21.1 Methods for First-Order ODEs

Tuesday, November 9, 2021 2:09 PM

- Consider an initial value, first order ODE:

$$y' = f(x,y) \quad ; \quad y(x_0) = y_0 \qquad \text{where } a<x<b \text{ contains } x_0$$

- We look at x values spaced equally in the interval:

$$x_1 = x_0 + h \ , \ x_2 = x_0 + 2h \ , \ x_3 = x_0 + 3h, \ \dots$$

h is the step size

- Taylor's formula for $y(x+h)$ is:

$$y(x+h) = y(x) + hy' + \frac{h^2}{2}y'' + \frac{h^3}{3!} + \dots$$

- For small h, h^2, h^3, etc are very small such that

$$y(x+h) \approx y(x) + hy'$$
$$= y(x) + hf(x,y) \quad \rightarrow \quad y' = f(x,y) = \frac{y(x+h) - y(x)}{h}$$

or

$$y_{n+1} = y_n + hf(x_n, y_n) \qquad (n = 0, 1, \dots)$$ Euler's Method

Note: $y(x+h) = y(x) + hy' + \underbrace{\frac{h^2}{2}y''(\xi)}_{\text{local truncation error } \mathcal{O}(h^2)}$ where $x \le \xi \le x+h$

- Improved Euler Method

First compute the 'predictor': $y^*_{n+1} = y_n + hf(x_n, y_n)$

Then compute the 'corrector" $y_{n+1} = y_n + \frac{1}{2}h\left[f(x_n, y_n) + f(x_{n+1}, y^*_{n+1})\right]$

Example: $y' = y + x \ ; \ y(0) = 0 \ ; \ h = 0.2$

true solution $y = e^x - x - 1$

Euler Method $y_n = y_{n-1} + hf(x_{n-1}, y_{n-1})$

n	x_n	y_n	f_n	$y(x_n)$	Error
0	0.0	0	0	0	0
1	0.2	0	0.2	0.021	0.021
2	0.4	0.04	0.44	0.092	0.052
3	0.6	0.128	0.728	0.222	0.094
4	0.8	0.274	1.074	0.426	0.152
5	1.0	0.488	0.977	0.718	0.230

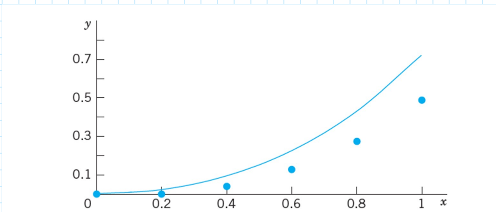


Fig. 9. Euler method: Approximate values in Table 1.1 and solution curve

Improved Euler Method

Algorithm:

For $n=0,1,\dots,N-1$

$x_{n+1} = x_n + h$

$k_1 = h f(x_n, y_n)$

$k_2 = h f(x_{n+1}, y_n + k_1)$

$y_{n+1} = y_n + \frac{1}{2}(k_1 + k_2)$

OUTPUT x_{n+1}, y_{n+1}

End

Table 21.2 **Improved Euler Method for (6). Errors**

n	x_n	y_n	Exact Values (4D)	Error of Improved Euler	Error of Euler
0	0.0	0.0000	0.0000	0.0000	0.000
1	0.2	0.0200	0.0214	0.0014	0.021
2	0.4	0.0884	0.0918	0.0034	0.052
3	0.6	0.2158	0.2221	0.0063	0.094
4	0.8	0.4153	0.4255	0.0102	0.152
5	1.0	0.7027	0.7183	0.0156	0.230

- Runge-Kutta Method

ALGORITHM RUNGE–KUTTA (f, x_0, y_0, h, N).

This algorithm computes the solution of the initial value problem $y' = f(x, y)$, $y(x_0) = y_0$ at equidistant points

(9) $$x_1 = x_0 + h, x_2 = x_0 + 2h, \cdots, x_N = x_0 + Nh;$$

here f is such that this problem has a unique solution on the interval $[x_0, x_N]$ (see Sec. 1.7).

INPUT: Function f, initial values x_0, y_0, step size h, number of steps N

OUTPUT: Approximation y_{n+1} to the solution $y(x_{n+1})$ at $x_{n+1} = x_0 + (n+1)h$, where $n = 0, 1, \cdots, N-1$

For $n = 0, 1, \cdots, N-1$ do:

$k_1 = hf(x_n, y_n)$

$k_2 = hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1)$

$k_3 = hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_2)$

$k_4 = hf(x_n + h, y_n + k_3)$

$x_{n+1} = x_n + h$

$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$

OUTPUT x_{n+1}, y_{n+1}

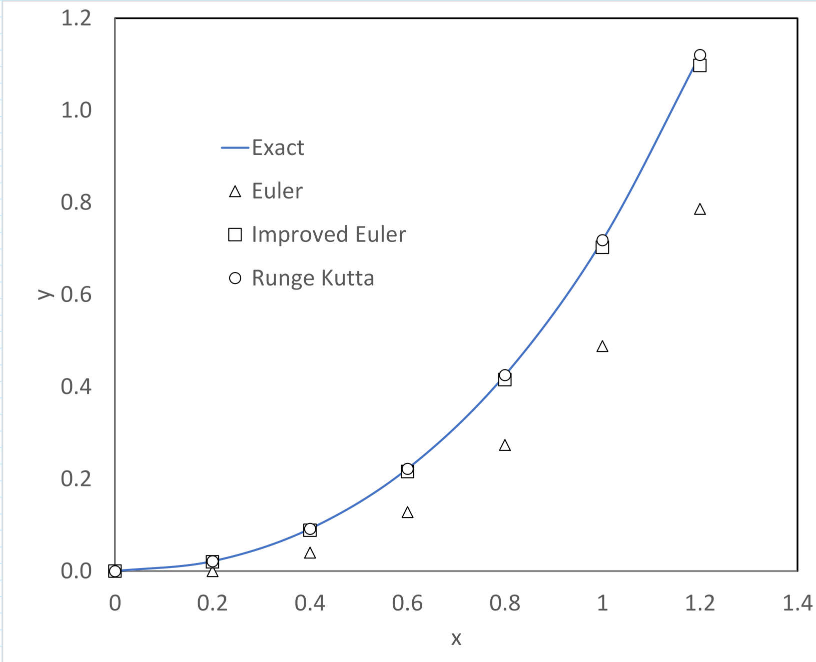
End

Stop

End RUNGE–KUTTA

for: $y' = f(x,y) = y + x$; $y(0) = 0$

x	$y = e^x - x - 1$	Error		
		Euler (Table 21.1)	Improved Euler (Table 21.3)	Runge–Kutta (Table 21.5)
0.2	0.021403	0.021	0.0014	0.000003
0.4	0.091825	0.052	0.0034	0.000007
0.6	0.222119	0.094	0.0063	0.000011
0.8	0.425541	0.152	0.0102	0.000020
1.0	0.718282	0.230	0.0156	0.000031



1.2
1.0
0.8
0.6
0.4
0.2
0.0
y
Exact
Euler
Improved Euler
Runge Kutta
0
0.2
0.4
0.6
0.8
1
1.2
1.4
x

21.2 Multistep Methods

Wednesday, November 10, 2021 9:25 AM

- **One-step methods** are self starting for i.v.p. b/c $y(0) = y_0$ is given.
- This allows easy calculation of y_{n+1}
- **Multistep methods** use two or more previous values to increase accuracy.

Adams-Bashforth Methods

$$y' = f(x,y) \; ; \; y(0) = y_0 \qquad (\text{an initial value problem})$$

$$\int_{x_n}^{x_{n+1}} y'(x)\,dx = y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x, y(x))\,dx$$

- Now, replace $f(x, y(x))$ with an interpolation polynomial

$$y_{n+1} - y_n = \int_{x_n}^{x_{n+1}} p(x)\,dx$$

let's use a cubic such that at equal spacings

$$f_n = f(x_n, y_n)$$

$$f_{n-1} = f(x_{n-1}, y_{n-1})$$

etc.

from Newton's backward difference formula (19.3)

$$P_3(x) = f(x) = f_0 + r\nabla f_0 + \frac{r(r+1)}{2!}\nabla^2 f_0 + \cdots + \frac{r(r+1)\cdots(r+n-1)}{n!}\nabla^n f_0$$

where:

$$\nabla f_j = f_j - f_{j-1}$$

$$\nabla^2 f_j = \nabla f_j - \nabla f_{j-1}$$

$$\vdots$$

$$\nabla^k f_j = \nabla^{k-1} f_j - \nabla^{k-1} f_{j-1}$$

$$r = \frac{x - x_n}{h}$$

(see book for intermediat steps of derivation)

(See book for intermediate steps of derivation)

$$\boxed{y_{n+1} = y_n + \frac{h}{24}\left(55 f_n - 59 f_{n-1} + 37 f_{n-2} - 9 f_{n-3}\right)}$$