Final Project Write-up

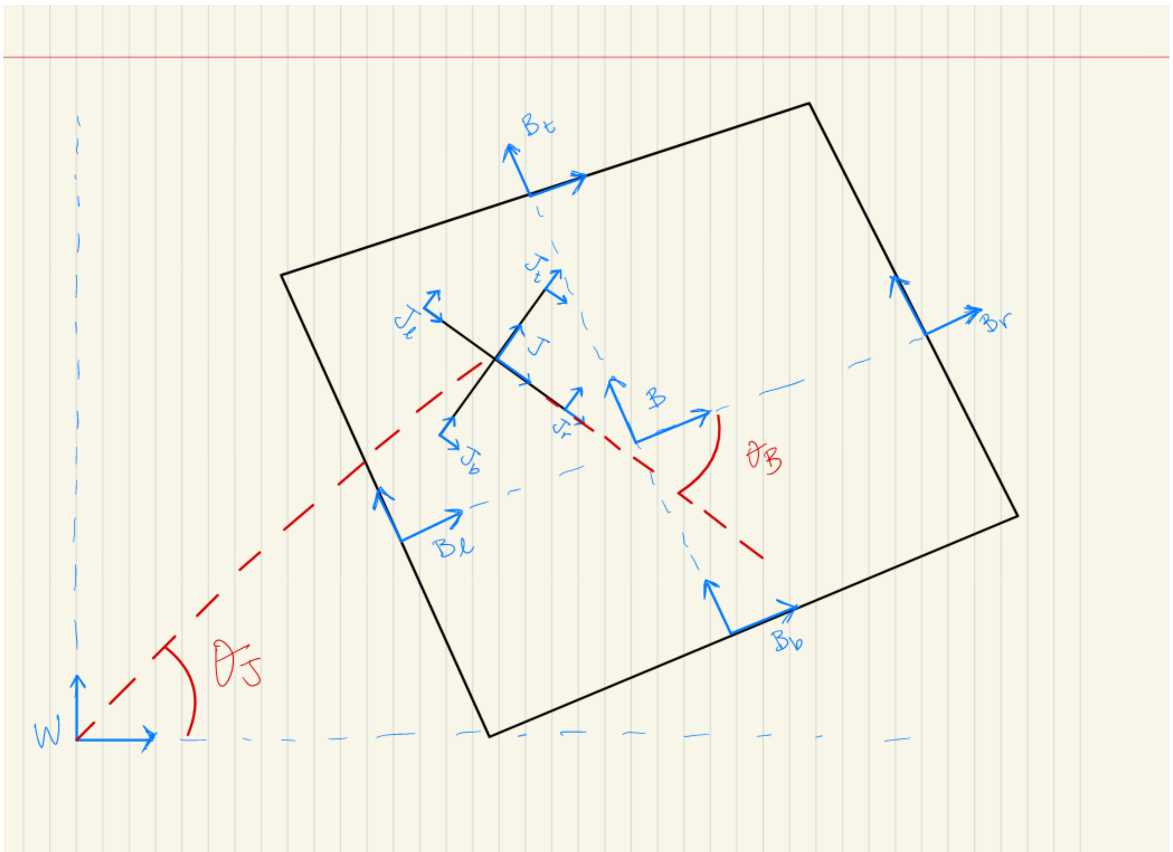
ME314 – Machine Dynamics – S2021

Numair Ahmed

Project Description

Jack in a box simulation and animation project done in Jupyter Notebooks

Diagram of jack and box system



Frame transformations

On paper	In code
World frame to jack center	gwj
World frame to Jack center to jack top	gwt
World frame to Jack center to jack bottom	gwb
World frame to Jack center to jack left	gwl
World frame to Jacket center to jack right	gwr

On paper	In code
World frame to jack center to box center	gwj_b
World frame to Jack center to box center to top box wall	gwb_t
World frame to Jack center to box center to bottom box wall	gwb_b
World frame to Jack center to box center to left box wall	gwb_l
World frame to Jacket center to box center to right box wall	gwb_r

Transformation for 16 impact cases

On paper	In code
world to jack bottom pt, to box top wall	Gbb_t
world to jack bottom pt, to box bottom wall	Gbb_b
world to jack bottom pt, to box left wall	Gbb_l
world to jack bottom pt, to box right wall	Gbb_r
world to jack top pt, to box top wall	Gtb_t
world to jack top pt, to box bottom wall	Gtb_b
world to jack top pt, to box left wall	Gtb_l
world to jack top pt, to box right wall	Gtb_r
world to jack left pt, to box top wall	Glb_t
world to jack left pt, to box bottom wall	Glb_b
world to jack left pt, to box left wall	Glb_l
world to jack left pt, to box right wall	Glb_r
world to jack right pt, to box top wall	Grb_t
world to jack right pt, to box bottom wall	Grb_b
world to jack right pt, to box left wall	Grb_l
world to jack right pt, to box right wall	Grb_r

After meticulously ensuring the transformations were happening as I had intended (this required a lot of printing in between each transformation to make sure the code was correct), I was able to compute the energy and Euler-Lagrange equations of the system.

In order to derive the E-L equations, we need the kinetic and potential energies. After some discussion with TAs and MSR cohort mates, it made sense to model the jack/box system to be rolling around on a surface so as to eliminate the effects of gravity in the Z direction (in & out of the screen); potential energy would be 0. KE of the jack and box have to be computed separately, given their separate body velocity and mass-inertia.

KEj = ((Vj.T * Ij * Vj) / 2)

KEb = ((Vb.T * Ib * Vb) / 2)

KE = KEj + KEb

PE = 0

Once system energy is computed, Lagrangrian = KE – PE can be computed, followed by Euler-Lagrangian (EL).

Where EL equation is derived using the relation: $\frac{d}{dt}(\frac{\delta L}{\delta(\frac{dq}{dt})}) - \frac{\delta L}{\delta q} = F$ (taken from wikipedia). Since gravity is not acting on the system, I chose to rotate the box via a sinusoidal input applied in the XY direction with an arbitrary amplitude of 500. If I had more time with the project, I would do further testing with the input force to figure out why the box is disappearing into the right side of the animation window.

From the code, the EL equations come out to be:

```
=================================================

Euler Lagrange equations for box and jack with input force on box
```

$$\begin{bmatrix} 4.0\frac{d^2}{dt^2}x_j(t) \\ 4.0\frac{d^2}{dt^2}y_j(t) \\ 25.0\frac{d^2}{dt^2}\theta_j(t) \\ 100.0\frac{d^2}{dt^2}x_b(t) \\ 100.0\frac{d^2}{dt^2}y_b(t) \\ 625.0\frac{d^2}{dt^2}\theta_b(t) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 500\sin(t) \\ 0 \\ 0 \end{bmatrix}$$

Since the jack is contained inside the box and the external force acts upon the box, we have to model an impact of each jack end against each of the four box walls; therefore, we have 16 constraint equations. The 16 constraint conditions are then aggregated into a list of phi's to then be processed during the impact update computations.

To compute the impact update for each of the 16 impacts, we need to solve for the following equations:

- $\frac{\partial L}{\partial \dot{q}}\Big|_{\tau^-}^{\tau^+} = \lambda \frac{\partial \phi}{\partial q}$ which tells us to equate the difference between the Lagrangian derivative w.r.t. q_dot to the derivative of the constraint equation phi scaled by lambda. (equation taken from: homework 5 assignment)
- $\left[\frac{\partial L}{\partial \dot{q}} \cdot \dot{q} - L(q, \dot{q})\right]\Big|_{\tau^-}^{\tau^+} = 0$ and to solve the Hamiltonian for just before and after the impact. (equation taken from: homework 5 assignment)

This has to be done for each constraint which requires its own impact update equation. This is all done in a conditional for loop in the code. The simulation runs through t = [0,10] seconds and checks for an impact. If an impact is detected, it'll solve the impact update equation for that particular impact condition (tracked throughout the simulation loop) using the configuration variables and time-derivative configuration variables at that condition.

The result of the animation makes intuitive sense upon seeing the external box begin to rotate and steadily move in the x direction, since the external force is applied to the x component of the box's trajectory. As it moves far enough to collide with the jack inside, the jack bounces off the wall (elastic impact) as its 1kg mass is much smaller than the wall's 20kg. The behavior seems reasonable, although, given additional time, I would like to experiment with different input forces (ie. linear forces applied at additional points either on jack or box). Overall a very tough and involved simulation project but achievable after tons of help!

Thank you to Dr. Murphey, Thomas, and Willa.