BIT
1998, Vol. 38, No. 2, pp. 101–104

0006-3835/98/3804-0101 $12.00

Dynamical systems and adaptive timestepping ODE solvers

H. Lamba[1]*

[1] *Department of Mathematical Sciences,*
George Mason University, MS 3F2,
4400 University Drive,
Fairfax, VA 22030, USA.
E-mail: `hlamba@gmu.edu`

Abstract.

Initial value problems for ODEs are often solved numerically using adaptive timestepping algorithms. These algorithms are controlled by a user-defined tolerance which bounds from above the estimated error commited at each step. We formulate a large class of such algorithms as discrete dynamical systems which are discontinuous and of higher dimension than the underlying ODE. By assuming sufficiently strong finite-time convergence results on some neighbourhood of an attractor of the ODE we prove existence and upper semicontinuity results for a nearby numerical attractor as the tolerance tends to zero.

This assumption of sufficiently strong finite-time convergence results is then examined for adaptive algorithms that use a pair of explicit Runge-Kutta methods of different order to estimate the one-step error. For arbitrary Runge-Kutta pairs the necessary finite-time convergence results fail to hold on a set of points in the phase space that includes all the equilibria of the ODE. Therefore, in general, the asymptotic convergence results cannot be applied to attractors containing equilibria. However, for a particular class of Runge-Kutta pairs, the finite-time convergence results can be strengthened to include neighbourhoods of equilibrium points for which the Jacobian is invertible.

AMS: 34C35, 65L07, 65L20, 65L50.
Keywords Error control, numerical integration, ordinary differential equations.

1 Introduction

Modern dynamical systems theory often relies upon numerical computations to reveal long-time behaviour which can then be analyzed mathematically. Thus a precise understanding of the relationship between the asymptotic behaviour of a dynamical system and its discrete numerical approximation is of great importance. Throughout, we study one-step numerical approximations of an autonomous set of ODEs

$$\frac{du}{dt} = f(u), \quad \text{where } f : \mathbb{R}^m \to \mathbb{R}^m. \tag{1.1}$$

*This work was partially supported by NSF Grant DMS-95-04879 and Stanford University.

Throughout we assume that the vector field f is sufficiently smooth on $\mathbb{R}^m$. These smoothness requirements are determined only by the order of the Runge-Kutta methods used to form the local error estimate.

When the exact solution $u(t)$ of (1.1) with initial condition $u(0) = U$ exists for all $t \geq 0$ we denote it by the continuous semigroup $S(t)U = u(t)$. For brevity, we shall assume, without explicitly stating it, that the exact solution exists over all necessary time intervals.

Let us first consider solving (1.1) numerically using a one-step method such as a Runge-Kutta scheme of order p with a fixed timestep Δt. This algorithm then generates a solution sequence $\{U_n\}_{n=0}^N$ where $U_n \approx u(t_n)$ for $t_n = n\Delta t$. The classical global error estimate for such a fixed timestepping implementation over the integration range $[0, T]$ with timestep Δt takes the form

$$\|U_n - u(t_n)\| \leq K(B, T)\Delta t^p \quad \forall n : 0 \leq t_n \leq T \tag{1.2}$$

for all initial conditions $u(0) = U$ in some compact set B. However as $T \to \infty$ then in general the error constant $K \to \infty$ also. Thus the error estimate says nothing in the general case about the asymptotic behaviour of the numerical method. However, if (1.1) has an attractor $\mathcal{A}$ then it is known [9, 6, 17] that for sufficiently small Δt the discrete dynamical system defined by the numerical method also has an attractor $\mathcal{A}_{\Delta t}$ and furthermore

$$\lim_{\Delta t \to 0} \operatorname{dist}(\mathcal{A}_{\Delta t}, \mathcal{A}) = 0 \tag{1.3}$$

where *dist* is the Hausdorff semi-distance between two sets.

The property (1.3) implies that for all sufficiently small Δt the numerical attractor is arbitrarily close to the actual attractor and is known as upper semi-continuity. The 'converse' property, that

$$\lim_{\Delta t \to 0} \operatorname{dist}(\mathcal{A}, \mathcal{A}_{\Delta t}) = 0,$$

is known as lower semicontinuity but does not hold in the general case. These results depend upon the availability of sufficiently strong finite-time convergence results such as (1.2) where the error constant $K(\bullet, T)$ is uniform on compact sets of initial conditions. Many of the proofs and further references can be found in [17].

However, most commercial and state-of-the-art software uses *adaptive* timestepping algorithms. These generate a solution sequence $\{U_n\}_{n=0}^N$, using inductively chosen timesteps Δt_n, so that (hopefully) $U_n \approx u(t_n)$ where $t_n = \sum_{i=0}^{i=n-1} \Delta t_i$. At each step, the algorithm enforces the requirement that an *estimate* $E(U_n, \Delta t_n)$ of the local (one-step) error is less than some quantity related to the user-defined *tolerance* τ. Such methods are much more efficient than fixed timestepping algorithms, reaching a comparable level of accuracy with substantially less computational effort on most test problems. This is especially significant for dynamical systems computations where the numerical integrations are often performed over very long time intervals and/or for many different initial conditions.

It should be stressed that we will only be considering adaptive timestepping algorithms where explicit Runge-Kutta methods are used both to provide the numerical solution and the local error estimate. These algorithms are used in the vast majority of codes for numerically integrating non-stiff ODEs (adaptive algorithms based upon multi-step methods tend not to be used for non-stiff problems because of the extra computational expense each time the stepsize is changed). It will be useful throughout this paper to consider the adaptive algorithm as consisting of two essentially independent components — the local error estimate and the timestep selection mechanism. While there are only a few basic ideas underlying adaptive algorithms, there are many possible differences in implementation — in particular there are many different ways of selecting candidate timesteps.

It is only recently that rigorous finite-time convergence results as $\tau \to 0$ for such algorithms have appeared. Previous analyses, such as [13, 14, 15], relied upon assumptions that do not hold in general. One such assumption was that the maximum timestep used during the integration would tend to zero as the tolerance $\tau \to 0$. This was shown to be false [7] for trajectories that remain in the neighbourhood of an equilibrium point for a sufficiently long time. Also, the timestep selection mechanisms analysed did not correspond exactly to ones used in actual algorithms since timestep rejections were ignored.

The first rigorous finite-time convergence result [16] analysed a specific timestep selection mechanism together with a local error estimate defined via a pair of explicit Runge-Kutta methods. The Runge-Kutta methods have orders p and $p-1$, thus the difference between the respective approximations for some candidate timestep Δt gives an estimate of the error for the lower-order method. It was proved that uniform convergence results of the form

$$\|U_n - u(t_n)\| \le K(B,T)\tau^{\gamma} \quad \forall n : 0 \le t_n \le T \tag{1.4}$$

hold for U inside a compact set B, over some finite time interval $[0,T]$, provided that the initial timestep estimate Δt_0 is sufficiently small. However, crucially, the set

$$\bigcup_{t\in[0,T],U\in B} S(t)U$$

must be bounded away from both the set of equilibria, $\Gamma(0)$, and the set $\Psi(0)$ where the leading term of the expansion of $E(u,\Delta t)$ (scaled by $\|f(u)\|$) vanishes. As a result, the error estimates obtained in [16] are not uniform with respect to the distance of exact trajectories from these sets and hence, in general, not uniform for initial data from a general compact set. The breakdown of the convergence result (1.4) near equilibria is especially important from a dynamical systems perspective since many invariant sets, including all invariant sets of gradient systems and global attractors generally, contain equilibria.

In [10] the MATLAB Version 4.2 ode23 algorithm [1], which uses a slightly different timestep selection mechanism to that studied in [16], was analysed. It was proved that uniform convergence close to non-singular equilibria holds if the local error estimate is derived using explicit Runge-Kutta pairs of orders p and $p-1$ that use only p stages (such pairs only exist for $p \le 4$).

This paper has three main aims: to define a *very general* class of adaptive algorithms as dynamical systems (using only weak, and verifiable, assumptions on the timestep selection mechanism); to extend the finite-time convergence results of [16, 10] to this class; and using these results prove the existence and upper semicontinuity of numerical attractors under suitable assumptions.

In §2 we formulate a class of adaptive algorithms as discrete dynamical systems that are both discontinuous (due to the timestep selection mechanism) and defined on a state space of dimension $m+1$, as opposed to m for the set of ODEs (1.1) (because the timestep must be included as an extra state-space variable). In §3 we redefine these dynamical systems to be set-valued at the points of discontinuity and thus show that these algorithms can be considered as closed relations. Then using the standard definition of attractors for such closed relations, and assuming sufficiently strong finite-time convergence results on a neighbourhood of some attractor $\mathcal{A}$ of (1.1), we prove existence and upper semicontinuity results for numerical attractors $\mathcal{A}_\tau$ as $\tau \to 0$. In §4 we extend the finite-time convergence results of [16, 10] to this class of algorithms. The superior convergence properties close to equilibria are seen to hold under the same assumptions on the local error estimate as in [10] — in particular for error estimates derived from the same class of explicit Runge-Kutta pairs. This has implications for the numerical simulation of say, gradient systems, where the equilibria and the connections between them are of fundamental interest, and also the accurate computation of stable and unstable manifolds of equilibria in more general ODEs.

Finally in §5 we present a heuristic argument estimating the probability that the convergence assumptions do not hold on the neighbourhood of an attractor for (1.1) and suggest possible modifications to the basic timestepping strategies currently used that can help eliminate these deficiencies.

2 Adaptive timestepping algorithms

An adaptive algorithm for solving (1.1) generates a sequence $(U_n, \Delta t_n)$ where Δt_n is the timestep used to advance the numerical solution from U_n and (hopefully) $U_n \approx u(t_n)$ where $t_n = \sum_{i=0}^{n-1} \Delta t_i$. At each step, an estimate of the local error, denoted by $E(U_n, \Delta t_n)$, is required to be less than some quantity $\sigma(\tau, U_n)$ which tends to zero with the user-defined tolerance τ. The local error estimate is often calculated by evaluating the difference (measured in some norm $\| \bullet \|$) between two one-step numerical methods of different order. In §4 the precise form of $E(U_n, \Delta t_n)$ will be crucial to the strength of the finite-time convergence results that can be proved.

The timestep selection mechanism operates as follows. Suppose that the numerical solution has already been advanced to a point U_n and an acceptable timestep Δt_n has been found. Then the numerical solution is advanced using a one-step method $S^{(1)}(u, \Delta t)$, that is, $U_{n+1} = S^{(1)}(U_n, \Delta t_n)$. The algorithm then generates a sequence of possible timesteps $\Delta t_{n+1}^{(k)}$, $k = 1, 2, \ldots$, decreasing in k, stopping at $k = l$ where l is the smallest positive integer such that $E(U_{n+1}, \Delta t_{n+1}^{(l)}) \leq \sigma(\tau, U_{n+1})$. This $\Delta t_{n+1}^{(l)}$ is then taken to be the timestep Δt_{n+1} and the process repeats.

However, before we can precisely define the algorithm as a mapping, we must consider carefully the behaviour of the algorithm at the first step $n = 0$ where the value Δt_0 is an initial timestep estimate. If $E(U_0, \Delta t_0) \leq \sigma(\tau, U_0)$ then the timestep is accepted and the algorithm behaves exactly as above. If not, then the step number is incremented to $n = 1$ but the solution is not advanced, i.e. $U_1 = U_0$. We also set $\Delta t_1^{(1)} = \Delta t_0$ and the sequence $\Delta t_1^{(k)}$ is then generated until an acceptable timestep Δt_1 is identified.

In order to make the timestep selection strategy more definite we introduce the function F_1, which provides the first candidate $\Delta t_{n+1}^{(1)}$ for Δt_{n+1} based upon the previous successful step, and functions $F_k,\ k \geq 2$ which provide the k^{th} candidates given that $\Delta t_{n+1}^{(k-1)}$ was rejected. The functions $F_k,\ k \geq 2$ are often identical but for the sake of generality we do not require this to be the case. For each fixed τ, the complete algorithm is defined by the following mapping, denoted by $S_\tau^1 : (U_p, \Delta t_p) \to (U_{p+1}, \Delta t_{p+1})$.

Given $(U_n, \Delta t_n)$ then
(A) If $E(U_n, \Delta t_n) \leq \sigma(\tau, U_n)$

$$
\begin{aligned}
(2.1)\ U_{n+1} &= S^{(1)}(U_n, \Delta t_n), \quad t_{n+1} = t_n + \Delta t_n \\
(2.2)\Delta t_{n+1}^{(0)} &= \Delta t_n, \\
(2.3)\Delta t_{n+1}^{(k)} &= F_k(\Delta t_{n+1}^{(k-1)}, \Delta t_n, U_n, U_{n+1}, \tau) \text{ for } k \geq 1, \\
(2.4)\Delta t_{n+1} &= \Delta t_{n+1}^{(l)} \ \text{ where } \ l = \min_{k \geq 1}\{k : E(U_{n+1}, \Delta t_{n+1}^{(k)}) \leq \sigma(\tau, U_{n+1})\}.
\end{aligned}
$$

(B) If $E(U_n, \Delta t_n) > \sigma(\tau, U_n)$

$$
\begin{aligned}
(2.5)\ U_{n+1} &= U_n, \quad t_{n+1} = t_n \\
(2.6)\Delta t_{n+1}^{(1)} &= \Delta t_n, \\
(2.7)\Delta t_{n+1}^{(k)} &= F_k(\Delta t_{n+1}^{(k-1)}, \Delta t_n, U_n, U_{n+1}, \tau) \text{ for } k \geq 2, \\
(2.8)\Delta t_{n+1} &= \Delta t_{n+1}^{(l)} \ \text{ where } \ l = \min_{k \geq 2}\{k : E(U_{n+1}, \Delta t_{n+1}^{(k)}) \leq \sigma(\tau, U_{n+1})\}
\end{aligned}
$$

For clarity, the updating of the integration time t_n has been included within the algorithm/mapping (we shall use the terms interchangeably) although it plays no part. It will only be used when comparing the numerical solution with the exact solution.

Note that part (B) of the mapping/algorithm only comes into effect on possibly the first timestep ($n = 0$), thereafter the error control criterion automatically ensures that part (A) is chosen. The division of the algorithm into parts (A) and (B) may seem unnatural but has two major benefits. Firstly, the mapping is now *autonomous* since the value of n does not explicitly appear. Secondly, it will greatly simplify the extension of the above (single-valued) mapping to a suitable set-valued mapping.

The timesteps $\Delta t_n,\ n \geq 1$ are all used to advance the solution, while Δt_0 will only be used to advance the solution if $E(U_0, \Delta t_0) \leq \sigma(\tau, U_0)$. This in turn means that U_n is an approximation to $u(t_n)$ where

$$
(2.9) \qquad \begin{array}{ll} t_n = \sum_{i=0}^{n-1} \Delta t_i & \text{if } E(U_0, \Delta t_0) \leq \sigma(\tau, U_n) \\ t_n = \sum_{i=1}^{n-1} \Delta t_i & \qquad \text{otherwise.} \end{array}
$$

This slight complication in the definition of t_n is necessary so that we can define the algorithm as a dynamical system on a simple phase space with variables $(u, \Delta t)$ but it will not cause any difficulties in the analysis.

The mapping defined by the above algorithm is discontinuous for two reasons. Firstly, if $E(u, \Delta t) = \sigma(\tau, u)$ then some nearby points in $\mathbb{R}^m \times \mathbb{R}^+$ will be advanced using (A) while others will be advanced using (B). Secondly, if $E(U_n, \Delta t_n^{(k)}) = \sigma(\tau, U_n)$ for some k then some nearby points in the phase space may undergo an extra timestep rejection while others will not. An alternative way of describing this discontinuity is that the integer l defined in (2.4) and (2.8) is a function of U_n and Δt_n and the dynamical system is discontinuous wherever the value of l is discontinuous.

We now impose the following conditions on the functions defined in the algorithm.

(C1) The functions $\sigma(\tau, u), S^{(1)}(u, \Delta t), E(u, \Delta t)$, and $F_k(\Delta s, \Delta t, u, v, \tau)$ are all continuous with respect to all their arguments.

(C2) There exists a maximum timestep $D > 0$, independent of τ, such that

$$0 < F_k(\Delta s, \Delta t, u, v, \tau) \leq D \quad \forall \Delta s, \Delta t \in (0, D],\ u, v \in \mathbb{R}^m,\ \tau > 0. \tag{2.10}$$

(C3) Let $J \subset \mathbb{R}^m$ be bounded. Then there exist constants $C_1(J), C_2(J)$ such that

$$0 < C_1(J) \leq \sigma(\tau, u)/\tau \leq C_2(J) \quad \forall \tau > 0 \text{ and } \forall u \in J. \tag{2.11}$$

(C4) Let $J \subset \mathbb{R}^m$ be bounded. Then, there exists positive integers s, q and constants $K_1(J, D), K_2(J, D)$ such that, for all $u \in J$, $\Delta t \in (0, D]$,

$$\|S(\Delta t)u - S^{(1)}(u, \Delta t)\| \leq K_1 \Delta t^{s+1} \tag{2.12}$$

and

$$E(u, \Delta t) \leq K_2 \Delta t^q. \tag{2.13}$$

(C5) If $E(U_n, \Delta t_n) \leq \sigma(\tau, U_n)$ then there exist constants $\alpha_1 > 1 > \theta_1$ such that

$$\theta_1 \Delta t_n \leq \Delta t_{n+1}^{(1)} \leq \alpha_1 \left(\frac{\sigma(\tau, U_n)}{E(U_n, \Delta t_n)} \right)^{\frac{1}{q}} \Delta t_n. \tag{2.14}$$

Furthermore, there also exist constants $\alpha_2 > 1 > \theta_2$ such that

$$E(U_n, \Delta t_n) \leq \theta_2 \sigma(\tau, U_n) \Rightarrow \Delta t_{n+1}^{(1)} \geq \min(\alpha_2 \Delta t_n, D). \tag{2.15}$$

(C6) If $E(U_n, \Delta t_n^{(k)}) \geq \sigma(\tau, U_n)$ for some $k \geq 1$ then there exist constants $\theta_3, \theta_4 < 1$ such that

$$\theta_3 \left(\frac{\sigma(\tau, U_n)}{E(U_n, \Delta t_n^{(k)})} \right)^{\frac{1}{q}} \Delta t_n^{(k)} \leq \Delta t_n^{(k+1)} \leq \theta_4 \Delta t_n^{(k)} \quad \forall k \geq 1. \tag{2.16}$$

Condition (C1) is a minimal requirement ensuring that the only discontinuities present in (2.1)—(2.8) are those described above. Condition (C2) asserts the existence of some maximum timestep $D > 0$, independent of τ, and implies that (2.1)—(2.8) can be defined as a mapping from $\mathbb{R}^m \times (0, D]$ to itself. This is a very common feature of adaptive codes and will be used several times in the analysis. Condition (C3) reflects the fact that many codes use a mixture of absolute and relative error control depending upon the size of the solution, for example requiring

$$E(U_n, \Delta t_n) \leq \tau \max(a, b\|U_n\|)$$

for fixed $a, b > 0$. Equation (2.12) of (C4) merely states that the one-step method $S^{(1)}(u, \Delta t)$ used to advance the solution is an order-s approximation to the exact solution while (2.13) gives the order of the local error estimate. The relationship between s and q will be stated in §4 when $E(u, \Delta t)$ is calculated via the difference between two Runge-Kutta methods of different order.

Conditions (C5) and (C6) are the assumptions on the generation of the sequences $\Delta t_n^{(k)},\ k \geq 1$. Condition (C5) places restrictions on the function F_1 which generates the initial candidate at each step while (C6) is concerned with the functions $F_k,\ k \geq 2$ which control the step rejections. These (very weak) assumptions, which are explained in more detail below, play three important roles. Firstly they imply that the error control criterion can always be satisfied at each step. Secondly they generate a lower bound on the sequence of accepted timesteps ensuring that the integration time t_n increases without bound if the numerical solution sequence U_n remains bounded. Finally, they determine upper bounds on the timestep sequence which in turn lead to bounds on the finite-time global error.

The right inequality of (2.14) simply asserts a very weak upper bound on $\Delta t_{n+1}^{(1)}$ in terms of the previous accepted timestep Δt_n. The exponent $1/q$ is introduced by considering the order of the local error estimate $E(U_n, \Delta t_n)$ as given in (C4) and should, for small timesteps, result in a value of $\Delta t_{n+1}^{(1)}$ which is both acceptable yet close to being maximal subject to the error control. This heuristic is very commonly used and so it has been incorporated into (C5), although the inequality could be weakened further. The left inequality of (2.14) states that, since the timestep Δt_n was accepted, there is no need to reduce $\Delta t_{n+1}^{(1)}$ by more than some factor $\theta_1 < 1$. Finally, (2.15) ensures that if the error criterion on the previous step was satisfied by a sufficiently large margin, $1/\theta_2$, then the first estimate for the current timestep will be larger. This will be used to prove that $t_n \to \infty$ if U_n remains bounded. Two commonly-used choices for F_1, satisfying (C2) and (C5) with given constants $0 < \theta < 1 < \alpha$, are

$$\Delta t_{n+1}^{(1)} = \min(\theta \left(\frac{\sigma(\tau, U_n)}{E(U_n, \Delta t_n)} \right)^{\frac{1}{q}} \Delta t_n, D)$$

(with $\theta_1 = \theta,\ \theta_2 = \left(\frac{\theta}{\alpha_2}\right)^q$, for any $\alpha_1, \alpha_2 > 1$) and

$$\Delta t_{n+1}^{(1)} = \min(\alpha \Delta t_n, \theta \left(\frac{\sigma(\tau, U_n)}{E(U_n, \Delta t_n)} \right)^{\frac{1}{q}} \Delta t_n, D)$$

(with $\theta_1 = \theta,\ \alpha_1 > 1,\ \theta_2 = \left(\frac{\theta}{\alpha}\right)^q,\ \alpha_2 = \alpha$). The constant $\alpha > 1$ is a maximum timestep ratio enforcing $\Delta t_{n+1}/\Delta t_n \leq \alpha$ which is often used to control the maximum rate of increase of the timestep sequence. These choices are based upon an asymptotic analysis for small timesteps and are designed to try and keep the timestep as large as possible without violating the error control (see for example [12] for more details).

We now describe the conditions on the functions $F_k,\ k \geq 2$. The right inequality of (C6) states that if a timestep estimate $\Delta t_n^{(k)}$ is rejected then the next estimate $\Delta t_n^{(k+1)}$ is smaller by a factor of at least $\theta_2 < 1$. This will ensure that the algorithm terminates on each timestep. The left inequality places a lower bound on the new estimate which prevents the algorithm from accepting arbitrarily small timesteps and will be used to prove that t_n can increase without bound provided that the solution sequence U_n remains bounded. Two commonly-used choices for $F_k, k \geq 2$, (consistent with (C6)), are

$$\Delta t_n^{(k+1)} = \theta \Delta t_n^{(k)}, \quad \forall k \geq 1$$

and

$$\Delta t_n^{(k+1)} = \theta \left(\frac{\sigma(\tau, U_n)}{E(U_n, \Delta t_n^{(k)})} \right)^{\frac{1}{q}} \Delta t_n^{(k)}, \quad \forall k \geq 1$$

with in each case $\theta_3 = \theta_4 = \theta < 1$.

Most software for solving (1.1) is of course written to be as efficient as possible and there are many different timestep selection mechanisms in use, usually based upon heuristic arguments rather than any mathematical analysis. The conditions (C5) and (C6) are sufficiently weak that they should be satisfied by most existing algorithms, in particular they are satisfied by the algorithms analysed in [16, 10].

The most restrictive assumption in the above framework is that the algorithm (2.1)—(2.8) defines a mapping from $(U_n, \Delta t_n)$ to $(U_{n+1}, \Delta t_{n+1})$. This implies that no other information from previous timesteps is used to generate candidate timesteps and need not necessarily be the case. For example a more sophisticated timestep selection mechanism might attempt to reduce the number of computationally wasteful timestep rejections by not allowing the timestep to increase for several steps after a timestep rejection has occurred. Many other variations on the basic algorithm are also possible (see [5] for examples of error-controls based upon ideas from control theory). So, although the finite-time convergence results proved in §4 may be easily extended to include such algorithms, they can only be defined as dynamical systems on more complicated phase spaces and for this reason they are excluded from the current analysis.

3 Set-valued dynamical systems and asymptotic convergence results

A fundamental difficulty in the analysis of discontinuous dynamical systems is that many of the basic results no longer hold. As a simple example consider the mapping

$$\begin{aligned} x_{k+1} &= x_k/2, \quad x_k \neq 0 \\ x_{k+1} &= 1, \quad x_k = 0. \end{aligned}$$

Then the ω-limit set $\omega(B)$ for any compact set B is the point $\{0\}$ but this set is not invariant. In general, the ω-limit sets of a discontinuous mapping need not be invariant and so, in particular, are not necessarily attractors.

However, many discontinuous mappings can be extended in a natural way to set-valued dynamical systems for which there is a well-developed theory that includes much of the standard single-valued theory as a special case. Therefore we introduce a class of generalised, or set-valued, mappings by making the algorithm (2.1)—(2.8) set-valued at the points of discontinuity. We then prove that, under conditions (C1)—(C6), these set-valued mappings define closed relations. This allows us to prove the existence of numerical attractors for these generalised dynamical systems.

We make the following changes to the algorithm (2.1)—(2.8) to define a set-valued mapping $\overline{S}_\tau^1$ such that $S_\tau^1(u,\Delta t) \in \overline{S}_\tau^1(u,\Delta t)$. If $E(U_n,\Delta t_n) = \sigma(\tau,U_n)$ then both parts (A) and (B) are used. Similarly, all $\Delta t_{n+1}^{(l)}$ such that $E(U_{n+1},\Delta t_{n+1}^{(l)}) \le \sigma(\tau,U_{n+1})$ and $E(U_{n+1},\Delta t_{n+1}^{(k)}) \ge \sigma(\tau,U_{n+1})$ $\forall k < l$ and the corresponding values of U_{n+1} lie in $\overline{S}_\tau^1(U_n,\Delta t_n)$. More precisely, given $(U_n,\Delta t_n) \in \mathbb{R}^m \times (0,D]$ then $\overline{S}_\tau^1(U_n,\Delta t_n)$ is the set of all points $(U_{n+1},\Delta t_{n+1})$ satisfying:
(A) If $E(U_n,\Delta t_n) \le \sigma(\tau,U_n)$

$$\begin{aligned}
&(3.1) & U_{n+1} &= S^{(1)}(U_n,\Delta t_n), \quad t_{n+1} = t_n + \Delta t_n\\
&(3.2) & \Delta t_{n+1}^{(0)} &= \Delta t_n,\\
&(3.3) & \Delta t_{n+1}^{(k)} &= F_k(\Delta t_{n+1}^{(k-1)},\Delta t_n,U_n,U_{n+1},\tau) \text{ for } k \ge 1,\\
& & \Delta t_{n+1} &= \Delta t_{n+1}^{(l)} \ \text{ for any } \ l \ge 1 : E(U_{n+1},\Delta t_{n+1}^{(l)}) \le \sigma(\tau,U_{n+1})\\
&(3.4) & & \text{and } \ E(U_{n+1},\Delta t_{n+1}^{(k)}) \ge \sigma(\tau,U_{n+1}) \ \forall k = 1,\dots,l-1.
\end{aligned}$$

(B) If $E(U_n,\Delta t_n) \ge \sigma(\tau,U_n)$

$$\begin{aligned}
&(3.5) & U_{n+1} &= U_n, \quad t_{n+1} = t_n\\
&(3.6) & \Delta t_{n+1}^{(1)} &= \Delta t_n,\\
&(3.7) & \Delta t_{n+1}^{(k)} &= F_k(\Delta t_{n+1}^{(k-1)},\Delta t_n,U_n,U_{n+1},\tau) \text{ for } k \ge 2,\\
& & \Delta t_{n+1} &= \Delta t_{n+1}^{(l)} \ \text{ for any } \ l \ge 2 : E(U_{n+1},\Delta t_{n+1}^{(l)}) \le \sigma(\tau,U_{n+1})\\
&(3.8) & & \text{and } \ E(U_{n+1},\Delta t_{n+1}^{(k)}) \ge \sigma(\tau,U_{n+1}) \ \forall k = 2,\dots,l-1.
\end{aligned}$$

Note again that if (B) is used then the solution is not advanced but that now this may happen for $n > 0$ since if $E(U_n,\Delta t_n) = \sigma(\tau,U_n)$ then both parts (A) and (B) will define $\overline{S}_\tau^1(U_n,\Delta t_n)$.

Because the adaptive timestepping algorithm is defined on the phase space $\mathbb{R}^m \times (0,D]$ we shall find it useful to define, for all sets $X \subseteq \mathbb{R}^m \times (0,D]$, the projection $\mathcal{P}_1$ where

$$\mathcal{P}_1(X) = \{u \in \mathbb{R}^m : \exists \Delta t \in (0,D], (u,\Delta t) \in X.\}$$

We now prove that $\overline{S}_\tau^1(U_n,\Delta t_n)$ can be calculated by computing only finitely many $\Delta t_n^{(k)}$ and furthermore for any solution sequence $\{(U_n,\Delta t_n)\}_{n=0}^{n=\infty}$ with $(U_{n+1},\Delta t_{n+1}) \in \overline{S}_\tau^1(U_n,\Delta t_n)$ $\forall n \ge 1$ and U_n bounded, $t_n \to \infty$ as $n \to \infty$.

Lemma 3.1. *Let $\{(U_n,\Delta t_n)\}_{n=0}^{n=\infty}$ be any sequence generated by (3.1)—(3.8) and $J \subset \mathbb{R}^m$ be compact. Then $\forall\ n : U_n \in J$, $\exists M(\tau,J)$ such that $\Delta t_n = \Delta t_n^{(k)}$ for some $k \le M(\tau,J)$. Also, for all sufficiently small τ, $\exists \Delta t_{\min}(\tau,J) \to 0$ as $\tau \to 0$ such that $\forall n : U_n, U_{n+1} \in J$,*

$$\Delta t_n \le \Delta t_{\min} \Rightarrow \Delta t_{n+1} \ge \alpha_2 \Delta t_n \tag{3.9}$$

and

$$\Delta t_n \ge \Delta t_{\min} \Rightarrow \Delta t_{n+1} \ge \alpha_2 \Delta t_{\min}. \tag{3.10}$$

Finally, if $U_n \in J\ \forall n \geq 0$ *then* $\lim_{n\to\infty} t_n = \infty$.

PROOF. Let $U_n \in J$. Then, irrespective of whether part (A) or (B) of the algorithm is used, (C2) implies $\Delta t_n^{(1)} \leq D$ and so by (C6), $\Delta t_n^{(k)} \leq \theta_4^{k-1} D$. We define

$$\phi = (\tau C_1/K_2)^{\frac{1}{q}}$$

and note that any candidate timestep $\Delta t < \phi$ implies

$$E(U_n, \Delta t) \leq K_2 \Delta t^q < C_1 \tau \leq \sigma(\tau, U_n). \tag{3.11}$$

Therefore $\Delta t_n = \Delta t_n^{(k)}$ for some $k \leq M(\tau, J) = [2 + (\ln \phi - \ln D)/\ln \theta_4]$ where $[\bullet]$ denotes the integer part.

Before continuing we note that if $U_n \in J$ then by (C6), no timestep rejection can result in a candidate timestep $\Delta t_n^{(k)}$, $k \geq 2$ smaller than $\theta_3 \phi$ since

$$\Delta t_n^{(k)} \geq \theta_3 \left(\frac{\sigma(\tau, U_n)}{E(U_n, \Delta t_n^{(k-1)})} \right)^{\frac{1}{q}} \Delta t_n^{(k-1)} \geq \theta_3 \left(\frac{C_1 \tau}{E(U_n, \Delta t_n^{(k-1)})} \right)^{\frac{1}{q}} \Delta t_n^{(k-1)} \geq \theta_3 \phi. \tag{3.12}$$

We now define

$$\Delta t_{\min}(\tau, J) = \frac{\theta_1 \phi}{\alpha_2} \min(\theta_3, \theta_2^{\frac{1}{q}}) \tag{3.13}$$

which tends to 0 as $\tau \to 0$ and choose τ sufficiently small that $\Delta t_{\min} \leq D/\alpha_2$.

To prove (3.9) suppose that $\Delta t_n \leq \Delta t_{\min}$ and $U_n \in J$. Then

$$\begin{aligned} \Delta t_n &\leq \theta_2^{\frac{1}{q}} \phi \\ \Rightarrow E(U_n, \Delta t_n) &\leq K_2 \theta_2 \phi^q \leq \theta_2 C_1 \tau \leq \theta_2 \sigma(\tau, U_n) \\ \Rightarrow \Delta t_{n+1}^{(1)} &\geq \alpha_2 \Delta t_n \text{ by (2.15).} \end{aligned}$$

If $\Delta t_{n+1}^{(1)}$ is accepted then (3.9) holds. If it is rejected and $U_{n+1} \in J$ then $\Delta t_{n+1} = \Delta t_{n+1}^{(j)}$ for some $j \geq 2$. Then by (3.12) $\Delta t_{n+1} \geq \theta_3 \phi \geq \alpha_2 \Delta t_{\min} \geq \alpha_2 \Delta t_n$ and again (3.9) holds.

We now prove (3.10). We first note that if $\Delta t_{n+1} = \Delta t_{n+1}^{(j)}$ for some $j \geq 2$ then $\Delta t_{n+1} \geq \theta_3 \phi \geq \alpha_2 \Delta t_{\min}$ and (3.10) holds. So suppose instead that (A) is chosen and $\Delta t_{n+1} = \Delta t_{n+1}^{(1)}$, in which case if $\Delta t_n \in [\Delta t_{\min}, \alpha_2 \Delta t_{\min}/\theta_1]$ then

$$E(U_n, \Delta t_n) \leq K_2 \Delta t_n^q \leq \theta_2 C_1 \tau \leq \theta_2 \sigma(\tau, U_n)$$

and so by (2.15), $\Delta t_{n+1} \geq \alpha_2 \Delta t_n \geq \alpha_2 \Delta t_{\min}$. If, however, $\Delta t_n \geq \alpha_2 \Delta t_{\min}/\theta_1$ then by (2.14) $\Delta t_{n+1}^{(1)} \geq \theta_1 \Delta t_n \geq \alpha_2 \Delta t_{\min}$.

Finally we prove that if $U_n \in J\ \forall n \geq 0$ then $t_n \to \infty$. Note that by (3.10) and (3.9) either $\Delta t_n \geq \Delta t_{\min}\ \forall n$ or Δt_n increase geometrically at a rate $\geq \alpha_2$ until $\Delta t_{\min}$ is reached. So if the increasing sequence t_n is bounded then there exists some integer N such that step (B) is chosen for all $n \geq N$ and hence that $E(U_n, \Delta t_n) = \sigma(\tau, U_n)\ \forall n \geq N$. But then, by (C6), $\Delta t_{n+1} \leq \theta_4 \Delta t_n\ \forall n \geq N$ and so $\Delta t_m < \phi$ for some $m > N$ and step (A) must be chosen at step $m+1$ contradicting our assumption. This completes the proof. ∎

Lemma 3.1 applies to the single-valued algorithm (2.1)—(2.8) since by construction $S^1_\tau(u,\Delta t) \in \overline{S}^1_\tau(u,\Delta t)$. We now prove that the generalised dynamical system (3.1)—(3.8) defines a closed relation on any forward invariant compact subset of $\mathbb{R}^m \times (0,D]$. This will allow us to apply standard definitions and existence results for attractors of closed relations.

DEFINITION 3.1. *Let F be a set-valued mapping $F : X_1 \to X_2$ where X_1 and X_2 are compact metric spaces. Then F is a* closed relation *if the set $\{(x,y) \in X_1 \times X_2 : y \in F(x)\}$ is a closed subset of $X_1 \times X_2$. The set $A \subseteq X_1$ is* forward invariant *if $F(A) \subseteq A$ and* invariant *if $F(A) = A$.*

LEMMA 3.2. *If $B^*_\tau \subset \mathbb{R}^m \times (0,D]$ is compact and forward invariant under $\overline{S}^1_\tau$, then $\overline{S}^1_\tau$ restricted to B^*_τ is a closed relation.*

PROOF. Suppose that there exists a convergent sequence in $\overline{\mathcal{S}}^1_\tau$,

$$(u_j, \Delta s_j, v_j, \Delta t_j) \to (u, \Delta s, v, \Delta t), \quad j \geq 1$$

with

$$(v_j, \Delta t_j) \in \overline{S}^1_\tau(u_j, \Delta s_j), \quad \forall j \geq 1.$$

By the compactness of B^*_τ, $(u, \Delta s) \in B^*_\tau$ and so it is sufficient to prove that

$$(v, \Delta t) \in \overline{S}^1_\tau(u, \Delta s).$$

For each j, $(v_j, \Delta t_j)$ was obtained from $(u_j, \Delta s_j)$ using either part (A) or (B) of the algorithm (3.1)—(3.8) and so we can generate a convergent subsequence consisting of points advanced using just (A) or (B). Let us suppose first that the subsequence uses (A). Then $E(u_j, \Delta s_j) \leq \sigma(\tau, u_j)\ \forall j$. The continuity of the functions E and σ then implies that $E(u, \Delta s) \leq \sigma(\tau, u)$ and the continuity of $S^{(1)}$ implies $v = S^{(1)}(u, \Delta s)$. A similar argument holds for a subsequence using (B) but with $v = u$ since the integration is not advanced.

Now from the proof of Lemma 3.1 we can form a further convergent subsequence such that each $\Delta t_j = \Delta t_j^{(l)}$ for some fixed $0 \leq l \leq M(\tau, \mathcal{P}_1(B^*_\tau))$. By the continuity of E, σ and F_k,

$$E(v, \Delta t^{(k)}) \geq \sigma(\tau, v) \quad \forall k = 1, \ldots, l-1$$

and

$$E(v, \Delta t^{(l)}) \leq \sigma(\tau, v).$$

Therefore $\Delta t = \Delta t^{(l)}$ satisfies (3.4) or (3.8) and so $(v, \Delta t) \in \overline{S}^1_\tau$. □

We are now in a position to prove the existence and upper semicontinuity of numerical attractors for (3.1)—(3.8). The proof uses a definition of attractors for closed relations and broadly follows the outline of the proof for the fixed timestepping case, which can be found in [17] and is based on [6].

In [8] a set-valued approach was also used to prove convergence results for attractors of adaptive algorithms. However it was *assumed* that the error control was working correctly and in particular that the actual error committed on each step $T(U_n, \Delta t_n) \leq K\tau^\gamma \Delta t_n$ for some $\gamma, K > 0$. The analysis presented here is substantially different since we consider numerical approximations generated by specific algorithms which allows us to precisely state the conditions under which the necessary finite-time convergence results hold.

We start by defining an attractor for (1.1) as follows:

DEFINITION 3.2. *Let* $A, B \subset \mathbb{R}^m$. *Then* A attracts B *under the semigroup* $S(t)$ *if, for any* $\epsilon > 0$ *there exists* $t^* = t^*(\epsilon, B, A)$ *such that* $S(t)B \subset \mathcal{N}(A, \epsilon)$, *for* $t \geq t^*$. *The set* A *is an* attractor *if it is compact, invariant and attracts a neighbourhood of itself.*

We now make the following definition for an attractor of a closed relation (for equivalent definitions see [2, Theorem 3(b)(1), Chapter 3]).

DEFINITION 3.3. *Let* F *be a closed relation and* A *be a compact invariant set. Then* A *is an attractor if there exists a closed neighbourhood* B *of* A *such that* $A = \bigcap_{n\geq 0} F^n B$. Additionally if the set B is forward invariant under F then

$$A = \omega(B) := \bigcap_{n\geq 0} \overline{\bigcup_{k\geq n} F^k(B)}$$

(see [2, Proposition 4a Chapter 2]).

Next we assume the existence of an attractor for (1.1).

ASSUMPTION 1. *Equation (1.1) has an attractor* $\mathcal{A}$ *which attracts a forward invariant compact neighbourhood* B *of itself. Therefore*

$$\mathcal{A} = \omega(B) = \bigcap_{s\geq 0} \bigcup_{t\geq s} S(t)B.$$

The existence of a forward invariant compact neighbourhood B of $\mathcal{A}$ contained within any ϵ-neighbourhood of $\mathcal{A}$ follows easily from the continuity of the vector field f.

Finally we make the following assumption about the finite-time convergence of the algorithm on a neighbourhood of the set B from Assumption 1, together with bounds on the accepted timestep sequence Δt_n for sufficiently small initial timestep Δt_0. The precise conditions under which this assumption holds will be proved in §4

ASSUMPTION 2. *Let* $U_0 \in B$ *and consider any sequence* $\{(U_n, \Delta t_n)\}_{n=0}^N$ *generated by the generalised algorithm (3.1)—(3.8) satisfying (C1)—(C6). Then for all sufficiently small* $d > 0$ *there exist positive constants* $\gamma, K(\mathcal{N}(B, d))$ *and* $L(\mathcal{N}(B, d))$ *and a continuous function* $h : u \rightarrow \mathbb{R}^+$ *with* $\min_{u\in\mathcal{N}(B,d)} h(u) = \kappa > 0$ *such that, for all sufficiently small* τ, *if* $\Delta t_0 \leq \min(h(U_0), D)$, *then for all* n *such that* $t_n \leq T$,

$$U_n \in \mathcal{N}(B, d),$$

$$\|U_n - S(t_n)U_0\| \leq K t_n \tau^\gamma \exp(L t_n) \tag{3.14}$$

and

$$\Delta t_n \leq \min(\frac{h(U_n)}{2}, D). \tag{3.15}$$

Equation (3.14) is the basic convergence result while (3.15) provides an upper bound on the sequence of accepted timesteps. Henceforth we fix $d > 0$ as above.

We now prove the existence of a forward invariant set B^*_τ for the generalised dynamical system (3.1)—(3.8) under Assumptions 1 and 2.

THEOREM 3.3. *Let Assumption 1 hold and consider the approximation of (1.1) by an adaptive timestepping method satisfying (C1)—(C6) and Assumption 2.*

Then for all sufficiently small ϵ, $\exists \tau_c(\epsilon) > 0$ such that $\forall \tau \in (0, \tau_c)$ there exists a set $B_\tau^* \subset \mathbb{R}^m \times (0, D]$ satisfying

$$\mathcal{N}(\mathcal{A}, 2\epsilon) \subseteq B \subseteq \mathcal{P}_1(B_\tau^*) \subseteq \mathcal{N}(B, \epsilon/2) \subseteq \mathcal{N}(B, d).$$

Furthermore, B_τ^* is compact and forward invariant under $\overline{S}_\tau^1$.

PROOF. We choose ϵ sufficiently small so that $\mathcal{N}(\mathcal{A}, 2\epsilon) \subseteq B$ and $\mathcal{N}(B, \epsilon/2) \subseteq \mathcal{N}(B, d)$. Since $\mathcal{A}$ attracts B there exists $T > D$ such that

$$S(t)B \subseteq \mathcal{N}(\mathcal{A}, \epsilon/2) \qquad \forall t \geq T. \tag{3.16}$$

For all τ sufficiently small such that $\Delta t_{\min}(\tau, \mathcal{N}(B, d)) < C\kappa/(2\alpha_2)$ we define the compact set

$$X(\tau) = \{(u, \Delta t) : u \in B \text{ and } \Delta t \in [\Delta t_{\min}, \min(h(u), D)]\}.$$

Now, by Assumption 2 $\exists \tau_c = \tau_c(\epsilon) > 0$ such that $\forall \tau \in (0, \tau_c)$, $\forall (U_0, \Delta t_0) \in X$ and for all n such that $t_n \in [0, 3T]$,

$$\|U_n - S(t_n)U_0\| \leq \frac{\epsilon}{2} \; \forall n : t_n \leq 3T \tag{3.17}$$

and

$$\Delta t_n \leq \min(h(U_n)/2, D).$$

Equation (3.17) and the forward invariance of B under the semigroup $S(t)$ also imply that

$$U_n \in \mathcal{N}(B, \epsilon/2) \subseteq \mathcal{N}(B, d), \quad \forall n : t_n \leq 3T \tag{3.18}$$

Thus, by Lemma 3.1 and (3.18), $\Delta t_n \geq \Delta t_{\min}$ and so there exists some integer $n_0(\tau, \mathcal{N}(B, d))$ such that for any solution sequence with $(U_0, \Delta t_0) \in X$ then $t_{n_0} \geq T$. We now define

$$B_\tau^* = \bigcup_{0 \leq n \leq n_0(\tau, \mathcal{N}(B,d))} \overline{S}_\tau^n(X)$$

and clearly $B \subseteq \mathcal{P}_1(B_\tau^*)$. To prove that $\mathcal{P}_1(B_\tau^*) \subseteq \mathcal{N}(B, \epsilon/2)$ consider any sequence $\{(U_i, \Delta t_i)\}_{i=0}^{n_0})$ with $(U_0, \Delta t_0) \in X$. Then $T \leq t_{n_0} \leq n_0 D < \infty$. Equations (3.16) and (3.17) together imply

$$U_n \in \mathcal{N}(\mathcal{A}, \epsilon) \subseteq B \quad \forall n : t_n \in [T, 3T] \tag{3.19}$$

and so by (3.14) and (3.10)

$$(U_n, \Delta t_n) \in X \; \forall n : t_n \in [T, 3T]. \tag{3.20}$$

Now let $n_{\min}$ be the smallest integer such that for some $(U_0, \Delta t_0) \in X$, $\exists U_{n_{\min}} \in \mathcal{P}_1(\overline{S}_\tau^{n_{\min}}(U_0, \Delta t_0))$ with $U_{n_{\min}} \notin \mathcal{N}(B, \epsilon/2)$. Then, since $T > D$, there exists some $p > 0$ such that $t_p \in [T, 3T]$ with $(U_p, \Delta t_p) \in \overline{S}_\tau^p(U_0, \Delta t_0)$ and $U_{n_{\min}} \in \mathcal{P}_1(\overline{S}^{n_{\min}-p}(U_p, \Delta t_p))$. But $(U_p, \Delta t_p) \in X$ and so $n_{\min}$ is not the smallest such integer, giving a contradiction.

It now remains to prove that B^*_τ is forward invariant under $\overline{S}^1_\tau$. If not, then for some $(U_0, \Delta t_0) \in X$ there exists some $(U_{n_0+1}, \Delta t_{n_0+1}) \in \overline{S}^{n_0+1}_\tau(U_0, \Delta t_0)$ such that $(U_{n_0+1}, \Delta t_{n_0+1}) \notin B^*_\tau$. Clearly, $t_{n_0+1} \geq T$ and so suppose first that $t_{n_0+1} \leq 3T$. Then by (3.20), $(U_{n_0+1}, \Delta t_{n_0+1}) \in X \subset B^*_\tau$ giving a contradiction. If $t_{n_0+1} > 3T$ then there exists some $p \geq 1$ such that $(U_p, \Delta t_p) \in \overline{S}^p_\tau(U_0, \Delta t_0)$ and $(U_{n_0+1}, \Delta t_{n_0+1}) \in \overline{S}^{n_0+1-p}_\tau(U_p, \Delta t_p)$ with $t_p \in [2T, 3T]$. Therefore by (3.20), $(U_p, \Delta t_p) \in X$ and $(U_{n_0+1}, \Delta t_{n_0+1}) \in B^*_\tau$.

Because X and each of its forward images under the closed relation $\overline{S}^1_\tau$ are closed, B^*_τ is also closed and hence compact. ◻

Since B^*_τ is compact and forward invariant, $\overline{S}^1_\tau$ restricted to B^*_τ is a closed relation and we now wish to prove that $\mathcal{A}_\tau = \omega(B^*_\tau)$ is an attractor. However, the set $\omega(B^*_\tau)$ may intersect the surface $\Delta t = D$ in which case there is no neighbourhood of $\mathcal{A}$ contained within $\mathbb{R}^m \times (0, D]$. Clearly we wish to allow this possibility but rather than modify the definition of an attractor we shall instead extend B^*_τ and the domain of $\overline{S}^1_\tau$. We extend $\overline{S}^1_\tau$ to $\mathbb{R}^m \times (0, 2D]$ by defining

$$\overline{S}^1_\tau(u, \Delta t) = \overline{S}^1_\tau(u, D) \quad \forall \Delta t \in (D, 2D].$$

We now define the sets

$$W = \{u \in \mathbb{R}^m : (u, D) \in B^*_\tau\}$$

and

$$B^{**}_\tau = B^*_\tau \cup (W \times (D, 2D]).$$

Then $\overline{S}^1_\tau$ restricted to B^{**}_τ is still a closed relation, B^{**}_τ is forward invariant and compact and $\omega(B^{**}_\tau) = \omega(B^*_\tau)$.

Theorem 3.4. *Let Assumption 1 hold and consider the approximation of (1.1) by an adaptive timestepping method satisfying conditions (C1)—(C6) and Assumption 2. Then, for all sufficiently small τ, $\overline{S}^1_\tau$ restricted to $\overline{B}^{**}_\tau$ is a closed relation with attractor*

$$\mathcal{A}_\tau = \omega(B^*_\tau) = \bigcap_{n \geq 0} \overline{\bigcup_{k \geq n} \overline{S}^k_\tau(B^*_\tau)}$$

and furthermore

$$\operatorname{dist}(\mathcal{P}_1(\mathcal{A}_\tau), \mathcal{A}) \to 0 \quad \text{as } \tau \to 0$$

Proof. We first prove that

$$\mathcal{P}_1(\overline{S}^n_\tau(B^*_\tau)) \subseteq \mathcal{N}(\mathcal{A}, \epsilon) \quad \forall n : t_n \geq T.$$

Consider any trajectory $(U_n, \Delta t_n)$, $n \geq 0$ and choose T, ϵ and τ as in the proof of Theorem 3.3. Then (3.19) and (3.20) hold and since $T > D$ there exists some integer $n^* \leq [3T/\Delta t_{\min} + 1]$ such that $t_{n^*} \in [T, 3T]$ with $U_{n^*} \in \mathcal{N}(\mathcal{A}, \epsilon)$ and $(U_{n^*}, \Delta t_{n^*}) \in X$. So (3.16) and (3.17) imply

$$U_n \in \mathcal{N}(\mathcal{A}, \epsilon) \subseteq B \; \forall n : t_n \in [T + t^*, 3T + t^*] \tag{3.21}$$

and hence $U_n \in \mathcal{N}(\mathcal{A}, \epsilon)\ \forall n : t_n \in [2T, 4T]$. Proceeding inductively completes the argument. Therefore

$$\mathcal{P}_1(\mathcal{A}_\tau) \subseteq \mathcal{N}(\mathcal{A}, \epsilon) \subset \mathcal{N}(\mathcal{A}, 2\epsilon) \subseteq B^*_\tau$$

giving $\mathrm{dist}(\mathcal{P}_1(\mathcal{A}_\tau, \mathcal{A})) \to 0$ as $\tau \to 0$ as required.

The set $\mathcal{A}_\tau$ is compact and invariant and so it is sufficient to prove that $\mathcal{A}_\tau$ lies in the interior of B^{**}_τ. Let $(u, \Delta t) \in \mathcal{A}_\tau$ with $\Delta t < D$. By (3.10) and Assumption 2, $\alpha_2 \Delta t_{\min} \leq \Delta t \leq h(u)/2$. Since $u \in \mathcal{N}(\mathcal{A}, \epsilon) \subset \mathrm{Int}(\mathcal{P}_1(X))$ and the function $h(u)$ is continuous, $(u, \Delta t)$ lies in the interior of $X \subseteq B^{**}_\tau$. If $\Delta t = D$ then by the continuity of $h(u)$ there exists some sufficiently small $\delta > 0$ such that

$$\mathcal{N}((u, \Delta t), \delta) \cap (\mathbb{R}^m \times (0, D]) \subset X \subset B^*_\tau.$$

Then by the construction of B^{**}_τ, $\mathcal{N}((u, \Delta t), \delta) \subset B^{**}_\tau$. This completes the proof.
$\square$

4 Finite-time convergence results

We now turn our attention from dynamical systems theory to numerical analysis. The results contained in this section are of independent interest since they extend the finite-time convergence results of [16, 10] to a much wider class of algorithms but they also allow us to state precisely the conditions under which Assumption 2 holds i.e. when the results of §3 apply.

We shall need to consider carefully the expansion in Δt of the local error estimate $E(u, \Delta t)$. We consider the most common case where $E(u, \Delta t)$ is derived via a pair of explicit Runge-Kutta methods $S^{(1)}(u, \Delta t)$ and $S^{(2)}(u, \Delta t)$ given by

$$\eta_i^{(k)} = u + \Delta t \sum_{j=1}^{i-1} a_{ij}^{(k)} f(\eta_j^{(k)}), \quad i = 1, \ldots, \nu \tag{4.1}$$

$$S^{(k)}(u, \Delta t) = u + \Delta t \sum_{j=1}^{\nu} b_j^{(k)} f(\eta_j^{(k)}). \tag{4.2}$$

The method $S^{(1)}(u, \Delta t)$ is the one used to advance the solution in (3.1)—(3.8). For each method, $k = 1, 2$, the ν stages $\eta_i^{(k)}$ are defined by the lower triangular matrix $A^{(k)}_{i,j} = a^{(k)}_{ij}$. The methods have orders p and $p-1$ (not necessarily respectively) where $S^{(k)}$ is of order r if for all $f \in C^{r+2}(\mathbb{R}^m, \mathbb{R}^m)$ and $u \in \mathbb{R}^m$

$$\lim_{\Delta t \to 0} \frac{\|S(\Delta t)u - S^{(k)}(u, \Delta t)\|}{\Delta t^{r+1}} < \infty$$

and for some $f \in C^{r+2}(\mathbb{R}^m, \mathbb{R}^m)$ and $u \in \mathbb{R}^m$

$$\lim_{\Delta t \to 0} \frac{\|S(\Delta t)u - S^{(k)}(u, \Delta t)\|}{\Delta t^{r+2}} = \infty.$$

In practice, $S^{(1)}$ and $S^{(2)}$ are *embedded*, that is, $A^{(1)} = A^{(2)}$. Thus both methods use the same stages and so the second method $S^{(2)}$ can be calculated very cheaply. We shall assume from now on that $S^{(1)}$ and $S^{(2)}$ are embedded and refer to them as a $(p-1,p)$ embedded explicit Runge-Kutta pair with ν stages.

For small Δt the difference between the two methods is an approximation to the actual error committed by the lower-order method. However, since the higher-order method has already been calculated, it is often used instead to advance the solution — this is known as extrapolation. The integer s from (C4) is therefore $s=p$ in extrapolation mode and $s=p-1$ in non-extrapolation mode. In addition, the error can either be controlled on an error-per-step (EPS) basis or error-per-unit-step basis (EPUS) where the allowable estimated error is scaled by the length of the timestep. Thus

$$E(u,\Delta t) = \|S^{(1)}(u,\Delta t) - S^{(2)}(u,\Delta t)\|/\Delta t^{\rho}$$

for some choice of norm $\|\bullet\|$, and where $\rho=0$ in EPS mode, $\rho=1$ in EPUS mode and the integer q in (C4) is given by $q=p-\rho$. This gives 4 different operating modes EPS, XEPS, EPUS and XEPUS, where X denotes extrapolation.

We now follow [16] and prove a convergence result for a general $(p-1,p)$ embedded explicit Runge-Kutta pair. With s,q defined as above, the local error estimate has the following form.

Lemma 4.1. *There exists functions $b_1(u)$ and $b_2(u,\Delta t)$ such that*

$$E(u,\Delta t) = \Delta t^{q}\|f(u)\|\ \|b_1(u)+\Delta t b_2(u,\Delta t)\|\,. \tag{4.3}$$

Furthermore, if $J\subset\mathbb{R}^m$ is compact then the following constants are finite

$$C_3 = \sup_{u\in J}\|b_1(u)\|,$$

$$C_4 = \sup_{(u,\Delta t)\in J\times(0,D]}\|b_2(u,\Delta t)\|.$$

Proof. See Lemma 2.4 of [16]. □

We define the following sets:

$$\Psi(\epsilon) := \{u\in\mathbb{R}^m : \|b_1(u)\|\le\epsilon\},\quad \Gamma(\delta) := \{u\in\mathbb{R}^m : \|f(u)\|\le\delta\}$$

and, given $J\subset\mathbb{R}^m$,

$$J_{\epsilon,\delta} = J\setminus(\Gamma(\delta)\cup\Psi(\epsilon)),\quad J_\epsilon = J\backslash\Psi(\epsilon).$$

We now use an inductive argument to prove an upper bound on a timestep sequence generated by (3.1)—(3.8) provided that the sequence U_n is bounded and remains away from the sets $\Psi(\epsilon)$ and $\Gamma(\delta)$ for some $\delta,\epsilon>0$.

Lemma 4.2. *Let $\{(U_n,\Delta t_n)\}_{n=0}^{N}$ be any sequence generated by (3.1)—(3.8) satisfying conditions (C1)—(C6) and $U_n\subseteq J_{\epsilon,\delta}$, $\forall n=0,\dots N$ for some compact set J and $\delta,\epsilon>0$. Then if*

$$\Delta t_0 \le \frac{\epsilon}{2C_4} \tag{4.4}$$

it follows that for all sufficiently small τ

$$\Delta t_n \le \frac{\epsilon}{4C_4}\quad \forall n>0 \tag{4.5}$$

and

$$\Delta t_n^q \leq \frac{2\sigma(t, U_n)}{\epsilon\delta} \quad \forall n \geq 0 \tag{4.6}$$

Proof. We first prove that if $u \in J_{\epsilon,\delta}$ then all $\Delta t \in (0, \frac{\epsilon}{2C_4}]$ satisfying $E(u, \Delta t) \leq \sigma(\tau, u)$ also satisfy $\Delta t^q \leq \frac{2\sigma(\tau,u)}{\epsilon\delta}$. We have

$$E(u, \Delta t) \leq \sigma(\tau, u) \text{ and } u \in J_{\epsilon,\delta} \Rightarrow \Delta t^q \|b_1(u) + \Delta t b_2(u, \Delta t)\| \leq \sigma(\tau, u)/\delta$$

and

$$\|b_1(u) + \Delta t b_2(u, \Delta t)\| \geq \|b_1(u)\| - \Delta t \|b_2(u, \Delta t)\| \geq \epsilon - C_4 \Delta t \geq \epsilon/2$$

giving $\Delta t^q \leq \frac{2\sigma(\tau,u)}{\epsilon\delta}$ as desired.

Note that for all $n \geq 1$,

$$\Delta t_n \leq \Delta t_n^{(1)}$$

and so it is sufficient to prove that $\Delta t_n^{(1)} \leq \frac{\epsilon}{4C_4}$. The result clearly holds for $n = 0$ and so we assume it holds for $n = 0, \ldots, p$ and prove it holds for $n = p+1$. If part (B) of the algorithm (3.1)—(3.8) was used to generate Δt_{p+1} then $U_{p+1} = U_p$, $\Delta t_{p+1} \leq \theta_4 \Delta t_p$ and so the result holds for $n = p + 1$. On the other hand, if (A) was used then by (C5)

$$\begin{aligned}
\Delta t_{p+1} &\leq \alpha_1 \left(\frac{\sigma(\tau, U_p)}{E(U_p, \Delta t_p)} \right)^{\frac{1}{q}} \Delta t_p \\
&\leq \alpha_1 \left(\frac{2\sigma(\tau, U_p)}{\epsilon\delta} \right)^{\frac{1}{q}} \\
&\leq \alpha_1 \left(\frac{2C_2\tau}{\epsilon\delta} \right)^{\frac{1}{q}} \\
&\leq \frac{\epsilon}{4C_4}
\end{aligned}$$

where the last inequality holds for all sufficiently small τ since $q \geq 1$. Thus (4.5) holds for $n = p + 1$ and since $U_{p+1} \in J_{\epsilon,\delta}$ so does (4.6). □

The upper bound (4.6) on the timestep sequence is the key step in proving a finite-time convergence result. Provided that the set

$$\bigcup_{t\in[0,T], U\in B} S(t)U$$

remains outside $\Gamma(2\delta)$ and $\Psi(2\epsilon)$ for some $\epsilon, \delta > 0$, then for sufficiently small τ the numerical sequence U_n can be bounded away from $\Gamma(\delta)$ and $\Psi(\epsilon)$. Then, some further elementary properties of Runge-Kutta approximations, together with a Gronwall argument, give a convergence result of the form (1.4). Since the argument is almost identical to that in [16] we do not reproduce it here but instead just state the convergence result for the special case of a forward invariant set B as in Assumption 1.

THEOREM 4.3. *Suppose that* $\mathcal{N}(B,d) \cap (\Gamma(\delta) \cup \Psi(\epsilon)) = \emptyset$ *for some* $\epsilon, \delta, d > 0$. *Then for some constant* $C_4 \geq 0$, *Assumption 2 holds with* $h(u) = \frac{\epsilon}{2C_4}, \gamma = \frac{s}{q}$ *and* $K = \mathcal{O}(1/(\delta\epsilon)^\gamma)$.

We postpone a discussion of the breakdown of the convergence results close to $\Psi(0)$ until §5 and instead concentrate on the failure close to the set of equilibria $\Gamma(0)$.

The above analysis fails to give uniform convergence results on neighbourhoods of equilibria because if a trajectory spends a sufficiently long time in such a neighbourhood then the timesteps increase to $\mathcal{O}(1)$ independently of τ. Then the higher-order term $b_2(u,\Delta t)$ in the expansion (4.3) is of the same order as the leading term $b_1(u,\Delta t)$ and cancellation may occur, resulting in an error estimate which is much smaller than the actual local error. This then leads to unacceptably large timesteps being taken with a corresponding degradation of the convergence results. A simple numerical example illustrating this failure of convergence can be found in [10].

In [10] an alternative upper bound on the timestep sequence was proved, that holds on neighbourhoods of equilibria, provided that the higher-order term $b_2(u,\Delta t)$ in (4.3) is $\mathcal{O}(\|f(u)\|^2)$. Furthermore the local error estimate has this form if the $(p-1,p)$ embedded explicit Runge-Kutta pair uses precisely p stages. In fact the minimum number of stages needed for an order p explicit Runge-Kutta method is p and this bound can be attained only for $p \leq 4$ (see [3]).

LEMMA 4.4. *Suppose that* $S^{(1)}(u,\Delta t)$ *and* $S^{(2)}(u,\Delta t)$ *are an embedded explicit* $(p-1,p)$ *Runge-Kutta pair with precisely* p *stages. Then*

$$E(u,\Delta t) = \Delta t^q \|f(u)\| \left\| \left(b_1(u) + \Delta t \|f(u)\| \tilde{b}_2(u,t) \right) \right\|. \tag{4.7}$$

Furthermore, if $J \subset \mathbb{R}^m$ *is compact then*

$$\tilde{C}_4 = \sup_{(u,\Delta t) \in J \times (0,D]} \|\tilde{b}_2(u,\Delta t)\| < \infty$$

PROOF. See [10]. □

We shall also need the following elementary result which holds for all Runge-Kutta methods $S^{(1)}$.

LEMMA 4.5. *For* $U_n \in J$, $\|f(U_{n+1})\| \leq \overline{K}\|f(U_n)\|$ *for some* $\overline{K}(J,D) > 0$.

PROOF. See [10]. □

The following Lemma, analagous to Lemma 4.2, gives an upper bound on the timestep sequence provided that U_n does not intersect J_ϵ.

LEMMA 4.6. *Let* $S^{(1)}(u,\Delta t)$ *and* $S^{(2)}(u,\Delta t)$ *be a* $(p-1,p)$ *embedded explicit Runge-Kutta pair with precisely* p *stages. Also let* $\{(U_j,\Delta t_j)\}_{n=0}^N$ *be any sequence generated by (3.1)—(3.8) satisfying conditions (C1)—(C6) with* $U_n \subseteq J_\epsilon$, $\forall n \in 0,\ldots,N$, *for some compact set* J *and* $\epsilon > 0$. *Then if*

$$\Delta t_0 \leq \frac{\epsilon}{2\tilde{C}_4\|f(U_0)\|} \tag{4.8}$$

it follows that for all sufficiently small τ

$$\Delta t_n \leq \frac{\epsilon}{4\tilde{C}_4\|f(U_n)\|} \quad \forall n > 0 \tag{4.9}$$

and

$$\Delta t_n^q \leq \frac{2\sigma(\tau, U_n)}{\epsilon \|f(U_n)\|}. \tag{4.10}$$

Proof. We first prove that if $u \in J_\epsilon$ then all $\Delta t \in (0, \frac{\epsilon}{2\tilde{C}_4 \|f(u)\|}]$ satisfying $E(u, \Delta t) \leq \sigma(\tau, u)$ also satisfy $\Delta t^q \leq \frac{2\sigma(t,u)}{\epsilon \|f(u)\|}$. We have

$$E(u, \Delta t) \leq \sigma(\tau, u) \text{ and } u \in J_{\epsilon,\delta} \Rightarrow \Delta t^q \left\| \left(b_1(u) + \Delta t \|f(u)\| \tilde{b}_2(u, \Delta t) \right) \right\| \leq \sigma(\tau, u)/\|f(u)\|$$

and

$$\left\| \left(b_1(u) + \Delta t \|f(u)\| \tilde{b}_2(u, \Delta t) \right) \right\| \geq \|b_1(u)\| - \Delta t \|f(u)\| \; \|\tilde{b}_2(u, \Delta t)\| \geq \epsilon - \tilde{C}_4 \Delta t \|f(u)\| \geq \epsilon/2$$

giving $\Delta t^q \leq \frac{2\sigma(t,u)}{\epsilon \|f(u)\|}$ as desired.

Since, for all n, $\Delta t_n \leq \Delta t_n^{(1)}$ it is sufficient to prove that $\Delta t_n^{(1)} \leq \frac{\epsilon}{4\tilde{C}_4 \|f(U_n\|}$ for all sufficiently small τ. The result holds for $n = 0$ and so we proceed inductively and assume it holds for $n = 0, \ldots, p$ and prove it holds for $n = p + 1$. If part (B) of the algorithm (3.1)—(3.8) was used to generate Δt_{p+1} then $U_{p+1} = U_p$, $\Delta t_{p+1} \leq \theta_4 \Delta t_p$ and so the result holds for $n = p + 1$. On the other hand, if (A) was used then by (C5)

$$\begin{aligned}
\Delta t_{p+1} &\leq \alpha_1 \left(\frac{\sigma(\tau, U_p)}{E(U_p, \Delta t_p)} \right)^{\frac{1}{q}} \Delta t_p \\
&\leq \alpha_1 \left(\frac{2\sigma(\tau, U_p)}{\epsilon \|f(U_p)\|} \right)^{\frac{1}{q}} \\
&\leq \alpha_1 \left(\frac{2C_2 \tau}{\epsilon \|f(U_p)\|} \right)^{\frac{1}{q}} \\
&\leq \alpha_1 \left(\frac{2C_2 \overline{K} \tau}{\epsilon \|f(U_{p+1})\|} \right)^{\frac{1}{q}} \quad \text{by Lemma 4.5} \\
&\leq \frac{\epsilon}{4\tilde{C}_4 \|f(U_{p+1})\|}
\end{aligned}$$

where the last inequality holds for sufficiently small τ since $q \geq 1$ and $\|f(u)\|$ is bounded on J_ϵ. ☐

The upper bound on the timestep sequence (4.6) therefore holds on some neighbourhood of an equilibrium point $\overline{u}$ provided that $\overline{u} \notin \Psi(0)$. Sufficient conditions for this to hold are that $\overline{u}$ is non-singular (i.e. the Jacobian $df(\overline{u})$ is invertible) and the $p - 1$ order method does not increase to order p on linear problems (this is a very weak requirement). Then, a similar argument to that in [10] gives the following convergence result on a neighbourhood of the set B from Assumption 1.

Theorem 4.7. *Let $S^{(1)}$ and $S^{(2)}$ be a $(p-1, p)$ embedded explicit Runge-Kutta pair with precisely p stages and assume that the lower-order method does not increase its order on linear problems. Suppose that $\mathcal{N}(B, d) \cap \Psi(\epsilon) = \emptyset$ for* *some $\epsilon, d > 0$ and also that all equilibria in $\mathcal{N}(B, d)$ are non-singular. Then for some constant $\tilde{C}_4 \geq 0$, Assumption 2 holds with $h(u) = \min(3D, \frac{\epsilon}{2C_4\|f(u)\|})$ and $K = \mathcal{O}((\frac{1}{\epsilon})^\gamma)$ where $\gamma = s/q$ for EPS, XEPS and EPUS modes and $\gamma = 1$ for XEPUS mode.*

5 Discussion

Equilibria, and their associated invariant manifolds, play an important role in dynamical systems theory. Theorems 4.3 and 4.7 suggest that if adaptive algorithms are used for computations involving these sets, then the local error estimate should be of the form (4.7). In particular, the existence and upper semicontinuity results for numerical attractors from §3 will only hold if this is the case.

As was mentioned in §4, such local error estimates can be derived using a $(p-1, p)$ embedded explicit Runge-Kutta pair with precisely p stages. Such methods only exist for $p \leq 4$ although higher-order Runge-Kutta pairs with suitable error estimates may also exist. Whether or not this is the case, an algorithm that uses an arbitrary high-order pair ($p > 4$) away from equilibria and switches to a $(p-1, p)$ pair with p stages when $\|f(u)\|$ is small should enjoy both the efficiency benefits of the main pair and the superior convergence properties of the second pair. Of course the class of algorithms (2.1)—(2.8) would have to be extended further to include such a modification and their description as dynamical systems would become more complicated.

Recall that the finite-time convergence results, Theorems 4.3 and 4.7, involve the quantity ϵ which is related to the distance between the sets B and $\Psi(0)$. For small ϵ the error constant K is large and if B and $\Psi(0)$ intersect then $\epsilon = 0$, the convergence results break down, and Assumption 2 may not hold. Since B can be any forward invariant neighbourhood of the attractor $\mathcal{A}$, it is the distance between the sets $\mathcal{A}$ and $\Psi(0)$ that is of interest and the following heuristic argument (motivated by [16, 10]) estimates the probability that ϵ is small.

Since $b_1(u) : \mathbb{R}^m \to \mathbb{R}^m$, the set $\Psi(0)$ of zeros of $b_1(u)$ will typically consist of isolated points. Furthermore, $b_1(u)$ depends upon both the vector field f and the coefficients used in the definition of the Runge-Kutta pair $S^{(1)}$ and $S^{(2)}$. These Runge-Kutta pairs usually exist in parametrised classes and so, by placing a Lebesgue measure on the free parameters, we can treat the zeros of $b_1(u)$ as random variables for a given vector field f. We can now estimate the probability that a point, chosen at random from a measure which is absolutely continuous with respect to Lebesgue measure, lies within $\mathcal{N}(\mathcal{A}, \epsilon)$ for small ϵ. We define $\underline{\dim}_B \mathcal{A}$ to be the lower box-counting dimension of the attractor $\mathcal{A}$ and $\mathrm{vol}(\mathcal{N}(\mathcal{A}, \epsilon))$ to be the m-dimensional volume of the set $\mathcal{N}(\mathcal{A}, \epsilon)$. Then (see [4])

$$\overline{\lim}_{\epsilon \to 0} \frac{\log \mathrm{vol}(\mathcal{N}(\mathcal{A}, \epsilon))}{\log \epsilon} = m - \underline{\dim}_B \mathcal{A}.$$

Therefore, for sufficiently small ϵ, there exists some constant C such that

$$\mathrm{vol}(\mathcal{N}(\mathcal{A}, \epsilon)) \leq C \epsilon^{m - \underline{\dim}_B \mathcal{A}}.$$

For many attractors of dissipative ODEs the lower box-counting dimension can be proved to be strictly smaller than the dimension of the phase space and hence, under the above heuristic argument, the probability that the attractor $\mathcal{A}$ intersects the set $\Psi(\epsilon)$ tends to 0 as $\epsilon \to 0$. If $\mathcal{A}$ does not intersect $\Psi(0)$ there exists $d > 0$ such that $\mathcal{N}(B, d)$ does not intersect $\Psi(0)$ for some forward invariant set B attracted by $\mathcal{A}$.

Finally, we note that in [11] another class of modified algorithms is suggested. These algorithms use two different embedded explicit Runge-Kutta pairs, one on odd-numbered steps and the other on even-numbered steps. Then for generic vector-fields f, the sets $\Psi(0)$ for each pair will not intersect and it is proved that this guarantees convergence everywhere in the phase space as $\tau \to 0$. Again, this modified algorithm does not fit into the class of algorithms (2.1)—(2.8) but can also be formulated as a dynamical system, albeit on a more complicated phase space. There are other possible approaches to improving the convergence behaviour of adaptive algorithms, including the possibility of adding extra criteria that must be satisfied at each timestep. Such modifications may also be important for proving stability results for classes of ODEs such as gradient or dissipative systems.

Acknowledgement.

The author thanks Andrew Stuart for his many helpful comments.

REFERENCES

1. *MATLAB. The Math Works Inc., Natick, MA.*
2. E. Akin, *The General Topology of Dynamical Systems*, American Mathematical Society, 1993.
3. J. Butcher, *The numerical analysis of ordinary differential equations*, Wiley, New York, 1992.
4. K. Falconer, *Fractal Geometry: Mathematical Foundations and Applications*, Wiley, 1990.
5. K. Gustafsson, M. Lundh, and G. Soderlind, *A PI-stepsize control for the numerical solution of ordinary differential equations*, BIT, 28 (1988), pp. 270–287.
6. J. Hale, X. Lin, and G. Raugel, *Upper semicontinuity of attractors for approximations of semigroups and partial differential equations*, Math. Comp., 50 (1988), pp. 89–123.
7. G. Hall, *Equilibrium states of Runge-Kutta schemes*, ACM Trans. Math. Software, 11 (1985), pp. 289–301.
8. A. Humphries, *Upper semicontinuity of attractors under approximation by variable time-stepping numerical methods.* Unpublished.
9. P. Kloeden and J. Lorenz, *Stable attracting sets in dynamical systems and their one-step discretizations*, SIAM J. Num. Anal., 23 (1986), pp. 986–995.
10. H. Lamba and A. Stuart, *Convergence results for the MATLAB ode23 routine*, BIT, 38 (1998), pp. 751–780.
11. ———, *Convergence proofs for numerical IVP software*, in Dynamics of Algorithms, vol. 118 of IMA Volumes in Mathematics and its Applications, 1999, pp. 107–127.
12. L. Shampine, *Numerical Solution of Ordinary Differential Equations*, Chapman and Hall, New York, 1994.

13. H. Stetter, *Considerations concerning a theory for ODE solvers.* Numerical Treatment of Differential Equations. Lecture Notes in Mathematics 631, Springer-Verlag, Berlin, 1976.
14. ———, *Tolerance proportionality in ODE codes.* Proc. second Conf. on Numerical Treatment of Ordinary Differential Equations, Humboldt University, 1980.
15. D. Stoffer and K. Nipp, *Invariant curves for variable step size integrators*, BIT, 31 (1991), pp. 169–180.
16. A. Stuart, *Probabilistic and deterministic convergence proofs for software for initial value problems*, Numerical Algorithms, 14 (1997), pp. 227–260.
17. A. Stuart and A. Humphries, *Dynamical systems and numerical analysis*, CUP, 1996.