Technion – Israel Institute of Technology

Faculty of Mechanical Engineering

Project – Part 1

Kinematics, Dynamics, and Control of Robots

036026

Alon Spinner	305184335	alonspinner@gmail.com
Shahar Tsadok		Shahar507@gmail.com

March 10, 2021

1 Finding the Direct Kinematics of the Robot

Transformation matrices transfer homogenous points $\boldsymbol{X} = [x, y, z, 1]^T$ from one coordinate system, say O_1, to the other, O_0.

$$\boldsymbol{X}_0 = {}^0\boldsymbol{A}_1 \boldsymbol{X}_1.$$

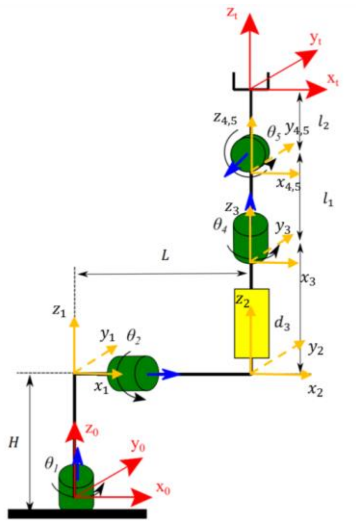
$${}^0\boldsymbol{A}_1 = \begin{bmatrix} \boldsymbol{R}^{3x3} & \boldsymbol{t}^{3x1} \\ \boldsymbol{0}^{1x3} & 1 \end{bmatrix}$$

A transformation matrix from basis O_1 to O_0 is structed from rotational and translational elements. The rotation matrix $\boldsymbol{R}$ has the basis vectors of O_1 in its columns with the representation of O_0. The translation vector $\boldsymbol{t}$ is the translation $\boldsymbol{t} = O_1 - O_0$ in the representation of O_0

$${}^0A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & H \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1A_2 = \begin{bmatrix} 1 & 0 & 0 & L \\ 0 & c_2 & -s_2 & 0 \\ 0 & s_2 & c_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^3A_4 = \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & l_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4A_5 = \begin{bmatrix} c_5 & 0 & -s_5 & 0 \\ 0 & 1 & 0 & 0 \\ s_5 & 0 & c_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5A_T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



$${}^0A_t = {}^0A_1\,{}^1A_2\,{}^2A_3\,{}^3A_4\,{}^4A_5\,{}^5A_t$$

$$\boxed{{}^0A_t = \begin{bmatrix} c_5(c_1c_4 - c_2s_1s_4) + s_1s_2s_5 & -c_1s_4 - c_2c_4s_1 & c_5s_1s_2 - s_5(c_1c_4 - c_2s_1s_4) & Lc_1 - l_2(s_5(c_1c_4 - c_2s_1s_4) - c_5s_1s_2) + d_3s_1s_2 + l_1s_1s_2 \\ c_5(c_4s_1 + c_1c_2s_4) - c_1s_2s_5 & c_1c_2c_4 - s_1s_4 & -s_5(c_4s_1 + c_1c_2s_4) - c_1c_5s_2 & Ls_1 - l_2(s_5(c_4s_1 + c_1c_2s_4) - c_1c_5s_2) - d_3c_1s_2 - l_1c_1s_2 \\ c_2s_5 + c_5s_2s_4 & c_4s_2 & c_2c_5 - s_2s_4s_5 & H + l_2(c_2c_5 - s_2s_4s_5) + d_3c_2 + l_1c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.0.0)}$$

2 Finding Inverse Kinematics Under Assumption of No Constraints

The problem of inverse kinematic can be formulated as such:
given a target transformation matrix which encapsulates the position and orientation of the tool in world coordinate system

$${}^0A_t = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{T} \\ \boldsymbol{0} & 1 \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{x}}_t & \hat{\boldsymbol{y}}_t & \hat{\boldsymbol{z}}_t & \boldsymbol{T} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_{x,1} & n_{y,1} & n_{z,1} & T_x \\ n_{x,2} & n_{y,2} & n_{z,2} & T_y \\ n_{x,3} & n_{y,3} & n_{z,3} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

Find the joint values $\boldsymbol{q} = [\theta_1, \theta_2, d_3, \theta_4, \theta_5]$.

To solve the inverse kinematics, we break the problem into 3 parts.

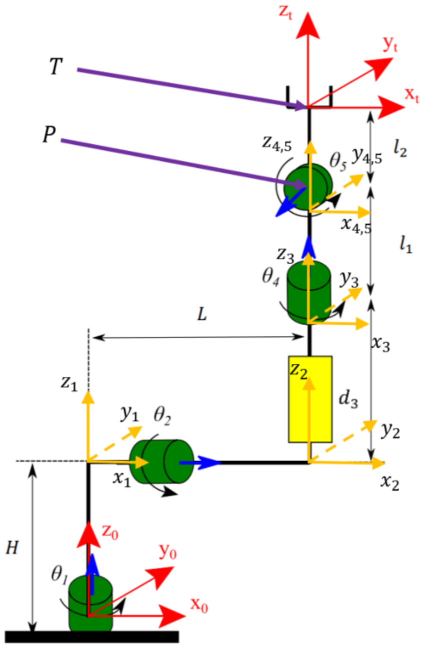
1) We denote point P to be the origin of coordinate system O_4 and solve for it given the target transformation matrix.

$$P_0 = T_0 - {}^0R_T \cdot \begin{bmatrix} 0 \\ 0 \\ l_2 \end{bmatrix}$$

2) Solve for θ_1, θ_2, d_3 using the transformation matrix ${}^0A_P$ found by direct kinematics.

$${}^0A_P[:,4] = P_0$$

3) Find θ_4, θ_5 by equating terms of ${}^0R_t$ from two different representations: the target matrix and the direct kinematics.



2.1 Finding Point P_0 Given Target Transformation Matrix

Given that the direction of the tool and position is known, finding P_0 is straightforward.

$$P_0 = T_0 - {}^0R_T \cdot \begin{bmatrix} 0 \\ 0 \\ l_2 \end{bmatrix} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} - \begin{bmatrix} n_{x,1} & n_{y,1} & n_{z,1} \\ n_{x,2} & n_{y,2} & n_{z,2} \\ n_{x,3} & n_{y,3} & n_{z,3} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ l_2 \end{bmatrix}$$

$$\boxed{P_0 = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} - \begin{bmatrix} n_{z,1} \\ n_{z,2} \\ n_{z,3} \end{bmatrix} l_2} \tag{2.1.0}$$

2.2 Finding θ_1, θ_2, d_3 From Point P_0 and Joint-Based Transformation Matrix ${}^0A_4$

$${}^0A_4 = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4$$

$${}^0A_P = \begin{bmatrix} c_1c_4 - c_2s_1s_4 & -c_1s_4 - c_2c_4s_1 & s_1s_2 & Lc_1 + d_3s_1s_2 + l_1s_1s_2 \\ c_4s_1 + c_1c_2s_4 & c_1c_2c_4 - s_1s_4 & -c_1s_2 & Ls_1 - d_3c_1s_2 - l_1c_1s_2 \\ s_2s_4 & c_4s_2 & c_2 & H + d_3c_2 + l_1c_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.2.0}$$

The location of P in system O_0 is therefore

$$P_0 = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} Lc_1 + d_3 s_1 s_2 + l_1 s_1 s_2 \\ Ls_1 - d_3 c_1 s_2 - l_1 c_1 s_2 \\ H + d_3 c_2 + l_1 c_2 \end{bmatrix} \tag{2.2.1}$$

A general formula can be applied to our case.

$$\begin{cases} acos\theta - bsin\theta = c \\ asin\theta + bcos\theta = d \end{cases} \rightarrow \begin{cases} \theta = atan2(ad - bc, ac + bd) \\ a^2 + b^2 = c^2 + d^2 \end{cases} \tag{2.2.2}$$

Rewriting the first two equations in (2.2.1)

$$\begin{cases} (d_3 + l_1) s_2 c_1 - L s_1 = -P_y \\ (d_3 + l_1) s_2 s_1 + L c_1 = P_x \end{cases}$$

Denoting terms and using the first equation in (2.2.2) we can find θ_1.

$$\boxed{\begin{matrix} a = (d_3 + l_1) s_2 \\ b = L \\ c = -P_y \\ d = P_x \end{matrix} \quad \rightarrow \theta_1 = atan2(ad - bc, ac + bd)} \tag{2.2.3}$$

Finding d_3 is done by substituting the terms in the second equation of (2.2.2), and reorganizing and squaring the third equation in (2.2.1)

$$(d_3 + l_1)^2 s_2^2 \ + L^2 = P_x^2 + P_y^2$$

$$(d_3 + l_1)^2 c_2^2 = (P_z - H)^2$$

Adding the two equations we can find d_3

$$(d_3 + l_1)^2 + L^2 = P_x^2 + P_y^2 + (P_z - H)^2$$

$$\boxed{d_3 = \pm\sqrt{P_x^2 + P_y^2 + (P_z - H)^2 - L^2} - l_1} \tag{2.2.4}$$

To find θ_2 we utilize the third equation in (2.2.2)

$$\boxed{\begin{matrix} c_2 = \dfrac{P_z - H}{d_3 + l_1} \\ s_2 = \pm\sqrt{(1 - c_2^2)} \\ \theta_2 = \mathrm{atan}\,(s_2, c_2) \end{matrix}} \tag{2.2.5}$$

If both d_3 and l_1 equate to zero, equation (2.2.5) will be impossible to solve, hence we make a small distinction which will nullify c_2 in such a case.

$$c_2 = \frac{P_z - H}{d_3 + l_1 + \epsilon} \quad \epsilon - small\ positive\ number$$
$$s_2 = \pm\sqrt{(1 - c_2^2)} \tag{2.2.6}$$
$$\theta_2 = \operatorname{atan}(s_2, c_2)$$

2.3 Finding θ_4, θ_5 by Equating Two Representations of ${}^0R_t$

Given a target rotation matrix, we can compare it to the rotational part of equation (1.0.0) and find the required joint values.

$$\begin{bmatrix} n_{x,1} & n_{y,1} & n_{z,1} \\ n_{x,2} & n_{y,2} & n_{z,2} \\ n_{x,3} & n_{y,3} & n_{z,3} \end{bmatrix} = \begin{bmatrix} c_5(c_1c_4 - c_2s_1s_4) + s_1s_2s_5 & -c_1s_4 - c_2c_4s_1 & c_5s_1s_2 - s_5(c_1c_4 - c_2s_1s_4) \\ c_5(c_4s_1 + c_1c_2s_4) - c_1s_2s_5 & c_1c_2c_4 - s_1s_4 & -s_5(c_4s_1 + c_1c_2s_4) - c_1c_5s_2 \\ c_2s_5 + c_5s_2s_4 & c_4s_2 & c_2c_5 - s_2s_4s_5 \end{bmatrix} \tag{2.3.0}$$

From the (3,2) placement in equation (2.3.0)

$$n_{y,3} = c_4s_2$$
$$c_4 = \frac{s_2}{n_{y,3}} \tag{2.3.1}$$

From the (2,2) placement in equation (2.3.0)

$$n_{y,2} = c_1c_2c_4 - s_1s_4$$
$$s_4 = \frac{c_1c_2c_4 - n_{y,2}}{s_1}$$

Substituting for c_4 in equation (2.3.1)

$$s_4 = \frac{\frac{s_2}{n_{y,3}}c_1c_2 - n_{y,2}}{s_1} \tag{2.3.2}$$

We can then find θ_4 with equations (2.3.1), (2.3.2)

$$\boxed{\theta_4 = atan2(s_4, c_4) = atan2\left(\frac{\frac{s_2}{n_{y,3}}c_1c_2 - n_{y,2}}{s_1}, \frac{s_2}{n_{y,3}}\right)} \tag{2.3.3}$$

To find θ_5 we will use the (3,1), (3,3) placements in equation (2.3.0)

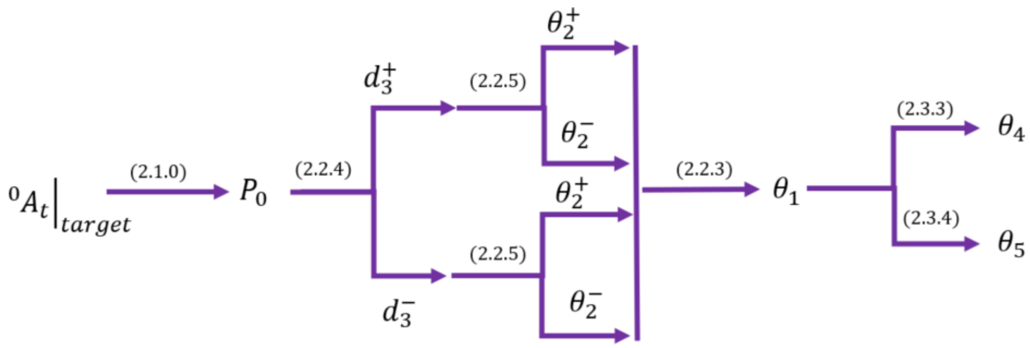
$$\begin{cases} n_{z,3} = c_2c_5 - s_2s_4s_5 \\ n_{x,3} = c_2c_5 + s_2s_4s_5 \end{cases}$$

Plugging into the general formula in (2.2.2)

$$\boxed{\begin{aligned} a &= c_2 \\ b &= s_2 s_4 \quad \rightarrow \theta_5 = atan2(ad - bc, ac + bd) \\ c &= n_{z,3} \\ d &= n_{x,3} \end{aligned}} \tag{2.3.4}$$

Inverse Kinematic Flow and Equations

To sum things up, we provide a figure of the computation flow from the target transformation matrix to the joints. Notice that two joints, d_3 and θ_2 have ambiguities that need to be decided prior to the computation.



3 Solving for the Jacobian

The Jacobian matrix transfers the n joint velocities to tool velocities. It can be written as two matrices: $\boldsymbol{J}_L$ transfers to linear velocity, and $\boldsymbol{J}_A$ transfers to angular velocity

$$\begin{bmatrix} \boldsymbol{v}^{3\times1} \\ \boldsymbol{\omega}^{3\times1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_L^{3\times n} \\ \boldsymbol{J}_A^{3\times n} \end{bmatrix} \dot{\boldsymbol{q}}^{n\times1}$$

To find $\boldsymbol{J}_L$ we differentiated the translation vector found by the direct kinematics (equation (1.0.0)) with respect to joint vector $\boldsymbol{q}$.

$$\boldsymbol{T} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = {}^0A_t(1{:}3,4) = \begin{bmatrix} Lc_1 - l_2(s_5(c_1c_4 - c_2s_1s_4) - c_5s_1s_2) + d_3s_1s_2 + l_1s_1s_2 \\ Ls_1 - l_2(s_5(c_4s_1 + c_1c_2s_4) - c_1c_5s_2) - d_3c_1s_2 - l_1c_1s_2 \\ H + l_2(c_2c_5 - s_2s_4s_5) + d_3c_2 + l_1c_2 \end{bmatrix}$$

$$\boldsymbol{J}_L = \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{q}} = \begin{bmatrix} \frac{\partial T_x}{\partial \theta_1} & \frac{\partial T_x}{\partial \theta_2} & \frac{\partial T_x}{\partial d_3} & \frac{\partial T_x}{\partial \theta_4} & \frac{\partial T_x}{\partial \theta_5} \\ \frac{\partial T_y}{\partial \theta_1} & \frac{\partial T_y}{\partial \theta_2} & \frac{\partial T_y}{\partial d_3} & \frac{\partial T_y}{\partial \theta_4} & \frac{\partial T_y}{\partial \theta_5} \\ \frac{\partial T_z}{\partial \theta_1} & \frac{\partial T_z}{\partial \theta_2} & \frac{\partial T_z}{\partial d_3} & \frac{\partial T_z}{\partial \theta_4} & \frac{\partial T_z}{\partial \theta_5} \end{bmatrix}$$

$$\boldsymbol{J}_L(:,1) = \frac{\partial \boldsymbol{T}}{\partial \theta_1} = \begin{bmatrix} l_2(s_5(c_4s_1 + c_1c_2s_4) + c_1c_5s_2) - Ls_1 + d_3c_1s_2 + l_1c_1s_2 \\ Lc_1 - l_2\big(s_5(c_1c_4 - c_2s_1s_4)\big) - c_5s_1s_2) + d_3s_1s_2 + l_1s_1s_2 \\ 0 \end{bmatrix}$$

$$\boldsymbol{J}_L(:,2) = \frac{\partial \boldsymbol{T}}{\partial \theta_2} = \begin{bmatrix} l_2(c_2c_5s_1 - s_1s_2s_4s_5) + d_3c_2s_1 + l_1c_2s_1 \\ -l_2(c_1c_2c_5 - c_1s_2s_4s_5) - d_3c_1c_2 - l_1c_1c_2 \\ -l_2(c_5s_2 + c_2s_4s_5) - d_3s_2 - l_1s_2 \end{bmatrix}$$

$$\boldsymbol{J}_L(:,3) = \frac{\partial \boldsymbol{T}}{\partial d_3} = \begin{bmatrix} s_1 s_2 \\ -c_1 s_2 \\ c_2 \end{bmatrix}$$

$$\boldsymbol{J}_L(:,4) = \frac{\partial \boldsymbol{T}}{\partial \theta_4} = \begin{bmatrix} l_2 s_5 (c_1 s_4 + c_2 c_4 s_1) \\ l_2 s_5 (s_1 s_4 - c_1 c_2 c_4) \\ -l_2 c_4 s_2 s_5 \end{bmatrix}$$

$$\boldsymbol{J}_L(:,5) = \frac{\partial \boldsymbol{T}}{\partial \theta_5} = \begin{bmatrix} -l_2 (c_5 (c_1 c_4 - c_2 s_1 s_4) + s_1 s_2 s_5) \\ -l_2 (c_5 (c_4 s_1 + c_1 c_2 s_4) - c_1 s_2 s_5) \\ -l_2 (c_2 s_5 + c_5 s_2 s_4) \end{bmatrix}$$

To find the Angular Jacobian $\boldsymbol{J}_A$ we used the Whitney method.
For each revolute joint i,the respective Jacobian column is the direction in which the joint turns about.

$$\boldsymbol{J}_A(:,i) = \widehat{\boldsymbol{u}}_i$$

The angular Jacobean for linear joints is the zero vector.
To find the direction for each revolute joint we used the rotation matrices computed in the direct kinematics found in 1.

$${}^{0}\boldsymbol{R}_i = {}^{0}\boldsymbol{R}_1 {}^{0}\boldsymbol{R}_1 \dots {}^{\mathrm{i-1}}\boldsymbol{R}_i$$

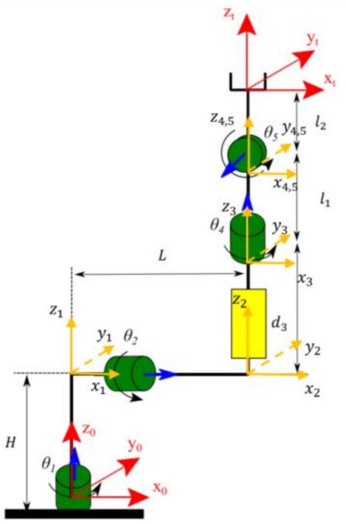
$$\widehat{\boldsymbol{u}}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\widehat{\boldsymbol{u}}_2 = {}^{0}\boldsymbol{R}_1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\widehat{\boldsymbol{u}}_3 = \boldsymbol{0}^{3\times1}$$

$$\widehat{\boldsymbol{u}}_4 = {}^{0}\boldsymbol{R}_3 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\widehat{\boldsymbol{u}}_5 = {}^{0}\boldsymbol{R}_4 \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$$



$$\boldsymbol{J}_A = [\widehat{\boldsymbol{u}}_1 \quad \widehat{\boldsymbol{u}}_2 \quad \widehat{\boldsymbol{u}}_3 \quad \widehat{\boldsymbol{u}}_4 \quad \widehat{\boldsymbol{u}}_5]$$

Plugging in the values

$$\boldsymbol{J}_A = \begin{bmatrix} 0 & c_1 & 0 & s_1 s_2 & c_1 s_4 + c_2 c_4 s_1 \\ 0 & s_1 & 0 & -c_1 s_2 & s_1 s_4 - c_1 c_2 c_4 \\ 1 & 0 & 0 & c_2 & -c_4 s_2 \end{bmatrix}$$

To find the Jacobian in the tools system coordinates we multiplied both linear and angular Jacobians by the rotation matrix ${}^{\mathrm{t}}\boldsymbol{R}_0$

$$\boldsymbol{J}_L^t = {}^{\mathrm{t}}\boldsymbol{R}_0 \boldsymbol{J}_L$$

$$\boldsymbol{J}_A^t = {}^{\mathrm{t}}\boldsymbol{R}_0 \boldsymbol{J}_A$$

$${}^{\mathrm{t}}\boldsymbol{R}_0 = {}^{0}\boldsymbol{R}_t^{\,T} = \begin{bmatrix} c_5(c_1c_4 - c_2s_1s_4) + s_1s_2s_5 & c_5(c_4s_1 + c_1c_2s_4) - c_1s_2s_5 & c_2s_5 + c_5s_2s_4 \\ -c_1s_4 - c_2c_4s_1 & c_1c_2c_4 - s_1s_4 & c_4s_2 \\ c_5s_1s_2 - s_5(c_1c_4 - c_2s_1s_4) & -s_5(c_4s_1 + c_1c_2s_4) - c_1c_5s_2 & c_2c_5 - s_2s_4s_5 \end{bmatrix}$$

4 Singular Robot States

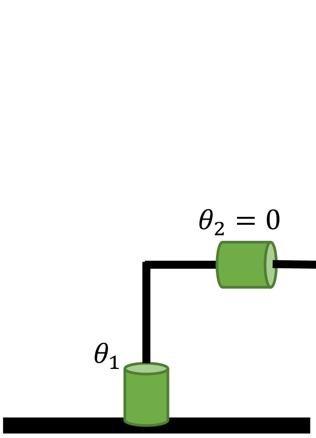
We assume $l_1 = l_2 = 0[m]$ and $\theta_1 = \theta_2 = 0°$.

The singular states of the robot are obtained by joint values that nullify the robot's Jacobean determinant. As the assumptions make our robot a de-facto smaller version of itself, we will be looking only on $\boldsymbol{J}_L(1{:}3{,}1{:}3)$.

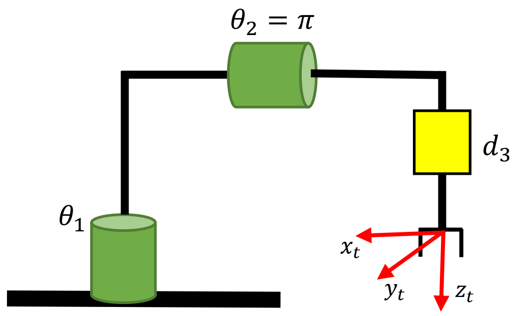
$$\det\left(\boldsymbol{J}_L(1{:}3{,}1{:}3)\right) = -d_3^2 s_2$$

This expression equates to zero when either $d_3 = 0$ or $\theta_2 = \pi k, \quad k \epsilon Z$.

For $\theta_2 = \pi k$, the robot can take the following two forms:



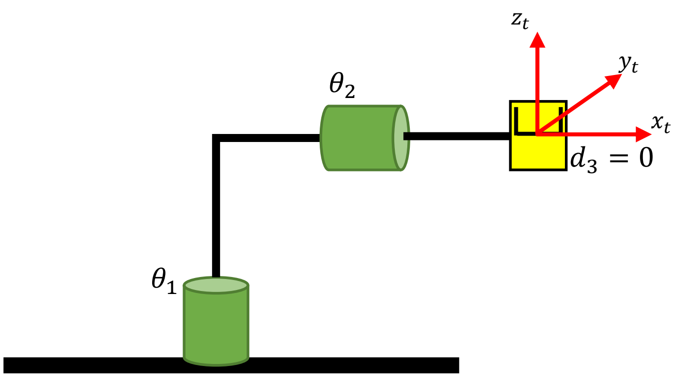
θ_2=0



$\theta_2 = \pi$

In these two positions, the robot movement in the x_t direction is impossible.
Changing d_3 will result in a movement in the z_t direction, while changing either θ_1 or θ_2 will move the tool in the y_t direction.

For $d_3 = 0$, the robot will appear as follows:



$d_3 = 0$

Here the motion is impossible in the x_t direction when θ_2 has no effect on the location of the tool.

We would like to note that when both $d_3 = 0$ **and** $\theta_2 = \frac{\pi}{2}$ two degrees of freedom will fall and the robot won't be able to create tool movement in both the x_t and y_t directions. We can prove that by plugging in said values to the linear Jacobean in the tool co-ordinate system:

$$J_L^t = \begin{bmatrix} d_3 s_2 & 0 & 0 \\ Lc_2 & -d_3 & 0 \\ -Ls_2 & 0 & 1 \end{bmatrix} \rightarrow \{d_3 \theta_2 = \pi/2\} \rightarrow J_L^t = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -L & 0 & 1 \end{bmatrix}$$

The first two rows of the Jacobian are zero making the motion in x_t, y_t impossible. Also, the second column is zero which prevents changes in θ_2 to affect the tool's location.

5 Motion Planning

Provided the constant lengths of the robot:

$$H = 0.2[m] \quad L = 0.1[m] \quad l_1 = l_2 = 0[m]$$

And movement constraints:

$$\theta_1 \epsilon [-\pi, \pi] \quad \theta_2 \epsilon [-\frac{\pi}{2}, \frac{\pi}{2}] \quad d_3 > 0$$

We were asked to move the tool from one location to the other, ignoring tool orientation, in three different trajectories: constant velocity, trapezoid velocity and polynomial with zero velocity and acceleration at the end points. The motion is to take $T = 2[s]$.

$$\boldsymbol{x}_0 = \begin{bmatrix} 0.1 \\ 0.05 \\ 0.25 \end{bmatrix} \rightarrow \boldsymbol{x}_f = \begin{bmatrix} 0.25 \\ -0.15 \\ 0.35 \end{bmatrix}$$

Analytical trajectory profiles were computed independently for each axis in the 3-D world.

$$[\boldsymbol{x}, \dot{\boldsymbol{x}}, \ddot{\boldsymbol{x}}] = f(Trajectory)$$

Inverse kinematics was then used to compute the joint position for each tool world position with ambiguity in the form of *elbows* being decided by movement constraints.

$$\boldsymbol{q} = InverseKinematics(\boldsymbol{x}, elbows)$$

Joint velocities and acceleration were computed in two methods:

- Differentiating the joint positions:

$$\dot{\boldsymbol{q}} = \frac{d\boldsymbol{q}}{dt}$$

$$\ddot{\boldsymbol{q}} = \frac{d\dot{\boldsymbol{q}}}{dt}$$

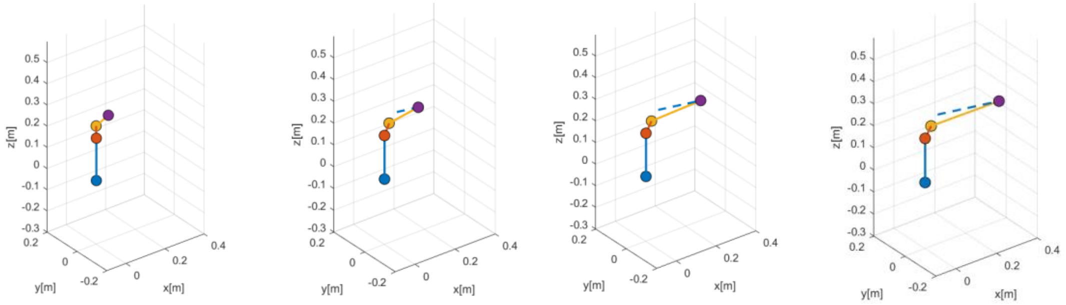
- The use of the linear Jacobean computed analytically

$$\dot{\boldsymbol{q}} = J\dot{\boldsymbol{x}}$$

$$\ddot{\boldsymbol{q}} = J\ddot{\boldsymbol{x}} + \dot{J}\dot{\boldsymbol{x}}$$

In the subsequent sub-sections, an analytical motion trajectory is derived for each of the required profiles, followed by graphic outputs of its results.
The last sub-section shows graphic comparison between the motion profiles.

Note: the joint constraints allowed for two possible θ_2 solutions, we chose the positive one for the inverse kinematics.



5.1 Constant Velocity

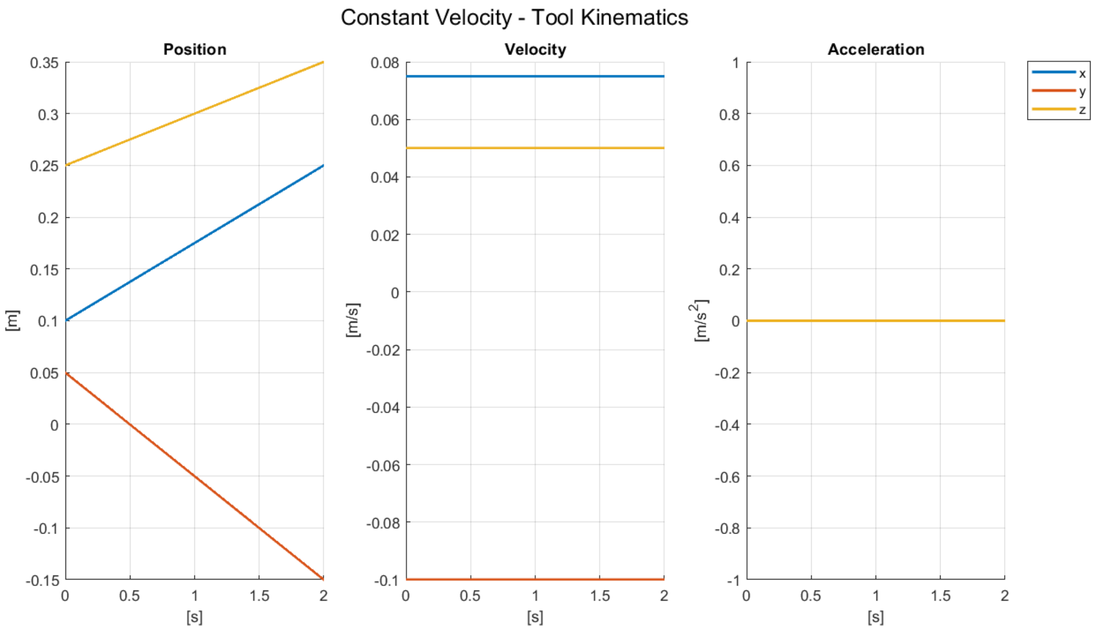
Provided that the velocity is constant between two points $[x_1, x_2]$ traversed in time T, the kinematics are:

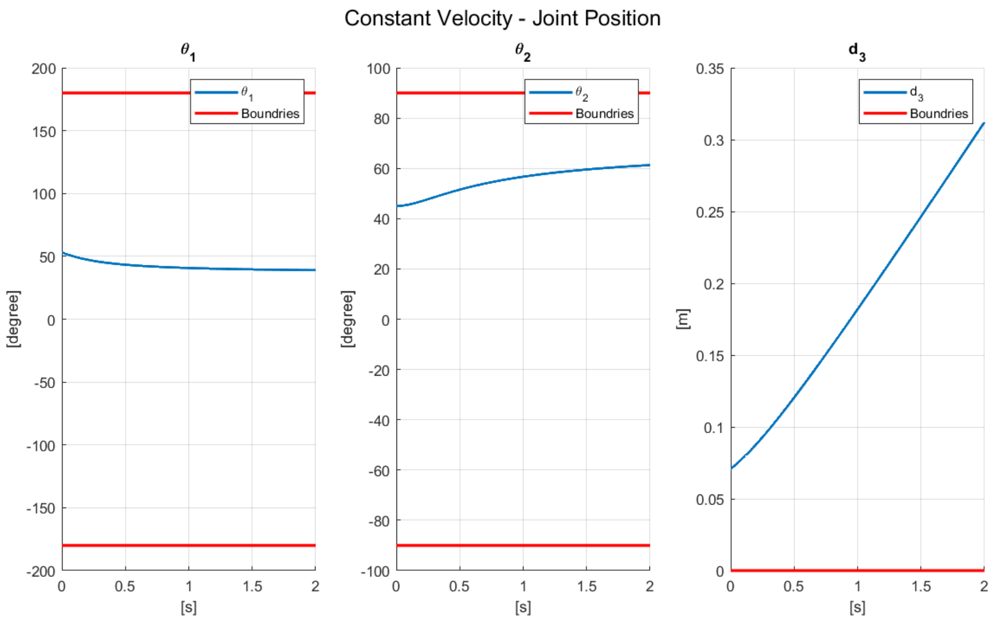
$$a = 0$$

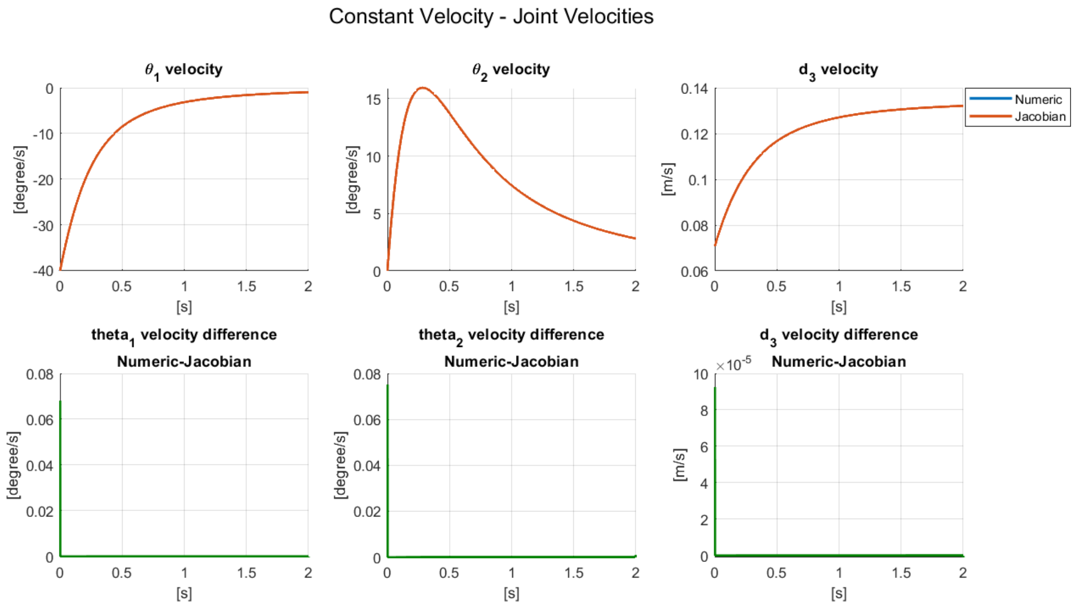
$$v = \frac{x_2 - x_1}{T}$$

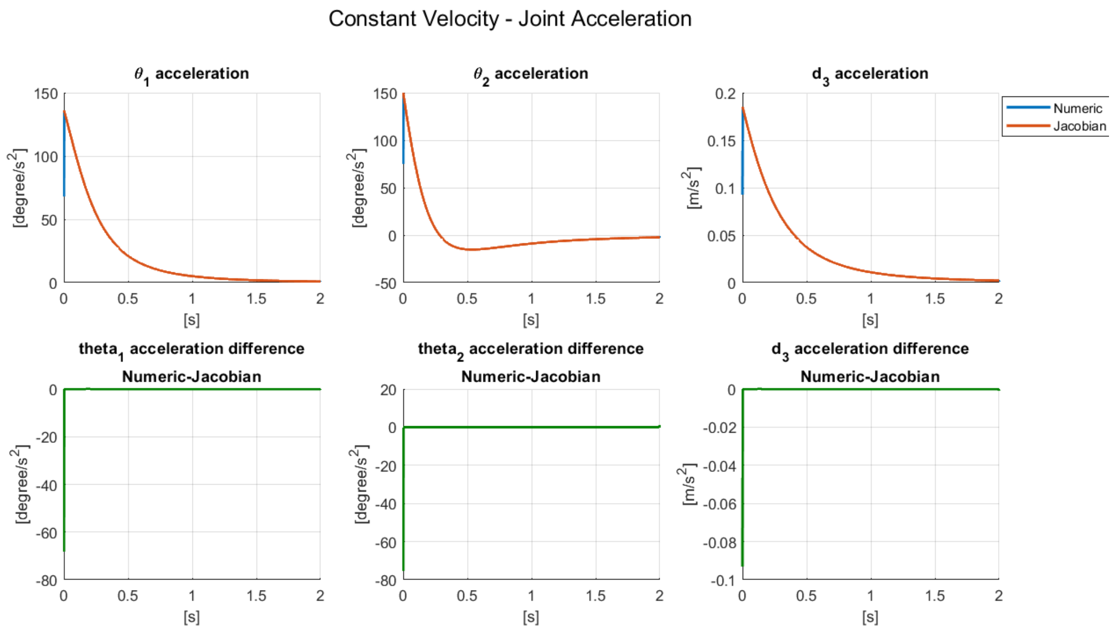
$$x = v \cdot t + x_1 \quad t\epsilon[0, T]$$

5.1.1 Graphic Results




Constant Velocity - Joint Position
θ_1
θ_2
d_3
Boundries
[degree]
[m]
[s]


Constant Velocity - Joint Velocities
θ_1 velocity
θ_2 velocity
d_3 velocity
Numeric
Jacobian
theta$_1$ velocity difference
Numeric-Jacobian
theta$_2$ velocity difference
Numeric-Jacobian
d_3 velocity difference
Numeric-Jacobian
$\times 10^{-5}$
[degree/s]
[m/s]
[s]

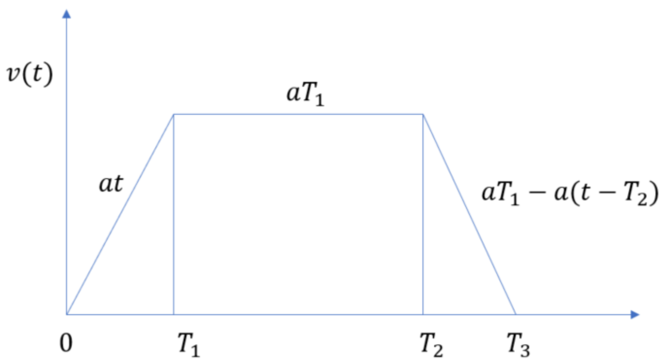


5.2 Trapezoid Velocity

Symmetrical trapezoid velocity movement can be described by as little as three parameters:

- Total Time,
- Zero Acceleration Time
- Amount Traveled

We derivate the acceleration on the velocity ramp-up from these properties.



$$T_1 = T_3 - T_2 = \Delta T$$

$$a(t) = \begin{cases} a & t \le T_1 \\ 0 & T_1 < t \le T_2 \\ -a & T_2 < t \le T_3 \end{cases}$$

$$v(t) = \begin{cases} a\,t & t \le T_1 \\ aT_1 & T_1 < t \le T_2 \\ aT_1 - a\,(t - T_2) & T_2 < t \le T_3 \end{cases}$$

$$x(t) = \begin{cases} \dfrac{a\,t^2}{2} & t \le T_1 \\ \underbrace{\dfrac{a\,T_1^2}{2}}_{x(T_1)} + aT_1(t - T_1) & T_1 < t \le T_2 \\ \underbrace{\dfrac{a\,T_1^2}{2} + aT_1(T_2 - T_1)}_{x(T_2)} + aT_1(t - T_2) - \dfrac{a\,(t - T_2)^2}{2} & T_2 < t \le T_3 \end{cases}$$

To solve for a we equate the final position $x(T_3)$ to a known number ΔX.

$$\frac{aT_1^2}{2} + aT_1(T_2 - T_1) + \frac{aT_1(T_3 - T_2)}{2} = \Delta X$$

$$a\left(T_1^2 + 2T_1(T_2 - T_1) + T_1(T_3 - T_2)\right) = 2\Delta X$$

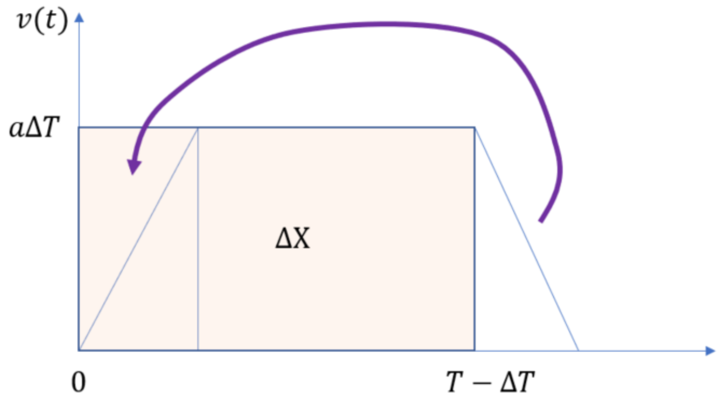
Subsisting $T_3 = T$, and $\Delta T = T_1 = T_3 - T_2$

$$a(\Delta T^2 + 2\Delta T(T - 2\Delta T) + \Delta T^2) = 2\Delta X$$

$$a(2\Delta T^2 + 2\Delta T \cdot T - 4\Delta T^2) = 2\Delta X$$

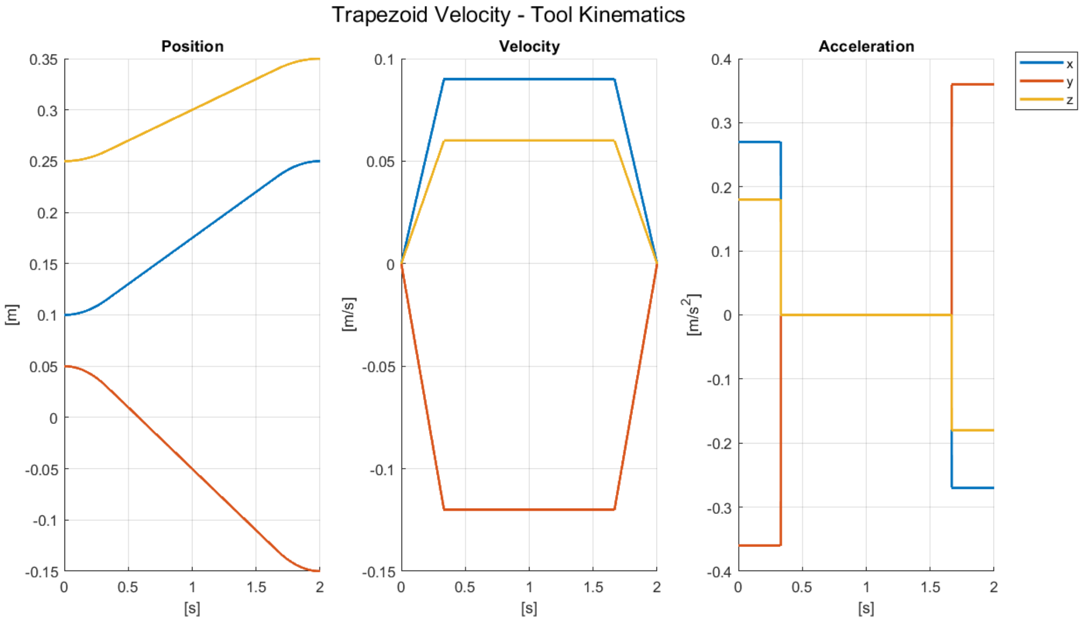
$$a = \frac{\Delta X}{\Delta T(T - \Delta T)}$$

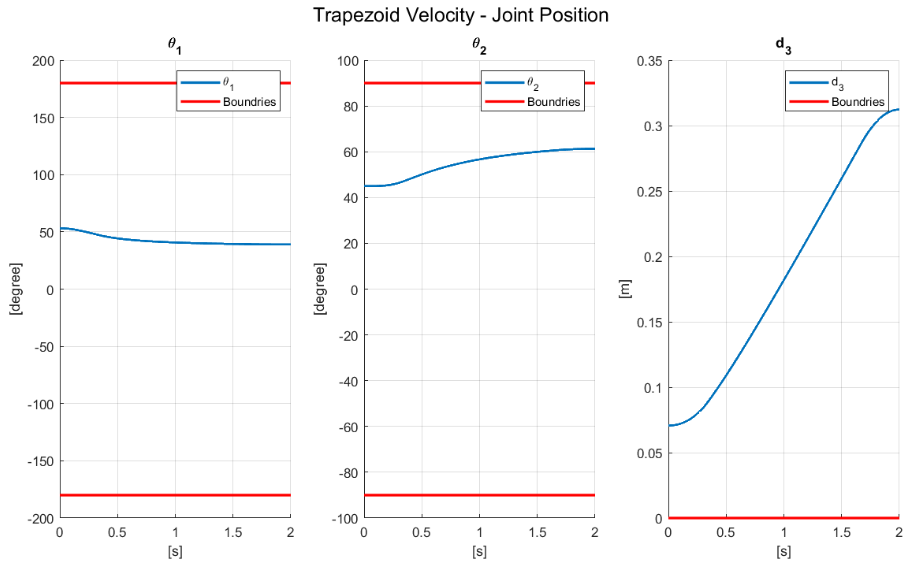
This result sits well with integration by 'rectangular-ing' the trapezoid:



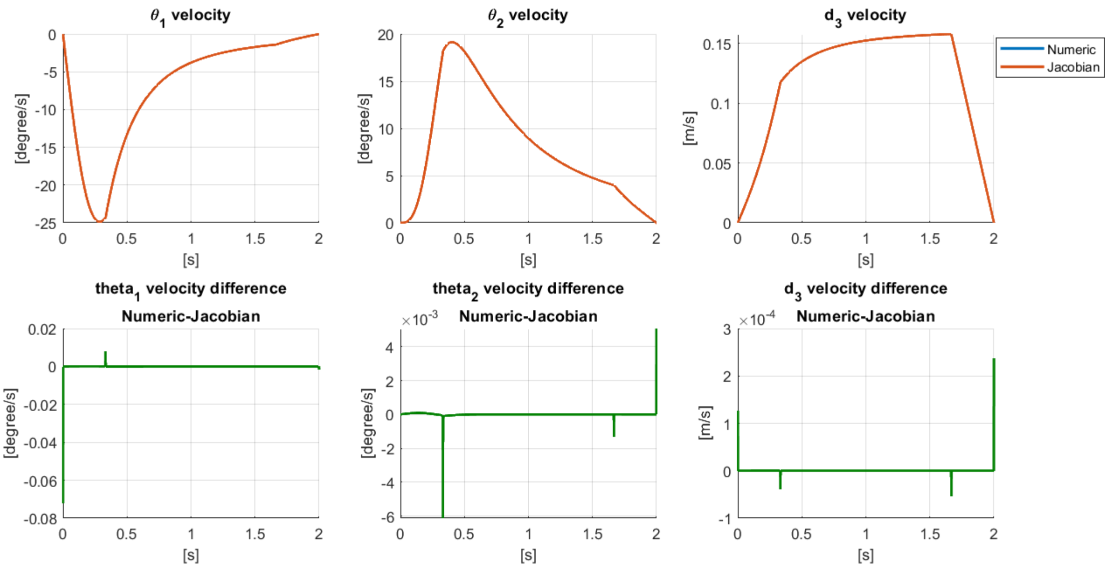
In our assignment $\Delta T = \frac{1}{6}T$.

5.2.1 Graphic Results

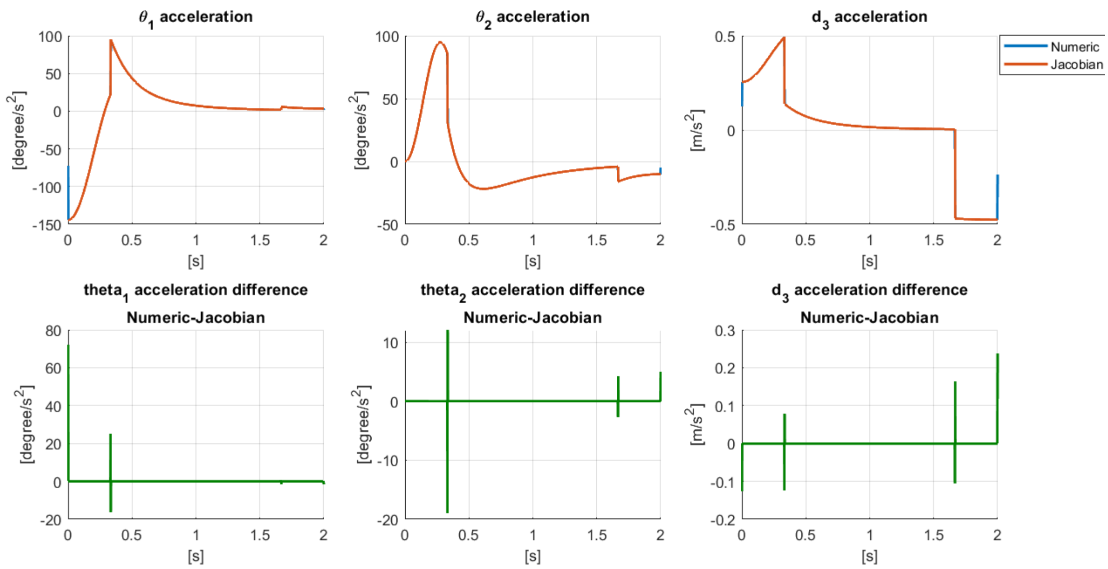




Trapezoid Velocity - Joint Velocities



Trapezoid Velocity - Joint Acceleration



5.3 Polynomial Profile - Minimum Jerk

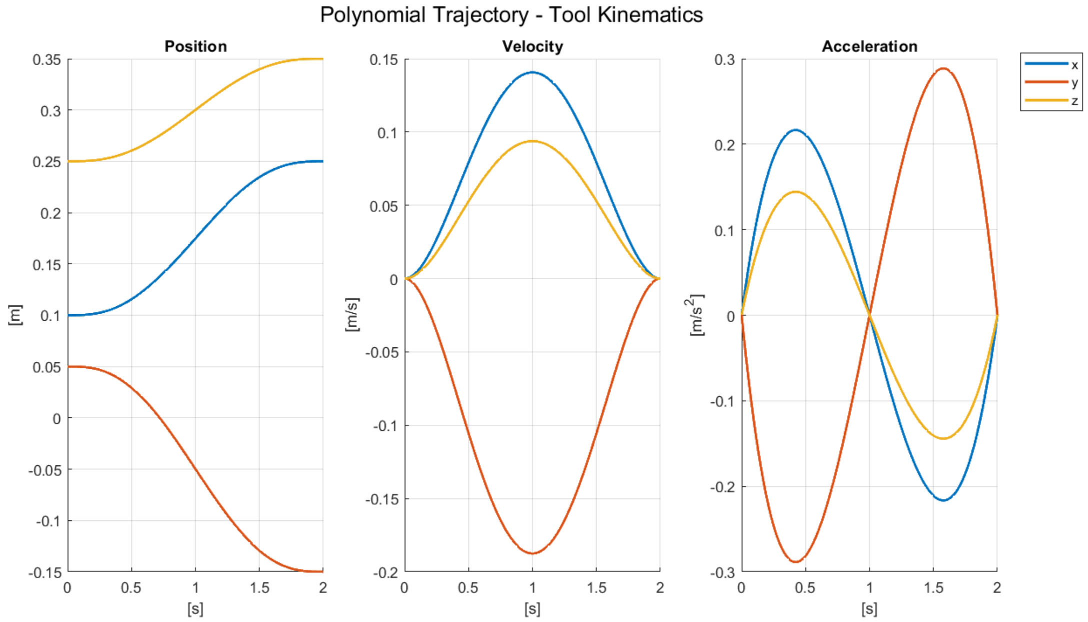
In their paper 'A Minimum-Jerk Trajectory' Kyriakopoulos et al. derivate a well-known 1-D polynomial trajectory which minimizes the motion's jerk [1]. This trajectory also offers zero velocity and acceleration at the end points.

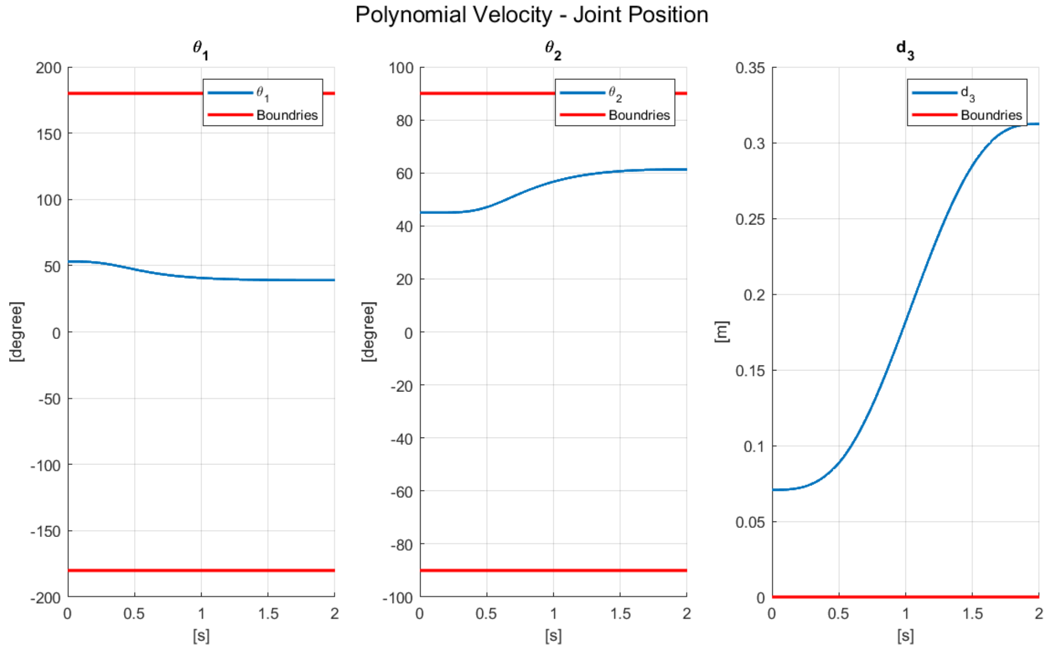
$$x(t) = x_i + (x_f - x_i)\left(10\left(\frac{t}{T}\right)^3 - 15\left(\frac{t}{T}\right)^4 + 6\left(\frac{t}{T}\right)^5\right)$$

$$v(t) = (x_f - x_i)\left(\frac{30t^2}{T^3} - \frac{60t^3}{T^4} + \frac{30t^4}{T^5}\right)$$

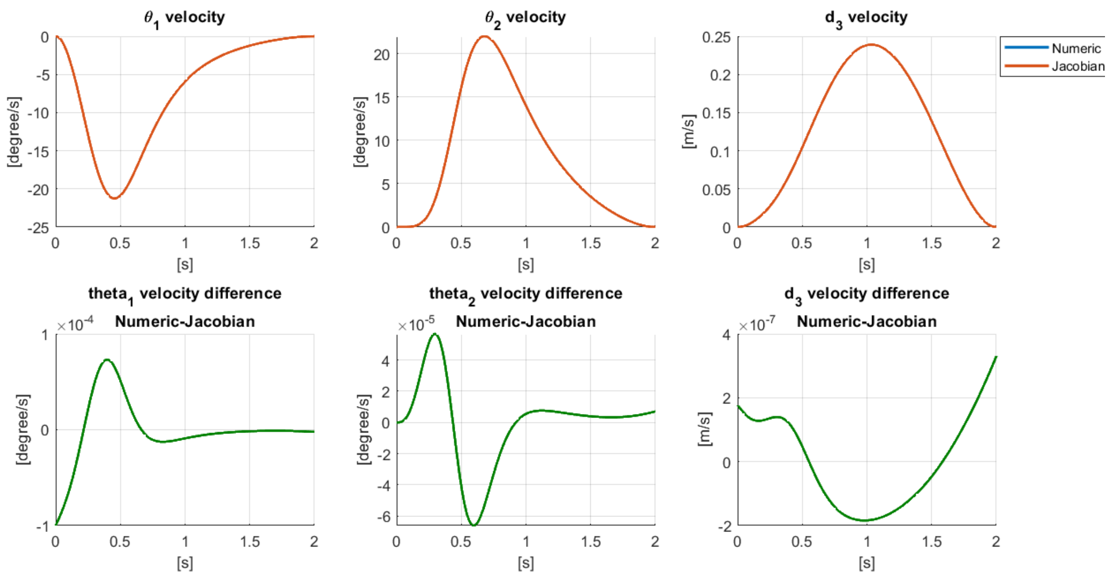
$$a(t) = (x_f - x_i)\left(\frac{60t}{T^3} - \frac{180t^2}{T^4} + \frac{120t^3}{T^5}\right)$$

5.3.1 Graphic Results

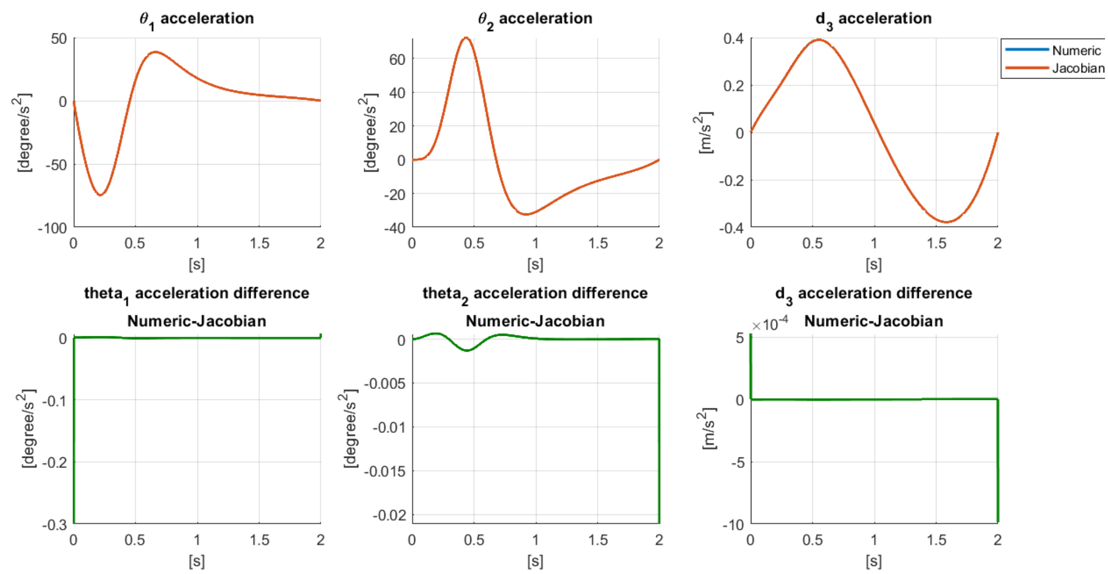




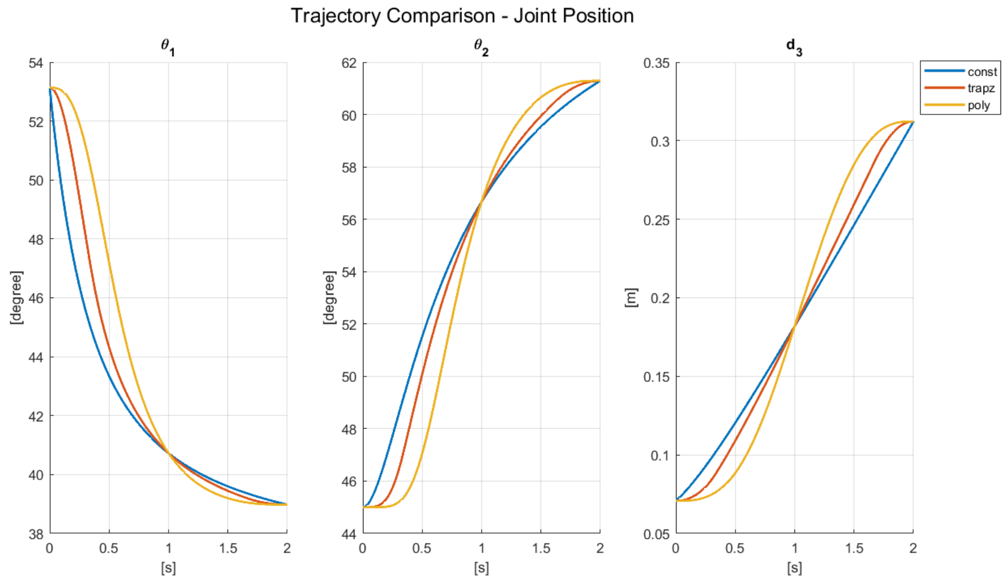
Polynomial Velocity - Joint Velocities

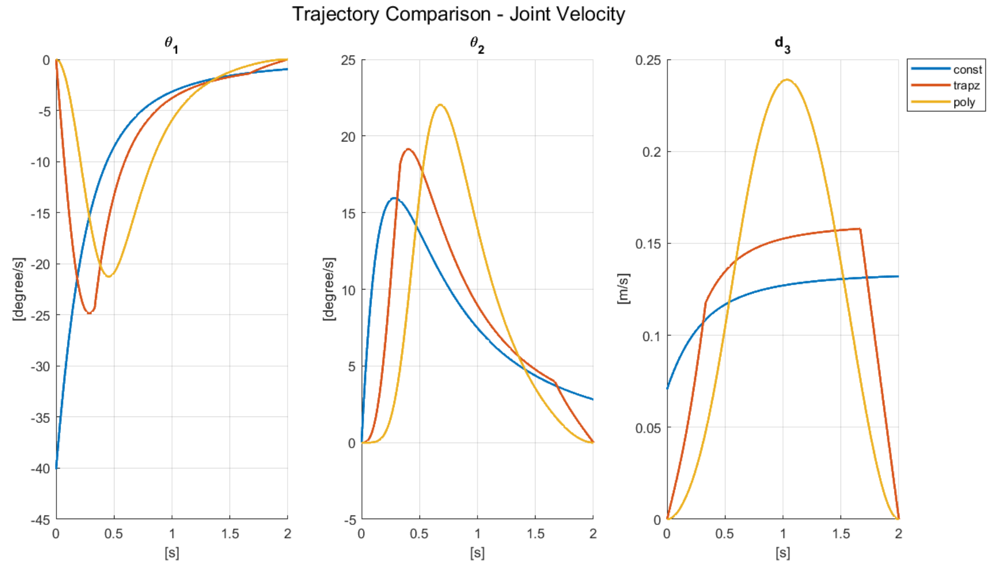


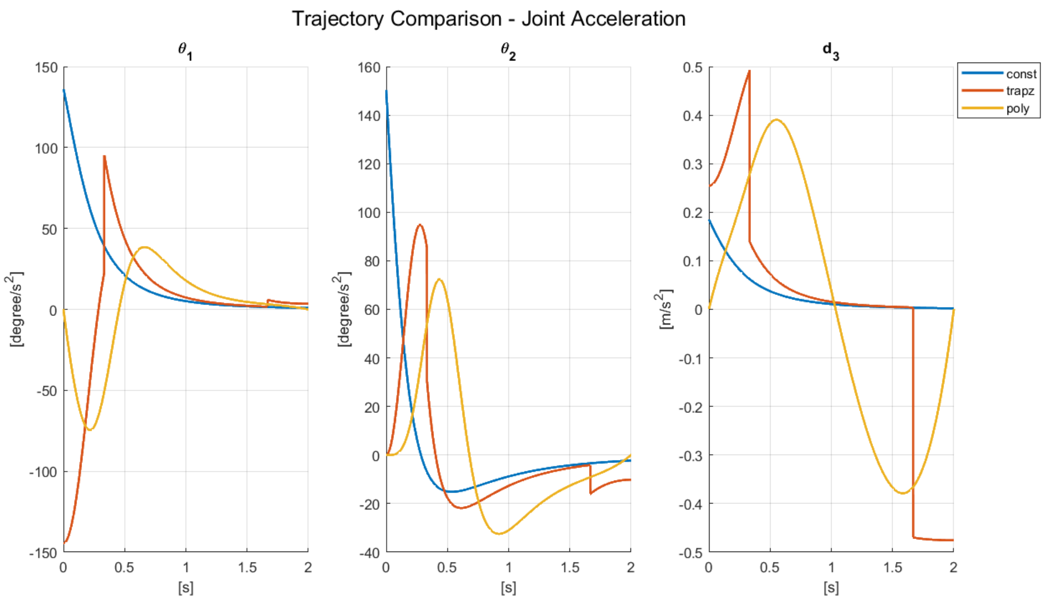
Polynomial Velocity - Joint Acceleration



5.4 Graphic Comparison Between Trajectories in Joint Space







6 References

[1] K. J. Kyriakopoulos and G. N. Saridis, "Minimum jerk trajectory planning for robotic manipulators," *Coop. Intell. Robot. Sp.*, vol. 1387, p. 159, 1991, doi: 10.1117/12.25421.