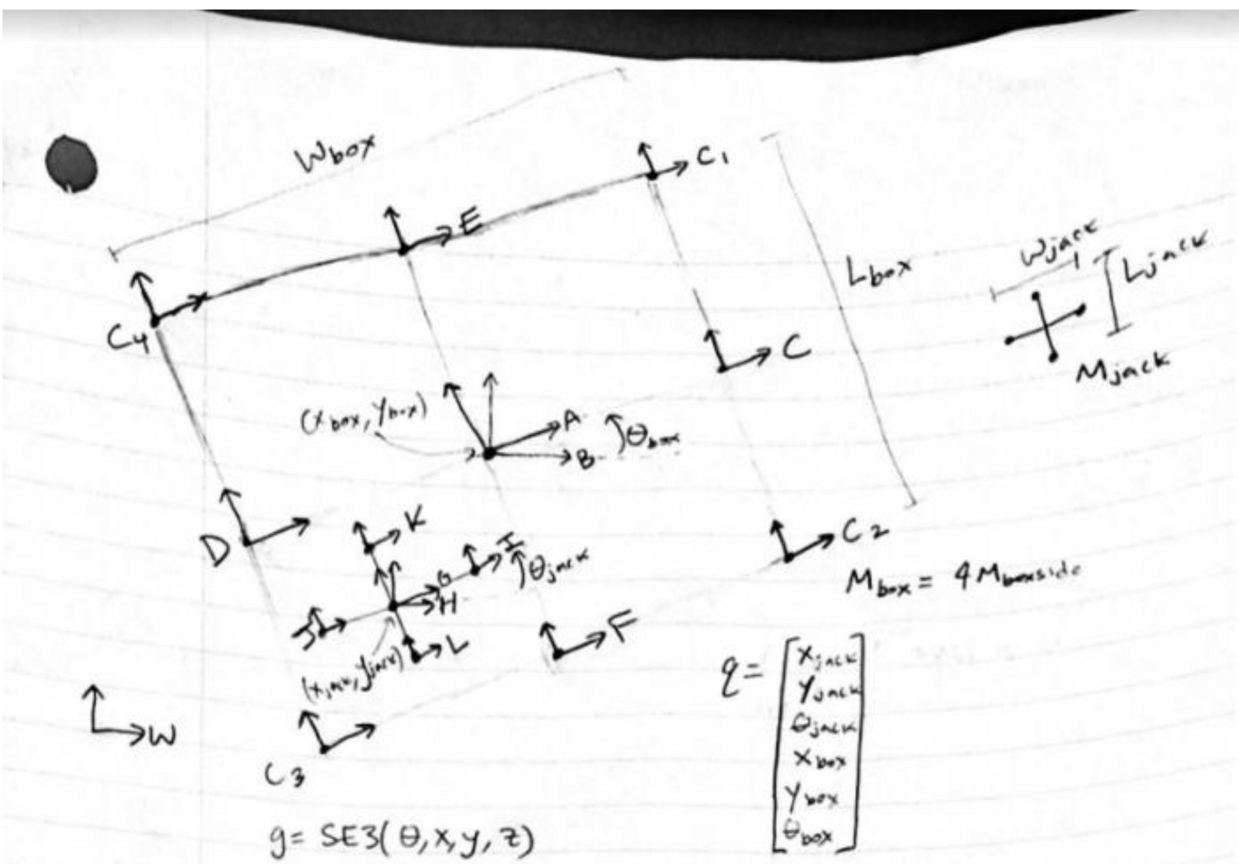
ME 314 Final Project Write-Up – Gilberto Guadiana

The project guidelines are below.

1. You should include a brief description of what you originally proposed and what changes you needed to make to your original proposal (and why you made them). If you chose the 'default project', you can simply state that you did so.

I chose the default project of the jack in a box. In my version of the project, we are using a bird's eye view to view the jack and box from above. Gravity is modeled as being into the screen.

2. You should include a drawing of the system you are modeling that includes all the frames you are using, with frame labels. In addition to the drawing, you should include all of the rigid body transformations you are using between the frames. These frames and their labels should be clearly identifiable in your code.



Box

$g_{WA} = SE3(0, x_{box}, y_{box}, 0)$ $\qquad g_{WB} = g_{WA} SE3(\theta_{box}, 0, 0, 0)$

$g_{WC} = g_{WB} SE3(0, \frac{W_{box}}{2}, 0, 0)$ $\qquad g_{WD} = g_{WB} SE3(0, -\frac{W_{box}}{2}, 0, 0)$

$g_{WE} = g_{WB} SE3(0, 0, \frac{L_{box}}{2}, 0)$ $\qquad g_{WF} = g_{WB} SE3(0, 0, -\frac{L_{box}}{2}, 0)$

$g_{WC1} = g_{WB} SE3(0, \frac{W_{box}}{2}, \frac{L_{box}}{2}, 0)$ $\qquad g_{WC2} = g_{WB} SE3(0, \frac{W_{box}}{2}, -\frac{L_{box}}{2}, 0)$

$g_{WC3} = g_{WB} SE3(0, -\frac{W_{box}}{2}, -\frac{L_{box}}{2}, 0)$ $\qquad g_{WC4} = g_{WB} SE3(0, -\frac{W_{box}}{2}, \frac{L_{box}}{2}, 0)$

Jack

$g_{WG} = SE3(0, x_{jack}, y_{jack}, 0)$ $\qquad g_{WH} = g_{WG} SE3(\theta_{jack}, 0, 0, 0)$

$g_{WI} = g_{WH} SE3(0, \frac{W_{jack}}{2}, 0, 0)$ $\qquad g_{WJ} = g_{WH} SE3(0, -\frac{W_{jack}}{2}, 0, 0)$

$g_{WK} = g_{WH} SE3(0, 0, \frac{L_{jack}}{2}, 0)$ $\qquad g_{WL} = g_{WH} SE3(0, 0, -\frac{L_{jack}}{2}, 0)$

3. Using the drawing and the rigid body transformations, you should say in writing how you calculate the Euler-Lagrange equations, the constraints, the external forces and impact update laws.

To find the Lagrangian of the system, I considered the jack and each of the four box sides. I represented the jack with four point masses attached with weightless rods. I calculated the rotational inertia of the jack and set up its inertial tensor. I did the same for each of the box sides. I let each of the box sides have the same amount of mass and have the center of mass for each box side be in the middle of the side.

I defined the frames shown in the picture above, and I then defined the body velocities. For the jack, the body velocity is in frame H. For the box sides, the body velocities are in frames C, D, E and F. I then used the body velocities and the inertial tensors to compute the kinetic energies of each component and added those components together to get the total kinetic energy of the system. Because gravity is into the screen in my interpretation of the project, there are no potential energy terms.

I decided to model the shaking of the box through a constant external force acting on the orientation of the box (theta box). After having the Lagrangian and external forces defined, I defined the Euler-Lagrange equations. With these equations, I solved them for the equations of motion of my coordinate variables using the Python Sympy solver. I used the Python Sympy Lambdify function to declare numerical functions for the equations of motion.

I then defined my impacts. I considered my expressions for impact in the box frame (frame B) and checked whether the jack was inside or outside of the box. I have 16 different impact expressions that come from the four corners of the jack being able to collide with any of the four box sides. To find the expressions, I converted the coordinates of each of the jack's corners to the box frame and considered the x and y positions of the box sides in the box frame. I defined my impact conditions based on these expressions. I use these impact conditions to check numerically whether the value of the expression is changing sign. If it is changing sign, then there is a collision occurring. I then setup the impact equations and solved for them numerically as the system progresses over time. My impact update functions take in the values of the state and use them to simplify the solving process. It then returns the updated, post-impact velocities of the coordinate variables.

My simulation function checks for a collision by calling the 16 impact condition functions. If there is a condition, the system update is handled by calling the appropriate impact update function. If there is no impact, the system updates by using the solutions to the forced Euler-Lagrange equations.

4. If your code works, you should describe in words what happens in the simulation and why you think it is correct (e.g., at a high level, describe why you think the behavior is reasonable or not). If your code works, this can be the end of your write-up. If your code works, your entire write-up will likely just be a few paragraphs.

My simulation has initial conditions where the jack is moving downward with no rotation and the box is rotating. At the start of the simulation, the jack hits the box's bottom side and continues to hit the different sides of the box multiple times. Through the collisions, the jack begins to rotate and the position of the box shifts downward. The simulation seems to show reasonable behavior. The jack never exited the box and whenever the jack hits the box, the box is visibly pushed in response to the jack. We note that although gravity is into the screen, the box moves down the screen over time. This behavior is consistent with the initial condition of the jack moving downward.