Robotics I HW

Marco Mustacchi

1 Equation of Motion

For derivation of the equation of motion in the *joint space* we have two methods: *Lagrange formulation* and *Newton-Euler formulation*. In this case, I have decided to adopt the first one, since it is a sistematic method and so it is easier than the second one. The first thing to do is to decide the generalized coordinates, two in our case, one for each DOF. I have chosen $\boldsymbol{q} = \begin{bmatrix} d_1 & \theta_2 \end{bmatrix}^T$.

Then, the equation of motion can be derived as follow:

$$\boxed{\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = \xi_i} \tag{1}$$

Then we have to introduce the so called *Lagrangian*:

$$\mathcal{L} = \mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \mathcal{U}(\boldsymbol{q}) \tag{2}$$

where T denotes the kinetic energy and U the potential energy, computed as follow:

$$\mathcal{T} = \sum_{i=1}^{n} (\mathcal{T}_{\ell_i} + \mathcal{T}_{m_i}) \qquad \mathcal{U} = \sum_{i=1}^{n} (\mathcal{U}_{\ell_i} + \mathcal{U}_{m_i})$$

After some computations we get for kinetic and potential energy:

$$\mathcal{T} = \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{B}(\boldsymbol{q})\dot{\boldsymbol{q}} \qquad \mathcal{U} = -\sum_{i=1}^{n} \left(m_{\ell_i} \boldsymbol{g}_0^T \boldsymbol{p}_{\ell_i} + m_{m_i} \boldsymbol{g}_0^T \boldsymbol{p}_{m_i} \right)$$

At this point, recalling equation 2, and taking the derivative required by the Lagrange equations of motion 1 we get:

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\xi} \tag{3}$$

Rewriting:

$$\underbrace{\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}}}_{\text{Inertia}} + \underbrace{\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}}_{\text{Centrifugal and Coriolis}} + \underbrace{\boldsymbol{g}(\boldsymbol{q})}_{\text{gravity}} = \underbrace{\boldsymbol{\xi}}_{\text{non conservative forces}} \tag{4}$$

Substituting the non-conservative forces:

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \underbrace{\boldsymbol{F}_v \dot{\boldsymbol{q}}}_{\text{Viscous friction}} + \underbrace{\boldsymbol{F}_s \,\mathrm{sgn}(\dot{\boldsymbol{q}})}_{\text{Static friction}} + \boldsymbol{g}(\boldsymbol{q}) = \underbrace{\boldsymbol{\tau}}_{\text{actuators torques}} - \underbrace{\boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{h}_e}_{\text{Environment}} \tag{5}$$

Using the assumption that all frictions are negligible and there are no end-effector contact forces with the environment, the resulting equation of motion in compact form is:

$$\boxed{\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}} \tag{6}$$

where:

$$\boldsymbol{B}(\boldsymbol{q}) = \sum_{i=1}^{n} \left(m_{\ell_i} \boldsymbol{J}_P^{(\ell_i)T} \boldsymbol{J}_P^{(\ell_i)} + \boldsymbol{J}_O^{(\ell_i)T} \boldsymbol{R}_i \boldsymbol{I}_{\ell_i}^i \boldsymbol{R}_i^T \boldsymbol{J}_O^{(\ell_i)} + m_{m_i} \boldsymbol{J}_P^{(m_i)T} \boldsymbol{J}_P^{(m_i)} + \boldsymbol{J}_O^{(m_i)T} \boldsymbol{R}_{m_i} \boldsymbol{I}_{m_i}^{m_i} \boldsymbol{R}_{m_i}^T \boldsymbol{J}_O^{(m_i)} \right)$$

So in order to compute $\boldsymbol{B}(\boldsymbol{q})$ we need to derive the geometric Jacobians $\boldsymbol{J}$, the rotation matrices $\boldsymbol{R}$ and the moments of inertia $\boldsymbol{I}$.

Given the positions we can compute the geometric Jacobians, recalling that they can be easily computed using the formula:

$$\boldsymbol{J} = \begin{bmatrix} J_{P_1} & \cdots & J_{P_n} \\ J_{O_1} & \cdots & J_{O_n} \end{bmatrix}$$

where:

$$\begin{bmatrix} J_{P_i} \\ J_{O_i} \end{bmatrix} = \begin{cases} \begin{bmatrix} z_{i-1} \\ 0 \end{bmatrix} & \text{for a prismatic joint} \\ \begin{bmatrix} z_{i-1} \times (p_e - p_{i-1}) \\ z_{i-1} \end{bmatrix} & \text{for a revolute joint} \end{cases}$$

respectively for the prismatic and for the revolute joint.
So we have to define position and orientation of each single link and motor with respect to the base frame. To do so, we need to assign some reference frames to our model. I used the Denavit-Hartenberg convention that we have seen in class:
It is worth noticing that, in order to keep the figure cleaner, the parameters a_1 and a_2 have been

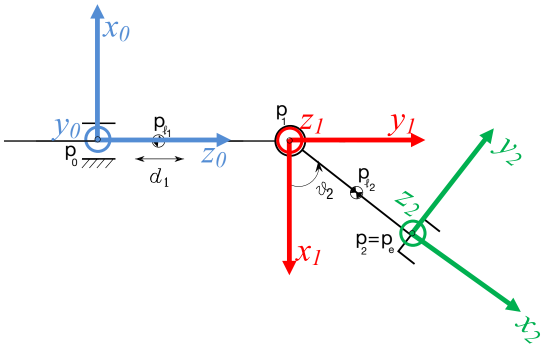


Figure 1: frames attached with the Denavit-Hartenberg convention

omitted. In addition, the gravitational component faces downwards, resulting in $g_0 = \begin{bmatrix} -g & 0 & 0 \end{bmatrix}^T$ as we will se later.

Giving that, we can compute positions and orientations both with respect to the base frame $[x_0, y_0, z_0]$. So for our particular case we obtain:

$$p_{m_1} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad p_{m_2} = \begin{bmatrix} 0 \\ 0 \\ d_1 \end{bmatrix} \quad p_{\ell_1} = \begin{bmatrix} 0 \\ 0 \\ d_1 - \ell_1 \end{bmatrix} \quad p_{\ell_2} = \begin{bmatrix} -c_2\ell_2 \\ 0 \\ d_1 + s_2\ell_2 \end{bmatrix} \quad p_e = \begin{bmatrix} -c_2 a_2 \\ 0 \\ d_1 + s_2 a_2 \end{bmatrix}$$

Now from we need to compute the rotation matrices to define the orientations, recalling that as far as the rotation matrices of the motors are concerned, we can only take care of the z-axis:

$$R_1 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad R_2 = \begin{bmatrix} -c_2 & s_2 & 0 \\ 0 & 0 & 1 \\ s_2 & c_2 & 0 \end{bmatrix} \quad R_{m_1} = \begin{bmatrix} * & * & 0 \\ * & * & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad R_{m_2} = \begin{bmatrix} * & * & 0 \\ 0 & 0 & 1 \\ * & * & 0 \end{bmatrix}$$

Furthermore, given the relation $p_{m_i} = p_{i_1}$ and $z_{m_i} = z_{i_1}$, for $i = 1, 2$, we can see that $R_{m_2} = R_1$.
Given these we can compute geometric Jacobians relative to the links:

$$\boldsymbol{J}^{(l)} = \begin{bmatrix} J_P^{(l_1)} & J_P^{(l_2)} \\ J_O^{(l_1)} & J_O^{(l_2)} \end{bmatrix} = \begin{bmatrix} \boldsymbol{z_0} & \boldsymbol{z_1} \times (\boldsymbol{p_2} - \boldsymbol{p_1}) \\ \boldsymbol{0} & \boldsymbol{z_1} \end{bmatrix}$$
$$= \begin{bmatrix} 0 & s_2 l_2 \\ 0 & 0 \\ 1 & c_2 l_2 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

The geometric Jacobian relative to the motors is a little bit more difficult to compute, since it is assumed that $p_{m_i} = p_{i_1}$ and $z_{m_i} = z_{i_1}$, for $i = 1, 2$.

Indeed, for instance, since motor 1 is attached to point 0, it's linear velocity is always 0. However, the angular velocity of the motor 1 is different from 0, and depends on the position on the joint 1 (since the radius can change). So it is better to express each geometric Jacobian explicity with respect to each of the two generalized coordinates, obtaining:

$$
\begin{aligned}
\boldsymbol{J_P}^{(m_1)} &= \begin{bmatrix} J_{P_1}^{(m_1)} & J_{P_2}^{(m_1)} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} & \boldsymbol{J_O}^{(m_1)} &= \begin{bmatrix} J_{O_1}^{(m_1)} & J_{O_2}^{(m_1)} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} & &= \begin{bmatrix} k_{r1}\boldsymbol{z_0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} & &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_{r1} & 0 \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix}
\end{aligned}
$$

for motor 1 and:

$$
\begin{aligned}
\boldsymbol{J_P}^{(m_2)} &= \begin{bmatrix} J_{P_1}^{(m_2)} & J_{P_2}^{(m_2)} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} & \boldsymbol{J_O}^{(m_2)} &= \begin{bmatrix} J_{O_1}^{(m_2)} & J_{O_2}^{(m_2)} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{z_0} & 0 \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} & &= \begin{bmatrix} 0 & k_{r2}\boldsymbol{z_1} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_{r1} & 0 \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} & &= \begin{bmatrix} 0 & 0 \\ 0 & k_{r2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix}
\end{aligned}
$$

for motor 2.
So back to our equation 6 and replacing the expressions we have just found and after some computations we get:

$$
\begin{aligned}
\boldsymbol{B}(\boldsymbol{q}) &= \sum_{i=1}^{n} \left(m_{\ell_i} \boldsymbol{J}_P^{(\ell_i)T} \boldsymbol{J}_P^{(\ell_i)} + \boldsymbol{J}_O^{(\ell_i)T} \boldsymbol{R}_i \boldsymbol{I}_{\ell_i}^{i} \boldsymbol{R}_i^T \boldsymbol{J}_O^{(\ell_i)} + m_{m_i} \boldsymbol{J}_P^{(m_i)T} \boldsymbol{J}_P^{(m_i)} + \boldsymbol{J}_O^{(m_i)T} \boldsymbol{R}_{m_i} \boldsymbol{I}_{m_i}^{m_i} \boldsymbol{R}_{m_i}^T \boldsymbol{J}_O^{(m_i))} \right) \\
&= 50 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + 50 \begin{bmatrix} 1 & c_2\ell_2 \\ c_2\ell_2 & \ell_2^2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & I_{\ell_{2zz}}^{\ell_2} \end{bmatrix} + 5 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} k_{r1}^2 I_{m_{1zz}}^{m_1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & k_{r2}^2 I_{m_{2zz}}^{m_2} \end{bmatrix}
\end{aligned}
$$

Summing up all the contributions, the inertia matrix is

$$
B(q) = \begin{bmatrix} b_{11} & b_{12}(\theta_2) \\ b_{21}(\theta_2) & b_{22} \end{bmatrix} \qquad \text{where} \qquad
\begin{aligned}
b_{11} &= m_{\ell_1} + m_{\ell_2} + m_{m_1} + k_{r1}^2 I_{m_{1zz}}^{m_1} \\
b_{12} &= b_{21} = m_{\ell_2} c_2 \ell_2 \\
b_{22} &= m_{\ell_2} l_2^2 + I_{\ell_{2zz}}^{\ell_2} + k_{r2}^2 I_{m_{2zz}}^{m_2}
\end{aligned}
$$

Then the choice of the matrix $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ in 6 is not unique. In this case I chose to derive it using the *Christoffel symbols of the first type.* By doing that, the generic element of $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ is computed as

$$
c_{ij} = \sum_{k=1}^{n} c_{ijk} \dot{q}_k \qquad \text{where} \quad c_{ijk} = \frac{1}{2} \left(\frac{\partial b_{ij}}{\partial q_k} + \frac{\partial b_{ik}}{\partial q_j} - \frac{\partial b_{jk}}{\partial q_i} \right)
$$

where c_{ijk} are the *Christoffel symbols.* So we have

$$
\begin{aligned}
c_{111} &= \tfrac{1}{2} \tfrac{\partial b_{11}}{\partial q_1} = 0 \\
c_{112} &= c_{121} = \tfrac{1}{2} \tfrac{\partial b_{11}}{\partial q_2} = 0 \\
c_{122} &= \tfrac{\partial b_{12}}{\partial q_2} - \tfrac{1}{2} \tfrac{\partial b_{22}}{\partial q_1} = -m_{\ell_2} s_2 \ell_2 \\
c_{211} &= \tfrac{\partial b_{21}}{\partial q_1} - \tfrac{1}{2} \tfrac{\partial b_{11}}{\partial q_2} = 0 \\
c_{212} &= c_{221} = \tfrac{1}{2} \tfrac{\partial b_{22}}{\partial q_1} = 0 \\
c_{222} &= \tfrac{1}{2} \tfrac{\partial b_{22}}{\partial q_2} = 0
\end{aligned}
\qquad \Rightarrow \qquad
\begin{aligned}
c_{11} &= c_{111}\dot{d}_1 + c_{112}\dot{\theta}_2 = 0 \\
c_{12} &= c_{121}\dot{d}_1 + c_{122}\dot{\theta}_2 = -m_{\ell_2} s_2 \ell_2 \dot{\theta}_2 \\
c_{21} &= c_{211}\dot{d}_1 + c_{212}\dot{\theta}_2 = 0 \\
c_{22} &= c_{221}\dot{d}_1 + c_{222}\dot{\theta}_2 = 0
\end{aligned}
$$

The gravitational terms from 6 is computed as follow

$$
g_i(q) = \frac{\partial \mathcal{U}}{\partial q_i} = -\sum_{j=1}^{n} \left(m_{\ell_j} g_0^T J_{P_i}^{(\ell_j)}(q) + m_{m_j} g_0^T J_{P_i}^{(m_j)}(q) \right)
$$

Since in this case $g_0 = \begin{bmatrix} -g & 0 & 0 \end{bmatrix}^T$ and so we get

$$g_1 = (-m_{\ell_1} - m_{\ell_2} - m_{m_2}) \begin{bmatrix} -g & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad g_2 = -m_{\ell_2} \begin{bmatrix} -g & 0 & 0 \end{bmatrix} \begin{bmatrix} s_2\ell_2 \\ 0 \\ c_2\ell_2 \end{bmatrix}$$
$$= 0 \qquad\qquad = m_{\ell_2} g \ell_2 s_2$$

So back to our equation 6 in matrix form

$$\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} \begin{bmatrix} \ddot{d}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} \begin{bmatrix} \dot{d}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \tag{7}$$

where $u = \begin{bmatrix} \tau_1 & \tau_2 \end{bmatrix}^T$ represents the actuators torques applied to the system.
Then after replacing the corresponding terms and some elementary computations

$$\begin{cases} \left(m_{\ell_1} + m_{\ell_2} + m_{m_1} + k_{r1}^2 I_{m_{1zz}}^{m_1}\right) \ddot{d}_1 + (m_{\ell_2} c_2 \ell_2)\, \ddot{\theta}_2 - m_{\ell_2} s_2 \ell_2 \dot{\theta}_2^2 = \tau_1 \\ (m_{\ell_2} \ell_2 c_2)\, \ddot{d}_1 + \left(m_{\ell_2} l_2^2 + I_{\ell_{2zz}}^{\ell_2} + k_{r2}^2 I_{m_{2zz}}^{m_2}\right) \ddot{\theta}_2 + m_{\ell_2} g \ell_2 s_2 = \tau_2 \end{cases} \tag{8}$$

and finally expressing in terms of the accelerations of the generalized coordinates, the equations of motion are:

$$\begin{cases} \ddot{d}_1 &= \frac{1}{\left(m_{\ell_1} + m_{\ell_2} + m_{m_1} + k_{r1}^2 I_{m_{1zz}}^{m_1}\right)} \left[(-m_{\ell_2} c_2 \ell_2)\, \ddot{\theta}_2 + m_{\ell_2} s_2 \ell_2 \dot{\theta}_2^2 + \tau_1\right] \\ \ddot{\theta}_2 &= \frac{1}{\left(m_{\ell_2} l_2^2 + I_{\ell_{2zz}}^{\ell_2} + k_{r2}^2 I_{m_{2zz}}^{m_2}\right)} \left[(-m_{\ell_2} \ell_2 c_2)\, \ddot{d}_1 - m_{\ell_2} g \ell_2 s_2 + \tau_2\right] \end{cases} \tag{9}$$

where as we can expect the motion of the generalized coordinates are dependent one with the other.

2 Property

The choice of the matrix $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is not unique. However, using the *Christoffel symbol of the first type* as we have done in this case, the matrix $\boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ which is defined as:

$$\boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \dot{\boldsymbol{B}}(\boldsymbol{q}) - 2\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$$

is skew-symmetric. This means that the following relation holds:

$$\boxed{A \text{ skew-symmetric} \quad \Longleftrightarrow \quad A^T = -A}$$

The definition of a skew-symmetric matrix is a square matrix whose transpose equals its negative. That is, it satisfies the condition:

$$\boldsymbol{w}^T \boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \boldsymbol{w} = 0$$

Let's try to see if it is true also in our example. The derivative of the matrix $\dot{\boldsymbol{B}}(\boldsymbol{q})$ is

$$\dot{\boldsymbol{B}}(\boldsymbol{q}) = \begin{bmatrix} \dot{b}_{11} & \dot{b}_{12} \\ \dot{b}_{21} & \dot{b}_{22} \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^2 \frac{\partial b_{11}}{\partial q_k} \dot{q}_k & \sum_{k=1}^2 \frac{\partial b_{12}}{\partial q_k} \dot{q}_k \\ \sum_{k=1}^2 \frac{\partial b_{21}}{\partial q_k} \dot{q}_k & \sum_{k=1}^2 \frac{\partial b_{22}}{\partial q_k} \dot{q}_k \end{bmatrix} = \begin{bmatrix} 0 & -m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 \\ -m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 & 0 \end{bmatrix}$$

The matrix $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is

$$\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \begin{bmatrix} 0 & -m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 \\ 0 & 0 \end{bmatrix}$$

The matrix $N(q, \dot{q})$ corresponds to

$$\begin{aligned} \boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}}) &= \dot{\boldsymbol{B}}(\boldsymbol{q}) - 2\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ &= \begin{bmatrix} 0 & -m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 \\ -m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 & 0 \end{bmatrix} - 2 \begin{bmatrix} 0 & -m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 \\ -m_{\ell_2} \ell_2 s_2 \dot{\theta}_2 & 0 \end{bmatrix} \end{aligned}$$

We can easily see now that the sum of the matrix with its transpose is equal to 0 and so the matrix is skew symmetric since and hence is true that $\boldsymbol{w}^T \boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \boldsymbol{w} = 0$. Of course this is a more general property which includes also the other property $\dot{\boldsymbol{q}}^T \boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \dot{\boldsymbol{q}} = 0$.

3 Trajectory Planning

The goal of trajectory planning is to generate the reference inputs to the motion control system which ensures that the manipulator executes the planned trajectories.

It is obvious that infinite solutions exist to this problem. Two possible choices are the *cubic polynomial*, which minimize the energy dissipation in the motor and the *trapezoidal velocity profile*.

In this case I have chosen the first method, since it gives us smooth reference signals and it can be computed easily, without the need of derivation and integrations of signals as in the second case, but just solving a systems of equations.

To determine the joint motion, the *cubic* polynomial

$$q(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0 \tag{10}$$

can be chosen, resulting into a *parabolic* velocity profile

$$\dot{q}(t) = 3a_3 t^2 + 2a_2 t + a_1 \tag{11}$$

and a *linear* acceleration profile

$$\ddot{q}(t) = 6a_3 t^3 + 2a_2 \tag{12}$$

Finally, as we previosly mentioned, the determination of the specific trajectory is given by the solution of the following systems of equations:

$$\begin{cases} q_i &= a_0 \\ \dot{q}_i &= a_1 \\ q_f &= a_3 t_f^3 + a_2 t_f^2 + a_1 t_f + a_0 \\ \dot{q}_f &= 3a_3 t_f^2 + 2a_2 t_f + a_1 \end{cases}$$

In this way I could find the desired reference trajectory for the second generalized coordinate. At this point I need to find the relation between the first and the second generalized coordinate exploiting the constraint of the moto so that the end effector respects the desired trajectory in the operational space.

The trajectory of the point p_e in the operational space can be parametrized as follow

$$p_e(\lambda) = p_{e_i} - \lambda(p_{e_i} - p_{e_f}) = \begin{bmatrix} a_2\lambda \\ 0 \\ a_1 + a_2(1-\lambda) \end{bmatrix}$$

where $\lambda \in [0,1]$. Indeed, we can easily see that when $\lambda = 0$ we are exactly in the configuration in Figure 1 of the homework test, while when $\lambda = 1$ we are in the configuration reported in Figure 2.

Now it is necessary to choose $d_1(\theta_2)$ so that the point p_e respects the desired trajectory in the operational space.

$$\begin{bmatrix} a_2\lambda \\ 0 \\ a_1 + a_2(1-\lambda) \end{bmatrix} = \begin{bmatrix} -c_2 a_2 \\ 0 \\ d_1 + s_2 a_2 \end{bmatrix} \Rightarrow \begin{cases} \lambda = -c_2 \\ a_1 + a_2(1-\lambda) = d_1 + s_2 a_2 \end{cases} \Rightarrow a_1 + a_2(1 + c_2) = d_1 + s_2 a_2$$

and so we get:

$$d_1(\theta_2) = a_1 + a_2(1 + \cos(\theta_2) - \sin(\theta_2)) \tag{13}$$

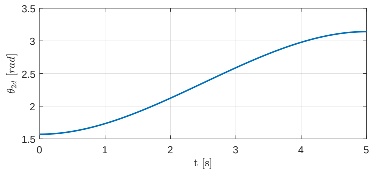
In this way we have found the relation between between the two generalized coordinates during the motion.

Now we have to compute the relation also for the first and second derivatives of the generalized coordinates, getting:

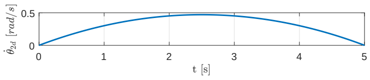
$$\dot{d}_1(\theta_2, \dot{\theta}_2) = \frac{\mathrm{d}}{\mathrm{d}t} d_1 = -a_2 \dot{\theta}_2 \left(\sin(\theta_2) + \cos(\theta_2)\right) \tag{14}$$

for the velocity and

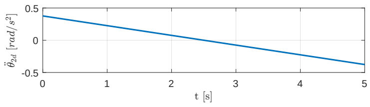
$$\ddot{d}_1(\theta_2, \dot{\theta}_2, \ddot{\theta}_2) = \frac{\mathrm{d}}{\mathrm{d}t} \dot{d}_1 = -a_2 \left(\left(\ddot{\theta}_2 - \dot{\theta}_2^2 \right) \sin(\theta_2) + \left(\ddot{\theta}_2 + \dot{\theta}_2^2 \right) \cos(\theta_2) \right) \tag{15}$$



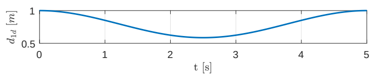
position ref q_2



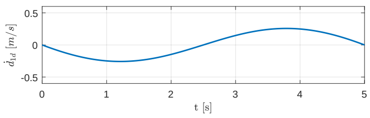
velocity ref $\dot{q}_2$



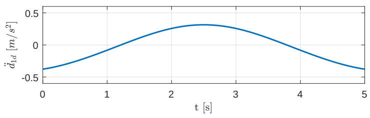
acceleration ref $\ddot{q}_2$



position ref q_1



velocity ref $\dot{q}_1$



acceleration ref $\ddot{q}_1$

for the acceleration. The final reference signals taken from the Simulink Simulation are shown in the figures above.

As we can see in both the previous figures, the references signals take constraints into account, in fact the initial and final velocity of the joints are zero, which implies that the final velocity in the operational space is zero.

4 Control

Since we would like to track a desired trajectory, defined in the joint space in terms of position, velocity and acceleration, I have chosen to use the **Inverse Dynamics Control**. It is a centralized control on the joint space, since we have found all the references with respect to the joint variables.

Moreover, since we have no interaction with the environment, we are not forced to use a controller defined in the operational space.

The scheme I used is reported in figure 2.

The idea is to do an *exact linearization* of the system dynamics, obtained by means of a *nonlinear*

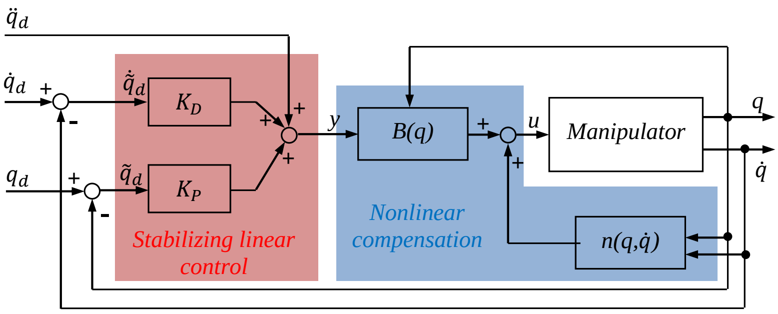


Figure 2: Inverse Dynamics Control scheme

state feedback and then finding a stabilizing control law. So the overall control input is

$$u = B(q)y + n(q, \dot{q})$$

where y is the output of the stabilizing controller:

$$y = \ddot{q}_d + K_P\tilde{q} + K_D\dot{\tilde{q}}$$

where $\tilde{q} = q_d - q$. In particular choosing K_P and K_D as diagonal matrices gives as a decoupled systems, meaning that motion of each actuator does not induce motion on a joint other than that actuated.

Other that, it is possible to prove that in order to ensure global asymptotic stability, the matrices K_P and K_D must be > 0 . Moreover, since we have no specifications on the rate of convergence, I just put them as identity matrices.

However, since we have supposed to have no disturbances, the trajectory given are enough to reach the goal, and our stabilizing controller is not really needed. The control components are put to obtain a more robust and precise control action in case of uncertainties in the model and in presence of disturbances.

So the input force and torque respectively of the two joints are: The final trajectory obtain

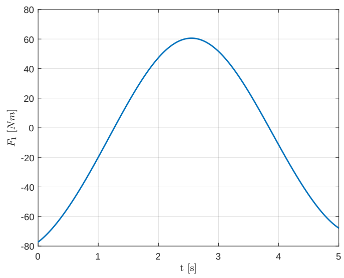


Figure 3: Joint 1 Force

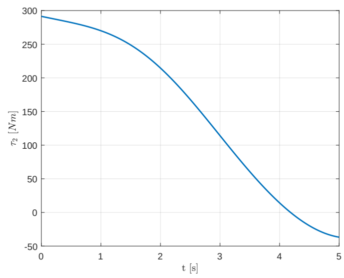


Figure 4: Joint 2 Torque

through simulation is the following:

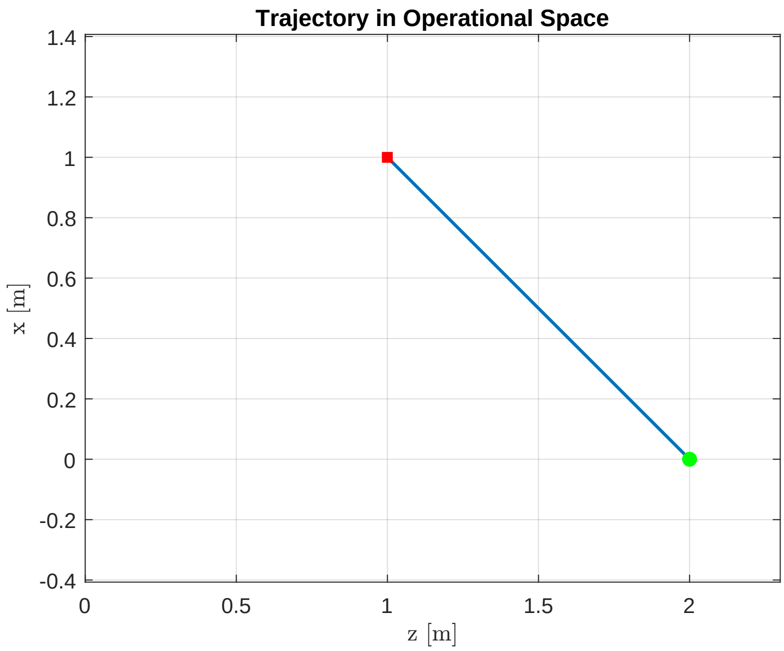


Figure 5: Trajectory in the Operational Space

Appendix

The Simulink scheme used for the simulations is the following:

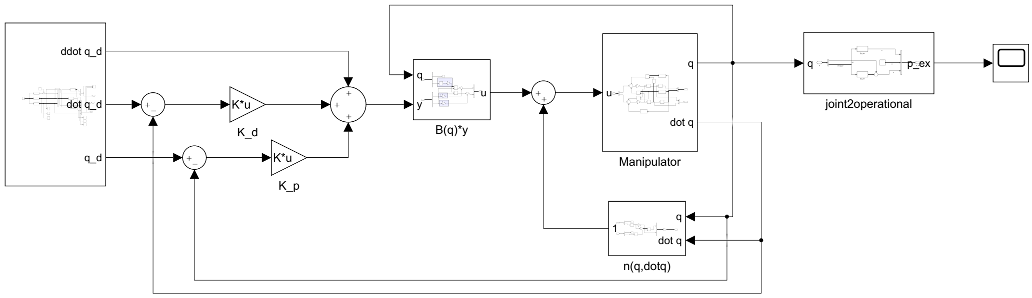


Figure 6: Simulink Model