NASA TECHNICAL NOTE

NASA TN D-4547

ON THE CONSTRUCTION OF HIGHLY STABLE, EXPLICIT, NUMERICAL METHODS FOR INTEGRATING COUPLED ORDINARY DIFFERENTIAL EQUATIONS WITH PARASITIC EIGENVALUES

by Harvard Lomax

Ames Research Center
Moffett Field, Calif.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1968

ON THE CONSTRUCTION OF HIGHLY STABLE, EXPLICIT, NUMERICAL METHODS FOR INTEGRATING COUPLED ORDINARY DIFFERENTIAL EQUATIONS WITH PARASITIC EIGENVALUES

By Harvard Lomax

Ames Research Center
Moffett Field, Calif.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151 – CFSTI price $3.00

ON THE CONSTRUCTION OF HIGHLY STABLE, EXPLICIT, NUMERICAL METHODS FOR INTEGRATING COUPLED ORDINARY DIFFERENTIAL EQUATIONS WITH PARASITIC EIGENVALUES

By Harvard Lomax

Ames Research Center

SUMMARY

A theory is developed for constructing explicit numerical methods for integrating coupled nonlinear ordinary differential equations with local eigenvalues that are greatly separated in magnitude. Applications are made to cases in which large negative eigenvalues are combined with small complex ones. The specific methods derived are compared with Runge-Kutta methods. The derived methods are not considered to be optimum and further improvements are anticipated.

INTRODUCTION

Large sets of coupled ordinary differential equations occur in the analysis of physical problems in a variety of ways. For example, the study of a chemically reacting gas in a one-dimensional flow leads directly to such sets. Less direct examples, but still practical and important ones, occur when partial differential equations are differenced in all but one of the independent variables, providing, thereby, large sets of loosely coupled ordinary differential equations. When these equations are complicated or large in number, it may be advantageous to adapt special numerical methods to their solution. This may be especially true if they contain "parasitic eigenvalues," a term which will be defined later on.

One purpose of this report is to present an approach that can be used to construct special numerical methods which will minimize computational time in certain special cases. For these methods to be truly beneficial, certain properties of the differential equations should be known a priori. For example, with a set of differential equations we will associate a matrix, and sometimes the eigenvalues of this matrix can be guaranteed to fall in special categories regardless of the details of the solutions to the differential equations. Thus, for one reason or another, we might know before hand that the eigenvalues of the associated matrix are always real, or always imaginary. Special methods exist which are optimum for either case, but a method designed for one might be unsatisfactory if used for the other.

The development of these special methods is still in a state of flux. The particular predictor-corrector formulas presented herein can be improved upon for a variety of reasons that are discussed as their development proceeds. Thus, the approach to their construction is more important than the details of their structure, and improved forms are anticipated for the future.

If one has no a priori knowledge about the differential equations, and wishes to use an explicit differencing scheme, the standard, fourth-order, Runge-Kutta method is highly recommended for general use. There are a variety of reasons for this recommendation. The specific ones regarding parasitic eigenvalues are discussed in a section entitled "Stability Polynomials."

SYMBOLS

$[\ ]$	matrix of enclosed quantity
$[\ ]^{-1}$	inverse of matrix
$[A_n]$	matrix in locally linearized equations
a_j	coefficients in stability polynomial (eq. (32))
det( )	determinant of enclosed quantity
E	the operator $e^{hd/dt}$, $E^k u_n = u_{n+k}$
er_p	truncation error in numerical method (eq. (11))
er_t	truncation error in local linearization (eq. (2a))
$\vec{F}_n$	function that determines magnitude of derivative
$\vec{f}_n$	See equation (2b).
H	effective distance that a numerical method advances the integration after time for two evaluations of the derivatives
h	step size used in the numerical integration
[I]	unit matrix
n	number of steps
t	independent variable
u_n	dependent variable in uncoupled form
$\vec{w}_n$	dependent variables in coupled form
$\vec{w}'_n$	$d\vec{w}_n/dt$

α,β,γ	See equation (22).
δ	See equation (27).
λ	eigenvalues of difference equations, $\lambda = \lambda(\sigma h)$
σ	eigenvalues of $[A_n]$ in differential equations, $\bar{\sigma}e^{i\theta}$
$\bar{\sigma}$	real number
$\lvert\sigma h\rvert_c, \lvert\sigma H\rvert_c$	induced stability boundary referred to calculation step h, and effective step H, respectively

Superscripts

$\rightarrow$	vector
T	transpose of vector

THE ASSOCIATED MATRIX

The General Case

Consider the set of m nonlinear, coupled, ordinary differential equations

$$w_i' \equiv \frac{dw_i}{dt} = F_i(t;w_1,w_2, \ldots, w_m) \qquad i = 1,2, \ldots, m \tag{1}$$

or

$$\vec{w}' = \vec{F}(t;\vec{w})$$

If each F_i is expanded about a local point referenced as n, where $t = nh$, and h is a small step interval

$$w_i' = F_{in} + (w_1 - w_{1n})\left(\frac{\partial F_i}{\partial w_1}\right)_n + \ldots + (w_m - w_{mn})\left(\frac{\partial F_i}{\partial w_m}\right)_n$$

$$+ \text{terms involving products of the various } (\vec{w} - \vec{w}_n)$$

Let the elements $(a_{ij}(t))_n$ of a matrix $[A_n(t)]$ be $\partial F_i/\partial w_j$. Since in the interval $h(n + 1) \geq t \geq nh$

$$\vec{w} - \vec{w}_n = h[(\vec{w} - \vec{w}_n)/h] = h\vec{w}_n' + O(h^2)$$

the terms involving the products of $(\vec{w} - \vec{w}_n)$ are $(1/2)(hw_n')^2(\partial^2 F_i/\partial w_j^2) + O(h^3)$ and equation (1) can be expressed as

$$\vec{w}' = [A_n(t)]\vec{w} + \vec{f}_n(t) + h^2\vec{er}_t \tag{2a}$$

where

$$\vec{f}_n = \vec{F}_n(t) - [A_n(t)]\vec{w}_n \tag{2b}$$

It is assumed throughout that

(a) The $\partial F_i/\partial w_j$ are continuous for all i and j
(b) $\vec{er}_t$ is bounded as $h \to 0$.

We refer to $[A_n(t)]$ as the local associated matrix.

Autonomous Equations

In many practical problems the right hand side of equation (1) does not depend explicitly on the independent variable t. In such cases, the equations are called autonomous and special numerical methods can be constructed to solve them. From the viewpoint of the applied mathematician who tries to make maximum use of the physical structure of his problem, there are two ways in which ordinary differential equations arising from the study of physical phenomena turn out to be autonomous. In one (the "natural" way), the derivation of the equations results in forms that contain no explicit dependency on the independent variable for physical reasons. In the other (the "artificial" way), the equations are made nonautonomous by a mathematical transformation. We next illustrate the simplest of such transformations as it applies to a representative nonautonomous equation. The example will prove to be useful later when we examine the conditions that must be satisfied to insure the accurate integration of autonomous equations by means of special methods designed for them.

Consider the representative, linear, nonautonomous equation

$$\frac{dw}{dt} = bw + e^{\mu t}$$
$$w(0) = 0 \tag{3}$$

where b and μ are constants. A nonlinear, autonomous set is formed by the transformations

$$w_1 = w \ , \quad w_2 = t$$

giving

$$\left.\begin{aligned} \frac{dw_1}{dt} &= bw_1 + e^{\mu w_2} \\ \frac{dw_2}{dt} &= 1 \\ w_1(0) &= w_2(0) = 0 \end{aligned}\right\} \tag{4}$$

which is in the form of equation (1). The expansion corresponding to equation (2) results in

$$\vec{w}' = [A_n]\vec{w} + \vec{f}_n + h^2\vec{er}_t \tag{5}$$

where

$$[A_n] = \begin{bmatrix} b & (\mu e^{\mu w_2})_n \\ 0 & 0 \end{bmatrix} \tag{6a}$$

$$\vec{f}_n = \begin{bmatrix} (e^{\mu w_2})_n - (\mu e^{\mu w_2})_n w_{2n} \\ 1 \end{bmatrix} \tag{6b}$$

and, since $w_2' = 1$,

$$h^2\vec{er}_t \approx \begin{bmatrix} (1/2)h^2(\mu^2 e^{\mu w_2})_n \\ 0 \end{bmatrix} \tag{6c}$$

Now if we approximate equation (5) by

$$\vec{w}' = [A_n]\vec{w} + \vec{f}_n$$

we have, in the step n to n + 1, a set of linear, coupled, differential equations with constant coefficients and constant values of $\vec{f}_n$. The fact that $\vec{f}_n$ and the coefficients in the associated matrix are constant is important and follows directly from the fact that the nonlinear equations from which they were derived were autonomous.

Role of the Associated Matrix

We have seen how the nonlinear equations

$$\vec{w}' = \vec{F}(\vec{w}) \tag{7}$$

can be approximated by

$$\vec{w}' = [A_n]\vec{w} + \vec{f}_n , \qquad nh \leq t \leq (n+1)h \tag{8}$$

with an error in the derivative proportional to h^2. When using implicit numerical methods, one actually puts equation (7) in the form of (8) and then numerically integrates the latter (see ref. 1). Calculating all the elements in $[A_n]$ can be quite troublesome, however, and a principal motivation for this report was an attempt to extend the range over which these calculations can be avoided while maintaining stable and accurate results. Therefore we are searching for methods that can be applied directly to equation (7) and not to equation (8). Some of the conditions under which this is possible when there are parasitic eigenvalues in the associated matrix are discussed in a section on nonlinear effects (p. 32). Such being the case, one may ask: Why study the associated matrix if it is not to be used in the numerical integration? The answer is that $[A_n]$ plays a fundamental role in constructing methods suitable for the direct differencing of equation (7). We now give two reasons why this is so. It follows from the study of linear autonomous equations that

1. If proper numerical techniques are employed, the approximate magnitudes of the maximum eigenvalues in the associated matrix can be automatically generated from the information carried in the solutions provided by the direct differencing of equation (7). This is demonstrated in the section beginning on page 34.

2. If, in a given step, all the differential equations are differenced by the same numerical method, the stability of the numerical integration (neglecting the effects of roundoff) is completely determined by the product of the step size and these same eigenvalues (it is otherwise independent of the size of the individual elements in $[A_n]$). Proof of this is given in reference 2.

For nonlinear equations these statements are valid locally insofar as equation (8) represents equation (7). We further hypothesize that local stability implies global stability. Practical experience has led to the general acceptance of this hypothesis except for singular cases.

A DISCUSSION OF STABILITY AND ACCURACY

Stability

The concepts of stability are very well known and are briefly reviewed here to establish the terminology, which is not universal. (For a more detailed discussion of this and the following material see ref. 2.) The matrix associated with a set of m simultaneous equations, of the form given by equation (8), has m eigenvalues which are designated σ_k, $k = 1,2, \ldots, m$. The eigenvalues are generally complex and may or may not be distinct. A set of linear autonomous differential equations is said to be inherently stable if all the eigenvalues σ_k are distinct and none has a positive real part. They are also inherently stable if all the eigenvalues have negative real parts, although cases can be constructed with multiple eigenvalues for which this is academic. If some of the eigenvalues are imaginary and multiple, the equations have a degenerate instability. We use the convenient terminology that a solution is more or less stable depending on whether the real parts of σ_k are more or less negative.

If the differential equations are inherently stable, but their numerical solution is not stable, the numerical method used is said to give an induced instability. This phenomenon has been studied extensively and is fundamental to the analysis presented herein. If equation (8) is differenced by some scheme, the resulting difference equations have some matrix associated with them and we designate the eigenvalues of this matrix by λ_{jk}. For most so-called one-step methods j is equal to 1, but for multistep methods, there can be several values of j for each k. Each λ_{jk} is some function of $h\sigma_k$. The solution to the difference equations depends, after n steps, on $(\lambda_{jk})^n$, so the necessary and sufficient condition for a numerical method to induce no instability is that all $|\lambda_{jk}| < 1$ (or, if there are no multiple roots, that $|\lambda_{jk}| \le 1$).

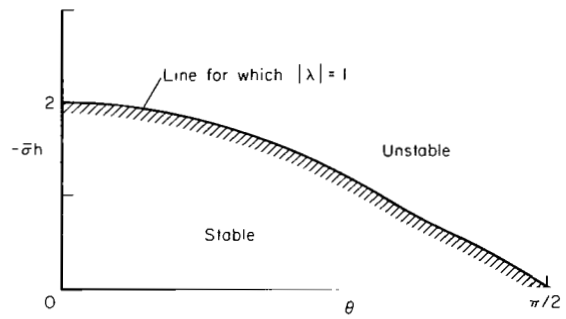
Two types of induced instability of general interest and frequently discussed in the literature are when $\sigma_k h \to 0$ and when $\sigma_k h \neq 0$. The former is called asymptotic stability (see ref. 3) and is concerned with whether or not any of the $|\lambda_{jk}|$ are greater than 1 when $h = 0$. The well-known Dahlquist theorem (see ref. 4) is valid for this kind of stability. The second type is more practical for our purposes and permits us to define an induced stability boundary which we designate by $|\sigma h|_c$.

Let $\sigma = \bar{\sigma} e^{i\theta}$ where $\bar{\sigma}$ is real. Then we can speak of two special induced stability boundaries for any given numerical method. The real induced stability boundary, $|\bar{\sigma} h|_c$, is that value of $\bar{\sigma} h$ for which any increase in h will cause some $|\lambda|$ to exceed unity. The imaginary induced stability boundary, $(i\bar{\sigma} h)_c$, is that value of $i\bar{\sigma} h$ for which any increase in h will cause some $|\lambda|$ to exceed unity. In general,

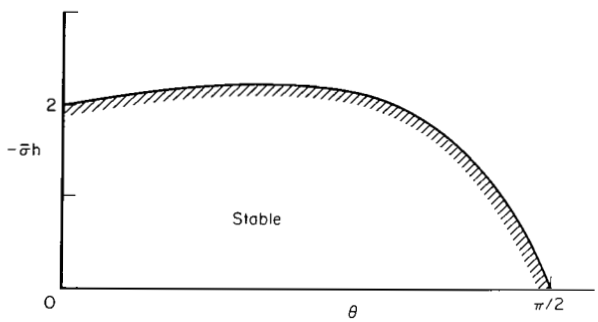
$$\left.\begin{array}{l} |\sigma h|_c \text{ is the value of } |\bar{\sigma} e^{i\theta} h| \text{ in the} \\ \text{interval } 0 \le \theta \le \pi/2 \text{ and } \bar{\sigma} h \le 0 \text{ for} \\ \text{which any increase in } h \text{ causes some} \\ |\lambda| \text{ to exceed unity.} \end{array}\right\} \qquad (9)$$

The simplest example of the above is given by applying Euler's method $w_{n+1} = w_n + hw'_n$ to the equation $w' = \sigma w$. There results $w_{n+1} = (1 + \sigma h)w_n$ so that

$$\lambda = 1 + \sigma h = 1 + \bar{\sigma} e^{i\theta} h$$



Sketch (a).- Euler's method.

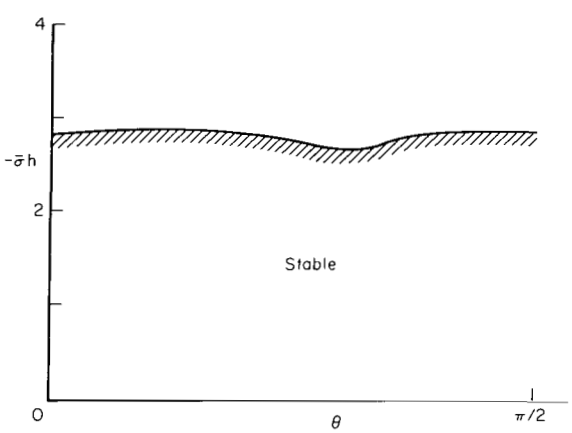


Sketch (b).- Second-order Runge-Kutta.

Sketch (a) shows the curve relating $\bar{\sigma}h$ and θ such that $|\lambda| = 1$. Clearly, Euler's method has a real induced stability boundary equal to 2. However, its imaginary induced stability boundary and, therefore, by definition (9), its general induced stability boundary are both zero. Sketch (b) gives the results for an Euler predictor followed by a modified Euler corrector (the second-order Runge-Kutta method)

$$w_{n+1}^{(1)} = w_n + hw'_n$$

$$w_{n+1} = w_n + \frac{1}{2} h(w_{n+1}^{(1)'} + w'_n)$$



Sketch (c).- Fourth-order Runge-Kutta.

Once again $|\bar{\sigma}h|_c$ is 2, but both $(i\bar{\sigma}h)_c$ and $|\sigma h|_c$ are zero. Sketch (c) illustrates the results for the fourth-order Runge-Kutta method, here $|\sigma h|_c \approx 2.6$.

Methods that have a finite induced stability boundary are referred to as being conditionally stable. All explicit methods are conditionally stable. Most, but not all, implicit methods are also conditionally stable.

Driving and Parasitic Eigenvalues

In order to discuss the accuracy of numerical integration methods, we need to be able to distinguish between what we shall call "driving" and "parasitic" eigenvalues in the associated matrix. In many problems requiring numerical integration one seeks to resolve the effects of all the eigenvalues. In such cases they are all referred to as driving eigenvalues, and the value of $|\sigma_k|_{max}$ normally determines the step size. In certain applications, however, (see refs. 1, 5-10) some of the eigenvalues are (relative to the absolute values of others in the coupled set) large negative numbers. The influence of these large negative eigenvalues can be completely negligible on the analytic solution over much of the range of integration. However, they can severely handicap the progress of the numerical solution, when one uses conditionally stable methods, since they force the step size to be unreasonably small due to the induced stability boundary defined in (9). These eigenvalues are called parasitic eigenvalues; and those much smaller in magnitude, the effects of which we do seek to resolve, and which should ideally be the reference for the step size, are again referred to as the driving eigenvalues. We will subsequently develop methods designed for problems with this particular mixture of driving and parasitic eigenvalues.

Clearly problems can occur when imaginary eigenvalues with very large magnitudes are coupled into equations with much smaller $|\sigma_k|$, and we seek to resolve the effects of those small in magnitude when the initial conditions are such that the effects of the large negative ones are (analytically) negligible. Physically, this is the case when we wish to study transient phenomena in the presence of low-amplitude, high-frequency noise. The methods to be described can also be used to develop numerical schemes that are optimum for this kind of problem.

Accuracy

All numerical methods discussed in this report can be identified with the recursive construction of a Taylor series expansion about each discrete point as the solution proceeds. This can also be regarded as a procedure in which a local polynomial is embedded in the data at each point. The accuracy of the polynomials is given by the highest degree exactly matched in the Taylor series expansion. For example, if each term in the modified Euler method

$$w_{n+1} = w_n + \frac{1}{2} h(w'_{n+1} + w'_n) \tag{10}$$

is expanded about the point n, one can easily show that the first nonzero term is $-(1/12)h^3 w'''_n$. Hence for the modified Euler method we can write for a continuous function of t

$$w_{n+1} = w_n + \frac{1}{2} h(w'_{n+1} + w'_n) + h^3 er_p \tag{11}$$

and combining equations (8) and (11), we find for $nh \geq t \geq (n+1)h$

$$\vec{w}_{n+1} = \vec{w}_n + \frac{1}{2} h[A_n](\vec{w}_{n+1} + \vec{w}_n) + hf_n + h^3(er_t + er_p) \tag{12a}$$

From this one derives the implicit method used in reference 1:

$$\left([I] - \frac{1}{2} h[A_n]\right)(\vec{w}_{n+1} - \vec{w}_n) = h\vec{F}_n + O(h^3) \tag{12b}$$

This method is unconditionally stable and is recommended when very large parasitic eigenvalues are present. It has the disadvantage, as has been pointed out, that the elements in the matrix $[A_n]$ must all be evaluated at each step.

We are now prepared to discuss the accuracy of the numerical methods that are derived in the following sections. These methods are all explicit and are all constructed so as to match a Taylor series expansion through the second-degree term; that is, they all have errors that can be represented by h^3er_p, just as the modified Euler method. For our purposes, methods with higher order accuracy are not justified because we are bound by the error term h^3er_t in the expansion that gives equation (8) from equation (7).

Let σ_p be the maximum (largest in absolute value) parasitic eigenvalue. Then, to provide an accurate numerical solution to equation (7), we must be sure that

$$|h\sigma_p| < |\sigma h|_c \tag{13a}$$

$$|h^3(er_p + er_t)| < \epsilon \tag{13b}$$

where $|\sigma h|_c$ is the induced stability boundary defined in (9), and ϵ represents the maximum truncation error one can tolerate. The methods to be developed can, with care (see the section entitled "The Largest Parasitic Eigenvalue"), be programmed so as to detect σ_p. Such being the case, by varying h (which, for all methods considered, can be changed after each step) the stability can be assured. The accuracy is then bounded by the truncation error in the two expansions. The best way to control this error appears to be by

1. testing the variation of the actual solutions

2. adjusting the step size according to the amount of this variation unless h is already limited by the stability.

In order to illustrate some of the above comments, consider the numerical integration of equations (4). From equation (6a), we see that the two eigenvalues in the associated matrix are zero and b. Hence, stability is assured if $|hb| < |h\sigma|_c$. If $b = -100$, $\mu = 1$, and $t > 1$, one seeks, for accuracy, to make $\mu h \approx 0.1$; and, for stability, to find a method for which $|\sigma h|_c > 10$. On the other hand, if $b = -1$, $\mu = 100$, and $t > 0$, accuracy would demand that $h \approx 0.001$ and stability would be no problem.

IMPLICIT AND EXPLICIT METHODS

The Representative Equation

Although all methods discussed in this report are intended for use on sets of coupled ordinary differential equations, their accuracy and stability can be classified according to how they reproduce the exact solution of the simple representative differential equation

$$w' = \sigma w + ae^{\mu t} \tag{14}$$

For further discussion of this point see reference 2.

Implicit Methods

If the modified Euler method (known also as the trapezoidal rule, or, in the study of parabolic partial differential equations, as the Crank-Nicholson method) is applied to the representative equation (14),

$$w_{n+1} = w_n + \frac{1}{2}\sigma h(w_{n+1} + w_n) + \frac{1}{2}ahe^{\mu hn}(e^{\mu h} + 1) \tag{15}$$

In operational notation ($E \equiv e^{hd/dt}$, $E^k w_n = w_{n+k}$) equation (15) becomes

$$\left[E\left(1 - \frac{1}{2}\sigma h\right) - 1 - \frac{1}{2}\sigma h\right] w_n = \frac{1}{2}ahe^{\mu hn}(e^{\mu h} + 1)$$

There is only one root to the characteristic equation

$$\left(1 - \frac{\sigma h}{2}\right)E - \left(1 + \frac{\sigma h}{2}\right) = 0 \tag{16}$$

which is

$$\lambda = \frac{1 + (\sigma h/2)}{1 - (\sigma h/2)} \tag{17}$$

For small σh, the solution to the homogeneous equation reduces to $w_n = c(\lambda)^n \approx (1 + \sigma h + (1/2)\sigma^2 h^2 + (1/4)\sigma^3 h^3 + \cdots)^n$ which represents an expansion of the exact solution, $w_n = c(e^{\sigma h})^n$, through the term with h^2. For large values of $|\sigma h|$, the solution is completely inaccurate but

$$\left|\frac{1 + (\sigma h/2)}{1 - (\sigma h/2)}\right| < 1 \tag{18}$$

if Real Part $(\sigma) \leq 0$

so it is unconditionally stable.

Another implicit method used in the study of nonequilibrium fluid flow (refs. 5, 10), and boundary-layer theory (ref. 11), is the two-step equation

$$w_{n+2} = \frac{1}{3}\left(4w_{n+1} - w_n + 2hw'_{n+2}\right) \tag{19}$$

Its characteristic equation is

$$\left(1 - \frac{2}{3}\sigma h\right)E^2 - \frac{4}{3}E + \frac{1}{3} = 0 \tag{20}$$

which has the two roots

$$\lambda_1 = \frac{2 + \sqrt{1 + 2\sigma h}}{3 - 2\sigma h}$$

$$\lambda_2 = \frac{2 - \sqrt{1 + 2\sigma h}}{3 - 2\sigma h}$$

Of these, λ_1 is the principal root and λ_2 is spurious. The expansion of λ_1 for small values of σh again coincides with the exact solution through terms with h^2; however, the error in the h^3 term is greater than that for the modified Euler method. The method given by equation (19) is also unconditionally stable. In fact, it is more stable than equation (10), since the roots to equation (20) $\rightarrow \pm 1/\sqrt{\sigma h}$ as $|\sigma h| \rightarrow \infty$. For unconditionally stable methods with higher accuracy (and correspondingly higher step number) see references 1, 5, and 10.

The characteristic equations (16) and (20) are of the form

$$P_k(\sigma h)E^k + P_{k-1}(\sigma h)E^{k-1} + \ldots + P_0(\sigma h) = 0 \tag{21}$$

where each $P_j(\sigma h)$ is a polynomial in σh. All implicit methods have characteristic equations of this form. In these two cases we see by inspection that an essential reason for their unconditional stability is that the leading coefficient $P_k(\sigma h)$ is <u>not</u> equal to 1.

Explicit Methods

The stability of three explicit methods has already been discussed in connection with sketches (a), (b), and (c). The characteristic equation for these and any other explicit method can also always be put in the form of equation (21), with the extremely important reservation that <u>for all explicit methods</u> $P_k(\sigma h) = 1$. Consider next the following:

Theorem: Let $E^k + P_{k-1}(\sigma h)E^{k-1} + \ldots + P_0(\sigma h) = 0$ be a monic (i.e., the coefficient of the highest power of E is 1) polynomial

in E, in which each coefficient, $P_j(\sigma h)$, is a finite polynomial in σh. Designate the roots for the polynomial in E, for some fixed value of σ, by $\lambda_j(\sigma h)$, $j = 1,2, \ldots, k$. If for one λ, $\lim_{h\to o}[\partial\lambda/\partial(\sigma h)] = 1$, then at least one $\lambda \to \infty$ as $h \to \infty$.

Proof: If $\lim_{h\to o}[\partial\lambda/\partial(\sigma h)] = 1$ (a necessary condition for accuracy), at least one coefficient of a nonzero power of σh in some $P_j(\sigma h)$ is not zero. Since all P_j are finite polynomials, then, for fixed σ, at least one $P_j \to \infty$ as $h \to \infty$. Therefore, since all of the coefficients of a monic polynomial can be expressed as sums of various combinations of products of the roots of the polynomial, at least one root $\to \infty$ as $h \to \infty$.

An immediate consequence of this theorem is that all explicit methods are conditionally stable.

EXPLICIT ONE-ROOT METHODS

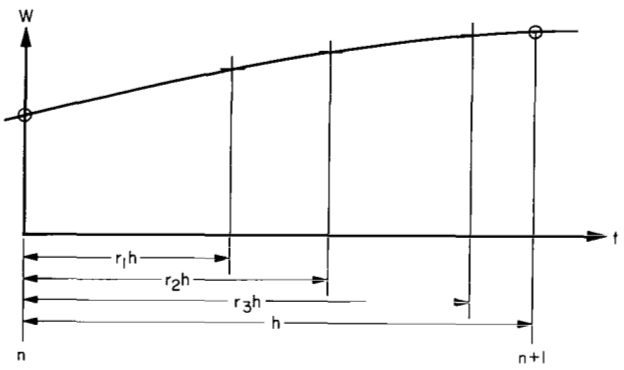
Introduction

Predictor-corrector methods are usually classified according to step number; that is, according to the number of equispaced locations at which previously calculated data are used to advance the solution one additional step in a cycle of computation. A more fundamental classification (at least for our purposes) is the number of roots in the characteristic equation generated by applying the method to the representative equation. The techniques outlined below can be extended to multiroot methods and the latter may have some advantages. However, this aspect is not persued further herein.

A Class of One-Root Methods

Nonautonomous.- Consider the predictor-corrector sequence given by the following:

$$\left.\begin{aligned}
w^{(1)}_{n+r_1} &= \alpha_1 w_n + \beta_1 h w'_n \\
w^{(2)}_{n+r_2} &= \gamma_{12} h w^{(1)\prime}_{n+r_1} + \alpha_2 w_n + \beta_2 h w'_n \\
w^{(3)}_{n+r_3} &= \gamma_{13} h w^{(1)\prime}_{n+r_1} + \gamma_{23} h w^{(2)\prime}_{n+r_2} + \alpha_3 w_n + \beta_3 h w'_n \\
&\vdots \\
w_{n+1} &= \gamma_{1k} h w^{(1)\prime}_{n+r_1} + \gamma_{2k} h w^{(2)\prime}_{n+r_2} + \cdots + \alpha_k w_n + \beta_k h w'_n
\end{aligned}\right\} \qquad (22)$$



Sketch (d)

where r_j are arbitrary weightings of h, the computational step size as shown in sketch (d), and, by convention, the superscript is omitted from the final family.

The first three of these equations are identical to the third-order Runge-Kutta (Heune's) method if the various terms have the following specific values:

j	r_j	γ_{1j}	γ_{2j}	α_j	β_j
1	1/3	---	---	1	1/3
2	2/3	2/3	---	1	0
3	1	0	3/4	1	1/4

and the first four equations are identical to the standard fourth-order Runge-Kutta method if

j	r_j	γ_{1j}	γ_{2j}	γ_{3j}	α_j	β_j
1	1/2	---	---	---	1	1/2
2	1/2	1/2	---	---	1	0
3	1	0	1	---	1	0
4	1	1/3	1/3	1/6	1	1/6

Introduce the representative equation (14) into equation (22) and we derive the operational matrix relation,

$$\begin{bmatrix} E^{r_1} & 0 & \cdots & -\alpha_1 - \beta_1\sigma h \\ -\sigma h\gamma_{12}E^{r_1} & E^{r_2} & & -\alpha_2 - \beta_2\sigma h \\ -\sigma h\gamma_{13}E^{r_1} & -\sigma h\gamma_{23}E^{r_2} & & -\alpha_3 - \beta_3\sigma h \\ \vdots & & & \vdots \\ -\sigma h\gamma_{1k}E^{r_1} & -\sigma h\gamma_{2k}E^{r_2} & \cdots & E - \alpha_k - \beta_k\sigma h \end{bmatrix} \times \begin{bmatrix} w_n^{(1)} \\ w_n^{(2)} \\ w_n^{(3)} \\ \\ w_n \end{bmatrix} = ahe^{\mu hn} \begin{bmatrix} \beta_1 \\ \beta_2 + \gamma_{12}e^{\mu hr_1} \\ \beta_3 + \gamma_{13}e^{\mu hr_1} + \gamma_{23}e^{\mu hr_2} \\ \vdots \\ \beta_k + \gamma_{1k}e^{\mu hr_1} + \cdots \end{bmatrix} \quad (23)$$

The characteristic equation is found from (23) by setting the determinant of the square matrix equal to zero. Notice that, for fixed j, E^{r_j} is common to all the elements in the jth column. Therefore, the characteristic equation of any method represented by equations (22) always reduces to

$$E^{r_1}E^{r_2} \cdots E^{r_{k-1}}[E - \lambda(\sigma h)] = 0 \tag{24}$$

Hence, all of the roots except one are zero - regardless of the values of r_j. Since the stability of a method is completely determined by the roots of its characteristic equation, the stability of equations (22) is independent of the choices of the various r_j.

Autonomous.- Consider next an autonomous set of differential equations. Such cases are represented in equation (23) when μ is set equal to zero. Under such a condition one can solve for w_n and show

$$w_n = c[\lambda(\sigma h)]^n - \frac{a}{\sigma}\frac{D_1}{D_2} \tag{25}$$

where

$$D_1 = \det \begin{pmatrix} 1 & 0 & \cdots & -\alpha_1 - \beta_1\sigma h \\ -\sigma h\gamma_{12} & 1 & & -\alpha_2 - \beta_2\sigma h \\ -\sigma h\gamma_{13} & -\sigma h\gamma_{23} & & -\alpha_3 - \beta_3\sigma h \\ \vdots & & & \vdots \\ -\sigma h\gamma_{1k} & -\sigma h\gamma_{2k} & \cdots & 1 - \alpha_k - \beta_k\sigma h \end{pmatrix}$$

and

$$D_2 = \det \begin{pmatrix} 1 & 0 & \cdots & -\sigma h\beta_1 \\ -\sigma h\gamma_{12} & 1 & & -\sigma h(\beta_2 + \gamma_{12}) \\ -\sigma h\gamma_{13} & -\sigma h\gamma_{23} & & -\sigma h(\beta_3 + \gamma_{13} + \gamma_{23}) \\ \vdots & & & \vdots \\ -\sigma h\gamma_{1k} & -\sigma h\gamma_{2k} & \cdots & -\sigma h(\beta_k + \gamma_{1k} + \gamma_{2k} + \cdots) \end{pmatrix}$$

Subtracting each of the columns (except the last) in D_2 from the last column in D_2 does not alter the value of the determinant and results in the form

$$D_2 = \det \begin{pmatrix} 1 & 0 & \cdots & -1 - \beta_1\sigma h \\ -\sigma h\gamma_{12} & 1 & & -1 - \beta_2\sigma h \\ -\sigma h\gamma_{13} & -\sigma h\gamma_{23} & & -1 - \beta_3\sigma h \\ \vdots & & & \vdots \\ -\sigma h\gamma_{1k} & -\sigma h\gamma_{2k} & \cdots & -\beta_k\sigma h \end{pmatrix}$$

Clearly $D_1 = D_2$ if $\alpha_j = 1$; $j = 1,2, \ldots, k$.

Now the exact solution of the representative differential equation (14) is

$$w_n = c(e^{\sigma h})^n + \frac{a}{\mu - \sigma} e^{\mu h n}$$

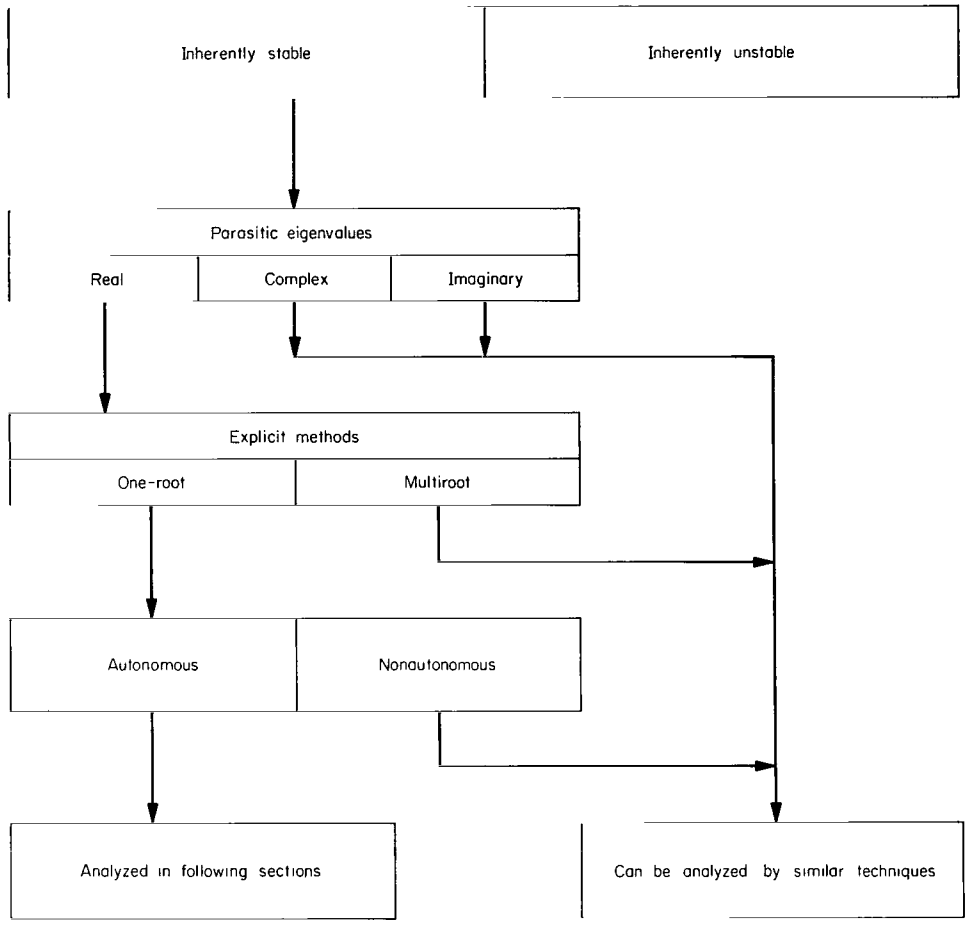
where t has been set equal to nh. If the equation is to be autonomous $\mu = 0$, and the exact solution is

$$w_n = c(e^{\sigma h})^n - \frac{a}{\sigma} \tag{26}$$

Comparing equations (25) and (26), and using the result just derived for D_1 and D_2, we see that if equations (22) are applied to an autonomous set of linear differential equations, the particular solution is calculated without any error whatsoever if all the α_j are set equal to 1.

Finally, notice that if the differential equations are autonomous, differencing them by means of equations (22) results in a method for which both the stability and accuracy are independent of r_j.

Discussion.- In the next sections two numerical methods are derived, both of which are suitable for use on differential equations with parasitic eigenvalues. The various specializations assumed are summarized in the following chart.



Special One-Root Methods

Case 1.- It was just shown that the predictor-corrector sequence given by equations (22) calculates, when applied to linear autonomous differential equations, the particular solution exactly if $\alpha_j = 1$; $j = 1,2, \ldots, k$. It also was shown that under the same conditions the sequence generates only one nonzero root and the value of this root is independent of the various r_j. Making use of these facts, one can construct certain forms of equations (22) in which the terms γ_{ij} and β_j are related to the coefficients in the characteristic root by quite simple formulas. For example, consider the form of equations (22) given by

$$
\left.
\begin{aligned}
w_{n+1}^{(1)} &= w_n + hw_n' \\
w_{n+1}^{(2)} &= w_n + hw_{n+1}^{(1)\prime} \\
&\vdots \\
w_{n+1}^{(k-1)} &= w_n + hw_{n+1}^{(k-2)\prime} \\
w_{n+1} &= w_n + h(\delta_1 w_n' + \delta_2 w_{n+1}^{(1)\prime} + \cdots + \delta_k w_{n+1}^{(k-1)\prime})
\end{aligned}
\right\} \qquad (27)
$$

where, for convenience in the subsequent expressions, we have set $\delta_1 = \beta_k$, $\delta_2 = \gamma_{1k}$, $\delta_3 = \gamma_{2k}$, etc. Applied to the representative equation (14), one finds the characteristic equation in determinant form

$$
\det \begin{pmatrix}
1 & 0 & \cdots & 0 & 0 & -(1+\sigma h) \\
-\sigma h & 1 & \cdots & 0 & 0 & -1 \\
0 & -\sigma h & \cdots & 0 & 0 & -1 \\
\vdots & \vdots & & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & -\sigma h & 1 & -1 \\
-\delta_2 \sigma h & -\delta_3 \sigma h & \cdots & -\delta_{k-1} \sigma h & -\delta_k \sigma h & E - (1 + \delta_1 \sigma h)
\end{pmatrix} = 0
$$

which expands to

$$
E - 1 - \sigma h \sum_{1}^{k} \delta_j - (\sigma h)^2 \sum_{2}^{k} \delta_j - \cdots - (\sigma h)^k \delta_k = 0
$$

having the single root

$$
\lambda = 1 + a_1 \sigma h + a_2 (\sigma h)^2 + \cdots + a_k (\sigma h)^k
$$

where

$$
a_j = \sum_{j}^{k} \delta_i , \qquad j = 1, \ldots, k \qquad (28a)
$$

Now recall that the application of a one-root, predictor-corrector sequence to the autonomous form of the representative differential equation

results in the numerical solution

$$w_n = c(1 + a_1\sigma h + a_2(\sigma h)^2 + \cdots + a_k(\sigma h)^k)^n - \frac{a}{\sigma}$$

whereas the exact solution (in expanded form) is

$$w_n = c\left(1 + \sigma h + \frac{1}{2}(\sigma h)^2 + \cdots + \frac{1}{k!}(\sigma h)^k + \cdots\right)^n - \frac{a}{\sigma}$$

Equation (28a) displays, therefore, the connection between the terms in the predictor-corrector formulas (27), and the numerical approximation to the exact solution of the differential equation resulting from their use.

Notice that equations (28a) can be inverted to form the simple recursion relations

$$\left.\begin{aligned} \delta_k &= a_k \\ \delta_j &= a_j - a_{j+1}, \qquad j = k-1, \ldots, 1 \end{aligned}\right\} \tag{28b}$$

If we seek very accurate methods, it is clear from a glance at the two expressions below equation (28a) that we should set $a_j = 1/j!$ and find the terms in the final corrector in the sequence (27) by means of equations (28b). However, if we seek very stable methods, the coefficients of the higher order terms are modified accordingly. Just what their values should be in such cases is discussed in the section entitled Stability Polynomials.

Case 2.- An alternative to the predictor-corrector sequence studied under Case 1 is the following scheme

$$\left.\begin{aligned} w_{n+1}^{(1)} &= w_n + \beta_1 h w_n' \\ w_{n+1}^{(2)} &= w_n + \beta_2 h w_{n+1}^{(1)\prime} \\ &\vdots \\ w_{n+1} &= w_n + \beta_k h w_{n+1}^{(k-1)\prime} \end{aligned}\right\} \tag{29}$$

where, for convenience in the subsequent development, γ_{12} in equations (22) has been replaced by β_2, γ_{23} by β_3, etc. The characteristic equation in determinant form is now

$$\det \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & -(1+\beta_1\sigma h) \\ -\beta_2\sigma h & 1 & 0 & \cdots & 0 & -1 \\ 0 & -\beta_3\sigma h & 1 & \cdots & 0 & -1 \\ 0 & 0 & -\beta_4\sigma h & \cdots & 0 & -1 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\beta_k\sigma h & E-1 \end{pmatrix} = 0$$

which expands to

$$E - 1 - \sigma h\beta_k - (\sigma h)^2 \prod_{k-1}^{k} \beta_j - \cdots - (\sigma h)^k \prod_{1}^{k} \beta_j$$

with the root

$$\lambda = 1 + a_1\sigma h + \cdots + a_k(\sigma h)^k$$

where

$$a_j = \prod_{k+1-j}^{k} \beta_i \tag{30a}$$

and, conversely,

$$\left.\begin{aligned} \beta_k &= a_1 \\ \beta_{k-1} &= a_2/a_1 \\ \beta_{k-2} &= a_3/a_2 \\ \beta_{k-j} &= a_{j+1}/a_j \end{aligned}\right\} \tag{30b}$$

The appropriate choice of a_j is discussed in the next section.

A discussion of the merits and deficiencies of this method, and that presented as case 1, is given in the section beginning on page 27.

STABILITY POLYNOMIALS

General Discussion

Two ways have been presented for constructing simple predictor-corrector sequences that produce characteristic equations having any given values for the coefficients a_j in the single root

$$\lambda = 1 + a_1(\sigma h) + \ldots + a_k(\sigma h)^k \tag{32}$$

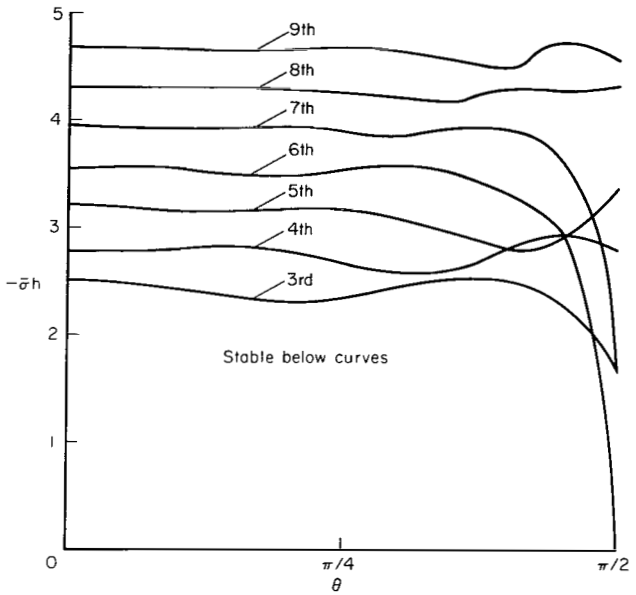
If all the values of a_j are set equal to the corresponding coefficient of $(\sigma h)^j$ in the truncated expression of $e^{\sigma h}$,

$$a_j = \frac{1}{j!}, \qquad j = 1,2, \ldots, k \tag{33}$$

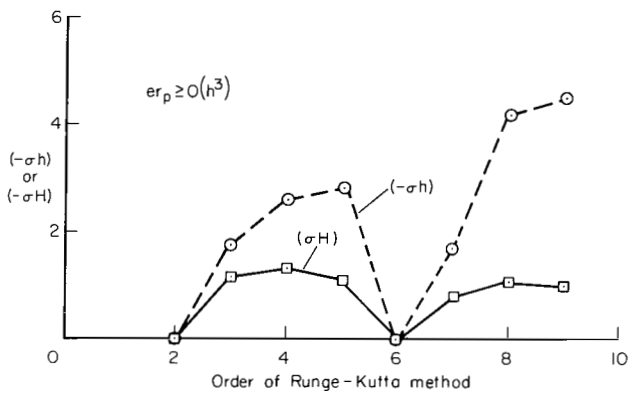
and the error in the local embedded polynomial is led by the term $(\sigma h)^{k+1}/(k+1)!$. The development of the inequality (13b), however, indicates that, from the point of view of accuracy, there may be no advantage in making $a_3 = 1/6$ (so the $er_p = O(h^4)$) because of the presence of the term $h^3 er_t$. Of course, the use of equation (33) for $j > 3$ also would be pointless if accuracy is the only factor involved and inequality (13b) is pertinent.

Let us next investigate the possibility that the choice of a_j given by equation (33) may still be the best from the point of view of stability, irrespective of the accuracy. This possibility can occur when the parasitic eigenvalues in the associated matrix may have any complex value during the numerical integration; that is, when there is no a priori knowledge of their behavior. When the a_j are given by equation (33), the root given by equation (32) is identical to that generated by the Runge-Kutta methods. The stability boundary for the fourth-order Runge-Kutta method has already been shown in sketch (c). Similar results for the third- through ninth-order methods (corresponding to using eq. (33) for $k = 3$ through 9) are shown in sketch (e). The fourth-, eighth-, and ninth-order methods appear to be about optimum in that the stability boundary is about the same for $0 \le \theta \le \pi/2$. The third-, fifth-, and seventh-order methods could be improved because of their behavior when θ is near $\pi/2$, and the sixth-order method is actually unstable for $\theta = \pi/2$.

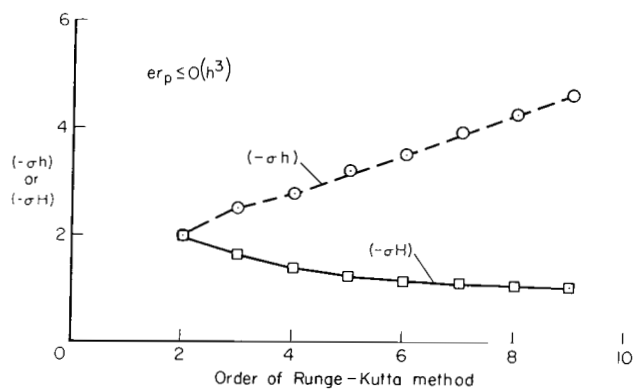
In many problems, requiring the numerical integration of differential equations, the evaluation of the derivative (i.e., the right side of eq. (1)) is, by far, the most time consuming of the various numerical processes involved. In order to compare various methods as they apply to such cases it is necessary to reference both the accuracy and stability to H, the representative step size (see definition of symbols), rather than to h, the step size actually used in the calculations. Comparisons of the general induced stability boundaries (determined by the smallest value of $(-\sigma h)$ on the curves in sketch (e)) for the two reference step sizes are shown in sketch (f) for all



Sketch (e).- Induced stability boundaries of third through ninth order Runge-Kutta methods or one-root methods with coefficients given by equation (33).



Sketch (f).- General induced stability boundary.



Sketch (g).- Real induced stability boundary.

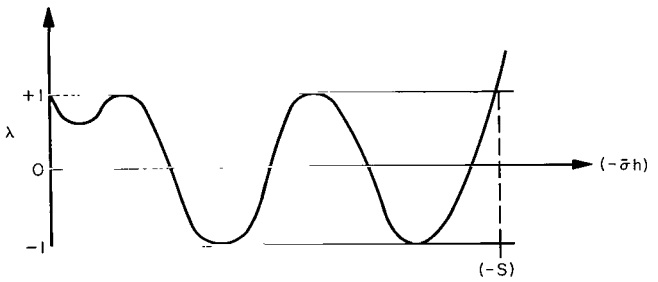
Runge-Kutta methods for which er_p is less than or equal to $O(h^3)$ up through the ninth order. When based on H, the fourth-order method has the highest stability boundary of all the Runge-Kutta methods shown (or any other one-root method with coefficients given by eq. (33)).

Let us next consider the case when all of the parasitic eigenvalues are known to be real. The one-root methods just described have the real induced stability boundaries shown in sketch (g). When based on the representative step size, H, the second-order method is now the best (from the viewpoint of stability) of all the Runge-Kutta methods with accuracy $O(h^3)$ or higher. However, for these cases much better choices of the a_j in equation (32) exist. Setting $er_p = O(h^3)$, to be consistent with inequality (13b), and insisting on real eigenvalues, we let

$$\lambda = 1 + (\bar{\sigma}h) + \frac{1}{2}(\bar{\sigma}h)^2 + a_3(\bar{\sigma}h)^3 + \ldots + a_k(\bar{\sigma}h)^k \qquad (34)$$

and choose the a_j, $j > 2$, so that λ is as close as possible to the optimum real stability polynomial defined as follows:

Definition: Consider the kth order polynomial given by equation (34) in which the a_j, $j = 3,4, \ldots, k$, are any real constants for which $|\lambda| \leq 1$ over the entire range for which $0 \leq (-\bar{\sigma}h) \leq (-S)$. The optimum real stability polynomial of order k is defined to be that one for which S is a maximum.



Sketch (h)

It is hypothesized that the optimum real stability polynomial looks like the one shown in sketch (h). The behavior for small $|\bar{\sigma}h|$ is governed by the term $1 + \bar{\sigma}h + (1/2)(\bar{\sigma}h)^2$, and (except for the first) the maximums and minimums of the curve for large values of $|\bar{\sigma}h|$ lie on the bounding lines $\lambda = \pm 1$, there being k - 1 local extremums for a kth order polynomial. For $k = 3$ this hypothesis is correct. In fact one finds the optimum third-order real stability polynomial by choosing that value of a_3 for which a maximum of λ is +1. This gives

$$\lambda = 1 + (\bar{\sigma}h) + \frac{1}{2}(\bar{\sigma}h)^2 + \frac{1}{16}(\bar{\sigma}h)^3 \tag{35}$$

for which $|\bar{\sigma}h|_c \approx 6.25$, or $|\bar{\sigma}H|_c \approx 4.17$, already a considerable improvement over the values shown in sketch (g).

The equations for higher order optimum stability polynomials are unknown to the author. However, in the next section a method is presented which generates highly stable polynomials, although not the optimum ones.

Stability Polynomials Found by the Method of Least Squares

The following is a simple way to generate a class of polynomials that will have highly stable properties in the sense discussed in the previous section. Consider the definition

$$P_k \equiv 1 - z + \frac{1}{2}z^2 + a_3 z^3 + \ldots + a_k z^k \tag{36}$$

The minus sign is chosen so that, in the subsequent analysis, z is positive; consequently, the signs of all the odd a_j derived in the following must be reversed before being used in equations (28) or (30). We seek that class of polynomials for which

(a) The a_j are real numbers such that

$$\frac{\partial}{\partial a_j}\int_0^r (P_k)^2\, dz = 0\,, \qquad j = 3,4,\ldots,k \tag{37a}$$

for any given r.

(b) r is the maximum positive number for which

$$|P_k| \le 1 \quad \text{for} \quad 0 \le z < r \tag{37b}$$

Conditions (37) immediately lead to the matrix equation

$$[X]\vec{AR} = [Y]\vec{R} \tag{38}$$

where for $k \ge 3$

$$[X] = \begin{bmatrix} \frac{1}{7} & \frac{1}{8} & \cdots & \frac{1}{k+4} \\ \frac{1}{8} & \frac{1}{9} & \cdots & \frac{1}{k+5} \\ \vdots & \vdots & & \vdots \\ \frac{1}{k+4} & \frac{1}{k+5} & \cdots & \frac{1}{k+k+1} \end{bmatrix} \tag{39a}$$

$$[Y] = \begin{bmatrix} -\frac{1}{4} & \frac{1}{5} & -\frac{1}{12} \\ -\frac{1}{5} & \frac{1}{6} & -\frac{1}{14} \\ \vdots & \vdots & \vdots \\ -\frac{1}{k+1} & \frac{1}{k+2} & -\frac{1}{2(k+3)} \end{bmatrix} \tag{39b}$$

$$\vec{R}^T = (1, r, r^2) \tag{39c}$$

and

$$\vec{AR}^T = (a_3 r^3,\ a_4 r^4,\ \ldots,\ a_k r^k) \tag{39d}$$

Solutions to equation (38) were found by numerically calculating the matrix

$$[Z] = [X]^{-1}[Y] \tag{40a}$$

which is independent of r, and then mechanically plotting the polynomial P_n obtained from the solution

$$\vec{AR} = [Z]\vec{R} \tag{40b}$$

for various choices of r. The matrix given by equation (39a) is a subclass of the Hilbert matrix which is very poorly conditioned for numerical inversion, so 16 place, floating-point arithmetic was used for finding $[X]^{-1}$ in all cases. Final results are shown in figure 1 for $k = 3$ through 10. The corresponding values of a_j (with the signs of the odd terms reversed so that they may be inserted directly into eqs. (28), (30), and (34)) are given in table I.

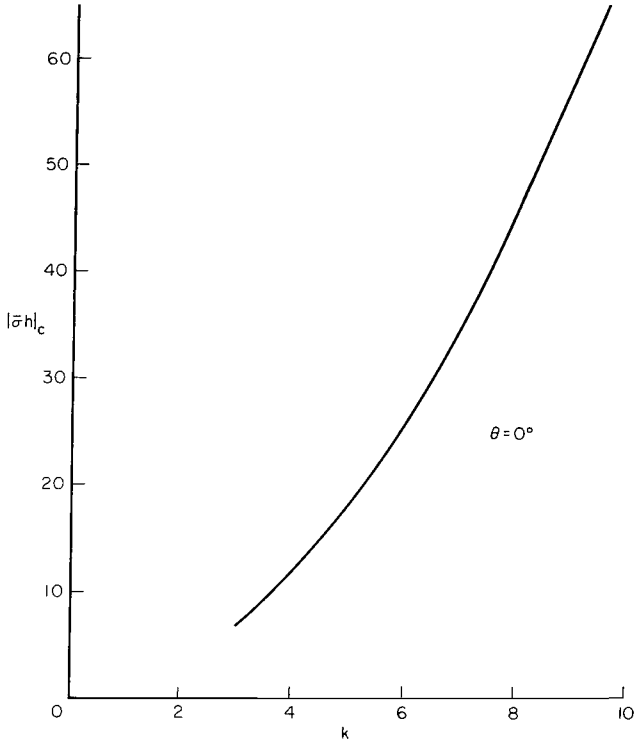
Notice that as k increases, the absolute values of the local extremums fall considerably below unity, except for the second, which is, in every case, the limiting one. (Care was taken so that in every case this first maximum was less than one; hence, stability for real eigenvalues is assured over the entire range indicated.) Obviously more sophisticated methods with restraints other than the simple least squares technique can be constructed to generate stability polynomials that are closer to the optimum.

The real induced stability boundaries for the class of polynomials given in figure 1 and table I are illustrated in sketches (i) and (j). The advantages of increasing the number of iterations (i.e., using more correctors) at a given step are still increasing even after nine iterations. This is shown in sketch (j). The improvements over the second- and fourth-order Runge-Kutta methods are also indicated.

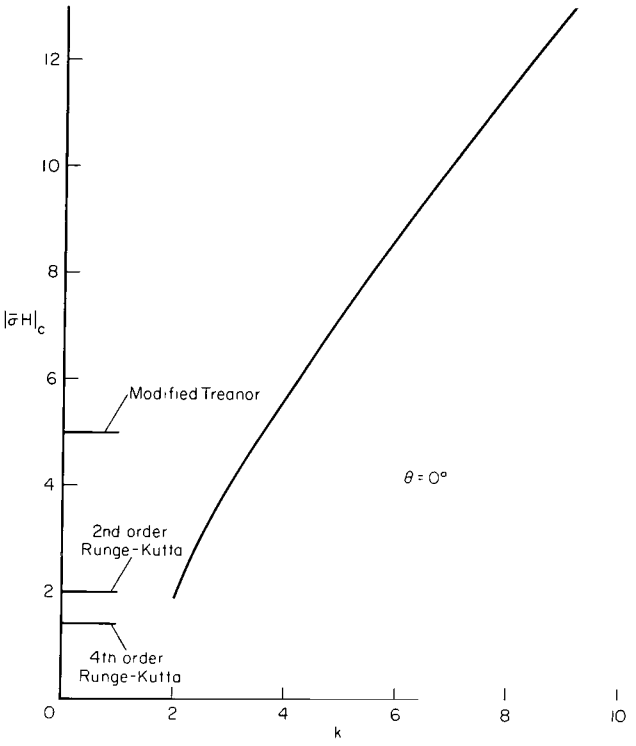
In reference 7 Treanor proposed a method to be used for integrating equations with parasitic real eigenvalues. This method requires four evaluations of the derivative at each step and, in effect, replaces the coefficients a_2, a_3, and a_4 in equation (32) by formulas with exponential terms containing data computed in the integration process. A thorough analysis of this method is given in reference 1. The real stability boundary of a modified form[1] of Treanor's method (the second modification given on p. 36 in ref. 1) is shown in sketch (j).

The "off-design" stability properties of the methods represented in table I are shown in sketch (k) over the complex range from $0 \le \theta \le \pi/2$. Although the very high stability characteristics found for real eigenvalues are drastically reduced if any imaginary component is present, the methods are still more stable than the fourth-order Runge-Kutta method for $0 \le \theta \lesssim 0.7 \times \pi/2$.

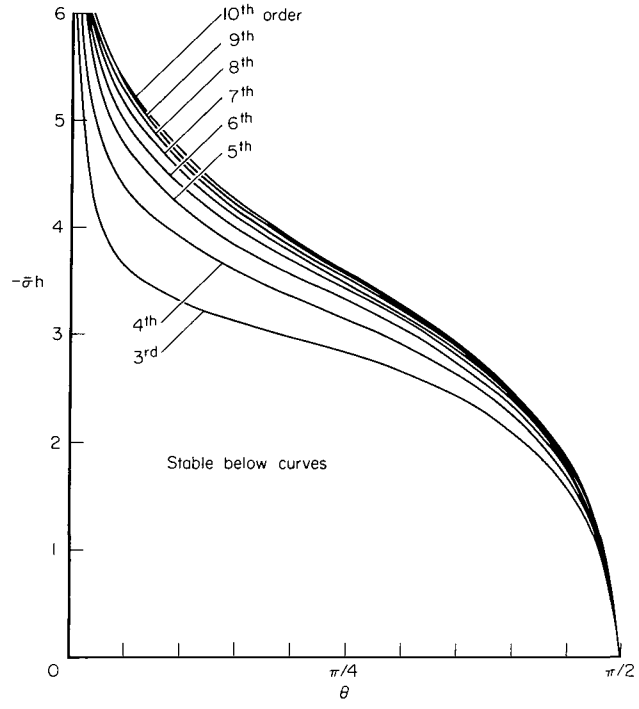
[1]The modified form is used in order that a rigorous comparison can be made.



Sketch (i).- Real induced stability boundaries for stability polynomials given in table I, based on calculation step size.



Sketch (j).- Real induced stability boundaries for stability polynomials given in table I, based on representative step size.



Sketch (k).- General induced stability boundaries for stability polynomials given in table I.

The behavior of the curves at $\theta = \pi/2$ requires some further consideration. Strictly speaking, none of the methods is stable for a pure imaginary eigenvalue. The reason for the sensitivity at this particular point is that the magnitude of the exact solution is $|e^{\pm i\bar{\sigma}nh}|$ which is exactly 1. Since there must be some error in the numerical calculation of the exact solution, the numerical root must fall just inside or just outside the unit circle (see, e.g., ref. 2, fig. 10) for small, but not zero, values of $\bar{\sigma}h$. It so happens that all of the methods are such that the roots start to go outside the circle. The rate at which they proceed outside of it is given in the following table.

$i\bar{\sigma}h$	Order of polynomial							
	3	4	5	6	7	8	9	10
0	1	1	1	1	1	1	1	1
.2i	1.00010	1.00008	1.00007	1.00007	1.00007	1.00006	1.00006	1.00006
.4i	1.00159	1.00128	1.00116	1.00109	1.00106	1.00103	1.00101	1.00100
.6i	1.00810	1.00652	1.00588	1.00555	1.00535	1.00522	1.00512	1.00507
.8i	1.02560	1.02059	1.01857	1.01752	1.01688	1.01647	1.01618	1.01599
1.0i	1.06208	1.05000	1.04508	1.04251	1.04096	1.03995	1.03924	1.03879

Values of $|\lambda|$ produced by the polynomials given by equation (32) and table I when an eigenvalue is imaginary.

The interpretation of these results depends on the nature of the problem. Recall that if $u' = i\bar{\sigma}hu$, $u_{n+k} = (\lambda)^k u_n$, where λ is the value in the table corresponding to the appropriate order and $i\bar{\sigma}h$. Note that if the value of u at $n + k$ is to increase by ϵ over its value at n, then k steps must be taken where

$$k = \frac{\ln(1+\epsilon)}{\ln(\lambda)}$$

Thus if $\bar{\sigma}$ is an important eigenvalue (i.e., contributes significantly to the solution for the various $\vec{w}$ when coupled into the equations) and h is chosen so that $\bar{\sigma}h \leq 0.2$, about $\ln(1.01)/\ln(1.0001) \approx 100$ steps can be taken with an error no greater than 1 percent regardless of the order. This has been built into the methods by ordering the truncation error to $O(h^3)$. On the other hand, if the role of the eigenvalue is not important (i.e., it represents low-amplitude noise), it may be more significant to notice that $i\bar{\sigma}h$ can be 0.8i, and 34 steps can be calculated before the original value is doubled.

AN ANALYSIS OF TWO KINDS OF PREDICTOR-CORRECTOR SEQUENCES THAT ARE STABLE FOR AUTONOMOUS LINEAR COUPLED EQUATIONS WITH PARASITIC REAL EIGENVALUES IN THE ASSOCIATED MATRIX

Type 1 - Weighting the Final Corrector Only

Two kinds of methods (corresponding to the two cases discussed in the section commencing on p. 17) are now analyzed in detail. Consider first the predictor-corrector sequence given by equation (27). If the δ_j are determined from equation (28b), in which the a_j are given in table I for a given k, the results are shown in table II. Suppose, for example, we choose to use the method for which $k = 5$. Then the actual predictor-corrector formulas are

$$
\left.
\begin{aligned}
u_{n+1}^{(1)} &= u_n + hu_n' \\
u_{n+1}^{(2)} &= u_n + hu_{n+1}^{(1)\prime} \\
u_{n+1}^{(3)} &= u_n + hu_{n+1}^{(2)\prime} \\
u_{n+1}^{(4)} &= u_n + hu_{n+1}^{(3)\prime} \\
u_{n+1} &= u_n + h(0.5u_n' + 0.414435674u_{n+1}^{(1)\prime} + 0.0798309969u_{n+1}^{(2)} \\
&\quad + 0.00560204966u_{n+1}^{(3)\prime} + 0.000131279869u_{n+1}^{(4)\prime})
\end{aligned}
\right\} \qquad (41)
$$

We see from table I that $a_3 \approx 0.086$, so, from the expression under equation (28a), the truncation error is led by the term $(1/6 - 0.086)h^3$ or

$$er_p \approx 0.081h^3 \qquad (42)$$

From sketch (i) we find that the method is stable for real eigenvalues such that $-\bar{\sigma}h \lesssim 17.5$. Suppose our problem is one in which the magnitudes of the driving eigenvalues are around one. According to equation (42), we require a step size of about 0.1 or less to calculate the solution with an error bounded by about $er_p \approx 0.000081$. (The term er_t must also be $\lesssim 0.08$ if the equations are nonlinear.) Hence, the method is both stable and accurate in the presence of negative eigenvalues up to about -175, provided all of the eigenvalues fall under the curves in sketch (j) if they are complex. From sketch (j) we see that, under the conditions just mentioned, the method given by equations (41) can (when H is the appropriate reference) be five times faster than the fourth-order, and three and one half times faster than the second-order Runge-Kutta methods.

The methods represented by equations (27) and table II are attractive because of their simplicity and flexibility in application. Notice that all but the last corrector are identical so the iterations can be terminated at any point in a cycle of computation if the appropriate final corrector is then applied. Furthermore, they can be used without any modification to find the approximate magnitude of the largest parasitic eigenvalue in a manner to be described later. Unfortunately, however, when used to integrate nonlinear equations of the form given by equation (8), they have a serious drawback. This is brought out in a later section which discusses nonlinear effects.

Case 2 - Weighting Successive Correctors

The predictor-corrector sequences given by equation (29), in which the β_j are determined from equation (30b) with the a_j in table I, can be immediately written using table III. Thus the actual predictor-corrector method for $k = 8$ is composed of the following eight steps

$$
\left.
\begin{aligned}
u_{n+1}^{(1)} &= u_n + 0.00548929287hu_n' \\
u_{n+1}^{(2)} &= u_n + 0.0135316886hu_{n+1}^{(1)\prime} \\
u_{n+1}^{(3)} &= u_n + 0.0260698673hu_{n+1}^{(2)\prime} \\
u_{n+1}^{(4)} &= u_n + 0.0473850056hu_{n+1}^{(3)\prime} \\
u_{n+1}^{(5)} &= u_n + 0.0885814236hu_{n+1}^{(4)\prime} \\
u_{n+1}^{(6)} &= u_n + 0.186192156hu_{n+1}^{(5)\prime} \\
u_{n+1}^{(7)} &= u_n + 0.5hu_{n+1}^{(6)\prime} \\
u_{n+1} &= u_n + hu_{n+1}^{(7)\prime}
\end{aligned}
\right\} \qquad (43)
$$

In this case $a_3 \approx 0.093$ and the truncation error (again a precaution must be made regarding the term er_t in eq. (33b)) is led by the term

$$er_p \approx 0.074h^3 \qquad (44)$$

Now, however, if a step size of 0.1 is used to resolve the driving eigenvalues (all complex values of which must lie in the stable range of sketch (k)), negative parasitic eigenvalues up to -450 can be coupled into the associated matrix without inducing instability. From sketch (i) we see that the use of equations (43) can be eight times faster than the fourth-order, and five and one half times faster than the second-order Runge-Kutta methods.

Roundoff Effects

As a test of the reliability of the methods just described in the presence of roundoff error, they were used to integrate three sets of three, coupled, linear equations. Each set has the same eigenvalues and, with appropriately chosen initial conditions, identical solutions in uncoupled form. The eigenvalues are -1, -500, and -1000; and the associated matrixes in the equation

$$\vec{w}' = [A]\vec{w} + \vec{f} \qquad (45)$$

are

$$[A_1] = \begin{bmatrix} -1084.5097 & 218.58539 & 1115.6165 \\ -146.40393 & -53.566240 & 634.12204 \\ -42.599544 & 77.516236 & -362.92405 \end{bmatrix} \tag{46a}$$

$$[A_2] = \begin{bmatrix} -549.42632 & 708.77517 & 22.528027 \\ 226.30682 & -387.05493 & -114.59380 \\ 132.34937 & -667.02699 & -564.51876 \end{bmatrix} \tag{46b}$$

$$[A_3] = \begin{bmatrix} -19409.333 & 74819.682 & -17686.144 \\ -4657.5586 & 17717.487 & -4627.3186 \\ 314.68009 & -2187.2134 & 190.84683 \end{bmatrix} \tag{46c}$$

The $\vec{f}$ vectors are

$$\vec{f}_1 = \begin{bmatrix} 0.66073210 \\ 1.7543727 \\ 0.29797822 \end{bmatrix} \quad \vec{f}_2 = \begin{bmatrix} 0.69796554 \\ 0.55576110 \\ -0.49391808 \end{bmatrix} \quad \vec{f}_3 = \begin{bmatrix} -1.8943767 \\ -0.14342850 \\ 1.4720800 \end{bmatrix} \tag{46d}$$

The uncoupled equations and initial conditions are

$$\left.\begin{array}{ll} u_1' = -u + 1 & u_1(0) = 0 \\ u_2' = -500u & u_2(0) = 0.002 \\ u_3' = -1000u & u_3(0) = 0.001 \end{array}\right\} \tag{47}$$

The elements along the diagonal of [A] in equation (46a) are (except for the one smallest in magnitude) about the same as the eigenvalues themselves; set (46b) has nearly equal elements along the diagonal; and set (46c) has widely varying elements along the diagonal, none of which are close to the eigenvalues. Theoretically any of the one-root methods that generate exactly the same root will give exactly the same solution when used to integrate equation (45) with any of the combinations in equations (46). This was verified by using 16 place, floating-point arithmetic and integrating the equations by means of the methods given by equations (27) and (29) and tables II and III for, in both cases, k = 8. The results after 100 steps using a step size of 0.045 are given below. (The figures in equations (46) have been truncated to

eight places from those actually used, so the following results are not exactly reproducible; but this has no effect on the discussion.)

Equations (27) applied to	Solution for $\vec{w}$	Uncoupled $\vec{w}$ or $\approx \vec{u}$
Equation (46a)	0.653386982323 . . .	0.988883367973 . . .
	1.73487000003 . . .	0
	0.294665704436 . . .	0.13877787 . . . E - 16
Equation (46b)	0.690206513917 . . .	0.988883367973 . . .
	0.549582907231 . . .	0
	-0.488427370653 . . .	-0.13877787 . . . E - 16
Equation (46c)	-1.87331762109 . . .	0.988883367888 . . .
	-0.141834062548 . . .	-0.58619775 . . . E - 12
	1.45571545005 . . .	0.39435121 . . . E - 12

Equations (29) applied to	Solution for $\vec{w}$	Uncoupled $\vec{w}$ or $\approx \vec{u}$
Equation (46a)	0.653386982365 . . .	0.988883368022 . . .
	1.73487000011 . . .	0.64281913 . . . E - 13
	0.294665704451 . . .	0.24234225 . . . E - 11
Equation (46b)	0.690206513949 . . .	0.988883368022 . . .
	0.549582907259 . . .	0.30814240 . . . E - 12
	-0.488427370676 . . .	0.91734952 . . . E - 12
Equation (46c)	-1.87331764033 . . .	0.988883367941 . . .
	-0.141834068684 . . .	-0.39274539 . . . E - 9
	1.45571544412 . . .	0.15559983 . . . E - 7

The exact solution for the first uncoupled u is 0.9888910 . . . . The difference between this and the corresponding values in the right column is a measure of the truncation error. The difference between corresponding numbers in the central column is a measure of the roundoff error. As would be expected, the poorly conditioned matrix given by equation (46c) leads to the largest roundoff error. From these results it also appears that the methods determined by equations (29) are the most seriously affected by roundoff.

Corresponding values of the $\vec{w}$ found by using eight place, floating-point arithmetic under otherwise identical conditions are given below. Using equations (27) one finds

Equation (46a)	Equation (46b)	Equation (46c)
0.65338612	0.69020604	-1.8717663
1.7348677	0.54958251	-0.14171782
0.29466531	-0.48842703	1.4545049

and using equations (29) one finds

Equation (46a)	Equation (46b)	Equation (46c)
0.65326662	0.69075020	-2.3270456
1.7348450	0.54926022	-0.28486464
0.29466745	-0.48908233	1.3223521

Again equations (29) show a serious effect of roundoff. Under "normal" conditions (the results for matrices (46a) and (46b)) the error appears in the third and fourth significant figure. In the "extreme" case (matrix (46c)) the first significant figure is affected (although the method remains stable).

Nonlinear Effects

It is not possible at this time to anticipate in full generality the result of using the methods just described on nonlinear equations with parasitic eigenvalues. They may be of value in some cases and useless in others. However, they are easily programmed and their worth in individual cases can be ascertained by judicious numerical experiments. An example of this philosophy is given at the end of this section.

One thing is certain: if the equations are locally linearized (transformed from eq. (7) to eq. (8)), and the linear form (eq. (8)) is advanced a single step, there can be no growth of the solution due to the parasitic eigenvalues if one stays within the stability boundary. Therefore, global stability is certainly implied for most cases: however, all of the elements of the associated matrix would have to be calculated and, if such is the case, the unconditionally stable implicit methods may well be preferred.

What we wish to discuss here is the effect of combining the concept of controlling parasitic eigenvalues in the direct integration of equation (7). It is important to recall that when dealing with parasitic eigenvalues we are, by definition, completely unconcerned with the accuracy of their resolution. Consider for a moment the standard, fourth-order, Runge-Kutta method applied with a step size of 0.1 to the equation $u' = -27.5u$. One can easily show that there results the sequence of families

$$u^{(1)}_{n+0.5} = \left(1+\frac{1}{2}\sigma h\right) u_n = -0.375u_n$$

$$u^{(2)}_{n+0.5} = \left(1+\frac{1}{2}\sigma h+\frac{1}{4}\sigma^2h^2\right) u_n = 1.516u_n$$

$$u^{(3)}_{n+1} = \left(1+\sigma h+\frac{1}{2}\sigma^2h^2+\frac{1}{4}\sigma^3h^3\right) u_n = -3.168u_n$$

$$u_{n+1} = \left(1+\sigma h+\frac{1}{2}\sigma^2h^2+\frac{1}{6}\sigma^3h^3+\frac{1}{24}\sigma^4h^4\right) u_n = 0.948u_n$$

The procedure is said to be stable because the value of $|u|$ at $n+1$, the only term in the sequence that is remembered, is less than the previous value of $|u|$ at n. Accuracy is of no concern since the value of u_n must already be so small (compared with others in $\vec{w}_n$) that it is negligible. Notice, however, that the value of $u^{(3)}_{n+1}$ is over three times the value of both u_n and u_{n+1}, and remember that in the nonlinear case, $u' = F(u)$, we would be evaluating u' at $F(-3.168u_n)$. This sampling of F with values of u such that $|u| > \max(|u_n|, |u_{n+1}|)$ may have serious consequences in nonlinear cases.

In the example just cited the situation is not critical for most cases, since, if F is well behaved for u_n (which is negligible), it is probably also well behaved for $-3.168u_n$. However, such is far from being the case with the sequence generated by equations (27). Take, for example, $k = 4$ and use a step size of 0.1 to integrate the equation $u' = -110u$. The sequence now is

$$u^{(1)}_{n+1} = (1+\sigma h)u_n = -10u_n$$

$$u^{(2)}_{n+1} = (1+\sigma h+\sigma^2h^2)u_n = 111u_n$$

$$u^{(3)}_{n+1} = (1+\sigma h+\sigma^2h^2+\sigma^3h^3)u_n = -1220u_n$$

$$u_{n+1} = (1+\sigma h+0.5\sigma^2h^2+0.07870\sigma^3h^3+0.003695\sigma^4h^4)u_n = -0.151$$

Again the method is stable for linear systems because $|u_{n+1}| < |u_n|$, but clearly the intermediate values of $|u^{(j)}_{n+1}|$ are far greater than $\max(|u_n|, |u_{n+1}|)$. For larger k the maximum intermediate value of $|u^{(j)}_{n+1}|$ is

about $|\bar{\sigma}h|_c^{k-1}|u_n|$ where $|\bar{\sigma}h|_c$ is shown in sketch (h). This becomes $(70)^9|u_n| \approx 0.405\times10^{17}|u_n|$ for $k = 10$. Values of $|u_{n+1}^{(j)}|$ with this order of magnitude actually appear in the solution of equations (45) and (46) when integrated by means of equations (27) with $k = 10$; although, in these linear computations, all three cases are stable and accurate if 16 place, floating-place arithmetic is employed. (Equation (46a) are also stable with 8 place arithmetic, but equations (46b) and (46c) are not.) In nonlinear cases, however, one would have to be extremely careful about using a derivative calculated from $F(\pm10^{17}u_n)$.

Now let us examine the sequence of predicted and corrected values generated when equations (29) are used. If $k = 8$ and $h = 0.1$, we have for equation $u' = -450u$

$$u_{n+1}^{(1)} = (1 + 0.005489\sigma h)u_n = 0.7530u_n$$

followed by $0.5415u_n$, $0.3647u_n$, $0.2222u_n$, $0.1142u_n$, $0.0434u_n$, and $0.0243u_n$ for $u_{n+1}^{(j)}$, $j = 2,3, \ldots, 7$; and finally $u_{n+1} = -0.0942u_n$. Again the process is stable since $|u_{n+1}| < |u_n|$, but now the method is more likely to be valid for nonlinear equations since $|u_{n+1}^{(j)}| < \max(|u_n|, |u_{n+1}|)$. Unfortunately, as was shown in the previous section, the method is more sensitive to roundoff error.

In order to test the methods represented by equations (29) on practical nonlinear formulas with parasitic eigenvalues, the method for which $k = 8$ was used to integrate the nonequilibrium flow equations discussed in reference 1. The results shown in figure 2 of that reference were duplicated exactly through three significant figures using 8 place, floating-point arithmetric throughout. The solution was carried to $x \approx 0.122$ in 121 steps, the last 39 of which were computed with $|\sigma h| \approx 45$, where $|\sigma|$ was the largest eigenvalue in the local associated matrix. The computing time was half of that required by Treanor's method (which coincides with the results shown in sketch (j)) but was about three times longer than that required by the implicit method described in reference 1.

THE LARGEST PARASITIC EIGENVALUE

It was shown in reference 1 (appendix B) that the intermediate calculations in the fourth-order Runge-Kutta process can be useful in estimating parasitic eigenvalues. The sequence presented in equations (27) and (29) can be used in the same way.

Consider the result of applying equations (27) to equation (8). There follows

$$\vec{w}_{n+1}^{(1)} = [[I] + h[A_n]]\vec{w}_n + hf_n$$

$$\vec{w}_{n+1}^{(1)'} = [[A_n] + h[A_n]^2]\vec{w}_n + h[A_n]f_n$$

or, after j iterations,

$$\left.\begin{aligned} \vec{w}_{n+1}^{(j)} &= [[I] + h[A_n] + \ldots + h^k[A_n]^k]\vec{w}_n + [h[I] + h^2[A_n] + \ldots + h^k[A_n]^{k-1}]\vec{f}_n \\ \vec{w}_{n+1}^{(j)'} &= [[A_n] + h[A_n]^2 + \ldots + h^k[A_n]^{k+1}]\vec{w}_n + [h[A_n] + h^2[A_n]^2 + \ldots + h^k[A_n]^k]\vec{f}_n \end{aligned}\right\} \quad (48)$$

From these one can form the ratio

$$\vec{\Lambda}^{(j)} = \frac{\vec{w}_{n+1}^{(j)'} - \vec{w}_{n+1}^{(j-1)'}}{\vec{w}_{n+1}^{(j)} - \vec{w}_{n+1}^{(j-1)}} = \frac{[A_n]^{j+1}\vec{w}_n + [A_n]^j\vec{f}_n}{[A_n]^j\vec{w}_n + [A_n]^{j-1}\vec{f}_n} \quad (49)$$

where division is defined to mean that an element in the vector in the numerator is divided by the corresponding element in the vector in the denominator.

Now let $\vec{g}_n$ be any linear combination of the eigenvectors of $[A_n]$. Define $\vec{T}^{(j)}$ by

$$\vec{T}^{(j)} = \frac{[A_n]^j\vec{g}_n}{[A_n]^{j-1}\vec{g}_n} \quad (50)$$

where division is defined as above. Since we are interested here in those cases for which $[A_n]$ has large negative eigenvalues, we can make use of the result (see ref. 12, pp. 205 and 206) that, when j is increased, all elements in $\vec{T}^{(j)}$ approach the largest parasitic eigenvalue whose vector has a nonzero weight in $\vec{g}_n$.

The underlined words are important in our application. It was demonstrated (by numerical experiment) in reference 1 that a method that is stable for a given parasitic eigenvalue loses information (delegates it to higher and higher order significant figures) about this eigenvalue as the integration proceeds. In other words the vector $\vec{w}_n$ in equation (49) can eventually be constructed from a set of eigenvectors in which those connected with the parasitic eigenvalues have very little weight.

Typical of what can happen in employing equation (49) to estimate $|\sigma|_{max}$ is shown below. Equations (27) were used with $k = 8$ to integrate equation (45) in which [A] had the form indicated with each set of numbers. The value of $|\sigma h|_{max}$ was 45 in each case and the largest $|\Lambda^{(j)}|$ is recorded.

j / No. steps	2	3	4	5	6	7	Error in uncoupled	
							e^{-500x}	e^{-1000x}
1	-5156	-1402	-1143	-1063	-1030	-1014	0.8×10^{-3}	0.9×10^{-4}
10	98	528	500	500	500	500	0.2×10^{-6}	0.5×10^{-13}
20	-1	-10	-1870	-501	-500	-500	0.2×10^{-10}	0.1×10^{-16}
40	-1	-1	-2	-526	-1030	1017	0.1×10^{-15}	0.5×10^{-16}

Value of $|\Lambda^{(j)}|_{max}$ with matrix given by equation (46a)

j / No. steps	2	3	4	5	6	7	Error in uncoupled	
							e^{-500x}	e^{-1000x}
1	-875	-928	-961	-980	-990	-995	0.8×10^{-3}	0.9×10^{-4}
10	752	-512	-500	-500	-500	-500	0.2×10^{-6}	0.2×10^{-12}
20	-1	22	-768	-509	-517	-534	0.2×10^{-10}	0.2×10^{-12}
40	-1	-1	-250	-986	-994	-997	0.4×10^{-12}	0.2×10^{-12}

Value of $|\Lambda^{(j)}|$ with matrix given by equation (46c)

The two right-hand columns give the errors in the terms representing the eigenvalues -500 and -1000 when the solutions were uncoupled. The terms corresponding to the column for j = 2 is the same as that which would be generated by ratioing the intermediate calculations in the fourth-order Runge-Kutta method. After 20 steps it "sees" only the lowest eigenvalue. Notice that in the examples shown the final ratio at the end of a given step correlates with the eigenvalue with maximum error in uncoupled form, whether or not it is the largest one. When the errors are small and nearly equal, the largest eigenvalue appears.

The above results were for calculations made with 16 place, floating-point arithmetic. The same calculations for 8 place numbers gave the following results:

j / No. steps	2	3	4	5	6	7
1	-5156	-1402	-1143	-1063	-1030	-1014
10	-97	-539	-521	-540	2516	-1699
20	-4	-631	-10613	-1464	-1158	-1068
40	-110	-1557	-1228	-1093	-1042	-1020

Value of $|\Lambda^{(j)}|$ with matrix given by equation (46a)

No. steps \ j	2	3	4	5	6	7
1	-874	-928	-961	-980	-990	-995
10	-892	-1193	-1078	-1036	-1007	-1008
20	-2365	-8099	-1437	-1152	-1066	-1031
40	3843	-1599	-1863	-1232	-1094	-1043

Values of $\Lambda^{(j)}$ with matrix given by equation (46c)

The principal effect of roundoff error, as far as these particular results are concerned, was to bring out the presence of the largest negative eigenvalue at the end of the iterations at each step.

If the method given by equations (29) is used, some additional calculations must be carried out to produce the vector $\vec{\Lambda}^{(j)}$. However, this can be accomplished in the following way. After each iteration in a given step, calculate a $\vec{\Delta}^{(j)}$ such that

$$
\left.
\begin{aligned}
\vec{\Delta}_n^{(1)} &= \vec{w}_n^{(1)} - \vec{w}_n \\
\vec{\Delta}_n^{(2)} &= \vec{w}_n^{(2)} - \vec{w}_n - \frac{\beta_2}{\beta_1}\vec{\Delta}_n^{(1)} \\
\vec{\Delta}_n^{(3)} &= \vec{w}_n^{(3)} - \vec{w}_n - \frac{\beta_3}{\beta_1}(\vec{\Delta}_n^{(2)} + \vec{\Delta}_n^{(1)}) \\
\vec{\Delta}_n^{(4)} &= \vec{w}_n^{(4)} - \vec{w}_n - \frac{\beta_4}{\beta_1}\left(\vec{\Delta}_n^{(3)} + \frac{\beta_3}{\beta_2}\vec{\Delta}_n^{(2)} + \vec{\Delta}_n^{(1)}\right) \\
\vec{\Delta}_n^{(5)} &= \vec{w}_n^{(5)} - \vec{w}_n - \frac{\beta_5}{\beta_1}\left[\vec{\Delta}_n^{(4)} + \frac{\beta_4}{\beta_2}(\vec{\Delta}_n^{(3)} + \vec{\Delta}_n^{(2)}) + \vec{\Delta}_n^{(1)}\right] \\
\vec{\Delta}_n^{(6)} &= \vec{w}_n^{(6)} - \vec{w}_n - \frac{\beta_6}{\beta_1}\left[\vec{\Delta}_n^{(5)} + \frac{\beta_5}{\beta_2}\left(\vec{\Delta}_n^{(4)} + \frac{\beta_4}{\beta_3}\vec{\Delta}_n^{(3)} + \vec{\Delta}_n^{(2)}\right) + \vec{\Delta}_n^{(1)}\right]
\end{aligned}
\right\} \qquad (51)
$$

One can show

$$
\vec{\Delta}_n^{(j)} = ([A_n]^j\vec{w}_n + [A_n]^{j-1}\vec{f}_n)h^j \prod_{i=1}^{j}\beta_i
$$

The same construction for the primed quantities (i.e., $\vec{\Delta}_n^{(1)'} = \vec{w}_n^{(1)'} - \vec{w}_n'$, etc.) gives

$$\vec{\Delta}_n^{(j)'} = ([A_n]^{j+1}\vec{w}_n + [A_n]^j\vec{f}_n)h^j \prod_{i=1}^{j} \beta_i$$

Hence,

$$\vec{\Lambda}^{(j)} = \frac{\vec{\Delta}_n^{(i)'}}{\vec{\Delta}_n^{(j)}} = \frac{[A_n]^{j+1}\vec{w}_n + [A_n]^j\vec{f}_n}{[A_n]^j\vec{w}_n + [A_n]^{j-1}\vec{f}_n}$$

and the right hand side is identical to that in equation (49). This means that the elements of $\vec{\Lambda}^{(j)}$ developed by using equation (49) with the method described by equations (27) are identical with those found from equation (51) using the method described by equations (29) except for the effect of roundoff.

The values of $\vec{\Lambda}^{(j)}$ determined by using equation (51) and the method described by equations (29) to integrate equations (45) and (46) are shown below. The differential equations and initial conditions are identical to those used in constructing the previous tables. The results are

No. steps \ j	2	3	4	5	6	7	Error in uncoupled e^{-500x}	Error in uncoupled e^{-1000x}
1	-5156	-1402	-1143	-1063	-1030	-1014	0.8×10^{-3}	0.9×10^{-4}
10	-98	-528	-500	-500	-500	-500	0.2×10^{-6}	0.7×10^{-12}
20	-1	-18	887	1331	-1820	-1197	0.2×10^{-10}	0.1×10^{-11}
40	-1	-145	-995	-1002	-1002	-1002	0.2×10^{-12}	0.2×10^{-12}

Value of $|\Lambda^{(j)}|_{max}$ with matrix given by equation (26a).

No. steps \ j	2	3	4	5	6	7	Error in uncoupled e^{-500x}	Error in uncoupled e^{-1000x}
1	-875	-928	-961	-980	-990	-995	0.8×10^{-3}	0.9×10^{-4}
10	686	-481	-476	-450	1023	2280	0.2×10^{-6}	0.3×10^{-8}
20	-7	1778	-999	-998	-998	-996	0.5×10^{-9}	0.7×10^{-8}
40	-75	-1123	-998	-998	-997	-997	0.5×10^{-9}	0.1×10^{-7}

Value of $|\Lambda^{(j)}|_{max}$ with matrix given by equation (26c).

The difference between these numbers and those presented for the same matrices using equations (27) is entirely due to roundoff. Since 16 place, floating-point arithmetic was used in both cases, the roundoff effect is quite pronounced. However, to call this a roundoff "error" gives the wrong

impression. The errors due to roundoff for the actual integration of $\vec{w}_n$ have already been presented and discussed on page 29. The differences brought about by roundoff that are shown in the above tables are quite a different matter, and we discuss this next.

The reason for the loss, in some cases, of all 16 significant figures in the evaluation of $\Lambda^{(j)}$ can be explained by referring to equation (50). The values of $\Lambda^{(j)}$ in the preceding tables are, in effect, the result of raising a matrix with a large (-10^3) eigenvalue to the eighth power, multiplying it by some vector, g_n say, repeating the process for the seventh power, and finding the ratio of corresponding elements. The usual statement of the rule is that all of the resulting ratios will be (nearly) the same regardless of the choice of $\vec{g}_n$ and all will be (nearly) equal to the largest eigenvalue. The qualifications that we have already discussed under equation (50) are then made. What actually happens in our application is, effectively, this: the $[A_n]^j$ is very large, but (due to the fact that the numerical process to find $\vec{w}_n$ is stable) the values in $\vec{g}_n$ turn out to represent a vector such that $[A_n]^j\vec{g}_n$ is small. This means that differences of numbers that are very close together are occurring in the calculations with the consequent loss of significant figures. However, it is important to realize that this large effect of roundoff on the values of $\Lambda^{(j)}$ is not in the nature of an error. In fact, since we are only seeking to identify the largest negative eigenvalue, it can be beneficial to the extent that it jars $\vec{g}_n$ away from its sensitive position, and one can actually obtain a more accurate approximation for the value of the largest parasitic eigenvalue with roundoff effects than without them.

Finally, we should mention the transition phenomenon ("struggle for dominance," ref. 12, p. 206) shown in sketch (c) and (h) of reference 1. This occurs in the calculations reported above and accounts for the behavior of rows 3 and 2 in the tables just presented. In both cases a study of values in neighboring steps showed that the eigenvalue -500 was in the process of disappearing from the sequence, and a struggle for dominance was indeed occurring, with the resulting large fluctuations in the values of $\Lambda^{(j)}$.

STEP CONTROL

Present methods for controlling the step size are not very satisfactory for integrating differential equations with parasitic eigenvalues using explicit (i.e., conditionally stable) techniques. It appears to be best to test two variations continually. One, the variation of the function (or its derivative) to make sure that the accuracy of the driving eigenvalues is maintained. This is especially true in nonlinear cases to make sure that $h^3 er_p \approx h^3(w_n')^2F''$ is sufficiently small. The other, the variation of the vector $\Lambda_n^{(j)}$, or its equivalent, in order to keep track of the maximum eigenvalue - which in nonlinear cases is varying as the integration proceeds (see, e.g., ref. 1, fig. 3). The control of the latter is made difficult by the fact that the very process of making a method stable tends to remove the information regarding the parasitic eigenvalues from the vector $\vec{w}_n$. Perhaps a

procedure that occasionally permits a few unstable steps (but not inaccurate steps in terms of truncation error) would be satisfactory. This subject is being studied.

Ames Research Center
National Aeronautics and Space Administration
Moffett Field, Calif., 94035, Jan. 18, 1968
129-04-03-02-00-21

REFERENCES

1. Lomax, Harvard; and Bailey, Harry E.: A Critical Analysis of Various Numerical Integration Methods for Computing the Flow of a Gas in Chemical Nonequilibrium. NASA TN D-4109, 1967.

2. Lomax, Harvard: An Operational Unification of Finite Difference Methods for the Numerical Integration of Ordinary Differential Equations. NASA TR R-262, 1967.

3. Henrici, Peter: Discrete Variable Methods in Ordinary Differential Equations. John Wiley and Sons, Inc., New York, 1962.

4. Dahlquist, Germund G.: Convergence and Stability in the Numerical Integration of Ordinary Differential Equations. Math. Scand., vol. 4, 1956, pp. 33-53.

5. Curtiss, C. F.; and Hirschfelder, J. O.: Integration of Stiff Equations. Proc. Natl. Acad. Sci. U. S., vol. 38, 1952, pp. 235-243.

6. Emanuel, George: Problems Underlying the Numerical Integration of the Chemical and Vibrational Rate Equations in a Near Equilibrium Flow. Rep. AEDC-TDR-63-82, Arnold Engineering Development Center, Tullahoma, Tenn., Mar. 1963.

7. Treanor, Charles E.: A Method for the Numerical Integration of Coupled First-Order Differential Equations With Greatly Different Time Constants. Math. Comp., vol. 20, no. 93, Jan. 1966, pp. 39-45.

8. Bray, K. N. C.; and Pratt, N. H.: Conditions for Significant Gasdynamically Induced Vibration-Recombination Coupling. Proc. Eleventh Symposium (International) on Combustion, Berkeley, Calif., Aug. 14-20, 1966.

9. Moretti, Gino: A New Technique for the Numerical Analysis of Nonequilibrium Flows. AIAA J., vol. 3, no. 2, Feb. 1965, pp. 223-229.

10. Tyson, T. J.: An Implicit Integration Method for Chemical Kinetics. Rep. 9840-6002-RU000, TRW Space Technology Lab., Sept. 1964.

11. Flügge-Lotz, I.; and Davis, R. T.: Laminar Compressible Flow Past Axisymmetric Blunt Bodies. Tech. Rep. 143, Div. Eng. Mech., Stanford Univ., Feb. 1964.

12. Faddeeva, V. N. (C. D. Benster, trans.): Computational Methods of Linear Algebra. Dover Pub., Inc., New York, 1959.

TABLE I.- COEFFICIENTS OF LEAST SQUARES STABILITY POLYNOMIALS,

SEE EQUATION (34) AND FIGURE 1

	k = 3	k = 4	k = 5	k = 6
a_3	0.62500000E-01	0.78703703E-01	0.85564326E-01	0.89289876E-01
a_4		.36954365E-02	.57333295E-02	.69424690E-02
a_5			.13127986E-03	.24382590E-03
a_6				.31760020E-05

	k = 7	k = 8	k = 9	k = 10
a_3	0.91576422E-01	0.93096078E-01	0.94164667E-01	0.94857293E-01
a_4	.77180994E-02	.82465831E-02	.86237831E-02	.88835625E-02
a_5	.32819519E-03	.39076438E-03	.43780978E-03	.47219783E-03
a_6	.68601032E-05	.10187175E-04	.12985567E-04	.15214503E-04
a_7	.56070983E-07	.13784969E-06	.22402858E-06	.30309201E-06
a_8		.75669732E-09	.20832725E-08	.36500460E-08
a_9			.80736327E-11	.24357641E-10
a_{10}				.69155050E-13

TABLE II.- COEFFICIENTS OF THE FINAL CORRECTOR IN EQUATIONS (27) WHICH PRODUCE A LEAST SQUARES STABILITY POLYNOMIAL OF kth ORDER

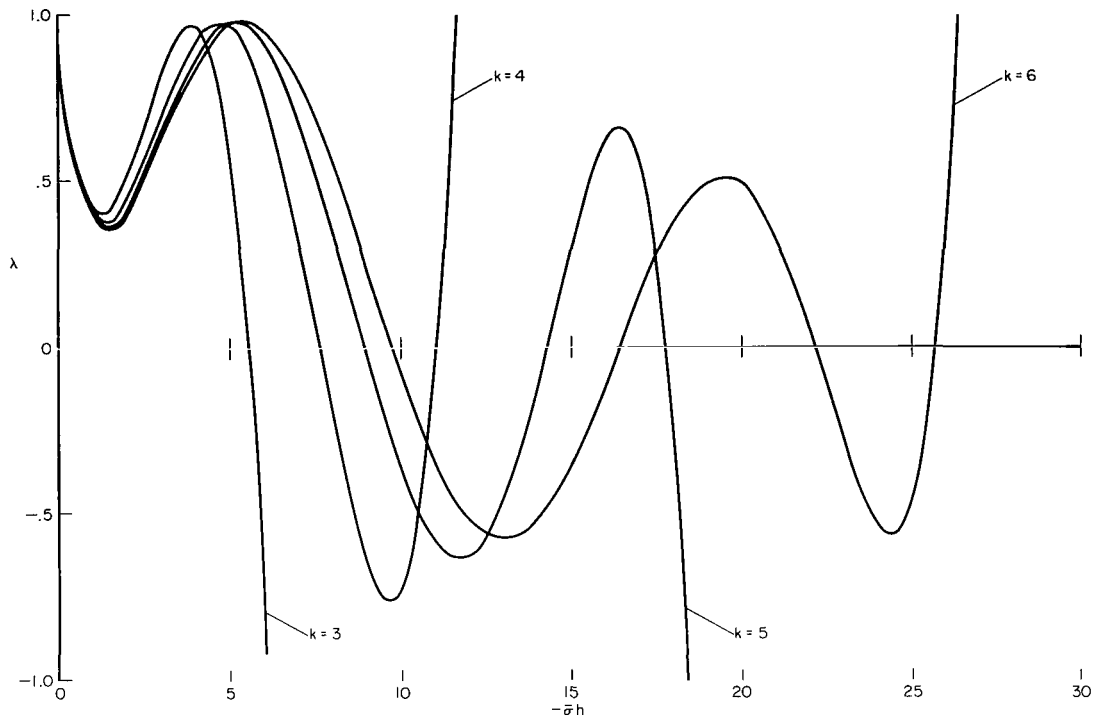
	k = 3	k = 4	k = 5	k = 6
δ_1	0.500000000E+00	0.500000000E+00	0.500000000E+00	0.500000000E+00
δ_2	.437500000E+00	.421296296E+00	.414435674E+00	.410710124E+00
δ_3	.625000000E+00	.750082672E-01	.798309969E-01	.823474072E-01
δ_4		.369543651E-02	.560204966E-02	.669864315E-02
δ_5			.131279869E-03	.240649899E-03
δ_6				.317600208E-05

	k = 7	k = 8	k = 9	k = 10
δ_1	0.500000000E+00	0.500000000E+00	0.500000000E+00	0.500000000E+00
δ_2	.408423577E+00	.406903922E+00	.405835333E+00	.405142706E+00
δ_3	.838583235E-01	.848494950E-01	.855408839E-01	.859737313E-01
δ_4	.738990422E-02	.785581874E-02	.818597340E-02	.841136476E-02
δ_5	.321335096E-03	.380577212E-03	.424824219E-03	.456983328E-03
δ_6	.680403231E-05	.100493261E-04	.127615385E-04	.149114113E-04
δ_7	.560709833E-07	.137092993E-06	.221945317E-06	.299441965E-06
δ_8		.756697322E-09	.207519896E-08	.362568844E-08
δ_9			.807363277E-11	.242884862E-10
δ_{10}				.691550502E-13

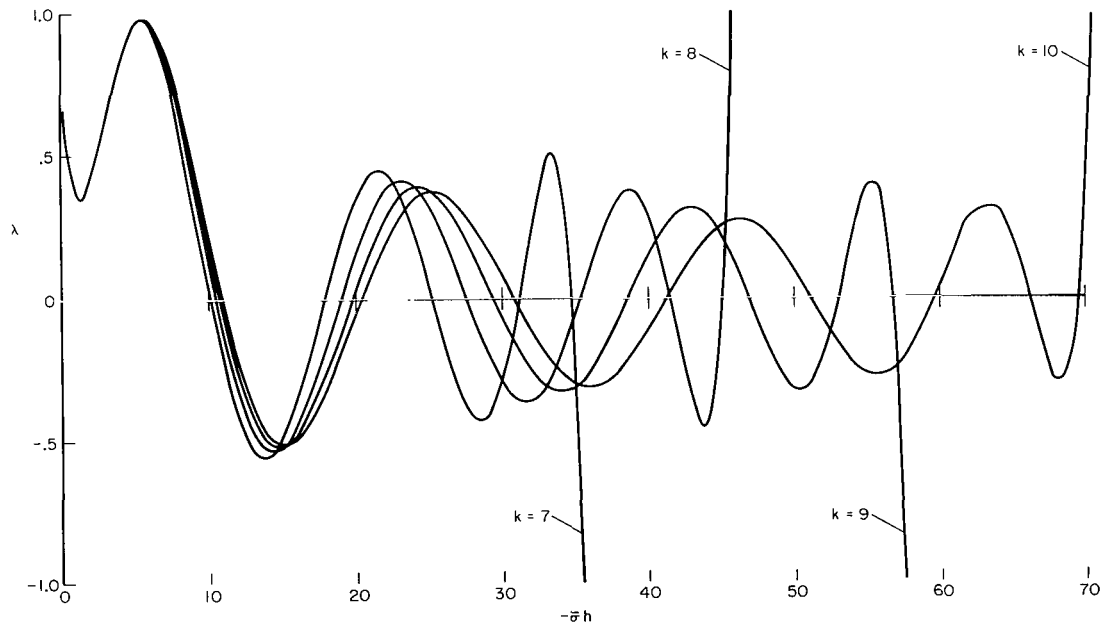
TABLE III.- COEFFICIENTS OF THE SEQUENTIAL CORRECTORS IN EQUATIONS (29) WHICH PRODUCE A LEAST SQUARES POLYNOMIAL OF kth ORDER

	k = 3	k = 4	k = 5	k = 6
β_1	0.125000000E+00	0.469537815E-01	0.228976667E-01	0.130256961E-01
β_2	.500000000E+00	.157407407E+00	.670060733E-01	.351209201E-01
β_3	1.000000000E+00	.500000000E+00	.171128653E+00	.777520290E-01
β_4		1.000000000E+00	.500000000E+00	.178579753E+00
β_5			1.000000000E+00	.500000000E+00
β_6				1.000000000E+00

	k = 7	k = 8	k = 9	k = 10
β_1	0.817348966E-02	0.548929287E-02	0.387545672E-02	0.283915218E-02
β_2	.209025096E-01	.135316886E-01	.929913722E-02	.667324211E-02
β_3	.425228001E-01	.260698673E-01	.172521222E-01	.120426997E-01
β_4	.842804204E-01	.473850056E-01	.296602941E-01	.199212558E-01
β_5	.183152846E+00	.885814236E-01	.507677173E-01	.322206123E-01
β_6	.500000000E+00	.186192156E+00	.915819431E-01	.531541064E-01
β_7	1.000000000E+00	.500000000E+00	.188329334E+00	.936518661E-01
β_8		1.000000000E+00	.500000000E+00	.189714588E+00
β_9			1.000000000E+00	.500000000E+00
β_{10}				1.000000000E+00



(a) Orders 3 through 6.



(b) Orders 7 through 10.

Figure 1.- Least squares stability polynomials determined from equation (34) with table I.

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Notes, and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION DIVISION

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546