JOURNAL OF COMPUTATIONAL PHYSICS 62, 26–39 (1986)

On the Stability of Mesh Equidistribution Strategies for Time-Dependent Partial Differential Equations*

J. MICHAEL COYLE, JOSEPH E. FLAHERTY, AND RAYMOND LUDWIG

Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, New York 12181

Received May 22, 1984; revised December 27, 1984

We study the stability of several mesh equidistribution schemes for time-dependent partial differential equations in one space dimension. The schemes move a finite difference or finite element mesh so that a given quantity is uniform over the domain. We consider mesh-moving methods that are based on solving a system of ordinary differential equations for the mesh velocities and show that some of these methods are unstable with respect to an equidistributing mesh when the partial differential system is dissipative. Using linear perturbation techniques, we are able to develop simple criteria for determining the stability of a particular method and show how to construct stable differential systems for the mesh velocities. Several examples illustrating stable and unstable mesh motions are presented.

1. INTRODUCTION

Many technological situations involve the rapid formation, propagation, and disintegration of small-scale structures. Some examples are shock waves in compressible flows, shear layers in laminar and turbulent flows, phase boundaries in nonequilibrium processes, combustion fronts, and classical boundary layers. With increasing complexity of the physical problem, there is an increasing need for reliable and robust software tools to accurately and efficiently describe the phenomena. Adaptive techniques have been widely used to solve problems involving ordinary differential equations with rapid transitions, and are thus likely candidates for providing the computational methods and codes necessary to solve more difficult problems involving partial differential equations.

Adaptive techniques for partial differential equations can be roughly divided into two categories: (i) local refinement methods, where uniform fine grids are added to coarse grids in regions where the solution is not adequately resolved, and (ii) moving mesh methods, where grids of a fixed number of finite difference cells or

* The authors were partially supported by the U.S. Air Force Office of Scientific Research, Air Force Systems Command, USAF, under Grant AFOSR 80-0192 and the U.S. Army Research Office under Contract DAAG29-82-K-0197.

0021-9991/86 $3.00

finite elements are moved so as to follow and resolve local nonuniformities in the solution. A representative sample of both types of methods is contained in Babuska, Chandra, and Flaherty [3]. Each technique has its advantages, for example, local refinement techniques can in principle add enough grids to resolve any fine-scale structure, while moving mesh methods are superior at reducing dispersive errors in the vicinity of wavefronts (cf. Hedstrom and Rodrigue [13]).

In this note, we study the stability of several moving mesh schemes that are based on equidistribution, i.e., schemes that move a mesh so that a particular quantity is uniform over the domain. More specifically, we consider equidistribution problems in one space dimension and determine a mesh $\{a = x_0 < x_1(t) < \cdots < x_{N-1}(t) < x_N = b\}$ at time t so that

$$\int_{x_{j-1}}^{x_j} w(x, t)\, dx = c(t) = (1/N) \int_a^b w(x, t)\, dx, \qquad j = 1, 2, \ldots, N. \tag{1.1}$$

The positive density or weight function $w(x, t)$ is usually chosen to be a function of the solution of the partial differential system. For example, w has been chosen to be proportional to the gradient, curvature, combinations of the gradient and curvature, and the local discretization error of the solution of the partial differential equations (cf. Anderson [1], Bell and Shubin [5], Davis and Flaherty [7], Dwyer [9, 10], Hyman and Naughton [16], Rai and Anderson [20], Smooke and Koszykowski [22], and Thompson [23]).

Equidistribution strategies have also been used in codes for variable knot spline interpolation (cf., e.g., de Boor [8]) and for two-point boundary value problems (cf. Ascher, Christiansen, and Russell [2], Lentini and Pereyra [17], and Russell and Christiansen [21]). In these cases, it has been shown (cf. de Boor [8] or Pereyra and Sewell [18]) that the task of selecting a mesh to minimize the discretization error is asymptotically equivalent (for large N) to equidistributing the local discretization error.

Equidistribution techniques are typically applied to time-dependent problems by (i) solving (1.1) simultaneously with the solution of the partial differential equations, (ii) extrapolating equidistributing meshes at past time levels to future time levels, or (iii) developing a system of ordinary differential equations for, say, the mesh velocities, $dx_j(t)/dt := \dot{x}_j(t)$, $j = 0, 1, \ldots, N$, that are equivalent to (1.1), and solving it numerically. Many researchers have reported problems with extrapolating equidistributing meshes or with integrating differential equations for the mesh velocities. For example, if sufficient care is not exercised, mesh trajectories can leave the domain $[a, b]$, cross each other, or oscillate wildly from time step to time step (cf. Fig. 3). These events can even occur when the solution of the partial differential equations is changing very little. In order to explain these phenomena, we use linear perturbation techniques to study the stability of several differential systems for the mesh velocities. In particular, we show that an intuitively obvious system that has (1.1) as its exact solution is unstable whenever w is a decreasing function of time, e.g., when the partial differential system is dissipative. We also show how to stabilize an unstable system and that a differential system for the mesh velocities that was suggested by Hyman [15] is unconditionally stable to small perturbations from an equidistributing mesh. Since many mesh extrapolation schemes are asymptotically equivalent (for small time increments) to ordinary differential equations for mesh velocities, we would expect our results to also apply to these schemes.

In Section 2 we discuss an algorithm for solving (1.1) at a given time t, in Section 3 we present our stability results for differential systems that approximate (1.1), and we summarize our findings in Section 4.

2. An Equidistribution Algorithm

The equidistribution problem (1.1) can most easily be solved by a technique due to de Boor [8]. Thus, we let

$$T(x, t) = \int_a^x w(s, t)\, ds, \qquad a \leqslant x \leqslant b. \tag{2.1}$$

Then

$$c(t) = (1/N)\, T(b, t) \tag{2.2}$$

and the equidistributing mesh $x_j(t)$, $j = 0, 1, \ldots, N$, is determined as the solution of the nonlinear system

$$\Phi_j(x_j(t), t) := T(x_j(t), t) - jc(t) = 0, \qquad j = 0, 1, \ldots, N. \tag{2.3}$$

The equidistribution problem has a nonunique solution whenever $w(x, t) := 0$; hence, we may expect numerical difficulties when $w(x, t)$ is small on any subinterval of $[a, b]$. This problem is usually handled by imposing a lower bound on w, e.g., it is common to replace $w(x, t)$ by $w(x, t) + \eta$. There are many choices for the positive parameter η. Davis and Flaherty [7] suggest that η should be related to the discretization error of the numerical method that is being used to solve the partial differential equations. Another popular choice (cf., e.g., Dwyer [9]) is to set η to unity, when the interval $[a, b]$ and w have been appropriately scaled. Among other things, both of these choices insure that the solution of (2.3) is a uniform mesh whenever w is small everywhere on $[a, b]$. Throughout the remainder of this note, we assume that $w(x, t) \geqslant 0$ for $a \leqslant x \leqslant b$, $t \geqslant 0$, with $w = 0$ only at a finite number of isolated points. This is sufficient to guarantee a unique solution of (2.3).

If $w(x, t)$ is a function of the numerical solution of the associated partial differential equations, it will generally only be known discretely. Suppose w is known at the points x_i^0, $i = 0, 1, \ldots, M$, then we approximate it between mesh points by a piecewise polynomial in x and integrate (2.1) to find a piecewise polynomial approximation to $T(x, t)$. The function $c(t)$ can then be determined approximately from (2.2). An approximation x_j^1, $j=0, 1,..., N$, to an equidistributing mesh is determined by solving (2.3) by, e.g., Newton iteration. If, in particular, we approximate $w(x, t)$ by a piecewise linear function of x with respect to the mesh x_i^0, $i=0, 1,..., M$, then $T(x, t)$ is a piecewise quadratic function of x, and (2.3) can be solved for x_j^1, $j=0, 1,..., N$, directly by the quadratic formula to give

$$x_j^1 = x_{i-1}^0 + \frac{2\gamma}{\beta + (\beta^2 + 2\alpha\gamma)^{1/2}}, \qquad j=0, 1,..., N, \tag{2.4a}$$

where

$$\alpha = \frac{w(x_i^0, t) - w(x_{i-1}^0, t)}{x_i^0 - x_{i-1}^0}, \tag{2.4b}$$

$$\beta = w(x_{i-1}^0, t), \qquad \gamma = jc - T(x_{i-1}^0, t), \tag{2.4c,d}$$

and i is such that $T(x_{i-1}^0, t) \leqslant jc < T(x_i^0, t)$.

The number of points, M, in the input mesh x_i^0, $i=0, 1,..., M$, and, N, in the output mesh x_j^1, $j=0, 1,..., N$, are not necessarily the same. This could be useful in situations where the function $w(x, t)$ is known very precisely, e.g., $w(x, 0)$ can often be calculated exactly using the initial conditions of the associated partial differential equations. In this case, M can be determined so that the integrals in (2.1) to (2.3) and the equidistributing mesh can be evaluated to a prescribed level of accuracy. For example, if the trapezoidal rule is used to approximate $T(x, t)$, an approximation of the equidistributing mesh $x_j(t)$, $j=0, 1, 2,..., N$, can be determined to tolerance ε by selecting

$$M^2 > [(b-a)^3/4\varepsilon] \max_{a<x<b} w_{xx}(x, t) / \min_{a<x<b} w(x, t), \tag{2.5}$$

when $w(x, t) > 0$. This estimate is obtained from standard error formulae for the trapezoidal rule and elementary continuity arguments.

When $N=M$ we may think of solving (2.1) to (2.3) iteratively. Thus, the mesh x_j^1, $j=0, 1,..., N$, can be used to calculate a new piecewise polynomial approximation to $w(x, t)$ and this can be used to calculate another approximation x_j^2, $j=0, 1,..., N$, to an equidistributing mesh, etc. However, this iterative strategy does not necessarily converge near a local minimum of $w(x, t)$ as illustrated by the following simple example.

EXAMPLE 2.1. Consider a three-point mesh ($M=N=3$), x_j, $j=0, 1, 2$, on $-1 \leqslant x \leqslant 1$ with $w(x, t) = x^2$. The endpoints $x_0 = -1$ and $x_2 = 1$ are fixed, and the only point that needs to be determined by iteration is x_1. The exact value of x_1 is, of course, zero; however, we start with an initial guess $x_1^0 = \varepsilon$, use piecewise linear approximations for w, and see if successive iterates x_1^ν, $\nu = 1, 2,...,$ converge to zero. We can show that:

(1) If $\varepsilon = \xi := 2 - \sqrt{3}$, then $x_1^{2\nu+1} = -\xi$ and $x_1^{2\nu} = \xi$, $\nu = 0, 1, \ldots$. Thus, the iterative solution is a two-point limit cycle.

(2) If $\varepsilon \neq 0$, then $|x_1^\nu| \leqslant \xi$, $\nu = 1, 2, \ldots$.

(3) If $|x_1^\nu| < \xi$, then $|x_1^\nu| < |x_1^{\nu+1}|$.

(4) If $x_1^\nu > 0$ then $x_1^{\nu+1} < 0$.

Items (2) to (4) imply that x_1^ν does not approach zero for any nonzero initial guess, but instead approaches a limit cycle, oscillating between ξ and $-\xi$ on alternate iterations.

3. Mesh Dynamics

The discussion of Section 2 involved the computation of an equidistributing mesh at one time level. To obtain an equidistributing mesh at subsequent time levels, we can either (i) solve (2.3) simultaneously with the partial differential system, (ii) extrapolate equidistributing grids from previous time levels, or (iii) construct a differential equation, e.g., for the mesh velocities, whose solution is an equidistributing mesh. As previously noted, some extrapolation schemes and differential equations may be asymptotically equivalent for small time steps. In this section, we use linear stability analyses to explain why many researchers have been experiencing difficulties with extrapolation procedures or in solving some differential equations for mesh velocities. There are no essential stability problems associated with solving (2.3) directly and the only possible objection to this approach is computational cost.

We begin our analysis by considering the differential system obtained by differentiating (2.3) with respect to time. Upon use of (2.1) this gives

$$w(x_j, t)\,\dot{x}_j = -\left[\int_a^{x_j} w_t(x, t)\,dx - j\dot{c}(t)\right], \qquad j = 1, 2, \ldots, N-1. \tag{3.1}$$

Suppose $x_j(t)$, $j = 0, 1, \ldots, N$, is an equidistributing mesh that exactly satisfies (2.3) and (3.1) for $t \geqslant 0$ and introduce a small perturbation $\delta x_j(0)$, $j = 0, 1, \ldots, N$, at $t = 0$ and study its effect on (3.1). If no additional errors are introduced, the perturbed system satisfies

$$w(x_j(t) + \delta x_j(t), t)(\dot{x}_j + \delta\dot{x}_j) = -\left[\int_a^{x_j + \delta x_j} w_t(x, t)\,dx - j\dot{c}(t)\right],$$
$$j = 1, 2, \ldots, N-1, \tag{3.2}$$

and is subject to the constraints $\delta x_0(t) = \delta x_N(t) = 0$. We further assume that $|\delta x_j| \ll b - a$, $j = 1, 2, \ldots, N-1$, and linearize (3.2) to get

$$\frac{d}{dt}\left[w(x_j(t), t)\,\delta x_j(t)\right] = 0, \qquad j = 1, 2, \ldots, N-1. \tag{3.3}$$

Integrating, we find

$$\delta x_j(t) = \frac{w(x_j(0), 0)}{w(x_j(t), t)} \delta x_j(0). \tag{3.4}$$

Therefore, the differential system (3.1) is stable to linear perturbations if

$$L(t) = \max_{0<j<N} \frac{w(x_j(0), 0)}{w(x_j(t), t)} \tag{3.5}$$

is less than unity. Unfortunately, most choices of $w(x, t)$ are likely to be decaying functions of time for dissipative parabolic systems and this will almost certainly yield a value of $L(t)$ that is larger than unity. Local instabilities can also occur whenever the mesh is moved so that $L(t)$ exceeds unity for some specific times. These instabilities may grow or decay as time progresses depending on the value of L.

The following two examples illustrate some of the instabilities that can occur.

EXAMPLE 3.1. Consider the heat conduction problem

$$u_t = u_{xx}, \qquad 0 < x < 1, \quad t > 0, \tag{3.6a}$$

$$u(x, 0) = \sin \pi x, \qquad u(0, t) = u(1, t) = 0. \tag{3.6b,c}$$

The exact solution of this problem is

$$u(x, t) = e^{-\pi^2 t} \sin \pi x. \tag{3.6d}$$

We take

$$w(x, t) = \sqrt{|u_{xx}(x, t)|}. \tag{3.6e}$$

Since this problem and $w(x, t)$ are separable, the correct strategy is to generate an equidistributed mesh at time $t = 0$ and use it for all time. However, $L(t) \propto \exp(\pi^2 t/2)$ and, thus, we expect the solution of (3.1) to be unstable. In Fig. 1, we display the meshes produced by both (2.3) and (3.1) and the unstable behavior of (3.1) is clearly visible. The trapezoidal rule with $M = 100$ was used to evaluate all integrals, a mesh of $N = 10$ elements was equidistributed, and an initial perturbation $\delta x_j(0) = 0.0075$, $j = 1, 2, ..., N-1$, was introduced. Equations (3.1) (and all differential equations appearing in Examples 3.2 to 3.4) were solved using the IMSL version of Gear's code [12].

EXAMPLE 3.2. We consider a problem for a partial differential equation that has the exact solution

$$u(x, t) = \tanh[r_1(x-1) + r_2 t], \qquad 0 \leqslant x \leqslant 1, \quad t \geqslant 0. \tag{3.7}$$

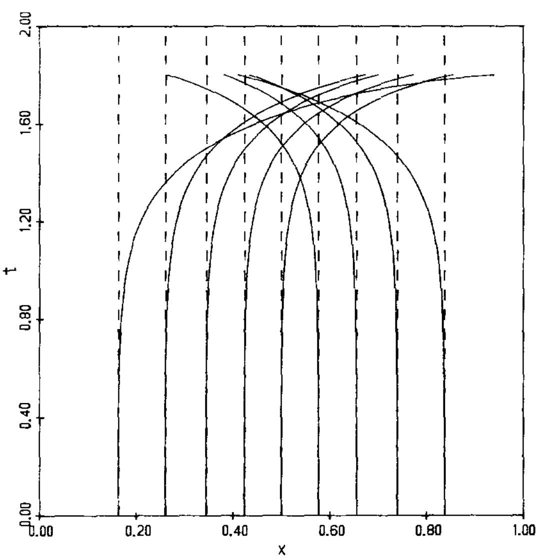

FIG. 1. Mesh trajectories for Example 3.1 calculated by (2.3) (broken curve) and (3.1) (solid curve). An initial perturbation of 0.0075 was introduced in the solution of (3.1) and this causes an instability to develop.

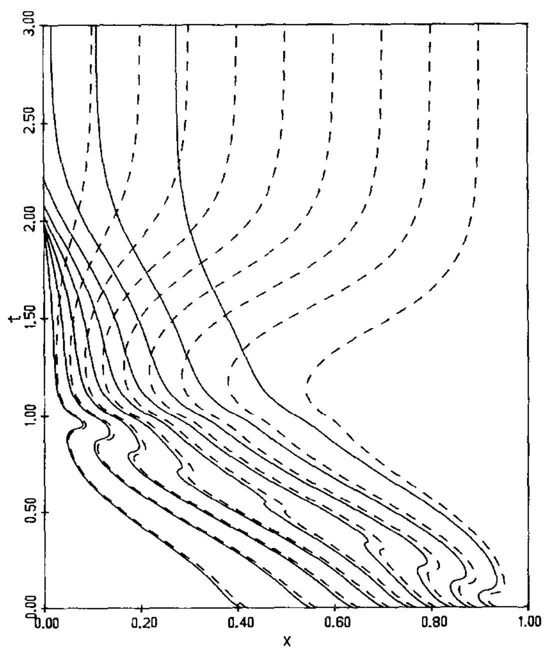

FIG. 2. Mesh trajectories for Example 3.2 calculated by (2.3) (broken curve) and (3.1) (solid curve). An initial perturbation of 0.015 was introduced in the solution of (3.1). The solution of (3.1) is marginally stable for small times, but becomes unstable as w decreases with time.

The function (3.7) is a wave that travels in the negative x direction when r_1 and r_2 are positive. The values of r_1 and r_2 determine the steepness of the wave and its propagation speed. Such solutions could arise from several types of partial differential equations, e.g., Davis and Flaherty [7] studied a heat conduction problem of the form

$$u_t + f(x, t) = (1/s)\, u_{xx}, \qquad 0 < x < 1, \quad t > 0, \tag{3.8}$$

where the initial conditions, Dirichlet boundary conditions, constant diffusion $1/s$, and source f were chosen so that the exact solution was given by (3.7).

The meshes produced by both (2.3) and (3.1) for $r_1 = r_2 = 5$ and

$$w(x, t) = \sqrt{|u_{xx}(x, t)|} + \eta, \tag{3.9}$$

where $\eta = 0.1$, are shown in Fig. 2. The solution of (3.1) is marginally stable for small times, but w decreases as the wavefront progresses towards $x = 0$ and the instability is apparent. In fact, some mesh trajectories have left the domain [0, 1]. The trapezoidal rule with $M = 100$ was used to evaluate all integrals, a mesh of $N = 10$ elements was equidistributed, and an initial perturbation $\delta x_j(0) = 0.015$, $j = 1, 2, \ldots, N-1$, was introduced.

In order to indicate what could happen when extrapolation is used to advance a mesh, we solve this problem using (2.3) to equidistribute a mesh at two time levels and use linear extrapolation to advance it to a third time level. Thus, for uniform time steps, we compute

$$z_j(t + \Delta t) = 2x_j(t) - x_j(t - \Delta t), \qquad j = 1, 2, \ldots, N-1, \tag{3.10}$$

where $z_j(t)$ is the extrapolated mesh position and x_j satisfies (2.3). Our results with $M = N = 10$ and a time step of $\Delta t = 0.01$ are compared with the exact solution of (2.3) in Fig. 3. The extrapolated solution follows the exact solution and oscillates about it for times less than unity.

Petzold [19] suggested that the following linear combination of equations (2.3) and (3.1) might yield stable meshes with some improved dynamic behavior:

$$\dot{\Phi}_j + \lambda \Phi_j = 0, \qquad j = 1, 2, \ldots, N-1. \tag{3.11}$$

Here $\lambda > 0$ is a parameter to be determined so that (3.11) is stable. A similar approach has been used by Holcomb and Hindman [14].

A linear stability analysis of (3.11) parallels the one used for (3.1), and in this case we find that Eq. (3.11) is stable to small perturbations provided that

$$L(t)\, e^{-\lambda t} < 1, \tag{3.12}$$

where $L(t)$ was defined in (3.5).

In practice (3.11) is solved numerically and its stability should be reexamined in this light. For example, if the explicit Euler method were used to solve (3.11), then

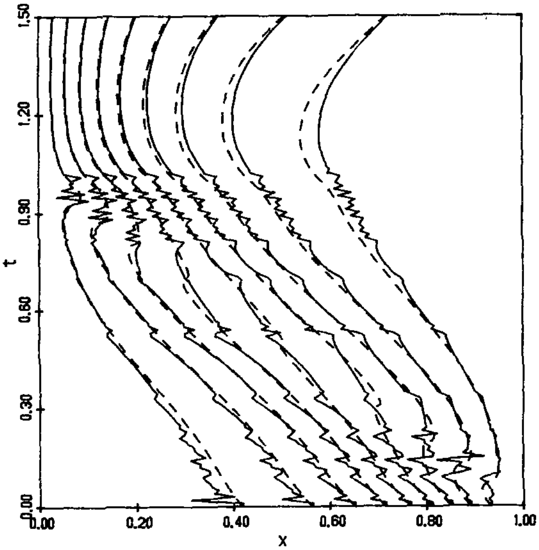

FIG. 3. Mesh trajectories for Example 3.2 calculated by (2.3) (broken curve) and linear extrapolation (solid curve). A mesh with $M = N = 10$ was used in both cases. The extrapolated solution was obtained using a uniform time step of $\Delta t = 0.01$.

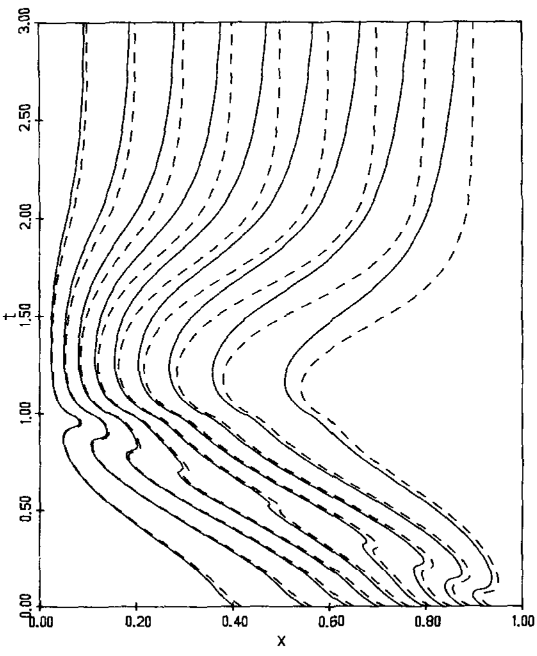

FIG. 4. Mesh trajectories for Example 3.3 calculated by (2.3) (broken curve) and (3.11) with $\lambda = 1$ (solid curve). An initial perturbation of 0.015 was introduced in the solution of (3.11). The solution obtained by (3.11) follows the exact solution reasonably closely and is not showing any signs of instability.

the appropriate stability condition is determined from (3.12) by replacing t by Δt and expanding the left-hand side of (3.12) in a Taylor's series to linear order in Δt. Using (3.5) we get the following stability condition for the time step $(t, t+\Delta t)$:

$$L_\Delta(t) := \max_{0<j<N} |1 - w_x(x_j(t), t)[x_j(t+\Delta t) - x_j(t)]/w(x_j(t), t)$$
$$- \Delta t[\lambda - w_t(x_j(t), t)/w(x_j(t), t)]| < 1. \tag{3.13}$$

This condition would be difficult to verify in practice.

The following example illustrates the performance of (3.11).

EXAMPLE 3.3. We solve (3.11) with $\lambda = 1$ using (3.7) and (3.9) with the same parameter values as Example 3.2. This solution is compared with the solution of (2.3) in Fig. 4. The mesh produced by integrating (3.11) is not showing any signs of instability.

Exact values of w and w_t were used to calculate the solution of (3.11) in the previous example. In practice these quantities would probably be approximated by finite differences, and this could introduce some instabilities. A differential system that does not require a knowledge of w_t, seems to be less sensitive to perturbations in w, and has solutions that approximate those of (2.3) is

$$\dot{z}_j(t) = -\lambda \Phi_j(z_j, t), \qquad j = 1, 2, ..., N-1. \tag{3.14}$$

If $\Phi_j(z_j, t)$ is positive then $z_j(t)$ is too large relative to the equidistributing mesh $x_j(t)$, and (3.14) will tend to reduce it when $\lambda > 0$. Similarly, if $\Phi_j(z_j, t)$ is negative, then $z_j(t)$ is smaller than $x_j(t)$, and (3.14) will increase it. Furthermore, larger values of the parameter λ will give shorter relaxation times of z_j to x_j; however, contrary to (3.1) and (3.11), the equidistributing mesh x_j is not a solution of (3.14). This strategy is similar to one suggested by Hyman and Naughton [16].

We again study the stability of this system with respect to linear perturbations. Thus, we let $x_j(t)$, $j = 0, 1, ..., N$, be an equidistributing mesh, introduce a perturbation at time $t = 0$, replace $z_j(t)$ by $x_j(t) + \delta x_j(t)$ in (3.14), use (2.3), and retain only linear terms in δx_j to get

$$\delta \dot{x}_j + \lambda w(x_j, t)\, \delta x_j = -\dot{x}_j, \qquad j = 1, 2, ..., N-1. \tag{3.15}$$

This system may be easily integrated to give

$$\delta x_j(t) = \delta x_j(0) \exp\left[-\lambda \int_0^t w(x_j(s), s)\, ds\right]$$
$$- \int_0^t \dot{x}_j(s) \exp\left[-\lambda \int_s^t w(x_j(\sigma), \sigma)\, d\sigma\right] ds, \qquad j = 1, 2, ..., N-1. \tag{3.16}$$

We bound (3.16) by

$$|\delta x_j(t)| \leqslant |\delta x_j(0)| \exp(-\lambda w_{\min} t) + (\dot{x}_{\max}/\lambda w_{\min})[1 - \exp(-\lambda w_{\min} t)],$$
$$j = 1, 2, \ldots, N-1, \quad (3.17a)$$

where

$$w_{\min} = \min_{\substack{a<x<b \\ 0<t<\infty}} |w(x, t)| \geqslant \eta, \qquad \dot{x}_{\max} = \max_{\substack{0<j<N \\ 0<t<\infty}} |\dot{x}_j(t)|. \qquad (3.17b,c)$$

Since $\dot{x}_{\max}$ is bounded (cf. (2.3)) and this, together with (3.17a), implies that the perturbations $\delta x_j(t)$, $j = 0, 1, \ldots, N$, are bounded for $\lambda > 0$, but they do not necessarily decay to zero. If, e.g., the partial differential system is dissipative and $w(x, t) \to \eta$ as $t \to \infty$, then $\dot{x}_j \to 0$, $j = 0, 1, \ldots, N$, and the perturbations δx_j, $j = 0, 1, \ldots, N$, decay (cf. (3.16)). If this is not the case, then the bound on $\delta x_j(t)$, $j = 0, 1, \ldots, N$, can be made arbitrarily small by selecting λ sufficiently large. This, however, will make the differential system (3.14) stiff.

When (3.14) is solved numerically, it will have to satisfy any additional stability

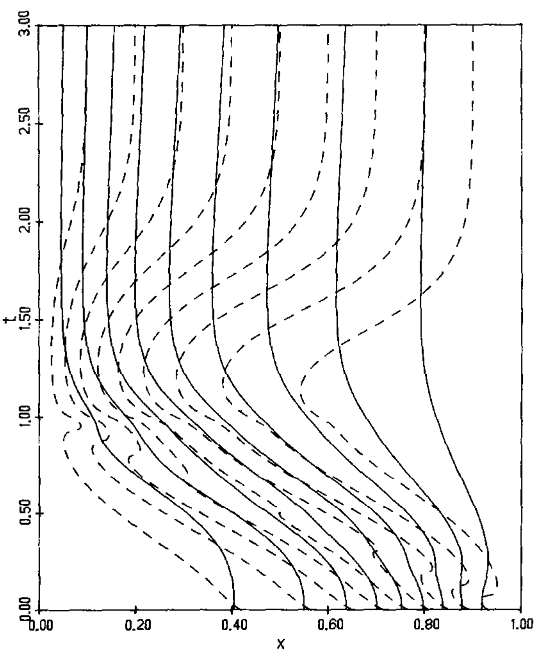

FIG. 5. Mesh trajectories for Example 3.4 calculated by (2.3) (broken curve) and (3.14) with $\lambda = 1$ (solid curve). An initial perturbation of 0.015 was introduced in the solution of (3.14). The solution obtained by (3.14) approaches the exact solution at a very slow rate.

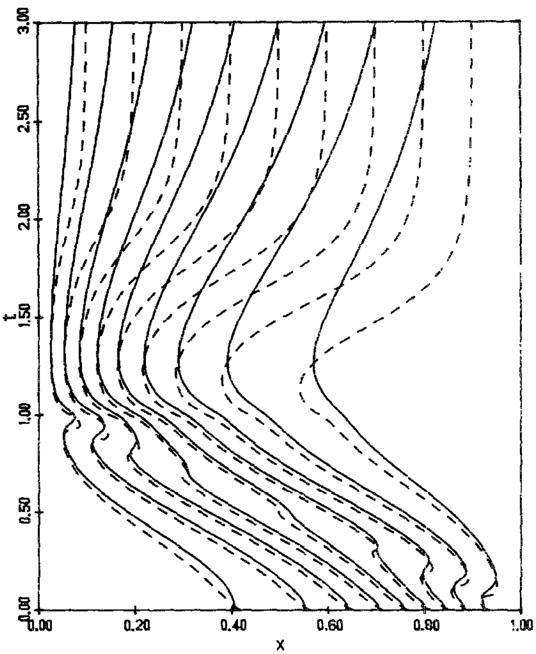

FIG. 6. Mesh trajectories for Example 3.4 calculated by (2.3) (broken curve) and (3.14) with $\lambda = 10$ (solid curve). An initial perturbation of 0.015 was introduced in the solution of (3.14). The solution obtained by (3.14) approaches the exact solution more quickly than when $\lambda = 1$.

requirements of the numerical integration technique. For example, if (3.14) is solved by the explicit Euler method, the stability restriction for the time step $(t, t+\Delta t)$ is

$$\lambda \, \Delta t \max_{a \leqslant x \leqslant b} w(x, t) < 2. \tag{3.18}$$

This suggests that λ should be kept relatively small; however, this is in conflict with the aim of selecting λ large in order to get quick decay of a trajectory to an equidistributing mesh.

The following examples illustrate the performance of (3.14).

EXAMPLE 3.4. We solve (3.14) using (3.7) and (3.9) with the same parameter values as Example 3.2. The solutions for $\lambda = 1$ and 10 are compared with the solution of (2.3) in Figs. 5 and 6, respectively. The time for the solution of (3.14) to relax to the exact solution is much longer when $\lambda = 1$ than when $\lambda = 10$.

4. DISCUSSION

We have explained why many intuitively obvious schemes for calculating equidistributing meshes for time-dependent partial differential equations are unstable. In particular, we have given a stability condition on the equidistribution density $w(x, t)$ that can easily be verified in practice. We have also suggested some methods of stabilizing unstable mesh-moving techniques. Specifically, the technique given by (3.14) seems to offer several advantages. It does not require knowledge nor continuity of w_t and it is stable for all positive values of the parameter λ. However, it is not asymptotically stable unless λ is large, and this introduces stiffness into (3.14). Thus, (3.11) may be preferable in those cases when precise mesh control is needed.

We note, however, that it is rarely necessary to calculate an equidistributing mesh very precisely. An $O(1/N)$ error in the location of the optimal equidistributing mesh will typically affect accuracy of the solution of the partial differential equations by $O(1/N^2)$ (cf. Babuska and Rheinboldt [4]). Thus, it is preferable to add finite difference cells or finite elements and reduce the magnitude of the error rather than devoting time to solving the equidistribution problem to great precision. If $w(x, t)$ is chosen so that the local discretization error is equidistributed, then refinement, when necessary, is done globally on $[a, b]$. This strategy should be simpler to implement than some local refinement finite difference and finite element methods (cf. Berger and Oliger [6] and Flaherty and Moore [11], respectively) which do not move meshes and use relatively sophisticated tree-structured grids.

Acknowledgments

The authors would like to thank Dr. J. M. Hyman of Los Alamos National Laboratories and Dr. L. Petzold of Sandia National Laboratories for their helpful comments and suggestions.

References

1. D. A. Anderson, *in* "Adaptive Computational Methods for Partial Differential Equations" (I. Babuska, J. Chandra, and J. E. Flaherty, Eds.), pp. 208–223, SIAM, Philadelphia, 1983.
2. U. Ascher, J. Christiansen, and R. D. Russell, *ACM Trans. Math. Software* **7** (1981), 209–222.
3. I. Babuska, J. Chandra, and J. E. Flaherty, Eds., "Adaptive Computational Methods for Partial Differential Equations," SIAM, Philadelphia, 1983.
4. I. Babuska and W. C. Rheinboldt, *Int. J. Numer. Methods Engrg.* **12** (1978), 1597–1615.
5. J. B. Bell and G. R. Shubin, *J. Comput. Phys.* **52** (1983), 569–591.
6. M. J. Berger and J. Oliger, *J. Comput. Phys.* **53** (1984), 484–512.
7. S. F. Davis and J. E. Flaherty, *SIAM J. Sci. Statist. Comput.* **3** (1982), 6–27.
8. C. De Boor, "A Practical Guide to Splines," Applied Mathematical Sciences, No. 27, Springer-Verlag, New York, 1978.
9. H. A. Dwyer, Grid adaption for problems with separation, cell Reynolds number, shock-boundary layer interaction, and accuracy, AIAA Paper No. 83-0449, AIAA Twenty First Aerospace Sciences Meeting, 1983.
10. H. A. Dwyer, *in* "Adaptive Computational Methods for Partial Differential Equations" (I. Babuska, J. Chandra, and J. E. Flaherty, Eds.), pp. 111–122, SIAM, Philadelphia, 1983.
11. J. E. Flaherty and P. K. Moore, *in* "Proceedings, Conference on Accuracy Estimates and Adaptive Refinements in Finite Element Computations, Lisbon, 1984," in press.

12. C. W. GEAR, "Numerical Initial Value Problems in Ordinary Differential Equations," Prentice–Hall, Englewood Cliffs, N.J., 1971.
13. G. W. HEDSTROM AND G. H. RODRIGUE, Adaptive-grid methods for time-dependent partial differential equations, UCRL-87242 preprint, Lawrence Livermore National Laboratory, Livermore, Calif., 1982.
14. J. E. HOLCOMB AND R. G. HINDMAN, Development of a dynamically adaptive grid method for multidimensional problems. AIAA paper 84-1668, June 1984.
15. J. M. HYMAN, "Adaptive Mesh Methods for Partial Differential Equations," Technical Report LA-UR-82-1637, Los Alamos National Laboratory, Los Alamos, 1982
16. J. M. HYMAN AND M. J. NAUGHTON, *in* "Proceedings, SIAM–AMS Conference on Large Scale Computation in Fluid Mechanics," SIAM, Philadelphia, 1984.
17 M. LENTINI AND V. PEREYRA, *SIAM J. Numer. Anal.* **14** (1977), 91–111.
18. V. PEREYRA AND E. G. SEWELL, *Numer. Math.* **23** (1975), 261–268.
19. L. PETZOLD, Personal communication, 1983.
20. M. M. RAI AND D. A. ANDERSON, The use of adaptive grids in conjunction with shock capturing methods, AIAA paper 81-1012, June 1981.
21. R. D. RUSSELL AND J. CHRISTIANSEN, *SIAM J. Numer. Anal.* **15** (1978), 59–80.
22. M. D. SMOOKE AND M. L. KOSZYKOWSKI, "Fully Adaptive Solutions of One-Dimensional Mixed Initial-Boundary Problems in Combustion," Technical Report SAND83-8219, Sandia National Laboratories, Livermore, Calif., 1983.
23. J. F. THOMPSON, A survey of dynamically-adaptive grids in the numerical solution of partial differential equations, AIAA paper 84-1606, June 1984.