Trajectory planning

Félix Monasterio-Huelin, Álvaro Gutiérrez & Blanca Larraga

September 6, 2022

Contents

List of Figures

List of Tables

1 Introduction

Trajectory planning is defined as the representation of the generalized coordinates as functions of time, $q_i(t)\,\forall i = 1, 2, \cdots, n$, in such a way that they satisfy certain conditions for a task implementation [1]. Typically, the tasks to be solved by a manipulator are defined in the Cartesian space, that is, as the temporal evolution of position and orientation of the hand's end, $\{d_0^n(t), R_0^n(t)\}$.

A manipulator robot is not only a mechanism with a mobility defined by its DOFs, but a complex system made up of actuators that allow to implement controlled movements. In this document, it is supposed that a closed-loop control system exists for every DOF of the robot. Moreover, it is assumed that the controller satisfies the reference signal tracking, that is, that the output $q_i(t)$ of every closed-loop control system matches, in the steady state, the predefined reference signal, $q_{ri}(t)$.

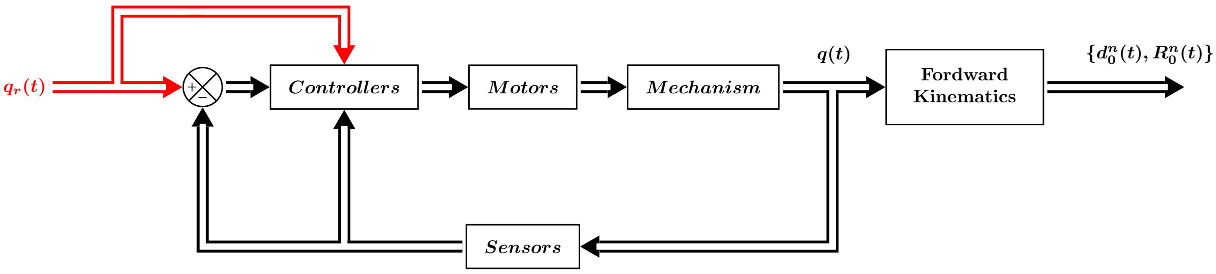


Figure 1.1: Depiction of a closed-loop control system

Figure 1.1 shows the typical block diagram of a closed-loop control system of robot manipulators. Double lines correspond to vectorial inputs or outputs. The objective of this kind of controllers is that $q(t) \to q_r(t)$ for every continuous time step t. This objective is not physically possible, because it will oblige to have an instantaneous response. Therefore, controllers are designed to allow this convergence in a small transient state. It is not possible to solve a control problem for every reference signal. Hence, they are typically designed to solve finite-grade polynomial reference signals, that is, reference signals of the form $q_{ri}(t) = a_{i0} + a_{i1}t + a_{i2}t^2 + \cdots + a_{ip}t^p$, where p is a design parameter; typically $p \leq 3$.

The objective of this document is to define reference signals that allow a smooth movement of the robot. For commodity in the notation and if it does not imply confusion, subindex r will be eliminated.

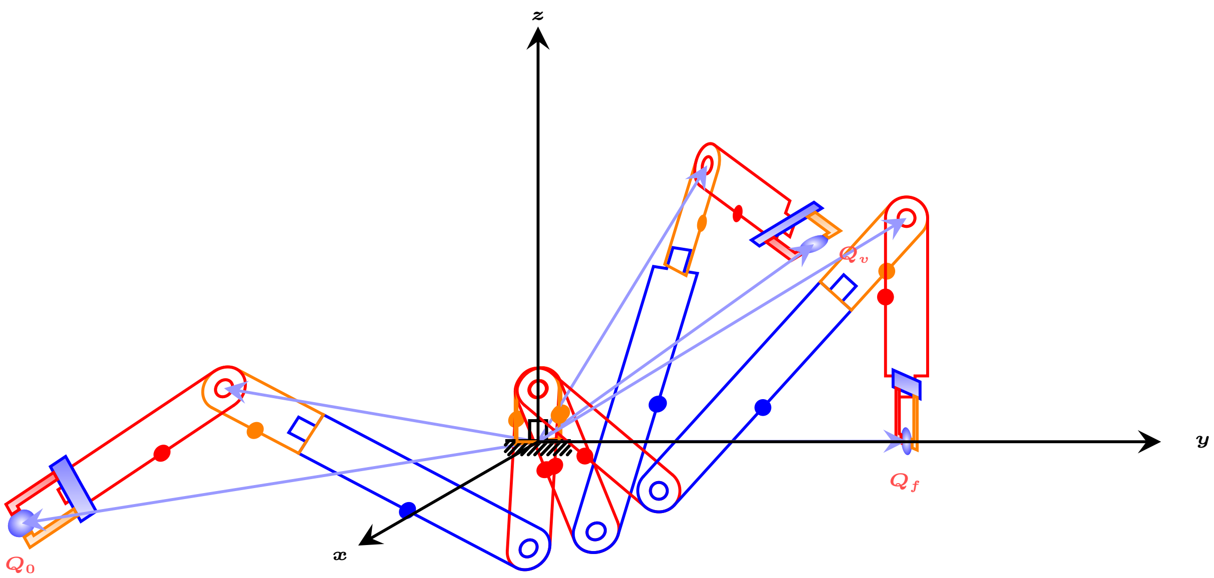


Figure 1.2: Robot in three different positions and orientations: $\{Q_0, R_0)\}$: Initial; $\{Q_v, R_v\}$: Intermediate; $\{Q_f, R_f\}$: Final

Figure 1.2 shows a 6 DOFs robot in three different positions and orientations. The idea is to generate a trajectory in the time domain for every generalized coordinate. The trajectory should move the robot from one position and orientation to another: $\{Q_0, R_0\} \Rightarrow \{Q_v, R_v\} \Rightarrow \{Q_f, R_f\}$.

There are two main ways of solving the trajectory planning problem. The first one consists of defining the trajectory at the hand's end, as shown in Figure 1.3.

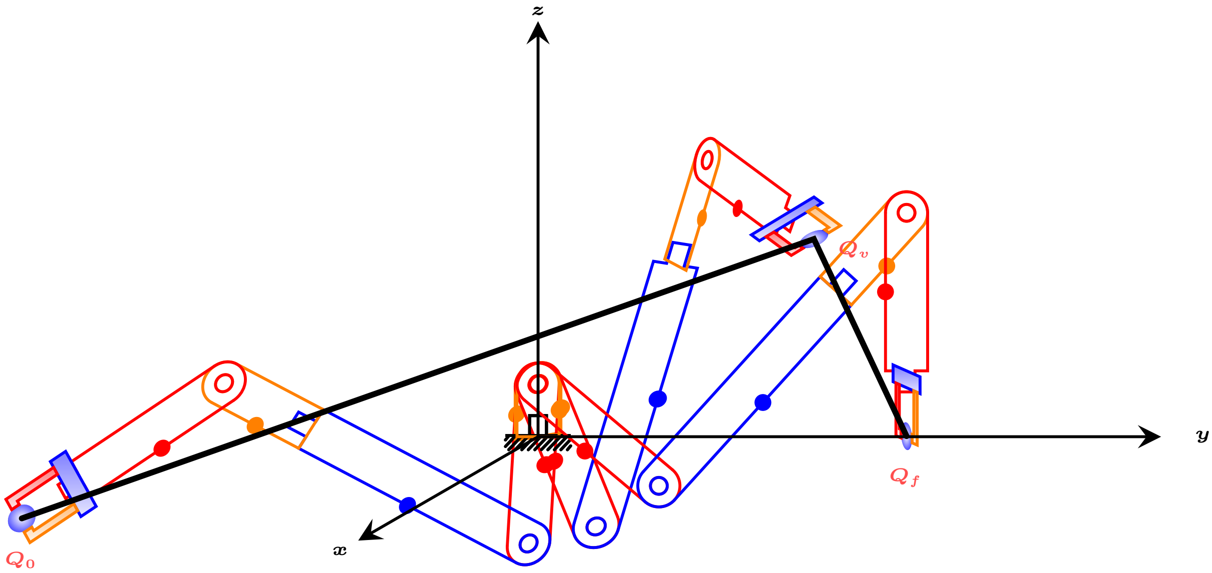


Figure 1.3: Trajectory of the hand's end on the Cartesian space.

This method obliges to solve the inverse kinematics problem for every time step, what forces to use powerful hardware and sophisticated controllers, because there is no warranty that the reference signals of the control system are smooth. In any case, there exist controller design techniques based on Cartesian reference signals that do not belong to the block diagram shown on Figure 1.1.

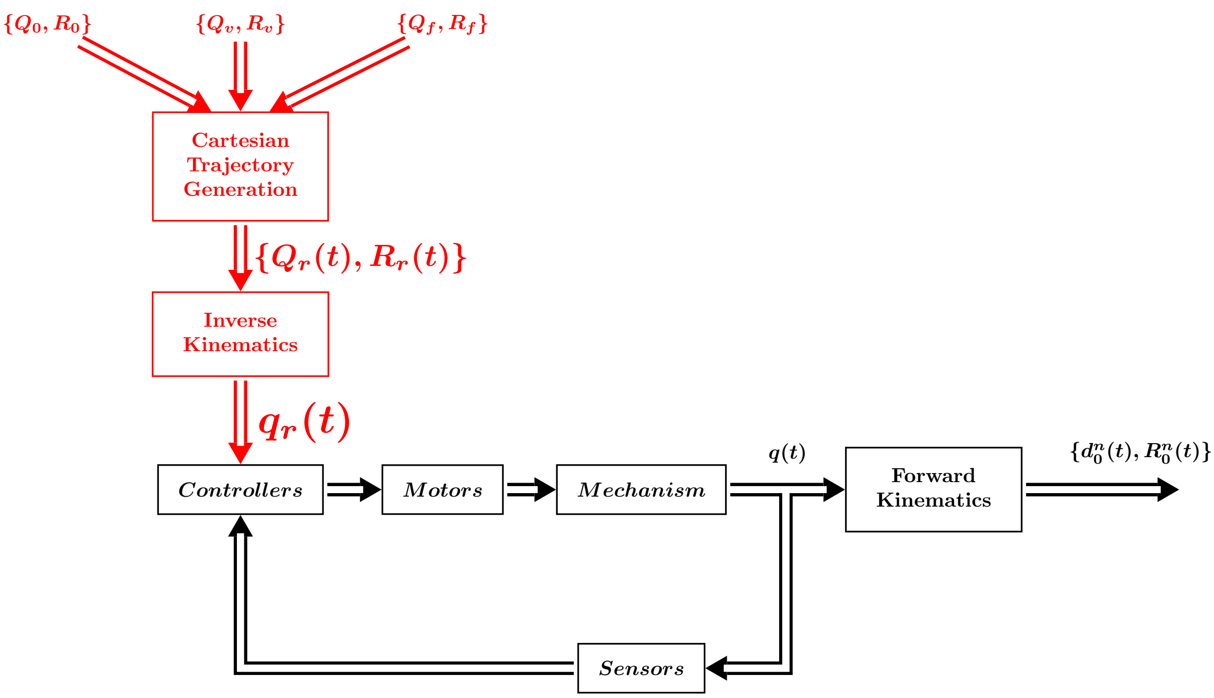


Figure 1.4: Block diagram of trajectory planning in the Cartesian space.

Figure 1.4 shows a block diagram with a trajectory planning in the Cartesian space. However, this method will not be used in this document. Nonetheless, it is convenient to understand that in some applications it is mandatory to fix a trajectory at the Cartesian space, at least, related to the hand's orientation. For example, when an object, as a glass of water, must be moved from one place to another without pouring the liquid.

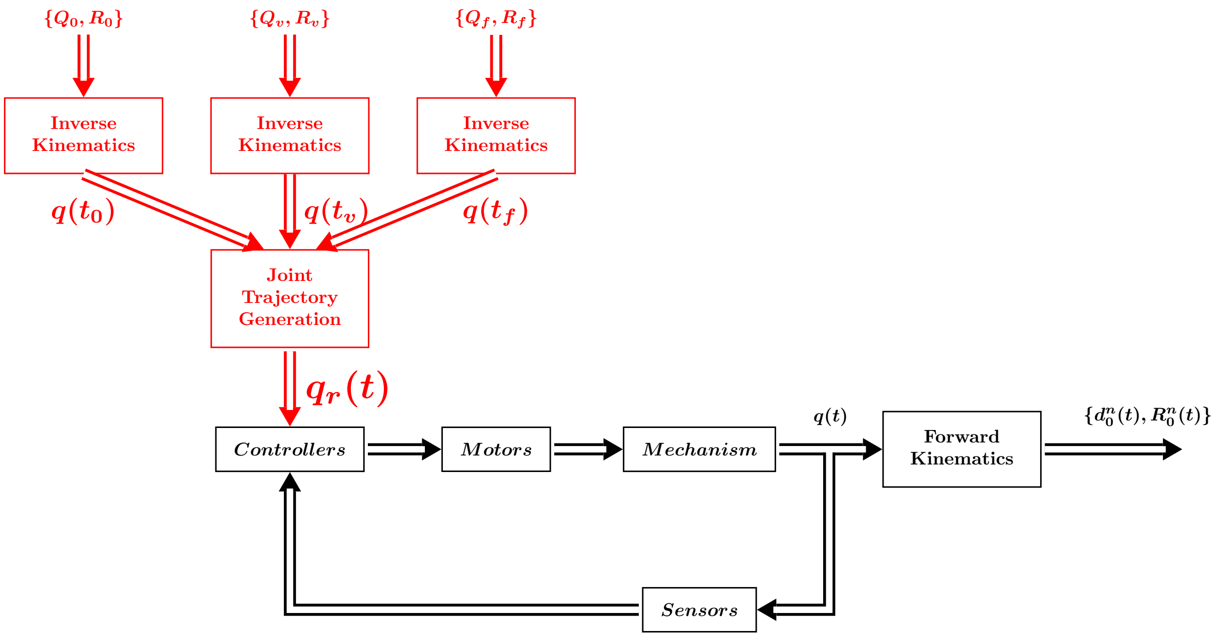
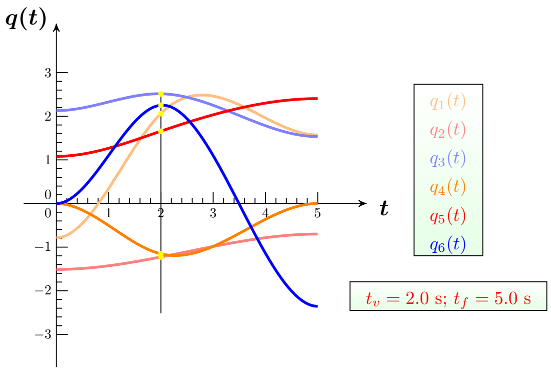
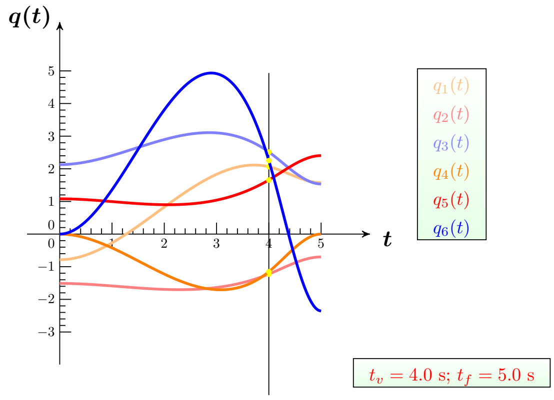


Figure 1.5: Block diagram of trajectory planning in the joints space.

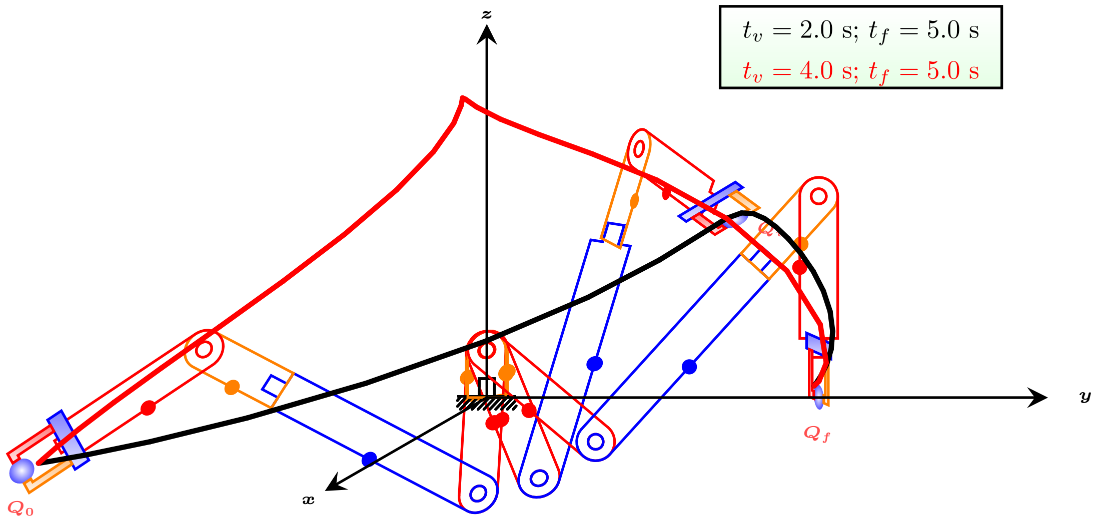
The second method, that will be tackled in this document, consists of defining the trajectories on the joints space. These trajectories must satisfy the condition for the robot to pass through certain predefined positions and orientations. Figure 1.5 shows a block diagram with a trajectory planning in the joints space. With this method, it is only necessary to solve the inverse kinematics problem for some predefined positions and orientations. Results of this problem will serve as contour conditions for the smooth trajectory planning.



(a) Generalized coordinates.



(b) Generalized coordinates.



(c) Trajectories on the Cartesian space.

Figure 1.6: Cubic trajectories with intermediate point in the joints space.

Figure 1.6 shows the generated curves for every joint $q_i(t)$ of a six DOFs robot and its trajectory in the Cartesian space. With this method no restriction about the hand's end position and orientation is taken into account, except a set of predefined positions and orientations.

The graphics of Figure 1.6a and Figure 1.6b are generated by following the technique explained in Section 3. With this technique, it is necessary to specify the time instant at which the robot is in the intermediate point (t_v) and the time instant at which the robot is in the final point (t_f). Figure 1.6c shows two trajectories generated in the Cartesian space: one when $t_v = 2\,s$ and the other when $t_v = 4\,s$. The trajectories at the Cartesian space are obtained by solving the forward kinematics problem from the generated trajectories in the joint space $q_r(t)$. As the generated functions $q_i(t)$ are polynomial functions of grade 3, and therefore smooth signals, that is, derivable in every time step, the trajectories generated in the Cartesian space will also be smooth, because the forward kinematics represents a continuous and derivable vectorial function. On the other hand, the non derivable point shown in Figure 1.6c for $t_v = 4$, which shows a change on the trajectory direction, can only be explained as numerical errors.

In practice, there are limitations on the variation range of the joint coordinates because of mechanical limitations. The techniques developed in this document for the trajectory planning of the joint do not have this restriction into account. Figure 1.6a and Figure 1.6b show that the movements requested when $t_v = 4\,s$ require a higher range of movement than for $t_v = 2$ s. Therefore, the selection of t_v is very important for a practical application. However, in this document only the case for one intermediate point is studied, although the methodology can be used for obtaining more than one intermediate point.

2 Cubic trajectory planning with initial and end points defined

The problem to solve in this Section is to define the generalized coordinates as cubic polynomials time dependent,

$$q_i(t) = a_{i0} + a_{i1}t + a_{i2}t^2 + a_{i3}t^3 \tag{2.1}$$

subject to known restrictions

$$\mathcal{C} = \{t_0, t_f, q_i(t_0), q_i(t_f), \dot{q}_i(t_0), \dot{q}_i(t_f)\} \tag{2.2}$$

This problem conducts to the following linear equations' system:

$$q_i(t_0) = a_{i0} + a_{i1}t_0 + a_{i2}t_0^2 + a_{i3}t_0^3 \tag{2.3a}$$

$$q_i(t_f) = a_{i0} + a_{i1}t_f + a_{i2}t_f^2 + a_{i3}t_f^3 \tag{2.3b}$$

$$\dot{q}_i(t_0) = a_{i1} + 2a_{i2}t_0 + 3a_{i3}t_0^2 \tag{2.3c}$$

$$\dot{q}_i(t_f) = a_{i1} + 2a_{i2}t_f + 3a_{i3}t_f^2 \tag{2.3d}$$

This system can be represented in matrix form as

$$g = Ta \tag{2.4}$$

where

$$g = \begin{bmatrix} q_i(t_0) \\ q_i(t_f) \\ \dot{q}_i(t_0) \\ \dot{q}_i(t_f) \end{bmatrix} \tag{2.5a}$$

$$T = \begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_0 & 3t_0^2 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \tag{2.5b}$$

$$a = \begin{bmatrix} a_{i0} \\ a_{i1} \\ a_{i2} \\ a_{i3} \end{bmatrix} \tag{2.5c}$$

The parameters of the cubic polynomial can be obtained by inverting matrix T,

$$a = T^{-1}g \tag{2.6}$$

In the following subsection this problem is solved for a particular case, when the initial and final velocities are zero. Moreover, a discretized form of the problem is solved for its programming in a computer or controller.

2.1 Zero initial and final velocity

If $t_0 = 0$ and the robot is initially stopped, that is $\dot{q}_i(0) = 0$, in a known configuration , $q_i(0)$ for $i = 1, 2, \cdots n$, and the condition of being stopped at the final time t_f is imposed, that is $\dot{q}_i(t_f) = 0$, in a known configuration, $q_i(t_f)$ for $i = 1, 2 \cdots n$, then, the set of restrictions is

$$\mathcal{C} = \{0, t_f, q_i(0), q_i(t_f), 0, 0\} \tag{2.7}$$

Therefore,

$$q_i(0) = a_{i0} \tag{2.8a}$$

$$\dot{q}_i(0) = 0 = a_{i1} \tag{2.8b}$$

$$q_i(t_f) = a_{i0} + a_{i1}t_f + a_{i2}t_f^2 + a_{i3}t_f^3 \tag{2.8c}$$

$$\dot{q}_i(t_f) = 0 = a_{i1} + 2a_{i2}t_f + 3a_{i3}t_f^2 \tag{2.8d}$$

And in matrix form,

$$\begin{bmatrix} 1 & t_f \\ 2 & 3t_f \end{bmatrix} \begin{bmatrix} a_{i2} \\ a_{i3} \end{bmatrix} = \frac{q_i(t_f) - q_i(0)}{t_f^2} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{2.9}$$

Then, the following solution is obtained:

$$\begin{bmatrix} a_{i2} \\ a_{i3} \end{bmatrix} = \frac{q_i(t_f) - q_i(0)}{t_f^2} \begin{bmatrix} 1 & t_f \\ 2 & 3t_f \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{q_i(t_f) - q_i(0)}{t_f^3} \begin{bmatrix} 3t_f & -t_f \\ -2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{2.10}$$

In summary,

$$\begin{bmatrix} a_{i0} \\ a_{i1} \\ a_{i2} \\ a_{i3} \end{bmatrix} = \begin{bmatrix} q_i(0) \\ 0 \\ 3\alpha_i t_f \\ -2\alpha_i \end{bmatrix} \tag{2.11}$$

where

$$\alpha_i = \frac{q_i(t_f) - q_i(0)}{t_f^3} \tag{2.12a}$$

Therefore,

$$\boxed{q_i(t) = q_i(0) + 3\alpha_i t_f t^2 - 2\alpha_i t^3} \tag{2.13}$$

If the solution is programmed in a computer, it must be discretized. Therefore, $t = kT$, with $k = 0, 1, 2, \cdots$ where T is a constant sampling period.

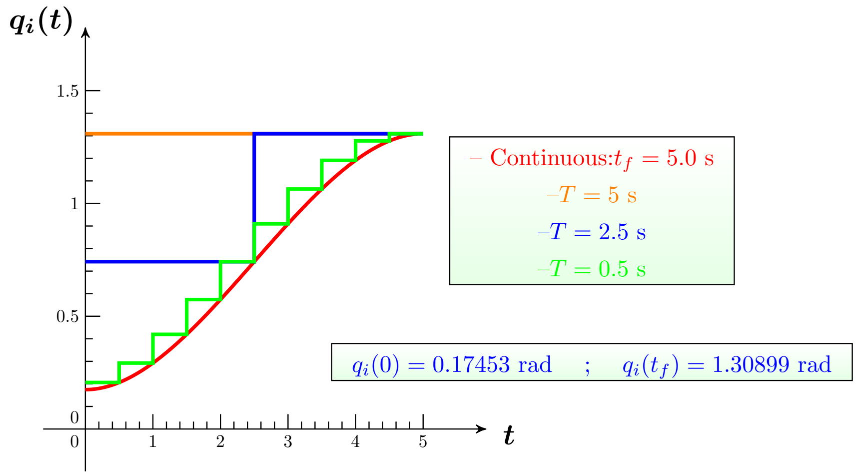


Figure 2.1: Discretization in advance.

Figure 2.1 shows an example of cubic trajectory and its discretization in advance, for different values of the sampling period T. In what follows, the solution for a discretization in advance is shown.

Let $t_f = NT$, where $N \in \mathbb{N}$. Therefore, if $t = kT$ where $k = 0, 1, 2, \cdots, N$,

$$q_i(k) = q_i(0) + \beta_i(3N - 2k)k^2 \tag{2.14}$$

where

$$\beta_i = \frac{q_i(N) - q_i(0)}{N^3} = T^3 \alpha_i \tag{2.15}$$

It is important to assume that the sampling, because of causality reasons, obliges to define the interval where the value is hold, $[k, k+1)$, with $k = 1, 2, \cdots, N$, closed on the left and open on the right. However, it is possible to program an anticipation by considering the interval $(k-1, k]$, with

$k = 1, 2, \cdots, N$. This has sense in the trajectory planning because it is known. Moreover, it will be the controller reference signal, so the steady state response will not be instantaneous and there will be a time delay called settling time. In this way, it is warrantied that in every sampling time step the system has been controlled, that is, that the output for every joint has achieved its desired value $q_i(k)$. This last affirmation is true if the controller is an ideal one, but it is not the typical case. In any case, because there is an advanced reference, a delay of one sampling time at the output of the system is avoided.

For programming in a computer, the following expression, valid for every continuous interval $(kT, (k+1)T]$, can be implemented:

$$\boxed{q_i(k+1) = q_i(0) + \beta_i \left(3N - 2(k+1)\right)(k+1)^2, \quad k \in \{0, 1, \cdots, N-1\}} \tag{2.16}$$

This document does not focus on how to select the sampling period T, but its importance must be shown. If T is excessively high, related to the dynamic response of the system, a jumping motion will be observed. This depends on the type of controller programmed and the mechanical response of the robot. In general, it is desirable to select a small sampling period. Moreover, it can also be appreciated that because β_i is inversely proportional to N^3, the higher N the lower β_i and therefore $q_i(k) - q_i(0)$. This means that low T provides better approximations to the cubic polynomial.

3 Cubic trajectory planning with initial, intermediate and end points defined

The problem to solve in this Section is to define the generalized coordinates as cubic polynomials time dependent, by connecting smoothly intermediate points without the robot stopping at them.

In this document only the case of a single intermediate point at $t = t_v$ is shown. Therefore, the trajectory will be composed of the following cubic polynomial equations:

$$q_i(t) = \begin{cases} a_{i0} + a_{i1}t + a_{i2}t^2 + a_{i3}t^3 & , \quad t \in [t_0, t_v] \quad (3.1a) \\ b_{i0} + b_{i1}t + b_{i2}t^2 + b_{i3}t^3 & , \quad t \in [t_v, t_f] \quad (3.1b) \end{cases}$$

subject to the following known restrictions

$$\mathcal{C} = \{t_0, t_v, t_f, q_i(t_0), q_i(t_v), q_i(t_f), \dot{q}_i(t_0), \dot{q}_i(t_f)\} \tag{3.2}$$

together with some conditions of continuity between the segments.

Continuity conditions can be selected in different ways:

1. By specifying the velocity $\dot{q}_i(t_v)$ obtained by solving an inverse kinematics problem of velocity.
2. By specifying the velocity $\dot{q}_i(t_v)$ by means of any heuristic criteria, e.g. a mean value between the slopes of the linear segments that joint the points $[q_i(t_0), q_i(t_v)]$ and $[q_i(t_v), q_i(t_f)]$ when they have the same sign and a null value when the sign is different,

$$\dot{q}_i(t_v) = \begin{cases} 0, & sgn(m_1) \neq sgn(m_2) \quad (3.3a) \\ \dfrac{m_1 + m_2}{2}, & sgn(m_1) = sgn(m_2) \quad (3.3b) \end{cases}$$

where $sgn(x)$ represents the function of the sign of x, and

$$m_1 = \frac{q_i(t_v) - q_i(t_0)}{t_{f1}} \tag{3.4a}$$

$$m_2 = \frac{q_i(t_f) - q_i(t_v)}{t_{f2}} \tag{3.4b}$$

where $t_{f1} = t_v - t_0$ y $t_{f2} = t_f - t_v$.

3. By imposing restriction of continuity in the speed and acceleration at $t = t_v$, $\dot{q}_i(t_v)$ and $\ddot{q}_i(t_v)$,

$$a_{i1} + 2a_{i2}t_v + 3a_{i3}t_v^2 = b_{i1} + 2b_{i2}t_v + 3b_{i3}t_v^2 \tag{3.5a}$$
$$2a_{i2} + 6a_{i3}t_v = 2b_{i2} + 6b_{i3}t_v \tag{3.5b}$$

All ways arrive to the following linear equation system,

$$q_i(t_0) = a_{i0} + a_{i1}t_0 + a_{i2}t_0^2 + a_{i3}t_0^3 \tag{3.6a}$$
$$q_i(t_f) = b_{i0} + b_{i1}t_f + b_{i2}t_f^2 + b_{i3}t_f^3 \tag{3.6b}$$
$$q_i(t_v) = a_{i0} + a_{i1}t_v + a_{i2}t_v^2 + a_{i3}t_v^3 \tag{3.6c}$$
$$q_i(t_v) = b_{i0} + b_{i1}t_v + b_{i2}t_v^2 + b_{i3}t_v^3 \tag{3.6d}$$
$$\dot{q}_i(t_0) = a_{i1} + 2a_{i2}t_0 + 3a_{i3}t_0^2 \tag{3.6e}$$
$$\dot{q}_i(t_f) = b_{i1} + 2b_{i2}t_f + 3b_{i3}t_f^2 \tag{3.6f}$$
$$\dot{q}_i(t_v) = a_{i1} + 2a_{i2}t_v + 3a_{i3}t_v^2 \tag{3.6g}$$
$$\dot{q}_i(t_v) = b_{i1} + 2b_{i2}t_v + 3b_{i3}t_v^2 \tag{3.6h}$$

This system can be represented as two systems in matrix form,

$$g_1 = T_1 a \tag{3.7a}$$
$$g_2 = T_2 b \tag{3.7b}$$

where

$$g_1 = \begin{bmatrix} q_i(t_0) \\ q_i(t_v) \\ \dot{q}_i(t_0) \\ \dot{q}_i(t_v) \end{bmatrix} \tag{3.8a}$$
$$T_1 = \begin{bmatrix} 1 & t_0 & t_0^2 & t_0^3 \\ 1 & t_v & t_v^2 & t_v^3 \\ 0 & 1 & 2t_0 & 3t_0^2 \\ 0 & 1 & 2t_v & 3t_v^2 \end{bmatrix} \tag{3.8b}$$
$$a = \begin{bmatrix} a_{i0} \\ a_{i1} \\ a_{i2} \\ a_{i3} \end{bmatrix} \tag{3.8c}$$
$$g_2 = \begin{bmatrix} q_i(t_v) \\ q_i(t_f) \\ \dot{q}_i(t_v) \\ \dot{q}_i(t_f) \end{bmatrix} \tag{3.8d}$$
$$T_2 = \begin{bmatrix} 1 & t_v & t_v^2 & t_v^3 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_v & 3t_v^2 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \tag{3.8e}$$
$$b = \begin{bmatrix} b_{i0} \\ b_{i1} \\ b_{i2} \\ b_{i3} \end{bmatrix} \tag{3.8f}$$

The parameters of the cubic polynomials can be obtained by inverting T_1 y T_2,

$$a = T_1^{-1} g_1 \tag{3.9}$$
$$b = T_2^{-1} g_2 \tag{3.10}$$

Moreover,the third proposal arrives to the following linear equations' system,

$$q_i(t_0) = a_{i0} + a_{i1}t_0 + a_{i2}t_0^2 + a_{i3}t_0^3 \tag{3.11a}$$
$$q_i(t_f) = b_{i0} + b_{i1}t_f + b_{i2}t_f^2 + b_{i3}t_f^3 \tag{3.11b}$$
$$q_i(t_v) = a_{i0} + a_{i1}t_v + a_{i2}t_v^2 + a_{i3}t_v^3 \tag{3.11c}$$
$$q_i(t_v) = b_{i0} + b_{i1}t_v + b_{i2}t_v^2 + b_{i3}t_v^3 \tag{3.11d}$$
$$\dot{q}_i(t_0) = a_{i1} + 2a_{i2}t_0 + 3a_{i3}t_0^2 \tag{3.11e}$$
$$\dot{q}_i(t_f) = b_{i1} + 2b_{i2}t_f + 3b_{i3}t_f^2 \tag{3.11f}$$
$$a_{i1} + 2a_{i2}t_v + 3a_{i3}t_v^2 = b_{i1} + 2b_{i2}t_v + 3b_{i3}t_v^2 \tag{3.11g}$$
$$2a_{i2} + 6a_{i3}t_v = 2b_{i2} + 6b_{i3}t_v \tag{3.11h}$$

This last method is extended in the following subsection for the case of zero initial and final velocities.

3.1 Zero initial and final velocities and continuity of velocity and acceleration in the intermediate point

When the initial and final velocities are zero, that is $\dot{q}_i(t_0) = \dot{q}_i(t_f) = 0$, it is more convenient to express the cubic polynomials as:

$$q_i(t) = \begin{cases} a_{i0} + a_{i1}(t - t_0) + a_{i2}(t - t_0)^2 + a_{i3}(t - t_0)^3 & , \quad t \in [t_0, t_v] \quad (3.12a) \\ b_{i0} + b_{i1}(t_f - t) + b_{i2}(t_f - t)^2 + b_{i3}(t_f - t)^3 & , \quad t \in [t_v, t_f] \quad (3.12b) \end{cases}$$

In this way,

$$a_{i0} = q_i(t_0) \tag{3.13a}$$
$$b_{i0} = q_i(t_f) \tag{3.13b}$$
$$a_{i1} = 0 \tag{3.13c}$$
$$b_{i1} = 0 \tag{3.13d}$$

where the problem is reduced to obtain the four first parameters instead of eight,

$$q_i(t) = \begin{cases} q_i(t_0) + a_{i2}(t - t_0)^2 + a_{i3}(t - t_0)^3 & , \quad t \in [t_0, t_v] \quad (3.14a) \\ q_i(t_f) + b_{i2}(t_f - t)^2 + b_{i3}(t_f - t)^3 & , \quad t \in [t_v, t_f] \quad (3.14b) \end{cases}$$

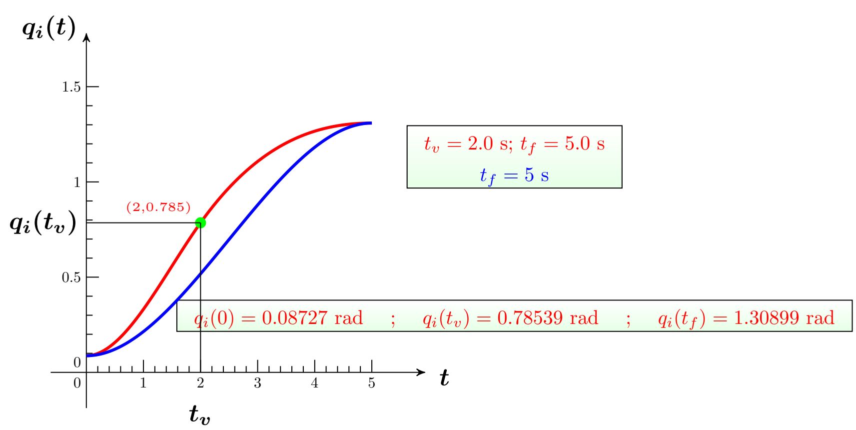


Figure 3.1: Cubic trajectory with (red) and without (blue) intermediate point

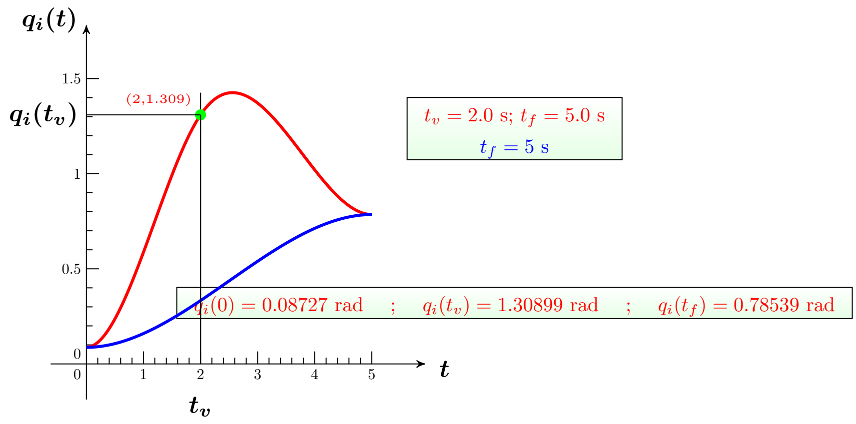


Figure 3.2: Cubic trajectory with (red) and without (blue) intermediate point: $q_i(t_v) > q_i(t_f)$

Figure 3.1 shows two curves for its comparison. The red curve shows the trajectory created by two cubic segments that passes by an intermediate point by following Equation 3.14. The blue curve shows the trajectory between the initial and final point without an intermediate point, studied in Section 2.

Figure 3.2 shows the same images but when $q_i(t_v) > q_i(t_f)$.

In what follows, the functions that define the parameters of every segment of the trajectory are calculated.

$$t_{f1} = t_v - t_0 \tag{3.15a}$$
$$t_{f2} = t_f - t_v \tag{3.15b}$$

By imposing continuity conditions, the four equations of the problem are obtained,

$$q_i(t_v) = q_i(t_0) + a_{i2}t_{f1}^2 + a_{i3}t_{f1}^3 \tag{3.16a}$$
$$q_i(t_v) = q_i(t_f) + b_{i2}t_{f2}^2 + b_{i3}t_{f2}^3 \tag{3.16b}$$
$$2a_{i2}t_{f1} + 3a_{i3}t_{f1}^2 = -2b_{i2}t_{f2} - 3b_{i3}t_{f2}^2 \tag{3.16c}$$
$$2a_{i2} + 6a_{i3}t_{f1} = 2b_{i2} + 6b_{i3}t_{f2} \tag{3.16d}$$

Appendix A demonstrates that

$$a_{i3} = -\frac{\alpha_{i1} + \alpha_{i2}t_{f2}}{t_{f1}(t_{f1} + t_{f2})} \tag{3.17a}$$
$$b_{i3} = -\frac{\alpha_{i1} - \alpha_{i2}t_{f1}}{t_{f2}(t_{f1} + t_{f2})} \tag{3.17b}$$
$$a_{i2} = \frac{3}{2}\frac{\alpha_{i1} + 2\alpha_{i2}t_{f2}}{t_{f1} + t_{f2}} \tag{3.17c}$$
$$b_{i2} = \frac{3}{2}\frac{\alpha_{i1} - 2\alpha_{i2}t_{f1}}{t_{f1} + t_{f2}} \tag{3.17d}$$

where

$$\alpha_{i1} = \frac{2(q_i(t_v) - q_i(t_0))}{t_{f1}} + \frac{2(q_i(t_v) - q_i(t_f))}{t_{f2}} \tag{3.18a}$$
$$\alpha_{i2} = \frac{q_i(t_v) - q_i(t_0)}{2t_{f1}^2} - \frac{q_i(t_v) - q_i(t_f)}{2t_{f2}^2} \tag{3.18b}$$

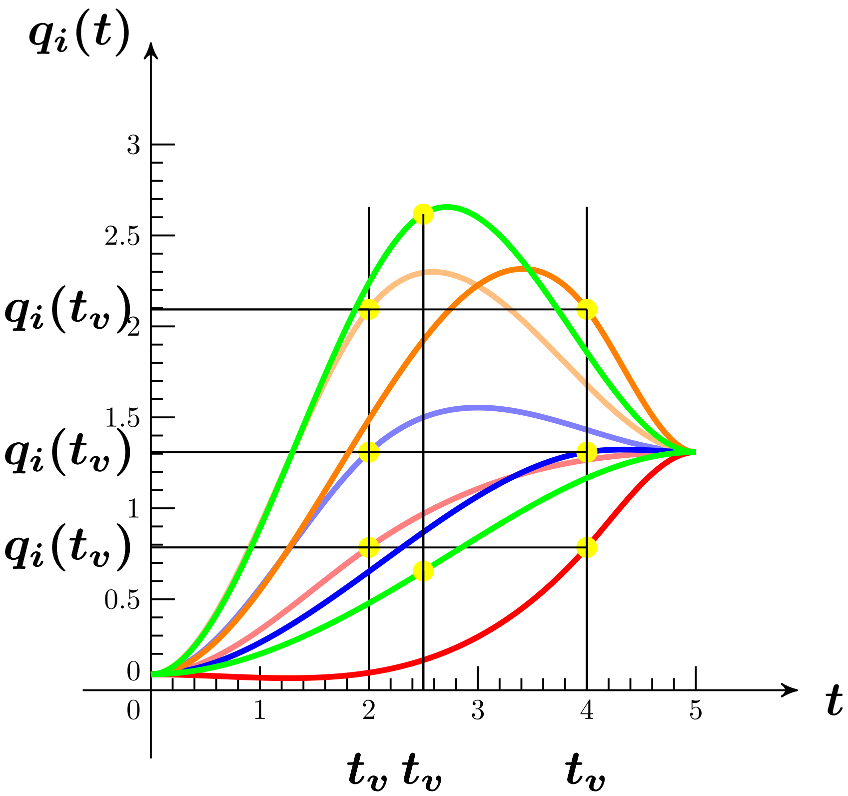


Figure 3.3: Cubic trajectories with intermediate point.

Figure 3.3 shows a collection of cubic trajectories with an intermediate point for different values of $q_i(t_v)$ and t_v. Green curves have been obtained for $t_v = t_f/2$, with $q_i(t_v) = q_i(t_f)/2$ and $q_i(t_v) = 2q_i(t_f)$. Blue curves have been obtained for $q_i(t_v) = q_i(t_f)$.

Cubic polynomials given by Equation 3.14 can be rewritten in discrete form, with $t = kT$, as:

$$q_i(k) = \begin{cases} q_i(0) + a'_{i2}k^2 + a'_{i3}k^3 & , k \in \{0, 1, 2, \cdots, N_1\} \\ q_i(N) + b'_{i2}(N-k)^2 + b'_{i3}(N-k)^3 & , k \in \{N_1, N_1+1, \cdots, N\} \end{cases} \quad (3.19a)(3.19b)$$

where $t_{f1} = N_1 T$, $t_{f2} = N_2 T$, $N = N_1 + N_2$ and

$$\beta_{i1} = \frac{2(q_i(N_1) - q_i(0))}{N_1} + \frac{2(q_i(N_1) - q_i(N_1+N_2))}{N_2} \quad (3.20a)$$

$$\beta_{i2} = \frac{q_i(N_1) - q_i(0)}{2N_1^2} - \frac{q_i(N_1) - q_i(N_1+N_2)}{2N_2^2} \quad (3.20b)$$

and

$$a'_{i3} = -\frac{\beta_{i1} + \beta_{i2}N_2}{NN_1} \quad (3.21a)$$

$$b'_{i3} = -\frac{\beta_{i1} - \beta_{i2}N_1}{NN_2} \quad (3.21b)$$

$$a'_{i2} = \frac{3}{2}\frac{\beta_{i1} + 2\beta_{i2}N_2}{N} \quad (3.21c)$$

$$b'_{i2} = \frac{3}{2}\frac{\beta_{i1} - 2\beta_{i2}N_1}{N} \quad (3.21d)$$

The discretized form in advance valid for the continuous interval $(kT, (k+1)T]$ can be obtained by modifying Equation 3.19,

$$q_i(k+1) = \begin{cases} q_i(0) + a'_{i2}(k+1)^2 + a'_{i3}(k+1)^3 & , k \in \{0, 1, 2, \cdots, N_1 - 1\} \\ q_i(N) + b'_{i2}(N-k-1)^2 + b'_{i3}(N-k-1)^3 & , k \in \{N_1, N_1+2, \cdots, N-1\} \end{cases}$$

(3.22a)

(3.22b)

A Cubic trajectory planning when initial, intermediate and final points are known

When the initial and final velocities are zero, $\dot{q}_i(t_0) = \dot{q}_i(t_f) = 0$, it is more convenient to express the cubic polynomials as:

$$q_i(t) = \begin{cases} a_{i0} + a_{i1}(t - t_0) + a_{i2}(t - t_0)^2 + a_{i3}(t - t_0)^3 & , \quad t \in [t_0, t_v] \quad \text{(A.1a)} \\ b_{i0} + b_{i1}(t_f - t) + b_{i2}(t_f - t)^2 + b_{i3}(t_f - t)^3 & , \quad t \in [t_v, t_f] \quad \text{(A.1b)} \end{cases}$$

Therefore

$$a_{i0} = q_i(t_0) \tag{A.2a}$$
$$b_{i0} = q_i(t_f) \tag{A.2b}$$
$$a_{i1} = 0 \tag{A.2c}$$
$$b_{i1} = 0 \tag{A.2d}$$

and the problem is reduced to obtaining four parameters instead of eight.

$$t_{f1} = t_v - t_0 \tag{A.3a}$$
$$t_{f2} = t_f - t_v \tag{A.3b}$$

By imposing continuity conditions, the four equations of the problem are obtained,

$$q_i(t_v) = q_i(t_0) + a_{i2}t_{f1}^2 + a_{i3}t_{f1}^3 \tag{A.4a}$$
$$q_i(t_v) = q_i(t_f) + b_{i2}t_{f2}^2 + b_{i3}t_{f2}^3 \tag{A.4b}$$
$$2a_{i2}t_{f1} + 3a_{i3}t_{f1}^2 = -2b_{i2}t_{f2} - 3b_{i3}t_{f2}^2 \tag{A.4c}$$
$$2a_{i2} + 6a_{i3}t_{f1} = 2b_{i2} + 6b_{i3}t_{f2} \tag{A.4d}$$

The first two equations can be rewritten as:

$$\frac{q_i(t_v) - q_i(t_0)}{t_{f1}^2} = a_{i2} + a_{i3}t_{f1} \tag{A.5a}$$
$$\frac{q_i(t_v) - q_i(t_f)}{t_{f2}^2} = b_{i2} + b_{i3}t_{f2} \tag{A.5b}$$

By substituting in the two last equations:

$$\frac{2(q_i(t_v) - q_i(t_0))}{t_{f1}} + a_{i3}t_{f1}^2 = -\frac{2(q_i(t_v) - q_i(t_f))}{t_{f2}} - b_{i3}t_{f2}^2 \tag{A.6a}$$
$$\frac{q_i(t_v) - q_i(t_0)}{t_{f1}^2} + 2a_{i3}t_{f1} = \frac{q_i(t_v) - q_i(t_f)}{t_{f2}^2} + 2b_{i3}t_{f2} \tag{A.6b}$$

If

$$\alpha_{i1} = \frac{2(q_i(t_v) - q_i(t_0))}{t_{f1}} + \frac{2(q_i(t_v) - q_i(t_f))}{t_{f2}} \tag{A.7a}$$
$$\alpha_{i2} = \frac{q_i(t_v) - q_i(t_0)}{2t_{f1}^2} - \frac{q_i(t_v) - q_i(t_f)}{2t_{f2}^2} \tag{A.7b}$$

then,

$$a_{i3}t_{f1}^2 + b_{i3}t_{f2}^2 = -\alpha_{i1} \tag{A.8a}$$
$$a_{i3}t_{f1} - b_{i3}t_{f2} = -\alpha_{i2} \tag{A.8b}$$

By solving the system,

$$a_{i3} = -\frac{\alpha_{i1} + \alpha_{i2} t_{f2}}{t_{f1}(t_{f1} + t_{f2})} \tag{A.9a}$$

$$b_{i3} = -\frac{\alpha_{i1} - \alpha_{i2} t_{f1}}{t_{f2}(t_{f1} + t_{f2})} \tag{A.9b}$$

and

$$a_{i2} = \frac{q_i(t_v) - q_i(t_0)}{t_{f1}^2} - a_{i3} t_{f1} = \frac{q_i(t_v) - q_i(t_0)}{t_{f1}^2} + \frac{\alpha_{i1} + \alpha_{i2} t_{f2}}{t_{f1} + t_{f2}} \tag{A.10a}$$

$$b_{i2} = \frac{q_i(t_v) - q_i(t_f)}{t_{f2}^2} - b_{i3} t_{f2} = \frac{q_i(t_v) - q_i(t_f)}{t_{f2}^2} + \frac{\alpha_{i1} - \alpha_{i2} t_{f1}}{t_{f1} + t_{f2}} \tag{A.10b}$$

It can be checked that,

$$\frac{\alpha_{i1}}{2} + 2\alpha_{i2} t_{f2} = (q_i(t_v) - q_i(t_0)) \frac{t_{f1} + t_{f2}}{t_{f1}^2} \tag{A.11a}$$

$$\frac{\alpha_{i1}}{2} - 2\alpha_{i2} t_{f1} = (q_i(t_v) - q_i(t_f)) \frac{t_{f1} + t_{f2}}{t_{f2}^2} \tag{A.11b}$$

Therefore,

$$a_{i2} = \frac{3}{2} \frac{\alpha_{i1} + 2\alpha_{i2} t_{f2}}{t_{f1} + t_{f2}} \tag{A.12a}$$

$$b_{i2} = \frac{3}{2} \frac{\alpha_{i1} - 2\alpha_{i2} t_{f1}}{t_{f1} + t_{f2}} \tag{A.12b}$$

Finally, it can be checked that:

$$\alpha_{i1} + 2\alpha_{i2} t_{f2} = \frac{(t_{f2} + 2t_{f1})(q_i(t_v) - q_i(t_0))}{t_{f1}^2} + \frac{q_i(t_v) - q_i(t_f)}{t_{f2}} \tag{A.13a}$$

$$\alpha_{i1} - 2\alpha_{i2} t_{f1} = \frac{q_i(t_v) - q_i(t_0)}{t_{f1}} + \frac{(2t_{f2} + t_{f1})(q_i(t_v) - q_i(t_f))}{t_{f2}^2} \tag{A.13b}$$

$$\alpha_{i1} + \alpha_{i2} t_{f2} = \frac{(4t_{f1} + t_{f2})(q_i(t_v) - q_i(t_0))}{2t_{f1}^2} + \frac{3}{2} \frac{q_i(t_v) - q_i(t_f)}{t_{f2}} \tag{A.13c}$$

$$\alpha_{i1} - \alpha_{i2} t_{f1} = \frac{3}{2} \frac{q_i(t_v) - q_i(t_0)}{t_{f1}} + \frac{(4t_{f2} + t_{f1})(q_i(t_v) - q_i(t_f))}{2t_{f2}^2} \tag{A.13d}$$

References

[1] J. Craig, *Introduction to Robotics. Mechanics & Control.* Addison-Wesley Publishing Company, Inc., 1986.