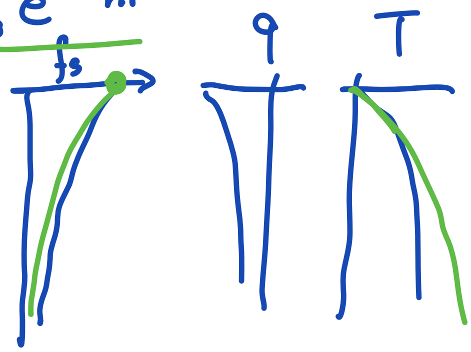
Lecture 17: Heat equation

Logistics: - HW 7 extension to Sat 11:45 pm

Last time: - steady heat equation

$$-\nabla\cdot[\kappa \nabla T] = \rho H_0 e^{-z/h_r}$$

$\Rightarrow$ Crustal geotherm



- Approx. of $f_s(x)$

$\Rightarrow$ best to average f_s over cell

Today: - transient heat equation
- Theta method
- Amplification matrix
- Decay of localized heat pulse

Energy balance equation

$$\overline{\rho c_p}\frac{\partial T}{\partial t} + \nabla\cdot[\cancel{q\rho_f c_{p,f}}\,T - \bar{\kappa}\,\nabla T] = \rho H$$

Today: $\phi = 0 \rightarrow q = 0$ $\quad \bar{\kappa} = \kappa_s = \kappa$

$\overline{\rho c_p} = \rho_s c_{p,s} = \rho c_p$

Heat equation

$$\boxed{\underbrace{\rho c_p}_{\text{new}}\frac{\partial T}{\partial t} - \nabla\cdot[\kappa\nabla T] = \rho H}$$

$\underline{u} = T(\underline{x}_e)$

$$\rho c_p \frac{\partial u}{\partial t} - \underline{\underline{D}} * [\underline{\underline{Kd}}\;\underline{\underline{G}}\;\underline{u}] = \underline{f}_s$$

$\Rightarrow$ just need to discretize time derivative

$$\frac{\partial u}{\partial t} = \frac{\underline{u}^{n+1} - \underline{u}^{n}}{\Delta t}$$

simple finite difference

$t^{n+1} = t^n + \Delta t = \cancel{t_{n+1}}$

$t^n = n\,\Delta t$

n = index for time level

substitute

$$\underline{\underline{S}}\,(\underline{u}^{n+1} - \underline{u}^{n}) + \Delta t\,\underline{\underline{L}}\,\underline{u}^{?} = \Delta t\,\underline{f}_s \qquad (\underline{\underline{L}} = -\underline{\underline{D}}\,\underline{\underline{Kd}}\,\underline{\underline{G}})$$

where $\underline{\underline{S}} = \rho c_p \underline{\underline{I}}$ $\quad \underline{\underline{I}}$ N by N

if $\rho c_p =$ const.

$$\underline{\underline{S}} = \begin{pmatrix} \ddots & & \\ & \rho c_p & \\ & & \ddots \end{pmatrix}$$

"$\underline{\underline{S}}$" storage of heat

Theta method

decide when to evaluate $\underline{\underline{L}}\,\underline{u}$!

$$\underline{u}^{\theta} = \theta\, \underline{u}^{n} + (1-\theta)\, \underline{u}^{n+1}$$

substitute:

$$\underline{\underline{S}} \left(\underline{u}^{n+1} - \underline{u}^{n} \right) + \Delta t\, \underline{\underline{L}} \left[\theta\, \underline{u}^{n} + (1-\theta)\, \underline{u}^{n+1} \right] = \Delta t\, \underline{f}_s$$

we know $\underline{u}^n$ we need to determine $\underline{u}^{n+1}$

move all known terms to r.h.s.

$$\underbrace{\left[\underline{\underline{S}} + \Delta t (1-\theta)\, \underline{\underline{L}} \right]}_{\underline{\underline{IM}}} \underline{u}^{n+1} = \Delta t\, \underline{f}_s + \underbrace{\left[\underline{\underline{S}} - \Delta t\, \theta\, \underline{\underline{L}} \right]}_{\underline{\underline{EX}}} \underline{u}^{n}$$

Linear system for a single timestep:

$$\boxed{\underline{\underline{IM}}\, \underline{u}^{n+1} = \Delta t\, \underline{f}_s + \underline{\underline{EX}}\, \underline{u}^{n}}$$

Implicit matrix : $\underline{\underline{IM}} = \underline{\underline{S}} + \Delta t\,(1-\theta)\,\underline{\underline{L}}$

Explicit matrix : $\underline{\underline{EX}} = \underline{\underline{S}} - \Delta t\,\theta\,\underline{\underline{L}}$

$\Rightarrow$ solved with solve_Lbup.m

Properties of the θ-method

For $\theta = 1$: Forward Euler Method

$$\underline{\underline{IM}} = \underline{\underline{S}} + \Delta t\,(\cancelto{\theta}{1} - 1)\,\underline{\underline{L}} = \underline{\underline{S}} \qquad \text{(diagonal)}$$

$$\Rightarrow \quad \underline{u}^{n+1} = \underline{\underline{S}}^{-1}\left(\Delta t\,\underline{f_s} + \underline{\underline{EX}}\,\underline{u}^n\right)$$

- explicit update (don't need to solve linear system)
- only matrix-vector multiply $\rightarrow$ cheap
- conditionally stable: $\Delta t \leq \dfrac{\Delta x^2}{2\kappa}$

  $\kappa = \dfrac{k}{\rho c_p}$

- first-order accurate

For $\theta = 0$: Backward Euler Method

$$\underline{\underline{EX}} = \underline{\underline{S}}$$

$$\underline{\underline{IM}}\,\underline{u}^{n+1} = \Delta t\,\underline{f_s} + \underline{\underline{EX}}\,\underline{u}^n$$

- implicit method

- solve linear system at <u>every</u> time step
- unconditionally stable
- first-order accurate

For $\theta = \frac{1}{2}$: Crank-Nicholson Method

$$\underline{\underline{LM}}\,\underline{u}^{n+1} = \Delta t\,\underline{f}_s + \underline{\underline{EX}}\,\underline{u}^{n}$$

- implicit method
- solve linear system + matrix vector mult.
- unconditionally stable
  (but has oscillation limit)
- second order accurate

Why do these methods behave this way?

Amplification Matrix

Linear system

$$\underline{\underline{IM}}\,\underline{u}^{n+1} = \underline{\underline{EX}}\,\underline{u}^{n} + \cancel{\underline{f}_s}^{\,0}$$

we know that without heat sources any T extrema will decay until T = const at thermal eqbm.

$$\underline{u}^{n+1} = \underbrace{\underline{\underline{IM}}^{-1}\underline{\underline{EX}}}_{\underline{\underline{A}} \,=\, \text{amplification matrix}}\,\underline{u}^{n}$$

$$\underline{u}^{n+1} = \underline{\underline{A}}\,\underline{u}^{n} = \underline{\underline{A}}\,(\underline{\underline{A}}\,\underline{u}^{n-1}) = \underline{\underline{A}}\,\underline{\underline{A}}\,(\underline{\underline{A}}\,\underline{u}^{n-2}) = \underline{\underline{A}}^{n}\,\underline{u}^{0}$$

where $\underline{u}^0$ is the initial condition

$$\boxed{\underline{u}^{n+1} = \underline{\underline{A}}^{n}\,\underline{u}^{0}} \qquad n \in \mathbb{N}$$

To evolve in time we just keep multiplying by $\underline{\underline{A}}$.

Compute matrix exponential using spectral decomposition

$$\underline{\underline{A}} = \underline{\underline{Q}}\,\underline{\underline{\Lambda}}\,\underline{\underline{Q}}^{-1}$$

Q = square matrix of eigenvectors

$\underline{\underline{\Lambda}}$ = diagonal matrix of eigenvalues

$$\underline{\underline{A}}\,\underline{\underline{A}} = \underline{\underline{A}}^2 = (\underline{\underline{Q}}\,\underline{\underline{\Lambda}}\,\underline{\underline{Q}}^{-1})(\underline{\underline{Q}}\,\underline{\underline{\Lambda}}\,\underline{\underline{Q}}^{-1})$$

$$= \underline{\underline{Q}}\,\underline{\underline{\Lambda}}\,\underbrace{\underline{\underline{Q}}^{-1}\underline{\underline{Q}}}_{\underline{\underline{I}}}\,\underline{\underline{\Lambda}}\,\underline{\underline{Q}}^{-1} = \underline{\underline{Q}}\,\underline{\underline{\Lambda}}\,\underline{\underline{\Lambda}}\,\underline{\underline{Q}}^{-1} = \underline{\underline{Q}}\,\underline{\underline{\Lambda}}^2\,\underline{\underline{Q}}^{-1}$$

$$\underline{\underline{A}}^n = \underline{\underline{Q}}\,\underline{\underline{\Lambda}}^n\,\underline{\underline{Q}}^{-1}$$

$$\underline{\underline{A}}\,\underline{\underline{B}} \neq \underline{\underline{B}}\,\underline{\underline{A}}$$

What happens when we multiply a vector by a matrix

$$\underline{\underline{A}}^n\,\underline{u}$$

Condition for stable time integration is that all $|\lambda_n| \leq 1$