Section 6.1: Multi-step methods

Tuesday, March 30, 2021 8:56 AM

What is something Euler's Method, Taylor Methods, RK Methods have in common?

- they only rely on previous step.
  w_{j+1} depends only on w_j (not $w_{j-1}, w_{j-2}, \ldots$ etc.)
  recall Euler's Method
  $$w_{j+1} = w_j + \Delta t\,[f(t_j, w_j)]$$

Goal: obtain higher accuracy by using more information. (ie, w_{j+1} will depend on $w_j, w_{j-1}, \ldots$)

General form for an "m" step method is

$$\left.\begin{aligned} w_0 &= \alpha_0 \\ w_1 &= \alpha_1 \\ w_2 &= \alpha_2 \\ &\vdots \\ w_{m-1} &= \alpha_{m-1} \end{aligned}\right\} \text{initial conditions}$$

$$w_{j+1} = a_0 w_j + a_1 w_{j-1} + a_2 w_{j-2} + \cdots + a_m w_{j+1-m} + \Delta t\,[b_0 f(t_{j+1}, w_{j+1}) + b_1 f(t_j, w_j) + b_2 f(t_{j-1}, w_{j-1}) + \cdots + b_m f(t_{j+1-m}, w_{j+1-m})]$$

$w_0 = \alpha_0$
$w_1 = \alpha_1$
$w_{j+1} = a_0 w_j + a_1 w_{j-1} + \Delta t[\ldots]$

Euler's Method

$j=1$: $w_2 = a_0 w_1 + a_1 w_0 + \Delta t[\ldots]$ ($2 = j+1$)

$j=2$: $w_3 = a_0 w_2 + a_1 w_1 + \Delta t[\ldots]$ ($3 = j+1$)

Special case

If $b_0 = 0$, we call method "explicit" (only depends on past)
If $b_0 \neq 0$, we call method "implicit" (w_{j+1} on both sides)

How to obtain a MS method?

Recall that we are solving $y' = f(t,y)$, $y(t_0) = \alpha$.

Let's integrate over $[t_j, t_{j+1}]$

$$\underbrace{\int_{t_j}^{t_{j+1}} y'(t)\,dt}_{\text{by FTC}} = \int_{t_j}^{t_{j+1}} f(t,y)\,dt$$

$$y(t_{j+1}) - y(t_j) = \int_{t_j}^{t_{j+1}} f(t,y)\,dt.$$

without

we can't solve w/o knowing y.
BUT we can approx. $f(t,y)$ with a Lagrange interpolating polynomial $P(t)$

$$y(t_{j+1}) = y(t_j) + \int_{t_j}^{t_{j+1}} P(t)\,dt$$

Let's do an example.

ex/ Use 2 points to create Lagrange Interpolating polynomial Of $f(t,y)$ explicitly (use t_j, t_{j-1})

$$P(t) = f(t_{j-1}, y_{j-1}) \underbrace{L_0(t)}_{\frac{t-t_j}{t_{j-1}-t_j} = -\frac{1}{\Delta t}(t-t_j)} + f(t_j, y_j) \underbrace{L_1(t)}_{\frac{t-t_{j-1}}{t_j - t_{j-1}} = \frac{1}{\Delta t}(t-t_{j-1})}$$

since $t_j - t_{j-1} = \Delta t$

Now plug into $\int_{t_j}^{t_{j+1}} P(t)\,dt$

$$\int_{t_j}^{t_{j+1}} P(t)\,dt = f(t_{j-1}, y_{j-1})\left(-\frac{1}{\Delta t}\right)\int_{t_j}^{t_{j+1}} (t-t_j)\,dt + f(t_j, y_j)\left(\frac{1}{\Delta t}\right)\int_{t_j}^{t_{j+1}} (t-t_{j-1})\,dt$$

$$= -\frac{f(t_{j-1}, y_{j-1})}{\Delta t}\left[\frac{t^2}{2} - t_j t\right]_{t_j}^{t_{j+1}} + \frac{f(t_j, y_j)}{\Delta t}\left[\frac{t^2}{2} - t_{j-1} t\right]_{t_j}^{t_{j+1}}$$

$\vdots$ left as exercise for students

$$= -\frac{f(t_{j-1}, y_{j-1})}{2}\Delta t + \frac{3 f(t_j, y_j)}{2}\Delta t$$

So all together

$$\boxed{y(t_{j+1}) = y(t_j) + \Delta t\left[\frac{3}{2} f(t_j, y_j) - \frac{1}{2} f(t_{j-1}, y_{j-1})\right]}$$

"Adams-Bashforth" 2-step Method (LTE $O(\Delta t^2)$)

$y_0 = \alpha_0,\ y_1 = \alpha_1$

ex/ What if we used L.I.P. implicitly (t_j, t_{j+1})?

$$P(t) = f(t_{j+1}, y_{j+1}) \underbrace{L_0(t)}_{\frac{1}{\Delta t}(t-t_j)} + f(t_j, y_j) \underbrace{L_1(t)}_{-\frac{1}{\Delta t}(t-t_{j+1})}$$

$\vdots$

$$w_{j+1} = w_j + \frac{\Delta t}{12}\left[5 f(t_{j+1}, w_{j+1}) + 8 f(t_j, w_j) - f(t_{j-1}, w_{j-1})\right]$$

"Adams-Moulton" 2-step Method (LTE $O(\Delta t^4)$)

$w_0 = \alpha_0,\ w_1 = \alpha_1$ Implicit

ex Consider $y' = y - t^2 + 1,\ 0 \le t \le 2,\ y(0) = 0.5$ Use AM2 with $N=10$ to estimate solution.

$$w_2 = w_1 + \frac{\Delta t}{12}\left[5f(t_2, w_2) + 8f(t_1, w_1) - f(t_0, w_0)\right]$$

$$= w_1 + \frac{\Delta t}{12}\left(5[w_2 - t_2^2 + 1] + 8[w_1 - t_1^2 + 1] - [w_0 - t_0^2 + 1]\right)$$

$$= w_1 + \left(\frac{5\Delta t}{12}w_2 - \frac{5\Delta t}{12}t_2^2 + \frac{5\Delta t}{12} + \frac{8\Delta t}{12}w_1 - \frac{8\Delta t}{12}t_1^2 + \frac{8\Delta t}{12} - \frac{\Delta t}{12}w_0 + \frac{\Delta t}{12}t_0^2 + \frac{\Delta t}{12}\right)$$

$$w_2 = \frac{12}{7\Delta t}\left[\left(1 + \frac{2}{3}\Delta t\right)w_1 - \frac{\Delta t}{12}w_0 + \frac{\Delta t}{12} - \frac{5\Delta t}{12}t_2^2 - \frac{2\Delta t}{3}t_1^2 + \frac{\Delta t}{12}t_0^2\right]$$

* Ugh. That was a lot of work.

ex Consider $y' = y - t^2 + 1,\ 0 \le t \le 2,\ y(0) = 0.5$. Use AB 2 method to approximate solution w/ $N=10$.

$$y(t_{j+1}) = y(t_j) + \Delta t\left[\frac{3}{2}f(t_j, y_j) - \frac{1}{2}f(t_{j-1}, y_{j-1})\right]$$

$$y(t_2) = y(t_1) + \Delta t\left[\frac{3}{2}[y_1 - t_1^2 + 1] - \frac{1}{2}[y_0 - t_0^2 + 1]\right]$$

$y_0 = 0.5$

$y_1 = ?$

How can we find y_1? *

* MATLAB

Use a 1-step method from Unit 5 (Euler, RK2, RK4) to generate our initial conditions)

In particular, use a 1-step method with comparable LTE.