Chapter 5 Manipulator Control

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]

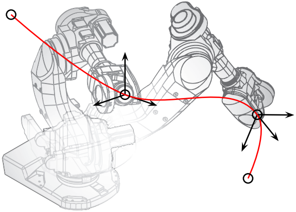
[1]ECE, Hong Kong University of Science & Technology

July 26, 2012

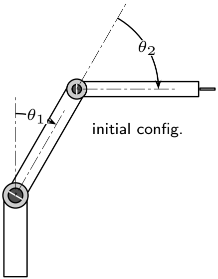
Table of Contents

Chapter 5 Manipulator Control

1. Trajectory Generation
   - Trajectory Generation in Joint Space
   - Trajectory Generation in Task Space
2. Position Control
   - Position control in joint space
   - Position control in task space
3. Hybrid Position/Force Control
   - Hybrid velocity/force control (Liu G.F., Li Z.X.)
   - Gauge invariant formulation (Aghili, F.)
   - References

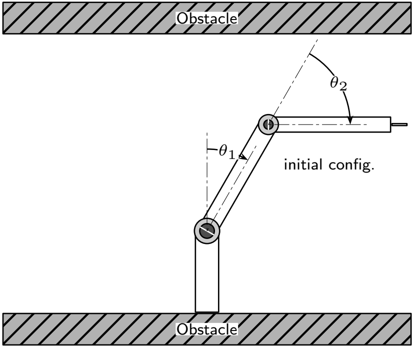


A Simple Robot Path Planning Example



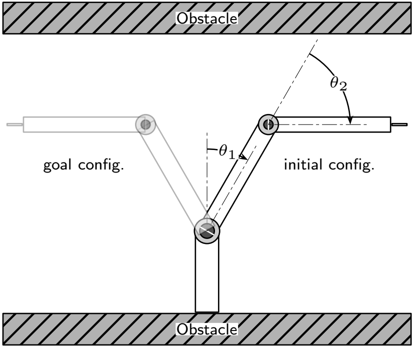
A Two-DoF robot arm

A Simple Robot Path Planning Example



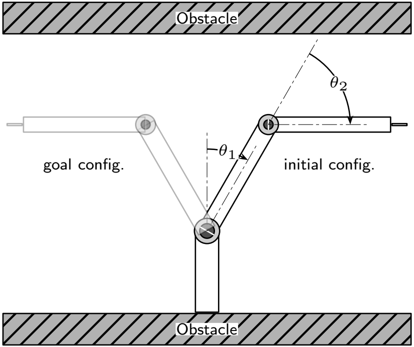
A Two-DoF robot arm

A Simple Robot Path Planning Example

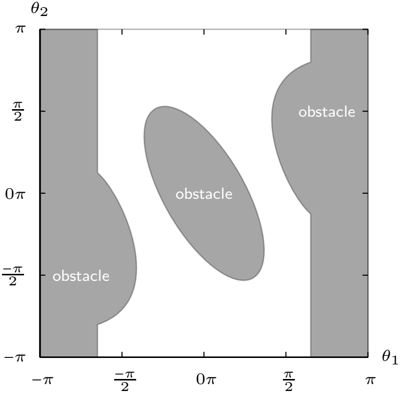


A Two-DoF robot arm

A Simple Robot Path Planning Example

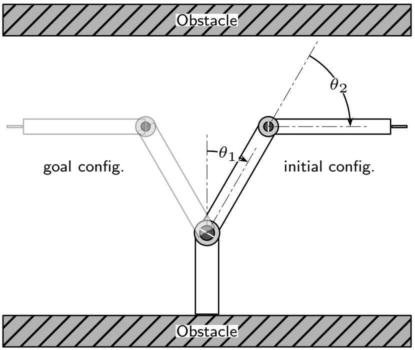


A Two-DoF robot arm

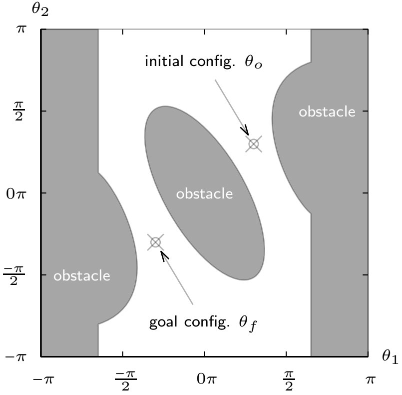


Configuration space $\mathcal{C}$

A Simple Robot Path Planning Example

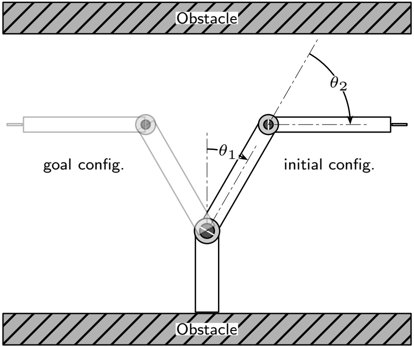


A Two-DoF robot arm

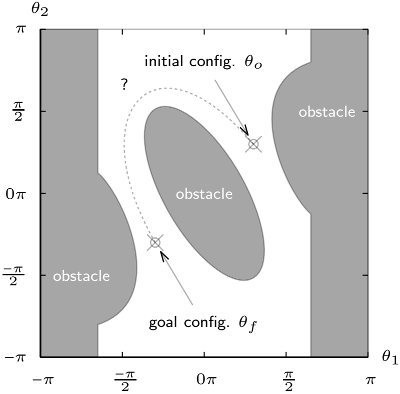


Configuration space $\mathcal{C}$

A Simple Robot Path Planning Example

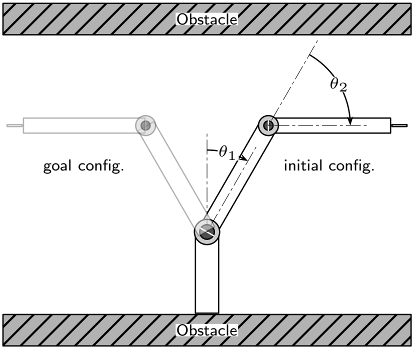


A Two-DoF robot arm

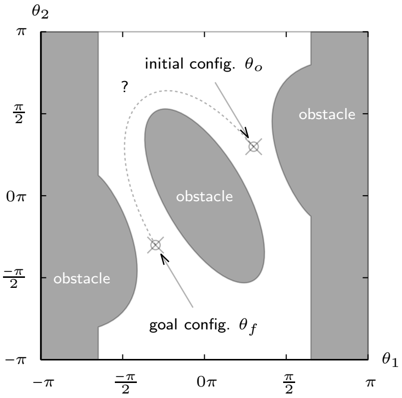


Configuration space $\mathcal{C}$

A Simple Robot Path 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 $\theta(1) = \theta_f$.

(see next page)

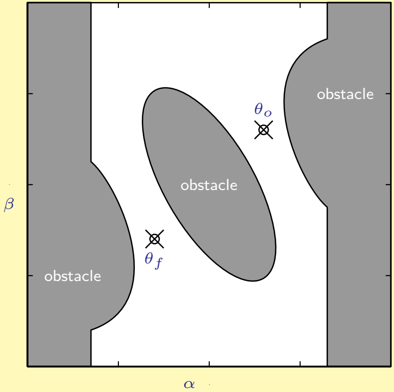
Path planning methods ([1, 2])

Probabilistic roadmap, Cell decomposition, numerical potential field,etc.

Path planning methods ([1, 2])

Probabilistic roadmap, Cell decomposition, numerical potential field,etc.

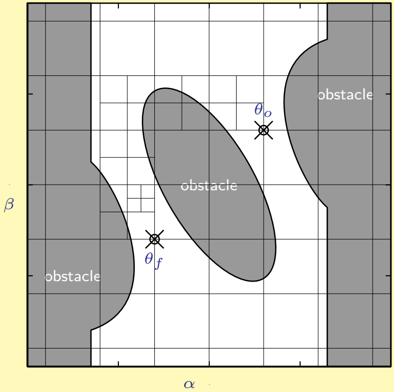
◇ Example: Cell decomposition



Path planning methods ([1, 2])

Probabilistic roadmap, Cell decomposition, numerical potential field,etc.

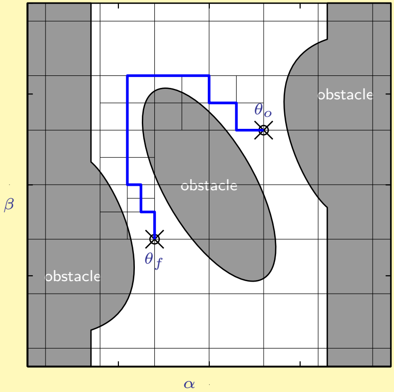
◇ Example: Cell decomposition

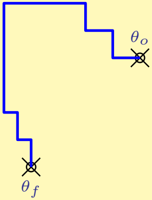


Path planning methods ([1, 2])

Probabilistic roadmap, Cell decomposition, numerical potential field,etc.

⋄ Example: Cell decomposition



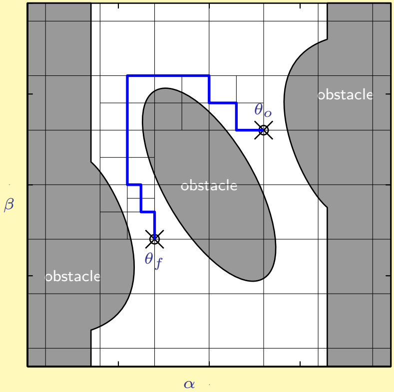


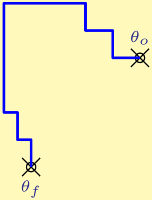
A collision free path

Path planning methods ([1, 2])

Probabilistic roadmap, Cell decomposition, numerical potential field,etc.

◇ Example: Cell decomposition

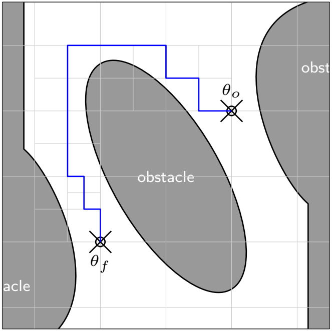




A collision free path

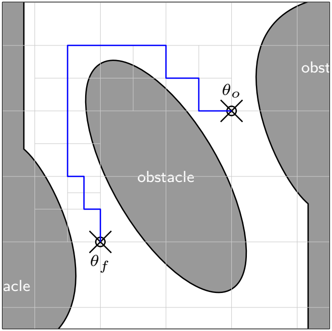
◇ Note: The generated path may not be suitable for robot control. E.g. not smooth.

Trajectory generation

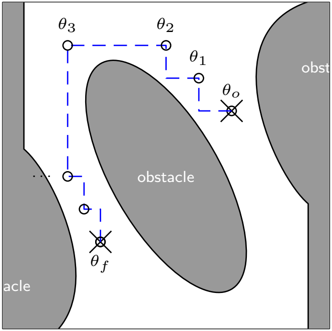


A collision free path

Trajectory generation

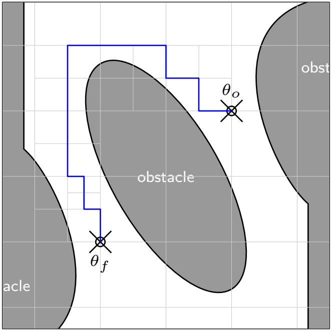


A collision free path

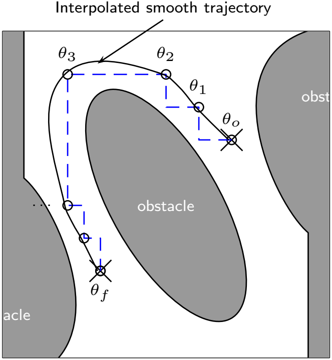


Generated via-points

Trajectory generation

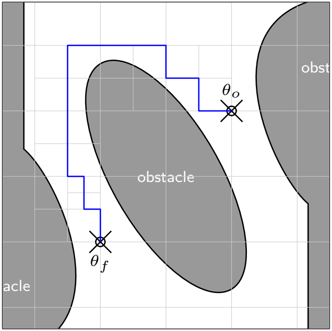


A collision free path

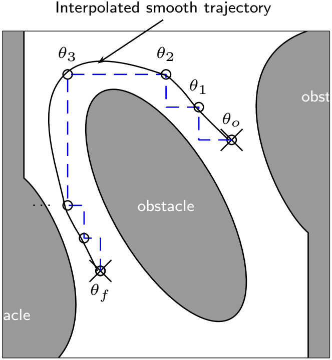


Generated via-points

Trajectory generation



A collision free path



Generated via-points

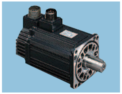
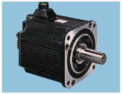
Definition: Trajectory generation

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 $\theta(t_k) = \theta_k$, $k = 1, \ldots, n-1$.

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

Main constraints on trajectory 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:

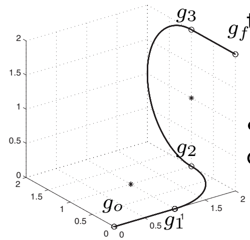
via points directly recorded as joint angles, no inverse kinematics required.

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):



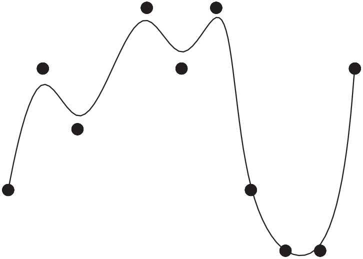
- use inverse kinematics and inverse Jacobian to obtain joint angles and velocity information:

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

- constraints from both joint and workspace need be considered.

From via-points to trajectory

interpolation

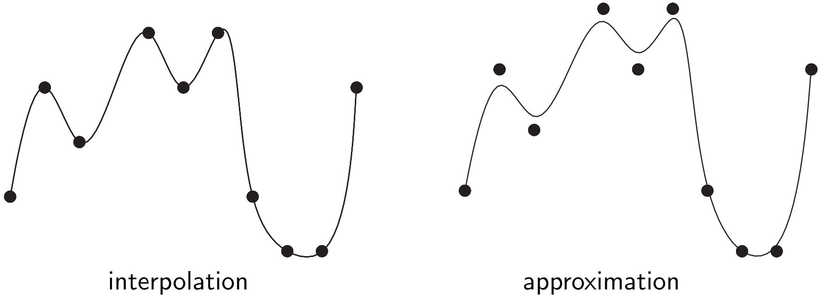


approximation

Definition: Interpolation

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

From via-points to trajectory

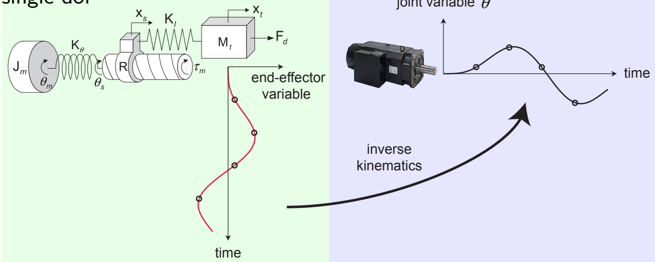


Definition: Interpolation
Constructing new data points within the range of a discrete set of known data points (exact fitting).

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

Types of trajectory

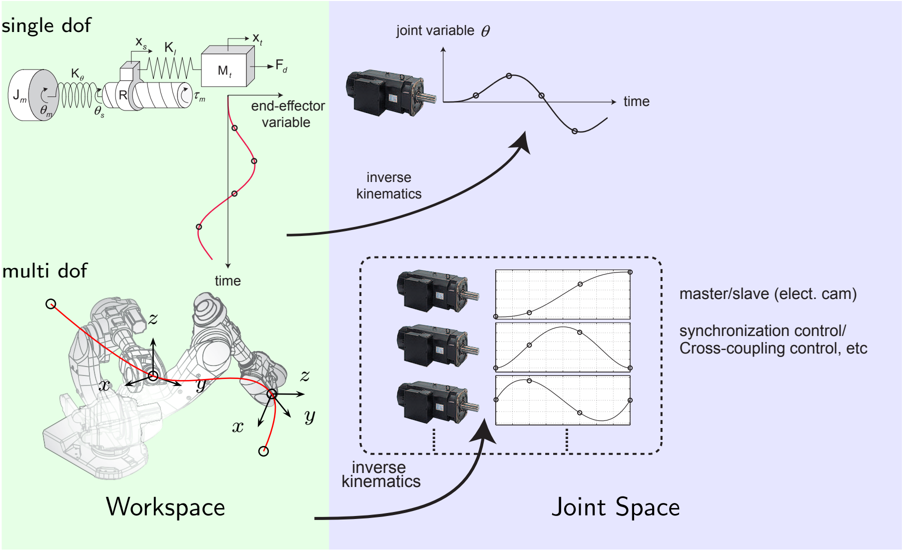
single dof



Workspace

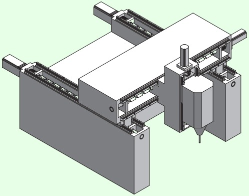
Joint Space

Types of trajectory



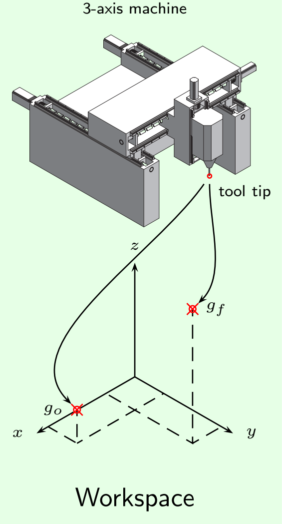
A simple example-Linear interpolation with no via-points

3-axis machine

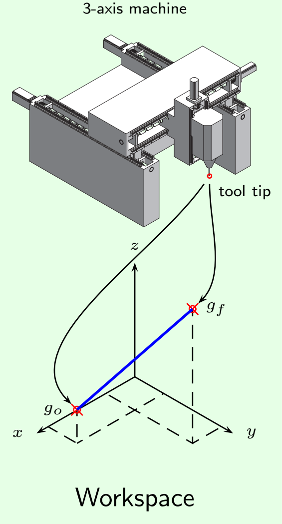


Workspace

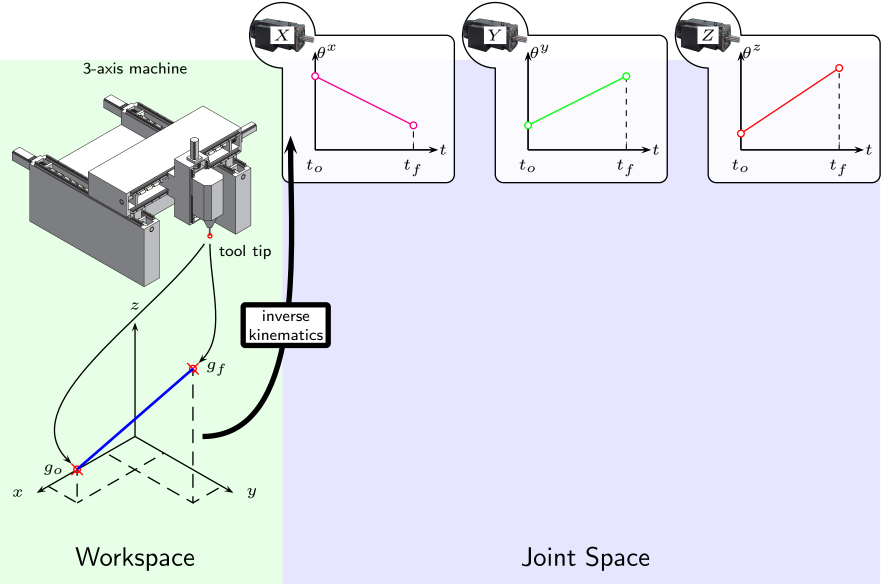
A simple example-Linear interpolation with no via-points



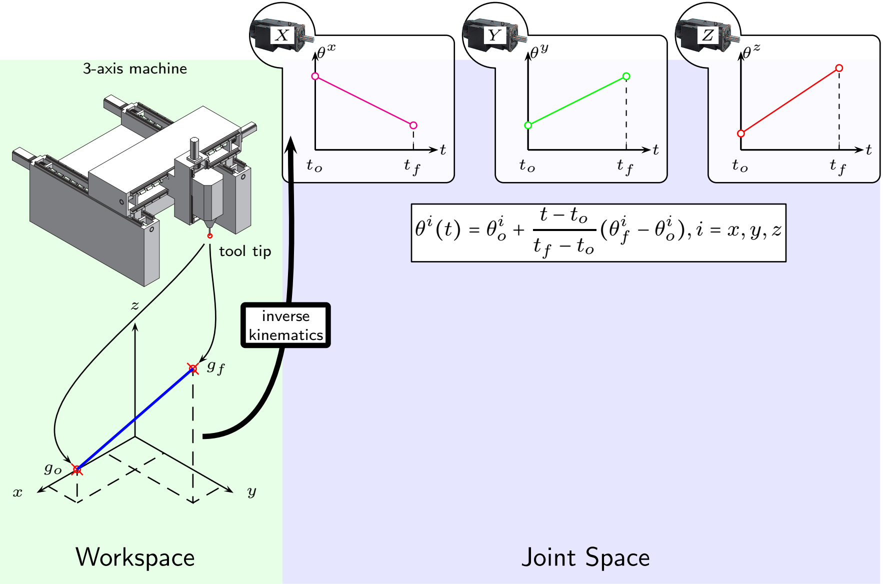
A simple example-Linear interpolation with no via-points



A simple example-Linear interpolation with no via-points

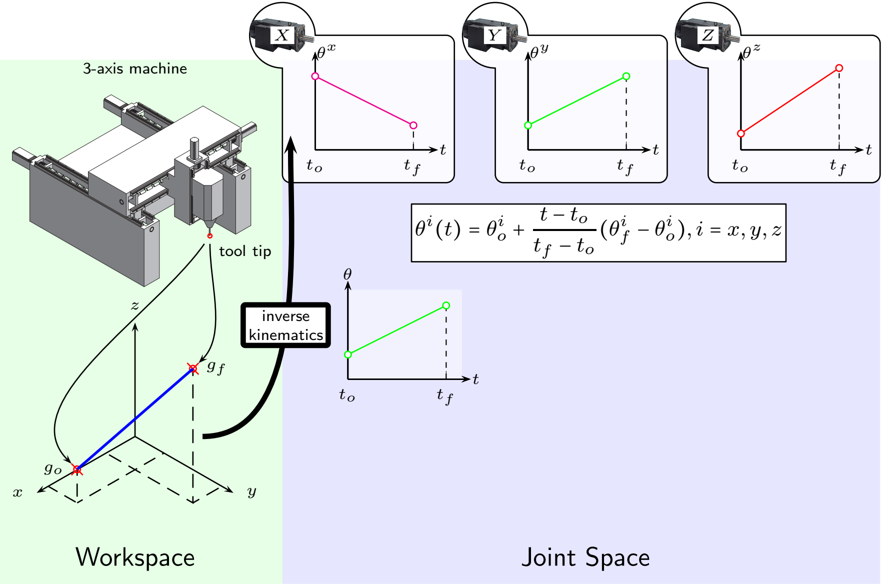


A simple example-Linear interpolation with no via-points



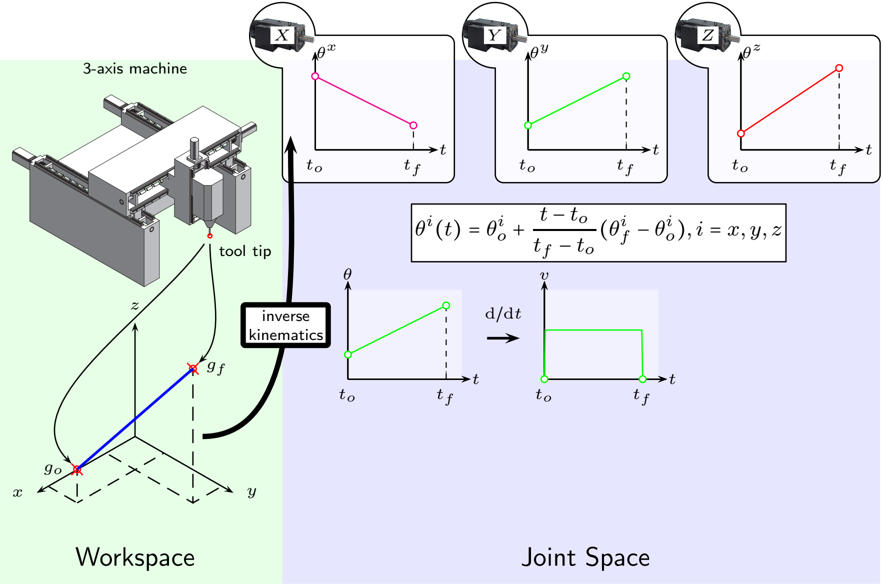
$$\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z$$

A simple example-Linear interpolation with no via-points



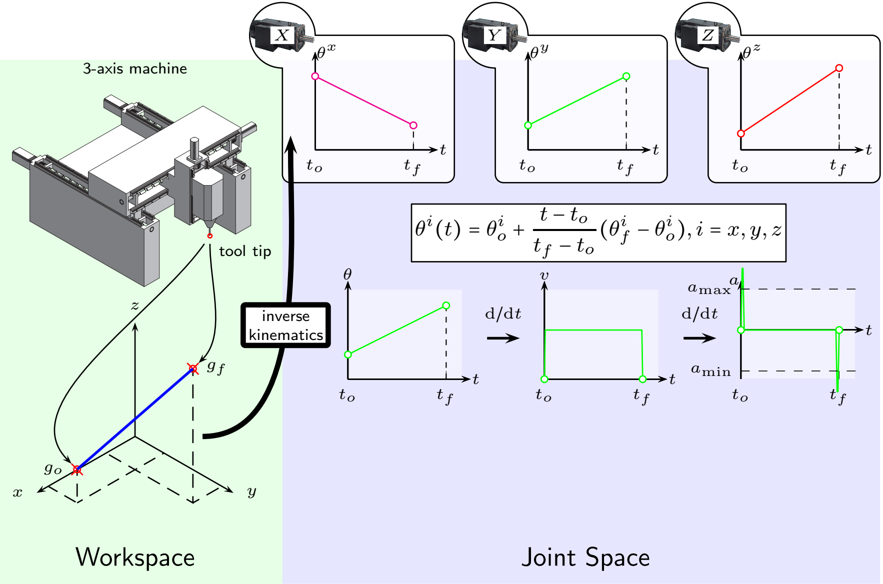
$$\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z$$

A simple example-Linear interpolation with no via-points



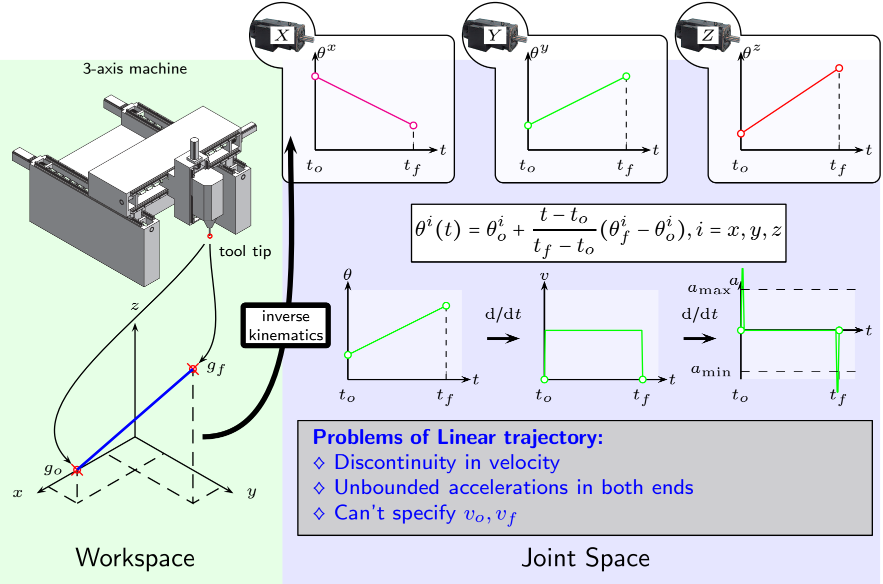
$$\boxed{\theta^i(t) = \theta^i_o + \frac{t - t_o}{t_f - t_o}(\theta^i_f - \theta^i_o), i = x, y, z}$$

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



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

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:

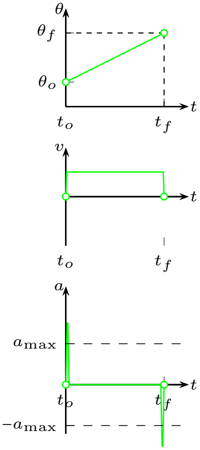
$$\theta(t) = \theta_o + \frac{t - t_o}{t_f - t_o}(\theta_f - \theta_o) = a_0 + a_1 t, a_0 = \frac{t_f \theta_o - t_o \theta_f}{t_f - t_o}, a_1 = \frac{\theta_f - \theta_o}{t_f - t_o}$$

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

Higher order trajectories:

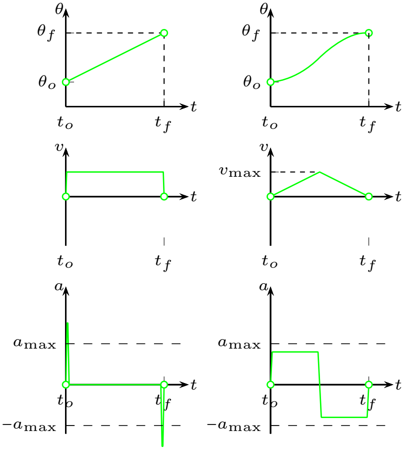
order of θ	order of v	order of a	allowable design variables
2 (parabolic)	1	0	$\theta_o, \theta_f, v_{\max}$
3 (cubic)	2	1	$\theta_o, \theta_f, v_o, v_f$
5 (quintic)	4	3	$\theta_o, \theta_f, v_o, v_f, a_o, a_f$

Higher order trajectories: time profile



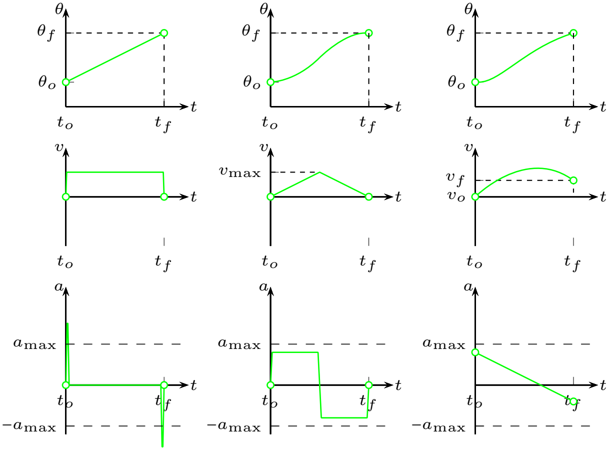
linear trajectory

Higher order trajectories: time profile



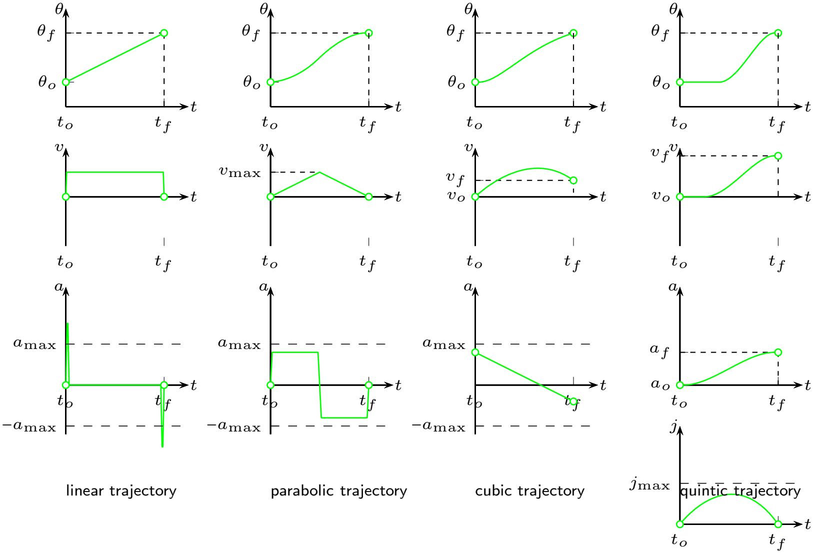
linear trajectory parabolic trajectory

Higher order trajectories: time profile



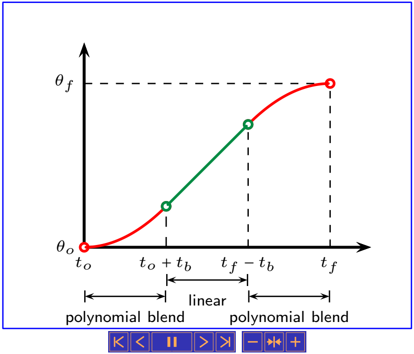
linear trajectory parabolic trajectory cubic trajectory

Higher order trajectories: time profile

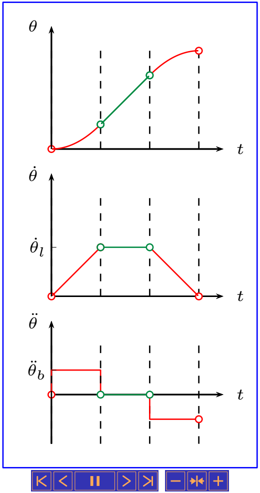


Better approaches

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



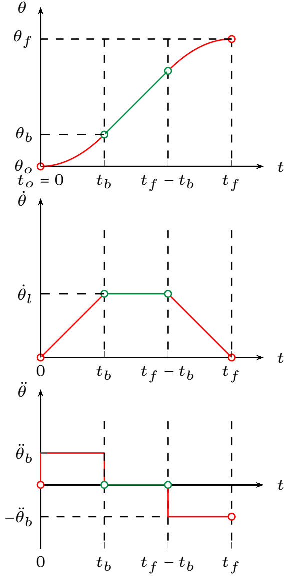
e.g. parabolic blend



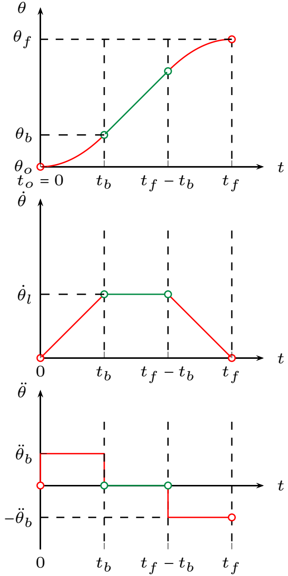
◇ A list of composite trajectories:

- □ 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

◇ Control Parameters:

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

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

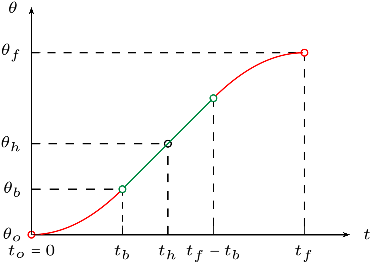
$t_b(0 < t_b \leq \frac{t_d}{2})$: blend time

$t_d - 2t_b$: linear time

◇ LFPB Trajectory:

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

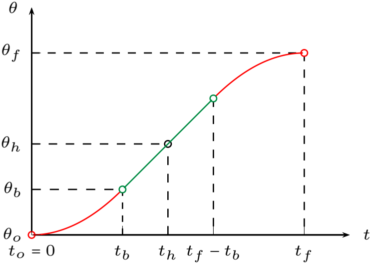
Derivation of the LFPB



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, (t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)) \quad (5.1.1)$$

Derivation of the LFPB



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, (t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)) \tag{5.1.1}$$

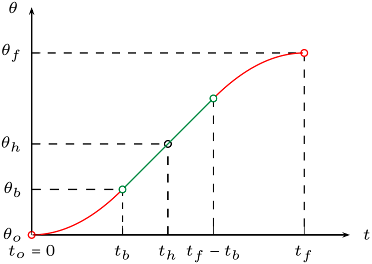
◇ Blend region:

$$\theta_b \triangleq \theta(t_b) = \theta_o + \frac{1}{2}\ddot{\theta}_b t_b^2 \tag{5.1.2}$$

$$(5.1.1) + (5.1.2) : \Rightarrow \ddot{\theta}_b t_b^2 - \ddot{\theta}_b t_d t_b + (\theta_f - \theta_o) = 0$$

$$\Rightarrow t_b = \frac{t_d}{2} - \frac{\sqrt{\ddot{\theta}_b^2 t_d^2 - 4\ddot{\theta}_b(\theta_f - \theta_o)}}{2\ddot{\theta}_b} \tag{5.1.3}$$

Derivation of the LFPB



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, (t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)) \tag{5.1.1}$$

◇ Blend region:

$$\theta_b \triangleq \theta(t_b) = \theta_o + \frac{1}{2}\ddot{\theta}_b t_b^2 \tag{5.1.2}$$

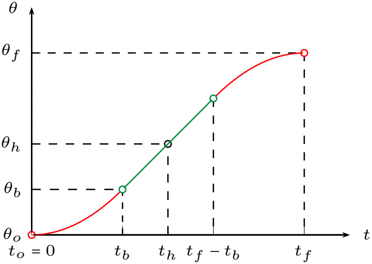
$$(5.1.1) + (5.1.2) : \Rightarrow \ddot{\theta}_b t_b^2 - \ddot{\theta}_b t_d t_b + (\theta_f - \theta_o) = 0$$

$$\Rightarrow t_b = \frac{t_d}{2} - \frac{\sqrt{\ddot{\theta}_b^2 t_d^2 - 4\ddot{\theta}_b(\theta_f - \theta_o)}}{2\ddot{\theta}_b} \tag{5.1.3}$$

◇ Constraints on $\ddot{\theta}_b$:

$$\ddot{\theta}_b \geq \frac{4(\theta_f - \theta_o)}{t_d^2} \tag{5.1.4}$$

Derivation of the LFPB



◇ Velocity match condition:

$$\ddot{\theta}_b t_b = \frac{\theta_h - \theta_b}{t_h - t_b}, (t_h \triangleq \frac{t_d}{2}, \theta_h \triangleq \theta(t_h)) \tag{5.1.1}$$

◇ Blend region:

$$\theta_b \triangleq \theta(t_b) = \theta_o + \frac{1}{2}\ddot{\theta}_b t_b^2 \tag{5.1.2}$$

$$(5.1.1) + (5.1.2) : \Rightarrow \ddot{\theta}_b t_b^2 - \ddot{\theta}_b t_d t_b + (\theta_f - \theta_o) = 0$$

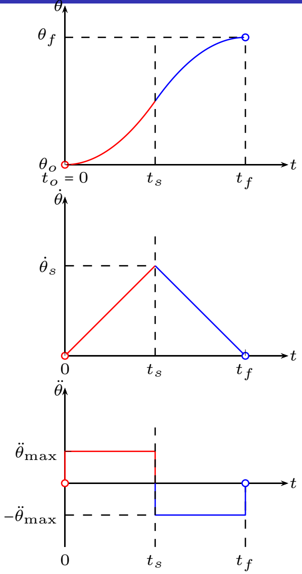
$$\Rightarrow t_b = \frac{t_d}{2} - \frac{\sqrt{\ddot{\theta}_b^2 t_d^2 - 4\ddot{\theta}_b(\theta_f - \theta_o)}}{2\ddot{\theta}_b} \tag{5.1.3}$$

◇ Constraints on $\ddot{\theta}_b$:

$$\ddot{\theta}_b \geq \frac{4(\theta_f - \theta_o)}{t_d^2} \tag{5.1.4}$$

◇ 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 (5.1.4), linear portion of LFPB shrinks to zero.

Minimum Time Trajectory (Bang Bang)



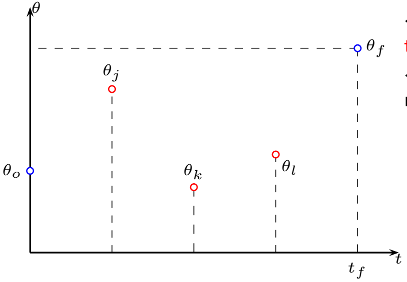
- ♢ Given: θ_o, θ_f and $\ddot{\theta}_{\max}$
- ♢ Minimize: t_d
- ♢ Solution: Bang-bang trajectory

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

where the switching time t_s is obtained from (5.1.3)

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

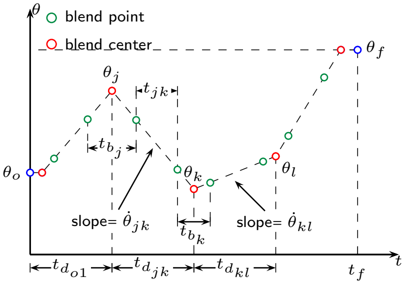
LFPB for a Path with Via Points ([3])



◇ 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])

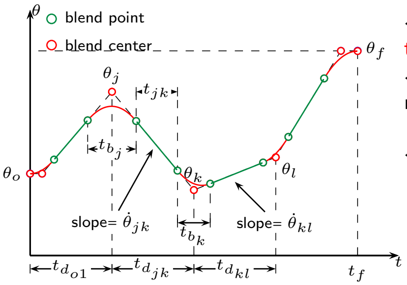


◇ Given: $t_o, \theta_o, t_f, \theta_f, \ddot{\theta}_b$, via points $\{\theta_i\}_1^m$ at time $\{t_i\}_1^m$ (time duration $t_{d_{jk}} \triangleq t_k - t_j$)

◇ 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])

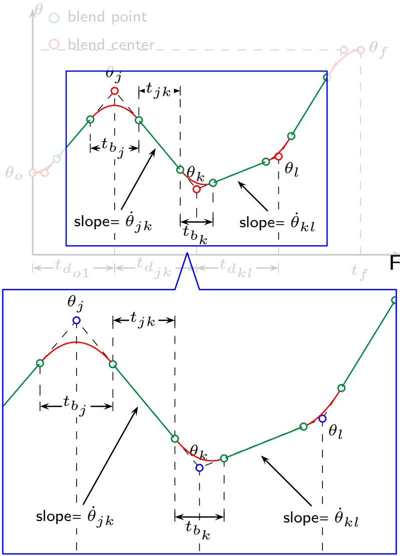


◇ Given: $t_o, \theta_o, t_f, \theta_f, \ddot{\theta}_b$, via points $\{\theta_i\}_1^m$ at time $\{t_i\}_1^m$ (time duration $t_{d_{jk}} \triangleq t_k - t_j$)

◇ 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 $t_{d_{jk}} \triangleq t_k - t_j$)

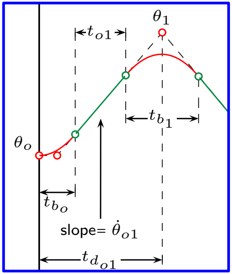
◊ Find: $\theta(t)$ interpolating θ_o, θ_f and approximating $\{\theta_i\}_1^m$.

◊ Solution: LFPB with via points

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

$$\begin{aligned}
\dot{\theta}_{jk} &= \frac{\theta_k - \theta_j}{t_{d_{jk}}} \ \text{(linear vel.)} \\
\ddot{\theta}_k &= \mathrm{Sgn}(\dot{\theta}_{kl} - \dot{\theta}_{jk}) \left|\ddot{\theta}_b\right| \ \text{(Blend acc.)} \\
t_{b_k} &= \frac{\dot{\theta}_{kl} - \dot{\theta}_{jk}}{\ddot{\theta}_k} \ \text{(Blend dur.)} \\
t_{jk} &= t_{d_{jk}} - \frac{1}{2} t_{b_j} - \frac{1}{2} t_{b_k} \ \text{(Linear dur.)}
\end{aligned} \quad (5.1.5)$$

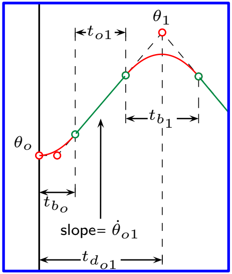
LFPB for a Path with Via Points ([3])



◇ **First Segment:**

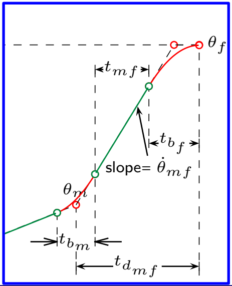
$$
\begin{aligned}
\ddot{\theta}_o &= \mathrm{Sgn}(\theta_1 - \theta_o)\left|\ddot{\theta}_b\right| \ \text{(Blend acc.)}\\
t_{b_o} &= t_{d_{o1}} - \sqrt{t_{d_{o1}}^2 - \frac{2(\theta_1 - \theta_o)}{\ddot{\theta}_o}} \ \text{(Blend dur.)}\\
\dot{\theta}_{o1} &= \frac{\theta_1 - \theta_o}{t_{d_{o1}} - \frac{1}{2}t_{b_o}} \ \text{(linear vel.)}\\
t_{o1} &= t_{d_{o1}} - t_{b_o} - \frac{1}{2}t_{b_1} \ \text{(linear dur.)}
\end{aligned}
\tag{5.1.6}
$$

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



◇ **First Segment:**

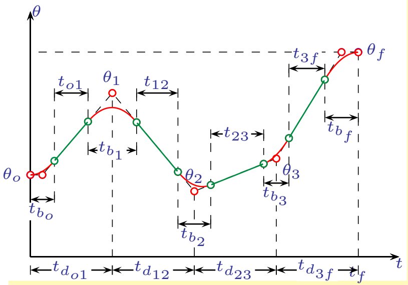
$$\begin{aligned}
\ddot{\theta}_o &= \mathrm{Sgn}(\theta_1 - \theta_o)\left|\ddot{\theta}_b\right| \text{ (Blend acc.)}\\
t_{b_o} &= t_{d_{o1}} - \sqrt{t_{d_{o1}}^2 - \frac{2(\theta_1 - \theta_o)}{\ddot{\theta}_o}} \text{ (Blend dur.)}\\
\dot{\theta}_{o1} &= \frac{\theta_1 - \theta_o}{t_{d_{o1}} - \frac{1}{2}t_{b_o}} \text{ (linear vel.)}\\
t_{o1} &= t_{d_{o1}} - t_{b_o} - \frac{1}{2}t_{b_1} \text{ (linear dur.)}
\end{aligned} \tag{5.1.6}$$



◇ **Last Segment:**

$$\begin{aligned}
\ddot{\theta}_f &= \mathrm{Sgn}(\theta_m - \theta_f)\left|\ddot{\theta}_b\right| \text{ (Blend acc.)}\\
t_{b_f} &= t_{d_{mf}} - \sqrt{t_{d_{mf}}^2 + \frac{2(\theta_f - \theta_m)}{\ddot{\theta}_f}} \text{ (Blend dur.)}\\
\dot{\theta}_{mf} &= \frac{\theta_f - \theta_m}{t_{d_{mf}} - \frac{1}{2}t_{b_f}} \text{ (linear vel.)}\\
t_{mf} &= t_{d_{mf}} - t_{b_f} - \frac{1}{2}t_{b_m} \text{ (linear dur.)}
\end{aligned} \tag{5.1.7}$$

Example: LFPB with 3 via points



Given:

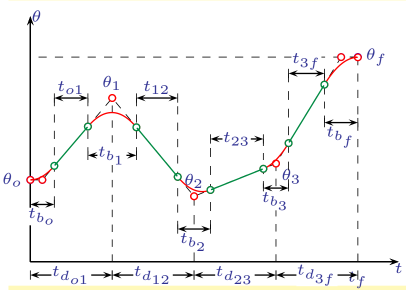
$$\theta_o = 1, \theta_f = 2.5, \ddot{\theta}_b = 4, \theta_1 = 2, \theta_2 = .8, \theta_3 = 1.2$$

Apply (5.1.6):

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

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

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 (5.1.6):

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

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

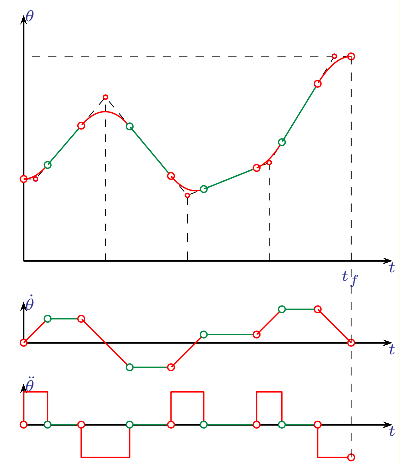
Apply (5.1.5):

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

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

(Continues next slide)

Example: LFPB with 3 via points



Apply (5.1.7):

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

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

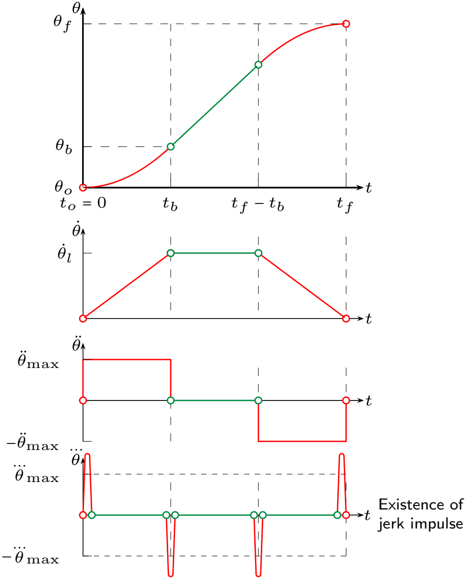
Apply (5.1.5):

$$\ddot\theta_3 = 4, t_{b_3} = \frac{1.63-0.4}{4} = 0.31,$$

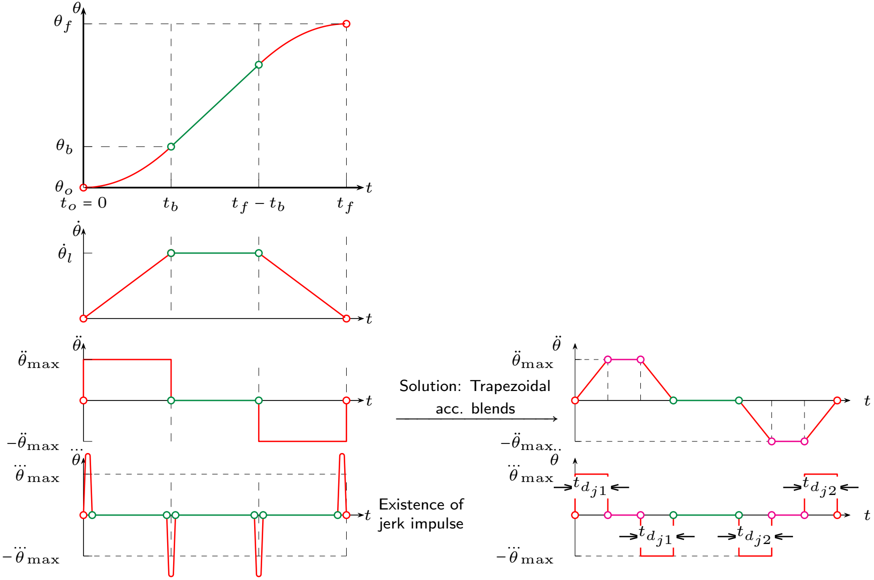
$$t_{3f} = 1 - \frac{1}{2}0.31 - 0.41 = 0.44,$$

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

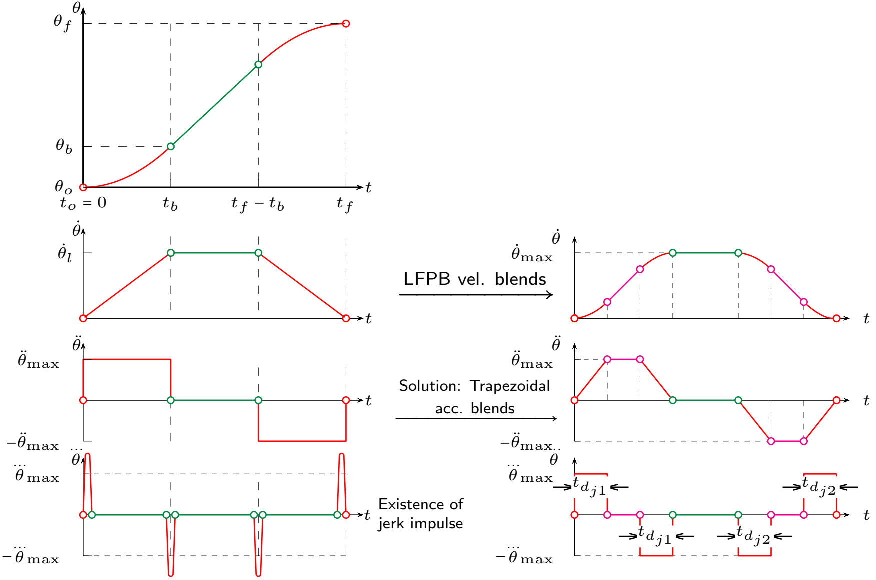
Disadvantage of LFPB



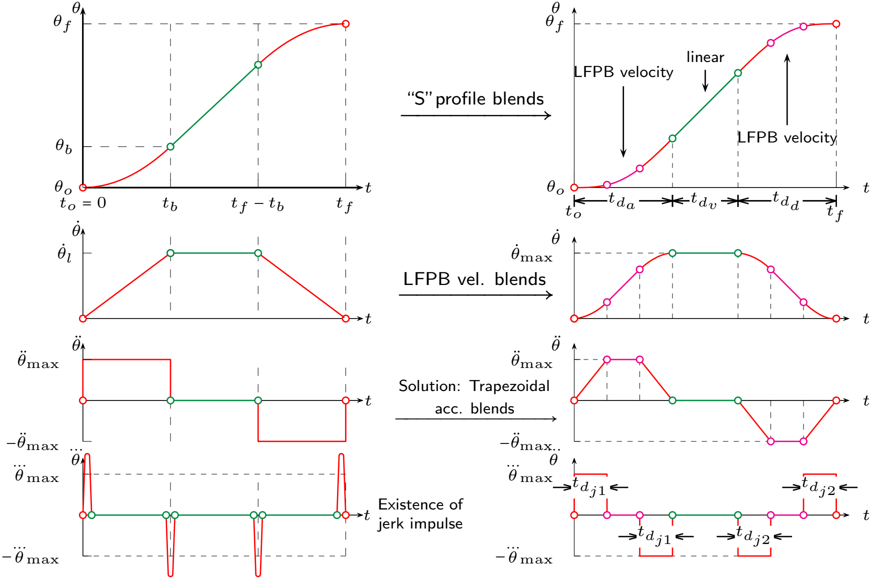
Disadvantage of LFPB



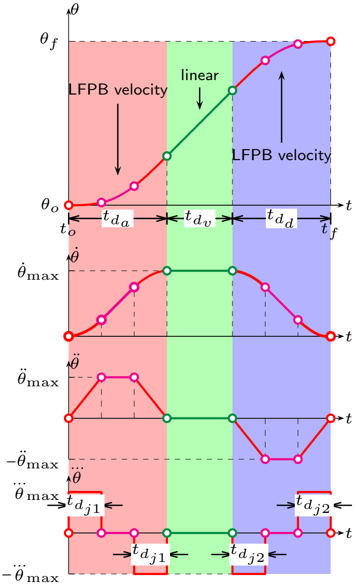
Disadvantage of LFPB



Disadvantage of LFPB



Linear function with Double S trajectory

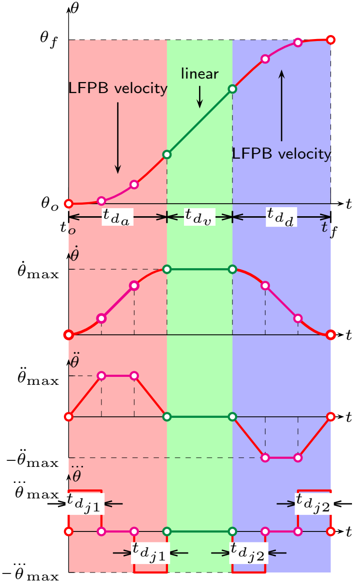


♢ **Double "S"trajectory:**

Linear trajectory with LFPB velocity blends

♢ **Advantage over LFPB:** Bounded jerk

Linear function with Double S trajectory



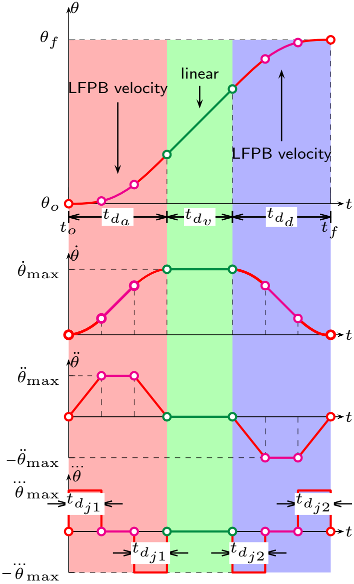
♢ **Double “S” trajectory:**
Linear trajectory with LFPB velocity blends

♢ **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 trajectory



◇ **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)

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

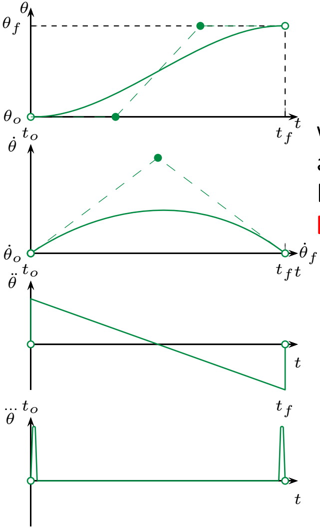
Constant velocity phase:

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

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**.

$$\begin{cases}\theta(t_o) = a_0 = \theta_o\\ \dot\theta(t_o) = a_1 = \dot\theta_o\\ \theta(t_f) = \sum_{i=0}^{3} a_i t_d^i = \theta_f\\ \dot\theta(t_f) = \sum_{i=0}^{2}(i+1)a_{i+1}t_d^i = \dot\theta_f\end{cases} \Rightarrow \begin{cases}a_0 = \theta_o\\ a_1 = \dot\theta_o\\ a_2 = \dfrac{3h - (2\dot\theta_o + \dot\theta_f)t_d}{t_d^2}\\ a_3 = \dfrac{-2h + (\dot\theta_o + \dot\theta_f)t_d}{t_d^3}\end{cases}$$

(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:

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

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 $\{a_{0k}, a_{1k}, a_{2k}, a_{3k}\}_o^m$.

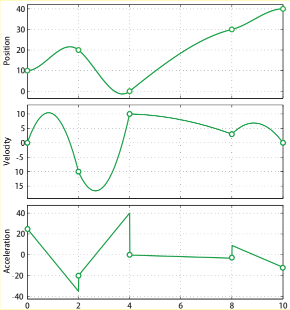
1. If via-point velocities $\{\dot{\theta}_k\}_1^m$ are directly assigned by user, solve the $m+1$ BVPs:

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

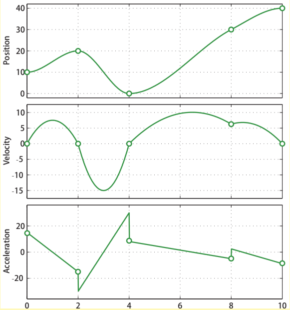
2. If only $\dot{\theta}_o, \dot{\theta}_f$ are given:
   1. compute $\{\dot{\theta}_k\}_1^m$ using a heuristic method; or
   2. design $\{\dot{\theta}_k\}_1^m$ so as to achieve acceleration continuity

(Continuous next slide)

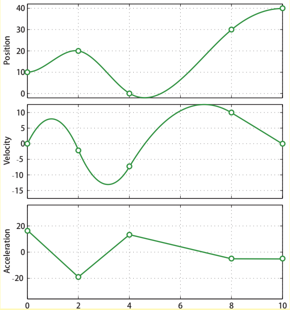
Example: Cubic interpolation with 3 via points



Approach 1



Approach 2

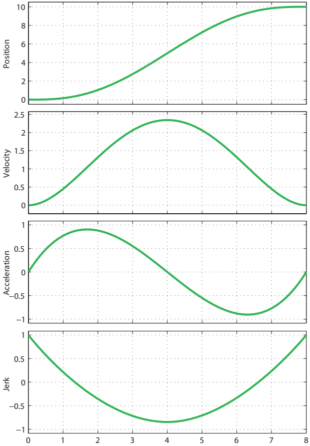


Approach 3

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

$$\boxed{\theta(t) = \sum_{i=0}^{5} a_i (t - t_o)^i, t \in [t_o, t_f]}$$



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

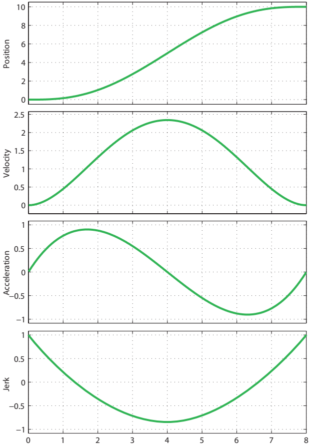
Properties:

◇ Smooth and bounded jerk

◇ Acc. continuity in composite curves.

Quintic polynomial trajectory

$$\boxed{\theta(t) = \sum_{i=0}^{5} a_i (t - t_o)^i, t \in [t_o, t_f]}$$



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

Properties:

◇ Smooth and bounded jerk

◇ Acc. continuity in composite curves.

Boundary conditions:

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

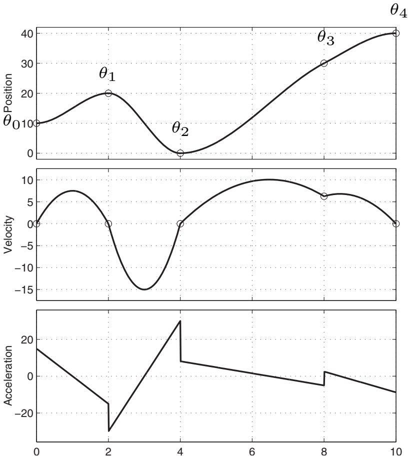
(Continues next slide)

Quintic polynomial trajectory

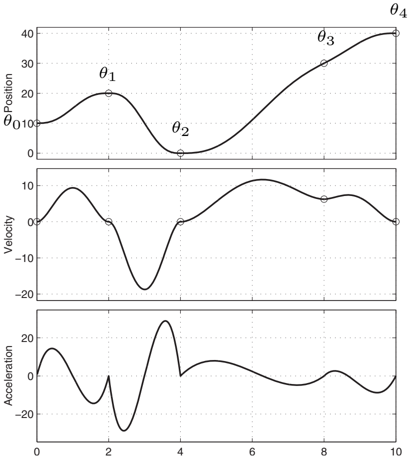
Define $t_d \triangleq t_f - t_o, h \triangleq \theta_f - \theta_o$, then:

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

Comparison of Cubic and Quintic Composites



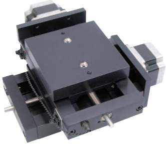
Composition of cubic polynomials: acceleration discontinuity.



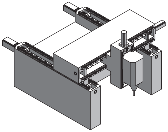
Composition of quintic polynomials: continuity in acceleration.

Trajectory Generation in Task Space

1. **Euclidean space:**



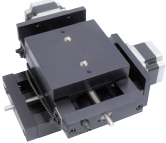
xy table ($\mathbb{R}^2$)



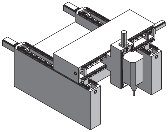
3-axis machine ($\mathbb{R}^3$)

Trajectory Generation in Task Space

1. **Euclidean space:**

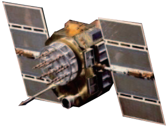


xy table ($\mathbb{R}^2$)



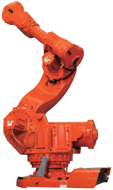
3-axis machine ($\mathbb{R}^3$)

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



Satellite ($SO(3)$)

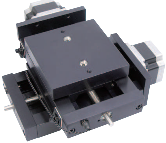
pick-and-place (X)



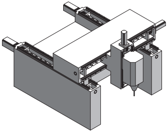
6 dof robot ($SE(3)$)

Trajectory Generation in Task Space

1. **Euclidean space:**

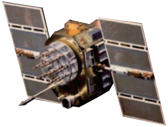


xy table ($\mathbb{R}^2$)



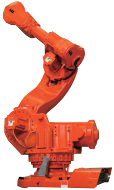
3-axis machine ($\mathbb{R}^3$)

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



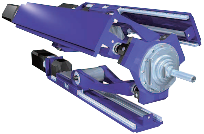
Satellite ($SO(3)$)

pick-and-place (X)

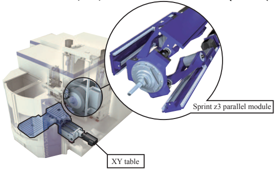


6 dof robot ($SE(3)$)

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

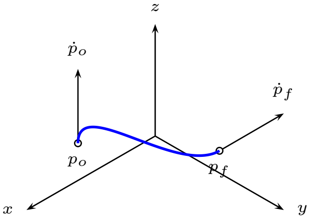
$$\text{e.g. } p(t) = \begin{bmatrix} a_{01} \\ \vdots \\ a_{0n} \end{bmatrix} + \begin{bmatrix} a_{11} \\ \vdots \\ a_{1n} \end{bmatrix} (t - t_o) + \cdots + \begin{bmatrix} a_{m1} \\ \vdots \\ a_{mn} \end{bmatrix} (t - t_o)^m, t \in [t_o, t_f]$$

◇ **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].

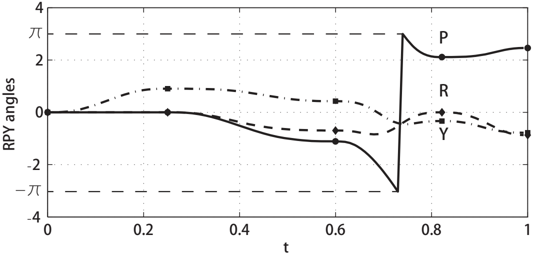
Trajectory Generation in $SO(3)$

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

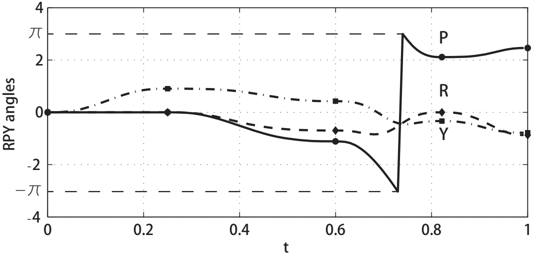
Trajectory Generation in $SO(3)$

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:

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

- Canonical coordinate:

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

A cubic trajectory 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 $\int_0^1 \dot{\omega}^T \dot{\omega} \mathrm{d}t$.

A cubic trajectory 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]:**

$$\omega^{(3)} + \omega \times \ddot{\omega} = 0 \tag{5.1.8}$$

which is hard to solve.

A cubic trajectory 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]:**

$$\omega^{(3)} + \omega \times \ddot{\omega} = 0 \tag{5.1.8}$$

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 $\dot{\omega}$:

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

$$(5.1.8): \omega^{(3)} + \omega \times \ddot{\omega} \approx \omega^{(3)} = r^{(4)} = 0$$

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 $\dot{\omega}$:

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

$$(5.1.8) : \omega^{(3)} + \omega \times \ddot{\omega} \approx \omega^{(3)} = r^{(4)} = 0$$

which shows that r is a cubic curve:

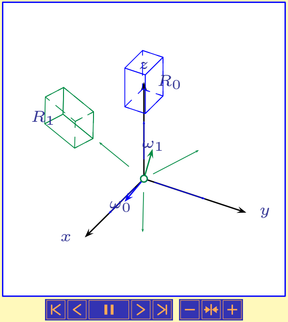
$$r(t) = at^3 + bt^2 + ct, t \in [0, 1]$$

◇ 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)

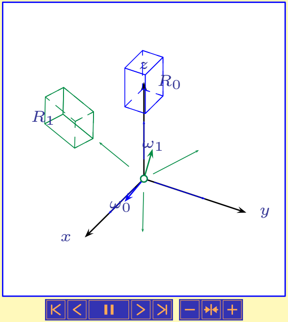
Example: A cubic traj. on $SO(3)$ (ctned)



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

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

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



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

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

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

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

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

Trajectory 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.

Trajectory 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]):

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

Trajectory 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]):

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

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

Manipulator control problems

Recall the manipulator dynamics equation:

$$\underbrace{M(\theta)\ddot{\theta}}_{\text{Inertia force}} + \underbrace{C(\theta,\dot{\theta})\dot{\theta}}_{\substack{\text{Coriolis \&} \\ \text{Centrifugal force}}} + \underbrace{N(\theta)}_{\text{gravity}} = \underbrace{\tau}_{\substack{\text{Joint} \\ \text{torque}}} - \left(\underbrace{A^T(\theta)\cdot F}_{\text{Interaction force}} \right)$$

Manipulator control problems

Recall the manipulator dynamics equation:

$$\underbrace{M(\theta)\ddot{\theta}}_{\text{Inertia force}} + \underbrace{C(\theta,\dot{\theta})\dot{\theta}}_{\substack{\text{Coriolis \&}\\ \text{Centrifugal force}}} + \underbrace{N(\theta)}_{\text{gravity}} = \underbrace{\tau}_{\substack{\text{Joint}\\ \text{torque}}} - \left(\underbrace{A^T(\theta)\cdot F}_{\text{Interaction force}}\right)$$

Problem 1: Position control ($F = 0$)
Given the dynamic equation of a manipulator:

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + N(\theta) = \tau$$

and a desired trajectory $\theta_d(t)$ in joint space (or $x_d(t)$ in task space), design the joint torque input τ so that the trajectory tracking error $e(t)$ (or $e_x(t)$):

$$e(t) \triangleq \theta_d - \theta(t) \to 0 \text{ (or } e_x(t) \triangleq x_d - x(t) \to 0\text{) asymptotically.}$$

Note: If $\theta_d(t) = \theta_d = \text{const.}$ (or $x_d(t) = x_d = \text{const.}$)$\Rightarrow$ *regulation problem*

Manipulator control problems

Problem 2: Force Control

Given a desired end-effector force $F_d(t)$, design the joint torque input τ so that:

$$e(t) \triangleq F(t) - F_d(t) \to 0$$

asymptotically.

Manipulator control problems

Problem 2: Force Control

Given a desired end-effector force $F_d(t)$, design the joint torque input τ so that:

$$e(t) \triangleq F(t) - F_d(t) \to 0$$

asymptotically.

Problem 3: Hybrid Position/Force Control

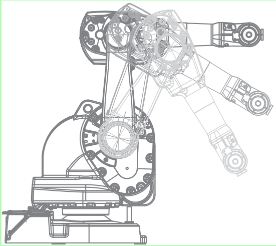
Given a desired constrained joint motion $\theta_d(t)$ satisfying $A(\theta_d)\dot{\theta}_d = 0$, and a desired constraint force $f_d = A^T F_d$, design the joint torque input τ so that:

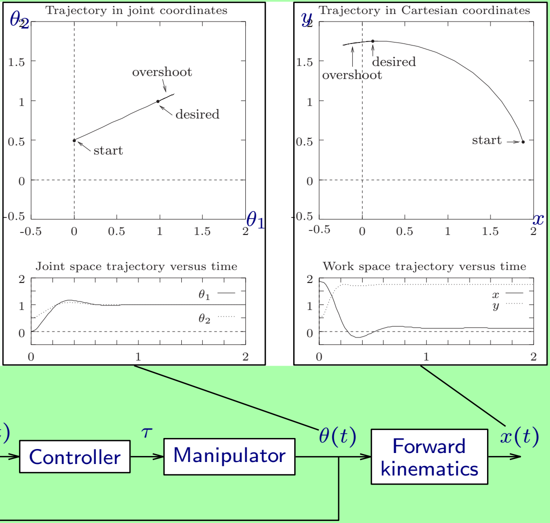
$$\theta(t) - \theta_d(t) \to 0$$
$$F(t) - F_d(t) \to 0$$

asymptotically.

Position Control in Joint Space

Problem 1.A: Position Control in Joint Space





Review: Ordinary Differential Equation

Given a first order ODE $\dot{x} = f(x,t)$, $x_0 = x(t_0)$ is called an *equilibrium point* if $\dot{x}(t_0) = f(x_0, t_0) \equiv 0$.

Review: Ordinary Differential Equation

Given a first order ODE $\dot{x} = f(x,t)$, $x_0 = x(t_0)$ is called an *equilibrium point* if $\dot{x}(t_0) = f(x_0,t_0) \equiv 0$.

Consider the second order ODE for $x(t) \in \mathbb{R}^n$, with initial condition $x(0) = x_0, \dot{x}(0) = v_0$:

$$\ddot{x}(t) + K_v\dot{x}(t) + K_p x(t) = 0$$

It can be converted into a first order ODE by defining $z_1 = x, z_2 = \dot{x}$,

$$\dot{z} = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & I \\ -K_p & -K_v \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \triangleq Az$$

$\sigma(A) \in \mathbb{C}^-$ if $K_v, K_p \in \mathbb{R}^{n\times n}$ are positive definite constant matrices:

Proposition:

The solution $x(t)$ converges asymptotically and exponentially to 0 if $K_v, K_p \in \mathbb{R}^{n\times n}$ are positive definite constant matrices.

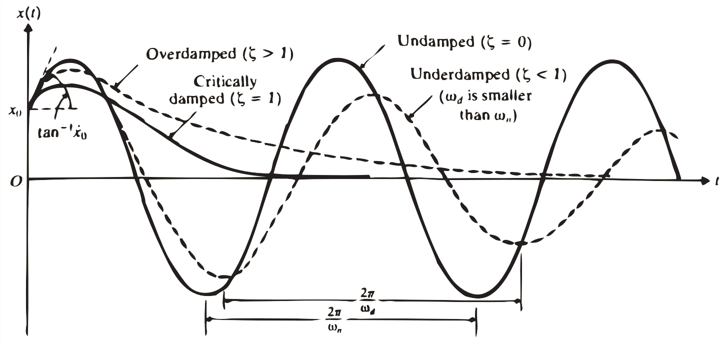
Example: Single DOF ODE

Given

$$\ddot{x}(t) + K_v\dot{x}(t) + K_px(t) = 0, x(0) = x_0, \dot{x}(0) = \dot{x}_0$$

Define:

$$\omega_n^2 = K_p/1, 2\zeta\omega_n = K_v, \omega_d = \sqrt{1-\zeta^2}\omega_n$$



Review: Lyapunov method and Lasalle's Principle

Proposition 1: Lyapunov stability

Consider the following first order ODE with equilibrium point $x_0 = 0$:

$$\dot{x} = f(x), f(0) = 0$$

If there exists a **Lyapunov function candidate** $V : U \subset \mathbb{R}^n \mapsto \mathbb{R}_+$ which is positive definite:

$$V(x) \geq 0, \forall x \in U, V(x) = 0 \text{ iff } x = 0$$

and $\dot{V} = \frac{\partial f}{\partial x} \cdot f$ is negative definite on U, then any $x(t), x(0) \in U$ converges to 0 asymptotically.

Review: Lyapunov method and Lasalle's Principle

Proposition 1: Lyapunov stability

Consider the following first order ODE with equilibrium point $x_0 = 0$:

$$\dot{x} = f(x), f(0) = 0$$

If there exists a **Lyapunov function candidate** $V : U \subset \mathbb{R}^n \mapsto \mathbb{R}_+$ which is positive definite:

$$V(x) \geq 0, \forall x \in U, V(x) = 0 \text{ iff } x = 0$$

and $\dot{V} = \frac{\partial f}{\partial x} \cdot f$ is negative definite on U, then any $x(t), x(0) \in U$ converges to 0 asymptotically.

Proposition 2: Lasalle's Principle

Given $\dot{x} = f(x)$. Let $V : \mathbb{R}^n \mapsto \mathbb{R}$ be a locally positive definite function such that on the compact set $\Omega_c \triangleq \{x \in \mathbb{R}^n | V(x) \leq c\}$, we have $\dot{V}(x) \leq 0$. Define

$$S = \{x \in \Omega_c | \dot{V}(x) = 0\}$$

As $t \mapsto \infty$, the trajectory tends to the largest invariant set inside S. In particular, if S contains no invariant sets other than $x = 0$, then $x(t), x(0) \in U$ converges to 0 asymptotically.

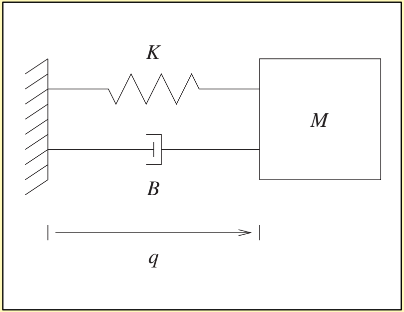
Example: Linear harmonic oscillator

Dynamics equation:

$$M\ddot{q} + B\dot{q} + Kq = 0$$

State space form:

$$\Rightarrow \frac{\mathrm{d}}{\mathrm{d}t}\underbrace{\begin{bmatrix} q \\ \dot{q} \end{bmatrix}}_{x} := \underbrace{\begin{bmatrix} \dot{q} \\ -\frac{K}{M}q - \frac{B}{M}\dot{q} \end{bmatrix}}_{f(x)}$$



Note that the jacobian A

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{K}{M} & -\frac{B}{M} \end{bmatrix}, \mathrm{Re}(\lambda(A)) = \frac{-B \pm \sqrt{B^2 - 4KM}}{2M} < 0$$

$\Rightarrow$ The system always exponentially stable (by Lyapunov indirect method)

(see next page)

Example: Linear harmonic oscillator

Choose *Lyapunov function* to be the system energy:

$$V(x) = \frac{1}{2}M\dot{q}^2 + \frac{1}{2}Kq^2, \dot{V} = M\dot{q}\ddot{q} + Kq\dot{q} = -B\dot{q}^2 \leq 0$$

Apply Lasalle's principle:

$$S = \{x \in \Omega_c | \dot{V}(x) = 0\} \Rightarrow \dot{q} = 0 \Rightarrow \ddot{q} = 0 \Rightarrow q = 0 \Rightarrow$$

$$x = \begin{bmatrix} 0 \\ 0 \end{bmatrix}: \text{ only quilibrium point inside } S.$$

Thus $x(t) \to \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ asymptotically.

◇

Example: Nonlinear spring mass system with damper

State space equation:

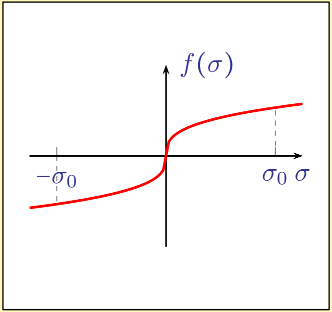
$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -f(x_2) - g(x_1)$$

Passivity of f and g:

$$\sigma f(\sigma) \geq 0, \forall \sigma \in [-\sigma_0, \sigma_0]$$
$$\sigma g(\sigma) \geq 0, \forall \sigma \in [-\sigma_0, \sigma_0]$$

Lyapunov function:

$$V(x) = \frac{x_2^2}{2} + \int_0^{x_1} g(\sigma)\mathrm{d}\sigma, \dot{V}(x) = -x_2 f(x_2)$$



(see next page)

Example: Nonlinear spring mass system with damper

State space equation:

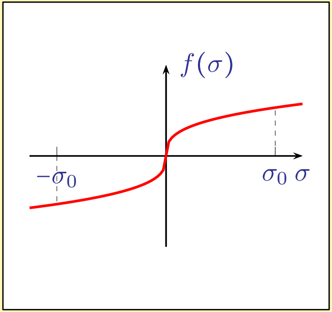
$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -f(x_2) - g(x_1)$$

Passivity of f and g:

$$\sigma f(\sigma) \geq 0, \forall \sigma \in [-\sigma_0, \sigma_0]$$
$$\sigma g(\sigma) \geq 0, \forall \sigma \in [-\sigma_0, \sigma_0]$$

Lyapunov function:

$$V(x) = \frac{x_2^2}{2} + \int_0^{x_1} g(\sigma)\mathrm{d}\sigma, \dot{V}(x) = -x_2 f(x_2)$$



(see next page)

Example: Nonlinear spring mass system with damper

Let

$$c \triangleq \min\left(V(-\sigma_0, 0), V(\sigma_0, 0)\right)$$
$$\dot{V}(x) \leq 0, \forall x \in \Omega_c \triangleq \{x | V(x) \leq c\}$$

$$\begin{aligned}
\dot{V}(x) &= 0 \Rightarrow x_2(t) = 0 \Rightarrow x_1(t) = x_{10} \Rightarrow \\
\dot{x}_2(t) &= 0 = -f(0) - g(x_{10}) \Rightarrow \\
g(x_{10}) &= 0 \Rightarrow x_{10} = 0
\end{aligned}$$

$$x = \begin{bmatrix} 0 \\ 0 \end{bmatrix} : \text{ largest invariant set inside } \Omega_c$$

Thus $x(t) \to \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ asymptotically.

◇

Regulation control: $\theta_d(t) \equiv \theta_d$ = constant

Dynamic equation:

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + N(\theta) = 0$$

Proposed control law (Computed torque):

$$\boxed{\tau = \underbrace{M(K_v\dot{e} + K_p e)}_{\text{feedback}} + \underbrace{M\ddot{\theta}_d + C\dot{\theta} + N}_{\text{feedforward}}, K_v, K_p > 0}$$

Closed loop equation:

$$M(\theta)(\ddot{e} + K_v\dot{e} + K_p e) = 0 \Leftrightarrow \ddot{e} + K_v\dot{e} + K_p e = 0$$

Regulation control: $\theta_d(t) \equiv \theta_d$ = constant

Dynamic equation:

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + N(\theta) = 0$$

Proposed control law (Computed torque):

$$\boxed{\tau = \underbrace{M(K_v\dot{e} + K_p e)}_{\text{feedback}} + \underbrace{M\ddot{\theta}_d + C\dot{\theta} + N}_{\text{feedforward}}, K_v, K_p > 0}$$

Closed loop equation:

$$M(\theta)(\ddot{e} + K_v\dot{e} + K_p e) = 0 \Leftrightarrow \ddot{e} + K_v\dot{e} + K_p e = 0$$

Practical issues:

1. computation demand
2. model sensitivity

PD control in joint space

PD control law: $\theta_d \equiv 0$

$$\boxed{\tau = K_v \dot{e} + K_p e, K_v, K_p > 0}$$

Closed-loop equation:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + K_v \dot{\theta} + K_p \theta = 0$$

Stability proof: Choose the Lyapunov function candidate:

$$V(\theta, \dot{\theta}) = \frac{1}{2}\dot{\theta}^T M(\theta)\dot{\theta} + \frac{1}{2}\theta^T K_p \theta \geq 0$$

$$\dot{V}(\theta, \dot{\theta}) = \dot{\theta}^T M \ddot{\theta} + \frac{1}{2}\dot{\theta}^T \dot{M}\dot{\theta} + \dot{\theta}^T K_p \theta = -\dot{\theta}^T K_v \dot{\theta} + \frac{1}{2}\dot{\theta}^T(\dot{M} - 2C)\dot{\theta}$$

$$= -\dot{\theta}^T K_v \dot{\theta} \leq 0$$

By Lasalle's invariance principle,

$$S = \{(\theta, \dot{\theta}) | \dot{\theta} \equiv 0\} \Rightarrow K_p \theta = 0 \Rightarrow \theta = 0$$

therefore $\theta \to 0$ asymptotically.

Augmented PD control in joint space

Augmented PD control:

$$\boxed{\tau = M(\theta)\ddot{\theta}_d + C(\theta,\dot{\theta})\dot{\theta}_d + N(\theta,\dot{\theta}) + K_v\dot{e} + K_p e, K_p, K_v > 0, \|\theta(0)\| \le \varepsilon}$$

Closed-loop equation:

$$M(\theta)\ddot{e} + C(\theta,\dot{\theta})\dot{e} + K_v\dot{\theta} + K_p\theta = 0$$

Stability proof: Choose the Lyapunov function candidate:

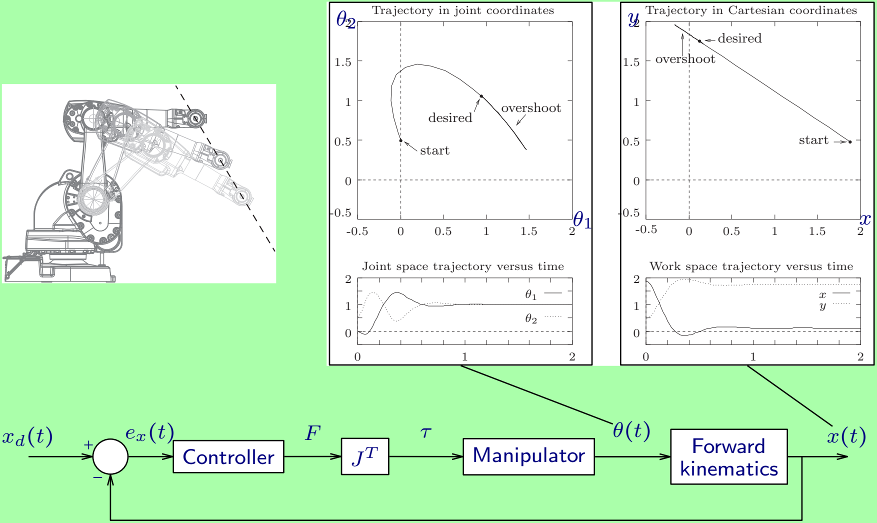
$$V(e,\dot{e},t) = \frac{1}{2}\dot{e}^T M(\theta)\dot{e} + \frac{1}{2}e^T K_p e + \varepsilon e^T M(\theta)\dot{e} > 0 \text{ for } \varepsilon \text{ small enough}$$

$$\begin{aligned}\dot{V} &= \dot{e}^T M\ddot{e} + \frac{1}{2}\dot{e}^T\dot{M}\dot{e} + \dot{e}^T K_p e + \varepsilon\dot{e}^T M\dot{e} + \varepsilon e^T(M\ddot{e} + \dot{M}\dot{e}) \\ &= -\dot{e}^T(K_v - \varepsilon M)\dot{e} + \frac{1}{2}\dot{e}^T(\dot{M} - 2C)\dot{e} + \varepsilon e^T(-K_p e - K_v\dot{e} - C\dot{e} + \dot{M}\dot{e}) \\ &= -\dot{e}^T(K_v - \varepsilon M)\dot{e} - \varepsilon e^T K_p e + \varepsilon e^T(-K_v + \frac{1}{2}\dot{M})\dot{e} < 0 \text{ for } \varepsilon \text{ small enough}\end{aligned}$$

Therefore $\theta(t) \to \theta_d(t)$ asymptotically exponentially.

Position Control in Task Space

Problem 1.B: Position Control in Task Space



Dynamic Equation in Task Space

Task space coordinates:

$$x = f(\theta) \in \mathbb{R}^n$$

Jacobian:

$$\dot{x} = J(\theta)\dot{\theta}, J(\theta) = \frac{\partial x}{\partial \theta}$$

Dynamic equation in task space:

$$\tilde{M}(\theta)\ddot{x} + \tilde{C}(\theta,\dot{\theta})\dot{x} + \tilde{N}(\theta,\dot{\theta}) = F$$

$$\tilde{M} \triangleq J^{-T} M J^{-1}$$

$$\tilde{C} \triangleq J^{-T}\left(C J^{-1} + M \frac{d}{dt}(J^{-1})\right)$$

$$\tilde{N} \triangleq J^{-T} N$$

$$\tau \triangleq J^T F$$

Structural properties of task space dynamics

Property 1:

1. $\tilde{M}(\theta)$ is symmetric and positive definite.
2. $\dot{\tilde{M}} - 2\tilde{C}$ is a skew-symmetric matrix.

Proof :

$\tilde{M}$ is symmetric:

$$\tilde{M}^T = (J^{-T} M J^{-1})^T = J^{-T} M J^{-1} = \tilde{M}$$

and positive definite:

$$\dot{x}^T \tilde{M} \dot{x} = \dot{\theta}^T M \dot{\theta} \geq 0, \dot{x}^T \tilde{M} \dot{x} = 0 \Leftrightarrow \dot{\theta} = 0 \Leftrightarrow \dot{x} = 0$$

$\dot{\tilde{M}} - 2\tilde{C}$ is skew symmetric:

$$\begin{aligned}\dot{\tilde{M}} - 2\tilde{C} &= J^{-T} \dot{M} J^{-1} + (\dot{J^{-T}}) M J^{-1} + J^{-T} M (\dot{J^{-1}}) - 2J^{-T} C J^{-1} - 2J^{-T} M (\dot{J^{-1}}) \\ &= J^{-T} (\dot{M} - 2C) J^{-1} + \left((\dot{J^{-T}}) M J^{-1} \right) - \left((\dot{J^{-T}}) M J^{-1} \right)^T \\ &= -(\dot{\tilde{M}} - 2\tilde{C})^T\end{aligned}$$

workspace control

PD control in workspace:

$$\boxed{\tau = J^T(K_v\dot{e}_x + K_p e_x), e_x \triangleq x_d - x}$$

Augmented PD control in workspace:

$$\boxed{\tau = J^T(\tilde{M}(\theta)\ddot{x}_d + \tilde{C}(\theta,\dot{\theta})\dot{x}_d + \tilde{N}(\theta,\dot{\theta}) + K_v\dot{e}_x + K_p e_x)}$$

workspace control

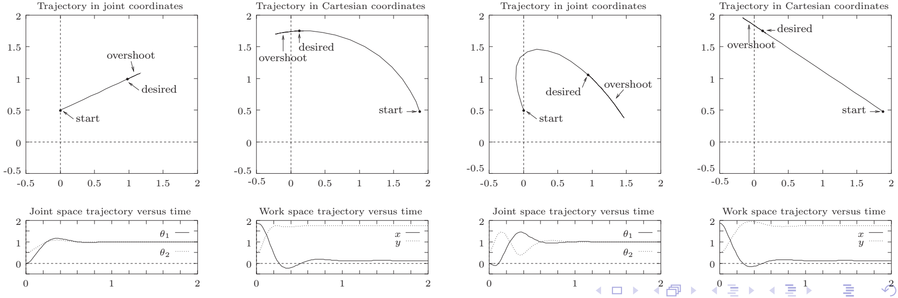
PD control in workspace:

$$\boxed{\tau = J^T(K_v \dot{e}_x + K_p e_x), e_x \triangleq x_d - x}$$

Augmented PD control in workspace:

$$\boxed{\tau = J^T(\tilde{M}(\theta)\ddot{x}_d + \tilde{C}(\theta,\dot{\theta})\dot{x}_d + \tilde{N}(\theta,\dot{\theta}) + K_v \dot{e}_x + K_p e_x)}$$

Comparison of joint space control and workspace control:



Adaptive computed torque control

Property 2: The equation of motion is linear in the inertia parameters:

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + N(\theta) = Y(\theta,\dot{\theta},\ddot{\theta})\pi$$

where $Y(\theta,\dot{\theta},\ddot{\theta})$ is called the regressor matrix and π is a constant vector, comprised of link masses, moments of inertia, etc.

Estimated equation of motion:

$$\hat{M}(\theta)\ddot{\theta} + \hat{C}(\theta,\dot{\theta})\dot{\theta} + \hat{N}(\theta) = Y(\theta,\dot{\theta},\ddot{\theta})\hat{\pi}$$

consider the following control law:

$$\begin{aligned}\tau &= \hat{M}(\theta)(\ddot{\theta}_d + K_v\dot{e} + K_p e) + \hat{C}(\theta,\dot{\theta})\dot{\theta} + \hat{N}(\theta,\dot{\theta}) \\ &= Y(\theta,\dot{\theta},\ddot{\theta})\hat{\pi} + \hat{M}(\theta)(\ddot{e} + K_v\dot{e} + K_p e)\end{aligned}$$

(see next page)

Adaptive computed torque control

The closed loop system:

$$Y(\theta,\dot\theta,\ddot\theta)(\pi-\hat\pi)=\hat M(\theta)(\ddot e+K_v\dot e+K_p e)$$

Define $x^T=(e^T,\dot e^T),\tilde\pi=\pi-\hat\pi$, then we have:

$$\dot x = Ax + B\hat M^{-1}(\theta)Y(\theta,\dot\theta,\ddot\theta)\tilde\pi,$$
$$A=\begin{bmatrix}0 & I\\ -K_p & -K_v\end{bmatrix}, B=\begin{bmatrix}0\\ I\end{bmatrix}$$

Choose the following Lyapunov function:

$$V(x,\tilde\pi)=\frac{1}{2}x^TPx+\frac{1}{2}\tilde\pi^T\Gamma\tilde\pi \text{ s.t. } P>0,\Gamma>0$$

then:

$$\dot V = x^TP\dot x+\tilde\pi^T\Gamma\dot{\tilde\pi}$$

(see next page)

Adaptive computed torque control

$$\begin{aligned}\dot{V} &= x^T P(Ax + B\hat{M}^{-1}(\theta)Y(\theta,\dot{\theta},\ddot{\theta})\tilde{\pi}) + \tilde{\pi}^T\Gamma\dot{\tilde{\pi}} \\ &= -x^T Qx + \tilde{\pi}^T(\Gamma\dot{\tilde{\pi}} + Y^T(\theta,\dot{\theta},\ddot{\theta})\hat{M}^{-1}(\theta)B^T Px)\end{aligned}$$

where $Q = -(PA + A^T P)/2 > 0$. If the following adaptive law:

$$\dot{\tilde{\pi}} = -\dot{\hat{\pi}} = -\Gamma^{-1}Y^T(\theta,\dot{\theta},\ddot{\theta})\hat{M}^{-1}(\theta)B^T Px$$

is adopted,

$$\dot{V} = -x^T Qx \leq 0$$

By Lasalle's principle, 0 is asymptotically stable.

Adaptive computed torque control

$$\begin{aligned}\dot{V} &= x^T P(Ax + B\hat{M}^{-1}(\theta)Y(\theta,\dot{\theta},\ddot{\theta})\tilde{\pi}) + \tilde{\pi}^T\Gamma\dot{\tilde{\pi}} \\ &= -x^T Qx + \tilde{\pi}^T(\Gamma\dot{\tilde{\pi}} + Y^T(\theta,\dot{\theta},\ddot{\theta})\hat{M}^{-1}(\theta)B^T Px)\end{aligned}$$

where $Q = -(PA + A^T P)/2 > 0$. If the following adaptive law:

$$\dot{\tilde{\pi}} = -\dot{\hat{\pi}} = -\Gamma^{-1}Y^T(\theta,\dot{\theta},\ddot{\theta})\hat{M}^{-1}(\theta)B^T Px$$

is adopted,

$$\dot{V} = -x^T Qx \le 0$$

By Lasalle's principle, 0 is asymptotically stable.

Proposition 3: Adaptive computed torque control

$$\begin{aligned}\tau &= Y(\theta,\dot{\theta},\ddot{\theta})\hat{\pi} + \hat{M}(\theta)(\ddot{e} + K_v\dot{e} + K_p e), K_v > 0, K_p > 0 \\ \dot{\hat{\pi}} &= \Gamma^{-1}Y^T(\theta,\dot{\theta},\ddot{\theta})\hat{M}^{-1}(\theta)B^T P\begin{bmatrix} e \\ \dot{e}\end{bmatrix}, P > 0, \Gamma > 0\end{aligned}$$

Metric,duality and orthogonality on T_qE

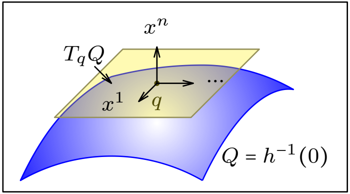


Figure 5.1

$$\mathcal{K} \triangleq \frac{1}{2} \ll \dot{q}, \dot{q} \gg_M = \frac{1}{2}\dot{q}^T M(q)\dot{q}$$

$$T_qQ^\perp = \{V_1 \in T_qE | \ll V_1, V_2 \gg_M \triangleq V_1^T M V_2 = 0, \forall V_2 \in T_qQ\}$$

$$T_q^*Q^\perp = \{f \in T_q^*E | \langle f, V \rangle = 0, \forall V \in T_qQ\}: \quad \text{constraint forces}$$

$$T_qE = T_qQ \oplus T_qQ^\perp, T_q^*E = T_q^*Q \oplus T_q^*Q^\perp$$

Metric,duality and orthogonality on T_qE

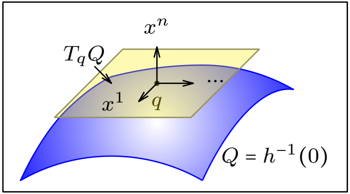


Figure 5.1

$$\mathcal{K} \triangleq \frac{1}{2} \ll \dot{q}, \dot{q} \gg_M = \frac{1}{2}\dot{q}^T M(q)\dot{q}$$

$$T_qQ^{\perp} = \{V_1 \in T_qE| \ll V_1, V_2 \gg_M \triangleq V_1^T M V_2 = 0, \forall V_2 \in T_qQ\}$$

$$T_q^*Q^{\perp} = \{f \in T_q^*E|\langle f, V\rangle = 0, \forall V \in T_qQ\}:$$ constraint forces

$$T_qE = T_qQ \oplus T_qQ^{\perp}, T_q^*E = T_q^*Q \oplus T_q^*Q^{\perp}$$

Definition:

$$M^{\flat} : T_qE \mapsto T_q^*E, \langle M^{\flat}V_1, V_2\rangle = V_1^T M V_2 = \ll V_1, V_2 \gg_M$$

$$M^{\sharp} : T_q^*E \mapsto T_qE, M^{\sharp} = {M^{\flat}}^{-1}$$

Unified geometric approach (Liu G.F., Li Z.X., [7])

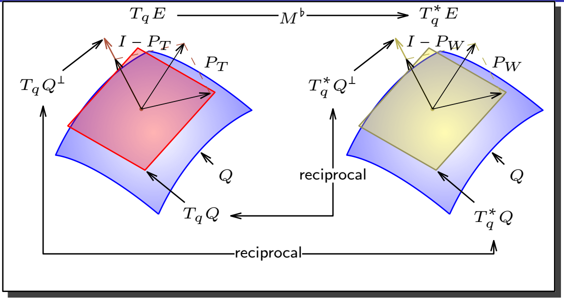


Figure 5.2

Unified geometric approach (Liu G.F., Li Z.X., [7])

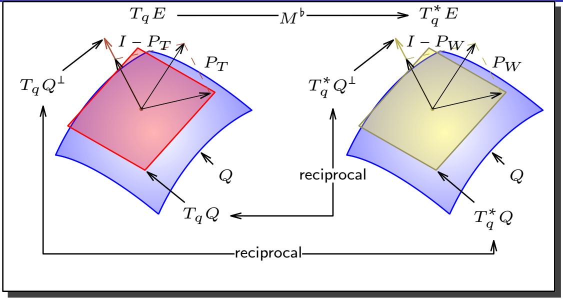


Figure 5.2

Property 3:
Under the basis $\frac{\partial}{\partial q_i}$ and $\mathrm{d}q_i, i = 1, \ldots, n$ of T_qE and T_q^*E respectively, the matrix representation of $M^\flat$ and $M^\sharp$ is M and M^{-1} respectively.

Unified geometric approach (Liu G.F., Li Z.X., [7])

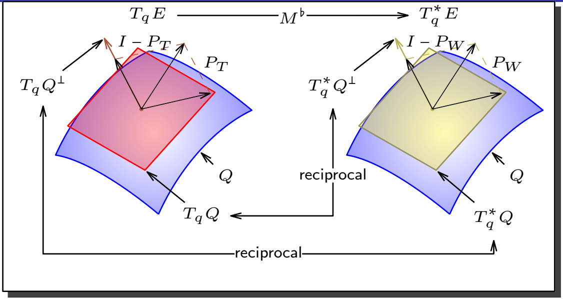


Figure 5.2

Property 3:
Under the basis $\frac{\partial}{\partial q_i}$ and $\mathrm{d}q_i, i = 1, \ldots, n$ of T_qE and T_q^*E respectively, the matrix representation of $M^\flat$ and $M^\sharp$ is M and M^{-1} respectively.

Property 4:

$$M^\sharp(T_q^*Q) = T_qQ$$
$$M^\sharp(T_q^*Q^\perp) = T_qQ^\perp$$

Given

$$h : E \mapsto \mathbb{R}^k, m = n - k$$

$$h_* \triangleq T_q h : T_q E \mapsto T_{h(q)}\mathbb{R}^k$$

$$h^* \triangleq T_q^* h : T_{h(q)}^* \mathbb{R}^k \mapsto T_q^* E$$

we have:

Property 5:

$$\ker h_* = T_q Q, h_*(T_q Q^\perp) = T_{h(q)}\mathbb{R}^k, h^*(T_{h(q)}^*\mathbb{R}^k) = T_q^* Q^\perp$$

$$\begin{tikzcd}
T_q^* E \arrow[d, "M^\sharp"'] & T_q^* \mathbb{R}^k \arrow[l, "h^*"'] \arrow[d, "M_2^\sharp"] \\
T_q E \arrow[r, "h_*"'] & T_{h(q)}\mathbb{R}^k
\end{tikzcd}$$

$$M_2^\sharp = h_* \circ M^\sharp \circ h^*$$

Lemma 1:

The map $(I - P_\omega) : T_q^* E \mapsto T_q^* Q^\perp$ given by

$$(I - P_\omega) = h^* \circ M_2^\flat \circ h_* \circ M^\sharp$$

is a well-defined projection map, with the property:

$$(I - P_\omega) f_1 = 0, \forall f_1 \in T_q^* Q$$

$$(I - P_\omega) f_2 = f_2, \forall f_2 \in T_q^* Q^\perp$$

Lemma 1:
The map $(I - P_\omega) : T_q^* E \mapsto T_q^* Q^\perp$ given by

$$(I - P_\omega) = h^* \circ M_2^\flat \circ h_* \circ M^\sharp$$

is a well-defined projection map, with the property:

$$(I - P_\omega) f_1 = 0, \forall f_1 \in T_q^* Q$$

$$(I - P_\omega) f_2 = f_2, \forall f_2 \in T_q^* Q^\perp$$

Proof :
Given $f_1 \in T_q^* Q, M^\sharp(f_1) \in T_q Q = \ker h_*$, then $(I - P_\omega) f_1 = 0$. For $f_2 \in T_q^* Q^\perp, \exists \lambda \in \mathbb{R}^{n-m}$ s.t. $f_2 = h^* \lambda$, and

$$(I - P_\omega) f_2 = h^* M_2^\flat h_* M^\sharp h^* \lambda = h^* \lambda = f_2$$

thus $P_\omega : T_q^* E \mapsto T_q^* Q$ is a well-defined projection map. Similarly,

$$P_T : T_q E \mapsto T_q Q, P_T = I - M^\sharp h^* M_2^b h_*$$

and

$$(I - P_T) : T_q E \mapsto T_q Q^\perp$$

are projection maps. □

Lemma 2:

$$
\begin{aligned}
P_\omega M &= M P_T \\
P_\omega h^* &= h_* P_T = 0 \\
P_T &= P_\omega^T
\end{aligned}
$$

Lemma 2:

$$P_\omega M = M P_T$$
$$P_\omega h^* = h_* P_T = 0$$
$$P_T = P_\omega^T$$

For nonholonomic constraints:

$$h_* \leftarrow A(q)$$
$$h^* \leftarrow A^*(q)$$
$$T_q Q \leftarrow \Delta_q$$
$$T_q^* Q^\perp \leftarrow \mathrm{span}\{a_i(q), i = 1, \ldots, k\}$$

application in hybrid velocity/force control.

□ Lagrange's equations of motion:

$$M(q)\ddot{q} + C(q,\dot{q}) + N = \tau + A^T(q)\lambda \Rightarrow$$
$$\lambda = (AM^{-1}A^T)^{-1}AM^{-1}(M\ddot{q} + (C + N - \tau))$$

□ Lagrange's equations of motion:

$$M(q)\ddot{q} + C(q,\dot{q}) + N = \tau + A^T(q)\lambda \Rightarrow$$
$$\lambda = (AM^{-1}A^T)^{-1}AM^{-1}(M\ddot{q} + (C + N - \tau))$$

Define $P_\omega = I - A^T(AM^{-1}A^T)^{-1}AM^{-1}$, then:

$$P_\omega M\ddot{q} + P_\omega C + P_\omega N = P_\omega \tau$$

Denote $\tilde{C} = P_\omega C, \tilde{N} = P_\omega N, \tilde{\tau} = P_\omega \tau$, $P_\omega M\ddot{\theta} \triangleq \tilde{M}\ddot{\theta}$ is the intertia force in T_q^*Q.

Lagrange's equations of motion:

$$M(q)\ddot{q} + C(q,\dot{q}) + N = \tau + A^T(q)\lambda \Rightarrow$$
$$\lambda = (AM^{-1}A^T)^{-1}AM^{-1}(M\ddot{q} + (C + N - \tau))$$

Define $P_\omega = I - A^T(AM^{-1}A^T)^{-1}AM^{-1}$, then:

$$P_\omega M\ddot{q} + P_\omega C + P_\omega N = P_\omega \tau$$

Denote $\tilde{C} = P_\omega C, \tilde{N} = P_\omega N, \tilde{\tau} = P_\omega \tau,\ P_\omega M\ddot{\theta} \triangleq \tilde{M}\ddot{\theta}$ is the intertia force in T_q^*Q.

Definition: Dynamics in T_q^*Q

$$\tilde{M}\ddot{\theta} + \tilde{C} + \tilde{N} = \tilde{\tau}$$

(Continues next slide)

Similarly

$$(I - P_\omega)(M\ddot{q} + C + N) = (I - P_\omega)\tau + A^T\tau$$

Let

$$P_T = I - M^{-1}A^T(AM^{-1}A^T)^{-1}A$$

then since $P_\omega M = MP_T$, we have:

Definition: Dynamics in $T_q^*Q^\perp$

$$M(I - P_T)(\ddot{q} + M^{-1}C) = (I - P_\omega)(\tau - N) + A^T\lambda$$

Similarly

$$(I - P_\omega)(M\ddot{q} + C + N) = (I - P_\omega)\tau + A^T\tau$$

Let

$$P_T = I - M^{-1}A^T(AM^{-1}A^T)^{-1}A$$

then since $P_\omega M = MP_T$, we have:

Definition: Dynamics in $T_q^*Q^\perp$

$$M(I - P_T)(\ddot{q} + M^{-1}C) = (I - P_\omega)(\tau - N) + A^T\lambda$$

Geometric Interpretation:

$$\nabla \leftrightarrow M$$

$$M\ddot{q} + C + N = \tau + A^T\lambda \Leftrightarrow M\nabla_{\dot{q}}\dot{q} = \tau - N + A^T\lambda$$

$\tilde{\nabla} \leftrightarrow$ induced metric on T_qQ

$S : TQ \otimes TQ \mapsto N(Q)$: 2^{nd} fundamental form

TQ : tangent vector field

$N(Q)$: normal vector field

$$\nabla_X Y = \tilde{\nabla}_X Y + S(X,Y)$$
$$M\underbrace{(I - P_T)(\ddot{q} + M^{-1}C)}_{S(\dot{q},\dot{q})} = (I - P_\omega)(\tau - N) + A^T\lambda$$

$MS(\dot{q},\dot{q})$: centrifugal force due to curvature of Q in E

$$\nabla_X Y = \tilde{\nabla}_X Y + S(X, Y)$$

$$M \underbrace{(I - P_T)(\ddot{q} + M^{-1} C)}_{S(\dot{q}, \dot{q})} = (I - P_\omega)(\tau - N) + A^T \lambda$$

$MS(\dot{q}, \dot{q})$: centrifugal force due to curvature of Q in E

Definition: Hybrid position/force control

$$M \tilde{\nabla}_{\dot{q}} \dot{q} = \tilde{\tau} - \tilde{N}$$

$$MS(\dot{q}, \dot{q}) = (I - P_\omega)(\tau - N) + A^T \lambda$$

Dynamics of a Spherical Pendulum

$$K = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) = \frac{1}{2}\dot{q}^T M \dot{q}$$

$$q = (x, y, z)^T, M = mI$$

$$h(q) = q^T q - r^2 = 0$$

$$A = (x, y, z), M_2^\sharp = AM^\sharp A^T = r^2/m$$

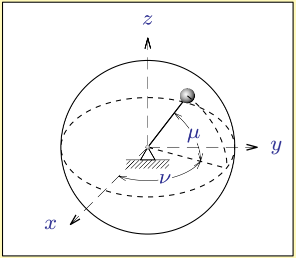


Figure 5.3

$$P_\omega = \frac{1}{r^2}\begin{bmatrix} y^2 + z^2 & -xy & -xz \\ -yz & x^2 + z^2 & -yz \\ -zx & -zy & x^2 + y^2 \end{bmatrix}$$

$$I - P_\omega = \frac{1}{r^2}\begin{bmatrix} x^2 & xy & xz \\ yx & y^2 & yz \\ zx & zy & z^2 \end{bmatrix}$$

(Continues next slide)

$$P_T = P_\omega^T = P_\omega$$

(μ, ν): Spherical coordinates

$$q = (r\cos\mu\cos\nu, r\cos\mu\sin\nu, r\sin\mu)^T$$

$$\begin{aligned}
\tilde{\nabla}_{\dot q}\dot q &= P_T(\nabla_{\dot q}\dot q) \\
&= \begin{bmatrix} -r\sin\mu\cos\nu & -r\sin\nu \\ -r\sin\mu\sin\nu & r\cos\nu \\ r\cos\mu & 0 \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \\
S(\dot q, \dot q) &= (I - P_T)(\nabla_{\dot q}\dot q) \\
&= (-\dot\mu^2 - \cos^2\mu\dot\nu^2)\begin{bmatrix} r\cos\mu\cos\nu \\ r\cos\mu\sin\nu \\ r\sin\mu \end{bmatrix}
\end{aligned}$$

where

$$\begin{aligned}
v_1 &= \ddot\mu + \sin\mu\cos\mu\dot\nu^2 \\
v_2 &= \cos\mu\ddot\nu - 2\sin\mu\dot\mu\dot\nu
\end{aligned}$$

Control Algorithm

1. holonomic constraints:

$$\tilde{q}: \text{coordinates of } Q$$

$$q = \psi(\tilde{q}) \Rightarrow \dot{q} = J \cdot \dot{\tilde{q}}$$

$$\tau = MJ(\ddot{\tilde{q}}_d - K_v \dot{\tilde{e}} - K_p \tilde{e}) + C_1 + N + A^T(-\lambda_d + K_I \int (\lambda - \lambda_d))$$

Control Algorithm

1. holonomic constraints:

$$\tilde{q} : \text{coordinates of } Q$$

$$q = \psi(\tilde{q}) \Rightarrow \dot{q} = J \cdot \dot{\tilde{q}}$$

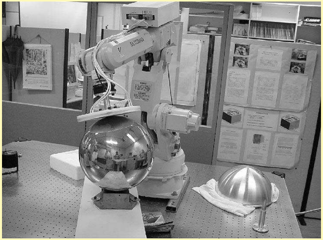
$$\tau = MJ(\ddot{\tilde{q}}_d - K_v \dot{\tilde{e}} - K_p \tilde{e}) + C_1 + N + A^T(-\lambda_d + K_I \int (\lambda - \lambda_d))$$

2. nonholonomic constraints:

Let $J(q) \in \mathbb{R}^{n \times m}$ be s.t. $AJ = 0$. Write $\dot{q} = J \cdot u$ for some u

$$\tau = MJ(\dot{u}_d - K_p(u - u_d)) + M\dot{J}u + C + N + A^T(-\lambda_d + K_I \int (\lambda - \lambda_d))$$

Example: 6-DoF manipulator on a sphere with frictionless point contact



- Contact constraint:

$$v_z = 0 \Leftrightarrow [0\ 0\ 1\ 0\ 0\ 0]\mathrm{Ad}_{g^{-1}_{f l_f}} V_{of} = 0$$

$\Rightarrow$ Holonomic constraint:

$\eta = (\alpha_o^T, \alpha_f^T, \psi)$: Parametrization of Q

$P_\omega = \mathrm{diag}(1,1,0,1,1,1)$

- Newton-Euler Equations of motion:

$$M\dot{V}_{of} - \mathrm{ad}^T_{V_{of}} M V_{of} = F_m + G + A^T\lambda$$

$$V_{of} = \begin{bmatrix} R_\psi M_o & -M_f & 0 \\ 0 & 0 & 0 \\ R_\psi R_o K_o M_o & -R_o K_f M_f & 0 \\ -T_o M_o & -T_f M_f & 1 \end{bmatrix} \begin{bmatrix} \dot{\alpha}_o \\ \dot{\alpha}_f \\ \dot{\psi} \end{bmatrix} \triangleq J\dot{\eta}$$

$$M J \ddot{\eta} + C_1 = F_m + G + A^T\lambda \qquad (*)$$

(Continues next slide)

$P_\omega(*)$:

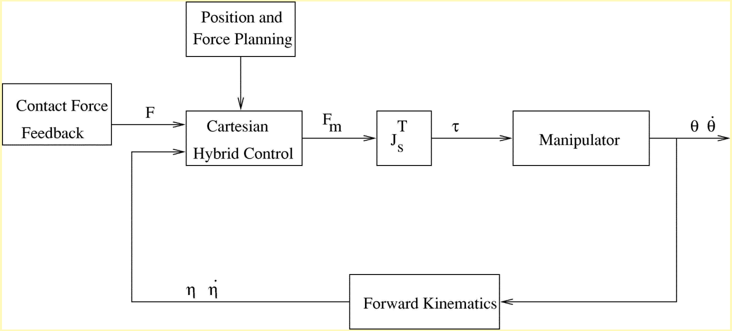
$$\tilde{M}\ddot{\eta} + \tilde{C}_1 = B_1 F_m + B_1 G$$

$$-\phi_3 - \lambda = b_2 F_m + b_2 G$$

$$\hat{F} = [f_1 \quad f_2 \quad f_4 \quad f_5 \quad f_6]^T = \tilde{M}(\ddot{\eta}_d - K_v \dot{\tilde{e}} - K_p \tilde{e}) + \tilde{C}_1 - B_1 G$$

$$f_3 = -\phi_3 - \lambda_d + K_I \int (\lambda - \lambda_d) - b_2 G$$

$$\tau = J_s^T F_m$$



Example: 6-DoF manipulator rolling on a sphere

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\dot{A}_1(q)} V_{l_o l_f} = 0$$

$$\begin{aligned} f_c &= A_1^T \lambda, \lambda \in \mathbb{R}^4 \\ \begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix} &= -R_0(K_f + R_\psi K_o R_\psi) M_f \dot{\alpha}_f \\ V_{of} &= \mathrm{Ad}_{g_{fl_f}} \cdot V_{l_o l_f} \\ &= \mathrm{Ad}_{g_{fl_f}} \begin{bmatrix} 0 \\ 0 \\ 0 \\ -R_o(K_f + R_\psi K_o R_\psi) M_f \\ o \end{bmatrix} \dot{\alpha}_f \\ &\triangleq J_f \dot{\alpha}_f \end{aligned}$$

$\mathrm{span}\{J_f\}$: Not involutive

(Continues next slide)

$$
\begin{aligned}
&MJ_f\ddot{\alpha}_f + (M\dot{J}_f\dot{\alpha}_f - \mathrm{ad}^T_{J_f\dot{\alpha}_f} MJ_f\dot{\alpha}_f) = F_m + G + A^T\lambda \\
F_m &= MJ_f(\ddot{\alpha}_{fd} - K_p(\dot{\alpha}_f - \dot{\alpha}_{fd})) + (M\dot{J}_f\dot{\alpha}_f - \mathrm{ad}^T_{J_f\dot{\alpha}_f} MJ_f\dot{\alpha}_f) \\
&+ A^T(-\lambda_d + \int(\lambda - \lambda_d)) - G
\end{aligned}
$$

$$M J_f \ddot{\alpha}_f + (M \dot{J}_f \dot{\alpha}_f - \mathrm{ad}^T_{J_f \dot{\alpha}_f} M J_f \dot{\alpha}_f) = F_m + G + A^T \lambda$$

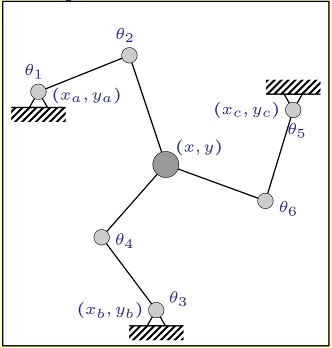
$$F_m = M J_f (\ddot{\alpha}_{fd} - K_p(\dot{\alpha}_f - \dot{\alpha}_{fd})) + (M \dot{J}_f \dot{\alpha}_f - \mathrm{ad}^T_{J_f \dot{\alpha}_f} M J_f \dot{\alpha}_f)$$
$$+ A^T(-\lambda_d + \int (\lambda - \lambda_d)) - G$$

◇ Example: Redundant parallel manipulator

$$\theta = (\theta_1, \ldots, \theta_6) \in E$$
$$\theta_a = (\theta_1, \theta_3, \theta_5)$$
$$\theta_p = (\theta_2, \theta_4, \theta_6)$$

$$H(\theta) = \begin{bmatrix} x_a + lc_1 + lc_{12} - x_b - lc_3 - lc_{34} \\ y_a + ls_1 + ls_{12} - y_b - ls_3 - ls_{34} \\ x_a + lc_1 + lc_{12} - x_c - lc_5 - lc_{56} \\ y_a + ls_1 + ls_{12} - y_c - ls_5 - ls_{56} \end{bmatrix} = 0$$

where $c_{ij} = \cos(\theta_i + \theta_j), s_{ij} = \sin(\theta_i + \theta_j)$.



(Continues next slide)

$$M_i(\theta) \in \mathbb{R}^{2\times 2} : i^{th} \text{ chain}$$

$$M(\theta) = \mathrm{diag}(M_1(\theta), \dots, M_3(\theta))$$

$$M(\theta)\ddot{\theta} + C + N = \tau + A^T\lambda$$

If all joints are actuated, we can achieve:

Position control of end-effector
+
internal grasping force

As $\tau_2, \tau_4, \tau_6 = 0$,

$$\tilde{\theta} \in \mathbb{R}^2 : \text{ local parametrization of } Q = H^{-1}(0)$$

$$\theta = \psi(\tilde{\theta}) : \text{ embedding of } Q \text{ in } E$$

$$\dot{\theta} = J\dot{\tilde{\theta}}$$

(Continues next slide)

Given $P_\omega : T_\theta^* E \mapsto T_\theta^* Q$, the dynamics in $T_\theta^* Q$ is given by:

$$
\begin{aligned}
P_\omega M J \ddot{\tilde{\theta}} + P_\omega (C_1 + N) &= P_\omega \tau \\
\tilde{\tau} &= (\tau_1, \tau_3, \tau_5) \\
\tilde{P}_\omega &= (P_1, P_3, P_5) \\
\hat{\tau} &= \hat{P}_\omega \tilde{\tau} = P_\omega \tau \in \mathbb{R}^6 \\
\hat{\tau} &= P_\omega M J (\ddot{\tilde{\theta}}_d - K_v \dot{\tilde{e}} - K_p \tilde{e}) + P_\omega (C_1 + N)
\end{aligned}
$$

◊

Gauge-invariant Formulation (Aghili, [8])

◇ **Square root factorization of inertia matrix:**

$$M = WW^T \text{(square root factorization)}$$

$$\begin{cases} v \triangleq W^T \dot{q} \in \mathbb{R}^n \\ u \triangleq W^{-1} \tau \in \mathbb{R}^n \end{cases} \qquad T = \frac{1}{2}\dot{q}^T M \dot{q} = \frac{1}{2} v^T v$$

Gauge-invariant Formulation (Aghili, [8])

◇ **Square root factorization of inertia matrix:**

$$M = WW^T \text{(square root factorization)}$$

$$\begin{cases} v \triangleq W^T \dot{q} \in \mathbb{R}^n \\ u \triangleq W^{-1} \tau \in \mathbb{R}^n \end{cases} \qquad T = \frac{1}{2}\dot{q}^T M \dot{q} = \frac{1}{2} v^T v$$

◇ **Lagrange's Equation:**

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial T}{\partial \dot{q}} - \frac{\partial T}{\partial q} = \tau \Rightarrow \frac{\mathrm{d}}{\mathrm{d}t}(Wv) - \frac{\partial v^T}{\partial q} v = \tau \Rightarrow$$

$$W\dot{v} + \dot{W}v - \frac{\partial v^T}{\partial q} v = \tau \Rightarrow \dot{v} + W^{-1}(\dot{W} - \frac{\partial v^T}{\partial q})v = W^{-1}\tau = u$$

Define $\Gamma \triangleq W^{-1}(\dot{W} - \frac{\partial v^T}{\partial q})$, then:

$$\boxed{\dot{v} + \Gamma v = u}$$

(Continues next slide)

Gauge-invariant Formulation (Aghili)

◊ **Change of coordinates:**

$$\bar{v} = V^T v, \bar{u} = V^T u, V \in U(n) \Rightarrow \frac{\mathrm{d}}{\mathrm{d}t}(V\bar{v}) + \Gamma(V\bar{v}) = V\bar{u} \Rightarrow$$

$$V\dot{\bar{v}} + \dot{V}\bar{v} + \Gamma V\bar{v} = V\bar{u} \Rightarrow \dot{\bar{v}} + V^T(\Gamma + \dot{V}V^T)V\bar{v} = \bar{u}$$

$$VV^T = I \Rightarrow \dot{V}V^T + V\dot{V}^T = 0 \Rightarrow \dot{V}V^T = -(\dot{V}V^T)^T =: -\Omega \Rightarrow$$

$$\bar{\Gamma} \triangleq V^T(\Gamma - \Omega)V, \Rightarrow \boxed{\dot{\bar{v}} + \bar{\Gamma}\bar{v} = \bar{u}}$$

Gauge-invariant Formulation (Aghili)

◇ **Change of coordinates:**

$$\bar{v} = V^T v, \bar{u} = V^T u, V \in U(n) \Rightarrow \frac{\mathrm{d}}{\mathrm{d}t}(V\bar{v}) + \Gamma(V\bar{v}) = V\bar{u} \Rightarrow$$

$$V\dot{\bar{v}} + \dot{V}\bar{v} + \Gamma V\bar{v} = V\bar{u} \Rightarrow \dot{\bar{v}} + V^T(\Gamma + \dot{V}V^T)V\bar{v} = \bar{u}$$

$$VV^T = I \Rightarrow \dot{V}V^T + V\dot{V}^T = 0 \Rightarrow \dot{V}V^T = -(\dot{V}V^T)^T =: -\Omega \Rightarrow$$

$$\bar{\Gamma} \triangleq V^T(\Gamma - \Omega)V, \Rightarrow \boxed{\dot{\bar{v}} + \bar{\Gamma}\bar{v} = \bar{u}}$$

◇ **Pfaffian constraint:**

$$A(q)\dot{q} = 0, A(q) \in \mathbb{R}^{m\times n}, \Lambda \triangleq AW^{-T} \Rightarrow \Lambda v = 0,$$

Gauge-invariant Formulation (Aghili)

◇ **Change of coordinates:**

$$\bar{v} = V^T v, \bar{u} = V^T u, V \in U(n) \Rightarrow \frac{\mathrm{d}}{\mathrm{d}t}(V\bar{v}) + \Gamma(V\bar{v}) = V\bar{u} \Rightarrow$$

$$V\dot{\bar{v}} + \dot{V}\bar{v} + \Gamma V\bar{v} = V\bar{u} \Rightarrow \dot{\bar{v}} + V^T(\Gamma + \dot{V}V^T)V\bar{v} = \bar{u}$$

$$VV^T = I \Rightarrow \dot{V}V^T + V\dot{V}^T = 0 \Rightarrow \dot{V}V^T = -(\dot{V}V^T)^T =: -\Omega \Rightarrow$$

$$\bar{\Gamma} \triangleq V^T(\Gamma - \Omega)V, \Rightarrow \boxed{\dot{\bar{v}} + \bar{\Gamma}\bar{v} = \bar{u}}$$

◇ **Pfaffian constraint:**

$$A(q)\dot{q} = 0, A(q) \in \mathbb{R}^{m\times n}, \Lambda \triangleq AW^{-T} \Rightarrow \Lambda v = 0,$$

◇ **Lagrange's equation with constraint:**

$$\boxed{\dot{v} + \Gamma v = u + \Lambda^T \lambda}$$

(Continues next slide)

Gauge-invariant Formulation (Aghili)

◊ **SVD of Λ:**

$$\boxed{\Lambda = U\Sigma V^T, \bar{v} = V^T v, \bar{u} = V^T u, \bar{\Lambda} \triangleq \Lambda V}$$

where $\bar{\Lambda}\bar{v} = 0$ and:

$$\sigma = \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix}, S = \mathrm{diag}(\sigma_1, \ldots, \sigma_r), \sigma_1 \geq \cdots \geq \sigma_r, r \leq m$$

$$\begin{cases} U = [U_1 \ U_2], U_1 \in \mathbb{R}^{m\times r}, U_2 \in \mathbb{R}^{m\times(m-r)} \\ V = [V_1 \ V_2], V_1 \in \mathbb{R}^{n\times r}, V_2 \in \mathbb{R}^{n\times(n-r)} \end{cases} \Rightarrow$$

$$\bar{\Lambda} = [\Lambda_r \ 0_{m\times(n-r)}], \Lambda_r \triangleq U_1 S, \bar{\Lambda}\bar{v} = 0 \Rightarrow$$

$$\bar{v} = \begin{bmatrix} 0_{r\times 1} \\ v_r \end{bmatrix}, v_r \triangleq V_2^T v = V_2^T W^T \dot{q}$$

(Continues next slide)

Gauge-invariant Formulation (Aghili)

$$\bar{u} = \begin{bmatrix} u_o \\ u_r \end{bmatrix}, \begin{cases} u_o = V_1^T W^{-1} \tau \\ u_r = V_2^T W^{-1} \tau \end{cases}$$

$$\bar{\Gamma}_{ij} \triangleq V_i^T (\Gamma - \Omega) V_j, i, j = 1, 2, \Gamma_r \triangleq \bar{\Gamma}_{22}, \Gamma_o \triangleq \bar{\Gamma}_{12}$$

Gauge-invariant Formulation (Aghili)

$$\bar{u} = \begin{bmatrix} u_o \\ u_r \end{bmatrix}, \begin{cases} u_o = V_1^T W^{-1} \tau \\ u_r = V_2^T W^{-1} \tau \end{cases}$$

$$\bar{\Gamma}_{ij} \triangleq V_i^T (\Gamma - \Omega) V_j, i, j = 1, 2, \Gamma_r \triangleq \bar{\Gamma}_{22}, \Gamma_o \triangleq \bar{\Gamma}_{12}$$

◇ **Decoupled equation of motion/constrained force:**

$$\boxed{\begin{aligned} \dot{v}_r + \Gamma_r v_r &= u_r \\ \Gamma_o v_r &= u_o + \Gamma_r^T \lambda \end{aligned}}$$

Gauge-invariant Formulation (Aghili)

$$\bar{u} = \begin{bmatrix} u_o \\ u_r \end{bmatrix}, \begin{cases} u_o = V_1^T W^{-1} \tau \\ u_r = V_2^T W^{-1} \tau \end{cases}$$

$$\bar{\Gamma}_{ij} \triangleq V_i^T (\Gamma - \Omega) V_j, i, j = 1, 2, \Gamma_r \triangleq \bar{\Gamma}_{22}, \Gamma_o \triangleq \bar{\Gamma}_{12}$$

◇ **Decoupled equation of motion/constrained force:**

$$\boxed{\begin{aligned} \dot{v}_r + \Gamma_r v_r &= u_r \\ \Gamma_o v_r &= u_o + \Gamma_r^T \lambda \end{aligned}}$$

◇ **Combined equation of motion (Kane's equation):**

$$\boxed{\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} q \\ v_r \end{bmatrix} = \begin{bmatrix} W^{-T} V_2 \\ -\Gamma_r \end{bmatrix} v_r + \begin{bmatrix} 0 \\ I \end{bmatrix} u_r}$$

(Continues next slide)

Gauge-invariant Formulation (Aghili)

◇ **Composite error vector** ε

$$q = q(\theta), \theta \in \mathbb{R}^{n-r} : \text{generalized coordinate (of } Q \subset \mathbb{R}^n\text{)}$$

$$v_r = B(\theta)\dot{\theta}, B(\theta) \triangleq V_2^T W^T J, J \triangleq \frac{\partial q}{\partial \theta}, \tilde{\theta} \triangleq \theta - \theta_d, \tilde{v}_r \triangleq v_r - v_{r_d} \Rightarrow$$

$$\varepsilon \triangleq \tilde{v}_r + BK_p\tilde{\theta} = B(\dot{\tilde{\theta}} + K_p\tilde{\theta}) : \text{ composite error}$$

$$s \triangleq v_{r_d} - BK_p\tilde{\theta} = v_r - \varepsilon = B(\dot{\theta}_d - K_p\tilde{\theta}), \tilde{\lambda} \triangleq \lambda - \lambda_d$$

Gauge-invariant Formulation (Aghili)

◇ **Composite error vector** ε

$$q = q(\theta), \theta \in \mathbb{R}^{n-r} : \text{generalized coordinate (of } Q \subset \mathbb{R}^n)$$

$$v_r = B(\theta)\dot{\theta}, B(\theta) \triangleq V_2^T W^T J, J \triangleq \frac{\partial q}{\partial \theta}, \tilde{\theta} \triangleq \theta - \theta_d, \tilde{v}_r \triangleq v_r - v_{r_d} \Rightarrow$$

$$\varepsilon \triangleq \tilde{v}_r + BK_p\tilde{\theta} = B(\dot{\tilde{\theta}} + K_p\tilde{\theta}) : \text{ composite error}$$

$$s \triangleq v_{r_d} - BK_p\tilde{\theta} = v_r - \varepsilon = B(\dot{\theta}_d - K_p\tilde{\theta}), \tilde{\lambda} \triangleq \lambda - \lambda_d$$

◇ **Hybrid position/force control**

$$\boxed{\begin{aligned} u_r &= \dot{s} + \Gamma_r s - K_d \varepsilon \\ u_o &= -\Lambda_r^T \lambda_d + \Gamma_o v_r \end{aligned}}$$

Note: integration term $K_I \int (\lambda - \lambda_d)$ is missing from u_o.

(Continues next slide)

Gauge-invariant Formulation (Aghili)

$$\dot{v}_r + \Gamma_r v_r = u_r = \frac{\mathrm{d}}{\mathrm{d}t}(v_r - \varepsilon) + \Gamma_r(v_r - \varepsilon) - K_d\varepsilon \Rightarrow \dot{\varepsilon} = -(\Gamma_r + K_d)\varepsilon$$

$$\Lambda_r^T \lambda + \Gamma_o v_r = u_o = -\Lambda_r^T \lambda_d + \Gamma_o v_r \Rightarrow \Lambda_r^T \tilde{\lambda} + K_I \int \tilde{\lambda} = 0$$

Gauge-invariant Formulation (Aghili)

$$\dot{v}_r + \Gamma_r v_r = u_r = \frac{\mathrm{d}}{\mathrm{d}t}(v_r - \varepsilon) + \Gamma_r(v_r - \varepsilon) - K_d\varepsilon \Rightarrow \dot{\varepsilon} = -(\Gamma_r + K_d)\varepsilon$$

$$\Lambda_r^T\lambda + \Gamma_o v_r = u_o = -\Lambda_r^T\lambda_d + \Gamma_o v_r \Rightarrow \Lambda_r^T\tilde{\lambda} + K_I\int\tilde{\lambda} = 0$$

◇ **Equivalence to the Geometric approach**

$$\boxed{\begin{aligned} P_\omega &= WV_2V_2^TW^{-1} \\ I - P_\omega &= WV_1V_1^TW^{-1} \end{aligned}} \Rightarrow \begin{aligned} V^TW^{-1}P_\omega\tau &= \begin{bmatrix} 0 \\ V_2^TW^{-1}\tau \end{bmatrix} = \begin{bmatrix} 0 \\ u_r \end{bmatrix} \\ V^TW^{-1}(I - P_\omega)\tau &= \begin{bmatrix} V_1^TW^{-1}\tau \\ 0 \end{bmatrix} = \begin{bmatrix} u_o \\ 0 \end{bmatrix} \end{aligned}$$

(Continues next slide)

Gauge-invariant Formulation (Aghili)

1. Geometric approach (K_p, K_I, K_d)**:**

$$u = \underbrace{\begin{bmatrix} V_1^T W^{-1} A^T(-\lambda_d + K_I \int(\lambda - \lambda_d)) \\ V_2^T W^T J(\ddot{\theta}_d - K_d \dot{\tilde{\theta}} - K_p \tilde{\theta}) \end{bmatrix}}_{fb} + \underbrace{V^T W^{-1} C_1}_{ff}$$

2. Gauge-invariant formulation $(\tilde{K}_p, \tilde{K}_d)$**:**

$$u = \begin{bmatrix} u_o \\ u_r \end{bmatrix} = \underbrace{\begin{bmatrix} V_1^T W^{-1} A^T(-\lambda_d) \\ V_2^T W^T J(\ddot{\theta}_d - K_d' \dot{\tilde{\theta}} - K_p' \tilde{\theta}) \end{bmatrix}}_{fb} + \underbrace{C_1'}_{ff} \Rightarrow$$

$$K_d' \triangleq \tilde{K}_p + (V_2^T W^T J)^{-1} \tilde{K}_d (V_2^T W^T J)$$

$$K_p' \triangleq (V_2^T W^T J)^{-1} \tilde{K}_d (V_2^T W^T J) \tilde{K}_p$$

† End of Section †

References

- J.C. Latombe. *Robot motion planning*. Springer Verlag, 1990.
- Z.X. Li and J.F. Canny. *Nonholonomic motion planning*. Kluwer Academic, 1993.
- J. Craig. *Introduction to Robotics: Mechanics and Control*. 2nd ed. Prentice Hall, 2005.
- L. Biagiotti and C. Melchiorri. *Trajectory Planning for Automatic Machines and Robots*. Springer Verlag, 2008.
- M. Zefran and V. Kumar. "Interpolation schemes for rigid body motions". In: *CAD Computer Aided Design* 30.3 (1998), pp. 179–189.
- F.C. Park and B. Ravani. "Smooth Invariant Interpolation of Rotations". In: *ACM Transactions on Graphics* 16.3 (1997), pp. 277–295.
- G.F. Liu and Z.X. Li. "A unified geometric approach to modeling and control of constrained mechanical systems". In: *Robotics and Automation, IEEE Transactions on* 18.4 (2002), pp. 574–587.
- F. Aghili. "A Gauge-Invariant Formulation for Constrained Mechanical Systems Using Square-Root Factorization and Unitary Transformation". In: *Journal of Computational and Nonlinear Dynamics* 4 (2009), p. 031010.
- B. Siciliano, L. Sciavicco, and L. Villani. *Robotics: modelling, planning and control*. Springer Verlag, 2009.