Two-Link Manipulator

2DOF ABB IRB 910SC (SCARA)

Content

1. Technologies
2. Forward /Inverse Kinematics
3. Differential Kinematics
4. Dynamics: Euler–Lagrange Equation
5. Motion Planning

Technologies

Technologies

Python version 3:

https://www.python.org/downloads/

Numpy (Array computing Lib.)

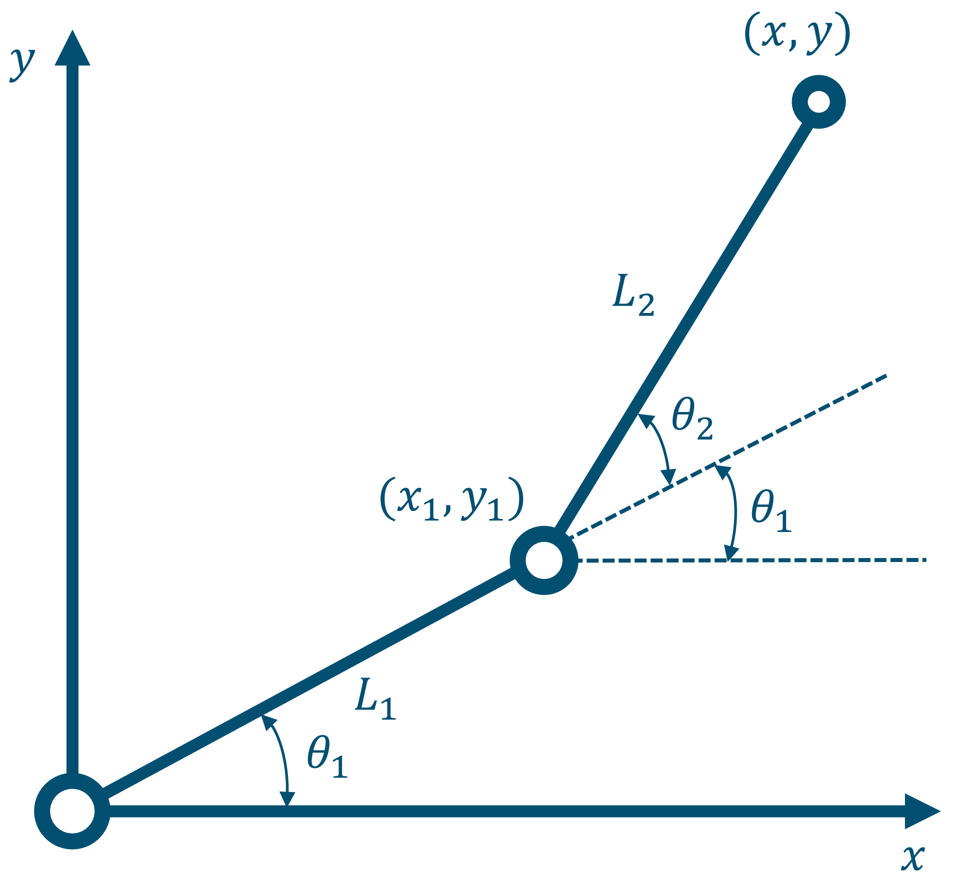
```
pip3 install numpy
```

Mataplotlib (Visualization Lib.)

```
pip3 install matplotlib
```

Forward /Inverse Kinematics

Forward/Inverse Kinematics Demonstration



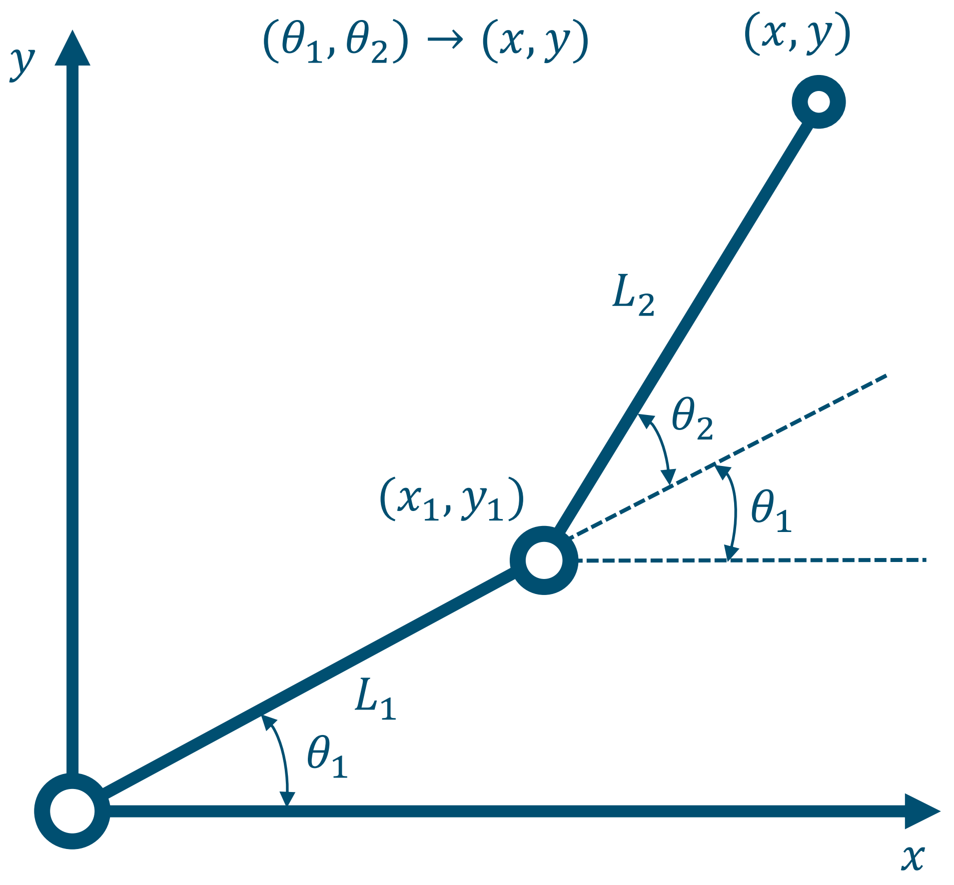
Forward Kinematics

$$(\theta_1, \theta_2) \rightarrow (x, y)$$

Inverse Kinematics

$$(\theta_1, \theta_2) \leftarrow (x, y)$$

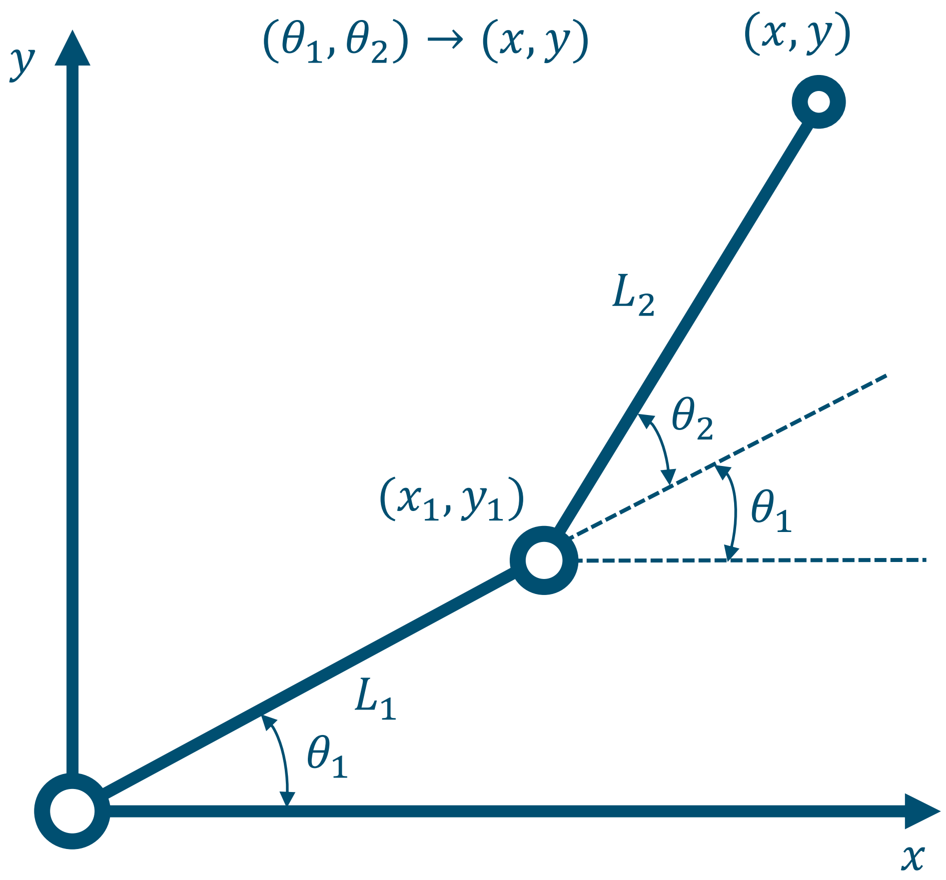
Forward Kinematics



a) DH Parameters – Table?

b) Position of the x_1, y_1 and x, y?

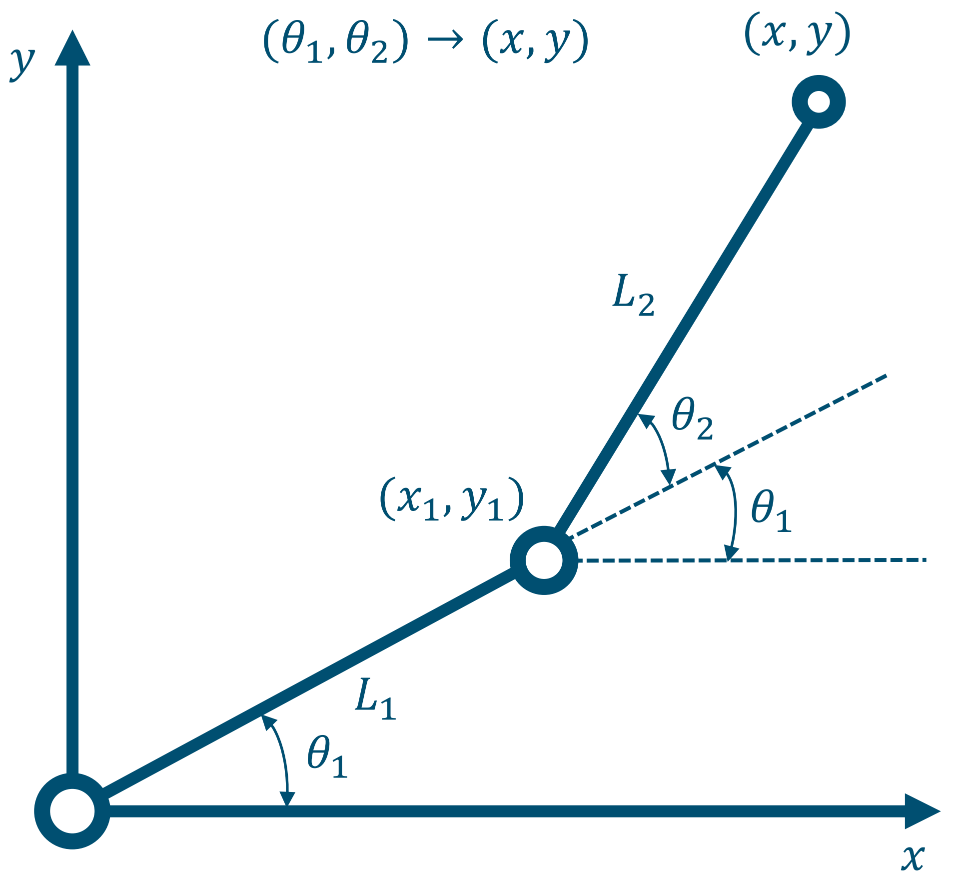
Forward Kinematics



a) DH Parameters – Table?

Link	a_i	α_i	d_i	θ_i
1	$a_1 = L_1$	0	0	θ_1
2	$a_2 = L_2$	0	0	θ_2

Forward Kinematics

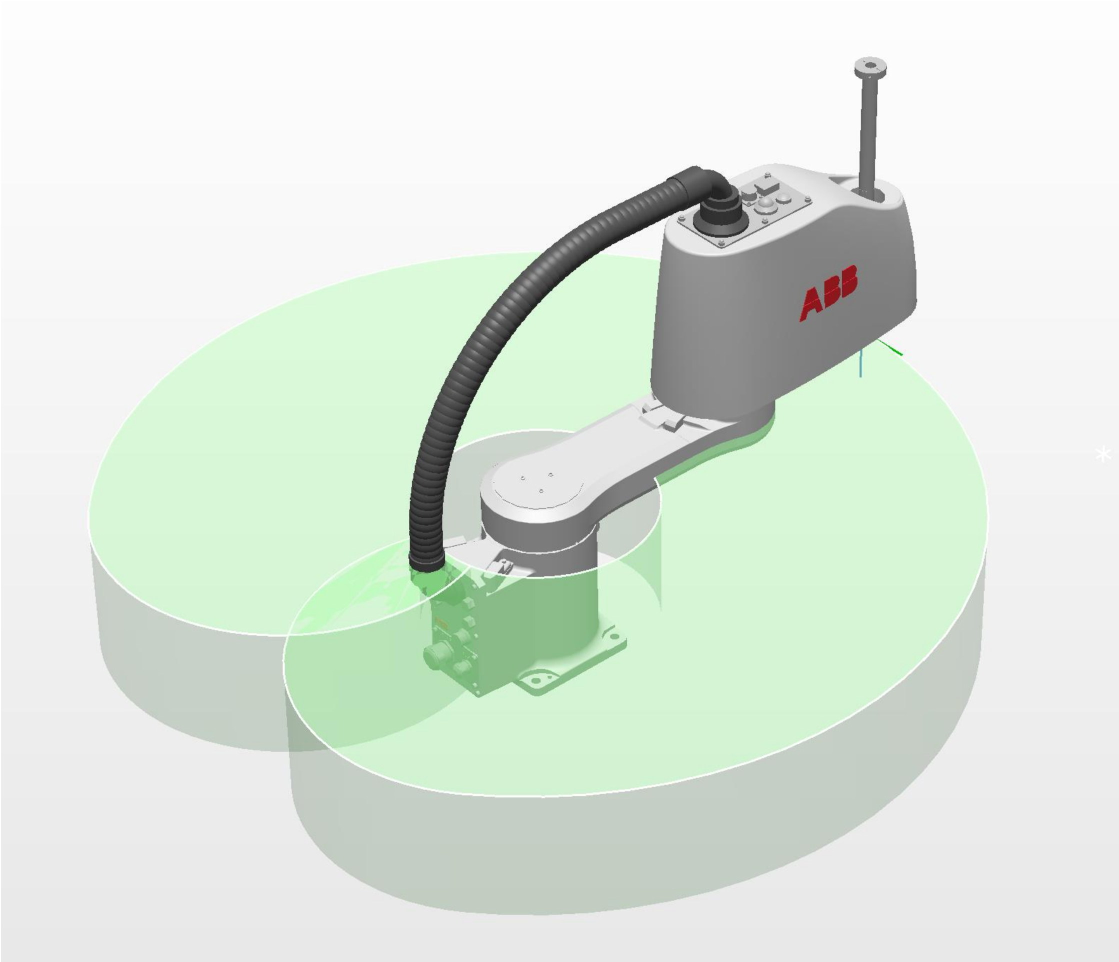


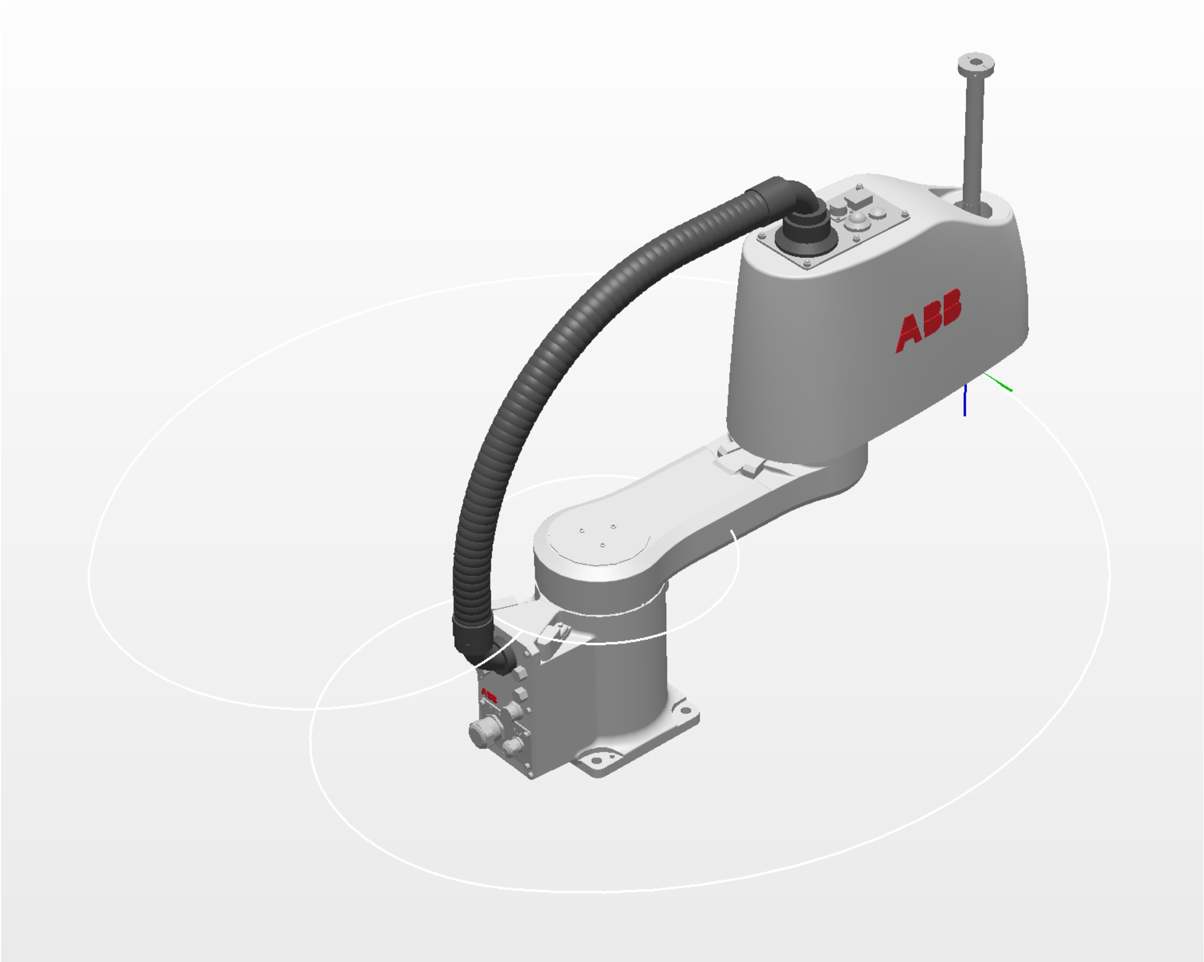
b) Position of the x_1, y_1 and x, y?

$$x_1 = L_1 cos\theta_1$$
$$y_1 = L_1 sin\theta_1$$

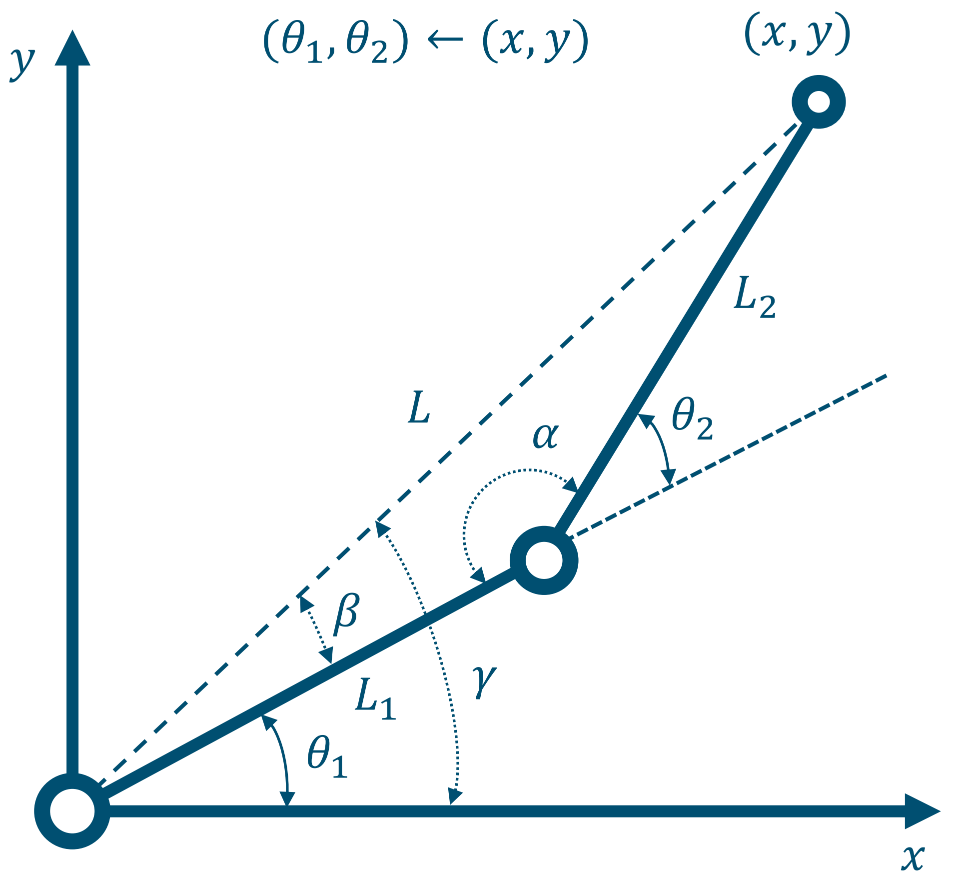
$$x = L_1 cos\theta_1 + L_2 cos(\theta_1 + \theta_2)$$
$$y = L_1 sin\theta_1 + L_2 sin(\theta_1 + \theta_2)$$

Work Envelope



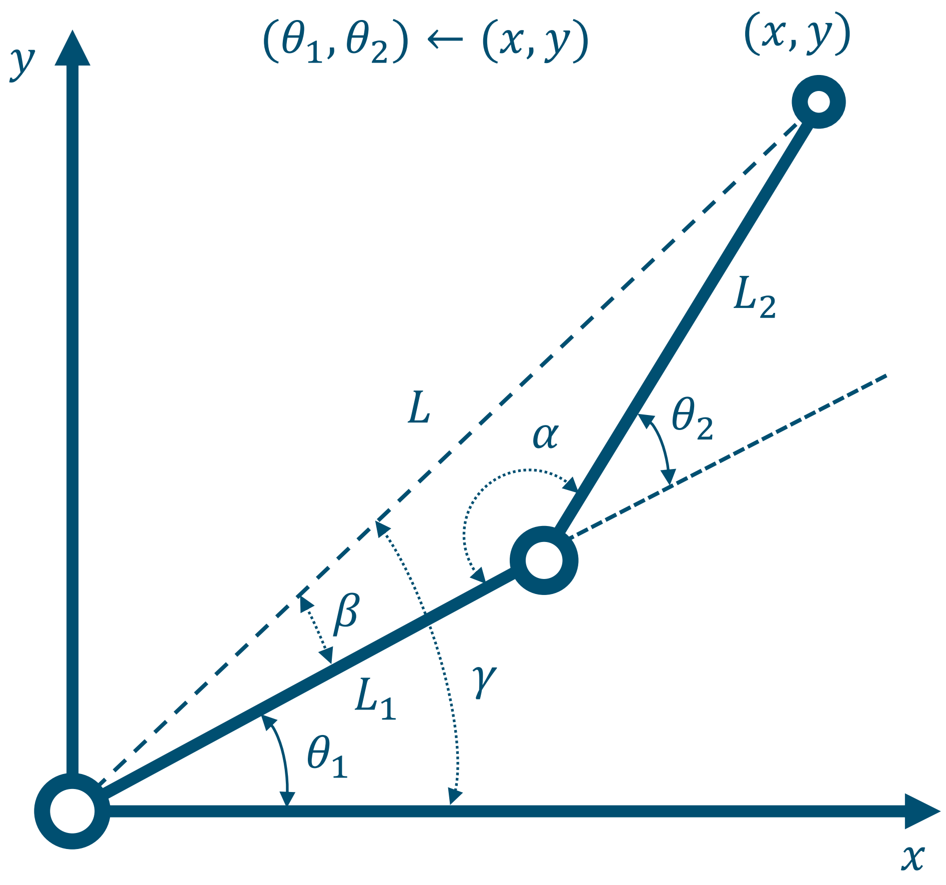


Inverse Kinematics



a) Rotation of the $\boldsymbol{\theta_1}, \boldsymbol{\theta_2}$?

Inverse Kinematics



$$L = \sqrt{x^2 + y^2}$$

$$R_{2DOF} \begin{cases} \theta_1 = \gamma - \beta \\ \theta_2 = \pi - \alpha \end{cases}$$

Law of cosines (Cosine Theorem)

$$L^2 = L_1^2 + L_2^2 - 2L_1L_2cos\alpha$$

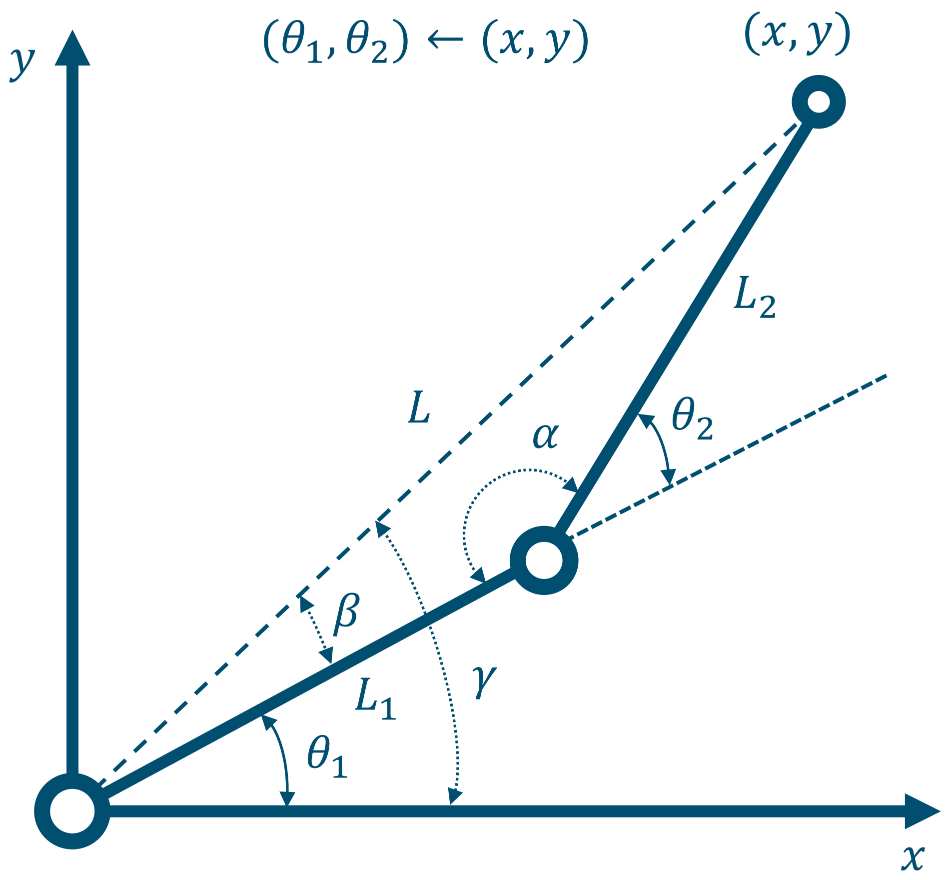
$$L_2^2 = L_1^2 + L^2 - 2L_1Lcos\beta$$

$$cos\alpha = \frac{L_1^2 + L_2^2 - L^2}{2L_1L_2}$$

$$cos\beta = \frac{L_1^2 + L^2 - L_2^2}{2L_1L}$$

$$tan\gamma = \frac{y}{x}$$

Inverse Kinematics



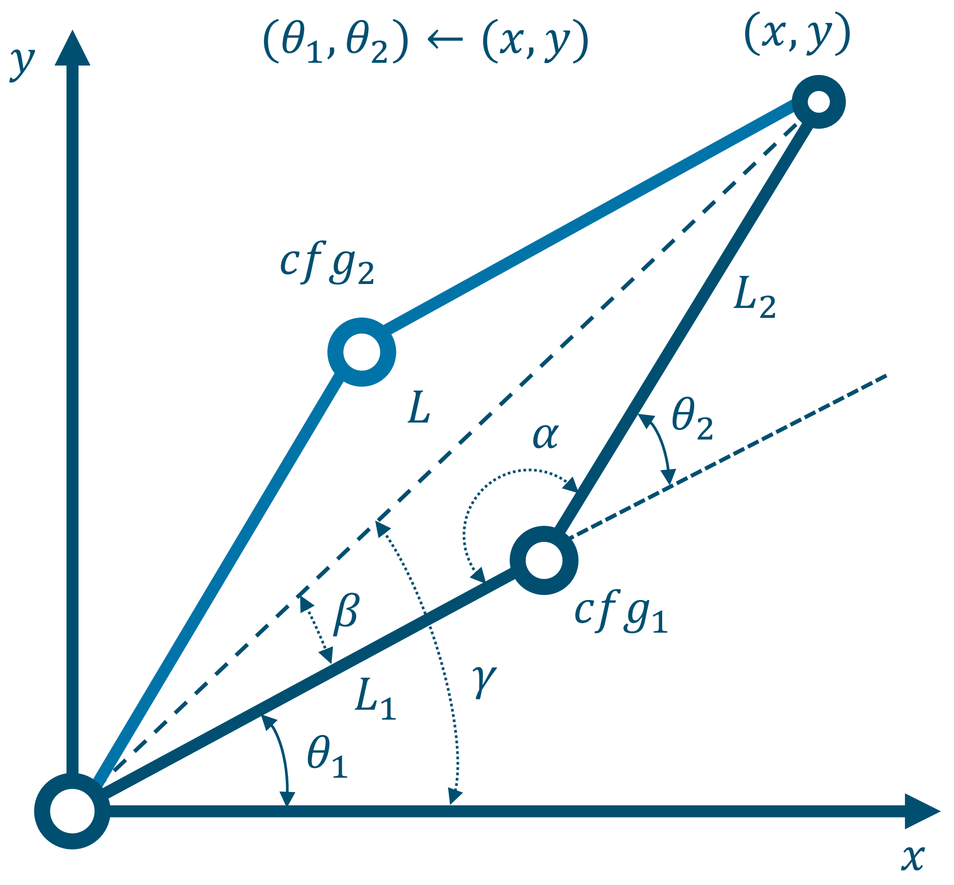
$$R_{2DOF}\begin{cases}\theta_1 = \gamma - \beta \\ \theta_2 = \pi - \alpha\end{cases}$$

Inverse Trigonometric Functions

$$\theta_1 = arctan\frac{y}{x} - \arccos\frac{L_1^2 + L^2 - L_2^2}{2L_1L}$$

$$\theta_2 = \pi - \arccos\frac{L_1^2 + L_2^2 - L^2}{2L_1L_2}$$

Inverse Kinematics



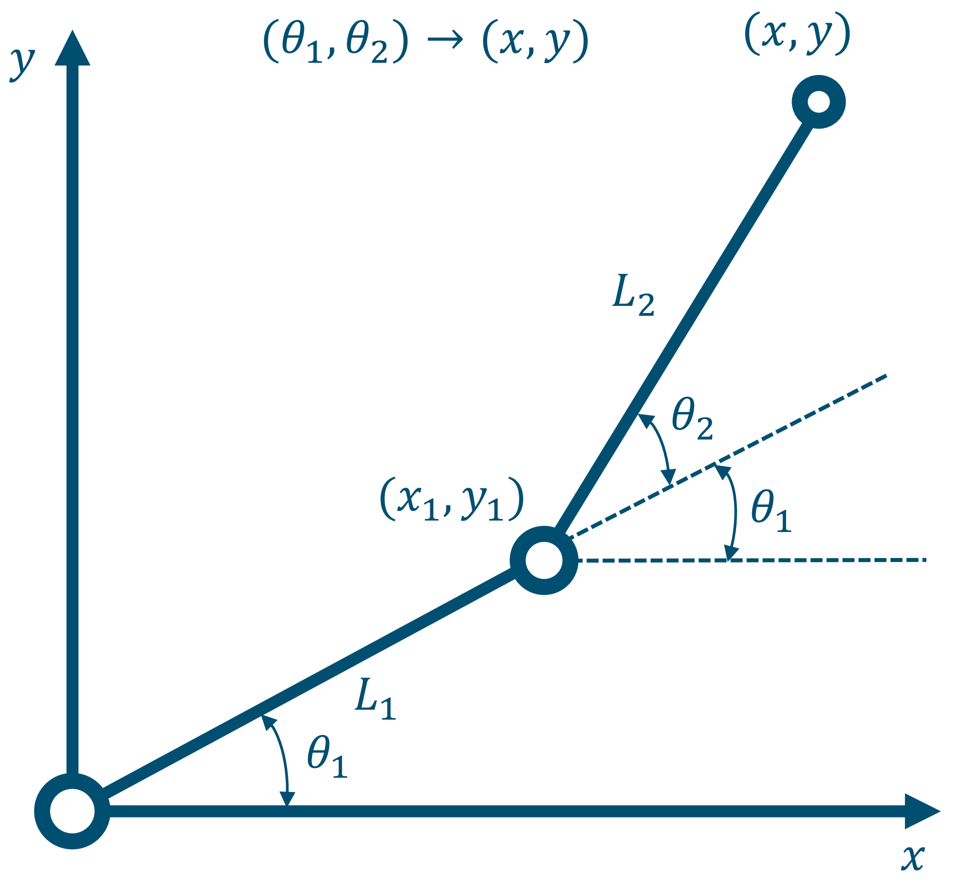
$$cfg_1 \begin{cases} \theta_1 = \gamma - \beta \\ \theta_2 = \pi - \alpha \end{cases}$$

$$cfg_2 \begin{cases} \theta_1 = \gamma + \beta \\ \theta_2 = \alpha - \pi \end{cases}$$

The number of solutions depends on the number of joints in the manipulator.

Differential Kinematics

Differential Kinematics



Forward Kinematics

$$x = L_1 cos\theta_1 + L_2 cos(\theta_1 + \theta_2)$$
$$y = L_1 sin\theta_1 + L_2 sin(\theta_1 + \theta_2)$$

Jacobian Matrix

$$\begin{bmatrix} p\dot{x} \\ p\dot{y} \end{bmatrix} = J(\theta_1, \theta_2) \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

Differential Kinematics

Jacobian Matrix

$$\begin{bmatrix} p\dot{x} \\ p\dot{y} \end{bmatrix} = J(\theta_1, \theta_2) \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} =$$

$$= \begin{bmatrix} -L_1 sin\theta_1 - L_2 sin(\theta_1 + \theta_2) & -L_2 sin(\theta_1 + \theta_2) \\ L_1 cos\theta_1 + L_2 cos(\theta_1 + \theta_2) & L_2 cos(\theta_1 + \theta_2) \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix}$$

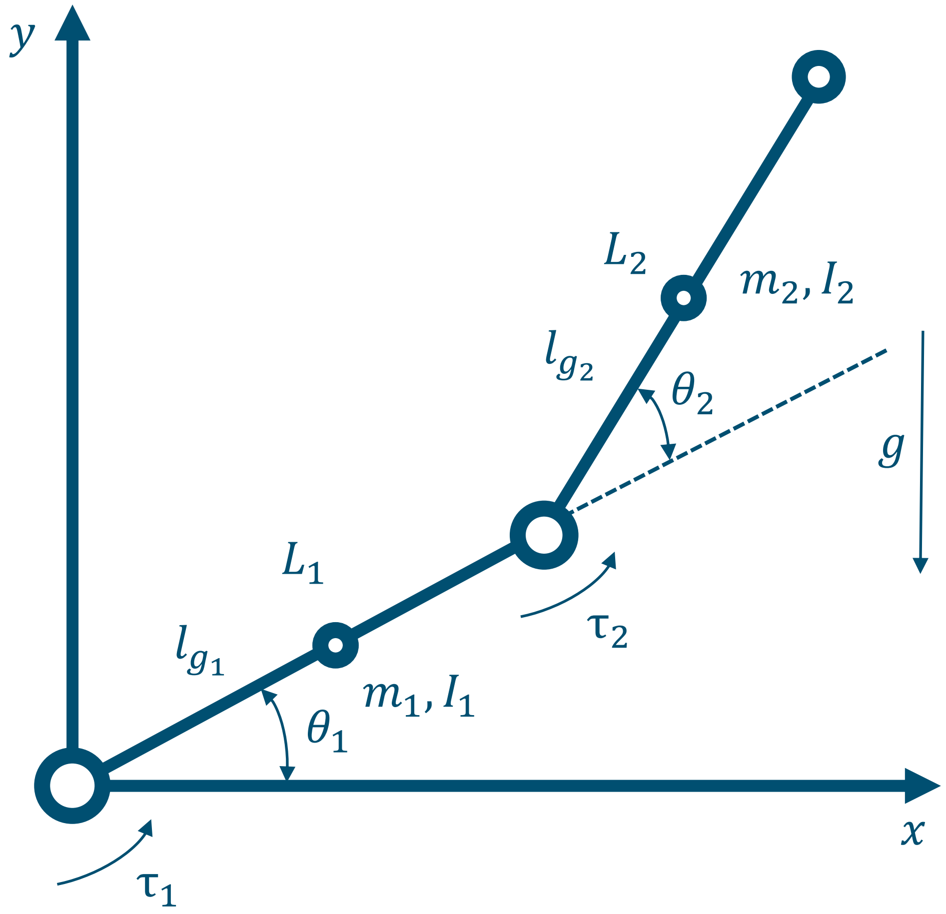
Singular Jacobian

$$\det\left(\begin{bmatrix} -L_1 sin\theta_1 - L_2 \sin(\theta_1 + \theta_2) & -L_2 \sin(\theta_1 + \theta_2) \\ L_1 cos\theta_1 + L_2 \cos(\theta_1 + \theta_2) & L_2 \cos(\theta_1 + \theta_2) \end{bmatrix}\right) = 0$$

Dynamics: Euler–Lagrange Equation

Dynamics: Euler–Lagrange Equation

$$\tau = M(\theta)\ddot{\theta} + b(\theta, \dot{\theta}) + g(\theta)$$



Lagrange Method

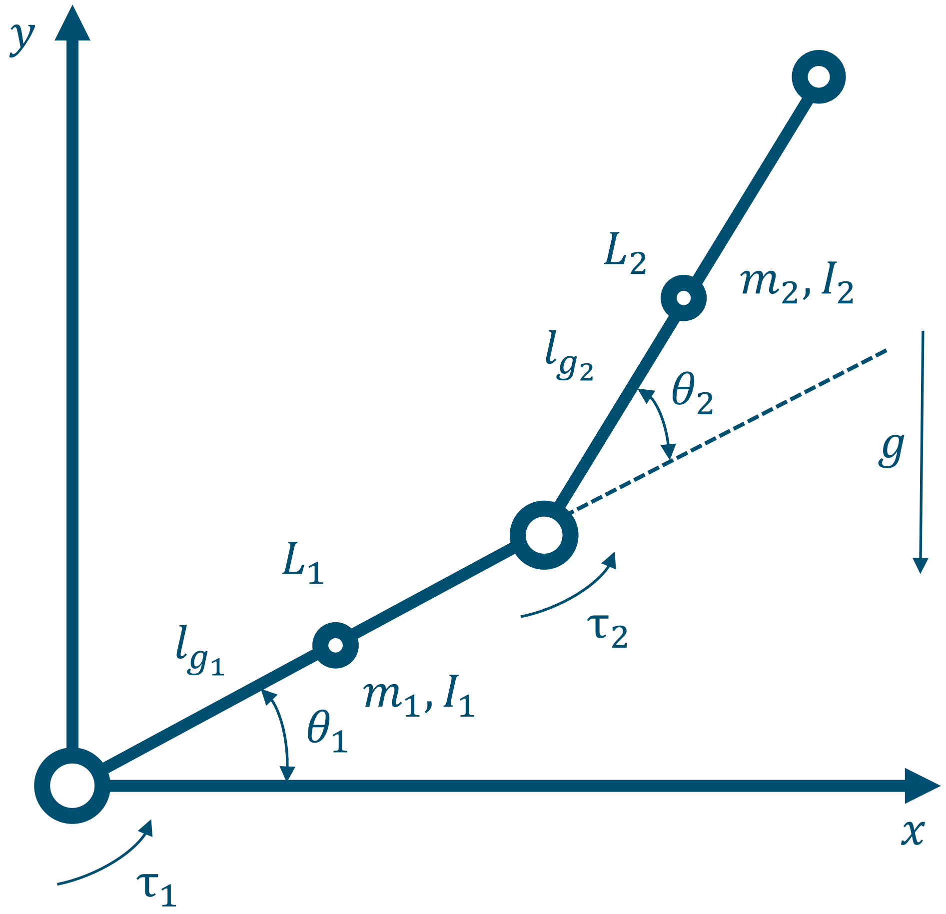
$$\tau_i = \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i}\right) - \left(\frac{\partial \mathcal{L}}{\partial \theta_i}\right)$$

$$\mathcal{L} = T - U$$

$$\mathcal{L}(\theta, \dot{\theta}) = T(\theta, \dot{\theta}) - U(\theta)$$

Dynamics: Euler–Lagrange Equation

$$\tau = M(\theta)\ddot{\theta} + b(\theta, \dot{\theta}) + g(\theta)$$



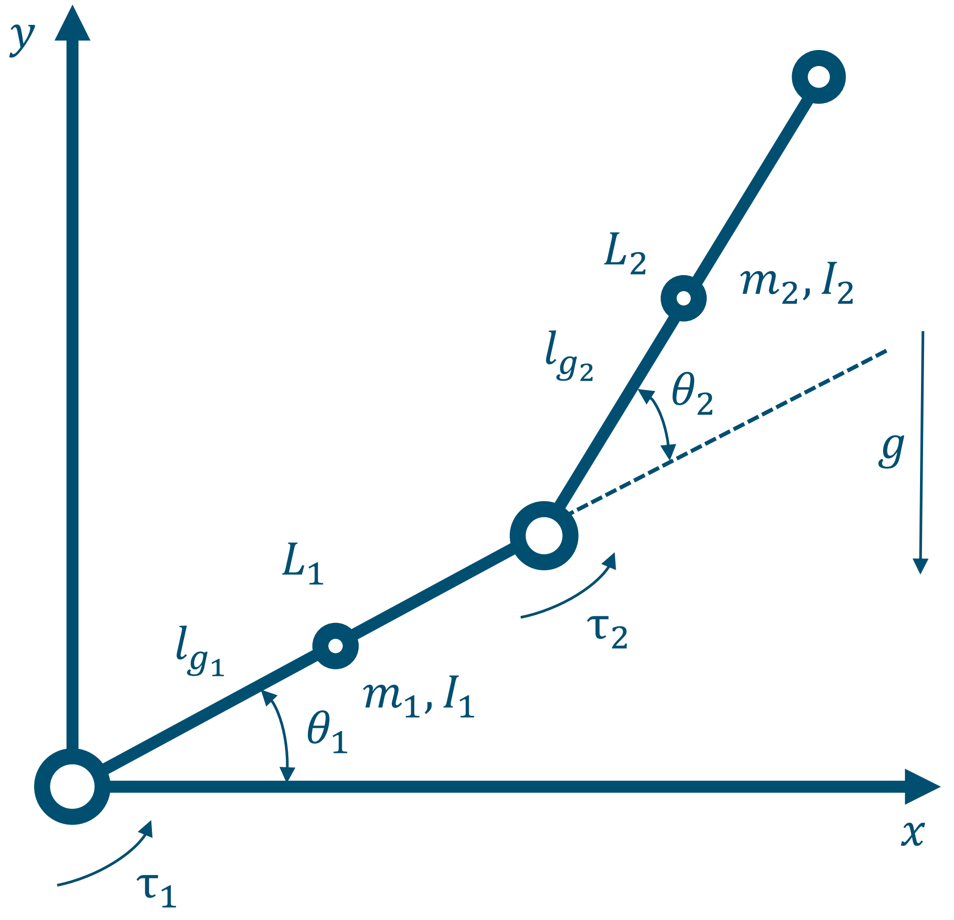
$$x_{m_1} = l_{g_1}\cos(\theta_1) \qquad \dot{x}_{m_1} = -l_{g_1}\dot{\theta}_1\sin(\theta_1)$$
$$y_{m_1} = l_{g_1}\sin(\theta_1) \qquad \dot{y}_{m_1} = l_{g_1}\dot{\theta}_1\cos(\theta_1)$$

$$x_{m_2} = L_1\cos(\theta_1) + l_{g_2}\cos(\theta_1 + \theta_2)$$
$$y_{m_2} = L_1\sin(\theta_1) + l_{g_2}\sin(\theta_1 + \theta_2)$$

$$\dot{x}_{m_2} = -L_1\dot{\theta}_1\sin(\theta_1) - l_{g_2}(\dot{\theta}_1 + \dot{\theta}_2)sin(\theta_1 + \theta_2)$$
$$\dot{y}_{m_2} = L_1\dot{\theta}_1\cos(\theta_1) + l_{g_2}(\dot{\theta}_1 + \dot{\theta}_2)cos(\theta_1 + \theta_2)$$

Dynamics: Euler–Lagrange Equation

$$\tau = M(\theta)\ddot{\theta} + b(\theta,\dot{\theta}) + g(\theta)$$



Kinetic energy

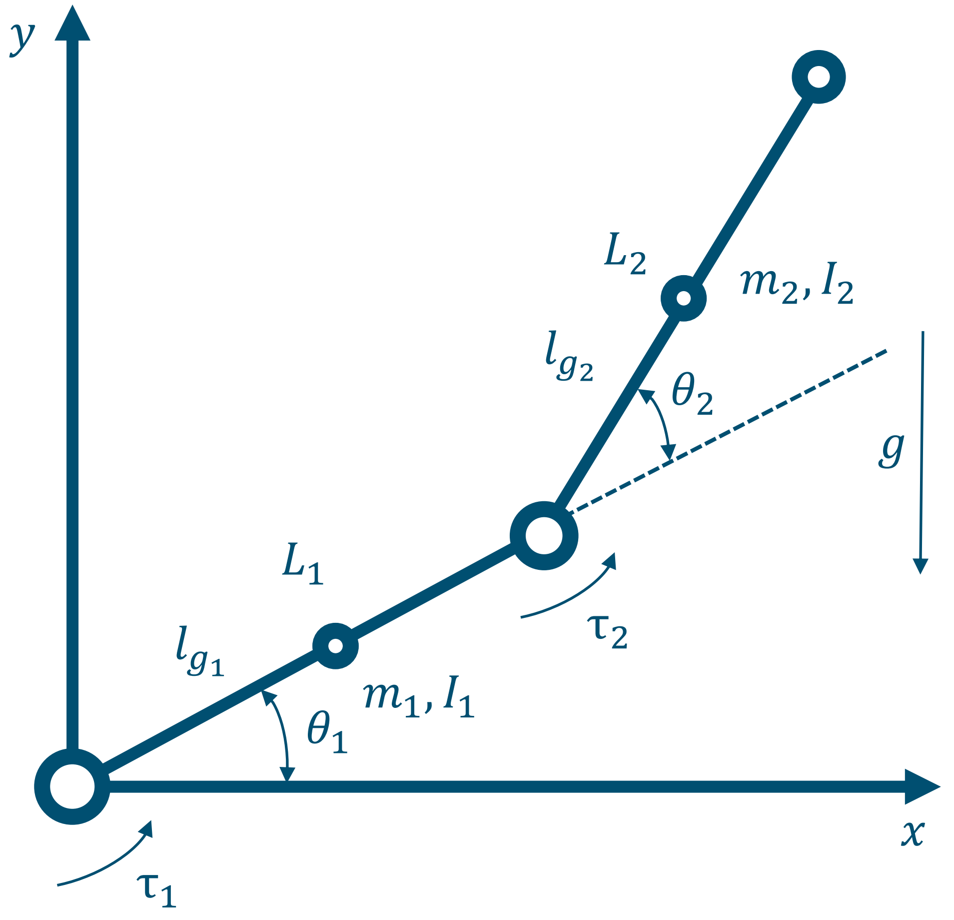
$$T_1 = \frac{1}{2} m_1\left(\dot{x}_{m_1}^2 + \dot{y}_{m_1}^2\right) + \frac{1}{2} I_1 \dot{\theta}_1^2$$
$$T_2 = \frac{1}{2} m_2\left(\dot{x}_{m_2}^2 + \dot{y}_{m_2}^2\right) + \frac{1}{2} I_2 (\dot{\theta}_1 + \dot{\theta}_2)^2$$

Potential energy

$$U_1 = m_1 l_{g_1} g sin(\theta_1)$$
$$U_2 = m_2 g(L_1 sin(\theta_1) + l_{g_2}\sin(\theta_1 + \theta_2))$$

Dynamics: Euler–Lagrange Equation

$$\tau = M(\theta)\ddot{\theta} + b(\theta, \dot{\theta}) + g(\theta)$$



Torque equation

$$\mathcal{L} = T - U$$

$$\tau_i = \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i}\right) - \left(\frac{\partial \mathcal{L}}{\partial \theta_i}\right)$$

$$\mathcal{L} = (T_1 + T_2) - (U_1 + U_1)$$

$$\tau = [\tau_1; \tau_2]^T, \theta = [\theta_1; \theta_2]^T$$

Dynamics: Euler–Lagrange Equation

Torque equations can then be split up into the form of the general equation. The different parameters can be extracted from this, by isolating the angular acceleration, velocities and positions.

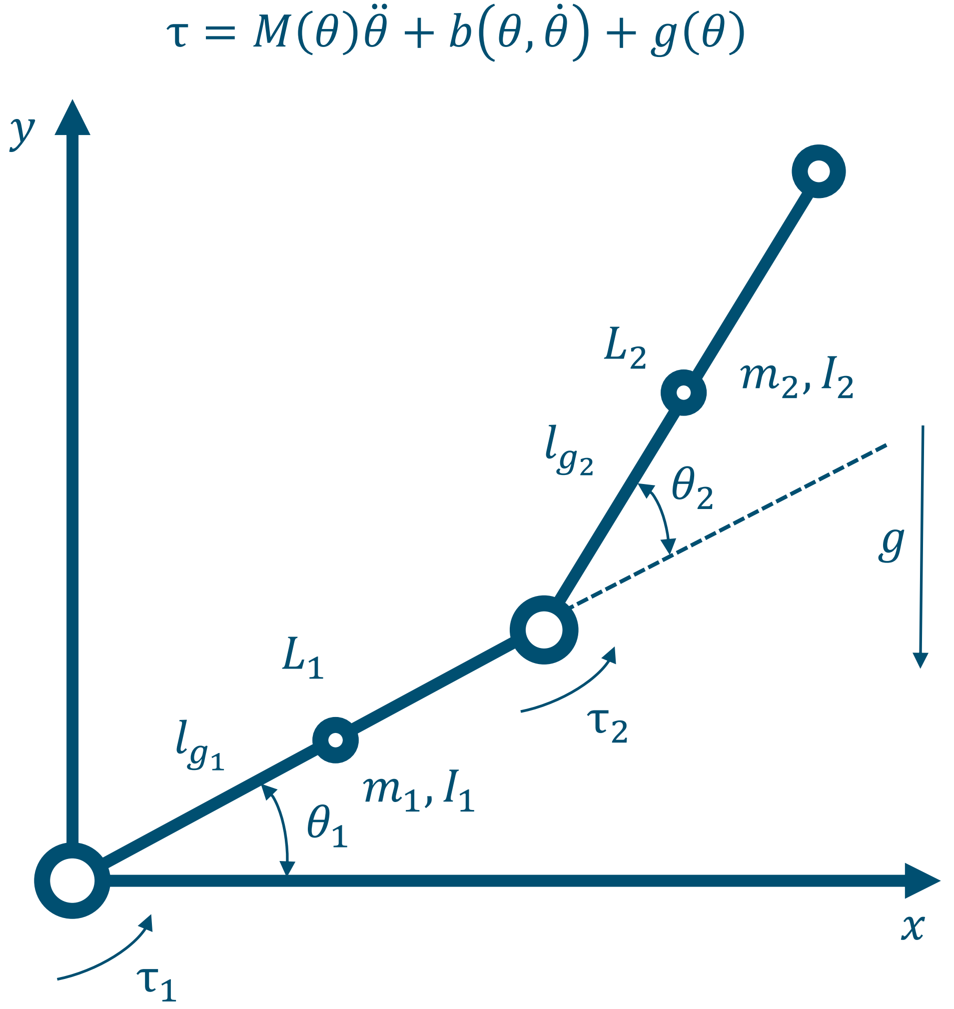
$$\tau = M(\theta)\ddot{\theta} + b\left(\theta, \dot{\theta}\right) + g(\theta)$$

$$M(\theta) = \begin{bmatrix} I_1 + I_2 + m_1 l_{g_1}^2 + m_2(L_1^2 + l_{g_2}^2 + 2L_1 l_{g_2} + 2L_1 l_{g_2}\cos(\theta_2)) & I_2 + m_2(l_{g_2}^2 + L_1 l_{g_2}\cos(\theta_2)) \\ I_2 + m_2(l_{g_2}^2 + L_1 l_{g_2}\cos(\theta_2)) & I_2 + m_2 l_{g_2}^2 \end{bmatrix}$$

$$b\left(\theta, \dot{\theta}\right) = \begin{bmatrix} -m_2 L_1 l_{g_2}\dot{\theta}_2\left(2\dot{\theta}_1 + \dot{\theta}_2\right)\sin(\theta_2) \\ m_2 L_1 l_{g_2}\dot{\theta}_1^2 \sin(\theta_2) \end{bmatrix}$$

$$g(\theta) = \begin{bmatrix} m_1 g l_{g_1}\cos(\theta_1) + m_2 g(L_1\cos(\theta_1) + l_{g_2}\cos(\theta_1 + \theta_2)) \\ m_2 g l_{g_2}\cos(\theta_1 + \theta_2) \end{bmatrix}$$

Dynamics: Euler–Lagrange Equation



Motion equation

$$\tau = M(\theta)\ddot{\theta} + b(\theta,\dot{\theta}) + g(\theta)$$

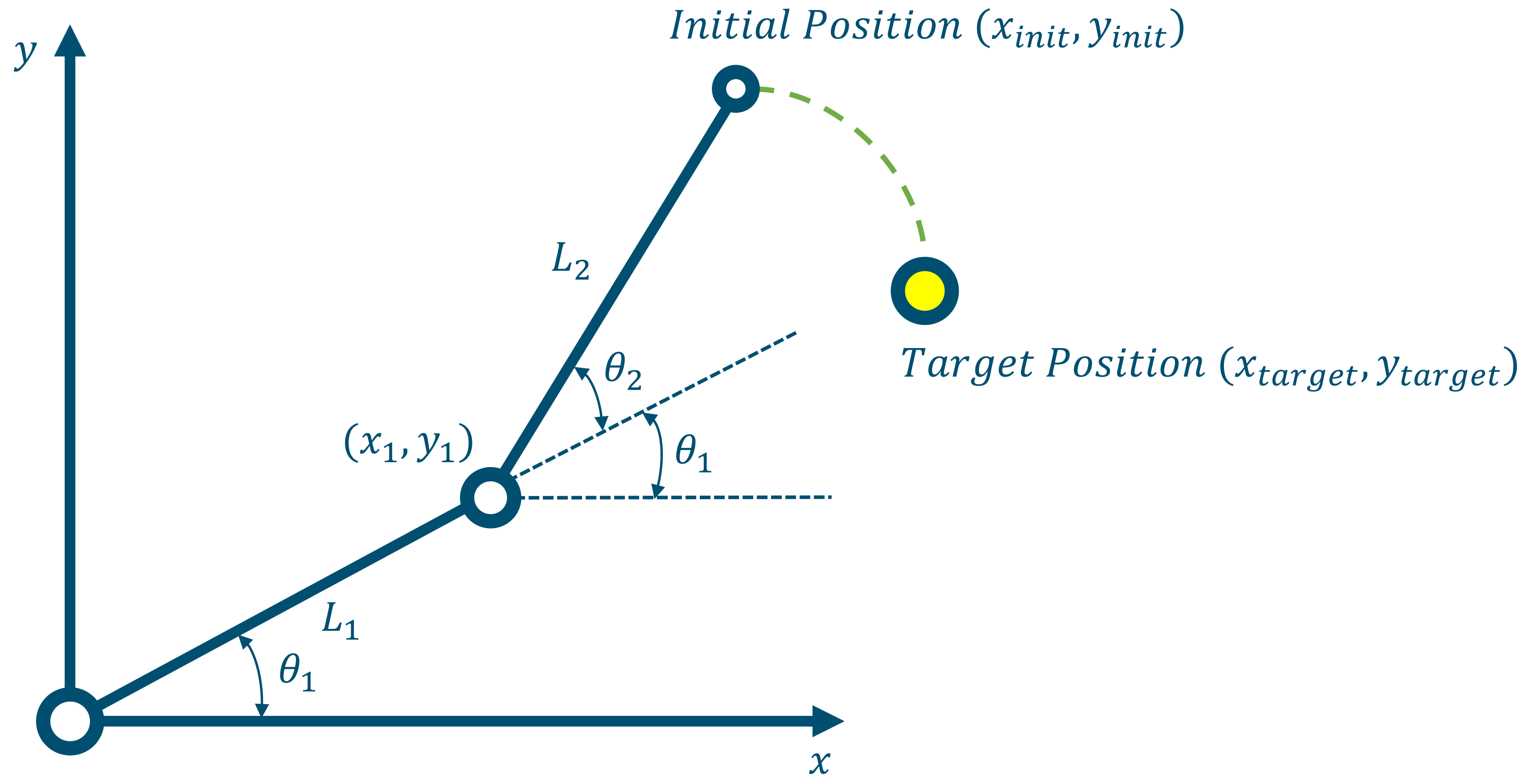
$$\ddot{\theta} = M(\theta)^{-1}\left(-b(\theta,\dot{\theta}) - g(\theta) + \tau\right)$$

Ordinary Differential Equations (ODE)

$$\frac{d}{dt}\begin{bmatrix}\theta \\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\dot{\theta} \\ M(\theta)^{-1}\left(-b(\theta,\dot{\theta}) - g(\theta) + \tau\right)\end{bmatrix}$$

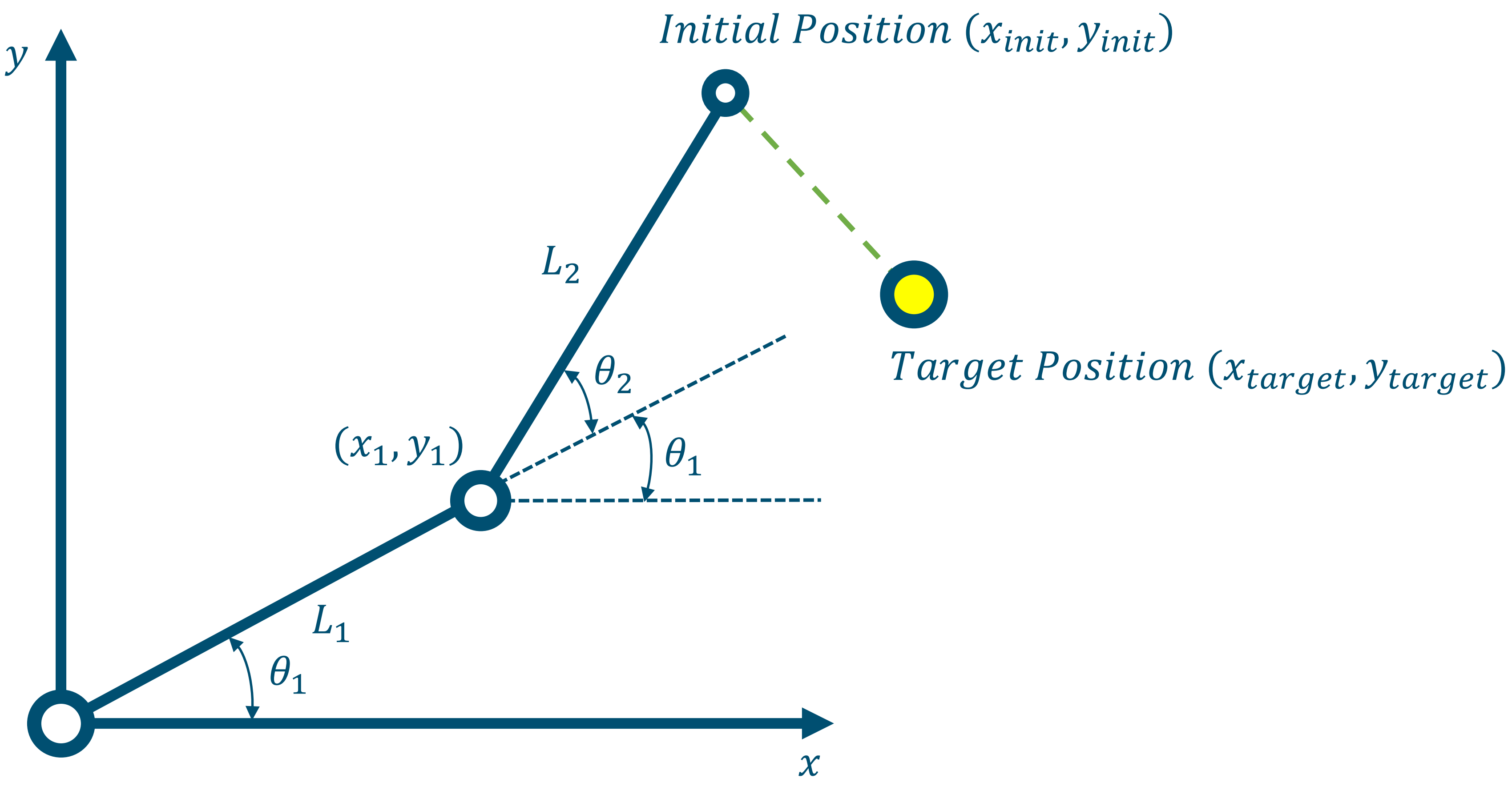
Motion Planning

Joint-Space (Joint Interpolation)



Joint-Space

Operational-Space (Cartesian-Space) Linear Interpolation



Operational-Space (Cartesian-Space)