Administrivia

• HW7 due Friday
• Review lecture on Thursday
• Exam 2 from 3/27-3/29
  • Remember to sign-up on the CBTF website!
  • No class on Tuesday
  • No office hours
• Guest Lecture 2 on 3/30
  • Reflection due following Friday
Meet Shakey the Robot:
An Experiment in Robot Planning and Learning

1. An operator types the command "push the block off the platform" at a computer console.
2. Shakey looks around, identifies a platform with a block on it, and locates a ramp in order to reach the platform.
3. Shakey then pushes the ramp over to the platform, rolls up the ramp onto the platform, and pushes the block off the platform.
Control Paradigm

desired behavior (robot position)

controller

actuators and transmissions

dynamics/kinematics of robot and env.

sensors

motions and forces

$\Theta_d, X_d$

$f(\theta)$

$T_{sv}(\theta)$
Trajectories and Paths

• The specification of a robot state as a function of time is called a trajectory.
  \[ \Theta : [0, T] \rightarrow \mathbb{R}^n \rightarrow \Theta(t) \text{ gives us joint angles at time } t \]

• Using forward kinematic maps, we can obtain the position of each link given as joint angles.
  • The trajectory of the end-effector is then \( T_{sb}(\Theta(t)) \)

• A path is a set of points.
Normalized Trajectories

- **Path** $\theta(s)$ maps a scalar path parameter $s \in [0,1]$ to a point in the robot's configuration space:

  $\theta : [0, 1] \rightarrow \Theta$

  $\theta(0)$ is start of path // $\theta(1)$ is end of path

- **A time-scaling** $s(t)$ is a monotonically increasing function:

  $s : [0, T] \rightarrow [0, 1]$

  $\rightarrow$ path + time scaling def trajectory

  $\Theta(t) \quad \rightarrow \quad \Theta(s(t))$
Straight-Line Paths

\( \Theta_{\text{start}} \quad \Theta_{\text{end}} \)

• Given \( \theta_0 \) and \( \theta_1 \), find straight-line path:
Straight-Line Paths

• Given $\theta_0$ and $\theta_1$, find straight-line path:
  \[ \Theta(s) = \Theta_0 + s(\Theta_1 - \Theta_0) \]
  \[ s \in [0, 1] \]

• Is this in the task or configuration space?
  • Straight lines in joint space do not lead to straight lines in end-effector/task space

• Straight line in task space:
  \[ x(s) = x_0 + s(x_1 - x_0) \]
  \[ s \in SE(3) \]
Straight-line Paths

$\Theta_{i, \text{min}} \leq \Theta \leq \Theta_{i, \text{max}}$
Straight-line Paths

\[\theta_2 \text{ (deg)}\]

\[\theta_1, \theta_2 \text{ (deg)}\]

"straight" in joint space
Straight-line Paths

"straight" in task (or tool) space
Straight lines in SE(3)

In $\mathbb{R}^2$, straight lines are characterized by a constant velocity

$$v(t) = v_0 + v(t) \implies \dot{y} = v$$

Recall: $\dot{T} = T[S]$

If $S$ is constant, then:

$$T(t) = T_0 e^{S t}$$

Given $X_0$ and $X_1$, a "straight" line in $SE(3)$ is:

$$X(s) = X_0 e^{\log(X_0^{-1}X_1)s}$$

$$\implies X_{\text{start, end}} = X_{\text{start, SE(3)}} \cdot X_{\text{SE(3), end}} = X_0^{-1}X_1$$

Use this to define "straight" in $SE(3)$
Straight lines in SE(3)

• We can decouple rotation and translation:

\[
X = (R, p) \rightarrow p(s) = p_0 + s(p_i - p_0) \\
R(s) = R_0 e^{\log(R_0^T R_i)s}
\]

\(\rightarrow \text{now pass to IK}\)
Time-scaling of straight-line paths

- Time scaling ensures that the motion is smooth and constraints are met

\[ \Theta(s) = \Theta_0 + s(t)(\Theta_1 - \Theta_0) \]

\[ \frac{d\Theta(t)}{dt} \Rightarrow \dot{\Theta}(s) = \frac{ds}{dt}(\Theta_1 - \Theta_0) \]

\[ \ddot{\Theta}(s) = \frac{d^2s}{dt^2}(\Theta_1 - \Theta_0) \]

Choose time-scaling \( s(t) \) to limit \( \dot{\Theta} \) and \( \ddot{\Theta} \)

\[ \rightarrow \text{often use parametric form of } s(t) \text{ like polynomial} \]
Polynomial Time-Scaling (1)

Let \( s(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \) (1)

Point-to-point motion in time \( T \) imposes constraints:

\[
\begin{align*}
\dot{s}(0) &= \dot{s}(0) = 0 \quad \text{and} \quad \ddot{s}(T) = 0 \\
\dddot{s}(0) &= \dddot{s}(0) = 0 \\
\end{align*}
\]

\( \Rightarrow \dddot{s}(t) = a_1 + 2a_2 t + 3a_3 t^2 \) (2)

Evaluate (1) and (2) \( \text{at } t = 0 \) and \( t = T, \) solve for

\[
\begin{align*}
a_0 &= 0, \quad a_1 = 0, \quad a_2 = \frac{3}{T^2}, \quad a_3 = -\frac{2}{T^3}
\end{align*}
\]

\( \Rightarrow s(t) = \frac{3t^2}{T^2} - \frac{2t^3}{T^3} \)
Polynomial Time-Scaling (2)

1. \( \Theta(t) = \Theta_0 + \left(3\frac{t^2}{T^2} - 2\frac{t^3}{T^3}\right)(\Theta_1 - \Theta_0) \)

2. \( \dot{\Theta}(t) = \left(6\frac{t}{T^2} - 6\frac{t^2}{T^3}\right)(\Theta_1 - \Theta_0) \)

3. \( \ddot{\Theta}(t) = \left(6\frac{1}{T^2} - 12\frac{t}{T^3}\right)(\Theta_1 - \Theta_0) \)

Evaluate (2) to find that max vel is at \( t = \frac{T}{2} \)

\[ \dot{\Theta}_{max} = \frac{3}{2T}(\Theta_1 - \Theta_0) \]

Evaluate (3) to find that max acc is at \( t = 0, t = T \)

\[ \ddot{\Theta}_{max} = \pm \frac{6}{T^2}(\Theta_1 - \Theta_0) \]

Choose \( T \) to meet joint limits on \( \dot{\Theta} \) and \( \ddot{\Theta} \).
Summary

• Defined **paths, time-scaling, and trajectories**
• Looked at how to find **straight-line paths** in various spaces
• We choose a **parametrization** $s(t)$, and computed the resulting velocity and acceleration profiles of the trajectory
  • Using a third-order polynomial, we tuned their maximal values to meet requirements with one parameter $T$
Summary

• Defined **paths, time-scaling, and trajectories**

• Looked at how to find **straight-line paths** in various spaces

• We choose a **parametrization** $s(t)$, and computed the resulting velocity and acceleration profiles of the trajectory
  • Using a third-order polynomial, we tuned their maximal values to meet requirements with one parameter $T$

• We can follow the same procedure with different parametrizations for $s(t)$ (e.g., polynomials of order 5, trapezoidal functions, splines, etc.)
  • Having more parameters allows us to meet more constraints. For example, using a fifth order polynomial, we can ensure that $\ddot{\theta}(0) = \ddot{\theta}(T) = 0$, meaning no jerk at beginning and end of the motion

• **Next topics** are on different concepts of / approaches to planning when the path may not be given