ROBOTICS, VISION AND CONTROL

Trajectory Planning. Point-to-Point Polynomials

Nicola Piccinelli

UNIVERSITÀ di VERONA
Dipartimento di INGEGNERIA PER LA MEDICINA DI INNOVAZIONE

ALTAIR

Outline

Problem statement

Joint Space Trajectories
Point-to-Point

PROJECT

Problem statement

Trajectory planning

Goal: compute the *reference inputs* for the motion control system to move the manipulator's end-effector to accomplish a specific task.

The trajectory planning problem consists in finding a relationship between two elements belonging to different domains: *time* and *space*.

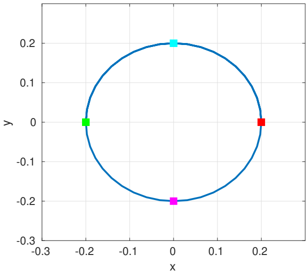
Planned trajectory is a *time sequence* of values, i.e. a parametric function of time

Geometric Path: the locus of points in the joint space ($\boldsymbol{q}$) or in the operational space ($\boldsymbol{x}$)

Trajectory: is a path on which a timing law (or motion law) is specified, for instance in terms of velocities and/or accelerations at each point, e.g. $\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t), \ddot{\boldsymbol{q}}(t)$, or $\boldsymbol{x}(t), \dot{\boldsymbol{x}}(t), \ddot{\boldsymbol{x}}(t)$

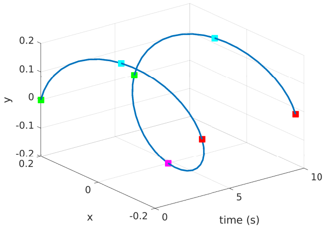
path

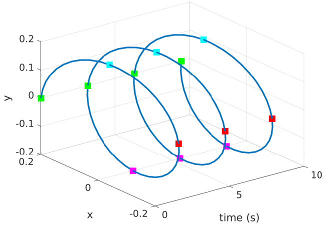
$$\boldsymbol{p} = \boldsymbol{p}(u)$$



trajectories

$$\boldsymbol{p}(t) = \boldsymbol{p}(u(t)) = \boldsymbol{p}(u) \circ u(t)$$





Trajectory planning

Inputs:

- path description
- path constraints (e.g. positions and velocities should be continuous functions of time)
- trajectory time (duration)
- dynamics constraints (manipulator)
- obstacles

$\rightarrow$

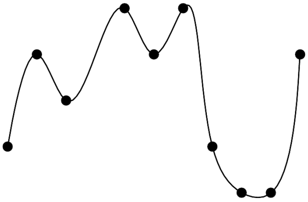
More often:

- extremal points
- intermediate points
- geometric primitives interpolating the points,
- velocity constraints
- acceleration constraints
- velocity and acceleration at particular points of interest

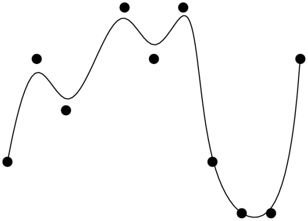
Outputs:

- time sequences of the positions (end-effector poses), velocities and accelerations which satisfy the constraints (*if a trajectory exists...*).

Interpolation: the curve crosses the given points for some values of the time



Approximation: the curve does not pass exactly through the points, but there is an error that may be assigned by specifying a prescribed tolerance



For given boundary conditions (initial and final positions, velocities, accelerations, etc.) and duration, the *typology of the trajectory* has a strong influence on the peak values of the velocity and acceleration in the intermediate points ($\rightarrow$ frequency aspects, vibrations)

	point-to-point motion (point-to-point trajectory)	motion through a sequence of points (multi-point trajectory)
joint space	$(q_i, t_i), (q_f, t_f)$	$(q_i, t_i), (q_1, t_1), \ldots, (q_f, t_f)$
operational space	$(x_i, t_i), (x_f, t_f)$	$(x_i, t_i), (x_1, t_1), \ldots, (x_f, t_f)$

Joint Space Trajectories
Point-to-Point

Polynomials

Assumption 1: *scalar case*, i.e. single actuator, or single axis of motion
Assumption 2: *no dynamic model is taken into account*

Data: initial joint configuration $q_i \in \mathbb{R}$ at t_i, final joint configuration $q_f \in \mathbb{R}$ at t_f ($t_f = t_i +$traveling time), no constraints

Trajectories described by polynomial functions

$$q(t) = a_n t^n + a_{n-1} t^{n-1} + \ldots + a_2 t^2 + a_1 t + a_0, \qquad t \in [t_i, t_f]$$

where the $n+1$ coefficients a_i are determined so that the initial and final constraints are satisfied.

- The degree n of the polynomial depends on the number of conditions to be satisfied (e.g. velocities and accelerations in specific time instant $t_j \in [t_i, t_f]$) and on the desired “smoothness” of the resulting motion.
- Since the number of boundary conditions is usually even, the degree n of the polynomial function is odd.

Linear trajectory

The *linear trajectory* (constant velocity) requires a first order polynomial

$$q(t) = a_0 + a_1(t - t_i), \qquad t \in [t_i, t_f]$$

Conditions on the coefficients:

$$\begin{cases} q(t_i) = q_i = a_0 \\ q(t_f) = q_f = a_0 + a_1(t_f - t_i) \end{cases}$$

We need to solve

$$\begin{bmatrix} 1 & 0 \\ 1 & \Delta T \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} q_i \\ q_f \end{bmatrix}$$

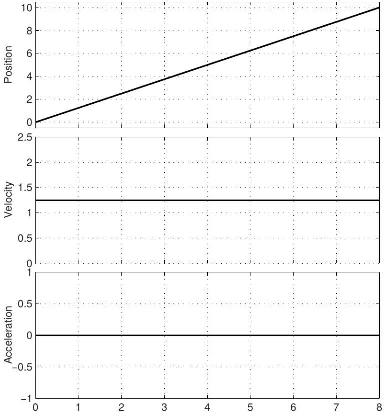
where $\Delta T = t_f - t_i$. Then, considering $\Delta q = q_f - q_i$ we have

$$\begin{cases} a_0 = q_i \\ a_1 = \frac{\Delta q}{\Delta T} \end{cases}$$

Linear trajectory

Remarks:

- The velocity $\dot{q}(t) = a_1$ is constant
- The acceleration is null in the interior of the trajectory and has an impulsive behaviour at the extremities



Parabolic trajectory

The *parabolic trajectory* (constant acceleration) requires the composition of two second degree polynomials, one from t_i to t_m (the flex point) and the second from t_m to t_f

$$t_m := \frac{t_i + t_f}{2}, \qquad q(t_m) = q_m := \frac{q_i + q_f}{2}$$

In the case of trajectory symmetric with respect to its middle point.

Acceleration period

$$q_a(t) = a_0 + a_1(t - t_i) + a_2(t - t_i)^2, \qquad t \in [t_i, t_m]$$

Conditions on the coefficients:

$$\begin{cases} q_a(t_i) = q_i = a_0 \\ q_a(t_m) = q_m = a_0 + a_1(t_m - t_i) + a_2(t_m - t_i)^2 \\ \dot{q}_a(t_i) = \dot{q}_i = a_1 \end{cases} \quad \Rightarrow \quad \begin{cases} a_0 = q_i \\ a_1 = \dot{q}_i \\ a_2 = \frac{2}{\Delta T^2}(\Delta q - \dot{q}_i \Delta T) \end{cases}$$

Parabolic trajectory

Deceleration period

$$q_d(t) = a_3 + a_4(t - t_m) + a_5(t - t_m)^2, \qquad t \in [t_m, t_f]$$

Conditions on the coefficients:

$$\begin{cases} q_d(t_m) = q_m = a_3 \\ q_d(t_f) = q_f = a_3 + a_4(t_f - t_m) + a_5(t_f - t_m)^2 \\ \dot{q}_d(t_f) = \dot{q}_f = a_4 + 2a_5(t_f - t_m) \end{cases} \Rightarrow \begin{cases} a_3 = q_m \\ a_4 = 2\frac{\Delta q}{\Delta T} - \dot{q}_f \\ a_5 = \frac{2}{\Delta T^2}(\dot{q}_f \Delta T - \Delta q) \end{cases}$$

Remarks:

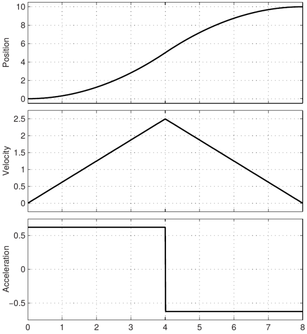
- if $\dot{q}_i \neq \dot{q}_f$ the velocity profile is discontinuous at t_m
- the maximum velocity in obtained at the flex point

$$\dot{q}_{max} = \dot{q}_a(t_m) = 2\frac{\Delta q}{\Delta T} - \dot{q}_i$$

- acceleration profile is piecewise constant with opposite sign in the acceleration/deceleration periods

$$\ddot{q}_a = \frac{4}{\Delta T^2}(\Delta q - \dot{q}_i \Delta T), \quad \ddot{q}_d = \frac{4}{\Delta T^2}(\dot{q}_f \Delta T - \Delta q)$$

- The jerk is always null except at the flex point, when the acceleration changes its sign and it assumes an infinite value.



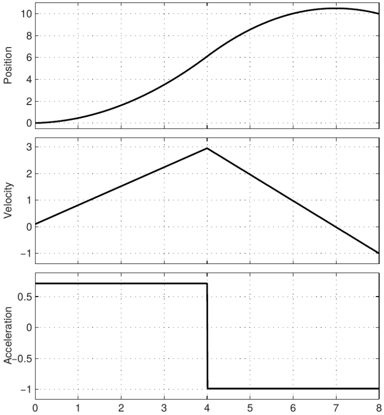
UNIVERSITÀ di VERONA
Dipartimento di INGEGNERIA PER LA MEDICINA DI INNOVAZIONE
ALTAIR

If $\dot{q}_i \neq \dot{q}_f$ (i.e. the velocity profile is discontinuous at t_m with the previous approach) and the constraint on the position at t_m (i.e. $q_m = \frac{q_i+g_f}{2}$) is not assigned, to have a continuous velocity profile ($\dot{q}_a(t_m) = \dot{q}_d(t_m)$) we can solve the following system of six equations in six unknowns

$$\begin{cases} q_a(t_i) = q_i = a_0 \\ \dot{q}_a(t_i) = \dot{q}_i = a_1 \\ q_d(t_f) = q_f = a_3 + a_4\frac{\Delta T}{2} + a_5\left(\frac{\Delta T}{2}\right)^2 \\ \dot{q}_d(t_f) = \dot{q}_f = a_4 + 2a_5\frac{\Delta T}{2} \\ q_a(t_m) = a_0 + a_1\frac{\Delta T}{2} + a_2\left(\frac{\Delta T}{2}\right)^2 = a_3 = q_d(t_m) \\ \dot{q}_a(t_m) = a_1 + 2a_2\frac{\Delta T}{2} = a_4 = \dot{q}_d(t_m) \end{cases} \Rightarrow \begin{cases} a_0 = q_i \\ a_1 = \dot{q}_i \\ a_2 = \dots \\ a_3 = \\ a_4 = \\ a_5 = \end{cases}$$

where $\frac{\Delta T}{2} = t_m - t_i = t_f - t_m$. It is possible to generate trajectory with asymmetric constant acceleration, i.e. $t_m \neq \frac{t_i+t_f}{2}$, but t_m is given.

Parabolic trajectory



Cubic Polynomials

$\Rightarrow$ Infinite solutions! We need a criterion / metric / performance index

$\Rightarrow$ optimization problem

We know from “Robotics” that a *cubic polynomial* (third-order polynomial function) minimizes the energy associated to the motion. Then

$$q(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

with

$$\dot{q}(t) = 3a_3 t^2 + 2a_2 t + a_1$$
$$\ddot{q}(t) = 6a_3 t + 2a_2$$

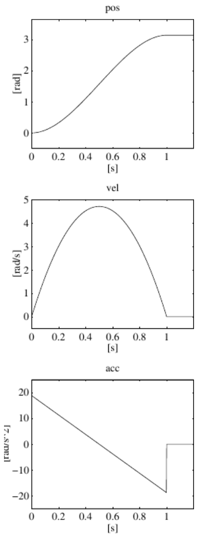
Cubic Polynomials

Since we have 4 unkowns, we need 4 constraints: $q_i, \dot{q}_i$ at t_i, $q_f, \dot{q}_f$ at t_f

$$
\begin{aligned}
a_3 t_i^3 + a_2 t_i^2 + a_1 t_i + a_0 &= q_i \\
3a_3 t_i^2 + 2a_2 t_i + a_1 &= \dot{q}_i \\
a_3 t_f^3 + a_2 t_f^2 + a_1 t_f + a_0 &= q_f \\
3a_3 t_f^2 + 2a_2 t_f + a_1 &= \dot{q}_f
\end{aligned}
$$

If $t_i = 0$

$$
\begin{aligned}
a_0 &= q_i \\
a_1 &= \dot{q}_i \\
a_3 t_f^3 + a_2 t_f^2 + a_1 t_f + a_0 &= q_f \\
3a_3 t_f^2 + 2a_2 t_f + a_1 &= \dot{q}_f
\end{aligned}
$$



Cubic Polynomials

Exercise: Derive the four conditions using the cubic polynomial

$$q(t) = a_3(t - t_i)^3 + a_2(t - t_i)^2 + a_1(t - t_i) + a_0, \quad t \in [t_i, t_f]$$

as a function of Δq and ΔT.

Solution:

$$\begin{cases} a_0 = q_i \\ a_1 = \dot{q}_i \\ a_2 = \frac{3\Delta q - (2\dot{q}_i + \dot{q}_f)\Delta T}{\Delta T^2} \\ a_3 = \frac{-2\Delta q + (\dot{q}_i + \dot{q}_f)\Delta T}{\Delta T^3} \end{cases}$$

By exploiting this result, it is very simple to compute a trajectory with continuous velocity through a sequence of n points by connecting with the previous equations pairs of points. (SEE LATER)

Remark: Discontinuities in the desired trajectory may generate vibrations in the robotic manipulator due to the induced discontinuities in the applied forces and the elastic effects of the mechanical system.

Solution 1: If we would like to assign also the initial and final values of acceleration, six constraints have to be satisfied and then a polynomial of at least *fifth order* is needed

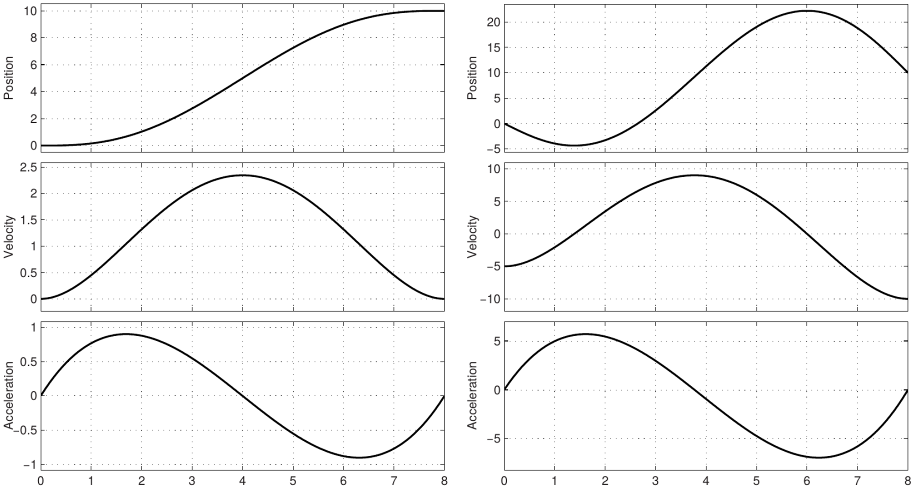
$$q(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

Then we also have the equations

$$\begin{aligned} \dots &= \ddot{q}_i \\ \dots &= \ddot{q}_f \end{aligned}$$

$\Rightarrow$ *trajectories with continuous acceleration, but discontinuous jerk*

5th-order Polynomials



Solution 2: If we would like to assign also the initial and final values of jerk, eight constraints have to be satisfied and then a polynomial of at least *seventh order* is needed

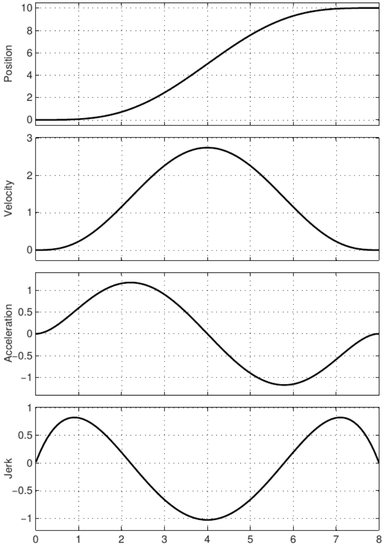
$$q(t) = a_7 t^7 + a_6 t^6 + a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

Then we also have the equations

$$\begin{aligned} \dots &= \dddot{q}_i \\ \dots &= \dddot{q}_f \end{aligned}$$

$\Rightarrow$ *trajectories with continuous jerk*

7th-order Polynomials




Project

TODO:

- Implement in Matlab 3rd– (cubic), 5th–, 7th–order polynomials for $q_i > q_f$ and $q_i < q_f$, and in both formulations

$$q(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0, \qquad t \in [t_i, t_f]$$

and

$$q(t) = a_3 (t - t_i)^3 + a_2 (t - t_i)^2 + a_1 (t - t_i) + a_0, \qquad t \in [0, \Delta T]$$