

GES 554  
Lessons 37-40

# Overview

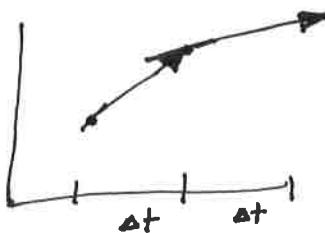
- Numerical Methods

Taylor's series (truncated)  $U(x+h) = U(x) + U'(x)h + \frac{U''(x)h^2}{2} + \dots$

$$U_x(j) = \frac{U_{j+1} - U_{j-1}}{2h}$$

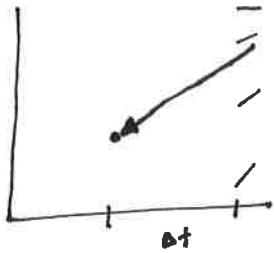
$$U_{xx}(j) = \frac{U_{j+1} + U_{j-1} - 2U_j}{h^2}$$

- ~~Implicit~~ Explicit Methods time-space



$$y^{n+1} = y^n + \frac{dy^n}{dx}$$

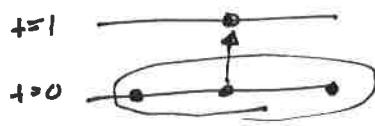
- Implicit Methods



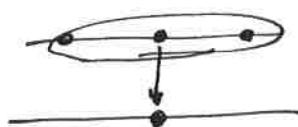
i) find new point such that derivative points to old point

$$y^{n+1} = y^n + \frac{dy^{n+1}}{dx}$$

- $\nabla^2 U = 0$



Explicit



Implicit

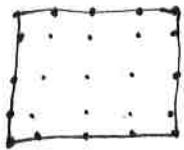


Crank-Nicolson

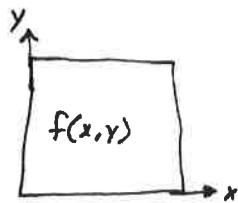
Stencil

# Numerical Solutions

Solve PDEs with discrete points



Simple approximation to



Taylor series

$$f(x+h)$$

$$f(x+h) = f(x) + \frac{df(x)}{dx} h + \frac{1}{2!} \frac{d^2 f(x)}{dx^2} h^2 + \dots + \frac{1}{n!} \frac{d^n f(x)}{dx^n} h^n$$

- Truncate at 2 terms

$$f(x+h) = f(x) + f'(x)h + O(h^2) + O(h^n)$$

$\nwarrow$  Error term

- Solve for  $f'$

$$f'(x) \approx \frac{f(x+h) - f(x)}{h} + O(h^2)$$

truncation  
The error is proportional to  $h$

Central difference

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$$

We gain one order for errors.  
 $O(h^2)$

why? take difference of  $f(x+h)$  and  $f(x-h)$   
Taylor Series; even order errors cancel.

Can you extend the stencil beyond 2 pts? Yes.

$$\begin{aligned} h f'(x) &\approx -\frac{1}{280} f(x+4h) + \frac{4}{105} f(x+3h) + -\frac{1}{5} f(x+2h) + \frac{4}{5} f(x+h) \\ &\quad - \frac{4}{5} f(x-h) + \frac{1}{5} f(x-2h) - \frac{4}{105} f(x-3h) + \frac{1}{280} f(x-4h) \\ &\quad + O(h^8) \end{aligned}$$

## 2<sup>nd</sup> Derivative

$$\frac{d^2f}{dx^2} \approx \frac{1}{h^2} \left( u(x+h) - 2u(x) + u(x-h) \right) \quad O(h^2)$$

As before, higher-order approximations exist

$$h^2 \frac{d^2f}{dx^2} \approx -\frac{1}{12} u(x+2h) + \frac{4}{3} u(x+h) - \frac{5}{2} u(x) + \frac{4}{3} u(x-h) - \frac{1}{12} u(x-2h)$$

with error  $O(h^4)$

## Mixed Derivatives

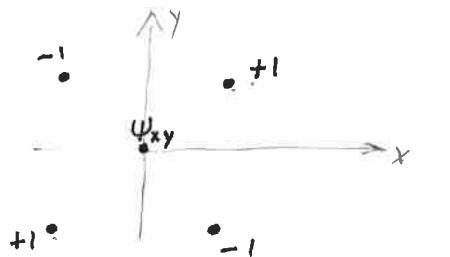
$$\frac{d^2\psi}{dxdy} = ?$$

piecewise. take FD deriv of FD deriv.

$$\begin{array}{ccccccc}
 y=1 & \bullet & \bullet & \bullet & \leftarrow & \psi'_1 = \frac{1}{h} (\psi(+1,1) - \psi(-1,1)) \\
 y=0 & \bullet & \bullet & \bullet & & = \frac{1}{2\Delta x} (\psi(1,1) - \psi(-1,1)) \\
 y=-1 & \bullet & \bullet & \bullet & \downarrow \Delta y & \\
 x=-1 & \xleftrightarrow{\Delta x} & x=0 & x=1 & \leftarrow & \psi'_{-1} = \frac{1}{2\Delta x} (\psi(1,-1) - \psi(-1,-1))
 \end{array}$$

Now take FD of  $\psi'_1$  and  $\psi'_{-1}$

$$\Psi_{xy} = \frac{1}{2\Delta y} (\psi'_1 - \psi'_{-1}) = \frac{1}{4\Delta x \Delta y} (\psi(+1,+1) - \psi(-1,1) - \psi(1,-1) + \psi(-1,-1))$$



Laplacian in FD approx.

$$\nabla^2 u = \frac{d^2 u}{dx^2} + \frac{d^2 u}{dy^2} = +1. \quad \begin{matrix} +1 \\ -4 \\ +1 \end{matrix}$$

Example.

Stencil for Stress

$$\frac{d^2 \sigma}{dx^2} + \frac{d^2 \sigma}{dkdy} + \frac{d^2 \sigma}{dy^2}$$
$$\begin{matrix} -1 & & +1 & & +1 \\ +1 & & -4 & & +1 \\ +1 & & +1 & & -1 \end{matrix}$$

# Finite Difference Check of Solutions

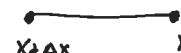
From Boyd's Chebyshev and Fourier Spectral Methods

Definition 16 IDIOT: Anyone who publishes a calculation without checking it against an identical computation with smaller  $N$  OR without evaluating the residual of the pseudospectral approximation via finite differences is an IDIOT

Harsh language.

For our purposes, a finite difference check is sufficient.

$$\frac{du}{dx} \approx \frac{u(x+\Delta x) - u(x)}{\Delta x}$$



$$\frac{d^2u}{dx^2} \approx \frac{u(x+\Delta x) - 2u(x) + u(x-\Delta x)}{\Delta x^2}$$

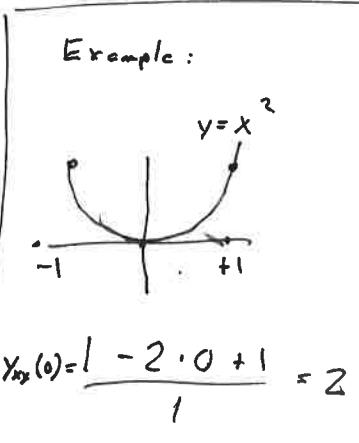
Example:

Project #2

$$U_t = U_{xx} \Rightarrow R = U_t - U_{xx} \text{ where } R \equiv \text{residual}$$

$$R = \left( \frac{U(x, t + \Delta t) - U(x, t)}{\Delta t} \right) - \left( \frac{U(x + \Delta x, t) - 2U(x, t) + U(x - \Delta x, t)}{\Delta x^2} \right)$$

This should be zero.



Demo

Project2-heat.py

FiniteDiffCheck()

# Numerical Finite Difference Example.

$$U_{xx} - 10U_c = 0$$

$$U(0) = 0$$

$$U(1) = 1$$



1) Convert to finite difference form.

$$\frac{U_w + U_E - 2U_c}{\Delta x^2} - 10U_c = 0$$



2) Solve for  $U_c$

$$10U_c + \frac{2U_c}{\Delta x^2} = \frac{U_w + U_E}{\Delta x^2} \Rightarrow U_c \left( 10 + \frac{2}{\Delta x^2} \right) = \frac{U_w + U_E}{\Delta x^2}$$

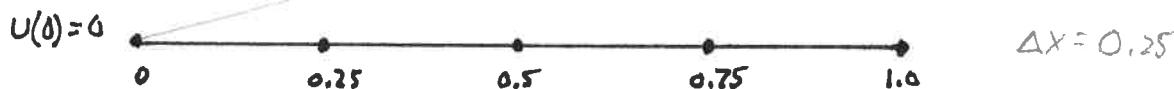
mult by  $\Delta x^2$

$$U_c (10\Delta x^2 + 2) = U_w + U_E \Rightarrow$$

$$\boxed{U_c = \frac{U_w + U_E}{2 + 10\Delta x^2}}$$

3) Create a grid and guess an initial condition (starting condition)

$$U(1) = 1$$



4) Iterate

	0	0.25	0.5	0.75	1.0
1	0	0.19	0.38	0.57	1.0
2	0	0.14	0.29	0.526	1.0
3	0				1.0
.					
15	0	0.0779	0.2045	0.4589	1.0

$$U_c = 0.381 \cdot (U_w + U_E)$$

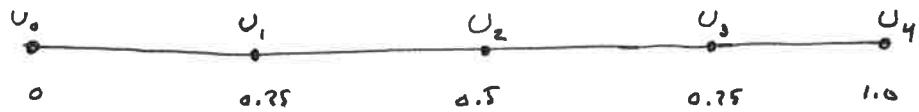
$\approx 135$  operations

Operations scale with # grid pts  
Unfortunately, convergence scales with # grid pts:

$O(\Delta x^{-3})$  or worse

4) Alternative One-step Matrix method  
 actually 0.380952

$$U_c = 0.381 (U_w + U_E) \Rightarrow 0.381 U_w - U_c + 0.381 U_E = 0$$



$$U_1) \quad U_w = U_0 \text{ and } U_E = U_2 \text{ and } U_c = U_1$$

$$\text{Substitute into update formula} \Rightarrow 0 - U_1 + 0.381 U_2 = 0$$

$$U_2) \quad U_w = U_1, \quad U_c = U_2, \quad U_E = U_3$$

$$\text{likewise} \Rightarrow 0.381 U_1 - U_2 + 0.381 U_3 = 0$$

$$U_3) \quad U_w = U_2, \quad U_c = U_3, \quad U_E = U_4 = 1.0$$

$$\text{likewise} \Rightarrow 0.381 U_2 - U_3 + 0.381 = 0$$

Place into matrix (the order doesn't matter)

$$\begin{bmatrix} 0 & -1 & 0.381 \\ 0.381 & 0.381 & -1 \\ -1 & 0.381 & 0.381 \end{bmatrix} \begin{pmatrix} U_3 \\ U_1 \\ U_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -0.381 \end{pmatrix}$$

↑ note the order!!

Rearrange (or just be smarter upfront!)

$$\begin{bmatrix} -1 & 0.381 & 0 \\ 0.381 & -1 & 0.381 \\ 0 & 0.381 & -1 \end{bmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -0.381 \end{pmatrix}$$

# TDMA

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ a_3 & b_3 & c_3 & & \\ \ddots & \ddots & \ddots & & \\ a_{n-1} & b_{n-1} & c_{n-1} & & \\ a_n & b_n & & & \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{n-1} \\ d_n \end{pmatrix}$$

The solution to this tri-diagonal matrix is easy with the "Thomas" aka "TDMA" method.

Compute:

$$c'_1 = \frac{c_1}{b_1} \quad \text{and} \quad d'_1 = \frac{d_1}{b_1}$$

$$c'_i = \frac{c_i}{b_i - a_i c'_{i-1}} \quad \text{and} \quad d'_i = \frac{d_i - a_i d'_{i-1}}{b_i - a_i c'_{i-1}}$$

Solution:

Start at  $n$ :

$$x_n = d'_n$$

step backwards to  $i = 1$

$$x_i = d'_i - c'_i x_{i+1}$$

TDMA applied to

$$\begin{bmatrix} -1 & 0.381 & & \\ 0.381 & -1 & 0.381 & \\ & 0.381 & -1 & \end{bmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -0.381 \end{pmatrix}$$

$$a = \begin{bmatrix} & 0.381 & 0.381 \end{bmatrix}$$

$$b = \begin{bmatrix} -1 & -1 & -1 \end{bmatrix}$$

$$c = \begin{bmatrix} 0.381 & 0.381 & \end{bmatrix}$$

$$d = \begin{bmatrix} 0 & 0 & -0.381 \end{bmatrix}$$

$$c' = \begin{bmatrix} 0.381 & -0.446 & -0.4589 \end{bmatrix}$$

$$d' = \begin{bmatrix} 0 & 0 & 0.4589 \end{bmatrix}$$

$$x = \begin{bmatrix} 0.0779 & 0.2045 & 0.4589 \end{bmatrix}$$

$\approx 23$  operations

scales with  $\mathcal{O}(n)$  !!

If you have a problem with  $10^6$  points,

TDMA is  $\approx 1$  trillion times faster than iteration.

5) Compare to exact solution (if possible)

$$U_{xx} - 10U = 0$$

• Substitute

$$U = A \sinh(\lambda x) + B \cosh(\lambda x)$$

$$U_{xx} = A\lambda^2 \sinh(\lambda x) + B\lambda^2 \cosh(\lambda x)$$

$$A\lambda^2 \sinh(\lambda x) + B\lambda^2 \cosh(\lambda x) - 10A \sinh(\lambda x) - 10B \cosh(\lambda x) = 0$$

$$\lambda^2 = 10$$

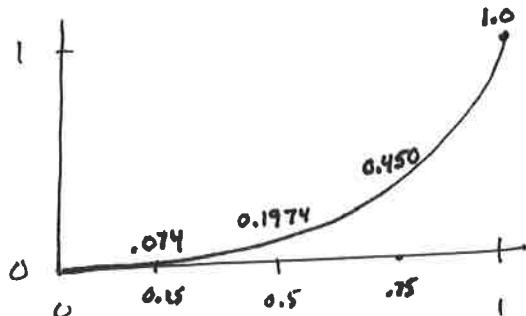
• Apply BCs

$$U(0) = 0 = A \sinh(\sqrt{10} \cdot 0) + B \cosh(\sqrt{10} \cdot 0) = B = 0$$

$$U(1) = 1 = A \sinh(\sqrt{10}) \Rightarrow A = \frac{1}{\sinh \sqrt{10}}$$

• Solution

$$U(x) = \frac{\sinh(\sqrt{10}x)}{\sinh \sqrt{10}}$$



What if you need to represent a Neumann BC?

- $U_x = C$  Introduce a ghost point

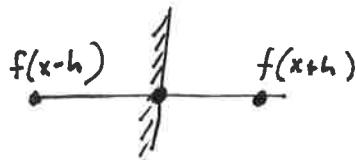


the ghost point is mirror image of the BC neighbor,

Effectively this means that the derivative is one-sided.

- Robin BC.

$$U_{x+h} + h U = C$$



Expand to

$$\frac{f(x+h) - f(x-h)}{2h} + h f(x) = C$$

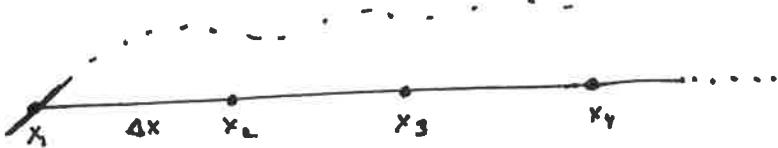
Mirror

$$f(x-h) = 2f(x) - f(x+h)$$

So Boundary values are (after some algebra)

$$f(x) = \frac{f(x+h) - hc}{2h} \cdot \frac{1}{1+h^2}$$

# Neumann BC with a numerical FD solution



- How can we specify a derivative?
- Introduce a ghost point outside the domain.



$$C = U_x(x_i) \approx \frac{U(x_2) - U(x_0)}{2\Delta x} \Rightarrow U(x_0) = U(x_2) - 2\Delta x C$$

- Update formula for  $x_1$  with  $\nabla^2 U = 0$ .

$$\frac{U_w + U_E - 2U_c}{\Delta x^2} = 0 \Rightarrow U_c = \frac{U_w + U_E}{2}$$

at  $x_1$ :

$$U_c = \frac{U(x_0) + U(x_2)}{2}$$

Substitute for  $U(x_0)$

$$U_1 = \frac{U(x_2) - 2\Delta x C + U(x_0)}{2} = U(x_2) - \Delta x C$$

- Conclusion

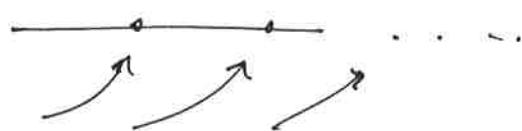
Update formula at this Neumann BC

$$U_1 = U_2 - \Delta x C$$



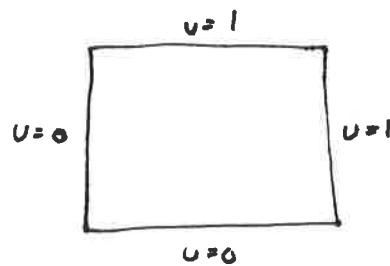
Update formula elsewhere

$$U_j = \frac{U_{j+1} + U_{j-1}}{2}$$



Example:

Use a finite difference formulation to solve  $\nabla^2 u = 0$  on a rectangular domain.



Governing Eqn

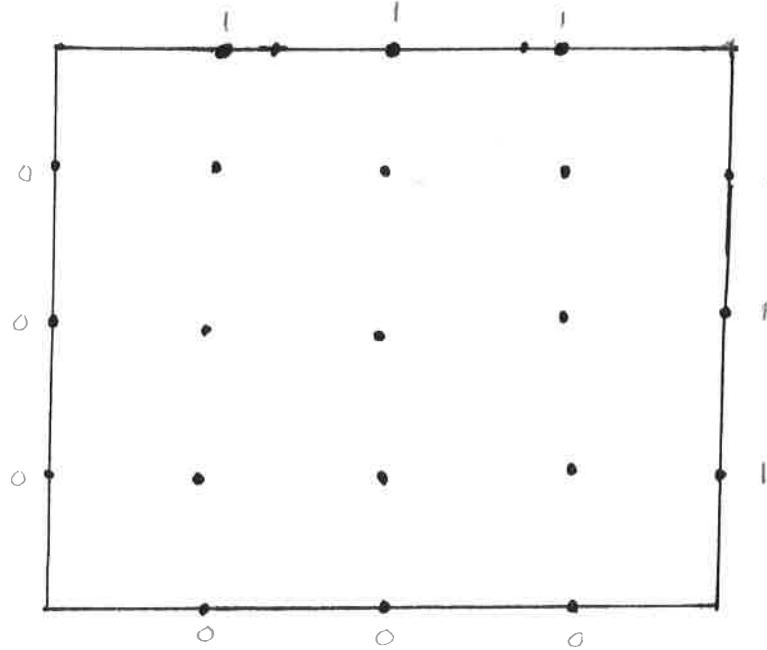
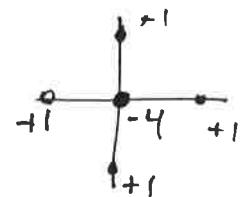
$$\nabla^2 u = \frac{d^2 u}{dx^2} + \frac{d^2 u}{dy^2}$$

FD:

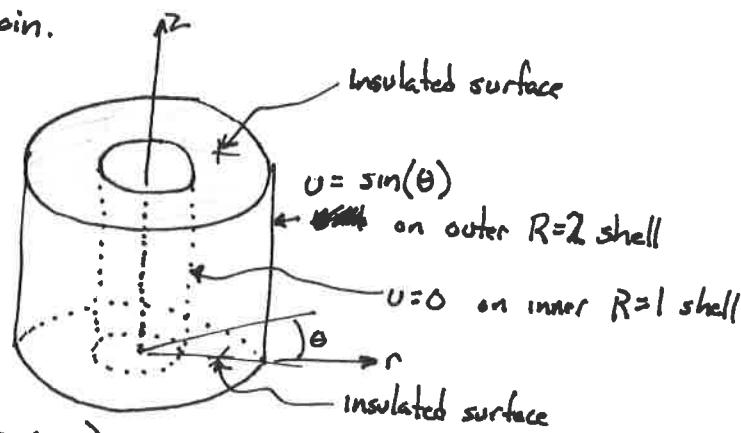
$$\nabla^2 u = \frac{U_{North} + U_{South} + U_{West} + U_{East} - 4U_c}{\Delta x^2} = 0$$

Thus,

$$U_c = \frac{1}{4}(U_N + U_S + U_W + U_E)$$



Example: Use a FD formulation to solve  $\nabla^2 u = 0$  on the annular cylindrical domain.



(ie. temperature across a pipe)

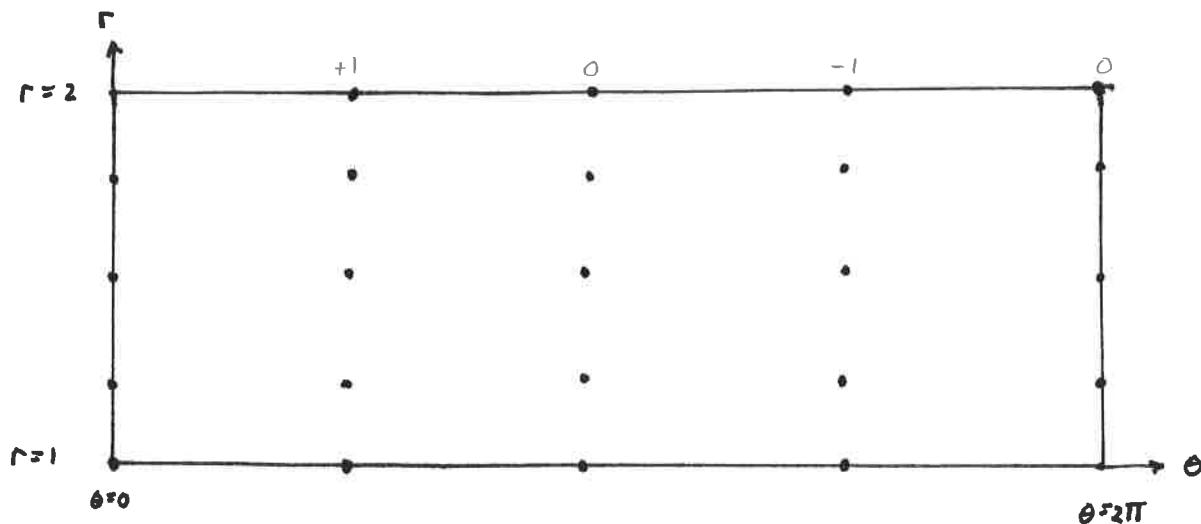
Simplify.

$$\frac{du}{dz} = 0 \quad \text{no change as } z \text{ changes.}$$

$$\frac{du}{d\theta} \neq 0 \text{ on } R=2$$

Gov Egu

$$\nabla^2 u = 0 = u_{rr} + \frac{1}{r} u_r + \frac{1}{r^2} u_{\theta\theta} + u_{zz}$$



What is the Gov Ego?

$$\frac{U_N + U_S - 2U_C}{\Delta r^2} + \frac{1}{r} \frac{U_N - U_S}{\Delta r} + \frac{1}{r^2} \frac{U_E + U_W - 2U_C}{\Delta \theta^2} = 0$$

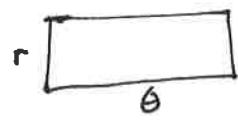
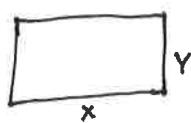
Solve for  $U_C$

$$+\frac{2U_C}{\Delta r^2} + \frac{2U_C}{\Delta \theta^2} = \frac{U_N + U_S}{\Delta r^2} + \frac{1}{r} \left( \frac{U_N - U_S}{\Delta r} \right) + \frac{1}{r^2} \left( \frac{U_E + U_W}{\Delta \theta^2} \right)$$

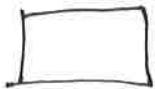
$$U_C = \frac{\Delta r^2 \Delta \theta^2}{2(\Delta \theta^2 + \Delta r^2)} \left( \frac{U_N + U_S}{\Delta r^2} + \frac{1}{r} \frac{U_N - U_S}{\Delta r} + \frac{1}{r^2} \frac{U_E + U_W}{\Delta \theta^2} \right)$$

Demo with excel

- Laplacian



- Diffusion



- Laplacian polar  $U_{rr} + \frac{1}{r} U_r + \frac{1}{r^2} U_{\theta\theta} = 0$

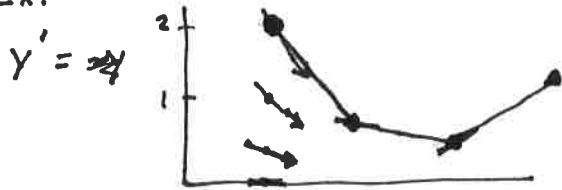
- Poisson problems

$$U_{xx} + U_{yy} = f$$

# Finite Difference Explicit Methods

Use past and current spatial information to compute future states

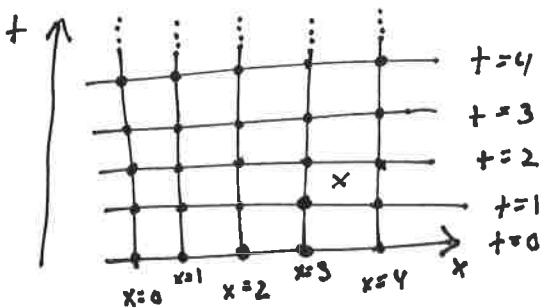
Ex:



Euler's Method.

$$y(t + \Delta t) = y(t) + \Delta t (-y(t))$$

- “Method of lines”



We only track and simulate the solution at intersections. The solution between points uses interpolation.

- Truncation Error

Difference between actual and numerical ~~derivatives~~ equations.

So substitute error equations into FD equations

$U_j^{n+1}$  is state  $u$  at  $j$  location and time  $n+1$

FD equation is

$$\Delta t U_t = U_j^{n+1} - U_j^n = \alpha \frac{\Delta t}{\Delta x^2} (U_{j+1}^n - 2U_j^n + U_{j-1}^n) = \alpha U_{xx}$$

Substitute to get (only look at 1st 2 terms)

$$U_t \Delta t + \underbrace{\frac{1}{2} U_{tt} \Delta t^2}_{\text{good parts}} = \alpha U_{xx}(\alpha x)^2 + \underbrace{\frac{1}{12} U^{(4)} \Delta x^4}_{\text{error parts}}$$

$$T_2(x, t) = \frac{1}{2} U_{tt} \Delta t^2 - \frac{\alpha^2}{12} U^{(4)} \Delta x^4$$

These 1st terms of the truncation error are called the “principle part”.

## Truncation Error Continued

We have  $U_{tt}$  in  $T(x,t)$ . Can we write this in terms of  $U_{xxxx}$ ?

Take the time derivative of  $U_t = \alpha^2 U_{xx}$

$$\frac{d}{dt}(U_t) = \alpha^2 \frac{d}{dt}(U_{xx}) = \alpha^2 \frac{d}{dt}\left(\frac{d^2 u}{dx^2}\right)$$

The derivative is a linear operator

$$\frac{d}{dt}(U_t) = \alpha^2 \frac{d^2}{dx^2} \left( \frac{du}{dt} \right)$$

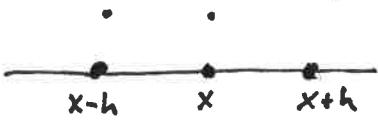
Thus

$$U_{tt} = \alpha^4 \frac{d^4 u}{dx^4}$$

Substitute into  $T(x,t)$

$$\begin{aligned} T(x,t) &= \frac{1}{2} \alpha^4 \frac{d^4 u}{dx^4} \Delta t^2 \frac{\alpha^2}{12} \frac{d^4 u}{dx^4} \Delta x^4 \\ &= \frac{1}{2} \alpha^2 \left( \alpha^2 \Delta t^2 - \frac{1}{6} \Delta x^4 \right) U_{xxxx} \end{aligned}$$

## Explicit Finite Difference Methods.

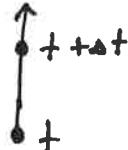


- Discretize the derivatives

$$U_{xx} \rightarrow \approx \frac{U(x+h) - 2U(x) + U(x-h)}{h^2}$$

- Use current values of space to advance time ( $t + \Delta t$ )

$$U_t = \frac{U(t + \Delta t) - U(t)}{\Delta t}$$

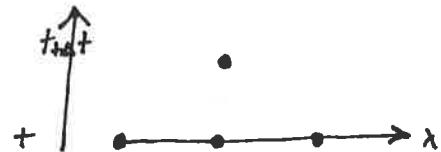


- Combine

Example  $U_t = U_{xx}$

$$\frac{U(t + \Delta t) - U(t)}{\Delta t} = \frac{U(x + \Delta x, t) - 2(U(x, t)) + U(x - \Delta x, t)}{\Delta x^2}$$

This notation is bulky.



- Errors

$$\delta_x^2 v(x, t) = v(x + \Delta x, t) - 2v(x, t) + v(x - \Delta x, t)$$

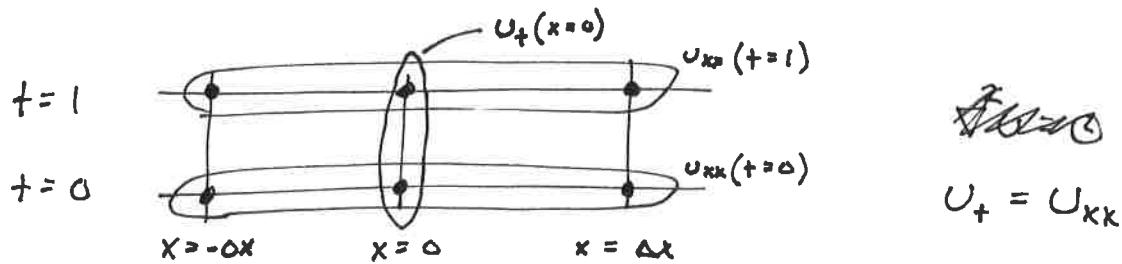
How does  $\delta_x^2 v$  compare to  $U_{xx}$ ?

$$\delta_x^2 v = U_{xx}(\Delta x)^2 + \underbrace{\frac{1}{12} U_{xxxx}(\Delta x)^4 + \dots}_{n \text{ is only even}} + \frac{1}{n!} U^{(n)} \Delta x^n$$

$$\begin{aligned} \Delta_t U(x, t) &= U(x, t + \Delta t) - U(x, t) \\ &\stackrel{\substack{\uparrow \\ \text{plus "t"}}}{=} U_t \Delta t + \frac{1}{2} U_{tt} \Delta t^2 + \frac{1}{6} U_{ttt} \Delta t^3 + \dots + \underbrace{\frac{1}{n!} U^{(n)} \Delta t^n}_{n \text{ is odd and even}} \end{aligned}$$

## Implicit Finite Difference (Crank - Nicolson)

Increase the stencil size to use derivatives at current and future times.



~~Stiff~~

$$U_+ = U_{xx}$$

The CN method takes a weighted average of central difference derivatives.

$$U_{xx} = \frac{\lambda}{\Delta x^2} \left( U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1} \right) + \frac{1-\lambda}{\Delta t^2} \left( \cancel{U_{j+1}^n} - 2U_j^n + U_{j-1}^n \right)$$

$$U_+ = \frac{1}{\Delta t} \left( U_j^{n+1} - U_j^n \right)$$

Again, you must solve the coupled  $U^{n+1}$  values simultaneously.

TDMA is a good choice.

For added insight, refer to a numerical methods book.

N.B.

All of these explicit and implicit methods are in the same family as Runge - Kutta (RK) methods.

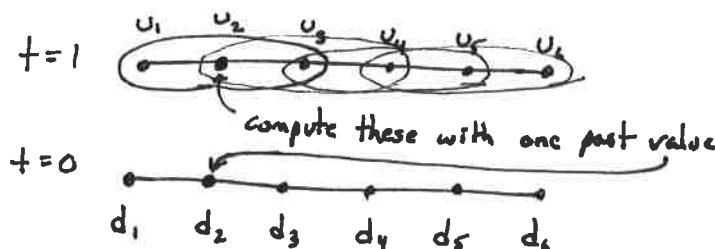
Implicit methods are often called "Stiff" Methods.

See "Solving Ordinary Differential Equations II" by Hairer and Wanner

There are special techniques for solving very stiff problems. These very stiff problems are not likely to occur for diffusion problems.

## Implicit Coupled Scheme.

- Now all of the future times are coupled, we can't just find each one individually
- Tridiagonal Matrix systems. Notice that the stencil is 3 wide and corresponds to a matrix



$$\frac{U_j^{n+1} - U_j^n}{\Delta x^2} = \frac{\alpha^2}{\Delta x^2} (U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1})$$

or with  $\nu = 1$

$$U_{j+1}^{n+1} + 3U_j^{n+1} + U_{j-1}^{n+1} = -U_j^n$$

or with  $\nu \neq 1$

$$-U_{j+1}^{n+1} + \left(2 - \frac{1}{\nu}\right)U_j^{n+1} - U_{j-1}^{n+1} = \frac{U_j^n}{\nu}$$

$$\begin{bmatrix} 3 & -1 & & & & \\ -1 & 3 & -1 & & & \\ & -1 & 3 & -1 & & \\ & & -1 & 3 & -1 & \\ & & & -1 & 3 & -1 \\ & & & & -1 & 3 \end{bmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

This is quickly solved with the TDMA (aka Thomas) algorithm.  
 See Farlow p321 Lesson 39.

What about iteration?

You might ask about an iterative method to solve the coupled problem.

This is called "fixed point iteration" and destroys the unconditional stability property of implicit methods.

## Fourier Analysis of the Error

$$U_t = \alpha^2 U_{xx}$$

Substitute a Fourier solution into the FD equations

$U_j^n = \lambda^n e^{ik(j\Delta x)}$  so  $\lambda^n$  is an amplification term in time  
also  $e^{ik(j\Delta x)}$  is a harmonic function in space

$$U_j^{n+1} = \lambda U_j^n$$

FD equation is  $\frac{U_j^{n+1} - U_j^n}{\Delta t} = \alpha^2 \left( \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{\Delta x^2} \right)$

Substitute  $\nu = \frac{\alpha^2 \Delta t}{\Delta x^2}$  and  $U_j^n$  into FD eqn.

$$\lambda U_j^n - U_j^n = \nu \left( \lambda^n e^{ik\Delta x} - 2\lambda^n e^{ik\Delta x} + \lambda^n e^{-ik\Delta x} \right)$$

$$\lambda^n e^{ik\Delta x} - \lambda^n e^{-ik\Delta x} = \dots$$

Reduce to

$$\lambda - 1 = \nu \left( e^{ik\Delta x} - 2 + e^{-ik\Delta x} \right)$$

Solve for  $\lambda$

$$\lambda = 1 + \nu (e^{ik\Delta x} - 2 + e^{-ik\Delta x})$$

$$\boxed{\lambda = 1 - 4\nu \sin^2 \frac{1}{2}k\Delta x} \approx 1 - 4\nu \left(\frac{1}{2}\right)^2 k^2 \Delta x^2 = 1 - \alpha^2 \Delta t k^2$$

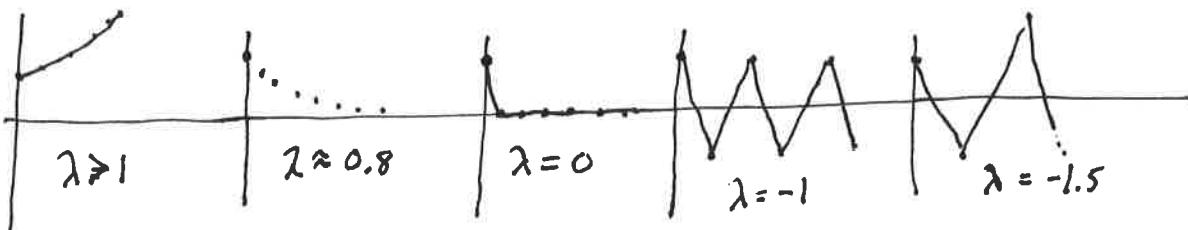
$$\boxed{\lambda \approx 1 - \alpha^2 k^2 \Delta t}$$

Taylor series for sin:  
 $\sin x \approx x + \dots$   
 $\sin^2 x \approx x^2 + \dots$

1)  $\lim_{\Delta t \rightarrow 0} \lambda = 1$  no amplification between successive values in time.

2) As  $k$  increases (spatial wavenumber increases), the amplification decreases to and past 0 (zero).

3) The amplification becomes negative when  $\nu$  or  $k$  is large. Thus, the explicit method overshoots.



We want  $\lambda$  between 0 and 1. The explicit method does not do this!

# Implicit Method

Compute spatial derivatives with future timesteps.

No DeLoreans involved.

$$\begin{array}{c} t=1 \quad \bullet \quad \bullet \quad \bullet \\ t=0 \quad \bullet \\ x=x_0 \quad x \quad x+\Delta x \end{array}$$



Ex.

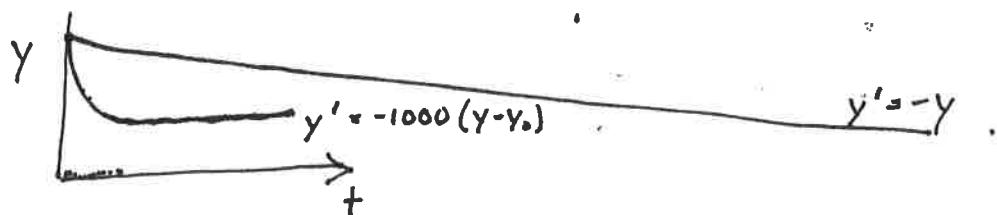
$$U_t = \alpha^2 U_{xx}$$

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} = \alpha^2 \left( \frac{U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1}}{\Delta x^2} \right)$$

- The implicit scheme operates along the so-called "slow manifold". The solution will not blow-up, but the error will be non-zero.
- Implicit schemes are strongly encouraged when you have a wide range of temporal scales.

Ex.

$$y' = -y - 1000(y - y_0)$$



The explicit method would need to resolve the  $y' = -1000(y - y_0)$  term everywhere.

# Fourier Analysis of an implicit method

$$U_t = \alpha^2 U_{xx}$$

FD equation

$$U_j^{n+1} - U_j^n = \nu \left( U_{j+1}^{n+1} - 2U_j^{n+1} + U_{j-1}^{n+1} \right)$$

Subs.

$$U_j^{n+1} = \lambda U_j^n$$

$$U_j^n = \lambda^n e^{ik(j\Delta x)}$$

$$\lambda U_j^n - U_j^n = \nu \left( \lambda U_{j+1}^n - 2\lambda U_j^n + \lambda U_{j-1}^n \right)$$

$$\lambda - 1 = \nu \left( \lambda e^{ik\Delta x} - 2 + \lambda e^{-ik\Delta x} \right)$$

$$\lambda(1 - \nu e^{ik\Delta x} - \nu e^{-ik\Delta x}) = 1 - 2\nu$$

Solve

$$\lambda = \frac{1 - 2\nu}{1 - \nu e^{ik\Delta x} - \nu e^{-ik\Delta x}}$$

$$\boxed{\lambda = \frac{1}{1 + 4\nu \sin^2 \frac{1}{2} k \Delta x}}$$

- $\lambda$  is always between 0 and 1.
- Implicit is unconditionally stable.
- This does not mean that the solution is exact, only that the solution will not blow-up.