The error at $t = 10.0$ s for $\Delta t = 1.0$ s is approximately 110,000 times smaller than the error presented in Table 7.3 for the explicit Euler method and 3,500 times smaller than the error presented in Table 7.6 for the modified Euler method. Results such as these clearly demonstrate the advantage of higher-order methods. An error analysis at $t = 10.0$ s gives

$$\text{Ratio} = \frac{E(\Delta t = 2.0)}{E(\Delta t = 1.0)} = \frac{-0.008855569}{-0.000260369} = 34.01$$

which shows that the method is fourth order, since the theoretical error ratio for $0(\Delta t^4)$ method is 16.0. The value Ratio = 34.01 instead of 16.0 is obtained since the higher-order terms in the truncation error are still significant.

7.7.4 Error Estimation and Error Control for Single-Point Methods

Consider a FDE of $0(\Delta t^m)$. For a single step:

$$\bar{y}(t_{n+1}) = y(t_{n+1}, \Delta t) + A\ \Delta t^{m+1} \tag{7.199}$$

where $\bar{y}(t_{n+1})$ denotes the exact solution at t_{n+1}, $y(t_{n+1}, \Delta t)$ denotes the approximate solution at t_{n+1} with increment Δt, and $A\ \Delta t^{m+1}$ is the local truncation error. How can the magnitude of the local truncation error $A\ \Delta t^{m+1}$ be estimated? Repeat the calculation using step size $\Delta t/2$. Thus,

$$\bar{y}(t_{n+1}) = y\left(t_{n+1}, \frac{\Delta t}{2}\right) + 2\left[A\left(\frac{\Delta t}{2}\right)^{m+1}\right] \tag{7.200}$$

This process is illustrated in Figure 7.14. Subtract Eq. (7.200) from Eq. (7.199) and solve for $A\ \Delta t^{m+1}$, which is the local trucation error.

$$\text{Error} = A\ \Delta t^{m+1} = \left[y_{n+1}\left(t_{n+1}, \frac{\Delta t}{2}\right) - y_{n+1}(t_{n+1}, \Delta t)\right]\left(\frac{2^m}{2^m - 1}\right) \tag{7.201}$$

If |Error| < (lower error limit), increase (double) the step size. If |Error| > (upper

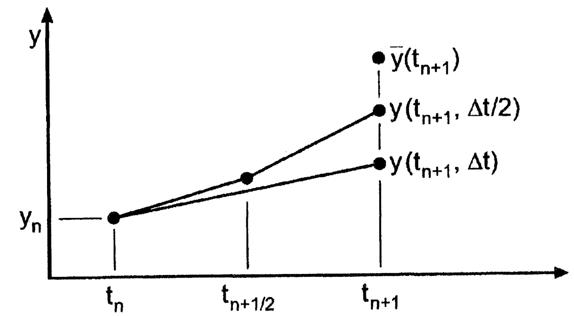

Figure 7.14 Step size halving for error estimation.

7.7.5 Runge-Kutta Methods with Error Estimation

Runge-Kutta methods with more function evaluations have been devised in which the additional results are used for error estimation. The *Runge-Kutta-Fehlberg method* [Fehlberg (1966)] uses six derivative function evaluations:

$$y_{n+1} = y_n + \left(\tfrac{16}{135}k_1 + \tfrac{6656}{12825}k_3 + \tfrac{28561}{56430}k_4 - \tfrac{9}{50}k_5 + \tfrac{2}{55}k_6\right) \quad 0(h^6) \tag{7.202}$$

$$\tilde{y}_{n+1} = y_n + \left(\tfrac{25}{216}k_1 + \tfrac{1408}{2565}k_3 + \tfrac{2197}{4101}k_4 - \tfrac{1}{5}k_5\right) \quad 0(h^5) \tag{7.203}$$

$$k_1 = \Delta t\, f(t_n, y_n) \tag{7.204a}$$

$$k_2 = \Delta t\, f(t_n + \tfrac{1}{4}h, y_n + \tfrac{1}{4}k_1) \tag{7.204b}$$

$$k_3 = \Delta t\, f(t_n + \tfrac{3}{8}h, y_n + \tfrac{3}{32}k_1 + \tfrac{9}{32}k_2) \tag{7.204c}$$

$$k_4 = \Delta t\, f(t_n + \tfrac{12}{13}h, y_n + \tfrac{1932}{2197}k_1 - \tfrac{7200}{2197}k_2 + \tfrac{7296}{2197}k_3) \tag{7.204d}$$

$$k_5 = \Delta t\, f(t_n + h, y_n + \tfrac{439}{216}k_1 - 8k_2 + \tfrac{3680}{513}k_3 - \tfrac{845}{4104}k_4) \tag{7.204e}$$

$$k_6 = \Delta t\, f(t_n + \tfrac{1}{2}h, y_n - \tfrac{8}{27}k_1 + 2k_2 - \tfrac{3544}{2565}k_3 + \tfrac{1859}{4104}k_4 - \tfrac{11}{40}k_5) \tag{7.204f}$$

The error is estimated as follows. Equations (7.202) and (7.203) can be expressed as follows:

$$\bar{y}_{n+1} = y_{n+1} + 0(h^6) \tag{7.205}$$

$$\bar{y}_{n+1} = \tilde{y}_{n+1} + 0(h^5) + 0(h^6) = \tilde{y}_{n+1} + \text{Error} + 0(h^6) \tag{7.206}$$

Substituting y_{n+1} and $\tilde{y}_{n+1}$, Eqs. (7.202) and (7.203), into Eqs. (7.205) and (7.206) and subtracting yields

$$\text{Error} = \tfrac{1}{360}k_1 - \tfrac{128}{4275}k_3 - \tfrac{2197}{75{,}240}k_4 + \tfrac{1}{50}k_5 + \tfrac{2}{55}k_6 + 0(h^6) \tag{7.207}$$

The error estimate, Error, is used for step size control. Use y_{n+1}, which is $0(h^6)$ locally, as the final value of y_{n+1}.