Modelling & Simulation of Chemical Engineering Systems

٥٠١ هعم : تمثيل الأنظمة الهندسية على الحاسب الآلى

Department of Chemical Engineering
King Saud University

LECTURE #9

Numerical Solution of Boundary-Value Differential Equations

BVP Examples

- Diffusion with chemical reaction in a catalyst. $$\frac{d^2c}{dx^2} = kc^2, \qquad c(0) = c_o, \qquad c(1) = c_1$$
- Diffusion or condition in a tubular reactor $$D\frac{d^2c}{dz^2} + v\frac{dc}{dz} - kc^2 = 0, \qquad c(0) = c_o, \qquad dc(L)/dz = c_1$$
- Heat & mass transfer in boundary layer problems.
- Disecretization of nonlinear PDEs

Initial & Boundary Conditions

$$\alpha \frac{\partial^2 T}{\partial x^2} = \frac{\partial T}{\partial t}, \qquad \text{Heat Conduction Equation}$$

$$\text{IC: } T = f(x) \quad or \quad T = T_o \qquad t = 0, \quad 0 \le x \le 1$$

BC :

Dirichlet BC:

$$T = f(t) \quad at \quad x = 0, \quad t > 0$$

$$T = T_2 \quad at \quad x = 1 \quad t > 0$$

Neumann

$$\frac{dT}{dx} = 0 \quad at \quad x = 1 \quad t > 0$$

Cauchy : Neumann+Dirichlet

Robbins Conditions

$$\frac{dT}{dx} = h(T - T_f) \quad at \quad x = 1 \quad t \ge 0$$

Numerical Methods

Shooting Method

Finite difference method

Methods of weight residuals, Collocation method

BVP Formulation

- Canonical Form

$$\frac{dy_j}{dx} = f_j(x, y_1, y_2, \cdots, y_n) \qquad x_o \le x \le x_f$$

$$y_j(x_o) = y_{jo} \qquad j = 1, 2, ..., r$$

$$y_j(x_f) = y_{jf} \qquad j = r+1,, ..., n$$

- Original Form

$$\frac{d^2y}{dx^2} = f\ (x, y, \frac{dy}{dx}) \qquad x_o \le x \le x_f$$

$$a_o y(x_o) + b_o \frac{dy(x_o)}{dx} = \gamma_o$$

$$a_f y(x_f) + b_f \frac{dy(x_f)}{dx} = \gamma_f$$

Shooting Method

- An initial-value technique
- Use successive approximation to solve for the missing initial conditions.
- It can utilize highly efficient initial value procedure.
- Shooting can be tried from either end of the BVP.

Shooting Method Algorithm

$$\frac{dy_1}{dx} = f_1(x, y_1, y_2) \qquad y_1(x_o) = y_{10}$$

$$\frac{dy_2}{dx} = f_1(x, y_1, y_2) \qquad y_2(x_f) = y_{2f}$$

- Problem:
- Guess initial condition for y_2 :

$$y_2(x_o) = \gamma$$

- Integrate the system forward, calculated y_2 at x_f is

$$y_2(x_f)\Big|_{calculated} = y_2(x_f, \gamma)$$

- Define the function

$$\phi(\gamma) = y_2(x_f, \gamma) - y_{2f} = 0$$

Nonlinear function need to be solved for γ

- Solution :

$$\Delta\gamma = \frac{-\phi(\gamma)}{\left[\dfrac{\partial\phi}{\partial\gamma}\right]} \qquad \gamma_{new} = \gamma_{old} + \rho\Delta\gamma, \qquad 0 < \rho \le 1$$

Example

- Solve the boundary-value differential equation:

$$\frac{d^2T}{dx^2} = 0.01(T-20)^{1.4} \qquad 0 \le x \le 10$$
$$T(0) = 200, \qquad T(10) = 40$$

- Canonical Form:

T'=z; z'=0.01(T-20)^1.4; T(0)=200; T(10)=40

Convert the BVP into IVP:

T'=z; z'=0.01(T-20)^1.4; T(0)=200; z(0)=γ

$$\Delta\gamma = \frac{-\phi(\gamma)}{\left[\frac{\partial\phi}{\partial\gamma}\right]} \qquad \gamma_{new} = \gamma_{old} + \rho\Delta\gamma, \qquad 0 < \rho \le 1$$

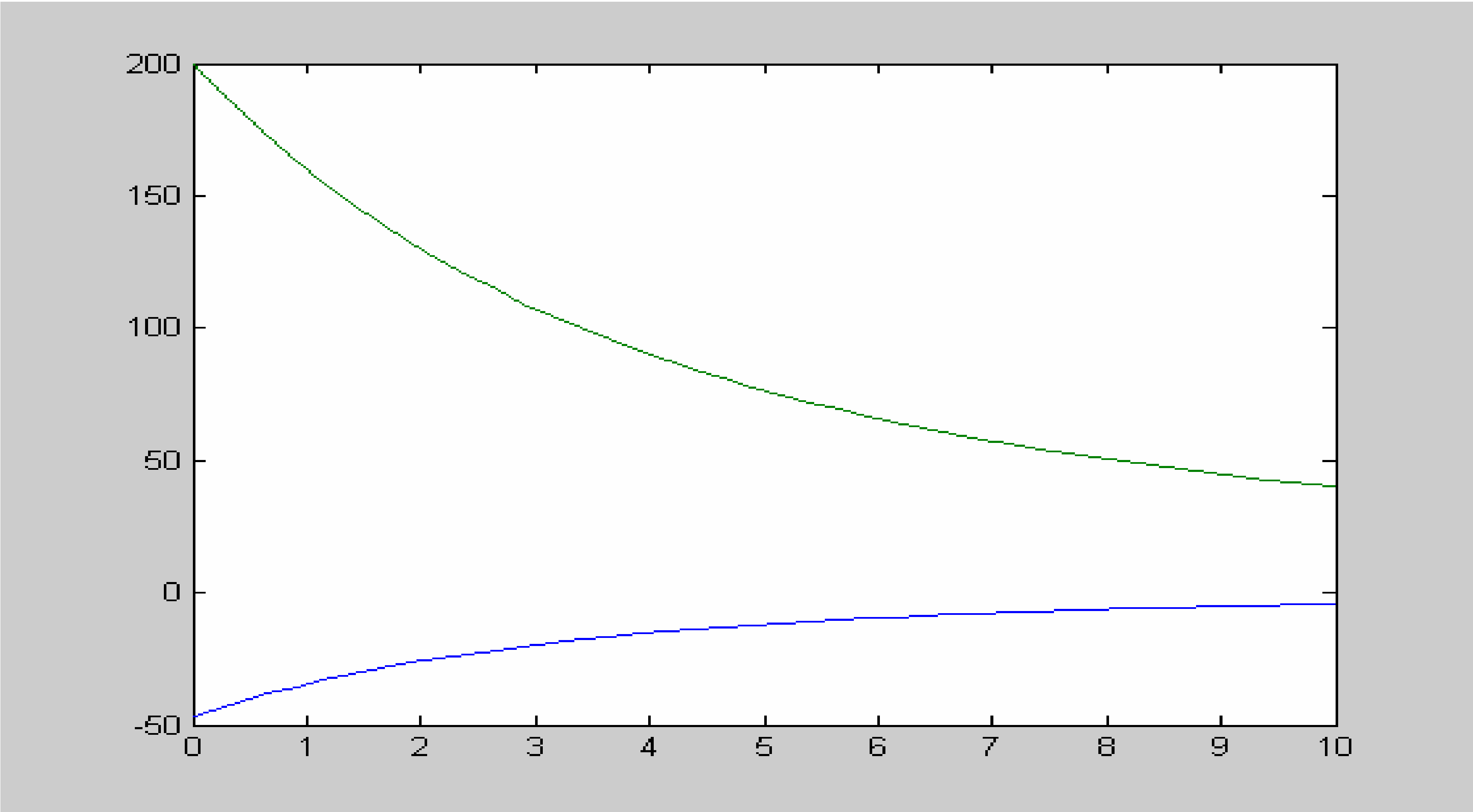
MATLAB Code

```
% Shooting Method for BVP
% dT^2/dx2=0.01(T-20)^1.4
% T(0)=200, T(10)=40
%
% Canonical Form: T'=z; z'=0.01(T-20)^1.4;
%T(0)=200; z(0)=g1
g0=20;
t0=200;
y2f_exact=40
tspan=[0 10]';
g=g0
 xx0=[t0 g]';
[x,y]=ode23('bvp1', tspan, xx0);
z=size(y)
y1f_g=y(z(1),1);
delta_g=0.1
delta_y2=1.0
```

```
while  delta_y2 >= 0.0001

   g=g+delta_g
   x0=[t0 g]';
yold=y1f_g
[x,y]=ode45('bvp1', tspan, x0);
z=size(y)
y1f_g=y(z(1),1)
delta_y2=y1f_g-y2f_exact
dif_y2=y1f_g-yold
Der_y2_g=dif_y2/delta_g
delta_g=-delta_y2/Der_y2_g

end
g
plot(x,y(:,2),x,y(:,1))
```



200
150
100
50
0
-50
0
1
2
3
4
5
6
7
8
9
10

Finite Difference Method

- Replace the derivative with finite difference approximation
- ODEs become algebraic equations

Forward

$$y' = \frac{y_{i+1} - y_i}{h}$$

Backward

$$y' = \frac{y_i - y_{i-1}}{h}$$

$$y' = \frac{y_{i+1} - y_{i-1}}{2h}$$

Central

$$y'' = \frac{y_{i+1} - 2y_i + y_{i+1}}{h^2}$$

Example:

$$\frac{d^2c}{dx^2} - 2c = 0 \qquad \Rightarrow \frac{c(x+h) - 2c(x) + c(x-h)}{h^2} - 2c(x) = 0$$

Example of FD method

$$\frac{d^2c}{dx^2} = 2c, \qquad \left.\frac{dc}{dx}\right|_{x=0} = 0 \qquad c(1) = 1$$

Let h=0.25

0 0.25 0.5 0.75 1

h

$$x = 0 \qquad \frac{c(.25) - c(0)}{h} = 0 \Rightarrow c(0.25) = c(0)$$

$$x = 0.25 \qquad \frac{c(0.5) - 2c(0.25) + c(0)}{0.25^2} - 2c(0.25) = 0$$

$$x = 0.5 \qquad \frac{c(0.75) - 2c(0.5) + c(0.25)}{0.25^2} - 2c(0.5) = 0$$

$$x = 0.75 \qquad \frac{c(1) - 2c(0.75) + c(0.5)}{0.25^2} - 2c(0.75) = 0$$

$$x = 1.0 \qquad c(1) = 1$$

c(0)=c(0.25)=0.546, c(0.5)=0.615, c(0.75)=0.760, c(1)=1

Example: solve the BVP using FD technique

$$\frac{d^2T}{dx^2} = 0.01(T-20)^{1.4} \qquad 0 \le x \le 10$$

$$T(0) = 200, \qquad T(10) = 40$$

$$\frac{T_{i+1} - 2T_i + T_{i-1}}{h^2} = 0.01(T_i - 20)^{1.4}$$

$$T_{i+1} - 2T_i + T_{i-1} - h^2 * 0.01(T_i - 20)^{1.4} = 0$$

```
function f=bvp_fd(t)
f(1)=t(2)-2*t(1)+200-0.01*(t(1)-20)^1.4;
for n=2:8
  f(n)=t(n+1)-2*t(n)+t(n-1)-0.01*(t(n)-20)^1.4;
end
f(9)=40-2*t(9)+t(8)-0.01*(t(9)-20)^1.4;
```

h=1

```
% Finite differnce method for solving
%nonlinear BVP
% Initial conditions
% nine nodes
to=[190 180 160 140 120 100 80 60 50]
xx=[1 2 3 4 5 6 7 8 9]'
t=fsolve('bvp_fd',to)
```

MATLAB command

BVP4C Solve two-point boundary value problems for ODEs by collocation.

SOL = BVP4C(ODEFUN,BCFUN,SOLINIT) integrates a system of ordinary differential equations of the form y' = f(x,y) on the interval [a,b],

- Example
- solinit = bvpinit([0 1 2 3 4],[1 0]);
- sol = bvp4c(@twoode,@twobc,solinit);
- solve a BVP on the interval [0,4] with differential equations and
- boundary conditions computed by functions twoode and twobc, respectively.
- This example uses [0 1 2 3 4] as an initial mesh, and [1 0] as an initial
- approximation of the solution components at the mesh points.
- xint = linspace(0,4);
- yint = deval(sol,xint);
- evaluate the solution at 100 equally spaced points in [0 4]. The first
- component of the solution is then plotted with
- plot(xint,yint(1,:));

$$\frac{d^2T}{dx^2} = 0.01(T-20)^{1.4} \qquad 0 \le x \le 10$$

$$T(0) = 200, \qquad T(10) = 40$$

```
function dydx = ge501pvb(x,y)

dydx = [ y(2); 0.01*(y(1)-20)^1.4 ];
```

```
function res = ge501_bc(ya,yb)
res = [ ya(1)-200;   yb(1)-40 ];
```

```
solin=bvpinit([0 2 4 6 8 10],[50 50]);

 sol=bvp4c(@ge501pvb,@ge501_bc,solin);
 xint=linspace(0,10);
 yint=deval(sol,xint);
 plot(xint,yint(1,:))
```

HW

- Solve the problem of flow of a Newtonian flow fluid in a pipe described by the equation:
- Compare the solutions obtained using the shooting method, finite difference and MATLAB bvp4c solver

$$\frac{d^2v}{dr^2} = -\frac{1}{\mu}\frac{\Delta p}{L} - \frac{1}{r}\frac{dv}{dr}$$

$$\frac{dv}{dr}(r=0) = 0, \qquad v(r=R) = 0$$

$$R = 0.0025$$

$$\Delta p = 2.8 \times 10^5$$

$$\mu = 0.492$$

$$L = 4.88$$