero real part arise. Hence, a straightforward linearization argument cannot be applied. We also note that although the only *orthogonal* fixed points of (2.11) correspond to Schur matrices of A, there are many other non-orthogonal fixed points. For example, $\sigma Q_\star$ for any $\sigma \in \mathbb{R}$ is also a fixed point. It follows that a numerical method that does not preserve orthogonality may drift towards a non-orthogonal steady-state. We have observed this behaviour in practice, and its consequences are illustrated in §4.

The following lemma forms the main part of our convergence proof.

LEMMA 3.3. *If a PRK or GLRK method is used to solve the* ODE*(2.11), then for sufficiently small Δt the kth column of the numerical solution, $q_j^{[k]}$, converges to a Schur vector $q_\star^{[k]}$ linearly:*

$$\|q_\star^{[k]} - q_j^{[k]}\| \le C(\widehat{r}_{\max} + D\Delta t^p)^{j\Delta t}, \tag{3.16}$$

where C and D are constants, with

$$\widehat{r}_{\max} = \max_{1\le i\le n} \widehat{r}^{[i]}, \quad \widehat{r}^{[k]} = \exp(-\epsilon^{[k]})$$

and $\epsilon^{[k]}$ is defined in (3.11) and (3.12).

Proof. First we let $\Psi(Q) := QH(Q)$, where $H(Q)$ is defined in (2.11). Now, note that $\Psi(Q)$ is locally Lipschitz, so given any bounded region $\mathcal{B}$ there exists a constant $L = L(\mathcal{B})$ such that

$$\|\Psi(W) - \Psi(Q_\star)\| \le L\|W - Q_\star\|, \quad \forall\, W \in \mathcal{B}, \text{ with } W^TW = I.$$

Since $\Psi(Q_\star) = 0$, we have

$$\|\Psi(W)\| \le L\|W - Q_\star\|, \quad \forall\, W \in \mathcal{B}, \text{ with } W^TW = I. \tag{3.17}$$

Also, we note that any Runge–Kutta method applied to $\dot{Q}(t) = \Psi(Q(t))$ has a factor of $\|\Psi(Q_j)\|$ in its local error expression—this follows from classical order theory [9]. Hence, if we let $Q_j(t)$ denote the local solution of (2.11) over $[t_i, t_{i+1}]$, so that $\dot{Q}_j(t) = \Psi(Q_j(t))$ and $Q_j(t_j) = Q_j$, then

$$\|Q_{j+1} - Q_j(t_{j+1})\| \le \kappa_2 \Delta t^{p+1}\|\Psi(Q_j)\|, \tag{3.18}$$

for any GLRK method. In the case of PRK methods, projection can no more than double the local error and "approximately projecting" onto the orthogonal QR factor increases the local error by at most a factor $1 + 2\sqrt{2}$ asymptotically [7]. So (3.18) is valid for both GLRK and PRK methods.

Now we know from [16] that for the exact flow of (2.11), the kth column of $Q(t)$, which we denote $q^{[k]}(t)$, converges linearly to a Schur vector $q_\star^{[k]}$ at rate $r^{[k]}$. So we may choose a time $\widehat{T}$ such that $Q(\widehat{T}) \in \mathcal{B}$, where $\mathcal{B}$ is a ball containing $Q_\star$ with the property that if $\hat{Q}(0) \in \mathcal{B}$ and $\hat{Q}(t)$ solves (2.11) then

$$\|\hat{q}^{[k]}(t + \Delta t) - q_\star^{[k]}\| \le (\widehat{r}^{[k]})^{\Delta t}\|\hat{q}^{[k]}(t) - q_\star^{[k]}\|, \quad \forall\, t \ge \widehat{T}. \tag{3.19}$$

Now, since the numerical method is convergent over finite time intervals, the inequality (3.16) will hold for $j\Delta t \le \widehat{T}$ and Δt sufficiently small. For later times, we have, from (3.17) and (3.18),

$$\|q_{j+1}^{[k]} - q_j^{[k]}(t_{j+1})\| \le \kappa_3 \Delta t^{p+1} \|Q_j - Q_\star\|. \tag{3.20}$$

For ease of notation, let

$$e_j^{[k]} := \|q_j^{[k]} - q_\star^{[k]}\|. \tag{3.21}$$

Then using (3.19) and (3.20), we obtain

$$\begin{aligned} e_{j+1}^{[k]} &\le \|q_{j+1}^{[k]} - q_j^{[k]}(t_{j+1})\| + \|q_j^{[k]}(t_{j+1}) - q_\star^{[k]}\| \\ &\le (\kappa_3 \Delta t^{p+1} + (\widehat{r}_{\max})^{\Delta t}) \max_{1\le i\le n} e_j^{[i]}. \end{aligned} \tag{3.22}$$

For Δt sufficiently small, it can be shown that

$$\kappa_3 \Delta t^{p+1} + (\widehat{r}_{\max})^{\Delta t} \le (\widehat{r}_{\max} + \kappa_4 \Delta t^p)^{\Delta t},$$

where $\kappa_4 = 8\kappa_3 \widehat{r}_{\max}$. Hence (3.22) gives

$$\max_{1\le i\le n} e_{j+1}^{[i]} \le (\widehat{r}_{\max} + \kappa_4 \Delta t^p)^{\Delta t} \max_{1\le i\le n} e_j^{[i]}.$$

It follows that

$$\max_{1\le i\le n} e_{j+1}^{[i]} \le \kappa_5 (\widehat{r}_{\max} + \kappa_4 \Delta t^p)^{\Delta t (j+1)}. \square \tag{3.23}$$

3.3.2. Trapezoidal Rule Error. The error in the Lyapunov exponent approximation $\ell^{[k]}$ in (2.19) satisfies

$$\begin{aligned} |\lambda^{[k]} - \ell^{[k]}| &= \left| (q_\star^{[k]T} A q_\star^{[k]}) - \frac{1}{T}\frac{\Delta t}{2} \sum_{j=1}^{N} \left[(q_{j-1}^{[k]}{}^T A q_{j-1}^{[k]}) + (q_j^{[k]T} A q_j^{[k]}) \right] \right| \\ &= \frac{1}{2N} \left| \sum_{j=1}^{N} \left[\left(q_\star^{[k]T} A q_\star^{[k]} - q_{j-1}^{[k]}{}^T A q_{j-1}^{[k]} \right) + \left(q_\star^{[k]T} A q_\star^{[k]} - q_j^{[k]T} A q_j^{[k]} \right) \right] \right| \\ &= \frac{1}{2N} \left| \sum_{j=1}^{N} \left[(q_\star^{[k]T} - q_{j-1}^{[k]}{}^T) A q_\star^{[k]} + q_{j-1}^{[k]}{}^T A (q_\star^{[k]} - q_{j-1}^{[k]}) \right] \right. \\ &\qquad \left. + \sum_{j=1}^{N} \left[(q_\star^{[k]T} - q_j^{[k]T}) A q_\star^{[k]} + q_j^{[k]T} A (q_\star^{[k]} - q_j^{[k]}) \right] \right| \\ &\le \frac{\|A\|}{N} \sum_{j=1}^{N} \left(e_{j-1}^{[k]} + e_j^{[k]} \right), \end{aligned}$$

where we have used $\|q_j^{[k]}\| = 1$, since the numerical scheme preserves orthogonality. Making use of Lemma 3.3, we obtain

$$|\lambda^{[k]} - \ell^{[k]}| \le \frac{\|A\|}{N} \sum_{j=1}^{N} \left[\kappa_5 (\widehat{r}_{\max} + \kappa_6 \Delta t^p)^{(j-1)\Delta t} + \kappa_5 (\widehat{r}_{\max} + \kappa_6 \Delta t^p)^{j\Delta t} \right]$$

ETNA
Kent State University
etna@mcs.kent.edu

$$\leq \frac{2\kappa_5 \|A\|}{N} \sum_{j=1}^{N} (\widehat{r}_{\max} + \kappa_6 \Delta t^p)^{\Delta t (j-1)}$$
$$= \frac{2\kappa_5 \|A\|}{N} \left[\frac{1 - (\widehat{r}_{\max} + \kappa_6 \Delta t^p)^T}{1 - (\widehat{r}_{\max} + \kappa_6 \Delta t^p)^{\Delta t}} \right].$$

For a sufficiently small Δt, we have

$$1 - (\widehat{r}_{\max} + \kappa_6 \Delta t^p)^T \leq 1 \quad \text{and} \quad 1 - (\widehat{r}_{\max} + \kappa_6 \Delta t^p)^{\Delta t} \geq \kappa_6 \Delta t.$$

Therefore,

$$|\lambda^{[k]} - \ell^{[k]}| \leq \frac{2\kappa_7 \|A\|}{N \Delta t} = \frac{\kappa_8}{T},$$

which establishes Theorem 2.2.

4. Numerical Tests.

4.1. Real Distinct Eigenvalues.

4.1.1. Discrete QR Algorithm. In this subsection we illustrate Theorem 2.1, testing the three cases $p > \alpha$, $p = \alpha$ and $p < \alpha$ for a 4×4 system. Given the Lyapunov exponents $\{\lambda^{[k]}\}_{k=1}^{4}$, we produce the Jacobian matrix A by forming $A = X \operatorname{diag}(\lambda^{[k]}) X^{-1}$, where X is a random matrix. (More precisely, X is formed using `rand('state',0)` and `X = rand(4,4)` in Matlab [14].) We perform QR factorizations using the modified Gram-Schmidt method; see [6].

In Figures 4.1–4.4 we plot the error in each Lyapunov exponent approximation (2.16) against Δt, on a log-log scale. The dashed line with '$\circ$' markers in each picture is of the slope $\min\{\alpha, p\}$, given by the convergence rate bound of Theorem 2.1.

In Figure 4.1 we take Lyapunov exponents 5, 2, 0, and -1. We use $S(z) = 1 + z + z^2/2 + z^3/6 + z^4/24$ in (2.15), which corresponds to a 4th order, 4 stage, explicit Runge–Kutta method, so $p = 4$. We set $K = 0.5$ and $\alpha = 1$.

For Figure 4.2 we use Lyapunov exponents $1, -1, -5$, and -10 and take $S(z) = 1 + z + z^2/2$, which corresponds to a 2nd order, 2 stage, explicit Runge–Kutta method, so $p = 2$. We set $K = 0.005$ and $\alpha = 2$.

Figure 4.3 arises with Lyapunov exponents $2, 1, -2$, and -2.5. In this case we use $S(z) = 1 + z + z^2/2$, so $p = 2$, and set $K = 0.1$ and $\alpha = 2.5$.

In Figure 4.4 we illustrate the use of an implicit ODE timestepping method. We take $S(z) = 1/(1 - z)$, which corresponds to the Backward Euler method [9], for which $p = 1$. We used Lyapunov exponents of 3.5, 1, -1, and -20, and set $K = 0.05$ and $\alpha = 1$.

In these tests, we see that the convergence rate of $\Delta t^{\min(\alpha,p)}$ arising in Theorem 2.1 is indeed an upper bound on the actual convergence rate, and it is generally sharp.

4.1.2. Continuous QR Algorithm. We now test the convergence of the continuous QR algorithm in a similar manner to §4.1.1. In Figure 4.5 we use Lyapunov exponents $3, 0, -2, -3$. We take $\Delta t = 0.1$ and solve (2.11) using the classical 4th order Runge–Kutta method with "projection" into the orthogonal QR factor using modified Gram-Schmidt.

In Figure 4.6 we take Lyapunov exponents $7, 6, 1, -1$. We set $\Delta t = 0.05$ and use the 1-stage 2nd order GLRK method to solve the ODE (2.11).

Figures 4.5 and 4.6 show that the bound in Theorem 2.2 is sharp—on a log-log scale the slope of each line is close to -1.

We include Figure 4.7 as an illustration of what may happen when a method that does not preserve orthogonality is used. In this case, we have taken Lyapunov exponents $8, 5, 2, 1$,

ETNA
Kent State University
etna@mcs.kent.edu

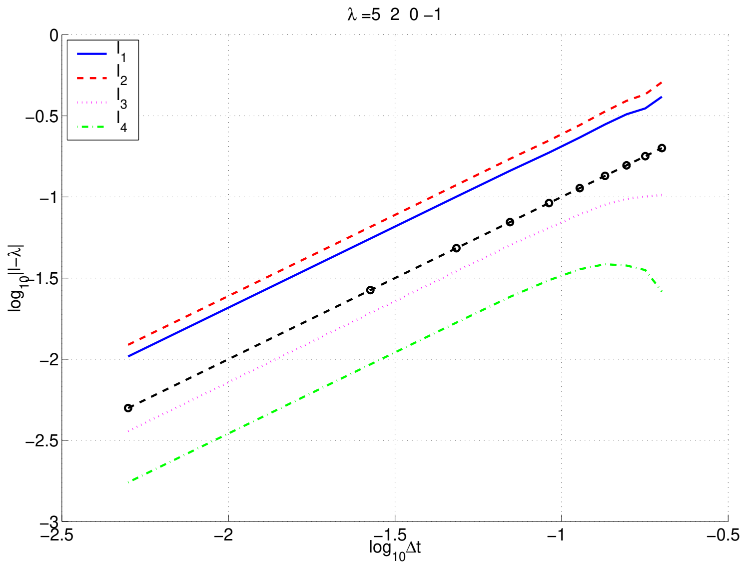

FIG. 4.1. *Discrete QR algorithm:* $\lambda = \{5, 2, 0, -1\}$*,* $p = 4$*,* $\alpha = 1$*.*

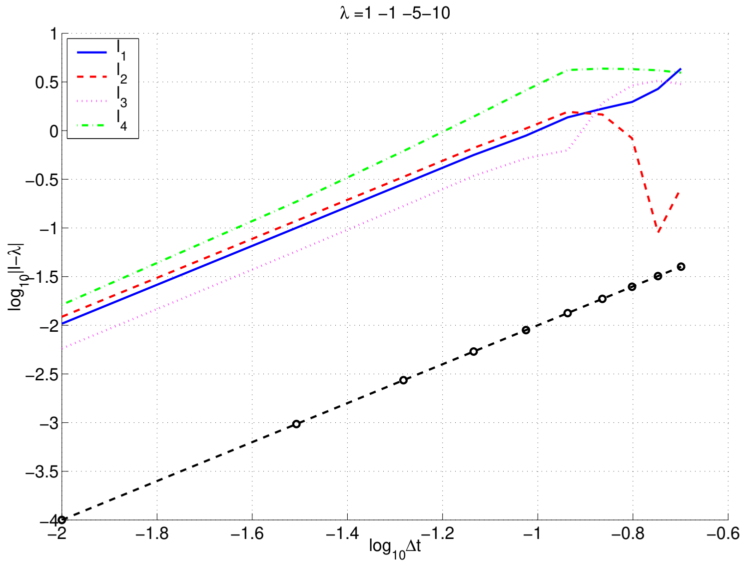

FIG. 4.2. *Discrete QR algorithm:* $\lambda = \{1, -1, -5, -10\}$*,* $p = 2$*,* $\alpha = 2$*.*

$\Delta t = 0.04$, and used the classical 4th order Runge–Kutta method. It is clear that the algorithm is no longer converging to the true Lyapunov exponents. Closer inspection has shown this non-convergence is caused by the ODE solver approaching a steady-state of (2.11) that is not orthogonal, and hence is not a Schur matrix.

4.2. Complex Conjugate Eigenvalues. We now give some numerical results for the case of complex conjugate eigenvalues. In this case the Lyapunov exponents are the real parts of the eigenvalues. The next subsection reviews the behaviour of orthogonal iteration

ETNA
Kent State University
etna@mcs.kent.edu

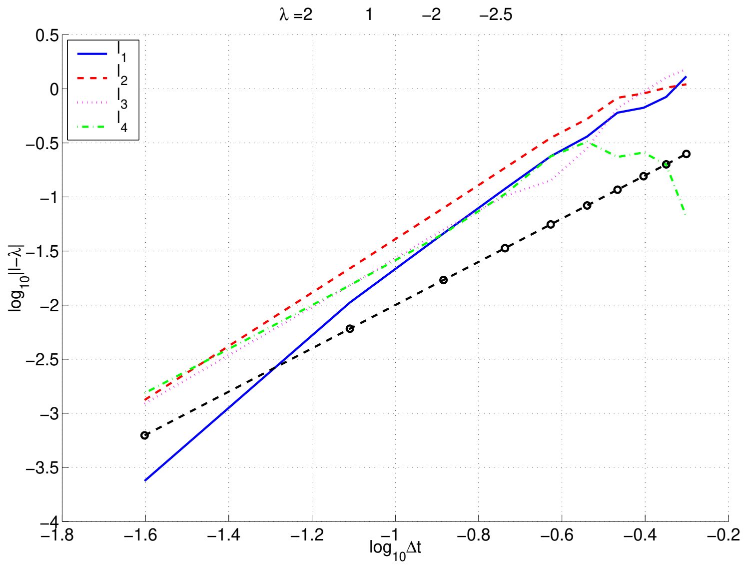

FIG. 4.3. *Discrete QR algorithm:* $\lambda = \{2, 1, -2, -2.5\}$, $p = 2$, $\alpha = 2.5$.

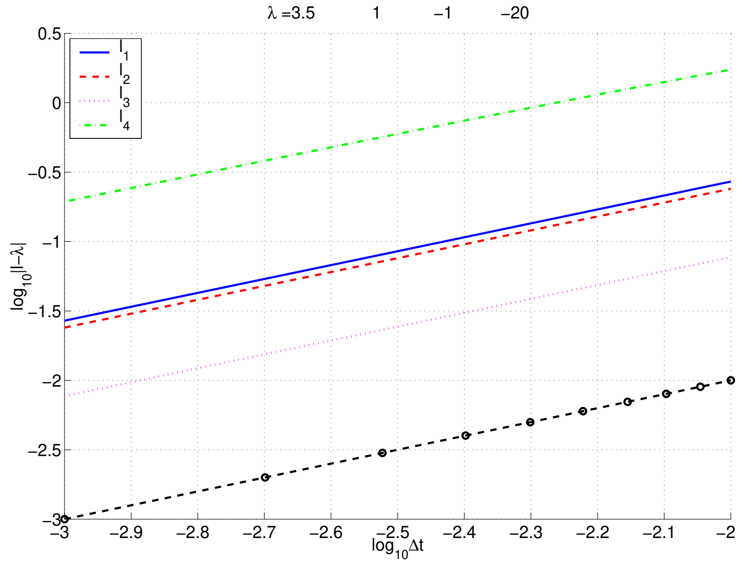

FIG. 4.4. *Discrete QR algorithm:* $\lambda = \{3.5, 1, -1, -20\}$, $p = 1$, $\alpha = 1$.

on a fixed matrix and then looks at the discrete QR algorithm. Subsection 4.2.2 deals with the continuous QR algorithm.

4.2.1. Discrete QR Algorithm. If the orthogonal iteration process described in §3.1 is applied to a matrix B that has a complex conjugate pair of eigenvalues, then $Q_j^T BQ_j$ converges to a block triangular form. The eigenvalues of the appropriate 2×2 block of $Q_j^T BQ_j$ converge to the corresponding complex conjugate eigenvalue pair (although the 2×2 block itself will not have a fixed limit). For a fuller explanation of convergence of the QR

ETNA
Kent State University
etna@mcs.kent.edu

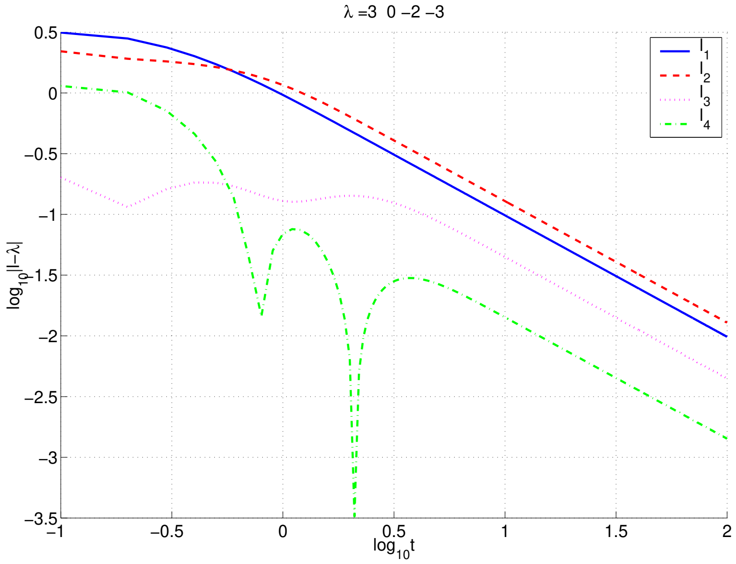

FIG. 4.5. *Continuous QR Algorithm,* $\lambda = \{3, 0, -2, -3\}$*, PRK4,* $\Delta t = 0.1$*.*

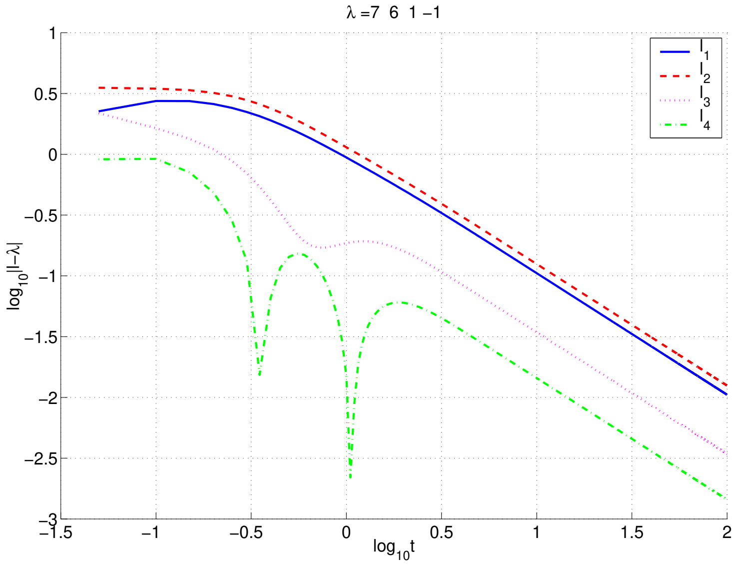

FIG. 4.6. *Continuous QR Algorithm:* $\lambda = \{7, 6, 1, -1\}$*, GLRK2,* $\Delta t = 0.05$*.*

algorithm in the complex case, see [1, 12, 17].

It can be shown that the sum of entries on the diagonal of B which correspond to the complex conjugate pair converge linearly to the sum of the real parts of the pair. This corresponds to the fact that the trace of a 2×2 block is equal to the sum of its eigenvalues. Therefore, we conclude that, when summed, the diagonal entries of $Q_j^T B Q_j$ contain the real part eigenvalue information that relates to the Lyapunov exponents.

The discrete QR algorithm for Lyapunov exponents, however, does not use $Q_j^T B Q_j$ but

ETNA
Kent State University
etna@mcs.kent.edu

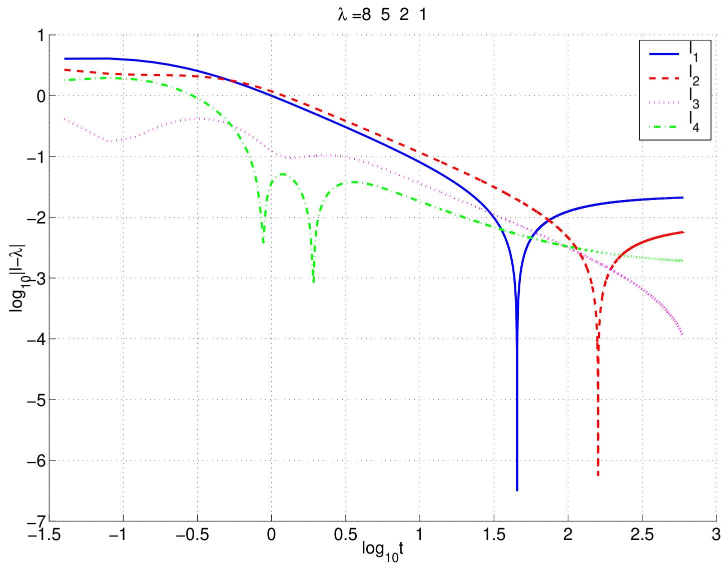

FIG. 4.7. *Continuous QR Algorithm:* $\lambda = \{8, 5, 2, 1\}$*, RK4,* $\Delta t = 0.04$*.*

the shifted version $R_{j+1} = Q_{j+1}^T B Q_j$. The columns of Q_{j+1} that correspond to a complex eigenvalue pair are typically different from the corresponding columns of Q_j; the space spanned by these columns is converging linearly but the columns themselves are not. Thus, the diagonal entries of the 2×2 block of R_{j+1} may differ greatly from the corresponding entries in B_{j+1}. Numerical experiments have shown that the two diagonal entries of R_{j+1}, even when summed, may not reveal any information about the real parts of the eigenvalues of B, and it is tempting to assert that the discrete QR algorithm will fail in the case of complex conjugate eigenvalues.

To test this assertion, Figure 4.8 gives results for the full discrete QR algorithm using a matrix A with eigenvalues $4, 1-3i, 1+3i, -2$ created as $A = XDX^{-1}$, where

$$D = \begin{bmatrix} 4 & & & \\ & 1 & 3 & \\ & -3 & 1 & \\ & & & 2 \end{bmatrix}$$

and X is a random matrix, as described in §4.1.1. We used $S(z) = 1 + z + z^2/2$, so $p = 2$, with $K = 0.01$ and $\alpha = 2$. The figure shows the surprising result that the full discrete QR algorithm is convergent with rate indicated by Theorem 2.1.

So why is the discrete QR algorithm still convergent for complex eigenvalues? Above we were considering a ‘shifted’ version of orthogonal iteration applied to a fixed matrix, while the example in Figure 4.8 deals with a matrix parametrized by Δt and zooms in on the limit $\Delta t \to 0$. A heuristic explanation for the success of the full discrete QR algorithm is provided by the observation that if A has a complex eigenvalue $\lambda = a + ib$, then the corresponding eigenvalue of $S(\Delta t A)$ looks like $1 + a\Delta t + ib\Delta t + \mathcal{O}\left(\Delta t^2\right)$. The modulus of this eigenvalue is $1 + 2a\Delta t + \mathcal{O}\left(\Delta t^2\right)$—the imaginary part of λ has an $\mathcal{O}\left(\Delta t^2\right)$ effect compared to the $\mathcal{O}(\Delta t)$ effect of the real part. Hence, in the limit $\Delta t \to 0$ we expect the real eigenvalue performance to be relevant.

ETNA
Kent State University
etna@mcs.kent.edu

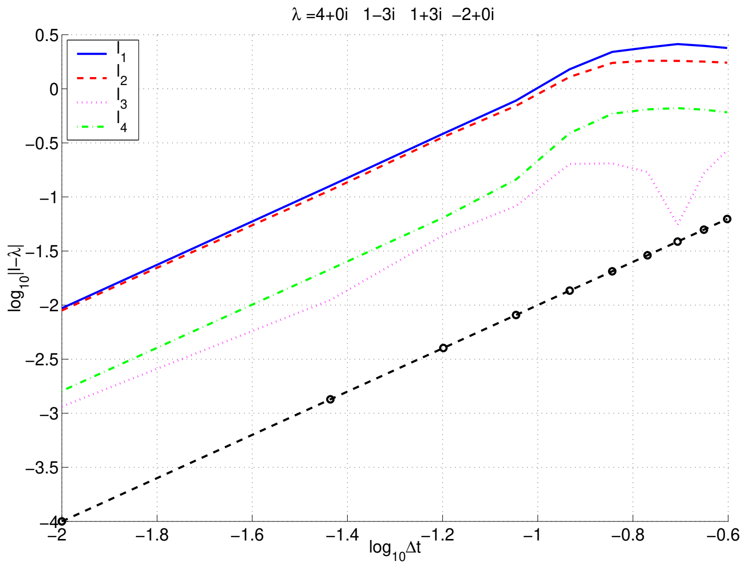

FIG. 4.8. *Discrete QR algorithm:* $\lambda = \{4, 1 \pm 3i, -2\}$, $p = 2$, $\alpha = 2$.

4.2.2. Continuous QR Algorithm. If the Jacobian matrix A contains a pair of complex conjugate eigenvalues, then its (real) Schur form will be block upper–triangular with 2×2 blocks, the eigenvalues of which correspond to each pair of complex eigenvalues. Despite the fact that the continuous QR algorithm uses only information about the diagonals, we observed that the algorithm converged in practice (as did the discrete QR algorithm discussed in the previous subsection). Figure 4.9 illustrates the behaviour.

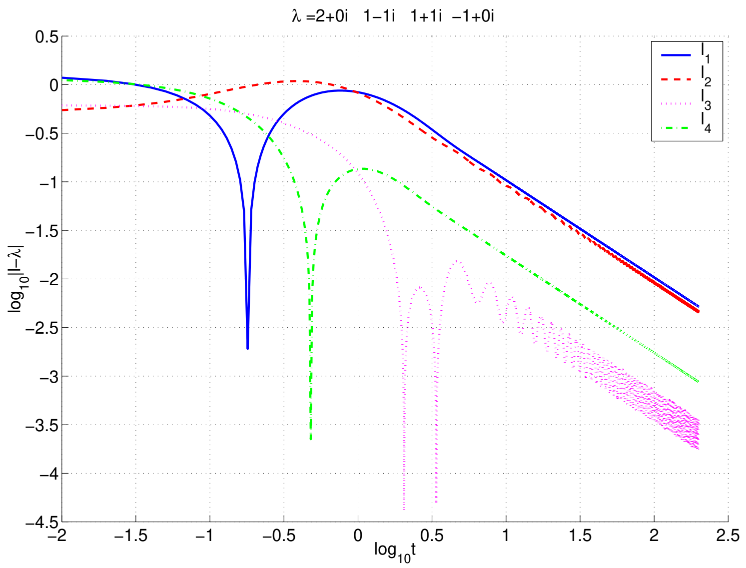

FIG. 4.9. *Continuous QR algorithm:* $\lambda = \{2, 1 \pm i, -1\}$, *PRK4,* $\Delta t = 0.01$.

5. Discussion. Our approach in this work was to analyse QR algorithms on a simple test problem, so that rigorous convergence rate bounds could be established. By choosing $A(t)$ constant in (2.1) and making the assumption (2.14), the mathematical problem reduces to one of linear algebra—find the eigenvalues of A, although the corresponding analysis of the numerical algorithms requires results from both numerical linear algebra and ODEs.

On this problem class the discrete QR algorithm is clearly not optimal. In particular, for each j in (2.16), $(R_j)_{kk}$ is approximating the same quantity. Since the accuracy increases with j, earlier values could be discarded. Indeed, the analysis in §3.2 can be used to show that taking the extreme case $\ell^{[k]} = (\log(R_N)_{kk})/\Delta t$ in (2.16) improves the error bound in Theorem 2.1 to $C_1 \Delta t^p$ (independent of $\alpha > 0$). However, for general time-dependent $A(t)$ it is clear from (2.7) that the averaging process inherent in (2.16) is necessary. Furthermore, in the case where $A(t)$ is constant but has complex conjugate eigenvalues, the averaging in (2.16) may compensate for the fact that the algorithm looks only at diagonal elements (rather than 2×2 blocks).

By a similar argument, the continuous QR algorithm loses optimality on this problem class by timestepping to steady state rather than jumping there in a single step, but the timestepping provides the averaging process that is needed for more general problems.

On a practical note, our analysis highlighted the need to deal simultaneously with the two limits $\Delta t \to 0$ and $T \to \infty$ when using the discrete QR algorithm. The relation (2.17) that we used to couple the two parameters may also be of use in more realistic simulations. In the general case where a convergence bound of the form (2.18) is not available, it would be possible to monitor convergence as Δt decreases, and hence adaptively refine the value of α in order to balance the errors.

There is much scope for further theoretical work in this area, including (a) fully analysing the case of complex conjugate eigenvalues and (b) extending the rigorous analysis to more general problem classes, such as the Floquet case [4, pages 412–413]. Given the importance of Lyapunov exponent computations in quantifying the dynamics of long-term simulations, it is clearly of interest to develop tools for analysing and comparing numerical methods, even on simple test problems.

Acknowledgements. We thank Pete Stewart for explaining to us how Lemma 3.2 follows from the traditional subspace convergence result, and thereby allowing us to shorten our original proof.

REFERENCES

[1] P. G. CIARLET, *Introduction to Numerical Linear Algebra and Optimisation*, Cambridge University Press, Cambridge, 1989.
[2] J. W. DEMMEL, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
[3] L. DIECI, R. D. RUSSELL, AND E. S. V. VLECK, *Unitary integrators and applications to continuous orthornormal techniques*, SIAM J. Numer. Anal., 31 (1994), pp. 261–281.
[4] ———, *On the computation of Lyapunov exponents for continuous dynamical systems*, SIAM J. Numer. Anal., 34 (1997), pp. 402–423.
[5] K. GEIST, U. PARLITZ, AND W. LAUTERBORN, *Comparison of different methods for computing Lyapunov exponents*, Prog. Theor. Phys., 83 (1990), pp. 875–893.
[6] G. H. GOLUB AND C. F. V. LOAN, *Matrix Computations*, John Hopkins University Press, Baltimore and London, 3rd ed., 1996.
[7] D. J. HIGHAM, *Time-stepping and preserving orthonormality*, BIT, 37 (1997), pp. 24–36.
[8] A. ISERLES, H. Z. MUNTHE-KAAS, S. P. NØRSETT, AND A. ZANNA, *Lie group methods*, Acta Numerica, 9 (2000), pp. 215–365.
[9] J. D. LAMBERT, *Numerical Methods for Ordinary Differential Systems*, John Wiley & Sons, Chichester, 1991.

[10] G. W. STEWART, *Methods of simultaneous iteration for calculating eigenvectors of matrices,* in *Topics in Numerical Analysis II*, John J. H. Miller, Academic Press, New York, 1975, pp. 169–185.
[11] ———, *Simultaneous iteration for computing invariant subspaces of non-Hermitian matrices*, Numer. Math., 25 (1976), pp. 123–136.
[12] J. STOER AND R. BULIRSCH, *Introduction to Numerical Analysis*, Springer–Verlag, New York, Inc., 2nd ed., 1992.
[13] A. M. STUART AND A. R. HUMPHRIES, *Dynamical Systems and Numerical Analysis*, Cambridge University Press, Cambridge, 1996.
[14] THE MATHWORKS, INC., *MATLAB User's Guide*, Natick, Massachusetts, 1992.
[15] D. S. WATKINS, *Understanding the QR algorithm*, SIAM Review, 24 (1982), pp. 427–440.
[16] ———, *Isospectral flows*, SIAM Review, 26 (1984), pp. 379–391.
[17] J. H. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford, 1965.