Dynamic System of Long Jump

——Project Report

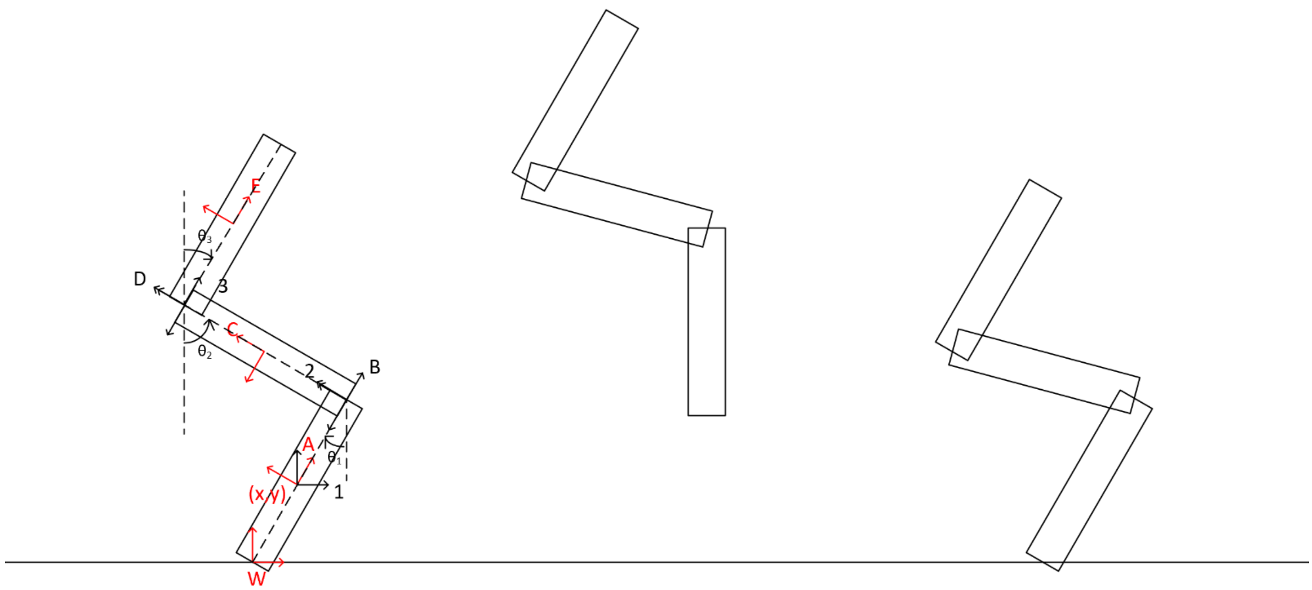
ID: 3226299 Name: Zhisheng Lin

Description of Project:

Original proposal: My project is to study the dynamic system of long jump. In the modeling part, I used three links to represent a man's body, thigh, and calf. In the external force part, to make him jump, I added forces in the x and y directions to the calf. In the degrees of freedom part, in total, I used three angle parameters and two position parameters to represent the specific position and posture of the man during the entire jumping process. In the impact part, after his feet touched the ground, his feet made contact with the ground and a plastic impact occurred.

Changes: There are two differences from the original idea. The first one is that in the maintaining posture part, at the beginning I planned to use joint force to control the posture of the man, but this was too complicated, so after I get the suggestion from you, I linked the three parts with torque springs to maintain his posture. Besides, I also changed the representation of angles since I think the angles I use now are easier to represent transformation matrices and the geometric structure in animation.

Drawing and Transformations:



$$g_{WA} = g\left(I, \begin{bmatrix} x \\ y \end{bmatrix}\right) \cdot g\left(R\left(\frac{\pi}{2} + \theta_1\right), 0\right)$$

$$g_{WC} = g_{WA} \cdot g\left(I, \begin{bmatrix} \frac{l}{2} \\ 0 \end{bmatrix}\right) \cdot g\left(R\left(\theta_2 - \theta_1\right), 0\right) \cdot g\left(I, \begin{bmatrix} \frac{l}{2} \\ 0 \end{bmatrix}\right)$$

$$g_{WE} = g_{WC} \cdot g\left(I, \begin{bmatrix} \frac{l}{2} \\ 0 \end{bmatrix}\right) \cdot g\left(R\left(\theta_3 - \theta_2\right), 0\right) \cdot g\left(I, \begin{bmatrix} \frac{l}{2} \\ 0 \end{bmatrix}\right)$$

Description of Calculation Process：

<u>The calculation of Euler-Lagrange equations, constraints and external forces:</u>

- Define two functions, which are unhat and hat.
- Define some constants and configuration variables

  $$q = \begin{bmatrix} x & y & \theta_1 & \theta_2 & \theta_3 \end{bmatrix}^\top \xrightarrow{\text{derive}} \dot{q} \;\&\; \ddot{q}$$

- Use transformation matrices we defined before to calculate body velocities

  $$V_A = \left(g_{WA}^{-1} \cdot \dot{g_{WA}}\right)^V \quad V_C = \left(g_{WC}^{-1} \cdot \dot{g_{WC}}\right)^V \quad V_E = \left(g_{WE}^{-1} \cdot \dot{g_{WE}}\right)^V$$

- Set up inertia tensor

  $$I_A = I_{6\times6} \quad I_C = I_{6\times6} \quad I_E = I_{6\times6}$$

- Calculate kinetic energy

  $$KE = \frac{1}{2}V_A^T I_A V_A + \frac{1}{2}V_C^T I_C V_C + \frac{1}{2}V_E^T I_E V_E$$

- Calculate potential energy, including elastic and gravity potential energy

  Elastic potential energy: I set k = 30

  $$SP_1 = \frac{1}{2} \cdot k\left(\theta_1 + \frac{\pi}{4}\right)^2 \quad SP_2 = \frac{1}{2} \cdot k\left(\theta_2 - \frac{\pi}{4}\right)^2 \quad SP_3 = \frac{1}{2} \cdot K\left(\theta_3 + \frac{\pi}{4}\right)^2$$

  Gravity potential energy: I set g = 9.8, m = 1

  $$GP = mg \cdot \left(g_{WA}[1,3] + g_{WC}[1,3] + g_{WE}[1,3]\right)$$

  Total potential energy:

  $$V = GP + SP_1 + SP_2 + SP_3$$

- Calculate Lagrangian

  $$L = KE - V$$

- Set up external force

  $F = [60 \quad 60 \quad 0 \quad 0 \quad 0]^T$

- Set up constraint (I just set a random parabola which the center of calf is at)

  $\phi_1 = -0.4503x^2 + 2.0697x - 0.3219 - y$

- Calculate Euler-Lagrange equations

  $$\begin{cases} \frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \lambda \bigtriangledown \phi_1(q) + F \\ \ddot{\phi}_1 = 0 \end{cases} \overset{\text{solve}}{\Longrightarrow} \ddot{q}$$

<u>The calculation of impact update laws:</u>

- Set up impact surface

  $\phi = y - \frac{1}{2}\cos\theta_1 \overset{\text{calculate}}{\longrightarrow} \bigtriangledown\phi(q) = \frac{\partial\phi}{\partial q}$

- Set up configuration variables

  $q^- = [x(\tau^-) \ \ y(\tau^-) \ \ \theta_1(\tau^-) \ \ \theta_2(\tau^-) \ \ \theta_3(\tau^-)]^T$

  $q^+ = [x(\tau^+) \ \ y(\tau^+) \ \ \theta_1(\tau^+) \ \ \theta_2(\tau^+) \ \ \theta_3(\tau^+)]^T$

  $\dot{q}^- = \dot{q} \,.\, subs \, q^-$

  $\dot{q}^+ = \dot{q} \,.\, subs \, q^+$

- Momentum conservation

  $$\left.\begin{matrix} p^- = \frac{\partial L}{\partial \dot{q}}.\text{subs } \dot{q}^- \\ p^+ = \frac{\partial l}{\partial \dot{q}}.\, subs \ \dot{q}^+ \end{matrix}\right\} \overset{Equation\ 1}{\longrightarrow} \left.\frac{\partial L}{\partial \dot{q}}\right|_{\tau^-}^{\tau^+} = \lambda \cdot \frac{\partial\phi}{\partial q}$$

- Speed normal to the impact surface

  $$\overset{Equation\ 2}{\longrightarrow} \quad \frac{\partial\phi}{\partial q} \cdot \dot{q}\,(\tau^+) = 0$$

- Put two equations together to generate the impact equations

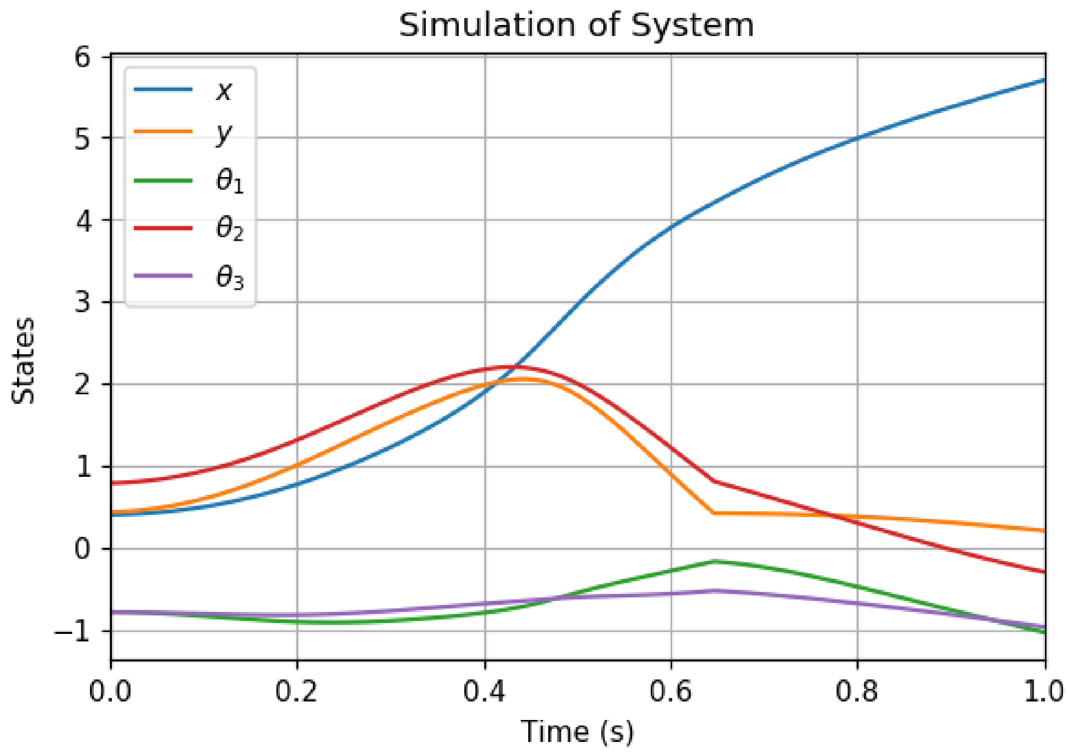
  $$\begin{cases} \left.\frac{\partial L}{\partial \dot{q}}\right|_{\tau^-}^{\tau^+} = \lambda \cdot \frac{\partial\phi}{\partial q} \\ \frac{\partial\phi}{\partial q} \cdot \dot{q}\,(\tau^+) = 0 \end{cases} \overset{solve}{\longrightarrow} Velocity\,Updates$$

- I used numerical method to solve the velocity update in "impact_update" function.

How Code Works:

<u>Reasons of making sense:</u>

I think my code and its simulation and animation make sense.



From the figure of simulation, we can see, x is increasing with time and y increases first and then decrease. Both of these two properties accord with the physical sense of long jump. The singular point in y, θ_1, θ_2, θ_3 also make sense, since, at the instant of plastic impact, y would suddenly stop making this singular point and at the meantime joints would subject to a sudden shake so that is also the reason for the singular point in joints curve. I think the reason of these parameters keeping changing is inertia since I set the constant external force in x and y direction as 60 N but its gravity is near 30 N. Besides, I think there is also existing another reason that after the instant of impact, since the external forces still exist, so the force will still drive them. In the animation, we can see that because of the inertia and external force, the man tends to fall forward and this makes sense too since sometimes the real person would do the same thing.

However, there is still a problem that I want to figure out is that if I increase the simulating time, like to 5, since the external forces always exist, it will ruin the simulation and animation. I have tried to eliminate the external force after plastic impact for many days and I have tried several ways like using conditional statements if, to separate the condition into 3 parts, including before plastic with external force, impact and after impact without external force, and try different dynamical equation to solve and simulate, but there are some problems in numerical computational process. I also tried to use a decreasing x directional force, and after an optimized time, the force in x-direction would decrease to 0 or near zero, but this method would make the man lose his power and I do not think this accord with common sense.

Finally, I think the simulation still makes sense since it accords with the common physical sense and the 1 second simulation time is enough for a man to finish its jumping.

<u>Time of running code:</u>

Solving Euler-Lagrange equations: 3894.623 s

Generating impact equations: 4.008 s

Simulation: 38.874 s

Animation: 9.364 s

Total: 3946.87 s (66 min)