Instabilities

Stiff equations

$$\left.\begin{aligned} y_1' &= 998 y_1 + 1998 y_2 \\ y_2' &= -999 y_1 - 1999 y_2 \end{aligned}\right\}$$

$$\left.\begin{aligned} y_1(0) &= 1 \\ y_2(0) &= 0 \end{aligned}\right\}$$

Solution:

$$y_1 = 2e^{-x} - e^{-1000x}$$

$$y_2 = \underbrace{-e^{-x}}_{\text{Slowly evolving part}} + \underbrace{e^{-1000x}}_{\text{Fast transient: dies away quickly.}}$$

When discretizing:

Δx must be chosen to be able to follow the transient e^{-1000x}, but the relevant part of the solution evolves with e^{-x}.

Can we get around this dilemma?

Backwards or implicit Euler integration

$$\vec{y}(x+\Delta x) = \vec{y}(x) + \Delta x \, \vec{f}(x+\Delta x, \vec{y}(x+\Delta x))$$

the future value!

Example

$$y' = -Cy$$

$$y(0) = 1$$

Solution $y(x) = e^{-Cx}$.

Explicit Euler discretization

$$y(x+\Delta x) = y(x) - \Delta x\, C\, y(x)$$

$$= (1 - C\Delta x)\, y(x)$$

This expression is <u>unstable</u> if

$$\Delta x > \frac{2}{C}$$

This is so since

$$C\Delta x - 1 > 1$$

so that

$$y(x+\Delta x) = \underbrace{(1 - C\Delta x)}_{\text{absolute value} > 1.}\, y(x) \to \pm\infty$$

Implicit Euler integration:

$$y(x+\Delta x) = y(x) - \Delta x\, C\, y(x+\Delta x)$$

$$\Rightarrow \quad y(x+\Delta x) = \underbrace{\frac{y(x)}{1+\Delta x C}}$$

Always $0 < 1 + C\Delta x < 1$;

Hence, $y(x) \to 0$ for all Δx,

<u>always stable.</u>

Implicit Euler integration:

An equation to solve for each time step:

<u>Expensive!</u>

Vol Mat

Stability ≠ accuracy.

Implicit Euler integration is always stable. No guarantees, however, with respect to accuracy.

Methods for improved accuracy

① Runge-Kutta integration.

We set $h = \Delta x$

One component: $y(x_n) = y_n$.

1st order (Explicit Euler):

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

Taylor expansion:

$$y_{n+1} = y_n + hy_n' + O(h^2)$$

2nd order Runge Kutta:

$$\begin{cases} k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1) \\ y_{n+1} = y_n + k_2 + O(h^3) \end{cases}$$

So we know this works, at a

Taylor expansion.

$$y_{n+1} = y_n + h f\left(x_n + \tfrac{1}{2}h,\ y_n + \tfrac{1}{2}h f(x_n, y_n)\right)$$

$$= y_n + h\left[f(x_n, y_n) + \tfrac{1}{2}h \frac{\partial f_n}{\partial x} + \tfrac{1}{2}h \frac{\partial f_n}{\partial y} \right]$$

$$= y_n + h f_n + \tfrac{1}{2}h^2 \left[\frac{\partial f_n}{\partial x} + \frac{\partial f_n}{\partial y} \right].$$

$$= y_n + h y_n' + \tfrac{1}{2}h^2 y_n'' + \cdots$$

(2) Richardson extrapolation

Best for smooth functions.

Stoer-Bulirsch algorithm - a variant of Richardson extrapolation is the highest-accuracy method with the least effort.

3 key ideas:

① $y(x+\Delta x) = y(x+\Delta x, h)$

h: Step size used in numerical integration.

y is an analytical function in h.

Exact result for $h \to 0$.

② Extrapolation of $y(x+\Delta x, h)$ as $h \to 0$:
We need to fit $y(x+\Delta x, h)$ to some functional form and take this to the limit $h \to 0$.

The functional form used is the rational function types

$$\frac{\sum_{i=1}^{N} c_i \cdot h^i}{\sum_{j=1}^{N} d_j \cdot h^j}$$

These remain good approximations to analytical functions even if the terms in the polynomials are all of the same magnitude.

(3) The choice of method to increase the number of integration points, $\frac{\Delta x}{h}$ leads to an error which goes as h^{2M} rather than h:

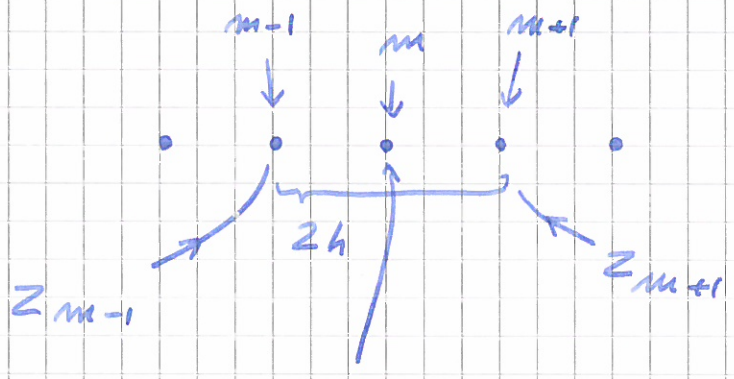
Modified midpoint method

We wish to integrate $y(x)$ from x to $x+\Delta x$ through N steps.

We set

- $z_0 = y(x)$
- $z_1 = z_0 + h f(x, z_0)$
- $z_{m+1} = z_{m-1} + 2h f(x + mh, z_m)$

$m = 1, 2, \ldots, N-1$



$f(x+mh, z_m)$

↑ midpoint evaluation

- $y(x+\Delta x) \approx y_N \equiv \frac{1}{2}(z_N + z_{N-1} + h f(x+\Delta x, z_N))$

↑ $z_N = z_{N-2} + 2h f(x+(N-1)h, z_{N-1})$

The error is $\delta y_N = y_N - y(x+\Delta x)$.

$$\delta y_N = \sum_{i=1}^{\infty} a_i h^{2i}$$

even powers!

This means that changing $h \to h/2$, the error is reduced by at least a factor 4.

In the Richardson extrapolation scheme, a good choice of N is

$$N = 2, 4, 6, 8, \ldots,$$

$$N_j = 2j.$$

Repeat modified midpoint method with N chosen from this sequence until an estimate is small enough.

Our estimate: Found through the <u>rational extrapolation scheme</u>.