CHAPTER 7. TRAJECTORY GENERATION

- {trajectory, motion, path (spatial)} + {generation, planning, prediction (biological systems)}

- Trajectory: desired time history of position, velocity, and acceleration for each DOF from initial to final configuration in multidimensional space; Joint space schemes vs. Cartesian space schemes

- Motions of end-effector tool frame $\{T\}$, relative to station frame $\{S\}$ (inertial frame), from current value $\{T_{initial}\}$ to a desired final value $\{T_{final}\}$

- Via points (frames): sequence of desired intermediate points (frames) between initial and final configurations
  - Via point (in configuration space) is a frame that specifies position and orientation of tool frame relative to station frame; tool frame must pass through a set of via points.

- Path points (frames): include all via points plus initial and final points

- Path update rate: time rate at which the trajectory points are computed

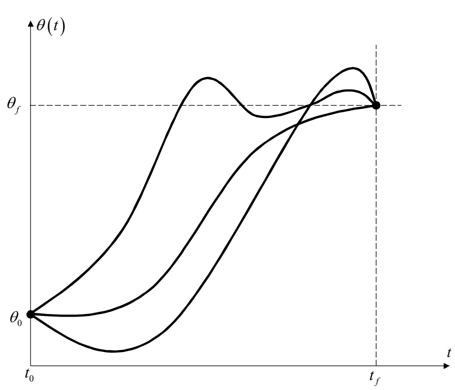
- Motion of smooth function: continuous up to first (or second) derivative (C^1, C^2)

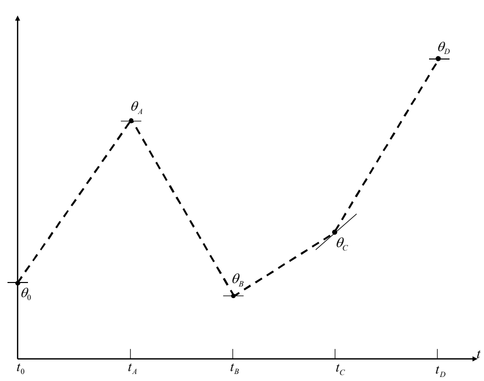
<u>Joint Space Schemes</u>

Path shapes (in space and in time) are described in terms of (smooth) joint variable functions.

- Given: Initial, final, and via points are specified in terms of desired position/orientation, velocity, and acceleration of end-effector tool frame $\{T\}$ relative to station frame $\{S\}$, and time duration
  → Converted into desired joint variables and velocities (by inverse kinematics and inverse Jacobian).
  → Calculate smooth function for each joint.
    - Time required for each segment is same for each joint; all joints reach a point at the same time.
    - No singularity problem

- Use numerical interpolation method for each joint: polynomials, piecewise polynomials, spline functions, B-splines, Hermite interpolations, Fourier transform (trigonometric), etc.
  → Calculate coefficients from initial, final, and via point conditions.





Cubic Polynomials (Joint Space)

- General *m*th degree polynomial: $\theta(t) = a_0 + a_1 t + a_2 t^2 \ldots + a_{m-1} t^{m-1} + a_m t^m$
- Cubic (m = 3) polynomial: $\theta(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3$

- Example: initial and final conditions (four constraints) $\theta(0) = \theta_0$; $\dot{\theta}(0) = \dot{\theta}_0$; $\theta(t_f) = \theta_f$; $\dot{\theta}(t_f) = \dot{\theta}_f$
  - can be solved for 4 unknowns

  → $a_0 = \theta_0$; $a_1 = \dot{\theta}_0$; $a_2 = \frac{3}{t_f^2}(\theta_f - \theta_0) - \frac{2}{t_f}\dot{\theta}_0 - \frac{1}{t_f}\dot{\theta}_f$; $a_3 = -\frac{2}{t_f^3}(\theta_f - \theta_0) + \frac{1}{t_f^2}(\dot{\theta}_f + \dot{\theta}_0)$

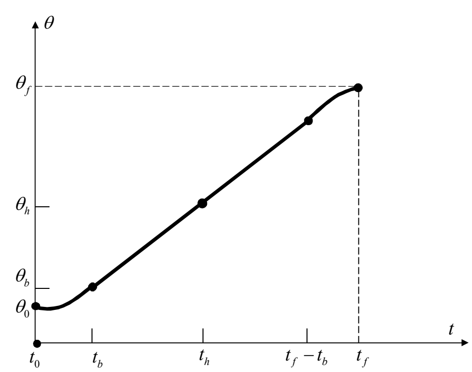
- If k via points are given → k+1 cubic segments
- Methods to specify desired velocity at via points
  (1) Specified by user in terms of Cartesian linear and angular velocities of tool frame $\{T\}$
  → Mapped to desired joint velocities using inverse Jacobian
  (If singular at a via point, then user is not free to assign arbitrary velocity at this point.)
  (2) Specified automatically by the system using a suitable heuristic in Cartesian or joint space
  → Reasonable choice of joint velocities at via points
  → Example: via points connected with straight line segments; if slope of these lines changes sign at via point, choose zero velocity; if slope of these lines does not change sign, choose the average of the two slopes
  (3) Specified automatically by the system such that the accelerations at via points are continuous
  → In spline, replace two velocity constraints at the connection of two cubics with two constraints of continuous velocity and continuous acceleration
  → For n cubic segments – tridiagonal matrix form

Linear Function with Parabolic Blends (Joint Space)

Linearly interpolate to move from initial to final joint position.
(But generally, end-effector does not necessarily move in a straight line in space.)

- For a smooth path with continuous position and velocity
  → linear function with a parabolic blend region at each path point
  During blend portion of trajectory, constant acceleration is used to change velocity smoothly.
  Linear function and two parabolic functions are splined together. → continuous q and $\dot{q}$

  $\theta_b = \theta_0 + \frac{1}{2}\ddot{\theta} t_b^2$ ($\ddot{\theta}$: acceleration during blend region)



- Assume
  - Parabolic blends both have same duration.
  - Same constant acceleration is used during both blends.

  → Symmetric about halfway point time t_h ($t = 2t_h$) and about halfway point position θ_h

- [Velocity at the end of blend region] = [velocity of linear section]

$$\ddot{\theta}t_b = \frac{\theta_h - \theta_b}{t_h - t_b} \;\Rightarrow\; \ddot{\theta}t_b^2 - \ddot{\theta}t_f t_b + (\theta_f - \theta_0) = 0 \;\Rightarrow\; t_b = \frac{t_f}{2} - \frac{\sqrt{\ddot{\theta}^2 t_f^2 - 4\ddot{\theta}(\theta_f - \theta_0)}}{2\ddot{\theta}}$$

where $\ddot{\theta} \geq \dfrac{4(\theta_f - \theta_0)}{t_f^2}$

  - If $\ddot{\theta} = 4(\theta_f - \theta_0)/t_f^2$ → no linear portion; path is composed of two blends that connect with equivalent slope
  - As $\ddot{\theta}$ increases → blend region decreases
  - If $\ddot{\theta} \to \infty$ → simple linear interpolation

- With via points
  (skip)

<u>Cartesian Space Schemes</u>

Cartesian position and orientation as functions of time (spatial path)

- Given: initial, final, and via points are specified in terms of desired position/orientation, velocity, and acceleration of end-effector tool frame $\{T\}$ relative to station frame $\{S\}$, and time duration

- Position: ${}^0\mathbf{P}_{AORG}$
- Orientation: use angle-axis representation $ROT({}^0\hat{K}_A, \theta_{0A})$ → ${}^0\mathbf{K}_A = \theta_{0A}\,{}^0\hat{K}_A$
  (Note: Rotation matrices cannot be interpolated.)
- 6x1 vector of Cartesian position and orientation as a function of time: ${}^0\boldsymbol{\chi}_A = \begin{bmatrix} {}^0\mathbf{P}_{AORG} \\ {}^0\mathbf{K}_A \end{bmatrix}$

- Recall: Angle-axis orientation representation is not unique, i.e., $({}^0\hat{K}_A, \theta_{0A}) = ({}^0\hat{K}_A, \theta_{0A} + n360^o)$.
  → Minimize total amount of rotation $\|{}^0\mathbf{K}_{final} - {}^0\mathbf{K}_{initial}\|$ for interpolation.

<u>Geometric problems with Cartesian Paths</u>

Workspace, singularities

- Intermediate points unreachable: Although the initial and final configuration of manipulator end-effector are within workspace, it is possible that not all points lying on a straight line connecting these two points are in the workspace.

- High joint rates near singularity: locations in workspace where it is impossible to choose finite joint rates that yield desired Cartesian velocity of end-effector → Certain Cartesian paths are impossible
  : if manipulator approaches a singular configuration, one or more joint velocities increase toward infinity → deviation from desired path

- Start and goal reachable in different solutions: Goal point cannot be reached in the same physical solution as the manipulator is in at the start point.

- Other problems: collision detection/avoidance, bifurcation, redundancy (optimization), etc.