Episode 5

Newton's Laws: Part 2
Predicting motion of particles

ENGN0040: Dynamics and Vibrations
Allan Bower, Yue Qi

School of Engineering
Brown University

Topics for todays class

Predicting the motion of a particle (Mostly Matlab)

1. Trajectory equations
2. Model suspension system
3. Centrifugal pump
4. Flying squirrel
5. Earthquake response of a building
6. Quadrupole filter

3.3 Predicting motion of particles subjected to known forces

General Approach:

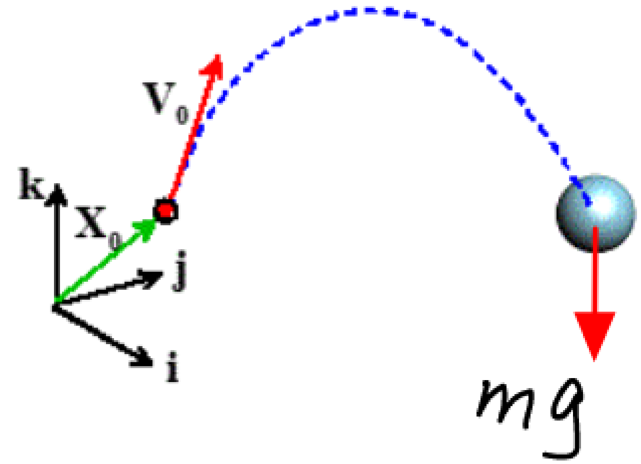
(1) $\underset{\sim}{F} = m\underset{\sim}{a}$ → Differential equation
(2) Solve (separate variables or MATLAB)

Illustrate with examples

3.3.1: Example: Predict motion of a projectile, launched from known initial position with known initial velocity. Neglect air resistance (for now)

Initial position and velocity

$$\left.\begin{array}{l} \mathbf{r} = X_0\mathbf{i} + Y_0\mathbf{j} + Z_0\mathbf{k} \\ \dfrac{d\mathbf{r}}{dt} = V_{x0}\mathbf{i} + V_{y0}\mathbf{j} + V_{z0}\mathbf{k} \end{array}\right\} t = 0$$



$$\underline{F} = m\underline{a} \Rightarrow -mg\underline{k} = m(a_x\underline{i} + a_y\underline{j} + a_z\underline{k})$$

Hence $a_x = 0 \Rightarrow \dfrac{dv_x}{dt} = 0 \Rightarrow \boxed{v_x = V_{x0}}$

Similarly $a_y = 0 \Rightarrow \boxed{v_y = V_{y0}}$

$$a_z = -g \Rightarrow \frac{dv_z}{dt} = -g \Rightarrow \boxed{v_z = V_{z0} - gt}$$

Now $\frac{dx}{dt} = v_x = V_{x0} \Rightarrow \boxed{x = X_0 + V_{x0} t}$

$$\frac{dy}{dt} = v_y = V_{y0} \Rightarrow \boxed{y = Y_0 + V_{y0} t}$$

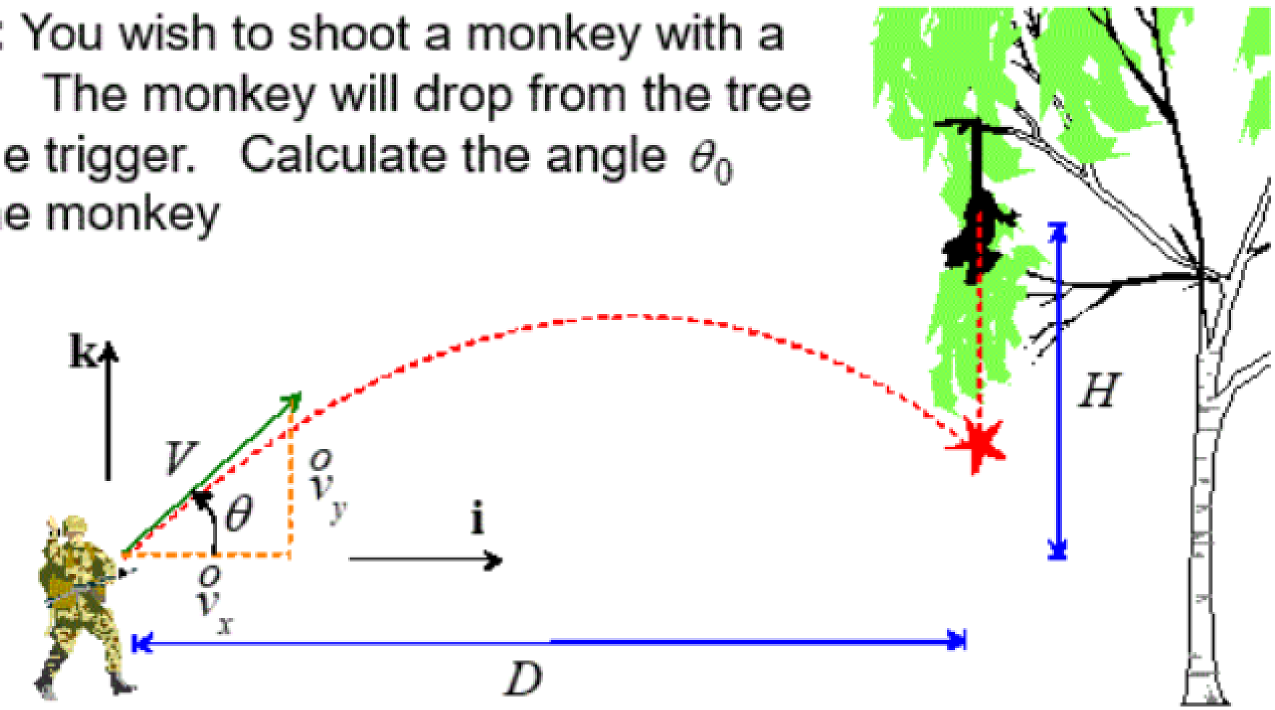
$$\frac{dz}{dt} = v_z = V_{z0} - gt \Rightarrow \boxed{z = Z_0 + V_{z0} t - \frac{1}{2} g t^2}$$

$$\boxed{\underline{v} = V_{x0}\,\underline{i} + V_{y0}\,\underline{j} + (V_{z0} - gt)\,\underline{k}}$$

$$\boxed{\underline{r} = (X_0 + V_{x0} t)\,\underline{i} + (Y_0 + V_{y0} t)\,\underline{j} + (Z_0 + V_{z0} t - \tfrac{1}{2} g t^2)\,\underline{k}}$$

"Trajectory equations"

3.3.2: Example: You wish to shoot a monkey with a tranquilizer dart. The monkey will drop from the tree when you pull the trigger. Calculate the angle θ_0 required to hit the monkey



Approach: monkey & dart have same position at impact $\Rightarrow$ $\underline{r}_m = \underline{r}_D$ - solve for θ

Trajectory eqs: $\underline{r}_m = D\underline{i} + (H - \frac{1}{2}gt^2)\underline{k}$

$\underline{r}_D = V\cos\theta\, t\, \underline{i} + (V\sin\theta\, t - \frac{1}{2}gt^2)\underline{k}$

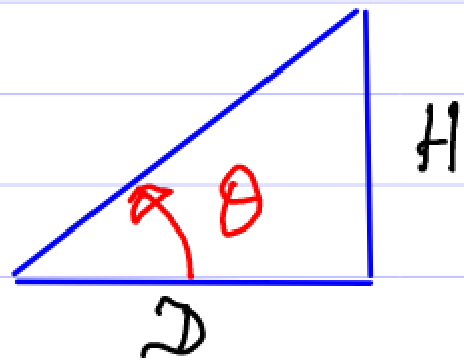
$(\underline{i}, \underline{k})$ components must agree

$$V\cos\theta\, t = D \qquad (1)$$

$$V\sin\theta\, t - \tfrac{1}{2}\cancel{g}t^2 = H - \tfrac{1}{2}\cancel{g}t^2 \qquad (2)$$

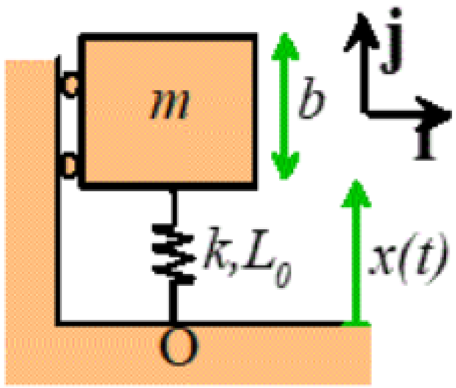
$(2)/(1) \Rightarrow \tan\theta = H/D$

Interpret graphically :

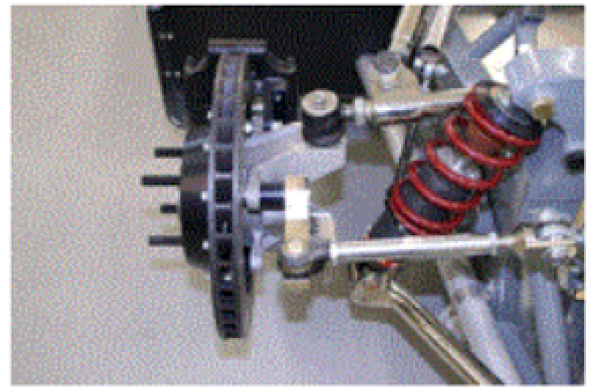


$\boxed{\text{Aim directly at monkey?}}$

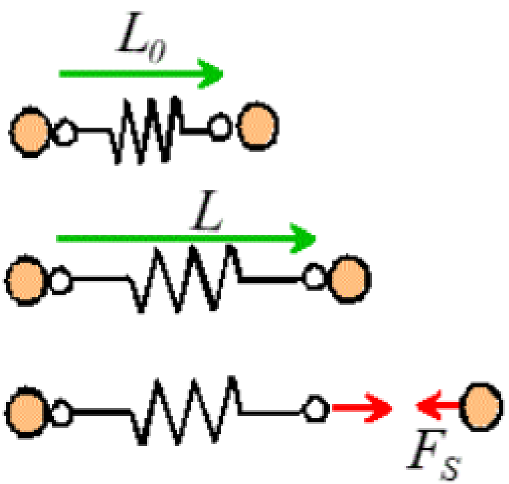
3.3.3: Example: A car suspension is idealized as a spring-mass system. The car is set vibrating by releasing the body from rest with the spring un-stretched. Calculate $x(t)$



$$\left.\begin{aligned} x &= L_0 \\ \frac{dx}{dt} &= 0 \end{aligned}\right\} t = 0$$



Preliminary: Forces exerted by springs



$$F_s = k \underbrace{(L - L_0)}_{\text{Increase in length}}$$

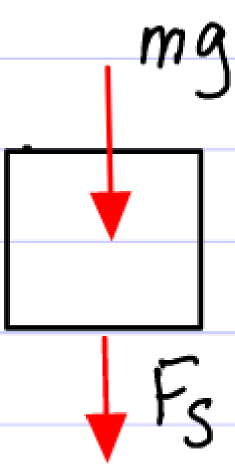
k – stiffness

L_0 – unstretched length

Assume springs pull on FBDs

Accel & Velocity : $v = \frac{dx}{dt}$ $\quad a = \frac{dv}{dt} = \frac{d^2x}{dt^2}$

FBD



$F = ma \Rightarrow ma = -mg - F_s$

Spring force law $F_s = k(x - L_0)$

Combine :

$$\boxed{m\frac{dv}{dt} = -k(x - L_0) - mg}$$

or

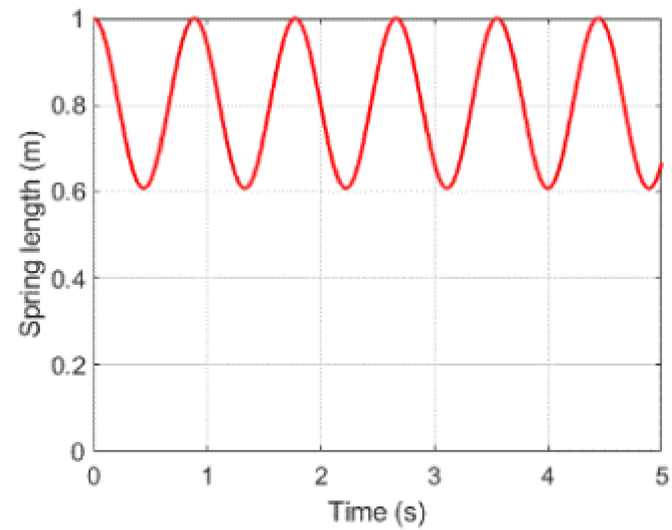
$$\boxed{m\frac{d^2x}{dt^2} = -k(x - L_0) - mg}$$

Solve with "Live Script" or ODE45

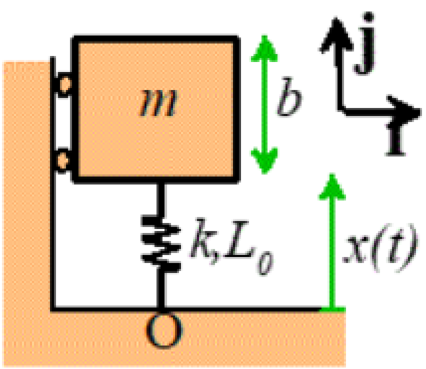
```
syms m t k L0 g real
syms x(t) v(t)
assume(k>0); assume(m>0);
diffeq = m*diff(x(t),t,2) == -k*(x(t) - L0) - m*g;
v(t) = diff(x(t),t);
IC = [x(0)==L0,v(0)==0];
sol = simplify(dsolve(diffeq,IC,symvar('x(t)')))
sol_with_nums = subs(sol,[L0,k,m,g],[1,10^5,2000,9.81]);
fplot(sol_with_nums,[0,5],'Linewidth',2,'Color',[1,0,0])
ylim([0 1])
set(gca,'FontSize',14)
xlabel('Time (s)')
ylabel('Spring length (m)')
grid on
```

sol =

$$\frac{L_0\, k - g\, m + g\, m \cos\left(\frac{\sqrt{k}\, t}{\sqrt{m}}\right)}{k}$$

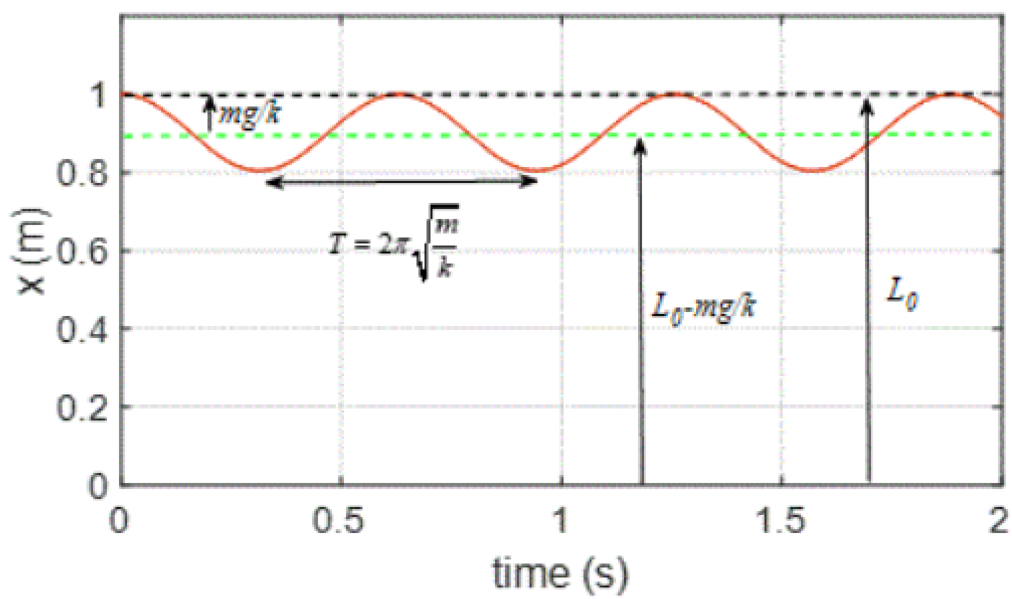


Summary of ‘Live Script’ solution



$$\left.\begin{aligned} x &= L_0 \\ \frac{dx}{dt} &= 0 \end{aligned}\right\} t = 0$$

$$x = \left(L_0 - \frac{mg}{k}\right) + \frac{mg}{k}\cos\sqrt{\frac{k}{m}}t$$



Solution with "ode45"

ode45 needs numbers for constants.

Assume:

$k = 10^5\ N/m \quad m = 2000\ kg \quad L_0 = 1m$

$g = 9.81\ m/s^2 \quad 0 < t < 5s$

Re-write diff eqs in form $\frac{dw}{dt} = f(t, w)$

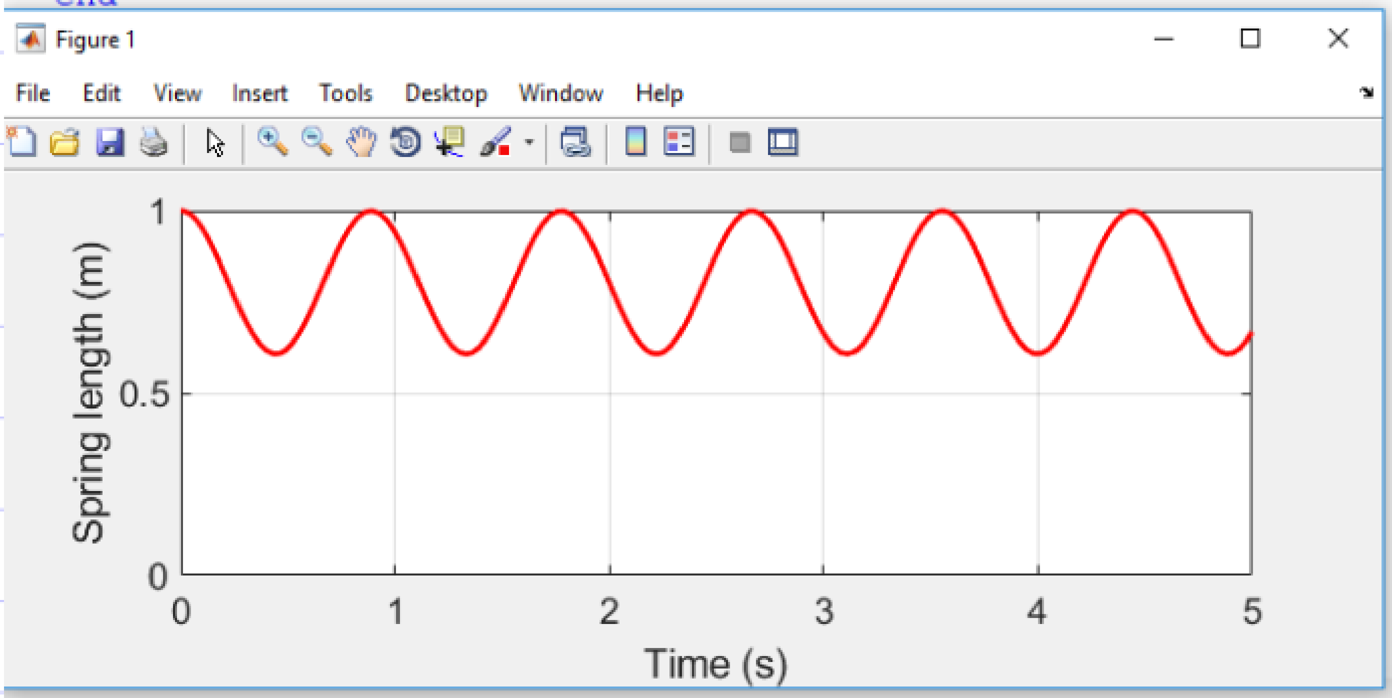
Unknowns $w = [x, v]$ (MATLAB vector)

Diff eqs:

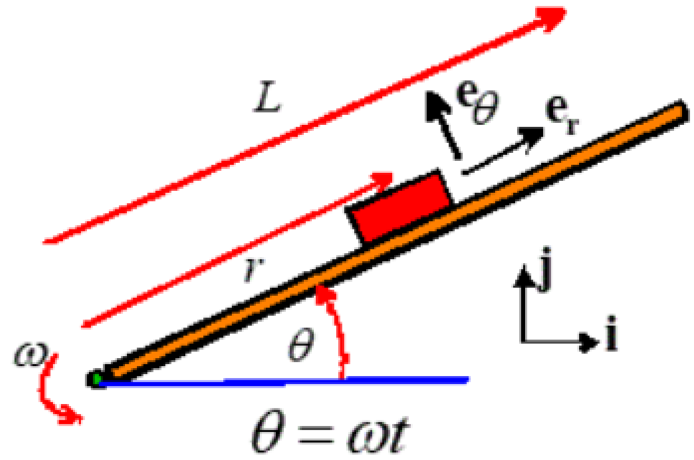
$$\frac{d}{dt}\begin{bmatrix} x \\ v \end{bmatrix} \qquad \boxed{\begin{matrix} \frac{dx}{dt} \\ \frac{dv}{dt} \end{matrix}} = \boxed{\begin{matrix} v \\ -k(x - L_0)/m - g \end{matrix}}$$

$$\equiv \frac{dw}{dt} \qquad\qquad f(t, w)$$

```
function suspension
  close all
  L0=1; k=1.e05; m=2000; g=9.81;
  tspan = [0,5];
  initial_w = [L0,0];

  [t_vals,sol_vals] = ode45(@(t,w) eom(t,w,k,L0,m,g),tspan,initial_w);
  plot(t_vals,sol_vals(:,1),'Linewidth',2,'Color','r');
  ylim([0 1]);
  set(gca,'FontSize',14);
  xlabel('Time (s)');
  ylabel('Spring length (m)');
  grid on
  pause(3)
  animatespringmass(t_vals,sol_vals,L0);
end
function dwdt = eom(t,w,k,L0,m,g)
   x = w(1); v = w(2);
   dxdt = v;
   dvdt = -k*(x-L0)/m - g;
   dwdt = [dxdt;dvdt];
end
```



3.3.4: Example: (Crude model of centrifugal pump) The rod rotates with constant angular speed ω . The mass starts at position *r=0* with radial speed $dr/dt = V_0$. Find a formula for *r* as a function of time



NEGLECT FRICTION AND GRAVITY

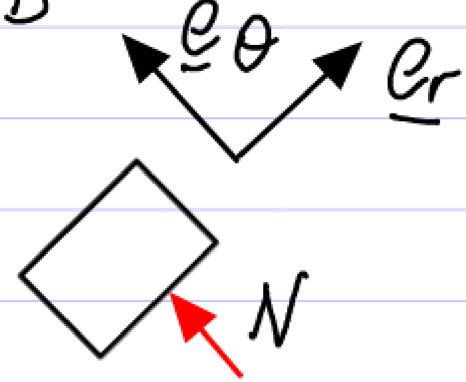
Approach: $\underline{F} = m\underline{a}$ in polar coords

Acceleration (from Episode 3)

$$\underline{a} = \left\{ \frac{d^2r}{dt^2} - r\left(\frac{d\theta}{dt}\right)^2 \right\} \underline{e_r} + \left\{ r\frac{d^2\theta}{dt^2} + 2\frac{dr}{dt}\frac{d\theta}{dt} \right\} \underline{e_\theta}$$

Here $\theta = \omega t \Rightarrow \frac{d\theta}{dt} = \omega \quad \frac{d^2\theta}{dt^2} = 0$

FBD



$\underline{F} = m\underline{a}$:

$$N\underline{e}_\theta = m\left(\left\{ \frac{d^2r}{dt^2} - r\omega^2 \right\} \underline{e}_r + \left\{ 2\frac{dr}{dt}\omega \right\} \underline{e}_\theta \right)$$

$\underline{e}_r$: $\boxed{\dfrac{d^2r}{dt^2} - r\omega^2 = 0}$ Diff eq for r

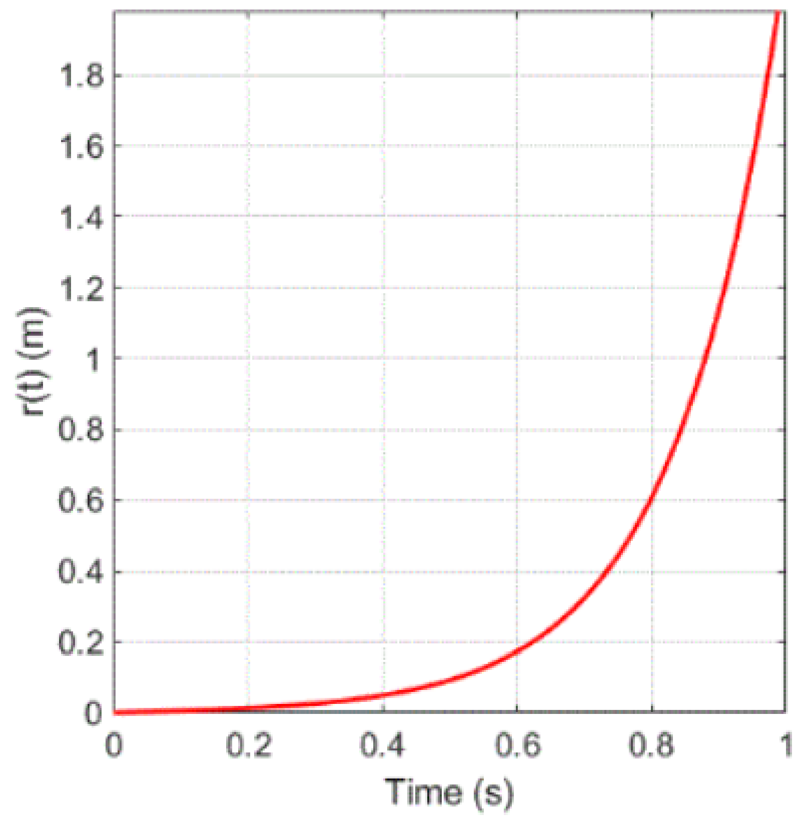
Initial Conditions: $r = 0$ $\quad dr/dt = V_0$ @ $t = 0$

Solve with "Live Script"

```
syms omega V0 real
syms r(t) v(t)
assume(omega>0); assume(V0>0);
diffeq = diff(r(t),t,2) - r(t)*omega^2 == 0;
v(t) = diff(r(t),t);
IC = [r(0)==0,v(0)==V0];
sol = simplify(dsolve(diffeq,IC,symvar('r(t)')))
sol_with_nums = subs(sol,[omega,V0],[2*pi,0.05]);
fplot(sol_with_nums,[0,1],'Linewidth',2,'Color',[1,0,0])
%ylim([0 1])
set(gca,'FontSize',14)
xlabel('Time (s)')
ylabel('r(t) (m)')
grid on
```

sol =

$$\frac{V_0 \sinh(\omega t)}{\omega}$$



Solution with ode45

ODE: $d^2r/dt^2 = r\omega^2$

Problem: rewrite in form $dw/dt = f(t,w)$

Solution: define $\dfrac{dr}{dt} = V$

Solve for $w = [r, V]$

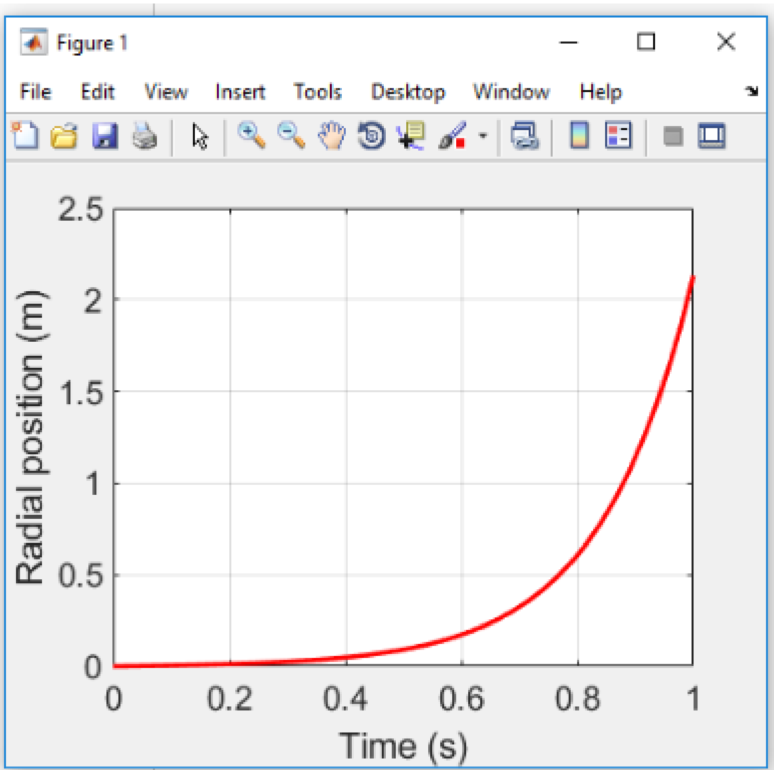
Differential eqs:

$$\begin{aligned} dr/dt &= V \\ dV/dt &= r\omega^2 \end{aligned} \equiv \frac{d}{dt}\begin{bmatrix} r \\ V \end{bmatrix} = \begin{bmatrix} V \\ r\omega^2 \end{bmatrix}$$

Solve for $\omega = 2\pi$ $\quad 0 < t < 1$

Initial condition $w = [0, 0.05]$ at time $t = 0$

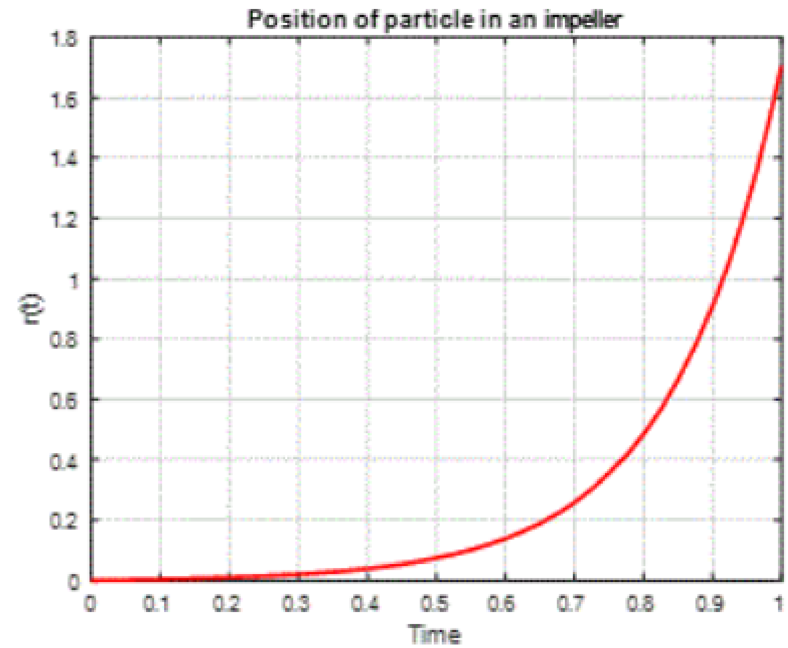
```
function pump
  close all
  omega = 2*pi;
  tspan = [0,1];
  initial_w = [0,0.05];

  [t_vals,sol_vals] = ode45(@(t,w) eom(t,w,omega),tspan,initial_w);
  plot(t_vals,sol_vals(:,1),'Linewidth',2,'Color','r');
  set(gca,'FontSize',14);
  xlabel('Time (s)');
  ylabel('Radial position (m)');
  grid on
  figure
  animate_impeller(t_vals,sol_vals,omega);
end
function dwdt = eom(t,w,omega)
   r = w(1); v = w(2);
   drdt = v;
   dvdt = r*omega^2;
   dwdt = [drdt;dvdt];
end
```



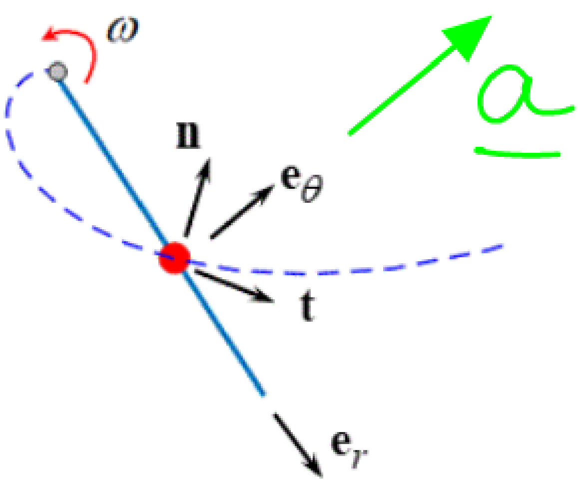
Solution

$$r(t) = \frac{V_0}{\omega} \sinh(\omega t)$$

$$\sinh(x) = \frac{1}{2}\left(e^{x} - e^{-x}\right)$$



Concept Question: What is the direction of the acceleration at the instant shown?

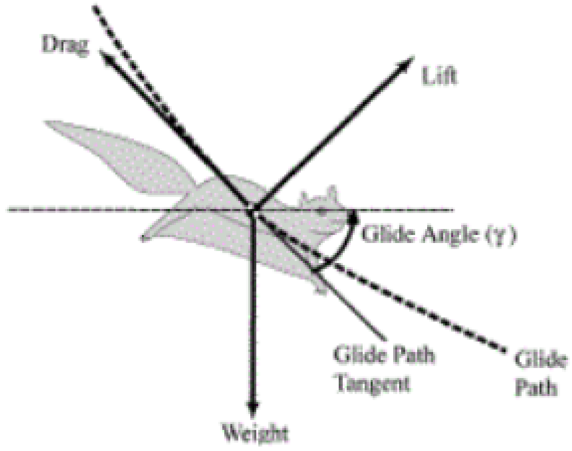


Solution: $\underline{a}$ must be parallel to $\underline{F}$

We know $\underline{F}$ is normal to bar

$\underline{a}$ is parallel to $\underline{e_\theta}$

3.3.5: Example: Calculate the trajectory of a flying squirrel, using the data below. Hence determine how far it flies.



Lift $F_L = c_L V^2$ $c_L = 0.0084 Ns^2/m^2$

Drag $F_D = c_D V^2$ $c_D = 0.00247 Ns^2 m^{-2}$

Mass $m = 0.075 kg$

Initial velocity $V_{x0} = 5.4 m/s$

$V_{y0} = 0$

Initial height 10m

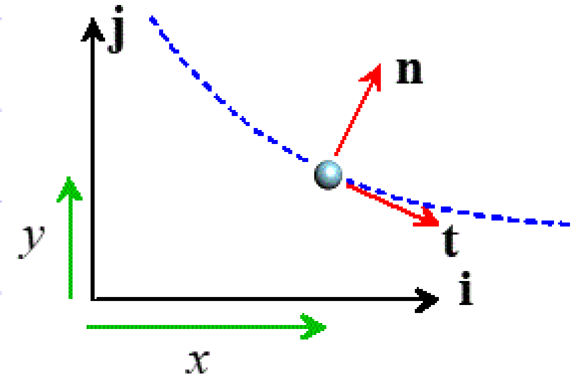
Note: F_D acts tangent to path
F_L acts normal to path

Approach

Calculate trajectory (x, y)

Find x when $y = 0$

Describing Motion



Velocity $\quad v_x = \dfrac{dx}{dt} \qquad v_y = \dfrac{dy}{dt}$

Speed $\quad V = \sqrt{v_x^2 + v_y^2}$

Tangent $\quad$ Recall $\underline{v} = V\,\underline{t} \Rightarrow \underline{t} = \underline{v}\,/\,V$

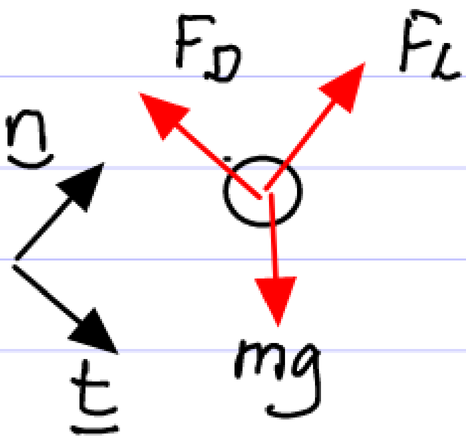
Hence $\quad \underline{t} = (v_x\,\underline{i} + v_y\,\underline{j})\,/\,V$

Normal $\quad \underline{n} = \underbrace{\underline{k} \times \underline{t}} = (-v_y\,\underline{i} + v_x\,\underline{j})\,/\,V$

$\underline{n}$ is perpendicular to $\underline{k}$ & $\underline{t}$

Acceleration $\quad a_x = \dfrac{dv_x}{dt} \qquad a_y = \dfrac{dv_y}{dt}$

FBD



Forces

Drag: $\underline{F}_D = -c_D V^2 \underline{t} = -c_D V (V_x \underline{i} + V_y \underline{j})$

Lift: $\underline{F}_L = c_L V^2 \underline{n} = c_L V (-V_y \underline{i} + V_x \underline{j})$

$\underline{F} = m\underline{a}$

$$\underline{F}_D + \underline{F}_L - mg\underline{j} = m\left(\frac{dV_x}{dt}\underline{i} + \frac{dV_y}{dt}\underline{j}\right)$$

$\underline{i}$ component $\Rightarrow$ $\dfrac{dV_x}{dt} = -C_D V V_x / m - C_L V V_y / m$

$\underline{j}$ " $\Rightarrow$ $\dfrac{dV_y}{dt} = -C_D V V_y / m + C_L V V_x / m - g$

Set up for "ode45"

Unknowns: $w = [x, y, V_x, V_y]$

Time Derivatives

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ V_x \\ V_y \end{bmatrix} = \begin{matrix} V_x \\ V_y \\ -C_D \bar{V} V_x / m - C_L \bar{V} V_y / m \\ -C_D \bar{V} V_y / m + C_L \bar{V} V_x / m - g \end{matrix}$$

Initial Conditions (given)

$w = [0, 10, 5.4, 0]$

Parameters $m = 0.075$ kg $C_D = 0.00247\ Ns^2m^{-2}$

$C_L = 0.0084\ Ns^2/m$

Use "event function" to detect $y = 0$

MATLAB ‘event’ function

(Reference: ENGN40 MATLAB tutorial)

Steps to using an ‘event’ function

Goal: Detect a special value in the solution to an ODE

Procedure:

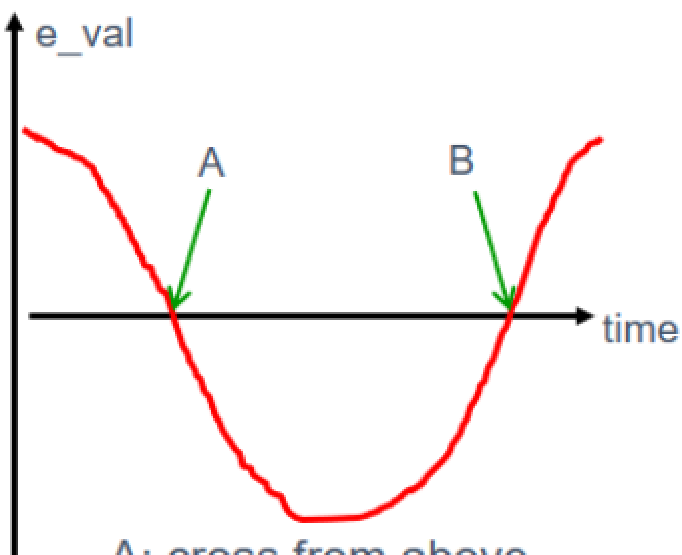
1. Write an ‘Event’ function that informs MATLAB of the special solution values you want to detect
2. Code the ODE solution in the usual way
3. Add an ‘options’ variable to the end of call to ode45 solver to tell MATLAB to use the ‘event’ function

Using an ‘event’ function in ode45

```
options = odeset('Event',@(t,w) event_function(t,w,parameters));

[time_vals,sol_vals] = ode45(@(t,w) eom_function(t,w,parameters), ...
                                                    tspan,options);
```

Format for an ‘event’ function

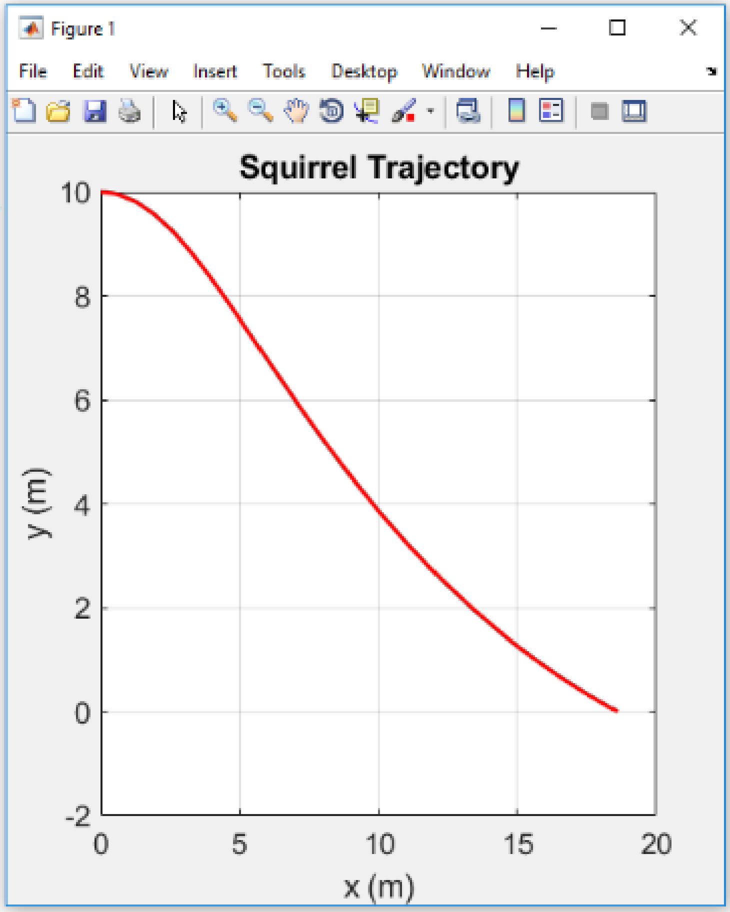
```
function [e_val,stop_flag,event_dir] = event_function(t,w,parameters)
% Extract solution variables out of vector w, eg
  x = w(1); y = w(2); % This is just an example!

% Find a formula that makes e_val go to zero at solution of interest
   e_val = y;  % This will detect y=0 - use any function you like!
% Use 'stop flag' to control MATLAB behavior.
   stop_flag = 1; % Stops MATLAB at solution of interest
%  stop_flag = 0; % This would allow MATLAB to continue after the event
% 'event_dir' Controls the type of zero crossing detected
   event_dir = 0; % Detects all zero crossings.  Can also use +1 or -1
end
```



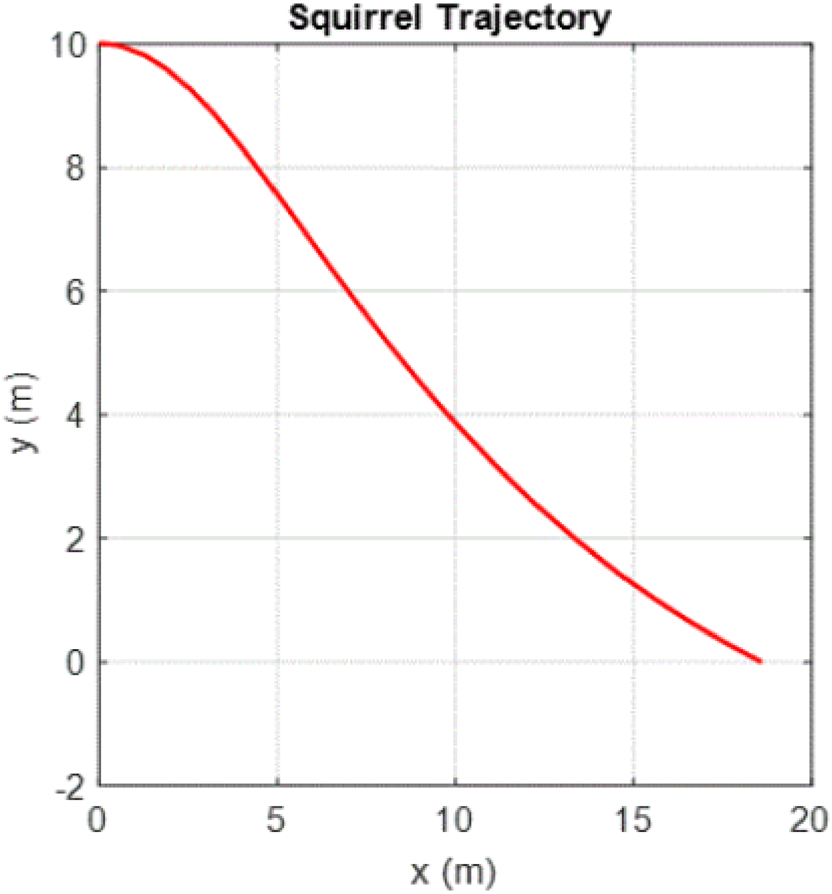
A: cross from above
B: cross from below

event_dir =1 detects B only
event_dir=-1 detects A only
event_dir=0 detects both

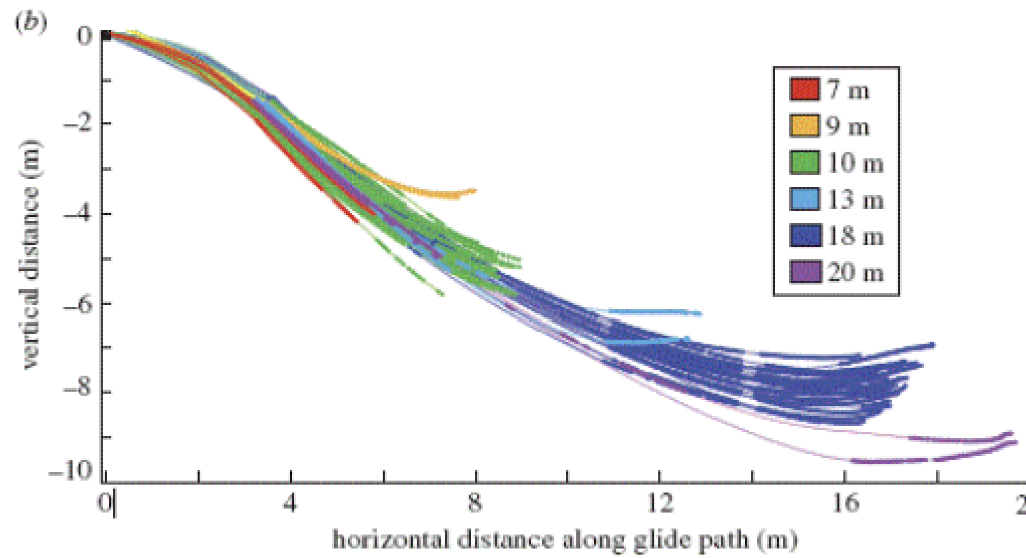
```
function squirrel
  close all
  m = 0.075; cD = 0.00247; cL = 0.0084; g=9.81;
  tspan = [0,5];
  initial_w = [0,10,5.4,0];

  options = odeset('Event',@(t,w) event_function(t,w,m,cD,cL,g))
  [t_vals,sol_vals] = ode45(@(t,w) eom(t,w,m,cD,cL,g),tspan,initial_w,options);
  plot(sol_vals(:,1),sol_vals(:,2),'Linewidth',2,'Color','r');
  set(gca,'FontSize',14);
  xlabel('x (m)');
  ylabel('y (m)');
  title('Squirrel Trajectory');
  grid on
end
function dwdt = eom(t,w,m,cD,cL,g)
   x = w(1); y = w(2); vx = w(3); vy = w(4);
   dxdt = vx;
   dydt = vy;
   V = sqrt(vx^2 + vy^2);
   dvxdt = -cD*V*vx/m - cL*V*vy/m;
   dvydt = -cD*V*vy/m + cL*V*vx/m - g;
   dwdt = [dxdt;dydt;dvxdt;dvydt];
end
function [e_val,stop_flag,event_dir] = event_function(t,w,m,cD,cL,g)
   x = w(1); y = w(2); vx = w(3); vy = w(4);
   e_val = y;
   stop_flag = 1;
   event_dir = -1;
end
```



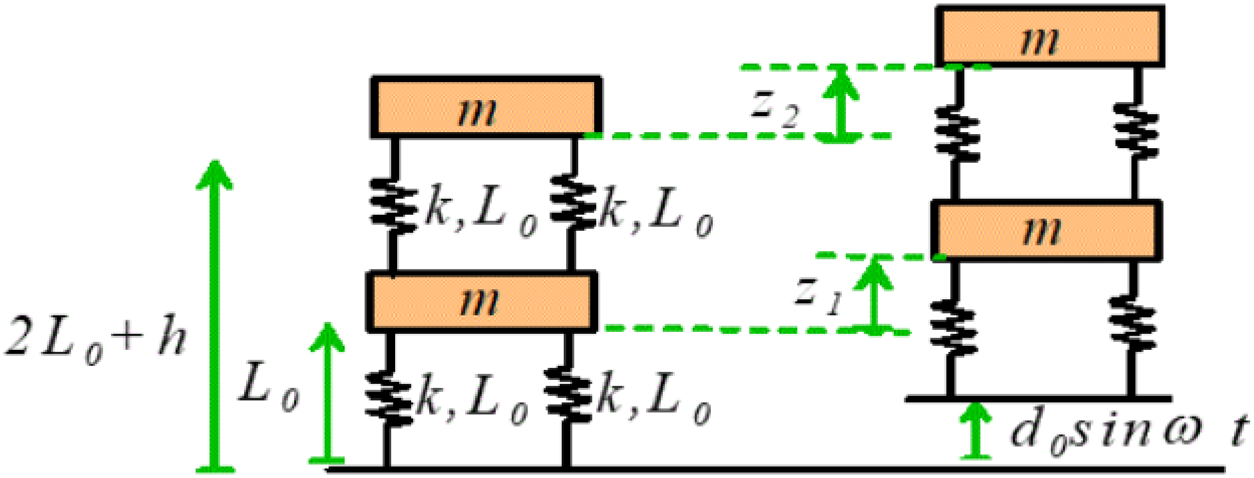
Predicted trajectory



Measurement



3.3.6: Example: Derive and solve the equations of motion for the vertical deflections z_1, z_2 of a base-excited 2 story building. Assume harmonic base excitation with frequency ω and amplitude d_0 **Neglect Gravity**



$$z_1 = z_2 = \frac{dz_1}{dt} = \frac{dz_2}{dt} = 0 \quad \text{at time } t=0$$

Parameters

$L_0 = 2m$

$k = 10^8 N/m$

$m = 25000kg$

$d0 = 0.01m$

Try

$\omega = 55\ rad/s$

$145\ rad/s$

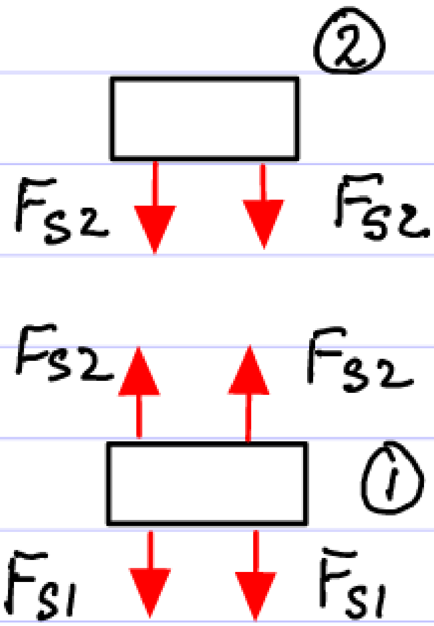
(can try other values as well)

Approach: $\underline{F} = m\underline{a}$ for each mass

→ Derive eqs for z_1, z_2, V_1, V_2

where $V_1 = \frac{dz_1}{dt}$ $V_2 = \frac{dz_2}{dt}$

FBD



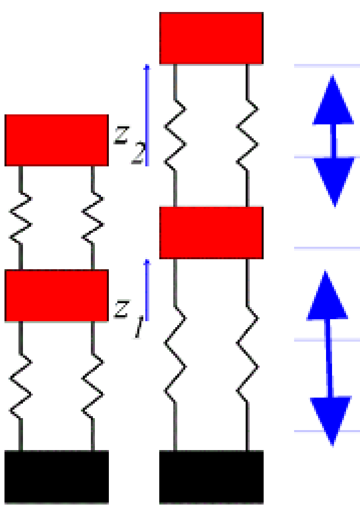
$\underline{F} = m\underline{a}$:

Mass 2 : $$-2F_{s2} = m\frac{dV_2}{dt}$$

Mass 1 :

$$2(F_{s2} - F_{s1}) = m\frac{dV_1}{dt}$$

Spring Force Law : $F_{s1} = k\Delta L_1$ $\quad F_{s2} = k\Delta L_2$



$$\Delta L_2 = z_2 - z_1$$

$$\Delta L_1 = z_1 - d_0 \sin \omega t$$

Set up for solution with ODE45

Unknowns: $[z_1, z_2, V_1, V_2] \equiv w$

Time Derivatives

$$\frac{d}{dt}\begin{bmatrix} z_1 \\ z_2 \\ V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} V_1 \\ V_2 \\ 2(F_{s2} - F_{s1})/m \\ -2F_{s2}/m \end{bmatrix}$$

$F_{s1} = k(z_1 - d_0 \sin \omega t)$ $\quad$ $F_{s2} = k(z_2 - z_1)$

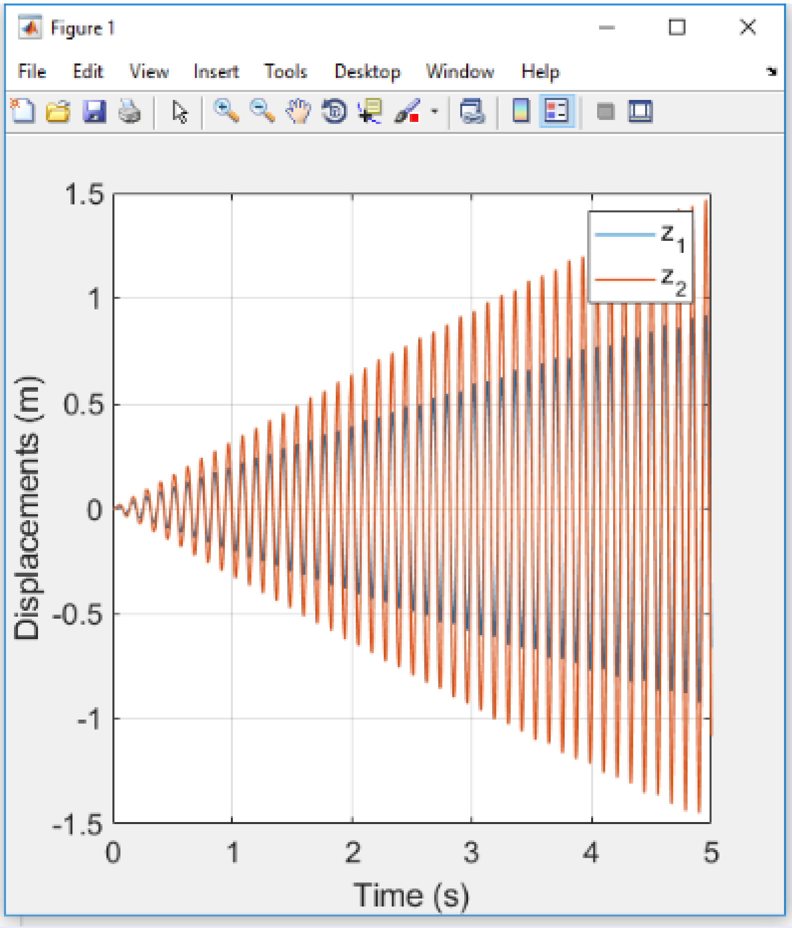
Parameters: $k = 10^8$ N/m

$m = 25000$ kg

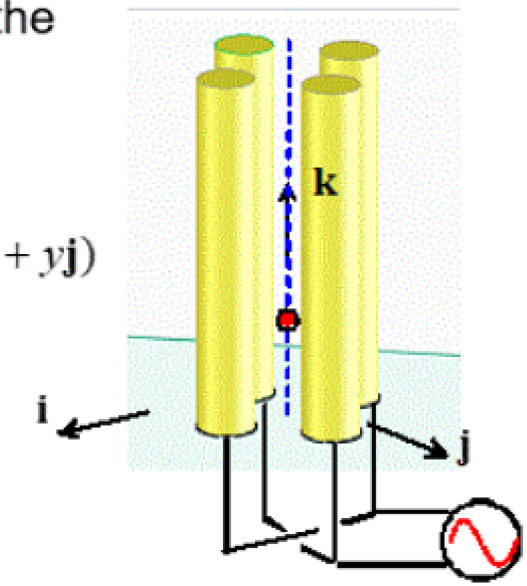
$d_0 = 0.01$ m

Initial Conditions $w = [0, 0, 0, 0]$ @ $t = 0$

```
function building
  close all
  k=1.e08; m=25000; d0 = 0.01; omega = 55;
  tspan = [0,5];
  initial_w = [0,0,0,0];

  [t_vals,sol_vals] = ode45(@(t,w) eom(t,w,k,m,d0,omega),tspan,initial_w);
  plot1 = plot(t_vals,sol_vals(:,1:2),'Linewidth',1);
  set(plot1(1),'DisplayName','z_1');
  set(plot1(2),'DisplayName','z_2');
  set(gca,'FontSize',14);
  xlabel('Time (s)');
  ylabel('Displacements (m)');
  legend(gca,'show');
  grid on
  plot_building(t_vals,sol_vals,d0,omega);
end
function dwdt = eom(t,w,k,m,d0,omega)
   z1 = w(1); z2 = w(2); v1 = w(3); v2 = w(4);
   dz1dt = v1;
   dz2dt = v2;
   Fs1 = k*(z1-d0*sin(omega*t));
   Fs2 = k*(z2-z1);
   dv1dt = 2*(Fs2-Fs1)/m;
   dv2dt = -2*Fs2/m;
   dwdt = [dz1dt;dz2dt;dv1dt;dv2dt];
end
```



3.3.7: Example: A particle with mass m and charge Q travels through the electric field induced by a quadrupole. It is subjected to a force



$$\mathbf{F} = Q\mathbf{E} \qquad \mathbf{E} = E_0 \frac{1+\beta\cos(\omega t)}{d}(x\mathbf{i} + y\mathbf{j})$$

(a) Show that the acceleration of the particle is $\mathbf{a} = \Omega^2(1+\beta\cos(\omega t))(x\mathbf{i}+y\mathbf{j})$

and give a formula for Ω

(b) At time t=0 the particle has position $(0.005, 0.005, 0)$ m and velocity $(0,0,0.01)$ m/s. Predict the path of the particle for $\omega = 10, \beta = 20 \quad \Omega = 1.1$

(c) Find the range of Ω for which the particle will pass through the filter

(a) Use $\underline{F} = m\underline{a} \Rightarrow$

$$\underline{a} = \underline{F}/m = \underbrace{\frac{E_0 Q}{dm}}_{\Omega^2}(1+\beta\cos\omega t)(x\underline{i} + y\underline{j})$$

Hence $\Omega = \sqrt{\dfrac{E_0 Q}{dm}}$

(b) Predict path with MATLAB

Unknowns : $W = [x, y, z, V_x, V_y, V_z]$

Differential eq: for ode45

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ z \\ V_x \\ V_y \\ V_z \end{bmatrix} = \begin{matrix} V_x \\ V_y \\ V_z \\ \Omega^2 (1+\beta \cos \omega t)\, x \\ \Omega^2 (1+\beta \cos \omega t)\, y \\ 0 \end{matrix}$$

Parameters: $\omega = 10 \quad \beta = 20 \quad \Omega = 1.1$

Initial conditions :

$$W = [0.005, 0.005, 0, 0, 0, 0.01]$$

$$0 < t < 100$$

```
function quadrupole
  close all
  Omega = 1.1; omega = 10; beta = 20;
  tspan = [0:0.2:100];
  initial_w = [0.005,0.005,0,0,0,0.01];
  [times,sols] = ode45(@(t,w) eom(t,w,Omega,omega,beta),tspan,initial_w);
  animate_quadrupole(times,sols);
%   solution to (c)
  for i=1:101
      Omega(i) = 0.6 + 1.1*(i-1)/100;
     [times,sols] = ode45(@(t,w) eom(t,w,Omega(i),omega,beta),tspan,initial_w);
     rval(i) = sqrt(sols(end,1)^2+sols(end,2)^2);
  end
  figure
  plot(Omega,rval,'LineWidth',2,'Color','r')
  ylim([0,1]);
  set(gca,'FontSize',14);
  xlabel('\Omega');
  ylabel('dist from quadrupole axis (m)');
  title('Solution to (c)');
end
function dwdt = eom(t,w,Omega,omega,beta)
  position = w(1:3); velocity = w(4:6);
  accel = Omega^2*(1+beta*cos(omega*t))*[position(1);position(2);0];
  dwdt = [velocity;accel];
end
```