ODEs: Backward Euler

October 18, 2021

Forward Euler

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

Forward Euler

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

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

$$\begin{aligned} y_{n+1} = y(t_n + \Delta t) &= y(t_n) + \Delta t y'(t_n) + \frac{1}{2}(\Delta t)^2 y''(t_n) + \cdots \\ &= y(t_n) + \Delta t f(t_n, y_n) + \frac{1}{2}(\Delta t)^2 y''(t_n) + \cdots \end{aligned}$$

so that the scheme become

$$y_{n+1} = y_n + \Delta t f(t_n, y_n)$$

Backward Euler

- Forward Euler:

$$y_{n+1} = y_n + \Delta t f(t_n, y_n)$$

- For Backward Euler, we use Taylor series, but expand differently:

$$\begin{aligned} y(t_n) = y(t_{n+1} - \Delta t) &= y(t_{n+1}) - \Delta t y'(t_{n+1}) + \frac{1}{2}(\Delta t)^2 y''(t_{n+1}) + \cdots \\ &= y(t_{n+1}) - \Delta t f(t_{n+1}, y_{n+1}) + \frac{1}{2}(\Delta t)^2 y''(t_{n+1}) + \cdots \end{aligned}$$

- This leads to the scheme

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

Backward Euler

- Backward Euler:
$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

- Backward Euler is an *implicit* method because it is involves solving an equation for y_{n+1}. This makes the method much more complicated.

- Why do we use backward Euler?

Backward Euler

- Backward Euler:

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

- Backward Euler is an *implicit* method because it is involves solving an equation for y_{n+1}. This makes the method much more complicated.

- Why do we use backward Euler?

- It is a *stable* method. This means it works well for stiff equations like

$$y' = \lambda y$$

Backward Euler

- Backward Euler:

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

- Another example: consider the ODE $y' = y\cos y$ which has $f(t, y) = y\cos y$.

Backward Euler

- Backward Euler:

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

- Another example: consider the ODE $y' = y \cos y$ which has $f(t, y) = y \cos y$.
- In this case, the scheme becomes

$$y_{n+1} - \Delta t y_{n+1} \cos y_{n+1} = y_n$$

which has no closed form solution.

Backward Euler

- Backward Euler:

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

- Another example: consider the ODE $y' = y \cos y$ which has $f(t, y) = y \cos y$.
- In this case, the scheme becomes

$$y_{n+1} - \Delta t y_{n+1} \cos y_{n+1} = y_n$$

  which has no closed form solution.
- Here, we need to use some root finder, like Newton's Method or the Bisection Method:

$$y_{n+1} - \Delta t y_{n+1} \cos y_{n+1} - y_n = 0$$

Backward Euler

- Backward Euler:

$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$

- Example problem: $y' = y \cos y$.

Backward Euler

- Backward Euler:
$$y_{n+1} = y_n + \Delta t f(t_{n+1}, y_{n+1})$$
- Example problem: $y' = y \cos y$.
- We need to solve the equation
$$y_{n+1} - \Delta t y_{n+1} \cos y_{n+1} - y_n = 0$$
- To find the value y_{n+1} at each step, we need to use Newton's Method to solve $g(y_{n+1}) = 0$:
$$\tilde{y}_{m+1} = \tilde{y}_m - \frac{g(\tilde{y}_m)}{g'(\tilde{y}_m)}$$
- For this problem,
$$g(\tilde{y}) = \tilde{y} - \Delta t \tilde{y} \cos \tilde{y} - y_n = 0$$

Backward Euler

- For these non-linear problems, we often times do not have an exact solution.
- How do we quantify the error?

Backward Euler

- For these non-linear problems, we often times do not have an exact solution.
- How do we quantify the error?
- Recall that the error looks like $e_h = Ch^p$ for some value of p.

Backward Euler

- For these non-linear problems, we often times do not have an exact solution.
- How do we quantify the error?
- Recall that the error looks like $e_h = Ch^p$ for some value of p.
- Suppose we have the values at the final time step for the runs with time steps h, $\frac{h}{2}$, and $\frac{h}{4}$, then we can do a *refinement study*. Then we have

$$\frac{y_h - y_{h/2}}{y_{h/2} - y_{h/4}} = \frac{y_h - y(T) - y_{h/2} + y(T)}{y_{h/2} - y(T) - y_{h/4} + y(T)} = \frac{e_h - e_{h/2}}{e_{h/2} - e_{h/4}}$$

$$= \frac{Ch^p - C\left(\frac{h}{2}\right)^p}{C\left(\frac{h}{2}\right)^p - C\left(\frac{h}{4}\right)^p} = \frac{\frac{2^p h^p - h^p}{2^p}}{\frac{2^p h^p - h^p}{4^p}} = 2^p$$

Backward Euler

- For these non-linear problems, we often times do not have an exact solution.
- How do we quantify the error?
- Recall that the error looks like $e_h = Ch^p$ for some value of p.
- Suppose we have the values at the final time step for the runs with time steps h, $\frac{h}{2}$, and $\frac{h}{4}$, then we can do a *refinement study*. Then we have

$$\begin{aligned}\frac{y_h - y_{h/2}}{y_{h/2} - y_{h/4}} &= \frac{y_h - y(T) - y_{h/2} + y(T)}{y_{h/2} - y(T) - y_{h/4} + y(T)} = \frac{e_h - e_{h/2}}{e_{h/2} - e_{h/4}} \\ &= \frac{Ch^p - C\left(\frac{h}{2}\right)^p}{C\left(\frac{h}{2}\right)^p - C\left(\frac{h}{4}\right)^p} = \frac{\frac{2^p h^p - h^p}{2^p}}{\frac{2^p h^p - h^p}{4^p}} = 2^p\end{aligned}$$

- To find the order of our method, we just need to compute

$$\log_2\left(\frac{y_h - y_{h/2}}{y_{h/2} - y_{h/4}}\right)$$