BIT 34 (1994), 88–98.

RUNGE-KUTTA STABILITY ON A FLOQUET PROBLEM

DESMOND J. HIGHAM*

Department of Mathematics and Computer Science
University of Dundee
Dundee, DD1 4HN

Abstract.

This work examines the stability of explicit Runge-Kutta methods applied to a certain linear ordinary differential equation with periodic coefficients. On this problem naïve use of the eigenvalues of the Jacobian results in misleading conclusions about stable behaviour. It is shown, however, that a valid analogue of the classical absolute stability theory can be developed. Further, using a suitable generalisation of the equilibrium theory of Hall [ACM Trans. on Math. Soft. 11 (1985), pp. 289–301], accurate predictions are made about the performance of modern, adaptive algorithms.

Key words: Runge-Kutta, absolute stability, Floquet, equilibrium, steady state.

AMS(MOS) Subject classification: 65L05.

1. Introduction.

Numerical analysts have built up a vast body of knowledge around the initial value ODE

$$y'(t) = Ay(t), \quad A \in \mathbb{R}^{N \times N} \text{ constant}, \quad y(t_0) \text{ given}. \tag{1.1}$$

The system (1.1) is sufficiently simple to allow precise statements to be made about the behaviour of numerical methods. Further, and equally importantly, (1.1) often provides a reasonable model for more general ODEs. Stability results have, of course, also been obtained for broader classes of ODEs, but here the properties under investigation, such as B-stability or algebraic stability [1], tend to be extremely demanding. Hence, although many powerful results have been established in the realm of non-linear stability, relatively little is known about the stability properties of standard explicit methods on problems outside the class (1.1).

This work takes a small step away from (1.1) and considers the absolute stability of explicit Runge-Kutta (ERK) methods on a particular class of linear test problems with periodic coefficients. We ask

* Supported by the University of Dundee Research Initiatives Fund.
Received January 1993. Revised September 1993.

1. For what range of stepsizes will an explicit Runge-Kutta formula behave acceptably?
2. Will an adaptive algorithm compute an acceptable solution?

Question 1 is considered in the next section. After introducing the test problem, we show that an analogue of the classical stability region can be defined. The stability restriction is different to, and not necessarily more stringent than that which arises when the classical definition is applied to the 'frozen' model (1.1). In section 3, we address question 2 by developing a suitable generalisation of the equilibrium theory of Hall [2]. This allows an accurate prediction of the long term behaviour of adaptive Runge-Kutta algorithms. Once again, the results are qualitatively different from those that would arise from the use of (1.1).

2. Absolute Stability.

Kreiss [7, page 24] introduces the initial value problem

$$y'(t) = \varepsilon^{-1} U^T(t) Z U(t) y(t) \equiv A(t) y(t), \quad y(t_0) = y_0, \tag{2.1}$$

where

$$U(t) = \begin{bmatrix} \cos(\alpha t) & -\sin(\alpha t) \\ \sin(\alpha t) & \cos(\alpha t) \end{bmatrix}, \quad Z = \begin{bmatrix} -1 & \eta \\ 0 & -1 \end{bmatrix}.$$

Here ε, α, $\eta \in \mathbb{R}$ are parameters that define a particular ODE. Note that $U(t)$ is orthogonal for any t, and that premultiplying a vector by $U(t)$ corresponds to an anticlockwise rotation through αt radians. (We remark that there is a minor error in [7]. This has been fixed by replacing $U(t)$ in [7] by $U^T(t)$.) It follows from the orthogonality of $U(t)$ that the time-dependent Jacobian $A(t)$ is always similar to $\varepsilon^{-1} Z$, and hence the eigenvalues of $A(t)$ are $-\varepsilon^{-1}$. Applying the transformation $v(t) = U(t) y(t)$ we find

$$v'(t) = \begin{bmatrix} -\varepsilon^{-1} & \eta\varepsilon^{-1} - \alpha \\ \alpha & -\varepsilon^{-1} \end{bmatrix} v(t). \tag{2.2}$$

Hence $v(t)$ solves a constant coefficient, linear system whose Jacobian has eigenvalues $-\varepsilon^{-1}(1 \pm \sqrt{\alpha\varepsilon(\eta - \alpha\varepsilon)})$. Since $\|v(t)\|_2 = \|y(t)\|_2$, it follows that for any initial value y_0 the long term growth or decay of $y(t)$ is governed by the real part of these eigenvalues. Kreiss [7] used this example to illustrate that an ODE could have an increasing solution despite the fact that the Jacobian $A(t)$ has eigenvalues with negative real parts for any 'frozen' value of t. Here, we are interested in the case where $\Re\{-\varepsilon^{-1}(1 \pm \sqrt{\alpha\varepsilon(\eta - \alpha\varepsilon)})\} < 0$, since we wish to examine the absolute stability of ERK methods applied to (2.1). Our interest in problems of the form (2.1) arose out of related work on stiff ODEs [5]. We mention that (2.1) is a special case of a Floquet

problem [6, page 231]; that is, a linear system with periodic coefficients. The eigenvalues of the constant Jacobian in (2.2) are the characteristic exponents of (2.1) and the long term behaviour of the solution follows from standard Floquet theory. We also mention that Sand [9] has analysed the stability of the θ-method on (2.1).

As an example of an ERK formula, we consider a two-stage, second order scheme. Applied to the general system $y'(t) = f(t, y(t))$ the formula advances from $y_n \approx y(t_n)$ to $y_{n+1} \approx y(t_n + h_n)$ according to

$$\begin{aligned} k_1 &= f(t_n, y_n), \\ k_2 &= f(t_n + h_n, y_n + h_n k_1), \\ y_{n+1} &= y_n + \frac{h_n}{2}[k_1 + k_2]. \end{aligned} \tag{2.3}$$

For the problem (2.1) we find

$$y_{n+1} = \left\{ I + \frac{h_n}{2}[A(t_n) + A(t_{n+1})] + \frac{h_n^2}{2} A(t_{n+1})A(t_n) \right\} y_n.$$

Using the relation $U(t_{n+1}) = U(h_n)U(t_n)$, this may be re-arranged as

$$U(t_{n+1})y_{n+1} = \left\{ U(h_n) + \frac{h_n}{2\varepsilon}[U(h_n)Z + ZU(h_n)] + \frac{h_n^2}{2\varepsilon^2} ZU(h_n)Z \right\} U(t_n)y_n,$$

which we write as

$$U(t_{n+1})y_{n+1} = S(h_n, \varepsilon, \alpha, \eta)U(t_n)y_n. \tag{2.4}$$

The key point is that the matrix $S(h_n, \varepsilon, \alpha, \eta)$, which we refer to as the *absolute stability matrix*, does not depend upon the location of the meshpoint t_n. In particular, if $h_n \equiv h$ (constant) then we have

$$\begin{aligned} U(t_{n+1})y_{n+1} &= S(h, \varepsilon, \alpha, \eta)U(t_n)y_n \\ &= S(h, \varepsilon, \alpha, \eta)^2 U(t_{n-1})y_{n-1} \\ &\vdots \\ &= S(h, \varepsilon, \alpha, \eta)^{n+1} U(t_0)y_0. \end{aligned} \tag{2.5}$$

Hence the long term growth or decay of $\|y_n\|_2$ is governed by $\rho(S(h, \varepsilon, \alpha, \eta))$, where $\rho(\cdot)$ denotes the spectral radius.

Moving from (2.3) to the general s-stage ERK formula

$$\begin{aligned} k_1 &= f(t_n, y_n), \\ k_i &= f\left(t_n + c_i h_n, y_n + h_n \sum_{j=1}^{i-1} a_{ij} k_j \right), \quad 2 \le i \le s, \\ y_{n+1} &= y_n + h_n \sum_{i=1}^{s} b_i k_i, \end{aligned} \tag{2.6}$$

the following result shows that the form of the recurrence (2.4) is maintained.

LEMMA 2.1. *Applying the formula (2.6) to (2.1) produces*

$$U(t_{n+1})y_{n+1} = S(h_n, \varepsilon, \alpha, \eta)U(t_n)y_n. \tag{2.7}$$

Here, the absolute stability matrix $S(h_n, \varepsilon, \alpha, \eta)$ depends on the coefficients of the Runge-Kutta formula, but is independent of t_n.

PROOF. We first prove by induction that k_j in (2.6) has the form

$$k_j = U^T(t_n)K_j(h_n, \varepsilon, \alpha, \eta)U(t_n)y_n, \tag{2.8}$$

where the matrix $K_j(h_n, \varepsilon, \alpha, \eta)$ depends upon the Runge-Kutta coefficients, but not upon t_n. It is trivial to check that (2.8) holds for $j = 1$. Now suppose that (2.8) holds for all $1 \le j \le i - 1$. Then

$$k_i = \varepsilon^{-1}U^T(t_n + c_ih_n)ZU(t_n + c_ih_n)\left\{y_n + h_n\sum_{j=1}^{i-1} a_{ij}U^T(t_n)K_j(h_n, \varepsilon, \alpha, \eta)U(t_n)y_n\right\}$$

$$= U^T(t_n)U^T(c_ih_n)\varepsilon^{-1}Z\left\{U(c_ih_n) + h_n\sum_{j=1}^{i-1} a_{ij}U(c_ih_n)K_j(h_n, \varepsilon, \alpha, \eta)\right\}U(t_n)y_n,$$

which has the required form. So, by induction, (2.8) is true for $1 \le j \le s$. Hence,

$$y_{n+1} = y_n + h_n\sum_{i=1}^{s} b_iU^T(t_n)K_i(h_n, \varepsilon, \alpha, \eta)U(t_n)y_n,$$

$$= U(t_n + h_n)^T\left\{U(h_n) + h_n\sum_{i=1}^{s} b_iU(h_n)K_i(h_n, \varepsilon, \alpha, \eta)\right\}U(t_n)y_n.$$

Multiplying on the left by $U(t_n + h_n)$ completes the proof. ■

This result leads naturally to the following definition.

DEFINITION 2.1. Given values of $\varepsilon, \alpha, \eta$ satisfying $\Re\{-\varepsilon^{-1}(1 \pm \sqrt{\alpha\varepsilon(\eta - \alpha\varepsilon)})\} < 0$, the method (2.6) is said to be *absolutely stable* for a particular stepsize h if the absolute stability matrix satisfies $\rho(S(h, \varepsilon, \alpha, \eta)) < 1$. The largest interval $(0, h_S)$ such that the method is absolutely stable for all $h \in (0, h_S)$ is called the *absolute stability interval.* ■

It is clear from (2.7) and (2.5) that asking for absolute stability is equivalent to requiring that the numerical solution, generated from a fixed stepsize, mimics the long term decay of the true solution.

Our stability criterion is different from the one that arises from the analysis of a "frozen" version of (2.1). For example, the two-stage method (2.3) has classical stability polynomial $p(z) = 1 + z + z^2/2$. Since $A(t)$ always has eigenvalues $-\varepsilon^{-1}$,

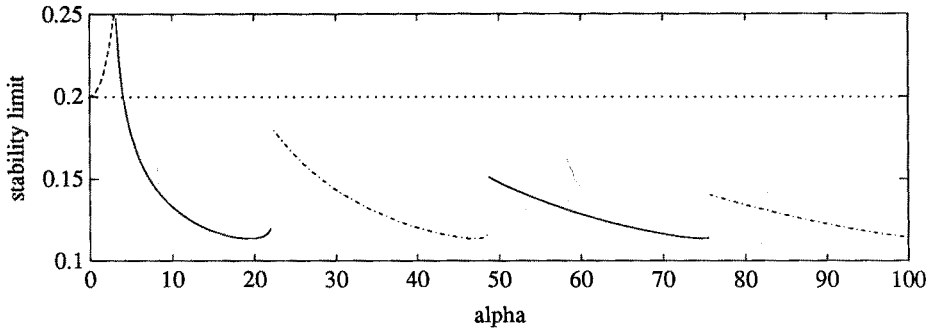

Figure 2.1. h_S versus α.

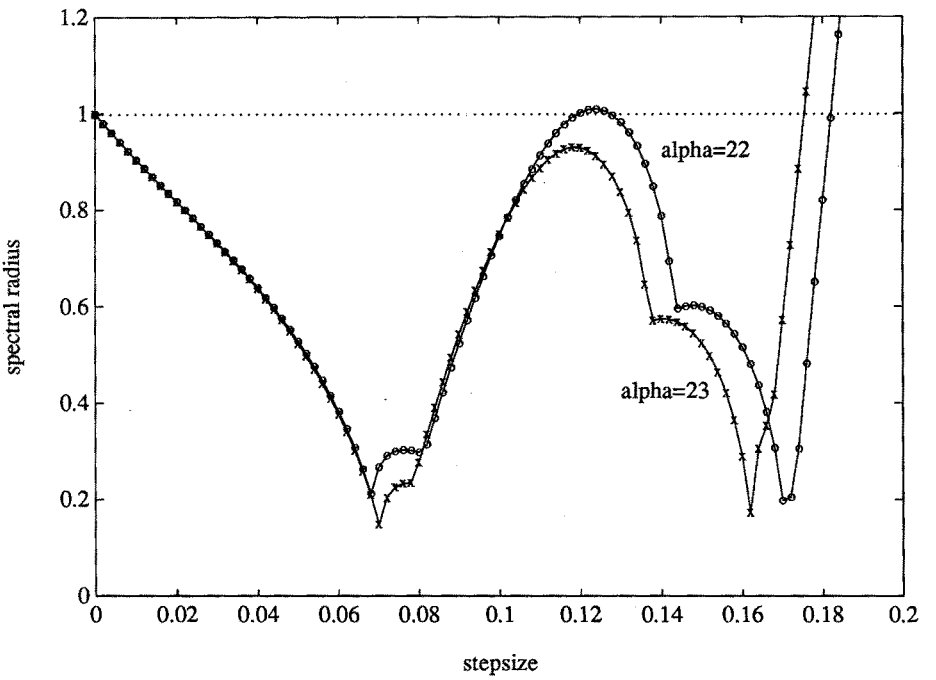

Figure 2.2 $\rho(S(h, \varepsilon, \alpha, \eta))$ versus h.

the classical $|p(-h/\varepsilon)| < 1$ stability restriction suggests a stability interval $(0, 2\varepsilon)$, which is independent of α and η. By contrast, the "correct" definition above produces a stability interval that depends markedly upon α and η. As an example, we fix $\varepsilon = .1$ and $\eta = 1.5$. Figure 2.1 plots the value of h_S from Definition 2.1 as α varies between

0 and 100. The h_S values were computed with a bisection algorithm. (An explanation of the line-types used in the plot will be given in the next section.) We see that h_S generally falls below the frozen limit of .2 (which is marked by a dotted line), but is greater than .2 for small values of α. In the latter case, the frozen Jacobian approach actually leads to an over-conservative stability definition. Also, as the figure suggests, h_S is a discontinuous function of α. To explain this, Figure 2.2 plots $\rho(S(h, \varepsilon, \alpha, \eta))$ versus h for $\alpha = 22$ and $\alpha = 23$. For $\alpha = 23$ the initial hump in $\rho(S(h, \varepsilon, \alpha, \eta))$ stays below 1 and hence h_S is determined by the later growth. However, at $\alpha = 22$ the initial hump dominates. This illustrates how a small change in the problem (2.1) can have a dramatic effect on numerical stability.

3. Equilibrium Theory.

In the previous section we derived a stability condition for a fixed stepsize implementation of an ERK formula. Here, we look at the performance of modern, variable-stepsize algorithms. Our approach is motivated by the work of Hall [2], [1, page 26], and the results below can be regarded as a generalization of [2] from constant coefficients to periodic coefficients.

Standard error control techniques for Runge-Kutta methods are based on local error estimation. After applying the formula (2.6), an error estimation. After applying the formula (2.6), an error-per-step estimate has the form

$$\text{est}_{n+1} = \|h_n \sum_{i=1}^{s} e_i k_i\|, \tag{3.1}$$

using certain coefficients $\{e_i\}$. Usually (3.1) is the norm of the difference between two Runge-Kutta approximations, and hence can be regarded as an estimate of the local error in the lower order approximation. (The error-per-unit-step analogue of (3.1), where $\text{est}_{n+1} = \|\sum_{i=1}^{s} e_i k_i\|$, is also covered by our analysis – equation (3.4) below holds with a different function $E(h_n, \varepsilon, \alpha, \eta)$.) A step is deemed acceptable if $\text{est}_{n+1} \leq \text{TOL}$, where TOL is a tolerance parameter supplied by the user. After an accepted step, the stepsize for the next step is computed from

$$h_{n+1} = \left(\frac{\theta \text{TOL}}{\text{est}_{n+1}}\right)^{1/q} h_n. \tag{3.2}$$

Here, the safety factor $\theta \in (0, 1)$ is included in an attempt to avoid unnecessary rejected steps, and the integer q comes from the expansion $\text{est}_{n+1} = O(h_n^q)$. Although the treatment of rejected steps is an important practical issue, we do not need to consider such details in our analysis.

We now show that the error estimate for (2.1) satisfies a similar recurrence to the numerical solution.

LEMMA 3.1. *Applying the formulae (2.6) and (3.1) to (2.1) produces*

$$U(t_{n+1})h_n \sum_{i=1}^{s} e_i k_i = E(h_n, \varepsilon, \alpha, \eta)U(t_n)y_n, \tag{3.3}$$

where the error matrix $E(h_n, \varepsilon, \alpha, \eta)$ *depends on the coefficients of the Runge-Kutta formula, but is independent of* t_n, *and hence, using* $\|\cdot\|_2$ *in (3.1)*

$$\text{est}_{n+1} = \|E(h_n, \varepsilon, \alpha, \eta)U(t_n)y_n\|_2. \tag{3.4}$$

PROOF. A proof follows almost immediately from the proof of Lemma 2.1. ■

Henceforth we will assume that $\|\cdot\|_2$ is used in (3.1) so that (3.4) holds.

We saw in section 2 that a stepsize h_S exists that lies on the boundary between stability and instability. We show below that if $S(h_S, \varepsilon, \alpha, \eta)$ has real eigenvalues then an *equilibrium state* exists where the algorithm continues with constant stepsize and constant solution norm.

THEOREM 3.1. *Suppose* $\|\cdot\|_2$ *is used in (3.1). If the matrix* $S(h_S, \varepsilon, \alpha, \eta)$ *has real eigenvalues then let* v, *with* $\|v\|_2 = 1$, *be a normalized eigenvector corresponding to an eigenvalue* μ *with modulus one. Further, let* $h_n = h_S$ *and let* y_n *be specified by* $U(t_n)y_n = \beta v$, *where* $|\beta| = \theta\text{TOL}/\|E(h_S, \varepsilon, \alpha, \eta)v\|_2$. *Then the algorithm continues with constant stepsize,* $h_n \equiv h_S$, *and constant solution norm,* $\|y_n\|_2 \equiv \theta\text{TOL}/\|E(h_S, \varepsilon, \alpha, \eta)v\|_2$. *The transformed numerical solution* $U(t_n)y_n$ *either remains constant for all n or oscillates in sign, depending upon whether* $\mu = +1$ *or* $\mu = -1$, *respectively.*

PROOF. Suppose first that $\mu = 1$. Then using h_n and y_n as defined above, we find, from (2.7),

$$U(t_{n+1})y_{n+1} = S(h_S, \varepsilon, \alpha, \eta)\beta v = \beta v = U(t_n)y_n,$$

and, from (3.2) and (3.4),

$$h_{n+1} = \left(\frac{\theta\text{TOL}}{|\beta|\,\|E(h_n, \varepsilon, \alpha, \eta)v\|_2}\right)^{1/q} h_S = h_S.$$

This gives the required result. If $\mu = -1$ then the recurrence changes to $U(t_{n+1})y_{n+1} = -U(t_n)y_n$ and $h_{n+1} = h_S$. ■

It is reasonable to expect a good error control scheme to eventually select stepsizes at roughly the level of h_S – smaller stepsizes would be inefficient and larger stepsizes would be unstable. Also, from (2.7), a power method-type argument suggests that $U(t_n)y_n$ should line up along a dominant eigenvector of $S(h_S, \varepsilon, \alpha, \eta)$. However, to see whether an exact equilibrium state will arise in practice, one must investigate its stability with respect to small perturbations. Regarding the adaptive Runge-Kutta algorithm applied to (2.1) as a nonlinear map

$$F\left(\begin{bmatrix}(U(t_n)y_n)_1\\(U(t_n)y_n)_2\\h_n\end{bmatrix}\right) = \begin{bmatrix}(U(t_{n+1})y_{n+1})_1\\(U(t_{n+1})y_{n+1})_2\\h_{n+1}\end{bmatrix}, \tag{3.5}$$

Theorem 3.1 identifies a fixed point $p_1 \in \mathbb{R}^3$ such that $F(p_1) = p_1$ in the case $\mu = 1$, and a period two point $p_2 \in \mathbb{R}^3$ such that $F(F(p_2)) = p_2$ in the case $\mu = -1$. Letting J denote the Jacobian of F, these solutions are stable to first order if and only if $\rho(J(p_1)) < 1$ and $\rho(J(p_2)J(F(p_2))) < 1$, respectively. Although we have no closed-form expression for these spectral radii, they are clearly computable.

At this point it is worth mentioning how these results compare with the original equilibrium results of Hall. For a constant coefficient, linear system $y'(t) = Ay(t)$, where A has a real dominant eigenvalue, Hall [2] demonstrated the existence of an equilibrium state. The state has h_n constant on the absolute stability boundary and y_n constant or oscillating in sign (depending on the sign of the dominant eigenvalue). Further, Hall showed that the stability of the equilibrium state depends only on the Runge-Kutta algorithm – a simple algebraic condition determines the equilibrium stability for all ODEs of that form. The equilibrium state that we have derived for problem (2.1) does not share this feature. Indeed, we saw in section 2 that the stability interval $(0, h_S)$ is not simply a function of the eigenvalues $-\varepsilon^{-1}$ of $A(t)$, but also depends nonlinearly upon α. Hence, as we will see below, for a particular algorithm the precise form and overall stability of the equilibrium state can change as we alter the paremeters in (2.1). We also mention that the results in [2] have been extended to the case where the constant Jacobian A is normal and has complex dominant eigenvalues [3, 4]. An analogous extension of Theorem 3.1 to complex eigenvalues is not possible. One reason for this is that the stability matrix $S(h_S, \varepsilon, \alpha, \eta)$ and the error matrix $E(h_S, \varepsilon, \alpha, \eta)$ no longer share the same eigenvectors. However, it is possible to show that the equilibrium stability is independent of TOL.

THEOREM 3.2. *The first order stability of the equilibrium state identified in Theorem 3.1 is independent of the tolerance parameter* TOL.

PROOF. Suppressing dependence upon the parameters ε, α and η, it follows from (2.7), (3.2) and (3.4) that the map F in (3.5) can be written

$$F\left(\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}\right)$$

$$= \begin{bmatrix} S_{11}(x_3)x_1 + S_{12}(x_3)x_2 \\ S_{21}(x_3)x_1 + S_{22}(x_3)x_2 \\ x_3[\theta \mathrm{TOL}/\sqrt{(E_{11}(x_3)x_1 + E_{12}(x_3)x_2)^2 + (E_{21}(x_3)x_1 + E_{22}(x_3)x_2)^2}]^{1/q} \end{bmatrix}.$$

Hence we have, for example,

$$\frac{\partial F_1}{\partial x_1} = S_{11}(x_3), \quad \frac{\partial F_1}{\partial x_2} = S_{12}(x_3), \quad \frac{\partial F_1}{\partial x_3} = S'_{11}(x_3)x_1 + S'_{12}(x_3)x_2.$$

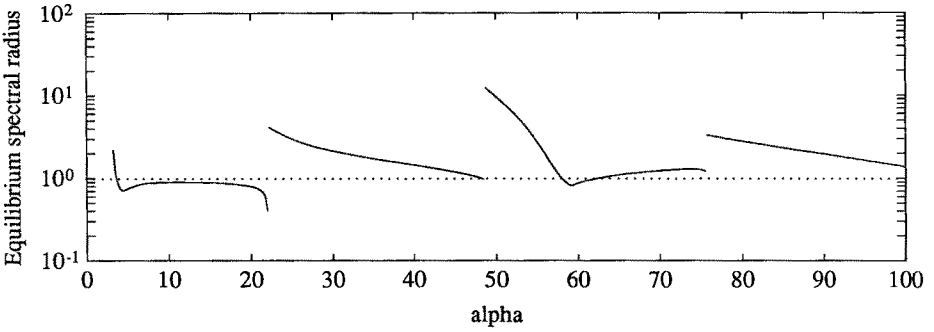

Figure 3.1. Equilibrium stability versus α.

We are interested in the value of the Jacobian at the equilibrium point defined in Theorem 3.1; that is, when $[x_1, x_2]^T = \beta v$ and $x_3 = h_S$. Note that h_S is independent of TOL and βv depends linearly upon TOL (through β). It follows that the corresponding values of $\partial F_1/\partial x_1$ and $\partial F_1/\partial x_2$ are independent of TOL whilst $\partial F_1/\partial x_3$ is linear in TOL. Continuing in this manner we find the following dependencies for the Jacobian matrix

$$\begin{bmatrix} \text{indep.} & \text{indep.} & \text{TOL} \\ \text{indep.} & \text{indep.} & \text{TOL} \\ \text{TOL}^{-1} & \text{TOL}^{-1} & \text{indep.} \end{bmatrix}.$$

Hence, letting $D = \text{diag}(1, 1, \text{TOL})$, the similarity transformations $DJ(p_1)D^{-1}$ and $DJ(p_2)D^{-1}\,DJ(F(p_2))D^{-1}$ show that the relevant Jacobian has eigenvalues that are independent of TOL. ■

In Figure 2.1, the line-types used to plot h_S are chosen according to the nature of the dominant eigenvalue of $S(h_S, \varepsilon, \alpha, \eta)$. Real negative, real positive, and complex eigenvalues correspond to solid, dash-dotted, and dashed lines, respectively. For those values of α where the eigenvalues are real, Figure 3.1 plots the size of the relevant spectral radius, $\rho(J(p_1))$ or $\rho(J(p_2)J(F(p_2)))$. Here, we have used an error estimate (3.1) that comes from differencing second and third order ERK approximations; that is, in addition to (2.3),

$$k_3 = f(t_n + h_n/2, y_n + h_n(k_1 + k_2)/4) \quad \text{and} \quad e = [1/3, 1/3, -2/3]^T,$$

with $q = 3$ in (3.2). Finite differences were used to approximate the Jacobian values. We see in Figure 3.1 that stable period two equilibrium states exist for two ranges of α; these are $4 \leq \approx \alpha \leq \approx 22$ and $58 \leq \approx \alpha \leq \approx 62$. For $0 < \alpha \leq \approx 4$ the eigenvalues of the stability matrix are complex, and for all other positive α values the equilibrium

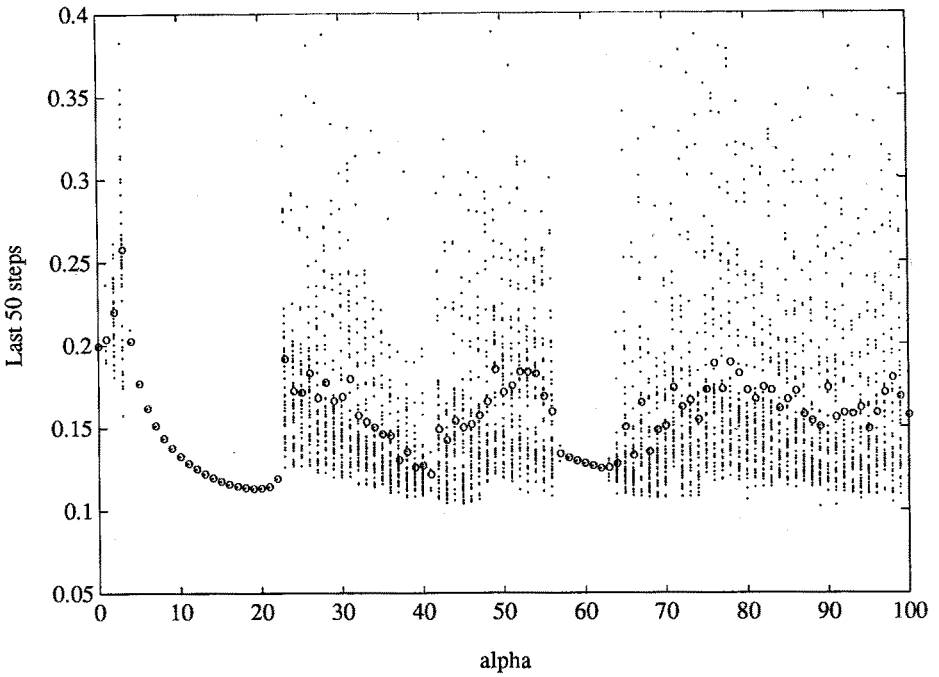

Figure 3.2 Last 50 stepsizes versus α.

state is unstable. For $\alpha = 0$, (2.1) becomes a constant coefficient problem and the original analysis of [2] shows that a stable fixed point equilibrium exists.

To illustrate the applicability of this theory, we present some numerical results computed with Matlab [8]. We used an adaptive Runge-Kutta algorithm based on Matlab's built-in ode23.m program, which implements a second and third order pair of formulas in a standard way. In order to make the results compatible with our Figures 2.1 and 3.1 we made the following minor changes to ode23.m.

- The second order formula (rather than the third order formula) was used to advance the solution.
- The two-norm was used to measure the error estimate, rather than the max-norm, so that (3.4) is valid.
- The minimum allowable stepsize was reduced in order to allow the initial fast transients to be tracked.

Fixing $\varepsilon = .1$ and $\eta = 1.5$, for each integer value of $\alpha \in [0, 100]$ we solved (2.1) over the range $t \in [0, 100]$, with the default tolerance of TOL $= 10^{-3}$. For every α, Figure 3.2 plots a sequence of dots corresponding to the last 50 stepsizes, with the exception that the final stepsize is not plotted since it is generally restricted artificially in order

to hit the output point $t = 100$. The mean of the stepsizes is plotted with an 'o' symbol. Comparing these results with Figures 2.1 and 3.1, we see that for those α where a stable equilibrium exists, the stepsize remains exactly constant (only a single point can be seen) at the h_S level. Further checks showed that the $U(t_n)y_n$ sequences are period two, having a size that depends linearly upon TOL, as predicted by Theorem 3.1. Hence in these tests, wherever a stable equilibrium state exists, it is seen in practice. With those α for which the stability matrix has complex eigenvalues, or for which an unstable equilibrium exists, the stepsize oscillates around the h_S limit and does not settle down to a fixed value. Overall, the theory derived in this section gives a full account of the long term behaviour of the algorithm.

In conclusion, we mention that our stability analysis could be extended in a straightforward way to cover the case where Z is replaced by a general 2×2 matrix in (2.1). Also, the techniques could be applied to other numerical methods. Neither extension, however, would add to the main point of our paper, which is that a generalisation of the standard model (1.1) to periodic coefficients can cause a significant, but predictable change in the stability properties of an ODE solver.

Acknowledgements.

I am indebted to Nick Trefethen for drawing my attention to the problem (2.1) during an extremely enjoyable collaboration [5]. I also thank Nick Higham and Nick Trefethen for useful comments on this paper.

REFERENCES

1. Hairer, E. and Wanner, G. (1991) *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, Springer-Verlag, Berlin.
2. Hall, G. (1985) *Equilibrium states of Runge Kutta schemes*, ACM Trans. Math. Soft. 11, 289–301.
3. Hall, G. (1986) *Equilibrium states of Runge Kutta schemes: part II*, ACM Trans. Math. Soft. 12 183–192
4. Hall, G. and Higham, D. J. (1988) *Analysis of stepsize selection schemes for Runge-Kutta codes*, IMA J. Numer. Anal. 8, 305–310.
5. Higham, D. J. and Trefethen, L. N. (1993) *Stiffness of ODEs*, BIT 33, 285–303.
6. Jordan, D. W. and Smith, P. (1977) *Nonlinear Ordinary Differential Equations*, Oxford University Press, Oxford.
7. Kreiss, H.-O. (1978) *Difference methods for stiff ordinary differential equations*, SIAM J. Numer. Anal. 15, 21–58.
8. Moler, C. B. Little, J. N. and Bangert, S. *386-Matlab User's Guide*, The Math-Works, Inc., 21 Eliot St., South Natick, Massachusetts 01760, 1989.
9. Sand, J. (1980) *A note on a differential system constructed by H. O. Kreiss*, Report TRITA-NA-8004, Dep. Numer. Anal. and Comp. Science, The Royal Institute of Technology, Stockholm, Sweden.