Lecture Notes for
A Geometrical Introduction to Robotics and Manipulation

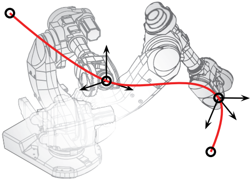
Richard Murray and Zexiang Li and Shankar S. Sastry
CRC Press

Zexiang Li[1] and Yuanqing Wu[1]

[1]ECE, Hong Kong University of Science & Technology

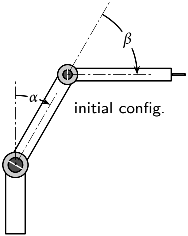
April 28, 2011

Chapter 5 Trajectory Generation



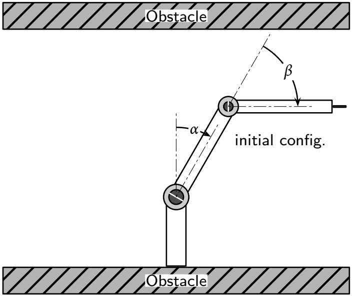
Section 5.1

□ A Simple Robot Planning Example:



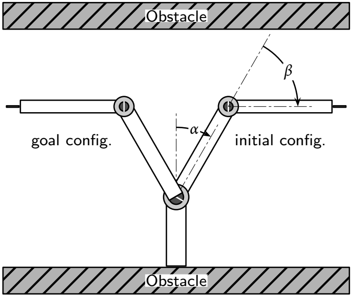
A Two-DoF robot arm

□ A Simple Robot Planning Example:



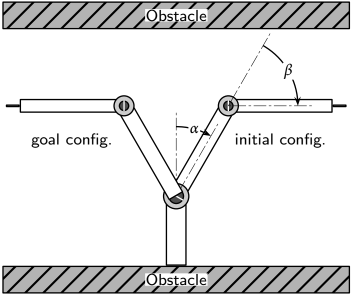
A Two-DoF robot arm

□ A Simple Robot Planning Example:

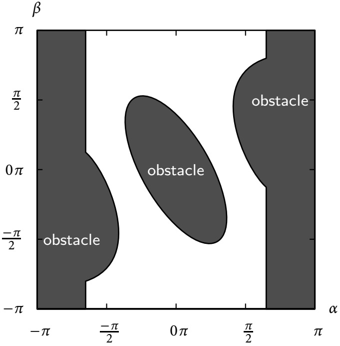


A Two-DoF robot arm

□ A Simple Robot Planning Example:

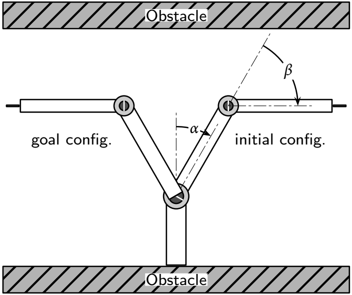


A Two-DoF robot arm

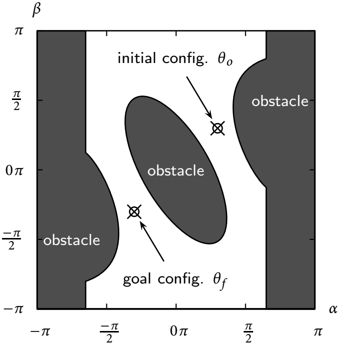


Configuration space $\mathcal{C}$

□ A Simple Robot Planning Example:

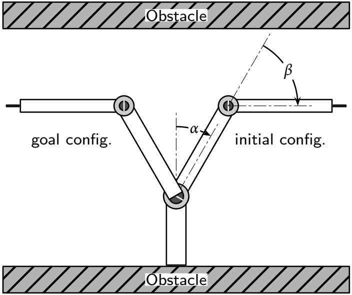


A Two-DoF robot arm

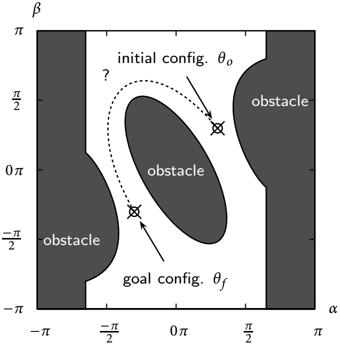


Configuration space $\mathcal{C}$

□ A Simple Robot Planning Example:

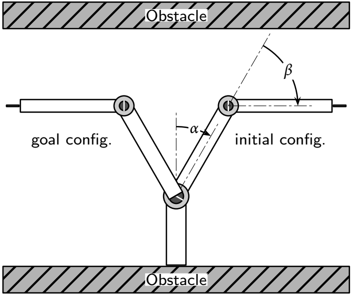


A Two-DoF robot arm

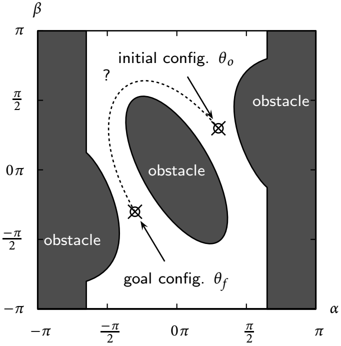


Configuration space $\mathcal{C}$

□ A Simple Robot Planning Example:



A Two-DoF robot arm



Configuration space $\mathcal{C}$

Definition: Path planning

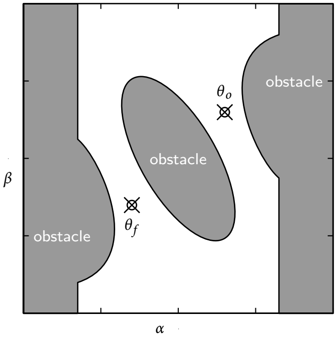
Given an initial and a final configuration θ_o and θ_f in the configuration space $\mathcal{C}$, find a collision-free path, $\theta : [0,1] \mapsto \mathcal{C}$ such that $\theta(0) = \theta_o$ and $\theta(1) = \theta_f$. (continues next slide)

◊ Path planning methods ([1, 2]): probabilistic roadmap, cell decomposition, numerical potential field,etc.

5.1 Introduction to Trajectory Generation

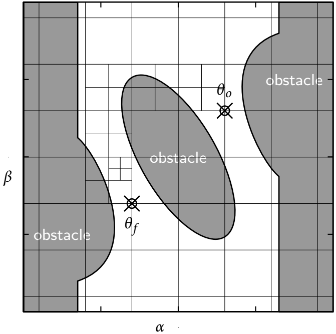
◇ Path planning methods ([1, 2]): probabilistic roadmap, cell decomposition, numerical potential field,etc.

◇ A Cell decomposition example:



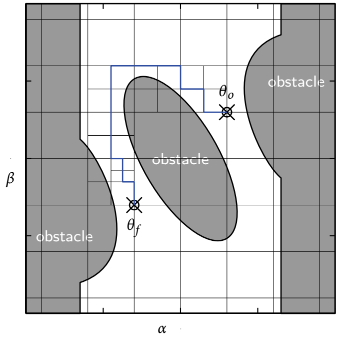
◇ Path planning methods ([1, 2]): probabilistic roadmap, cell decomposition, numerical potential field,etc.

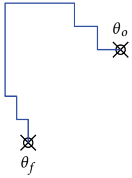
◇ A Cell decomposition example:



◇ Path planning methods ([1, 2]): probabilistic roadmap, cell decomposition, numerical potential field,etc.

◇ A Cell decomposition example:



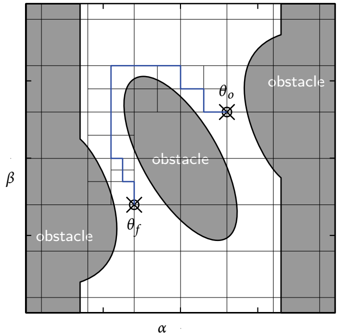


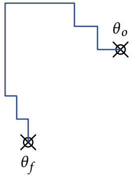
A collision free path

5.1 Introduction to Trajectory Generation

♢ Path planning methods ([1, 2]): probabilistic roadmap, cell decomposition, numerical potential field,etc.

♢ A Cell decomposition example:



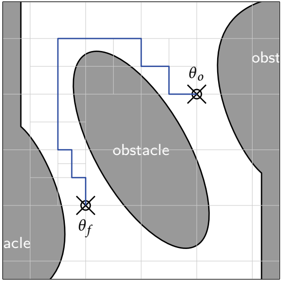


A collision free path

♢ Note: The generated path may not be suitable for robot control. E.g. not smooth.

5.1 Introduction to Trajectory Generation

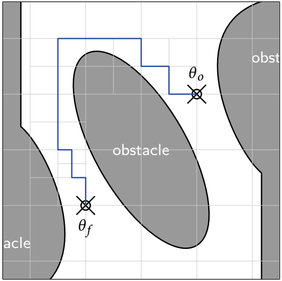
1 Motion planning and trajectory generation



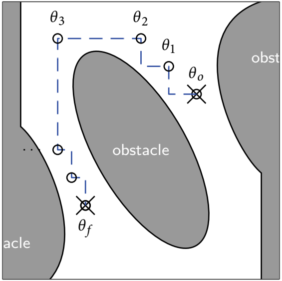
A collision free path

5.1 Introduction to Trajectory Generation

1 Motion planning and trajectory generation

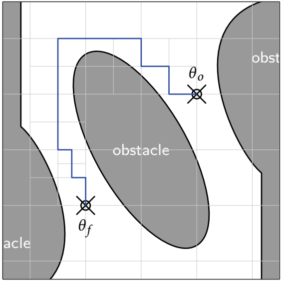


A collision free path

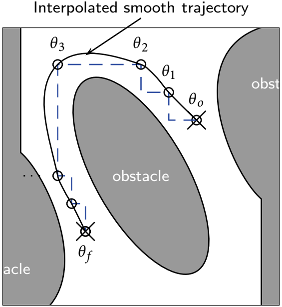


Generated via-points

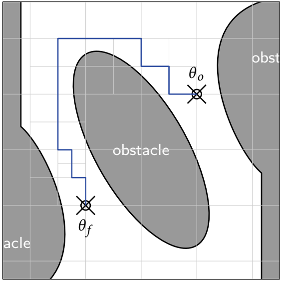
5.1 Introduction to Trajectory Generation



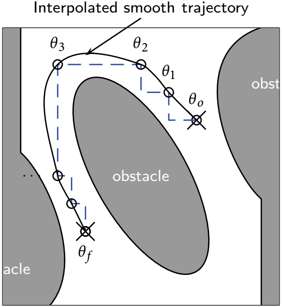
A collision free path



Generated via-points



A collision free path



Generated via-points

Definition: Trajectory generation

Given θ_o and θ_f, and a sequence of via points $\theta_k, k = 1, \ldots, n-1$, compute a trajectory $\theta : [t_0, t_n] \mapsto \mathcal{C}$ such that $\theta(t_0) = \theta_o$, $\theta(t_n) = \theta_f$, and $\theta(t_k) = \theta_k, k = 1, \ldots, n-1$.

♢ Note: the trajectory should be easy to specify, store and generate in real-time.

5.1 Introduction to Trajectory Generation

□ Main constraints on traj. generation:

1. Rated speed $|\dot{\theta}^i(t)| \leq \dot{\theta}^i_{\max}$
2. Rated Acceleration $|\ddot{\theta}^i(t)| \leq \ddot{\theta}^i_{\max}$
3. Bounded Jerk (avoiding excitation): $|\dddot{\theta}^i(t)| \leq \dddot{\theta}^i_{\max}$
4. Continuity in velocity, acceleration for bounded jerk

		Small Capacity		Medium Capacity		Large Capacity
		SGMAH	SGMPH	SGMSH	SGMGH	SGMBH
Yaskawa Σ series motor specification						
Rated Torque Range	[lb-in]	0.8-21	2.8-42	28.2-140	28-845	1239-3100
Peak Torque Range	[lb-in]	2.5-63	8.4-126	84.4-422	79-1988	2478-6120
Rated Speed	[rpm]	3000	3000	3000	1500	1500
Max. Speed	[rpm]	5000	5000	5000	3000	2000
Rated Acc.	[Rad/s^2]	57500	38500	12780	1575	1780
Power Range	[W]	30-750	100-1.5k	1k-5k	500-15k	22k-55k
Inertia		Low	Medium	Low	Medium	Medium

Generation of via-points:

1. by a path planner (the previous example)

Generation of via-points:

1. by a path planner (the previous example)
2. by a teach pendant:

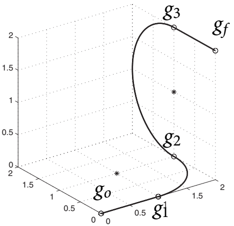
via points directly recorded as joint angles, no inverse kinematics required.

Generation of via-points:

1. by a path planner (the previous example)
2. by a teach pendant:

via points directly recorded as joint angles, no inverse kinematics required.

3. by G-code (through CAM software):

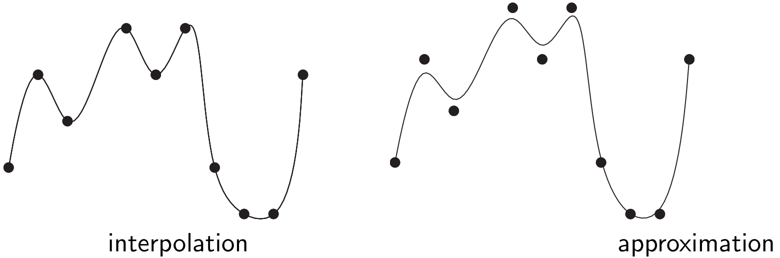


- use inverse kinematics and inverse Jacobian to obtain joint angles and velocity information:

$$g_i, V_i \xrightarrow{g^{-1}, J^{-1}} \theta_i, \dot{\theta}_i, i = 0, 1, 2, \ldots$$

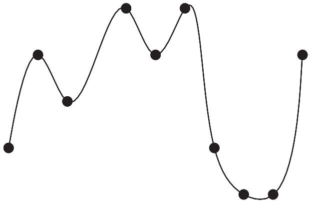
- constraints from both joint and workspace need be considered.

From via-points to trajectory:

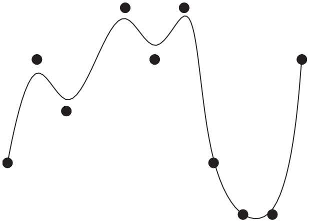


Definition: Interpolation
Constructing new data points within the range of a discrete set of known data points (exact fitting).

□ From via-points to trajectory:



interpolation



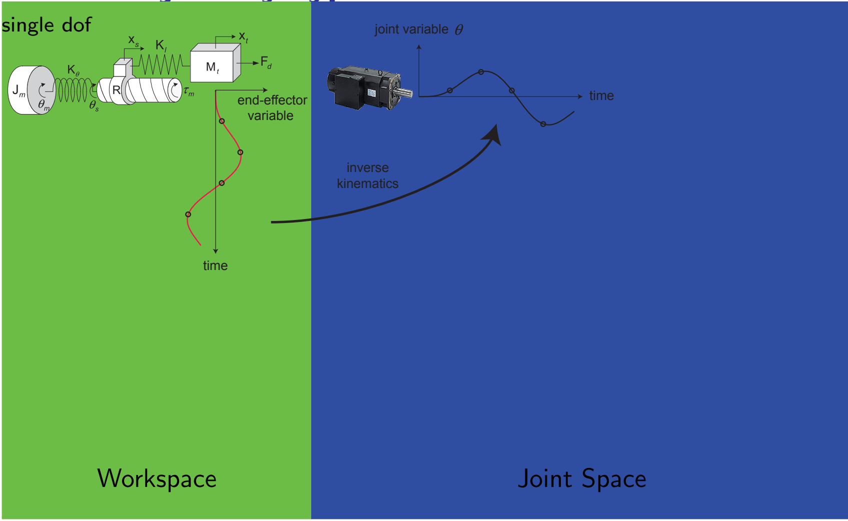
approximation

Definition: Interpolation
Constructing new data points within the range of a discrete set of known data points (exact fitting).

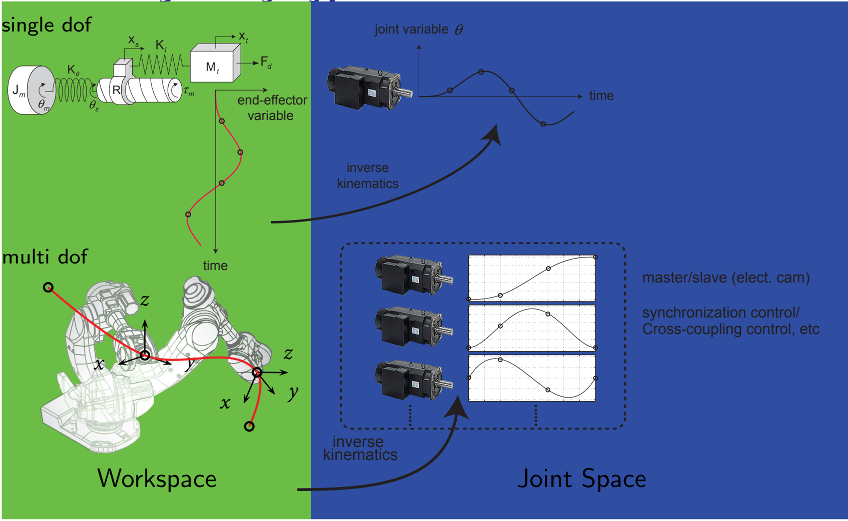
Definition: Approximation
Inexact fitting of a discrete set of known data points.

5.1 Introduction to Trajectory Generation

□ Trajectory type:

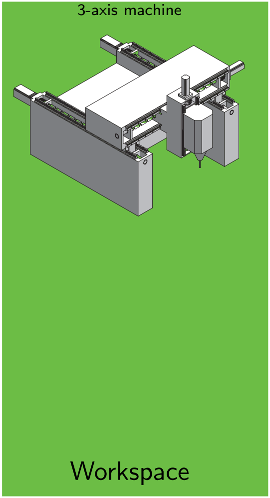


□ Trajectory type:

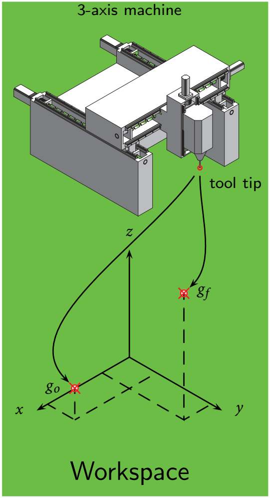


Section 5.2

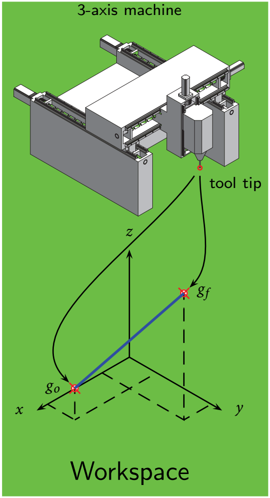
□ A simple example-Linear interpolation with no via-points:



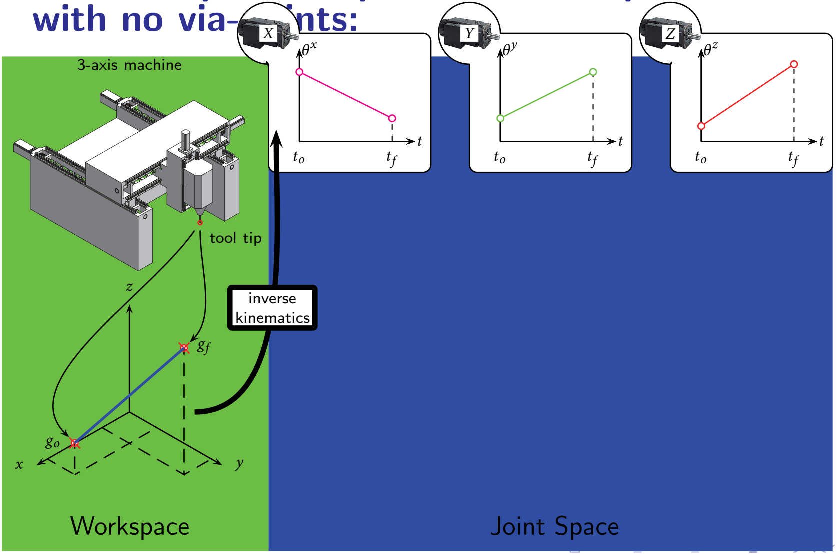
□ A simple example-Linear interpolation with no via-points:



□ A simple example-Linear interpolation with no via-points:

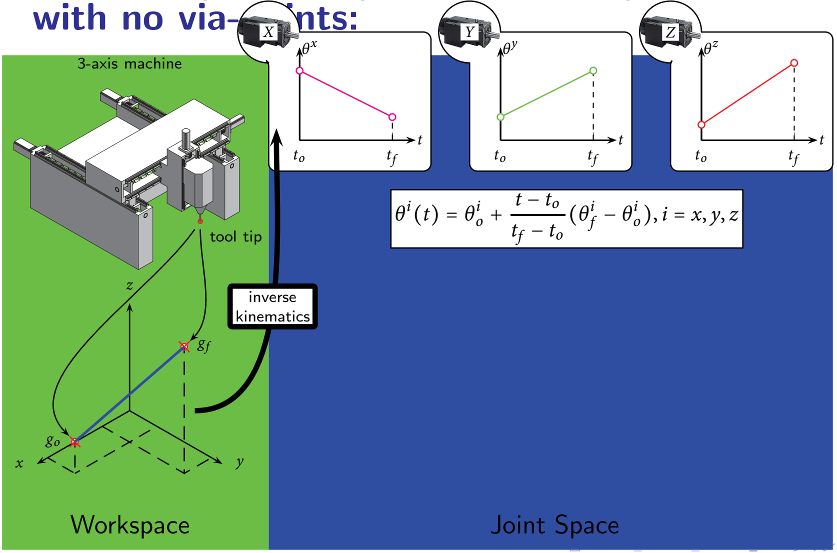


□ A simple example-Linear interpolation with no via-points:



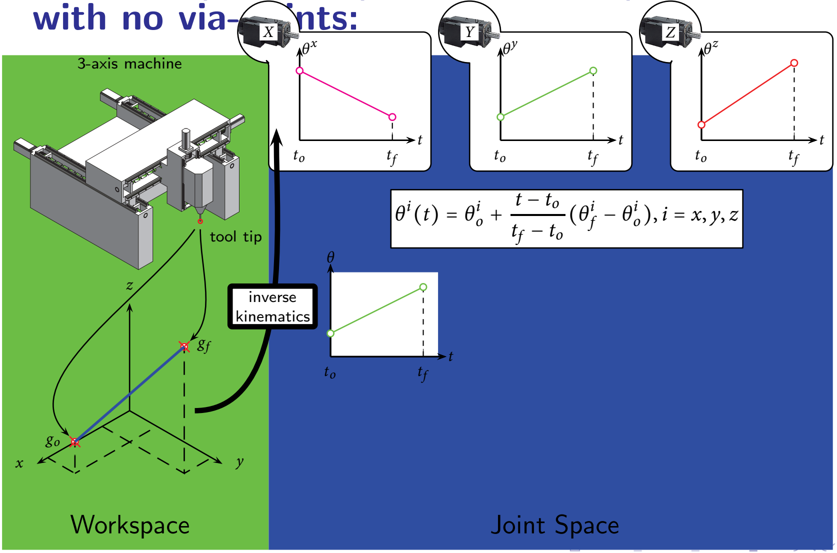
5.2 Trajectory Generation in Joint Space

□ A simple example-Linear interpolation with no via-points:



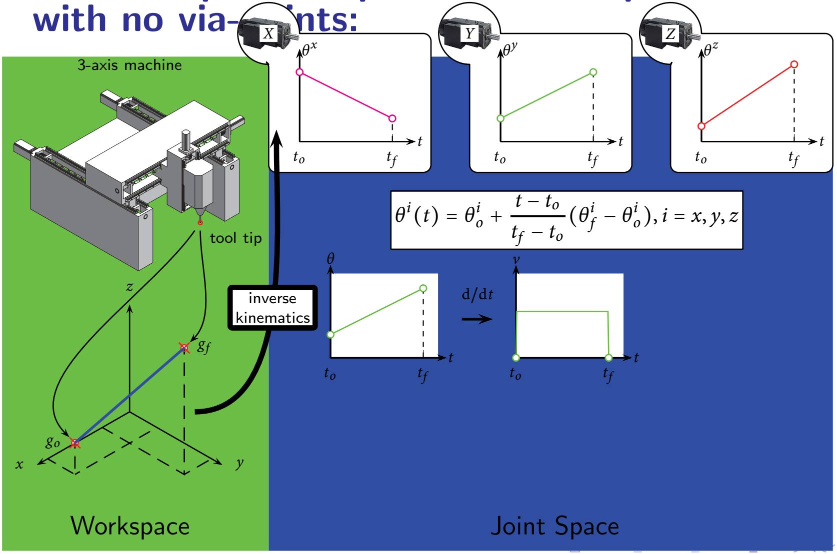
$$\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z$$

□ A simple example-Linear interpolation with no via-points:



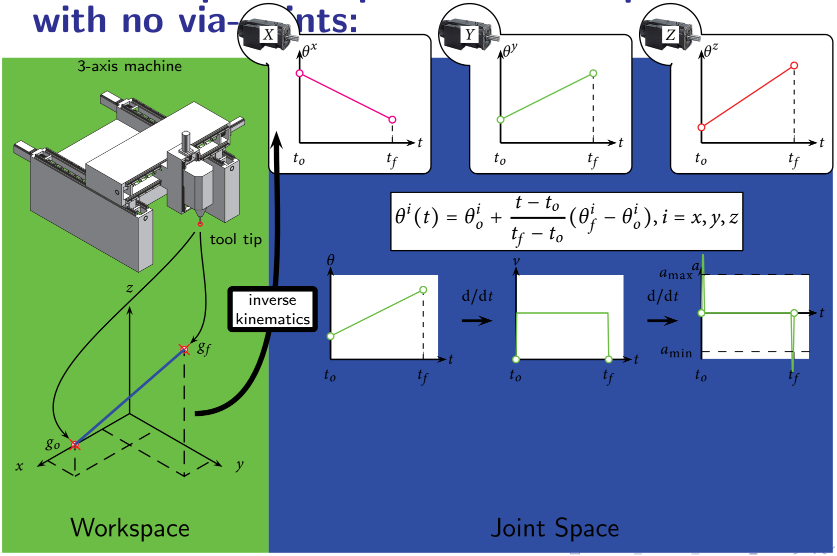
$$\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z$$

□ A simple example-Linear interpolation with no via-points:



$$\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z$$

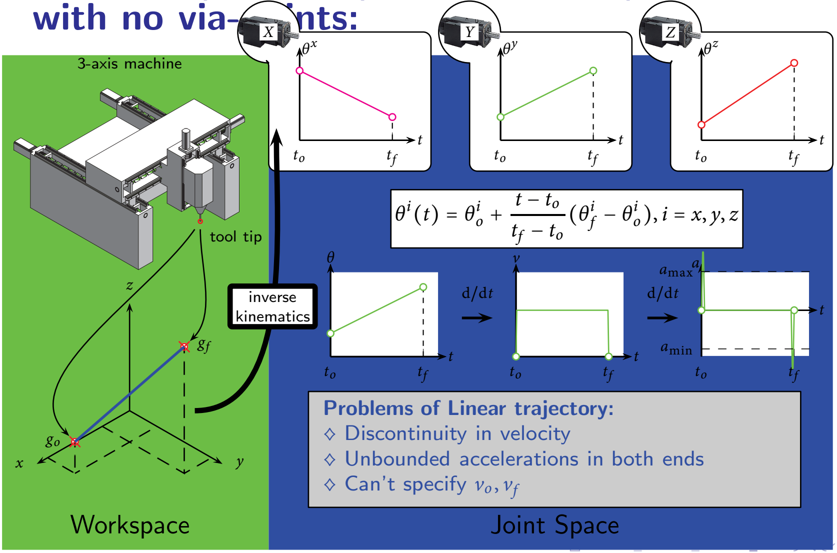
□ A simple example-Linear interpolation with no via-points:



$$\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z$$

5.2 Trajectory Generation in Joint Space

□ A simple example-Linear interpolation with no via-points:



$$\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z$$

Problems of Linear trajectory:

- ◇ Discontinuity in velocity
- ◇ Unbounded accelerations in both ends
- ◇ Can't specify v_o, v_f

Better Approaches:

1. Increase the order of the trajectory:

Better Approaches:

1. Increase the order of the trajectory:

Linear trajectory:

$$\theta(t) = \theta_o + \frac{t - t_o}{t_f - t_o}(\theta_f - \theta_o)$$

Better Approaches:

1. Increase the order of the trajectory:

Linear trajectory:

$$\theta(t) = \theta_o + \frac{t - t_o}{t_f - t_o}(\theta_f - \theta_o) = a_0 + a_1 t, a_0 = \frac{t_f\theta_o - t_o\theta_f}{t_f - t_o}, a_1 = \frac{\theta_f - \theta_o}{t_f - t_o}$$

$\rightarrow 1^{st}$ order polynomial in t, $v(t)$ const., $a(t)$ impulse at t_o, t_f.

Better Approaches:

1. Increase the order of the trajectory:

Linear trajectory:

$$\theta(t) = \theta_o + \frac{t - t_o}{t_f - t_o}(\theta_f - \theta_o) = a_0 + a_1 t, a_0 = \frac{t_f\theta_o - t_o\theta_f}{t_f - t_o}, a_1 = \frac{\theta_f - \theta_o}{t_f - t_o}$$

$\rightarrow 1^{st}$ order polynomial in t, $v(t)$ const., $a(t)$ impulse at t_o, t_f.

Higher order trajectories:

order of θ	order of v	order of a	allowable design variables
2 (parabolic)	1	0	$\theta_o, \theta_f, v_{\max}$

□ Better Approaches:

1. Increase the order of the trajectory:

Linear trajectory:

$$\theta(t) = \theta_o + \frac{t - t_o}{t_f - t_o}(\theta_f - \theta_o) = a_0 + a_1 t, a_0 = \frac{t_f \theta_o - t_o \theta_f}{t_f - t_o}, a_1 = \frac{\theta_f - \theta_o}{t_f - t_o}$$

$\rightarrow 1^{st}$ order polynomial in t, $v(t)$ const., $a(t)$ impulse at t_o, t_f.

Higher order trajectories:

order of θ	order of v	order of a	allowable design variables
2 (parabolic)	1	0	$\theta_o, \theta_f, v_{\max}$
3 (cubic)	2	1	$\theta_o, \theta_f, v_o, v_f$

Better Approaches:

1. Increase the order of the trajectory:

Linear trajectory:

$$\theta(t) = \theta_o + \frac{t - t_o}{t_f - t_o}(\theta_f - \theta_o) = a_0 + a_1 t, a_0 = \frac{t_f\theta_o - t_o\theta_f}{t_f - t_o}, a_1 = \frac{\theta_f - \theta_o}{t_f - t_o}$$

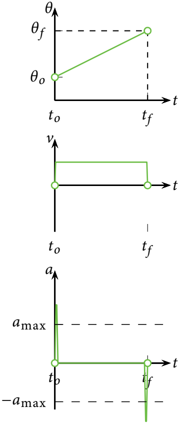
$\rightarrow 1^{st}$ order polynomial in t, $v(t)$ const., $a(t)$ impulse at t_o, t_f.

Higher order trajectories:

order of θ	order of v	order of a	allowable design variables
2 (parabolic)	1	0	$\theta_o, \theta_f, v_{\max}$
3 (cubic)	2	1	$\theta_o, \theta_f, v_o, v_f$
5 (quintic)	4	3	$\theta_o, \theta_f, v_o, v_f, a_o, a_f$

5.2 Trajectory Generation in Joint Space

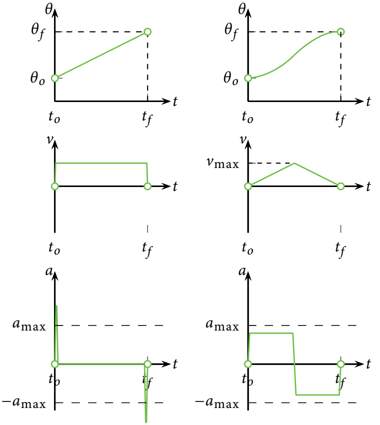
☐ Time profiles:



linear trajectory

5.2 Trajectory Generation in Joint Space

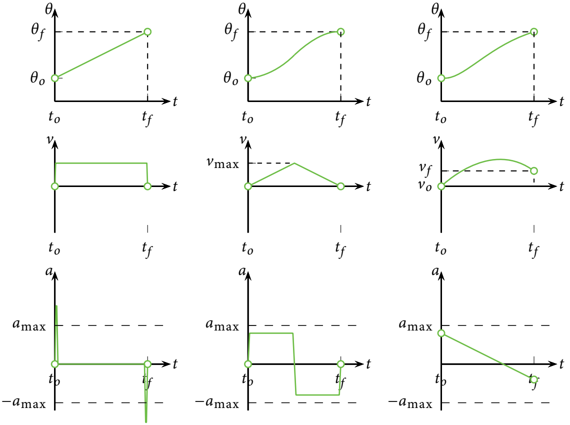
☐ Time profiles:



linear trajectory

parabolic trajectory

☐ Time profiles:

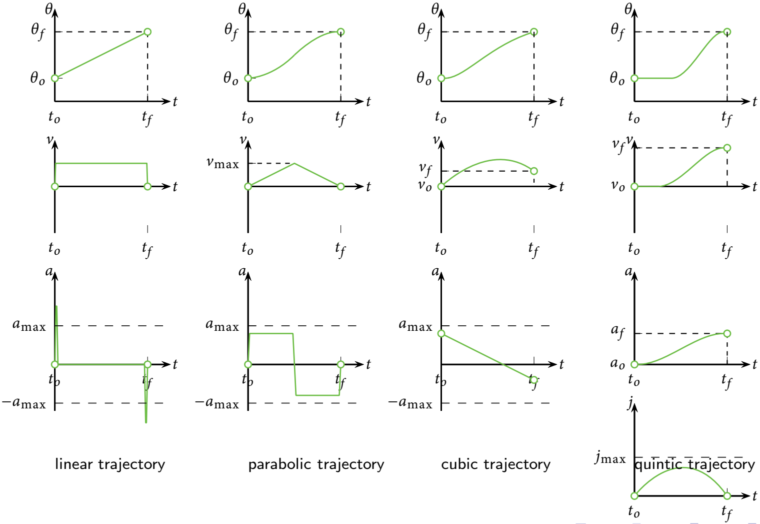


linear trajectory parabolic trajectory cubic trajectory

5.2 Trajectory Generation in Joint Space

1 Point to point trajectory generation

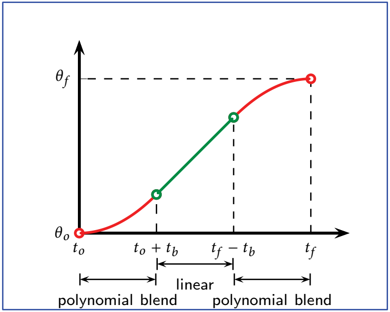
□ Time profiles:



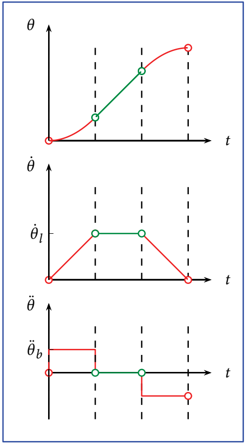
linear trajectory | parabolic trajectory | cubic trajectory | quintic trajectory

5.2 Trajectory Generation in Joint Space

2 Composition of elementary trajectories:
E.g., linear trajectory with polynomial blends:



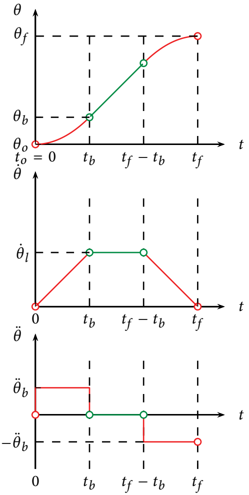
e.g. parabolic blend



◇ A list of composite trajectories:

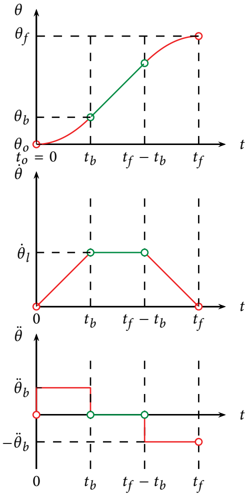
- ☐ Linear with parabolic (Trapezoidal): 2-1-2
- ☐ Linear with circular
- ☐ Linear with quintic: 5-1-5
- ☐ Linear with S (Double S): 3-2-3-1-3-2-3

□ Linear Function with Parabolic Blends (LFPB):

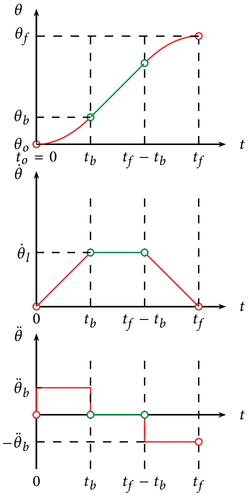


□ Linear Function with Parabolic Blends (LFPB):

◇ Input: $\theta(t_o) = \theta(0) = \theta_o$
$\theta(t_f) = \theta_f$
$t_d = t_f - t_o$: Duration of travel



□ Linear Function with Parabolic Blends (LFPB):



◇ Input: $\theta(t_o) = \theta(0) = \theta_o$
$\theta(t_f) = \theta_f$
$t_d = t_f - t_o$: Duration of travel

◇ Control Parameters:

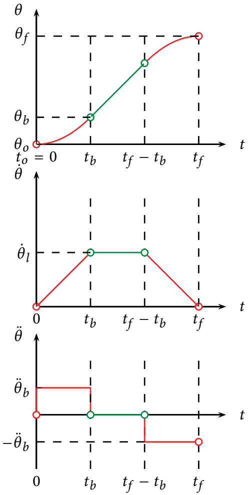
$\dot{\theta}_l(\leq \dot{\theta}_{\max})$: linear velocity

$\ddot{\theta}_b(\leq \ddot{\theta}_{\max})$: blend acceleration

$t_b(0 < t_b \leq \frac{t_d}{2})$: blend time

$t_d - 2t_b$: linear time

□ Linear Function with Parabolic Blends (LFPB):



◇ Input: $\theta(t_o) = \theta(0) = \theta_o$

$\theta(t_f) = \theta_f$

$t_d = t_f - t_o$: Duration of travel

◇ Control Parameters:

$\dot{\theta}_l(\leq \dot{\theta}_{\max})$: linear velocity

$\ddot{\theta}_b(\leq \ddot{\theta}_{\max})$: blend acceleration

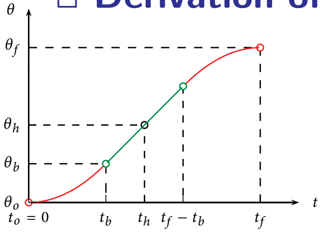
$t_b(0 < t_b \leq \frac{t_d}{2})$: blend time

$t_d - 2t_b$: linear time

◇ LFPB Trajectory:

$$\theta(t) = \begin{cases} \theta_o + \frac{1}{2}\ddot{\theta}_b(t-t_o)^2 & t_o \leq t < t_o + t_b \\ \theta_o + \ddot{\theta}_b t_b\left(t - t_o - \frac{t_b}{2}\right) & t_o + t_b \leq t < t_f - t_b \\ \theta_f - \frac{1}{2}\ddot{\theta}_b(t_f - t)^2 & t_f - t_b \leq t \leq t_f \end{cases}$$

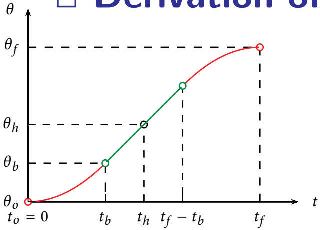
□ Derivation of the LFPB:



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, (t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)) \tag{5.2.1}$$

□ Derivation of the LFPB:



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, \left(t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)\right) \qquad (5.2.1)$$

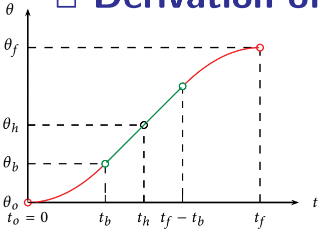
◇ Blend region:

$$\theta_b \triangleq \theta(t_b) = \theta_o + \frac{1}{2}\ddot{\theta}_b t_b^2 \qquad (5.2.2)$$

$$(5.2.1) + (5.2.2): \quad \Rightarrow \ddot{\theta}_b t_b^2 - \ddot{\theta}_b t_d t_b + (\theta_f - \theta_o) = 0$$

$$\Rightarrow t_b = \frac{t_d}{2} - \frac{\sqrt{\ddot{\theta}_b^2 t_d^2 - 4\ddot{\theta}_b(\theta_f - \theta_o)}}{2\ddot{\theta}_b} \qquad (5.2.3)$$

□ Derivation of the LFPB:



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, (t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)) \qquad (5.2.1)$$

◇ Blend region:

$$\theta_b \triangleq \theta(t_b) = \theta_o + \frac{1}{2}\ddot{\theta}_b t_b^2 \qquad (5.2.2)$$

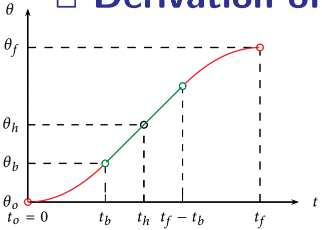
$$(5.2.1) + (5.2.2): \quad \Rightarrow \ddot{\theta}_b t_b^2 - \ddot{\theta}_b t_d t_b + (\theta_f - \theta_o) = 0$$

$$\Rightarrow t_b = \frac{t_d}{2} - \frac{\sqrt{\ddot{\theta}_b^2 t_d^2 - 4\ddot{\theta}_b(\theta_f - \theta_o)}}{2\ddot{\theta}_b} \qquad (5.2.3)$$

◇ Constraints on $\ddot{\theta}_b$:

$$\ddot{\theta}_b \geq \frac{4(\theta_f - \theta_o)}{t_d^2} \qquad (5.2.4)$$

□ Derivation of the LFPB:



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, (t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)) \qquad (5.2.1)$$

◇ Blend region:

$$\theta_b \triangleq \theta(t_b) = \theta_o + \frac{1}{2}\ddot{\theta}_b t_b^2 \qquad (5.2.2)$$

$$(5.2.1) + (5.2.2): \quad \Rightarrow \ddot{\theta}_b t_b^2 - \ddot{\theta}_b t_d t_b + (\theta_f - \theta_o) = 0$$

$$\Rightarrow t_b = \frac{t_d}{2} - \frac{\sqrt{\ddot{\theta}_b^2 t_d^2 - 4\ddot{\theta}_b(\theta_f - \theta_o)}}{2\ddot{\theta}_b} \qquad (5.2.3)$$

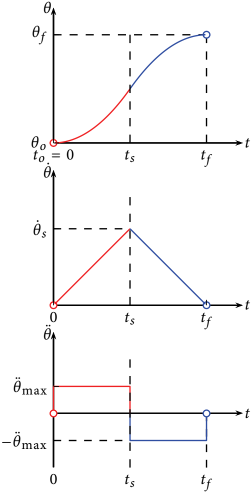
◇ Constraints on $\ddot{\theta}_b$:

$$\ddot{\theta}_b \geq \frac{4(\theta_f - \theta_o)}{t_d^2} \qquad (5.2.4)$$

◇ Observation:

- As $\ddot{\theta}_b \uparrow$, $t_b \downarrow$ and linear time $t_d - 2t_b \uparrow$; as $\ddot{\theta}_b \to \infty$, LFPB becomes linear interpolation.
- With equality in (5.2.4), linear portion of LFPB shrinks to zero.

☐ Minimum Time Trajectory (Bang Bang):



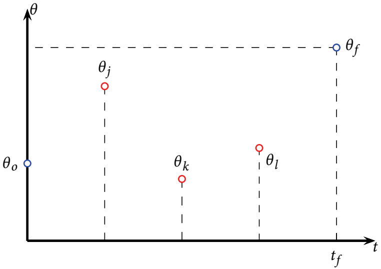
- ◇ Given: θ_o, θ_f and $\ddot{\theta}_{\max}$
- ◇ Minimize: t_d
- ◇ Solution: Bang-bang trajectory

$$\ddot{\theta}(t) = \begin{cases} \ddot{\theta}_{\max} & 0 \le t \le t_s \\ -\ddot{\theta}_{\max} & t_s \le t \le t_d \end{cases}$$

where the switching time t_s is obtained from (5.2.3)

$$\boxed{t_s = \frac{t_d}{2} = \sqrt{\frac{\theta_f - \theta_o}{\ddot{\theta}_{\max}}}}$$

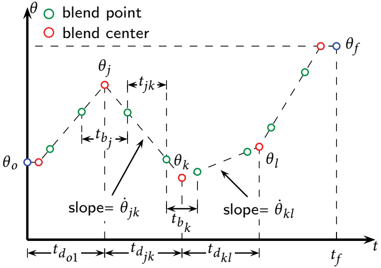
□ LFPB for a Path with Via Points ([3]):



◇ Given: $t_o, \theta_o, t_f, \theta_f, \ddot{\theta}_b$, via points $\{\theta_i\}_1^m$ at time $\{t_i\}_1^m$ (time duration $t_{d_{jk}} \triangleq t_k - t_j$)

◇ Find: $\theta(t)$ interpolating θ_o, θ_f and approximating $\{\theta_i\}_1^m$.

□ LFPB for a Path with Via Points ([3]):

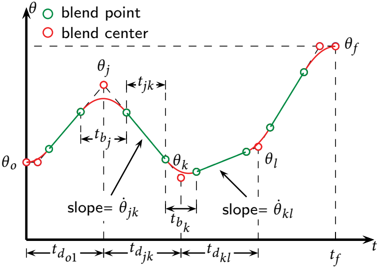


◇ Given: $t_o, \theta_o, t_f, \theta_f, \ddot{\theta}_b$, via points $\{\theta_i\}_1^m$ at time $\{t_i\}_1^m$ (time duration $t_{d_{jk}} \triangleq t_k - t_j$)

◇ Find: $\theta(t)$ interpolating θ_o, θ_f and approximating $\{\theta_i\}_1^m$.

◇ Solution: LFPB with via points

□ LFPB for a Path with Via Points ([3]):



◇ Given: $t_o, \theta_o, t_f, \theta_f, \ddot{\theta}_b$, via points $\{\theta_i\}_1^m$ at time $\{t_i\}_1^m$ (time duration $t_{d_{jk}} \triangleq t_k - t_j$)

◇ Find: $\theta(t)$ interpolating θ_o, θ_f and approximating $\{\theta_i\}_1^m$.

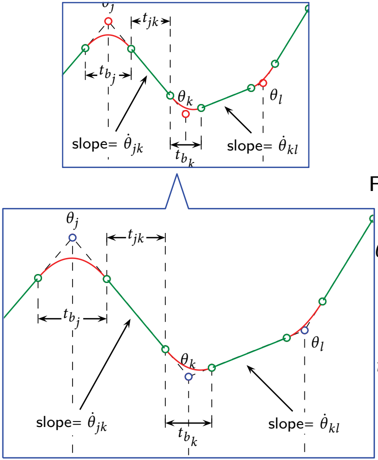
◇ Solution: LFPB with via points

□ LFPB for a Path with Via Points ([3]):

◇ Given: $t_o, \theta_o, t_f, \theta_f, \ddot{\theta}_b$, via points $\{\theta_i\}_1^m$ at time $\{t_i\}_1^m$ (time duration $t_{d_{jk}} \triangleq t_k - t_j$)

◇ Find: $\theta(t)$ interpolating θ_o, θ_f and approximating $\{\theta_i\}_1^m$.

◇ Solution: LFPB with via points

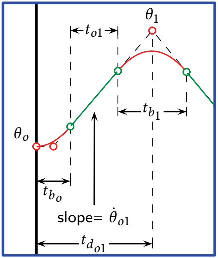


For via points $j, k, l = 1, \dots, m$:

$$\begin{aligned} \dot{\theta}_{jk} &= \frac{\theta_k - \theta_j}{t_{d_{jk}}} \quad \text{(linear vel.)} \\ \ddot{\theta}_k &= \operatorname{Sgn}(\dot{\theta}_{kl} - \dot{\theta}_{jk})\left|\ddot{\theta}_b\right| \quad \text{(Blend acc.)} \\ t_{b_k} &= \frac{\dot{\theta}_{kl} - \dot{\theta}_{jk}}{\ddot{\theta}_k} \quad \text{(Blend dur.)} \\ t_{jk} &= t_{d_{jk}} - \frac{1}{2}t_{b_j} - \frac{1}{2}t_{b_k} \quad \text{(Linear dur.)} \end{aligned} \tag{5.2.5}$$

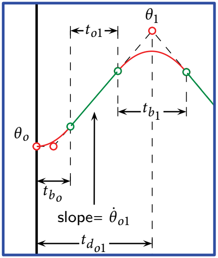
5.2 Trajectory Generation in Joint Space

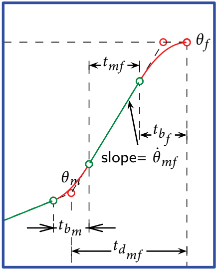
2 Linear Function with Parabolic Blends



◇ **First Segment:**

$$\begin{aligned} \ddot{\theta}_o &= \mathrm{Sgn}(\theta_1 - \theta_o)\left|\ddot{\theta}_b\right| \ \text{(Blend acc.)} \\ t_{b_o} &= t_{d_{o1}} - \sqrt{t_{d_{o1}}^2 - \frac{2(\theta_1 - \theta_o)}{\ddot{\theta}_o}} \ \text{(Blend dur.)} \\ \dot{\theta}_{o1} &= \frac{\theta_1 - \theta_o}{t_{d_{o1}} - \frac{1}{2}t_{b_o}} \ \text{(linear vel.)} \\ t_{o1} &= t_{d_{o1}} - t_{b_o} - \frac{1}{2}t_{b_1} \ \text{(linear dur.)} \end{aligned} \tag{5.2.6}$$





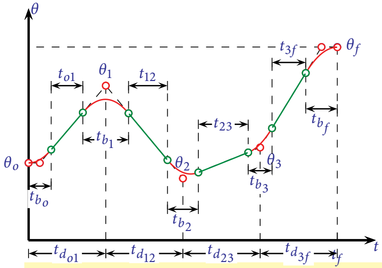
◇ **First Segment:**

$$\begin{aligned}
\ddot{\theta}_o &= \mathrm{Sgn}(\theta_1 - \theta_o)\left|\ddot{\theta}_b\right| \quad \text{(Blend acc.)} \\
t_{b_o} &= t_{d_{o1}} - \sqrt{t_{d_{o1}}^2 - \frac{2(\theta_1 - \theta_o)}{\ddot{\theta}_o}} \quad \text{(Blend dur.)} \\
\dot{\theta}_{o1} &= \frac{\theta_1 - \theta_o}{t_{d_{o1}} - \frac{1}{2}t_{b_o}} \quad \text{(linear vel.)} \\
t_{o1} &= t_{d_{o1}} - t_{b_o} - \frac{1}{2}t_{b_1} \quad \text{(linear dur.)}
\end{aligned} \tag{5.2.6}$$

◇ **Last Segment:**

$$\begin{aligned}
\ddot{\theta}_f &= \mathrm{Sgn}(\theta_m - \theta_f)\left|\ddot{\theta}_b\right| \quad \text{(Blend acc.)} \\
t_{b_f} &= t_{d_{mf}} - \sqrt{t_{d_{mf}}^2 + \frac{2(\theta_f - \theta_m)}{\ddot{\theta}_f}} \quad \text{(Blend dur.)} \\
\dot{\theta}_{mf} &= \frac{\theta_f - \theta_m}{t_{d_{mf}} - \frac{1}{2}t_{b_f}} \quad \text{(linear vel.)} \\
t_{mf} &= t_{d_{mf}} - t_{b_f} - \frac{1}{2}t_{b_m} \quad \text{(linear dur.)}
\end{aligned} \tag{5.2.7}$$

◇ Example: LFPB with 3 via points



Given:

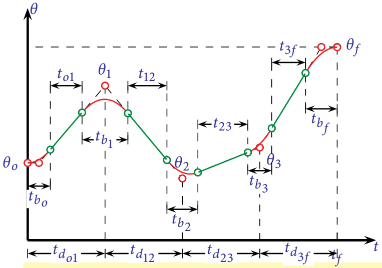
$\theta_o = 1, \theta_f = 2.5, \ddot{\theta}_b = 4, \theta_1 = 2, \theta_2 = .8, \theta_3 = 1.2$

Apply (5.2.6):

$$\ddot{\theta}_o = 4, t_{b_o} = 1 - \sqrt{1^2 - \frac{2(2-1)}{4}} = 0.29,$$

$$\dot{\theta}_{o1} = \frac{2-1}{1 - \frac{1}{2}0.29} = 1.17$$

◇ Example: LFPB with 3 via points



Given:

$\theta_o = 1, \theta_f = 2.5, \ddot{\theta}_b = 4, \theta_1 = 2, \theta_2 = .8, \theta_3 = 1.2$

Apply (5.2.6):

$$\ddot{\theta}_o = 4, t_{b_o} = 1 - \sqrt{1^2 - \frac{2(2-1)}{4}} = 0.29,$$

$$\dot{\theta}_{o1} = \frac{2-1}{1 - \frac{1}{2}0.29} = 1.17$$

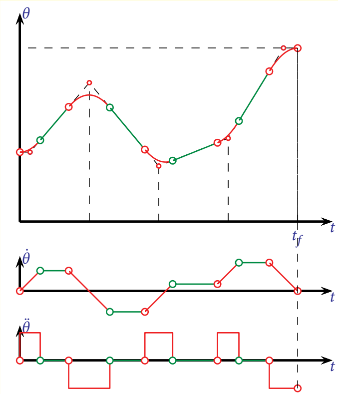
Apply (5.2.5):

$$\dot{\theta}_{12} = \frac{0.8-2}{1} = -1.2, \ddot{\theta}_1 = -4, t_{b_1} = \frac{-1.2-1.17}{-4} = 0.59, t_{o1} = 1 - 0.29 - \frac{1}{2}0.59 = 0.41$$

$$\dot{\theta}_{23} = \frac{1.2-0.8}{1} = 0.4, \ddot{\theta}_2 = 4, t_{b_2} = \frac{0.4+1.2}{4} = 0.4, t_{12} = 1 - \frac{1}{2}0.59 - \frac{1}{2}0.4 = 0.51$$

(Continues next slide)

◇ Example: LFPB with 3 via points (continued)



Apply (5.2.7):

$$\ddot{\theta}_f = -4,\ t_{b_f} = 1-\sqrt{1^2 + \frac{2(2.5-1.2)}{-4}} = 0.41,$$

$$\dot{\theta}_{3f} = \frac{2.5-1.2}{1-\frac{1}{2}0.41} = 1.63$$

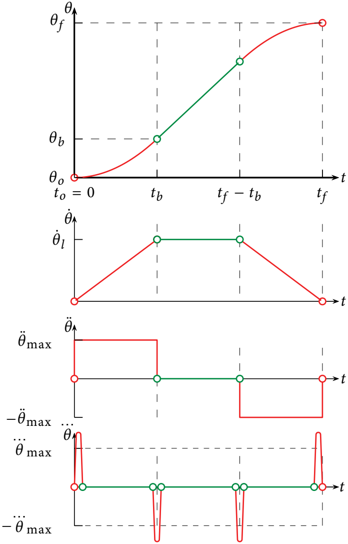
Apply (5.2.5):

$$\ddot{\theta}_3 = 4,\ t_{b_3} = \frac{1.63-0.4}{4} = 0.31,$$

$$t_{3f} = 1 - \frac{1}{2}0.31 - 0.41 = 0.44,$$

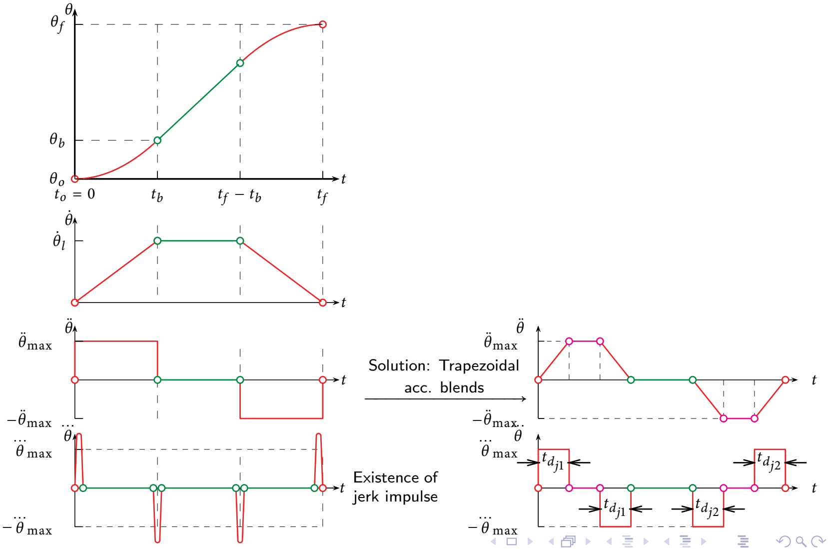
$$t_{23} = 1 - \frac{1}{2}0.4 - \frac{1}{2}0.31 = 0.65$$

Disadvantage of LFPB:

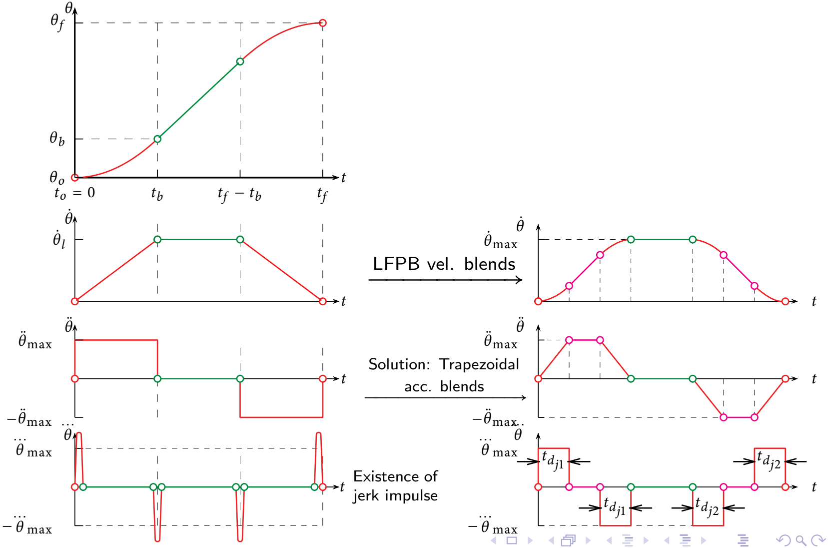


Existence of jerk impulse

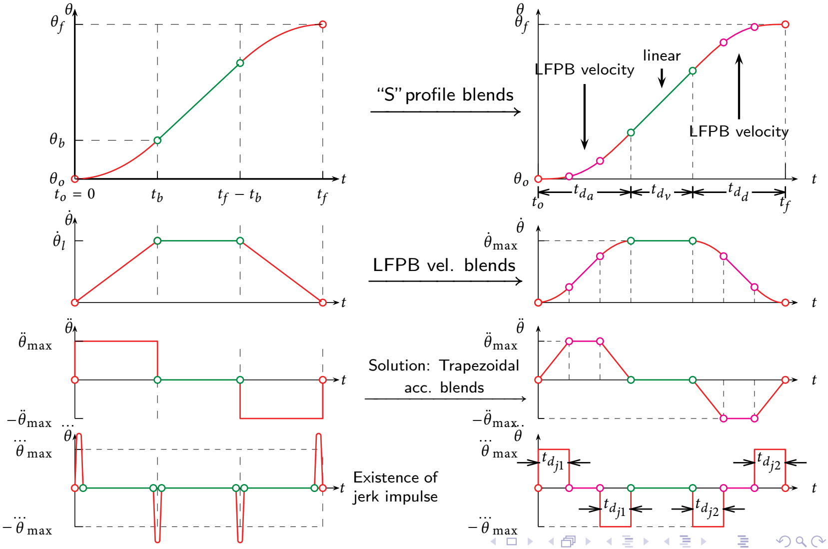
□ Disadvantage of LFPB:



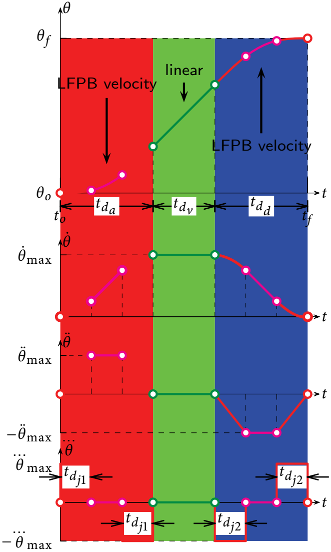
□ Disadvantage of LFPB:



□ Disadvantage of LFPB:



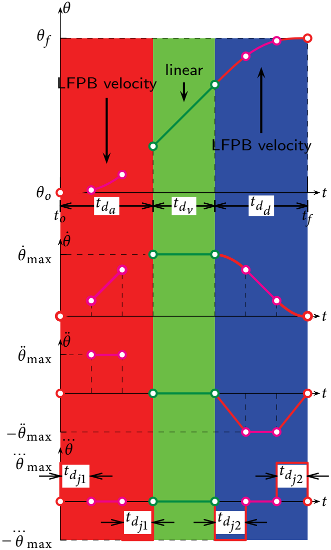
□ Linear function with Double S traj.:



♢ **Double "S" trajectory:** Linear trajectory with LFPB velocity blends

♢ **Advantage over LFPB:** Bounded jerk

□ Linear function with Double S traj.:



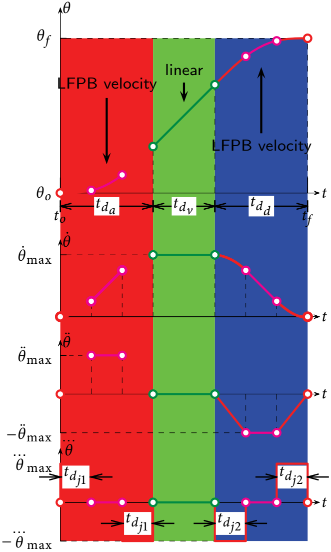
◇ **Double "S" trajectory:** Linear trajectory with LFPB velocity blends

◇ **Advantage over LFPB:** Bounded jerk

◇ **Input:**

$$\theta_o, \theta_f, \dot\theta_o, \dot\theta_f, \ddot\theta_o, \ddot\theta_f, \dot\theta_{\max}, \ddot\theta_{\max}, \dddot\theta_{\max}$$

□ Linear function with Double S traj.:



◇ **Double "S" trajectory:** Linear trajectory with LFPB velocity blends

◇ **Advantage over LFPB:** Bounded jerk

◇ **Input:**

$$\theta_o, \theta_f, \dot\theta_o, \dot\theta_f, \ddot\theta_o, \ddot\theta_f, \dot\theta_{max}, \ddot\theta_{max}, \dddot\theta_{max}$$

◇ **Output (for details see [4]):**

t_{d_a}: Acceleration duration

t_{d_v}: Linear duration

t_{d_d}: Deceleration duration

$t_{d_{j1}}, t_{d_{j2}}$: Jerk duration for acceleration and deceleration

◇ **Generalization:** Double S with via points (similar to LFPB with via points)

☐ Computation of the double S trajectory ($\theta_f > \theta_0$):

Notations:

$\dot{\theta}_{\lim}(\leq \dot{\theta}_{\max})$:	maximal velocity
$\ddot{\theta}_{\lim_a}(\leq \ddot{\theta}_{\max})$:	maximal acceleration in the acceleration phase
$\ddot{\theta}_{\lim_d}(\leq \ddot{\theta}_{\max})$:	maximal acceleration in the deceleration phase

Acceleration phase:

$$\theta(t) = \begin{cases} \theta_o + \dot{\theta}_o t + \dddot{\theta}_{\max}\frac{t^3}{6} & t \in [0, t_{d_{j1}}] \\ \theta_o + \dot{\theta}_o t + \frac{\ddot{\theta}_{\lim_a}}{6}(3t^2 - 3t_{d_{j1}}t + t_{d_{j1}}^2) & t \in [t_{d_{j1}}, t_{d_a} - t_{d_{j1}}] \\ \theta_o + (\dot{\theta}_{\lim} + \dot{\theta}_o)\frac{t_{d_a}}{2} - \dot{\theta}_{\lim}(t_{d_a} - t) - \dddot{\theta}_{\max}\frac{(t_{d_a} - t)^3}{6} & t \in [t_{d_a} - t_{d_{j1}}, t_{d_a}] \end{cases}$$

(Continues next slide)

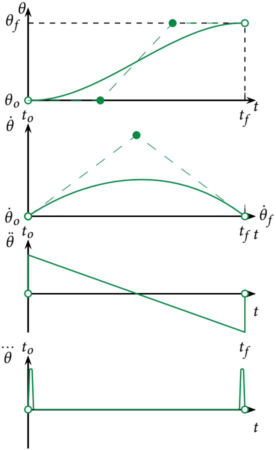
Constant velocity phase:

$$\theta(t) = \theta_o + (\dot{\theta}_{\lim} + \dot{\theta}_o)\frac{t_{d_a}}{2} + \dot{\theta}_{\lim}(t - t_{d_a}), t \in [t_{d_a}, t_{d_a} + t_{d_v}]$$

Deceleration phase: Define $t_d = t_{d_a} + t_{d_v} + t_{d_d}$:

$$\theta(t) = \begin{cases} \theta_f - (\dot{\theta}_{\lim} + \dot{\theta}_f)\frac{t_{d_d}}{2} + \dot{\theta}_{\lim}(t - t_d + t_{d_d}) - \dddot{\theta}_{\max}\frac{(t-t_d+t_{d_d})^3}{6} & t \in [t_{d_a} + t_{d_v}, t_{d_a} + t_{d_v} + t_{d_{j2}}] \\ \theta_f - (\dot{\theta}_{\lim} + \dot{\theta}_f)\frac{t_{d_a}}{2} + \dot{\theta}_{\lim}(t - t_d + t_{d_d}) + \frac{\ddot{\theta}_{\lim_d}}{6}\left(3(t - t_d + t_{d_d})^2 - 3t_{d_{j2}}(t - t_d - t_{d_d}) + t_{d_{j2}}^2\right) & t \in [t_{d_a} + t_{d_v} + t_{d_{j2}}, t_d - t_{d_{j2}}] \\ \theta_f - \dot{\theta}_f(t_d - t) - \dddot{\theta}_{\max}\frac{(t_d-t)^3}{6} & t \in [t_d - t_{d_{j2}}, t_d] \end{cases}$$

□ Cubic polynomial trajectory:



$$\boxed{\begin{aligned}\theta(t) = a_0 + a_1(t-t_o) + a_2(t-t_o)^2 + a_3(t-t_o)^3,\\ t\in[t_o,t_f], t_d \triangleq t_f - t_o\end{aligned}}$$

where 4 parameters a_0, a_1, a_2, a_3 are to be determined by boundary conditions.
Property: **bounded acceleration**, **jerk impulse at both ends**.

$$\begin{cases} \theta(t_o) = a_0 = \theta_o \\ \dot\theta(t_o) = a_1 = \dot\theta_o \\ \theta(t_f) = a_0 + a_1 t_d + a_2 t_d^2 + a_3 t_d^3 = \theta_f \\ \dot\theta(t_f) = a_1 + 2a_2 t_d + 3a_3 t_d^2 = \dot\theta_f \end{cases} \Rightarrow \begin{cases} a_0 = \theta_o \\ a_1 = \dot\theta_o \\ a_2 = \dfrac{3h - (2\dot\theta_o + \dot\theta_f)t_d}{t_d^2} \\ a_3 = \dfrac{-2h + (\dot\theta_o + \dot\theta_f)t_d}{t_d^3} \end{cases}$$

(Continuous next slide)

Multipoint Cubic interpolation:

Given $\theta_o, \theta_f, \dot{\theta}_o, \dot{\theta}_f$ at t_o, t_f and via points $\{\theta_k\}_1^m$ at time $\{t_k\}_1^m$, solve $\boxed{a_{0k} + a_{1k}(t - t_k) + a_{2k}(t - t_k)^2 + a_{3k}(t - t_k)^3}$ for the unknowns $\{a_{0k}, a_{1k}, a_{2k}, a_{3k}\}_o^m$.

1. If via-point velocities $\{\dot{\theta}_k\}_1^m$ are directly assigned by user, solve the $m+1$ BVPs:

$$\begin{cases} a_{0k} = \theta_o, & a_{1k} = \dot{\theta}_o \\ a_{2k} = \dfrac{3h - (2\dot{\theta}_o + \dot{\theta}_f)t_d}{t_d^2}, & a_{3k} = \dfrac{-2h + (2\dot{\theta}_o + \dot{\theta}_f)t_d}{t_d^3} \end{cases}, k = o, 1, \ldots, m$$

◻ Multipoint Cubic interpolation:

Given $\theta_o, \theta_f, \dot{\theta}_o, \dot{\theta}_f$ at t_o, t_f and via points $\{\theta_k\}_1^m$ at time $\{t_k\}_1^m$, solve $\boxed{a_{0k} + a_{1k}(t - t_k) + a_{2k}(t - t_k)^2 + a_{3k}(t - t_k)^3}$ for the unknowns $\{a_{0k}, a_{1k}, a_{2k}, a_{3k}\}_o^m$.

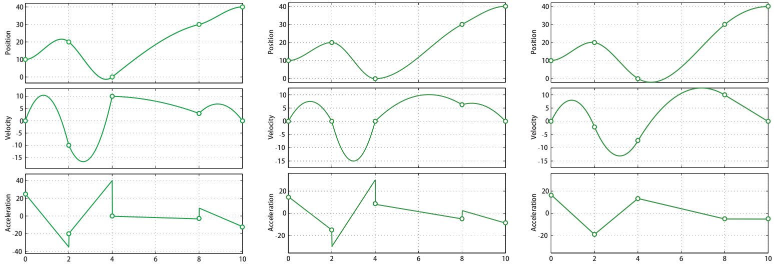
1. If via-point velocities $\{\dot{\theta}_k\}_1^m$ are directly assigned by user, solve the $m + 1$ BVPs:

$$\begin{cases} a_{0k} = \theta_o, & a_{1k} = \dot{\theta}_o \\ a_{2k} = \dfrac{3h - (2\dot{\theta}_o + \dot{\theta}_f)t_d}{t_d^2}, & a_{3k} = \dfrac{-2h + (2\dot{\theta}_o + \dot{\theta}_f)t_d}{t_d^3} \end{cases}, k = o, 1, \ldots, m$$

2. If only $\dot{\theta}_o, \dot{\theta}_f$ are given:
   1. compute $\{\dot{\theta}_k\}_1^m$ using a heuristic method; or
   2. design $\{\dot{\theta}_k\}_1^m$ so as to achieve acceleration continuity

(Continuous next slide)

□ Example: Cubic interpolation with 3 via points:

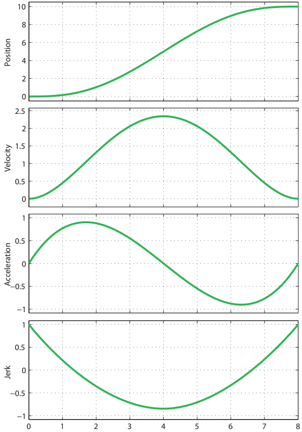


Approach 1 Approach 2 Approach 3

In Approach 1, via point velocities are arbitrarily assigned. This may lead to large and discontinuous accelerations. In Approach 2, $\dot{\theta}_k = 0$, if $\mathrm{Sign}(d_k) \neq \mathrm{Sign}(d_{k+1})$, and $\frac{1}{2}(d_k + d_{k+1})$, otherwise. Here $d_k = \frac{\theta_k - \theta_{k-1}}{t_{d_{k-1,k}}}$ is the slope from θ_{k-1} to θ_k. Note the discontinuity in acceleration. In Approach 3, we choose the polynomials so that acceleration is continuous.

□ Quintic polynomial trajectory:

$$\boxed{\theta(t) = a_0 + a_1(t-t_o) + a_2(t-t_o)^2 + a_3(t-t_o)^3 + a_4(t-t_o)^4 + a_5(t-t_o)^5, t \in [t_o, t_f]}$$



with 6 unknowns coefficients $a_i, i = 0, \ldots, 5$.

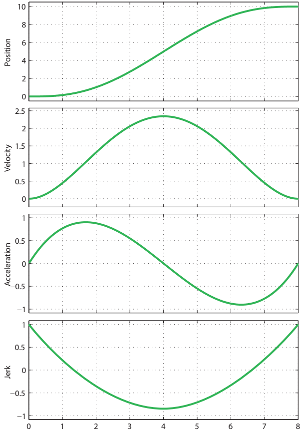
Properties:

◇ Smooth and bounded jerk

◇ Acc. continuity in composite curves.

□ Quintic polynomial trajectory:

$$\boxed{\theta(t) = a_0 + a_1(t - t_o) + a_2(t - t_o)^2 + a_3(t - t_o)^3 + a_4(t - t_o)^4 + a_5(t - t_o)^5, t \in [t_o, t_f]}$$



with 6 unknowns coefficients $a_i, i = 0, \ldots, 5$.

Properties:

◇ Smooth and bounded jerk

◇ Acc. continuity in composite curves.

Boundary conditions:

$$\theta(t_o) = \theta_o, \quad \theta(t_f) = \theta_f$$
$$\dot{\theta}(t_o) = \dot{\theta}_o, \quad \dot{\theta}(t_f) = \dot{\theta}_f$$
$$\ddot{\theta}(t_o) = \ddot{\theta}_o, \quad \ddot{\theta}(t_f) = \ddot{\theta}_f$$

(Continues next slide)

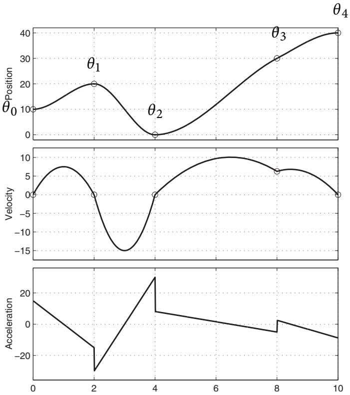
□ Quintic polynomial trajectory:

Define $t_d \triangleq t_f - t_o, h \triangleq \theta_f - \theta_o$, then:

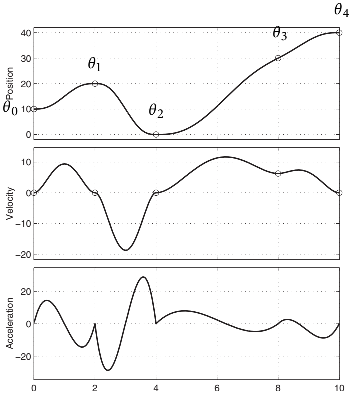
$$
\begin{aligned}
a_0 &= \theta_o \\
a_1 &= \dot{\theta}_o \\
a_2 &= \frac{1}{2}\ddot{\theta}_o \\
a_3 &= \frac{1}{2t_d^3}[20h - (8\dot{\theta}_f + 12\dot{\theta}_o)t_d - (3\ddot{\theta}_o - \ddot{\theta}_f)t_d^2] \\
a_4 &= \frac{1}{2t_d^4}[-30h - (14\dot{\theta}_f + 16\dot{\theta}_o)t_d - (3\ddot{\theta}_o - 2\ddot{\theta}_f)t_d^2] \\
a_5 &= \frac{1}{2t_d^5}[12h - 6(\dot{\theta}_f + \dot{\theta}_o)t_d - (\ddot{\theta}_f - \ddot{\theta}_o)t_d^2]
\end{aligned}
$$

5.2 Trajectory Generation in Joint Space

☐ Comparison of Cubic and Quintic Composites:



Composition of cubic polynomials: acceleration discontinuity.

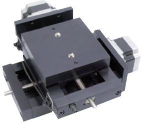


Composition of quintic polynomials: continuity in acceleration.

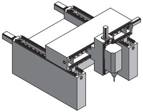
Section 5.3

☐ Types of Task Space:

1. **Euclidean space:**



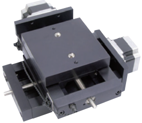
xy table ($\mathbb{R}^2$)

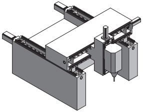


3-axis machine ($\mathbb{R}^3$)

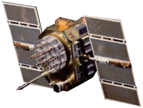
☐ Types of Task Space:

1. **Euclidean space:**


xy table ($\mathbb{R}^2$)


3-axis machine ($\mathbb{R}^3$)

2. **Subgroups (of $SE(3)$):**

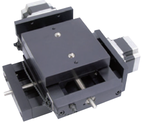

Satellite ($SO(3)$)

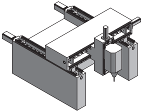

pick-and-place (X)


6 dof robot ($SE(3)$)

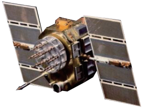
□ Types of Task Space:

1. **Euclidean space:**


xy table ($\mathbb{R}^2$)


3-axis machine ($\mathbb{R}^3$)

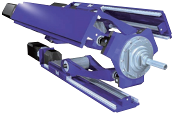
2. **Subgroups (of $SE(3)$):**

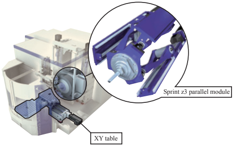

Satellite ($SO(3)$)


pick-and-place (X)


6 dof robot ($SE(3)$)

3. **Submanifolds of $SE(3)$:**


tooling module ($SE(3)/PL(z)$)



five-axis machining ($SE(3)/R(o,z)$)

□ Trajectory Generation in $\mathbb{R}^n$:

◇ **A trajectory in** $\mathbb{R}^3$ $p : [t_o, t_f] \mapsto \mathbb{R}^n$

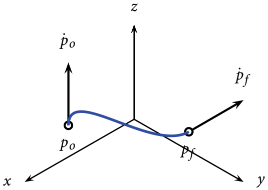
$$\text{e.g. } p(t) = \begin{bmatrix} a_{01} \\ \vdots \\ a_{0n} \end{bmatrix} + \begin{bmatrix} a_{11} \\ \vdots \\ a_{1n} \end{bmatrix} (t - t_o) + \cdots + \begin{bmatrix} a_{m1} \\ \vdots \\ a_{mn} \end{bmatrix} (t - t_o)^m, t \in [t_o, t_f]$$

◇ **A cubic example:**

Given $p_o, p_f, \dot{p}_o, \dot{p}_f, t_o, t_f, t_d = t_f - t_d, \vec{h} = p_f - p_o$, generate:

$$\vec{a}_o + \vec{a}_1(t - t_o) + \vec{a}_2(t - t_o)^2 + \vec{a}_3(t - t_o)^3, t \in [0, 1], \vec{a}_i \in \mathbb{R}^n$$

$$\Rightarrow \begin{cases} \vec{a}_0 = p_o \\ \vec{a}_1 = \dot{p}_o \\ \vec{a}_2 = \dfrac{3\vec{h} - (2\dot{p}_o + \dot{p}_f)t_d}{t_d^2} \\ \vec{a}_3 = \dfrac{-2\vec{h} + (\dot{p}_o + \dot{p}_f)t_d}{t_d^3} \end{cases}$$



For more information, see [4].

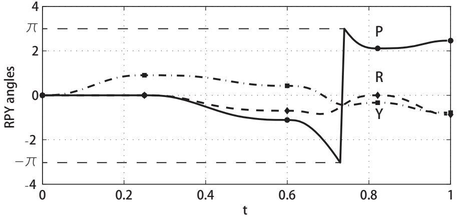
□ Trajectory Generation in $SO(3)$:

A naive approach:

Generate a trajectory using Euler angles, e.g., roll-pitch-yaw (RPY) angles or ZYZ angles.

Problems:

1. Parametrization singularity!



e.g., RPY angles, defined on $[-\pi, \pi]^3$ encounter a parametrization singularity

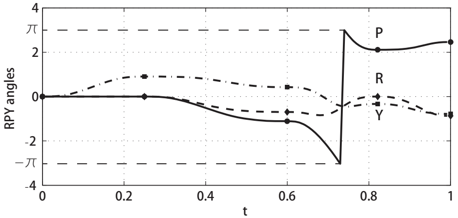
□ Trajectory Generation in $SO(3)$:

A naive approach:

Generate a trajectory using Euler angles, e.g., roll-pitch-yaw (RPY) angles or ZYZ angles.

Problems:

1. Parametrization singularity!



e.g., RPY angles, defined on $[-\pi, \pi]^3$ encounter a parametrization singularity

2. Derivatives of the Euler angles have no physical meaning!

□ Trajectory Generation in $SO(3)$:

A more meaningful approach:

1. Choose physically meaningful coordinates;
2. Add via-points to avoid parametrization singularity;
3. Generate trajectory and use inverse kinematics to obtain joint trajectory

□ Trajectory Generation in $SO(3)$:

A more meaningful approach:

1. Choose physically meaningful coordinates;
2. Add via-points to avoid parametrization singularity;
3. Generate trajectory and use inverse kinematics to obtain joint trajectory

Candidate coordinates:

- Unit quaternion:

$$Q(R) = (\cos\frac{\theta}{2}, \omega\sin\frac{\theta}{2}), \hat{\omega} = \frac{R - R^T}{2\sin\theta}, \theta = \arccos\frac{\mathrm{Tr}R - 1}{2}$$

□ Trajectory Generation in $SO(3)$:

A more meaningful approach:

1. Choose physically meaningful coordinates;
2. Add via-points to avoid parametrization singularity;
3. Generate trajectory and use inverse kinematics to obtain joint trajectory

Candidate coordinates:

- Unit quaternion:

$$Q(R) = (\cos\frac{\theta}{2}, \omega\sin\frac{\theta}{2}), \hat{\omega} = \frac{R - R^T}{2\sin\theta}, \theta = \arccos\frac{\text{Tr}R - 1}{2}$$

- Canonical coordinate:

$$\hat{r}(R) = \log R = \hat{\omega}\theta, \hat{\omega} = \frac{R - R^T}{2\sin\theta}, \theta = \arccos\frac{\text{Tr}R - 1}{2}$$

◇ Example: A cubic traj. on $SO(3)$

Given R_0, R_1 and $\omega_0 = R^T(0)\dot{R}(0)$, $\omega_1 = R^T(1)\dot{R}(1)$, consider a *minimum angular acceleration curve*:

$$R(t) = R_0 e^{\hat{r}(t)}, t \in [0, 1]$$

that minimizes $\int_0^1 \dot{\omega}^T \dot{\omega} dt$.

◇ Example: A cubic traj. on $SO(3)$

Given R_0, R_1 and $\omega_0 = R^T(0)\dot{R}(0)$, $\omega_1 = R^T(1)\dot{R}(1)$, consider a *minimum angular acceleration curve*:

$$R(t) = R_0 e^{\hat{r}(t)}, t \in [0,1]$$

that minimizes $\int_0^1 \dot{\omega}^T \dot{\omega} \mathrm{d}t$.

□ **Exact solution [5]:**

$$\omega^{(3)} + \omega \times \ddot{\omega} = 0 \tag{5.3.8}$$

which is hard to solve.

◇ Example: A cubic traj. on $SO(3)$

Given R_0, R_1 and $\omega_0 = R^T(0)\dot{R}(0)$, $\omega_1 = R^T(1)\dot{R}(1)$, consider a *minimum angular acceleration curve*:

$$R(t) = R_0 e^{\hat{r}(t)}, t \in [0,1]$$

that minimizes $\int_0^1 \dot{\omega}^T \dot{\omega} \mathrm{d}t$.

□ **Exact solution [5]:**

$$\omega^{(3)} + \omega \times \ddot{\omega} = 0 \tag{5.3.8}$$

which is hard to solve.

□ **Approximate Solution [6]:**

$$r(0) = 0, r(1) = \log(R_0^T R_1)^\vee, \omega = A(r)\dot{r},$$

$$A(r) = I + \frac{\cos\|r\| - 1}{\|r\|^2}\hat{r} + \frac{\|r\| - \sin\|r\|}{\|r\|^3}\hat{r}^2 \quad r \neq 0, A(0) = I$$

(Continues next slide)

◇ Example: A cubic traj. on $SO(3)$ (ctned)

◇ Approximation of $\dot{\omega}$:

$$\dot{\omega} \approx \ddot{r}$$

$$(5.3.8) : \omega^{(3)} + \omega \times \ddot{\omega} \approx \omega^{(3)} = r^{(4)} = 0$$

which shows that r is a cubic curve:

$$r(t) = at^3 + bt^2 + ct, t \in [0, 1]$$

◇ Example: A cubic traj. on $SO(3)$ (ctned)

◇ Approximation of $\dot{\omega}$:

$$\dot{\omega} \approx \ddot{r}$$

$$(5.3.8): \omega^{(3)} + \omega \times \ddot{\omega} \approx \omega^{(3)} = r^{(4)} = 0$$

which shows that r is a cubic curve:

$$r(t) = at^3 + bt^2 + ct, t \in [0, 1]$$

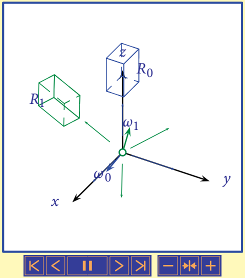
◇ Approximate solution:

$$\begin{aligned} \dot{r}(0) &= c = \omega_0 \\ r(1) &= a + b + c = \log(R_0^T R_1)^\vee \\ \dot{r}(1) &= 3a + 2b + c = A^{-1}(r(1))\omega_1 \end{aligned}$$

(Continues next slide)

◇ Example: A cubic traj. on $SO(3)$ (ctned)

Example:

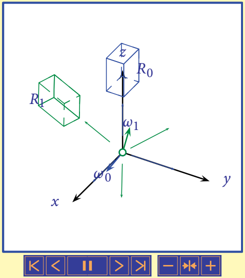


$$\log(R_0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \log(R_1) = \begin{bmatrix} 0.6 \\ 0.4 \\ 0.4 \end{bmatrix},$$

$$\omega_0 = c = \begin{bmatrix} 0.5 \\ 0.1 \\ 0.1 \end{bmatrix}, \omega_1 = \begin{bmatrix} 0.2 \\ 0.2 \\ 0.5 \end{bmatrix},$$

◇ Example: A cubic traj. on $SO(3)$ (ctned)

Example:



$$\log(R_0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \log(R_1) = \begin{bmatrix} 0.6 \\ 0.4 \\ 0.4 \end{bmatrix},$$

$$\omega_0 = c = \begin{bmatrix} 0.5 \\ 0.1 \\ 0.1 \end{bmatrix}, \omega_1 = \begin{bmatrix} 0.2 \\ 0.2 \\ 0.5 \end{bmatrix},$$

$$a + b + c = \log(R_0^T R_1)^\vee = \begin{bmatrix} 0.6 \\ 0.4 \\ 0.4 \end{bmatrix},$$

$$3a + 2b + c = A^{-1}(r(1))\omega_1 = \begin{bmatrix} 0.2688 \\ 0.0920 \\ 0.5048 \end{bmatrix}$$

$$\Rightarrow a = \begin{bmatrix} -0.4312 \\ -0.6080 \\ -0.1952 \end{bmatrix}, b = \begin{bmatrix} 0.5312 \\ 0.9080 \\ 0.4952 \end{bmatrix}$$

□ Traj. Generation on $SE(3)$:

◇ Candidate approaches:

1. Observe that $SE(3) \cong \mathbb{R}^3 \rtimes SO(3)$, we can interpolate position ($\mathbb{R}^3$) and orientation ($SO(3)$) separately.

□ Traj. Generation on $SE(3)$:

◇ Candidate approaches:

1. Observe that $SE(3) \cong \mathbb{R}^3 \rtimes SO(3)$, we can interpolate position ($\mathbb{R}^3$) and orientation ($SO(3)$) separately.

2. Canonical coordinate ([5]):

$$\xi \in \mathbb{R}^6 \mapsto e^{\hat{\xi}} \in SE(3)$$

□ Traj. Generation on $SE(3)$:

◇ Candidate approaches:

1. Observe that $SE(3) \cong \mathbb{R}^3 \rtimes SO(3)$, we can interpolate position ($\mathbb{R}^3$) and orientation ($SO(3)$) separately.

2. Canonical coordinate ([5]):

$$\xi \in \mathbb{R}^6 \mapsto e^{\hat{\xi}} \in SE(3)$$

3. Frenet frame following ([4]):

$$g(t) = \begin{bmatrix} R(t) & p(t) \\ 0 & 1 \end{bmatrix}, R(t) = [T \; N \; B], T = \frac{\dot{p}(t)}{\|\dot{p}(t)\|}$$

$$\begin{bmatrix} \dot{T} \\ \dot{N} \\ \dot{B} \end{bmatrix} = \begin{bmatrix} 0 & k & 0 \\ -k & 0 & -\tau \\ 0 & \tau & 0 \end{bmatrix} \begin{bmatrix} T \\ N \\ B \end{bmatrix},$$

5.4 References

[1] J.C. Latombe. *Robot motion planning*. Springer Verlag, 1990.

[2] Z.X. Li and J.F. Canny. *Nonholonomic motion planning*. Kluwer Academic, 1993.

[3] J. Craig. *Introduction to Robotics: Mechanics and Control*. 2nd ed. Prentice Hall, 2005.

[4] L. Biagiotti and C. Melchiorri. *Trajectory Planning for Automatic Machines and Robots*. Springer Verlag, 2008.

[5] M. Zefran and V. Kumar. "Interpolation schemes for rigid body motions". In: *CAD Computer Aided Design* 30.3 (1998), pp. 179–189.

[6] F.C. Park and B. Ravani. "Smooth Invariant Interpolation of Rotations". In: *ACM Transactions on Graphics* 16.3 (1997), pp. 277–295.

[7] B. Siciliano and O. Khatib. *Springer Handbook of Robotics*. Springer Verlag, 2008.