Name :Shamsulhaq Basir

HW #8 ROM

1) data is generated as in the following pseudo code

```
n          = 5;
xi1        = linspace(-1,1,n);
xi2        = linspace(-1,1,n);
[Xi1,Xi2] = meshgrid(xi1,xi2);
A =[];
for j=1:n
    for i=1:n

        eta(1,1) = Xi1(j,i);
        eta(1,2) = Xi2(j,i);
        u        = SolveState(t,u0,nu,eta);
        A        = [A u];

    end
end
```

2)

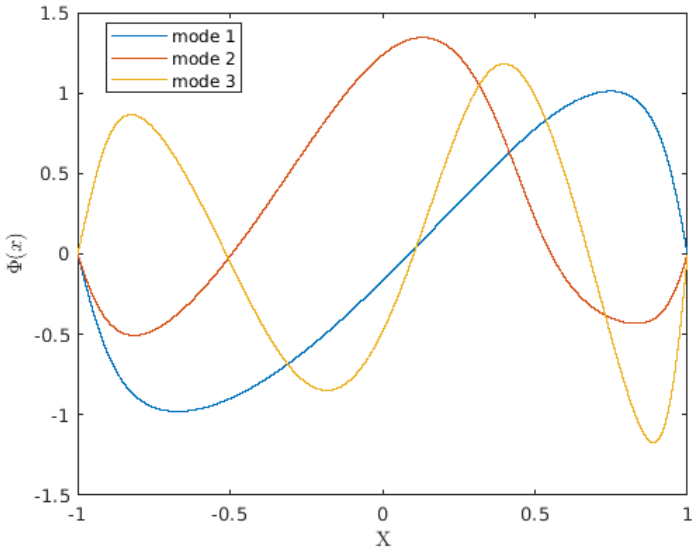


Figure 1 : Three dominant POD modes

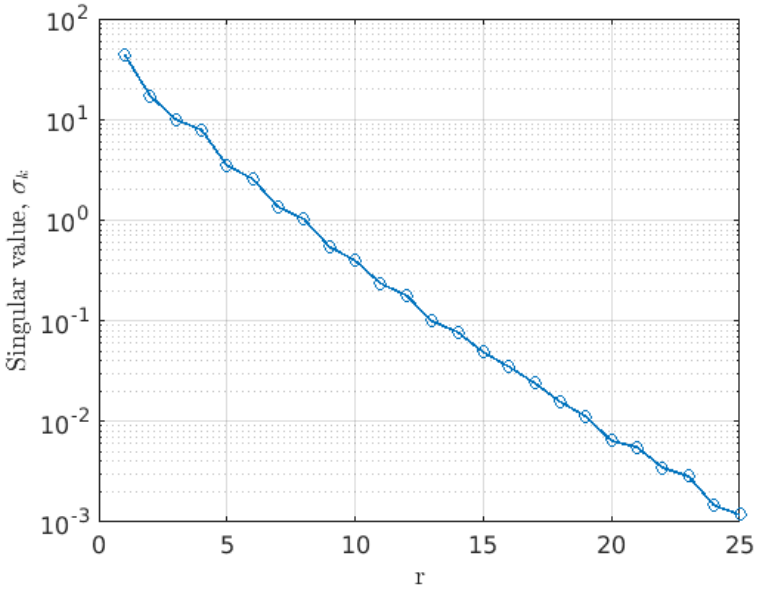


Figure 2: Singular values associated with the first dominant 25 modes

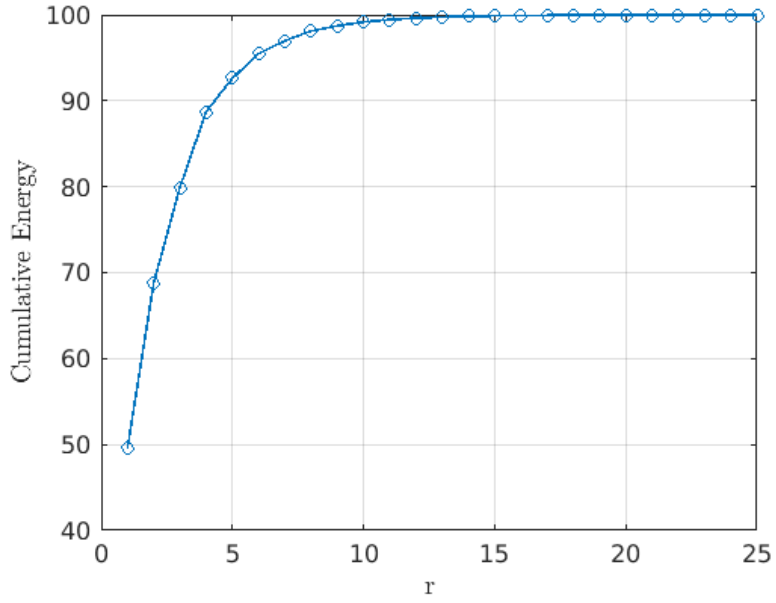


Figure 3: cumulative energy of the modes.

From figure (3) it was computed that 99.99 % of the energy is captured with 20 POD modes.

3) Following is the derivation of the ROM model from the governing equation (1) in terms of the POD basis (2).

$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial^2 x} + f(x, \eta_1, \eta_2)$, where $u = u(x, t; \eta)$ ----- (1),

$u(0) = u(\eta) = 0, x \in [0, L]$

We construct the ROM

$u(x, t, ; \eta) = \sum y_i(t)\Phi_i(x) + \epsilon(x, t; \eta)$, $< \Phi_i(x), \Phi_j(x) > = \delta_{ij}$----(2)

$\dot{y_J}\Phi_j + (y_i\Phi_i)\left(\frac{y_k d\Phi_k}{dx}\right) = \nu y_j \frac{d^2\Phi_j}{dx^2} + f$

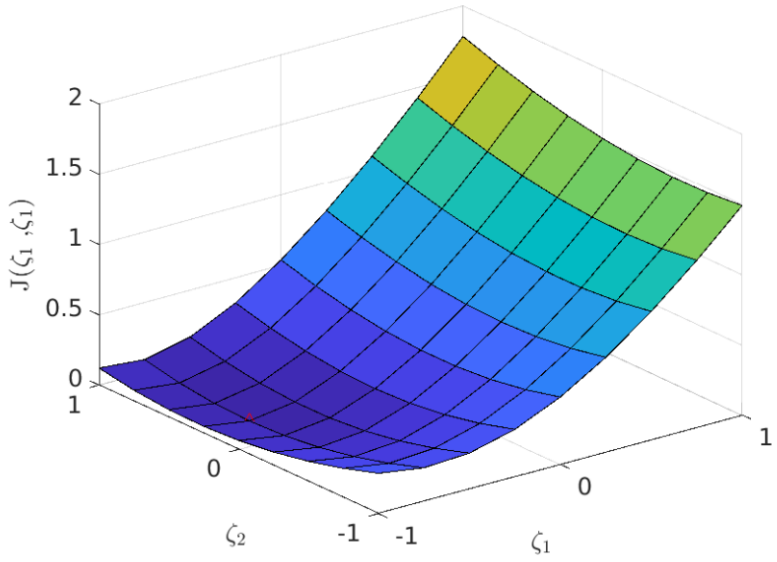
$< \dot{y_J}\Phi_j + (y_i\Phi_i)\left(\frac{y_k d\Phi_k}{dx}\right), \Phi_i > = < \nu y_j \frac{d^2\Phi_j}{dx^2} + f , \Phi_i >$

$\frac{dy_i(t;\eta)}{dt} = -\nu D_{ij} y_j(t;\eta) - N_{ijk} y_i(t;\eta) y_k(t;\eta) + b_i(t;\eta)$

Where $N_{ijk} = < \Phi_j \frac{d\Phi_k}{dx} \Phi_i >$, $D_{ij} = < \frac{d\Phi_i}{dx}, \frac{d\Phi_j}{dx} >$

4)

Because of the computation time of ROM model, the surface plot is obtained for (9x9) eta space. However, the response surface can be obtained for any domain of eta. The optimum is obtained as (-0.75,0.25) as can be seen in figure (4)



```
function [u]=SolveState(t,u0,nu,eta)
dom = [-1 1];
x = chebfun('x',dom);
opts     = pdeset('Eps', 1e-4, 'Ylim', [-40,40]);
pdefun   = @(t,x,u) -u.*diff(u) + nu.*diff(u,2) +f(t,x,eta);
bc.left  = {'dirichlet', 0};
bc.right = {'dirichlet', 0};
u        = pde15s(pdefun,t,u0,bc,opts);
end


function f=f(t,x,eta)
f= eta(1,1)*sin(pi*x-t) + eta(1,2)*sin(2*pi*x-3*t);
end


function [J]= Jcalc(u,ud,dt)
% calculate the objective function J(eta1,eta2)

Udiff = u-ud;
Nt    = size(Udiff,2);
Ix  = zeros(1,1); % integration at constant time along x
for i = 1:Nt
    % abs is taken because (u-ud)^2
    I_loc  = Udiff(:,i)'*Udiff(:,i); % chebfun takes the integral
    Ix(i) = abs(I_loc);
end
J = sum(dt*(Ix(2:end)+Ix(1:end-1))/2);
end



% Shamsulhaq Basir
% HW #8
clear ;  close  ; clc
LW = 'linewidth'; FS = 'fontsize'; IN = 'interpret'; LT = 'latex';
%% ------------- Space-Time Domains -----------

Tf  = 2.0; dt = 0.01;
t   = 0.0:dt:Tf;
nu  = 0.05;
dom = [-1 1];
%% ---------------- Initial state ------------
x   = chebfun(@(x) x, dom);
% Initial condition.
u0  = sin(pi*x);

%% --------------- POD Computation ---------
n = 5;
xi1      = linspace(-1,1,n); xi2= linspace(-1,1,n);
[Xi1,Xi2] = meshgrid(xi1,xi2);
A =[];
for j=1:n
    for i=1:n

        eta(1,1) = Xi1(j,i);
```

```
        eta(1,2) = Xi2(j,i);
        u        = SolveState(t,u0,nu,eta);
        A        = [A u];

    end
end
eta_d(1,1) = -0.75;
eta_d(1,2) =  0.25;
u_d = SolveState(t,u0,nu,eta_d);
[U,S,V] = svd(A);

sigma = diag(S);
sigma = sigma(1:25);
figure
semilogy(sigma,'-o',LW,1.2)
grid on;
set(gca,FS,12)
ylabel("Singular value, $\sigma_k$",IN,LT);
xlabel("r");



figure()
plot(cumsum(sigma)/sum(sigma)*100,'-o',LW,1.2)
grid; set(gca,FS,12)
ylabel("Cumulative Energy");
xlabel("r");

for i = 1:numel(sigma)
    percentage = 100 - cumsum(sigma(i))/sum(sigma)*100;
    if(percentage >= 99.99)
        r = i;
        break;
    end
end

%-- 3 POD modes
figure()
plot(U(:,1));
hold on
plot(U(:,2));
plot(U(:,3));
xlabel('X');
ylabel('$\Phi(x)$');
legend('mode 1','mode 2','mode 3');

save data


%% ------------------------------------------------------------------------

load data
%% (4) Response Surface
Tf = 2.0; dt = 0.01;
t = 0.0:dt:Tf;
snap = numel(t);
```

```
nu = 0.05;
dom = [-1 1]; x = chebfun('x',dom);
u0 = sin(pi*x);

Ubase   = U(:,1:r);                                % Basis
Ubase_p = diff(Ubase);                             % Derivative of Basis
Mik     = Ubase'*Ubase;                            % Mass matrix = I
y0      = (u0'*Ubase);                             % projection coefficients, t = 0
y0      = y0';
Dik     = -Ubase_p'*Ubase_p;                       % D matrix


dimEta = 9;
xi1=linspace(-1,1,dimEta);
xi2=linspace(-1,1,dimEta);
[Xi1,Xi2]= meshgrid(xi1,xi2);
J = zeros(1,1);

for j=1:dimEta
    for i=1:dimEta

        eta(1,1) = Xi1(i,j);
        eta(1,2) = Xi2(i,j);

        dydt = @(t,y) nu*Dik*y-Ubase'*((Ubase*y).*(Ubase_p*y))+Ubase'*f(t,x,eta);
        [t,y]=ode23(@(t,y) dydt(t,y), t, y0);
        u_rom = Ubase*y';                          % reduced order model solution
        J(i,j) =Jcalc(u_rom, u_d,dt);
    end
end
% find the minimum of J
[nx,ny] = find(J == min(min(J)));

%% Response surface
figure
surf(Xi1,Xi2,J)
et1 = Xi1(nx,ny);
et2 =Xi2(nx,ny);
hold on
plot3(et1,et2,J(nx,ny),'^','color','r');
xlabel('$\zeta_{1}$',FS,12);
ylabel('$\zeta_{2}$',FS,12);
zlabel('J($\zeta_{1}$ ,$\zeta_{1}$)',FS,12);
set(gca,FS,12);
%% optimum point
disp("Optimum is : ");
fprintf("(%2.3f, %2.3f) \n", Xi1(nx,ny),Xi2(nx,ny));
```