ENGR I1100
Engineering Analysis

Discretization

Nov 12, 2025

Rahul Pandare
PhD Candidate
Dept. of Chemical Engineering
rpandar000@citymail.cuny.edu

Discretization

- Discretization is the process of transforming continuous mathematical models, like ordinary differential equations (ODEs), into discrete versions suitable for numerical computation

Euler's Method

$$\frac{dy}{dt} = f(y, t)$$

At each **step**, we approximate the next value y(n+1) as,

$$y_{n+1} = y_n + h \cdot f(t_n, y_n)$$

Where,

$h = t_{n+1} - t_n$ (Discretization interval in t)

Euler's Method is used for simpler initial value problems involving ODEs, especially when high accuracy is not critical.

Finite Difference Method

$$\frac{dy}{dt} = f(y, t)$$

First **derivative** at a point t can be approximated as,

$$\frac{dy}{dt} \approx \frac{y(t+h) - y(t)}{h}$$

h = Discretization interval in t

Finite Difference Method is used for boundary value problems, multi-dimensional problems, or PDEs where you need to approximate derivatives at discrete points across a spatial domain.

Discretization

- Discretization is the process of transforming continuous mathematical models, like ordinary differential equations (ODEs), into discrete versions suitable for numerical computation

Euler's Method

Eq. from ex. 2 from last presentation →

$$\frac{dy}{dx} = -2y + x^2 = f(x, y)$$

Euler eq. →

$$y_{n+1} = y_n + h \cdot f(t_n, y_n)$$

Discretized form →

$$y_{n+1} = y_n + h \cdot f(x_n, y_n)$$

Finite Difference Method

Heat eq. →

$$\frac{\partial u(x,t)}{\partial t} = \alpha \cdot \left[\frac{\partial^2 u(x,t)}{\partial x^2}\right]$$

Finite difference method →

$$\frac{dy}{dt} \approx \frac{y(t+h) - y(t)}{h}$$

Discretized form →

$$U(i, m+1) = U(i, m) + \alpha \cdot (k/h^2) \cdot [U(i-1, m) - 2 * U(i, m) + U(i+1, m)]$$

Where,
i, m : index of discretized position and time element
k, h : unit length of discretized position and time element

Discretization (example 3)

Q. Numerically solve the 1D heat equation for a rod of unit length

x = 0 u(x , t) x = 1

Heat equation (1D) :

$$\dot{u} = \alpha \nabla^2 u$$

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

u(x,t) : temperature at position x at time t

α : thermal conductivity of the material

Assumption: The rod is long, thin, insulated. So that the heat is only exchanged at the ends.

$u(x, t = 0) = \sin\left(\pi * \frac{x}{L}\right)$ **Initial cond. (IC)**

$u(x = 0, t) = 0$
$u(x = L, t) = 0$ **Boundary cond. (BC)**

Discretization (example 3)

Discretizing the initial condition (IC) and boundary condition (BC):

u(x,t): actual state variable
U(i,m): discretized variable

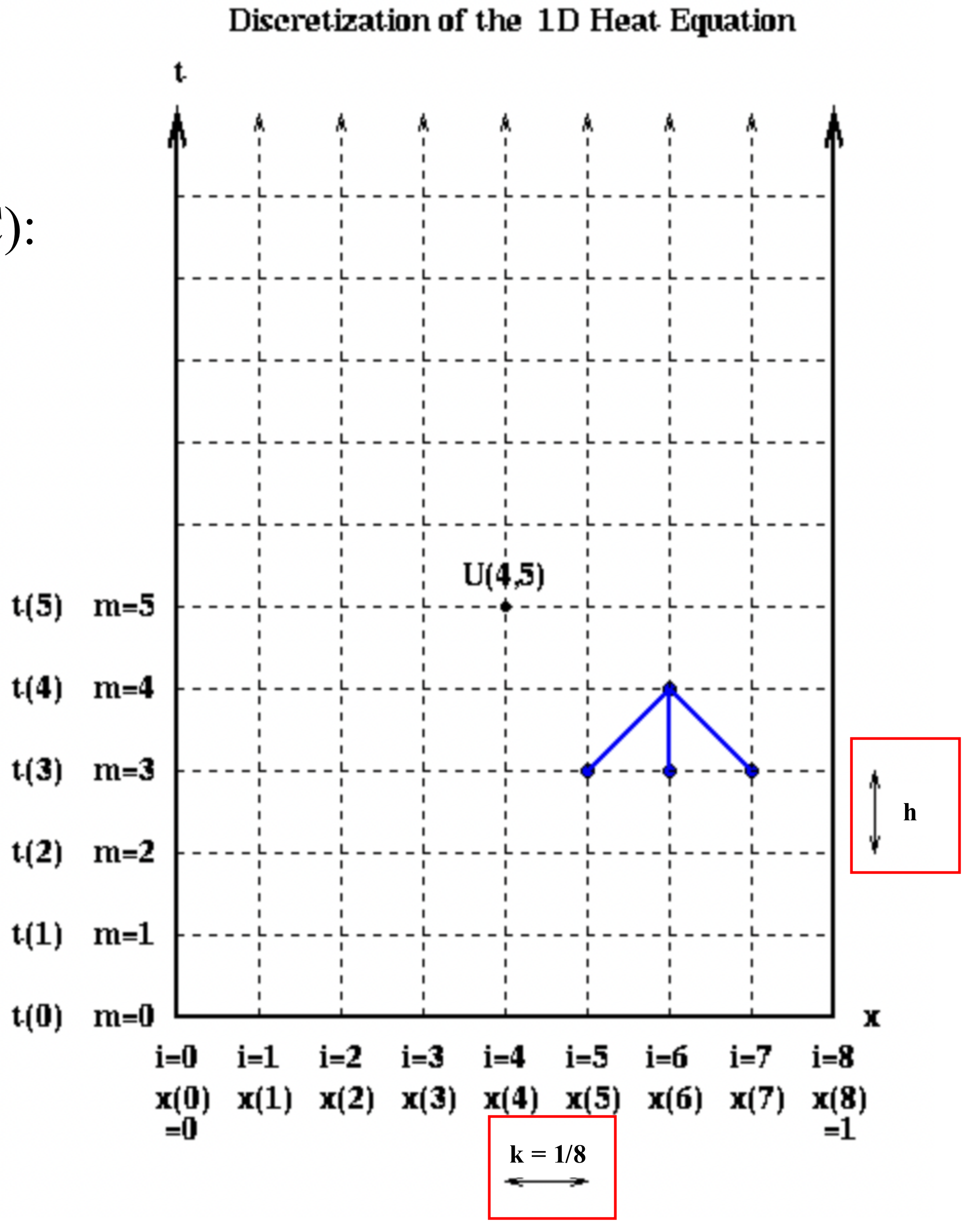
IC:

u(x,0) → U(i,0)

BC:

u(0,t) → U(0,m)
u(1,t) → U(n,m)

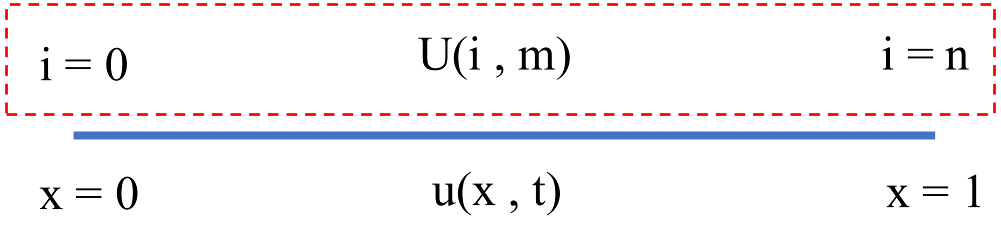
Where,
x, t : position and time
i, m: index of discretized position and time element
n: total number of discretized elements in the position variable



k, h : unit length of discretized position and time element respectively

Discretization (example 3)

Q. Numerically solve the 1D heat equation for a rod of unit length



Continuous form

$$\frac{\partial u(x,t)}{\partial t} = \alpha \cdot \left[\frac{\partial^2 u(x,t)}{\partial x^2}\right]$$

$u(x, t = 0) = \sin\left(\pi * \frac{x}{L}\right)$ **Initial cond. (IC)**

$u(x = 0, t) = 0$
$u(x = L, t) = 0$ **Boundary cond. (BC)**

u(x, t) : temperature at position x at time t

α : thermal conductivity of the material

Index form

general notation →
$$\boxed{\frac{\partial u(x(i), t(m))}{\partial t}} = \alpha \cdot \boxed{\left[\frac{\partial^2 u(x(i), t(m))}{\partial x^2}\right]}$$

Index notation →
$$\frac{\partial U(i,m)}{\partial t} = \alpha \cdot \left[\frac{\partial^2 U(i,m)}{\partial x^2}\right]$$

both eqs above are the same!

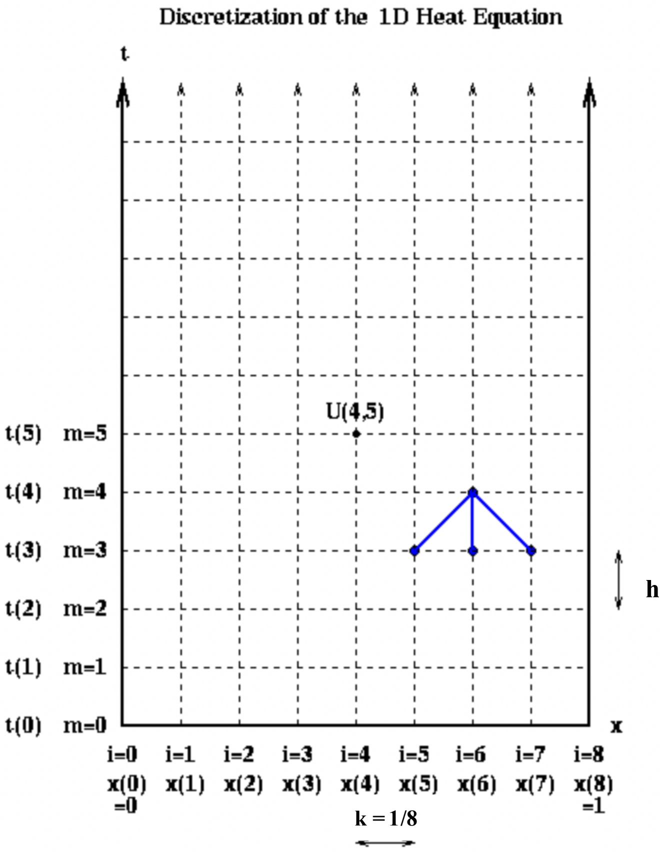
$U(i, m = 0) = \sin\left(\pi * \frac{i}{L}\right)$ **Initial cond. (IC)**

$U(i = 0, m) = 0$
$U(i = n, m) = 0$ **Boundary cond. (BC)**

Discretization (example 3)

Finite difference method → $\dfrac{dy}{dt} \approx \dfrac{y(t+h)-y(t)}{h}$

h is the temporal interval



Discretizing the governing equation (RHS):

Using finite difference method-

$$\frac{\partial^2 u(x(i),t(m))}{\partial x^2} \sim \frac{\boxed{\frac{\partial u(x(i)+h/2,t(m))}{\partial x}} - \boxed{\frac{\partial u(x(i)-h/2,t(m))}{\partial x}}}{h}$$

$$\sim \frac{u(x(i-1),t(m)) \; - \; 2*u(x(i),t(m)) \; + \; u(x(i+1),t(m))}{h^2}$$

$$\sim \frac{U(i-1,m) \; - \; 2*U(i,m) \; + \; U(i+1,m)}{h^2}$$

Using finite difference method-

$$\frac{\partial u(x(i)+h/2,t(m))}{\partial x} \sim \frac{u(x(i)+h,t(m))-u(x(i),t(m))}{h} \sim \frac{U(i+1,m)-U(i,m)}{h}$$

$$\frac{\partial u(x(i)-h/2,t(m))}{\partial x} \sim \frac{u(x(i),t(m))-u(x(i)-h,t(m)}{h} \sim \frac{U(i,m)-U(i-1,m)}{h}$$

Discretization (example 3)

Therefore, discretized heat equation (RHS) is:

$$\frac{\partial^2 u(x(i),t(m))}{\partial x^2} \sim \frac{U(i-1,m)\ -\ 2*U(i,m)\ +\ U(i+1,m)}{h^2} \qquad \text{(Eq. 1)}$$

Discretizing the governing equation (LHS):

$$\frac{\partial u(x(i),t(m))}{\partial t} \sim \frac{u(x(i),t(m+1))\ -\ u(x(i),t(m))}{k} \sim \frac{U(i,m+1)\ -\ U(i,m)}{k} \qquad \text{(Eq. 2)}$$

From equation 1 and 2 we get the discretized heat eq:

$$\frac{U(i,m+1)-U(i,m)}{k} = \alpha \cdot \frac{U(i-1,m)\ -\ 2*U(i,m)+U(i+1,m)}{h^2}$$

Heat equation in continuous form:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \cdot \left[\frac{\partial^2 u(x,t)}{\partial x^2}\right]$$

Discretization (example 3)

The discretized equation:

$$\frac{U(i,m+1) - U(i,m)}{k} = \alpha \cdot \frac{U(i-1,m) \; - \; 2 * U(i,m) + U(i+1,m)}{h^2}$$

$$\boxed{U(i,m+1) = U(i,m) + \alpha \cdot (k/h^2) \cdot [U(i-1,m) - 2 * U(i,m) + U(i+1,m)]}$$

We use the above equation in our MATLAB script.

Stability criterion

Stability criterion for discrete 1D heat equation (finite difference method):

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

1D heat equation:

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

Stability criterion for discrete 1D wave equation (finite difference method):

$$\Delta t \leq \frac{\Delta x}{c}$$

1D wave equation:

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

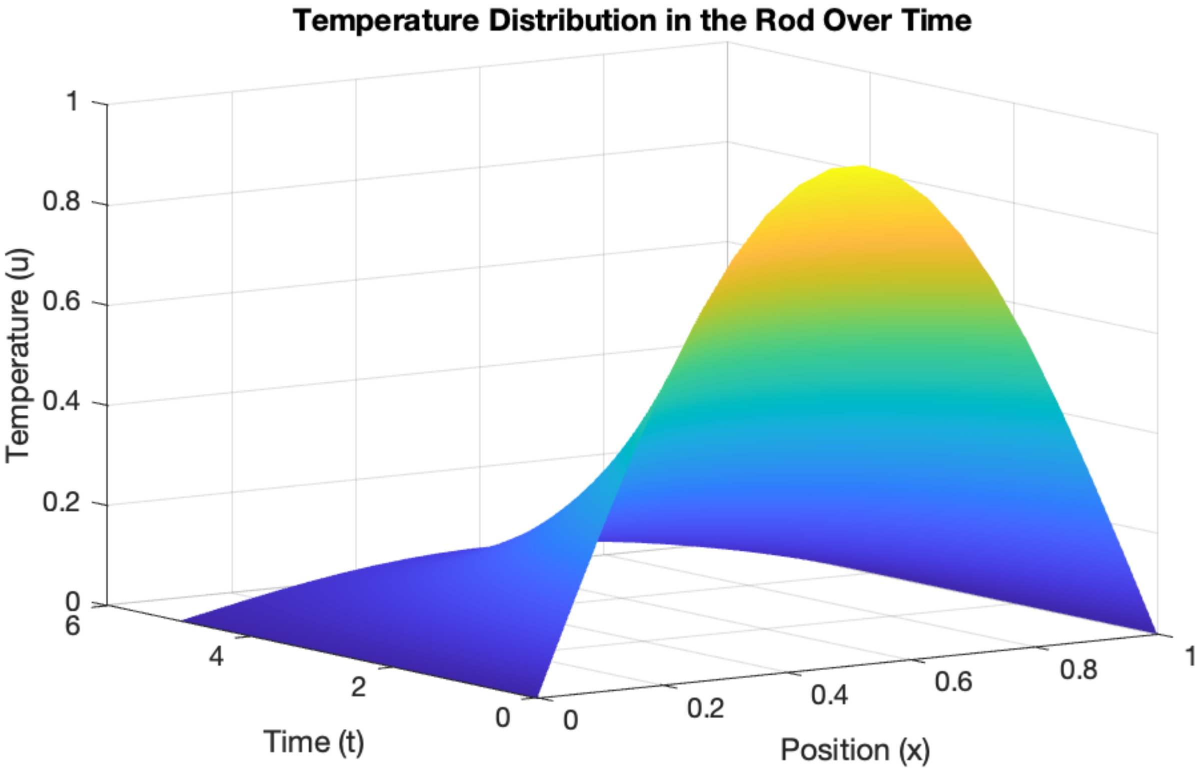
Δt is the temporal interval
Δx is the spatial interval

Derivation for the stability criterion linked [here]

Discretization (example 3)

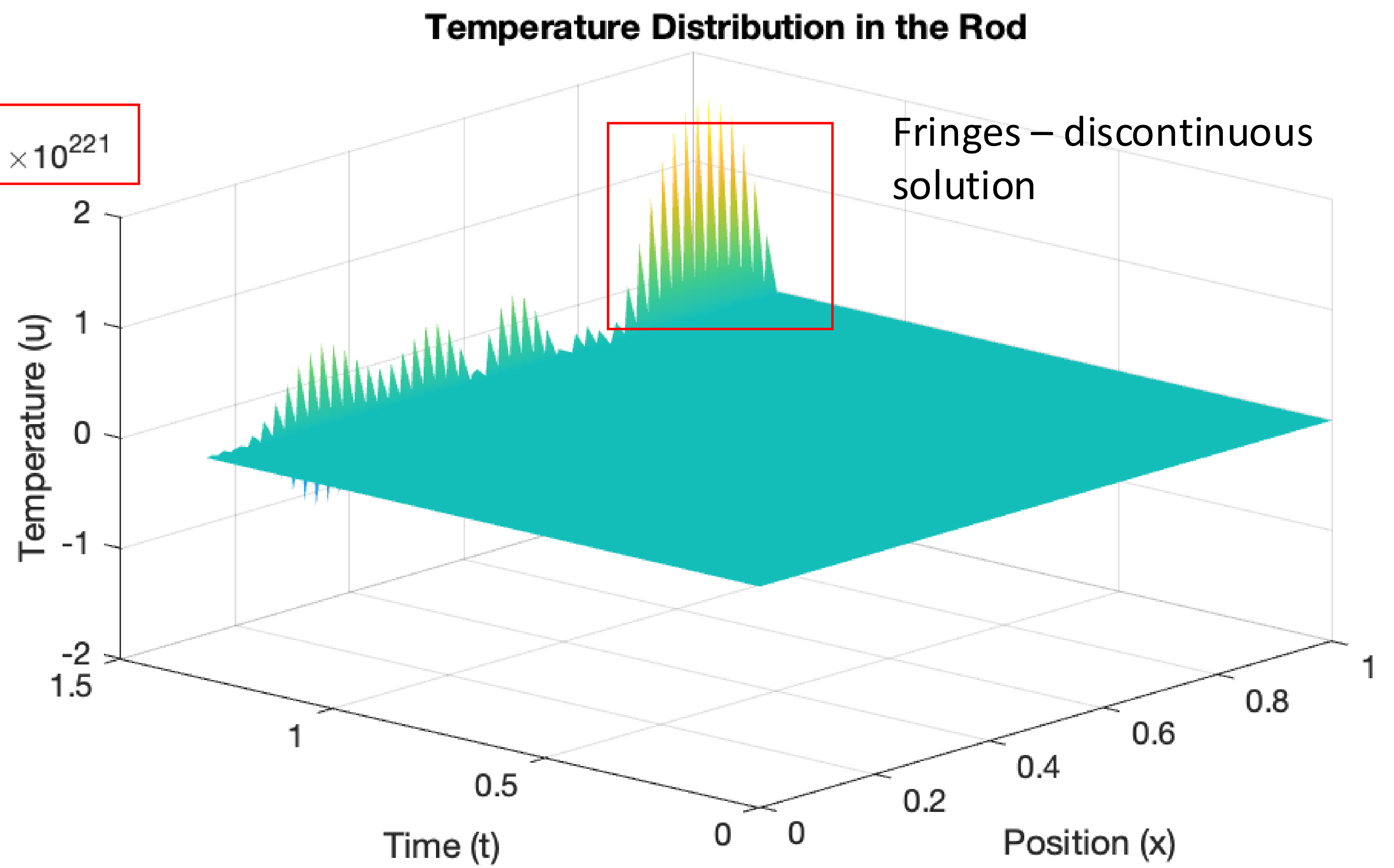
The discretized equation:

$$U(i,m+1) = U(i,m) + \alpha \cdot (k/h^2) \cdot [U(i-1,m) - 2 * U(i,m) + U(i+1,m)]$$




```matlab
clc;    % Clear the command window
clear;  % Clear all variables in workspace
clf;    % Clear the current figure handle

% Parameters
L = 1;               % Length of the rod
T = 5;               % Total simulation time (increased for steady state)
alpha = 0.05;        % Thermal diffusivity
Nx = 20;             % Number of spatial grid points
Nt = 200;            % Number of time steps (increased for finer resolution)
dx = L / (Nx - 1);   % Spatial step size

% Adjust dt to satisfy the stability criterion
dt = min(T / Nt, 0.5 * dx^2 / alpha);

% Discretized domain
x = linspace(0, L, Nx);
t = linspace(0, T, Nt);

% Initial condition
u = sin(pi * x / L);  % Sinusoidal initial condition

% Initialize U to store results
U = zeros(Nx, Nt);
U(:, 1) = u;          % Store the initial condition in the first column

% Time-stepping loop
for n = 1:Nt-1         % Temporal index
    % Update each spatial point (excluding boundaries)
    for i = 2:Nx-1
        u(i) = U(i, n) + alpha * dt / dx^2 * (U(i+1, n) - 2*U(i, n) + U(i-1, n));
    end
    % Enforce boundary conditions
    u(1)  = 0;
    u(Nx) = 0;

    % Store the updated values in U for the next time step
    U(:, n+1) = u;
end

% Create a 3D surface plot
[X, T] = meshgrid(x, t);

%figure;
surf(X, T, U');  % Transpose U for correct orientation
title('Temperature Distribution in the Rod Over Time');
xlabel('Position (x)');
ylabel('Time (t)');
zlabel('Temperature (u)');
shading interp; % Optional: for smoother plot
```

Absurd temperatures



Fringes – discontinuous solution

Unstable condition – this is what we don't want.
Make sure the stability criterion is met (slide 11).

Discretization (example 3)

- What if thermal conductivity α is not constant?

1. α varies with position (e.g. inhomogeneous material)

$$\alpha(x) = k' * \sin(x)$$

2. α varies with temperature – u(x,t) (usually the case)

$$\begin{aligned}\alpha(u) &= k_1 \quad \text{for} \quad u_1 \leq u < u_2 \\ &= k_2 \quad \text{for} \quad u_2 \leq u < u_3 \\ &= k_3 \quad \text{for} \quad u_3 \leq u < u_4\end{aligned}$$

Makes the PDE Non-Linear

Note: These equations are just examples!

Summary

- Discretization is indeed a powerful approach for obtaining approximate solutions PDEs, especially when analytical solutions are difficult or impossible to derive.

- Discretization of 1D heat eq:

$$\frac{\partial u(x,t)}{\partial t} = \alpha \cdot \left[\frac{\partial^2 u(x,t)}{\partial x^2}\right] \longrightarrow \begin{aligned} U(i,m+1) = U(i,m) + \alpha \cdot (k/h^2) \cdot [U(i-1,m) \\ -2 * U(i,m) + U(i+1,m)] \end{aligned}$$

Few Important points for project:

- Focus on the interpretation of the results than the algorithm part
- Show that you have understood the discretization method
- Talk about any limitations or cons about this method

Thank you!

Scripts used here and additional study material: https://rahul-pandare.github.io/teaching/matlab-intro

---

Project due: Dec 8

Office hours: Dec 2 or 3 (Tentative)

For question: rpandar000@citymail.cuny.edu ; Office: ST-305

Download MATLAB: https://www.mathworks.com/academia/tah-portal/city-university-of-new-york-1111017.html
(sign in with your cuny.edu email ID)