3.1.1_SmoothOneDimensionalTrajectories

Wednesday, May 30, 2012
10:06 AM

3.1 Trajectories

Curves in space from one pose to another.

Position & orientation vary smoothly over time.

3.1.1 One-dimensional trajectories

Pose, velocity, and acceleration should be continuous.

Quintic polynomial:

$S(t) = At^5 + Bt^4 + Ct^3 + Dt^2 + Et + F$ ← position

$\frac{d}{dt}S = \dot{S}(t) = 5At^4 + 4Bt^3 + 3Ct^2 + 2Dt + E$ ← velocity

$\ddot{S}(t) = 20At^3 + 12Bt^2 + 6Ct + 2D$ ← acceleration

A robot motion planning algorithm can produce a sequence of positions for the robot to follow and possibly the desired velocity and acceleration at these points. Consider one pair of consecutive points, $s(0)$ & $s(T)$.

Let $t \in [0, T]$.

Choose $s_0, s_T, \dot{s}_0, \dot{s}_T, \ddot{s}_0, \ddot{s}_T$

Solve for coefficients

$$\begin{bmatrix} s_0 \\ s_T \\ \dot{s}_0 \\ \dot{s}_T \\ \ddot{s}_0 \\ \ddot{s}_T \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ T^5 & T^4 & T^3 & T^2 & T & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 20T^3 & 12T^2 & 6T & 2 & 0 & 0 \end{bmatrix}}_{\text{non-singular}} \begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \end{bmatrix} \qquad \text{(pg. 44)}$$

Toolbox function

$s = \text{tpoly}(s_0, s_T, n)$ ← number of points returned

$\boxed{\text{see figure 3.1, pg 44.}}$

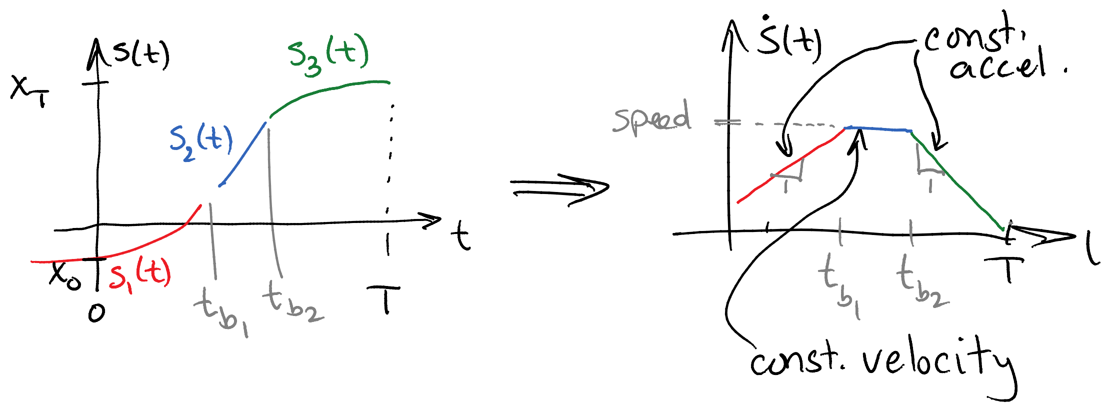
Problems:

① speed is not high for long, so motions of robot will be slow.

② position can overshoot (top plot of fig 3.1b)

One solution:

Linear segments w/ parabolic blends



$\boxed{\text{Define } s(t):}$

$$\left\{\begin{array}{lll} s_1(t) = a_1 t^2 + b_1 t + c_1 & & 0 \le t \le t_{b_1} \\ s_2(t) = \qquad\quad b_2 t + c_2 & & t_{b_1} \le t \le t_{b_2} \\ s_3(t) = a_3 t^2 + b_3 t + c_3 & & t_{b_2} \le t \le T \end{array}\right\} \text{ 11 parameters}$$

Note: To shift the initial time to t_0, simply replace t with $t - t_0$ on the right side of the equal signs.

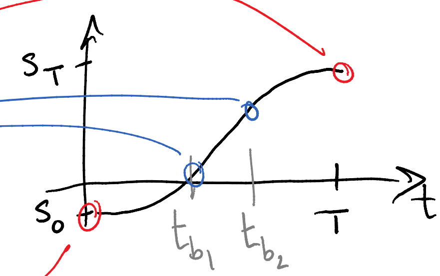
Trajectory design

Position continuity constraints:

$$s_1(0) = s_0 \leftarrow \text{given value}$$

$$\left.\begin{aligned} s_1(t_{b_1}) &= s_2(t_{b_1}) \\ s_2(t_{b_2}) &= s_3(t_{b_2}) \end{aligned}\right\}$$

$$s_3(T) = s_T \leftarrow \text{given value}$$



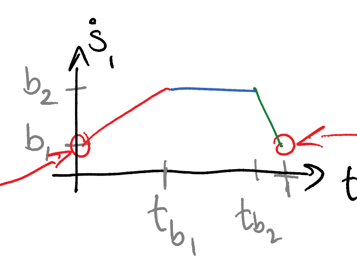
Velocity continuity constraints:

$$\dot{s}_1(0) = v_0 \leftarrow \text{given}$$

$$\dot{s}_1(t_{b_1}) = \dot{s}_2(t_{b_1})$$

$$\dot{s}_2(t_{b_2}) = \dot{s}_3(t_{b_2})$$

$$\dot{s}_3(T) = v_T \leftarrow \text{given}$$



Eight constraint equations:

$$\left.\begin{aligned}
c_1 &= s_0 \\
a_1 t_{b_1}^2 + b_1 t_{b_1} + c_1 - b_2 t_{b_1} - c_2 &= 0 \\
a_3 t_{b_2}^2 + b_3 t_{b_2} + c_3 - b_2 t_{b_2} - c_2 &= 0 \\
a_3 T^2 + b_3 T + c_3 &= s_T
\end{aligned}\right\}\ \text{Position}$$

Nonlinear in the unknowns

$$\left.\begin{aligned}
b_1 &= v_0 \\
2a_1 t_{b_1} + b_1 - b_2 &= 0 \\
2a_3 t_{b_2} + b_3 - b_2 &= 0 \\
2a_3 T + b_3 &= v_T
\end{aligned}\right\}\ \text{velocity}$$

Other constraints:

$$\left.\begin{aligned}
|\ddot{s}_1| &\le a_{max} \geq 0 \\
|\ddot{s}_3| &\le a_{max} \geq 0 \\
|\dot{s}_2| &\le v_{max} \geq 0
\end{aligned}\right\}\ \text{robot limitations. } a_{max}, v_{max} \text{ given}$$

$$0 \le t_{b_1} \le t_{b_2} \le T$$

One possible solution: solve for t_{b_1}, t_{b_2}, T first. Then remaining equations are linear!

Find minimum execution time.

$$0 \xrightarrow{a_{max}} t_{b_1} \xrightarrow{v_{max}} t_{b_2} \xrightarrow{-a_{max}} T$$

Assume: $-a_{max} \le \text{acceleration} \le a_{max}$; $a_{max} > 0$

$-v_{max} \le \text{speed} \le v_{max}$; $v_{max} > 0$

Minimum time implies:

$\ddot{s} = a_{max}$, $0 \le t \le t_{b_1}$

$\dot{s} = v_{max}$, $t_{b_1} \le t \le t_{b_2}$

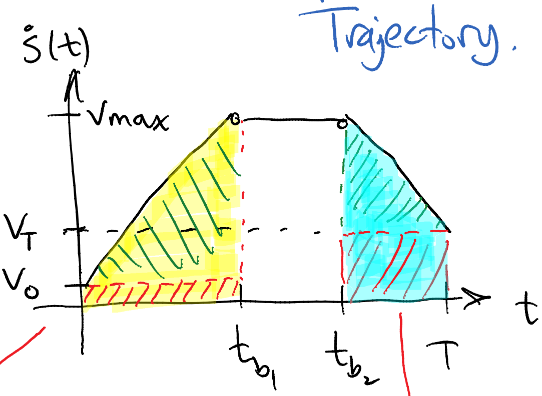
$\ddot{s} = -a_{max}$, $t_{b_2} \le t \le T$

Also assume $|v_0|, |v_T| \le v_{max}$

$$t_{b_1} = \frac{V_{max} - V_0}{a_{max}} \quad (1)$$

$$T - t_{b_2} = \frac{V_{max} - V_T}{a_{max}} \quad (2)$$

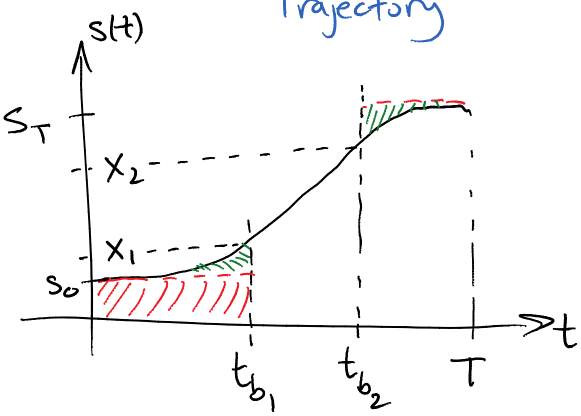
Minimum-Time Velocity Trajectory.



$$\boxed{x_1 - s_0 = V_0 t_{b_1} + \frac{1}{2} a_{max} t_{b_1}^2} \quad (3)$$

$$\boxed{s_T - x_2 = V_T (T - t_{b_2}) \cdots + \frac{1}{2} a_{max} (T - t_{b_2})^2} \quad (4)$$

Minimum-Time Position Trajectory



Similarly, the total displacement change equation is:

$$s_T - s_0 = V_0 t_{b_1} + \frac{1}{2} a_{max} t_{b_1}^2 + V_{max}(t_{b_2} - t_{b_1}) + V_T (T - t_{b_2}) + \frac{1}{2} a_{max} (T - t_{b_2})^2 \quad (5)$$

The minimum distance will achieving V_{max} occurs when $t_{b_2} = t_{b_1}$

Using eqs. (1) & (2) above, T and t_{b_1} are known:

$$T = \frac{V_{max} - V_T - V_0}{a_{max}} \qquad t_{b_1} = t_{b_2} = \frac{V_{max} - V_0}{a_{max}}$$

Substitute into the $\underbrace{\text{right-hand side of eq. (5)}}_{\text{RHS}}$

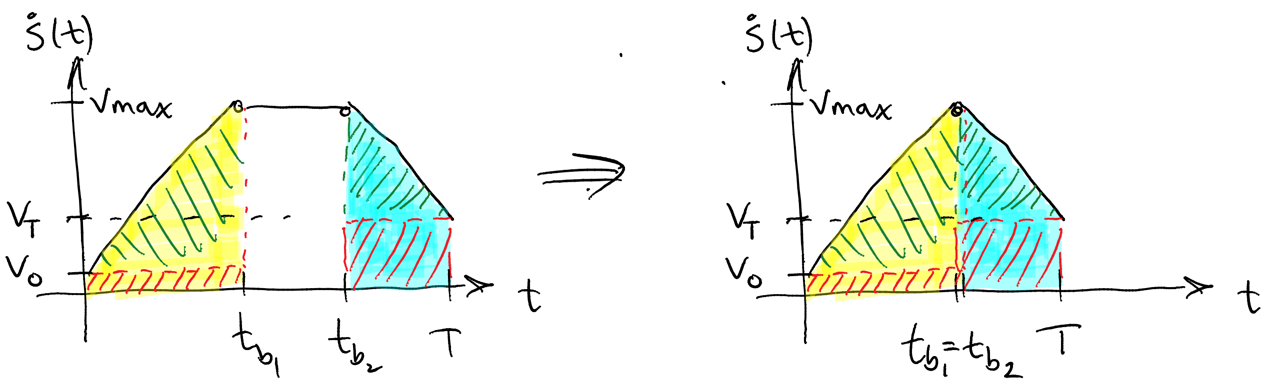
Two cases:

Case A: If $s_T - s_0 \geq RHS$, then more distance must be covered.

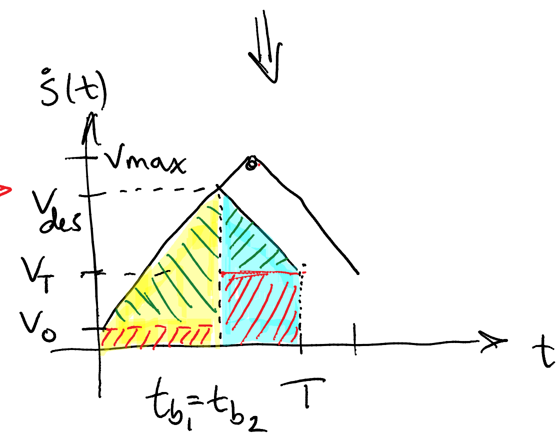
$$\therefore \quad t_{b_2} = t_{b_1} + \frac{s_T - s_0 - RHS}{V_{max}}$$

Continue to (A) below

Case B: If $s_T - s_0 < RHS$, then must reduce distance



Reduce maximum speed to reduce distance



(6) $$t_{b_1} = \frac{V_{des} - V_0}{a_{max}}$$

(7) $$T - t_{b_1} = \frac{V_{des} - V_T}{a_{max}}$$

Substitute equations ⑥ and ⑦ into equation ⑤

$$s_T - s_0 = v_0\left(\frac{v_{des} - v_0}{a_{max}}\right) + \frac{1}{2}a_{max}\left(\frac{v_{des} - v_0}{a_{max}}\right)^2$$

$$+ v_T\left(\frac{v_{des} - v_T}{a_{max}}\right) + \frac{1}{2}a_{max}\left(\frac{v_{des} - v_T}{a_{max}}\right)^2$$

This is quadratic in the only unknown, v_{des}.

Solve for v_{des}. Keep physically meaningful solution.

Substitute v_{des} into ⑥ to find t_{b_1}

" " and t_{b_1} into ⑦ to find T.

Continue to Ⓐ below

End of cases

Ⓐ The final steps are to solve for the unknown coeffs.: $a_1, b_1, c_1, b_2, c_2, a_3, b_3$, and c_3

Now use the 8 position and velocity constraints:

$$b_1 = v_0$$

$$c_1 = s_0$$

Finally solve the following equation:

$$\begin{bmatrix} t_{b_1}^2 & -t_{b_1} & -1 & 0 & 0 & 0 \\ 0 & -t_{b_2} & -1 & t_{b_2}^2 & t_{b_2} & 1 \\ 0 & 0 & 0 & T^2 & T & 1 \\ 2t_{b_1} & -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 2t_b^2 & 1 & 0 \\ 0 & 0 & 0 & 2T & 1 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ b_2 \\ c_2 \\ a_3 \\ b_3 \\ c_3 \end{bmatrix} = \begin{bmatrix} -c_1 - b_1 t_{b_1} \\ 0 \\ s_T \\ -b_1 \\ 0 \\ v_T \end{bmatrix}$$

Toolbox functions:

tpoly(s_0, s_T, n) where n is the # of points

lspb(s_0, s_T, n)