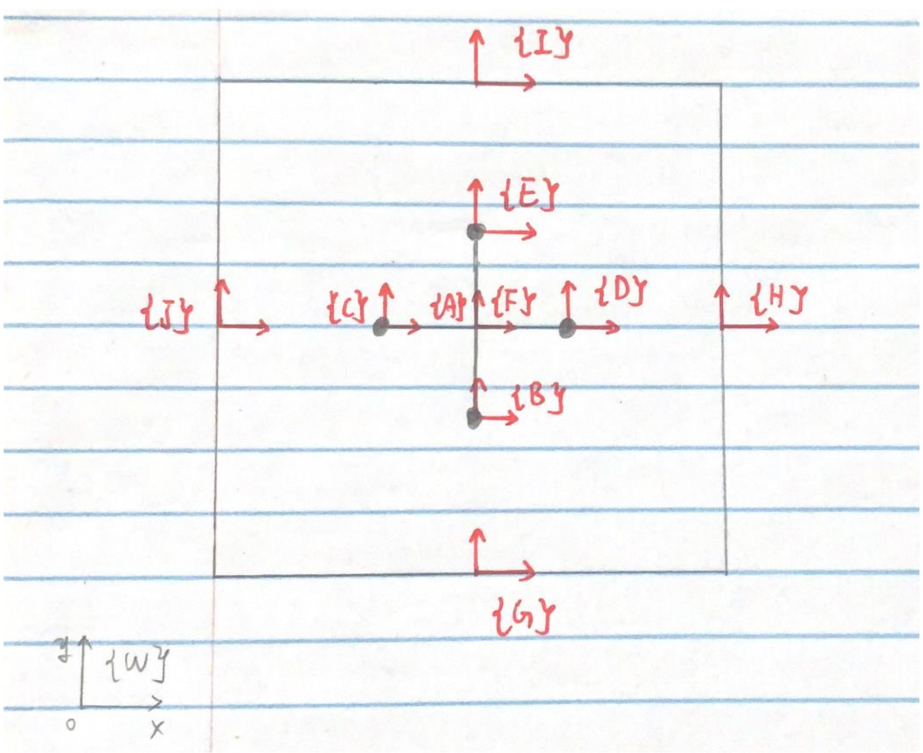
Project guidelines

1, Brief description of what I did in the “default project”

In this final project, I chose the “default project”. In this project, I will animate a jack in a box. The box will move to the upper right. This jack will go into free fall and then impact with the bottom of the box. After that, the jack will bounce up and rotate. And then it will impact with the box again. Due to the impact of the jack, the box will also rotate and go down a little bit. But it will go up again with the external force. Then these two bodies will impact with each other several times.

I assume the mass of each point of the jack and the mass of each edge of the box are all equal to 1kg. I apply a 0.5*m*g force on the x direction of the Box to move the Box. I assume that the jack and box both have gravity. So, I apply an 8*m*g force on the y direction of the Box so that they will not go into free fall. Besides, I assume the length of the edge of the box is 4m. And the distance between two points in the Jack is 1m.

2, The draw of my system



In this system:

Frame {w} is the world frame.

Frame {A} is at the center of the Jack.

Frame {B} is at the bottom point of the Jack.

Frame {C} is at the left point of the Jack.

Frame {D} is at the right point of the Jack.

Frame {E} is at the top point of the Jack.

Frame {F} is at the center of the box.

Frame {G} is at the center of the bottom edge of the box.

Frame {H} is at the center of the right edge of the box.

Frame {I} is at the center of the top edge of the box.

Frame {J} is at the center of the left edge of the box.

I assume in the initial condition the Jack is at the center of the box. So, frame{A} and frame{F} are coincident temporarily.

In the python, I made a function called "SE3" to get the rigid body transformations between the frames. The rigid body transformations between frame{A} and {w} is gwa = SE3(x2, y2, theta2), where x2, y2, and theta2 are actuallyx_{jack}, y_{jack} and θ_{jack}. So, similarly,

1. The rigid body transformation between frame{A} and {B} is gab = SE3(0, -0.5, 0)
2. The rigid body transformation between frame{A} and {C} is gac = SE3(-0.5, 0, 0)
3. The rigid body transformation between frame{A} and {D} is gad = SE3(0.5, 0, 0)
4. The rigid body transformations between frame{A} and {E} is gae = SE3(0, 0.5, 0)

5. The rigid body transformations between frame{B} and {w} is gwb = gwa*gab
6. The rigid body transformations between frame{C} and {w} is gwc = gwa*gac
7. The rigid body transformations between frame{D} and {w} is gwd = gwa*gad
8. The rigid body transformations between frame{E} and {w} is gwe = gwa*gae

9. The rigid body transformation between frame{F} and {w} is gwf = SE3(x1, y1, theta1), where x1, y1, and theta1 are x_{Box}, y_{Box} and θ_{Box}

10. The rigid body transformation between frame{F} and {G} is gfg = SE3(0, -2, 0)
11. The rigid body transformation between frame{F} and {H} is gfh = SE3(2, 0, 0)
12. The rigid body transformation between frame{F} and {I} is gfi = SE3(0, 2, 0)
13. The rigid body transformations between frame{F} and {J} is gfj = SE3(-2, 0, 0)

14. The rigid body transformations between frame{G} and {w} is gwg = gwf*gfg
15. The rigid body transformations between frame{H} and {w} is gwh = gwf*gfh
16. The rigid body transformations between frame{I} and {w} is gwi = gwf*gfi
17. The rigid body transformations between frame{J} and {w} is gwj = gwf*gfj

18. The rigid body transformations between frame{B} and {F} is gfb= gfw*gwb
19. The rigid body transformations between frame{C} and {F} is gfc= gfw*gwc
20. The rigid body transformations between frame{D} and {F} is gfd= gfw*gwd
21. The rigid body transformations between frame{E} and {F} is gfe= gfw*gwe

3, How I calculate the EL equation, impact update laws and external forces.

In this project, there is no constraint in this system. So, I will write how to calculate the Euler-Lagrange equation, the external forces and impact update laws.

Calculation of the external forces

I assume there is an 8*m*g external force applied on the y direction of the box. Because I assume these two bodies both have gravity. The mass of the jack is 4*m*g. And the mass of the box is also 4*m*g. After applying this external force, these two bodies will not go into free fall. Besides, I assume a 0.5*m*g force will be applied on the x direction of the box so that the box will move right and then get impact with the jack. Finally, I made a 1*6 matrix about force and add it into the right side of the Euler-lagrangian equation.

Calculation of the Euler-lagrangian equation

Similar to the homework 7, firstly I get the eight rigid body transformations between eight frames and the world frame. After that, I assume the rotational inertia J=1 because each frame is in the center of the geometry. Then according to this below function in the lecture note

$$KE = \frac{1}{2}(V^b)^T \begin{bmatrix} mI_{n\times n} & 0 \\ 0 & \mathrm{I} \end{bmatrix} V^b$$

I can calculate the KE. PE can directly be obtained from the rigid body transformations. Then I can get the Lagrangian equation. After using the equation “dLdqdot.diff(t) – dLdq”, I can get the left side of the Euler-lagrangian equation. Combined with the right side of the Euler-lagrangian equation, I can solve this equation.

Calculation of the impact update laws

In this project, there are 16 phi and 16 impact update functions. Firstly, I inversed gwf to get the gfw. Then I can get the positions of four points of the jack in the frame{F}. Then I assume the x component and y component of these four rigid body transformations are less than 3 and larger than 3. These are phi functions. When I get the phi function, I can get the impact update function based on these two functions

$$\left.\frac{\partial L}{\partial \dot{q}}\right|_{\tau^-}^{\tau^+} = \lambda \frac{\partial \phi}{\partial q},$$

$$\left[\frac{\partial L}{\partial \dot{q}} \cdot \dot{q} - L(q,\dot{q})\right]_{\tau^-}^{\tau^+} = 0.$$

After getting the impact update functions, I define 16 impact condition functions. When the function shows “True”, that means there is an impact. The impact update functions will change the original function. Then I added these condition functions into the simulate function. Finally, I complete the impact update laws.

4, The description of my simulation.

My code works successfully. I set the q as [4, 4, 0.2, 4, 4, -0.2] so that two bodies are both slanted. In this simulation, due to the external forces, the box moved up and right at the beginning time. The jack was falling down and impacted on the bottom edge of the box. After this impact, the jack bounced up to the left side of the box and then bounced again. The box went down a little bit and then went up again with the y direction external force. Then with these two external forces and some impacts, these two bodies started to rotate and move right.

I think it is correct. Because each time when the impact happened, the jack will bounce up. And the jack did not “escape” from the box in the whole process. This system has six degrees of freedom. And there are rotational inertias in both two bodies. The animation shows the effect of external force and impacts.