

```
function [t,z,radius] = Q5_Realization(r0,sigma)
% generate a random disturbance and dynamics realization
```

```
global f
% How do I "scale" these?
R = 0.5;
I = 1;
m = 1;
c = 10;
k = 50;
% r0 = 0.5;
% sigma = 0.1;
f = R/2;
t0 = 0;
tf = 1;
tspan = [t0 tf];

% Geometry
R1 = 2*R/pi;
R2 = R-2*R/pi;
rho = norm([R1,R2]);
rho1 = [-R2; R1];
rho2 = [R1; -R2];
phi1 = atan(R2/R1);
phi2 = atan(R1/R2);

% Initial Conditions
x0 = 0;    xd0 = 0;
y0 = 0;    yd0 = 0;
a0 = 0;    ad0 = 0;
z0 = [x0 xd0 y0 yd0 a0 ad0]';
```

```
% Disturbance Input
Dt = linspace(t0,tf,100)';
D = zeros(length(Dt),2);
% Random Variables
r = sigma*randn(length(Dt),1) + r0;
theta = 2*pi*rand(length(Dt),1);
% impulse input
D(1,:) = [r(1)*cos(theta(1)), r(1)*sin(theta(1))];
% continuous random input
% D = [r.*cos(theta), r.*sin(theta)];
```

```

% Control Input
Ut = linspace(t0,tf,100)';
U = zeros(length(Ut),2);

% integrate state equations
options=odeset('RelTol',1e-6); % set tolerances for integration
[t,z] = ode45(@state_eqns,tspan,z0,options,Dt,D,Ut,U);

x      = z(:,1);    % CoM x-position
xd     = z(:,2);    % CoM x-velocity
y      = z(:,3);    % CoM y-position
yd     = z(:,4);    % CoM y-velocity
alpha  = z(:,5);    % mirror angular displacement
alphad = z(:,6);    % mirror angular rate

% mirror vertex position
x2_nom = rho2(1);
y2_nom = rho2(2);
x2     = x2_nom + x + rho*alpha*cos(phi2);
y2     = y2_nom + y + rho*alpha*sin(phi2);

% focal point position
xfoc_nom = x2_nom;
yfoc_nom = y2_nom + f;
xfoc     = x2 + f*sin(alpha);
yfoc     = y2 + f*cos(alpha);

%% Calculate radius of focal point evolution ball centered on the nominal
d = zeros(length(t),1);
for i = 1:length(t)
    d(i) = norm([xfoc_nom - xfoc(i), yfoc_nom - yfoc(i)]);
end

radius = max(d);

%% nested function with state derivatives
function zdot = state_eqns(t,z,Dt,D,Ut,U)
% state equations for qualifying exam question 5, problem 1

zdot = zeros(6,1);

Ux = U(:,1);
Ux = interp1(Ut,Ux,t);

```

```
Uy = U(:,2);
Uy = interp1(Ut,Uy,t);

Dx = D(:,1);
Dx = interp1(Dt,Dx,t);
Dy = D(:,2);
Dy = interp1(Dt,Dy,t);

x      = z(1);
xd     = z(2);
y      = z(3);
yd     = z(4);
alpha  = z(5);
alphad = z(6);

x1 = x - rho*alpha*cos(phi1);
x1d = xd - rho*alphad*cos(phi1);
x2 = x + rho*alpha*cos(phi2);
x2d = xd + rho*alphad*cos(phi2);
y1 = y - rho*alpha*sin(phi1);
y1d = yd - rho*alphad*sin(phi1);
y2 = y + rho*alpha*sin(phi2);
y2d = yd + rho*alphad*sin(phi2);
x0 = x1 + Ux;
x0d = x1d;
y0 = y1 + Uy;
y0d = y1d;

zdot(1) = xd;
zdot(2) = (1/m)*(-c*x0d - k*x0 - c*x2d - k*x2 + Dx);
zdot(3) = yd;
zdot(4) = (1/m)*(-c*y0d - k*y0 - c*y2d - k*y2 + Dy);
zdot(5) = alphad;
zdot(6) = (1/I)*(R1*(c*x0d + k*x0 - c*y2d - k*y2 + Dy)...
             + R2*(-c*x2d - k*x2 + Dx + c*y0d + k*y0));

end

end
```