Advanced Control Systems: RPP manipulator

Filippo Grotto VR460638

October 20, 2021

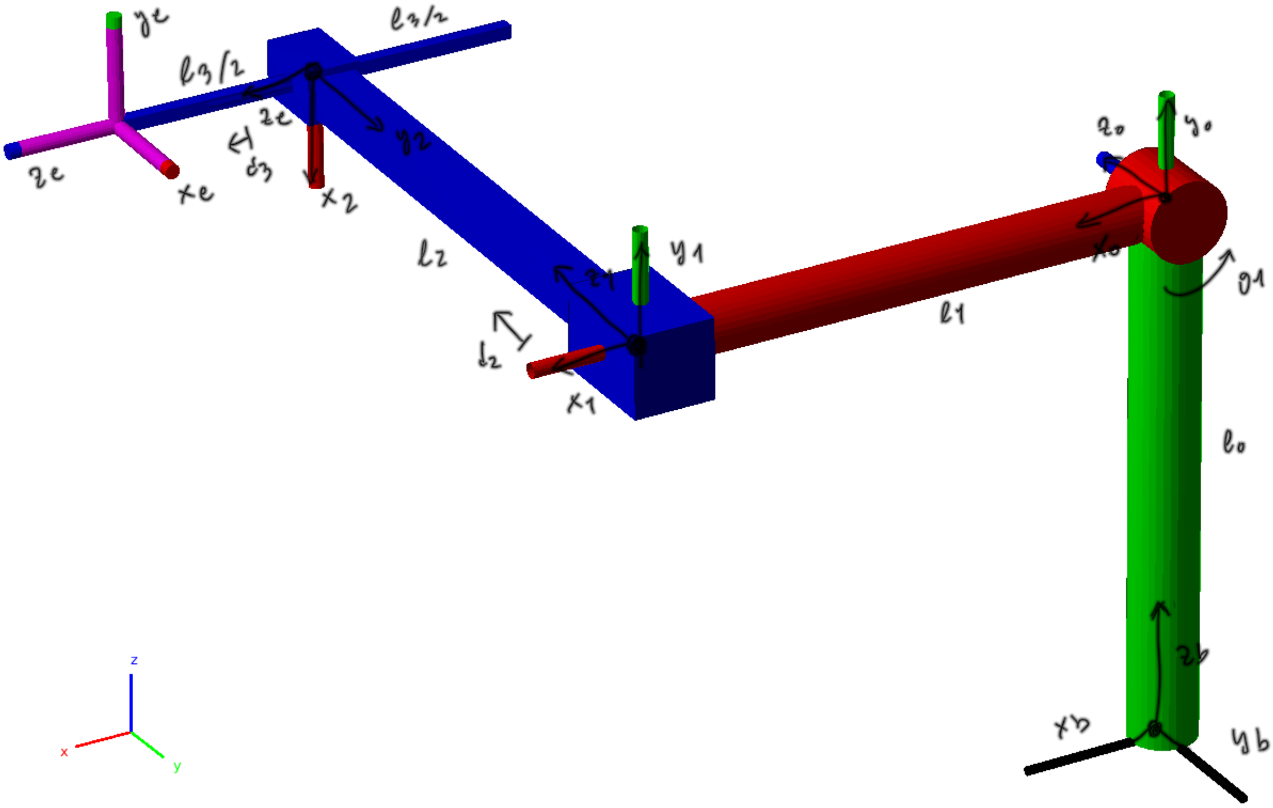
Contents

1 Kinematics **2**
- 1.1 Direct Kinematics 2
- 1.2 Inverse Kinematics 3

2 Jacobians **3**
- 2.1 Geometric Jacobians 3
- 2.2 Analytical Jacobian 4

1 Kinematics

1.1 Direct Kinematics



Lets define the DH table for our manipulator:

Σ_i	d_i	θ_i	a_i	α_i
$b-0$	ℓ_0	0	0	$\frac{\pi}{2}$
$0-1$	0	θ_1	ℓ_1	0
$1-2$	ℓ_2+d_2	$\frac{\pi}{2}$	0	$\frac{\pi}{2}$
$2-3$	ℓ_3+d_3	$\frac{\pi}{2}$	0	0
$3-e$	0	0	0	0

The homogenous transformation is defined according to the following matrix and calculated for each row of the DH table. By multiplying $H_0^b H_1^0 H_2^1 H_3^2 H_e^3$ we obtain the final transformation

$$H_i^{i-1}(q_i) = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_0^b = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & \ell_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad H_1^0(\theta_1) = \begin{bmatrix} c_1 & -s_1 & 0 & \ell_1 c_1 \\ s_1 & c_1 & 0 & \ell_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad H_2^1(d_2) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_2+\ell_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_3^2(d_3) = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & d_3+l_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad H_e^3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$H_e^b(\boldsymbol{q}) = \begin{bmatrix} 0 & -s_1 & c_1 & c_1(\ell_1+\ell_3+d_3) \\ -1 & 0 & 0 & -\ell_2-d_2 \\ 0 & -c_1 & s_1 & s_1(\ell_1+\ell_3+d_3)+\ell_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

1.2 Inverse Kinematics

Let's consider the position of ee with respect of the base frame to calculate the value of the joints.

$$p_e^b = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} c_1(\ell_1+\ell_3+d_3) \\ -\ell_2-d_2 \\ s_1(\ell_1+\ell_3+d_3)+\ell_0 \\ 1 \end{bmatrix}$$

It is easy to see that

$$d_2 = -\ell_2 - y$$

$$\theta_1 = Atan2(z-\ell_0, x)$$

For d_3 we can apply sum of squares and the result is:

$$d_3 = -\ell_1 \pm \sqrt{x^2+(z-\ell_0)^2} - \ell_3$$

2 Jacobians

2.1 Geometric Jacobians

The geometric jacobian is defined as follow with $\boldsymbol{q} = [\theta_1, d_2, d_3]^\top$. Note that the matlab robotic toolbox defines the angular velocities above the linear velocities:

$$\begin{bmatrix} \dot{p}_e \\ \omega_e \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \left[\begin{array}{c|c|c} J_{P_1} & J_{P_2} & J_{P_3} \\ J_{O_1} & J_{O_2} & J_{O_3} \end{array} \right] \begin{bmatrix} \dot{\theta}_1 \\ \dot{d}_2 \\ \dot{d}_3 \end{bmatrix}$$

$$J_{P_1} = z_0 \times (d_e^0 - d_0^0) = \begin{bmatrix} -s_1(\ell_1 + \ell_3 + d_3) \\ 0 \\ c_1(\ell_1 + \ell_3 + d_3) \end{bmatrix} \qquad J_{O_1} = z_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$J_{P_2} = z_1 = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \qquad J_{O_2} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$J_{P_3} = z_2 = \begin{bmatrix} c_1 \\ 0 \\ s_1 \end{bmatrix} \qquad J_{O_3} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

We can finally put all the pieces together and obtain the final geometric jacobian:

$$J(\boldsymbol{q}) = \begin{bmatrix} -s_1(\ell_1 + \ell_3 + d_3) & 0 & c1 \\ 0 & -1 & 0 \\ c_1(\ell_1 + \ell_3 + d_3) & 0 & s1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

2.2 Analytical Jacobian

The analytical jacobian can be easily calculated by using partial derivatives of p_e^b

$$p_e^b = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} c_1(\ell_1 + \ell_3 + d_3) \\ -\ell_2 - d_2 \\ s_1(\ell_1 + \ell_3 + d_3) + \ell_0 \\ 1 \end{bmatrix}$$

Finally we end up with the analytical jacobian

$$Ja(\boldsymbol{q}) = \begin{bmatrix} -s_1(\ell_1 + \ell_3 + d_3) & 0 & c1 \\ 0 & -1 & 0 \\ c_1(\ell_1 + \ell_3 + d_3) & 0 & s1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Another possibility is to use the relation between the geometric and analytical jacobian as follow using ZYZ:

$$\omega_e = T(\phi_e)\dot{\phi}_e \qquad T(\phi_e) = \begin{bmatrix} 0 & -s_\varphi & c_\varphi s_\theta \\ 0 & c_\varphi & s_\varphi s_\theta \\ 1 & 0 & c_\theta \end{bmatrix}$$

$$J(\boldsymbol{q}) = T_A(\phi_e) J_A(\boldsymbol{q})$$

$$T_A(\phi_e) = \begin{bmatrix} \mathbb{I}_3 & \emptyset_3 \\ \emptyset_3 & T(\phi_e) \end{bmatrix}$$