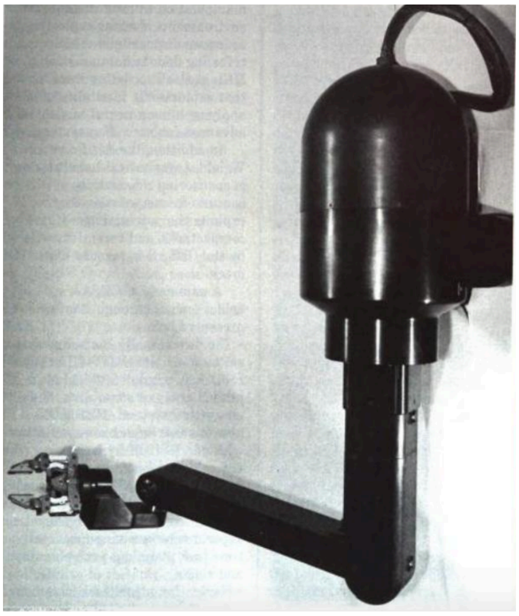
Robotics I

January 7, 2020

Exercise 1

Consider the 7R CESAR research manipulator in Fig. 1, developed at the Oak Ridge National Laboratories, USA. The robot has a spherical wrist, and the ordered sequence of the last three axes is pitch, yaw, and roll (see the naming of joint axes in the figure). Assume that the upper arm roll axis intersects the elbow pitch axis (we neglect here a small existing offset). The geometric dimensions are: upper arm length = 0.635; lower arm length = 0.508; shoulder offset = 0.356; distance from the wrist center to the center of the end-effector gripper jaws = 0.343 (all in [m]). Determine a frame assignment and the associated table of parameters following the Denavit-Hartenberg (DH) convention, and assign the geometric data to the corresponding constant DH parameters. Place the first DH frame so that $a_1 = d_1 = 0$, and the last frame with the origin O_7 at the center of the gripper jaws and with the axis $\boldsymbol{z}_7$ in the approach direction. Use the provided Extra Sheet #1 and return it, with your name added.



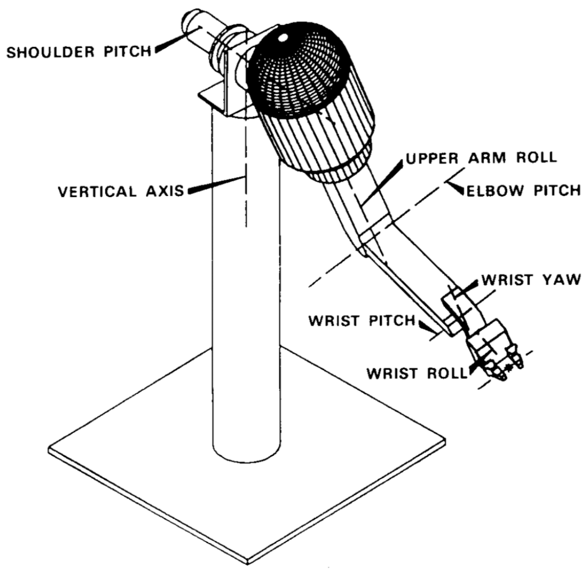


Figure 1: The 7R CESAR manipulator and its drawing with the names of the joint axes.

Exercise 2

For the CESAR manipulator of Exercise 1, determine the Jacobian matrix $\boldsymbol{J}_{L,w}(\boldsymbol{q})$ that relates the joint velocity $\dot{\boldsymbol{q}} \in \mathbb{R}^7$ to the linear velocity $\boldsymbol{v}_w \in \mathbb{R}^3$ of the wrist center. Note that the expression of this Jacobian becomes simpler when expressed in frame 2 or 3 (i.e., ${}^2\boldsymbol{J}_{L,w}$ or ${}^3\boldsymbol{J}_{L,w}$). Sketch the elements used by an iterative numerical scheme based on the Gradient method used to solve the inverse kinematics problem for a given desired position $\boldsymbol{p}_w \in \mathbb{R}^3$ of the center of the wrist, *without* considering unnecessary joint variables. In the generic case, how many inverse kinematic solutions are there for this problem?

Exercise 3

Consider the 3-dof, planar RPR robot in Fig. 2, with an associated base frame RF_0. The robot end-effector should move in contact with the shown surface from point $\boldsymbol{A} = (5.2, 1.5)$ to point $\boldsymbol{B} = (2.2, -2.5)$ (both in [m]). Moreover, the orientation of the end-effector with respect to the normal $\boldsymbol{n}$ to the surface should change continuously from the initial angle $\alpha_A = -60°$ to the final angle $\alpha_B = 30°$.

a) For the (Cartesian) task variables $\boldsymbol{r} = [\boldsymbol{p}^T\ \alpha]^T \in \mathbb{R}^3$, provide a spatial description $\boldsymbol{r} = \boldsymbol{r}(s)$ of the defined task in terms of a normalized parameter $s \in [0, 1]$.

b) At time $t = 0$, the robot end-effector is in point $\boldsymbol{A}$, with the correct orientation α_A, and its initial non-zero velocity is *consistent* with the execution of the desired task, with a linear speed $V = 2.5$ [m/s] and an angular speed $\Omega = 45$ [°/s]. From this state, plan a state-to-rest coordinated Cartesian motion that will complete the task in a given time $T = 2$, with continuity up to the acceleration (including at the two ends of the path).

c) What will be the value of the task velocity $\dot{\boldsymbol{r}}(t)$ at the half-time $t = T/2$?

d) Associate next the DH variables to the RPR robot, and assume that the prismatic joint range is limited to non-negative values of q_2 and that the third link has length $L = 1$ [m]. Show that the parametrized Cartesian task implies also a unique parametrized path in the robot joint space, and provide the analytic expression of $\boldsymbol{q} = \boldsymbol{q}(s)$.

e) What will be the value of the joint velocity $\dot{\boldsymbol{q}}(t)$ at the half-time $t = T/2$?

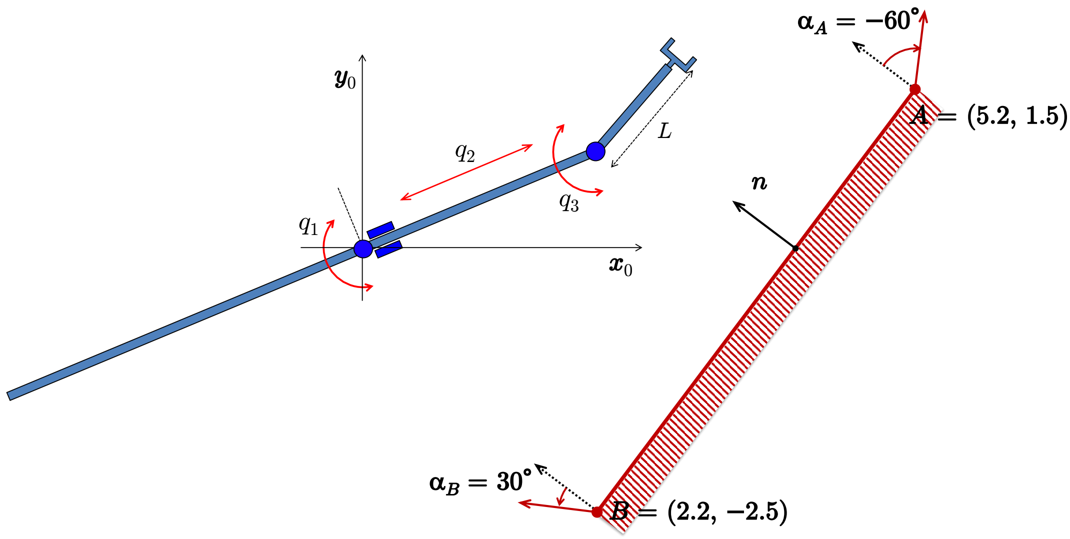


Figure 2: The planar RPR robot and the Cartesian task to be executed.

Exercise 4

A number of questions and statements are reported on the Extra Sheet #2. Fill in your answers and/or comments on the same sheet, providing also a short motivation/explanation for each item. Add your name on the sheet and return it.

[240 minutes, open books]

Solution

January 7, 2020

Exercise 1

A possible Denavit-Hartenberg frame assignment for the 7R CESAR manipulator is shown in Fig. 3, with the associated parameters reported in Tab. 1. In the table, a zero (with an asterisk) is set for a_3, according to the assumption about neglecting the small offset at the elbow joint. In the real robot, it is $a_3 = 0.029$ [m].

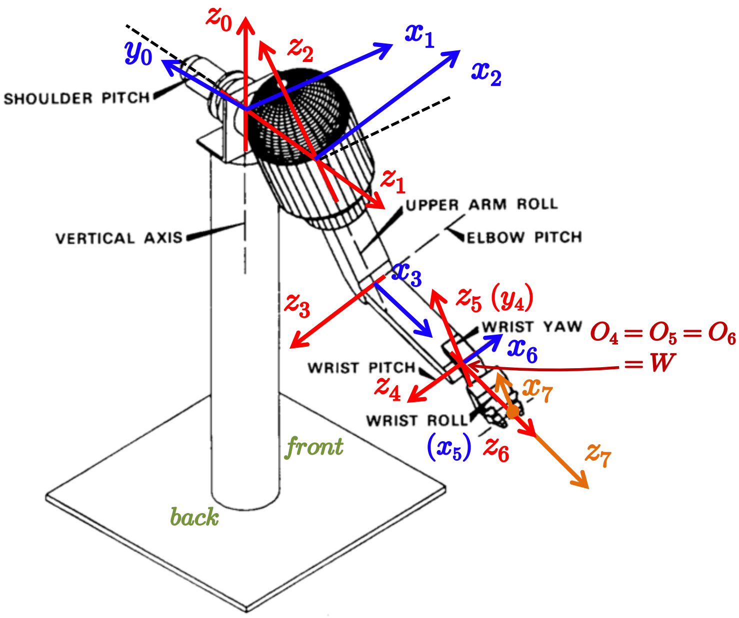


Figure 3: Assignment of the DH frames for the 7R CESAR manipulator (view from the back).

i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	0	q_1
2	$-\pi/2$	0	$d_2 = 0.356$	q_2
3	$\pi/2$	0^*	$d_3 = -0.635$	q_3
4	0	$a_4 = 0.508$	0	q_4
5	$-\pi/2$	0	0	q_5
6	$\pi/2$	0	0	q_6
7	0	0	$d_7 = 0.343$	q_7

Table 1: The DH table of parameters for the 7R CESAR manipulator corresponding to Fig. 3.

The manipulator is viewed from the front side (and with the same frame assignment) in Fig. 4, which is taken from the original paper:

[1] R.V. Dubey, J.A. Euler, and S.M. Babcock, "Real-time implementation of an optimization scheme for seven-degree-of-freedom redundant manipulators," *IEEE Trans. on Robotics and Automation*, vol. 7, no. 5, pp. 579–588, 1991.

Therein, also the non-zero parameters a_i and d_i are shown. The robot is in a slightly different configuration in the two figures: in Fig. 4, it is $\boldsymbol{q} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & \pi/2 & \pi/2 \end{pmatrix}^T$ [rad], whereas q_1 is slightly negative, q_2 is positive, and $q_3 \simeq -\pi/2$ in Fig. 3.

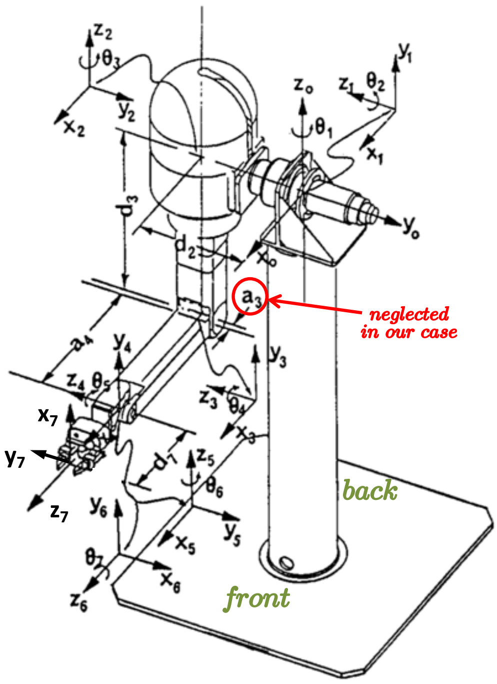


Figure 4: Front view of the DH assignment for the 7R CESAR manipulator (modified from [1]).

Exercise 2

Based on Tab. 1, in order to determine the position $\boldsymbol{p}_w$ of the center of the spherical wrist, i.e., the position of the origin O_4, we need only the following DH homogenous transformation matrices:

$$
{}^0\boldsymbol{A}_1(q_1) = \begin{pmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} {}^0\boldsymbol{R}_1(q_1) & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 \end{pmatrix}, \quad {}^1\boldsymbol{A}_2(q_2) = \begin{pmatrix} c_2 & 0 & -s_2 & 0 \\ s_2 & 0 & c_2 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} {}^1\boldsymbol{R}_2(q_2) & {}^1\boldsymbol{p}_{12} \\ \boldsymbol{0}^T & 1 \end{pmatrix},
$$

$$
{}^2\boldsymbol{A}_3(q_3) = \begin{pmatrix} c_3 & 0 & s_3 & 0 \\ s_3 & 0 & -c_3 & 0 \\ 0 & 1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \qquad {}^3\boldsymbol{A}_4(q_4) = \begin{pmatrix} c_4 & -s_4 & 0 & a_4 c_4 \\ s_4 & c_4 & 0 & a_4 s_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},
$$

where the shorthand notations $s_i = \sin q_i$, $c_i = \cos q_i$ have been used.

The position $\boldsymbol{p}_w$ is computed from

$$\boldsymbol{p}_{w,H} = \begin{pmatrix} \boldsymbol{p}_w \\ 1 \end{pmatrix} = {}^0\boldsymbol{A}_1(q_1)\left({}^1\boldsymbol{A}_2(q_2)\left({}^2\boldsymbol{A}_3(q_3)\left({}^3\boldsymbol{A}_4(q_4)\begin{pmatrix} \boldsymbol{0} \\ 1 \end{pmatrix}\right)\right)\right),$$

giving

$$\boldsymbol{p}_w(\boldsymbol{q}) = \begin{pmatrix} d_2 s_1 - d_3 c_1 s_2 - a_4 c_1 s_2 s_4 - a_4 c_4 (s_1 s_3 - c_1 c_2 c_3) \\ -d_2 c_1 - d_3 s_1 s_2 - a_4 s_1 s_2 s_4 + a_4 c_4 (c_1 s_3 + s_1 c_2 c_3) \\ d_3 c_2 + a_4 c_2 s_4 + a_4 s_2 c_3 c_4 \end{pmatrix}.$$

The 3×7 Jacobian matrix $\boldsymbol{J}_{L,w}$ in $\boldsymbol{v}_w = \dot{\boldsymbol{p}}_w = \boldsymbol{J}_{L,w}(\boldsymbol{q})\dot{\boldsymbol{q}}$ can be computed in two alternative but equivalent ways (i.e., analytically or geometrically) as

$$\boldsymbol{J}_{L,w}(\boldsymbol{q}) = \frac{\partial \boldsymbol{p}_w(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{pmatrix} \boldsymbol{z}_0 \times \boldsymbol{p}_{0,w} & \boldsymbol{z}_1 \times \boldsymbol{p}_{1,w} & \boldsymbol{z}_2 \times \boldsymbol{p}_{2,w} & \boldsymbol{z}_3 \times \boldsymbol{p}_{3,w} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{pmatrix},$$

where $\boldsymbol{p}_{0,w} = \boldsymbol{p}_w$. Since there is no actual dependence of $\boldsymbol{p}_w(\boldsymbol{q})$ on q_5, q_6, and q_7, the last three columns of $\boldsymbol{J}_{L,w}(\boldsymbol{q})$ are automatically zero and can be skipped. We will denote by $\bar{\boldsymbol{J}}_{L,w}(\boldsymbol{q})$ the resulting 3×4 Jacobian matrix. Performing computations (possibly with the Symbolic Toolbox of Matlab) one obtains

$$\bar{\boldsymbol{J}}_{L,w}(\boldsymbol{q}) = \left(\begin{matrix} d_2 c_1 + d_3 s_1 s_2 + a_4 s_1 s_2 s_4 - a_4 c_4 (c_1 s_3 + s_1 c_2 c_3) & -c_1 (d_3 c_2 + a_4 c_2 s_4 + a_4 s_2 c_3 c_4) \\ d_2 s_1 - d_3 c_1 s_2 - a_4 c_1 s_2 s_4 - a_4 c_4 (s_1 s_3 - c_1 c_2 c_3) & -s_1 (d_3 c_2 + a_4 c_2 s_4 + a_4 s_2 c_3 c_4) \\ 0 & -d_3 s_2 - a_4 s_2 s_4 + a_4 c_2 c_3 c_4 \end{matrix}\right.$$
$$\left.\begin{matrix} -a_4 c_4 (s_1 c_3 + c_1 c_2 s_3) & -a_4 c_1 s_2 c_4 + a_4 s_4 (s_1 s_3 - c_1 c_2 c_3) \\ a_4 c_4 (c_1 c_3 - s_1 c_2 s_3) & -a_4 s_1 s_2 c_4 - a_4 s_4 (c_1 s_3 + s_1 c_2 c_3) \\ -a_4 s_2 s_3 c_4 & a_4 c_2 c_4 - a_4 s_2 c_3 s_4 \end{matrix}\right).$$

By noticing the presence of some recurrent trigonometric terms, one may obtain simpler expressions of this Jacobian by expressing it in a rotated reference frame (say, RF_1, RF_2 or RF_3). Following the hint given in the text, the simplest form is in fact obtained when working in RF_2, i.e., expressing the linear velocity of center of the wrist as ${}^2\boldsymbol{v}_w = {}^2\boldsymbol{J}_{L,w}(\boldsymbol{q})\dot{\boldsymbol{q}}$. We have

$$\begin{aligned} {}^2\bar{\boldsymbol{J}}_{L,w}(\boldsymbol{q}) &= {}^1\boldsymbol{R}_2^T(q_2)\left({}^0\boldsymbol{R}_1^T(q_1)\,\bar{\boldsymbol{J}}_{L,w}(\boldsymbol{q})\right) = {}^1\boldsymbol{R}_2^T(q_2)\,{}^1\bar{\boldsymbol{J}}_{L,w}(\boldsymbol{q}) \\ &= \begin{pmatrix} c_2 (d_2 - a_4 s_3 c_4) & -d_3 - a_4 s_4 & a_4 s_3 c_4 & -a_4 c_3 s_4 \\ -d_2 s_2 - a_4 s_2 s_4 + a_4 c_2 c_3 c_4 & 0 & a_3 c_3 c_4 & -a_4 s_3 s_4 \\ -s_2 (d_2 - a_4 s_3 c_4) & a_4 c_3 c_4 & 0 & a_4 c_4 \end{pmatrix}. \end{aligned} \tag{1}$$

To find a solution to the inverse kinematics problem for a given desired position $\boldsymbol{p}_{w,d}$ of the center of the wrist, we can only resort to an iterative numerical scheme. The problem has in fact an infinite number of solutions that cannot be obtained in closed form. The Gradient method will use the transpose of the Jacobian matrix $\bar{\boldsymbol{J}}_{L,w}(\boldsymbol{q})$, possibly expressed in the simpler form (1). Moreover, the updates of the algorithm will concern only the first four joint variables $\bar{\boldsymbol{q}} = \begin{pmatrix} q_1 & q_2 & q_3 & q_4 \end{pmatrix}^T$ —the remaining ones being irrelevant. Thus, the generic iteration $k \geq 1$ of the Gradient method will be

$$\bar{\boldsymbol{q}}^{[k+1]} = \bar{\boldsymbol{q}}^{[k]} + \alpha^{[k]} \cdot {}^2\bar{\boldsymbol{J}}_{L,w}^T\left(\bar{\boldsymbol{q}}^{[k]}\right)\,{}^0\boldsymbol{R}_2^T(q_1^{[k]}, q_2^{[k]})\left(\boldsymbol{p}_{w,d} - \boldsymbol{p}_w\left(\bar{\boldsymbol{q}}^{[k]}\right)\right), \tag{2}$$

with a stepsize $\alpha^{[k]} > 0$ possibly varying over iterations. Note the added rotation matrix needed to express the Cartesian position error $\boldsymbol{e}^{[k]} = \boldsymbol{p}_{w,d} - \boldsymbol{p}_w^{[k]}$ at iteration k in the frame RF_2 as ${}^2\boldsymbol{e}^{[k]}$.

Exercise 3

The first part of the problem, items a) to c), is concerned only with the specification of the task, independently from the robot that has to execute it (in this case a RPR robot).

The desired planar task is three-dimensional, involving the motion in position $\boldsymbol{p}(t) \in \mathbb{R}^2$ along the linear surface from $\boldsymbol{A}$ to $\boldsymbol{B}$ and the simultaneous change of orientation $\alpha(t) \in \mathbb{R}$ from α_A to α_B, with the angle $\alpha(t)$ being defined w.r.t. the constant normal $\boldsymbol{n}$ to the surface as in Fig. 2. We approach the problem by decomposing the definition of the task trajectory in (normalized) space and time, i.e.,

$$\boldsymbol{r}(t) = \begin{pmatrix} \boldsymbol{p}(t) \\ \alpha(t) \end{pmatrix}, \text{ for } t \in [0,T] \quad \Rightarrow \quad \boldsymbol{r}(s) = \begin{pmatrix} \boldsymbol{p}(s) \\ \alpha(s) \end{pmatrix}, \text{ with } s \in [0,1], \quad s = s(t), \text{ for } t \in [0,T].$$

The simplest parametrization of the desired task is through the linear expressions

$$\boldsymbol{p}(s) = \boldsymbol{A} + (\boldsymbol{B} - \boldsymbol{A})\, s \in \mathbb{R}^2, \qquad \alpha(s) = \alpha_A + (\alpha_B - \alpha_A)\, s \in \mathbb{R}, \qquad \text{with } s \in [0,1], \tag{3}$$

or, replacing numerical values,

$$\boldsymbol{r}(s) = \begin{pmatrix} \boldsymbol{A} \\ \alpha_A \end{pmatrix} + \begin{pmatrix} \boldsymbol{B} - \boldsymbol{A} \\ \alpha_B - \alpha_A \end{pmatrix} s = \begin{pmatrix} 5.2 \\ 1.5 \\ -60^\circ \end{pmatrix} + \begin{pmatrix} -3 \\ -4 \\ 90^\circ \end{pmatrix} s, \qquad s \in [0,1].$$

As a consequence, we have also

$$\dot{\boldsymbol{p}}(t) = \frac{d\boldsymbol{p}(t)}{dt} = \frac{d\boldsymbol{p}(s)}{ds}\,\dot{s}(t) = \boldsymbol{p}'(s)\dot{s}(t) = (\boldsymbol{B} - \boldsymbol{A})\,\dot{s}(t), \qquad \ddot{\boldsymbol{p}}(t) = (\boldsymbol{B} - \boldsymbol{A})\,\ddot{s}(t), \tag{4}$$

and

$$\dot{\alpha}(t) = \frac{d\alpha(t)}{dt} = \frac{d\alpha(s)}{ds}\,\dot{s}(t) = \alpha'(s)\dot{s}(t) = (\alpha_B - \alpha_A)\,\dot{s}(t), \qquad \ddot{\alpha}(t) = (\alpha_B - \alpha_A)\,\ddot{s}(t). \tag{5}$$

The definition of the timing law $s = s(t)$ takes into account the smoothness requirement (continuity up to the acceleration $\ddot{s}(t)$, for $t \in [0,T]$) and the boundary conditions at the initial time $t = 0$ and final time $t = T$. In particular, the linear and angular velocity should satisfy the non-zero initial conditions

$$\dot{\boldsymbol{p}}(0) = (\boldsymbol{B} - \boldsymbol{A})\,\dot{s}(0) = \frac{\boldsymbol{B} - \boldsymbol{A}}{\|\boldsymbol{B} - \boldsymbol{A}\|}\,V \qquad \text{and} \qquad \dot{\alpha}(0) = (\alpha_B - \alpha_A)\,\dot{s}(0) = \Omega,$$

implying the following common conditions on the initial speed

$$\dot{s}(0) = \frac{V}{\|\boldsymbol{B} - \boldsymbol{A}\|} = \frac{2.5\,[\text{m/s}]}{5\,[\text{m}]} = 0.5 = v_i \qquad \text{and} \qquad \dot{s}(0) = \frac{\Omega}{\alpha_B - \alpha_A} = \frac{45\,[^\circ/\text{s}]}{90\,[^\circ]} = 0.5 = v_i. \tag{6}$$

The equality of the two numerical values for the alternative expressions of $\dot{s}(0) = v_i$ in (6) is indeed necessary in order to have an initial velocity that is *consistent* with the desired task. On the other hand, the condition of zero velocity at the final instant (a rest state) implies for the final speed

$$\dot{s}(T) = v_f = 0. \tag{7}$$

For the continuity of the scalar acceleration $\ddot{s}(t)$ also at $t = 0$ and $t = T$, we have to impose boundary accelerations too, though with arbitrary values a_i and a_f, i.e.,

$$\ddot{s}(0) = a_i, \qquad \ddot{s}(T) = a_f. \tag{8}$$

In fact, leaving instead these accelerations unconstrained would result in specific values at the boundaries (as outcome of a lower-order interpolation problem) that may not match those before the start and after the end of the planned motion (i.e., $\ddot{s}(0^-) \neq \ddot{s}(0^+)$ and/or $\ddot{s}(T^-) \neq \ddot{s}(T^+)$). As a result of the boundary conditions (6–8), the function interpolating an initial value $s(0) = s_i$ to a final value $s(T) = s_f$ is chosen to be a quintic polynomial. Its general expression can be given in terms of the normalized time $\tau = t/T$ as (see, e.g., the lecture slides)

$$\begin{aligned} s(\tau) &= (1-\tau)^3 \left(s_i + (3s_i + v_i T)\tau + 0.5\left(12 s_i + 6 v_i T + a_i T^2\right)\tau^2\right) \\ &\quad + \tau^3 \left(s_f + (3s_f - v_f T)(1-\tau) + 0.5\left(12 s_f - 6 v_f T + a_f T^2\right)(1-\tau)^2\right), \quad \tau \in [0,1]. \end{aligned} \tag{9}$$

Specializing (9) to the case at hand ($s_i = 0$, $s_f = 1$, $v_i = 0.5$, $v_f = 0$, and $T = 2$ [s]) and choosing for simplicity zero values for the boundary accelerations, $a_i = a_f = 0$, results in

$$s(\tau) = (1-\tau)^3(\tau + 3\tau^2) + \tau^3(1 + 3(1-\tau) + 6(1-\tau)^2) = 3\tau^5 - 7\tau^4 + 4\tau^3 + \tau, \quad \tau \in [0,1]. \tag{10}$$

Furthermore,

$$\dot{s}(\tau) = \frac{ds(\tau)}{dt} = \frac{ds(\tau)}{d\tau}\frac{d\tau}{dt} = \frac{1}{T}\frac{ds(\tau)}{d\tau} = 0.5\left(15\tau^4 - 28\tau^3 + 12\tau^2 + 1\right), \qquad \tau \in [0,1], \tag{11}$$

and

$$\ddot{s}(\tau) = \frac{1}{T^2}\frac{d^2 s(\tau)}{d\tau^2} = 0.25\left(60\tau^3 - 84\tau^2 + 24\tau\right) = \left(15\tau^2 - 21\tau + 6\right)\tau, \qquad \tau \in [0,1]. \tag{12}$$

The plots of (10–12) over the actual time $t \in [0,T] = [0,2]$ are shown in Fig. 5. Note in particular the asymmetry of the speed profile. Also, at the half-time $t = T/2 = 1$ (or $\tau = 0.5$) more than half of the path length $\|\boldsymbol{B} - \boldsymbol{A}\| = 5$ [m] will have been traced, being $s_m = s(0.5) = 0.6562 > 0.5$.

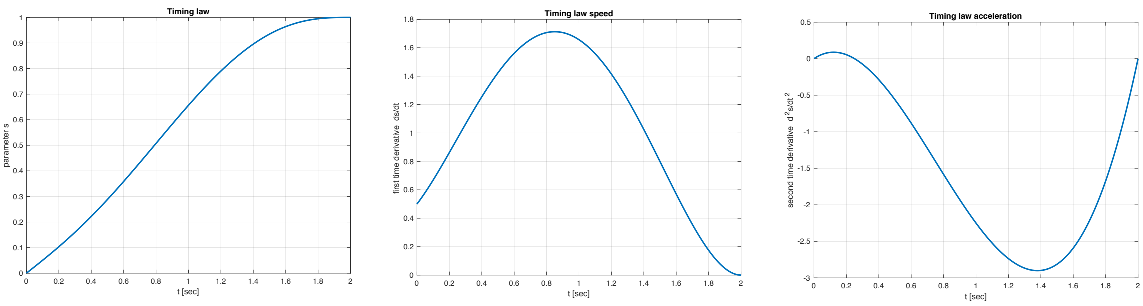


Figure 5: The time evolution of $s(t)$, $\dot{s}(t)$, and $\ddot{s}(t)$, as given by eqs. (10–12).

The task velocity at the half-time is obtained from (4), (5) and (11) as[1]

$$\dot{\boldsymbol{r}}(1) = \begin{pmatrix} \dot{\boldsymbol{p}}(1) \\ \dot{\alpha}(1) \end{pmatrix} = \begin{pmatrix} \boldsymbol{B} - \boldsymbol{A} \\ \alpha_B - \alpha_A \end{pmatrix} \dot{s}(0.5) = \begin{pmatrix} -3 \\ -4 \\ 90^\circ \end{pmatrix} 1.6562 = \begin{pmatrix} -4.9688 \\ -6.6250 \\ 149.06\ [^\circ/\mathrm{s}] \end{pmatrix}.$$

Figures 6–8 show the time evolutions of the two coordinates, respectively, of the Cartesian position, linear velocity, and linear acceleration, placed side by side with the evolution of the angle with respect to the surface normal $\boldsymbol{n}$, its speed and acceleration. As it can be seen, all boundary conditions are satisfied. Moreover, coordinated motion follows from the chosen planning approach, with space-time decomposition and a common timing law for all variables: all quantities start and end their motion at the same time.

[1] In this formula, $\dot{\boldsymbol{p}}$ and $\dot{\alpha}$ are expressed with respect to time t, whereas $\dot{s}$ uses the normalized time τ.

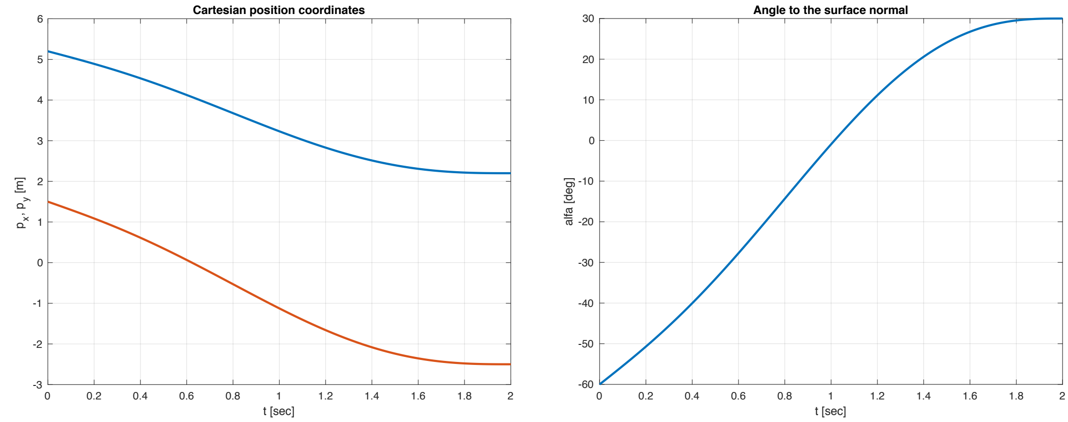


Figure 6: Cartesian position $\boldsymbol{p}(t) = (p_x(t), p_y(t))$ [left] and angle $\alpha(t)$ to the surface normal [right].

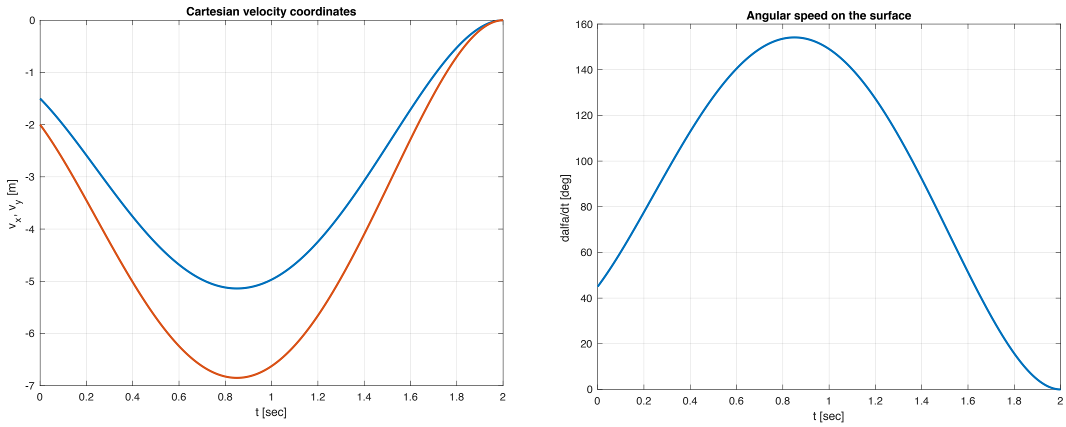


Figure 7: Cartesian velocity $\dot{\boldsymbol{p}}(t) = (v_x(t), v_y(t))$ [left] and angular speed $\dot{\alpha}(t)$ [right].

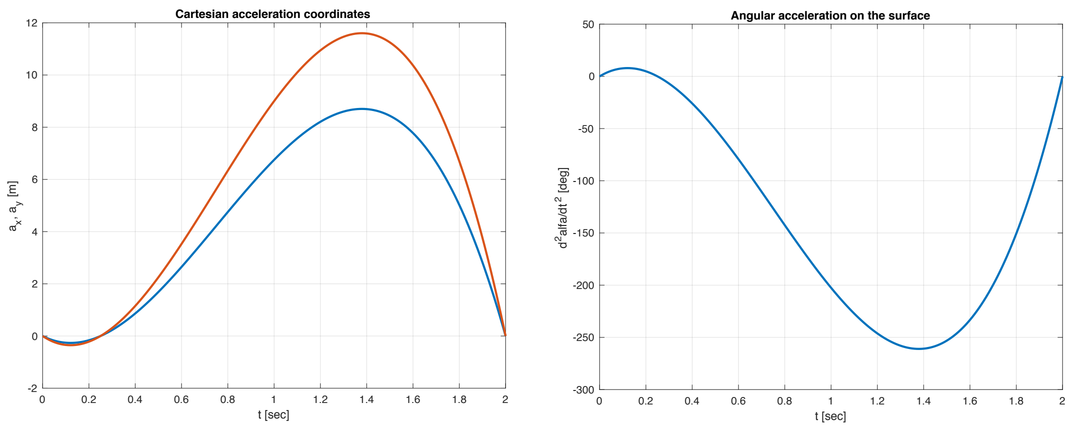


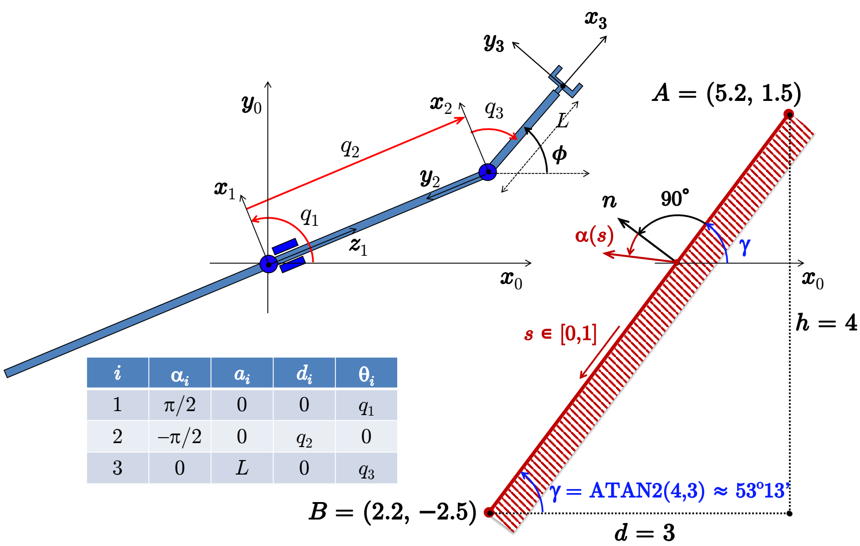
Figure 8: Cartesian acceleration $\ddot{\boldsymbol{p}}(t) = (a_x(t), a_y(t))$ [left] and angular acceleration $\ddot{\alpha}(t)$ [right].

In the second part of the problem, items (d) and (e), we have to map the task planned so far into a joint motion of the planar RPR robot. The assigned DH frames and the associated joint variables are defined in Fig. 9. The direct kinematics for the robot end-effector position $\boldsymbol{p}_e \in \mathbb{R}^2$ and its absolute orientation (as given by an angle $\phi_e \in \mathbb{R}$ defined w.r.t. the axis $\boldsymbol{x}_0$ of the robot base frame) is

$$\begin{pmatrix} \boldsymbol{p}_e \\ \phi_e \end{pmatrix} = \boldsymbol{f}_e(\boldsymbol{q}) = \begin{pmatrix} L\cos(q_1+q_3) + q_2 \sin q_1 \\ L\sin(q_1+q_3) - q_2\cos q_1 \\ q_1 + q_3 \end{pmatrix}, \tag{13}$$

and the associated Jacobian is

$$\boldsymbol{J}_e(\boldsymbol{q}) = \frac{\partial \boldsymbol{f}_e(\boldsymbol{q})}{\partial \boldsymbol{q}} = \begin{pmatrix} -L\sin(q_1+q_3) + q_2\cos q_1 & \sin q_1 & -L\sin(q_1+q_3) \\ L\cos(q_1+q_3) + q_2\sin q_1 & -\cos q_1 & L\cos(q_1+q_3) \\ 1 & 0 & 1 \end{pmatrix}. \tag{14}$$



i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	0	q_1
2	$-\pi/2$	0	q_2	0
3	0	L	0	q_3

Figure 9: DH frames and joint variables for the planar RPR robot.

To match the orientation of the task with the robot end-effector orientation, we should consider the different definition of the two angles ϕ_e (pertaining to the robot) and α (pertaining to the task). Still with reference to Fig. 9, it is easy to see that we have the relation[2]

$$\begin{aligned} \phi_e &= \alpha_{\text{abs}} - 180° = \left(\text{ATAN2}\left\{A_y - B_y, A_x - B_x\right\} + 90° + \alpha\right) - 180° \\ &= \alpha + \text{ATAN2}\{4,3\} - 90° = \alpha - 36.87°. \end{aligned} \tag{15}$$

The angle α_{abs} is the absolute orientation w.r.t. the axis $\boldsymbol{x}_0$ imposed by the task, based on the desired angle α specified w.r.t. the normal $\boldsymbol{n}$ to the surface. The subtraction (or also the addition) of a half turn is due to the fact that the robot end-effector has to approach the desired orientation from the external side of the surface.

[2]Although most of the angular quantities in this exercise are expressed in degrees (just to match the initial format of the problem data), remember that computations (e.g., arguments of trigonometric functions) are always, more conveniently expressed in radians.

For any desired value $\boldsymbol{r}(s)$ of the parametrized task in (3), we obtain, using also (15), equivalent conditions for the robot end-effector variables as

$$\boldsymbol{p}_{ed}(s) = \begin{pmatrix} p_{ed,x}(s) \\ p_{ed,y}(s) \end{pmatrix} = \boldsymbol{p}(s), \qquad \phi_{ed}(s) = \alpha(s) - 36.87°, \qquad \text{with } s \in [0,1]. \tag{16}$$

The inverse kinematics problem consists in finding one (or more) $\boldsymbol{q}(s) = \begin{pmatrix} q_1(s) & q_2(s) & q_3(s) \end{pmatrix}^T$ in parametrized form, such that

$$\begin{pmatrix} L\cos\left(q_1(s)+q_3(s)\right) + q_2(s)\sin q_1(s) \\ L\sin\left(q_1(s)+q_3(s)\right) - q_2(s)\cos q_1(s) \\ q_1(s)+q_3(s) \end{pmatrix} = \begin{pmatrix} p_{ed,x}(s) \\ p_{ed,y}(s) \\ \phi_{ed}(s) \end{pmatrix}, \qquad \forall s \in [0,1]. \tag{17}$$

Thanks to the non-negative assumption made on the prismatic joint variable, $q_2 \geq 0$, equations 17 admit one and only one solution
$qv(s)$ for each $s \in [0,1]$. In fact, we have

$$q_1(s) + q_3(s) = \phi_{ed}(s) \quad \Rightarrow \quad \begin{pmatrix} q_2(s)\sin q_1(s) \\ -q_2(s)\cos q_1(s) \end{pmatrix} = \begin{pmatrix} p_{ed,x}(s) - L\cos\phi_{ed}(s) \\ p_{ed,y}(s) - L\sin\phi_{ed}(s) \end{pmatrix},$$

and so

$$\begin{aligned} q_1(s) &= \text{ATAN2}\left\{-\left(p_{ed,y}(s) - L\sin\phi_{ed}(s)\right), p_{ed,x}(s) - L\cos\phi_{ed}(s)\right\}, \\ q_2(s) &= \sqrt{\left(p_{ed,x}(s) - L\cos\phi_{ed}(s)\right)^2 + \left(p_{ed,y}(s) - L\sin\phi_{ed}(s)\right)^2} \geq 0, \\ q_3(s) &= \phi_{ed}(s) - q_1(s). \end{aligned} \tag{18}$$

As a result, the parametrized Cartesian task $\boldsymbol{r} = \boldsymbol{r}(s)$ implies also a unique parametrized path $\boldsymbol{q} = \boldsymbol{q}(s)$ in the robot joint space.

At the motion half-time $t = T/2 = 1$ [s], for the desired task expressed in time we have

$$\boldsymbol{r}(1) = \begin{pmatrix} \boldsymbol{p}(1) \\ \alpha(1) \end{pmatrix} = \begin{pmatrix} 3.2313 \\ -1.1250 \\ -0.94° \end{pmatrix}.$$

Evaluating (16) at the corresponding $s = s_m = 0.6562$ gives

$$\boldsymbol{p}_{ed}(s_m) = \begin{pmatrix} 3.2313 \\ -1.1250 \end{pmatrix} \text{[m]}, \qquad \phi_{ed}(s_m) = -37.80° \ (= -0.6599 \text{ [rad]}),$$

and so, from (18),

$$\boldsymbol{q}_m = \boldsymbol{q}(s_m) = \begin{pmatrix} 78.15° \\ 2.4943 \\ -115.96° \end{pmatrix} = \begin{pmatrix} 1.3641 \\ 2.4943 \\ -2.0239 \end{pmatrix} \text{[rad, m, rad]}.$$

The configuration of the RPR robot at this stage along the path from $\boldsymbol{A}$ to $\boldsymbol{B}$ is shown in Fig. 10. Note that in practice the end-effector is almost oriented along the normal to the linear surface.

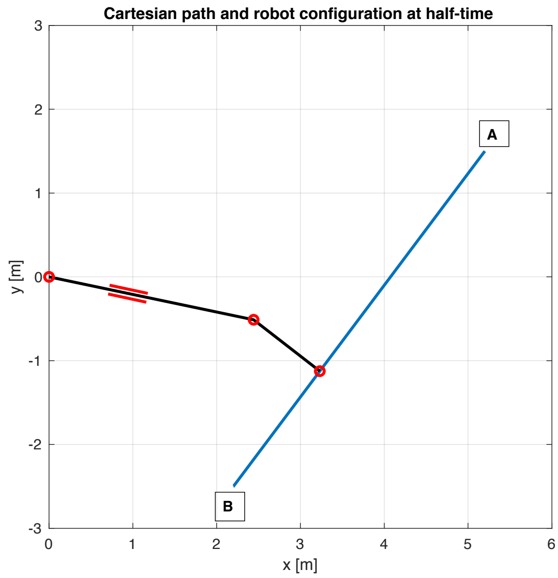


Figure 10: The RPR robot configuration $\boldsymbol{q}_m$ reached at the half-time $t = T/2 = 1$ [s] of the task.

Finally, using $\boldsymbol{q}_m$ to evaluate the Jacobian in (14) and plugging in the link length $L = 1$, we compute the joint velocity $\dot{\boldsymbol{q}}_m$ at the motion half-time from the corresponding task velocity $\dot{\boldsymbol{r}}_m$:

$$\dot{\boldsymbol{q}}_m = \boldsymbol{J}_e^{-1}(\boldsymbol{q}_m)\dot{\boldsymbol{r}}_m = \begin{pmatrix} 1.1250 & 0.9787 & 0.6130 \\ 3.2313 & -0.2053 & 0.7901 \\ 1 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} -4.9688 \\ -6.6250 \\ 2.6016 \end{pmatrix}$$

$$= \begin{pmatrix} -3.9462 \\ -4.6420 \\ 6.5478 \end{pmatrix} [\text{rad/s, m/s, rad/s}] = \begin{pmatrix} -226.1 \\ -4.64 \\ 375.16 \end{pmatrix} [^\circ\text{/s, m/s, }^\circ\text{/s}].$$

Exercise 4

Answer to the questions or comment/complete the statements, providing also a *short* motivation/explanation for each of the following 8 items.

1. At the same level of resolution, the cost of incremental encoders is usually less than that of absolute encoders because ...

A: ... incremental encoders have a simpler structure for generating pulses, with a regular optical disc and only three channels, while absolute encoders have more electronic components inside (arrays of opaque/transparent LED/sensor pairs), Gray-to-binary code converters, multi-turn option, an optical disc with more complex patterns to be engraved, etc.

2. What is the purpose of using Wheatstone bridge configurations in the electronics of strain gages?

A: Strain gages are configured in Wheatstone bridge circuits (the electrical equivalent of two parallel voltage divider circuits) so as to better detect small changes in resistance (and thus in the applied strain). Moreover, special such configurations (e.g., two strain gages used in a quarter-bridge) help in further minimize undesired effects due to temperature changes.

3. Compare the link position resolution of an incremental encoder with 600 pulses per revolution (PPR) mounted on the motor having a transmission of reduction ratio $n_r = 30$, with that of an incremental encoder with 4000 PPR and quadrature electronics mounted directly on the link. Which is better?

A: *The position resolution at the link level is* $r_1 = 360°/(600 \cdot 30) = 0.02°$ *in the first case, and* $r_2 = 360°/(4000 \cdot 4) = 0.0225°$ *in the second case. Thus, the first setup is (slightly) better.*

4. Given a desired end-effector position of a planar 3R robot, the iterative Newton method can find all solutions to the inverse kinematics problem out of singularities. True or false? Why?

A: *False. The considered problem has in fact an infinite number of solutions, so neither a numerical nor an analytical method can generate all of them.*

5. Which is the relation between the second derivative $\ddot{\boldsymbol{R}}$ of a time-varying rotation matrix $\boldsymbol{R}(t)$ and the associated angular velocity $\boldsymbol{\omega}$ and acceleration $\dot{\boldsymbol{\omega}}$?

A: *We differentiate the known relation* $\dot{\boldsymbol{R}} = \boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{R}$ *w.r.t. time. We have thus:*

$$\ddot{\boldsymbol{R}} = \dot{\boldsymbol{S}}(\boldsymbol{\omega})\boldsymbol{R} + \boldsymbol{S}(\boldsymbol{\omega})\dot{\boldsymbol{R}} = \left(\boldsymbol{S}(\dot{\boldsymbol{\omega}}) + \boldsymbol{S}^2(\boldsymbol{\omega})\right)\boldsymbol{R} = \left(\boldsymbol{S}(\dot{\boldsymbol{\omega}}) + \boldsymbol{\omega}\boldsymbol{\omega}^T - \boldsymbol{I}\|\boldsymbol{\omega}\|^2\right)\boldsymbol{R},$$

with the 3×3 *identity matrix* $\boldsymbol{I}$. *Note that the dependence on the angular acceleration* $\dot{\boldsymbol{\omega}}$ *is linear, while* $\boldsymbol{S}^2$ *leads to quadratic terms in the angular velocity* $\boldsymbol{\omega}$.

6. For a joint that needs to move by $\Delta q > 0$, if the bounds on maximum absolute velocity and acceleration are related by $A_{max} = V_{max}^2/\Delta q$, is the minimum time acceleration profile always bang-coast-bang?

A: *No. In a trapezoidal speed profile, the coast phase for the acceleration (the time interval of motion with zero acceleration and maximum speed) exists if and only if* $V_{max}^2/A_{max} < \Delta q$. *The above relation provides instead* $V_{max}^2/A_{max} = \Delta q$. *Thus, the maximum speed is reached in just one instant (at the half-time of motion), not for a finite interval.*

7. The uniform time scaling procedure allows to obtain the minimum motion time along a parametrized path under maximum velocity and acceleration constraints. True or false? Why?

A: *False. By uniformly scaling time, the fastest motion profile that is obtained will have in general just one velocity or acceleration constraint saturated, and in one instant only. On the other hand, we would get faster motions by speeding up where the bounds are largely satisfied and slowing down where they are violated. The minimum motion time along the given parametrized path will be obtained by suitably choosing such a non-uniform time scaling of the original trajectory.*

8. Kinematic control laws designed at the Cartesian level are better than those designed at the joint level because ..., and are worse because ...

A: *Cartesian (or task) kinematic control guarantees asymptotically stable tracking of trajectories that are defined directly in the most relevant space for evaluating robot performance, with errors that will converge to zero exponentially and in a decoupled way. The downside is that the control law is more complex to be implemented in real time. Moreover, singularities that would highly affect robot motion may be encountered at run time and should be carefully handled online.*

* * * * *