Numerical Heat Transfer

Larry Caretto

Mechanical Engineering 375

Heat Transfer

May 9 and 14, 2007

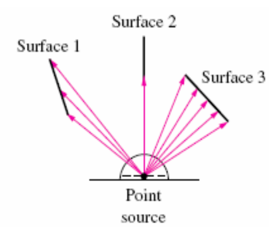
California State University Northridge

Outline

- What is numerical analysis
- Considerations of conduction, convection and radiation
- Review numerical analysis basics
  - Derivative expressions, truncation error and roundoff error
- Numerical solutions of the conduction equation in one space dimension
- Explicit versus implicit algorithms

Review View Factor, $F_{i\to j}$ or F_{ij}

- $F_{i\to j}$ or F_{ij} is the fraction of radiation, leaving surface i, that strikes surface j
  - $A_iF_{ij} = A_jF_{ji}$
  - $\Sigma_j F_{ij} = 1$
  - $F_{1\to 2+3} = F_{1\to 2} + F_{1\to 3}$



- Find view factors from charts or equations
- Basic component of radiation exchange

Figure 13-1 from Çengel, *Heat and Mass Transfer*

Review All Black Surfaces

- Heat transfer from surface 1 reaching surface 2 is $A_1F_{12}\sigma T_1^4$
- Heat transfer from surface 2 reaching surface 1 is $A_2F_{21}\sigma T_2^4 = A_1F_{12}\sigma T_2^4$
- Net heat transfer from surface 1 is $A_1F_{12}\sigma(T_1^4 - T_2^4)$
  - Negative value indicates heat into surface 1
  - For multiple surfaces

$$\dot{Q}_i = \sum_{j=1}^{N} \dot{Q}_{i\to j} = \sum_{j=1}^{N} A_i F_{ij} \sigma \left(T_i^4 - T_j^4\right)$$

Review Gray Diffuse Opaque

- Kirchoff's law applies to the average: $\alpha = \varepsilon$ at all temperatures
- For opaque surfaces $\tau = 0$ so $\alpha + \rho = 1$
- For gray, diffusive, opaque surfaces then $\rho = 1 - \alpha = 1 - \varepsilon$
- Define radiosity, $J = \varepsilon E_b + \rho G$ = emitted and reflected radiation

$$\dot{Q}_i = \frac{A_i \varepsilon_i}{1-\varepsilon_i}\left(E_{bi} - J_i\right) = \frac{E_{bi} - J_i}{R_i} \quad where \quad R_i = \frac{1-\varepsilon_i}{A_i \varepsilon_i}$$

Review Gray Diffuse Opaque II

$$\dot{Q}_i = \sum_{j=1}^{N} \dot{Q}_{ij} = \sum_{j=1}^{N} A_i F_{ij}\left(J_i - J_j\right) = \sum_{j=1}^{N} \frac{J_i - J_j}{R_{ij}} \qquad R_{ij} = \frac{1}{A_i F_{ij}}$$

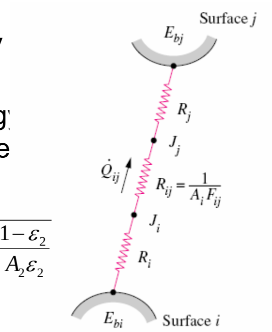
- Combining two equations for $\dot{Q}_i$

$$\sum_{j=1}^{N} \frac{J_i - J_j}{R_{ij}} = \frac{E_{bi} - J_i}{R_i} \quad \Rightarrow \quad \sum_{j=1}^{N} \frac{J_i - J_j}{R_{ij}} + \frac{J_i - E_{bi}}{R_i} = 0$$

- Solve system of N simultaneous linear equations for N values of J_i
- Black or reradiating surface ($\dot{Q}_i = 0$) has $J_i = E_{bi} = \sigma T_i^4$

Review Circuit Analogy

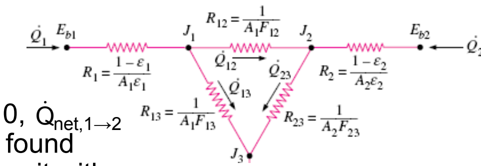
- Look at simple enclosure with only two surfaces
- Apply circuit analogy with total resistance



$$\dot{Q}_{12} = \frac{E_{b1} - E_{b2}}{R_{Total}} = \frac{E_{b1} - E_{b2}}{\frac{1-\varepsilon_1}{A_1\varepsilon_1} + \frac{1}{A_1 F_{12}} + \frac{1-\varepsilon_2}{A_2\varepsilon_2}}$$

California State University Northridge 7

Review Three-Surface Circuit



- If $\dot{Q}_3 = 0$, $\dot{Q}_{net,1\to2}$ can be found from circuit with two parallel resistances

$$\dot{Q}_{net,12} = \frac{E_{b1} - E_{b2}}{R_{Total}} = \frac{E_{b1} - E_{b2}}{R_1 + R_{\Updownarrow} + R_2} \qquad R_{\Updownarrow} = \frac{1}{\frac{1}{R_{12}} + \frac{1}{R_{13} + R_{23}}}$$

California State University Northridge 8

Review Radiation Exchange

- Two possible surface conditions: (1) known temperature, (2) known $\dot{Q}_i$

$$\dot{Q}_i = \frac{A_i\varepsilon_i}{1-\varepsilon_i}\left(E_{b_i} - J_i\right) = \sum_{j=1}^{N} A_i F_{ij}\left(J_i - J_j\right) \qquad i = 1, \ldots, N$$

(1) $$\left(1 + \frac{1-\varepsilon_i}{\varepsilon_i}\sum_{j=1, j\neq i}^{N} F_{ij}\right) J_i - \frac{1-\varepsilon_i}{\varepsilon_i}\sum_{j=1, j\neq i}^{N} F_{ij} J_j = E_{b_i} = \sigma T_i^4$$

(2) $$\left(\sum_{j=1, j\neq i}^{N} A_i F_{ij}\right) J_i - \sum_{j=1, j\neq i}^{N} A_i F_{ij} J_j = \dot{Q}_i$$

Solve this set of N simultaneous equations for N values of J_i

California State University Northridge 9

Review Radiation Exchange II

- Once all J_i values are known we can compute unknown values of T_i and $\dot{Q}_i$
  - For known T_i

$$\dot{Q}_i = \frac{A_i\varepsilon_i}{1-\varepsilon_i}\left(E_{b_i} - J_i\right) = \frac{A_i\varepsilon_i}{1-\varepsilon_i}\left(\sigma T_i^4 - J_i\right)$$

  - For known $\dot{Q}_i$

$$E_{b_i} = J_i + \frac{1-\varepsilon_i}{A_i\varepsilon_i}\dot{Q}_i \quad \Rightarrow \quad T_i = \frac{1}{\sigma}\sqrt[4]{J_i + \frac{1-\varepsilon_i}{A_i\varepsilon_i}\dot{Q}_i}$$

California State University Northridge 10

Numerical Analysis

- Differential equations give analytical solution at any point
- Usually not possible to obtain analytical solutions for real problems
- Numerical analysis provides algebraic equations for values at a set of points in region
  - Larger sets give better values of temperature and temperature gradient required for heat transfer

California State University Northridge 11

Numerical Analysis II

- Use different methods
  - Finite differences use Taylor series to get pointwise expressions for differential equations with measure of error
  - Finite element methods use integral approach to get similar results
  - Finite volume methods are midway between the two
  - Can use physical arguments to derive algebraic equations (see text)

California State University Northridge 12

Finite Difference Grids

- Subdivide region into discrete points
- Spacing between the points may be uniform or non-uniform
- Example: grid for $x_{min} \le x \le x_{max}$ with N+1 nodes numbered from zero to N
- Initial node value, $x_0 = x_{min}$
- Final grid node value, $x_N = x_{max}$
- Node spacing between $\Delta x_i = x_i$ and x_{i-1}
- Uniform spacing, $h = \Delta x_i = (x_{min} - x_{max})/N$
- N+1 nodes give N spaces

California State University Northridge 13

Finite Difference Grids II

- Non-uniform grid illustrated below

●----●--------●-----------●---~ ~---●--------●-----●

x_0 x_1 x_2 x_3 x_{N-2} x_{N-1} x_N

- Uniform grid below

●---●----●----●----●----●---~ ~---●----●----●

x_0 x_1 x_2 x_3 x_4 x_5 x_{N-2} x_{N-1} x_N

- Can have two, three or four-dimensional grids for x, y, z, and time
- Notation $T_{ijk}^n = T(x_i, y_j, z_k, t_n)$

California State University Northridge 14

Derivative Expressions

- Obtain from differentiating interpolation polynomials or from Taylor series
- Manipulate equation to get expression for derivative plus error term

$$f(x) = f(a) + \left.\frac{df}{dx}\right|_{x=a}(x-a) + \frac{1}{2!}\left.\frac{d^2f}{dx^2}\right|_{x=a}(x-a)^2 + \frac{1}{3!}\left.\frac{d^3f}{dx^3}\right|_{x=a}(x-a)^3 + \ldots$$

- Apply to $a = x_i$ and $x = x_{i+1}$

$$f(x_{i+1}) = f(x_i) + \left.\frac{df}{dx}\right|_{x=x_i}(x_{i+1}-x_i) + \frac{1}{2!}\left.\frac{d^2f}{dx^2}\right|_{x=x_i}(x_{i+1}-x_i)^2 + \frac{1}{3!}\left.\frac{d^3f}{dx^3}\right|_{x=x_i}(x_{i+1}-x_i)^3 + \ldots$$

$$\left.\frac{df}{dx}\right|_{x=x_i} = \frac{f(x_{i+1})-f(x_i)}{x_{i+1}-x_i} + \frac{1}{2!}\left.\frac{d^2f}{dx^2}\right|_{x=x_i}\frac{(x_{i+1}-x_i)^2}{x_{i+1}-x_i} + \frac{1}{3!}\left.\frac{d^3f}{dx^3}\right|_{x=x_i}\frac{(x_{i+1}-x_i)^2}{x_{i+1}-x_i} + \ldots$$

California State University Northridge 15

Derivative Expressions II

- Look at the first term in the error
- Can show that this equals error if location if derivative is unknown

$$\left.\frac{df}{dx}\right|_{x=x_i} = \frac{f(x_{i+1})-f(x_i)}{x_{i+1}-x_i} + \frac{1}{2!}\left.\frac{d^2f}{dx^2}\right|_{x=x_i}\frac{(x_{i+1}-x_i)^2}{x_{i+1}-x_i} + \frac{1}{3!}\left.\frac{d^3f}{dx^3}\right|_{x=x_i}\frac{(x_{i+1}-x_i)^2}{x_{i+1}-x_i} + \ldots$$

$$\left.\frac{df}{dx}\right|_{x=x_i} = \frac{f(x_{i+1})-f(x_i)}{x_{i+1}-x_i} + \frac{1}{2!}\left.\frac{d^2f}{dx^2}\right|_{x=\xi}\frac{(x_{i+1}-x_i)^2}{x_{i+1}-x_i}$$

- For uniform $x_{i+1} - x_i = \Delta x = h$

$$\left.\frac{df}{dx}\right|_{x=x_i} = \frac{f(x_{i+1})-f(x_i)}{h} + \frac{1}{2!}\left.\frac{d^2f}{dx^2}\right|_{x=\xi} h = \frac{f(x_{i+1})-f(x_i)}{h} + O(h)$$

California State University Northridge 16

O(h) is Order of the Error

$$\left.\frac{df}{dx}\right|_{x=x_i} = \frac{f(x_{i+1})-f(x_i)}{h} + O(h) \quad or \quad \left.\frac{df}{dx}\right|_i = f_i' = \frac{f_{i+1}-f_{i+1}}{h} + O(h)$$

- Notation $O(h^n)$ means ***truncation error*** is proportional to h^n
- Error proportional to h^n called n^{th} order
- Changing step size from h_1 to h_2 changes n^{th} order error power of n

$$\varepsilon_2 \approx \varepsilon_1\left(\frac{h_2}{h_1}\right)^n$$

California State University Northridge 17

First Derivative Expressions

First order forward
$$f_i' = \frac{f_{i+1}-f_i}{h} + O(h)$$

First order backward
$$f_i' = \frac{f_i - f_{i-1}}{h} + O(h)$$

Second order central
$$f_i' = \frac{f_{i+1}-f_{i-1}}{2h} + O(h^2)$$

Second order forward
$$f_i' = \frac{-f_{i+2} + 4f_{i+1} - 3f_i}{2h} + O\left(h^2\right)$$

Second order backwards
$$f_i' = \frac{f_{i-2} - 4f_{i-1} + 3f_i}{2h} + O\left(h^2\right)$$

California State University Northridge 18

Second Derivatives

- Second-order, central-difference, second derivative

$$f_i'' = \frac{f_{i+1} + f_{i-1} - 2f_i}{h^2} + O\left(h^2\right)$$

- Other expressions available, but this is most common

California State University Northridge 19

Find f' and f'' for sin at x = 1

Second order central

$$f_i' = \frac{f_{i+1} - f_{i-1}}{2h} + O(h^2) \qquad f_i'' = \frac{f_{i+1} + f_{i-1} - 2f_i}{h^2} + O\left(h^2\right)$$

$$f_i' = \frac{\sin(1+.1) - \sin(1-.1)}{2(.1)} \qquad f_i'' = \frac{\sin(1+.1) + \sin(1-.1) - 2\sin(1)}{(.1)^2}$$

$$f_i' = \frac{\sin(1+.01) - \sin(1-.01)}{2(.01)} \qquad f_i'' = \frac{\sin(1+.01) + \sin(1-.01) - 2\sin(1)}{(.01)^2}$$

$$f_i' = \frac{\sin(1+.001) - \sin(1-.001)}{2(.001)} \qquad f_i'' = \frac{\sin(1+.001) + \sin(1-.001) - 2\sin(1)}{(.001)^2}$$

California State University Northridge 20

Roundoff Error

- Possible in derivative expressions from subtracting close differences
- Example f(x) = e^x: f'(x) ≈ $(e^{x+h} - e^{x-h})/(2h)$ and error at x = 1 is $(e^{1+h} - e^{1-h})/(2h) - e$

$$E = \frac{3.004166 - 2.722815}{2(0.1)} - 2.718282 = 4.5x10^{-3}$$

Second order error

$$E = \frac{2.7185536702 - 2.7180100139}{2(0.0001)} - 2.718281828459 = 4.5x10^{-9}$$

$$E = \frac{2.71828210028724 - 2.71828155660388}{2(0.0000001)} - 2.718281828 = 5.9x10^{-9}$$

California State University Northridge 21

Error *vs.* Step Size Log Plots

- Previously said that $\varepsilon_2 \approx \varepsilon_1 \left(\frac{h_2}{h_1}\right)^n$

$$\log(\varepsilon_2) \approx \log(\varepsilon_1) + n\log\left(\frac{h_2}{h_1}\right) = \log(\varepsilon_1) + n[\log(h_2) - \log(h_1)]$$

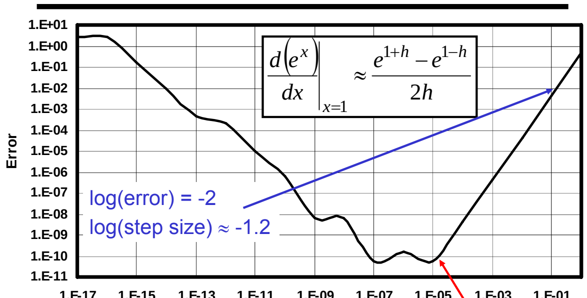
$$\log(\varepsilon_2) - \log(\varepsilon_1) \approx n[\log(h_2) - \log(h_1)]$$

$$n \approx \frac{\log(\varepsilon_2) - \log(\varepsilon_1)}{\log(h_2) - \log(h_1)}$$

- The slope of a line on a log(error) *vs.* log(step size) plot is order of the error

California State University Northridge 22

Figure 2-1. Effect of Step Size on Error



$$n \approx \frac{\log(\varepsilon_2) - \log(\varepsilon_1)}{\log(h_2) - \log(h_1)} \approx \frac{-10 - (-2)}{-5.2 - (-1.2)} = 2$$, Size

log(error) = -10

log(step size) ≈ -5.2

California State University Northridge 23

Numerical PDE Solutions

- Define a finite-difference grid in the independent variables (x, y, z, t)
- Place grid points on region boundary whose values are found from boundary conditions for the problem
- At some grid location convert differential equation into a finite difference equation
  - Observe truncation error in process
  - Neglect truncation error to get set of algebraic equations to solve

California State University Northridge 24

Unsteady Heat Transfer

- Apply difference formulas derived for ordinary derivatives to partial derivatives
- Use notation to consider different coordinate directions
- Apply to diffusion equation $\left[\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}\right]_i^n$
- Grids $x_i = x_0 + i\Delta x$ and $t_n = t_0 + n\Delta t$
- Try finite difference expressions below to get simple finite-difference equation

$$\left.\frac{\partial T}{\partial t}\right|_i^n = \frac{T_i^{n+1} - T_i^n}{\Delta t} + O(\Delta t) \quad and \quad \left.\frac{\partial^2 T}{\partial x^2}\right|_i^n = \frac{T_{i+1}^n + T_{i-1}^n - 2T_i^n}{(\Delta x)^2} + O[(\Delta x)^2]$$

Unsteady Heat Transfer II

- Substitute finite difference expressions into differential equation

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = \alpha \frac{T_{i+1}^n + T_{i-1}^n - 2T_i^n}{(\Delta x)^2} + O[\Delta t, (\Delta x)^2]$$

- Ignore truncation error, solve for T_i^{n+1}

$$T_i^{n+1} = \frac{\alpha \Delta t}{(\Delta x)^2}\left(T_{i+1}^n + T_{i-1}^n\right) + \left(1 - \frac{2\alpha\Delta t}{(\Delta x)^2}\right)T_i^n$$

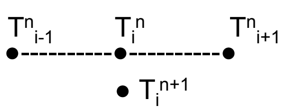
- Obtain temperature at $x = x_i$ and $t = t_{n+1}$ in terms of T values at old time step

Explicit (FTCS) Method

- Method just derived is called explicit method; can solve one equation at a time

$$T_i^{n+1} = \frac{\alpha \Delta t}{(\Delta x)^2}\left(T_{i+1}^n + T_{i-1}^n\right) + \left(1 - \frac{2\alpha\Delta t}{(\Delta x)^2}\right)T_i^n = f\left(T_{i+1}^n + T_{i-1}^n\right) + (1 - 2f)T_i^n$$

$$f \equiv \frac{\alpha \Delta t}{(\Delta x)^2}$$



- T_i^{n+1} does not depend on other T values at the new time step (n+1)

Explicit Method Example

- Pick $\alpha = 1$, $\Delta x = 0.25$, $N_x = 4$, $\Delta t = 0.01$
- $f = \alpha\Delta t/(\Delta x)^2 = 1(.01)/(.25)^2 = 0.16$
- Pick initial $T_i^0 = 1000$ and boundaries, $T^n_0 = T^n_4 = 0$ for time > 0 ($n \geq 0$)

$$Apply\ T_i^{n+1} = f\left(T_{i+1}^n + T_{i-1}^n\right) + (1 - 2f)T_i^n$$

$$T_1^1 = f\left[T_0^0 + T_2^0\right] + (1 - 2f)T_1^0 = 0.16[0 + 1000] + 0.68[1000] = 840$$
$$T_2^1 = f\left[T_1^0 + T_3^0\right] + (1 - 2f)T_2^0 = 0.16[1000 + 1000] + 0.68[1000] = 1000$$
$$T_3^1 = f\left[T_2^0 + T_4^0\right] + (1 - 2f)T_3^0 = 0.16[1000 + 0] + 0.68[1000] = 840$$

- Repeat for subsequent time steps

Explicit Method Results f = 0.16

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0.00	x = 0.25	x = 0.50	x = 0.75	x = 1.00
	t = 0	1000	1000	1000	1000	1000
n = 0	t = 0+	0	1000	1000	1000	0
n = 1	t = 0.01	0	840	1000	840	0
n = 2	t = 0.02	0	731.2	948.8	731.2	0
n = 3	t = 0.03	0	649	879.2	649	0
n = 4	t = 0.04	0	582	805.5	582	0
n = 5	t = 0.05	0	524.6	734	524.6	0
n = 6	t = 0.06	0	474.2	667	474.2	0
n = 7	t = 0.07	0	429.2	605.3	429.2	0
n = 8	t = 0.08	0	388.7	548.9	388.7	0

Explicit Method Results f = 0.16

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0.00	x = 0.25	x = 0.50	x = 0.75	x = 1.00
n = 12	t = 0.12	0	262	370.5	262	0
n = 13	t = 0.13	0	237.5	335.8	237.5	0
n = 14	t = 0.14	0	215.2	304.4	215.2	0
n = 15	t = 0.15	0	195	275.8	195	0
n = 16	t = 0.16	0	176.8	250	176.8	0
n = 17	t = 0.17	0	160.2	226.5	160.2	0
n = 18	t = 0.18	0	145.2	205.3	145.2	0
n = 19	t = 0.19	0	131.6	186.1	131.6	0
n = 20	t = 0.20	0	119.2	168.6	119.2	0
Exact	t = 0.20	0	125.1	176.9	125.1	0
Error	t = 0.20	0	5.8	8.2	5.8	0

Explicit Results f = 0.32

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0.00	x = 0.25	x = 0.50	x = 0.75	x = 1.00
	t = 0	1000	1000	1000	1000	1000
n = 0	t = 0+	0	1000	1000	1000	0
n = 1	t = 0.02	0	680	1000	680	0
n = 2	t = 0.04	0	564.8	795.2	564.8	0
n = 3	t = 0.06	0	457.9	647.7	457.9	0
n = 8	t = 0.16	0	162.2	229.4	162.2	0
n = 9	t = 0.18	0	131.8	186.4	131.8	0
n = 10	t = 0.20	0	107.1	151.4	107.1	0
Exact	t = 0.20	0	125.1	176.9	125.1	0
Error	t = 0.20	0	18	25.4	18	0

California State University Northridge

31

Explicit Results f = 0.64

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0.00	x = 0.25	x = 0.50	x = 0.75	x = 1.00
	t = 0	1000	1000	1000	1000	1000
n = 0	t = 0+	0	1000	1000	1000	0
n = 1	t = 0.04	0	360	1000	360	0
n = 2	t = 0.08	0	539.2	180.8	539.2	0
n = 3	t = 0.12	0	-35.3	639.6	-35.3	0
n = 4	t = 0.16	0	419.2	-224.2	419.2	0
n = 5	t = 0.20	0	-260.9	599.3	-260.9	0
Exact	t = 0.20	0	125.1	176.9	125.1	0
Error	t = 0.20	0	385.9	422.5	385.9	0

California State University Northridge

32

What Happened?

- We are seeing effects of instability
- Difference equations may not converge
  - Unstable equations grow without bound
  - May have stable equations that produce incorrect results
  - Conditional stability requires step size less than that needed for accuracy
  - Goal of absolute stability not always possible
  - Discussions of stability complex, can sometimes use physical arguments

California State University Northridge

33

Stability of Explicit Method

- If the values of T_{i+1} and T_{i-1} are fixed an increase in T_i^n should increase T_i^{n+1}

$$Numerical: \quad T_i^{n+1} = f\left(T_{i+1}^n + T_{i-1}^n\right) + (1-2f)T_i^n$$

- If f is greater than 0.5, an increase in T_i^n will cause a decrease in T_i^{n+1}
- We can avoid this incorrect result by keeping $f = \alpha\Delta t/(\Delta x)^2 \le 0.5$
- This imposes a time step limit that may be less than the limit required for accuracy in the solution

California State University Northridge

34

Crank-Nicholson Method

- Seek more accurate time derivative
- Provides implicit method
  - Value of T_i^{n+1} depends on other T^{n+1}
  - More work per step, but can take longer time steps with this method
  - Apply to differential equation at time n + 1/2

$$\left.\frac{\partial T}{\partial t}\right|_i^{n+\frac{1}{2}} = \frac{T_i^{n+1} - T_i^n}{2\frac{\Delta t}{2}} + O[(\Delta t)^2] = \frac{T_i^{n+1} - T_i^n}{\Delta t} + O[(\Delta t)^2] = \alpha \left.\frac{\partial^2 T}{\partial x^2}\right|_i^{n+\frac{1}{2}}$$

California State University Northridge

35

Crank-Nicholson Equation

- See derivation on slides 52-55
  - Final result shown below

$$-\frac{f}{2}T_{i-1}^{n+1} + (1+f)T_i^{n+1} - \frac{f}{2}T_{i+1}^{n+1} = \frac{f}{2}\left[T_{i+1}^n + T_{i-1}^n\right] + (1-f)T_i^n$$

$$-fT_{i-1}^{n+1} + 2(1+f)T_i^{n+1} - fT_{i+1}^{n+1} = f\left[T_{i+1}^n + T_{i-1}^n\right] + 2(1-f)T_i^n$$

- System of equations easily solved by special application of Gauss elimination called Thomas algorithm (slides 56-58)

$$-fT_{i-1}^{n+1} + 2(1+f)T_i^{n+1} - fT_{i+1}^{n+1} = R_i^n$$

California State University Northridge

36

Crank Nicholson Results

- Results for α = 1, L = 1, Δx = 0.01, Δt = 0.0005, f = $\alpha\Delta t/(\Delta x)^2$ = 5

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0	x = .01	x = .02	x = .03	x = .04
	t = 0	1000	1000	1000	1000	1000
n = 0	t = 0+	0	1000	1000	1000	1000
n = 1	t = 0.0005	0	-73.35	423.96	690.85	834.09
n = 2	t = 0.001	0	352.75	305.27	440.73	599.81
n = 3	t = 0.0015	0	25.7	320.81	439.19	533.34
n = 4	t = 0.002	0	203.86	209.57	347.52	473.02
n = 5	t = 0.0025	0	56.79	252.91	334.12	422.43
n = 6	t = 0.003	0	141.46	177.47	298.2	397.48

California State University Northridge 37

Crank Nicholson Results II

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0	x = .01	x = .02	x = .03	x = .04
n = 7	t = 0.0035	0	66.73	209.02	279.22	363.26
n = 8	t = 0.004	0	109.4	160.3	263.81	347.29
n = 9	t = 0.0045	0	68.71	179.63	245.68	324.49
n = 10	t = 0.005	0	90.79	148.2	237.92	311.75
n = 11	t = 0.0055	0	67.5	159.07	222.68	296.08
n = 12	t = 0.006	0	78.99	138.51	217.76	285.25
n = 13	t = 0.0065	0	65.08	144.07	205.56	273.92
n = 14	t = 0.007	0	70.94	130.31	201.68	264.62
n = 15	t = 0.0075	0	62.29	132.69	192.04	255.97
n = 16	t = 0.008	0	65.1	123.21	188.58	247.99
n = 17	t = 0.0085	0	59.5	123.75	180.95	241.06

California State University Northridge 38

Crank Nicholson Results III

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0	x = .01	x = .02	x = .03	x = .04
n = 18	t = 0.009	0	60.65	117	177.71	234.21
n = 19	t = 0.0095	0	56.86	116.5	171.59	228.43
n = 20	t = 0.01	0	57.1	111.53	168.52	222.53
n = 21	t = 0.0105	0	54.43	110.47	163.53	217.57
n = 22	t = 0.011	0	54.19	106.68	160.64	212.45
n = 23	t = 0.0115	0	52.22	105.35	156.49	208.11
n = 24	t = 0.012	0	51.73	102.36	153.78	203.64
n = 25	t = 0.0125	0	50.21	100.93	150.27	199.78
Exact	t = 0.0125	0	50.43	100.66	150.48	199.72
Error	t = 0.0125	0	0.216	0.272	0.212	0.061

California State University Northridge 39

Fully Implicit Method

- Discretize diffusion equation at t_{n+1}

$$\left.\frac{\partial T}{\partial t}\right|_i^{n+1} = \frac{T_i^{n+1} - T_i^n}{\Delta t} + O(\Delta t) \quad and \quad \left.\frac{\partial^2 T}{\partial x^2}\right|_i^{n+1} = \frac{T_{i+1}^{n+1} + T_{i-1}^{n+1} - 2T_i^{n+1}}{(\Delta x)^2} + O[(\Delta x)^2]$$

$$\left.\frac{\partial T}{\partial t}\right|_i^{n+1} - \alpha\left.\frac{\partial^2 T}{\partial x^2}\right|_i^{n+1} = \frac{T_i^{n+1} - T_i^n}{\Delta t} - \alpha\frac{T_{i+1}^{n+1} + T_{i-1}^{n+1} - 2T_i^{n+1}}{(\Delta x)^2} + O[(\Delta t),(\Delta x)^2] = 0$$

$$-fT_{i-1}^{n+1} + (1+2f)T_i^{n+1} - fT_{i+1}^{n+1} = T_i^n$$

- Tridiagonal system of equations
- Almost same work as CN and no spurious oscillations, but less accuracy

California State University Northridge 40

Fully Implicit Results

- Same as CN results: α = 1, L = 1, Δx = 0.01, Δt = 0.0005, f = $\alpha\Delta t/(\Delta x)^2$ = 5

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0	x = .01	x = .02	x = .03	x = .04
	t = 0	1000	1000	1000	1000	1000
n = 0	t = 0+	0	1000	1000	1000	1000
n = 1	t = 0.0005	0	358.26	588.17	735.71	830.39
n = 2	t = 0.001	0	218.22	408.43	562.69	682.35
n = 3	t = 0.0015	0	166.26	322.13	460.74	578.96
n = 4	t = 0.002	0	139.05	272.65	396.35	507.18
n = 5	t = 0.0025	0	121.84	240.25	352.17	455.26
n = 6	t = 0.003	0	109.75	217.08	319.77	415.99

California State University Northridge 41

Fully Implicit Results II

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0	x = .01	x = .02	x = .03	x = .04
n = 7	t = 0.0035	0	100.65	199.49	294.81	385.13
n = 8	t = 0.004	0	93.50	185.57	274.85	360.14
n = 9	t = 0.0045	0	87.68	174.19	258.43	339.38
n = 10	t = 0.005	0	82.82	164.67	244.62	321.81
n = 11	t = 0.0055	0	78.69	156.56	232.81	306.69
n = 12	t = 0.006	0	75.13	149.54	222.55	293.50
n = 13	t = 0.0065	0	72.00	143.38	213.53	281.87
n = 14	t = 0.007	0	69.24	137.93	205.52	271.52
n = 15	t = 0.0075	0	66.77	133.05	198.35	262.22
n = 16	t = 0.008	0	64.55	128.66	191.88	253.82
n = 17	t = 0.0085	0	62.54	124.67	186.01	246.17

California State University Northridge 42

Fully Implicit Results III

		i = 0	i = 1	i = 2	i = 3	i = 4
		x = 0	x = .01	x = .02	x = .03	x = .04
n = 18	t = 0.009	0	60.70	121.03	180.64	239.17
n = 19	t = 0.0095	0	59.02	117.70	175.71	232.74
n = 20	t = 0.01	0	57.47	114.62	171.16	226.79
n = 21	t = 0.0105	0	56.03	111.78	166.95	221.28
n = 22	t = 0.011	0	54.70	109.13	163.04	216.16
n = 23	t = 0.0115	0	53.46	106.67	159.38	211.37
n = 24	t = 0.012	0	52.30	104.36	155.96	206.88
n = 25	t = 0.0125	0	51.21	102.20	152.76	202.67
Exact	t = 0.0125	0	50.43	100.66	150.48	199.72
Error	t = 0.0125	0	0.779	1.542	2.273	2.956

California State University Northridge 43

Richardson/Leapfrog

- Use two time step central differences

$$\left.\frac{\partial T}{\partial t}\right|_i^n = \frac{T_i^{n+1} - T_i^{n-1}}{2\Delta t} + O[(\Delta t)^2] = \alpha \left.\frac{\partial^2 T}{\partial x^2}\right|_i^n = \alpha \frac{T_{i+1}^n + T_{i-1}^n - 2T_i^n}{(\Delta x)^2} + O[(\Delta x)^2]$$

- Result is explicit with second order accuracy in time

$$T_i^{n+1} = T_i^{n-1} + \frac{2\alpha\Delta t}{(\Delta x)^2}\left(T_{i+1}^n + T_{i-1}^n - 2T_i^n\right) = T_i^{n-1} + 2f\left(T_{i+1}^n + T_{i-1}^n - 2T_i^n\right)$$

- However result is unstable for any f and cannot be used

California State University Northridge 44

DuFort Frankel

- Modification of Richardson method to provide stability
- Replace $2T_i^n$ in second derivative by average at time steps n+1 and n-1
- Introduces another $O[(\Delta t)^2]$ error

$$\left.\frac{\partial T}{\partial t}\right|_i^n = \frac{T_i^{n+1} - T_i^{n-1}}{2\Delta t} + O[(\Delta t)^2] = \alpha \left.\frac{\partial^2 T}{\partial x^2}\right|_i^n = \alpha \frac{T_{i+1}^n + T_{i-1}^n - 2T_i^n}{(\Delta x)^2} + O[(\Delta x)^2]$$

$$2T_i^n = T_i^{n+1} + T_i^{n-1} + O[(\Delta t)^2]$$

$$\frac{T_i^{n+1} - T_i^{n-1}}{2\Delta t} = \alpha \frac{T_{i+1}^n + T_{i-1}^n - T_i^{n+1} - T_i^{n-1}}{(\Delta x)^2} + O\left[(\Delta x)^2, (\Delta t)^2, \frac{(\Delta t)^2}{(\Delta x)^2}\right]$$

California State University Northridge 45

DuFort Frankel

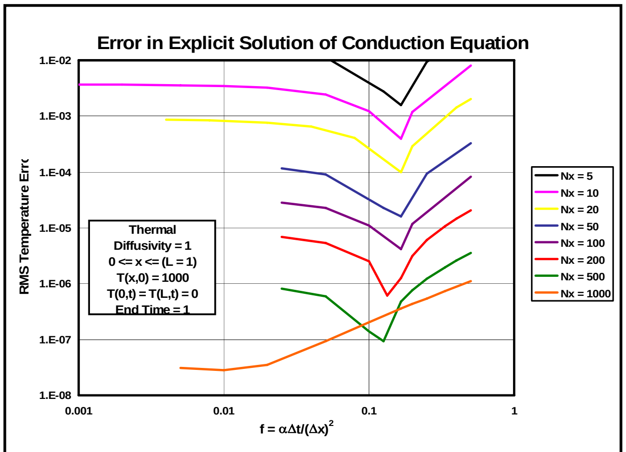
- Rearrange and introduce $f = \alpha\Delta t/(\Delta x)^2$

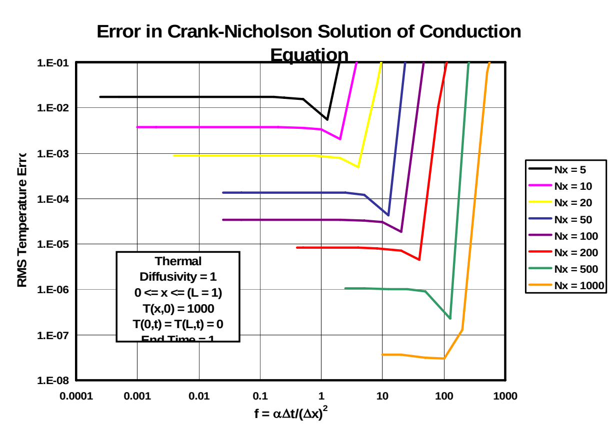
$$T_i^{n+1} - T_i^{n-1} = \frac{2\alpha\Delta t}{(\Delta x)^2}\left(T_{i+1}^n + T_{i-1}^n - T_i^{n+1} - T_i^{n-1}\right) = 2f\left(T_{i+1}^n + T_{i-1}^n - T_i^{n+1} - T_i^{n-1}\right)$$

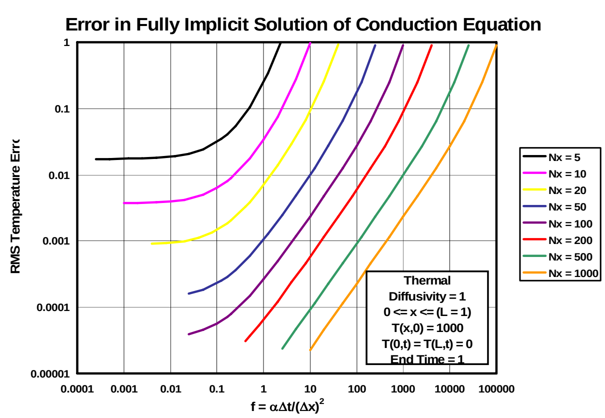
$$(1 + 2f)T_i^{n+1} = T_i^{n-1}(1 - 2f) + 2f\left(T_{i+1}^n + T_{i-1}^n\right)$$

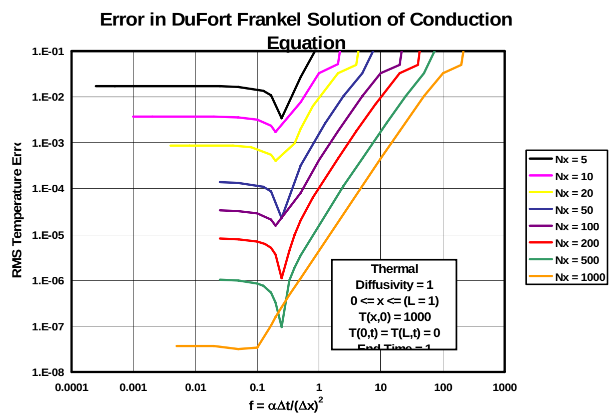
- Result is explicit for values at time n+1
- Explicit start required to get first set of values at time n-1

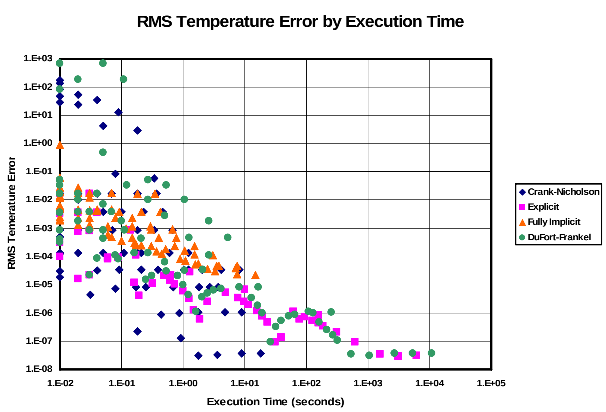
California State University Northridge 46











Derivation Details

- Slides 53 to 55 cover derivation of Crank-Nicholson method
  - Show how using midpoint of interval for difference equation gives truncation error that is O[$(\Delta x)^2$, $(\Delta t)^2$]
- Slides 56 to 58 discuss Thomas algorithm
  - Provides efficient numerical solutions for simultaneous equations of following form

$$-fT_{i-1}^{n+1}+2(1+f)T_i^{n+1}-fT_{i+1}^{n+1}=R_i^n$$

California State University Northridge

52

Space Derivative at $t_{n+1/2}$

- Take average of space derivative at time steps n and n + 1
- Show average is second order accurate

$$f_{i+1}=f_i+f_i'h+f_i''\frac{h^2}{2}+f_i'''\frac{h^3}{6}+.....$$

$$+\quad f_{i-1}=f_i-f_i'h+f_i''\frac{h^2}{2}-f_i'''\frac{h^3}{6}+.....$$

$$f_{i+1}+f_{i-1}=2f_i+2f_i''\frac{h^2}{2}+2f_i''''\frac{h^4}{24}+.....$$

$$f_i=\frac{f_{i+1}+f_{i-1}}{2}-f_i''\frac{h^2}{4}-f_i''''\frac{h^4}{48}+.....=\frac{f_{i+1}+f_{i-1}}{2}+O(h^2)$$

California State University Northridge

53

Using Space Derivative at $t_{n+1/2}$

- Apply average to space derivative

$$\left.\frac{\partial^2 T}{\partial x^2}\right|_i^{n+\frac{1}{2}}=\frac{1}{2}\left[\left.\frac{\partial^2 T}{\partial x^2}\right|_i^n+\left.\frac{\partial^2 T}{\partial x^2}\right|_i^{n+1}\right]+O[(\Delta t)^2]$$

- Substitute into unsteady equation

$$\left.\frac{\partial T}{\partial t}\right|_i^{n+\frac{1}{2}}-\alpha\left.\frac{\partial^2 T}{\partial x^2}\right|_i^{n+\frac{1}{2}}=\frac{T_i^{n+1}-T_i^n}{\Delta t}$$

$$-\frac{\alpha}{2}\left[\frac{T_{i+1}^{n+1}+T_{i-1}^{n+1}-2T_i^{n+1}}{(\Delta x)^2}+\frac{T_{i+1}^n+T_{i-1}^n-2T_i^n}{(\Delta x)^2}\right]+O[(\Delta t)^2,(\Delta x)^2]=0$$

- Introduce f = $\alpha\Delta t/(\Delta x)^2$ and rearrange

$$-\frac{f}{2}T_{i-1}^{n+1}+(1+f)T_i^{n+1}-\frac{f}{2}T_{i+1}^{n+1}=\frac{f}{2}\left[T_{i+1}^n+T_{i-1}^n\right]+(1-f)T_i^n$$

California State University Northridge

54

Crank-Nicholson Equations

$$-fT_{i-1}^{n+1} + 2(1+f)T_i^{n+1} - fT_{i+1}^{n+1} = R_i^n = f\left[T_{i+1}^n + T_{i-1}^n\right] + 2(1-f)T_i^n$$

- Rewrite equations in matrix form to show tridiagonal structure (boundary values T_0 and T_N specified)

$$\begin{bmatrix} 2(1+f) & -f & 0 & 0 & \cdots & 0 & 0 \\ -f & 2(1+f) & -f & 0 & \cdots & 0 & 0 \\ 0 & -f & 2(1+f) & -f & \cdots & 0 & 0 \\ 0 & 0 & -f & 2(1+f) & & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 2(1+f) & -f \\ 0 & 0 & 0 & 0 & \cdots & -f & 2(1+f) \end{bmatrix} \begin{bmatrix} T_1^{n+1} \\ T_2^{n+1} \\ T_3^{n+1} \\ \vdots \\ \vdots \\ T_{N-2}^{n+1} \\ T_{N-1}^{n+1} \end{bmatrix} = \begin{bmatrix} R_1^n + fT_0^n \\ R_2^n \\ R_3^n \\ \vdots \\ \vdots \\ R_{N-2}^n \\ R_{N-1}^n + fT_N^n \end{bmatrix}$$

California State University Northridge

Thomas Algorithm

- General format for tridiagonal equations

$$\begin{bmatrix} B_0 & C_0 & 0 & 0 & \cdots & 0 & 0 \\ A_1 & B_1 & C_1 & 0 & \cdots & 0 & 0 \\ 0 & A_2 & B_2 & C_2 & \cdots & 0 & 0 \\ 0 & 0 & A_3 & B_3 & & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & B_{N-1} & C_{N-1} \\ 0 & 0 & 0 & 0 & \cdots & A_N & B_N \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_{N-1} \\ x_N \end{bmatrix} = \begin{bmatrix} D_0 \\ D_1 \\ D_2 \\ \vdots \\ \vdots \\ D_{N-1} \\ D_N \end{bmatrix}$$

California State University Northridge

Thomas Algorithm II

- Gauss elimination upper triangular form

$$\begin{bmatrix} 1 & -E_0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & -E_1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & -E_2 & \cdots & 0 & 0 \\ 0 & 0 & 0 & 1 & & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -E_{N-1} \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_{N-1} \\ x_N \end{bmatrix} = \begin{bmatrix} F_0 \\ F_1 \\ F_2 \\ \vdots \\ \vdots \\ F_{N-1} \\ F_N \end{bmatrix}$$

California State University Northridge

Thomas Algorithm III

- Forward computations
  - Initial: $E_0 = -C_0 / B_0$ $F_0 = D_0 / B_0$
  - For i = 1,… N-1:

$$E_i = \frac{-C_i}{B_i + A_i E_{i-1}} \qquad F_i = \frac{D_i - A_i F_{i-1}}{B_i + A_i E_{i-1}}$$

- Get last x value first $\quad x_N = F_N = \dfrac{D_N - A_N F_{N-1}}{B_N + A_N E_{N-1}}$
- Back substitute: $x_i = F_i + E_i x_{i+1}$

California State University Northridge