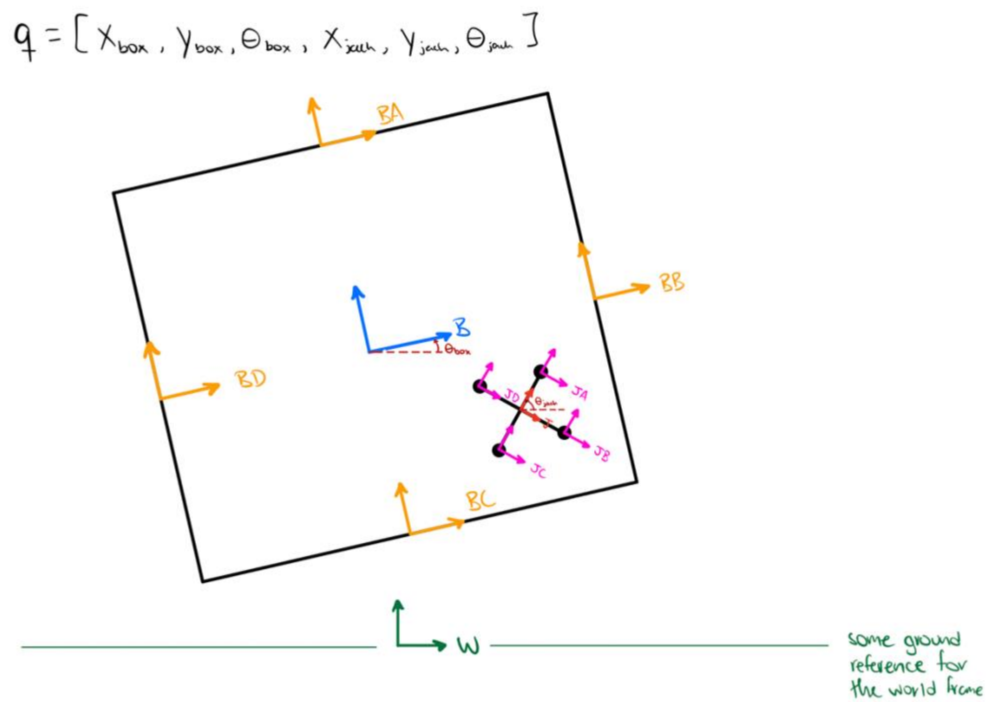
Ilan Mayer
ME314
Professor Todd Murphey
Due 06/08/2023

ME314 Final Project

Description

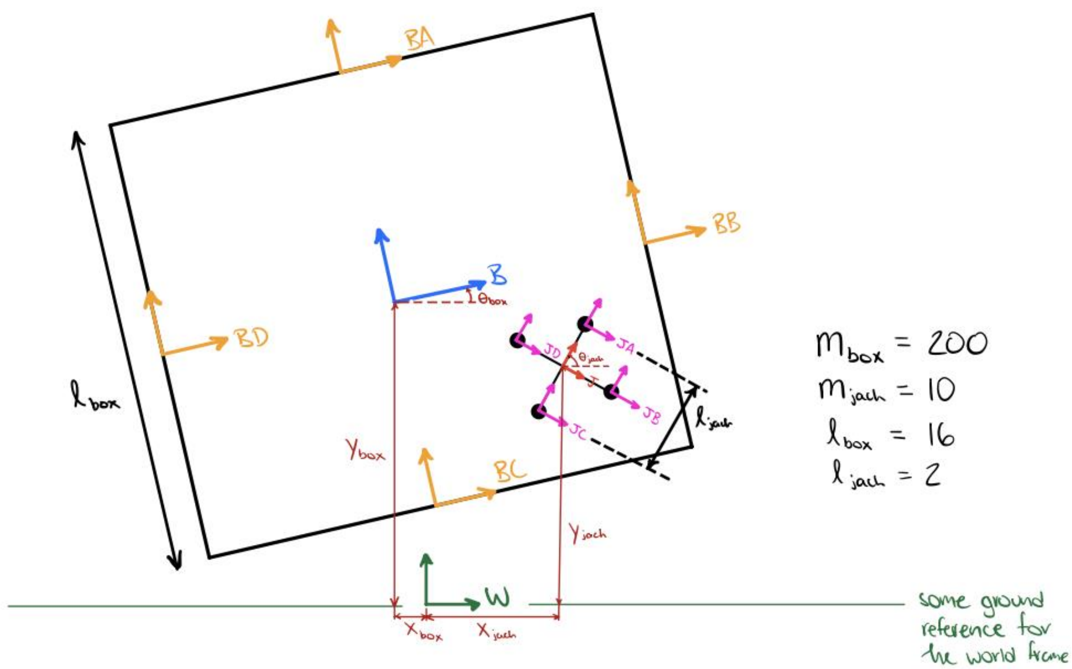
For my final project in Mechanical Engineering 314, I decided to choose the default project. This consists of a planar representation of the behavior of a jack or dice in a box. The jack has 4 corners, which can come into contact with any of the 4 walls of the box (impact). The idea behind this project is to simulate and animate the trajectory of the jack as it moves around in the box and comes in contact with the walls. It is important to mention that the box also rotates due to external forces acting on it. The project requires a complete incorporation of several things we have studied throughout the year; configuration variables, Euler Lagrange equations, impact update laws, constraints, homogenous rigid body transformations; rotational inertia, amongst several others. The image below is a drawing that models the several frames used to describe the translations and rotations of both the jack and the box. These frames allow us to map coordinate points from anybody frame to the world frame. More specifically in the jack in the box, we can determine rotational and translational kinetic energy for any body (4 walls of box and 4 edges of jack) in terms of the world frame, allowing us to determine kinetic energy, potential energy, and eventually the Euler Lagrange equation.

System Drawing



The frames used are B, BA, BB, BC, BD, J, JA, JB, JC, JD. The B frame is located at the center of mass of the box, which happens to be its central position. The B-frame is rotated by an amount theta_box (which is an angle between the x-axis of the B frame and the x-axis of the world frame), and translated by an amount of x_box, y_box. Therefore, B represents the location (x,y) and rotation of the box. BA, BB, BC, and BD are located at the center of mass of each wall of the box (which happens to be the midpoint for each wall). The homogenous rigid body transformation from the B frame to any frame BA, BB, BC, or BD is simply a translation by an amount of l_box/2 in either positive or negative x, positive or negative y, or both, depending on which BA, BB, BC, BD you are moving to. These body frames work to determine the KE and PE for each wall with respect to the world frame and allow us to calculate the lagrangian to see the acceleration of the configuration variables for the box.

The J frame is located at the center of mass of the jack, which happens to be its central position. The J frame is rotated by an angle of theta_jack (which is the angle between the x-axis of the J frame and the x-axis of the world frame), and translated by an amount of x_jack, y_jack. Therefore, J represents the location (x,y) and rotation of the jack. JA, JB, JC, JD are located at the edges of the jack. The homogenous rigid body transformation from the J frame to any frame JA, JB, JC, JD is simply a translation by an amount of l_jack/2 in either positive or negative x, positive or negative y, or both, depending on which JA, JB, JC, JD you are moving to. These body frames are used to find the kinetic energy and potential energy of the jack with respect to the world frame.



The configuration variables of the system can be illustrated above in the wine-red color. The configuration q is given by the following variables: the x-position of the center of mass of the box (the center) relative to the world frame, the y-position of the center of mass of the box (the center) relative to the world frame, the rotation of the box as an angle measured from the x-axis of the world frame to the x-axis of the B frame, the x-position of the center of mass of the jack (the center) relative to the world frame, the y-position of the center of mass of the jack (the center) relative to the world frame, and the rotation of the jack as an angle measured from the x-axis of the world frame to the x-axis of the J frame.

The dimensions of the system are also illustrated above. The length of the box (which is a square) is given by l_box and has a magnitude of 16 units. The length of the jack, from one edge to another, is given by l_jack and has a magnitude of 2 units. Furthermore, the mass of the box is given by m_box and has a magnitude of 200 units. The mass of the jack is given by m_jack and has a magnitude of 10 units.

Code Breakdown

The homogeneous rigid body transformations were defined using the figure shown above. First, the transformations from the world frame to B, BA, BB, BC, BD framed were computed. This was done by using the SE3 helper function I created in HW7. The transformations from the world to the frame at the COM of the jack (J frame) and all the frames at the edges of the jack were computed in the same manner. The purpose of these frames is so that we can find the KE and PE of the jack and box in the J and B frames and use the transformations to get KE and PE in terms of the world frame. This is what allows us to compute the lagrangian of the entire system. To get KE, the rotational inertia tensors for both the jack and box were done so just like HW7, knowing that there is only rotational inertia about the z-axis. Once the tensors were computed, the dynamic matrix for both jack and box were computed by making a 6x6 like in HW7. The place in where the rigid body transformations came into play was in computing the body velocities of the jack and box. The body velocity of the box was done by taking the inverse of the g_W_B matrix and multiplying it by its derivative. g_W_B is the transformation from the world frame to the box COM frame, which we had defined earlier. With the body velocity and inertia tensor, we are able to compute the kinetic energy of the box in the world frame. This shows how rigid body transformations make something like computing the KE of a moving object wrt the world frame much simpler. The exact same is done for the jack's KE. For the PE, it was simply obtaining the y-coordinate of the position of the origin of J and B frames wrt world frame. This gave us the height of J and B frames wrt world frame, which we could then plug into PE formula. With KE and PE, getting the lagrangian was simply finding the difference between them, and the left-hand side was simply found by using the regular EL formula. L.diff(qdot).diff(t) – L.diff(q).

The external forces were created in a way that there would be rotation for theta_box, but the box would stay in the same x, y coordinates. This was done by placing a force in the y-direction to counteract the force of gravity on the box, no force along x, and a force along theta that was used by plugging in a desired theta trajectory (inspired from HW7). This composed the rhs of the EL equations. A force vector F, was created, were the only non-zero components were the 2 and 3 elements (corresponding to y-box and theta-box configurations)… as the force only acts on those configurations. With the lhs and rhs of the EL, it could be then solved to find the acceleration of all configurations. Solutions were then lambdified.

The heart of the code lied within impacts, which used rigid body transformations to determine the impact constraints (phi), of which there were 16… for the 16 different possible impacts. I needed to find the transformation from each wall to each edge (16 different transformations). This is because I knew that when, for example, the y-coordinate of the transformation from the BA frame (wall A) to the JA frame (A edge of jack) was 0, there was an impact between wall A and edge A. Why so? Because this meant that the y-coordinate position of the origin of the JA frame wrt BA was 0. And since the wall A is at y=0 on the BA frame, this meant an impact was occurring. This was one of my impact constraints (where impact occurs). I multiplied the inverse of the rigid body transformation from the world to a wall frame by the rigid body transformation of the world frame to a jack edge frame in order to give me the homogeneous rigid body transformation from that wall frame to that edge frame. I repeated this process for each different combination of wall and edge (which represented each unique impact… 16 of them). Once I had all these transforms, I could begin writing my impact constraints. As I mentioned before, for the BA frame, impact occurred when the y-coordinate was 0, but for the BB frame impact was when the x-coordinate was 0. This is bc of how the wall aligns with the frame (aka wall B is at x=0 on BB frame). So I got either the x or y coordinate of each transformation (depending on which I needed) and set that equal to the constraint phi. I now had 16 of them.

To get the impact update laws, I computed the conserved quantity (P) and the Hamiltonian (H) of the system. P was computed by taking the derivative of the lagrange with respect to the previously defined derivative of configuration matrix q_dot. H was done so by multiplying the transpose of P and qdot and subtracting the lagrangian from that. I then needed to define a dummy dictionary for the configurations and configuration derivatives before impact (tau plus) and configurations and configuration derivatives after impact (tau minus). I substituted the matrix P for tp dummies and P for tm dummies, and did the same for the scalar H. I then subtracted P tau plus from P tau minus and the same for H. I created matrices for both to store the values. One of my impact update laws was simple, setting H_tau_plus minus H_tau_minus equal to zero. The challenge arose for the other 16 equations, which are defined by setting P_tau_plus minus P_tau_minus equal to the constraint wrt q (configuration) times a variable lambda. I set up a matrix for all phi I had previously determined, and then took the derivative of each one wrt to all 6 configurations. This gave me a 16x7 matrix. The first column corresponded to all derivatives taken wrt the first configuration variable q[0], the second column to all derivatives taken wrt to second configuration q[1], and so on and so forth. The Ptp – Ptm matrix worked in a way where the first row was the derivatives taken wrt to the first configuration variable q[0] the second for the second configuration, and so on. The first impact law was essentially the first element of the Ptp – Ptm matrix set equal to the element in the first row and first column of the 16x7 dphi_dq matrix. The second equation was the second element of P set equal to the element in the first row but second column of the 16x7 matrix. As you can see, the each side of the equation was a derivative wrt the same configuration variable. The way this worked, is that I generated a for loop of the length of elements in phi (16) and iterated a value "i" over this list. I took the first row of the 16x7 multiplied it by lambda and set it equal to the corresponding P element. Then the second row, then so on. The equations were appended to an impact_eqns list where I stored all my impact update laws.

Simulation Explanation

The first simulation graph was the position of the configurations corresponding to the box over time. As I wanted, the x_box and y_box configurations barely changed over time, meanwhile the theta_box rotated as it was being forced by the external force I defined to act theta_box. The second graph was the positions of the jack configuration variables over time. As seen in the plot, the x-jack, y-jack, and theta-jack are all changing value. However, what shows impact is the sharp edges in the plots. As we can see, these sharp points in x_jack, y_jack, and theta_jack align at the same time stamp. By counting these sharp edges on the graph, we can see about 8 impacts. Another way of seeing it, is that if you look at the derivatives (line tangent) to each plot, at each sharp edge, the tangent line changes direction instantaneously. These instantaneous changes in velocity represent impact! Therefore, this simulation seems to represent impact of the jack. The third graph is the velocities of the box configuration variables. Here, we can see the theta_box velocity follows the forcing trajectory it was subjected to. The x_box and y_box velocities have sharp changes (but very small). These are impacts with the jack and the box, because since the box is floating (standstill) along x and y, when it is hit by the jack, it has a sharp change in velocity. Lastly, where impacts can be most clearly seen, is in the graph of velocities of the jack configuration variables. For the velocity of the jack along x, y, and theta, there are instantenous changes in velocity that lign up at certain timestamps. These are clearly impacts, as the x, y, and theta velocities change and are updated by the impact update laws at that instant.