c. $y' = -(y+1)(y+3)$, for $0 \le t \le 3$, with $y(0) = -2$; actual solution $y(t) = -3 + 2(1+e^{-2t})^{-1}$.

d. $y' = (t+2t^3)y^3 - ty$, for $0 \le t \le 2$, with $y(0) = \frac{1}{3}$; actual solution $y(t) = (3+2t^2+6e^{t^2})^{-1/2}$.

5. Let $P(t)$ be the number of individuals in a population at time t, measured in years. If the average birth rate b is constant and the average death rate d is proportional to the size of the population (due to overcrowding), then the growth rate of the population is given by the *logistic equation*

$$\frac{dP(t)}{dt} = bP(t) - k[P(t)]^2$$

where $d = kP(t)$. Suppose $P(0) = 50,976$, $b = 2.9 \times 10^{-2}$, and $k = 1.4 \times 10^{-7}$. Find the population after 5 years.

5.6 Adaptive Techniques

The appropriate use of varying step size was seen in Section 4.6 to produce integral approximating methods that are efficient in the amount of computation required. This might not be sufficient to favor these methods due to the increased complication of applying them, but they have another important feature. The step-size selection procedure produces an estimate of the local error that does not require the approximation of the higher derivatives of the function. These methods are called *adaptive* because they adapt the number and position of the nodes used in the approximation to keep the local error within a specified bound.

You might like to review the Adaptive Quadrature material beginning on page 138 before considering this material.

There is a close connection between the problem of approximating the value of a definite integral and that of approximating the solution to an initial-value problem. It is not surprising, then, that there are adaptive methods for approximating the solutions to initial-value problems, and that these methods are not only efficient but incorporate the control of error.

Any one-step method for approximating the solution, $y(t)$, of the initial-value problem

$$y' = f(t, y), \quad \text{for } a \le t \le b, \quad \text{with } y(a) = \alpha$$

can be expressed in the form

$$w_{i+1} = w_i + h_i \phi(t_i, w_i, h_i), \quad \text{for } i = 0, 1, \ldots, N-1,$$

for some function ϕ. An ideal difference-equation method would have the property that given a tolerance $\varepsilon > 0$, the minimal number of mesh points would be used to ensure that the global error, $|y(t_i) - w_i|$, would not exceed ε for any $i = 0, 1, \ldots, N$. Having a minimal number of mesh points and also controlling the global error of a difference method is, not surprisingly, inconsistent with the points being equally spaced in the interval. In this section we examine techniques used to control the error of a difference-equation method in an efficient manner by the appropriate choice of mesh points.

Although we cannot generally determine the global error of a method, there is often a close connection between the local error and the global error. If a method has local error $O(h^{n+1})$, then the global error of the method is $O(h^n)$. By using methods of differing order we can predict the local error and, using this prediction, choose a step size that will keep the global error in check.

To illustrate the technique, suppose that we have two approximation techniques. The first is an nth-order method obtained from an nth-order Taylor method of the form

$$y(t_{i+1}) = y(t_i) + h\phi(t_i, y(t_i), h) + O(h^{n+1}).$$

This method produces approximations

$$w_0 = \alpha$$
$$w_{i+1} = w_i + h\phi(t_i, w_i, h) \quad \text{for} \quad i > 0,$$

which satisfy, for some $\hat{K}$ and all relevant h and i,

$$|y(t_i) - w_i| < \hat{K}h^n.$$

In general, the method is generated by applying a Runge-Kutta modification to the Taylor method, but the specific derivation is unimportant.

The second method is similar but of higher order. For example, let us suppose it comes from an $(n+1)$st-order Taylor method of the form

$$y(t_{i+1}) = y(t_i) + h\tilde{\phi}(t_i, y(t_i), h) + O(h^{n+2}),$$

producing approximations

$$\tilde{w}_0 = \alpha$$
$$\tilde{w}_{i+1} = \tilde{w}_i + h\tilde{\phi}(t_i, \tilde{w}_i, h) \quad \text{for} \quad i > 0,$$

which satisfy, for some $\tilde{K}$ and all relevant h and i,

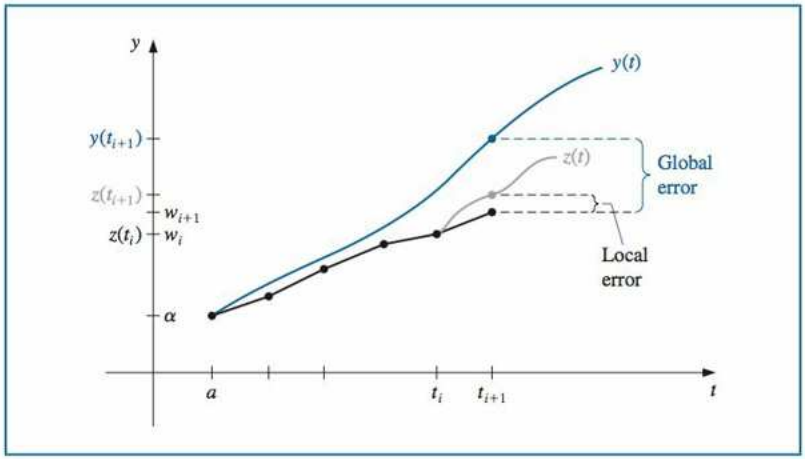
$$|y(t_i) - \tilde{w}_i| < \tilde{K}h^{n+1}.$$

We assume now that at the point t_i we have

$$w_i = \tilde{w}_i = z(t_i),$$

where $z(t)$ is the solution to the differential equation that does not satisfy the original initial condition but instead satisfies the condition $z(t_i) = w_i$. The typical difference between $y(t)$ and $z(t)$ is shown in Figure 5.3.

Figure 5.3



Applying the two methods to the differential equation with the fixed step size h produces two approximations, w_{i+1} and $\tilde{w}_{i+1}$, whose differences from $y(t_i+h)$ represent global errors but whose differences from $z(t_i + h)$ represent local errors.

Now consider

$$z(t_i + h) - w_{i+1} = (\tilde{w}_{i+1} - w_{i+1}) + (z(t_i + h) - \tilde{w}_{i+1}).$$

The term on the left side of this equation is $O(h^{n+1})$, the local error of the nth order method, but the second term on the right side is $O(h^{n+2})$, the local error of the $(n+1)$st order method. This implies that the dominant portion on the right side comes from the first term; that is,

$$z(t_i + h) - w_{i+1} \approx \tilde{w}_{i+1} - w_{i+1} = O(h^{n+1}).$$

So, a constant K exists with

$$Kh^{n+1} = |z(t_i + h) - w_{i+1}| \approx |\tilde{w}_{i+1} - w_{i+1}|,$$

and K can be approximated by

$$K \approx \frac{|\tilde{w}_{i+1} - w_{i+1}|}{h^{n+1}}. \tag{5.3}$$

Let us now return to the global error associated with the problem we really want to solve,

$$y' = f(t, y), \quad \text{for } a \le t \le b, \quad \text{with } y(a) = \alpha,$$

and consider the adjustment to the step size needed if this global error is expected to be bounded by the tolerance ε. Using a multiple q of the original step size implies that we need to ensure that

$$|y(t_i + qh) - w_{i+1}(\text{using the new step size } qh)| < K(qh)^n < \varepsilon.$$

This implies, from Eq. (5.3), that for the approximations w_{i+1} and $\tilde{w}_{i+1}$ using the original step size h we have

$$Kq^n h^n \approx \frac{|\tilde{w}_{i+1} - w_{i+1}|}{h^{n+1}} q^n h^n = \frac{q^n |\tilde{w}_{i+1} - w_{i+1}|}{h} < \varepsilon.$$

Solving this inequality for q tells us how we should choose the new step size, qh, to ensure that the global error is within the bound ε:

$$q < \left[\frac{\varepsilon h}{|\tilde{w}_{i+1} - w_{i+1}|}\right]^{1/n}.$$

Erwin Fehlberg developed this and other error control techniques while working for the NASA facility in Huntsville, Alabama during the 1960s. In 1969 he received NASA's Exceptional Scientific Achievement Medal for his work.

The Runge-Kutta-Fehlberg Method

One popular technique that uses this inequality for error control is the **Runge-Kutta-Fehlberg method** (see [Fe]). It uses a Runge-Kutta method of order 5,

$$\tilde{w}_{i+1} = w_i + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6,$$

to estimate the local error in a Runge-Kutta method of order 4,

$$w_{i+1} = w_i + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5,$$

where the coefficient equations are

$$\begin{aligned}
k_1 &= hf(t_i, w_i),\\
k_2 &= hf\left(t_i + \frac{h}{4}, w_i + \frac{1}{4}k_1\right),\\
k_3 &= hf\left(t_i + \frac{3h}{8}, w_i + \frac{3}{32}k_1 + \frac{9}{32}k_2\right),\\
k_4 &= hf\left(t_i + \frac{12h}{13}, w_i + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3\right),\\
k_5 &= hf\left(t_i + h, w_i + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4\right),\\
k_6 &= hf\left(t_i + \frac{h}{2}, w_i - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5\right).
\end{aligned}$$

An advantage of this method is that only six evaluations of f are required per step, whereas arbitrary Runge-Kutta methods of order 4 and 5 used together would require (see Table 5.9 on page 188) at least four evaluations of f for the fourth-order method and an additional six for the fifth-order method.

In the theory of error control, an initial value of h at the ith step was used to find the first values of w_{i+1} and $\tilde{w}_{i+1}$, which led to the determination of q for that step. Then the calculations were repeated with the step size h replaced by qh. This procedure requires twice the number of functional evaluations per step as without error control. In practice, the value of q to be used is chosen somewhat differently in order to make the increased functional-evaluation cost worthwhile. The value of q determined at the ith step is used for two purposes:

- When $q < 1$, to reject the initial choice of h at the ith step and repeat the calculations using qh, and
- When $q \geq 1$, to accept the computed value at the ith step using the step size h and to increase the step size to qh for the $(i + 1)$st step.

Because of the penalty in terms of functional evaluations that must be paid if many steps are repeated, q tends to be chosen conservatively. In fact, for the Runge-Kutta-Fehlberg method with $n = 4$, the usual choice is

$$q = \left(\frac{\varepsilon h}{2|\tilde{w}_{i+1} - w_{i+1}|}\right)^{1/4} \approx 0.84\left(\frac{\varepsilon h}{|\tilde{w}_{i+1} - w_{i+1}|}\right)^{1/4}.$$

The program RKFVSM55 implements the Runge-Kutta-Fehlberg method.

Program RKFVSM55 incorporates a technique to eliminate large modifications in step size. This is done to avoid spending too much time with very small step sizes in regions with irregularities in the derivatives of y, and to avoid large step sizes, which might result in skipping sensitive regions nearby. In some instances the step-size-increase procedure is omitted completely and the step-size-decrease procedure is modified to be incorporated only when needed to bring the error under control.

Example 1 Use the Runge-Kutta-Fehlberg method with a tolerance $TOL = 10^{-5}$, a maximum step size $hmax = 0.25$, and a minimum step size $hmin = 0.01$ to approximate the solution to the initial-value problem

$$y' = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5,$$

and compare the results with the exact solution $y(t) = (t+1)^2 - 0.5e^t$.

Solution We will work through the first step of the calculations and then apply the program RKFVSM55 to determine the remaining results. The initial condition gives $t_0 = 0$ and $w_0 = 0.5$. To determine w_1 using $h = 0.25$, the maximum allowable stepsize, we compute

$$k_1 = hf(t_0, w_0) = 0.25(0.5 - 0^2 + 1) = 0.375;$$

$$k_2 = hf\left(t_0 + \frac{1}{4}h, w_0 + \frac{1}{4}k_1\right) = 0.25f\left(\frac{1}{4}0.25, 0.5 + \frac{1}{4}0.375\right) = 0.3974609;$$

$$k_3 = hf\left(t_0 + \frac{3}{8}h, w_0 + \frac{3}{32}k_1 + \frac{9}{32}k_2\right)$$

$$= 0.25f\left(0.09375, 0.5 + \frac{3}{32}0.375 + \frac{9}{32}0.3974609\right) = 0.4095383;$$

$$k_4 = hf\left(t_0 + \frac{12}{13}h, w_0 + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3\right)$$

$$= 0.25f\left(0.2307692, 0.5 + \frac{1932}{2197}0.375 - \frac{7200}{2197}0.3974609 + \frac{7296}{2197}0.4095383\right)$$

$$= 0.4584971;$$

$$k_5 = hf\left(t_0 + h, w_0 + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4\right)$$

$$= 0.25f\left(0.25, 0.5 + \frac{439}{216}0.375 - 8(0.3974609) + \frac{3680}{513}0.4095383 - \frac{845}{4104}0.4584971\right)$$

$$= 0.4658452;$$

$$k_6 = hf\left(t_0 + \frac{1}{2}h, w_0 - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5\right)$$

$$= 0.25f\left(0.125, 0.5 - \frac{8}{27}0.375 + 2(0.3974609) - \frac{3544}{2565}0.4095383\right.$$

$$\left. + \frac{1859}{4104}0.4584971 - \frac{11}{40}0.4658452\right)$$

$$= 0.4204789.$$

The two approximations to $y(0.25)$ are then found to be

$$\tilde{w}_1 = w_0 + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6$$

$$= 0.5 + \frac{16}{135}0.375 + \frac{6656}{12825}0.4095383 + \frac{28561}{56430}0.4584971 - \frac{9}{50}0.4658452$$

$$+ \frac{2}{55}0.4204789$$

$$= 0.9204870,$$

and

$$\begin{aligned} w_1 &= w_0 + \frac{25}{216}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4104}k_4 - \frac{1}{5}k_5 \\ &= 0.5 + \frac{25}{216}0.375 + \frac{1408}{2565}0.4095383 + \frac{2197}{4104}0.4584971 - \frac{1}{5}0.4658452 \\ &= 0.9204886. \end{aligned}$$

The approximations with $h = 0.25$ and $\varepsilon = 10^{-5}$ give

$$q = \left(\frac{h\varepsilon}{2|\tilde{w}_1 - w_1|}\right)^{1/4} = 0.9462100.$$

Since $q < 1$, we need to recompute the values of $\tilde{w}_1$ and w_1 with the new step size

$$h = 0.9462100(0.05) = 0.2365525.$$

Recalculating with this step size gives us $q = 0.9986023$ which is again less than 1. Redoing the calculations with the step size

$$h = (0.9986023)(0.2365525) = 0.2362221.$$

gives a value of $q = 0.9999643$ which is still less than one. This requires a change in step size to

$$h = (0.9999643)(0.236222) = 0.2362137.$$

For this step size we get $q = 1.000002 > 1$. So, we accept that, with the step size 0.2362137, the approximation w_1 is sufficiently accurate approximation to $y(0.2362137)$.

The results from program RKFVSM55 are shown in Table 5.13. Notice that the step size on the last line has been adjusted so that an approximation is given for $y(2)$. Keep in mind that the errors listed are global errors, and the Runge-Kutta-Fehlberg method controls local errors. ■

Table 5.13

			RKF-4		RKF-5	
t_i	$y(t_i)$	h_i	w_i	$\lvert y(t_i) - w_i\rvert$	$\hat{w}_i$	$\lvert y(t_i) - \hat{w}_i\rvert$
0.2362137	0.8950018	0.2362137	0.8950028	1.0×10^{-6}	0.8950016	2.0×10^{-7}
0.4724278	1.3661020	0.2362142	1.3661042	2.2×10^{-6}	1.3661031	1.1×10^{-6}
0.7147675	1.9185718	0.2423397	1.9185755	3.7×10^{-6}	1.9185745	2.7×10^{-6}
0.9647675	2.5482226	0.2500000	2.5482282	5.6×10^{-6}	2.5482272	4.6×10^{-6}
1.2147675	3.2204398	0.2500000	3.2204475	7.7×10^{-6}	3.2204469	7.1×10^{-6}
1.4647675	3.9118102	0.2500000	3.9118204	1.02×10^{-5}	3.9118202	1.00×10^{-5}
1.7147675	4.5922707	0.2500000	4.5922836	1.29×10^{-5}	4.5922839	1.32×10^{-5}
1.9647675	5.2232194	0.2500000	5.2232351	1.57×10^{-5}	5.2232362	1.68×10^{-5}
2.0000000	5.3054720	0.0352325	5.3054883	1.63×10^{-5}	5.3054883	1.63×10^{-5}

An adaptive Runge-Kutta method is available in MATLAB using the command `ode45`. To use this command for our sample initial value problem, we first need to define the function $f(t, y)$ using an `M-file` which we name `odefunction2.m`.

```
function YY = odefunction2(t,y)
YY=y-t^2+1
```

We then need to make the function known to MATLAB using

```
f = @odefunction2
```

We define a range of t values for which we want to approximate the $y(t)$ values beginning at the initial value $a = 0$ and ending at the value $b = 2$. The range of t values are placed in the structure called `tspan`

```
tspan = [0.0 0.2 0.4 0.6 0.8 1.0 1.2 1.4 1.6 1.8 2.0]
```

and we define the initial value $y(0) = 0.5$ with

```
y0=0.5
```

We invoke the function `ode45` using the following command

```
[T,YY] = ode45(f,tspan,y0)
```

which produces the values of t and corresponding approximations to $y(t)$ in the two 11×1 arrays

$$T = \begin{bmatrix} 0 \\ 0.200000000000000 \\ 0.400000000000000 \\ 0.600000000000000 \\ 0.800000000000000 \\ 1.000000000000000 \\ 1.200000000000000 \\ 1.400000000000000 \\ 1.600000000000000 \\ 1.800000000000000 \\ 2.000000000000000 \end{bmatrix} \quad \text{and} \quad YY = \begin{bmatrix} 0.500000000000000 \\ 0.829298806102213 \\ 1.214087852350423 \\ 1.648940818452941 \\ 2.127229773242783 \\ 2.640859343211129 \\ 3.179941816703143 \\ 3.732400315281681 \\ 4.283484106130412 \\ 4.815176603278762 \\ 5.305472351203669 \end{bmatrix}$$

Variable Step-Size Predictor-Corrector Methods

The Runge-Kutta-Fehlberg method is popular for error control because at each step it provides, at little additional cost, *two* approximations that can be compared and related to the local error. Predictor-corrector techniques always generate two approximations at each step, so they are natural candidates for error-control adaptation.

To demonstrate the procedure, we construct a *variable-step-size predictor-corrector* method using the explicit Adams-Bashforth Four-Step method as predictor and the implicit Adams-Moulton Three-Step method as corrector.

The Adams-Bashforth Four-Step method comes from the equation

$$y(t_{i+1}) = y(t_i) + \frac{h}{24}[55f(t_i, y(t_i)) - 59f(t_{i-1}, y(t_{i-1})) + 37f(t_{i-2}, y(t_{i-2})) - 9f(t_{i-3}, y(t_{i-3}))] + \frac{251}{720}y^{(5)}(\hat{\mu}_i)h^5$$

for some $\hat{\mu}_i$ in (t_{i-3}, t_{i+1}). Suppose we assume that the approximations $w_0, w_1, \ldots, w_i$ are all exact and, as in the case of the one-step methods, we let z represent the solution to the differential equation satisfying the initial condition $z(t_i) = w_i$. Then the difference between the actual solution and the predicted solution $w_{i+1,p}$ is

$$z(t_{i+1}) - w_{i+1,p} = \frac{251}{720}z^{(5)}(\hat{\mu}_i)h^5 \quad \text{for some } \hat{\mu}_i \text{ in } (t_{i-3}, t_i). \tag{5.4}$$

A similar analysis of the Adams-Moulton Three-Step corrector method leads to the local error

$$z(t_{i+1}) - w_{i+1} = -\frac{19}{720} z^{(5)}(\tilde{\mu}_i) h^5 \quad \text{for some } \tilde{\mu}_i \text{ in } (t_{i-2}, t_{i+1}). \tag{5.5}$$

To proceed further, we must make the assumption that for small values of h,

$$z^{(5)}(\hat{\mu}_i) \approx z^{(5)}(\tilde{\mu}_i),$$

and the effectiveness of the error-control technique depends directly on this assumption.

If we subtract Eq. (5.5) from Eq. (5.4), and assume that $z^{(5)}(\hat{\mu}_i) \approx z^{(5)}(\tilde{\mu}_i)$, we have

$$w_{i+1} - w_{i+1,p} = \frac{h^5}{720}\left[251 z^{(5)}(\hat{\mu}_i) + 19 z^{(5)}(\tilde{\mu}_i)\right] \approx \frac{3}{8} h^5 z^{(5)}(\tilde{\mu}_i),$$

so

$$z^{(5)}(\tilde{\mu}_i) \approx \frac{8}{3h^5}(w_{i+1} - w_{i+1,p}).$$

Using this result to eliminate the term involving $h^5 z^{(5)}(\tilde{\mu}_i)$ from Eq. (5.5) gives the approximation to the error

$$|z(t_{i+1}) - w_{i+1}| \approx \frac{19h^5}{720} \cdot \frac{8}{3h^5} |w_{i+1} - w_{i+1,p}| = \frac{19|w_{i+1} - w_{i+1,p}|}{270}.$$

This expression was derived assuming that $w_0, w_1, \ldots, w_i$ are the exact values of $y(t_0), y(t_1), \ldots, y(t_i)$, respectively, which means that this is an approximation to the local error. As in the case of the one-step methods, the global error is of order one degree less, so for the function y that is the solution to the original initial-value problem,

$$y' = f(t, y), \quad \text{for } a \le t \le b, \quad \text{with } y(a) = \alpha,$$

the global error can be estimated by

$$|y(t_{i+1}) - w_{i+1}| \approx \frac{|z(t_{i+1}) - w_{i+1}|}{h} \approx \frac{19|w_{i+1} - w_{i+1,p}|}{270h}.$$

Suppose that we now reconsider the situation with a new step size qh generating new approximations $\hat{w}_{i+1,p}$ and $\hat{w}_{i+1}$. To control the global error to within ε, we want to choose q so that

$$\frac{|z(t_i + qh) - \hat{w}_{i+1} \text{ (using the step size } qh)|}{qh} < \varepsilon.$$

But from Eq. (5.5),

$$\frac{|z(t_i + qh) - \hat{w}_{i+1}(\text{using } qh)|}{qh} = \frac{19}{720}|z^{(5)}(\tilde{\mu}_i)|q^4h^4 \approx \frac{19}{720}\left[\frac{8}{3h^5}|w_{i+1} - w_{i+1,p}|\right]q^4h^4,$$

so we need to choose q with

$$\frac{19}{720}\left[\frac{8}{3h^5}|w_{i+1} - w_{i+1,p}|\right] q^4h^4 = \frac{19}{270}\frac{|w_{i+1} - w_{i+1,p}|}{h} q^4 < \varepsilon.$$

Consequently, we need the change in step size from h to qh, where q satisfies

$$q < \left(\frac{270}{19}\frac{h\varepsilon}{|w_{i+1} - w_{i+1,p}|}\right)^{1/4} \approx 2\left(\frac{h\varepsilon}{|w_{i+1} - w_{i+1,p}|}\right)^{1/4}.$$

A number of approximation assumptions have been made in this development, so in practice q is chosen conservatively, usually as

$$q = 1.5\left(\frac{h\varepsilon}{|w_{i+1} - w_{i+1,p}|}\right)^{1/4}.$$

A change in step size for a multistep method is more costly in terms of functional evaluations than for a one-step method because new equally-spaced starting values must be computed. As a consequence, it is also common practice to ignore the step-size change whenever the global error is between $\varepsilon/10$ and ε; that is, when

$$\frac{\varepsilon}{10} < \frac{|y(t_{i+1}) - w_{i+1}|}{h} \approx \frac{19}{270h}|w_{i+1} - w_{i+1,p}| < \varepsilon.$$

The program VPRCOR56 implements the Adams Variable Predictor-Corrector method.

In addition, q is generally given an upper bound to ensure that a single unusually accurate approximation does not result in too large a step size. Program VPRCOR56 incorporates this safeguard with an upper bound of four times the previous step size.

It should be emphasized that because the multistep methods require equal step sizes for the starting values, any change in step size necessitates recalculating new starting values at that point. In VPRCOR56, this is done by incorporating as a subroutine the program RKO4M53, which is the Runge-Kutta method of order 4.

Example 2 Use the Adams Variable Step-Size Predictor-Corrector method with maximum step size $hmax = 0.2$, minimum step size $hmin = 0.01$, and tolerance $TOL = 10^{-5}$ to approximate the solution of the initial-value problem

$$y' = y - t^2 + 1, \quad 0 \le t \le 2, \quad y(0) = 0.5.$$

Solution We begin with $h = hmax = 0.2$, and obtain w_0, w_1, w_2, and w_3 using Runge-Kutta, then find w_{4p} and w_4 by applying the predictor-corrector method. These calculations were done in Example 2 of Section 5.3 where it was determined that the Runge-Kutta approximations are

$$y(0) = w_0 = 0.5,\ y(0.2) \approx w_1 = 0.8292933,\ y(0.4) \approx w_2 = 1.2140762,\ \text{and}$$
$$y(0.6) \approx w_3 = 1.6489220.$$

The predictor and corrector method in Example 2 of Section 5.4 gave

$$y(0.8) \approx w_{4p} = w_3 + \frac{0.2}{24}(55f(0.6, w_3) - 59f(0.4, w_2) + 37f(0.2, w_1) - 9f(0, w_0))$$
$$= 2.1272892,$$

and

$$y(0.8) \approx w_4 = w_3 + \frac{0.2}{24}(9f(0.8, w_{4p}) + 19f(0.6, w_3) - 5f(0.42, w_2) + f(0.2, w_1))$$
$$= 2.1272056.$$

We now need to determine if these approximations are sufficiently accurate or if there needs to be a change in the step size. First we find

$$\sigma = \frac{19}{270h}|w_4 - w_{4p}| = \frac{19}{270(0.2)}|2.1272056 - 2.1272892| = 2.941 \times 10^{-5}.$$

Because this exceeds the tolerance of 10^{-5} a new step size is needed and the new step size is

$$qh = \left(\frac{10^{-5}}{2\sigma}\right)^{1/4} = \left(\frac{10^{-5}}{2(2.941 \times 10^{-5})}\right)^{1/4} (0.2) = 0.642(0.2) \approx 0.128.$$

As a consequence, we need to begin the procedure again computing the Runge-Kutta values with this step size, and then use the predictor-corrector method with this same step size to compute the new values of w_{4p} and w_4. We then need to run the accuracy check on these approximations to see that we have been successful. Table 5.14 shows that this second run is successful and lists all the results obtained using the Adams Variable Predictor-Corrector method. ■

Table 5.14

t_i	$y(t_i)$	w_i	h_i	σ_i	$\lvert y(t_i) - w_i \rvert$
0	0.5	0.5			
0.1257017	0.7002323	0.7002318	0.1257017	4.051×10^{-6}	0.0000005
0.2514033	0.9230960	0.9230949	0.1257017	4.051×10^{-6}	0.0000011
0.3771050	1.1673894	1.1673877	0.1257017	4.051×10^{-6}	0.0000017
0.5028066	1.4317502	1.4317480	0.1257017	4.051×10^{-6}	0.0000022
0.6285083	1.7146334	1.7146306	0.1257017	4.610×10^{-6}	0.0000028
0.7542100	2.0142869	2.0142834	0.1257017	5.210×10^{-6}	0.0000035
0.8799116	2.3287244	2.3287200	0.1257017	5.913×10^{-6}	0.0000043
1.0056133	2.6556930	2.6556877	0.1257017	6.706×10^{-6}	0.0000054
1.1313149	2.9926385	2.9926319	0.1257017	7.604×10^{-6}	0.0000066
1.2570166	3.3366642	3.3366562	0.1257017	8.622×10^{-6}	0.0000080
1.3827183	3.6844857	3.6844761	0.1257017	9.777×10^{-6}	0.0000097
1.4857283	3.9697541	3.9697433	0.1030100	7.029×10^{-6}	0.0000108
1.5887383	4.2527830	4.2527711	0.1030100	7.029×10^{-6}	0.0000120
1.6917483	4.5310269	4.5310137	0.1030100	7.029×10^{-6}	0.0000133
1.7947583	4.8016639	4.8016488	0.1030100	7.029×10^{-6}	0.0000151
1.8977683	5.0615660	5.0615488	0.1030100	7.760×10^{-6}	0.0000172
1.9233262	5.1239941	5.1239764	0.0255579	3.918×10^{-8}	0.0000177
1.9488841	5.1854932	5.1854751	0.0255579	3.918×10^{-8}	0.0000181
1.9744421	5.2460056	5.2459870	0.0255579	3.918×10^{-8}	0.0000186
2.0000000	5.3054720	5.3054529	0.0255579	3.918×10^{-8}	0.0000191

MATLAB uses the command `ode113` to implement a variable-step size and variable-order Adams-Bashforth-Moulton method. It can be more efficient than `ode45` for smaller error tolerances because it can use higher-order methods if needed.

EXERCISE SET 5.6

1. The initial-value problem

$$y' = \sqrt{2 - y^2}e^t, \quad \text{for } 0 \le t \le 0.8, \quad \text{with } y(0) = 0$$

has the exact solution $y(t) = \sqrt{2}\sin(e^t - 1)$.

a. Use the Runge-Kutta-Fehlberg method with tolerance $TOL = 10^{-4}$ to find w_1. Compare the approximate solution to the actual solution.

b. Use the Adams Variable-Step-Size Predictor-Corrector method with tolerance $TOL = 10^{-4}$ and starting values from the Runge-Kutta method of order 4 to find w_4. Compare the approximate solution to the actual solution.

2. The initial-value problem

$$y' = -y + 1 - \frac{y}{t}, \quad \text{for } 1 \le t \le 2, \quad \text{with } y(1) = 1$$

has the exact solution $y(t) = 1 + (e^{1-t} - 1)t^{-1}$.

   a. Use the Runge-Kutta-Fehlberg method with tolerance $TOL = 10^{-3}$ to find w_1 and w_2. Compare the approximate solutions to the actual values.

   b. Use the Adams Variable-Step-Size Predictor-Corrector method with tolerance $TOL = 0.002$ and starting values from the Runge-Kutta method of order 4 to find w_4 and w_5. Compare the approximate solutions to the actual values.

3. Use the Runge-Kutta-Fehlberg method with tolerance $TOL = 10^{-4}$ to approximate the solution to the following initial-value problems.

   a. $y' = \left(\frac{y}{t}\right)^2 + \frac{y}{t}$, for $1 \le t \le 1.2$, with $y(1) = 1$, $hmax = 0.05$, and $hmin = 0.02$.

   b. $y' = \sin t + e^{-t}$, for $0 \le t \le 1$, with $y(0) = 0$, $hmax = 0.25$, and $hmin = 0.02$.

   c. $y' = (y^2 + y)t^{-1}$, for $1 \le t \le 3$, with $y(1) = -2$, $hmax = 0.5$, and $hmin = 0.02$.

   d. $y' = -ty + 4ty^{-1}$, for $0 \le t \le 1$, with $y(0) = 1$, $hmax = 0.2$, and $hmin = 0.01$.

4. Use the Runge-Kutta-Fehlberg method with tolerance $TOL = 10^{-6}$, $hmax = 0.5$, and $hmin = 0.05$ to approximate the solutions to the following initial-value problems. Compare the results to the actual values.

   a. $y' = \frac{y}{t} - \frac{y^2}{t^2}$, for $1 \le t \le 4$, with $y(1) = 1$; actual solution $y(t) = t/(1 + \ln t)$.

   b. $y' = 1 + \frac{y}{t} + \left(\frac{y}{t}\right)^2$, for $1 \le t \le 3$, with $y(1) = 0$; actual solution $y(t) = t\tan(\ln t)$.

   c. $y' = -(y+1)(y+3)$, for $0 \le t \le 3$, with $y(0) = -2$; actual solution $y(t) = -3+2(1+e^{-2t})^{-1}$.

   d. $y' = (t+2t^3)y^3 - ty$, for $0 \le t \le 2$, with $y(0) = \frac{1}{3}$; actual solution $y(t) = (3+2t^2+6e^{t^2})^{-1/2}$.

5. Use the Adams Variable-Step-Size Predictor-Corrector method with $TOL = 10^{-4}$ to approximate the solutions to the initial-value problems in Exercise 3.

6. Use the Adams Variable-Step-Size Predictor-Corrector method with tolerance $TOL = 10^{-6}$, $hmax = 0.5$, and $hmin = 0.02$ to approximate the solutions to the initial-value problems in Exercise 4.

7. An electrical circuit consists of a capacitor of constant capacitance $C = 1.1$ farads in series with a resistor of constant resistance $R_0 = 2.1$ ohms. A voltage $\mathcal{E}(t) = 110\sin t$ is applied at time $t = 0$. When the resistor heats up, the resistance becomes a function of the current i,

$$R(t) = R_0 + ki, \quad \text{where } k = 0.9,$$

and the differential equation for i becomes

$$\left(1 + \frac{2k}{R_0}i\right)\frac{di}{dt} + \frac{1}{R_0C}i = \frac{1}{R_0C}\frac{d\mathcal{E}}{dt}.$$

Find the current i after 2 s, assuming $i(0) = 0$.

5.7 Methods for Systems of Equations

The most common application of numerical methods for approximating the solution of initial-value problems concerns not a single problem, but a linked system of differential equations. Why, then, have we spent the majority of this chapter considering the solution of a single equation? The answer is simple: to approximate the solution of a system of initial-value problems, we successively apply the techniques that we used to solve problems involving a single equation. As is so often the case in mathematics, the key to the methods