UDRI-TM-67-137

AN ALGORITHM FOR APPROXIMATING THE SOLUTION TO A GENERAL FIRST ORDER SYSTEM OF DIFFERENTIAL EQUATIONS

Prepared for
U. S. Naval Weapons Laboratory
Dahlgren, Virginia

By

Herbert Dershem

August 1967

UNIVERSITY OF DAYTON
DAYTON, OHIO

UDRI-TM-67-137

AN ALGORITHM FOR APPROXIMATING THE SOLUTION TO A GENERAL FIRST ORDER SYSTEM OF DIFFERENTIAL EQUATIONS

Prepared for
U. S. Naval Weapons Laboratory
Dahlgren, Virginia

By

Herbert Dershem

August 1967

UNIVERSITY OF DAYTON

Research Institute

Dayton, Ohio

PROBLEM

This report presents an algorithm for approximating the solution to a general first order system of ordinary differential equations of the form

$$(1) \quad \bar{y}' = \bar{f}(t, \bar{y})$$

$$(2) \quad \bar{y}(t_o) = \bar{y}_o$$

where $\bar{y}$ is a vector of dependent variables, t an independent variable and $\bar{f}$ a function vector. The algorithm utilizes a multistep technique of Crane and Klopfenstein [3] , and special procedures for starting the approximation, stability control and interval modification. These procedures are discussed at some length in this report.

Included in the Appendicies are sample applications of some of the discussed techniques.

METHOD OF APPROXIMATING THE SOLUTION

The common methods for approximating the solution of a system of the form (1) - (2) can be divided into four groups:

(i) One-step schemes of the Runge-Kutta type.

(ii) One-step schemes of the Nordsieck type. [5]

(iii) Predictor-corrector schemes of the Adam's type.

(iv) Predictor-corrector schemes of the multistep type (terminology from Gear [4]).

Group (i) schemes can be ruled out immediately in most cases because they usually require four times as many costly evaluations of $\bar{f}$ as methods from any of the other groups and do not provide an economical measure of the truncation error.

Group (ii) has much to recommend it, especially in the case when one is working with a system of higher order equations. Since schemes of this form are one-step, they have the advantage of being self-starting in a qualified manner. They also provide an easily attainable measure of the truncation error and stability and allow the interval size to be changed easily. However, for first order systems, the equivalent group (iv) scheme has greater simplicity and is therefore chosen in this special case.

In reality, group (iii) is a subset of group (iv), but since the use of Adams Type methods are so common, they rate a class of their own. This type of method is generally not preferred, however, over those of group (iv) because a multistep method which uses the same number of previous points has either a higher order truncation error or a larger region of stability, with the only cost being storage, which is not a high price on modern machines.

Hence, it is an algorithm for solving the system (1) - (2) using a method of group (iv) that will be described in the following. The particular member of this group that is recommended is the one suggested by Crane and Klopfenstein [3], especially if one wishes to obtain a large region of stability near the real axis.

GENERAL DESCRIPTION OF THE ALGORITHM

A flow diagram of the algorithm for obtaining an approximation to (1) - (2) by the predictor-corrector multistep method is found in Figure 1. The automatic starting procedure STARTER, the interval modification procedures, HALVE and DOUBLE, and the boolean procedure for detecting instability, STABLE, are described in separate sections and flow diagrams. A list of the input to the algorithm and explanation of the other symbols appearing in the flow diagrams are found in Table 1.

The algorithm begins by calling on procedure STARTER which determines an initial step size which will keep the truncation error within specified bounds (ϵ_{max} and ϵ_{min}) and insure absolute stability. STARTER then integrates forward the required number of steps using a suitable one-step method. One which is highly recommended for this purpose is the Runge-Kutta Technique of Ralston [6]. The number of steps evaluated by STARTER depends upon the number of previous points needed by the chosen multistep method.

After STARTER has completed its task, a predicted value $\bar{p}_{n+1}$ of the dependent variable vector and a corrected value $\bar{y}_{n+1}$ are computed by the multistep scheme. The corrector is reiterated, using the last obtained corrected value to compute the derivative, until the specified number of derivative evaluations, DPS, have been performed. Note that DPS can be, and usually will be, one, in which case the corrector is not reiterated at all.

Once this is done, an estimation of the truncation error is obtained. Letting $e_i = |p_{n_i} - y_{n_i}|$ where p_{n_i} and y_{n_i} are the ith components of $\bar{p}_n$ and $\bar{y}_n$, the error at the nth step is expressed by

$$E_{n_i} = w_i \min \left[e_i , \frac{e_i}{|y_{n_i}|} \right]$$

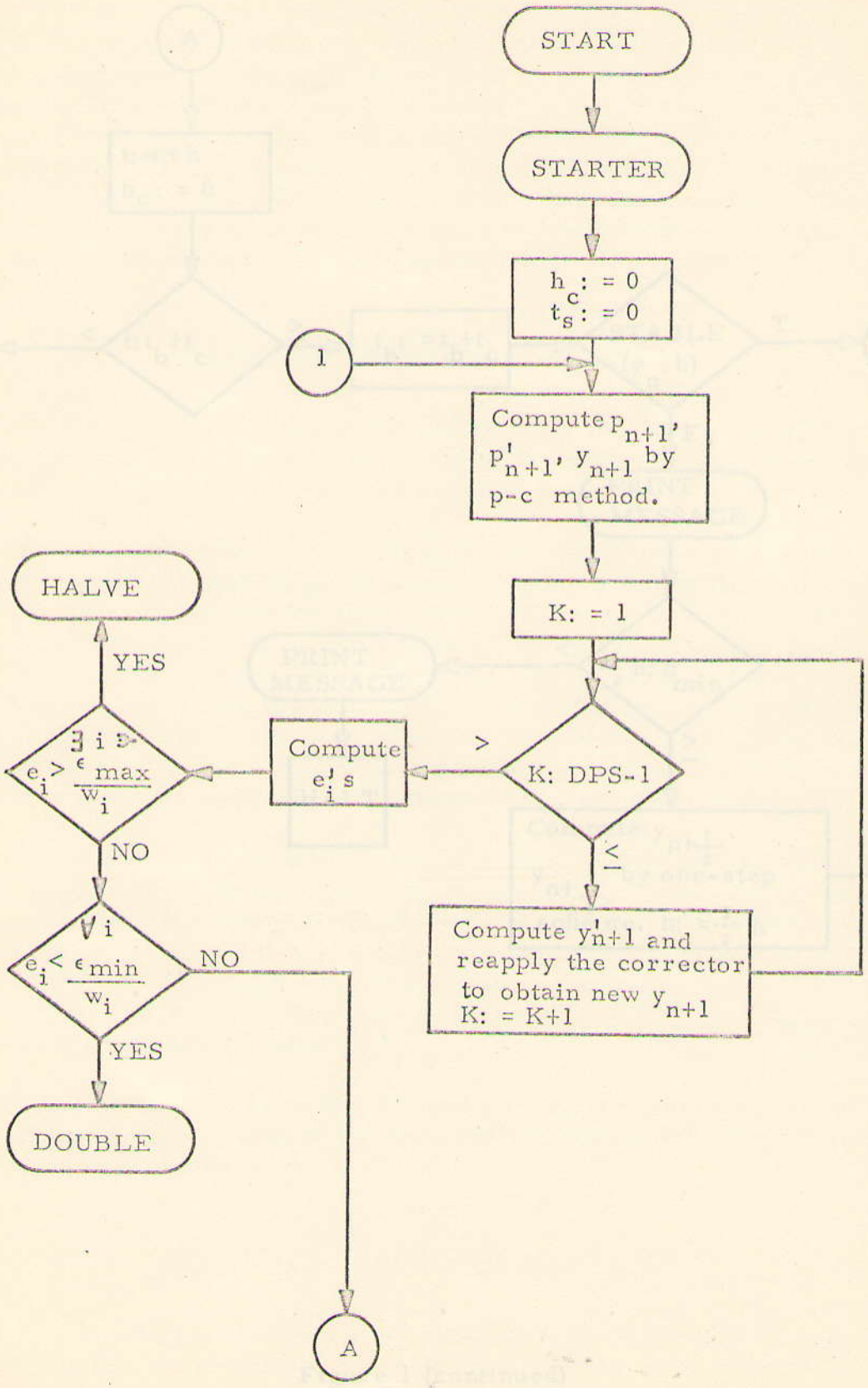
START
STARTER
$h_c := 0$
$t_s := 0$
1
Compute p_{n+1}, p'_{n+1}, y_{n+1} by p-c method.
K: = 1
K: DPS-1
>
≤
Compute e'_i s
$\exists i \ni e_i > \frac{\epsilon}{w_i}$ max
YES
HALVE
NO
$\forall i$ $e_i < \frac{\epsilon}{w_i}$ min
NO
YES
DOUBLE
Compute y'_{n+1} and reapply the corrector to obtain new y_{n+1}
K: = K+1
A

Figure 1

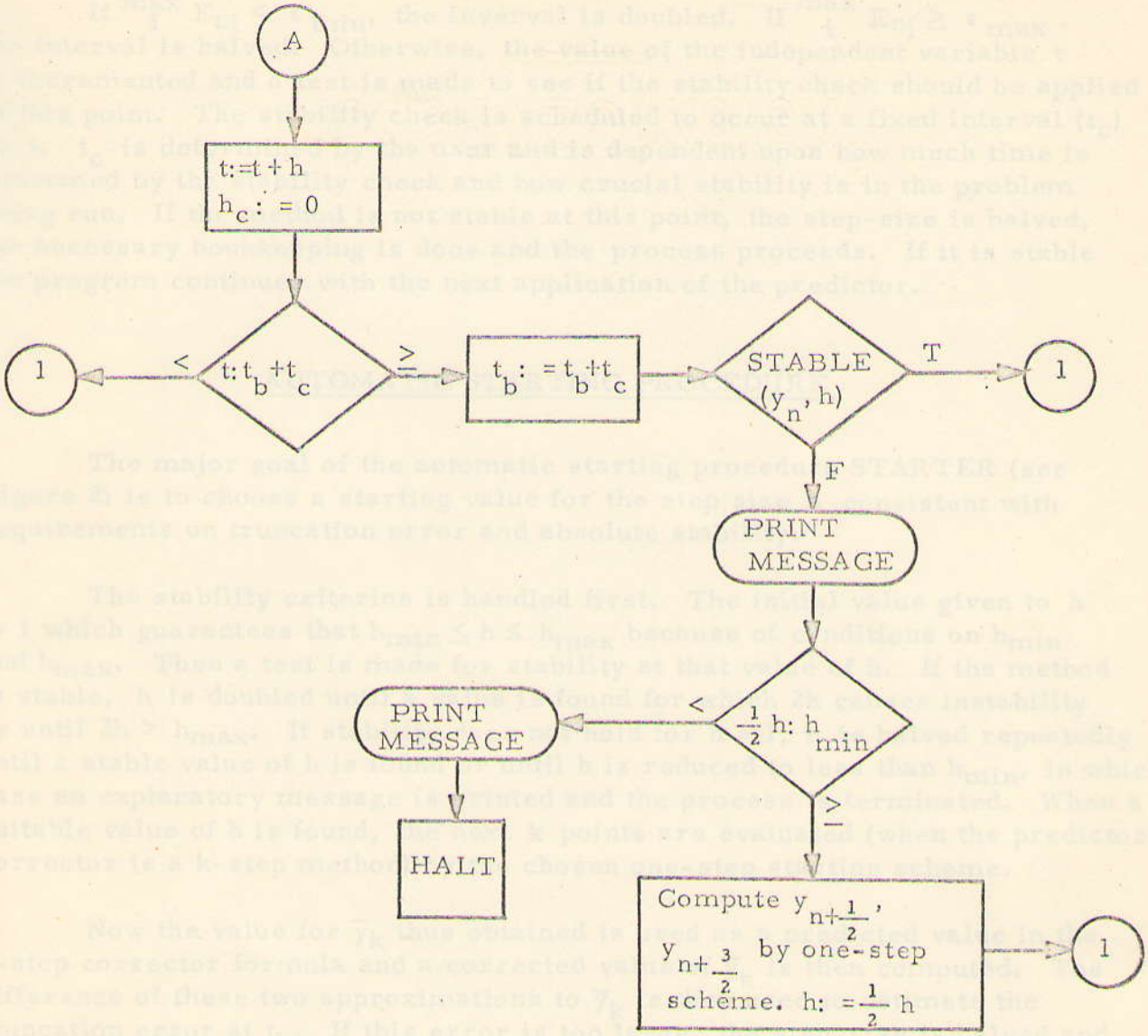

Figure 1 (continued)

where the w_i are weight factors to be determined by the user ($0 \leq w_i \leq 1$).

If $\max_i E_{n_i} < \epsilon_{min}$, the inverval is doubled. If $\max_i E_{n_i} \geq \epsilon_{max}$, the interval is halved. Otherwise, the value of the independent variable t is incremented and a test is made to see if the stability check should be applied at this point. The stability check is scheduled to occur at a fixed interval (t_c) on t. t_c is determined by the user and is dependent upon how much time is consumed by the stability check and how crucial stability is in the problem being run. If the method is not stable at this point, the step-size is halved, the necessary bookkeeping is done and the process proceeds. If it is stable the program continues with the next application of the predictor.

AUTOMATIC STARTING PROCEDURE

The major goal of the automatic starting procedure STARTER (see Figure 2) is to choose a starting value for the step size h consistent with requirements on truncation error and absolute stability.

The stability criterion is handled first. The initial value given to h is 1 which guarantees that $h_{min} \leq h \leq h_{max}$ because of conditions on h_{min} and h_{max}. Then a test is made for stability at that value of h. If the method is stable, h is doubled until a value is found for which 2h causes instability or until $2h > h_{max}$. If stability does not hold for h = 1, h is halved repeatedly until a stable value of h is found or until h is reduced to less than h_{min}, in which case an explanatory message is printed and the process is terminated. When a suitable value of h is found, the next k points are evaluated (when the predictor-corrector is a k-step method) by the chosen one-step starting scheme.

Now the value for $\bar{y}_k$ thus obtained is used as a predicted value in the k-step corrector formula and a corrected value of $\bar{y}_k$ is then computed. The difference of these two approximations to $\bar{y}_k$ is then used to estimate the truncation error at t_k. If this error is too large, the step-size is halved and the one-step method is again applied starting at the initial point. This process is continued until a suitable step-size is found or h falls below h_{min}. In the latter case the process is halted.

It should be noted here that the estimate of the truncation error obtained in this manner at t_k will tend to be conservative so that the starting step size may be smaller than necessary.

STABILITY CHECKING PROCEDURE

A method is known to be stable at a given point if all the zeros of the characteristic polynomial of the method lie within or on the unit circle and those lying on the unit circle are simple [3]. The coefficients of the characteristic polynomial involve terms containing $\bar{h}$ where $\bar{h} = h \cdot \lambda_i$. The λ_i are

STARTER:

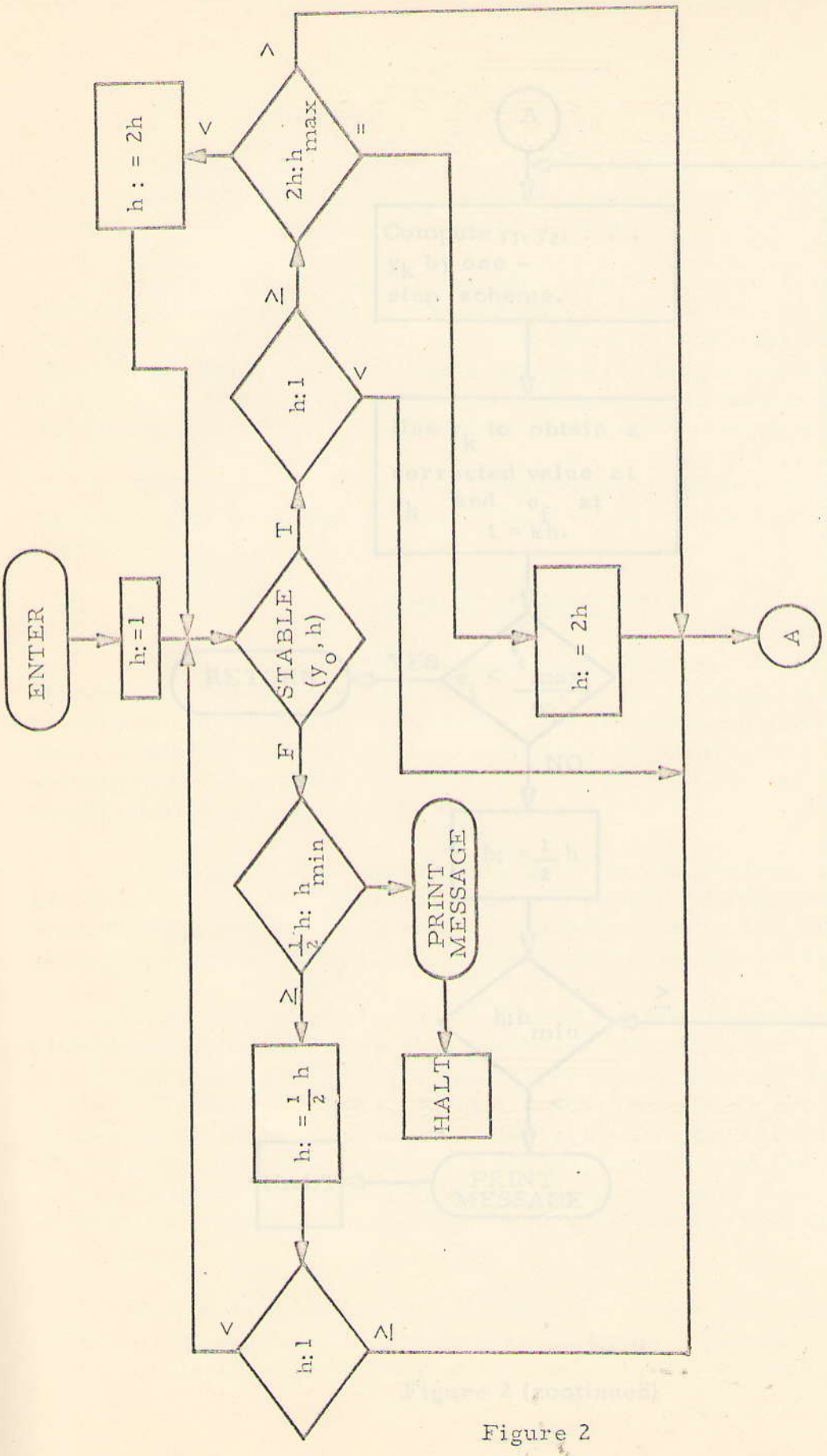

Figure 2

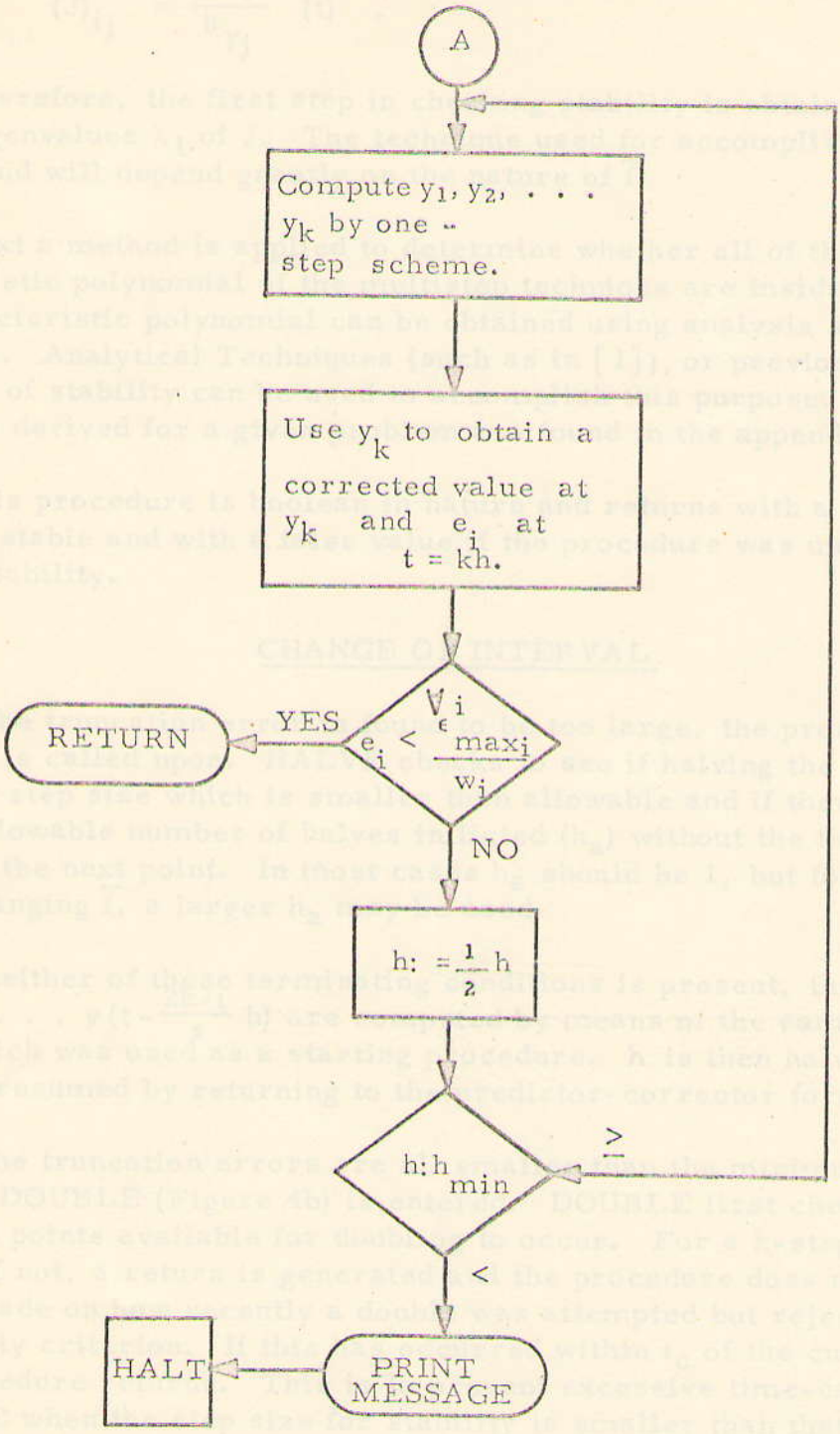

Figure 2 (continued)

eigenvalues of the matrix J which can be represented by

$$(J)_{ij} = \frac{\partial f_i}{\partial y_j} \quad (t) \quad .$$

Therefore, the first step in checking stability is obtaining approximations for the eigenvalues λ_i of J. The technique used for accomplishing this is up to the user and will depend greatly on the nature of $\bar{f}$.

Next a method is applied to determine whether all of the zeros of the characteristic polynomial of the multistep technique are inside the unit circle. The characteristic polynomial can be obtained using analysis similar to that of Chase [2]. Analytical Techniques (such as in [1]), or previous knowledge of the region of stability can be used to accomplish this purpose. Examples of techniques derived for a given problem are found in the appendices.

This procedure is boolean in nature and returns with a true value if the method is stable and with a false value if the procedure was unable to verify absolute stability.

CHANGE OF INTERVAL

If the truncation error is found to be too large, the procedure HALVE (Figure 4) is called upon. HALVE checks to see if halving the step size would result in a step size which is smaller than allowable and if there have been more than the allowable number of halves inflicted (h_a) without the integration proceeding to the next point. In most cases h_a should be 1, but for extremely rapidly changing $\bar{f}$, a larger h_a may be used.

If neither of these terminating conditions is present, then $\bar{y}(t-\frac{1}{2}h)$, $\bar{y}(t-\frac{3}{2}h)$, . . . $y(t-\frac{2k-1}{2}h)$ are computed by means of the same one-step method which was used as a starting procedure. h is then halved and the integration is resumed by returning to the predictor-corrector formulas.

If the truncation errors are all smaller than the minimum tolerable, procedure DOUBLE (Figure 4b) is entered. DOUBLE first checks that there are enough points available for doubling to occur. For a k-step method "enough" is 2k-1. If not, a return is generated and the procedure does nothing. Next a check is made on how recently a double was attempted but rejected because of a stability criterion. If this has occurred within t_c of the current value of t, the procedure returns. This is to prevent excessive time-consuming calls on STABLE when the step size for stability is smaller than that required by truncation error.

Next stability is checked for step size 2h and 2h is compared with h_{max} to insure that doubling would not cause the bound on h to be exceeded. If both of these tests allow, h is doubled and the necessary bookkeeping handled.

STABLE: (y, h):

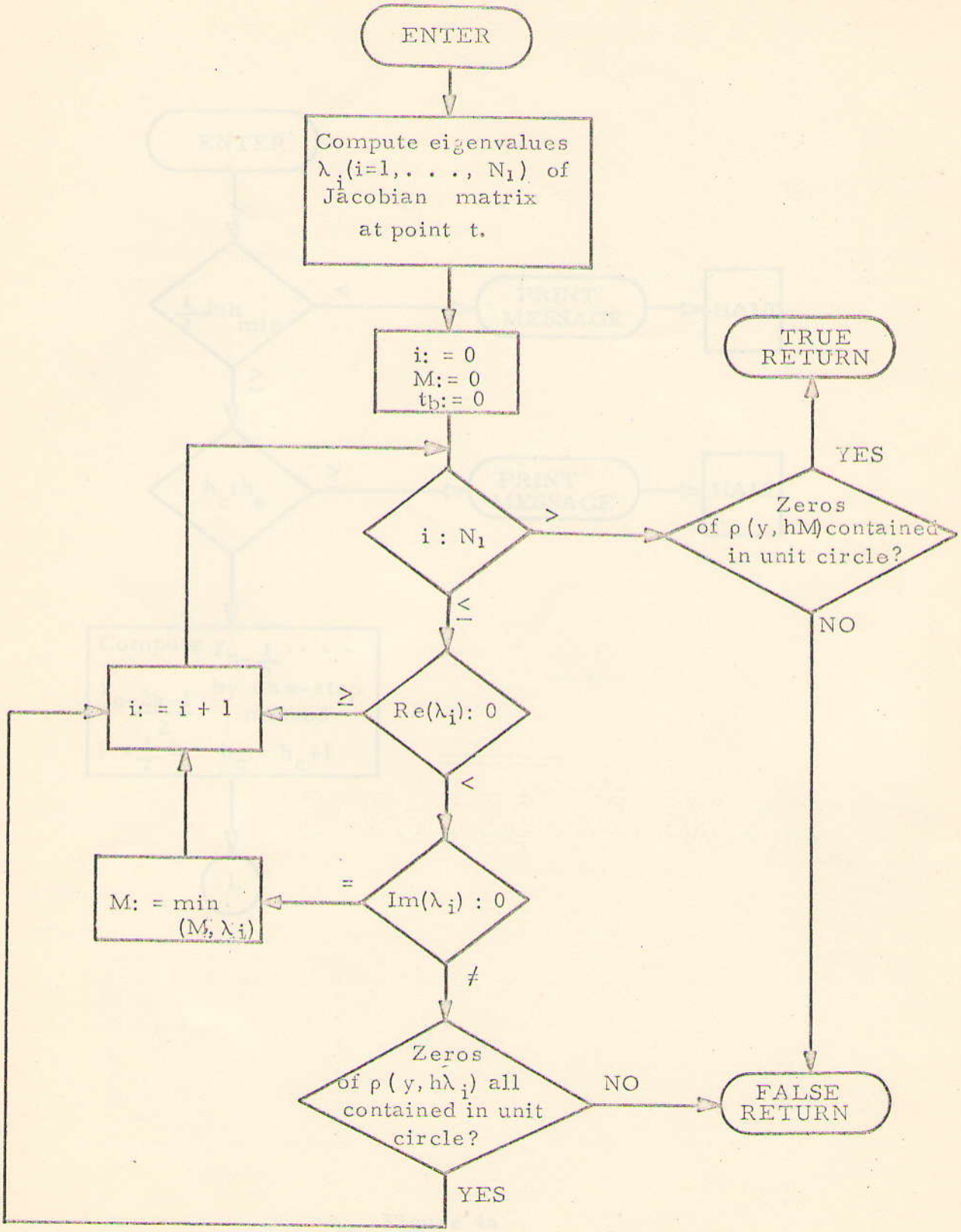

Figure 3

HALVE:

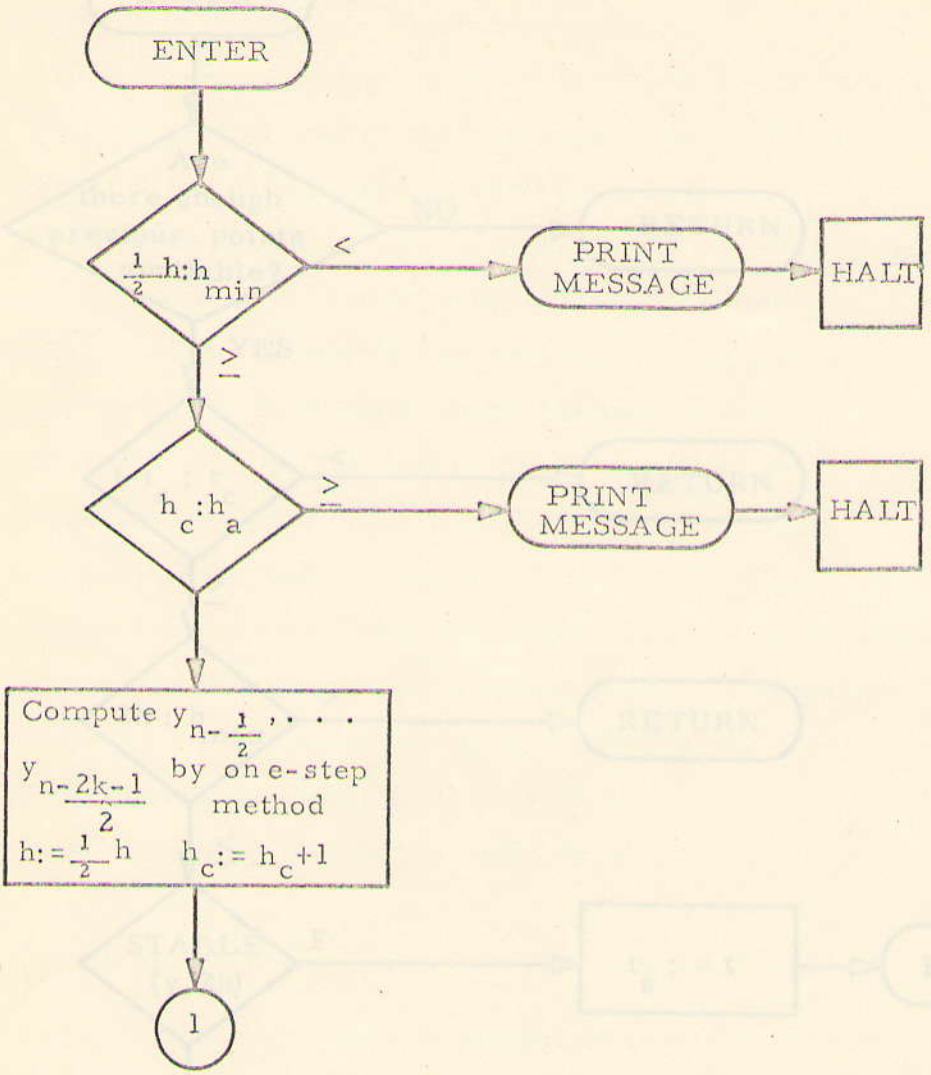

Figure 4a

DOUBLE:

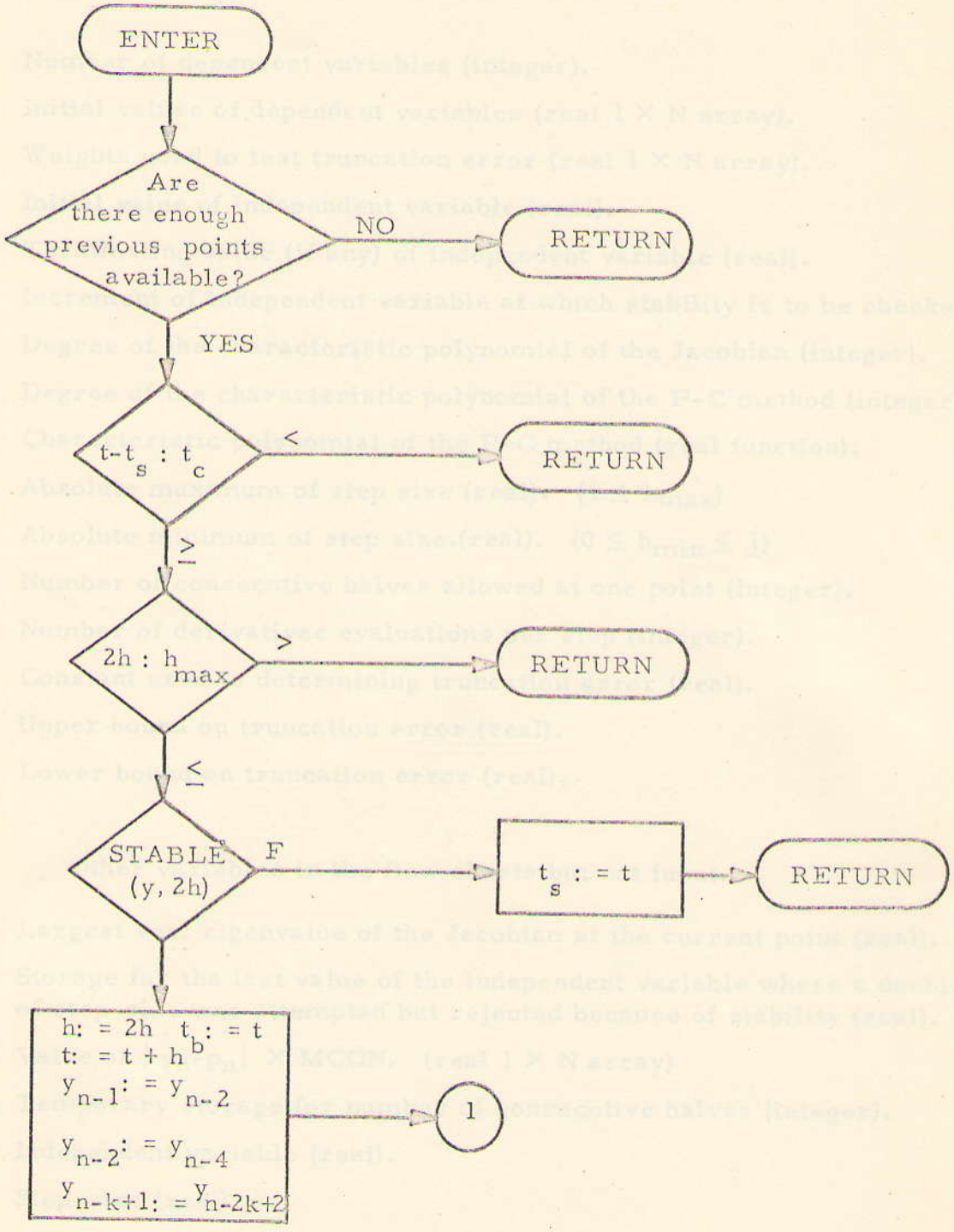

Figure 4b

TABLE 1

INPUT TO PREDICTOR-CORRECTOR PROGRAM

N	Number of dependent variables (integer).
y_o	Initial values of dependent variables (real 1 × N array).
w	Weights used to test truncation error (real 1 × N array).
t_o	Initial value of independent variable (real).
t_f	Terminating value (if any) of independent variable (real).
t_c	Increment of independent variable at which stability is to be checked (real).
N_1	Degree of the characteristic polynomial of the Jacobian (integer).
N_2	Degree of the characteristic polynomial of the P-C method (integer).
$\rho(y, \bar{h})$	Characteristic polynomial of the P-C method (real function).
h_{max}	Absolute maximum of step size (real). $(1 \leq h_{max})$
h_{min}	Absolute minimum of step size.(real). $(0 \leq h_{min} \leq 1)$
h_a	Number of consecutive halves allowed at one point (integer).
DPS	Number of derivatives evaluations per step (integer).
MCON	Constant used in determining truncation error (real).
ϵ_{max}	Upper bound on truncation error (real).
ϵ_{min}	Lower bound on truncation error (real).

Other variables in the flow charts but not input.

M	Largest real eigenvalue of the Jacobian at the current point (real).
t_s	Storage for the last value of the independent variable where a double of step-size was attempted but rejected because of stability (real).
e	Value of $\lvert y_n - p_n \rvert \times$ MCON. (real 1 × N array)
h_c	Temporary storage for number of consecutive halves (integer).
t	Independent variable (real).
h	Step-size (real).
t_b	Value of the independent variable at the previous stability check (real).

APPENDIX A

STABILITY CRITERION FOR A SAMPLE PROBLEM

Let $\bar{y} = (y_1, y_2, y_3, y_4, y_5, y_6)$ and $\bar{f} = (f_1, f_2, f_3, f_4, f_5, f_6)$. Then consider the system

$$\bar{y}' = \bar{f}(t, \bar{y})$$

where

$$\begin{aligned}
f_1(t, \bar{y}) &= y_4 \\
f_2(t, \bar{y}) &= y_5 \\
f_3(t, \bar{y}) &= y_6 \\
f_4(t, \bar{y}) &= -E(y_4 - w_x) - A_z y_5 + A_y y_6 - G_r y_1 \\
f_5(t, \bar{y}) &= -E y_5 - g \\
f_6(t, \bar{y}) &= -E(y_6 - w_x) - A_y y_4 + A_x y_5
\end{aligned}$$

where

$$E = E(y_2, y_4, y_5, y_6) = \gamma \cdot k \cdot \rho \cdot V \cdot K_D$$

$$= \gamma \cdot k \cdot \rho\,(y_2)\, V(y_4, y_5, y_6) \cdot K_D(y_4, y_5, y_6).$$

Now γ and k are constants, ρ depends on y_2 from a table, K_D depends on V from a table and

$$V = [(y_4 - w_x)^2 + y_5^2 + (y_6 - w_z)^2]^{1/2}.$$

Therefore

$$\frac{\partial E}{\partial y_2} = \gamma \cdot V \cdot k \cdot K_D \cdot \frac{\partial \rho}{\partial y_2}$$

$$\frac{\partial E}{\partial y_i} = k \cdot \gamma \cdot \rho \left[\frac{\partial V}{\partial y_i} K_D + \frac{\partial K_D}{\partial V} \cdot \frac{\partial V}{\partial y_i} V \right], \quad (i = 4, 5, 6.)$$

But

$$\frac{\partial V}{\partial y_4} = \frac{y_4 - w_x}{V}, \quad \frac{\partial V}{\partial y_5} = \frac{y_5}{V}, \quad \frac{\partial V}{\partial y_6} = \frac{y_6 - w_z}{V}$$

$$\therefore \quad \frac{\partial E}{\partial y_4} = k \cdot \gamma \cdot \rho \cdot (y_4 - w_x) \left[\frac{K_D}{V} + \frac{\partial K_D}{\partial V} \right]$$

$$\frac{\partial E}{\partial y_5} = k \cdot \gamma \cdot \rho \cdot y_5 \left[\frac{K_D}{V} + \frac{\partial K_D}{\partial V}\right]$$

$$\frac{\partial E}{\partial y_6} = k \cdot \gamma \cdot \rho \cdot (y_6 - w_z) \left[\frac{K_D}{V} + \frac{\partial K_D}{\partial V}\right] .$$

$\frac{\partial K_D}{\partial V}$ and $\frac{\partial \rho}{\partial y_2}$ can be obtained from the tables used to find K_D and ρ. If interpolation is linear the slope is constant between any two tabular values.

Now let

$$f_{i_j} = \frac{\partial f_i}{\partial y_j} .$$

We compute the following:

$$f_{44} = - E - (y_4 - W_x) \frac{\partial E}{\partial y_4}$$

$$f_{45} = - A_z - (y_4 - W_x) \frac{\partial E}{\partial y_5}$$

$$f_{46} = + A_y - (y_4 - W_x) \frac{\partial E}{\partial y_6}$$

$$f_{54} = - y_5 \frac{\partial E}{\partial y_4}$$

$$f_{55} = - E - y_5 \frac{\partial E}{\partial y_5}$$

$$f_{56} = - y_5 \frac{\partial E}{\partial y_6}$$

$$f_{64} = - A_y - (y_6 - W_z) \frac{\partial E}{\partial y_4}$$

$$f_{65} = A_x - (y_6 - W_z) \frac{\partial E}{\partial y_5}$$

$$f_{66} = - E - (y_6 - W_z) \frac{\partial E}{\partial y_6} .$$

For

$$i = 4, 5, 6,$$

$$\frac{df_i}{dy_2} = \frac{dy'_{i-3}}{dy_2} = \frac{\frac{dy'_{i-3}}{dt}}{\frac{dy_2}{dt}} = \frac{y''_{i-3}}{y'_2} \quad .$$

Also,

$$\frac{df_i}{dy_2} = \sum_{j=1}^{6} f_{ij} \frac{dy_j}{dy_2} = \frac{y''_{i-3}}{y'_2}$$

$$f_{i_2} = \left(\frac{y''_{i-3}}{y'_2} - \sum_{\substack{j=1 \\ j\neq i}}^{6} f_{ij} \frac{dy_j}{dy_2}\right) \frac{1}{\frac{dy_i}{dy_2}}$$

$$= \left(\frac{y''_{i-3}}{y'_2} - \sum_{j=1}^{3} f_{ij} \frac{y'_j}{y'_2} - \sum_{\substack{j=4 \\ j\neq i}}^{6} f_{ij} \frac{y''_{i-3}}{y'_2}\right) \frac{y'_2}{y''_{i-3}} \quad .$$

Hence all f_{i_2} can be computed.

Therefore, the Jacobian Matrix J is

$$J = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -G_r & f_{42} & 0 & f_{44} & f_{45} & f_{46} \\ 0 & f_{52} & 0 & f_{54} & f_{55} & f_{56} \\ 0 & f_{62} & 0 & f_{64} & f_{65} & f_{66} \end{bmatrix} .$$

The characteristic polynomial of this matrix is

$$\begin{aligned} \rho(t) = {} & t^6 + (f_{44}+f_{55}+f_{66})\, t^5 + (f_{44}f_{55} + f_{44}f_{66} + f_{55}f_{66} - f_{46}f_{64} - f_{45}f_{54} \\ & - f_{65}f_{56} - f_{52} + G_r)\, t^4 + (f_{44}f_{55}f_{66} + f_{45}f_{56}f_{64} + f_{46}f_{65}f_{54} \\ & - f_{64}f_{46}f_{55} - f_{45}f_{54}f_{66} - f_{44}f_{65}f_{56} + f_{54}f_{42} + f_{56}f_{62} - f_{44}f_{52} \\ & - f_{66}f_{52} + G_r f_{55} + G_r f_{66})\, t^3 + (f_{54}f_{42}f_{66} + f_{44}f_{56}f_{62} + f_{46}f_{64}f_{52} \\ & - f_{54}f_{46}f_{62} - f_{44}f_{66}f_{52} - f_{64}f_{56}f_{42} + G_r f_{55}f_{66} + G_r f_{52} - G_r f_{65}f_{56})\, t^2 \\ & - (G_r f_{56}f_{62} - G_r f_{66}f_{52})\, t \quad . \end{aligned}$$

APPENDIX B

TESTING TO SEE IF ZEROS OF A POLYNOMIAL LIE INSIDE THE UNIT CIRCLE

Two possible techniques will be discussed for locating zeros of the characteristic polynomial. The first is more time consuming, but also more accurate than the second.

The first is based upon a technique suggested by Cain (1), and uses the concept of the winding number of the polynomial on the unit circle. This is accomplished by evaluating the polynomial at certain small increments around the circle and counting the number of times the image curve winds around the origin. The increment used in the actual program was $\frac{\pi}{20}$, but it was decreased whenever both the real and imaginary part of the image changed sign over a single increment.

This second technique uses prior knowledge of the region of stability to determine directly from the eigenvalues whether or not the method is stable. For example, one could simply check the largest circle inside the region to see if all eigenvalues were contained in it. This would allow considerable savings when calculating the eigenvalues since only the largest modulus need be found.

REFERENCES

1) Cain, G. L., "A Method for Locating Zeros of Complex Functions," Comm. ACM Vol. 9, No. 4, 1966, pp. 305-306.

2) Chase, P. E., "Stability Properties of Predictor-Corrector Methods for Ordinary Differential Equations," Journal ACM, Vol. 9, 1962, pp. 457-468.

3) Crane, R. L., and Klopfenstein, R. W. "A Predictor-Corrector Algorithm with an Increased Range of Absolute Stability," Journal ACM, Vol. 12, 1965, pp. 227-241.

4) Gear, C. W., "The Numerical Integration of Ordinary Differential Equations of Various Orders," Report #ANL-7126, Argonne National Laboratory, Argonne, Ill., 1966.

5) Nordsieck, A., "On Numerical Integration of Ordinary Differential Equations," Math. Comp., Vol. 16, 1962, pp. 22-49.

6) Ralston, A., "Runge-Kutta Methods with Minimum Error Bounds," Math. Comp., Vol. 16, 1962, pp. 431-437.