Solution

January 11, 2022

Exercise #1

A possible assignment of Denavit-Hartenberg (D-H) frames is shown in Fig. 4. The associated D-H parameters are given in Table 1. The signs of the q_i's correspond to the robot configuration shown in the figure. Note that the L-shaped form of the forearm is equivalent from a kinematic point of view to a straight link of length $D = \sqrt{L^2 + M^2}$ connecting the origin O_2 of frame RF_2 with the point P, where the origin O_3 of the last D-H frame had to be placed. Accordingly, $\boldsymbol{x}_3$ is chosen along the direction of this equivalent link.

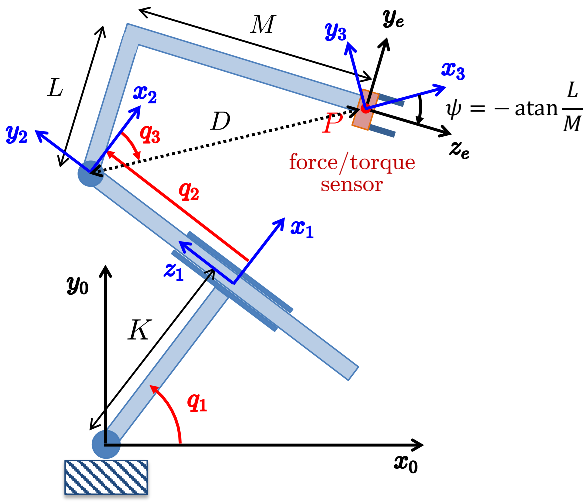

Figure 4: Assignment of D-H frames for the planar RPR robot.

i	α_i	a_i	d_i	θ_i
1	$-\pi/2$	K	0	$q_1 > 0$
2	$\pi/2$	0	$q_2 > 0$	0
3	0	$D = \sqrt{L^2 + M^2}$	0	$q_3 < 0$

Table 1: Table of D-H parameters for the planar RPR robot.

Figure 5 shows the robot in the configuration $\boldsymbol{q} = \boldsymbol{0}$. In this configuration, the position of the point $P = O_3$ and the orientation of the D-H frame RF_3, as computed from the direct kinematics using the D-H homogeneous transformation matrices ${}^{i-1}\boldsymbol{A}_i(q_i)$, are

$$\boldsymbol{p} = \begin{pmatrix} K + D \\ 0 \\ 0 \end{pmatrix}, \qquad {}^0\boldsymbol{R}_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

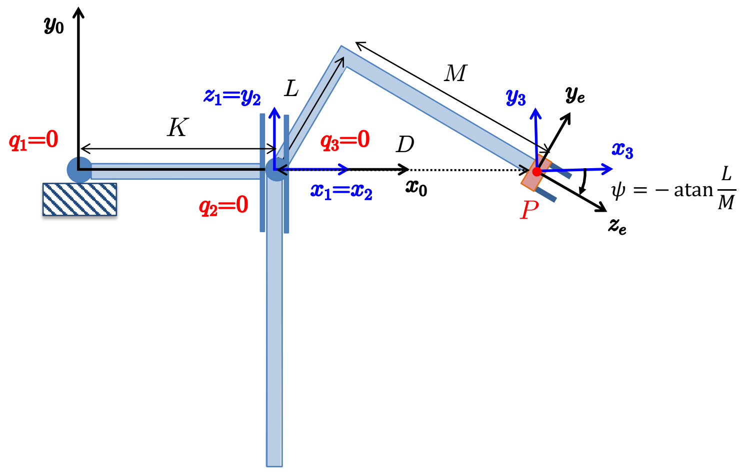

Figure 5: The RPR robot in the configuration $\boldsymbol{q} = \mathbf{0}$.

The constant homogeneous transformation matrix ${}^3\boldsymbol{T}_e$ that aligns the last D-H frame RF_3 with the end-effector (sensor) frame RF_e is given by

$$
{}^3\boldsymbol{T}_e = \begin{pmatrix} {}^3\boldsymbol{R}_e & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix}, \quad \text{with} \quad {}^3\boldsymbol{R}_e = \begin{pmatrix} 0 & L/D & M/D \\ 0 & M/D & -L/D \\ -1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\sin\psi & \cos\psi \\ 0 & \cos\psi & \sin\psi \\ -1 & 0 & 0 \end{pmatrix}, \quad (1)
$$

where $\psi = -\arctan\left(L/M\right) < 0$.
The XYX Euler angles $(\alpha_1, \alpha_2, \alpha_3)$ define the rotation matrix

$$
\begin{aligned}
\boldsymbol{R}_{XYX} &= \boldsymbol{R}_X(\alpha_1)\boldsymbol{R}_Y(\alpha_2)\boldsymbol{R}_X(\alpha_3) \\
&= \begin{pmatrix} \cos\alpha_2 & \sin\alpha_2 \sin\alpha_3 & \sin\alpha_2\cos\alpha_3 \\ \sin\alpha_1\sin\alpha_2 & \cos\alpha_1\cos\alpha_3 - \sin\alpha_1\cos\alpha_2\sin\alpha_3 & -\cos\alpha_1\sin\alpha_3 - \sin\alpha_1\cos\alpha_2\cos\alpha_3 \\ -\cos\alpha_1\sin\alpha_2 & \sin\alpha_1\cos\alpha_3 + \cos\alpha_1\cos\alpha_2\sin\alpha_3 & \cos\alpha_1\cos\alpha_2\cos\alpha_3 - \sin\alpha_1\sin\alpha_3 \end{pmatrix}.
\end{aligned}
$$

We need to solve the inverse orientation problem for this minimal representation of Euler angles:

$$
\boldsymbol{R}_{XYX}\left(\alpha_1, \alpha_2, \alpha_3\right) = {}^3\boldsymbol{R}_e(\psi).
$$

Since the two elements in the first and second row of the first column of ${}^3\boldsymbol{R}_e$ are not simultaneously zero, two *regular* solutions for $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$ are obtained in symbolic form as:

$$
\boldsymbol{\alpha}^+ = \begin{pmatrix} 0 \\ \frac{\pi}{2} \\ -\psi \end{pmatrix}, \qquad \boldsymbol{\alpha}^- = \begin{pmatrix} \pi \\ -\frac{\pi}{2} \\ -\psi + \pi \end{pmatrix}.
$$

With the numerical values $L = M = 1$ [m], we have $\psi = -45° = -0.7854$ [rad] and thus

$$
\boldsymbol{\alpha}^+ = \begin{pmatrix} 0 \\ 1.5708 \\ -0.7854 \end{pmatrix}, \qquad \boldsymbol{\alpha}^- = \begin{pmatrix} 3.1416 \\ -1.5708 \\ 2.3562 \end{pmatrix} \quad \text{[rad]}.
$$

Exercise #2

The requested task kinematics for the RPR robot in Fig. 1 is easily obtained as[1]

$$\boldsymbol{r} = \begin{pmatrix} p_x \\ p_y \\ \phi \end{pmatrix} = \begin{pmatrix} K\cos q_1 - q_2 \sin q_1 + D\cos(q_1+q_3) \\ K\sin q_1 + q_2\cos q_1 + D\sin(q_1+q_3) \\ q_1+q_3 \end{pmatrix} = \boldsymbol{t}(\boldsymbol{q}), \tag{2}$$

with $D = \sqrt{L^2+M^2}$. The closed-form expression of the inverse kinematics

$$\boldsymbol{q} = \boldsymbol{t}^{-1}(\boldsymbol{r}_d), \quad \text{for a given } \boldsymbol{r}_d = \begin{pmatrix} p_{xd} \\ p_{yd} \\ \phi_d \end{pmatrix}, \tag{3}$$

is found from (2) and (3) as follows. Substituting the third relation $q_1 + q_3 = \phi_d$ in the first two leads to

$$\begin{aligned} K\cos q_1 - q_2\sin q_1 &= p_{xd} - D\cos\phi_d \\ K\sin q_1 + q_2\cos q_1 &= p_{yd} - D\sin\phi_d. \end{aligned} \tag{4}$$

Squaring each equation in (4) and summing, we obtain after simplifications

$$q_2^2 = p_{xd}^2 + p_{yd}^2 + D^2 - 2D\left(p_{xd}\cos\phi_d + p_{yd}\sin\phi_d\right) - K^2 \triangleq A \qquad \Rightarrow \qquad q_2^{\pm} = \pm\sqrt{A}. \tag{5}$$

When $A > 0$, we get two (real) solutions for q_2. If $A = 0$, the two solutions collapse into the single value $q_2 = 0$ (singular case). When $A < 0$,, the inverse problem has no solution because the input data are not compatible with the *secondary* workspace of the robot. When a solution exists (either two or only one), substituting in (4) each value of q_2 from (5), we obtain a linear system of two equations in the two unknowns $s_1 = \sin q_1$ and $c_1 = \cos q_1$:

$$\begin{pmatrix} K & -q_2^{\pm} \\ q_2^{\pm} & K \end{pmatrix} \begin{pmatrix} c_1 \\ s_1 \end{pmatrix} = \begin{pmatrix} p_{xd} - D\cos\phi_d \\ p_{yd} - D\sin\phi_d \end{pmatrix}. \tag{6}$$

The determinant of the coefficient matrix is $\det = K^2 + |A| > 0$ (in the assumed situation). Solving (6) provides a value for each $q_2^{\pm}$

$$\begin{aligned} q_1^{\pm} &= \text{atan2}\{s_1, c_1\} \\ &= \text{atan2}\left\{-q_2^{\pm}\left(p_{x_d} - D\cos\phi_d\right) + K\left(p_{yd} - D\sin\phi_d\right), K\left(p_{xd} - D\cos\phi_d\right) + q_2^{\pm}\left(p_{yd} - D\sin\phi_d\right)\right\}, \end{aligned} \tag{7}$$

and finally

$$q_3^{\pm} = \phi_d - q_1^{\pm}. \tag{8}$$

For the following data

$$K = L = M = 1 \text{ [m]} \quad \Rightarrow \quad D = \sqrt{2} \text{ [m]} \qquad \text{and} \qquad \boldsymbol{r}_d = \begin{pmatrix} p_{xd} \\ p_{yd} \\ \phi_d \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \\ -\pi/6 \end{pmatrix} \text{ [m,m,rad]},$$

we have that $A = 2.5152$ (regular case) and the two inverse solutions are

$$\boldsymbol{q}^+ = \begin{pmatrix} 7.81^\circ \\ 1.5859 \\ -37.81^\circ \end{pmatrix} = \begin{pmatrix} 0.1363 \\ 1.5859 \\ -0.6599 \end{pmatrix} \text{ [rad,m,rad]} \quad \boldsymbol{q}^- = \begin{pmatrix} 123.34^\circ \\ -1.5859 \\ -153.34^\circ \end{pmatrix} = \begin{pmatrix} 2.1527 \\ -1.5859 \\ -2.6763 \end{pmatrix} \text{ [rad,m,rad]}.$$

[1] Extract the expressions in (2) from ${}^0\boldsymbol{T}_3(\boldsymbol{q}) = {}^0\boldsymbol{A}_1(q_1)\,{}^1\boldsymbol{A}_2(q_2)\,{}^2\boldsymbol{A}_3(q_3)$ or just use direct inspection of the figure.

Exercise #3

The (3×3) Jacobian matrix associated to the task (2) is

$$\boldsymbol{J}_t(\boldsymbol{q}) = \frac{\partial \boldsymbol{t}(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{pmatrix} -K\sin q_1 - q_2\cos q_1 - D\sin(q_1+q_3) & -\sin q_1 & -D\sin(q_1+q_3) \\ K\cos q_1 - q_2\sin q_1 + D\cos(q_1+q_3) & \cos q_1 & D\cos(q_1+q_3) \\ 1 & 0 & 1 \end{pmatrix}. \tag{9}$$

Its determinant is $\det \boldsymbol{J}_t(\boldsymbol{q}) = -q_2$. When the robot is in a task singularity $\boldsymbol{q}_s = (q_1, 0, q_3)$, with q_1 and q_3 being arbitrary, the Jacobian becomes

$$\boldsymbol{J}_s = \boldsymbol{J}_t(\boldsymbol{q}_s) = \begin{pmatrix} -K\sin q_1 - D\sin(q_1+q_3) & -\sin q_1 & -D\sin(q_1+q_3) \\ K\cos q_1 + D\cos(q_1+q_3) & \cos q_1 & D\cos(q_1+q_3) \\ 1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{J}_1 & \boldsymbol{J}_2 & \boldsymbol{J}_3 \end{pmatrix}. \tag{10}$$

It is evident that its first column $\boldsymbol{J}_1$ is a linear combination of the other two: $\boldsymbol{J}_1 = K\boldsymbol{J}_2 + \boldsymbol{J}_3$. Moreover, $\text{rank}\{\boldsymbol{J}_s\} = \text{rank}\left\{\begin{pmatrix} \boldsymbol{J}_2 & \boldsymbol{J}_3 \end{pmatrix}\right\} = 2$, constant for all (q_1, q_3). Therefore, the requested subspaces $\mathcal{N}\{\boldsymbol{J}_s\}$ and $\mathcal{R}\{\boldsymbol{J}_s\}$ associated to the singular matrix $\boldsymbol{J}_s$ are one-dimensional and, respectively, two-dimensional, with *global* bases given by

$$\mathcal{N}\{\boldsymbol{J}_s\} = \text{span}\left\{\begin{pmatrix} -1 \\ K \\ 1 \end{pmatrix}\right\}, \qquad \mathcal{R}\{\boldsymbol{J}_s\} = \text{span}\left\{\begin{pmatrix} -\sin q_1 \\ \cos q_1 \\ 0 \end{pmatrix}, \begin{pmatrix} -\sin(q_1+q_3) \\ \cos(q_1+q_3) \\ 1 \end{pmatrix}\right\}.$$

Set now $K = L = M = 1$ (and thus $D = \sqrt{2}$) in (10). A simple choice of a feasible task velocity is

$$\dot{\boldsymbol{r}}_f = \gamma \begin{pmatrix} -\sin q_1 \\ \cos q_1 \\ 0 \end{pmatrix} \in \mathcal{R}\{\boldsymbol{J}_s\}.$$

There is indeed an infinity of joint velocities $\dot{\boldsymbol{q}}_f \in \mathbb{R}^3$ realizing $\dot{\boldsymbol{r}}_f$. Two possible solutions are

$$\dot{\boldsymbol{q}}_f' = \begin{pmatrix} 0 \\ \gamma \\ 0 \end{pmatrix} \qquad \text{or} \qquad \dot{\boldsymbol{q}}_f'' = \begin{pmatrix} \gamma \\ 0 \\ -\gamma \end{pmatrix} \qquad \Rightarrow \qquad \boldsymbol{J}_s\,\dot{\boldsymbol{q}}_f' = \boldsymbol{J}_s\,\dot{\boldsymbol{q}}_f'' = \dot{\boldsymbol{r}}_f.$$

The first solution uses only the prismatic joint of the robot, while the second uses only the two revolute joints. Note that it is useless to ask which solution has a smaller norm, i.e., it involves a smaller motion in the joint space (apparently, $\left\|\dot{\boldsymbol{q}}_f'\right\| < \left\|\dot{\boldsymbol{q}}_f''\right\|$). In fact, the first solution has [m] as units while the other uses [rad]. These units are not commensurable[2], and the straightforward norm minimization would not be unit-independent. The problem arises because of the different nature (prismatic and revolute) of the robot joints. For this reason, the pseudoinverse solution

$$\dot{\boldsymbol{q}}_f^{\#} = \boldsymbol{J}_s^{\#}\,\dot{\boldsymbol{r}}_f = \frac{\gamma}{K^2+1}\begin{pmatrix} K \\ 1 \\ -K \end{pmatrix} \qquad \text{[rad/s,m/s,rad/s]}$$

makes little sense here.

[2]This is a typical 'adding apples and oranges' issue: which is larger, 1 radiant or 1 meter? 1 radiant or 100 centimeters?

Exercise #4

A main issue here is the expression of forces/torques from one reference frame to another: in particular, from the sensor frame RF_e at the robot gripper, where measures of the wrench ${}^e\boldsymbol{F}$ (i.e., forces $\boldsymbol{f} \in \mathbb{R}^3$ and torques $\boldsymbol{m} \in \mathbb{R}^3$) are collected, to the absolute frame RF_0. Because of the set up of the axes of these two reference frames, the problem is naturally embedded in 3D. Moreover, this change of representation is needed also when using the (transpose of the) *geometric* Jacobian $\boldsymbol{J}(\boldsymbol{q})$ for computing the joint torques $\boldsymbol{\tau} \in \mathbb{R}^n$ associated to a wrench at the end-effector gripper. In fact, with the $(6 \times n)$ geometric Jacobian $\boldsymbol{J}(\boldsymbol{q})$ we usually express the end-effector linear and angular velocities $\boldsymbol{v} \in \mathbb{R}^3$ and $\boldsymbol{\omega} \in \mathbb{R}^3$ directly in frame RF_0. The dual map requires then also wrenches to be expressed in the same frame. In the following, quantities expressed in RF_e carry a preceding superscript e, whereas quantities without a preceding superscript are expressed (by default) in the absolute frame RF_0.

With the above in mind, we have in general

$$
\begin{aligned}
\begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{pmatrix} &= \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} = \begin{pmatrix} \boldsymbol{J}_L(\boldsymbol{q}) \\ \boldsymbol{J}_A(\boldsymbol{q}) \end{pmatrix} \dot{\boldsymbol{q}} \\
\Rightarrow \begin{pmatrix} {}^e\boldsymbol{v} \\ {}^e\boldsymbol{\omega} \end{pmatrix} &= \begin{pmatrix} {}^e\boldsymbol{R}_0(\boldsymbol{q})\,\boldsymbol{v} \\ {}^e\boldsymbol{R}_0(\boldsymbol{q})\,\boldsymbol{\omega} \end{pmatrix} = \begin{pmatrix} {}^e\boldsymbol{R}_0(\boldsymbol{q}) & \boldsymbol{0} \\ \boldsymbol{0} & {}^e\boldsymbol{R}_0(\boldsymbol{q}) \end{pmatrix} \begin{pmatrix} \boldsymbol{J}_L(\boldsymbol{q}) \\ \boldsymbol{J}_A(\boldsymbol{q}) \end{pmatrix} \dot{\boldsymbol{q}} = \begin{pmatrix} {}^e\boldsymbol{J}_L(\boldsymbol{q}) \\ {}^e\boldsymbol{J}_A(\boldsymbol{q}) \end{pmatrix} \dot{\boldsymbol{q}}.
\end{aligned}
\tag{11}
$$

On the other hand

$$
\boldsymbol{F} = \begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{m} \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{R}_e(\boldsymbol{q})\,{}^e\boldsymbol{f} \\ {}^0\boldsymbol{R}_e(\boldsymbol{q})\,{}^e\boldsymbol{m} \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{R}_e(\boldsymbol{q}) & \boldsymbol{0} \\ \boldsymbol{0} & {}^0\boldsymbol{R}_e(\boldsymbol{q}) \end{pmatrix} \begin{pmatrix} {}^e\boldsymbol{f} \\ {}^e\boldsymbol{m} \end{pmatrix}. \tag{12}
$$

Thus, the map from end-effector wrenches to joint torques can be written in equivalent ways as

$$
\begin{aligned}
\boldsymbol{\tau} = \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F} &= \begin{pmatrix} \boldsymbol{J}_L^T(\boldsymbol{q}) & \boldsymbol{J}_A^T(\boldsymbol{q}) \end{pmatrix} \begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{m} \end{pmatrix} \\
&= \begin{pmatrix} \boldsymbol{J}_L^T(\boldsymbol{q}) & \boldsymbol{J}_A^T(\boldsymbol{q}) \end{pmatrix} \begin{pmatrix} {}^e\boldsymbol{R}_0(\boldsymbol{q}) & \boldsymbol{0} \\ \boldsymbol{0} & {}^e\boldsymbol{R}_0(\boldsymbol{q}) \end{pmatrix}^T \begin{pmatrix} {}^e\boldsymbol{R}_0(\boldsymbol{q}) & \boldsymbol{0} \\ \boldsymbol{0} & {}^e\boldsymbol{R}_0(\boldsymbol{q}) \end{pmatrix} \begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{m} \end{pmatrix} \\
&= \left(\begin{pmatrix} {}^e\boldsymbol{R}_0(\boldsymbol{q}) & \boldsymbol{0} \\ \boldsymbol{0} & {}^e\boldsymbol{R}_0(\boldsymbol{q}) \end{pmatrix} \begin{pmatrix} \boldsymbol{J}_L(\boldsymbol{q}) \\ \boldsymbol{J}_A(\boldsymbol{q}) \end{pmatrix} \right)^T \begin{pmatrix} {}^e\boldsymbol{f} \\ {}^e\boldsymbol{m} \end{pmatrix} \\
&= \begin{pmatrix} {}^e\boldsymbol{J}_L^T(\boldsymbol{q}) & {}^e\boldsymbol{J}_A^T(\boldsymbol{q}) \end{pmatrix} \begin{pmatrix} {}^e\boldsymbol{f} \\ {}^e\boldsymbol{m} \end{pmatrix} \\
&= {}^e\boldsymbol{J}^T(\boldsymbol{q})\,{}^e\boldsymbol{F}.
\end{aligned}
\tag{13}
$$

In the above computations, one needs ${}^0\boldsymbol{R}_3(\boldsymbol{q})$ from the robot direct kinematics and ${}^3\boldsymbol{R}_e$ from (1). Further, we can use conveniently the task Jacobian $\boldsymbol{J}_t(\boldsymbol{q})$ in (9) to build the geometric Jacobian $\boldsymbol{J}(\boldsymbol{q})$. These quantities are evaluated when the robot is in $\boldsymbol{q} = \bar{\boldsymbol{q}} = (\pi/2, -1, 0)$ [rad,m,rad], using the data $K = L = M = 1$ [m] (thus, $D = \sqrt{2}$ [m] and $\psi = -\pi/4$ [rad]). We have

$$
\begin{aligned}
{}^0\boldsymbol{R}_e(\bar{\boldsymbol{q}}) = {}^0\boldsymbol{R}_3(\bar{\boldsymbol{q}})\,{}^3\boldsymbol{R}_e &= \begin{pmatrix} \cos(\bar{q}_1 + \bar{q}_3) & -\sin(\bar{q}_1 + \bar{q}_3) & 0 \\ \sin(\bar{q}_1 + \bar{q}_3) & \cos(\bar{q}_1 + \bar{q}_3) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & -\sin\psi & \cos\psi \\ 0 & \cos\psi & \sin\psi \\ -1 & 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & \sqrt{2}/2 & \sqrt{2}/2 \\ 0 & \sqrt{2}/2 & -\sqrt{2}/2 \\ -1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -0.7071 & 0.7071 \\ 0 & 0.7071 & 0.7071 \\ -1 & 0 & 0 \end{pmatrix}
\end{aligned}
\tag{14}
$$

and

$$\boldsymbol{J}_t(\bar{\boldsymbol{q}}) = \begin{pmatrix} -\sin\bar{q}_1 - \bar{q}_2\cos\bar{q}_1 - \sqrt{2}\,\sin(\bar{q}_1+\bar{q}_3) & -\sin\bar{q}_1 & -\sqrt{2}\,\sin(\bar{q}_1+\bar{q}_3) \\ \cos\bar{q}_1 - \bar{q}_2\sin\bar{q}_1 + \sqrt{2}\,\cos(\bar{q}_1+\bar{q}_3) & \cos\bar{q}_1 & \sqrt{2}\,\cos(\bar{q}_1+\bar{q}_3) \\ 1 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} -1-\sqrt{2} & -1 & -\sqrt{2} \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} = \begin{array}{c} \begin{pmatrix} -2.4142 & -1 & -1.4142 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} \end{array} \begin{array}{l} \leftarrow\ \dot{p}_x \\ \leftarrow\ \dot{p}_y \\ \leftarrow\ \dot{\phi}_z \end{array}$$

Therefore, the mapping

$$\dot{\boldsymbol{q}} \in \mathbb{R}^3 \quad \longrightarrow \quad \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{pmatrix} = \begin{pmatrix} v_x \\ v_y \\ v_z \\ \omega_x \\ \omega_y \\ \omega_z \end{pmatrix}$$

is given by the (6×3) geometric Jacobian, expressed in the frames RF_0 and RF_e respectively as[3]

$$\boldsymbol{J}(\bar{\boldsymbol{q}}) = \begin{pmatrix} \boldsymbol{J}_L(\bar{\boldsymbol{q}}) \\ \boldsymbol{J}_A(\bar{\boldsymbol{q}}) \end{pmatrix} = \begin{pmatrix} -2.4142 & -1 & -1.4142 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{array}{l} \leftarrow\ v_x = \dot{p}_x \\ \leftarrow\ v_y = \dot{p}_y \\ \leftarrow\ v_z = 0 \\ \leftarrow\ \omega_x = 0 \\ \leftarrow\ \omega_y = 0 \\ \leftarrow\ \omega_z = \dot{\phi}_z \end{array}$$

and, using the transpose of (14),

$${}^e\boldsymbol{J}(\bar{\boldsymbol{q}}) = \begin{pmatrix} {}^e\boldsymbol{J}_L(\bar{\boldsymbol{q}}) \\ {}^e\boldsymbol{J}_A(\bar{\boldsymbol{q}}) \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{R}_e^T(\bar{\boldsymbol{q}})\,\boldsymbol{J}_L(\bar{\boldsymbol{q}}) \\ {}^0\boldsymbol{R}_e^T(\bar{\boldsymbol{q}})\,\boldsymbol{J}_A(\bar{\boldsymbol{q}}) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 2.4142 & 0.7071 & 1 \\ -1 & -0.7071 & -1 \\ -1 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{array}{l} \leftarrow\ {}^e v_x = 0 \\ \leftarrow\ {}^e v_y \\ \leftarrow\ {}^e v_z \\ \leftarrow\ {}^e\omega_x = -\dot{\phi}_z \\ \leftarrow\ {}^e\omega_y = 0 \\ \leftarrow\ {}^z\omega_y = 0. \end{array}$$

The two posed problems i. and ii. have then the following answers. From the measured data

$${}^e\boldsymbol{F} = \begin{pmatrix} {}^e\boldsymbol{f}^T & {}^e\boldsymbol{m}^T \end{pmatrix}^T = \begin{pmatrix} {}^e f_x & {}^e f_y & {}^e f_z & {}^e m_x & {}^e m_y & {}^e m_z \end{pmatrix}^T = \begin{pmatrix} 0 & -1 & -2 & 2 & 0 & 0 \end{pmatrix}^T,$$

we compute the gripper wrench expressed in the base frame using (12) and (14):

$$\boldsymbol{F} = \begin{pmatrix} \boldsymbol{f} \\ \boldsymbol{m} \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{R}_e(\bar{\boldsymbol{q}})\,{}^e\boldsymbol{f} \\ {}^0\boldsymbol{R}_e(\bar{\boldsymbol{q}})\,{}^e\boldsymbol{m} \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{R}_e(\bar{\boldsymbol{q}}) & \boldsymbol{0} \\ \boldsymbol{0} & {}^0\boldsymbol{R}_e(\bar{\boldsymbol{q}}) \end{pmatrix} {}^e\boldsymbol{F} = \begin{pmatrix} -0.7071 \\ -2.1213 \\ 0 \\ 0 \\ 0 \\ -2 \end{pmatrix} \begin{array}{l} \leftarrow\ f_x\ [\mathrm{N}] \\ \leftarrow\ f_y\ [\mathrm{N}] \\ \\ \\ \\ \leftarrow\ m_z\ [\mathrm{Nm}]. \end{array}$$

[3] We simply embed here the rows of $\boldsymbol{J}_t(\bar{\boldsymbol{q}})$ in the correct rows of $\boldsymbol{J}(\bar{\boldsymbol{q}})$.

From (13), the joint torques needed to balance in static conditions the gripper wrench (applied by the environment and measured by the sensor) are given by

$$\boldsymbol{\tau} = -\,{}^{e}\boldsymbol{J}^{T}(\bar{\boldsymbol{q}})\,{}^{e}\boldsymbol{F} = -\boldsymbol{J}^{T}(\bar{\boldsymbol{q}})\,\boldsymbol{F} = \begin{pmatrix} 2.4142 \\ -0.7071 \\ 1 \end{pmatrix} \text{ [Nm,N,Nm]}.$$

Note here the minus sign!

Exercise #5

The elliptic path in Fig. 2 can be smoothly parametrized by

$$\boldsymbol{p}_d(s) = \begin{pmatrix} p_{dx}(s) \\ p_{dy}(s) \end{pmatrix} = \begin{pmatrix} -a\sin 2\pi s \\ b\cos 2\pi s \end{pmatrix}, \qquad s \in [0,1].$$

In this way we have $\boldsymbol{p}_d(0) = (0, b)$, the coordinates of the point P_0, and the path is traced counterclockwise for increasing values of the parameter s. The first and second path derivatives are

$$\boldsymbol{p}_d'(s) = \frac{d\boldsymbol{p}_d(s)}{ds} = -2\pi \begin{pmatrix} a\cos 2\pi s \\ b\sin 2\pi s \end{pmatrix}, \qquad \boldsymbol{p}_d''(s) = \frac{d^2\boldsymbol{p}_d(s)}{ds^2} = 4\pi^2 \begin{pmatrix} a\sin 2\pi s \\ -b\cos 2\pi s \end{pmatrix}, \qquad s \in [0,1].$$

Figure 6 shows the plots of the x and y components of $\boldsymbol{p}_d(s)$, $\boldsymbol{p}_d'(s)$, and $\boldsymbol{p}_d''(s)$, when choosing $a = 1$ and $b = 0.3$ [m] as lengths for the semi-axes of the ellipse.

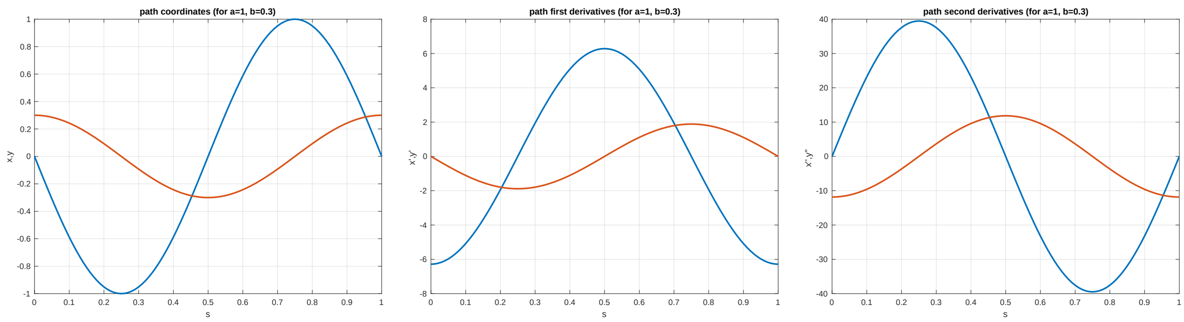

Figure 6: Components of $\boldsymbol{p}_d(s)$, $\boldsymbol{p}_d'(s)$, and $\boldsymbol{p}_d''(s)$ (x in blue, y in red).

The desired timing on the path is simply

$$s = s(t) = vt, \qquad \dot{s}(t) = v > 0, \qquad \ddot{s}(t) = 0, \qquad t \in [0, T],$$

where $T = 1/v$ is the motion time needed to trace a full ellipse with constant speed v. Accordingly, the velocity and the acceleration along the trajectory $\boldsymbol{p}_d(t)$ are

$$\dot{\boldsymbol{p}}_d(t) = \boldsymbol{p}_d'\,\dot{s} = -2\pi v \begin{pmatrix} a\cos 2\pi vt \\ b\sin 2\pi vt \end{pmatrix} \qquad \ddot{\boldsymbol{p}}_d(t) = \boldsymbol{p}_d'\,\ddot{s} + \boldsymbol{p}_d''\,\dot{s}^2 = 4\pi^2 v^2 \begin{pmatrix} a\sin 2\pi vt \\ -b\cos 2\pi vt \end{pmatrix},$$

with associated norms

$$\|\dot{\boldsymbol{p}}_d(t)\| = 2\pi v\,\sqrt{a^2\cos^2 2\pi vt + b^2\sin^2 2\pi vt}, \qquad \|\ddot{\boldsymbol{p}}_d(t)\| = 4\pi^2 v^2\,\sqrt{a^2\sin^2 2\pi vt + b^2\cos^2 2\pi vt}.$$

It is easy to see that, being $a > b$, the maximum values of these norms are

$$\max_{t\in[0,T]} \|\dot{\boldsymbol{p}}_d(t)\| = 2\pi v a, \qquad \text{attained at } t = \{0,\, T/2,\, T\}$$

and, respectively,

$$\max_{t\in[0,T]} \|\ddot{\boldsymbol{p}}_d(t)\| = 4\pi^2 v^2 a, \qquad \text{attained at } t = \{T/4,\, 3T/4\}.$$

From the required bounds on these norms

$$2\pi v a \leq V_{max}, \qquad 4\pi^2 v^2 a \leq A_{max},$$

we obtain the maximum feasible speed v_f for this motion as

$$v_f = \min\left\{\frac{V_{max}}{2\pi a}, \sqrt{\frac{A_{max}}{4\pi^2 a}}\right\} = \frac{1}{2\pi}\min\left\{\frac{V_{max}}{a}, \sqrt{\frac{A_{max}}{a}}\right\}.$$

Using the given numerical data $a = 1$, $b = 0.3$ [m], $V_{max} = 3$ [m/s] and $A_{max} = 6$ [m/s^2], we obtain for the speed and the motion time

$$v_f = 0.3898 \text{ [s}^{-1}] \qquad \Rightarrow \qquad T_f = 2.5651 \text{ [s]}.$$

The norm of the acceleration saturates the value $A_{max} = 6$ [m/s^2] while the maximum norm of the velocity equals 2.4495 [m/s], remaining below the limit V_{max}. Figure 7 shows the resulting evolution of the norms of $\dot{\boldsymbol{p}}_d(t)$ and $\ddot{\boldsymbol{p}}_d(t)$ along the trajectory.

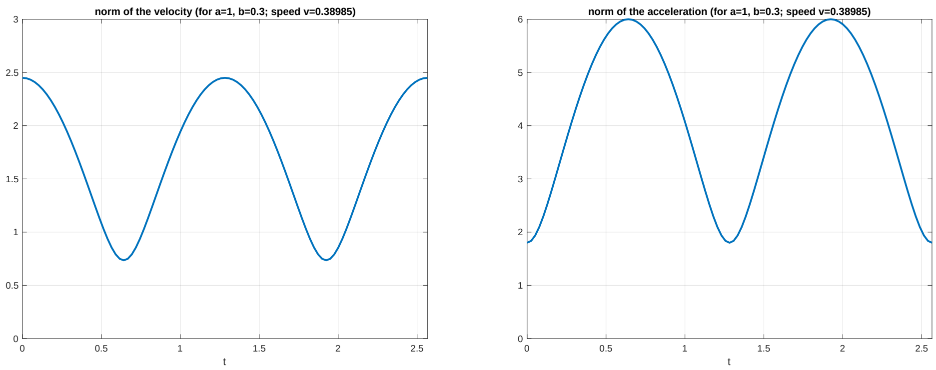

Figure 7: Evolution of $\|\dot{\boldsymbol{p}}_d(t)\|$ and $\|\ddot{\boldsymbol{p}}_d(t)\|$ for $v_f = 0.3898$ [s^{-1}].

Exercise #6

For the planar 2R robot shown in Fig. 3 to be able to execute the task, the elliptic path defined in Exercise #5 should entirely belong to its primary workspace. Since the robot has strictly different link lengths $l_1 = a$ and $l_2 = b < a$, the workspace is a circular annulus with internal radius $\rho_{min} = a - b > 0$ and external radius $\rho_{min} = a + b > 0$. Therefore, the lengths a and b of the semi-axes of the ellipse should satisfy the inequalities

$$\rho_{min} = a - b \leq a \leq a + b = \rho_{max}, \quad \rho_{min} = a - b \leq b \leq a + b = \rho_{max} \qquad \Rightarrow \qquad b < a \leq 2b.$$

However, the limit value $a = 2b$ would certainly lead to a singularity when the robot end effector is placed at $P_0 = (0, b)$, i.e., at the trajectory start (on the inner boundary of the workspace). In this case, the only inverse kinematics solution is $\boldsymbol{q}_s = (\pi/2, \pi)$, a singular configuration with the second link folded on the first one. The Jacobian of the 2R robot is then

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{pmatrix} -a\cos q_1 - b\cos(q_1+q_2) & -b\cos(q_1+q_2) \\ a\sin q_1 + b\sin(q_1+q_2) & b\sin(q_1+q_2) \end{pmatrix} \quad \Rightarrow \quad \boldsymbol{J}(\boldsymbol{q}_s) = \begin{pmatrix} 0 & 0 \\ a-b & -b \end{pmatrix}$$

so that $\det \boldsymbol{J}(\boldsymbol{q}_s) = 0$. The same happens also at the opposite point of the ellipse, $P_{-0} = (-b, 0)$. Therefore, a has to belong to the *open* interval $a \in (b, 2b)$ in order to avoid singularities[4]. For illustration, we choose the numerical values $a = 1$ and $b = 0.6$. The following results will be qualitatively similar for other admissible choices of these two geometric parameters. The position and the velocity of the desired Cartesian trajectory for $v = 0.4$ [s^{-1}] are shown in Fig. 8. The motion time is $T = 1/v = 2.5$ [s].

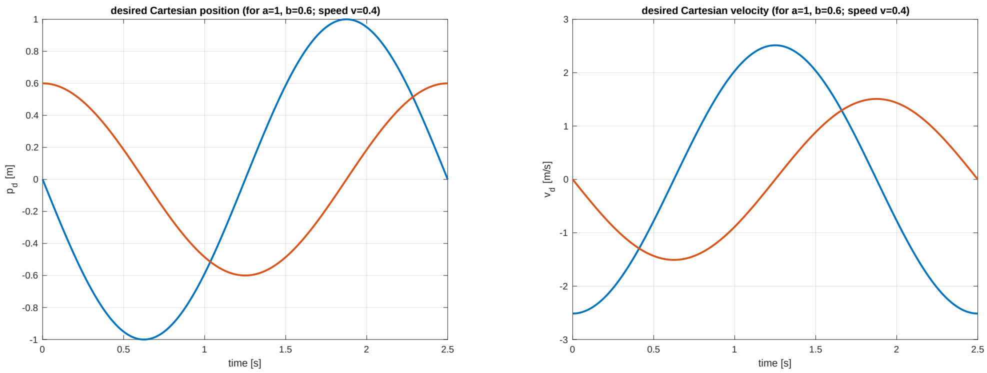

Figure 8: Components of $\boldsymbol{p}_d(t)$ and $\dot{\boldsymbol{p}}_d(t)$ (x in blue, y in red).

To proceed, we solve first the inverse kinematics for $P_0 = (p_{0x}, p_{0y}) = (0, b) = (0, 0.6)$, yielding two initial (regular) configurations. From the known formulas for the 2R robot, we have

$$c_2 = \frac{p_{0x}^2 + p_{0y}^2 - (a^2 + b^2)}{2ab} = -0.8333, \qquad s_2 = \sqrt{1 - c_2^2} = 0.5528,$$

to be used in

$$\begin{aligned} \boldsymbol{q}_0^+ &= \begin{pmatrix} \operatorname{atan2}\{p_{0y}(a + bc_2) - p_{0x} b\, s_2, p_{0x}(a + bc_2) + p_{0y} b\, s_2\} \\ \operatorname{atan2}\{s_2, c_2\} \end{pmatrix} \\ &= \begin{pmatrix} 56.44° \\ 146.44° \end{pmatrix} = \begin{pmatrix} 0.9851 \\ 2.5559 \end{pmatrix} \text{[rad]} \qquad \text{(right arm solution)} \end{aligned}$$

and

$$\begin{aligned} \boldsymbol{q}_0^- &= \begin{pmatrix} \operatorname{atan2}\{p_{0y}(a + bc_2) + p_{0x} b\, s_2, p_{0x}(a + bc_2) - p_{0y} b\, s_2\} \\ \operatorname{atan2}\{-s_2, c_2\} \end{pmatrix} \\ &= \begin{pmatrix} 123.56° \\ -146.44° \end{pmatrix} = \begin{pmatrix} 2.1565 \\ -2.5559 \end{pmatrix} \text{[rad]} \qquad \text{(left arm solution).} \end{aligned}$$

[4] It should be noted that the velocity vector $\dot{\boldsymbol{p}}_d$ is actually feasible even in the two singular situations and can be obtained by the use of the pseudoinverse of $\boldsymbol{J}$. However, too large joint velocities would be generated in that case while approaching a singularity.

With both choices, the position of the robot end effector will be matched with the desired trajectory $\boldsymbol{p}_d(t)$ at time $t = 0$ ($\boldsymbol{p}_d(0) = P_0$). With such an initialization, the nominal joint velocity command $\dot{\boldsymbol{q}}_n(t)$ that will execute perfectly the entire trajectory $\boldsymbol{p}_d(t)$, for $t \in [0, T]$, is given by

$$\dot{\boldsymbol{q}}_n = \boldsymbol{J}^{-1}(\boldsymbol{q}_n)\dot{\boldsymbol{p}}_d, \qquad \boldsymbol{q}_n(0) = \boldsymbol{q}_0^{\pm}. \tag{15}$$

Note that the robot Jacobian $\boldsymbol{J}(\boldsymbol{q}_n(t))$ will never become singular because the end-effector path remains always *strictly* inside the robot workspace. Therefore, the right or left arm configuration chosen at the start will be kept throughout the entire trajectory. Choosing, e.g., the right arm solution at start, $\boldsymbol{q}_n(0) = \boldsymbol{q}_0^{+}$, yields the joint velocity command $\dot{\boldsymbol{q}}_n(t)$ and the associated joint evolution $\boldsymbol{q}_n(t)$ shown in Fig. 9. It is apparent that the velocity commands and the motion of the joints are cyclic (modulo 2π for $q_{n1}(t)$). The initial value of the joint velocity command (15) is $\dot{\boldsymbol{q}}_n(0) = \begin{pmatrix} 4.1888 & 0 \end{pmatrix}^T$ [rad/s].

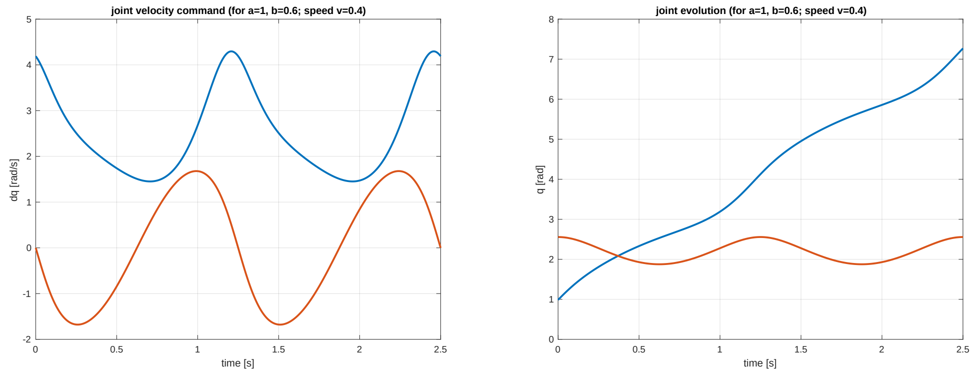

Figure 9: Nominal velocity command $\dot{\boldsymbol{q}}_n(t)$ and resulting evolution $\boldsymbol{q}_n(t)$ (joint 1 in blue, 2 in red).

Next, let the robot start from another initial configuration $\boldsymbol{q}(0)$ such that the end-effector position error is $\boldsymbol{e}(0) = \boldsymbol{p}_d(0) - \boldsymbol{f}(\boldsymbol{q}(0)) \neq \boldsymbol{0}$, with $\boldsymbol{p} = \boldsymbol{f}(\boldsymbol{q})$ being the direct kinematics of the 2R robot, but $e_y(0) = 0$. For instance, we choose $\boldsymbol{q}(0) = (0, \pi/2)$ (still a 'right arm' configuration), corresponding to an error $e_x(0) = -a = -1$ [m] only along the x-direction. In order to obtain asymptotic tracking of the desired trajectory $\boldsymbol{p}_d(t)$ together with the requested performance during the initial transient, the joint velocity control law $\dot{\boldsymbol{q}} = \dot{\boldsymbol{q}}_c(\boldsymbol{q}, t)$ is designed using feedback from the *Cartesian* error. The control law is then

$$\dot{\boldsymbol{q}}_c = \boldsymbol{J}^{-1}(\boldsymbol{q})\big(\dot{\boldsymbol{p}}_d + \boldsymbol{K}_P\left(\boldsymbol{p}_d - \boldsymbol{f}(\boldsymbol{q})\right)\big), \qquad \boldsymbol{K}_P = r \cdot \boldsymbol{I}_{2\times 2} > 0, \tag{16}$$

with the rate $r = 5$ introduced in the diagonal, positive definite gain matrix $\boldsymbol{K}_P$. This choice guarantees that, in the absence of further disturbances, we have

$$e_x(t) = e_x(0)\exp\left(-5t\right), \qquad e_y(t) \equiv 0, \qquad \forall t \geq 0.$$

Figure 10 shows the desired and the actually executed Cartesian trajectory, together with the Cartesian position error, the feedback control commands, and the resulting motion of the robot joints. Finally, the initial value of the joint velocity control law (16) is $\dot{\boldsymbol{q}}_c(0) = \begin{pmatrix} 0 & 12.5221 \end{pmatrix}^T$ [rad/s].

Exercise #7

The following self-explanatory Matlab code computes the minimum number of bits of the multi-turn absolute encoder which satisfies the given specifications. This number is `bits` $= 15$: 5 bits

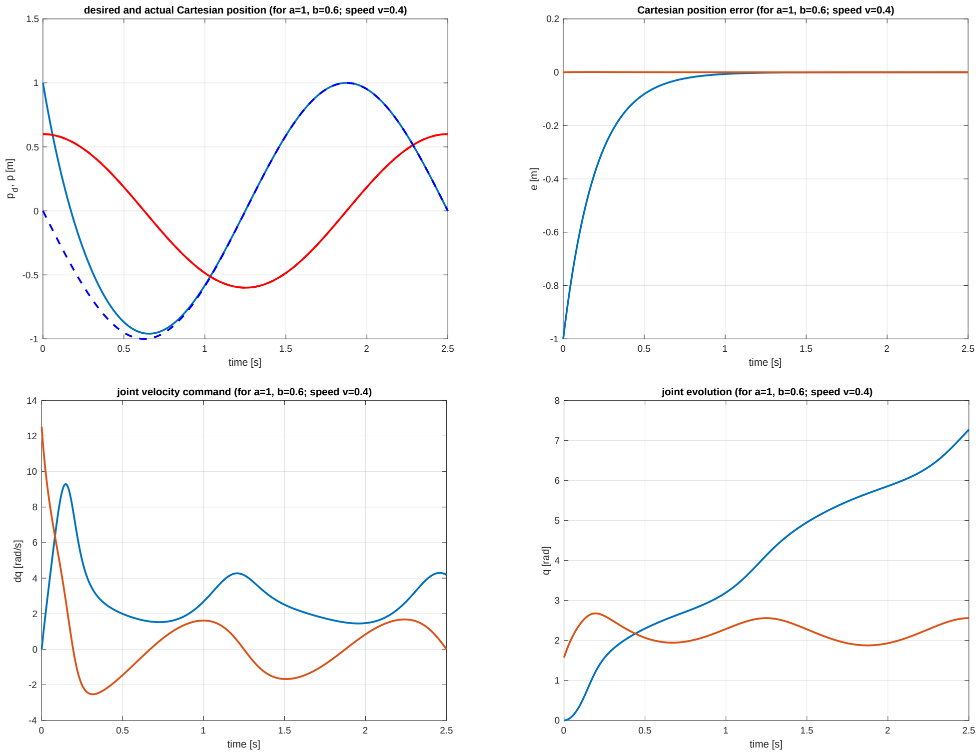

Figure 10: [Top] Components of the desired $\boldsymbol{p}_d(t)$ (dashed) and actually obtained $\boldsymbol{p}(t)$ (continuous) and those of the associated error $\boldsymbol{e}(t) = \boldsymbol{p}_d(t) - \boldsymbol{p}(t)$ (x in blue, y in red). [Bottom] Velocity control law $\dot{\boldsymbol{q}}_c(t)$ and resulting evolution $\boldsymbol{q}_c(t)$ (joint 1 in blue, 2 in red).

count separately the number of motor turns that covers the entire joint range of the flange, while 10 bits (equal to the number of tracks of the main encoder wheel) allow to achieve the desired angular resolution on the flange side of the transmission.

```
% data
joint_range=700 %[deg]      % range of the flange rotation
nr=30                       % reduction ratio
res_joint=0.02  %[deg]      % desired resolution at the flange side
% computation
disp('all angles are in degrees')
turns_joint=joint_range/360
turns_motor=nr*turns_joint
bits_turn=ceil(log2(turns_motor))-1
sectors_joint=360/res_joint
tracks_motor=sectors_joint/nr
res_motor=sectors_motor/360
bits_res=ceil(log2(sectors_motor))
bits=bits_turn+bits_res
% end
```

* * * * *