JOURNAL OF COMPUTATIONAL PHYSICS 5, 244–264 (1970)

Variable Mesh Multistep Methods for Ordinary Differential Equations*

RODNEY VAN WYK

North American Rockwell Corporation, Rocketdyne Division, Canoga Park, California 91304

Received April 1, 1969

A multistep predictor-corrector method for the numerical solution of ordinary differential equations is developed. The difference equations employed are generalizations, for the case of variable mesh spacing, of previous formulas requiring fixed step size. In addition to retaining the high local accuracy of the earlier methods, the variable mesh method is developed in a form conducive to the generation of effective criteria for the selection of subsequent step sizes in the step by step solution of differential equations. These criteria are based on considerations of truncation error, convergence of corrector iterations, and an extensive treatment of relative numerical stability. The algorithm has been tested extensively and compared with other methods. The results of the comparison favor the new method. An application of the method to a heat transfer problem is discussed.

1. INTRODUCTION

A great deal of research effort has been directed toward the numerical solution of first order nonlinear ordinary differential equations because of the practical importance of such problems. The most widely used numerical methods that have been developed for these problems provide approximate values of the solution at discrete points according to a stepwise computation beginning at an initial point for which the solution is known. These methods are called one-step methods if the calculation of the solution at a given point depends explicitly on values of the solution and one or more of its derivatives at only one previous point. Multistep methods require values at two or more previous points. One-step (Runge-Kutta) methods are very convenient because the step increments can be changed readily from step to step as desired and because the solution in the initial steps is calculated with the same formulas as used in subsequent steps. Multistep methods, although less convenient, are usually more efficient because, by making use of the calcula-

* This work was supported by the National Aeronautics and Space Administration Office of Advanced Research and Technology under Contract No. NAS7-470.

tions of more than one previous step, less computer time is required to achieve the same accuracy as achieved with a one-step method.

The research reported here was directed toward the development and testing of variable mesh multistep methods which not only preserve the efficiency due to the multistep structure but improve this efficiency by permitting as much freedom in the variation of the step increments as is afforded by one-step methods. Care was taken to formulate the basic difference equations in a manner conducive to the development of effective criteria for selecting the variable mesh increments as the calculation progresses. In the following pages, the basic algorithm is described and the analysis and practical considerations justifying the mesh criteria are presented. The mesh criteria were subjected to extensive numerical testing, and in addition, the algorithm was compared with known methods in the numerical solution of selected differential equations. The results of this and other experimental work are summarized in later sections.

The problem of starting the computation, that is, the requirement of computing the solution at the first few points by a separate technique, in order to initialize multistep methods, is not emphasized here for two reasons. First, because of the variable mesh formulation, the calculation is only initialized once and never has to be restarted as would be required in changing the step size while using a fixed step size, multistep method. In the second place, fairly general starting procedures are readily available for incorporation with the variable mesh method because the step increments used in the starting procedure can be smaller than those used in the subsequent calculations. For example, the starting procedure outlined in [1] for the variable mesh method consists of simply using the one-step Adams-Bashforth/Adams-Moulton formulas for the first step, the two-step formulas for the second step, and the three-step formulas for third step. The same step size is used for each of these three initial steps, and it is chosen small enough to yield the desired accuracy at the first point. There is little danger of exceeding this error at the second and third points since higher order formulas are used.

2. Variable Mesh Multistep Formulas

The initial value problems considered are represented by differential equations of the form

$$dy/dx = F(x, y), \tag{1}$$

with initial condition $y(x_0) = y_0$. Equation (1) represents a single differential equation; however, some special considerations required for systems of differential equations will also be given in the sequel. It is assumed at the outset that F is continuous and satisfies the Lipschitz condition that guarantees the existence of a

unique, continuous and differentiable solution ([2], p. 15). The continuity of higher derivatives will be required later in the discussion of truncation error.

We will use the usual notation in which y_n denotes the computed value of $y(x_n)$ and y_n' denotes $F(x_n, y_n)$. It is assumed that the computed solution is obtained recursively by one or more formulas of the following type:

$$y_{n+1} = a_0 y_n + a_1 y_{n-1} + a_2 y_{n-2} + a_3 y_{n-3} + h(b_{-1} y'_{n+1} + b_0 y_n' + b_1 y'_{n-1} + b_2 y'_{n-2} + b_3 y'_{n-3}). \tag{2}$$

Here h denotes the current step size, $x_{n+1} - x_n$, and is permitted to vary with n. The coefficients a_i and b_i are also variable and it will be convenient later to express them in terms of mesh parameters α, β, and γ defined by

$$\alpha = (x_n - x_{n-1})/h,$$
$$\beta = (x_n - x_{n-2})/h, \tag{3}$$

and

$$\gamma = (x_n - x_{n-3})/h.$$

We require $\gamma > \beta > \alpha$.

We will restrict the present discussion to fourth order methods—that is, methods with error terms proportional to the fifth power of h. The optimum order to use in a given application depends heavily on the degree of accuracy desired, but fourth order is a reasonable compromise for medium range accuracy—say two to six significant figures. With fixed step size, it is often desirable to vary the order within a given application in order to maintain a desired accuracy. The variable mesh procedure, on the other hand, has the advantage of achieving the same objective without switching from formulas of one order to those of another.

Both explicit (predictor) and implicit (corrector) variable mesh formulas are used. The explicit equation has $b_{-1} \equiv 0$, while the implicit usually can be solved by iteration. The coefficients in (2) for the two formulas are determined as follows: By requiring exactness for $F(x, y) \equiv 0$, the relation $a_0 + a_1 + a_2 + a_3 = 1$ is imposed. Because of the known advantage in numerical stability (for the case of fixed step size) resulting from the choice $a_0 = 1, a_1 = a_2 = a_3 = 0$ (as in Adams-type formulas), we make the same choice for the more general variable mesh case. Requiring exactness also for $y = x - x_n$, $\frac{1}{2}(x - x_n)^2$, $\frac{1}{3}(x - x_n)^3$, and $\frac{1}{4}(x - x_n)^4$, the b_i of the predictor

$$y_{n+1} = y_n + h(b_0 y_n' + b_1 y'_{n-1} + b_2 y'_{n-2} + b_3 y'_{n-3}) \tag{4}$$

are determined recursively as follows:

$$b_3 = \frac{2(2 + 3\alpha)(\beta + \alpha) + 3(1 - 2\alpha^2)}{12\gamma(\gamma - \alpha)(\beta - \gamma)},$$

$$b_2 = \frac{2 + 3\alpha - 6\gamma(\gamma - \alpha)\, b_3}{6\beta(\beta - \alpha)},$$

$$b_1 = -(1 + 2\gamma b_3 + 2\beta b_2)/2\alpha,$$

$$b_0 = 1 - b_3 - b_2 - b_1 . \tag{5}$$

Similarly, a corrector of the form

$$y_{n+1} = y_n + h(d_{-1}y'_{n+1} + d_0 y_n' + d_1 y'_{n-1} + d_2 y'_{n-2}) \tag{6}$$

is found with coefficients

$$d_2 = \frac{1 + 2\alpha}{12\beta(1 + \beta)(\beta - \alpha)};$$

$$d_1 = -\frac{2\beta + 1}{12\alpha(1 + \alpha)(\beta - \alpha)};$$

$$d_0 = \tfrac{1}{2} - d_2(1 + \beta) - d_1(1 + \alpha);$$

$$d_{-1} = 1 - d_2 - d_1 - d_0 . \tag{7}$$

For the special case of fixed step size where α, β, and γ have the constant values of 1, 2, and 3, respectively, the above predictor and corrector formulas reduce to the widely accepted Adams-Bashforth and Adams-Moulton formulas, respectively. In this connection one is reminded of the formulas presented by Nordsieck in a paper which, like the present paper, also emphasizes the advantages of changing step size [3]. Although the algorithm of Nordsieck is substantially different from that presented here, it is similar in the sense that his basic integration formulas are equivalent to the Adams formulas. However, the formulation used by Nordsieck appears to be much less conducive to the development of effective mesh selection criteria than is the formulation presented above. This claim is corroborated by evidence obtained when both methods, complete with their respective recommended mesh selection criteria, were applied to selected differential equations. (This work is described in more detail in a later section.) This deficiency of the Nordsieck method may result from his assumption of "fixed point" operations rather than the more commonly used "floating point," as also is suggested by Lewis and Stovall [4].

Assuming continuous higher derivatives of $F(x, y)$, it is evident upon comparing Eq. (4) with an appropriate Taylor Series representation for $y(x_{n+1})$ that the truncation error in (4) can be represented as

$$P_n(h^5/5!)\, y_n^{v} + 0(h^6),$$

where the coefficient P_n depends on α, β, and γ. If we consider the residual error resulting from the application of (4) to the polynomial $(x - x_n)^5$, we find that

$$\begin{aligned} P_n &= 1 - 5(b_1\alpha^4 + b_2\beta^4 + b_3\gamma^4) \\ &= 1 + 5[3(\alpha + \beta + \gamma) + 4(\alpha\beta + \alpha\gamma + \beta\gamma) + 6\alpha\beta\gamma]/12. \end{aligned} \tag{8}$$

Similarly, if the error in (6) is taken in the form

$$C_n(h^5/5!)\, y_n^{v} + 0(h^6),$$

C_n is found to be given by

$$\begin{aligned} C_n &= 1 - 5(d_{-1} + d_1\alpha^4 + d_2\beta^4) \\ &= 1 - 5(3 + 2\alpha\beta + \alpha + \beta)/12. \end{aligned} \tag{9}$$

Various alternative modes of utilization of the predictor and corrector formulas are available in practice. For example, the predictor can be used without employing the corrector at all. On the other hand, if the corrector is used, it usually is used iteratively, with the predictor providing the first guess. Qualitatively, some of the arguments for and against the various alternatives are as follows:

(a) *Number of Derivative Evaluations Per Step*

The "predictor-only" mode requires only one evaluation per step. If one correction is employed, a second evaluation is usually made after the correction to enhance numerical stability. In general, n corrections require either n or $n + 1$ derivative evaluations, depending on whether a final evaluation is or is not carried out. Evaluations of complicated derivative functions freqeuently require a predominant portion of the total computer time.

(b) *Truncation Error*

Implementation of the corrector reduces the truncation error. (It is a simple exercise to show that $| C_n | < | P_n |$.)

(c) *Numerical Stability*

With regard to both absolute and relative stability, the regions of stability become less restrictive as the number of correction-evaluation iterations is increased. Incidentally, these regions become more restrictive as order is increased.

(d) *Availability of Mesh Criteria*

More effective procedures for automatically selecting the mesh increments can be developed for some modes than for others. This consideration favors a predictor-corrector mode with at least two applications of the corrector.

An empirical program was carried out whereby the various modes were compared in the actual numerical solution of selected differential equations. The mesh increments were selected in a manner such that the total number of derivative evaluations was the same for each mode. This work is not reported in detail here since an even more extensive testing program of a similar nature for the case of fixed step size was carried out and reported in detail by Hull and Creemer [5]. Their conclusions, favoring the mode p-d-c-d-c, are in agreement with those reached in the present study. (Here p denotes predictor, d derivative evaluation, and c corrector.) Consequently, the discussion in the remaining sections will be directed primarily toward this mode.

3. Numerical Stability for Single Differential Equations

First note that each corrector iteration is performed according to the equation

$$c_{n+1}^{(k+1)} = y_n + hd_{-1}F(x_{n+1}, c_{n+1}^{(k)}) + h\sum_{i=0}^{2} d_i y'_{n-i}, \tag{10}$$

where the superscript k denotes the k-th iteration. Subtracting this equation from (6), and employing the mean value theorem gives

$$y_{n+1} - c_{n+1}^{(k+1)} = \lambda d_{-1}[y_{n+1} - c_{n+1}^{(k)}],$$

where

$$\lambda = h(\partial F/\partial y)_{\substack{x=x_{n+1} \\ y=\eta}},$$

for some η between y_{n+1} and $c_{n+1}^{(k)}$. Thus, the following condition is required for convergence of the corrector iterations:

$$|\lambda d_{-1}| < 1. \tag{11}$$

It is assumed that condition (11) is satisfied in the following discussion, and, in fact, this condition will be used in the mesh selection procedures described in the next section.

It is also assumed for the purpose of the numerical stability analysis that λ is constant, a standard assumption in the literature for fixed step size. By appropriate

choice of h at each step, λ can be made nearly constant in the variable mesh case. In practice, however, this assumption is usually violated with fixed mesh methods, even when procedures to frequently double or halve the step size are included. Furthermore, when numerical stability is the controlling factor, it is good policy to keep h as large as possible without forcing λ beyond its limitation imposed by the threat of instability. Thus, in this case, the mesh increments used are actually considerably suboptimal at most steps with fixed mesh methods. On the other hand, the variable mesh feature obviously allows much better optimization when the integration is stability limited. Of course, when it is not stability limited, variations in λ are inconsequential.

Initially, let us consider the mode which employs a prediction and k corrections with a derivative evaluation after each prediction and correction. Let ϵ_n denote the propagated error, $y(x_n) - c_n^{(k)}$. Then it can be shown that ϵ_n satisfies the difference equation

$$\epsilon_{n+1} = \epsilon_n \left[1 + \sum_{j=1}^{k} \lambda^j d_{-1}^{j-1}(d_0 + d_{-1}) + \lambda^{k+1} d_{-1}^k b_0\right] + \sum_{i=1}^{2} \epsilon_{n-i} \sum_{j=1}^{k} \lambda^j d_{-1}^{j-1} d_i + \sum_{i=1}^{3} \epsilon_{n-i} \lambda^{k+1} d_{-1}^k b_i ,$$

except for the fifth order truncation error. The effect of the predictor on the propagated error decreases with increasing k because the factor $(\lambda d_{-1})^k$ multiplies the b_i in the above equation. In the limit, the corrector alone determines the error propagation, the equation being given by

$$\epsilon_{n+1}(1 - \lambda d_{-1}) - \epsilon_n(1 + \lambda d_0) - \epsilon_{n-1}\lambda d_1 - \epsilon_{n-2}\lambda d_2 = 0. \tag{12}$$

In practice, when the mesh increments are small enough to provide a reasonably small truncation error, the corrector iterations beyond the second are essentially redundant. Hence, the above difference equation for the propagated error in the corrector alone is adequately representative, for practical purposes, of the error propagation for the recommended mode, p-d-c-d-c.

If the difference Eq. (12) has constant coefficients, its solution ϵ_n can be expressed in terms of the roots ρ_i of the polynomial equation

$$\rho^3(1 - \lambda d_{-1}) - \rho^2(1 + \lambda d_0) - \rho\lambda d_1 - \lambda d_2 = 0 \tag{13}$$

by $\epsilon_n = k_1\rho_1{}^n + k_2\rho_2{}^n + k_3\rho_3{}^n$ (slightly modified in the case of a multiple root), where the k_i are constants. Equations (12) has constant coefficients as required provided the d_i are constant, as well as λ. The d_i are constant in the case of fixed mesh. In the variable mesh case, it is this investigator's experience that the d_i vary

very slowly when the integration is stability limited. This is due to the fact that the ratio α of mesh increments from step to step remains nearly constant, and the d_i are constant when the mesh parameters α, β, and γ are constant. (When α is constant, β and γ are the constants $\alpha + \alpha^2$ and $\alpha + \alpha^2 + \alpha^3$, respectively.) Thus, it is reasonable to add the assumption of constant d_i, for the purposes of the stability analysis only, and in view of the above remarks it becomes convenient to treat numerical stability in terms of the two parameters λ and α.

When $\lambda = 0$, the fundamental root of the characteristic Eq. (13) is unity and the others are zero. When $\lambda \neq 0$, one or both of the latter roots may become larger in modulus than the fundamental root. This is a condition of relative numerical

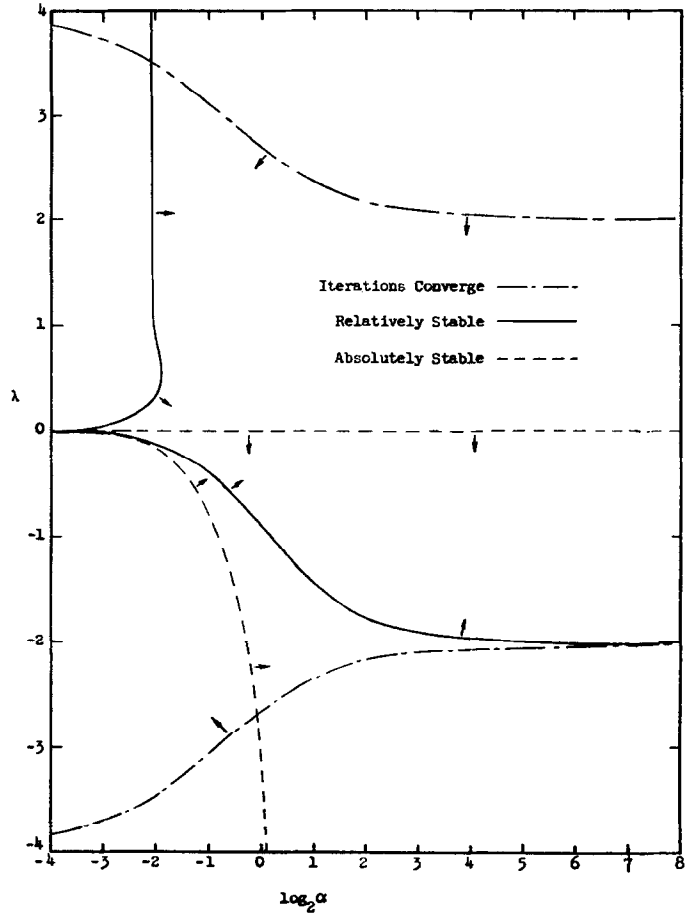

FIG. 1. Regions of stability and convergence of corrector iterations of variable mesh method.

instability [6], whereas absolute numerical instability occurs whenever any root is greater than one in modulus or when a root of unit modulus is a multiple root. Applying these conditions as definitions, regions of both relative and absolute stability have been computed by tracking the roots of (13). These regions are shown in Fig. 1 in terms of the parameters λ and α. Although it is interesting to note the behavior for very large and small α, in practice α actually remains fairly close to unity. Also shown in Fig. 1 are the curves $\lambda d_{-1} = \pm 1$, which indicate the region for which the corrector iterations converge, and within which the stability regions have meaning.

4. Numerical Stability for Systems of Differential Equations

The variable mesh formulas are applicable for systems of differential equations of the form

$$dy^{(i)}/dx = F_i(x, y^{(1)}, y^{(2)}, \ldots, y^{(N)}), \qquad i = 1, 2, \ldots, N. \tag{14}$$

In considering numerical stability for this case, Eq. (12) for the propagated error is replaced by

$$(I - d_{-1}hG)\,\bar{\epsilon}_{n+1} - (I + d_0 hG)\,\bar{\epsilon}_n - d_1 hG\bar{\epsilon}_{n-1} - d_2 hG\bar{\epsilon}_{n-2} = 0, \tag{15}$$

where $\bar{\epsilon}_n$ denotes the vector with components $y^{(i)}(x_n) - y^{(i)}$. I is the identity matrix and G is the Jacobian matrix with elements $G_{ij} = \partial F_i/\partial y^{(j)}$ which are assumed constant, as in the case of a single equation. A cursory analysis of numerical stability is available through consideration of a characteristic polynomial corresponding to a majorization of Eq. (15). However, a more detailed approach involving the eigenvalues of the matrix G has been pursued in the present study.

Pre-multiplying Eq. (15) by a matrix T, representing a nonsingular linear transformation such that $TGT^{-1} = J$ is in canonical form, gives

$$(I - d_{-1}hJ)\,\bar{\eta}_{n+1} - (I + d_0 hJ)\,\bar{\eta}_n - d_1 hJ\bar{\eta}_{n-1} - d_2 hJ\bar{\eta}_{n-2} = 0, \tag{16}$$

where $\bar{\eta}_n = T\bar{\epsilon}_n$. The diagonal elements of J are the eigenvalues of G, and if these are distinct, all the off diagonal elements of J are zero. In this case, the system of difference equations for the propagated errors becomes uncoupled in passing from (15) to (16), and the relevant characteristic polynomial equation is again given by (13), with λ taking on the values hJ_{ii}. If the eigenvalues of G are not distinct, the analysis is more complicated, as indicated in [1], but the results are essentially the same. In either case, however, Fig. 1 is inadequate because some of the J_{ii} may have nonzero imaginary parts.

It is easy to show that the zeros of any polynomial whose coefficients are themselves polynomials in a complex variable λ are the complex conjugates of the zeros

of the same polynomial with λ replaced by its conjugate. Thus, we need only track the roots of (13) for values of λ with positive imaginary parts, the regions of numerical stability in the lower half of the λ plane then being given by symmetry.

The problem of determining regions of stability for fixed α has, thus, been reduced to computing the roots of (13) for incremental values of λ in the upper half λ plane and deciding at each point whether or not we have stability according to some appropriate definition involving the roots. We will limit ourselves to relative stability.

Choosing a definition of relative numerical stability presents an interesting situation. (We ignored this situation in the case of a single differential equation. It was present but rather inconsequential.) One would like a definition which not only provides a unique decision regarding stability at each point but also reflects one's intuitive notions of relative stability. For example, it is distressing to find it possible to pass repeatedly back and forth from stability to instability as $|\lambda|$ increases along some specified path. Two definitions were considered in the present study—one an extension of the Ralston definition used above for single differential equations, and the other a definition used by Crane and Klopfenstein [7] and also by Krogh [8]. Both definitions lead to meaningless relative stability boundaries for fairly large complex λ. As a practical matter, however, it should be remembered that numerical stability is irrelevant for sufficiently large λ since either the truncation error becomes prohibitively large or convergence of the corrector iterations is not obtained.

The generalization of Ralston's definition to apply to systems was considered by Lea [9]. Lea defined the principal root of the characteristic polynomial equation as the continuous function of h satisfying the polynomial equation and taking on the value unity at $h = 0$. All others were called extraneous. Actually, however, this "definition" fails to distinguish between the principal and extraneous roots because two of them may satisfy the requirements of the principal root. The following example illustrates this deficiency and further illustrates the inability to decide between stability and instability for a particular value of λ.

For $\alpha \equiv 1$ (fixed step size) the three roots of Eq. (13) are shown in the ρ plane (Fig. 2). The values corresponding to $\lambda = (-1, 2)$ are indicated by circles. Moving from the origin in the λ plane counterclockwise around the rectangle to $(-1, 2)$, the roots proceed in the ρ plane from the points (1, 0), (0, 0), and (0, 0) to the circled points along the paths indicated by the arrows. The point $\lambda = (-1, 2)$ appears stable according to the Lea definition since the root which started at (1, 0) is the largest. However, as we continue around the rectangle in the λ plane, we see, upon returning to the origin, that the root which started at (1, 0) is now at (0, 0), while one of the roots which started at (0, 0) is now at (1, 0). In other words, if we had proceeded *clockwise* in the λ plane, the point $\lambda = (-1, 2)$ would appear unstable.

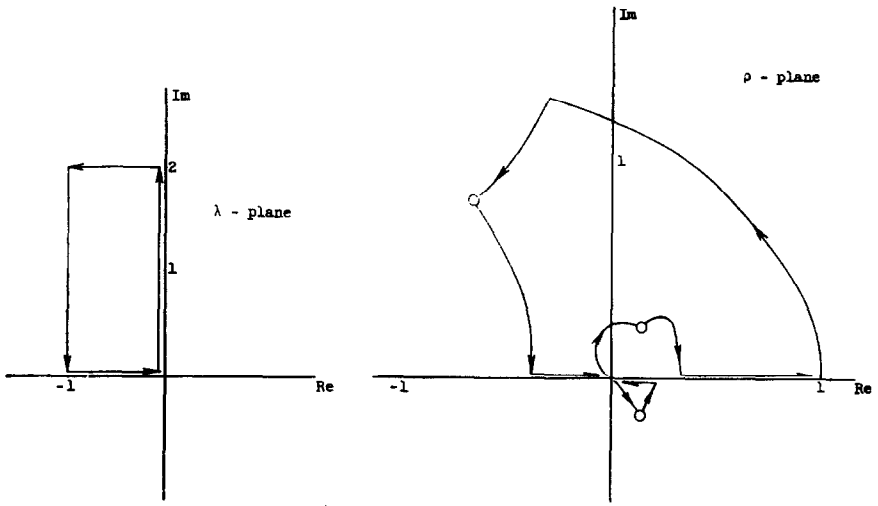

FIG. 2. Example of ambiguity in relative stability definition.

This problem does not develop with small λ; that is, when we consider a somewhat smaller rectangle the roots return to their starting points. On the other hand, the problem does preclude a complete partitioning of the λ plane into meaningful regions of stability and instability by this procedure.

The alternate definition does uniquely partition the λ plane into regions of stability and instability, but these regions are not acceptable for large λ. The problem here, although not recognized in either [7] or [8], is the one mentioned earlier of alternating between stability and instability as λ increases. According to this definition, a method is relatively stable if the modulus of each of the roots, other than the one nearest $\exp(\lambda)$, is less than or equal to $\exp[Re(\lambda)]$, with equality permitted for simple roots only.

To illustrate the problem with this definition we note first that for $\alpha = 1$, the roots of Eq. (13) go from the "source points" (1, 0), (0, 0), and (0, 0), to the "sink points," approximately (−2.37, 0.0), (0.13, −0.17), and (0.13, +0.17), not necessarily respectively, as λ goes from the origin to infinity along any path in the λ plane. Consider now, for example, λ moving along the real axis to (0.5, 0.0) and then vertically to infinity. For the vertical portion, $\exp(\lambda)$ traverses again and again the circle in the ρ-plane with radius $\exp(0.5)$ and center (0, 0). Eventually, when the three roots are sufficiently close to their sink points, they are each nearest $\exp(\lambda)$ for a portion of each cycle of $\exp(\lambda)$. Thus, by definition, the method is relatively stable for the portion of each cycle when the root near (−2.37, 0.0) is the closest to $\exp(\lambda)$ and unstable otherwise. In this manner, on the vertical line

$\mathrm{Re}(\lambda) = 0.5$, we have stability up to $\mathrm{Im}(\lambda) = 3.0$, then instability to about 8.3, stability again to about 9.9, etc.

Since the second definition has the practical advantage that its application is independent of path in the λ plane, and since the problem just noted apparently occurs only for excessively large λ, there is no practical difficulty in its usage: one simply ignores stable regions lying "outside" unstable regions.

Consequently the results shown in Fig. 3 were obtained by applying the second definition. The two definitions give very similar results for small λ and reasonable values of α, say $\frac{1}{4} < \alpha < 4$.

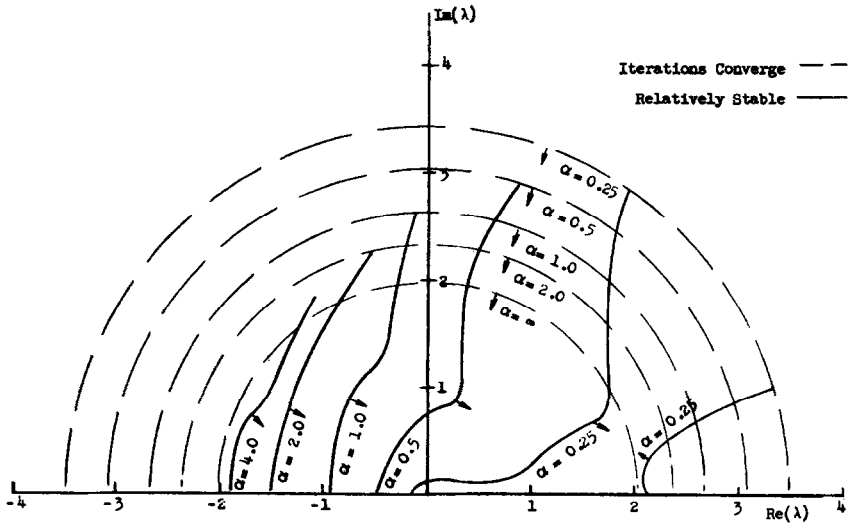

FIG. 3. Regions of relative stability and convergence of corrector iterations of variable mesh method for systems of differential equations.

Also shown in Fig. 3 are the curves $| \lambda d_{-1} | = 1$. In a manner analogous to the case of a single differential equation, it can be shown that for the dominating eigenvalue of the Jacobian matrix of the system, the condition $| \lambda d_{-1} | < 1$ is necessary for convergence of the corrector iterations.

5. CRITERIA FOR SELECTING MESH INCREMENTS

An algorithm for the solution of differential equations by variable mesh procedures would be incomplete without a reasonably sound, general purpose criteria for deciding what step size to use at each step of the integration. The main informa-

tion required for specifying effective criteria was developed in the previous sections. In essence, the mesh selection procedure discussed below represents an attempt to choose each step size just small enough so that the following three criteria are satisfied in the numerical solution of single differential equations:

(a) The relative truncation error must remain within a prescribed tolerance δ.

(b) The condition for convergence of the corrector iterations must be satisfied.

(c) The method must possess relative numerical stability.

Let p_{n+1} and c_{n+1} denote the predicted and final corrected approximations for $y(x_{n+1})$. Let $H = x_{n+2} - x_{n+1}$ be the step size to be used in computing the solution at x_{n+2}, and let α_t be the new value of α as determined by the truncation error criterion in a manner described below. (Thus, from the truncation error criterion we will get $H = h/\alpha_t$.)

Using the truncation error terms for the predictor and corrector formulas obtained in Section 2, we can eliminate the factor $(h^5/5!)\, y_n^v$ and obtain, through fifth order in h, the equation

$$y(x_{n+1}) - c_{n+1} = \left(\frac{c_{n+1} - p_{n+1}}{P_n - C_n}\right) C_n ,$$

where P_n and C_n are given by Eqs. (8) and (9), respectively. We want to find α_t such that the relative error in c_{n+2} is δ, that is,

$$| y(x_{n+2}) - c_{n+2} | = \delta \, | y(x_{n+2}) | .$$

In practice, we actually set

$$\left| \left(\frac{c_{n+1} - p_{n+1}}{P_n - C_n}\right) \frac{C_n}{h^5} \right| H^5 = \delta \, | c_{n+1} |,$$

giving

$$\alpha_t = \left| \frac{C_n(c_{n+1} - p_{n+1})}{\delta c_{n+1}(P_n - C_n)} \right|^{1/5}, \qquad c_{n+1} \neq 0. \tag{17}$$

If $c_{n+1} = 0$, absolute rather than relative truncation error must be considered, the allowable tolerance depending on the range of the machine.

Criteria b an c above are combined to produce a single value α_c for the mesh parameter α at the new step. To this end, we solve $H = h/\alpha_c$ simultaneously with expressions approximating the boundary of the intersection of the regions of

relative stability and iteration convergence shown in Fig. 1. For this purpose, the following expressions have been found to fit the boundary data accurately:

$$\begin{aligned}
F_y < 0: \quad & 0 < \alpha_c < .25: \quad HF_y = -3.2\alpha_c^{5/2} \\
& .25 \leqslant \alpha_c < 1.0: \quad HF_y = .17 - 1.09\alpha_c \\
& 1.0 \leqslant \alpha_c < \infty: \quad HF_y = 1.08/\alpha_c - 2, \\
F_y > 0: \quad & \alpha_c < .25: \quad \text{not permitted (see Fig. 1)} \\
& .25 \leqslant \alpha_c \leqslant 1.0: \quad HF_y = \tfrac{4}{3}[2 + (1 - \alpha_c)^{7/4}] \\
& 1.0 < \alpha_c < \infty: \quad HF_y = 2 + \frac{2}{3\alpha_c}.
\end{aligned}$$

An approximation for F_y can be obtained from computations from the completed step

$$F_y \approx \frac{F(x_{n+1}, p_{n+1}) - F(x_{n+1}, c_{n+1})}{p_{n+1} - c_{n+1}}.$$

For the case $F_y > 0$, $.25 \leqslant \alpha_c \leqslant 1.0$, an iterative scheme is used to solve for α_c:

$$\alpha_c^{(i+1)} = \frac{3hF_y}{8 + 4(1 - \alpha_c^{(i)})^{7/4}}.$$

It has been determined (by actual calculations) that with $\alpha_c^{(0)} = hF_y/3$, $\alpha_c^{(1)}$ is always correct to within two units in the second decimal place. Thus, α_c is computed according to the following simultaneous solutions of each of the above equations with the equation $H = h/\alpha_c$:

$$\begin{aligned}
-\infty < hF_y \leqslant -.92: \quad & \alpha_c = (1.08 - hF_y)/2, \\
-.92 < hF_y \leqslant -.025: \quad & \alpha_c = (.17 + \sqrt{.03 - 4.36hF_y})/2.18, \\
-.025 < hF_y < 0: \quad & \alpha_c = (-hF_y/3.2)^{2/7}, \\
0 \leqslant hF_y \leqslant .875: \quad & \alpha_c = .25, \\
.875 < hF_y < 8/3: \quad & \alpha_c = 3hF_y/[8 + 4(1 - hF_y/3)^{7/4}], \\
8/3 \leqslant hF_y < \infty: \quad & \alpha_c = (hF_y - 2/3)/2.
\end{aligned}$$

The new step size H can then be taken as h/α, where $\alpha = \max(\alpha_t, \alpha_c)$. Although this policy has proven satisfactory in practice, it is possible that it could produce a new step size which is substantially different from the preceding one (but not likely because of the contracting character of the fifth root), and this in turn could result in a subsequent loss of accuracy. Therefore, the writer recommends the

addition of a precautionary restriction, such as $\frac{2}{3} \leqslant \alpha \leqslant \frac{3}{2}$, using a smaller or larger interval depending on the requirements of the particular problem being solved.

Experience indicates that when even only a moderate degree of accuracy is required, the numerical solution of most problems is limited by the truncation rather than the stability (or convergence of the iterations) criterion. Of course, it may be that the truncation criterion is limiting the step size by detecting numerical instability of the predictor; we know, for example, that the numerical integration of stiff equations is limited by stability. At any rate, when we begin to examine mesh criteria for large systems of differential equations, it is fortuitous that satisfying the truncation error criterion usually precludes instability because, in this case, the truncation criterion may be the only one which can be feasibly incorporated into the algorithm. For large systems, the amount of computing time required to evaluate either the Jacobian matrix G or its eigenvalues at each step would usually be prohibitive. Of course, for certain small systems it may not be prohibitive, and then the results shown in Fig. 3 can be incorporated in a manner analogous to that given above for obtaining α_c in the case of a single differential equation. This procedure has proved successful for selected systems although it did not alter the mesh increments substantially from those selected by the truncation criterion alone when reasonably small values of δ were used in the latter criterion.

The mesh selection procedure recommended for most large systems, thus, consists of using only the truncation error criterion. Values of α_i^5 are computed from Eq. (17) for each component of the system, and then α is set equal to the fifth root of the largest of these.

6. Numerical Testing and Comparison with Other Methods

The variable mesh multistep method has been tested by applying it to several single differential equations and to several systems of differential equations. This testing has given a fairly thorough demonstration of the effectiveness and reliability of the algorithm. One system of substantial importance for which the variable mesh approach proved especially effective was the heat transfer problem discussed in the next section. Another system, discussed in [10], was a stochastic model of enzymatically controlled cooperative unwinding and template replication of biological macromolecules. Several simpler test problems are listed in Table I.

Most of the test problems in Table I were selected because of their inherent potential, both in the behavior of the solutions and in the behavior of the partial derivatives of the right sides with respect to the dependent variables, for producing numerical difficulties. Some are particularly suited to a variable mesh treatment

while others, nos. 5, 6, 10, and 11 can be solved efficiently with constant mesh increments. In the latter cases, it is important to note that the accuracy obtained by the variable mesh method was about the same as that obtained using constant increments with the same number of steps. This indicates that the variable mesh procedures do not have a degrading effect when they are used unnecessarily.

TABLE I

Problems Used to Test Variable Mesh Method

Problem	Differential Equation(s)	Integration Interval	Initial Value	Exact Solution
1	$y' = -40xy$	$-1 \leqslant x \leqslant 1$	$\exp(-10)$	$\exp(10 - 20x^2)$
2	$y' = (2xy)^{-1}$	$1 \leqslant x \leqslant 10^{20}$	0	$\sqrt{\ln(x)}$
3	$y' = y/x - (1/x)\cos(1/x)$	$-1 \leqslant x \leqslant -0.01$	$\sin(1)$	$x \sin(1/x)$
4	$y' = -\exp(x)\, y$	$0 \leqslant x \leqslant 5$	$\exp(-1)$	$\exp[-\exp(x)]$
5	$y' = -y$	$0 \leqslant x \leqslant 10$	1	$\exp(-x)$
6	$y' = y$	$0 \leqslant x \leqslant 10$	1	$\exp(x)$
7	$y' = -y/z$	$0 \leqslant x \leqslant 5$	$\exp(-1)$	$\exp[-\exp(x)]$
	$z' = -z$		1	$\exp(-x)$
8	$y' = z(y/z + 1)$	$0 \leqslant x \leqslant 5$	$\exp(-1)$	$\exp[x - \exp(x)]$
	$z' = y$		$\exp(-1)$	$\exp[-\exp(x)]$
9	$y' = y^2/z - 40z$	$-1 \leqslant x \leqslant 1$	$40 \exp(-10)$	$-40x \exp(10 - 20x^2)$
	$z' = y$		$\exp(-10)$	$\exp(10 - 20x^2)$
10	$y' = -2(y + z)$	$0 \leqslant x \leqslant 100$	0	$-2 \exp(-x) \sin(x)$
	$z' = y$		1	$\exp(-x)[\sin(x) + \cos(x)]$
11	$y' = -\exp(-x) - 100z$	$0 \leqslant x \leqslant 1.5$	2	$\exp(-x) + \exp(-100x)$
	$z' = -100z$		1	$\exp(-100x)$
12	$y' = -z/x^4$	$-1 \leqslant x \leqslant -0.01$	$\cos(1) - \sin(1)$	$\sin(1/x) - (1/x)\cos(1/x)$
	$z' = y$		$\sin(1)$	$x \sin(1/x)$

Each equation was solved on the IBM System 360 using single precision starting values and double precision floating point operations to advance the solution. Values of δ, the target relative truncation error, ranging from 10^{-6} to 10^{-1} were used for each equation. The accuracy obtained was roughly proportionate to the values of δ specified. It was noted that the step lengths were limited almost entirely by the truncation error for the smaller values of δ with the stability/convergence criterion becoming of increasing importance with increasing δ.

Some of these problems, 1, 9, and 12, were used in comparing the new algorithm with other fourth order numerical methods which also permit some variability in the mesh increments. The other methods used were the standard fourth order Runge-Kutta method, the Nordsieck method, and the basic Adams-Bashforth/Adams-Moulton method, allowing doubling and halving of the incre-

ments with the latter. As indicated below, the new method proved superior to the other methods for these problems.

Since the Runge-Kutta method requires four derivative evaluations per step while the others were used with only two evaluations per step, half as many steps were used with the Runge-Kutta method as with the other three. For this method the step sizes were obtained by linear interpolation of an input table, reapplying the method with different tables until no improvement could be obtained.

The Nordsieck method permits increasing (or decreasing) the step size by a factor θ (or $1/\theta$). The test problems used in the present study were solved with $\theta = 2$, the value emphasized in [3] where the symbol "β" is used for this factor, and also with smaller values to permit more gradual varying of the increments. In addition, Nordsieck's interval control mechanism requires a parameter "e" used in a manner to imply a target error θ^{-e}. For each value of θ, the problems used here were solved with several values of e, seeking one which produced the number of steps commensurate with the number used by the other methods. However, for $\theta = 2$, the Nordsieck method used too many steps even when e was reduced to unity. (In fact, considerable difficulty was encountered in trying to locate values of θ which were usable in this sense. Successful choices are indicated in Table II.) It is also noted here that it was not necessary to use Nordsieck's starting procedure for the test problems since all the required initial information was available.

TABLE II

Comparison of Relative Error

Problem	Variable Mesh	Adams	Runge-Kutta	Nordsieck
1	1.5×10^{-4}	2.3×10^{-4}	2.8×10^{-3}	3.0×10^{-3} ($\theta = 1.01$)
9	1.8×10^{-3}	6.0×10^{-3}	5.2×10^{-2}	1.7×10^{-2} ($\theta = 1.01$)
12	2.0×10^{-2}	5.5×10^{-2}	1.0×10^{-1}	6.5×10^{-1} ($\theta = 1.5$)

For Problem 1, the absolute value of the relative error in the solution obtained by each of the four methods is shown in Fig. 4. For this problem, the entries in Table II are the areas under the curves of Fig. 4. For the other two problems, the entries in Table II reflect alternative measures of relative error which are more appropriate for the numerical solutions obtained for those two systems of equations. As can be seen from the table, the new variable mesh method gave the best performance; and the basic Adams method, augmented with interpolation procedures to permit doubling and halving, also did considerably better than the other two methods.

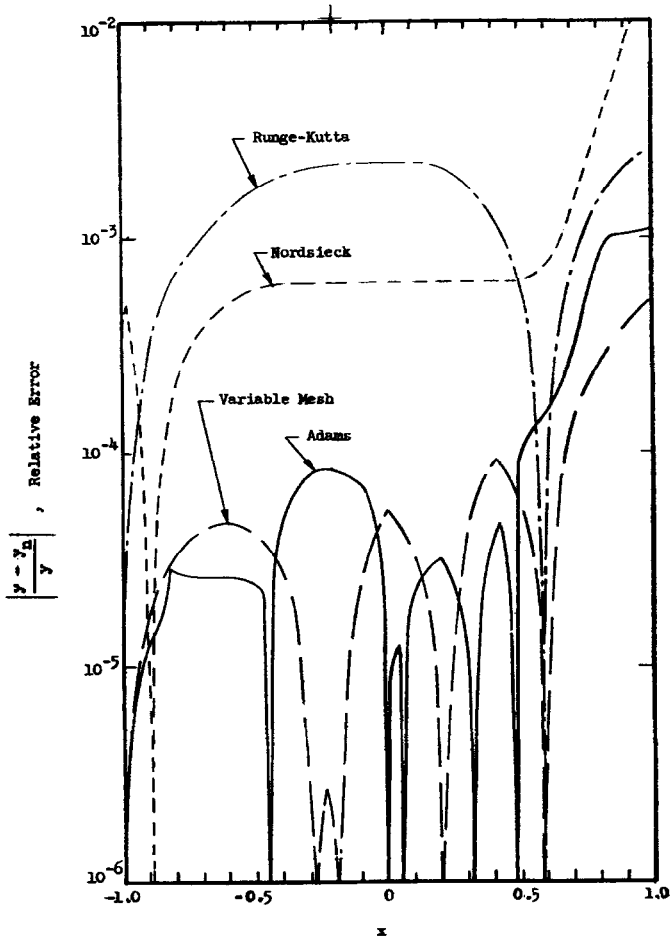

FIG. 4. Relative errors for Problem 1.

7. AN APPLICATION: HEAT TRANSFER TO SUPERCRITICAL HYDROGEN

The numerical method developed in the previous sections has been applied to a problem of heat transfer for fully developed turbulent flow of a supercritical fluid with variable properties in a smooth tube. This problem was first formulated by Deissler [11] who established the solution for the case of water. The results given here are for the case of hydrogen—an important problem in the regenerative cooling process of liquid rocket engines.

Two first order simultaneous ordinary differential equations arise for dimensionless velocity and temperature parameters, u and t, with the independent variable being a dimensionless wall-distance parameter x. Near the wall, $0 \leqslant x < \bar{x}$, the following equations apply:

$$\frac{du}{dx} = \left\{\frac{\mu}{\mu_0} + \frac{\rho}{\rho_0} m^2 ux \left[1 - \exp\left(-\frac{m^2 ux\rho\mu_0}{\rho_0\mu}\right)\right]\right\}^{-1}, \tag{18}$$

$$\frac{dt}{dx} = \left\{\frac{K}{K_0 P_{r_0}} + \frac{\rho}{\rho_0}\frac{C_p}{C_{p_0}} m^2 ux \left[1 - \exp\left(-\frac{m^2 ux\rho\mu_0}{\rho_0\mu}\right)\right]\right\}^{-1}, \tag{19}$$

with

$$u(0) = t(0) = 0.$$

Here, ρ, μ, C_p, K, and P_r are the density, viscosity, specific heat, thermal conductivity, and Prandtl number, respectively, with the subscript "0" indicating the

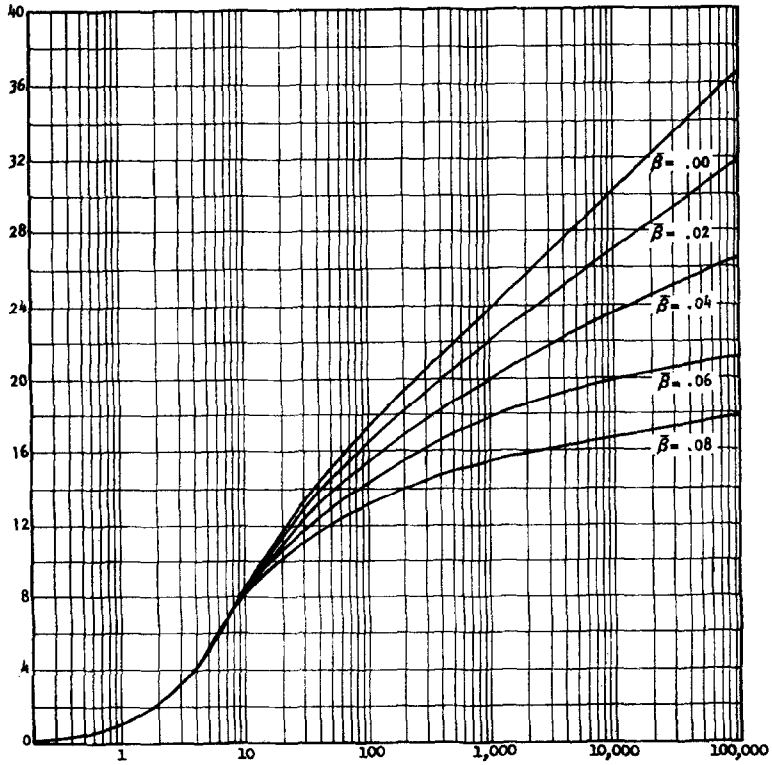

FIG. 5. Velocity parameter versus wall-distance parameter.

values at the wall temperature T_0. (Temperature is related to the dimensionless temperature parameter t by the equation $T = T_0(1 - \bar{\beta}t)$, where $\bar{\beta}$ is a dimensionless heat-transfer parameter.) The empirical constant m was taken as 0.124.

In the turbulent core, $\bar{x} \leqslant x \leqslant x_L$, Eqs. (18) and (19) are replaced by the equations

$$\frac{du}{dx} = \left(\frac{du}{dx}\right)_{x=\bar{x}} \exp\left[-\kappa \int_{\bar{x}}^{x} \left(\frac{du}{d\xi} \Big/ \sqrt{\frac{1 - \frac{\mu}{\mu_0}\frac{du}{d\xi}}{\rho/\rho_0}}\right) d\xi\right], \tag{20}$$

and

$$\frac{dt}{dx} = \frac{du}{dx} \Big/ \left[\frac{C_p}{C_{p_0}} + \frac{du}{dx}\left(\frac{K}{K_0 P_{r_0}} - \frac{C_p \mu}{C_{p_0}\mu_0}\right)\right], \tag{21}$$

where κ is an empirical constant taken as 0.36 (see [11]).

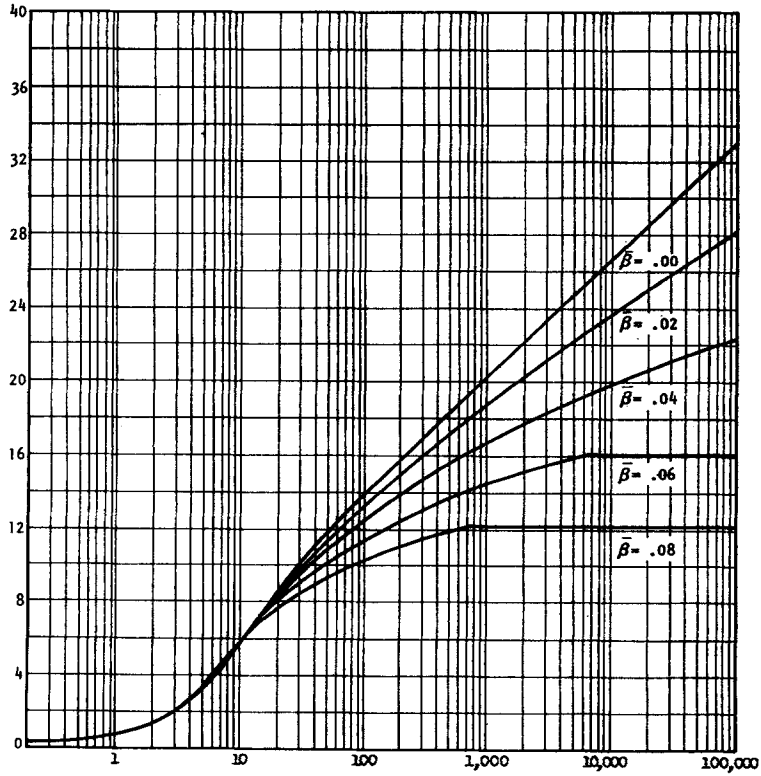

FIG. 6. Temperature parameter versus wall-distance parameter.

Starting values were obtained by the procedure outlined in the introduction. Because the derivative du/dx appears implicitly in Eq. (20), successive substitutions were used for its evaluation. With only that one additional requirement, the variable mesh method was applied exactly as recommended in the preceding sections. The results are given in Figs. 5 and 6. These results are for a wall temperature of $1160°R$ and a pressure of 1000 pounds per square inch, with $\bar{x}$ taken as 26. Note in Fig. 6 that the profile of the temperature parameter t reaches a plateau for the higher values of $\bar{\beta}$. This is due to the fact that the temperature T was not permitted to decrease below the critical temperature for hydrogen (36°R).

The variable mesh method proved remarkably efficient for these integrations. For each value of $\bar{\beta}$, the integration of the equations out to $x_L = 10^5$ was repeated for three specified values of the relative truncation error δ, 0.01, 0.001, and 0.0001. Even though the initial step sizes for these fifteen cases ranged from 0.003 to 3.9, never more than 39 steps were required for $\delta = 0.01$, 53 for $\delta = 0.001$, and 74 for $\delta = 0.0001$.

REFERENCES

1. R. VAN WYK, A variable mesh method for the efficient numerical solution of ordinary differential equations, Rocketdyne Research Report No. 65–26, 1965, presented at the 1966 National SIAM meeting, Iowa City, Iowa.

3. A. NORDSIECK, On numerical integration of ordinary differential equations, *Math. Comp.* **16** (1962), 22–49.
4. H. R. Lewis, JR. AND E. J. STOVALL, JR., Comments on a floating point version of Nordsieck's scheme for the numerical integration of differential equations, *Math. Comp.* **21** (1967), 157–161.
5. T. E. HULL, AND A. L. CREEMER, Efficiency of predictor-corrector procedures, *J. ACM* **10** (1963), 291–301.
6. A. RALSTON, Relative stability in the numerical solution of ordinary differential equations, *SIAM Rev.* **7** (1965), 114–125.
7. R. L. CRANE AND R. W. KLOPFENSTEIN, A predictor-corrector algorithm with an increased range of absolute stability, *J. ACM* **12** (1965), 227–241.
8. F. T. KROGH, Predictor-corrector methods of high order with improved stability characteristics, *J. ACM* **13** (1966), 374–385.
9. R. N. LEA, On the stability of numerical solutions of ordinary differential equations, NASA Technical Note No. D-3760, 1967.
10. J. M. ZIMMERMAN AND R. SIMHA, The kinetics of cooperative unwinding and template replication of biological macromolecules, *J. Theor. Biol.* **13** (1966), 106–130.
11. R. G. DEISSLER, Heat transfer and fluid friction for fully developed turbulent flow of air and supercritical water with variable fluid properties, *ASME Trans.* **76** (1954), 73–85.