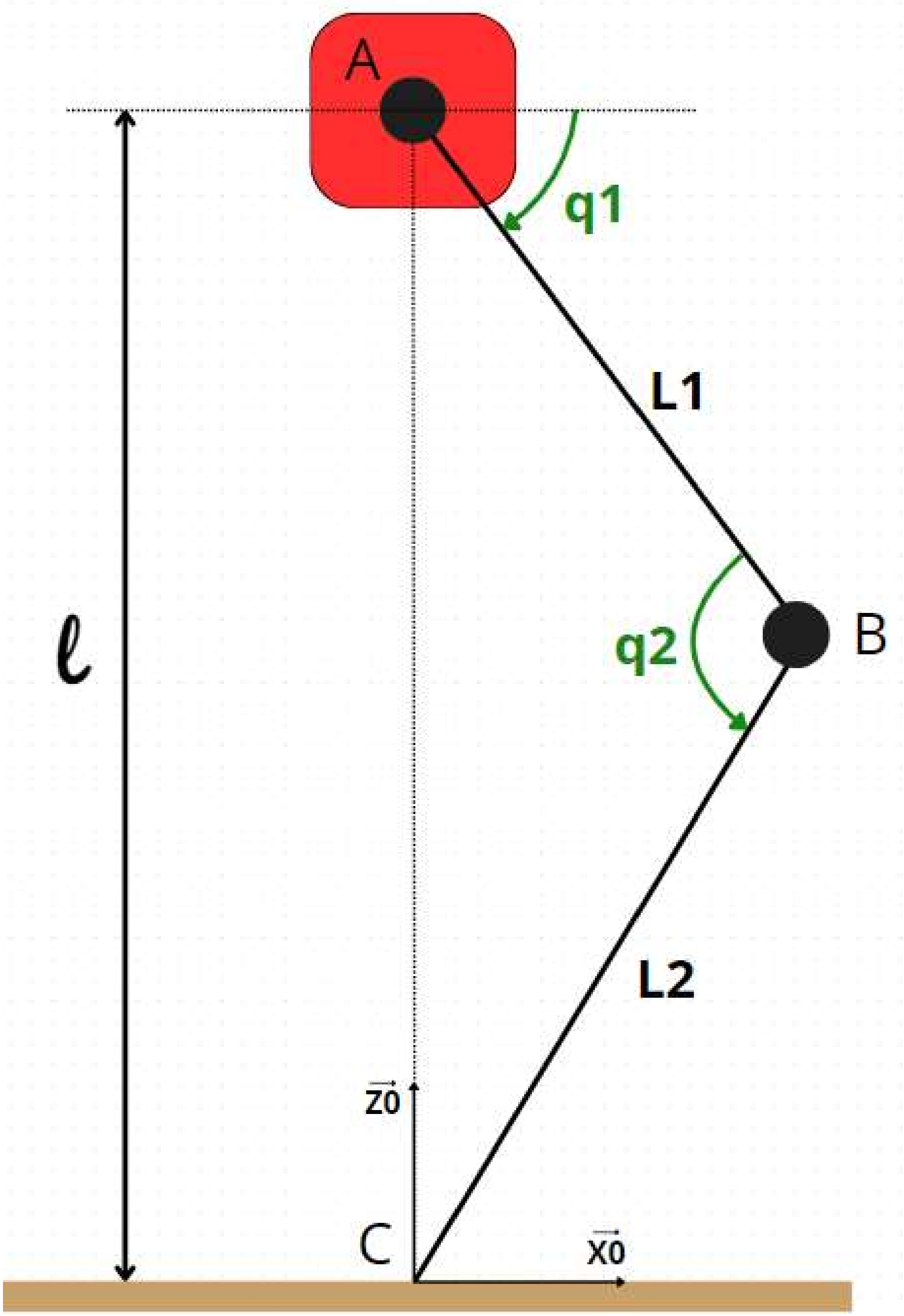
TRAJECTORY ANALYSIS - Jacobian method

We want to analyse the motion of a quadruped robot leg during the pushing phase of a vertical jump.



Direct Geometrics

The leg deployment l between it foot and it frame can be expressed as a function of it joints angles :

$$l(q) = \begin{bmatrix} L_1 cos(q_1) + L_2 cos(q_1 + q_2) \\ L_1 sin(q_1) + L_2 sin(q_1 + q_2) \end{bmatrix}$$

Direct Kinematics

Using that last formula, can express the speed and acceleration of the deployement length l.

$$l(t) = l(q(t)) \text{ where } q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

so we can write the temporal derivate of l as following :

$$\dot{l} = \frac{\partial l}{\partial q} \frac{dq}{dt}$$

$\frac{\partial l}{\partial q}$ is called the jacobian.

$$J = \begin{bmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial x}{\partial q_2} \\ \frac{\partial y}{\partial q_1} & \frac{\partial y}{\partial q_2} \end{bmatrix} = \begin{bmatrix} -L_1 sin(q_1) - L_2 sin(q_1 + q_2) & -L_2 sin(q_1 + q_2) \\ L_1 cos(q_1) + L_2 cos(q_1 + q_2) & L_2 cos(q_1 + q_2) \end{bmatrix}$$

$$\ddot{l} = \dot{J}\,\dot{q} + J\ddot{q}$$

where

$$\dot{J} = \begin{bmatrix} -L_1 \dot{q}_1 cos(q_1) - L_2 \dot{q}_1 cos(q_1 + q_2) & -L_2 \dot{q}_2 cos(q_1 + q_2) \\ -L_1 \dot{q}_1 sin(q_1) - L_2 \dot{q}_1 sin(q_1 + q_2) & -L_2 \dot{q}_2 sin(q_1 + q_2) \end{bmatrix}$$

Inverse Kinematics

We want to calculate the angular speed and acceleration :

$$\dot{q} = J^{-1} \dot{l}$$

and

$$\ddot{q} = J^{-1}(\ddot{l} - \dot{J}\,\dot{q})$$

Inverse Dynamics

By équality of powers, we can find the matrix of the motors torque :

$$P = \dot{q}^T \tau = \dot{l}^T F \Leftrightarrow \dot{q}^T \tau = (J\dot{q})^T F$$

so $\tau = J^T F$ were F is the pushing force.

Tuning the trjectoy

Since we admit that l is purely vertical, we also can express l depending on a polynom of n degree:
$l = \sum_{i=0}^{n} \alpha_i \frac{t^i}{i!}$

We still have to find the degree n of the polynom such as the initial conditions α.

In order to do that, we have to choose some constraints for our movement. The degree of our polynom must be equal to the number of constraints we choose.

constraints matrix : $c = \begin{bmatrix} l_0 = 0.2 \\ v_0 = 0 \\ a_0 = 0 \\ l_f = 0.4 \\ v_f = 4 \end{bmatrix}$

Here for example, we choose to set the initial high of the leg, set initial speed and acceleration to 0, and we want a final speed at a final leg deployement length. Those are 5 constraints ; so our polynom will be a 5 degree polynom.
Once that is done, we still have to chose the duration of the jump that want. We call it T.
Now we express in a matrix the equation of each condition we choose depending on the time :

sate matrix : $C = \begin{bmatrix} l_0 \\ v_0 \\ a_0 \\ l_T \\ v_T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & T & \frac{T^2}{2} & \frac{T^3}{6} & \frac{T^4}{24} \\ 0 & 1 & T & \frac{T^2}{2} & \frac{T^3}{6} \end{bmatrix}$

In that way, we can set this relation :

$c = C\alpha$ where $alpha = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix}$

Finally we can find the initial conditions that correspond to the trajectory we wanted :

$\alpha = C^{-1}c$

If the trajectory that we want does not match with the time chose to realize it, or simply the trajectory can not be done, the C matrix will not be inversible.