10.34, Numerical Methods Applied to Chemical Engineering
Professor William H. Green
Lecture #20: Boundary Value Problems Lecture 3. Finite Differences, Method of Lines, and Finite Elements.

Finite Differences

$$\hat{D}\varphi + S(\varphi) = 0 \xrightarrow{\text{mesh finite difference for } \hat{D}} \underline{\underline{A}} \cdot \underline{\phi} + \underline{S} = 0$$

$F_i(\underline{\phi}) = (\Sigma A_{ij}\phi_j) + S(\phi_i) = 0 \quad i = 1, N_{mesh}*N_{scalar\ fields}$

$J_{in} = \dfrac{\partial F_i}{\partial \phi_n}$ Solve $\underline{F}(\underline{\phi}) = \underline{0}$ by Newton-type methods

Need Jacobian $\underline{\underline{J}}$: $J_{in} = A_{in} + \delta_{in} S'(\phi_i)$.

$$\underline{\underline{J}} = \begin{pmatrix} \ddots & \ddots & & & & & \ddots & & 0 \\ \ddots & \ddots & \ddots & & & & & \ddots & \\ & \ddots & \ddots & \ddots & & & 0 & & \ddots \\ & & \ddots & \ddots & \ddots & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & \ddots & \ddots & \ddots & & \\ \ddots & & 0 & & & \ddots & \ddots & \ddots & \\ & \ddots & & & & & \ddots & \ddots & \ddots \\ 0 & & \ddots & & & & & \ddots & \ddots \end{pmatrix} + \begin{pmatrix} \ddots & 0 \\ 0 & \ddots \end{pmatrix}$$

Iterative, need good initial guess $\underline{\phi}^{guess}$!

(normally, $\underline{\underline{J}}\Delta\underline{\phi} = -\underline{F}$)

Method of Lines

e.g. 2D, $v_y = 0$, $v_{\underline{x}}$ independent of φ:

$$\underbrace{v_x(x,y)\left.\frac{\partial\varphi}{\partial x}\right|_{x,y_i} + \varphi\left.\frac{\partial v_x}{\partial x}\right|_{x,y_i}}_{\text{convection}} + D\left(\frac{\partial^2\varphi}{\partial x^2} + \boxed{\frac{\partial^2\varphi}{\partial y^2}}\right) + S(\varphi(x,y)) = 0 \quad \Big| \quad \nabla\cdot(\varphi\underline{v}) = \frac{\partial}{\partial x}(\varphi v_x) + \cancel{\frac{\partial}{\partial y}(\varphi v_y)}$$

diffusion coefficient D replace 0 if $v_y = 0$

$$\frac{\varphi(x,y_{i+1}) - 2\varphi(x,y_i) + \varphi(x,y_{i-1})}{(\Delta y)^2}$$

$u_i(x) = \varphi(x, y_i)$

$$\frac{d}{dx}\begin{pmatrix} u_1 \\ w_1 \\ u_2 \\ w_2 \\ \vdots \\ u_i \\ w_i \\ \vdots \\ u_{N_y} \\ w_{x_y} \end{pmatrix} = \begin{pmatrix} w_i \\ -\frac{v_x(x,y_i)}{D} w_i - \frac{1}{D}\frac{\partial v_x}{\partial x}\bigg|_{x,y_i} u_i - \frac{u_{i+1}}{(\Delta y)^2} - 2\frac{u_i}{(\Delta y)^2} - \frac{u_{i-1}}{(\Delta y)^2} - \frac{S(u_i)}{D} \end{pmatrix}$$

$w(x) = {}^{\partial\varphi}/_{\partial x}\big|_{(x,y_i)}$

A mess, but we can solve stiff ODEs better than PDEs. Variables: $2N_{mesh}$. Coarse discretization easy; finer mesh makes complexity rise. Better for steep front.

Diffusion/conduction dominated – Elliptic PDEs

- microfluidics, cells, reaction no convection
- no sharp fronts

$\Big\}$ Finite Differences, Method of Lines

every $\varphi(x_i, y_j)$ depends on all the others

Convection-dominated, wave equations – Hyperbolic PDEs

- information flows in a direction, shockwaves, flames
- sharp fronts ←-----→ numerical instabilities, oscillations

$\Big\}$ Stiff ODEs solve with adaptive mesh

Finite Element Method (FEM) ➔ FEMLAB

Galerkin's Method

$\phi = \sum \underbrace{c_n \chi_n(\underline{x})}$ want to retain sparsity

local basis functions

$\chi_n(\underline{x} - \underline{x}_n)$ points connected only to neighbors. Used in fluid mechanics and quantum mechanics (must consider electrostatic attraction).

1D example

$$\chi_n = \begin{cases} 1 - \dfrac{|x - x_n|}{\Delta x} & \text{if } x_{n-1} < x < x_{n+1} \\ 0 & \text{otherwise} \end{cases}$$

Add lots of x_n close to steep gradient

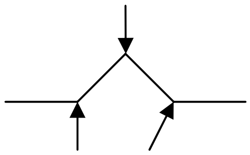

Figure 1. Add points close to steep gradient. Separate into parts at singularity.

Singularities but can integrate by parts.

$$\int \chi_n \nabla^2 \varphi = -\int \underline{\nabla} \chi_n \cdot \underline{\nabla} \varphi - \underbrace{\cancel{\int \underline{\nabla} \cdot \left(\chi_n \underline{\nabla} \varphi \right)}}_{0}$$

$$\int dx\, \chi_n(x) \left(\hat{D} \varphi(x) + S(\varphi) \right) = 0 \qquad n = 1, N_{mesh}$$

Local integral: 0 except in range $x_{n-1} < x < x_{n+1}$. Nonzero close to mesh point.

$$f_n(\phi_{n-1}, \phi_n, \phi_{n+1}) = 0$$

$$\underline{F}(\underline{\phi}) = 0$$

Method good when there are only short range forces.

This equation is difficult to write. Usually use software that is already written: e. g. MATLAB PDE Tool Box or FEMLab

$$\underline{\underline{J}} = \begin{pmatrix} \ddots & \ddots & 0 \\ \ddots & \ddots & \ddots \\ 0 & \ddots & \ddots \end{pmatrix}$$

{ easy to solve this Jacobian for finite differences, sparse matrix

Discussion

$\nabla^2 \varphi$

cylindrical: $\dfrac{1}{r} \dfrac{\partial}{\partial r} \left(r \dfrac{\partial \varphi}{\partial r} \right)$

3D $\quad N_{mesh} \sim 10^6$

$\underline{\underline{J}} \sim 10^6 \times 10^6 = 10^{12}$ (computer cannot store this)

Conjugate gradient-type solvers for $\underline{\underline{J}} \Delta \underline{\phi} = -\underline{F}$ (N steps: still 10^6)

MatLAB solver: $\boxed{\text{GMRes}}$ Works even if matrix is not positive definite

$f(\Delta \underline{\phi}) = || \underline{\underline{J}} \Delta \underline{\phi} + \underline{F} ||^2$ (minimize this within N steps)

Preconditioning

Text has a number of tricks for good preconditioning (i.e. Jacobi method see textbook)

Initial guess of ϕ is important for convergence

Finite Volumes

$\frac{\partial \varphi}{\partial t} = \hat{D}\varphi + S(\varphi)$ Instead of mesh points, we use mesh volumes.

Operator splitting Method

$$\varphi(x_i, y_i, .t) \xrightarrow[\text{t}\to\text{t}+\Delta\text{t}/2]{\text{transport}} \varphi_{new}(x_i, y_i, .\tilde{t})$$

$\frac{d}{dt}\varphi_i = s(\varphi_i)$ using N_{mesh} ODE solver: $\varphi(t_0) = \varphi_{new}$

$\varphi_{new} \xrightarrow{\text{chemistry}, \Delta t} \varphi_i(t_f) \xrightarrow{\text{transport}, (\Delta t/2)} \varphi(x_i, y_i, t + \Delta t)$ splitting error $\sim(\Delta t)^2$

$\Delta t < \Delta x/v_x$ to prevent numerical instability

Model system must have no mixing. $N_{species}$ compare to $N_{species}N_{mesh}$ for Method of Lines