SIAM J. SCI. COMPUT.
Vol. 21, No. 6, pp. 2275–2294

PHASE SPACE ERROR CONTROL FOR DYNAMICAL SYSTEMS*

D. J. HIGHAM†, A. R. HUMPHRIES‡, AND R. J. WAIN§

Abstract. Variable time-stepping algorithms for initial value ordinary differential equations are traditionally designed to solve a problem for a fixed initial condition and over a finite time. It can be shown that these algorithms may perform poorly for long time computations with initial conditions that lie in a small neighborhood of a fixed point. In this regime there are orbits that are bounded in space but unbounded in time, and the classical error-per-step or error-per-unit-step philosophy may be improved upon. A new error criterion is introduced that essentially bounds the truncation error at each step by a fraction of the solution arc length over the corresponding time interval. This new control can be incorporated within a standard algorithm as an additional constraint at negligible additional computational cost. It is shown that this new criterion has a positive effect on the linear stability properties and hence improves behavior in the neighborhood of stable fixed points. Furthermore, spurious fixed points and period two solutions are prevented. The new criterion is shown to be admissible in the sense that it can always be satisfied with nonzero stepsizes. Implementation details and numerical results are given.

Key words. adaptivity, fixed point, long time simulations, stability, stepsize

AMS subject classifications. 65L06

PII. S1064827597331400

1. Introduction.

We are concerned with explicit numerical methods for dynamical systems defined by autonomous initial value ordinary differential equations (ODEs)

$$u_t = f(u), \qquad u(0) = u_0 \in \mathbb{R}^m. \tag{1.1}$$

The function $f : \mathbb{R}^m \to \mathbb{R}^m$ is assumed to be continuous. Additional continuity conditions will be stated where required.

In a dynamical systems context an accurate solution of (1.1) over a given finite time-interval with a particular u_0 is often of little relevance; rather, it is the global behavior of the system for general values of u_0 in the limit as $t \to \infty$ that is of interest.

When a fixed time-stepping numerical method is used to approximate the flow of (1.1), the classical error bound between the numerical approximation and exact solution of (1.1) grows exponentially in time. Moreover, at least in the case of chaotic attractors, the actual error grows exponentially in time. This leads us naturally to question the meaningfulness of our numerical solution, and any conclusions drawn from it, in the limit as $t \to \infty$. This issue has been studied in detail over the last decade or so, and the approach of considering the numerical solution as a discrete dynamical system in its own right, and then comparing the dynamics of this system

*Received by the editors December 8, 1997; accepted for publication (in revised form) May 4, 1999; published electronically May 19, 2000.

http://www.siam.org/journals/sisc/21-6/33140.html

†Department of Mathematics, University of Strathclyde, Glasgow, G1 1XH, UK (na.dhigham@na-net.ornl.gov). The work of this author was supported by the Engineering and Physical Sciences Research Council of the UK under grants GR/K80228 and GR/M42206.

‡School of Mathematical Sciences, University of Sussex, Brighton BN1 9QH, UK (a.r.humphries@susx.ac.uk). The work of this author was supported by the Engineering and Physical Sciences Research Council of the UK under grants GR/H63456 and GR/J75258.

§Formerly at Department of Mathematics, University of Dundee, Dundee DD1 4HN, UK. The work of this author was supported by a Research Studentship from the Engineering and Physical Sciences Research Council of the UK.

with the dynamics of (1.1), has been particularly fruitful. Further details can be found in [13] and the references therein.

It is widely accepted that to be efficient an ODE algorithm must be adaptive; that is, the stepsize must be varied according to some locally based error measure. In contrast to the fixed-stepsize case, a dynamical systems oriented theory for variable stepsize algorithms is far from complete. Contributions to this area include studies [2, 5, 6] on behavior near stable equilibria, [8, 12] on systems with particular nonlinear structures, and [1] on spurious fixed points.

To motivate our work, we mention two areas in which typical adaptive ODE algorithms perform badly. The first area is behavior around a stable fixed point. Hall [5] showed that typical methods fail to capture the correct dynamics in this very simple and important scenario. An illustration of this behavior is given in section 2. A second area where poor behavior can arise was identified in [1], where it was shown that almost all adaptive explicit Runge–Kutta methods admit stable *spurious* fixed points for arbitrarily small tolerances.

In this work, we present a new type of error control that is designed to overcome these two difficulties. Moreover, we aim to illustrate that traditional error control algorithms are fundamentally tied to the finite-time/fixed initial value paradigm, and hence other approaches can be beneficial for adaptive, long time simulations.

In the next section we outline the traditional error control approach and illustrate its shortcomings near fixed points. In section 3 we motivate and introduce a new error control. The properties of this control are then analyzed; in sections 4, 5, and 6 we consider admissibility, prevention of spuriosity, and linear stability, respectively. Sections 7 and 8 cover implementation details and numerical tests. The work is summarized in section 9.

2. Standard error control. Most of the ideas in this work apply to general variable stepsize algorithms. However, in order to state precise results we focus on explicit Runge–Kutta (ERK) embedded pairs. We describe below the main details of a typical adaptive ERK algorithm of the type found in numerical software libraries. Further details can be found, for example, in [4, 11].

Let t_n denote a sequence of (unequally spaced) grid points in time and let U_n denote an approximation to $u(t_n)$. Given U_n and a stepsize $\Delta t_n := t_{n+1} - t_n$, the ERK pair is defined by

$$Y_i = U_n + \Delta t_n \sum_{j=1}^{i-1} a_{ij} f(Y_j), \qquad 1 \le i \le s, \tag{2.1}$$

$$U_{n+1} = U_n + \Delta t_n \sum_{i=1}^{s} b_i f(Y_i), \tag{2.2}$$

$$V_{n+1} = U_n + \Delta t_n \sum_{i=1}^{s} \widetilde{b}_i f(Y_i). \tag{2.3}$$

Here $\{a_{ij}, b_i, \widetilde{b}_i\}$, for $1 \le i \le s$ and $1 \le j \le i-1$, are the coefficients of the formula pair and V_{n+1} is a subsidiary approximation that is used for error control. If V_{n+1} is a lower-order approximation than U_{n+1}, then the pair is said to be operating in local extrapolation mode.

The approximation U_{n+1} is regarded as acceptable if an error criterion of the form

$$\mathrm{est}_{n+1} \le \tau \tag{2.4}$$

is satisfied, where τ is a user-specified tolerance. The two common error measures for (2.4) are

$$\text{est}_{n+1} := \|U_{n+1} - V_{n+1}\|/\Delta t_n \quad \text{and} \quad \text{est}_{n+1} := \|U_{n+1} - V_{n+1}\|, \tag{2.5}$$

which lead to *error-per-unit-step* (EPUS) and *error-per-step* (EPS) control, respectively.

The constraint (2.4) must be coupled to a stepsize selection mechanism. The theory that we develop will be largely independent of this mechanism; thus we will not consider it in detail but merely note that it is usually based on the formula

$$\Delta t_{n+1} = \theta \left(\frac{\tau}{\text{est}_{n+1}} \right)^{1/q} \Delta t_n, \tag{2.6}$$

where $\theta \in (0,1)$ is a safety factor and q is the largest integer such that $\text{est}_{n+1} = \mathcal{O}(\Delta t_n^q)$.

When (2.2)–(2.6) is applied to the scalar problem $u_t = \lambda u$, where $\lambda < 0$, the discrete solution can be regarded as a map

$$\begin{pmatrix} U_{n+1} \\ \Delta t_{n+1} \end{pmatrix} = \begin{pmatrix} R(\Delta t_n \lambda) U_n \\ \theta \left(\frac{\tau \Delta t_n^{1-k}}{|E(\Delta t_n \lambda) U_n|} \right)^{1/q} \Delta t_n \end{pmatrix}, \tag{2.7}$$

where R and E are polynomials that depend on the ERK coefficients, $k = 0$ for EPUS control and $k = 1$ for EPS control. (Here, for the moment, we assume that there are no step rejections.) Hall [5] observed that the map (2.7) has a steady state solution with $\Delta t_n \equiv \Delta t_L$, $|U_n| \equiv U_L$, where Δt_L and U_L satisfy $|R(\Delta t_L \lambda)| = 1$ and $|E(\Delta t_L \lambda)| U_L = \theta^q \Delta t_L^{1-k} \tau$. (Note that Δt_L is a stepsize that lies on the boundary of the linear stability region.) If $R(\Delta t_L \lambda) = +1$, then $(\Delta t_L, U_L)$ and $(\Delta t_L, -U_L)$ are both period one steady states, whereas if $R(\Delta t_L \lambda) = -1$, then $(\Delta t_L, \pm U_L)$ is a period two steady state. In both cases, the error criterion (2.4) is satisfied and hence there are no step rejections. Similar results for complex λ were given in [7].

These steady state solutions may be regarded as acceptable in the sense that they are all within $\mathcal{O}(\tau)$ of the correct steady state $u \equiv 0$. However, it is unsettling that the underlying dynamics cannot be reproduced exactly—in neither case is the numerical solution driven to the correct fixed point, and the period two solution does not capture the correct qualitative behavior. Furthermore, as solutions of the map (2.7), these states are not necessarily stable. Hall showed that the stability is independent of τ but depends on the coefficients of the ERK pair. In the unstable case, the numerical solution and stepsize are seen to oscillate around these steady state values.

To illustrate this behavior, we applied the Matlab [10] `ode23` routine to (1.1) with

$$f(u) = \begin{bmatrix} -10 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{2.8}$$

taking $u(0) = [10^{-4}, 10^{-4}]^T$. We remark that there is no loss of generality in placing the fixed point at the origin. We used the default value for the error tolerance, which corresponds to $\tau = 10^{-3}$. (More specifically, when we refer to Matlab's `ode23` in this work, we mean the `ode23.m` code from version 4 of Matlab. Similar behavior was observed with `ode23` from Matlab's current version 5. However, version 4 has a much shorter code that is easier to follow and edit and hence was the natural choice for testing the new controls introduced in this work.) The left-hand picture in Figure 2.1

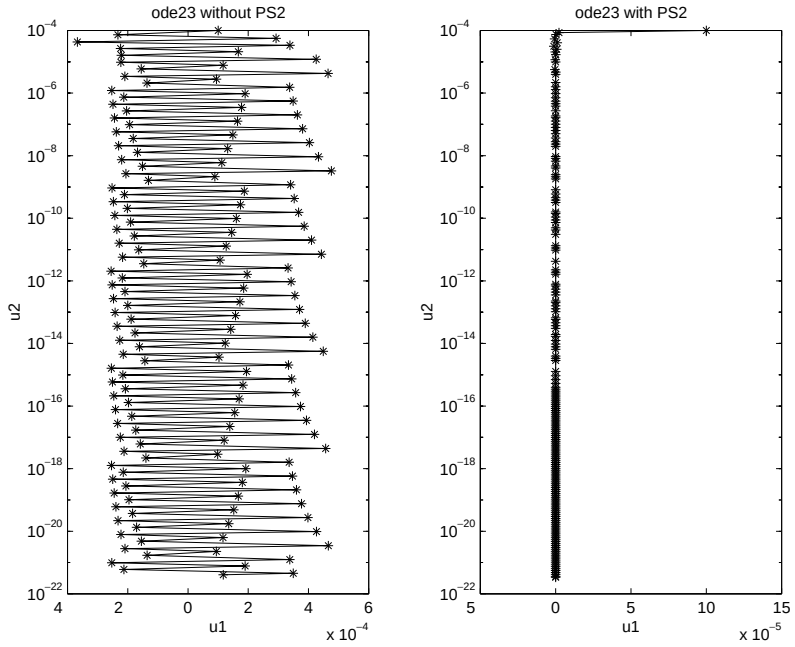

FIG. 2.1. *Numerical solutions from an ERK pair around a stable fixed point.*

illustrates the numerical solution in the phase plane for $0 \leq t \leq 20$. The discrete solution is plotted with the $*$ symbol, and these points are joined with straight lines for clarity. For this ERK pair we have $R(z) = 1 + z + z^2/2 + z^3/6$ and $R(\Delta t_L \lambda) = -1$ at the linear stability limit $\Delta t_L \lambda \approx -2.5$. The effect of the period two steady state identified by Hall can be clearly seen—the u_1 component has $\mathcal{O}(\tau)$ oscillations about zero, and these oscillations persist for all time. The oscillations are not smooth because for this ERK pair the period two solution of (2.7) is unstable. The right-hand picture in Figure 2.1 shows the same ERK method when the PS control derived in this work is incorporated. Further details are given in section 8.

In a dynamical systems context it is important to obtain a good approximation to the dynamics in the neighborhood of unstable fixed points. This is because it is often the stable and unstable manifolds of the fixed points that organize the flow in a chaotic attractor. Thus, if we cannot obtain a good approximation to the dynamics in a neighborhood of a fixed point, we may not reproduce the qualitative features of the attractor.

To illustrate this, we applied an ERK method with EPUS control and $\tau = 10^{-2}$ to (1.1) with

$$f(u) = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{2.9}$$

taking $u(0) = [0.99, 10^{-10}]^T$; very close to the stable manifold of the origin. The exact solution for $0 \leq t \leq 20$, given by the solid line in Figure 2.2, is smooth and passes close to the fixed point before following its unstable manifold—the u_2-axis. In contrast, the numerical solution oscillates about the unstable manifold of the fixed point, and although these oscillations die out as time increases, the numerical solution can ultimately end up either side of the unstable manifold. In the example given the

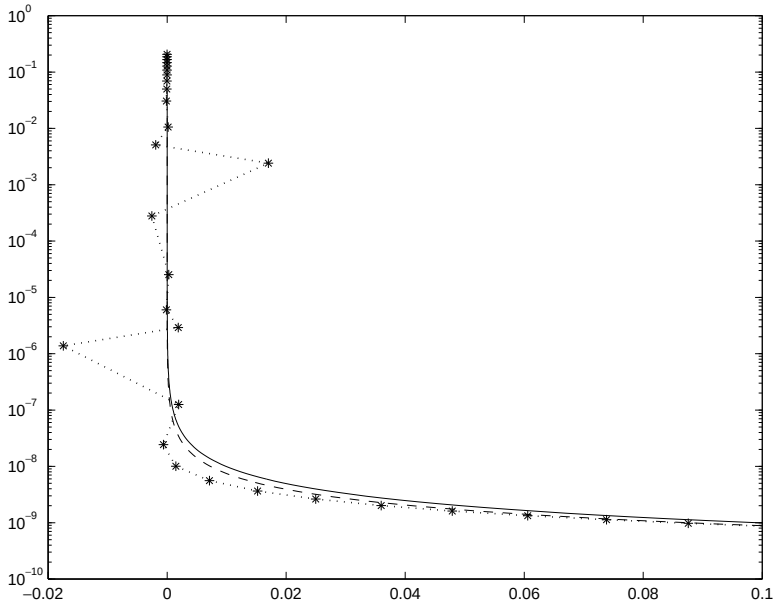

FIG. 2.2. *Numerical solutions from an ERK pair around a saddle point.*

u_1 component of the numerical solution is negative for all large n; thus, although the unstable manifold of the fixed point is a separatrix for the dynamics of the continuous problem (1.1), this is no longer true for the numerical approximation. In the general case of a chaotic attractor, numerically generated orbits passing close to fixed points may cross separatrices and consequently have different asymptotic behavior than the continuous orbit that they are intended to model. Although $\tau = 10^{-2}$ as taken in the example is relatively large, $\mathcal{O}(\tau)$ oscillations persist for all values of $\tau > 0$, and thus a modification of the error control is needed to prevent this behavior. When the PS error control is incorporated we obtain the dashed line in Figure 2.2 which follows the exact solution much more closely without oscillations. Further details are given in section 8.

3. New error control.

3.1. Motivation. The poor behavior illustrated at the end of the last section can be explained in several ways. We mention two related arguments here.

Let u^* be a fixed point of (1.1), i.e., $f(u^*) = 0$. Then given any neighborhood of u^*, there are orbits which spend arbitrarily long time intervals in that neighborhood. Both EPS and EPUS control are designed to solve the initial value problem (1.1) over a *finite* time interval, and hence we might expect them to perform badly in the neighborhood of a fixed point. Alternatively, note that from the form of the ERK formulas, in a small neighborhood of u^* we have $\| U_{n+1} - V_{n+1} \| = \Delta t_n \mathcal{O}(\| f(U_n) \|)$. It follows that by making the neighborhood sufficiently small, either EPS or EPUS may allow a step for which $\| U_{n+1} - V_{n+1} \| = \mathcal{O}(\| f(U_n) \|)$. Further, the corresponding arc length over which the solution evolves is also of size $\mathcal{O}(\| f(U_n) \|)$. Hence, the error control does not necessarily keep the local error estimate less than the arc length. Also, in this scenario there is little validity for the expansion $\text{est}_{n+1} = \mathcal{O}(\Delta t_n^q)$ on which the stepsize selection formula (2.6) is based.

As a first attempt at finding a new constraint, we may aim to control the local error as a fraction of the evolved arc length of the underlying solution of (1.1) over the corresponding time-interval. In (2.1)–(2.3), the difference $U_{n+1} - V_{n+1}$ approximates the local error in the lower order formula, and a suitable measure of $V_{n+1} - U_n$ approximates the solution arc length. Thus to bound the local error at each step as a fraction of the solution arc length we may augment (2.4) with the extra constraint

$$\|U_{n+1} - V_{n+1}\| \leq \varphi \|V_{n+1} - U_n\|, \tag{3.1}$$

where φ is a constant, chosen so that $\varphi \in (0, 1)$.

Since (3.1) relies on the interaction of the two individual ERK formulas, it is difficult to perform a general analysis of its benefits. However, note that the constraint (3.1) forces closeness in an $\mathcal{O}(1)$ sense (rather than closeness to within some power of Δt). Hence, we may replace V_{n+1} in (3.1) by some other Runge–Kutta formula. The replacment does not need to be of high order and can be chosen for its properties with regard to, for example, linear stability or spuriosity.

3.2. Phase space error control. We now study the constraint (3.1) with V_{n+1} replaced by the trapezoidal rule. The analysis of this new phase space error control forms the rest of the paper.

Phase space control (PS). In addition to (2.4), at each step we require

$$\left\| U_{n+1} - U_n - \frac{1}{2}\Delta t_n (f(U_{n+1}) + f(U_n)) \right\| \leq \frac{1}{2}\varphi \Delta t_n \|f(U_{n+1}) + f(U_n)\|, \tag{3.2}$$

where $\varphi \in (0, 1)$ is a constant.

Three key features should be noted. First, condition (3.2) is formulated solely in terms of the solution sequence $\{U_n\}_{n=0}^{\infty}$, and hence its applicability is not limited to ERK pairs. Second, the condition has little impact on the computational expense of the overall algorithm. The value of $f(U_n)$, which is required in (3.2), is already evaluated in order to compute Y_i, $i = 1, \ldots, s$. The evaluation of $f(U_{n+1})$ is also required, but if the step is accepted then this quantity is needed on the next step. (Also, ERK pairs with the "first-same-as-last" property automatically compute $f(U_{n+1})$ as a stage value for the current step [4, 11].) A third feature of condition (3.2) is that, for any consistent method, the left-hand side of the inequality is $\mathcal{O}(\Delta t_n^l)$, for some $l \geq 2$, whilst the right-hand side is $\mathcal{O}(\Delta t_n)$. Hence, away from fixed points, we would expect the condition to hold automatically for schemes of the form (2.1)–(2.6).

In the next section we show that the PS control is realistic in the sense that it can be satisfied for nonzero stepsizes. Sections 5 and 6 continue the theoretical investigations by studying the prevention of spuriosity and the endowment of linear stability properties.

4. Admissibility of phase space error controls. In this section we consider the ERK formula (2.1)–(2.2) subject to PS control (3.2) and show that the scheme is *admissible* in the sense that we can find an infinite solution sequence $\{U_n\}_{n=0}^{\infty}$ such that the PS error control is satisfied at every step. Moreover, we show that for this solution sequence it is not possible to have both $\sum_{n=0}^{\infty} \Delta t_n$ and $\{U_n\}_{n=0}^{\infty}$ bounded; hence we avoid the circumstance where $\{U_n\}_{n=0}^{\infty}$ remains bounded but the numerical solution does not progress beyond some *finite* time interval.

The term *admissible* was introduced in [12]. In that paper structural assumptions were made on f which meant that it was possible under appropriate conditions to prove both $\{U_n\}_{n=0}^{\infty}$ bounded and $\sum_{n=0}^{\infty} \Delta t_n$ unbounded for certain embedded pairs.

However, in this paper we make no structural assumptions on f and so will be able to show only that one of $\{U_n\}_{n=0}^{\infty}$ or $\sum_{n=0}^{\infty}\Delta t_n$ is unbounded.

The results in this section generalize straightforwardly to implicit Runge–Kutta methods.

We require the following notation. Let

$$\mathbb{A} = \max_i \sum_{j=1}^{i-1} |a_{ij}| \quad \text{and} \quad \mathbb{B} = \sum_{i=1}^{s} |b_i|.$$

Note that consistency of the Runge–Kutta method implies that $\mathbb{B} \geq 1$. We also require the following lemma.

LEMMA 4.1. *If f is Lipschitz on $B \subseteq \mathbb{R}^m$ with Lipschitz constant L, $U_n \in B$, and $\Delta t_n < 2/L(2\mathbb{A}+\mathbb{B})$, then any solution of (2.1)–(2.2) which satisfies $Y_i \in B$ for all i also satisfies*

$$\left\| f(Y_i) - \frac{1}{2}f(U_n) - \frac{1}{2}f(U_{n+1}) \right\| \leq \frac{1}{2}\left[\frac{L(2\mathbb{A}+\mathbb{B})\Delta t_n}{2 - L(2\mathbb{A}+\mathbb{B})\Delta t_n} \right] \|f(U_n) + f(U_{n+1})\| \tag{4.1}$$

for all $i = 1, \ldots, s$.

Proof. Using the Lipschitz continuity and the triangle inequality

$$\begin{aligned}
&\left\| f(Y_i) - \frac{1}{2}f(U_n) - \frac{1}{2}f(U_{n+1}) \right\| \\
&\qquad \leq \frac{1}{2}L\|Y_i - U_n\| - \frac{1}{2}L\|Y_i - U_{n+1}\| \\
&\qquad = \frac{1}{2}L \left\| \Delta t_n \sum_{j=1}^{i-1} a_{ij} f(Y_j) \right\| + \frac{1}{2}L \left\| \Delta t_n \sum_{j=1}^{i-1} a_{ij} f(Y_j) - \Delta t_n \sum_{j=1}^{s} b_j f(Y_j) \right\| \\
&\qquad \leq \frac{1}{2}\Delta t_n L \left\| \sum_{j=1}^{s} b_j f(Y_j) \right\| + \Delta t_n L \left\| \sum_{j=1}^{i-1} a_{ij} f(Y_j) \right\| \\
&\qquad \leq \frac{1}{2}\Delta t_n L(2\mathbb{A}+\mathbb{B})M, \qquad (4.2)
\end{aligned}$$

where $M = \max_i \|f(Y_i)\|$. Using the triangle inequality and (4.2) gives

$$\begin{aligned}
\|f(Y_i)\| &\leq \left\| f(Y_i) - \frac{1}{2}f(U_n) - \frac{1}{2}f(U_{n+1}) \right\| + \frac{1}{2}\|f(U_n) + f(U_{n+1})\| \\
&\leq \frac{1}{2}\Delta t_n L(2\mathbb{A}+\mathbb{B})M + \frac{1}{2}\|f(U_n) + f(U_{n+1})\|
\end{aligned}$$

and hence

$$M \leq \frac{1}{2}\Delta t_n L(2\mathbb{A}+\mathbb{B})M + \frac{1}{2}\|f(U_n) + f(U_{n+1})\|.$$

This rearranges to

$$M \leq \frac{1}{2 - L(2\mathbb{A}+\mathbb{B})\Delta t_n}\|f(U_n) + f(U_{n+1})\|$$

and the result follows from (4.2). ∎

We continue by proving a result in the case where f is globally Lipschitz, and then consider the more general case of f locally Lipschitz. (Recall that f is said to be locally Lipschitz if f satisfies a Lipschitz condition on every bounded subset $B \subset \mathbb{R}^m$, where the Lipschitz constant may depend upon B [9, 13].) We do this because the essence of the proofs of the two results is the same, but the globally Lipschitz case is easier and clearer to follow since it does not require some technicalities that arise in the locally Lipschitz case.

THEOREM 4.2. *Let $f : \mathbb{R}^m \to \mathbb{R}^m$ be globally Lipschitz. Then the solution sequence of the ERK formula* (2.1)–(2.2) *satisfies the PS condition* (3.2) *at every step if*

$$\Delta t_n \leq \frac{2\varphi}{L(2\mathbb{A}+\mathbb{B})(\mathbb{B}+\varphi)}. \tag{4.3}$$

Proof. We have

$$\begin{aligned}\left\| U_{n+1} - U_n - \frac{1}{2}\Delta t_n (f(U_{n+1}) + f(U_n)) \right\| &\\ = \Delta t_n \left\| \sum_{i=1}^{s} b_i \left(f(Y_i) - \frac{1}{2} f(U_{n+1}) - \frac{1}{2} f(U_n) \right) \right\| &\\ \leq \Delta t_n \mathbb{B} \max_{1\leq i\leq s} \left\| f(Y_i) - \frac{1}{2} f(U_{n+1}) - \frac{1}{2} f(U_n) \right\|. &\end{aligned}$$

Since $\varphi \in (0,1)$ and $\mathbb{B} \geq 1$, (4.3) implies that $\Delta t_n < 2/L(2\mathbb{A}+\mathbb{B})$ and by Lemma 4.1

$$\begin{aligned}\left\| U_{n+1} - U_n - \frac{1}{2}\Delta t_n (f(U_{n+1}) + f(U_n)) \right\| &\\ \leq \frac{1}{2}\Delta t_n \left[\frac{L(2\mathbb{A}+\mathbb{B})\mathbb{B}\Delta t_n}{2 - L(2\mathbb{A}+\mathbb{B})\Delta t_n} \right] \|f(U_n) + f(U_{n+1})\|. &\end{aligned}$$

Now, (4.3) implies

$$\frac{L(2\mathbb{A}+\mathbb{B})\mathbb{B}\Delta t_n}{2 - L(2\mathbb{A}+\mathbb{B})\Delta t_n} \leq \varphi$$

and condition (3.2) holds, as required. ∎

The above theorem shows that when f is globally Lipschitz, for any U_n we can find Δt_n and hence U_{n+1} such that the PS error control is satisfied. Thus we can always find a solution sequence $\{U_n\}_{n=0}^{\infty}$ when f is globally Lipschitz. Moreover, (4.3) shows that we can choose the solution sequence so that $\{\Delta t_n\}_{n=0}^{\infty}$ is uniformly bounded away from zero, and hence that $\sum_{n=0}^{\infty} \Delta t_n$ is unbounded.

We now consider the case where f is locally Lipschitz. We require the following lemma.

LEMMA 4.3. *Let f be Lipschitz with Lipschitz constant L on $\mathcal{N}(B,\varepsilon)$, where $B \subset \mathbb{R}^m$, $\varepsilon > 0$, and*

$$\mathcal{N}(B,\varepsilon) = \Big\{ x \in \mathbb{R}^m : \operatorname{dist}(x,B) < \varepsilon \Big\}, \tag{4.4}$$

and let

$$M = \sup_{u \in \mathcal{N}(B,\varepsilon)} \|f(u)\| < \infty. \tag{4.5}$$

If

$$\Delta t_n < \min\left(\frac{\varepsilon}{\mathbb{A}M}, \frac{1}{L\mathbb{A}}\right), \tag{4.6}$$

then for any $U_n \in B$ *the solution of* (2.1)–(2.2) *satisfies*

$$Y_i \in B(U_n, \varepsilon) \subset \mathcal{N}(B, \varepsilon) \qquad \text{for all } i = 1, \ldots, s,$$

where

$$B(U_n, \varepsilon) = \{x \in \mathbb{R}^m : \|x - U_n\| < \varepsilon\}.$$

Proof. See Lemma 4.2.4 in [13]. □

THEOREM 4.4. *Suppose* $f : \mathbb{R}^m \to \mathbb{R}^m$ *is locally Lipschitz. Then for any bounded set* B *and any* $U_n \in B \subset \mathbb{R}^m$ *there exists* $\widehat{\Delta t} = \widehat{\Delta t}(B) > 0$ *such that* U_{n+1} *in the ERK formula* (2.1)–(2.2) *satisfies the PS condition* (3.2) *for all* $\Delta t \in (0, \widehat{\Delta t}(B))$.

Proof. Choose $\varepsilon > 0$ and define $\mathcal{N}(B, \varepsilon)$ and M by (4.4) and (4.5), and let L be a Lipschitz constant for f on $\mathcal{N}(B, \varepsilon)$. Define

$$\widehat{\Delta t} = \min\left(\frac{\varepsilon}{\mathbb{A}M}, \frac{2\varphi}{L(2\mathbb{A} + \mathbb{B})(\mathbb{B} + \varphi)}, \frac{\varepsilon}{\mathbb{B}M}\right)$$

and note that $\varphi \in (0, 1)$ and $\mathbb{B} \geq 1$ imply that

$$\frac{2\varphi}{L(2\mathbb{A} + \mathbb{B})(\mathbb{B} + \varphi)} < \frac{1}{L\mathbb{A}}.$$

Thus Lemma 4.3 shows that $Y_i \in B(U_n, \varepsilon)$ for all i. Since $\Delta t < \varepsilon/\mathbb{B}M$, we also conclude that $U_{n+1} \in B(U_n, \varepsilon)$.

Now follow the proof of Theorem 4.2, applying (4.1) from Lemma 4.1 with $B = B(U_n, \varepsilon)$ to derive the result. □

Theorem 4.4 shows that if U_0 is in some bounded set B, then by choosing $0 < \Delta t_i < \widehat{\Delta t}(B)$ for all $i \geq n$ either the PS condition is satisfied for all $t_i \geq t_n$ or the solution sequence $\{U_i\}_{i \geq n}$ leaves B. Hence, it is possible to choose a stepsize sequence subject to PS control that has either $\sum_{i=0}^{\infty} \Delta t_i$ or $\{U_i\}_{i=0}^{\infty}$ unbounded.

5. Prevention of spuriosity. We now show that PS control, like the fixed-stepsize trapezoidal rule, does not allow either spurious fixed points or period two solutions. Note that the following result is independent of the method used to generate the solution sequence $\{U_n\}_{n=0}^{\infty}$ or the stepsize sequence $\{\Delta t_n\}_{n=0}^{\infty}$. We assume without further comment that $\Delta t_n > 0$ for all n.

THEOREM 5.1. *An algorithm that satisfies the PS constraint* (3.2) *does not admit spurious fixed points or period two solutions.*

Proof. If $U_{n+1} = U_n = U^*$ in (3.2), then

$$(1 - \varphi)\Delta t_n \|f(U^*)\| \leq 0,$$

from which it follows that $f(U^*) = 0$ as required.

Now, suppose that $U_{2n} = u$, $U_{2n+1} = v$ for all $n \geq 0$, with $u \neq v$. Consider two successive steps. From (3.2) we must have

$$\|2(v - u) - \Delta t_n(f(v) + f(u))\| \leq \varphi \Delta t_n \|f(v) + f(u)\|, \tag{5.1}$$

$$\|2(u - v) - \Delta t_{n+1}(f(v) + f(u))\| \leq \varphi \Delta t_{n+1} \|f(v) + f(u)\|. \tag{5.2}$$

From the triangle inequality,

$$(\Delta t_n + \Delta t_{n+1})\|f(u) + f(v)\| \le \|2(v-u) - \Delta t_n(f(v)+f(u))\| + \|2(u-v) - \Delta t_{n+1}(f(v)+f(u))\|.$$

Hence, using (5.1)–(5.2)

$$(\Delta t_n + \Delta t_{n+1})\|f(u) + f(v)\| \le \varphi(\Delta t_n + \Delta t_{n+1})\|f(u) + f(v)\|. \tag{5.3}$$

Since $\varphi \in (0,1)$, (5.3) implies $f(u) = -f(v)$. Hence, from (5.1), we have $\|2(v-u)\| \le 0$. So $u = v$, giving the required contradiction. ∎

6. Linear stability analysis. When the linear, scalar test problem

$$u_t = \lambda u, \tag{6.1}$$

where $\lambda \in \mathbb{R}$ or $\lambda \in \mathbb{C}$, is solved with an adaptive ERK method (2.1)–(2.3), the numerical solution advances according to

$$U_{n+1} = R(z_n)U_n, \tag{6.2}$$

where $z_n = \lambda \Delta t_n$. Here $R(z)$ is the linear stability polynomial of the Runge–Kutta formula (2.1)–(2.2) with Δt_n determined by the particular time-stepping strategy in use. We recall that the (linear) stability region, $\mathcal{S}$, for the ERK formula is defined as

$$\mathcal{S} := \{z \in \mathbb{C} : |R(z)| < 1\}.$$

In the next two subsections we investigate the behavior of adaptive ERK methods under PS error control when applied to this test problem for $\lambda \in \mathbb{R}$ and $\lambda \in \mathbb{C}$, respectively. In the third subsection we discuss the relevance of the analysis for more general linear systems.

Note that with (6.2) the PS condition (3.2) becomes

$$\left| R(z_n) - 1 - \frac{1}{2} z_n (R(z_n) + 1) \right| \le \frac{1}{2}\varphi |z_n (R(z_n) + 1)|. \tag{6.3}$$

6.1. Real λ. We begin by showing that the error control always preserves the stability of the fixed point when λ is real.

LEMMA 6.1. *Suppose the ERK formula* (2.1)–(2.2) *is applied to the linear test problem* (6.1) *with* $\lambda \in \mathbb{R}$, *and suppose that the PS condition* (3.2) *is satisfied.*

(i) *If* $\lambda < 0$, *then* $|R(\lambda \Delta t_n)| < 1$.

(ii) *If* $\lambda > 0$, *then* $|R(\lambda \Delta t_n)| > 1$.

Proof. Suppose $\lambda < 0$ and $|R(z_n)| \ge 1$. Then since $z_n < 0$, by inspection it follows that $R(z_n) - 1$ and $-z_n(R(z_n) + 1)$ have the same sign. Therefore

$$\left| R(z_n) - 1 - \frac{1}{2} z_n (R(z_n) + 1) \right| = |R(z_n) - 1| + \frac{1}{2}|z_n (R(z_n) + 1)| > \frac{1}{2}\varphi |z_n (R(z_n) + 1)|,$$

which contradicts (6.3) and thus proves (i). A similar proof works for (ii). ∎

Lemma 6.1 shows that the numerical and exact solutions both decay in modulus if $\lambda < 0$ and both grow in modulus if $\lambda > 0$. We would like to show further that the

numerical approximation to the solution of (6.1) satisfies $U_n \to 0$ as $n \to \infty$ if $\lambda < 0$ and $|U_n| \to \infty$ as $n \to \infty$ if $\lambda > 0$. To do this we must bound $|R(z_n)|$ strictly away from 1. This motivates the next theorem.

We denote the stability function of the trapezoidal rule by $R_{TR}(z)$; that is,

$$R_{TR}(z) = \frac{2+z}{2-z}. \tag{6.4}$$

Note that $|R_{TR}(z)| < 1$ for $z < 0 \in \mathbb{R}$ and $|R_{TR}(z)| > 1$ for $z > 0 \in \mathbb{R}$. We now show how the stability function $R(z_n)$ of an ERK formula subject to PS control can be bounded in terms of $R_{TR}(z)$.

THEOREM 6.2. *Suppose the ERK formula* (2.1)–(2.2) *is applied to the linear test problem* (6.1) *with* $\lambda \in \mathbb{R}$, *and suppose that the PS condition* (3.2) *is satisfied.*

(i) *If* $\lambda < 0$, *then*

$$-1 < R_{TR}((1+\varphi)\lambda\Delta t_n) \le R(\lambda\Delta t_n) \le R_{TR}((1-\varphi)\lambda\Delta t_n) < 1.$$

(ii) *If* $\lambda > 0$, *then*

$$\begin{aligned}
1 < R_{TR}((1-\varphi)\lambda\Delta t_n) \le R(\lambda\Delta t_n) \le R_{TR}((1+\varphi)\lambda\Delta t_n) \quad &\text{for} \quad \lambda\Delta t_n < \tfrac{2}{1+\varphi},\\
1 < R_{TR}((1-\varphi)\lambda\Delta t_n) \le R(\lambda\Delta t_n) \quad &\text{for} \quad \lambda\Delta t_n = \tfrac{2}{1+\varphi},\\
R(\lambda\Delta t_n) \le R_{TR}((1-\varphi)\lambda\Delta t_n) < -1 \quad &\text{for} \quad \lambda\Delta t_n = \tfrac{2}{1-\varphi},\\
R_{TR}((1-\varphi)\lambda\Delta t_n) \le R(\lambda\Delta t_n) \le R_{TR}((1+\varphi)\lambda\Delta t_n) < -1 \quad &\text{for} \quad \lambda\Delta t_n > \tfrac{2}{1-\varphi},
\end{aligned}$$

and for each $\lambda\Delta t_n \in (\frac{2}{1+\varphi}, \frac{2}{1-\varphi})$ *either*

$$R(\lambda\Delta t_n) \le R_{TR}((1+\varphi)\lambda\Delta t_n) < -1 \quad \text{or} \quad 1 < R_{TR}((1-\varphi)\lambda\Delta t_n) \le R(\lambda\Delta t_n).$$

Proof. Let $z_n = \lambda\Delta t_n$. First consider case (i), where $\lambda < 0$. The extreme right- and left-hand inequalities follow from the stability properties of the trapezoidal rule.

Suppose that the second inequality fails so that $R_{TR}((1+\varphi)z_n) > R(z_n)$. Using (6.4) and rearranging (noting that $2 - (1+\varphi)z_n > 0$), we find

$$R(z_n) - 1 - \frac{1}{2}(R(z_n)+1)z_n < \frac{1}{2}\varphi z_n(R(z_n)+1).$$

Now, by Lemma 6.1(i), $R(z_n) + 1 > 0$ and so the term on the right-hand side is negative. Therefore

$$\left| R(z_n) - 1 - \frac{1}{2}(R(z_n)+1)z_n \right| > \frac{1}{2}\varphi |z_n(R(z_n)+1)|,$$

which contradicts (6.3); thus the second inequality holds. The proof of the third inequality is similar.

Now consider case (ii), where $\lambda > 0$. Lemma 6.1(ii) shows that either $R(z_n) > 1$ or $R(z_n) < -1$. Suppose that $R(z_n) > 1$. Then we claim that

$$R(z_n)[2 - (1-\varphi)z_n] \ge 2 + (1-\varphi)z_n. \tag{6.5}$$

To establish (6.5), suppose that it is false and rearrange to obtain

$$2(R(z_n) - 1) - z_n(R(z_n)+1) < -\varphi z_n(R(z_n)+1).$$

But this implies that

$$|2(R(z_n)-1) - z_n(R(z_n)+1)| > \varphi |z_n(R(z_n)+1)|,$$

which contradicts (6.3), and so (6.5) must hold when $R(z_n) > 1$. Since the right-hand side of (6.5) is positive, we also require the left-hand side to be positive, which implies that

$$R(z_n) > 1 \qquad \text{only if} \qquad z_n < \frac{2}{1-\varphi}.$$

Now, dividing (6.5) by $2-(1-\varphi)z_n$ gives

$$R(z_n) \geq R_{TR}((1-\varphi)z_n).$$

Now suppose that $R(z_n) < -1$. Then we claim that

$$R(z_n)[2-(1+\varphi)z_n] \geq 2 + (1+\varphi)z_n. \tag{6.6}$$

To establish (6.6), suppose that it is false and rearrange to obtain

$$2(R(z_n)-1) - z_n(R(z_n)+1) < \varphi z_n(R(z_n)+1).$$

But, since $R(z_n)+1 < 0$, this implies that

$$|2(R(z_n)-1) - z_n(R(z_n)+1)| > \varphi |z_n(R(z_n)+1)|,$$

which contradicts (6.3), and so (6.6) must hold when $R(z_n) < -1$. Since the right-hand side of (6.6) is positive, we also require the left-hand side to be positive, which implies that

$$R(z_n) < -1 \qquad \text{only if} \qquad z_n > \frac{2}{1+\varphi}.$$

Now dividing (6.6) by $2-(1+\varphi)z_n$ implies that

$$R(z_n) \leq R_{TR}((1+\varphi)z_n).$$

The remaining inequalities are similar and straightforward to establish. □

Note that Theorem 6.2 is sharp in the limit $\varphi \to 0$, since setting $\varphi = 0$ in (3.2) forces the numerical solution to match a solution from the trapezoidal rule.

Theorem 4.2 shows that on the linear test problem it is possible to satisfy the PS constraint with a stepsize sequence that is strictly bounded away from zero. Hence, if the stepsizes are also bounded above, then for any $0 < \varphi < 1$ it follows from Theorem 6.2 that $U_n \to 0$ as $n \to \infty$ if $\lambda < 0$ and $U_n \to \infty$ as $n \to \infty$ if $\lambda > 0$.

6.2. Complex λ. We now indicate how far the results of the previous subsection can be extended to the case of complex λ in (6.1).

First we note that Lemma 6.1 does not carry through to complex λ. To see this, note that for a consistent method the left-hand side of (6.3) is of order $\mathcal{O}(z^k)$ with $k \geq 2$, whilst the right-hand side is of order $\mathcal{O}(z)$. Thus there will be a neighborhood $\mathcal{N}$ of the origin such that for all $z_n \in \mathcal{N}$ the condition (6.3) is satisfied (if this were not true we would also have a contradiction to Theorem 4.2). Generally, the boundary of the stability region intersects the imaginary axis in this neighborhood *only* at the

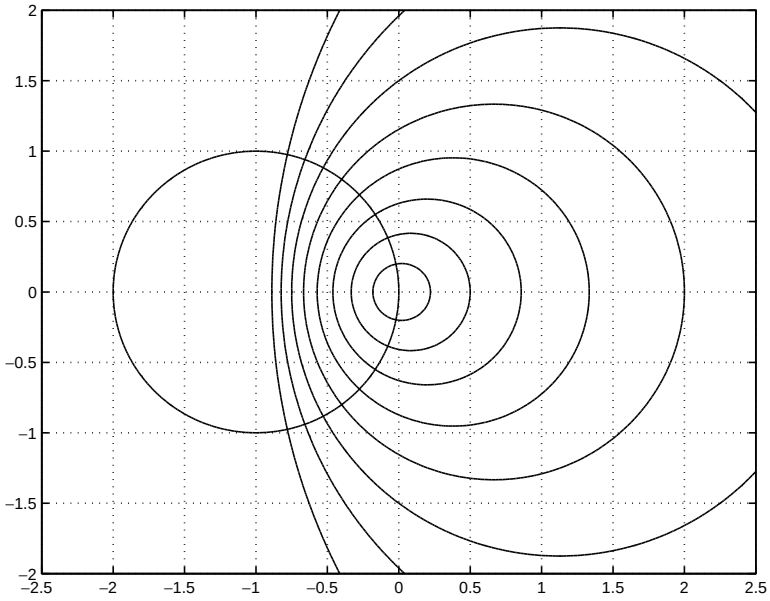

FIG. 6.1. *Euler's method: Boundaries of stability region and PS acceptable region for* $\varphi = 0.1, 0.2, \ldots, 0.8$.

origin. Hence, there must exist points close to the origin and the imaginary axis which satisfy the error control (3.2) but for which either $\mathrm{Re}(\lambda) < 0$ and $|R(\lambda\Delta t_n)| > 1$ or $\mathrm{Re}(\lambda) > 0$ and $|R(\lambda\Delta t_n)| < 1$.

Example 6.3. Consider Euler's method subject to PS control. The stability polynomial for Euler's method is $R(z) = 1 + z$, for which (6.3) reduces to $|z_n| \leq \varphi|2 + z_n|$. Letting $z_n = x + iy$, this condition simplifies to

$$\left(x - \frac{2\varphi^2}{1-\varphi^2}\right)^2 + y^2 \leq \frac{4\varphi^2}{(1-\varphi^2)^2}.$$

Thus the acceptable region $\mathcal{Q}(\varphi)$ of points that satisfy the PS condition (3.2) is given by the closed disc

$$\mathcal{Q}(\varphi) = \left\{ z : \left| z - \frac{2\varphi^2}{1-\varphi^2} \right| \leq \frac{2\varphi}{1-\varphi^2} \right\}. \tag{6.7}$$

The boundaries of the stability region $\mathcal{S} := \{z : |z + 1| < 1\}$ and acceptable region (6.7) for $\varphi = .1, .2, .3, \ldots, .8$ are shown in Figure 6.1. (Note that as φ increases the region $\mathcal{Q}(\varphi)$ becomes larger.)

Note that for this method, if $Re(\lambda) > 0$, then $|R(z_n)| > 1$ (irrespective of the condition (3.2)). However, in line with the remarks before this example, both $|R(z_n)| > 1$ and $|R(z_n)| < 1$ can occur when (3.2) is satisfied with $Re(\lambda) < 0$.

The previous example suggests the following generalization of A(α)-stability [11] to variable time-stepping methods.

DEFINITION 6.4. *An* ERK *formula subject to PS control is said to be A(α)-stable if*

$$S_\alpha \cap \mathcal{Q} \subseteq \mathcal{S},$$

where $S_\alpha = \{z : |\arg(-z)| \le \alpha\}$, $\mathcal{S}$ *is the stability region of the formula, and* $\mathcal{Q}$ *is the PS acceptable region; that is,* $\mathcal{Q}$ *is the set of* $z_n \in \mathbb{C}$ *for which* (6.3) *holds.*

THEOREM 6.5. *Euler's method subject to PS control is A(*α*)-stable with* $\alpha = \tan^{-1}(1/\varphi)$.

Proof. Using (6.7), a straightforward calculation shows that $\partial\mathcal{S}$ and $\partial\mathcal{Q}$ intersect at $z_\pm = (-2\varphi^2/(1+\varphi^2) \pm i2\varphi/(1+\varphi^2)$ with $\arg(-z_\pm) = \tan^{-1}(\pm 1/\varphi)$. ∎

We now briefly discuss how these results generalize to other ERK methods. We note first that it follows trivially from (6.3) that the acceptable region $\mathcal{Q}(\varphi)$ is monotonically increasing; that is, $\mathcal{Q}(\varphi_1) \subseteq \mathcal{Q}(\varphi_2)$ if $\varphi_1 \le \varphi_2$. In order to gain some insight, we investigate $\mathcal{Q}(0)$. From (6.3), $\mathcal{Q}(0)$ is given by solving

$$R(z) - 1 - \frac{1}{2}z(R(z)+1) = 0. \tag{6.8}$$

Now for an explicit s-stage method (6.8) is a polynomial of degree $s+1$ and hence has $s+1$ roots. If the method is of at least second order, then $R(z) = 1+z+z^2/2+\mathcal{O}(z^3)$ and three of these roots will be at the origin. The location of the other $s-2$ roots influences the acceptable region. By (3.2), the solutions of (6.8) correspond to values of z for which the method agrees with the trapezoidal rule, so it follows that

$$\text{if } \xi \text{ is a root of (6.8), then } \begin{cases} \mathrm{Re}(\xi) < 0 & \Rightarrow \quad \xi \in \mathrm{int}(\mathcal{S}), \\ \mathrm{Re}(\xi) = 0 & \Rightarrow \quad \xi \in \partial\mathcal{S}, \\ \mathrm{Re}(\xi) > 0 & \Rightarrow \quad \xi \notin \mathcal{S}. \end{cases} \tag{6.9}$$

In Figures 6.2 and 6.3 we present the acceptable regions $\mathcal{Q}(\varphi)$ along with the stability regions for two popular ERK formulas, namely the fourth- and fifth-order pairs of Fehlberg and Dormand and Prince (see [4, pp. 177–178]), which we refer to as FEHL4(5) and DOPRI5(4), respectively. Note that FEHL4(5) is normally used in nonextrapolation mode (so the stability function of the fourth-order formula is used), whilst DOPRI5(4) is designed to be applied in extrapolation mode (and thus stability function of the fifth-order method is used to advance the solution). We see from the figures that, for $\varphi \approx 1$, both methods admit points z with $\mathrm{Re}(z) \approx -2.5$ such that $z \in \mathcal{Q}(\varphi)$ but $z \notin \mathcal{S}$. This implies that that for these methods there are points away from the imaginary axis such that condition (6.3) is satisfied, but stability of the fixed point is lost; and so the qualitative dynamics are not preserved. To prevent this, we must choose a smaller value for φ.

By inspecting Figure 6.2 we see that for $\varphi \le 0.9$ the FEHL4(5) formula subject to PS control is A(α)-stable in the sense of Definition 6.4 for significant values of α. From Figure 6.3 we see that for $\varphi \le 0.7$ the DOPRI5(4) formula is A(α)-stable for $\alpha \approx \pi/2$.

For FEHL4(5) the three nonzero roots of (6.8) are contained in $\mathcal{S}$. However, two of these roots are very close to the imaginary axis and hence, by (6.9) and continuity, these roots are close to $\partial\mathcal{S}$. This results in a significant region of points z such that $Re(z) < 0$, $z \in \mathcal{Q}(\varphi)$ but $z \notin \mathcal{S}$, even for very small values of φ. In such cases the stability of the fixed point is lost. The DOPRI5(4) formula has its four nonzero roots of (6.8) further from the imaginary axis, and hence this difficulty does not arise.

6.3. Linear systems. By the Hartman–Grobman theorem [3] the behavior of a dynamical system in the neighborhood of a hyperbolic fixed point u^* is governed by the behavior of the linearized system

$$u_t = Au, \tag{6.10}$$

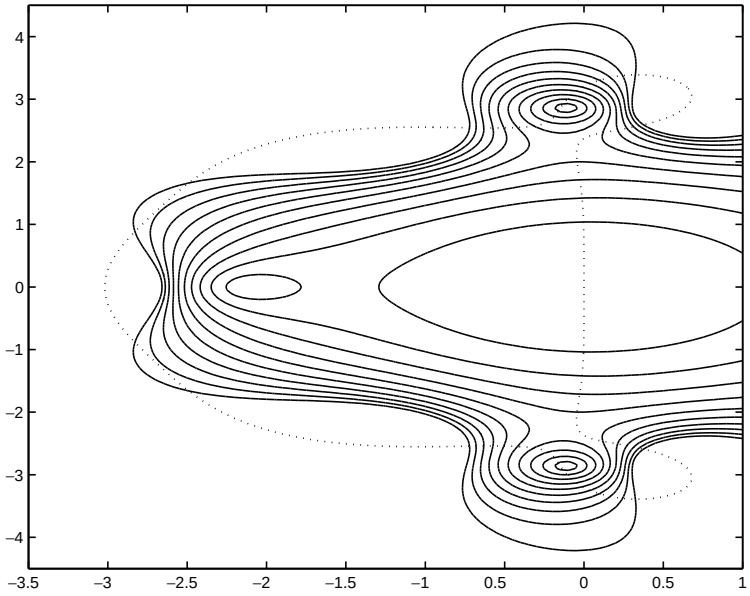

FIG. 6.2. FEHL4(5): *Boundaries of stability region (dotted line) and PS acceptable region for* $\varphi = 0.1,\ 0.2, \ldots, 0.9, 1.0$.

where A is the Jacobian of f evaluated at u^*. It can be shown that the real scalar analysis in subsection 6.1 is directly relevant to the case where the stability matrix $R(A\Delta t_n)$ has a real dominant eigenvalue and the complex scalar analysis in subsection 6.2 is directly relevant to the case where $R(A\Delta t_n)$ has a complex conjugate pair of dominant eigenvalues. Details can be found in [14].

7. Algorithm. In the previous sections we gave theoretical results about the effect of PS control. There remains the question of how to incorporate the constraints into a practical variable time-stepping algorithm. In particular, we must say how to choose a new stepsize when the PS constraint is violated (or is close to being violated). We now outline a strategy that has been arrrived at after extensive numerical testing. Our aim is to show that PS control can be added to a traditional time-stepping algorithm with few changes to the code.

It is convenient to use the following common representation of an ERK method:

$$k_i = f\left(U_n + \Delta t_n \sum_{j=1}^{i-1} a_{ij} k_j\right), \quad 1 \le i \le s, \tag{7.1}$$

$$U_{n+1} = U_n + \Delta t_n \sum_{i=1}^{s} b_i k_i, \tag{7.2}$$

with

$$\mathrm{est}_{n+1} = \left\| \sum_{i=1}^{s} e_i k_i \right\| \tag{7.3}$$

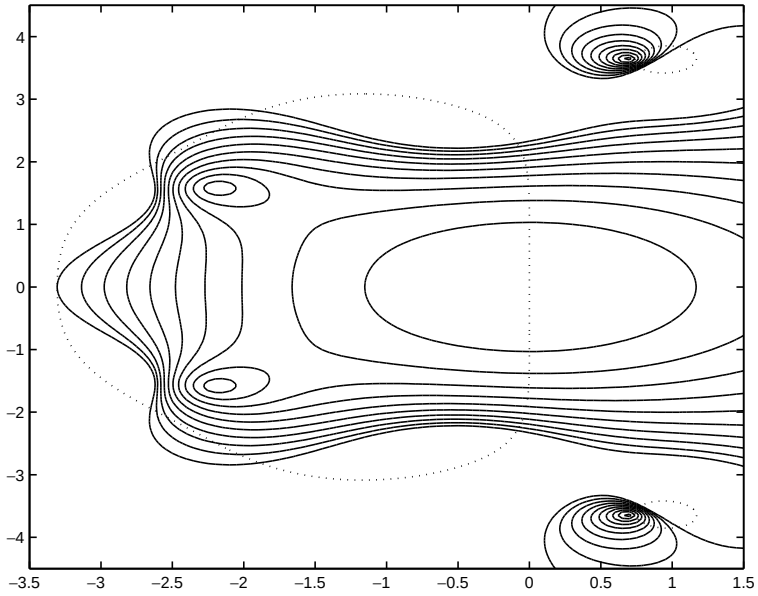

FIG. 6.3. DOPRI5(4): *Boundaries of stability region (dotted line) and PS acceptable region for* $\varphi = 0.1,\ 0.2, \ldots, 0.9, 1.0$.

for EPUS control and

$$\text{est}_{n+1} = \Delta t_n \left\| \sum_{i=1}^{s} e_i k_i \right\| \tag{7.4}$$

for EPS control. This is identical to (2.1)–(2.5), noting that $k_i = f(Y_i)$, for $i = 1 \ldots s$, and setting $e_i = b_i - \widetilde{b}_i$.

Some care is needed when implementing PS control in finite precision arithmetic. Although Theorem 4.4 shows that there is always an acceptable stepsize, since both the right- and left-hand sides of (3.2) tend to zero as $\Delta t \to 0$, in practice rounding errors could cause the rejection of what is otherwise an acceptable stepsize. To avoid unnecessary cancellation, we implement (3.2) in the equivalent form

$$\left\| \left(b_1 - \frac{1}{2} \right) f(U_n) - \frac{1}{2} f(U_{n+1}) + \sum_{i=2}^{s} b_i f(Y_i) \right\| \leq \frac{1}{2} \varphi \| f(U_{n+1}) + f(U_n) \|. \tag{7.5}$$

The basic algorithm for solving (1.1) over $0 \leq t \leq T$ can be summarized as follows.

ALGORITHM 7.1 (PS).
set $n = 0$, $U_0 = u(0)$, $t_0 = 0$, $k_1 = f(U_0)$ *and choose* Δt_0
while $t_n < T$
  compute k_i, $i = 2, \ldots s$ *from* (7.1)
  $U_{\text{new}} = U_n + \Delta t_n \sum_{i=1}^{s} b_i k_i$
  $f_{\text{new}} = f(U_{\text{new}})$
  $\text{est}_{n+1} = \| \sum_{i=1}^{s} e_i k_i \|$ *for EPS*
  $\text{est}_{n+1} = \Delta t_n \| \sum_{i=1}^{s} e_i k_i \|$ *for EPUS*
  $T_l = \| (b_1 - \frac{1}{2}) k_1 - \frac{1}{2} f_{\text{new}} + \sum_{i=2}^{s} b_i k_i \|$

$T_r = \frac{1}{2}\|f_{\text{new}} + k_1\|$
if $\text{est}_{n+1} \le \tau$ **and** $T_l \le \varphi T_r$
$U_{n+1} = U_{\text{new}}$
$k_1 = f_{\text{new}}$
$t_{n+1} = t_n + \Delta t_n$
compute Δt_{new} *and set* $\Delta t_{n+1} = \Delta t_{\text{new}}$
increment n *to* $n+1$
else
compute Δt_{new} *and set* $\Delta t_n = \Delta t_{\text{new}}$
end
end

Next, we elaborate on the strategy for computing Δt_{new}. It is common to include a *maximum stepsize ratio*, $\alpha > 1$, in a code. A typical choice is $\alpha = 5$. Consecutive stepsizes must satisfy $\Delta t_{n+1} \le \alpha \Delta t_n$; this restricts the relative increase of the stepsize over each step. It is also common to impose a maximum stepsize, $\Delta t_{\max}$, so that $\Delta t_n \le \Delta t_{\max}$ for all n. Thus, using the standard formula (2.6), we calculate

$$\Delta t_{\text{est}} = \theta \left(\frac{\tau}{\text{est}_{n+1}} \right)^{1/q} \Delta t_n \tag{7.6}$$

and set

$$\Delta t_{\text{new}} = \min\{\Delta t_{\text{est}}, \alpha \Delta t_n, \Delta t_{\max}, T - t_n\}. \tag{7.7}$$

In our new stepsize selection strategy, we allow α to change on each step in order to take account of the extra constraint (7.5). Recall that our overall aim is to depart from the stepsize that would be predicted by the local error based formula only when the phase space error is significant. Hence, letting $r := T_l/T_r$, we set $\alpha = \alpha_1$ if $r < \beta_{\min}$, where α_1 is the maximum stepsize ratio used by the traditional strategy and $\beta_{\min}$ is a small parameter, such as 0.01. In this way, we expect the new strategy to be invisible away from fixed points.

If the constraint $r \le \varphi$ is violated, then we allow the stepsize to be halved; that is, we set $\alpha = .5$. In our numerical tests, we found that it is important to take action when r is close to φ. Hence, we introduce a parameter $\beta_{\max}$ (say, $\beta_{\max} = 0.1$) such that α decreases linearly from α_1 to 1 as r inceases from $\beta_{\min}$ to $\beta_{\max}$ and α decreases linearly from 1 to .5 as r inceases from $\beta_{\max}$ to φ.

Overall, this defines $\alpha = \alpha(r)$ as follows:

$$\alpha(r) = \begin{cases} \alpha_1, & r \le \beta_{\min}, \\ \frac{\alpha_1(\beta_{\max} - r) + (r - \beta_{\min})}{\beta_{\max} - \beta_{\min}}, & \beta_{\min} \le r \le \beta_{\max}, \\ \frac{(\varphi - r) + .5(r - \beta_{\max})}{\varphi - \beta_{\max}}, & \beta_{\max} \le r \le \varphi, \\ .5, & \varphi \le r. \end{cases}$$

With this dynamic choice of α, (7.7) defines the new stepsize selection process.

One more point must be made about Algorithm 7.1. In the case where the numerical solution is driven to a fixed point, both T_l and T_r tend to zero. Hence, to avoid division by zero errors, we computed r as follows:

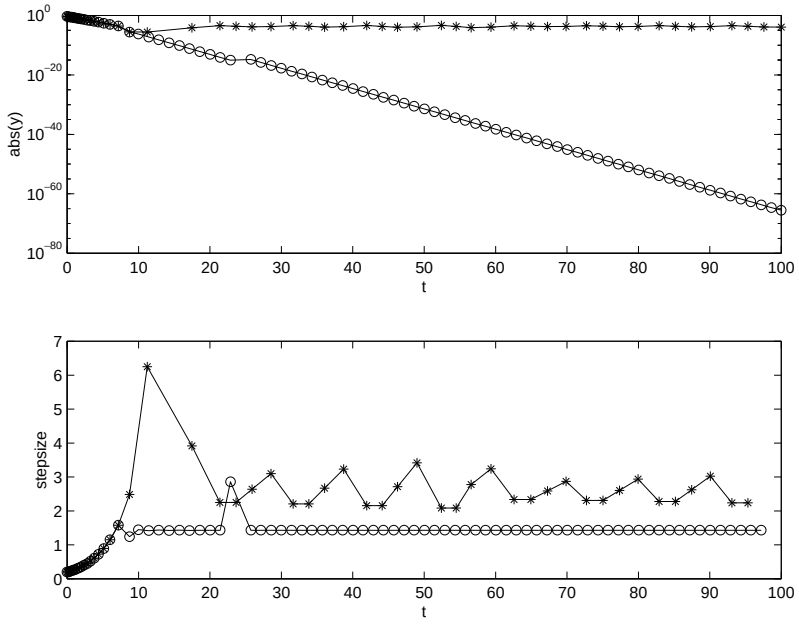

FIG. 8.1. `ode23` *around a scalar stable fixed point.*

if $T_r > \delta$
  $r := T_l/T_r$
else
  if $T_l \leq \delta$
    $r := \beta_{\max}$
  else
    $r := \varphi$
  end
end

Here, we force a decrease in the stepsize if T_r is small but T_l is not, and we keep the same stepsize if both T_r and T_l are small.

Finally, we mention that choosing a trial value for the initial time step, Δt_0, is a separate practical issue that does not significantly impact our algorithm. See [4, p. 169] or [11, p. 377] for details about initial stepsize selection.

8. Numerical tests. In this section we briefly describe some numerical experiments with PS control. Extensive testing has been done on other problems, and the conclusions shown here have been found to be valid in general.

We present results for the `ode23` code from version 4 of Matlab [10], which uses a third-order ERK formula and a secondary formula of order two. The default tolerance of $\tau = 10^{-3}$ was used. We compare the unmodified form of the code with an alternative where PS control has been implemented, as described in section 7. We used parameter values $\varphi = 0.7$, $\beta_{\min} = 0.01$, $\beta_{\max} = 0.1$, $\alpha_1 = 5$, and $\delta = 10^{-15}$, and measured vectors in the infinity norm.

We first consider the scalar linear test problem (6.1) with $\lambda = -1$ for $0 \leq t \leq 100$. The upper picture in Figure 8.1 shows the absolute value of the numerical solution for

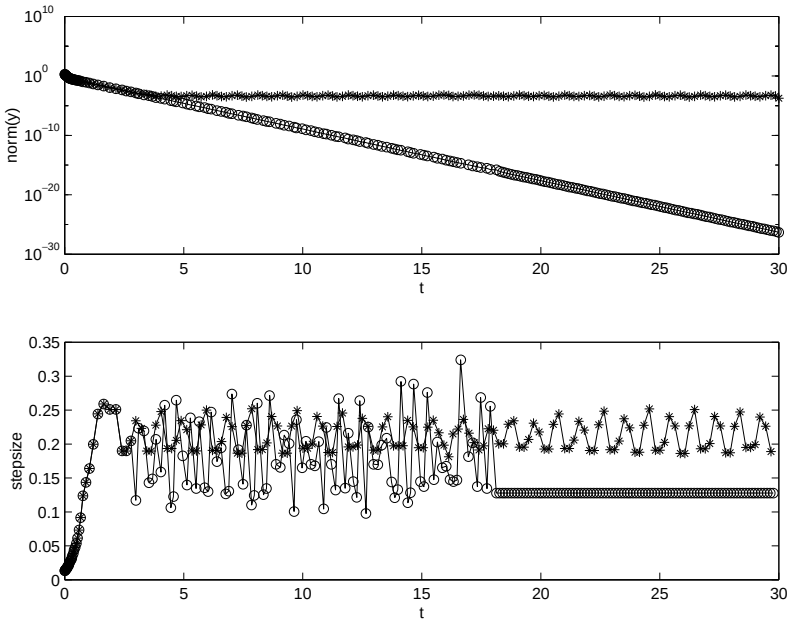

FIG. 8.2. `ode23` *around a stable fixed point with nonreal eigenvalues.*

the unmodified method (marked with `*`) and for the method with PS control (marked with `o`). It is clear that the unmodified solution remains at $\mathcal{O}(\tau)$, whilst the PS solution is driven to zero. The stepsizes used in the two cases are shown in the lower picture of Figure 8.1. For the PS method, the stepsize settles to the value $\Delta t \approx 1.43$.

Figure 2.1 in section 2 illustrates similar behavior. On the system (2.8), the unmodified method leaves $\mathcal{O}(\tau)$ oscillations in the direction of the dominant eigenvector $[1, 0]^T$. These are not present with PS control.

Next we illustrate the behavior around a stable fixed point with nonreal eigenvalues. We use the problem $u_t = Au$, where $A = Q^T BQ$, with Q a full orthogonal matrix and

$$B = \begin{bmatrix} -10 & 5 & 0 & 0 \\ -5 & -10 & 0 & 0 \\ 0 & 0 & -2 & 1 \\ 0 & 0 & -1 & -2 \end{bmatrix}.$$

(More precisely, Q was computed from `[Q R] = qr(magic(4))` in Matlab.) Note that A has eigenvalues $-10 \pm 5i$ and $-2 \pm i$. We took $y(0) = [1, 1, 1, 1]^T$ and $0 \le t \le 30$. Figure 8.2 gives the solution norm and stepsizes, using the same key as Figure 8.1. As on the scalar problem, PS control has the effect of driving the solution towards equilibrium.

We also implemented PS control with the DOPRI5(4) pair. Aves, Griffiths, and Higham [1] showed that given any tolerance, it is possible to construct a smooth function f in (1.1) for which the pair with traditional error control admits a stable spurious fixed point. Our tests confirmed that PS control avoids these solutions, in line with Theorem 5.1.

Finally, Figure 2.2 in section 2 illustrates a saddle point. For this example we took the RK1(2) method consisting of the forward Euler method with second-order EPUS

control in nonextrapolation mode with $\tau = 10^{-2}$. For PS control we took $\varphi = 0.1$, $\beta_{\min} = 0.004$, $\beta_{\max} = 0.04$, and the other values as before. In a forthcoming paper we will show how the accuracy of the numerical solution depends on φ and that the PS control method gives more accurate solutions for much less work than the unmodified method. PS control always outperforms the unmodified method. However, for stiff saddle point problems, although oscillations never occur, PS control as presented in this paper can give rise to solutions that cross the separatrix; further details and a modification to PS control to prevent this will appear in the forthcoming paper.

9. Summary. We have introduced a new error control that was motivated from a geometrical, or phase space, viewpoint. The new control does not influence the numerical solution in most regions of phase space but improves the performance near fixed points. More precisely, the new control is designed to affect positively the linear stability properties around true fixed points. This enhancement is particularly relevant when the numerical solution is to be driven to a stable fixed point and, more generally, when computations take place around (stable or unstable) invariant manifolds. The new control was also proved to prevent spurious fixed points and period two solutions that might otherwise be allowed by the adaptive algorithm.

The PS constraint analyzed here was motivated by a residual test based on the trapezoidal rule. There are many other geometrically-based controls that could be considered, for example, by using residuals from other implicit formulas. Analyzing the benefits of such controls is clearly of interest.

REFERENCES

[1] M. A. Aves, D. F. Griffiths, and D. J. Higham, *Does error control suppress spuriosity?*, SIAM J. Numer. Anal., 34 (1997), pp. 756–778.

[2] D. F. Griffiths, *The dynamics of some linear mutlistep methods*, in Proceedings of the 1987 Dundee Conference on Numerical Analysis, D. F. Griffiths and G. A. Watson, eds., Pitman Res. Notes in Math. Ser., Longman, Harlow, UK, 1988, pp. 115–134.

[3] J. Guckenheimer and P. Holmes, *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields*, Appl. Math. Sci. 42, Springer-Verlag, New York, 1983.

[4] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations* I*, Nonstiff Problems*, 2nd ed., Springer Verlag, New York, 1993.

[5] G. Hall, *Equilibrium states of Runge–Kutta schemes*, ACM Trans. Math. Software, 11 (1985), pp. 289–301.

[6] G. Hall, *Equilibrium states of Runge–Kutta schemes*—II, ACM Trans. Math. Software, 12 (1986), pp. 183–192.

[7] G. Hall and D. J. Higham, *Analysis of stepsize selection schemes for Runge–Kutta codes*, IMA J. Numer. Anal., 8 (1988), pp. 305–310.

[8] D. J. Higham and A. M. Stuart, *Analysis of the dynamics of local error control via a piecewise continuous residual*, BIT, 38 (1998), pp. 44–57.

[9] A. R. Humphries, *Spurious solutions of numerical methods for initial value problems*, IMA J. Numer. Anal., 13 (1993), pp. 263–290.

[10] The Math Works, Inc., *MATLAB User's Guide*, Math Works, Natick, MA, 1992.

[11] L. F. Shampine, *Numerical Solution of Ordinary Differential Equations*, Chapman and Hall, London, 1994.

[12] A. M. Stuart and A. R. Humphries, *The essential stability of local error control for dynamical systems*, SIAM J. Numer. Anal., 32 (1995), pp. 1940–1971.

[13] A. M. Stuart and A. R. Humphries, *Dynamical Systems and Numerical Analysis*, Cambridge University Press, Cambridge, UK, 1996.

[14] R. J. Wain, *Long Term Dynamics of Adaptive Algorithms for the Numerical Solution of Ordinary Differential Equations*, Ph.D. thesis, University of Dundee, Dundee, UK, 1998.