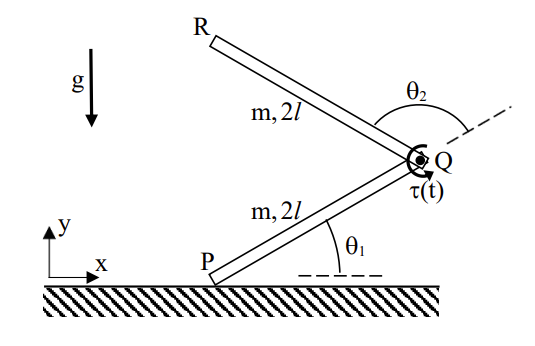
**Final Project –hybrid dynamics of a robotic hopper**

We declare that the project was solely done on our owns. And the main source of resource is based on the lecture notes.

|  |  |
| --- | --- |
| Name | ID |
| Jonathan Oh | 941170383 |
| Xingkun Huang | 941164840 |

**Configuration**

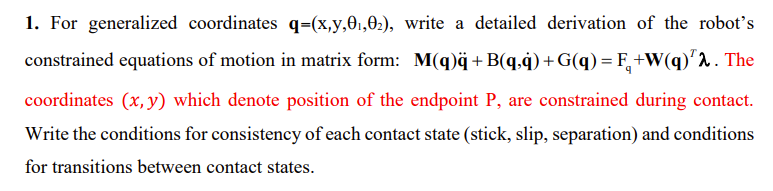
Our robot has following configuration. It’s a simplified model of a robotic hopper composed of two identical links connected by a rotational joint which is actuated by a torque.



We choose the generalized coordinates to be where is the position of the endpoint and are the relative angle between the links. During the entire motion, the actuation torque satisfies the bound . The physical values for the simulations are:

Our objective of the project is to formulate the hybrid dynamics of the robot in all possible contact states, the conditions for transitions between contact states, and the impact law. Also, we will write MATLAB code functions for dynamic simulations of “jumping forward from rest” under a prescribed actuation torque. Finally, we will choose time-profile of actuation torque in attempt to maximize the jump distance .

**Task**



First, we lay out the kinematics of the system. We find the center of gravity of each link. We denote the lower link as link number 1 and the upper one as 2. CG is the center of gravity of the robot.

Their respective velocities are:

Then we can obtain the kinetic energy of the system.

The expression is long, and we won’t display it. The exact expression can be calculated via MATLAB. As for the potential energy,

Then we can write the EOM in a matrix form via Lagrange method.

As for the constraint matrix **,** we would require the velocity at point P.

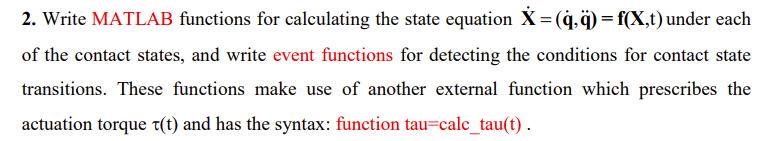
There are three phases for this configuration. To use ODE 45, we need equation of motion for each phase.

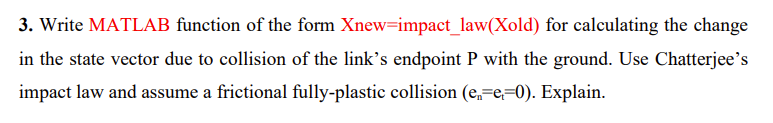
* Stick:
* Slip:
* Contact separation:

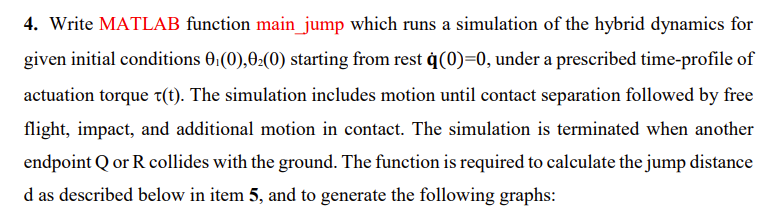
Condition for transition:

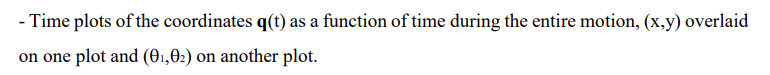
|  |  |
| --- | --- |
| No-slip |  |
| No-slip to slippage |  |
| Slipping |  |
| Slipping to separation |  |
| Separation |  |
| Separation to re-impact |  |

For MATLAB code, we need event function to detect when the transition occurs, and we can terminate the phase calculation and move to a new state.

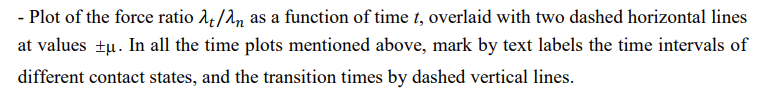


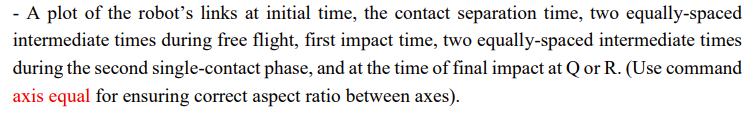


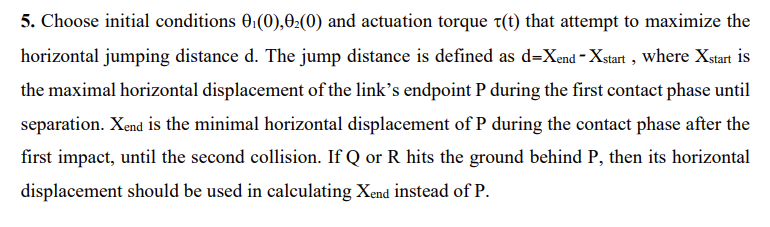












For the jumping challenge, we take the broad jump as consideration. After analyzing the motion of the athlete, we use the movement of his thigh and calf to simulate the two rods in our project. As it shows in the figure, the movement contains 4 stages: Firstly a negative torque is applied on the joint so that the rods would eject out until the two rods are almost parallel. Then a positive torque is applied on the joint in order to clamp the angle between the two rods while it is flying. This would help extend the flying time of the rods. Before the rods landing, a negative torque should be applied on the rods again which enlarge the angle between the two rods. And in this way, the lower rod could reach as far as possible. In The end of the movement, a negative torque should be applied in order to clamp the angle so that the rod on the left hand side would come closer to the rod on the right hand side as much as possible. When we are measuring the distance of jumping, we should count the last point on the left hand side, therefore the less the distance between the landing point of the two rods, the better.