Boundary Value Problems

So far: IVP $\quad \dfrac{d\vec{u}}{dt} = \vec{f}(t, \vec{u})$

$$\vec{u}(0) = \vec{u}_0 \quad \text{(initial condition)}$$

$\vec{f}$ "nice" $\rightarrow$ solution always exists and is unique

lternative: only <u>some</u> component of $\vec{u}$ are prescribed,
but they are prescribed for different $t_0, t_1, \ldots$

rises in many physical / engineering problems.

: <u>Trajectory reconstruction</u>

we observe the motion of a particle in a force field

$$\frac{d^2x}{dt^2} = \frac{1}{m} F(\cancel{t}) \quad , \quad F(t) \text{ is known}$$

observation data: $x(t_0) = x_0$, $x(t_1) = x_1$

what was the trajectory in between?

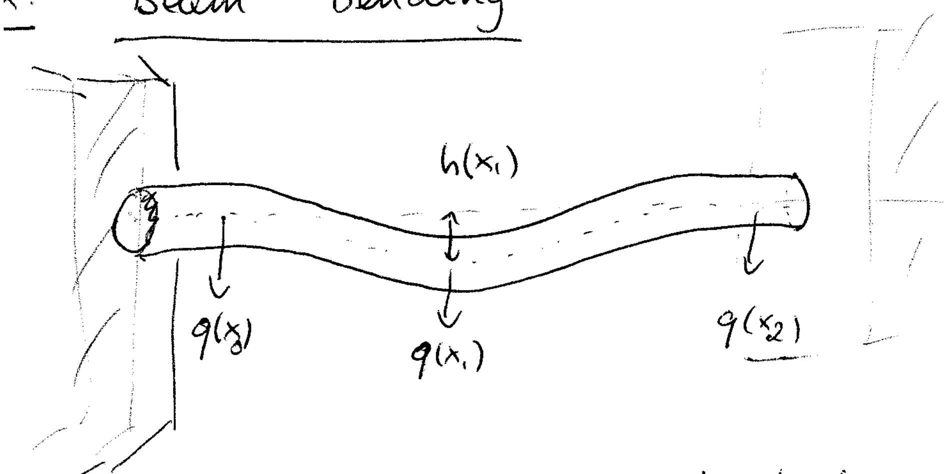
$$\boxed{\frac{d}{dt}\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} u_2 \\ F(t)/m \end{pmatrix}, \quad \vec{u}(t_1) = \begin{pmatrix} x_1 \\ ? \end{pmatrix} \quad \vec{u}(t_2) = \begin{pmatrix} x_2 \\ ? \end{pmatrix}}$$

(BVP)

Remarks: (*) Generally, no existence and uniqueness theorem like for IIPs, except in special conditions

(*) ODE integrators do not work, we don't know the initial data!

Ex: Beam Bending



onstant-cross section beam subject to position-dependent load.

Euler-Bernoulli eqn:

$$\alpha \frac{d^4 h}{dx^4} = q(x)$$

α : rigidity

$h(x)$: deflection

$q(x)$: load

eam is fixed to walls at both ends:

$$h(x_1) = 0$$
$$h'(x_1) = 0$$
$$h(x_2) = 0$$
$$h'(x_2) = 0$$

) BVP

$$\frac{d}{dx}\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} = \begin{pmatrix} u_2 \\ u_3 \\ u_4 \\ q(u_1)/a \end{pmatrix}, \quad \vec{u}(x_1) = \begin{pmatrix} 0 \\ 0 \\ ? \\ ? \end{pmatrix}, \quad \vec{u}(x_2) = \begin{pmatrix} 0 \\ 0 \\ ? \\ ? \end{pmatrix}$$

)DE Approach: Shooting method

Idea:
- guess the unknown initial conditions at t_0
- use ODE solver to obtain s/n at t_1
- refine the guess until both b.c.s are satisfied

Difficulty: need to solve

$$\vec{u}^{\,guess}(t_0) = \begin{pmatrix} u_1(0) \\ u_2^{guess} \end{pmatrix}$$

$$\rightarrow \quad \vec{u}^{\,guess}(t_1) = \begin{pmatrix} u_1^{guess}(t_1) \\ u_2^{guess}(t_1) \end{pmatrix}$$

$$\rightarrow \quad \boxed{u_1^{guess}(t_1 ; u_2^{guess}) - u_1^{desired}(t_1) = 0}$$

(nonlinear root finding problem)

Root finding is <u>much harder</u> than ODE solving!

Linear - Algebra approach, Finite - difference Method

- extends to higher dimensions
- useful for PDEs
- no non-linear root - finding necessary !

Recall finite - difference stencils for a function $f(x)$ on $[a,b]$

$$\vec{f} = \begin{pmatrix} f_1 \\ \vdots \\ f_N \end{pmatrix}, \quad \vec{f}'' = \begin{pmatrix} f_1'' \\ \vdots \\ f_n'' \end{pmatrix}$$

$$f_n = f(a+nh), \quad f_n'' = f''(a+nh), \quad h = \frac{b-a}{N+1}$$

$$\rightarrow \quad \vec{f}'' = A\vec{f} \quad (\text{with } f(a) = f(b) = 0)$$

$$A = \frac{1}{h^2}\begin{pmatrix} -2 & 1 & 0 & \cdots & & \\ 1 & -2 & 1 & 0 & \cdots & \\ 0 & 1 & -2 & 1 & 0 & \\ & & \ddots & \ddots & & \end{pmatrix}$$

$$\Rightarrow \quad \vec{f} = A^{-1}\vec{f}''.$$

in BVPs, we usually <u>know</u> f'' in terms of other functions

$\rightarrow$ only need to solve a linear system.

Ex: Beam equation on $[a,b]$

$$\frac{d^4 f}{dx^4} = \frac{1}{\alpha} q(x) \quad , \quad f(a) = f'(a) = f(b) = f'(b) = 0$$

nite-difference stencil for $\frac{d^4 f}{dx^4}$

$$f^{(4)}(x,h) = \frac{f(x-2h) - 4f(x-h) + 6f(x) - 4f(x+h) + f(x+2h)}{h^4}$$

(second-order convergence)

pply this to our discretized ~~the~~ interval: need alues for f_{-1}, f_0, f_{N+1}, f_{N+2}.

$f_0 = f_{N+1} = 0$ (boundary condition)

$f'(a) = f'(b) = 0 \Rightarrow f_{-1} = f_{N+2} = 0$ (second boundary condition)

$\Rightarrow$ finite - difference matrix

$$A = \frac{1}{h^4} \begin{pmatrix} 6 & -4 & 1 & 0 & \cdots & & & \\ -4 & 6 & -4 & 1 & 0 & \cdots & & \\ 1 & -4 & 6 & -4 & 1 & 0 & \cdots & \\ 0 & 1 & -4 & 6 & -4 & 1 & \cdots & \\ & & & & \ddots & & & \\ & & & & & -4 & 6 & -4 \\ & & & & & & -4 & 6 \end{pmatrix}$$

$\Rightarrow \quad \vec{f} = A^{-1} \cdot \vec{f}^{(4)}$, Beam - eqn : $\vec{f}^{(4)} = \frac{1}{\alpha} \vec{q}$

$q_i = q(a + n h)$

$$\Rightarrow \quad \boxed{\vec{f} = A^{-1} \left(\frac{1}{\alpha} \vec{q} \right)}$$

Richardson extrapolation

General method for improving accuracy of numerical methods.

Say we want to compute some number

$F(\Delta)$, where the exact value is obtained as $\Delta \to 0$

x) Integration with composite trapezoidal rule

$$\int_a^b f(x)\,dx \approx \underbrace{\underbrace{\frac{b-a}{N}}_{=\Delta}\left(\frac{f(a)}{2} + \sum_{k=1}^{N-1} f\left(a+k\cdot\frac{b-a}{N}\right) + \frac{f(b)}{2}\right)}_{= F(\Delta)}$$

Idea: If we <u>know</u> how $F(\Delta)$ behaves as $\Delta \to 0$,

$$F(\Delta) - F(0) = A\,\Delta^p + \mathcal{O}(\Delta^{p+1})$$

⇒ can construct a <u>new</u> rule with <u>higher</u> accuracy (i.e., ~~with~~ a higher p)

Consider

$$F(\Delta) = F(0) + A\,\Delta^p + \mathcal{O}(\Delta^{p+1})$$

$$F\left(\frac{\Delta}{2}\right) = F(0) + A\left(\frac{\Delta}{2}\right)^p + \mathcal{O}(\Delta^{p+1}) \quad // \cdot 2^p$$

$$2^p F\left(\frac{\Delta}{2}\right) - F(\Delta) = 2^p F(0) - F(0) + \underbrace{\Delta^p - \Delta^p}_{\Delta^p \text{ error term cancels out!}} + \mathcal{O}(\Delta^{p+1})$$

$$\Rightarrow \quad F(0) = \underbrace{\frac{2^p F\left(\frac{\Delta}{2}\right) - F(\Delta)}{2^p - 1}}_{} + \mathcal{O}(\Delta^{p+1})$$

$$= F^{\text{Richardson}}(\Delta)$$

$\Rightarrow$ new approximation

$$\boxed{F^{\text{Richardson}}(\Delta) = \frac{2^p F\left(\frac{\Delta}{2}\right) - F(\Delta)}{2^p - 1}}$$

has order p+1 if F has order p.

<u>Cost:</u> more evaluations ~~of f~~ $\longleftrightarrow$ higher accuracy

<u>Ex.</u> Trapezoid rule

$$F(\Delta) = \underbrace{\frac{b-a}{N}}_{=\Delta}\left(\frac{f(a)}{2} + \sum_{k=1}^{N-1} f\left(a + k\frac{b-a}{N}\right) + \frac{f(b)}{2}\right)$$

$$\Delta \to \frac{\Delta}{2} \Leftrightarrow \frac{b-a}{2N} \rightsquigarrow N \to 2N$$

so in total, $F^{\text{Richardson}}$ requires $3N$ evaluations of $f(x)$.

Advantage: requires no excessively small stepsize for given accuracy.

Richardson + Newton-Cotes = "Romberg integration"

Richardson + Midpoint Runge-Kutta = "Burlisch-Stoer method"