NSFD Schemes for the Heat Equation

Miles Smith, 2023

July 27, 2022

Structure

1. Introduce the Heat Equation
2. Analytic Solution
3. Numerical Solutions
4. NSFD Approach
5. Current Progress and Next Steps

Introduction to the Heat Equation

The Heat Equation In $\mathbb{R}^1$:

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

The Heat Equation In $\mathbb{R}^1$:

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

What? A *Partial* Differential Equation (PDE), that describes heat, $u(x, t)$, in terms of time, t, and space, x

α represents thermal diffusivity

Introduction to the Heat Equation

The Heat Equation In $\mathbb{R}^1$:

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

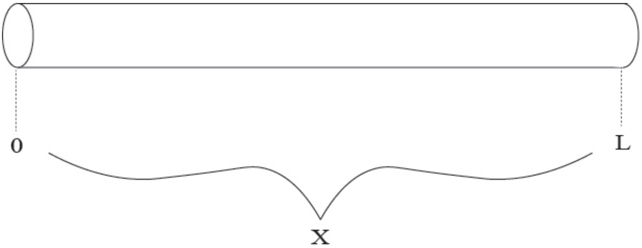
What? A *Partial* Differential Equation (PDE), that describes heat, $u(x, t)$, in terms of time, t, and space, x

α represents thermal diffusivity

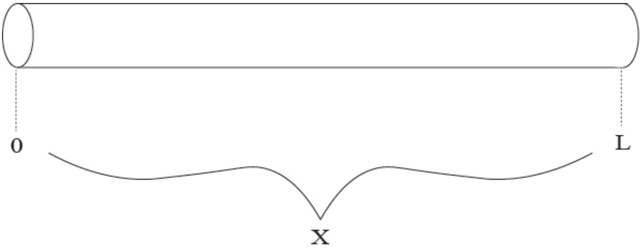
Solution

Heat Flow: How heat spreads out over time

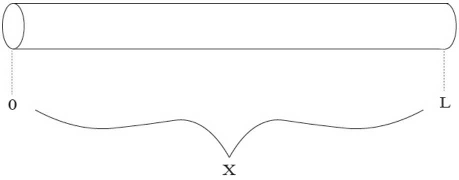
What 1-D Means



What 1-D Means



Question: If this rod starts out with some initial heat distribution, how does heat flow out of it over time? (What's the rod's heat in 1 second?)



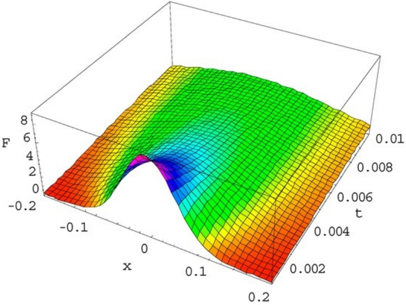
Question: how does heat flow over time?

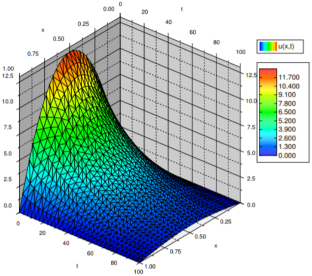
To Answer...

Initial heat distribution (IC): $u(x,0) = f(x)$

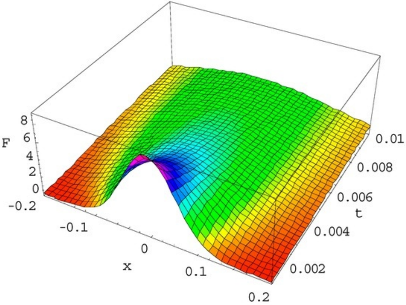
Heat at end-points (BC): $u(0,t) = ?\quad u(L,t) = ?$

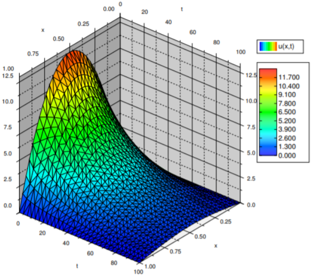
Heat Equation Visualization





Heat Equation Visualization





Here, $u(x,0)$ is the initial heat distribution
and $u(0,t) = u(L,t) = 0$

True Analytic Solution

Methods

Fourier Series (traditional)

Green's Function

Dirac-delta function for fundamental solution

True Analytic Solution

Methods

Fourier Series (traditional)
Green's Function
Dirac-delta function for fundamental solution

Let $u(0,t) = u(L,t) = 0$
and $u(x,0) = f(x)$

True Analytic Solution

Methods

Fourier Series (traditional)
Green's Function
Dirac-delta function for fundamental solution

Let $u(0,t) = u(L,t) = 0$
and $u(x,0) = f(x)$

The Answer...

$$u(x,t) = \sum_{n=1}^{\infty} b_n \cdot \sin\left(\frac{n\pi x}{L}\right) \cdot e^{-\left(\frac{\alpha n\pi}{L}\right)^2 t}$$

$$b_n = \frac{2}{L}\int_0^L f(x) \cdot \sin\left(\frac{n\pi x}{L}\right) dx$$

True Analytic Solution

The Answer...

$$u(x,t) = \sum_{n=1}^{\infty} b_n \cdot \sin\left(\frac{n\pi x}{L}\right) \cdot e^{-\left(\frac{\alpha n\pi}{L}\right)^2 t}$$

$$b_n = \frac{2}{L}\int_0^L f(x) \cdot \sin\left(\frac{n\pi x}{L}\right) dx$$

Problem: This is complicated... and what if $f(x)$ is complicated and makes it difficult to integrate?

General Problem

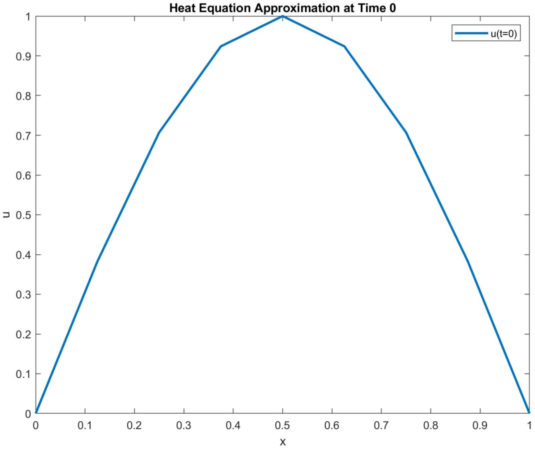
Let $L = 1,\ T = 1$

General Problem

Let $L = 1,\ T = 1$
BC: $u(0,t) = u(L,t) = 0$
IC: $u(x,0) = sin(\pi x)$

General Problem

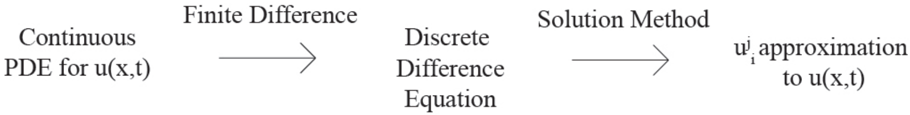
Let $L = 1,\ T = 1$
BC: $u(0, t) = u(L, t) = 0$
IC: $u(x, 0) = sin(\pi x)$



Numerical Solutions to the Heat Equations

Continuous PDE for u(x,t) $\xrightarrow{\text{Finite Difference}}$ Discrete Difference Equation $\xrightarrow{\text{Solution Method}}$ u^j_i approximation to u(x,t)

Numerical Solutions to the Heat Equations



Solution Methods:

- Explicit Euler
- Implicit Euler
- Crank-Nicolson
- Thomas Algorithm

Numerical Set-Up: (2) Finite Difference

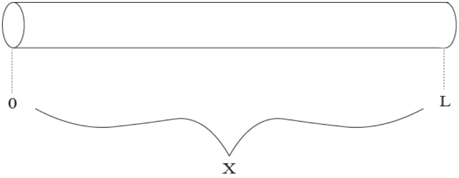
Recall:

$$\frac{df(x)}{dx} \approx \frac{f(x+h) - f(x)}{h}$$

Numerical Set-Up: (2) Finite Difference

Recall:

$$\frac{df(x)}{dx} \approx \frac{f(x+h) - f(x)}{h}$$

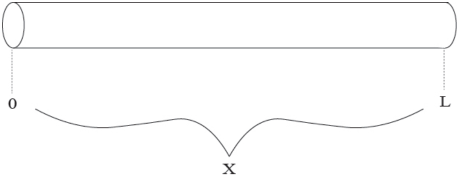


If $L = 1$, and we want to find the change in heat around $x = .5$ at $t = 0$ with respect to *time*.

Numerical Set-Up: (2) Finite Difference

Recall:

$$\frac{df(x)}{dx} \approx \frac{f(x+h) - f(x)}{h}$$



If $L = 1$, and we want to find the change in heat around $x = .5$ at $t = 0$ with respect to *time*. Then,

$$u_t(.5, 0) \approx \frac{u(.5, 0 + \Delta t) - u(.5, 0)}{\Delta t}$$

Numerical Set-Up: (2) Finite Difference

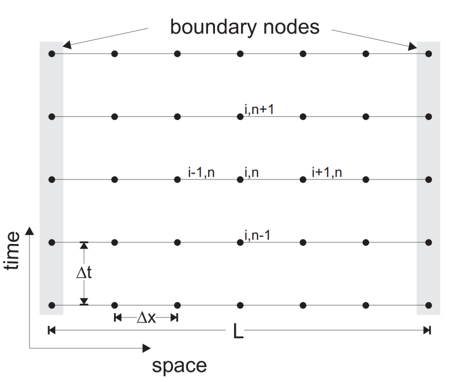
If both x and t go from 0 to 1, and we want to break them into discrete chunks...

Numerical Set-Up: (2) Finite Difference

If both x and t go from 0 to 1, and we want to break them into discrete chunks...

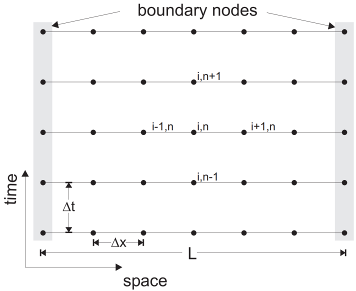
Finite Difference Mesh

$$\partial t \longrightarrow \Delta t \qquad \partial x \longrightarrow \Delta x$$



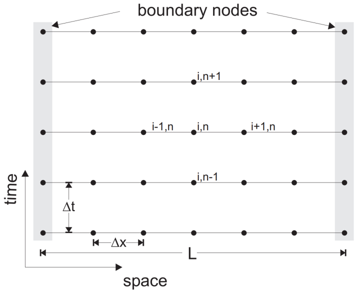
This mesh includes appoximations of heat, Δx and Δt apart, for all values of x and t.

Numerical Set-Up: (2) Finite Difference



Bottom line:

$$u^{j=1} = \begin{bmatrix} u_{i=1}^{j=1} & u_{i=2}^{j=1} & \dots & u_{i=7}^{j=1} \end{bmatrix}$$



Bottom line:

$$u^{j=1} = \begin{bmatrix} u_{i=1}^{j=1} & u_{i=2}^{j=1} & \dots & u_{i=7}^{j=1} \end{bmatrix}$$

$$u_{i=5}^{j=2} \approx u(x = .5, t = .2)$$

Numerical Set-Up: (2) Finite Difference

$$u_t(.5, 0) \approx \frac{u(.5, 0 + \Delta t) - u(.5, 0)}{\Delta t}$$

Numerical Set-Up: (2) Finite Difference

$$u_t(.5,0) \approx \frac{u(.5, 0+\Delta t) - u(.5,0)}{\Delta t}$$

Discretize Derivatives Using Finite Difference

$$\frac{\partial u}{\partial t} \approx \frac{u_i^{j+1} - u_i^j}{\Delta t}$$

and

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{\Delta x^2}$$

Numerical Set-Up: (3) Discrete Difference Equation

Bringing pieces together...

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} = \alpha \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{\Delta x^2}$$

Numerical Set-Up: (3) Discrete Difference Equation

Bringing pieces together...

$$\frac{u_i^{j+1} - u_i^j}{\Delta t} = \alpha \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{\Delta x^2}$$

Rearranged:

Discrete Heat Equation

$$u_i^{j+1} = (1 - 2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$

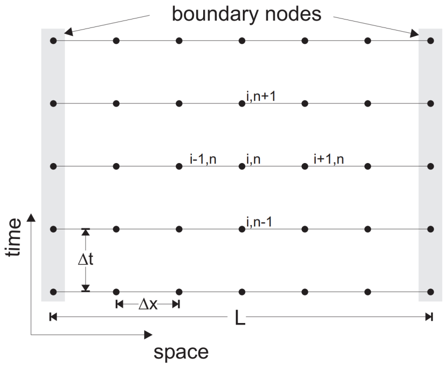
$$R = \frac{\Delta t \cdot \alpha}{\Delta x^2}$$

Numerical Set-Up: (4) Solution Methods

1. **Explicit Euler**
2. Implicit Euler

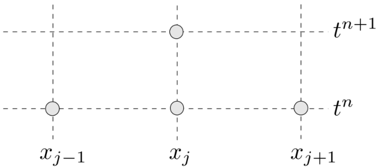
Solution Method: Explicit Euler

$$u_i^{j+1} = (1 - 2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$



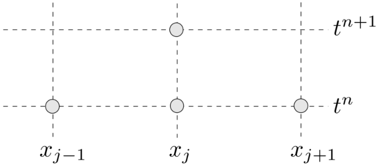
Solution Method: Explicit Euler

$$u_i^{j+1} = (1 - 2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$



Solution Method: Explicit Euler

$$u_i^{j+1} = (1-2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$



$$u_3^4 = (1-2R)u_3^3 + R(u_4^3 + u_2^3) \approx u(x=.5, t=.05)$$

Solution Method: Explicit Euler

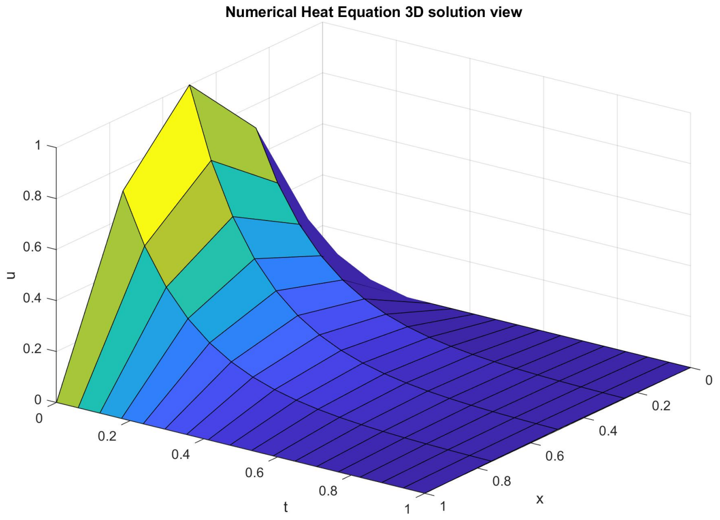
How do you create each time-step vector?

Explicit For-Loop

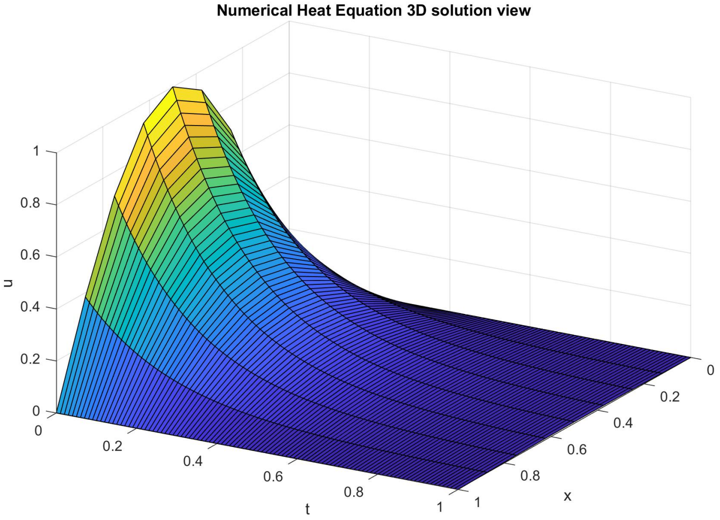
```
a = @(t) 0;              %boundary condition x=0
b = @(t) 0;              %boundary condition x=L
g = @(x) sin(pi.*x);     %initial condition, t=0

u=zeros(M+1, K+1);

u(1,:)=a(t'); %input BCs
u(end,:)=b(t');
u(:,1)=g(x); %input ICs

% Forward Difference Scheme
for j=1:K
    for i=2:M
        u(i, j+1) = (1-2*R)*u(i,j) + R*(u(i+1,j) + u(i-1,j));
    end
end
```

K and M are the number of steps for time and space, respectively

Explicit Euler Example: Simple Mesh



Explicit Euler Example: Refined Mesh



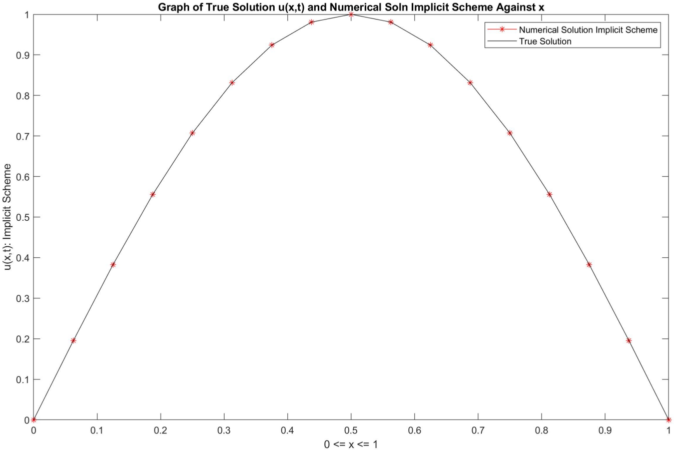
Approximate True Solution

How do we know if our approximation is good?

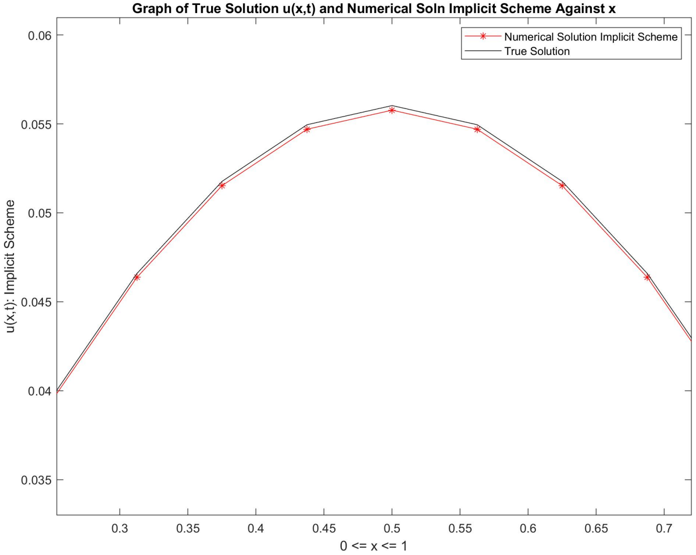
True Solution Baseline

```
%loop for exact solution
u_true=zeros(M+1,K+1);

for j=1:K+1
    for i=1:M+1
        u_true(i,j) = sin(pi.*x(i))*exp(-D*pi^2*t(j));
    end
end
```

True versus Explicit Euler: First Time Step



True versus Explicit Euler: 300th Time Step



Explicit Method Analysis

Order of Convergence:

$$\mathcal{O}(\Delta x)^2 + \mathcal{O}(\Delta t)$$

Explicit Method Analysis

Order of Convergence:

$$\mathcal{O}(\Delta x)^2 + \mathcal{O}(\Delta t)$$

$\mathcal{O}(\Delta t) \approx 1$, and $\mathcal{O}(\Delta x)^2 \approx 2$

The total order is *dominated* by Δt, so it should be closer to 1

Explicit Method Analysis

Conditionally Stable: Under the CFL condition

$$R = \frac{\Delta t \cdot \alpha}{\Delta x^2} < 1/2$$

Explicit Method Analysis

Conditionally Stable: Under the CFL condition

$$R = \frac{\Delta t \cdot \alpha}{\Delta x^2} < 1/2$$

Means: Δt has to be *much* smaller than Δx

Numerical Set-Up: (4) Solution Methods

1. Explicit Euler
2. **Implicit Euler**

Solution Method: Implicit Euler

$$u_i^{j+1} = (1 - 2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$

Solution Method: Implicit Euler

$$u_i^{j+1} = (1-2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$

$$A \cdot \vec{x} = \vec{b}$$

Coefficient Matrix, A: A matrix representing the coefficients in the system of equations

Solution Method: Implicit Euler

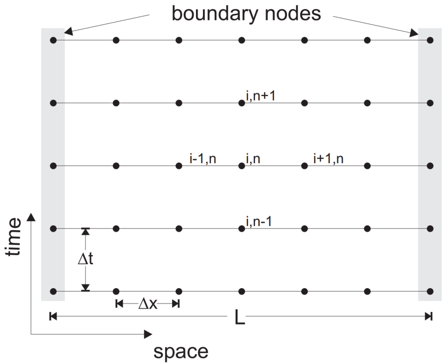
$$u_i^{j+1} = (1-2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$

$$A \cdot \vec{x} = \vec{b}$$

Coefficient Matrix, A: A matrix representing the coefficients in the system of equations

$$\vec{x} = A^{-1}\vec{b}$$

Solution Method: Implicit Euler



Solution Method: Implicit Euler

$$u_i^{j+1} = (1-2R)u_i^j + R\left(u_{i+1}^j + u_{i-1}^j\right)$$

$$\underbrace{\begin{bmatrix} 1-2R & R & 0 & 0 & 0 & \dots & 0 \\ R & 1-2R & R & 0 & 0 & \dots & 0 \\ 0 & R & 1-2R & R & 0 & \dots & 0 \\ \vdots & & & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & R & 1-2R \end{bmatrix}}_{A} \cdot \begin{bmatrix} u_2^{j+1} \\ u_3^{j+1} \\ u_4^{j+1} \\ \vdots \\ u_M^{j+1} \end{bmatrix} = \begin{bmatrix} u_2^j \\ u_3^j \\ u_4^j \\ \vdots \\ u_M^j \end{bmatrix} + R \begin{bmatrix} u_1^j \\ 0 \\ \vdots \\ 0 \\ u_{M+1}^j \end{bmatrix}$$

Note: A is a *Tridiagonal Matrix*

Solution Method: Implicit Euler

Since out boundary condition is always 0,

$$\begin{bmatrix} u_1^{j+1} \\ u_2^{j+1} \\ u_3^{j+1} \\ \vdots \\ u_M^{j+1} \end{bmatrix} = A^{-1} \cdot \begin{bmatrix} u_1^{j} \\ u_2^{j} \\ u_3^{j} \\ \vdots \\ u_M^{j} \end{bmatrix}$$

$$A = \begin{bmatrix} 1-2R & R & 0 & 0 & 0 & \dots & 0 \\ R & 1-2R & R & 0 & 0 & \dots & 0 \\ 0 & R & 1-2R & R & 0 & \dots & 0 \\ \vdots & & & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & R & 1-2R \end{bmatrix}$$

```
% Data structures for the linear tridiagonal system
A = zeros(N_x-1, N_x-1);

    for i=1:N_x-1
    A(i,i)=1-2*lambda; % diagonal
    end

for i=2:N_x-1
    A(i,i-1)=lambda; %subdiagonal
end

for i=1:N_x-2
    A(i,i+1)=lambda; %superdiagonal
end
```

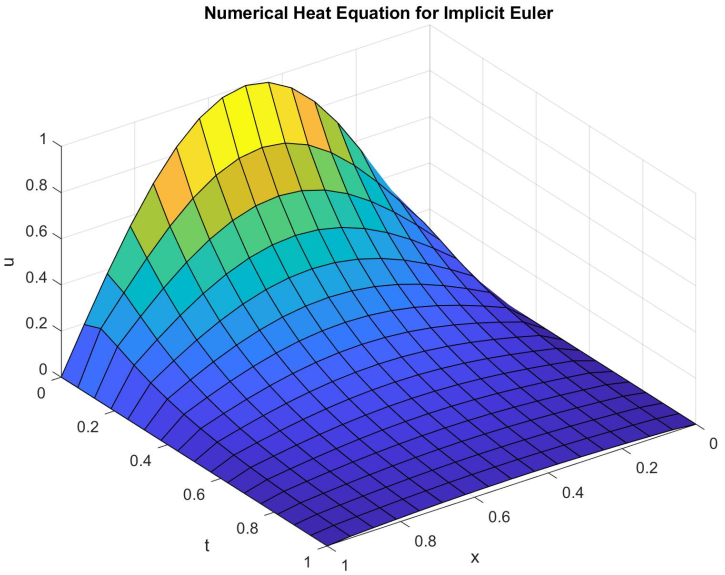
Implicit Code

```
v=zeros(N_x-1,1);        %BC
for j=1:N_t-1
% Solve triagonal system
v(1,1)=0;                    %BC at start
v(N_x-1,1)=0;                %BC at end

u(:,j+1)=A\(u(:,j)+lambda*v);    %x = A^-1 * b
end
```

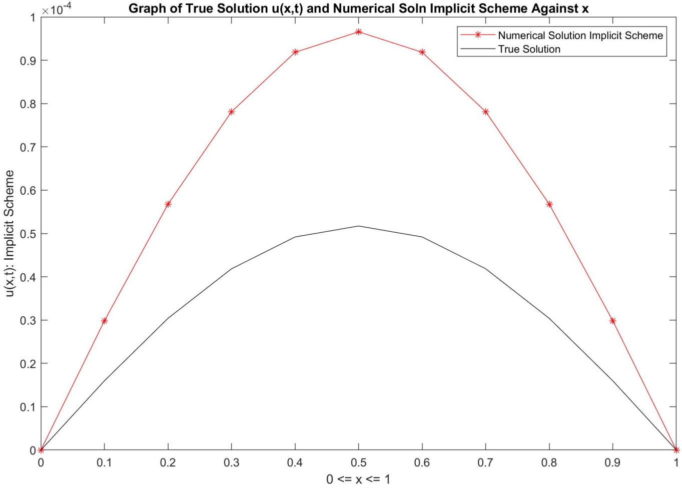
Implicit Code

```
v=zeros(N_x-1,1);       %BC
for j=1:N_t-1
% Solve triagonal system
v(1,1)=0;               %BC at start
v(N_x-1,1)=0;           %BC at end

u(:,j+1)=A\(u(:,j)+lambda*v);    %x = A^-1 * b
end
```

Implicit Graph



True versus Implicit Euler



Implicit Method Analysis

Unconditionally Stable: *always works*

Unconditionally Stable: *always works*

L^1, L^2, L^∞ Error for Implicit Scheme for the Heat Equation: For fixed Δx and when Δt is cut in half.

Δx	Δt	L^1 Error	L^2 Error	L^∞ Error
0.01000	0.01000	$2.42598905e-03$	$2.69481799e-04$	$3.81104814e-05$
0.01000	0.00500	$1.08459172e-03$	$1.20477760e-04$	$1.70381282e-05$
0.01000	0.00250	$5.13610852e-04$	$5.70525148e-05$	$8.06844401e-06$
0.01000	0.00125	$2.50928217e-04$	$2.78734099e-05$	$3.94189542e-06$
0.01000	0.000625	$1.25033217e-04$	$1.38888410e-05$	$1.96417873e-06$

Implicit Method Analysis: Order

$$\mathcal{O}(\Delta x)^2 + \mathcal{O}(\Delta t) \approx 1$$

Δx	Δt	Δt Ratio	L^∞ Error	Ratio of Errors	Order
0.01000	0.01000	$\sim$	$3.81104814e-05$	$\sim$	$\sim$
0.01000	0.00500	0.50000	$1.70381282e-05$	$1.70381282e-05/3.81104814e-05 = 0.44707$	1.1614
0.01000	0.00250	0.50000	$8.06844401e-06$	$8.06844401e-06/1.70381282e-05 = 0.47355$	1.0784
0.01000	0.00125	0.50000	$3.94189542e-06$	$3.94189542e-06/8.06844401e-06 = 0.48855$	1.0334
0.01000	0.000625	0.50000	$1.96417873e-06$	$1.96417873e-06/3.94189542e-06 = 0.49828$	1.0049

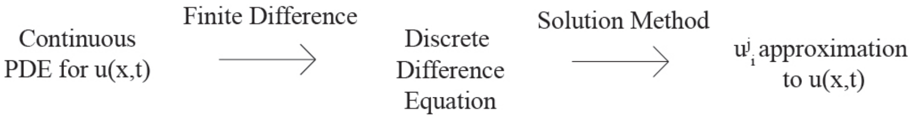
$$log_2(0.44707)/log_2(0.5) = 1.1614$$

$$log_2(0.47355)/log_2(0.5) = 1.0784$$

$$log_2(0.48855)/log_2(0.5) = 1.0334$$

$$log_2(0.49828)/log_2(0.5) = 1.0049$$

NSFD Approach



Change Finite Difference Scheme: Then, see how explicit and implicit change as a result

Unity Approximations: Averaging

$$u_i^{j+1} = (1 - 2R)u_i^j \times 1_0 + R(u_{i+1}^j + u_{i-1}^j) \times 1_1$$

Unity Approximations: Averaging

$$u_i^{j+1} = (1 - 2R)u_i^j \times 1_0 + R(u_{i+1}^j + u_{i-1}^j) \times 1_1$$

2-point Average:

$$\frac{u_A^j + u_B^j}{2u_C^j}, \quad A, B, C \in \{i, i+1, i-1\}$$

Unity Approximations: Averaging

$$u_i^{j+1} = (1-2R)u_i^j \times 1_0 + R(u_{i+1}^j + u_{i-1}^j) \times 1_1$$

2-point Average:

$$\frac{u_A^j + u_B^j}{2u_C^j}, \quad A, B, C \in \{i, i+1, i-1\}$$

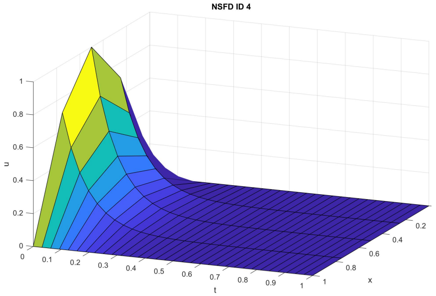
3-point Average:

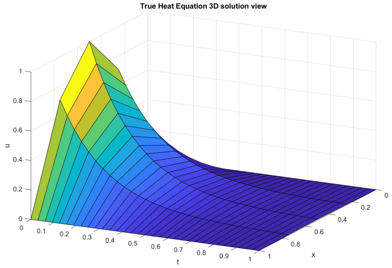
$$\frac{u_A^j + u_B^j + u_C^j}{3u_D^j}, \quad A, B, C, D \in \{i, i+1, i-1\}$$

2-pt Avg Technique at 1_0

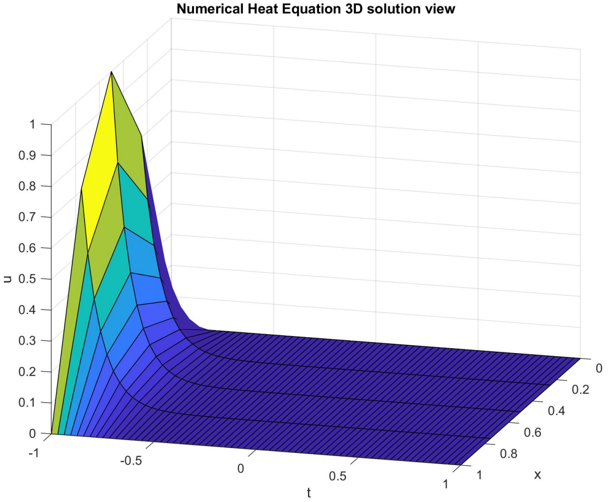
ID	A	B	C	$u_i^{j+1} =$
1	i	i + 1	i - 1	$\dfrac{(1-2R)\,u_i^j\left(u_i^j+u_{i+1}^j\right)}{2u_{i-1}^j}+R\left(u_{i+1}^j+u_i^j\right)$
3	i + 1	i	i - 1	$\dfrac{(1-2R)u_i^j\left(u_i^j+Ru_{i+1}^j\right)}{2u_{i-1}^j}+R/2$
4	i + 1	i - 1	i	$\dfrac{u_{i+1}^j+u_{i-1}^j}{2}$
6	i - 1	i + 1	i	$\dfrac{u_{i-1}^j+u_{i+1}^j}{2}$

NSFD ID 4: Explicit Euler

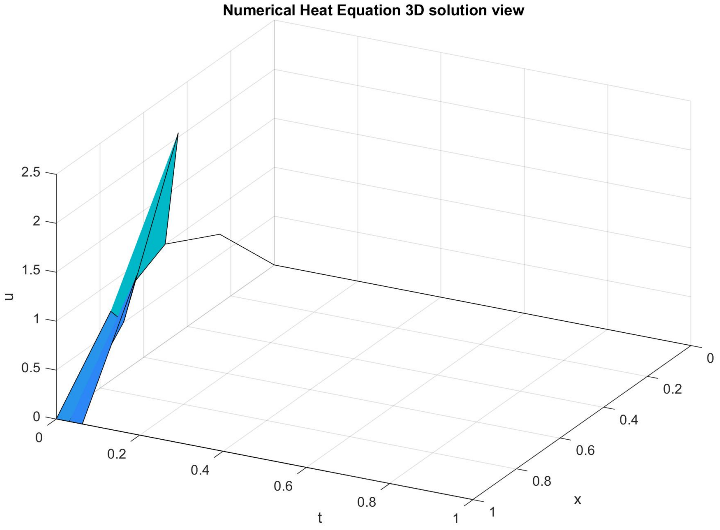




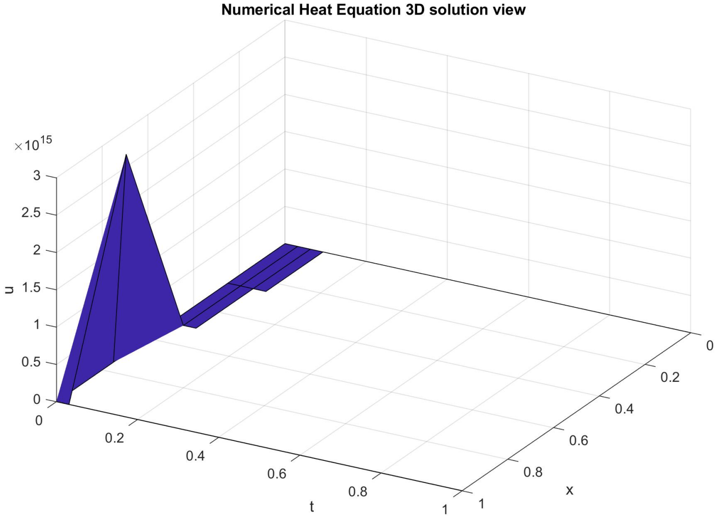
NSFD ID 16 (3pt Avg): Explicit Euler



NSFD ID 17 (3pt Avg): Explicit Euler



NSFD ID 25 (3pt Avg): Explicit Euler



Implicit NSFD Schemes

Goal: Create the A matrix

2-point Average:

$$\frac{u_A^X + u_B^Y}{2u_C^Z}, \quad A, B, C \in \{i, i+1, i-1\} \quad X, Y, Z \in \{j, j+1, j-1\}$$

Implicit NSFD Schemes

Goal: Create the A matrix

2-point Average:

$$\frac{u_A^X + u_B^Y}{2u_C^Z}, \quad A, B, C \in \{i, i+1, i-1\} \quad X, Y, Z \in \{j, j+1, j-1\}$$

When $j = 1$, $u_i^1 =$ function in terms of u_i^0

When $j = 2$, $u_i^2 =$ function in terms of u_i^1

Then can find the matrix A such that

$$A \cdot u^{j+1} = u^j$$

Next Steps

1. More Averages
2. Build out Implicit NSFDs
3. Find orders and compare against Implicit SFDs

Thank you!

Questions?