Solution

February 3, 2022

Exercise #1

A possible assignment of Denavit-Hartenberg (D-H) frames is shown in Fig. 5. The three $\boldsymbol{z}$ axes with a double arrow are coming out of the plane. The associated D-H parameters are given in Table 1. The signs of the q_i's in the table correspond to the robot configuration shown in the figure. The Crane-X7 robot has no offsets, it has both shoulder and wrist spherical, and a kinematics equivalent to that of the KUKA LWR IV robot.

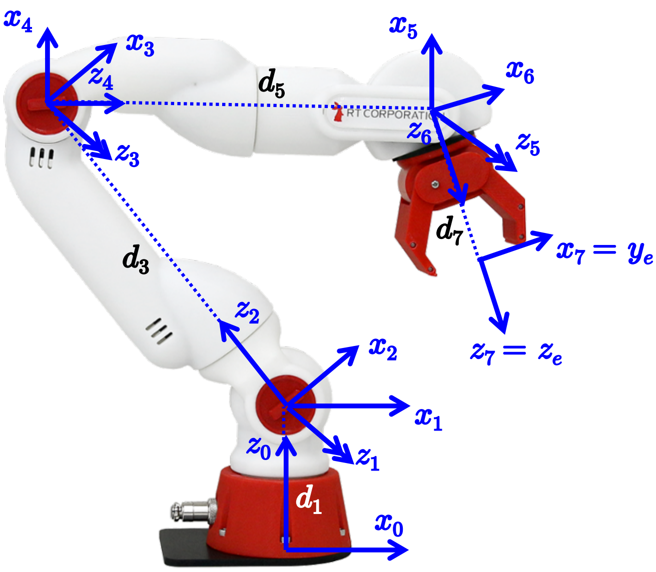

Figure 5: Assignment of D-H frames for the Crane-X7 robot.

i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	$d_1 = 105$	$q_1 = 0$
2	$-\pi/2$	0	0	$q_2 > 0$
3	$\pi/2$	0	$d_3 = 250$	$q_3 = 0$
4	$\pi/2$	0	0	$q_4 > 0$
5	$-\pi/2$	0	$d_5 = 250$	$q_5 = 0$
6	$\pi/2$	0	0	$q_6 < 0$
7	0	0	$d_7 = 103$	$q_7 = 0$

Table 1: Table of D-H parameters for the frame assignment of Fig. 5. Lengths are in [mm].

The constant rotation matrix from the seventh D-H frame to the end-effector frame is

$$ {}^7\boldsymbol{R}_e = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}. $$

Exercise #2

The YXY sequence of Euler angles $(\alpha_1, \alpha_2, \alpha_3)$ defines the following rotation matrix:

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

Therefore, we evaluate the initial orientation by the rotation matrix

$$ \boldsymbol{R}_i = \boldsymbol{R}_{YXY}\left(\frac{\pi}{4}, -\frac{\pi}{4}, \frac{\pi}{3}\right) = \begin{pmatrix} -0.7866 & -0.5 & 0.3624 \\ -0.6124 & 0.7071 & -0.3536 \\ -0.0795 & -0.5 & -0.8624 \end{pmatrix}. $$

From the final rotation matrix

$$ \boldsymbol{R}_f = \begin{pmatrix} 0 & 0.8660 & 0.5 \\ 0 & 0.5 & -0.8660 \\ -1 & 0 & 0 \end{pmatrix}, $$

we compute the relative rotation matrix as

$$ {}^i\boldsymbol{R}_f = \boldsymbol{R}_i^T\boldsymbol{R}_f = \begin{pmatrix} 0.0795 & -0.9874 & 0.1370 \\ 0.5 & -0.0795 & -0.8624 \\ 0.8624 & 0.1370 & 0.4874 \end{pmatrix}. $$

Denoting by R_{ij} the elements of ${}^i\boldsymbol{R}_f$, we use the solution formulas of the inverse axis/angle representation problem. This is not a singular case since

$$ \sin\theta = \frac{1}{2}\sqrt{(R_{21}-R_{12})^2 + (R_{13}-R_{31})^2 + (R_{32}-R_{23})^2} = 0.9666 \neq 0. $$

Using $\cos\theta = \frac{1}{2}\left(\text{trace}\left\{{}^i\boldsymbol{R}_f\right\} - 1\right) = -0.2563$ and the atan2 function, we obtain the two solutions

$$ \theta' = 1.83 \text{ [rad]}, \quad \boldsymbol{r}' = \begin{pmatrix} 0.5170 \\ -0.3752 \\ 0.7694 \end{pmatrix} \qquad \text{and} \qquad \theta'' = -\theta', \quad \boldsymbol{r}'' = -\boldsymbol{r}'. $$

With a constant angular velocity $\boldsymbol{\omega} = 1.1 \cdot \boldsymbol{r}'$ [rad/s], one traces the total angle θ' in a time

$$ T_\omega = \frac{1.83}{1.1} = 1.6636 \text{ [s]}. $$

Exercise #3

We need to formulate the complete task at the second-order differential level, i.e., in terms of accelerations. For the positional task of the end-effector, we set the common length of the links to $L = 1$ and compute

$$\boldsymbol{p} = \begin{pmatrix} \cos q_1 + \cos(q_1+q_2) + \cos(q_1+q_2+q_3) \\ \sin q_1 + \sin(q_1+q_2) + \sin(q_1+q_2+q_3) \end{pmatrix} = \boldsymbol{f}_p(\boldsymbol{q}).$$

Differentiating once, we build the Jacobian for the positional task

$$\dot{\boldsymbol{p}} = \frac{\partial \boldsymbol{f}_p(\boldsymbol{q})}{\partial \boldsymbol{q}}\,\dot{\boldsymbol{q}} = \begin{pmatrix} -\left(s_1+s_{12}+s_{123}\right) & -\left(s_{12}+s_{123}\right) & -s_{123} \\ c_1+c_{12}+c_{123} & c_{12}+c_{123} & c_{123} \end{pmatrix} = \boldsymbol{J}_p(\boldsymbol{q})\,\dot{\boldsymbol{q}},$$

where the shorthand notation has been used for trigonometric quantities (e.g., $s_{12} = \sin(q_1+q_2)$). Differentiating again, we have

$$\ddot{\boldsymbol{p}} = \boldsymbol{J}_p(\boldsymbol{q})\,\ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_p(\boldsymbol{q})\,\dot{\boldsymbol{q}} = \boldsymbol{J}_p(\boldsymbol{q})\,\ddot{\boldsymbol{q}} + \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}),$$

with

$$\boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \dot{\boldsymbol{J}}_p(\boldsymbol{q})\,\dot{\boldsymbol{q}} = -\begin{pmatrix} c_1\,\dot{q}_1^2 + c_{12}\left(\dot{q}_1+\dot{q}_2\right)^2 + c_{123}\left(\dot{q}_1+\dot{q}_2+\dot{q}_3\right)^2 \\ s_1\,\dot{q}_1^2 + s_{12}\left(\dot{q}_1+\dot{q}_2\right)^2 + s_{123}\left(\dot{q}_1+\dot{q}_2+\dot{q}_3\right)^2 \end{pmatrix}. \tag{1}$$

As for the angular velocity of the end effector, in this planar case it is

$$\omega_z = \dot{q}_1 + \dot{q}_2 + \dot{q}_3,$$

and thus

$$\dot{\omega}_z = \ddot{q}_1 + \ddot{q}_2 + \ddot{q}_3 = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}\ddot{\boldsymbol{q}} = \boldsymbol{J}_\omega\,\ddot{\boldsymbol{q}}.$$

The requested task is executed by imposing the desired linear acceleration $\ddot{\boldsymbol{p}} = \ddot{\boldsymbol{p}}_d$ to the end effector, while zeroing its angular acceleration $\dot{\omega}_z = \dot{\omega}_{z,d} = 0$ in order to keep $\omega_z = \omega_{z,d} = $ constant (whatever this value may be). Thus, the Jacobian of the complete task will be the (3×3) matrix

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{pmatrix} \boldsymbol{J}_p(\boldsymbol{q}) \\ \boldsymbol{J}_\omega \end{pmatrix} = \begin{pmatrix} -\left(s_1+s_{12}+s_{123}\right) & -\left(s_{12}+s_{123}\right) & -s_{123} \\ \left(c_1+c_{12}+c_{123}\right) & \left(c_{12}+c_{123}\right) & c_{123} \\ 1 & 1 & 1 \end{pmatrix}, \tag{2}$$

which is singular if and only if $\det \boldsymbol{J}(\boldsymbol{q}) = \sin q_2 = 0$. As long as the robot is away from the singularity $q_2 = 0$ or π, we can solve the relation for the task accelerations

$$\begin{pmatrix} \ddot{\boldsymbol{p}} \\ \dot{\omega}_z \end{pmatrix} = \begin{pmatrix} \boldsymbol{J}_p(\boldsymbol{q}) \\ \boldsymbol{J}_\omega \end{pmatrix}\ddot{\boldsymbol{q}} + \begin{pmatrix} \dot{\boldsymbol{J}}_p(\boldsymbol{q})\dot{\boldsymbol{q}} \\ 0 \end{pmatrix} = \boldsymbol{J}(\boldsymbol{q})\,\ddot{\boldsymbol{q}} + \begin{pmatrix} \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) \\ 0 \end{pmatrix} \tag{3}$$

in nominal conditions (i.e., for $\ddot{\boldsymbol{p}} = \ddot{\boldsymbol{p}}_d$ and $\dot{\omega}_z = \dot{\omega}_{z,d} = 0$) in terms of the joint acceleration command as

$$\ddot{\boldsymbol{q}} = \boldsymbol{J}^{-1}(\boldsymbol{q})\begin{pmatrix} \ddot{\boldsymbol{p}}_d - \boldsymbol{n}(\boldsymbol{q},\dot{\boldsymbol{q}}) \\ 0 \end{pmatrix}. \tag{4}$$

At the current robot state, in the nominal conditions

$$\boldsymbol{q}_d = \begin{pmatrix} \pi/4 \\ \pi/3 \\ -\pi/2 \end{pmatrix} \ \text{[rad]}, \qquad \dot{\boldsymbol{q}}_d = \begin{pmatrix} -0.8 \\ 1 \\ 0.2 \end{pmatrix} \ \text{[rad/s]},$$

the position of the end effector and its linear velocity are, respectively,

$$\boldsymbol{p}_d = \boldsymbol{f}_p(\boldsymbol{q}_d) = \begin{pmatrix} 1.4142 \\ 1.9319 \end{pmatrix} \ \text{[m]} \qquad \text{and} \qquad \dot{\boldsymbol{p}}_d = \boldsymbol{J}_p(\boldsymbol{q}_d)\,\dot{\boldsymbol{q}}_d = \begin{pmatrix} 0.2690 \\ -0.2311 \end{pmatrix} \ \text{[m/s]},$$

while the end-effector angular velocity is

$$\omega_{z,d} = \dot{q}_{1,d} + \dot{q}_{2,d} + \dot{q}_{3,d} = 0.4 \ \text{[rad/s]}.$$

For a desired linear acceleration of the end effector

$$\ddot{\boldsymbol{p}}_d = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \ \text{[m/s}^2\text{]},$$

the joint acceleration command (4) is evaluated, using (1) and (2), as

$$\begin{aligned}
\ddot{\boldsymbol{q}} &= \boldsymbol{J}^{-1}(\boldsymbol{q}) \begin{pmatrix} \ddot{\boldsymbol{p}}_d - \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ 0 \end{pmatrix} \\
&= \begin{pmatrix} -1.9319 & -1.2247 & -0.2588 \\ 1.4142 & 0.7071 & 0.9659 \\ 1 & 1 & 1 \end{pmatrix}^{-1} \begin{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} -0.5967 \\ -0.5326 \end{pmatrix} \\ 0 \end{pmatrix} = \begin{pmatrix} 1.2322 \\ -3.7873 \\ 2.5551 \end{pmatrix} \ \text{[rad/s}^2\text{]}.
\end{aligned}$$

To correct an error (in any component) that may arise during the execution of the complete task by the robot end effector, feedback terms should be added to the nominal command (4). Due to the task structure, a proportional-derivative (PD) action is used on the error along the positional trajectory tracking task and a simpler proportional (P) action is used on the error in the regulation task of the angular velocity. The resulting law is

$$\begin{aligned}
\ddot{\boldsymbol{q}} &= \boldsymbol{J}^{-1}(\boldsymbol{q}) \begin{pmatrix} \ddot{\boldsymbol{p}}_d + \boldsymbol{K}_D\,(\dot{\boldsymbol{p}}_d - \dot{\boldsymbol{p}}) + \boldsymbol{K}_P\,(\boldsymbol{p}_d - \boldsymbol{p}) - \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ k_\omega\,(\omega_{z,d} - \omega_z) \end{pmatrix} \\
&= \boldsymbol{J}^{-1}(\boldsymbol{q}) \begin{pmatrix} \ddot{\boldsymbol{p}}_d + \boldsymbol{K}_D\,(\dot{\boldsymbol{p}}_d - \boldsymbol{J}_p(\boldsymbol{q})\,\dot{\boldsymbol{q}}) + \boldsymbol{K}_P\,\big(\boldsymbol{p}_d - \boldsymbol{f}_p(\boldsymbol{q})\big) - \boldsymbol{n}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ k_\omega\,(\omega_{z,d} - \boldsymbol{J}_\omega\,\dot{\boldsymbol{q}}) \end{pmatrix},
\end{aligned} \tag{5}$$

with diagonal gain matrices $K_P > 0$ and $K_D > 0$, and a scalar gain $k_\omega > 0$. In fact, define the task errors as

$$\boldsymbol{e}_p(t) = \boldsymbol{p}_d(t) - \boldsymbol{p}(t) \quad \Rightarrow \quad \dot{\boldsymbol{e}}_p(t) = \boldsymbol{e}_v(t) = \dot{\boldsymbol{p}}_d(t) - \dot{\boldsymbol{p}}(t), \qquad\qquad e_\omega(t) = \omega_{z,d} - \omega_z(t).$$

Plugging (5) into (3) and simplifying terms, yields the two (decoupled) error dynamics

$$\ddot{\boldsymbol{e}}_p + \boldsymbol{K}_D \dot{\boldsymbol{e}}_p + \boldsymbol{K}_P \boldsymbol{e}_p = 0, \qquad\qquad \dot{e}_\omega + k_\omega e_\omega = 0,$$

whose evolutions will exponentially converge to zero thanks to the choice of the gains. Indeed, if only some of the errors are present, the same control law (5) will work accordingly.

Exercise #4

The problem is addressed by adding a via point P_{mid} in the Cartesian space so as to avoid the obstacle, converting then start, via, and goal points into the joint space of the PR robot, and fitting a spline trajectory to this to guarantee the desired smoothness. It is convenient to add the via point at the middle of the x-displacement D, sufficiently below the obstacle (but not too far away, so as to keep the robot travel limited). In the following, we choose $P_{mid} = (S + \Delta/2, L/4)$.

Note also that the start and goal positions correspond to a singularity for the PR robot (with $q_2 = \pi/2$, its Jacobian $\boldsymbol{J}_s = \boldsymbol{J}(q_2 = 0)$ has rank one). A simple path planning with straight lines joining P_{start} to P_{mid} and P_{mid} to P_{goal} would be unfeasible because the directions of these lines would not belong to $\mathcal{R}\{\boldsymbol{J}_s\}$ at P_{start} and P_{goal}. Moreover, the tangent discontinuity of that path at P_{mid} would force a stop of the robot, contrary to the desired motion requirements.

With the above in mind, we convert first the three positions in the joint space. We have immediately

$$P_{start} = \begin{pmatrix} S \\ L \end{pmatrix} \rightarrow \boldsymbol{q}_s = \begin{pmatrix} S \\ \pi/2 \end{pmatrix}, \qquad P_{goal} = \begin{pmatrix} S+\Delta \\ L \end{pmatrix} \rightarrow \boldsymbol{q}_g = \begin{pmatrix} S+\Delta \\ \pi/2 \end{pmatrix}.$$

For the via point, we use the closed-form expression of the inverse kinematics of the PR robot[2],

$$\boldsymbol{p} = \begin{pmatrix} p_x \\ p_y \end{pmatrix} = \begin{pmatrix} q_1 + L\cos q_2 \\ L \sin q_2 \end{pmatrix} \rightarrow \boldsymbol{q}^{[+/-]} = \begin{pmatrix} q_1^{[+/-]} \\ q_2^{[+/-]} \end{pmatrix} = \begin{pmatrix} p_x \pm \sqrt{L^2 - p_y^2} \\ \operatorname{atan2}\left\{p_y, p_x - q_1^{[+/-]}\right\} \end{pmatrix},$$

and choose the '$-$' solution (i.e., with the base of the second link on the left of P_{mid})

$$P_{mid} = \begin{pmatrix} S + \Delta/2 \\ L/4 \end{pmatrix} \rightarrow \boldsymbol{q}_m = \begin{pmatrix} S + \Delta/2 - L\sqrt{15}/4 \\ \operatorname{atan2}\left\{1, \sqrt{15}\right\} \end{pmatrix}.$$

Moreover, again because of the Cartesian symmetry, we will impose the passage through the via point P_{mid} (or, equivalently, through the joint configuration $\boldsymbol{q}_m$) at the mid motion time $t = T/2$.

The construction of the spline trajectory can be performed in a straightforward way, being composed by only two cubic polynomials (for each joint component), namely $\boldsymbol{q}_A(t)$, for $t \in [0, T/2]$, and $\boldsymbol{q}_B(t)$, for $t \in [T/2, T]$. Both $\boldsymbol{q}_A$ and $\boldsymbol{q}_B$ are (timed) vectors in $\mathbb{R}^2$. As in the general case of multiple cubics, we introduce the joint velocity at the mid knot as the (unknown) vector parameter $\boldsymbol{v}_m = \dot{\boldsymbol{q}}_A(T/2) = \dot{\boldsymbol{q}}_B(T/2) \in \mathbb{R}^2$. The two cubics are expressed in normalized times as

$$\boldsymbol{q}_A(\tau_A) = \boldsymbol{q}_s + \boldsymbol{c}_{A,1}\,\tau_A + \boldsymbol{c}_{A,2}\,\tau_A^2 + \boldsymbol{c}_{A,3}\,\tau_A^3, \quad \tau_A = \frac{t}{T/2} = \frac{2t}{T} \in [0,1] \tag{6}$$

and[3]

$$\boldsymbol{q}_B(\tau_B) = \boldsymbol{q}_g + \boldsymbol{c}_{B,1}\,(\tau_B - 1) + \boldsymbol{c}_{B,2}\,(\tau_B - 1)^2 + \boldsymbol{c}_{B,3}\,(\tau_B - 1)^3, \quad \tau_B = \frac{t - (T/2)}{T/2} = \frac{2t}{T} - 1 \in [0,1], \tag{7}$$

Next, we impose the boundary conditions. For $\boldsymbol{q}_A(\tau_A)$ in (6), we have

$$\begin{array}{lllll} t = 0 & \rightarrow \tau_A = 0: & \dot{\boldsymbol{q}}_A(0) = \boldsymbol{0} & \rightarrow & \boldsymbol{c}_{A,1} = \boldsymbol{0} \\ t = \dfrac{T}{2} & \rightarrow \tau_A = 1: & \boldsymbol{q}_A(1) = \boldsymbol{q}_m & \rightarrow & \boldsymbol{c}_{A,2} + \boldsymbol{c}_{A,3} = \boldsymbol{q}_m - \boldsymbol{q}_s \\ t = \dfrac{T}{2} & \rightarrow \tau_A = 1: & \dot{\boldsymbol{q}}_A(1) = \boldsymbol{v}_m & \rightarrow & 4\,\boldsymbol{c}_{A,2} + 6\,\boldsymbol{c}_{A,3} = \boldsymbol{v}_m\,T, \end{array}$$

[2]Remember that the two arguments of the atan2 function can be arbitrarily scaled by a positive factor.

[3]One could use also a cubic with powers of τ_B, rather than of $(\tau_B - 1)$. Indeed, the final result would be the same. The choice (7) gives time specularity to the treatment, introducing the goal value $\boldsymbol{q}_g$ as constant in the cubic.

yielding

$$\boldsymbol{c}_{A,2} = 3\,(\boldsymbol{q}_m - \boldsymbol{q}_s) - \frac{T}{2}\,\boldsymbol{v}_m, \qquad \boldsymbol{c}_{A,3} = 2\,(\boldsymbol{q}_s - \boldsymbol{q}_m) + \frac{T}{2}\,\boldsymbol{v}_m.$$

Similarly, for $\boldsymbol{q}_B(\tau_B)$ in (7), we have

$$\begin{array}{llllll} t = T & \rightarrow & \tau_B = 1: & \dot{\boldsymbol{q}}_B(1) = \boldsymbol{0} & \rightarrow & \boldsymbol{c}_{B,1} = \boldsymbol{0} \\ t = \dfrac{T}{2} & \rightarrow & \tau_B = 0: & \boldsymbol{q}_B(0) = \boldsymbol{q}_m & \rightarrow & \boldsymbol{c}_{B,2} - \boldsymbol{c}_{B,3} = \boldsymbol{q}_m - \boldsymbol{q}_g \\ t = \dfrac{T}{2} & \rightarrow & \tau_B = 0: & \dot{\boldsymbol{q}}_B(0) = \boldsymbol{v}_m & \rightarrow & -4\,\boldsymbol{c}_{B,2} + 6\,\boldsymbol{c}_{B,3} = \boldsymbol{v}_m\,T, \end{array}$$

yielding in this case

$$\boldsymbol{c}_{B,2} = 3\left(\boldsymbol{q}_m - \boldsymbol{q}_g\right) + \frac{T}{2}\,\boldsymbol{v}_m, \qquad \boldsymbol{c}_{B,3} = 2\left(\boldsymbol{q}_m - \boldsymbol{q}_g\right) + \frac{T}{2}\,\boldsymbol{v}_m.$$

Finally, we find the unknown value $\boldsymbol{v}_m$ by imposing continuity of the accelerations at the mid point instant $t = T/2$ (i.e., at $\tau_A = 1$ and $\tau_B = 0$). Since

$$\ddot{\boldsymbol{q}}_A(\tau_A) = \frac{8}{T^2}\,\left(\boldsymbol{c}_{A,2} + 3\,\boldsymbol{c}_{A,3}\,\tau_A\right), \qquad \ddot{\boldsymbol{q}}_B(\tau_B) = \frac{8}{T^2}\,\left(\boldsymbol{c}_{B,2} + 3\,\boldsymbol{c}_{B,3}\,(\tau_B - 1)\right),$$

we have

$$\ddot{\boldsymbol{q}}_A(0) = \ddot{\boldsymbol{q}}_B(1) \quad \Rightarrow \quad \boldsymbol{c}_{A,2} + 3\,\boldsymbol{c}_{A,3} = \boldsymbol{c}_{B,2} - 3\,\boldsymbol{c}_{B,3} \quad \Rightarrow \quad \boldsymbol{v}_m = \frac{3}{2T}\left(\boldsymbol{q}_g - \boldsymbol{q}_s\right).$$

Note that the value of $\boldsymbol{q}_m$ plays no role in the definition of the (unique) midpoint velocity $\boldsymbol{v}_m$. Moreover, it is clear that $\boldsymbol{v}_m \neq \boldsymbol{0}$, satisfying the condition of no stops during motion.

As a result, replacing the coefficients of the cubics with the symbolic expression that have been found, we obtain from (6) and (7)

$$\boldsymbol{q}_A(\tau_A) = \boldsymbol{q}_s + \frac{1}{4}\left(12\,\boldsymbol{q}_m - 3\,\boldsymbol{q}_g - 9\,\boldsymbol{q}_s\right)\tau_A^2 + \frac{1}{4}\left(5\,\boldsymbol{q}_s + 3\,\boldsymbol{q}_g - 8\,\boldsymbol{q}_m\right)\tau_A^3, \tag{8}$$

and

$$\boldsymbol{q}_B(\tau_B) = \boldsymbol{q}_g + \frac{1}{4}\left(12\,\boldsymbol{q}_m - 9\,\boldsymbol{q}_g - 3\,\boldsymbol{q}_s\right)(\tau_B - 1)^2 + \frac{1}{4}\left(8\,\boldsymbol{q}_m - 5\,\boldsymbol{q}_g - 3\,\boldsymbol{q}_s\right)(\tau_B - 1)^3. \tag{9}$$

Finally, we provide and example using the following numerical data:

$$L = 1, \qquad S = 0, \qquad \Delta = 3 \text{ [m]}, \qquad T = 4 \text{ [s]}. \tag{10}$$

These lead to the following values in the previous formulas

$$P_{start} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \qquad P_{mid} = \begin{pmatrix} 1.5 \\ 0.25 \end{pmatrix}, \qquad P_{goal} = \begin{pmatrix} 3 \\ 1 \end{pmatrix},$$

$$\boldsymbol{q}_s = \begin{pmatrix} 0 \\ \pi/2 \end{pmatrix}, \qquad \boldsymbol{q}_m = \begin{pmatrix} 0.5318 \\ 0.2527 \end{pmatrix}, \qquad \boldsymbol{q}_g = \begin{pmatrix} 3 \\ \pi/2 \end{pmatrix} \quad \Rightarrow \quad \boldsymbol{v}_m = \begin{pmatrix} 1.1250 \\ 0 \end{pmatrix},$$

from which

$$\boldsymbol{q}_A(\tau_A) = \begin{pmatrix} 0 \\ 1.5708 \end{pmatrix} - \begin{pmatrix} 0.6547 \\ 3.9543 \end{pmatrix}\tau_A^2 + \begin{pmatrix} 1.1865 \\ 2.6362 \end{pmatrix}\tau_A^3$$

and

$$\boldsymbol{q}_B(\tau_B) = \begin{pmatrix} 3 \\ 1.5708 \end{pmatrix} - \begin{pmatrix} 5.1547 \\ 3.9543 \end{pmatrix} (\tau_B - 1)^2 - \begin{pmatrix} 2.6865 \\ 2.6362 \end{pmatrix} (\tau_B - 1)^3 .$$

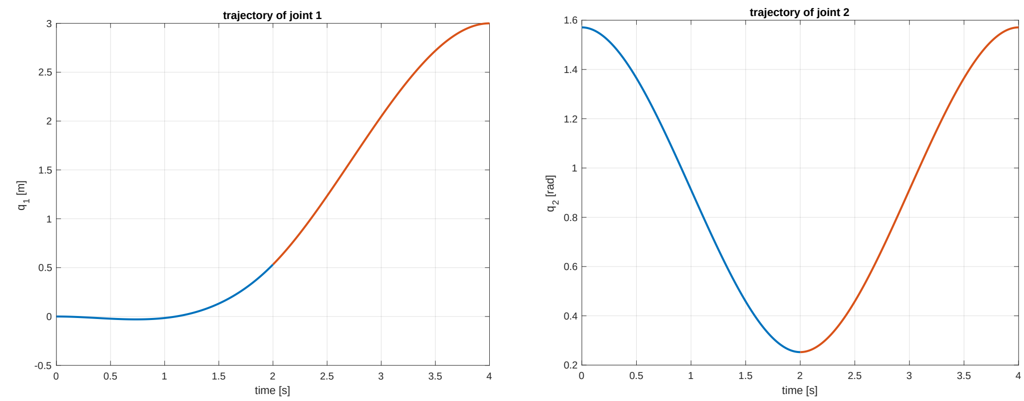

Figure 6: Spline joint trajectories of the PR robot.

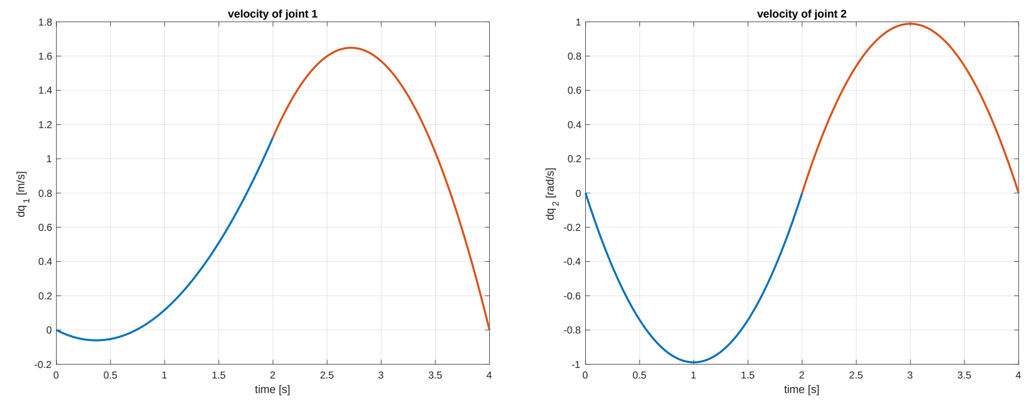

Figure 7: Joint velocities of the PR robot.

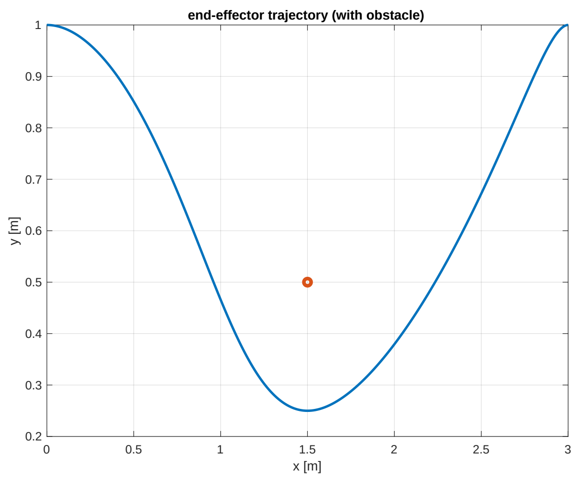

Figure 8: Cartesian path of the end effector of the PR robot. The obstacle (circle in red) is avoided.

Figures 6–8 show the results of the trajectory planning. The Cartesian path avoids the obstacle, although it is slightly asymmetric w.r.t. the via point P_{mid}. The first joint retracts a bit in the initial phase of the motion (when $\dot{q}_1 < 0$), before increasing constantly until the goal is reached. Note that the Cartesian path starts and ends with an horizontal tangent, being this the only admissible direction in the range of the singular Jacobian $\boldsymbol{J}$ at $\boldsymbol{q}_s$ and $\boldsymbol{q}_g$. Finally, there is no instant $t \in (0, T) = (0, 3)$ (excluding obviously the interval boundaries) such that $\dot{\boldsymbol{q}}(t) = \mathbf{0}$.

As an additional comment, the path will remain the same when scaling the motion time. Figure 9 shows the same instance of motion planning for $T = 1$ and $T = 10$. Indeed, the joint velocities have a scaled profile (by a factor $k = 4/1 = 4$ in the first case and $k = 4/10 = 0.4$ in the second).

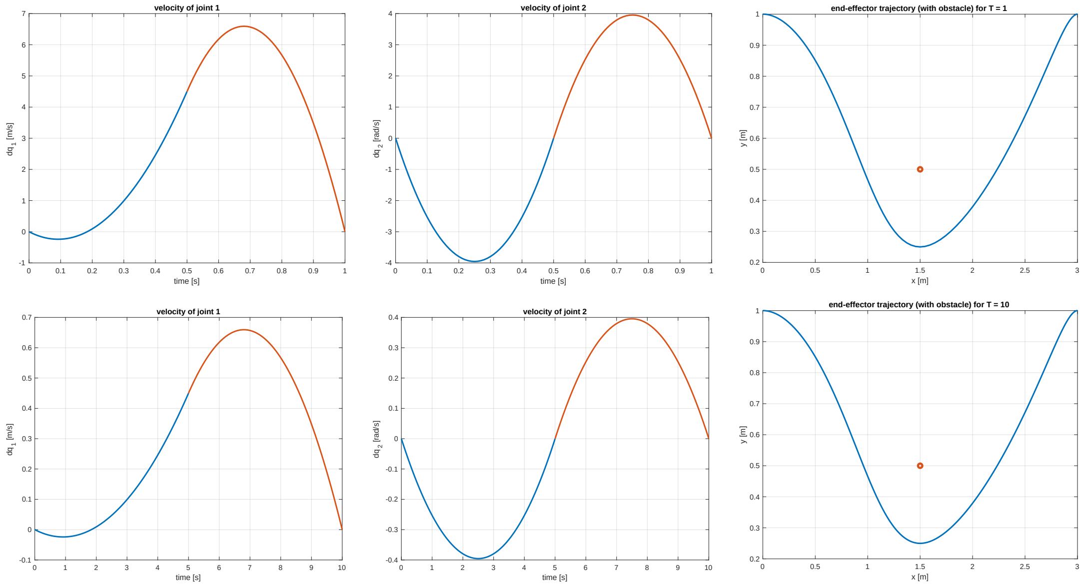

Figure 9: Trajectory planning for the same problem, with $T = 1$ [s] (top) and $T = 10$ [s] (bottom): joint velocities scale linearly while the Cartesian path remains the same.

Note in conclusion that the assumed condition $\Delta > L/2$ is quite stringent. For shorter displacements D along the x-direction between P_{start} and P_{goal}, one via point will not be sufficient to obtain obstacle avoidance of the Cartesian path, at least with the chosen joint space planning method. On the other hand, longer displacements Δ for a given link length L will provide more symmetric solutions. These two aspects are illustrated in Fig. 10 for $L = 1$. The Cartesian path hits the obstacle when $\Delta = 0.5L = 0.5$, while its is practically symmetric with $\Delta = 10L = 10$.

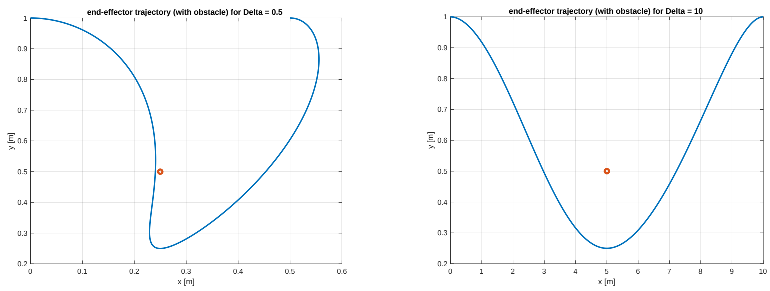

Figure 10: Cartesian paths resulting from the used trajectory planning method for the same problem, but with $D = 0.5$ [m] (left) and $D = 10$ [m] (right).

Final remark. An alternative solution can be developed when assuming that the range of the second (revolute) joint is unlimited. In this case, one could replace the final target configuration $\boldsymbol{q}_g = \begin{pmatrix} S+\Delta & \pi/2 \end{pmatrix}^T$ by $\boldsymbol{q}'_g = \begin{pmatrix} S+\Delta & \pi/2+2\pi \end{pmatrix}^T$, and let a single smooth trajectory (a cubic or quintic polynomial) connect $\boldsymbol{q}_s$ to $\boldsymbol{q}'_g$. The second link would make then a complete counter-clockwise rotation of 360° while the first joint is translating forward, avoiding thus the obstacle. However, even in the absence of joint limits, the price to pay with this strategy is a much higher speed reached by joint 2 at the trajectory midpoint (the motion should be coordinated, i.e., started and completed at the same instants of time for both joints). With the numerical values in (10), there would be a peak $\dot{q}_{2,max} = \dot{q}_2(T/2) = 3\pi/T \simeq 2.36$ [rad/s] for a rest-to-rest cubic trajectory and a peak $\dot{q}_{2,max} = \dot{q}_2(T/2) = 3.75\pi/T \simeq 2.95$ [rad/s] for a quintic trajectory (with zero acceleration at the boundaries), as opposed to $|\dot{q}_{2,max}| = 1$ [rad/s] with the solution in Fig. 7.

Exercise #5

The reduction ratio of the complete transmission is the product of the ratio $n_{r,g} = r_1/r_2$ of the radiuses of the two gear wheels, times the ratio $n_{r,p} = r_3/r_4$ of the radiuses of the two pulleys connected by the belt. While the gears invert the rotation direction, the pulleys preserve the same direction. Thus, the link will rotate in the opposite way of the motor (around their respective axes, $\boldsymbol{z}_m$ and $\boldsymbol{z}_j$). The following (symbolic/numeric) Matlab code computes the complete reduction ratio

$$n_r = \left| \frac{\dot{\theta}_m}{\dot{\theta}_l} \right| \geq 1$$

of the transmission and, accordingly, the time $T_\theta > 0$ needed for the link to rotate by 90°. With the given numerical values, when the motor spins at $\dot{\theta}_m = 10$ [rad/s], it is

$$n_r = 12, \qquad \dot{\theta}_l = -\frac{\dot{\theta}_m}{n_r} = -0.8333 \text{ [rad/s]}, \qquad T_\theta = \frac{\pi/2}{|\dot{\theta}_l|} = \frac{\pi/2}{|\dot{\theta}_m|/n_r} = 1.8850 \text{ [s]},$$

and the link rotates clockwise (CW).

```
syms r1 r2 r3 r4 dtheta_m real   % D and L are irrelevant...

% reduction ratio (symbolic)

dtheta_g=-(r1/r2)*dtheta_m
dtheta_l=(r3/r4)*dtheta_g
nr=abs(dtheta_m/dtheta_l)

% time for 90 [deg] of link rotation

T_th=(pi/2)/abs(dtheta_l)

% numerical values (radiuses in [mm])

dtheta_m=10     % rad/s
dtheta_g=subs(dtheta_g,{r1,r2},{20,60});
dtheta_g=eval(dtheta_g)
dtheta_l=subs(dtheta_l,{r1,r2,r3,r4},{20,60,8,32});
dtheta_l=eval(dtheta_l)
```

```
nr=subs(nr,{r1,r2,r3,r4},{20,60,8,32});
nr=eval(nr)
if dtheta_l > 0
   disp('link rotation is CCW')
else
   disp('link rotation is CW')
end
T_th=subs(T_th,{r1,r2,r3,r4},{20,60,8,32});
T_th=eval(T_th)

% end
```

* * * * *