COMPUTER SCIENCE 349A
Handout Number 35

NOTES ON THE ANALYSIS OF EULER'S METHOD

(Section 25.1.1 : pages 710-715 of the 6th ed. or pages 712-717 of the 7th ed.)

Initial-value problem:

$$y'(x) = f(x, y(x)), \qquad a \le x \le b$$

$$y(x_0) = y_0 \ .$$

Exact solution at x_i is denoted by $y(x_i)$.

Computed approximation at x_i is denoted by y_i.

Definition

$|y(x_i) - y_i|$ is called the **global truncation error** at x_i .

Definition

If the global truncation error is $O(h^k)$, the numerical method used to compute the values y_i is said to be of **order *k*** (or a k^{th} order method).

The order of a method is a measure of the accuracy of the computed approximations, or of the rate of convergence of the computed approximations y_i to the exact solutions $y(x_i)$ as $h \to 0$. For any fixed value of the step size *h*, the <u>larger the order</u> *k*, the more accurate are the computed approximations.

Definition

A numerical method is said to be **convergent** (with respect to the differential equation it approximates) if

$$\lim_{h \to 0} \max_{1 \le i \le N} |y(x_i) - y_i| = 0 .$$

(That is, the global truncation error $\to 0$ at all grid points x_i in [*a*, *b*] as $h \to 0$.)

The total amount of truncation error in each computed approximation y_{i+1} (using <u>any</u> numerical method) is composed of two parts: the **local truncation error** is the amount of truncation error that results from <u>a single application</u> of a numerical method (that is, from the computation of y_{i+1} from y_i), whereas the **global truncation error** contains the accumulated local truncation errors from all of the steps leading up to the computation of y_{i+1}.

Definition

The **local truncation error** at any point x_{i+1} is the amount of truncation error that would result from using a numerical method with the exact value $y(x_i)$ rather than the computed approximation y_i.

Example: the derivation of the local truncation error for Euler's method

Euler's method is

$$y_{i+1} = y_i + h\, f(x_i, y_i).$$

Using the exact value $y(x_i)$ in this formula instead of the computed approximation y_i, define

$$v_{i+1} = y(x_i) + h\, f(x_i, y(x_i)). \tag{1}$$

Then the **local truncation error** at x_{i+1} is equal to

$$|y(x_{i+1}) - v_{i+1}|.$$

Note that this local truncation error does not contain all of the accumulated local truncation errors from all of the steps leading up to the computation of y_{i+1} (as v_{i+1} is computed using the exact solution at x_i, namely $y(x_i)$).

Determination of the order of the local truncation error for Euler's method: use Taylor's Theorem (see pages 710-711 of the 6th ed. or pages 712-713 of the 7th ed.). Since, by Taylor's Theorem,

$$\begin{aligned} y(x_{i+1}) &= y(x_i) + h\, y'(x_i) + \frac{h^2}{2} y''(\xi_i) \\ &= y(x_i) + h\, f(x_i, y(x_i)) + \frac{h^2}{2} y''(\xi_i) \end{aligned} \tag{2}$$

for some value ξ_i between x_i and x_{i+1}, from (1) and (2) we obtain

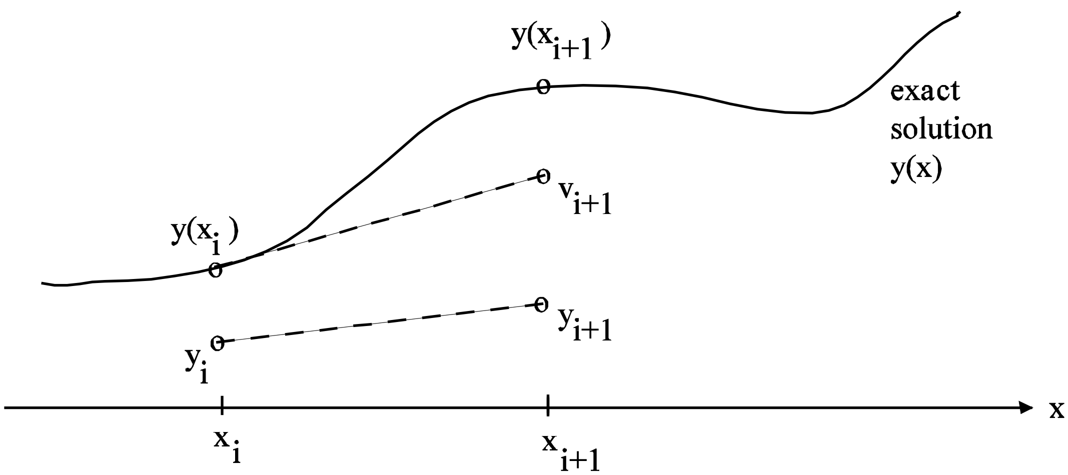
$$|y(x_{i+1}) - v_{i+1}| = \left|\frac{h^2}{2} y''(\xi_i)\right|.$$

That is, for **Euler's method**, the local truncation error is $O(h^2)$.

Relation between the local and global truncation error for any numerical method

If the local truncation error is $O(h^{k+1})$, then the global truncation error is $O(h^k)$. That is, the numerical method used to compute the approximate solution has **order *k***.

Thus, the **global truncation error** for Euler's method is $O(h)$, and Euler's method has order 1.



An informal justification for the relationship between the local and global truncation errors for Euler's method is as follows. The local truncation error in each step of Euler's method is, as shown above, $O(h^2)$. After N steps of Euler's method, the global truncation error $|y(x_N) - y_N|$ of the final computed approximation y_N at $x_N = b$ will depend on the N local truncation errors of $y_1, y_2, \ldots, y_N$. But the magnitude of N local truncation errors is

$$N \times O(h^2) = \frac{b-a}{h} \times O(h^2) = O(h) \quad \text{since} \quad h = \frac{b-a}{N} .$$

At a point x_{i+1}, the global truncation error $|y(x_{i+1}) - y_{i+1}|$ is the actual difference between the exact value and the computed approximation, so ideally we would like to be able to control or estimate its magnitude. However, the global truncation error of a numerical method is difficult to compute or to estimate directly. But there are good ways of estimating the local truncation error of a numerical method, and because of the above relationship between the global and local truncation errors, these estimates of the local truncation error can be used to obtain accurate estimates of the global truncation error, and thus of the actual error of the computed approximations. This is the done in Section 25.5 for Runge-Kutta methods: **adaptive Runge-Kutta methods**.

Disadvantage of Euler's method -- it is not sufficiently accurate. Because the global truncation error is only $O(h)$, a very small step size h is required to compute highly accurate approximations. Methods with a higher order of accuracy are used in practice.

Section 25.1.3 (page 719 of the 6th ed. or page 721 of the 7th ed.)

One approach to deriving higher-order numerical methods is to use Taylor's theorem: since Euler's method can be derived from the first two terms of the Taylor expansion, higher order methods can be obtained by <u>keeping more terms</u> from the Taylor expansion.

$$\begin{aligned} y(x_{i+1}) &= y(x_i) + hy'(x_i) + \frac{h^2}{2}y''(x_i) + \cdots + \frac{h^n}{n!}y^{(n)}(x_i) + \frac{h^{n+1}}{(n+1)!}y^{(n+1)}(\xi_i) \\ &= y(x_i) + h\,f(x_i, y(x_i)) + \frac{h^2}{2}f'(x_i, y(x_i)) + \cdots + \frac{h^n}{n!}f^{(n-1)}(x_i, y(x_i)) + O(h^{n+1}) \end{aligned}$$

since $y'(x) = f(x, y(x))$. Note that all derivatives are with respect to x.

Dropping the $O(h^{n+1})$ remainder term in the above Taylor expansion, gives a numerical method

$$y_{i+1} = y_i + h\,f(x_i, y_i) + \frac{h^2}{2}f'(x_i, y_i) + \cdots + \frac{h^n}{n!}f^{(n-1)}(x_i, y_i),$$

for any integer $n \geq 1$. This is called **the Taylor method of order *n*** (as its local truncation error is $O(h^{n+1})$, and thus its global truncation error is $O(h^n)$). **Euler's method** is just the case $n = 1$.

Example

The Taylor method of order $n = 2$ is

$$y_{i+1} = y_i + h\,f(x_i, y_i) + \frac{h^2}{2}f'(x_i, y_i).$$

Application of this method to solve the initial-value problem

$$y' = y - x^2 + 1 \qquad \text{subject to } y(0) = 0.5 .$$

Evaluate

$$f'(x, y(x)) = \frac{d}{dx}\left(y - x^2 + 1\right) = y' - 2x = y - x^2 + 1 - 2x.$$

Thus, the **iterative formula** for the Taylor method of order $n = 2$ is

$$\begin{aligned} y_{i+1} &= y_i + h\, f(x_i, y_i) + \frac{h^2}{2} f'(x_i, y_i) \\ &= y_i + h\left[y_i - x_i^2 + 1\right] + \frac{h^2}{2}\left[y_i - x_i^2 + 1 - 2x_i\right] \\ &= y_i + h\left[\left(1 + \frac{h}{2}\right)\left(y_i - x_i^2 + 1\right) - h x_i\right] \end{aligned}$$

The first step of the numerical computation: the given initial condition is $y_0 = 0.5$ at $x_0 = 0$. If we choose $h = 0.2$ then we obtain (with $i = 1$ in the above iterative formula)

$$\begin{aligned} y_1 &= y_0 + h\left[\left(1 + \frac{h}{2}\right)\left(y_0 - x_0^2 + 1\right) - h x_0\right] \\ &= 0.5 + (0.2)\left[\left(1 + \frac{0.2}{2}\right)\left(0.5 - 0^2 + 1\right) - (0.2)(0)\right] \\ &= 0.83 \end{aligned}$$

Note: the exact solution is $y(0.2) = 0.8292986$.

****

For $n \geq 2$, use of a Taylor method requires evaluation of the derivatives of the function $f(x, y(x))$ with respect to *x*.

Although the <u>Taylor methods of order *n*</u> have high accuracy for values of n = 3, 4 or 5, they are <u>seldom used</u> in practice because of the difficulty and expense in evaluating the required higher derivatives.

Runge-Kutta methods are a class of higher-order methods that are more often used in practice.