Chapter 9 Trajectory Generation

14.5.23

The robot's position as a function of time is called a trajectory.

Can be :

- specified by task : e.g following an object
- move from one position to another in a given time

Trajectory : path and time scaling

- path : geometric description of sequence of configurations
- time scaling : times when these configurations are reached

Trajectory should be a smooth function of time and respect limits : joint velocities, accelerations, torques.

Three cases considered :

- Point-to-point straight-line trajectories passing through a sequence of timed via points
    - in joint space
    - in task space
- Minimum-time trajectories along specified paths taking actuator limits into considerations

9.1 Definitions

<u>Trajectory</u>: Robot configuration as function of time

$\theta(t), \quad t \in [0, T]$

<u>Path</u> : Robot movement decoupled from time (curve)

$\theta(s), \quad s \in [0, 1]$

↳ $\theta(s(t)), \quad s : [0, T] \to [0, 1]$

- $s(t)$: maps time range 0 to t to range 0 to 1
- $s(t)$: called '<u>time scaling</u>'
  ↳ controls how fast the path is followed

Acceleration and velocity along the trajectory:

Velocity: $\dot{\theta} = \frac{d\theta}{ds}\dot{s}$

Acceleration: $\ddot{\theta} = \frac{d\theta}{ds}\ddot{s} + \frac{d^2\theta}{ds^2}\dot{s}^2$

↳ each $\theta(s)$ and $s(t)$ must be twice differentiable

9.2.1 Straight-Line Paths

"Straight line" from θ_{start} to θ_{end} can be defined in

- joint space: simple (and in convex space)
- task space

Joint space:

$$\theta(s) = \theta_{start} + s(\theta_{end} - \theta_{start}),\quad s \in [0,1]$$

with:

$$\frac{d\theta}{ds} = \theta_{end} - \theta_{start}$$

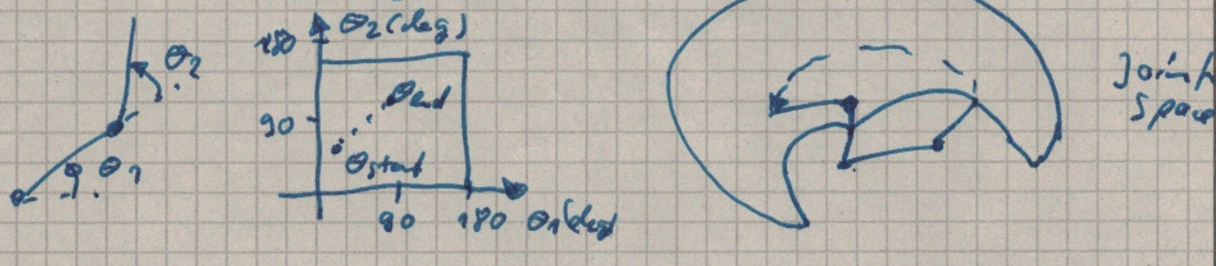
$$\frac{d^2\theta}{ds^2} = 0$$

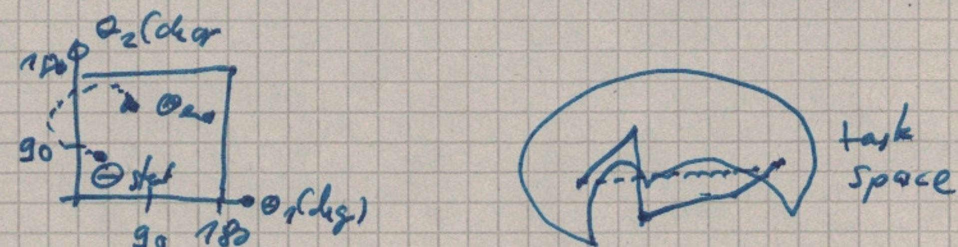
Straight lines in joint space usually don't lead to straight lines in task space

Task space:

$$X_{start}(s) = X_{start} + s(X_{end} - X_{start}),\quad s \in [0,1]$$

- X_{start} and X_{end} are represented by minimum set of coordinates
- kinematic singularities are dangerous (velocities may large)
- Task space might not be convex: some points on a straight line might not be reachable





If x_{start} and x_{end} represented in $SE(3)$
there is not a concept of straight line (in $SE(3)$)

$$x_{start} + s(x_{end} - x_{start}) \text{ generally is not in } SE(3)$$

↳ use screw motion (rotation and translation)
to move end-effector from $x_{start} = X(0)$ to $x_{end} = X(1)$
- write start and end configuration in frame {s}

$$x_{start,end} = x_{start,s}\, x_{s,end} = x_{s,start}^{-1}\, x_{s,end}$$

⇒ $\log(x_{s,start}^{-1} x_{s,end})$ is matrix representation of the twist expressed in {start} frame that takes x_{start} to x_{end} in unit time

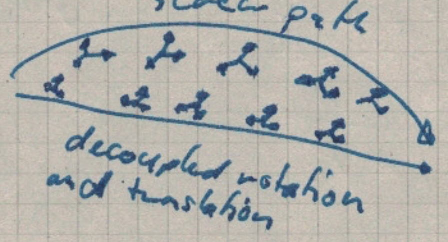
Path: $X(s) = x_{start} \exp(\log(x_{start}^{-1} x_{end})s)$
- x_{start} is post multiplied by matrix exponential since twist is represented in {start} frame (not in {s} frame)

↳ screw motion is constant ⇒ "straight line"

Decouple $X = (R, p)$
- R: rotation
- p: translation

$$p(s) = p_{start} + s(p_{end} - p_{start})$$

$$R(s) = R_{start} \exp(\log(R_{start}^T R_{end})s)$$



9.2.2 Time scaling a Straight-Line Path

25.5.23

Time scaling $s(s)$ of a path shoul ensure:

- smooth motion
- satisfy constraints

Joint space:

- $\theta(s) = \theta_{start} + s(\theta_{end} - \theta_{start})$, $s \in [0, 1]$
- $\dot{\theta}(s) = \dot{s}(\theta_{end} - \theta_{start})$
- $\ddot{\theta}(s) = \ddot{s}(\theta_{end} - \theta_{start})$

analogous for taskspace (replace $\theta, \dot{\theta}, \ddot{\theta}$ with $X, \dot{X}, \ddot{X}$)

Polynomial Time Scaling

Third order Polynomials (cubic):

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

constraints $s(0) = 0$ $\quad$ $s(T) = 1$

$\dot{s}(0) = 0$ $\quad$ $\dot{s}(T) = 0$

$\hookrightarrow \dot{s}(t) = a_1 + 2a_2 t + 3a_3 t^2$

$\Rightarrow$ solve with 4 constraint at $t=0$ and $t=T$

$\hookrightarrow$ $a_0 = 0$, $a_1 = 0$, $a_2 = \frac{3}{T_2}$, $a_3 = -\frac{2}{T_3}$

$\hookrightarrow$ $s = a_2 t^2 + a_3 t^3$

$$\Rightarrow \theta(t) = \theta_{start} + \left(\frac{3t^2}{T^2} - \frac{2t^3}{T^3}\right)(\theta_{end} - \theta_{start})$$

$$\dot{\theta}(t) = \left(\frac{6t}{T^2} - \frac{6t^2}{T^3}\right)(\theta_{end} - \theta_{start})$$

$$\ddot{\theta}(t) = \left(\frac{6}{T^2} - \frac{12t}{T^3}\right)(\theta_{end} - \theta_{start})$$

and

$$\dot{\theta}_{max} = \frac{3}{2T}(\theta_{end} - \theta_{start})$$

$$\ddot{\theta}_{max} = \left|\frac{6}{T_2}(\theta_{end} - \theta_{start})\right|$$

$$\ddot{\theta}_{min} = \left|\frac{6}{T_2}(\theta_{end} - \theta_{start})\right|$$

known limits to θ, $\dot{\theta}$ and $\ddot{\theta}$ can be checked.

Fifth-Order Polynomials:

3rd Order Polynomials lead to an jump in the acceleration at $t=0$ and $t=T$:

infinite jerk (derivative of acceleration)

→ can cause vibrations of robot

Improvement:

additional constraints ($\ddot{s}(0)=0, \ddot{s}(T)=0$)

↳ $s(0)=0$ $\quad s(T)=1$

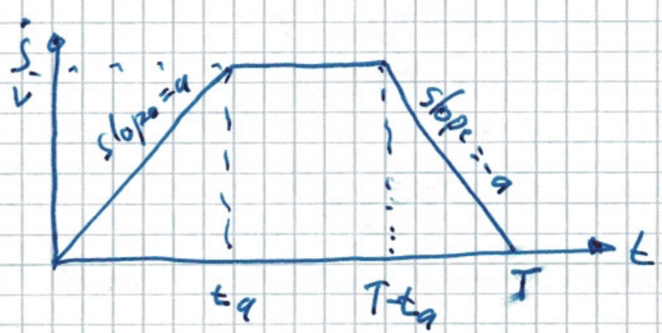
$\dot{s}(0)=0$ $\quad \dot{s}(T)=0$

$\ddot{s}(0)=0$ $\quad \ddot{s}(T)=0$

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

↳ solve for $a_0, \dots, a_5$

Trapezoidal Motion Profiles



1. constant acceleration phase $\ddot{s}=a$ for time t_a
2. constant velocity phase $\dot{s}=v$ for time $t_v = T-2t_a$
3. constant deceleration phase $\ddot{s}=-a$ for time t_a

Not as smooth as cubic time scaling.

If there are known constant limits on joint velocities $\dot{\theta}_{limit} \in \mathbb{R}^n$ and on joint accelerations $\ddot{\theta}_{limit} \in \mathbb{R}^n$

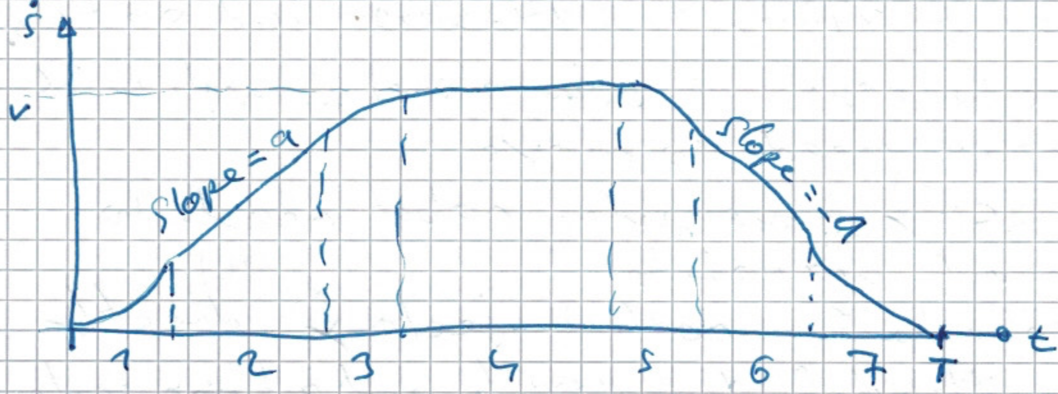
$$|(\theta_{end} - \theta_{start})v| \leq \dot{\theta}_{limit}$$

$$|(\theta_{end} - \theta_{start})a| \leq \ddot{\theta}_{limit}$$

↳ fastest straight-line motion possible

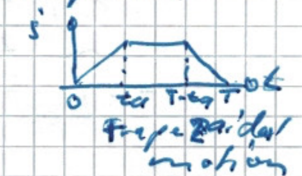
see more details in book

S - Curve Time Scalings



1, 3, 5, 7 : constant jerk (derivative of acceleration)
2, 6 : constant acceleration
4 : constant velocity

Trapezoidal motions cause discontinuous jumps in acceleration at $t \in \{0, t_a, T - t_a, T\}$.



Solution:

Smoother S-curve time scaling

7 stages

1.) constant jerk $\frac{d^3 s}{dt} = J$ until desired acceleration $\ddot{s} = a$

2.) constant acceleration until desired velocity $\dot{s} = v$

3.) constant jerk $-J$ until $\ddot{s} = 0$ at time when $\dot{s} = v$

4.) coasting at constant v

5.) constant negative jerk $-J$

6.) constant deceleration $-a$

7.) constant positive jerk J until $\ddot{s} = 0$ and $\dot{s} = 0$ when $s = 1$

Given subset of v, a, J and $T \Rightarrow$ calculate switching time between stages.

Ensure that all stages are actually achieved.

Polynomial Via Point Trajectories

For more flexibility on path shape and on the speed:

- specify points (configurations) through which robot needs to move
- specify times, when robot needs to reach via points

↳ solve for ~~t~~ smooth trajectory $\theta(t)$ directly without finding a path $q_{\theta(s)}$ and a time scale $q_{s(t)}$ first.

n-dimensional joint space

for joint i moving from via point j to $j+1$ define motion as 3rd order polinomial of time.

$a_0 + a_1\theta + a_2\theta^2 + a_3\theta^3$

Apply 4 terminal constraints:

- initial and final position
- initial and final velocity

of joint i

↳ solve for coefficients $a_0, \dots, a_3$

Each via point has

- the time that robot passes throug configuration
- velocity at that time

Eeach segment between via points has:

for each degree of freedom:

- 4 coefficients
- 4 terminal constraints

↳ solve for trajectories between via points

Tangent path at via points: specified velocities at via points

↳ velocities at via point can be adjusted to shape the path

- Velocities and positions are continuous at via points
- accelerations are <u>not</u> continuous at via points

↳ possible to leave velocities at via point free but require the velocities and accelerations to be the same before and after a via point.

B-splines are also a common way for trajectory generation.

9.4 Time-Optimal Time Scaling

18.5.23

When path $\theta(s)$ is fully specified.
i.e. given by task, path planer...

Time scaling can be minimized for:
- energy consumed
- meeting constrains
- time of motion (speed)
- ...

Most robots: state-dependent joint actuator limits:

$$\tau_i^{min}(\theta, \dot{\theta}) \leq \tau_i \leq \tau_i^{max}(\theta, \dot{\theta})$$: limits on the i-th actuator

Dynamics:

$$M(\theta)\ddot{\theta} + \underbrace{c(\theta, \dot{\theta})}_{= \dot{\theta}^T \Gamma(\theta) \dot{\theta}} + g(\theta) = \tau$$

: $\Gamma(\theta)$ 3-dimensional tensor of Christoffel symbols

$$\dot{\theta} = \frac{d\theta}{ds}\dot{s}, \quad \ddot{\theta} = \frac{d\theta}{ds}\ddot{s} + \frac{d^2\theta}{ds^2}\cdot\dot{s}^2$$

$$\hookrightarrow \underbrace{\left(M(\theta(s))\frac{d\theta}{ds}\right)}_{m(s) \in \mathbb{R}^n}\ddot{s} + \underbrace{\left(M(\theta(s))\frac{d^2\theta}{ds^2} + \left(\frac{d\theta}{ds}\right)^T \Gamma(\theta(s))\frac{d\theta}{ds}\right)}_{c(s) \in \mathbb{R}^n}\dot{s}^2$$

$$+ \underbrace{g(\theta(s))}_{g(s) \in \mathbb{R}^n} = \tau$$

as vector equation:

$$m(s)\ddot{s} + c(s)\dot{s}^2 + g(s)$$

- $m(s)$: effective inertia of the robot when confined to the path $\theta(s)$
- $c(s)\dot{s}$: quadratic velocity terms
- $g(s)$: gravitational torque

Actuation constraints can be written as function of s:

$$\tau_i^{min}(s,\dot{s}) \leq \tau_i \leq \tau_i^{max}(s,\dot{s})$$

$$\Leftrightarrow \tau_i^{min}(s,\dot{s}) \leq m_i(s)\ddot{s} + c_i(s)\dot{s}^2 + g_i(s) \leq \tau_i^{max}(s,\dot{s})$$

Let: • $L_i(s,\dot{s})$: minimum acceleration $\ddot{s}$ satifying ith component

• $U_i(s,\dot{s})$: maximum "

Depending on sign of $m_i(s)$:

if $m_i(s) > 0$, $L_i(s,\dot{s}) = \dfrac{\tau_i^{min}(s,\dot{s}) - c(s)\dot{s}^2 - g(s)}{m_i(s)}$

$$U_i(s,\dot{s}) = \frac{\tau_i^{max}(s,\dot{s}) - c(s)\dot{s}^2 - g(s)}{m_i(s)}$$

if $m_i(s) < 0$, $L_i(s,\dot{s}) = \dfrac{\tau_i^{max}(s,\dot{s}) - c(s)\dot{s}^2 - g(s)}{m_i(s)}$

$$U_i(s,\dot{s}) = \frac{\tau_i^{min}(s,\dot{s}) - c(s)\dot{s}^2 - g(s)}{m_i(s)}$$

if $m_i(s) = 0$: zero inertia point

Define :

$$L(s,\dot{s}) = \max_i L_i(s,\dot{s})$$

$$U(s,\dot{s}) = \min_i U_i(s,\dot{s})$$

Actuator limits (as state-dependent time-scaling constraints):

$$L(s,\dot{s}) \leq \ddot{s} \leq U(s,\dot{s})$$

<u>Time-optimal time-scaling problem:</u>

can be generalized for $\dot{s}(0) > 0$, $\dot{s}(T) > 0$

Given: path $\theta(s)$, $s \in [0,1]$
initial state $(s_0, \dot{s}_0) = (0,0)$
final state $(s_f, \dot{s}_f) = (1,0)$

find monotonically increasing twice-differentiable time scaling $s: [0,T] \to [0,1]$ that:

- $s(0) = 0$, $\dot{s}(0) = 0$, $s(T) = 1$, $\dot{s}(T) = 0$
- minimize total travel time T along path, respecting the actuator constraints

see details in book