ODEs: Runge-Kutta methods

September 27, 2021

Forward Euler

- Problem: Solve the ordinary differential equation $y'(t) = f(y,t)$ with initial data $y(0) = y_0$ on the domain $[0,L]$

Forward Euler

- Problem: Solve the ordinary differential equation $y'(t) = f(y,t)$ with initial data $y(0) = y_0$ on the domain $[0, L]$

- We can derive Forward Euler's Method by using Taylor series:

$$\begin{aligned} y_{n+1} = y(t_n + h) &= y(t_n) + hy'(t_n) + \frac{1}{2}(h)^2 y''(t_n) + \cdots \\ &= y(t_n) + hf(t_n, y_n) + \frac{1}{2}(h)^2 y''(t_n) + \cdots \end{aligned}$$

so that the scheme become

$$y_{n+1} = y_n + hf(y_n, t_n)$$

Runge-Kutta Methods

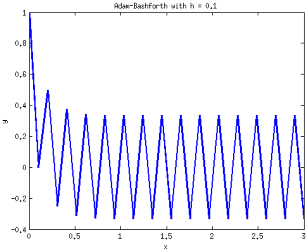
- Forward Euler is a *first order* method. If we half the time step, we should half the error.

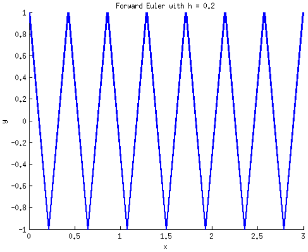
Runge-Kutta Methods

- Forward Euler is a *first order* method. If we half the time step, we should half the error.
- Last week we looked at Adams Bashforth, a *second order multistep method*. This means if we half the error, we should decrease the error by a factor of 4.

Runge-Kutta Methods

- Forward Euler is a *first order* method. If we half the time step, we should half the error.
- Last week we looked at Adams Bashforth, a *second order multistep method*. This means if we half the error, we should decrease the error by a factor of 4.
- In using the second order Adams Bashforth method, our method became less stable.
- Can we get a second order *explicit* method without sacrificing stability?





Runge-Kutta Methods

- Recall Forward Euler gave us the method:

$$y_{n+1} = y_n + hf(t_n, y_n) = y_n + hy'(t_n)$$

- Solving for y', we get

$$\frac{y_{n+1} - y_n}{h} \approx y'(t_n)$$

- This is a *first order* approximation of the derivative.
- Instead, we can approximate $y'(t_n + \frac{h}{2})$ by

$$\frac{y_{n+1} - y_n}{h} \approx y'\left(t_n + \frac{h}{2}\right)$$

This is a *second order* approximation.

Runge-Kutta Methods

- Idea: Replace $y'(t_n)$ in Euler's Method with $y'(t_n + \frac{h}{2})$ to get a more accurate method:

$$y_{n+1} = y_n + hy'\left(t_n + \frac{h}{2}\right) = y_n + hf\left(t_n + \frac{h}{2}, y\left(t_n + \frac{h}{2}\right)\right)$$

- We don't know the value $y(t_n + \frac{h}{2})$, so Taylor Expand!:

$$y\left(t_n + \frac{h}{2}\right) = y(t_n) + \frac{h}{2}y'(t_n) + \cdots \approx y(t_n) + \frac{h}{2}f(t_n, y(t_n))$$

- This gives us the scheme

$$y_{n+1} = y_n + hf\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}f(t_n, y_n)\right)$$

This is a second order Runge-Kutta method (also called the midpoint method).

Runge-Kutta Methods

- Second order Runge-Kutta method:

$$y_{n+1} = y_n + hf\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} f(t_n, y_n)\right)$$

- There are a whole family of Runge-Kutta methods. Another popular one is a *fourth order* method

$$\begin{aligned} y_{n+1} &= y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ k_1 &= f(t_n, y_n) \\ k_2 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_1\right) \\ k_3 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_2\right) \\ k_4 &= f(t_n + h, y_n + hk_3) \end{aligned}$$