which possesses obvious advantages (in general) over the corresponding formula

$$y_{n+1} = y_n + \frac{h}{2}(y'_{n+1} + y'_n) - \frac{h^2}{12}(y''_{n+1} - y''_n) + \frac{h^4}{720}(y^{\text{iv}}_{n+1} - y^{\text{iv}}_n) - \frac{h^7}{30240}\, y^{\text{vii}}(\xi) \qquad (6.12.9)$$

obtained from (6.12.1). An appropriate predictor formula can be obtained in the form

$$y_{n+1} = y_{n-1} + 2h(4y'_n - 3y'_{n-1}) - \frac{2h^2}{5}(8y''_n + 7y''_{n-1}) + \frac{2h^3}{15}(7y'''_n - 3y'''_{n-1}) + \frac{13h^7}{6300}\, y^{\text{vii}}(\xi) \qquad (6.12.10)$$

An infinite variety of other formulas can be derived by employing data relevant to more than two points for correction, and to more than three points for prediction. Thus, for example, the three-point formula of highest precision, using first and second derivatives, is readily found to be

$$y_{n+1} - 2y_n + y_{n-1} = \frac{3h}{8}(y'_{n+1} - y'_{n-1}) - \frac{h^2}{24}(y''_{n+1} - 8y''_n + y''_{n-1}) + \frac{h^8}{60480}\, y^{\text{viii}}(\xi) \qquad (6.12.11)$$

6.13 A Simple Runge-Kutta Method

The methods associated with the names of Runge [1895], Kutta [1901], Heun [1900], and others as applied to the numerical solution of the problem

$$y' = F(x, y) \qquad y(x_0) = y_0 \qquad (6.13.1)$$

effectively replace the result of truncating a Taylor-series expansion of the form

$$y_{n+1} = y_n + hy'_n + \frac{h^2}{2} y''_n + \frac{h^3}{6} y'''_n + \cdots \qquad (6.13.2)$$

by an approximation in which y_{n+1} is calculated from a formula of the type

$$y_{n+1} = y_n + h[\alpha_0 F(x_n, y_n) + \alpha_1 F(x_n + \mu_1 h, y_n + b_1 h) + \alpha_2 F(x_n + \mu_2 h, y_n + b_2 h) + \cdots + \alpha_p F(x_n + \mu_p h, y_n + b_p h)] \qquad (6.13.3)$$

Here the α's, μ's, and b's are so determined that, if the right-hand member of (6.13.3) were expanded in powers of the spacing h, the coefficients of a certain number of the leading terms would agree with the corresponding coefficients in (6.13.2).

They possess the advantages that they are self-starting but do not require the evaluation of derivatives of $F(x, y)$ and hence can be used (even at the beginning of the solution) when $F(x, y)$ is not given by an analytical expression, and also that a change in spacing is easily effected at any intermediate stage of the calculation. On the other hand, each step involves several evaluations of $F(x, y)$, which may be excessively laborious and/or time-consuming, and also the estimation of errors is less simply accomplished than in the previously described methods.

It is convenient, in order both to simplify the derivation and also to systematize the formulation, to express each of the b's in (6.13.3) as a linear combination of the preceding values of F. Thus, in place of using the notation of (6.13.3), it is desirable to write the approximation in the form

$$y_{n+1} = y_n + \alpha_0 k_0 + \alpha_1 k_1 + \cdots + \alpha_p k_p \tag{6.13.4}$$

with

$$\begin{aligned}
k_0 &= hF(x_n, y_n) \\
k_1 &= hF(x_n + \mu_1 h, y_n + \lambda_{10} k_0) \\
k_2 &= hF(x_n + \mu_2 h, y_n + \lambda_{20} k_0 + \lambda_{21} k_1) \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
k_p &= hF(x_n + \mu_p h, y_n + \lambda_{p0} k_0 + \lambda_{p1} k_1 + \cdots + \lambda_{p,p-1} k_{p-1})
\end{aligned} \tag{6.13.5}$$

where the coefficients α_i, μ_i, and λ_{ij} are to be determined.

Since the actual derivation of such formulas involves considerable algebraic manipulation, we consider in detail only the very simple case $p = 1$, which may serve to illustrate the procedure in the more general case. Thus, writing μ for μ_1 and λ for λ_{10}, we proceed to determine α_0, α_1, μ, and λ such that

$$y_{n+1} = y_n + \alpha_0 k_0 + \alpha_1 k_1 \tag{6.13.6}$$

with

$$k_0 = hF(x_n, y_n) \qquad k_1 = hF(x_n + \mu h, y_n + \lambda k_0) \tag{6.13.7}$$

possesses an expansion in powers of h whose leading terms agree, insofar as possible, with the leading terms of (6.13.2).

We first obtain the expansion

$$\begin{aligned} k_1 &= h[F + (\mu h F_x + \lambda k_0 F_y) \\ &\qquad + \tfrac{1}{2}(\mu^2 h^2 F_{xx} + 2\mu\lambda h k_0 F_{xy} + \lambda^2 k_0^2 F_{yy}) + O(h^3)] \\ &= hF + h^2(\mu F_x + \lambda F F_y) \\ &\qquad + \frac{h^3}{2}(\mu^2 F_{xx} + 2\mu\lambda F F_{xy} + \lambda^2 F^2 F_{yy}) + O(h^4) \end{aligned} \tag{6.13.8}$$

where $F \equiv F(x_n, y_n)$, $F_x \equiv F_x(x_n, y_n)$, and so forth. Hence (6.13.6) becomes

$$\begin{aligned} y_{n+1} &= y_n + h(\alpha_0 + \alpha_1)F + h^2\alpha_1(\mu F_x + \lambda F F_y) \\ &\qquad + \frac{h^3}{2}\alpha_1(\mu^2 F_{xx} + 2\mu\lambda F F_{xy} + \lambda^2 F^2 F_{yy}) + O(h^4) \end{aligned} \tag{6.13.9}$$

On the other hand, with the same abbreviated notation, we obtain from (6.13.1) the relations

$$\begin{aligned} y' &= F \\ y'' &= F_x + F F_y \\ y''' &= F_{xx} + 2F F_{xy} + F^2 F_{yy} + F_y(F_x + F F_y) \end{aligned} \tag{6.13.10}$$

so that (6.13.2) becomes

$$\begin{aligned} y_{n+1} &= y_n + hF + \frac{h^2}{2}(F_x + F F_y) \\ &\qquad + \frac{h^3}{6}[F_{xx} + 2F F_{xy} + F^2 F_{yy} + F_y(F_x + F F_y)] + O(h^4) \end{aligned} \tag{6.13.11}$$

Thus, if we identify the coefficients of hF, $h^2 F_x$, and $h^2 F F_y$ in (6.13.9) and (6.13.11), we obtain the three conditions

$$\alpha_0 + \alpha_1 = 1 \qquad \mu\alpha_1 = \tfrac{1}{2} \qquad \lambda\alpha_1 = \tfrac{1}{2} \tag{6.13.12}$$

involving the four adjustable parameters, which are satisfied if and only if

$$\alpha_0 = 1 - c \qquad \alpha_1 = c \qquad \mu = \frac{1}{2c} \qquad \lambda = \frac{1}{2c}$$

where c is an arbitrary nonzero constant. The expansion (6.13.9) then reduces to

$$\begin{aligned} y_{n+1} &= y_n + hF + \frac{h^2}{2}(F_x + F F_y) \\ &\qquad + \frac{h^3}{8c}(F_{xx} + 2F F_{xy} + F^2 F_{yy}) + O(h^4) \end{aligned} \tag{6.13.13}$$

and reference to (6.13.10) shows that (6.13.13) or, equivalently, (6.13.6) would then be brought into agreement with (6.13.11) or (6.13.2) if a truncation-error term of the form

$$T_n = -\left(\frac{h^3}{8c} - \frac{h^3}{6}\right)[(F_{xx} + 2FF_{xy} + F^2F_{yy}) + F_y(F_x + FF_y)] + \frac{h^3}{8c}F_y(F_x + FF_y) + O(h^4)$$

or

$$T_n = -\frac{h^3}{24c}[(3 - 4c)y_n''' - 3F_y(x_n, y_n)y_n''] + O(h^4) \quad (6.13.14)$$

were added to its right-hand member.

The remaining free parameter c clearly cannot be determined so that T_n is of order h^4, except in trivial special cases. One convenient choice is $c = \frac{1}{2}$, in which case the second abscissa involved in (6.13.6) and (6.13.7) is x_{n+1}, and the formula becomes

$$y_{n+1} = y_n + \tfrac{1}{2}(k_0 + k_1) + T_n, \quad (6.13.15)$$

with

$$k_0 = hF(x_n, y_n) \qquad k_1 = hF(x_n + h, y_n + k_0) \quad (6.13.16)$$

where

$$T_n = -\frac{h^3}{12}[y_n''' - 3F_y(x_n, y_n)y_n''] + O(h^4) \quad (6.13.17)$$

Stepwise calculation based on this formula is sometimes known as *Heun's method.*

If, for all values of x and y involved in the calculation, it is known that

$$|F_y(x, y)| \leqq K \quad (6.13.18)$$

then, as in earlier developments, it is readily shown that the propagated error ε_n in the nth step is dominated by the solution of the difference equation

$$e_{n+1} = e_n + \frac{hK}{2}e_n + \frac{hK}{2}(e_n + hKe_n) + E$$

or

$$e_{n+1} = \left(1 + hK + \frac{h^2K^2}{2}\right)e_n + E \quad (6.13.19)$$

where

$$e_0 = 0 \qquad E = |T_n + R_n|_{\max} \qquad hK + \frac{h^2K^2}{2} < 1 \quad (6.13.20)$$

Further, it can be shown that (6.13.17) can be replaced by

$$T_n = -\frac{h^3}{12}[y'''(\xi_1) - 3F_y(x_{n+1}, \eta)y''(\xi_2)] \qquad (6.13.21)$$

where ξ_1 and ξ_2 are intermediate between x_n and x_{n+1}, and η between y_{n+1} and $y_n + hy'_n$. Thus, if the roundoff error R_n is ignored, and if

$$|y''(x)| \leqq M_2 \qquad |y'''(x)| \leqq M_3 \qquad (6.13.22)$$

it follows after a simple calculation that

$$|\varepsilon_n| \leqq \frac{h^2(M_3 + 3KM_2)}{12K(1 + \frac{1}{2}hK)}\left[\left(1 + hK + \frac{h^2K^2}{2}\right)^n - 1\right] \qquad (6.13.23)$$

The formula (6.13.15), using (6.13.16), is of limited accuracy. Indeed, it can be considered to be a modification of the result of retaining only the first difference in (6.3.2)

$$y_{n+1} = y_n + \frac{h}{2}(y'_n + y'_{n+1}) - \frac{h^3}{12}y'''(\xi) \qquad (6.13.24)$$

in which the unknown derivative $y'_{n+1} \equiv F(x_{n+1}, y_{n+1})$ is replaced by the approximation $y'_{n+1} \approx F(x_{n+1}, y_n + hy'_n)$. This consideration is useful in deriving (6.13.21). The details of the analysis were presented here principally to illustrate the similar but more complicated analysis relevant to formulas of higher-order accuracy, certain of which are listed in the following section.

It is of some importance to notice that the error (6.13.21), associated with (6.13.15) and (6.13.16), depends upon the form of the function $F(x, y)$ as well as upon the solution y itself. This situation is characteristic of formulas of the Runge-Kutta type. For example, whereas the equations $y' = 2(x + 1)$ and $y' = 2y/(x + 1)$ both define the function $y = (x + 1)^2$ when the condition $y(0) = 1$ is imposed, the formula defined by (6.13.15) and (6.13.16) would yield this solution *exactly* when applied to the first equation, if no roundoffs were committed, but would not do so when applied to the second form. On the other hand, the formula (6.13.24) would yield exact results when applied to *either* form, or to any *other* first-order equation whose required solution is a polynomial of degree 2 or less (see also Milne [1950, 1970]).

At the same time, the mere fact that (6.13.15), with (6.13.16), does not have this last property does not imply that its interpretation as a weakened modification of (6.13.24) is proper in the more general case when the true solution is not such a polynomial. For example, it is easily seen that the use of (6.13.15) and (6.13.16) would yield exact results when applied to the problem $y' = -y/(x + 1)$, $y(0) = 1$, for which the solution is $y = 1/(x + 1)$, whereas the use of (6.13.24) would lead only to an approximation.

6.14 Runge-Kutta Methods of Higher Order

When k_0, k_1, and k_2 are employed in (6.13.4), corresponding to $p = 2$, it is found that the requirement that the expansion of the right-hand member be correct through h^3 terms imposes only six conditions on the eight arbitrary parameters involved, so that a doubly infinite set of such formulas with third-order accuracy can be obtained.

One such formula, due to Kutta, is of the form

$$y_{n+1} = y_n + \tfrac{1}{6}(k_0 + 4k_1 + k_2) + O(h^4) \tag{6.14.1}$$

with

$$\begin{aligned} k_0 &= hF(x_n, y_n) \\ k_1 &= hF(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_0) \\ k_2 &= hF(x_n + h, y_n + 2k_1 - k_0) \end{aligned} \tag{6.14.2}$$

A second formula, due to Heun, is of the form

$$y_{n+1} = y_n + \tfrac{1}{4}(k_0 + 3k_2) + O(h^4) \tag{6.14.3}$$

with

$$\begin{aligned} k_0 &= hF(x_n, y_n) \\ k_1 &= hF(x_n + \tfrac{1}{3}h, y_n + \tfrac{1}{3}k_0) \\ k_2 &= hF(x_n + \tfrac{2}{3}h, y_n + \tfrac{2}{3}k_1) \end{aligned} \tag{6.14.4}$$

These two formulas are generally of about equal accuracy, with each possessing certain obvious computational advantages. Kutta's form is seen to be analogous to the formula of Simpson's rule and would reduce to that formula if F were independent of y.

It is also possible to derive a two-parameter family of formulas of fourth-order accuracy, by retaining an additional k in (6.13.4). The simplest such formula, due to Kutta, is of the form

$$y_{n+1} = y_n + \tfrac{1}{6}(k_0 + 2k_1 + 2k_2 + k_3) + O(h^5) \tag{6.14.5}$$

with

$$\begin{aligned} k_0 &= hF(x_n, y_n) \\ k_1 &= hF(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_0) \\ k_2 &= hF(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1) \\ k_3 &= hF(x_n + h, y_n + k_2) \end{aligned} \tag{6.14.6}$$

and would also reduce to Simpson's rule if F were independent of y.

Such formulas can also be generalized to the treatment of *simultaneous* equations of the form

$$\frac{dy}{dx} = F(x, y, u) \qquad \frac{du}{dx} = G(x, y, u) \tag{6.14.7}$$

where y and u are prescribed when $x = x_0$. In particular, the preceding formula generalizes as follows:

$$\begin{aligned} y_{n+1} &= y_n + \tfrac{1}{6}(k_0 + 2k_1 + 2k_2 + k_3) + O(h^5) \\ u_{n+1} &= u_n + \tfrac{1}{6}(m_0 + 2m_1 + 2m_2 + m_3) + O(h^5) \end{aligned} \tag{6.14.8}$$

with

$$\begin{aligned} k_0 &= hF(x_n, y_n, u_n) \\ k_1 &= hF(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_0, u_n + \tfrac{1}{2}m_0) \\ k_2 &= hF(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1, u_n + \tfrac{1}{2}m_1) \\ k_3 &= hF(x_n + h, y_n + k_2, u_n + m_2) \end{aligned} \tag{6.14.9}$$

and

$$\begin{aligned} m_0 &= hG(x_n, y_n, u_n) \\ m_1 &= hG(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_0, u_n + \tfrac{1}{2}m_0) \\ m_2 &= hG(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1, u_n + \tfrac{1}{2}m_1) \\ m_3 &= hG(x_n + h, y_n + k_2, u_n + m_2) \end{aligned} \tag{6.14.10}$$

A consideration of this form indicates the way in which other formulas are so generalized.

In particular, when $F = u$, so that (6.14.7) is equivalent to

$$\frac{d^2y}{dx^2} = G(x, y, y') \tag{6.14.11}$$

with $u \equiv y'$, (6.14.9) gives

$$k_0 = hy_n' \qquad k_1 = hy_n' + \frac{h}{2}m_0 \qquad k_2 = hy_n' + \frac{h}{2}m_1 \qquad k_3 = hy_n' + hm_2$$

and hence (6.14.8) and (6.14.10) reduce to

$$\begin{aligned} y_{n+1} &= y_n + hy_n' + \frac{h}{6}(m_0 + m_1 + m_2) + O(h^5) \\ y_{n+1}' &= y_n' + \tfrac{1}{6}(m_0 + 2m_1 + 2m_2 + m_3) + O(h^5) \end{aligned} \tag{6.14.12}$$

with

$$
\begin{aligned}
m_0 &= hG(x_n, y_n, y_n') \\
m_1 &= hG(x_n + \tfrac{1}{2}h,\ y_n + \tfrac{1}{2}hy_n',\ y_n' + \tfrac{1}{2}m_0) \\
m_2 &= hG(x_n + \tfrac{1}{2}h,\ y_n + \tfrac{1}{2}hy_n' + \tfrac{1}{4}hm_0,\ y_n' + \tfrac{1}{2}m_1) \\
m_3 &= hG(x_n + h,\ y_n + hy_n' + \tfrac{1}{2}hm_1,\ y_n' + m_2)
\end{aligned}
\tag{6.14.13}
$$

The use of this formula is clearly simplified in those cases when G is independent of y'.

Many variations and generalizations of these formulas are present in the literature, some of which afford certain computational advantages in certain situations. One such modification, due to Gill [1951], is of particular usefulness when the computation is to be effected by a computer in which it is desirable to minimize the *storage* of data.

No simple expressions are known for the precise truncation errors in the preceding formulas. An *estimate* of the error can be obtained, in practice, in the following way. Let the truncation error associated with a formula of rth-order accuracy, in progressing from the ordinate at x_n to that at $x_{n+1} = x_n + h$, in a single step, be denoted by $C_n h^{r+1}$, and suppose that C_n varies slowly with n and is nearly independent of h when h is small. Then if the true ordinate at x_{n+1} is denoted by Y_{n+1}, the value obtained by two steps starting at x_{n-1} by $y_{n+1}^{(h)}$, and the value obtained by a single step with *doubled* spacing $2h$ by $y_{n+1}^{(2h)}$, there follows approximately

$$
\begin{aligned}
Y_{n+1} - y_{n+1}^{(h)} &\approx 2C_n h^{r+1} \\
Y_{n+1} - y_{n+1}^{(2h)} &\approx 2^{r+1} C_n h^{r+1}
\end{aligned}
\tag{6.14.14}
$$

when h is small. The result of eliminating C_n from these approximate relations is then the *extrapolation* formula†

$$
Y_{n+1} \approx y_{n+1}^{(h)} + \frac{y_{n+1}^{(h)} - y_{n+1}^{(2h)}}{2^r - 1} \tag{6.14.15}
$$

Thus if, at certain stages of the advancing calculation, the newly calculated ordinate y_{n+1} is recomputed from y_{n-1} with a doubled spacing, the truncation error in the originally calculated value is approximated by the result of dividing the difference between the two values by the factor $2^r - 1$, that is, by 3 in (6.13.15), by 7 in (6.14.1) or (6.14.3), and by 15 in the formulas of fourth-order accuracy.

It is apparent that an arbitrary change in spacing can be introduced at any stage of the forward progress, when a method of the Runge-Kutta type is used, without introducing any appreciable complication.

† This is another example of so-called *Richardson extrapolation* (see Sec. 3.8).