



School of Engineering  
Brown University

## EN40: Dynamics and Vibrations

### Homework 3: Kinematics and Dynamics of Particles Due Friday Feb 15, 2019

Please submit your solutions to the MATLAB coding problems 4, 5, 6 by uploading a single file, with a .m extension, to Canvas.

**1. Simple Newton's law problem.** The European standard EN12195-1:2010 regulates [safe securing of loads for transportation](#). For a cargo secured by a single top-over lashing (see the figure) they give the following formula for the largest mass that can be transported without slipping on the truck-bed

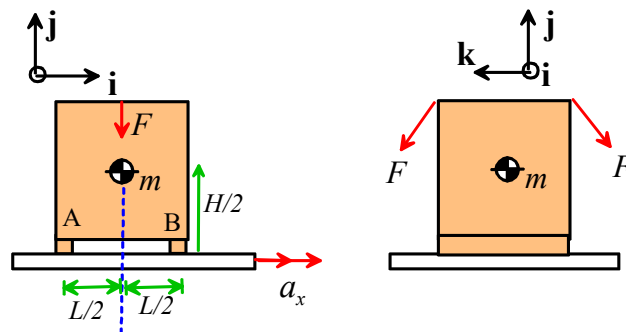
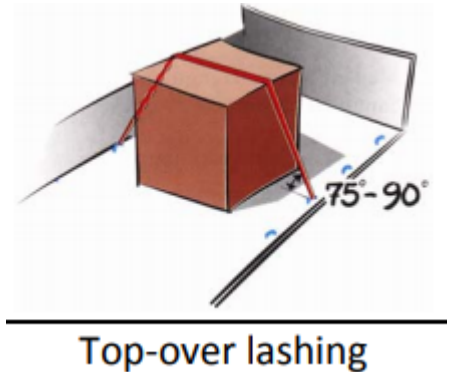
$$m = \frac{2\mu F \sin \alpha}{a_x - \mu g}$$

where  $\mu$  is the coefficient of friction between the load and truck-bed;  $a_x$  is the vehicle's acceleration (the formula assumes  $a_x > \mu g$ );  $F$  is the tension force in the lashing,  $\alpha$  is the angle

between the cable and the truck-bed (indicated as 75-90 degrees in the sketch), and  $g$  is the gravitational acceleration. The standard also states that for a load with COM at its center, the maximum mass that can be transported without tipping can be calculated from the formula

$$m = \frac{2F \sin \alpha}{a_x(H/L) - g}$$

The goal of this problem is to derive these two formulas.



1.1 Draw a (2D) free body diagram showing the forces acting on the mass. Assume no slip occurs at the contact between the crate and truckbed.

1.2 Write down  $\mathbf{F} = m\mathbf{a}$  and  $\sum \mathbf{M}_{COM} = \mathbf{0}$  for the mass

1.3 Hence, find formulas for the reaction forces acting at A and B (you will only be able to calculate the sum of the horizontal forces acting at A and B – see the ‘people mover’ example from class)

1.4 Use your solution to 1.3 to derive the two formulas in the European standard.

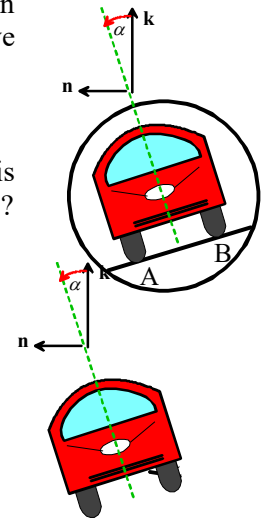
**2. Newtons laws in Normal-Tangential Coordinates.** Suppose that the Virgin Hyperloop vehicle (see HW2) travels at constant speed 1080 km/hr around a curve with constant radius  $R$ .

2.1 If the magnitude of the vehicle's acceleration cannot exceed  $0.2g$  ( $g=9.81 \text{ m/s}^2$  is gravitational acceleration) what is the smallest allowable value for the curve radius  $R$ ?

2.2 Curved track will be banked at an angle  $\alpha$  to minimize the lateral forces acting on the vehicle (and passengers inside the vehicle). On a copy of the figure to the right, draw a free body diagram showing the forces acting on the vehicle (you can neglect air resistance).

2.3 Assume that the vehicle travels at 1080 km/hr around a curve with the minimum allowable radius calculated in 2.1. Find formulas for the reaction forces acting on the vehicle normal and tangential to the banked track, in terms of the mass  $m$  of the vehicle and the bank angle  $\alpha$ . (You only need to find the sum of the forces acting on the two wheels; there is no need to calculate the forces at A and B separately)

2.4 Find the bank angle that will ensure that the force acting tangential to the track is zero.



**3. In this problem we will analyze the motion of the first stage of the Falcon-9 launch vehicle**

To keep the calculation simple, we will assume that

- The mass of the rocket decreases with time as propellant is ejected, and can be calculated from the formula  $m = M - \mu t$  where  $M$  and  $\mu$  are constants.
- Assume a vertical trajectory and include gravity
- Neglect air drag



Assume that the distance traveled by the vehicle  $x(t)$  satisfies the differential equation

$$(M - \mu t) \frac{d^2 x}{dt^2} = T_0 - (M - \mu t)g$$

where  $T_0$  is the rocket thrust (we will assume its constant for this problem) and  $t$  is time. (this looks like  $\mathbf{F}=\mathbf{ma}$ , but it is derived by considering the rate of change of momentum of the rocket and propellant – applying Newton's law to an object that does not have constant mass is a bit tricky).

3.1 Write a MATLAB 'Live Script' to solve the differential equation for  $x$ , (using the 'dsolve' function)

with initial conditions  $x = 0$   $\frac{dx}{dt} = 0$  at time  $t=0$ . Please upload your script to CANVAS.

3.2 Hence (using your 'Live Script') plot the speed as a function of time. Use the following values for parameters:

- Total initial mass  $M=575000\text{kg}$
- Propellant mass flow rate  $\mu = 2500\text{kg} / \text{s}$
- Thrust  $T_0 = 7.6 \times 10^6 \text{ N}$
- Time interval  $0 < t < 162 \text{ sec.}$

4. We can set up a more sophisticated model of the Falcon-9 by using the MATLAB ode solver to integrate the equations of motion. For the complete calculation, assume that

- The mass of the rocket can be calculated from the formula

$$m = \begin{cases} 550000 - 2500t & t < 165\text{s} \\ 137500 - 263(t - 165) & 165 < t < 584\text{s} \\ 27000 & t > 584\text{s} \end{cases}$$

The mass is in kg, and the discontinuities in the formula account for the two stages of the rocket (and the burnout at 584s)

- The thrust varies with time as

$$T_0 = \begin{cases} 7.6 \times 10^6 & t < 165\text{s} \\ 1.6 \times 10^6 & 165 < t < 584\text{s} \\ 0 & t > 584\text{s} \end{cases}$$

- A drag force

$$F_D = 50\rho v^2$$

(in Newtons) acts on the rocket, where  $v$  is the rocket's speed in m/s, and  $\rho$  is the air density, which varies with altitude as  $\rho = 1.22 \exp(-x / 8.5 \times 10^3) \text{ kg m}^{-3}$ , where  $x$  is in  $m$ .

- Assume as an approximation that the trajectory is vertical, and neglect variations in gravitational force.

4.1 Show that the equation of motion

$$(M - \mu t) \frac{d^2 x}{dt^2} = T_0 - (M - \mu t)g - 50\rho v^2$$

can be written in the form required by the MATLAB ode solver as follows

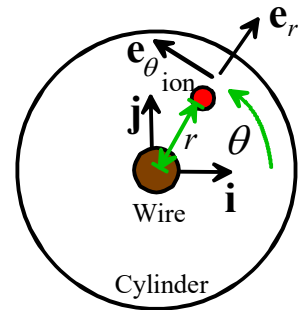
$$\frac{d}{dt} \begin{bmatrix} x \\ v \end{bmatrix} = \begin{bmatrix} v \\ (T_0 - 50\rho v^2) / m - g \end{bmatrix}$$

4.2 Write a MATLAB script that will calculate  $x$  and  $v$  for a time interval  $0 < t < 700\text{s}$ . Plot the variation of speed  $v$  with time, and compare the prediction with the experimental data (use the data file from HW2, and plot the prediction and measurement on the same graph)

5. A ‘[Dynamic Kingdon trap](#)’ is a device for storing charged particles (see e.g. this [patent](#)). It consists of a wire that runs down the axis of a cylinder. A time varying electric field is applied to the wire, which subjects the particle to a radial force

$$\mathbf{F} = q(V_0 + \Delta V \cos \Omega t) \frac{1}{2\pi\epsilon_0 r} \mathbf{e}_r$$

Here,  $V_0, \Delta V$  are a static and fluctuating voltage applied between the wire and cylinder;  $\Omega$  is the frequency of the fluctuating voltage,  $q$  is the charge of the trapped ion, and  $\epsilon_0$  is the permittivity of free space.



The goal of this problem is to analyze the motion of an ion in the trap. We will assume that the particle moves in the  $r, \theta$  plane, and so must solve for  $[r, \theta]$  as a function of time

5.1 Write down Newton’s laws for the particle (use polar coordinates) and show that the  $\mathbf{e}_r, \mathbf{e}_\theta$  components of the equation can be re-arranged into the form required by matlab by re-writing them as

$$\frac{d}{dt} \begin{bmatrix} r \\ \theta \\ v \\ \omega \end{bmatrix} = \begin{bmatrix} v \\ \omega \\ r\omega^2 + (\lambda + \mu \cos \Omega t) / r \\ -2v\omega / r \end{bmatrix}$$

Find formulas for  $\lambda, \mu$  in terms of  $V_0, \Delta V, \epsilon_0, q, m$ .

5.2 Write a MATLAB code that will calculate  $[r, \theta, v, \omega]$ . For the parameter values listed below, plot the path of the particle, for a time period  $0 < t < 50$  time units. You can use the following plot function to do the polar plot for you

```
polarplot(theta, r, 'Displayname', '(Problem a)', 'Color', [1, 0, 0]);
```

where theta is a vector of values of  $\theta$  and r is a vector of values of  $r$  at successive time intervals (these can be extracted from the matrix that stores values of  $[r, \theta, v, \omega]$  returned by ode45).

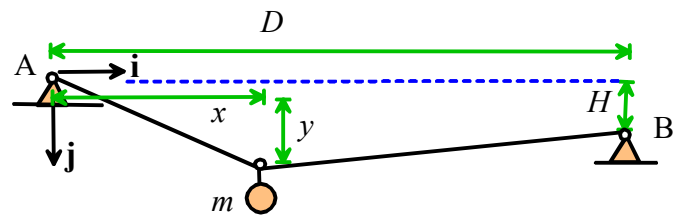
(a)  $\lambda = -1, \mu = 0, \Omega = 2$  (this is a static trap – the electric field does not vary with time), with initial conditions  $r = 1, \theta = 0, v = 0, \omega = 1$ . For this case you will need to specify the time intervals you would like in the solution, otherwise ode45 will take huge time steps and the trajectory will look funny. You can do this, eg, with

```
[time, sols] = ode45(@(t,w) trap(t,w,lambda,mu,Omega), [0:0.1:50], initial_w);
```

Here, the 0:0.1:50 tells MATLAB you would like solution values every 0.1 time intervals.

(b)  $\lambda = -1, \mu = 0, \Omega = 2$  with initial conditions  $r = 0.8, \theta = 0, v = 0, \omega = 0.4$ . (use ‘hold on’ to put this on the same plot as (a)). The static trap is stable – a particle remains trapped, as long as nothing slows it down.

- (c)  $\lambda = -1, \mu = 8, \Omega = 2$  (this is a dynamic trap – the electric field varies with time) with initial conditions  $r = 2.5, \theta = 0, v = 0, \omega = 0.25$ . Plot this case on the same graph as (a) and (b). Particles with the right radius and velocity are trapped.
- (d)  $\lambda = -1, \mu = 8, \Omega = 2$  with initial conditions  $r = 1.5, \theta = 0, v = 0, \omega = 0.25$ . Run this case for 100 time units, and plot it on a new figure. A dynamic trap will expel particles that are too fast or slow, and are too close or too far from the wire



6. The figure shows an idealization of a person with mass  $m$  on a zip-line. The person starts at end  $A$  of the line and travels to end  $B$ , a horizontal distance  $D$  and vertical distance  $H$  below  $A$ . The cable is idealized as a spring with stiffness  $k$  and un-stretched length  $L_0$ , with  $L_0 > \sqrt{D^2 + H^2}$ . The tension in the cable is related to its stretched length by  $T = k(L - L_0)$ .

6.1 Show that the length of the cable is  $L = \sqrt{x^2 + y^2} + \sqrt{(D-x)^2 + (y-H)^2}$  (this is just geometry)

6.2 Draw a free body diagram showing the forces acting on the mass (include the cable tension, which is the same on both sides of the mass). Assume that an air drag force with magnitude  $F_D = \frac{1}{2} \rho A C_D V^2$  and direction opposite to the velocity vector acts on the rider. Here,  $\rho$  is the air density,  $A$  is the frontal area of the rider;  $C_D$  is the drag coefficient and  $V = \sqrt{v_x^2 + v_y^2}$  is the rider's speed.

6.3 Use Newton's law to derive equations of motion for the mass, and show that they can be re-arranged in the following form (suitable for the MATLAB ODE solver)

$$\frac{d}{dt} \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ (T/m)(D-x)/d_2 - (T/m)x/d_1 - cVv_x \\ g - (T/m)(y-H)/d_2 - (T/m)y/d_1 - cVv_y \end{bmatrix}$$

$$d_1 = \sqrt{x^2 + y^2}$$

$$d_2 = \sqrt{(D-x)^2 + (y-H)^2}$$

$$c = \frac{1}{2m} \rho A C_D$$

Use ode45 to solve the equation. Use the following values for parameters

- Rider mass  $m$  80kg
- Dimensions  $D=400\text{m}$ ,  $H=30\text{m}$
- Drag parameters:  $A=1\text{m}^2$ ,  $\rho = 1.22 \text{ kg m}^{-3}$   $C_D = 0.2$
- Line stiffness  $k=10^6\text{Nm}^{-1}$  and unstretched length  $L_0 = \sqrt{D^2 + H^2} + 2 \text{ m}$
- Initial position  $x = 0, y = (L_0^2 - D^2 - H^2) / 2(L_0 - H), v_x = v_y = 0$

Add an 'event' function to your script that will stop the calculation when the rider reaches  $x=D$ .

Hence, plot (i) The trajectory of the rider; and (ii) a graph of the riders speed as a function of time. You can compare your predictions to data for a number of ziplines [here](#)

6.4 The rider arrives at  $x=D$  at over 10 m/s. How would you change the design to reduce the speed to below 7 m/s (but without changing  $H$  or  $D$ )?