Robotics 2

June 12, 2024

Exercise 1

Table 1 contains the Denavit-Hartenberg parameters of a robot with three revolute joints.

i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	0	θ_1
2	0	$a_2 > 0$	0	θ_2
3	0	$a_3 > 0$	0	θ_3

Table 1: Denavit-Hartenberg parameters of a 3R robot

Use the recursive algorithm based on moving frames to compute the kinetic energy of this robot, making reasonable assumptions to simplify the inertial properties of the links, e.g., assume that each link has a cylindric body with uniformly distributed mass and center of mass on the kinematic axis of the link. Provide at the end the robot inertia matrix $\boldsymbol{M}(\boldsymbol{q})$ and determine a minimal linear parametrization of the inertial term $\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}$.

Exercise 2

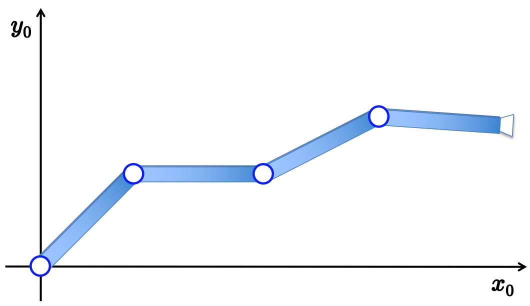

Figure 1: A 4R planar robot.

Consider the 4R planar robot in Fig. 1, having links with unit length. The primary task for the end-effector is to point at a moving target in the plane (x_0, y_0). The available extra degrees of freedom of the robot are used to keep the joints close to the middle of their ranges, which are defined, using D-H variables, as

$$q_1 \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \qquad q_2 \in \left[0, \frac{\pi}{2}\right] \qquad q_i \in \left[-\frac{\pi}{4}, \frac{\pi}{4}\right], \quad i = 3, 4. \tag{1}$$

Define the primary task Jacobian $\boldsymbol{J}(\boldsymbol{q})$ and determine the joint velocity command $\dot{\boldsymbol{q}} \in \mathbb{R}^4$ that realizes at best the desired robot behavior. Provide the numerical value of $\dot{\boldsymbol{q}}$ when the robot is at $\boldsymbol{q} = (0, \pi/2, 0, -\pi/4)$, while the target is being correctly pointed at and has an instantaneous velocity $\dot{\boldsymbol{p}}_t = (-1, -1)$ [m/s]. How would you modify the joint velocity command if the target was placed at $\boldsymbol{p}_t = (0, 2.5)$ [m] and had the same previous velocity $\dot{\boldsymbol{p}}_t$? Provide the new numerical value of the command $\dot{\boldsymbol{q}}$ with your modified control strategy.

Exercise 3

The dynamics of a robot with n elastic joints moving in the absence of gravity is described by the $2n$ second-order differential equations

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{K}\,(\boldsymbol{q}-\boldsymbol{\theta}) = \boldsymbol{0} \tag{2}$$

$$\boldsymbol{B}_m\ddot{\boldsymbol{\theta}} + \boldsymbol{K}\,(\boldsymbol{\theta}-\boldsymbol{q}) = \boldsymbol{u}, \tag{3}$$

where (2) are the n link equations and (3) are the n motor equations, with link positions $\boldsymbol{q} \in \mathbb{R}^n$ and motor positions $\boldsymbol{\theta} \in \mathbb{R}^n$ as generalized coordinates. $\boldsymbol{K}$ is the diagonal, positive definite stiffness matrix of the joints, while $\boldsymbol{B}_m$ is the diagonal, positive definite inertia matrix of the motors.

For the input torque $\boldsymbol{u} \in \mathbb{R}^n$, consider the PD control law on the motor variables

$$\boldsymbol{u} = \boldsymbol{K}_P\,(\boldsymbol{\theta}_d - \boldsymbol{\theta}) - \boldsymbol{K}_D\,\dot{\boldsymbol{\theta}}, \tag{4}$$

where $\boldsymbol{K}_P$ and $\boldsymbol{K}_D$ are diagonal and positive definite gain matrices, and $\boldsymbol{\theta}_d \in \mathbb{R}^n$ is a desired constant motor position. Prove that

$$(\boldsymbol{q},\boldsymbol{\theta},\dot{\boldsymbol{q}},\dot{\boldsymbol{\theta}}) = (\boldsymbol{\theta}_d,\boldsymbol{\theta}_d,\boldsymbol{0},\boldsymbol{0})$$

is the unique, globally asymptotically stable equilibrium state for the closed-loop system made by eqs. (2) and (3) under the control law (4).

Exercise 4

The PR robot in Fig. 2 may be subject to a generic unknown fault u_{f1} on the force produced by first actuator. Based on the symbolic terms of the dynamic model of this robot, design a scalar residual function $r_1(t)$ such that $r_1(t) \equiv 0$ in the absence of a fault of this actuator, while it evolves otherwise as $\dot{r}_1(t) = k_1\,(u_{f1}(t) - r_1(t))$, for a given $k_1 > 0$. If the unknown fault consists in a constant force $u_{f1} = 2$ N being subtracted to the commanded force u_1 at the first joint starting from the instant $t_0 = 0$, what will be the evolution of $r_1(t)$ for $t \geq t_0$ and its value at steady state?

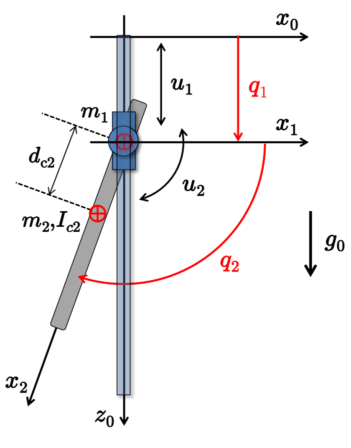

Figure 2: A PR planar robot under gravity.

[240 minutes (4 hours); open books]

Solution

June 12, 2024

Exercise 1

From the Denavit-Hartenberg table of parameters (Tab. 1), we prepare the vectors and matrices needed in each step of the recursive algorithm with moving frames for computing the kinetic energy.

Rotation matrix

$$
{}^0\boldsymbol{R}_1(q_1) = \begin{pmatrix} c_1 & 0 & s_1 \\ s_1 & 0 & -c_1 \\ 0 & 1 & 0 \end{pmatrix} \quad {}^1\boldsymbol{R}_2(q_2) = \begin{pmatrix} c_2 & -s_2 & 0 \\ s_2 & c_2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad {}^2\boldsymbol{R}_3(q_3) = \begin{pmatrix} c_3 & -s_3 & 0 \\ s_3 & c_3 & 0 \\ 0 & 0 & 1 \end{pmatrix}.
$$

Position vector between origins in previous frame

$$
{}^0\boldsymbol{r}_{01} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \qquad {}^1\boldsymbol{r}_{12}(q_2) = \begin{pmatrix} a_2 c_2 \\ a_2 s_2 \\ 0 \end{pmatrix} \qquad {}^2\boldsymbol{r}_{23}(q_3) = \begin{pmatrix} a_3 c_3 \\ a_3 s_3 \\ 0 \end{pmatrix}.
$$

Position vector between origins in moving frame

$$
{}^1\boldsymbol{r}_{01} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \qquad {}^2\boldsymbol{r}_{12} = \begin{pmatrix} a_2 \\ 0 \\ 0 \end{pmatrix} \qquad {}^3\boldsymbol{r}_{23} = \begin{pmatrix} a_3 \\ 0 \\ 0 \end{pmatrix}.
$$

Based on the given assumptions on mass distribution of the links, we have also:

Position of link CoM in moving frame

$$
{}^1\boldsymbol{r}_{c1} = \begin{pmatrix} 0 \\ d_{c1} \\ 0 \end{pmatrix} \qquad {}^2\boldsymbol{r}_{c2} = \begin{pmatrix} d_{c2} - a_2 \\ 0 \\ 0 \end{pmatrix} \qquad {}^3\boldsymbol{r}_{c3} = \begin{pmatrix} d_{c3} - a_3 \\ 0 \\ 0 \end{pmatrix},
$$

being d_{ci} the distance from the origin O_{i-1} to the center of mass (CoM) of link i along the $\boldsymbol{x}_i$ axis.

Link inertia matrix in moving frame

$$
{}^1\boldsymbol{I}_{c1} = \begin{pmatrix} I_{c1,x} & 0 & 0 \\ 0 & I_{c1,y} & 0 \\ 0 & 0 & I_{c1,x} \end{pmatrix} \quad {}^2\boldsymbol{I}_{c2} = \begin{pmatrix} I_{c2,x} & 0 & 0 \\ 0 & I_{c2,z} & 0 \\ 0 & 0 & I_{c2,z} \end{pmatrix} \quad {}^3\boldsymbol{I}_{c3} = \begin{pmatrix} I_{c3,x} & 0 & 0 \\ 0 & I_{c3,z} & 0 \\ 0 & 0 & I_{c3,z} \end{pmatrix},
$$

where we used the symmetry of the cylindrical links with respect to the two minor axes that are transversal to their major axis (which are, respectively, y_1, x_2 and x_3).

With the above structures, the algorithm is initialized with ${}^0\boldsymbol{\omega}_0 = \boldsymbol{0}$, ${}^0\boldsymbol{v}_0 = \boldsymbol{0}$ and yields:

Step 1

$$
{}^1\boldsymbol{\omega}_1 = \begin{pmatrix} 0 \\ \dot{q}_1 \\ 0 \end{pmatrix} \qquad {}^1\boldsymbol{v}_1 = \boldsymbol{0} \qquad {}^1\boldsymbol{v}_{c1} = \boldsymbol{0}
$$

Kinetic energy of link 1

$$
T_1 = \frac{1}{2}\, I_{c1,y}\dot{q}_1^2.
$$

Step 2

$$
{}^2\boldsymbol{\omega}_2 = \begin{pmatrix} s_2\dot{q}_1 \\ c_2\dot{q}_1 \\ \dot{q}_2 \end{pmatrix} \qquad {}^2\boldsymbol{v}_2 = \begin{pmatrix} 0 \\ a_2\dot{q}_2 \\ -a_2c_2\dot{q}_1 \end{pmatrix} \qquad {}^2\boldsymbol{v}_{c2} = \begin{pmatrix} 0 \\ d_{c2}\dot{q}_2 \\ -d_{c2}c_2\dot{q}_1 \end{pmatrix}
$$

Kinetic energy of link 2

$$
T_2 = \frac{1}{2}\left(I_{c2,x}\, s_2^2 + (I_{c2,z} + m_2 d_{c2}^2)\, c_2^2\right)\dot{q}_1^2 + \frac{1}{2}\left(I_{c2,z} + m_2 d_{c2}^2\right)\dot{q}_2^2.
$$

Step 3

$$
{}^3\boldsymbol{\omega}_3 = \begin{pmatrix} s_{23}\dot{q}_1 \\ c_{23}\dot{q}_1 \\ \dot{q}_2 + \dot{q}_3 \end{pmatrix} \quad {}^3\boldsymbol{v}_3 = \begin{pmatrix} a_2s_3\dot{q}_2 \\ a_2c_3\dot{q}_2 + a_3\left(\dot{q}_2 + \dot{q}_3\right) \\ -\left(a_2c_2 + a_3c_{23}\right)\dot{q}_1 \end{pmatrix} \quad {}^3\boldsymbol{v}_{c3} = \begin{pmatrix} a_2s_3\dot{q}_2 \\ a_2c_3\dot{q}_2 + d_{c3}\left(\dot{q}_2 + \dot{q}_3\right) \\ -\left(a_2c_2 + d_{c3}c_{23}\right)\dot{q}_1 \end{pmatrix}
$$

Kinetic energy of link 3

$$
\begin{aligned}
T_3 = {} & \frac{1}{2}\left(I_{c3,x}\, s_{23}^2 + I_{c3,z}\, c_{23}^2 + m_3\left(a_2c_2 + d_{c3}c_{23}\right)^2\right)\dot{q}_1^2 + \frac{1}{2}m_3a_2^2\dot{q}_2^2 \\
& + \frac{1}{2}\left(I_{c3,z} + m_3 d_{c3}^2\right)\left(\dot{q}_2 + \dot{q}_3\right)^2 + \frac{1}{2}\left(2\, m_3 a_2 d_{c3} c_3\, \dot{q}_2\left(\dot{q}_2 + \dot{q}_3\right)\right).
\end{aligned}
$$

Thus, the kinetic energy of the 3R robot is

$$
T = T_1 + T_2 + T_3 = \frac{1}{2}\,\dot{\boldsymbol{q}}^T\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}.
$$

In the expressions of the elements of the inertia matrix $\boldsymbol{M}(\boldsymbol{q})$, one can eliminate all appearances of square roots of the *sine* functions by setting

$$
s_3^2 = 1 - c_3^2 \qquad s_{23}^2 = 1 - c_{23}^2.
$$

With these substitutions, one recognizes the presence of six independent dynamic coefficients ρ_i in the inertia matrix:

$$
\boldsymbol{M}(\boldsymbol{q}) = \begin{pmatrix} \rho_1 + \rho_2\, c_2^2 + \rho_3\, c_{23}^2 + 2\rho_5\, c_2c_{23} & 0 & 0 \\ 0 & \rho_4 + 2\rho_5\, c_3 & \rho_6 + \rho_5\, c_3 \\ 0 & \rho_6 + \rho_5\, c_3 & \rho_6 \end{pmatrix},
$$

where

$$
\begin{aligned}
\rho_1 &= I_{c1,y} + I_{c2,x} + I_{c3,x} \\
\rho_2 &= I_{c2,z} + m_2 d_{c2}^2 - I_{c2,x} + m_3 a_2^2 \\
\rho_3 &= I_{c3,z} + m_3 d_{c3}^2 - I_{c3,x} \\
\rho_4 &= I_{c2,z} + m_2 d_{c2}^2 + I_{c3,z} + m_3\left(a_2^2 + d_{c3}^2\right) \\
\rho_5 &= m_3 a_2 d_{c3} \\
\rho_6 &= I_{c3,z} + m_3 d_{c3}^2.
\end{aligned}
$$

Note that other parametrizations with the same number of coefficients can be defined, but the minimum number of parameters (six in the present case) is unique.

As a result, the inertial terms in the dynamic model can be linearly parametrized by

$$\boldsymbol{\rho} = \begin{pmatrix} \rho_1 & \rho_2 & \rho_3 & \rho_4 & \rho_5 & \rho_6 \end{pmatrix}^T$$

as

$$\boldsymbol{M}(\boldsymbol{q})\,\ddot{\boldsymbol{q}} = \boldsymbol{Y}(\boldsymbol{q},\ddot{\boldsymbol{q}})\,\boldsymbol{\rho},$$

where the 3×6 regressor matrix $\boldsymbol{Y}$ is

$$\boldsymbol{Y}(\boldsymbol{q},\ddot{\boldsymbol{q}}) = \begin{pmatrix} \ddot{q}_1 & c_2^2\,\ddot{q}_1 & c_{23}^2\,\ddot{q}_1 & 0 & 2c_2c_{23}\,\ddot{q}_1 & 0 \\ 0 & 0 & 0 & \ddot{q}_2 & c_3\,(2\ddot{q}_2+\ddot{q}_3) & \ddot{q}_3 \\ 0 & 0 & 0 & 0 & c_2\,\ddot{q}_2 & \ddot{q}_2+\ddot{q}_3 \end{pmatrix}.$$

Exercise 2

The variables $\boldsymbol{r} \in \mathbb{R}^m$ that are used to define a robot task are always functions of the configuration $\boldsymbol{q} \in \mathbb{R}^n$, i.e., $\boldsymbol{r} = \boldsymbol{f}(\boldsymbol{q})$. On the other hand, the desired behavior of the task variables (which needs to be imposed by the control law) is usually expressed by an exogenous, time-varying signal $\boldsymbol{r}_d(t)$ (a constant $\boldsymbol{r}_d$ for regulation tasks). However, the desired behavior of the primary task assigned to the robot in Fig. 1, i.e., pointing with the end effector to a (moving) target, has the peculiarity of depending also on the current robot configuration $\boldsymbol{q}(t)$, or $\boldsymbol{r}_d(t,\boldsymbol{q}(t))$. This requires some caution in the definition of the task control problem at a differential level.

The task assigned to the 4R planar robot ($n = 4$) involves the orientation angle α of the last link

$$\alpha = f_\alpha(\boldsymbol{q}) = q_1 + q_2 + q_3 + q_4,$$

as well as the position of the end effector

$$\boldsymbol{p} = \begin{pmatrix} p_x \\ p_y \end{pmatrix} = \begin{pmatrix} c_1 + c_{12} + c_{123} + c_{1234} \\ s_1 + s_{12} + s_{123} + s_{1234} \end{pmatrix} = \boldsymbol{f}_p(\boldsymbol{q}).$$

Accordingly, we have at the differential level

$$\dot{\alpha} = \frac{\partial f_\alpha}{\partial \boldsymbol{q}}\,\dot{\boldsymbol{q}} = \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}\dot{\boldsymbol{q}} = \boldsymbol{J}_\alpha\,\dot{\boldsymbol{q}} \tag{5}$$

and

$$\dot{\boldsymbol{p}} = \frac{\partial \boldsymbol{f}_p}{\partial \boldsymbol{q}}\,\dot{\boldsymbol{q}} = \boldsymbol{J}_p(\boldsymbol{q})\,\dot{\boldsymbol{q}}, \tag{6}$$

with

$$\boldsymbol{J}_p(\boldsymbol{q}) = \begin{pmatrix} -(s_1+s_{12}+s_{123}+s_{1234}) & -(s_{12}+s_{123}+s_{1234}) & -(s_{123}+s_{1234}) & -s_{1234} \\ c_1+c_{12}+c_{123}+c_{1234} & c_{12}+c_{123}+c_{1234} & c_{123}+c_{1234} & c_{1234} \end{pmatrix}.$$

The two Jacobians $\boldsymbol{J}_\alpha$ and $\boldsymbol{J}_p$ will be needed in the following.

With reference to Fig. 3, the pointing task has dimension $m = 1$ and the desired behavior for the task variable $r = \alpha$ is expressed as

$$\alpha_d(t,\boldsymbol{q}(t)) = \operatorname{atan2}\left\{p_{ty}(t) - p_y(\boldsymbol{q}(t)), p_{tx}(t) - p_x(\boldsymbol{q}(t))\right\},$$

where $\boldsymbol{p}_d(t) = (p_{tx}(t), p_{ty}(t))$ is the current position of the target. It is apparent that this scalar function depends both on the target motion (which is the exogenous part) and on the robot configuration $\boldsymbol{q}$. We need then to define the differential mapping between $\dot{\boldsymbol{q}}$ and $\dot{\alpha}_d$.

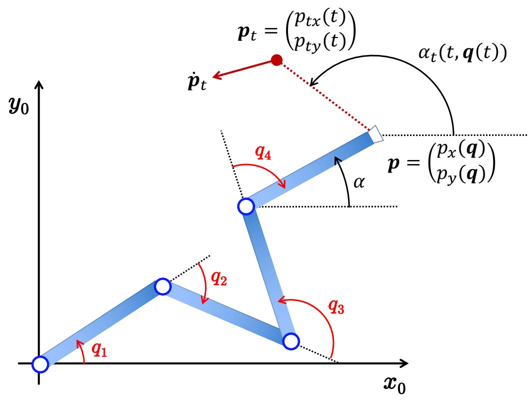

Figure 3: The task variable α and its desired value α_d at a given instant.

For this, remember that for a generic scalar function $g(z)$

$$\frac{d}{dz} \arctan g(z) = \frac{1}{1 + g^2(z)} \frac{dg(z)}{dz}$$

and note that the two functions $\operatorname{atan2}\{y, x\}$ and $\arctan(y/x)$ behave in the same way at the differential level. After some lengthy but straightforward computations, we obtain

$$\dot{\alpha}_d = \frac{(\dot{p}_{ty} - \dot{p}_y)(p_{tx} - p_x) - (\dot{p}_{tx} - \dot{p}_x)(p_{ty} - p_y)}{(p_{tx} - p_x)^2 + (p_{ty} - p_y)^2}.$$

This can be reorganized as

$$\begin{aligned} \dot{\alpha}_d &= \begin{pmatrix} \dfrac{p_{ty} - p_y}{(p_{tx} - p_x)^2 + (p_{ty} - p_y)^2} & -\dfrac{p_{tx} - p_x}{(p_{tx} - p_x)^2 + (p_{ty} - p_y)^2} \end{pmatrix} \begin{pmatrix} \dot{p}_x - \dot{p}_{tx} \\ \dot{p}_y - \dot{p}_{ty} \end{pmatrix} \\ &= \boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \left(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_t\right) = \boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \boldsymbol{J}_p(\boldsymbol{q}) \dot{\boldsymbol{q}} - \boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \dot{\boldsymbol{p}}_t, \end{aligned} \tag{7}$$

where we used the Jacobian in (6). Note that in order to evaluate the 1×2 Jacobian $\boldsymbol{J}_{\alpha_d}$, we need to know in general also the current position $\boldsymbol{p}_t$ of the target.

The task equation corresponding to the desired condition (i.e., with the robot end effector always pointing at the target) is then $\alpha(t) = \alpha_d(t)$, for all $t \geq 0$. In the nominal case, this equality holds true at the initial time $t = 0$ and thus it can be replaced by the identity $\dot{\alpha}(t) = \dot{\alpha}_d(t)$ at the differential level or, using also the Jacobian in (5),

$$\boldsymbol{J}_\alpha \, \dot{\boldsymbol{q}} = \boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \boldsymbol{J}_p(\boldsymbol{q}) \dot{\boldsymbol{q}} - \boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \, \dot{\boldsymbol{p}}_t,$$

which can be reorganized as

$$\boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \, \dot{\boldsymbol{p}}_t = \left(\boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \boldsymbol{J}_p(\boldsymbol{q}) - \boldsymbol{J}_\alpha\right) \dot{\boldsymbol{q}} = \boldsymbol{J}_r(\boldsymbol{q}) \dot{\boldsymbol{q}} \tag{8}$$

with the 1×4 task matrix $\boldsymbol{J}_r$. This relation also shows that if the target is not moving ($\dot{\boldsymbol{p}}_t = \boldsymbol{0}$), any robot velocity $\dot{\boldsymbol{q}}$ that keeps task satisfaction would have to be in the null space of $\boldsymbol{J}_r$ (and not necessarily also in the null space of $\boldsymbol{J}_p$, i.e., without a change of the end effector position!).

The minimum norm solution to (8) is given by

$$\dot{\boldsymbol{q}}_0 = \boldsymbol{J}_r^{\#}(\boldsymbol{q}) \boldsymbol{J}_{\alpha_d}(\boldsymbol{q}) \, \dot{\boldsymbol{p}}_t,$$

whereas the simultaneous minimization of an objective function $H(\boldsymbol{q})$ leads to the PG method

$$\begin{aligned}\dot{\boldsymbol{q}}_H &= \boldsymbol{J}_r^{\#}(\boldsymbol{q})\boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\,\dot{\boldsymbol{p}}_t - \big(\boldsymbol{I} - \boldsymbol{J}_r^{\#}(\boldsymbol{q})\boldsymbol{J}_r(\boldsymbol{q})\big)\beta\,\nabla H(\boldsymbol{q}) \\ &= -\beta\,\nabla H(\boldsymbol{q}) + \boldsymbol{J}_r^{\#}(\boldsymbol{q})\,\big(\boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\,\dot{\boldsymbol{p}}_t + \boldsymbol{J}_r(\boldsymbol{q})\,\beta\,\nabla H(\boldsymbol{q})\big),\end{aligned} \tag{9}$$

for some stepsize $\beta > 0$ in the direction of the negative gradient of H.

When there is a task error $e_r = \alpha_d - \alpha \neq 0$, a feedback action should be incorporated in the joint velocity command (9) as

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_r^{\#}(\boldsymbol{q})\,(\boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\,\dot{\boldsymbol{p}}_t - k e_r) - \big(\boldsymbol{I} - \boldsymbol{J}_r^{\#}(\boldsymbol{q})\boldsymbol{J}_r(\boldsymbol{q})\big)\,\beta\,\nabla H(\boldsymbol{q}), \tag{10}$$

for some scalar control gain $k > 0$. When the matrix $\boldsymbol{J}_r$ has full rank, i.e., when its single row does not vanish, then $\boldsymbol{J}_r\,\boldsymbol{J}_r^{\#} = 1$ and using (10) it follows that

$$\begin{aligned}\dot{e}_r = \dot{\alpha}_d - \dot{\alpha} &= \boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\boldsymbol{J}_p(\boldsymbol{q})\dot{\boldsymbol{q}} - \boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\dot{\boldsymbol{p}}_t - \boldsymbol{J}_\alpha\dot{\boldsymbol{q}} = \boldsymbol{J}_r(\boldsymbol{q})\dot{\boldsymbol{q}} - \boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\dot{\boldsymbol{p}}_t \\ &= \boldsymbol{J}_r(\boldsymbol{q})\big(\boldsymbol{J}_r^{\#}(\boldsymbol{q})\,(\boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\dot{\boldsymbol{p}}_t - k e_r) - \big(\boldsymbol{I} - \boldsymbol{J}_r^{\#}(\boldsymbol{q})\,\boldsymbol{J}_r(\boldsymbol{q})\big)\,\beta\,\nabla H(\boldsymbol{q})\big) - \boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\dot{\boldsymbol{p}}_t \\ &= \boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\dot{\boldsymbol{p}}_t - k e_r - \boldsymbol{J}_{\alpha_d}(\boldsymbol{q})\dot{\boldsymbol{p}}_t \\ &= -k e_r,\end{aligned}$$

showing exponential recovery of the task error.

In order to keep the joints close to the middle values $\bar{q}_i = (q_{m,i} + q_{M,i})/2$ of their ranges $[q_{m,i}, q_{M,i}]$, for $i = 1, \ldots, 4$, the following objective function (in our case, with $N = 4$)

$$H(\boldsymbol{q}) = \frac{1}{2N}\sum_{i=1}^{N}\left(\frac{q_i - \bar{q}_i}{q_{M,i} - q_{m,i}}\right)^2$$

will be minimized in the null space of the primary task. The gradient of H is

$$\nabla H(\boldsymbol{q}) = \left(\frac{\partial H}{\partial \boldsymbol{q}}\right)^T = \frac{1}{N}\begin{pmatrix}\dfrac{q_1 - \bar{q}_1}{(q_{M,1} - q_{m,1})^2} \\ \vdots \\ \dfrac{q_N - \bar{q}_N}{(q_{M,N} - q_{m,N})^2}\end{pmatrix}.$$

With the robot in the configuration $\boldsymbol{q} = (0, \pi/2, 0, -\pi/4)$, we can evaluate a number of terms useful for the control expressions (9) or (10):

$$\boldsymbol{p} = \begin{pmatrix}1.7071 \\ 2.7071\end{pmatrix} \qquad \boldsymbol{J}_p = \begin{pmatrix}-2.7071 & -2.7071 & -1.7071 & -0.7071 \\ 1.7071 & 0.7071 & 0.7071 & 0.7071\end{pmatrix} \qquad \alpha = \pi/4.$$

Moreover, using the given joint limits we have

$$H = 0.0625 \qquad \nabla H = \begin{pmatrix}0 \\ 0.0796 \\ 0 \\ -0.0796\end{pmatrix}.$$

Consider first the nominal initial condition in which the end effector is correctly pointing at the target (see Fig. 4). In this case, the actual position $\boldsymbol{p}_t$ of the target is not specified in the problem

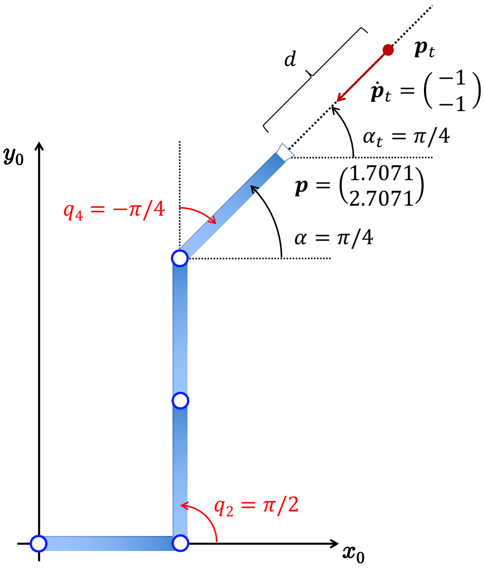

Figure 4: Nominal initial condition for the task, with the robot end effector pointing at the target. In addition, the target moves along the joining direction with $\dot{\boldsymbol{p}}_t = (-1, -1)$ [m/s].

and, as already mentioned, we cannot evaluate the Jacobian $\boldsymbol{J}_{\alpha_d}$ nor the task matrix $\boldsymbol{J}_r$. However, we are in a very special situation since the velocity $\dot{\boldsymbol{p}}_t$ will keep the target along the direction joining it to the robot end effector. Analytically, the target position is on the half-line

$$\boldsymbol{p}_t = \boldsymbol{p} + d \begin{pmatrix} \cos\alpha \\ \sin\alpha \end{pmatrix} = \begin{pmatrix} 1.7071 \\ 2.7071 \end{pmatrix} + d \begin{pmatrix} 0.7071 \\ 0.7071 \end{pmatrix} \qquad d > 0,$$

at a (unknown) distance d from the robot end effector. Since $\sin\alpha = \cos\alpha = 0.7071$ and the velocity of the target is $\dot{\boldsymbol{p}}_t = (-1, -1)$ [m/s], we have from (7)

$$\boldsymbol{J}_{\alpha_d} \dot{\boldsymbol{p}}_t = \begin{pmatrix} \dfrac{\sin\alpha}{d} & -\dfrac{\cos\alpha}{d} \end{pmatrix} \begin{pmatrix} -1 \\ -1 \end{pmatrix} = 0.$$

Moreover

$$\begin{aligned} \boldsymbol{J}_r = \boldsymbol{J}_{\alpha_d} \boldsymbol{J}_p - \boldsymbol{J}_\alpha &= \frac{0.7071}{d} \begin{pmatrix} -4.4142 & -3.4142 & -2.4142 & -1.4142 \end{pmatrix} - \begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix} \\ &= \frac{1}{d} \begin{pmatrix} -(3.2113 + d) & -(2.4142 + d) & -(1.7071 + d) & -(1 + d) \end{pmatrix}. \end{aligned}$$

Without pursuing further the numerical computation, note that since $\boldsymbol{J}_r$ is just a row, its pseudo-inverse is $\boldsymbol{J}_r^{\#} = \boldsymbol{J}_r^T / \| \boldsymbol{J}_r^T \|^2$. The solution (9) takes then the form

$$\dot{\boldsymbol{q}}_H = -\beta \left(\boldsymbol{I} - \frac{1}{\| \boldsymbol{J}_r^T \|^2} \boldsymbol{J}_r^T \boldsymbol{J}_r \right) \nabla H.$$

The distance $d > 0$, which is embedded in the product $\boldsymbol{J}_r^{\#} \boldsymbol{J}_r$, plays a limited role in the solution. On the other hand, the larger is $\beta > 0$, the more the joints will try to get close to their midrange.

In the second situation, i.e., when there is an initial pointing error $e_r \neq 0$, the target position is given and thus all numerical data are available for evaluating the control law (10). In fact, we have

$$\alpha_d = -3.0209 \qquad e_r = -3.8063 \qquad \text{[rad]}$$

and

$$\boldsymbol{J}_r = \begin{pmatrix} 0.1751 & -0.4022 & -0.4722 & -0.5423 \end{pmatrix}.$$

Choosing for instance $k = 0.2$ and $\beta = 20$ in eq. (10) yields finally

$$\dot{\boldsymbol{q}}_H = \begin{pmatrix} 0.3681 & -2.4370 & -0.9927 & 0.4516 \end{pmatrix}^T \qquad [\text{rad/s}].$$

In the chosen configuration, the second joint is at its upper limit, i,e., $q_2 = q_{M,2} = \pi/2$, while the fourth joint is at its lower limit, i.e., $q_4 = q_{m,4} = -\pi/4$; therefore, in order to remain within the feasible range, the velocity of the second and fourth joints should be respectively $\dot{q}_2 \leq 0$ (remain at rest, or rotate clockwise) and $\dot{q}_4 \geq 0$ (remain at rest, or rotate counterclockwise), which is what happens with the solution $\dot{\boldsymbol{q}}_H$. Note however that this is obtained by fine tuning the gain k and (especially) the stepsize β in the control law (10), since the joint range limits are not considered as hard constraints in the problem but only within the (soft) objective H.

Exercise 3

Consider the closed-loop equations (2), (3) with the control law (4)

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{S}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta}) = \boldsymbol{0} \tag{11}$$

$$\boldsymbol{B}_m\ddot{\boldsymbol{\theta}} + \boldsymbol{K}(\boldsymbol{\theta}-\boldsymbol{q}) = \boldsymbol{K}_P(\boldsymbol{\theta}_d-\boldsymbol{\theta}) - \boldsymbol{K}_D\dot{\boldsymbol{\theta}}, \tag{12}$$

where we replaced $\boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{S}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ using any factorization of the Coriolis and centrifugal terms. It is easy to see that $\boldsymbol{x} = (\boldsymbol{q},\boldsymbol{\theta},\dot{\boldsymbol{q}},\dot{\boldsymbol{\theta}}) = (\boldsymbol{\theta}_d,\boldsymbol{\theta}_d,\boldsymbol{0},\boldsymbol{0}) = \boldsymbol{x}_e$ is the unique equilibrium state of such a controlled robot with elastic joints (in the absence of gravity).

To show that this is a globally asymptotically stable equilibrium, define the energy-based Lyapunov candidate

$$V = \frac{1}{2}\dot{\boldsymbol{q}}^T\boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} + \frac{1}{2}\dot{\boldsymbol{\theta}}^T\boldsymbol{B}_m\dot{\boldsymbol{\theta}} + \frac{1}{2}(\boldsymbol{q}-\boldsymbol{\theta})^T\boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta}) + \frac{1}{2}(\boldsymbol{\theta}_d-\boldsymbol{\theta})^T\boldsymbol{K}_P(\boldsymbol{\theta}_d-\boldsymbol{\theta}),$$

which contains the kinetic energy of the links and the motors, the potential energy due to the elasticity of the joints (quadratic in the joint deformation $\boldsymbol{\delta} = \boldsymbol{q}-\boldsymbol{\theta}$) , and a virtual potential energy introduced by the control (in terms of the motor position error, with $\boldsymbol{K}_P > 0$). For this function, it is $V \geq 0$ for all $\boldsymbol{x}$ and $V = 0$ if and only if $\boldsymbol{x} = \boldsymbol{x}_e$.

The time derivative of V, evaluated along the trajectories of the closed-loop system, is

$$\begin{aligned}
\dot{V} &= \dot{\boldsymbol{q}}^T\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \frac{1}{2}\dot{\boldsymbol{q}}^T\dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \dot{\boldsymbol{\theta}}^T\boldsymbol{B}_m\ddot{\boldsymbol{\theta}} + (\dot{\boldsymbol{q}}-\dot{\boldsymbol{\theta}})^T\boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta}) - \dot{\boldsymbol{\theta}}^T\boldsymbol{K}_P(\boldsymbol{\theta}_d-\boldsymbol{\theta}) \\
&= -\dot{\boldsymbol{q}}^T\left(\boldsymbol{S}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta})\right) + \frac{1}{2}\dot{\boldsymbol{q}}^T\dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \dot{\boldsymbol{\theta}}^T\left(\boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta}) + \boldsymbol{K}_P(\boldsymbol{\theta}_d-\boldsymbol{\theta}) - \boldsymbol{K}_D\dot{\boldsymbol{\theta}}\right) \\
&\quad + \dot{\boldsymbol{q}}^T\boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta}) - \dot{\boldsymbol{\theta}}^T\boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta}) - \dot{\boldsymbol{\theta}}^T\boldsymbol{K}_P(\boldsymbol{\theta}_d-\boldsymbol{\theta}) - \dot{\boldsymbol{\theta}}^T\boldsymbol{K}_D\dot{\boldsymbol{\theta}} \\
&= -\dot{\boldsymbol{\theta}}^T\boldsymbol{K}_D\dot{\boldsymbol{\theta}} \leq 0,
\end{aligned}$$

where we used the identity

$$\dot{\boldsymbol{q}}^T\left(\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\boldsymbol{S}(\boldsymbol{q},\dot{\boldsymbol{q}})\right)\dot{\boldsymbol{q}} = 0, \qquad \text{for all } \boldsymbol{q},\dot{\boldsymbol{q}}.$$

Thus, the closed-loop system is certainly stable.

To conclude about asymptotic stability, we use LaSalle theorem. Since

$$\dot{V} = 0 \qquad \Longleftrightarrow \qquad \dot{\boldsymbol{\theta}} = \boldsymbol{0},$$

we analyze the closed-loop equations for $\dot{\boldsymbol{\theta}} \equiv \mathbf{0}$. From eq. (12), since $\ddot{\boldsymbol{\theta}}$ must also vanish in order for any set contained in $\mathcal{S} = \{\boldsymbol{x} : \dot{V} = 0\}$ to be invariant, we have

$$\boldsymbol{K}\,(\boldsymbol{\theta} - \boldsymbol{q}) = \boldsymbol{K}_P\,(\boldsymbol{\theta}_d - \boldsymbol{\theta}) = \text{constant}. \tag{13}$$

Being $\boldsymbol{\theta}$ constant itself, this equation implies that also $\boldsymbol{q}$ must remain constant. Therefore, since $\dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{q}}$ must also vanish, from eq. (11) it follows that $\boldsymbol{K}(\boldsymbol{q}-\boldsymbol{\theta}) = \mathbf{0}$, and then $\boldsymbol{q} = \boldsymbol{\theta}$. Substituting this in (13) leads to $\boldsymbol{\theta} = \boldsymbol{\theta}_d$ (thus, the constant therein is necessarily zero). Summarizing, $\boldsymbol{q} = \boldsymbol{\theta} = \boldsymbol{\theta}_d$ and the maximal set of invariant states contained in $\mathcal{S}$ reduces to the singleton $\boldsymbol{x}_e = (\boldsymbol{\theta}_d, \boldsymbol{\theta}_d, \mathbf{0}, \mathbf{0})$, which is then a global, asymptotically stable equilibrium. This concludes the proof.

Exercise 4

The method of residuals for fault detection is based on terms appearing in the robot dynamics. We need then to derive first the dynamic model of the PR robot. Using the variables $\boldsymbol{q} = (q_1, q_2)$ defined in Fig. 2, we have in particular

$$\boldsymbol{p}_{c2} = \begin{pmatrix} d_{c2}c_2 \\ 0 \\ q_1 - d_{c2}s_2 \end{pmatrix} \quad \Rightarrow \quad \boldsymbol{v}_{c2} = \begin{pmatrix} -d_{c2}s_2\,\dot{q}_2 \\ 0 \\ \dot{q}_1 - d_{c2}c_2\,\dot{q}_2 \end{pmatrix} \quad \Rightarrow \quad \|\boldsymbol{v}_{c2}\|^2 = \dot{q}_1^2 + d_{c2}^2\,\dot{q}_2^2 - 2\,d_{c2}c_2\,\dot{q}_1\dot{q}_2.$$

Therefore, the kinetic and potential energies of the two links are

$$T_1 = \frac{1}{2}\,m_1\dot{q}_1^2 \qquad T_2 = \frac{1}{2}\,I_{c2}\,\dot{q}_2^2 + \frac{1}{2}\,m_2\left(\dot{q}_1^2 + d_{c2}^2\,\dot{q}_2^2 - 2\,d_{c2}c_2\,\dot{q}_1\dot{q}_2\right) \qquad \Rightarrow \qquad T = T_1 + T_2$$

and

$$U_1 = -m_1 g_0\,q_1 \qquad U_2 = -m_2 g_0\,(q_1 - d_{c2}s_2) \qquad \Rightarrow \qquad U = U_1 + U_2.$$

with $g_0 = 9.81$ m/s^2.

The dynamic model of the robot, assuming the possible presence of a fault $u_{f1}(t)$ on the first actuator, is

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{u} - \boldsymbol{u}_f. \tag{14}$$

with inertia matrix

$$\boldsymbol{M}(\boldsymbol{q}) = \begin{pmatrix} m_1 + m_2 & -m_2 d_{c2}\,c_2 \\ -m_2 d_{c2}\,c_2 & I_{c2} + m_2 d_{c2}^2 \end{pmatrix},$$

velocity terms (as computed from Christoffel symbols)

$$\boldsymbol{c}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \begin{pmatrix} m_2 d_{c2} s_2\,\dot{q}_2^2 \\ 0 \end{pmatrix},$$

gravity vector

$$\boldsymbol{g}(\boldsymbol{q}) = \frac{\partial U}{\partial \boldsymbol{q}} = g_0 \begin{pmatrix} -(m_1 + m_2) \\ m_2 d_{c2} c_2 \end{pmatrix},$$

and fault vector

$$\boldsymbol{u}_f = \begin{pmatrix} u_{f1} \\ 0 \end{pmatrix}.$$

The expression of the scalar residual for a fault on the first actuator[1] is

$$r_1(t) = k\left(\int_0^t \left(u_1 - \alpha_1(\boldsymbol{q},\dot{\boldsymbol{q}}) - r_1\right)d\tau - p_1(\boldsymbol{q},\dot{\boldsymbol{q}})\right) \qquad r_1(0) = 0, \tag{15}$$

[1] It is assumed that the robot starts at rest at $t = 0$, i.e., $\dot{\boldsymbol{q}}(0) = \mathbf{0}$.

where $k > 0$, u_1 is the commanded force for the actuator at the first prismatic joint, the function α_1 is defined as

$$\alpha_1(\boldsymbol{q}, \dot{\boldsymbol{q}}) = -\frac{1}{2}\,\dot{\boldsymbol{q}}^T\,\frac{\partial \boldsymbol{M}}{\partial q_1}\,\dot{\boldsymbol{q}} + g_1(\boldsymbol{q}),$$

and $p_1 = \boldsymbol{m}_1^T(\boldsymbol{q})\dot{\boldsymbol{q}}$ is the first component of the generalized momentum $\boldsymbol{p} = \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}}$, being $\boldsymbol{m}_1$ the first column of the inertia matrix. Since q_1 never appears in the inertia matrix, one has simply

$$\alpha_1 = -(m_1 + m_2)\, g_0,$$

while

$$p_1 = (m_1 + m_2)\,\dot{q}_1 - m_2 d_{c2}\, c_2\, \dot{q}_2.$$

Substituting these expressions in (15) gives finally

$$r_1(t) = k_1 \left(\int_0^t (u_1 + (m_1 + m_2)\, g_0 - r_1)\, d\tau + m_2 d_{c2}\, c_2\, \dot{q}_2 - (m_1 + m_2)\, \dot{q}_1 \right) \qquad r_1(0) = 0. \tag{16}$$

The theory says that the evolution of $r_1(t)$ is governed by $\dot{r}_1 = k_1\,(u_{f1} - r_1)$, allowing detection of the fault, whenever present, through the response of a first-order filter with time constant $1/k_1$ excited by the unknown input signal u_{f1}. One can also verify this property by differentiating (16) and using the model terms in (14). In fact

$$\begin{aligned}
\dot{r}_1 &= k_1 \big((u_1 + (m_1 + m_2)\, g_0 - r_1) + m_2 d_{c2}\, c_2\, \ddot{q}_2 - m_2 d_{c2}\, s_2\, \dot{q}_2^2 - (m_1 + m_2)\, \ddot{q}_1 \big) \\
&= k_1\, ((u_1 + (m_1 + m_2)\, g_0 - r_1) - (u_1 - u_{f1} + (m_1 + m_2)\, g_0)) \\
&= k_1\, (u_{f1} - r_1)\,.
\end{aligned}$$

If the fault is $u_{f1}(t) = 2$, for $t \geq 0$, the solution trajectory $r_1(t)$ and its steady-state value are

$$r_1(t) = 2(1 - \exp(-k_1 t)) \qquad \Rightarrow \qquad r_{1,ss} = \lim_{t \to \infty} r_1(t) = 2.$$

* * * * *