DOCUMENT RESUME

ED 024 575 SE 004 989

By- Hull, T. E.

Monograph - The Numerical Integration of Ordinary Differential Equations.

Committee on the Undergraduate Program in Mathematics, Berkeley, Calif.

Spons Agency- National Science Foundation, Washington, D.C.

Pub Date 66

Note- 34p.

EDRS Price MF-$0.25 HC-$1.80

Descriptors- *College Mathematics, Computer Assisted Instruction, *Curriculum, *Instructional Materials, Mathematical Concepts, *Mathematics

Identifiers- National Science Foundation

The materials presented in this monograph are intended to be included in a course on ordinary differential equations at the upper division level in a college mathematics program. These materials provide an introduction to the numerical integration of ordinary differential equations, and they can be used to supplement a regular text on this subject. The techniques provided are important in applications for finding approximate solutions to problems for which analytic solutions either cannot be found or are too difficult to evaluate. (RP)

COMMITTEE ON THE UNDERGRADUATE PROGRAM IN MATHEMATICS

MONOGRAPH

THE NUMERICAL INTEGRATION OF ORDINARY DIFFERENTIAL EQUATIONS

by

T. E. Hull

MATHEMATICAL ASSOCIATION OF AMERICA

Financial support for the
Committee on the Undergraduate Program in Mathematics
has been provided by the
National Science Foundation

THE NUMERICAL INTEGRATION OF ORDINARY DIFFERENTIAL EQUATIONS

T.E. Hull
Professor of Mathematics and Computer Science
University of Toronto

The Committee on the Undergraduate Program in Mathematics
P. O. Box 1024, Berkeley, California 94701

CONTENTS

Chapter 1. Purpose 4

Chapter 2. Outline 5

Chapter 3. Euler's Procedure 7

Chapter 4. Convergence of Euler's Procedure 9

Chapter 5. Error Analysis of Euler's Procedure 11

Chapter 6. Second-Order Runge-Kutta Procedures 15

Chapter 7. Higher-Order Runge-Kutta Procedures 18

Chapter 8. Adams Predictor-Corrector Procedures 21

Chapter 9. General Predictor-Corrector Procedures 24

Chapter 10. Other Procedures 26

Chapter 11. Error Control 27

Chapter 12. Programming Considerations 30

Bibliography 31

Chapter 1
PURPOSE

Most students taking advanced mathematics will take a course on ordinary differential equations at about the Junior level. The material presented in this monograph is intended to be included in such a course. It provides an introduction to the numerical integration of ordinary differential equations, and can be used to supplement a regular text on this subject.

The material on numerical integration could be given at almost any stage in the course. It would be preferable to give it in the latter half, after the student has gained some familiarity with the analytical behavior of solutions of ordinary differential equations.

Some of the exercises require the preparation of computer programs. These problems can be omitted, but it would be highly desirable to include them if possible.

There is considerable motivation for including material on numerical analysis in a course on ordinary differential equations. For anyone with any interest at all in applications there is the very practical reason of using the methods to find approximate solutions to those problems for which analytic solutions either cannot be found, or are too difficult to evaluate. Computing machines have made the task much easier by taking over practically all of the tedious parts of the numerical work. There is additional motivation for the mathematician because the field is rich in results which are of mathematical interest, especially in connection with questions of convergence and stability, and mathematical results in this area are increasing steadily.

Chapter 2
OUTLINE

We begin by describing one very simple procedure, known as Euler's procedure, which generates approximations to the solutions of ordinary differential equations. Convergence of this procedure is then discussed, and the basic existence theorem for ordinary differential equations is stated without proof.

We then find a bound for the error which is propagated through a calculation with Euler's procedure. Many of the important features of more complicated methods are already present in the results for Euler's procedure.

Although Euler's procedure does converge (under suitable circumstances), its rate of convergence is not good enough for numerical work. In Chapters 6 through 9 we therefore introduce the two main classes of numerical methods which are in general use. They are the Runge-Kutta and Predictor-Corrector methods, and they will appear as natural generalizations of Euler's procedure. In each case an error analysis is given, which is quite similar to the one given for Euler's procedure.

Brief mention is made of other methods in Chapter 10. Error control and a few programming considerations are discussed briefly in Chapters 11 and 12. References to the literature are given in the bibliography.

Throughout it is intended that the relative merits of the various procedures be made as clear as possible. The merits are judged in terms of realistic criteria, as they arise in the construction and use of computer programs. We will be concerned mainly with accuracy and reliability. In principle, any method (as long as it converges) can be made arbitrarily accurate, and reasonably reliable. But it is much more expensive to do so with some than with others, and therefore the relative merits of different procedures depend mainly on cost. We will also be concerned to some extent with convenience.

It should be emphasized, however, that even a good computer program will not do everything. There is no substitute for sound analysis of the original problem, and careful preparation of that problem for computing. This may involve judicious changes of variable, special treatment of singularities, awareness of inherent instabilities in the differential equations, and so on. But it is equally important to understand well the nature and limitations of the numerical procedures that one is going to use. Our concern here will be only with what is relevant to the numerical procedures.

For convenience the discussion is mainly in terms of single first-order equations with initial conditions. Except for two relatively unimportant restrictions which will be mentioned specifically later on, it is quite easy to apply the results we obtain to systems of first-order equations with initial conditions. Some scalar quantities must be replaced by vectors, others by matrices, and some absolute values must be replaced by norms of vectors or matrices. Moreover, higher order equations can also be handled, because they can be replaced by systems of first-order equations.

Only initial-value problems will be considered. Procedures for initial-value problems are often

used as a basis for "hit-or-miss" attempts to solve boundary-value problems, especially when the latter are non-linear. Otherwise the handling of boundary-value problems usually involves matrix calculations, and the subject is therefore more appropriate to a course on linear algebra.

Chapter 3
EULER'S PROCEDURE

Let the differential equation be denoted by

$$y' = f(x, y),$$

and the associated initial condition by

$$y(x_0) = y_0.$$

We must have a procedure for generating approximate solutions. The simplest is Euler's procedure, which is based on the following recurrence:

$$y_0^* = y_0,$$
$$y_i^* = y_{i-1}^* + h\, y_{i-1}^{*\prime}, \quad i = 1, 2, 3, \ldots,$$

Here h is the "step-size", $y_i^{*\prime} = f(x_i, y_i^*)$, and y_i is the approximation to the value of the true solution $y_i = y(x_i)$.

It is convenient to define the approximate solution $y^*(x)$ to be the polygon one obtains by joining all pairs of points (x_{i-1}, y_{i-1}^*) and (x_i, y_i^*) by straight line segments. We then have the situation shown in Figure 1.

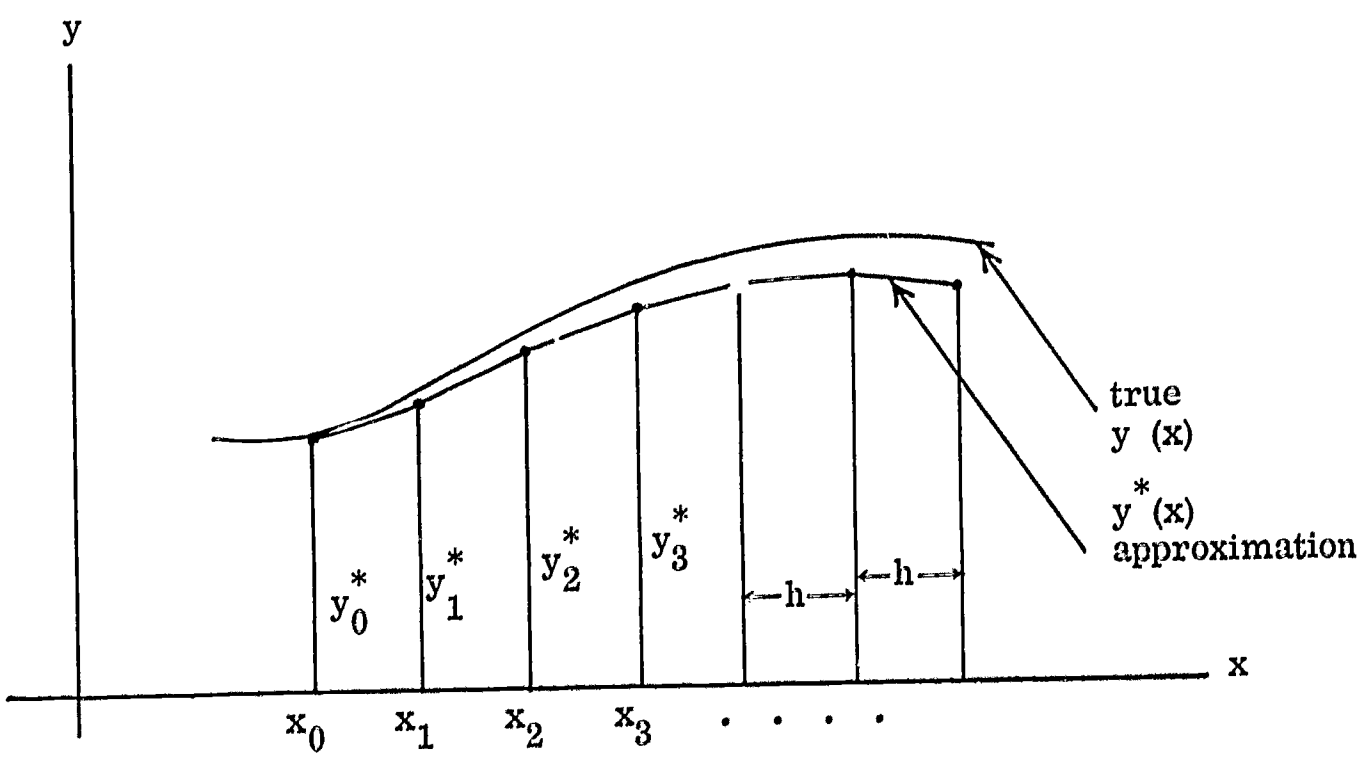

FIGURE 1. Graphs of the true solution and a polygonal approximation to the solution.

In the next two chapters we discuss the convergence of Euler's procedure, and obtain estimates of the error $y^* - y$. Although Euler's procedure is not good enough for numerical work it is a

convenient example to use for establishing the basic results we will need. It will then be relatively easy to extend these results to the more sophisticated methods which are used in practice.

EXERCISES

1. Write a computer program based on Euler's procedure to provide an approximation to $y(4)$, where $y' = 1-y$ and $y(0) = 0$. Use $h = 1/4$.
2. Repeat the calculation in (1) with $h = 1/8$, and with $h = 1/16$.
3. On the basis of the results in (1) and (2), how good an approximation can you be reasonably sure of having?

Chapter 4
CONVERGENCE OF EULER'S PROCEDURE

The polygonal approximation provided by Euler's procedure depends on the step-size h. Of course we expect the approximation to be better for smaller values of h. To be more precise we need a theorem which guarantees the convergence of Euler's method, i.e. which guarantees that the approximations will approach the true solution as h approaches zero.

The theorem can be established for functions $f(x,y)$ which satisfy the following two conditions:

(1) Continuity condition. The function $f(x,y)$ is a continuous function of both x and y, in some region, say R, of the x-y plane which contains the point (x_0, y_0). This means in particular that f is bounded in R, i.e. there is a constant M such that $|f(x,y)| < M$ in R.

(2) Lipschitz condition. The function $f(x,y)$ satisfies a Lipschitz condition in R with respect to y, i.e. there is a constant L such that $|f(x,y)-f(x,\bar{y})| < L|y-\bar{y}|$ for any (x,y) and $(x,\bar{y})$ in R.

The theorem then guarantees convergence over a certain interval of the x-axis. The interval is most easily described by means of the diagram in Figure 2, where it is denoted by I.

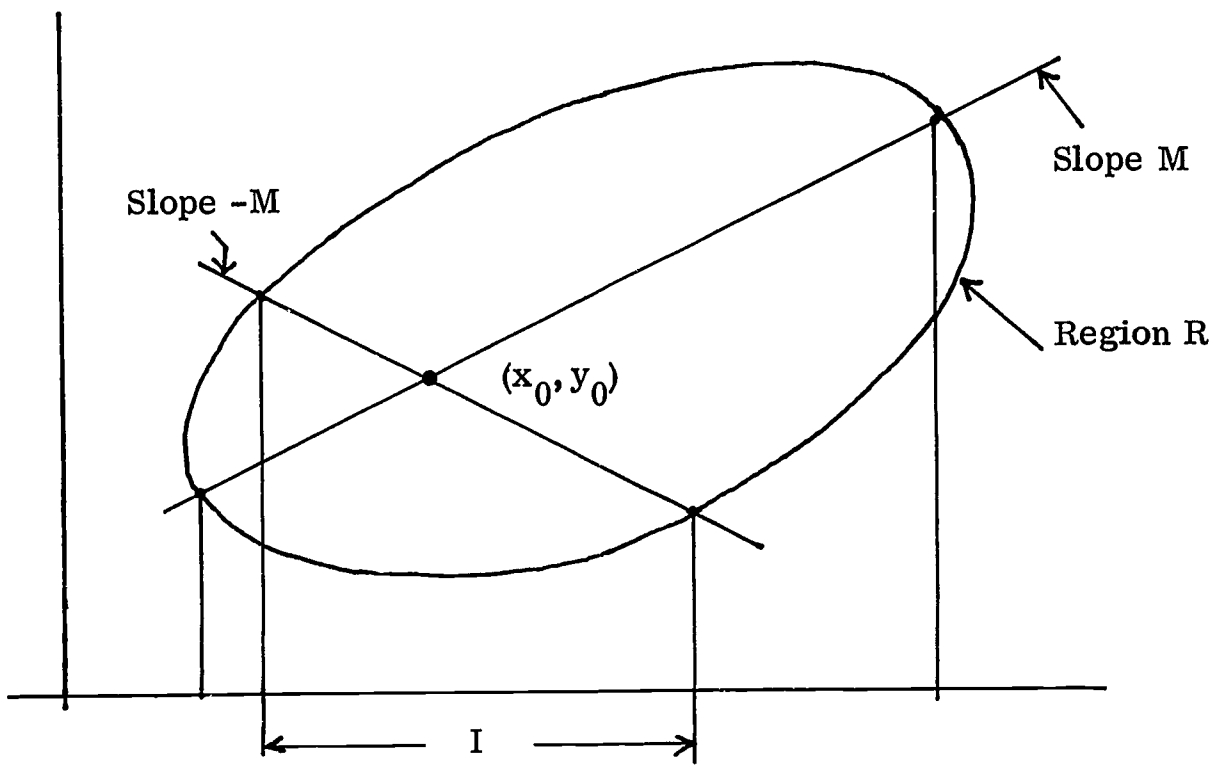

FIGURE 2. The interval I depends on the region R, the bound M and the initial point (x_0, y_0).

The straight lines through (x_0, y_0) with slopes M and -M determine four sectors of the x-y plane. The interval I extends to the left as far as possible while keeping in R all the points of the "western" sector which are above I. Similarly I extends to the right, keeping in R the points of the "eastern" sector which are above I.

The theorem can now be stated as follows:

THEOREM: If $f(x,y)$ satisfies the continuity and Lipschitz conditions then the Euler approximations

converge in I to the unique solution of the differential equation which satisfies the initial condition. The proof will be omitted, but a reference is given in the bibliography.

Our statement of the theorem emphasizes the convergence of the Euler procedure because this is so important from the numerical point of view. But the theorem can be appreciated from another point of view, since it guarantees the existence and uniqueness of a solution, under the stated conditions. Thus the proof of convergence of a numerical procedure can, at the same time, be an existence proof.

When Euler approximations are used to prove existence the theorem is associated with the names of Cauchy and Lipschitz. The details of the proof show that only the continuity condition is needed to prove existence, whereas the Lipschitz condition is needed only for the proof of uniqueness. (Both conditions can be weakened slightly.) Another form of the existence and uniqueness proof is due to Picard, but it uses quite different approximations which are not related to a common numerical procedure.

Chapter 5

ERROR ANALYSIS OF EULER'S PROCEDURE

From now on we assume that $f(x,y)$ satisfies the continuity and Lipschitz conditions in some region of the x-y plane which contains (x_0, y_0). In the absence of rounding errors this leads us to consider values of x in the interval I. But we do not want to ignore the effect of rounding errors and we therefore assume that the values of x are restricted somewhat further. The purpose is only to ensure that $f(x,y)$ is still bounded by M, and that the Lipschitz condition still holds for all points of the x-y plane that enter into the calculations.

In a numerical calculation one does not deal with the values of $y^*(x)$, since these values are defined by a procedure which uses exact arithmetic. To include the effect of rounding errors we introduce a new approximation $y^{**}(x)$ which is defined by the same procedure, except that rounding errors are allowed. Of course these errors depend on details of the computer program and of the machine being used. But in any given instance it is usually possible to find a bound for the rounding error in an individual step in the calculation, and this is often all that is required. In any event we can define the rounding error in the i-th step of Euler's procedure to be r_i where

$$y_i^{**} = y_{i-1}^{**} + hf(x_{i-1}, y_{i-1}^{**}) - r_i.$$

The rounding error r_i is therefore just the amount by which the Euler equation is not satisfied by y^{**}. In an analogous way we can define the truncation error T_i by means of

$$y_i = y_{i-1} + hf(x_{i-1}, y_{i-1}) + T_i.$$

The truncation error is then the amount by which the Euler equation is not satisfied by the true solution. (The signs for r_i and T_i were chosen merely for later convenience.)

If $f(x,y)$ is sufficiently differentiable Taylor's formula gives

$$T_i = \frac{h^2}{2} y'(\xi),$$

where $x_{i-1} \leqq \xi \leqq x_i$. Of course one would not in general know the numerical value of the truncation error. However, we will make use of the fact that it is $0(h^2)$ as $h \to 0$, at least when $f(x,y)$ is sufficiently differentiable.

Subtracting our two equations we obtain

$$y_i - y_i^{**} = y_{i-1} - y_{i-1}^{**} + h\,[f(x_{i-1}, y_{i-1}) - f(x_{i-1}, y_{i-1}^{**})] + r_i + T_i$$

We now define the error $e_i = y_i - y_i^{**}$, and we define g_i to be 0 if $y_i - y_i^{**} = 0$, otherwise by $f(x_i, y_i) - f(x_i, y_i^{**}) = g_i(y_i - y_i^{**})$. Then our error equation becomes

$$e_i = (1 + hg_{i-1})e_{i-1} + r_i + T_i.$$

Let us now introduce the difference equation

$$E_i = A\,E_{i-1} + B,$$

where $|1 + hg_i| \leqq A$, and $|r_i + T_i| \leqq B$.
It is clear that $|e_i| \leqq E_i$, provided $|e_0| \leqq E_0$. The equation for E_i is called a dominating difference equation for the error.

We obtain

$$E_1 = A E_0 + B,$$
$$E_2 = A^2 E_0 + (A+1)B$$
$$\cdot$$
$$\cdot$$
$$\cdot$$

so that

$$E_i = A^i E_0 + \frac{A^i - 1}{A-1} B,$$

provided $A \neq 1$. (No special result is needed for the case $A = 1$, since it can be obtained from the result for $A \neq 1$, by taking the limit at $A = 1$.)

We find it convenient to assume $h > 0$. We will thereby avoid having to write $|h|$ in place of h in a large number of places.

An interesting result is obtained by taking $A = 1 + hL$. If moreover $|r_i| < r$ and $|T_i| < T$ we can take $B = r + T$. We also note that $(1 + hL)^i < e^{L(x_i - x_0)}$. Putting $E_0 = |e_0|$ we then obtain

$$|e_i| < e^{L(x_i - x_0)} |e_0| + \frac{e^{L(x_i - x_0)} - 1}{hL} (r + T).$$

The main contribution to the right side is

$$\frac{r + T}{hL} \quad e^{L(x_i - x_0)}.$$

Of course we have derived only a bound for the propagated error; the error itself may be very much less than the bound. However, in some circumstances the error can also approach the bound, and the term given above is a good indication of how the various factors contribute to the accumulated error. The exponential factor in this term is to be expected when g is close to L, because then the differential equation itself is close to $y' = Ly + f(x)$, and the general solution of this equation includes a term proportional to e^{Lx}. Since $T = 0(h^2)$ when $f(x, y)$ is sufficiently differentiable, we see that the bound will decrease as h decreases, until the contribution due to rounding becomes dominant, at which point a further decrease in h causes the bound to increase. This behavior of the bound, and often of the error itself, is quite typical.

Of course we have been considering only a bound, and the preceding discussion cannot be expected to apply to all cases. One important possibility is that the value taken for A, namely $1 + hL$, may be much too crude. The value of g may be quite variable so that most of the time $|g| << L$ and the bound is therefore much too pessimistic. But more than that, it could happen that $-1 < hg < -h\ell < 0$, so that we could put $A = 1 - h\ell$. This leads to

$$|e_i| < e^{-\ell(x_i - x_0)} |e_0| + \frac{1 - e^{-\ell(x_i - x_0)}}{h\ell} (r + T).$$

in which the main contribution to the right side is

$$\frac{r + T}{h\ell}.$$

The decrease of error with h, until rounding is dominant, is as before. But we no longer have the exponential growth with $x_i - x_0$. (With systems of equations, instead of a single equation, it is the dominant eigenvalue of the matrix $I + hg$ which plays the role of A in this discussion.)

So far we have used only a bound for r_i. In practice the rounding errors accumulate in a way very much like the accumulation of random quantities. And this means, very roughly speaking, that the accumulated rounding error will be proportional to $r/\sqrt{i}$, after i steps, instead of r. We need not consider any further details of this phenomenon because the truncation error is almost

always the most important contribution. If it is not, one should be using a larger value of h, or one should be carrying more significant figures in the calculation.

Finally we now consider briefly how one would use Euler's procedure in practice. Apart from what has already been said about the method, the main problem in practice is to control the errors. One way to do this is to find several approximations using different step-sizes, say h, h/2, h/4. Here h is a value which one would have to choose on the basis of qualitative knowledge of the solution of the problem. For example, noting that Euler's method is exact only for straight line solutions, we need to choose h to be small enough so that straight line segments can approximate the solution sufficiently closely. On the other hand, there is no point in taking h so small that the rounding errors will be dominant.

Once several solutions have been obtained one can compare them at several values of x. If they agree to a number of decimal digits which is accurate enough for the purpose, these results can probably be taken to be satisfactory. If they do not agree it may still be possible to extrapolate from several results to a better one, especially if these results correspond to values of h for which the propagated error is approximately proportional to a power of h, as it probably will be if the truncation error is dominant. In the case of Euler's procedure it will be the first power of h in most cases.

The main disadvantage of this approach is that the same value of the step-size is being used throughout each calculation with the differential equation. This means that no advantage is taken of the possibility that larger step-sizes might be suitable for parts of the calculation, thereby reducing the cost of the calculation. We will return to this possibility in Chapter 11.

EXERCISES

1. Use a computer program (such as the one developed for Exercise 1 of Chapter 3) to solve a number of differential equations. In each case results should be obtained for several different values of h, including some that are somewhat extreme in order to show the change of error with h both when truncation is dominant and when rounding error is dominant.

   The following problems can be used as examples:

   (a) $y' = (\cos x)y$, $y(0) = 1$, Exact solution: $y = e^{\sin x}$

   (b) $y' = 2xy$, $y(0) = 1$, Exact solution: $y = e^{-x^2}$

   (c) $y' = y - \frac{2x}{y}$, $y(0) = 1$, Exact solution: $y = \sqrt{2x + 1}$.

   (d) $y' = y^2$, $y(0) = 1$, Exact solution: $y = \frac{1}{1-x}$.

   You should have trouble with the third equation if you try to integrate to $x = 100$. Why? How close can you get to the singularity in the fourth equation?

2. Develop a program for handling systems of equations and try it on:

   (a) $y' = z$, $y(0) = 0$, Exact solution: $y = \sin x$

   $z' = -y$, $z(0) = 1$, $z = \cos x$

   (b) $y' = z/(y^2 + z^2)^{3/2}$, $y(0) = 0$, Exact solution: $y = \sin x$

   $z' = -y/(y^2 + z^2)^{3/2}$, $z(0) = 1$, $z = \cos x$

3. Integrate each of the following to $x = 4$.

(a) $y'' + 3y' + 2y = 0$

(b) $y'' - 3y - 4y = 0$

(c) $y'' + 101y' + 100y = 0$

If the initial conditions are $y(0) = 1$, $y'(0) = -1$, the exact solution is $y = e^{-x}$ in each case. Why are the numerical approximations so different from each other?

Chapter 5

SECOND-ORDER RUNGE-KUTTA PROCEDURES

So far only Euler's procedure has been used to generate approximations to the required solution. Analogous results can be obtained for other procedures which are more accurate than Euler's. (By a more accurate procedure we mean one in which the truncation error is smaller. This does not necessarily mean that the propagated error will be smaller.) Most of our new results are simple generalizations of what we obtained for Euler's procedure. However, some new features arise, including the possibility of procedures which do not converge.

There are two main generalizations of Euler's method. In this section and the next we consider one which produces Runge-Kutta methods. (Special cases were developed originally by C. Runge, W. Kutta, and K. Heun in the period 1895-1901.) In this section we develop the second-order Runge-Kutta methods. They will be described with the help of figure 3.

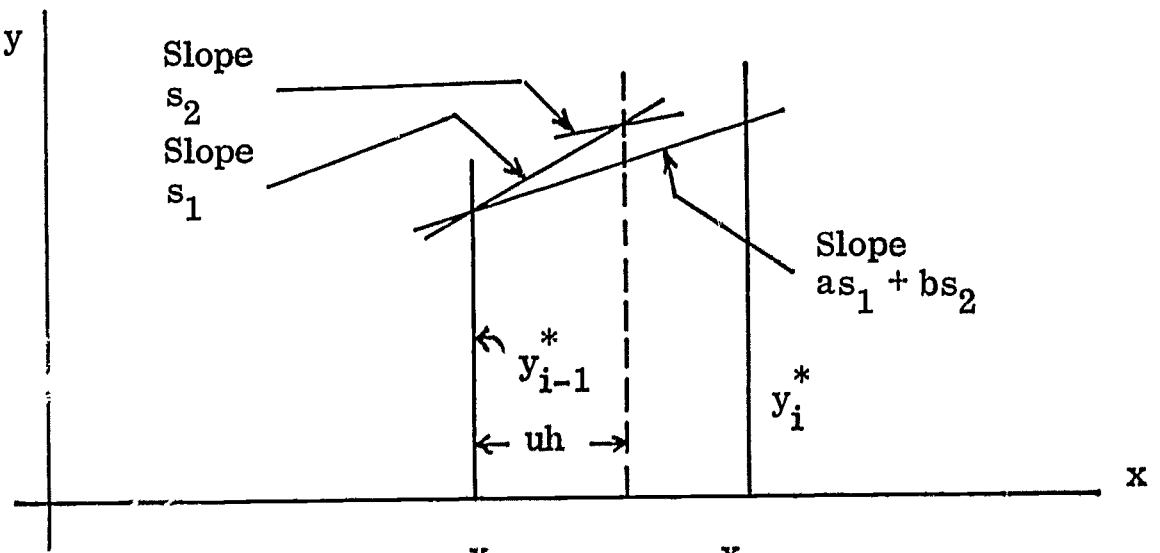

FIGURE 3. A geometrical interpretation of second-order Runge-Kutta methods.

Suppose the calculations have been carried as far as the point (x_{i-1}, y^*_{i-1}). From here Euler's procedure would use the slope

$$s_1 = f(x_{i-1}, y^*_{i-1})$$

to continue the polygon all the way across the next step of length h. We would expect it to be more accurate if we used this slope to go only part of the way, say a step of length uh, and then evaluated the slope s_2 at the intermediate point, i.e.

$$s_2 = f(x_{i-1} + uh, y^*_{i-1} + uhf(x_{i-1}, y^*_{i-1}))$$

and then finally used some linear combination of these two slopes to make the full step. This leads us to

$$y^*_i = y^*_{i-1} + h(as_1 + bs_2)$$

as the basic recurrence formula. The question of choosing the fraction u, and the weights a and b, still remains.

In a standard notation the procedure is defined by

$$y_i^* = y_{i-1}^* + ak_0 + bk_1,$$

where

$$k_0 = hf(x_{i-1}, y_{i-1}^*),$$
$$k_1 = hf(x_{i-1} + uh,\ y_{i-1}^* + uk_0).$$

To obtain values of the parameters a, b and u, we substitute the true solution y in place of y^* in both sides of the formula. The true solution does not satisfy this equation in general, but, if the functions involved are sufficiently differentiable, the two sides can be expanded in powers of h, and we can choose the parameters so that the coefficients on either side are equal, up to as high a power of h as possible. The coefficients of h^0 are already equal. For the coefficients of h^1 to be equal we find that

$$a + b = 1,$$

a requirement which would have been expected from our geometrical considerations. For the coefficients of h^2 to be equal we find that $2bu = 1$. To make the coefficients of h^3 equal we are led to two more equations which must be satisfied by a, b and u, and, not surprisingly, it turns out that all four equations cannot be satisfied simultaneously. Taking only the first two equations, and solving for a and b in terms of u, leads us to the following one-parameter family of methods:

$$y_i^* = y_{i-1}^* + \frac{2u-1}{2u} k_0 + \frac{1}{2u} k_1,$$

where

$$k_0 = hf(x_{i-1},\ y_{i-1}^*),$$
$$k_1 = hf(x_{i-1} + uh,\ y_{i-1}^* + uk_0).$$

If the functions involved are sufficiently differentiable, the truncation error associated with these methods turns out to be

$$T_i = \frac{h^3}{12} [3uf_y y'' + (2u-3)y'''] + 0(h^4),$$

where f_y, y'' and y''' can be evaluated at any points within distances from $(x_i,\ y_i^*)$ which are 0(h).

It is natural to try to choose the free parameter so that the truncation error is in some sense minimized. But for this family of methods there does not seem to be a simple connection between this idea and any specific value of u. In any event we would expect u to be somewhere between 1/2 and 1. With $u = 1$ we obtain the following special case:

$$y_i^* = y_{i-1}^* + (1/2)k_0 + (1/2)k_1,$$

where

$$k_0 = hf(x_{i-1},\ y_{i-1}^*),$$
$$k_1 = hf(x_{i-1} + h,\ y_{i-1}^* + k_0),$$

with

$$T_i = \frac{h^3}{12} [3f_y y'' - y'''] + 0(h^4).$$

Proof of the convergence of any of these methods can be carried through as with Euler's method. If we assume existence to begin with, the proof is very much simplified. Under this circumstance the best approach is to first find a bound for the propagated error. It turns out that with $e_0 = 0$ and $r = 0$, this bound approaches zero as h approaches zero, and this fact establishes the convergence of the method, merely as a by-product of the error analysis.

The derivation of the bound follows closely the one used with Euler's procedure. To simplify the formulas a little we will consider only the special case obtained with $u = 1$, but the argument is applicable to any member of the family. The rounding errors are defined by

$$y_1^{**} = y_{i-1}^{**} + (1/2)k_0^{*} + (1/2)k_1^{**} - r_i,$$

where the asterisks on the k's have an obvious significance. The truncation error is defined by

$$y_i = y_{i-1} + (1/2)k_0 + (1/2)k_1 + T_i.$$

On subtracting and making the same substitutions as with Euler's method we obtain the following error equation:

$$e_i = (1 + (1/2)hg + (1/2)hg + (1/2)hghg)e_{i-1} + r_i + T_i.$$

The different appearances of g in this equation do not in general have the same values. Nevertheless each one represents a value of

$$\frac{f(x, y) - f(x, z)}{y - z}$$

(or 0, if $y = z$) for some y and z near to y_{i-1} and y_{i-1}^{**}. This means, for example, that each value of $|g|$ is bounded by L, and this in turn means that we could put $A = 1 + hL + \frac{h^2L^2}{2}$. Since in this case $A^i < e^{L(x_i - x_0)}$ we are led to the following bound:

$$|e_i| < e^{L(x_i - x_0)} |e_0| + \frac{e^{L(x_i - x_0)} - 1}{hL} (r + T).$$

This bound is exactly the same as with Euler's method, except that now the value of T will in general be smaller.

Note that the question of convergence is concerned only with the case where $e_0 = 0$ and $r = 0$. In this case the bound will clearly approach zero as h approaches zero, as long as $T = o(h)$. This is certainly true of T when the functions involved are sufficiently differentiable, for then $T = 0(h^3)$. In other cases a proof must be supplied, but as one would expect it follows from the conditions we have imposed on $f(x, y)$. Any method for which $T = o(h)$, at least for the class of functions $f(x, y)$ which we have been considering, is said to be "consistent." It is left as an exercise to show that the condition $a + b = 1$ is all that is needed to guarantee consistency for the methods of this section.

EXERCISES

1. Verify the calculations needed to obtain the one-parameter family of this section, and also the expression for the truncation error.
2. How would you modify the error bound in case $g < 0$? If $-\ell = g < 0$, what is the largest value of h for which you can be sure that the propagated error will not increase exponentially with $x_i - x_0$?
3. Show that any second-order Runge-Kutta method is consistent, as long as $a + b = 1$.
4. Suppose that rounding errors are neglected. Then show that, for any particular problem, a member of the one-parameter family given in this section will always be better than Euler's method, as long as one's accuracy requirements are stringent enough. (By "better" we mean it will give the same accuracy at less cost.)
5. Modify your earlier program to use the special Runge-Kutta method given in this section in place of Euler's. Use it on some of the same examples and compare the results with those obtained by Euler's method.

Chapter 7

HIGHER-ORDER RUNGE-KUTTA PROCEDURES

The discussion of second-order methods can now be generalized in a straightforward way to higher-order methods. For example, instead of using a combination of the two slopes s_1 and s_2 to take the full step h, a combination of these slopes could be used to take another partial step, of size vh say. Then a third slope could be obtained at this new point, and finally the three slopes could be combined to give an average slope for taking the full step. This leads to the class of third order Runge-Kutta methods, which can be defined by:

$$y_i^* = y_{i-1}^* + ak_0 + bk_1 + ck_2,$$

where

$$k_0 = hf(x_{i-1},\ y_{i-1}^*),$$
$$k_1 = hf(x_{i-1} + uh,\ y_{i-1}^* + uk_0),$$
$$k_2 = hf(x_{i-1} + vh,\ y_{i-1}^* + wk_0 + (v-w)k_1).$$

It is left as an exercise to find the four equations which must be satisfied by the six parameters a, b, c, u, v and w, in order to ensure that the truncation error is $0(h^4)$. A special solution of these equations due to Heun yields the method:

$$y_i^* = y_{i-1}^* + (1/4)\ (k_0 + 3k_2),$$

where

$$k_0 = hf(x_{i-1},\ y_{i-1}^*),$$
$$k_1 = hf(x_{i-1} + (1/3)h,\ y_{i-1}^* + (1/3)k_0),$$
$$k_2 = hf(x_{i-1} + (2/3)h,\ y_{i-1}^* + (2/3)k_1).$$

Another special case, due to Kutta is:

$$y_i^* = y_{i-1}^* + (1/6)\ (k_0 + 4k_1 + k_2),$$

where

$$k_0 = hf(x_{i-1},\ y_{i-1}^*).$$
$$k_1 = hf(x_{i-1} + \frac{h}{2},\ y_{i-1}^* + \frac{k_0}{2}),$$
$$k_2 = hf(x_{i-1} + h,\ y_{i-1}^* - k_0 + 2k_1).$$

The four equations for the parameters will in general have a two-parameter family of solutions. (There also are some singular solutions.) It is natural to ask if the two free parameters can be chosen in a way so that the truncation error is in some sense minimized. The problem is not precisely defined, but the small amount of experimental evidence that is available suggests that there is a little that can be done in this respect, but also that the choice is not particularly critical. The commonly used methods seem to be reasonably well chosen.

By allowing still another intermediate evaluation of the slope, one is led to fourth-order methods. This time there are 10 parameters and 8 conditions with $T = 0(h^5)$. The derivation of the equations is straightforward but extremely tedious. The best known example from this family of methods, and probably the most frequently used of all Runge-Kutta methods, is due to Kutta. It is defined by:

$$y_i^* = y_{i-1}^* + (1/6)\,(k_0 + 2k_1 + 2k_2 + k_3),$$

where

$$k_0 = hf(x_{i-1},\ y_{i-1}^*),$$
$$k_1 = hf(x_{i-1} + \frac{h}{2},\ y_{i-1}^* + \frac{k_0}{2}),$$
$$k_2 = hf(x_{i-1} + \frac{h}{2},\ y_{i-1}^* + \frac{k_1}{2}),$$
$$k_3 = hf(x_{i-1} + h,\ y_{i-1}^* + k_2).$$

Again there is evidence to suggest that this method is a reasonable choice from amongst all possible fourth-order methods. With care one might reduce the truncation error by a factor of 2 or so, over some fairly general class of problems, but the possible advantage is not very large, and anyway the question is not really settled.

In all of the above examples the error bounds and the resulting proof of convergence are almost exactly as before. (See Exercise 2.) However, one additional point should be kept in mind. We will illustrate by considering the simple differential equation $y' = -\ell y$. Here the solution is a decreasing function of x. But the error is proportional to the i-th power of some polynomial in ℓh. When ℓh is small this polynomial is close to $e^{-\ell h}$, as one would expect. But when ℓh is large this polynomial can be considerably larger than $e^{-\ell h}$, and in fact it can even be larger than 1. The procedure is said to be unstable. It is relative instability if the error dominates the solution. It is absolute instability if the error not only dominates the solution but is also growing exponentially. The phenomenon was hinted at in Exercise 2 of the preceding chapter.

We can continue the generalizations to still higher orders, but the algebra becomes very formidable. It turns out that allowing five function evaluations does not lead to $T = 0(h^6)$, but only to $T = 0(h^5)$, as with four evaluations. One needs six evaluations per step to achieve $T = 0(h^6)$. It also turns out that the largest possible value of p in $T = 0(h^{p+1})$ may depend on whether the formulas are being applied to single equations, or to systems of equations. This anomaly does not appear if the largest value of p is less than five. (This "largest value" is usually called the order of the method, but we prefer to call it the degree, to be consistent with later sections. We will use the term "order" to denote the number of evaluations per step. Order and degree are the same when they are less than five.)

EXERCISES

1. Find the four equations for the parameters of third-order methods. Verify that the two examples given in the text do satisfy the requirements. Can you find the two-parameter, and the one-parameter families of solutions of these equations?
2. Verify that expressions for the error bound can be found which are exactly the same for the Heun method and the fourth-order Kutta method as was found earlier for Euler's method. What is the effect on the error bound for the third-order Kutta method of having a negative value for v?
3. The following method was once proposed:

$$y_i^* = y_{i-1}^* + k_3,$$

where

$$k_0 = hf(x_{i-1}, y_{i-1}^*),$$
$$k_1 = hf(x_{i-1} + \frac{h}{4}, y_{i-1}^* + \frac{k_0}{4}),$$
$$k_2 = hf(x_{i-1} + \frac{h}{3}, y_{i-1}^* + \frac{k_1}{3}),$$
$$k_3 = hf(x_{i-1} + \frac{h}{2}, y_{i-1}^* + \frac{k_2}{2}).$$

What is wrong with it?

4. Show that, when $f(x, y)$ satisfies our usual continuity and Lipschitz conditions, then every Runge-Kutta method will be consistent, as long as the sum of the coefficients of $k_0, k_1, \ldots$ is equal to one.

5. For what range of values of x is $1 + x + x^2/2 + x^3/6 > 1$? For what range is it $> e^x$?

Chapter 8

ADAMS PREDICTOR-CORRECTOR PROCEDURES

There is a second class of methods which can be obtained as natural generalizations of Euler's method. We will introduce it in the following way. We know from the mean value theorem that

$$y_i = y_{i-1} + hf(\xi, y(\xi)),$$

where ξ satisfies $x_{i-1} \leq \xi \leq x_i$. We obtain a numerical procedure if we replace this exact result by

$$y_i^* = y_{i-1}^* + hs,$$

where s is some average of known values of $f(x, y)$ near the point (x_{i-1}, y_{i-1}^*). Euler's method defines s to be $f(x_{i-1}, y_{i-1}^*)$, and we have already described the special way in which s is obtained for Runge-Kutta methods.

As long as we are not near the beginning of the calculation, we will always have, in addition to $f(x_{i-1}, y_{i-1}^*)$, some earlier values of the slope, such as $f(x_{i-2}, y_{i-2}^*)$, $f(x_{i-3}, y_{i-3}^*)$, and so on. It is natural to consider defining s in terms of these earlier values. If we denote $f(x_j, y_j^*)$ by f_j^*, we are led to consider a formula such as

$$y_i^* = y_{i-1}^* + h(af_{i-1}^* + bf_{i-2}^* + cf_{i-3}^*).$$

Of course we still have to choose the parameters a, b, and c. But this time the process of substituting y in place of y^* and expanding each side in powers of h is very much simpler than it was for Runge-Kutta methods. If we expand about x_i in this example, and equate coefficients of powers of h, we obtain the following:

$$\text{from } h^1\text{:} \quad a + b + c = 1,$$
$$\text{from } h^2\text{:} \quad a + 2b + 3c = 1/2,$$
$$\text{from } h^3\text{:} \quad a + 4b + 9c = 1/3.$$

After solving for a, b and c, we finally obtain

$$y_i^* = y_{i-1}^* + \frac{h}{12}(23 f_{i-1}^* - 16f_{i-2}^* + 5f_{i-3}^*).$$

We can also show that

$$T_i = \frac{9}{24}h^4 y^{(4)}(x_i) + 0(h^5).$$

The details of the above derivation will be left as an exercise.

Four points are involved in the above formula, and it is called a third-order formula. It is a straightforward matter to derive formulas of higher order. Results like these are usually obtained with the aid of finite difference formulas, but the development here is more direct. A procedure similar to the one obtained above was first used by J. C. Adams in work on capillary action which was published with F. Bashforth in 1883.

Procedures like the above are extrapolation procedures. It turns out that their performance can be improved if they are used in conjunction with certain interpolation formulas. An appropriate formula in this case is obtained by keeping terms involving f_{i-1}^* and f_{i-2}^* and adding a term involving f_i^*. The derivation is almost exactly as before, and one obtains the following second-order formula:

$$y_i^* = y_{i-1}^* + \frac{h}{12}(5f_i^* + 8f_{i-1}^* - f_{i-2}^*),$$

along with

$$T_i = \frac{-1}{24} h^4 y^{(4)}(x_i) + 0(h^5).$$

The reason why it is appropriate to use a second-order interpolation formula in conjunction with a third-order extrapolation formula will be explained later when error control is being considered. It turns out to be very convenient if the two truncation errors are proportional to the same power of h. The largest value of p for which $T = 0(h^{p+1})$ is the degree of a formula.

Again it is a straightforward matter to derive higher-order formulas.

It is clear that the interpolation formula is more accurate than the corresponding extrapolation formula. This situation is typical. It would be better to use the interpolation formula, but there is a difficulty because the interpolation formula defines y_i^* only implicitly. This leads us to the idea of a predictor-corrector method, which uses an extrapolation formula to "predict" an approximation to y_i, and then uses this "predicted" value to "evaluate" an approximate to f_i, which is then used in the right side of an interpolation formula to produce a "corrected" value of the approximation to y_i. The corrected value can then be used to produce a new evaluation of f, which leads to a new corrected value, and so on. The process can be represented schematically by PECEC..., for Predict, Evaluate, Correct, Evaluate, Correct, and so on.

A decision must be made about when to stop the process. Procedures based on the decision represented by PE do not use the corrector at all, and are known as Adams-Bashforth procedures. The original proposal for predictor-corrector methods was of the PECE type, and was used by F. R. Moulton for ballistics calculations during the first world war. Procedures of this type are often called Adams-Moulton procedures. There is evidence to suggest that either PECE or PECEC is best in practice. PEC would be best, except that it is prone to instability, a phenomenon which we will discuss in the next section. More than two evaluations of $f(x, y)$ per step seem to be not worthwhile.

Of course we must also decide on the degree of the predictor-corrector formulas. This choice depends on one's accuracy requirements, the more stringent the requirement the higher the degree. In practice one usually uses formulas of degree 3, as in the examples of this section, or possibly a little higher.

We will now carry out an error analysis of the Adams methods based on the formulas of this section. In order to avoid a lot of detail we will content ourselves with deriving a bound which is slightly larger than we would like to have.

We can define r_i as before with

$$y_i^{**} = y_{i-1}^{**} + \frac{h}{12}(5f_i^{**} + 8f_{i-1}^{**} - f_{i-2}^{**}) - r_i,$$

but this time r_i includes more than just the effect of rounding; it also includes the effect of not iterating indefinitely on the corrector. We define T_i by means of

$$y_i = y_{i-1} + \frac{h}{12}(5f_i + 8f_{i-1} - f_{i-2}) + T_i.$$

Subtracting, and substituting as before, we obtain

$$e_i = \frac{(1 + 8hg/12)}{1 - 5hg/12} e_{i-1} - \frac{hg/12}{1 - 5hg/12} e_{i-2} + \frac{r_i + T_i}{1 - 5hg/12},$$

where we have again used the generic symbol g to represent different values of $(f(x,y) - f(x,z))/(y-z)$ in the region of interest.

If $5hL/12 < 1$ the error is dominated by the solution of

$$E_i = AE_{i-1} + B,$$

with

$$A = \frac{1 + 9hL/12}{1 - 5hL/12}, \quad B = \frac{r + T}{1-5hL/12},$$

and with $E_0 = \max(|e_0|, |e_1|)$. A bound is easily found and, for small h, it becomes approximately

$$e^{\frac{7}{6}L(x_i-x_0)} E_0 + \frac{e^{\frac{7}{6}L(x_i-x_0)}-1}{hL} (r + T).$$

This expression is very similar to the bound obtained earlier for Runge-Kutta methods. The factor 7/6 is unfortunate, and turns out to be even worse with higher-order methods. But with more care this factor can be reduced effectively to 1. To show that this is true, and also to deal more carefully with the effect of the predictor and other details, requires a more tedious analysis which we will not consider.

With Adams procedures there is some choice in how you define convergence. The simplest way is to consider only the exact solutions of the corrector equation so that $r = 0$, thus ignoring the predictor entirely. With exact initial conditions we also have $E_0 = 0$. It then follows from the error bounds that Adams procedures do converge, as long as $T = o(h)$ as $h \to 0$. If this last condition does hold the corrector formula is said to be "consistent." (See Exercise 6.)

Before concluding this section we will consider briefly the relative merits of Adams methods and Runge-Kutta methods. The Adams methods are more difficult to use, but they involve only two function evaluations per step. Since this is true even for higher-order methods, it is clear that Adams methods are likely to be much faster than Runge-Kutta methods, at least when one requires relatively high accuracy. (We are assuming that the function $f(x, y)$ is fairly complicated so that its evaluation accounts for most of the computing time.)

EXERCISES

1. Derive the third-order predictor and the second-order corrector given above, along with their truncation errors.
2. Derive predictor and corrector formulas of degree 4, i.e. with $T = 0(h^5)$, and obtain the corresponding truncation errors.
3. Derive an approximate error bound for a method based on the formulas of Exercise 2.
4. Write a program based on the formulas of this section of type PECE. You can use your Runge-Kutta program to provide starting values. Then find new approximate solutions to the earlier examples, and compare the results with what you got with Runge-Kutta.
5. Consider the idealized process of iterating indefinitely with the second-order corrector formula of this section in the absence of rounding errors. Show that the iterations converge if $5hL/12 < 1$.
6. Show that any formula of the form
$$y_i^* = y_{i-1}^* + h(af_i^* + bf_{i-1}^* + \ldots)$$
is consistent for functions which satisfy our continuity and Lipschitz conditions, as long as $a + b + \ldots = 1$.

Chapter 9

GENERAL PREDICTOR-CORRECTOR PROCEDURES

A more general class of predictor-corrector procedures, which includes the Adams procedures as special cases, uses earlier values of the ordinates as well as the slopes. Such general methods are often called multi-step methods.

The predictor formulas are of the form

$$y_i^* = \alpha y_{i-1}^* + \beta y_{i-2}^* + \ldots + h(af_{i-1}^* + bf_{i-2}^* + \ldots),$$

and the correctors are of the form

$$y_i^* = \gamma y_{i-1}^* + \delta y_{i-2}^* + \ldots + h(cf_i^* + df_{i-1}^* + \ldots).$$

One could now proceed as with the Adams formulas, substituting y for y^*, expanding, and matching coefficients of like powers of h. But one soon runs into difficulties. The main difficulty is that the methods obtained in this way will usually be unstable, especially the higher-order methods. We will not be able to consider this question in detail. But it is important to understand the basic idea, and we will therefore give a brief indication of the nature of the instability, and also discuss what this implies regarding a choice of methods.

With Euler's procedure we replaced a first-order differential equation by an approximation which was a first-order difference equation. But now, in trying to achieve higher accuracy, we are replacing the first-order differential equation by a higher-order difference equation. A kth-order difference equation will in general have k independent solutions.

One of these solutions should approximate the solution of the original differential equation, but the other $k - 1$ are extraneous. The trouble is that one of the extraneous solutions may be so large that it completely dominates the calculation.

The theory about this phenomenon can be quite complicated. But in the limiting case when $h \to 0$ it is very precise and elegant. And this case gives important indications about what to expect in the general case when h may be small, but not zero.

As $h \to 0$ the difference equation becomes a linear difference equation with constant coefficients. To illustrate, let us assume that the effect of the predictor is negligible, and let us consider only a third-order corrector formula. Then the limiting difference equation becomes

$$y_i^* = \gamma y_{i-1}^* + \delta y_{i-2}^* + \varepsilon y_{i-3}^*.$$

If we attempt to satisfy this equation with $y_i^* = s^i$, we find that

$$s^3 - \gamma s^2 - \delta s - \varepsilon = 0.$$

If the roots are s_1, s_2, and s_3, the general solution of our difference equation is a linear combination of s_1^i, s_2^i and s_3^i, provided the roots are distinct. (Our argument needs to be modified slightly when the roots are not distinct.) It turns out that one root must be $s_1 = 1$, and this root corresponds to the desired solution of the differential equation. The other two are extraneous. If either of them, say s_2, is much bigger than 1 in magnitude, then it is clear that the solution of the difference equation will in general be dominated by the term s_2^i, at least when i is large.

In the limiting case as $h \to 0$ one is led to identify the notion of stability with the requirement

that the extraneous roots be in or on the unit circle $|s| = 1$ of the complex s- plane, the roots which are on the circle being simple. And there is a remarkable theorem, due to Dahlquist, concerning the maximum degree of a stable formula. It states that, if the corrector formula is stable and of order k, then the maximum possible degree is $k + 1$, except that when k is even it is possible to achieve $k + 2$ in very special circumstances.

All Adams formulas are stable in the limiting sense considered here because their limiting polynomials are $s^k - s^{k-1} = 0$, so that all the extraneous roots are at the origin. Moreover, all Adams corrector formulas are of degree $k + 1$, so we already know that this degree can be achieved with stable formulas. But the theorem states essentially that you cannot improve the degree, even though you have an additional k parameters which you are free to choose.

What you can do with these k parameters is to adjust them in order to reduce the coefficient in the truncation error, while still pre-serving the stability. Except for special cases it turns out that one cannot gain very much in this way, in fact not as much as one can achieve by simply using a higher-order method. For general purposes then, it is probably more economical to use Adams procedures, if one is going to use predictor-corrector methods at all.

A rigorous treatment of the stability problem in the limiting case as $h \to 0$ leads to a fundamental theorem about convergence which states that the two requirements of stability and consistency are necessary and sufficient for convergence.

If we consider the more realistic case when h is not zero we find that the analysis is much more complicated. Nevertheless we are still concerned with the zeros of certain polynomials whose zeros are near to those of the limiting case. The difficulty is that the positions of these zeros change in the course of a calculation, and they also depend on the problem being considered. It is moreover not easy to know whether or not they are too large. However a method can still be defined to be stable, at a particular point in a calculation, if the effect of the extraneous zeros is negligible. It would be relative stability if the effect is negligible in comparison to the solution, but absolute stability if in comparison to 1. The Adams methods are more likely to be stable than any of the others.

The best known example of a general predictor-corrector procedure is due to Milne, and is based on

$$y_i^* = y_{i-4}^* + \frac{4}{3}h\,(2f_{i-1}^* - f_{i-2}^* + 2f_{i-3}^*)$$

for the predictor, which has $T_i = \frac{28}{90}h^5 y^{(5)}(x_i) + 0(h^6)$,

and on

$$y_i^* = y_{i-2}^* + \frac{h}{3}(f_i^* + 4f_{i-1}^* + f_{i-2}^*)$$

for the corrector, which has $T_i = \frac{-1}{90}h^5 y^{(5)}(x_i) + 0(h^6)$.

The corrector formula in Milne's method is one of the exceptional cases when the order k (here it is 2) is even and the degree is $k + 2$. But note that the limiting polynomial in this case is $s^2 - 1$ so that the extraneous zero is at $s = -1$. This can cause serious instability when $h \neq 0$.

EXERCISES

1. Find the corrector formula of order 3 with largest possible degree, and show that it is unstable, at least in the limiting case as $h \to 0$.

2. Find a bound for the propagated error in Milne's method. (If a first-order dominating difference equation is derived in the usual way, the exponential factor in the error bound turns out to be approximately $e^{2L(x_i-x_0)}$. To improve this result one must be more careful and deal directly with the second-order dominating difference equation which follows directly from the second-order error equations.)

3. Show that the corrector formula

$$\sum_{i=0}^{k} a_i y^*_{n-i} = h \sum_{i=0}^{k} b_i f^*_{n-i}$$

is consistent, for functions which satisfy the usual continuity and Lipschitz conditions, provided

$$\sum_{i=0}^{k} a_i = 0, \text{ and } \sum_{i=0}^{k} (ia_i + b_i) = 0.$$

Notice that the first of these ensures a zero at $s = 1$ of the limiting polynomial.

4. Suppose that the corrector formula in Exercise 3 is solved exactly and suppose also that g is constant. Find the error equation, and then the polynomial associated with this error equation. Show that the zero which is at $s = 1$ in the limiting case becomes $e^{hg} + 0(h^{p+1})$ where p is the degree of the corrector formula.

5. Consider the circumstances of exercise 4 and show that Milne's method is unstable if $hg < 0$.

6. If you are not convinced that instability can be catastrophic then write a program to try it. Use the formula of Exercise 1, or use a high-order Adams method with moderately large, but negative hg.

Chapter 10
OTHER PROCEDURES

We will mention very briefly a number of variants of the two classes of methods described so far in this pamphlet.

One can always deal with differential equations of order higher than the first by replacing them with equivalent systems of first-order equations. However some methods, both of the Runge-Kutta and the multi-step types, have also been designed especially for dealing directly with the higher-order equations, mostly for second-order equations. An important modification of these is the so-called "summed" form.

There have also been developed some "hybrid" multi-step methods which involve two formulas which give two predicted values, one at x_i and one at another point between x_{i-1} and x_i. The corrector uses both of these values. The remarkable thing about these methods is that they can be of relatively high degree and still remain stable, in contrast to the multi-step methods. But these hybrid methods are quite new and they have not been sufficiently tested.

There are also a number of other special results, such as implicit Runge-Kutta methods, and multi-step methods based on fitting the solution with exponentials, or trigonometric functions, in place of polynomials. But none of these has received much attention.

One other class of methods deserves special mention. These are the higher-derivative methods. Typical formulas in this class are the following:

$$y_i^* = 3y_{i-1}^* - 3y_{i-2}^* + y_{i-3}^* + h^2(y_{i-1}^{*\prime\prime} - y_{i-2}^{*\prime\prime})$$

which can be used as a predictor, and which has

$$T_i = \frac{60}{720} h^5 y^{(5)}(x_i) + 0(h^6),$$

and

$$y_i^* = y_{i-1}^* + \frac{h}{2}(y_i^{*\prime} + y_{i-1}^{*\prime}) - \frac{h^2}{12}(y_i^{*\prime\prime} - y_{i-1}^{*\prime\prime}),$$

which can be used as a corrector, and which has

$$T_i = \frac{1}{720} h^5 y^{(5)}(x_i) + 0(h^6).$$

The formulas have relatively small truncation errors, and yet some of them (the corrector here is an example) have none of the stability problems of the multi-step methods. The disadvantage is of course the need to evaluate the higher derivatives. This might be very time consuming or, if the function $f(x, y)$ contains tabulated information for example, it could be virtually impossible. However, methods based on formulas such as these may be the best of all when the functions can be readily differentiated.

EXERCISE

1. Derive the corrector formula given above, and its truncation error.

Chapter 11
ERROR CONTROL

In connection with Euler's method we have already mentioned one way of estimating the accuracy of a calculation. The same idea can be used with Runge-Kutta and predictor-corrector procedures as well. It involves solving the problem several times with different step-sizes, and comparing the results.

For example, suppose it is known that the propagated error is proportional to the pth power of the step-size. The result of a calculation using the step-size h is then the true result plus Kh^p where K is assumed constant. The result when using the step-size $h/2$ is the true result plus $Kh^p/2^p$. The difference between the two results is then $(2^p-1)Kh^4/2^p$, or (2^p-1) times the error in the second calculation. The argument is not rigorous of course. But if further runs are consistent, the conclusions are at least extremely plausible.

One might consider extrapolating from the results of several calculations to obtain an improved result. Just how this is done will depend on whether one assumes he knows the power of h that is involved, as we have been doing, or whether one assumes only that the error is proportional to h^p, with the value of p not known in advance.

If it is obviously appropriate to change the step-size at various points during the calculation, then this modification can be incorporated in whatever scheme one is using.

There is another approach which is often used. This is to estimate the error at each step in the calculation, and then to make any necessary adjustments of the step-size before proceeding to the next step. The usual way of accomplishing this with Runge-Kutta methods is similar to the above. One compares the result of taking one step of size h with the result of taking two steps of size $h/2$.

If the error is Ch^{p+1}, where C is assumed constant and p is known, then the difference between the two results is taken to be a measure of the size of the error, in this case (2^p-1) times the error in one step, using the smaller step-size. (Note that this time the error in using the smaller step-size is committed twice during the calculation being considered.)

With this idea a procedure can be arranged which will automatically adjust the step-size so that the error is kept below some prescribed tolerance. Many programs based on this idea control the error per step, without making allowance for the number of steps. It should be the error per unit increase in x which is controlled. This means keeping $Ch^{p+1}/h = Ch^p$ below a prescribed tolerance.

With predictor-corrector procedures one has a relatively easy way of estimating the error in a single step. The difference between the predicted and corrected values is a measure of this error. For example, with the first Adams method given earlier, the truncation error in the predicted value can be taken, for the sake of this argument, to be approximately $9Ch^4$. In the corrector it is then approximately $-Ch^4$. The difference between the predicted and the corrected values is therefore approximately $10Ch^4$ which is -10 times the error in the corrected value. The argument is of course not rigorous.

If this way of estimating the local error is accepted, the decisions about when to raise or lower

the step-size are still quite complicated, because of the fact that it is so troublesome to change step-size with predictor-corrector methods. In fact the value of predictor-corrector methods depends heavily on whether or not the step-size will have to be changed very often, and this clearly depends on the problem being considered.

A weakness is this second approach to the problem of error control, whether with Runge-Kutta or predictor-corrector methods, is that it does not take into account the way in which errors are propagated through a calculation. Even though the local errors are kept small it may be that their effect is growing exponentially in such a way that the accumulated error will not be tolerable. There is no simple way of dealing with this situation except by knowing something about the way the errors are being propagated. This knowledge might be obtained theoretically, or it might come from experimental runs with sample problems of the right type.

Another weakness of the second approach to error control is that it will not detect instability in the numerical procedure. The difficulty is analogous to what has just been described. The local errors may be kept very small, but instability could still cause enormous growth of the error.

It is possible to monitor stability in a fairly simple way with single differential equations, but no efficient procedure seems to be possible for systems of equations. The trouble is that one really needs to know something about the eigenvalues of the matrix $I + hg$, or $I + hg + h^2g^2/2$, etc. In the limiting case, when $h = 0$, there is no problem concerning the stability of the methods which are generally used. But in practice, when $h \neq 0$, the situation is quite complicated, as we have already indicated. There does not seem to be any simple course of action which can be recommended, other than simply knowing something about the way the errors are propagated, as we have already suggested.

Chapter 11
PROGRAMMING CONSIDERATIONS

A good program library should have subroutines for the numerical integration of ordinary differential equations. There should be at least one subroutine based on a Runge-Kutta method, such as the fourth-order method quoted earlier, and another based on an Adams procedure of moderately high order. It is clear that the Runge-Kutta program would be quite a bit easier to write, mainly because of the need in an Adams subroutine for separate procedures at the beginning of the calculation, or after the step-size has been changed. A Runge-Kutta procedure could be used for this purpose, but there are other possibilities as well.

In the parameter list for any such subroutine the user most provide x_0, y_0, xend, and fcn, where xend is the value of x to which the integration is to be taken, and fcn is the name of the procedure for evaluating the function f(x, y). He would also have to provide something about the step-size. One good idea would be for him to provide a maximum step-size, which could be used as a starting value and also as a bound for later choices of h. This would serve to prevent the program from trying to use any ridiculously large values, as it might in regions where the solution is very smooth.

The user would also have to provide an error bound which could, for example, be a vector giving bounds for the local error in each component. A decision must be made about whether or not this bound is to be an absolute error bound, or a relative error bound. If the latter, it is relative to the solution, or to f(x, y), and what if either of these becomes zero?

The user would also have to provide some indication of whether or not the entry to the subroutine is the first entry. This will enable the subroutine to initialize when necessary, and to choose an appropriate value for h. Some attempt should be made to minimize the accululated effect of rounding errors. One way is to accumulate the approximations to the solution in double precision. This device is known as partial double precision. If this is done, then the initialization will involve setting the least significant halves to zero.

Finally, with most programming languages, the user will have to provide the order of the system of differential equations, and also the working space needed by the subroutine.

With local error control a common technique is to halve the step-size whenever it is decided that the step-size should be reduced, and to double it when it should be increased. There could be some slight saving in computing time when this is done, but it is doubtful if it is worthwhile in the long run. On the other hand, very little has been done about trying to find any better schemes.

In some applications it is important to make provision for an error exit in case the program has difficulty with a singularity.

Project: Write a subroutine based on a Runge-Kutta procedure with automatic error control. Provide a clear description for the potential user of exactly how the subroutine is to be used, what it is supposed to do, and what its limitations are. How do you know your subroutine fits the description?

BIBLIOGRAPHY

For an advanced discussion, particularly of stability questions in the limiting case $h \to 0$, and for a complete bibliography, one should consult the following books:

Henrici, P. Discrete Variable Methods in Ordinary Differential Equations (Wiley, 1962).
Henrici, P. Error Propagation for Difference Methods (Wiley, 1963)

Earlier books, which are also very comprehensive, are the following:

Milne, W. E. Numerical Solution of Differential Equations (Wiley, 1953).
Collatz, L. The Numerical Treatment of Differential Equations, 3rd Ed. (Springer, 1960).

Chapter 6 in the following book is also recommended:

Hildebrand, F. B. Introduction to Numerical Analysis (McGraw-Hill, 1956)

For further information about existence theorems the following is recommended:

Coddington, E. A., Levinson, N. Theory of Ordinary Differential Equations (McGraw-Hill, 1955).

Most of the required information on Runge-Kutta methods is contained in the books given above on numerical methods. For further information about choosing the remaining free parameters in these methods see:

Hull, T. E., Johnston, R. L. Optimum Runge-Kutta Methods, Math. Comp. 18 (1864) 306-310.

For further information on higher order Runge-Kutta methods see:

Butcher, J. C. On Runge-Kutta Processes of High Order, J. Austral. Math Soc. 4 (1964) 179-194.

References to the earlier work of Bashforth and Adams, Moulton, Milne and others are given in the texts referred to above. References to the work of Dahlquist and further development of these ideas will be found in Henrici's books. For further information about trying to find general multi-step formulas which are better than Adams formulas see:

Hull, T. E., and Newbery, A. C. R. Corrector Formulas for Multi-step Integration Methods, J. Soc. Indust. Appl. Math. 10 (1962) 351-369.

For details about the need to have two function evaluations per step with predictor-corrector methods see:

Hull, T. E. and Creemer, A. L. Efficiency of Predictor-Corrector Procedures, J. Assoc. Comput. Mach. 10 (1963) 291-301.

Further details about methods for higher-order differential equations, and also about higher-derivative methods will be found in the texts given above. For the latter see also:

Lambert, J. D., and Mitchell, A. R. On the Solution of $y' = f(x, y)$ by a class of high accuracy formulae of low order, Z. Angew. Math. Phys. 13 (1962) 223-231.

For more information about hybrid methods see:

Gragg, W. B. and Stetter, H. J. Generalized Multistep Predictor-Corrector Methods, J. Assoc. Comput. Mach. 11 (1964) 188 - 209.
Butcher, J. C. A Modified Multistep Method for the Numerical Integration of Ordinary Differential Equations, J. Assoc. Comput. Mach. 12 (1965) 124-135.

For implicit Runge-Kutta methods see:

Butcher, J. C. Implicit Runge-Kutta Processes, Math. Comp. 18 (1964) 50-64.

For methods which are exact for trigonometric functions see:

Gautschi, W. Numerical Integration of Ordinary Differential Equations Based on Trigonometric Polynomials, Numer. Math 3 (1961) 381-397;

and for those which are exact for exponentials see:

Brock, P. and Murray, F. J. The Use of Exponential Sums in Step by Step Integration, Math Tables Other Aids Comp. 6 (1952) 63-78.

Details about programming have been published in the following:

Anderson, W. H. The Solution of Simultaneous Ordinary Differential Equations Using a General Purpose Digital Computer, Comm. ACM 3 (1960) 355-360.
Hain, K., Hertweck, F. Numerical Integration of Ordinary Differential Equations by Difference Methods with Automatic Determination of Steplength, Symposium on the Numerical Treatment of Ordinary Differential Equations, Integral and Integro-Differential Equations (Birkhauser Verlag, 1960) 122-128.

An interesting development using interval arithmetic to provide guaranteed error bounds is described in:

Moore, Ramon E., The Automatic Analysis and Control of Error in Digital Computing Based on the Use of Interval Numbers, Error in Digital Computation, vol. 1, (Wiley. 1965) ed. by Louis B. Rall, 61-130.

An extension of this work is to appear in vol. 2.