3.1.2 Multi-dimensional cases

$s \in \mathbb{R}^M$ eg. $s = (x, y, \theta) \in SE(2)$

$s = (x, y, z, \mathring{q}) \in SE(3)$

$s = (q_1, q_2, \cdots, q_M) \in$ Joint space

Generate splines by calling:

$[s, \dot{s}, \ddot{s}] = \text{mtraj}(@\text{tpoly}, s_0, s_T, N)$

$[\cdot \cdot \cdot \cdot \cdot] = \text{mtraj}(@\text{lspd}, \cdot \cdot \cdot \cdot)$

must be row vectors

To see results:

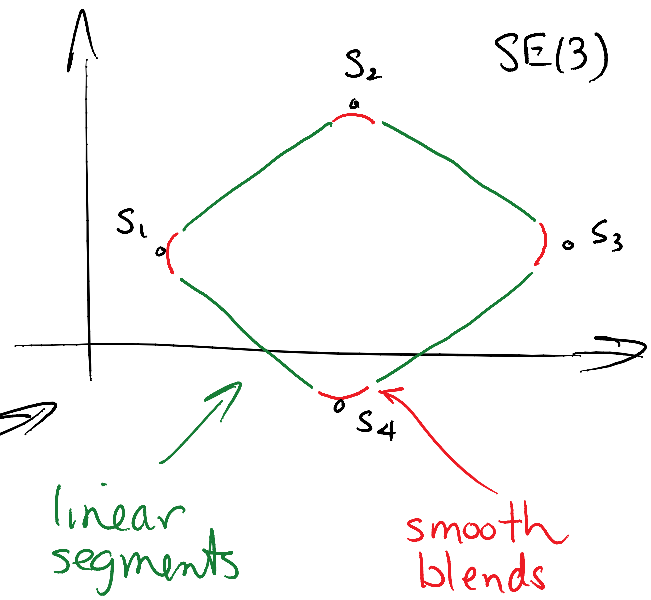
figure; plot(s);

figure; plot($\dot{s}$);

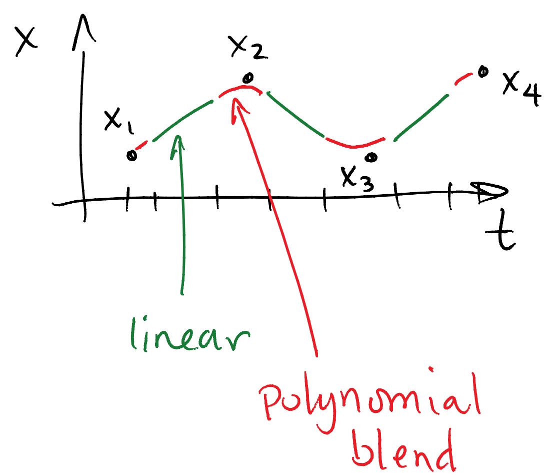
figure; plot(norm(s)); etc.

3.1.3 Multi-Segment Trajectories

Follow a curve in SE(3) or joint space that passes through or near several "via points."



Consider the case of a single dimension.



Assume constant speed $\dot{x}$ along green segments

Assume the red curve blend smoothly between the linear segments.

We need more assumptions to construct a unique spline.

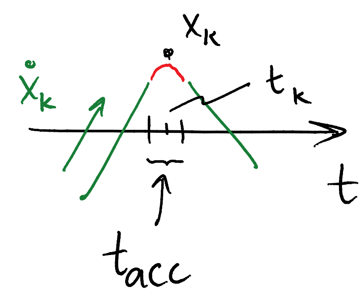
Assumptions:

speeds are constant along linear segments

$\dot{x}_1 = \dot{x}_N = 0$

$\dot{x}_2, \cdots, \dot{x}_{N-1}$ are given

t_{acc} = acceleration time interval selected by the user.



The assumptions impose constraints on the blend curve.

$s(t_k - t_{acc}/2)$ = end position of linear segment k

$s(t_k + t_{acc}/2)$ = start " " " " $k+1$

$\dot{s}(t_k - t_{acc}/2) = \dot{x}_k$

$\dot{s}(t_k + t_{acc}/2) = \dot{x}_{k+1}$

$\ddot{s}(t_k - t_{acc}/2) = 0$

$\ddot{s}(t_k + t_{acc}/2) = 0$

Five constraints suggest a polynomial blend with at least 6 coefficients – at least a quintic polynomial.

The toolbox uses a quintic polynomial

$$s(t) = at^5 + bt^4 + ct^3 + dt^2 + et + f \ .$$

3.1.4 Interpolation of Orientation in 3D.

$$R(t) = tR(0) + (1-t)R(1) \ ; \quad 0 \le t \le 1$$

This does not work, because $R(t) \ni SO(3)$!

What about interpolating roll, pitch, yaw or Euler angles?

The approach discussed thus far is:

$$[r_1, p_1, y_1] \xrightarrow[\text{lspb}]{\text{mtraj}} [r_2, p_2, y_2] \Rightarrow R(t)$$

where $R(0) = \text{rpy2r}(r_1, p_1, y_1)$

$R(1) = \text{rpy2r}(r_2, p_2, y_2)$

Problem:

Rotation axis changes over the interpolation,

- Uncoordinated looking

∘ ⇒ changing angular velocity over time

Quaternions to the rescue!

Slerp$(\mathring{q}_0, \mathring{q}_f)$ → a great circle on S^3 in $\mathbb{R}^4$

$$\mathring{q}(t) = \frac{\sin((1-t)\Omega)}{\sin(\Omega)}\mathring{q}_0 + \frac{\sin(t\Omega)}{\sin(\Omega)}\mathring{q}_f$$

$$\|\mathring{q}(t)\| = 1$$

$$\Omega = \cos^{-1}(\mathring{q}_0 \cdot \mathring{q}_f)$$

where $\mathring{q}$ is treated as an element of $\mathbb{R}^4$

and "·" is the usual dot product

See wikipedia page (google "geometric slerp")

Toolbox example:

$$\left.\begin{array}{l} \text{q0 = Quaternion(rotx}(\pi)) \\ \text{q1 = Quaternion(roty}(\pi/2) * \text{rotz}(-\pi/2)) \\ \text{q = interp(q0, q1, [0:49]'/49)} \end{array}\right\} (*)$$

These calls yield 50 points linearly interpolated and equally spaced on a great circle on S^3 embedded in $\mathbb{R}^4$.

Notes:

- In 3D space, the rotation axis is fixed!
- If orientation curves pass through singularities $\frac{d}{dt}$(orient parameters) $\longrightarrow \infty$

  (The right end of figures 3.6b and 3.7b exhibit this problem since yaw $= \pi/2$)

  Experiment a bit with this. Change 49 to 300.
  and change $\pi/2$ to $\pi/2 + 0.2$
  and change $-\pi/2$ to $-\pi/2 - 0.2$

- Trajectories produced as done by (*) will have discontinuous velocity and accel. at the ends.

One possible solution

trinterp can accept normalized segment intervals

Rs = trinterp(R0, R1, lspb(0, 1, 50))

3.1.5 Cartesian Motion

Simply put orientation & translation together.

Ts = trinterp(T0, T1, lspb(0, 1, 50))

Ts = ctraj(T0, T1, 50)