2.29 Numerical Fluid Mechanics
Spring 2015 – Lecture 14

REVIEW Lecture 13:

• **Stability**: Von Neumann Ex.: 1st order linear convection/wave eqn., F-B scheme

• **Hyperbolic PDEs and Stability**
  - 2nd order wave equation and waves on a string
    - Characteristic finite-difference solution
    - Stability of C – C (CDS in time/space, explicit): \( C = \frac{c \Delta t}{\Delta x} < 1 \)
    - Example: Effective numerical wave numbers and dispersion
  - CFL condition:
    - “Numerical domain of dependence” must include “Mathematical domain of dependence”
    - Examples: 1st order linear convection/wave eqn., 2nd order wave eqn.
    - Other FD schemes (C 2nd – C 4th)
  - Von Neumann: 1st order linear convection/wave eqn., F- C: unstable
  - Stability summary: Tables of schemes for 1st order linear convection/wave eqn.

• **Elliptic PDEs**
  - FD schemes for 2D problems (Laplace, Poisson and Helmholtz eqns.)
    - Direct 2nd order and Iterative (Jacobi, Gauss-Seidel)
  - Boundary conditions

\[ C = \frac{c \Delta t}{\Delta x} < 1 \]
TODAY (Lecture 14): FINITE DIFFERENCES, Cont’d

• Elliptic PDEs, Continued
  – Examples, Higher order finite differences
  – Irregular boundaries: Dirichlet and Von Neumann BCs
  – Internal boundaries

• Parabolic PDEs and Stability
  – Explicit schemes (1D-space)
    • Von Neumann
  – Implicit schemes (1D-space): simple and Crank-Nicholson
    • Von Neumann
  – Examples
  – Extensions to 2D and 3D
    • Explicit and Implicit schemes
    • Alternating-Direction Implicit (ADI) schemes
TODAY (Lecture 14, Cont’d): FINITE VOLUME METHODS

• Integral forms of the conservation laws
• Introduction to FV Methods
• Approximations needed and basic elements of a FV scheme
  – FV grids
  – Approximation of surface integrals (leading to symbolic formulas)
  – Approximation of volume integrals (leading to symbolic formulas)
• Summary: Steps to step-up FV scheme
• Examples: One Dimensional examples
  – Generic equations
  – Linear Convection (Sommerfeld eqn.): convective fluxes
    • 2\textsuperscript{nd} order in space, 4\textsuperscript{th} order in space, links to CDS
  – Unsteady Diffusion equation: diffusive fluxes
    • Two approaches for 2\textsuperscript{nd} order in space, links to CDS
References and Reading Assignments

• Lapidus and Pinder, 1982: Numerical solutions of PDEs in Science and Engineering. Section 4.5 on “Stability”.


Elliptic PDEs
Iterative Schemes: Laplace equation \[ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \]

Finite Difference Scheme
\[ u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k = 0 \]

Liebman Iterative Scheme (Jacobi/Gauss-Seidel)
\[ u_{i,j}^{k+1} = u_{i,j}^k + r_{i,j}^k \]
\[ r_{i,j}^k = \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k}{4} \]

SOR Iterative Scheme, Jacobi
\[ u_{i,j}^{k+1} = u_{i,j}^k + \omega r_{i,j}^k \]
\[ = u_{i,j}^k + \omega \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k}{4} \]
\[ = (1 - \omega) u_{i,j}^k + \omega \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k}{4} \]

Optimal SOR (Equidistant Sampling \( h \))
\[ \omega = \frac{4}{2 + \sqrt{4 - \left[ \cos \left( \frac{\pi}{m-1} \right) + \cos \left( \frac{\pi}{m-1} \right) \right]^2}} \]
Elliptic PDE: Poisson Equation

\[ \nabla^2 u = g(x, y) \]

\[ g_{i,j} = g(x_i, y_j) \]

SOR Iterative Scheme, with Jacobi

\[ u_{i,j}^{k+1} = u_{i,j}^k + \omega r_{i,j}^k \]

\[ = u_{i,j}^k + \omega \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k - h^2 g_{i,j}}{4} \]

\[ = (1 - \omega) u_{i,j}^k + \omega \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - h^2 g_{i,j}}{4} \]
Elliptic PDE: 
Poisson Equation

\[ \nabla^2 u = g(x, y) \]

\[ g_{i,j} = g(x_i, y_j) \]

**SOR Iterative Scheme, with Gauss-Seidel**

\[
\begin{align*}
    u_{i,j}^{k+1} &= u_{i,j}^k + \omega r_{i,j}^k \\
    &= u_{i,j}^k + \omega \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k - h^2 g_{i,j}}{4} \\
    &= (1 - \omega) u_{i,j}^k + \omega \frac{u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - h^2 g_{i,j}}{4}
\end{align*}
\]
Laplace Equation

Steady Heat diffusion (with source: Poisson eqn)

\[ \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = g(x, y) \]

**BCs:**  \[ u(x, 0, t) = f(x) = 4x - 4x^2 \]

Three other BCs are null
Helmholtz Equation

\[ \nabla^2 u + f(x, y)u = g(x, y) \]

\[ f_{i,j} = f(x_i, y_j) \]

\[ g_{i,j} = g(x_i, y_j) \]

SOR Iterative Scheme, with Gauss-Seidel

\[ u_{i,j}^{k+1} = u_{i,j}^k + \omega r_{i,j}^k \]

\[ = u_{i,j}^k + \omega \left( u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - (4 - h^2 f_{i,j}) u_{i,j}^k - h^2 g_{i,j} \right) / (4 - h^2 f_{i,j}) \]

\[ = (1 - \omega) u_{i,j}^k + \omega \left( u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - h^2 g_{i,j} \right) / (4 - h^2 f_{i,j}) \]
Elliptic PDE’s
Higher Order Finite Differences

CD, 4th order (see tables in eqs. sheet)

\[
\left( \frac{\partial^2 u}{\partial x^2} \right)_{CD, 4th \ order} = \frac{-u_{i+2,j} + 16 u_{i+1,j} + 30 u_{i,j} + 16 u_{i-1,j} - u_{i-2,j}}{12h^2}
\]

The resulting 9 point “cross” stencil (not drawn) is more challenging computationally (boundary, etc) than CD 2nd order.

Use more compact scheme instead

Square stencil (see figure):
- Use Taylor series, then cancel the terms so as to get a 4th order scheme
- Leads to:

\[
\nabla^2 u_{i,j} = \frac{1}{6h^2} \left[ u_{i+1,j-1} + u_{i-1,j-1} + u_{i+1,j+1} + u_{i-1,j+1} + 4u_{i+1,j} + 4u_{i-1,j} + 4u_{i,j+1} + 4u_{i,j-1} - 20u_{i,j} \right] + O(h^4)
\]
Elliptic PDEs: Irregular Boundaries

- Many elliptic problems don’t have simple boundaries/geometries
- One way to handle them is through “irregular” discrete boundary cells (e.g. shaved cells)

1) Boundary Stencils (with Dirichlet BCs)

\[
\left( \frac{\partial u}{\partial x} \right)_{i-1,i} \approx \frac{u_{i,j} - u_{i-1,j}}{\alpha_1 \Delta x}
\]

\[
\left( \frac{\partial u}{\partial x} \right)_{i,i+1} \approx \frac{u_{i+1,j} - u_{i,j}}{\alpha_2 \Delta x}
\]

\[1^{st} \text{ derivatives evaluated at center of edges, hence } \Delta x \text{ is sum of half edge lengths on each side}\]

\[
\frac{\partial^2 u}{\partial x^2} = \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) = \frac{\partial u_{i,i+1}}{\partial x} - \frac{\partial u_{i-1,i}}{\partial x} \frac{\alpha_1 \Delta x + \alpha_2 \Delta x}{2}
\]

\[
\frac{\partial^2 u}{\partial x^2} = \frac{2}{\Delta x^2} \left[ \frac{u_{i-1,j} - u_{i,j}}{\alpha_1 (\alpha_1 + \alpha_2)} + \frac{u_{i+1,j} - u_{i,j}}{\alpha_2 (\alpha_1 + \alpha_2)} \right]
\]

\[
\frac{\partial^2 u}{\partial y^2} = \frac{2}{\Delta y^2} \left[ \frac{u_{i,j-1} - u_{i,j}}{\beta_1 (\beta_1 + \beta_2)} + \frac{u_{i,j+1} - u_{i,j}}{\beta_2 (\beta_1 + \beta_2)} \right]
\]

Can be used directly with Dirichlet BCs

- Leads to direct and iterative elliptic solvers as before, but with updated coefficients for the boundary stencils
- Other options possible: curved boundary elements
2) Neumann Boundary Conditions
(e.g. normal derivative given)

\[
\frac{\partial u}{\partial \eta} = \frac{u_1 - u_7}{L_{17}}
\]

\[
L_{78} = \Delta x \tan \theta
\]

\[
L_{17} = \Delta x / \cos \theta
\]

Linear interpolation at point 7:
\[
u_7 = u_8 + (u_6 - u_8) \frac{\Delta x \tan \theta}{\Delta y}
\]

This approach is given in Chapra & Canale.
One may instead estimate \( u_3 \) from neighbor nodes, then take the derivative along 1-3.
Elliptic PDEs
Internal (Fixed) Boundaries

Velocity and Stress Continuity (heat flux or viscous stress)

\[ u^+ = u^- \]

\[ \mu^+ \frac{\partial u^+}{\partial y} = \mu^- \frac{\partial u^-}{\partial y} \]

Derivative Finite Differences (1st order)

\[ \mu^+ \frac{\partial u^+}{\partial y} = \mu^+ \left( \frac{u_{i,j+1} - u_{i,j}}{h} \right) \]

\[ \mu^- \frac{\partial u^-}{\partial y} = \mu^- \left( \frac{u_{i,j} - u_{i,j-1}}{h} \right) \]

Finite Difference Equation at bnd.

\[ (\mu^- + \mu^+)u_{i,j} = \mu^+ u_{i,j+1} + \mu^- u_{i,j-1} \]

SOR Finite Difference Scheme at bnd.

\[ u_{i,j}^{k+1} = (1 - \omega)u_{i,j}^k + \omega \frac{\mu^+ u_{i,j+1}^k + \mu^- u_{i,j-1}^k}{\mu^- + \mu^+} \]

FIGURE 29.13
A heated plate with unequal grid spacing, two materials, and mixed boundary conditions.

© McGraw-Hill. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/fairuse.
Elliptic PDEs

**Internal (Fixed) Boundaries – Higher Order**

**Velocity and Stress Continuity**

\[ u^+ = u^- \]

\[ \mu^+ \frac{\partial u^+}{\partial y} = \mu^- \frac{\partial u^-}{\partial y} \]

**Taylor Series, inserting the PDE**

\[
\begin{align*}
    u_{i,j-1} & \approx u_{i,j} - hu_y(x_i,y_j) + \frac{h^2}{2} u_{yy}(x_i,y_j) \\
    & = u_{i,j} - hu_y(x_i,y_j) + \frac{h^2}{2} (g^-_{i,j} - u_{xx}(x_i,y_j)) \\
    u_{i,j+1} & \approx u_{i,j} + hu_y(x_i,y_j) + \frac{h^2}{2} u_{yy}(x_i,y_j) \\
    & = u_{i,j} + hu_y(x_i,y_j) + \frac{h^2}{2} (g^+_{i,j} - u_{xx}(x_i,y_j))
\end{align*}
\]

**Derivative Finite Differences (2nd order)**

\[
\begin{align*}
    \mu^+ \frac{\partial u^+}{\partial y} & = \mu^+ \left[ \frac{u_{i,j+1} - u_{i,j}}{h} + \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{2h} - \frac{h}{2} g^+_{i,j} \right] \\
    \mu^- \frac{\partial u^-}{\partial y} & = \mu^- \left[ \frac{u_{i,j} - u_{i,j-1}}{h} - \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{2h} + \frac{h}{2} g^-_{i,j} \right]
\end{align*}
\]

**Finite Difference Equation at bnd.**

\[
\left[ \frac{2(\mu^+ u_{i,j+1} + \mu^- u_{i,j-1})/(\mu^- + \mu^+)}{4} + u_{i+1,j} + u_{i-1,j} - 4u_{i,j} - h^2 \vec{g}_{i,j} \right] = 0
\]

**SOR Finite Difference Scheme at bnd.**

\[ u^{k+1}_i = (1-\omega)u^{k}_i + \omega \left[ \frac{2(\mu^+ u^{k}_{i,j+1} + \mu^- u^{k}_{i,j-1})/(\mu^- + \mu^+)}{4} + u^{k+1}_{i+1,j} + u^{k+1}_{i-1,j} - h^2 \vec{g}^{k}_{i,j} \right] \]

\[
\mu^\pm \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) = f(x,y), \quad \frac{f(x,y)}{\mu^\pm} = g^\pm(x,y)
\]
Partial Differential Equations

Parabolic PDE: \( B^2 - 4AC = 0 \)

Heat conduction equation, forced or not (dominant in 1D)

\[
\frac{\partial T}{\partial t} = \frac{\kappa}{\rho c} \nabla^2 T + f, \quad (\alpha = \frac{\kappa}{\rho c})
\]

Unsteady, diffusive, small amplitude flows or perturbations (e.g. Stokes Flow)

Examples

- Usually smooth solutions ("diffusion effect" present)
- "Propagation" problems
- Domain of dependence of solution is domain \( D \) (\( x, y \), and \( 0 < t < \infty \)):

Finite Differences/Volumes, Finite Elements

\[
\frac{\partial u}{\partial t} = \nu \nabla^2 u + g
\]

BC 1:
\[ T(0,0,t) = f_1(t) \]

BC 2:
\[ T(L_x,L_y,t) = f_2(t) \]

IC:
\[ T(x,y,0) = F(x,y) \]

(from Lecture 9)
Partial Differential Equations
Parabolic PDE: 1D Heat Conduction example

Heat Conduction Equation

\[ T_t(x,t) = \alpha T_{xx}(x,t), \quad 0 < x < L, \quad 0 < t < \infty \]

Initial Condition

\[ T(x,0) = f(x), \quad 0 \leq x \leq L \]

Boundary Conditions

\[ T(0,t) = g_1(t), \quad 0 < t < \infty \]
\[ T(L,t) = g_2(t), \quad 0 < t < \infty \]
Parabolic PDE

1D Heat Conduction: Forward in time, centered in space, explicit

Equidistant Sampling

\[ h = \frac{L}{n} \]
\[ k = \frac{T}{m} \]

Discretization

\[ x_i = (i - 1)h, \ i = 2, \ldots, n - 1 \]
\[ t_j = (j - 1)k, \ j = 1, \ldots, m \]

Forward (Euler) Finite Difference in time

\[ T_i(x,t) = \frac{T(x_i, t_{j+1}) - T(x_i, t_j)}{k} + O(k) \]

Centered Finite Difference in space

\[ T_{xx}(x,t) = \frac{T(x_{i-1}, t_j) - 2T(x_i, t_j) + T(x_{i+1}, t_j)}{h^2} + O(h^2) \]

\[ T_{i,j} = T(x_i, t_j) \]

Finite Difference Equation

\[ \frac{T_{i,j+1} - T_{i,j}}{k} = \alpha \frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{h^2} \]
Parabolic PDE
1D Heat Conduction: Forward in time, centered in space, explicit

Dimensionless diffusion coefficient

\[ r = \frac{\alpha k}{h^2} \]

Explicit Finite Difference Scheme

\[ T_{i,j+1} = (1 - 2r)T_{i,j} + r(T_{i-1,j} + T_{i+1,j}) \]

Stability Requirement

\[ r \leq 0.5 \]

Conditionally stable (von Neumann)
Shown in class on blackboard
Heat Conduction Equation

Explicit Finite Differences

(1D-in-space, unsteady case; similar to steady elliptic problem seen previously)

\[ u_t(x,t) = u_{xx}(x,t), \quad 0 < x < 1, \quad 0 < t < 0.2 \]

**ICs:** \( u(x,0) = f(x) = 4x - 4x^2 \)

**BCs:**
\[ u(0,t) = g_1(t) \equiv 0 \]
\[ u(1,t) = g_2(t) \equiv 0 \]

**Forward Euler - \( r = 0.5 \)**

\( T \) denoted by \( u \), i.e. \( u_{i,j} \equiv T_{i,j} \)
\( \alpha \) denoted by \( c^2 \), i.e. \( c^2 \equiv \alpha \)

heat_fw.m

```matlab
L=1; T=0.2; c=1;
N=5; h=L/N;
M=10; k=T/M;
r=c^2*k/h^2

x=[0:h:L]';
t=[0:k:T];
fx='4*x-4*x.^2';
g1x='0';
g2x='0';
f=inline(fx,'x');
g1=inline(g1x,'t');
g2=inline(g2x,'t');
n=length(x);
m=length(t);
u=zeros(n,m);
u(2:n-1,1)=f(x(2:n-1));
u(1,1:m)=g1(t);
u(n,1:m)=g2(t);
for j=1:m-1
    for i=2:n-1
        u(i,j+1)=(1-2*r)*u(i,j) + r*(u(i+1,j)+u(i-1,j));
    end
end

figure(4)
mesh(t,x,u);
a=ylabel('x');
set(a,'Fontsize',14);
a=xlabel('t');
set(a,'Fontsize',14);
a=title(['Forward Euler - r = ' num2str(r)]);
set(a,'Fontsize',16);
```
2.29

Heat Conduction Equation

Explicit Finite Differences

\[ u_t(x, t) = u_{xx}(x, t), \quad 0 < x < 1, \quad 0 < t < 0.33 \]

**ICs:** \[ u(x, 0) = f(x) = 4x - 4x^2 \]

**BCs:**
\[ u(0, t) = g_1(t) \equiv 0 \]
\[ u(1, t) = g_2(t) \equiv 0 \]

**Forward Euler - \( r = 0.8325 \)**

```matlab
L=1; T=0.333; c=1; 
N=5; h=L/N; 
M=10; k=T/M; 
r=c^2*k/h^2 

x=[0:h:L]'; 
t=[0:k:T]; 
fx='4*x-4*x.^2'; 
g1x='0'; 
g2x='0'; 
f=inline(fx,'x'); 
g1=inline(g1x,'t'); 
g2=inline(g2x,'t'); 
n=length(x); 
m=length(t); 
u=zeros(n,m); 
u(2:n-1,1)=f(x(2:n-1)); 
u(1,1:m)=g1(t); 
u(n,1:m)=g2(t); 
for j=1:m-1 
    for i=2:n-1 
        u(i,j+1)=(1-2*r)*u(i,j) + r*(u(i+1,j)+u(i-1,j)); 
    end 
end 
figure(4) 
mesh(t,x,u); 
a=ylabel('x'); 
set(a,'Fontsize',14); 
a=xlabel('t'); 
set(a,'Fontsize',14); 
a=title(['Forward Euler - r = ' num2str(r)]); 
set(a,'Fontsize',16); 
```

heat fw 2.m
Parabolic PDE: **Implicit Schemes**

Leads to a system of equations to be solved at each time-step

B-C (Backward-Centered):
- 1\textsuperscript{st} order accurate in time,
- 2\textsuperscript{nd} order in space
- Unconditionally stable

Crank-Nicolson:
- 2\textsuperscript{nd} order accurate in time,
- 2\textsuperscript{nd} order in space
- Unconditionally stable

**Simple implicit method**

\[ t^l \quad t^{l+1} \]
\[ x_{i-1} \quad x_i \quad x_{i+1} \]

- Backward in time
- Centered in space
- Evaluates RHS at time \( t+1 \) instead of time \( t \) (for the explicit scheme)

**Crank-Nicolson method**

\[ t^l \quad t^{l+1/2} \quad t^{l+1} \]
\[ x_{i-1} \quad x_i \quad x_{i+1} \]

Time: centered FD, but evaluated at mid-point

2\textsuperscript{nd} derivative in space determined at mid-point by averaging at \( t \) and \( t+1 \)

Parabolic PDE: Implicit Schemes

Crank-Nicolson Scheme

Equidistant Sampling

\[ h = \frac{L}{n} \]
\[ k = \frac{T}{m} \]

Discretization

\[ x_i = (i - 1)h, \ i = 2, \ldots, n - 1 \]
\[ t_j = (j - 1)k, \ j = 1, \ldots, m \]

Mid-point Finite Difference in time

\[ u_t(x, t + \frac{k}{2}) = \frac{u(x, t + k) - u(x, t)}{k} + O(k^2) \]

Mid-point Finite Difference in time

\[ u_{i,j+1} - u_{i,j} = \frac{k}{2} \left( u_{i-1,j+1} - 2u_{i,j+1} + u_{i+1,j+1} \right) \]
\[ + \frac{2h^2}{2} \left( u_{i-1,j} - 2u_{i,j} + u_{i+1,j} \right) \]

Crank-Nicholson Implicit Scheme

Unconditionally stable (by Von Neumann)

\[ r = \frac{c^2k}{h^2} \]

\[ -ru_{i-1,j+1} + (2 + 2r)u_{i,j+1} - ru_{i+1,j+1} = (2 - 2r)u_{i,j} + r(u_{i-1,j} + u_{i+1,j}) \]
Parabolic PDEs: Implicit Schemes

Crank-Nicolson – special case of $r = 1$

\[ r = \frac{c^2 k}{h^2} = 1 \]

\[ k = \frac{h^2}{c^2} \]

\[-u_{i-1,j+1} + 4u_{i,j+1} - u_{i+1,j+1} = u_{i-1,j} + u_{i+1,j} \]

\[
\begin{bmatrix}
4 & -1 \\
-1 & 4 & -1 \\
& \ddots & \ddots \\
-1 & 4 & -1 \\
0 & -1 & 4 & -1 \\
& & & -1 & 4
\end{bmatrix}
\begin{bmatrix}
u_{2,j+1} \\
u_{3,j+1} \\
\vdots \\
u_{i,j+1} \\
u_{n-2,j+1} \\
u_{n-1,j+1}
\end{bmatrix}
= \begin{bmatrix}
g_{1,j} + u_{2,j} + g_{1,j+1} \\
u_{2,j} + u_{4,j} \\
\vdots \\
u_{i-1,j} + u_{i+1,j} \\
u_{n-2,j} + u_{n-1,j} \\
u_{n-2,j} + g_{n,j} + g_{n,j+1}
\end{bmatrix} \]
Heat Flow Equation
Implicit Crank-Nicolson Scheme

\[ u_t(x,t) = u_{xx}(x,t) \quad 0 < x < 1, \quad 0 < t < 0.0 \]

**ICs:** \[ u(x, 0) = f(x) = 4x - 4x^2 \]

**BCs:**
\[ u(0, t) = g_1(t) = 0 \]
\[ u(1, t) = g_2(t) = 0 \]

Crank-Nicholson - \( r = 0.8325 \)

```matlab
heat_cn.m
```

L=1; T=0.333; c=1;
N=5; h=L/N;
M=10;
k=T/M;
r=c^2*k/h^2

x=[0:h:L]';
t=[0:k:T];
fx='4*x-4*x.^2';
g1x='0';
g2x='0';
f=inline(fx,'x');
g1=inline(g1x,'t');
g2=inline(g2x,'t');
n=length(x); m=length(t);
u=zeros(n,m);
u(2:n-1,1)=f(x(2:n-1));
u(1,1:m)=g1(t);
u(n,1:m)=g2(t);

% set up Crank-Nicholson coef matrix
\( d = (2+2*r) \times \text{ones(n-2,1)}; \)
\( b = -r \times \text{ones(n-2,1)}; \)
c=b;

% LU factorization
[alf,bet]=lu_tri(d,b,c);
for j=1:m-1
    rhs=r*(u(1:n-2,j)+u(3:n,j)) + (2-2*r)*u(2:n-1,j);
    rhs(1) = rhs(1)+r*u(1,j+1);
    rhs(n-2)=rhs(n-2)+r*u(n,j+1);
end
```

```matlab
% Forward substitution
z=forw_tri(rhs,bet);
% Back substitution
y_b=back_tri(z,alf,c);
for i=2:n-1
    u(i,j+1)=y_b(i-1);
end
end
```
Heat Flow Equation
Implicit Crank-Nicolson Scheme

L=1; T=0.1; c=1;
N=10; h=L/N;
M=10;
k=T/M;
r=c^2*k/h^2

x=[0:h:L]';
t=[0:k:T];
fx='sin(pi*x)+sin(3*pi*x)';
g1x='0';
g2x='0';
f=inline(fx,'x');
g1=inline(g1x,'t');
g2=inline(g2x,'t');
n=length(x); m=length(t); u=zeros(n,m);
u(2:n-1,1)=f(x(2:n-1));
u(1,1:m)=g1(t);
u(n,1:m)=g2(t);

% set up Crank-Nicholson coef matrix
d=(2+2*r)*ones(n-2,1);
b=-r*ones(n-2,1);
c=b;

% LU factorization
[alf,bet]=lu_tri(d,b,c);

for j=1:m-1
    rhs=r*(u(1:n-2,j)+u(3:n,j))+(2-2*r)*u(2:n-1,j);
    rhs(1)=rhs(1)+r*u(1,j+1);
    rhs(n-2)=rhs(n-2)+r*u(n,j+1);
end

% Forward substitution
z=forw_tri(rhs,bet);

% Back substitution
y_b=back_tri(z,alf,c);
for i=2:n-1
    u(i,j+1)=y_b(i-1);
end

Initial Condition
\[ f(x) = \sin \pi x + \sin 3\pi x \]

Analytical Solution
\[ u(x, t) = e^{-\pi^2 t} \sin \pi x + e^{-9\pi^2 t} \sin 3\pi x \]
Parabolic PDEs: Two spatial dimensions

• Example: Heat conduction equation/unsteady diffusive (e.g. negligible flow, no convection)

\[
\frac{\partial T}{\partial t} = c^2 \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right), \quad (0 \leq t < \infty, 0 < x < L_x, 0 < y < L_y)
\]

• Standard explicit and implicit schemes \((t = n\Delta t, \ x = i\Delta x, \ y = j\Delta y)\)

  – Explicit: \(\frac{T_{i,j}^{n+1} - T_{i,j}^n}{\Delta t} = c^2 \frac{T_{i-1,j}^n - 2T_{i,j}^n + T_{i+1,j}^n}{\Delta x^2} + c^2 \frac{T_{i,j-1}^n - 2T_{i,j}^n + T_{i,j+1}^n}{\Delta y^2} \) \(\left( O(\Delta t), \ O(\Delta x^2 + \Delta y^2) \right)\)

    • Stringent stability criterion:

      \[
      \Delta t \leq \frac{1}{8} \frac{\Delta x^2 + \Delta y^2}{c^2}
      \]

      For uniform grid: \(r = \frac{\Delta t \ c^2}{\Delta x^2} \leq \frac{1}{4}\)

  – Implicit: \(\frac{T_{i,j}^{n+1} - T_{i,j}^n}{\Delta t} = c^2 \frac{T_{i-1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i+1,j}^{n+1}}{\Delta x^2} + c^2 \frac{T_{i,j-1}^{n+1} - 2T_{i,j}^{n+1} + T_{i,j+1}^{n+1}}{\Delta y^2} \) \(\left( O(\Delta t^2), \ O(\Delta x^2) \right)\)

  – Crank-Nicolson Implicit (for \(\Delta x = \Delta y\)) \(\left( O(\Delta t^2), \ O(\Delta x^2) \right)\)

\[
(1 + 2r)T_{i,j}^{n+1} - (1 - 2r)T_{i,j}^n = r \left( T_{i-1,j}^{n+1} + T_{i+1,j}^{n+1} + T_{i,j+1}^{n+1} + T_{i,j-1}^{n+1} \right) + \frac{r}{2} \left( T_{i-1,j}^n + T_{i+1,j}^n + T_{i,j+1}^n + T_{i,j-1}^n \right)
\]

• Centered in time over \(\Delta t = \text{Sum explicit and implicit RHSs (given above), divided by two}\)
Parabolic PDEs: Two spatial dimensions

• Crank-Nicolson Implicit (for $\Delta x=\Delta y$):

$$(1+2r)T_{i,j}^{n+1} - (1-2r)T_{i,j}^n = \frac{r}{2}(T_{i-1,j}^{n+1} + T_{i+1,j}^{n+1} + T_{i,j+1}^{n+1} + T_{i,j-1}^{n+1}) + \frac{r}{2}(T_{i-1,j}^n + T_{i+1,j}^n + T_{i,j+1}^n + T_{i,j-1}^n)$$

• Five unknowns at the (n+1) time => penta-diagonal

• Either elimination procedure or iterative scheme (Jacobi/Gauss-Seidel/SOR)

• but not always efficient

• Alternating-Direction Implicit (ADI) schemes

– Provides a mean for solving parabolic PDEs with tri-diagonal matrices

– In 2D: each time increment is executed in two half steps: each step is conditionally stable, but “combination of two half-steps” is unconditionally stable (similar to Crank-Nicolson behavior)

– It is one but a group of schemes called “splitting methods”

– Extended to 3D (time increment divided in 3): varied stability properties
Parabolic PDEs: Two spatial dimensions

**ADI scheme (Two Half steps in time)**

1) From time $n$ to $n+1/2$: Approximation of 2\textsuperscript{nd} order $x$ derivative is explicit, while the $y$ derivative is implicit. Hence, tri-diagonal matrix to be solved:

$$
\frac{T_{i,j}^{n+1/2} - T_{i,j}^n}{\Delta t / 2} = c^2 \frac{T_{i-1,j}^n - 2T_{i,j}^n + T_{i+1,j}^n}{\Delta x^2} + c^2 \frac{T_{i,j-1}^{n+1/2} - 2T_{i,j}^{n+1/2} + T_{i,j+1}^{n+1/2}}{\Delta y^2}
$$

(\(O(\Delta x^2 + \Delta y^2)\))

2) From time $n+1/2$ to $n+1$: Approximation of 2\textsuperscript{nd} order $x$ derivative is implicit, while the $y$ derivative is explicit. Another tri-diagonal matrix to be solved:

$$
\frac{T_{i,j}^{n+1} - T_{i,j}^{n+1/2}}{\Delta t / 2} = c^2 \frac{T_{i-1,j}^{n+1} - 2T_{i,j}^{n+1} + T_{i+1,j}^{n+1}}{\Delta x^2} + c^2 \frac{T_{i,j-1}^{n+1/2} - 2T_{i,j}^{n+1/2} + T_{i,j+1}^{n+1/2}}{\Delta y^2}
$$

(\(O(\Delta x^2 + \Delta y^2)\))
For $\Delta x = \Delta y$:

1) From time $n$ to $n+1/2$:

$$-rT_{i,j-1}^{n+1/2} + 2(1+r)T_{i,j}^{n+1/2} - rT_{i,j+1}^{n+1/2} = rT_{i-1,j}^n + 2(1-r)T_{i,j}^n + rT_{i+1,j}^n$$

2) From time $n+1/2$ to $n+1$:

$$-rT_{i-1,j}^{n+1} + 2(1+r)T_{i,j}^{n+1} - rT_{i+1,j}^{n+1} = rT_{i,j-1}^{n+1/2} + 2(1-r)T_{i,j}^{n+1/2} + rT_{i,j+1}^{n+1/2}$$