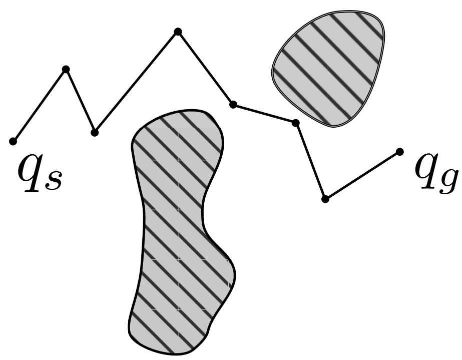
CS 294-115 Algorithmic Human-Robot Interaction **Fall 2015**

Lecture 3: Trajectory Optimization

Scribes: Yan Duan, Chelsea Finn

3.1 Wrap up/Review: RRT & RRT*

3.1.1 Bi-Directional RRT



The bi-directional RRT algorithm grows two trees towards each other, iteratively following these two steps until a solution is found:

1. Sample a configuration in C_{free}.
2. Run simple planner - try to connect sample to closest node in each tree.

Pros/cons:

- (+) Scales well in high dimensions
- (-) Not optimal, but short-cutting post-processing improves result: pick two nodes at random and try to connect them
- (-) Not complete, but probabilistically complete: $\lim_{t\to\infty} P(sol \mid \exists\ sol) = 1$. *Note: This is a weak property.*

RRT is single tree (unidirectional) version, where you sample the goal with some probability as opposed to always sampling a random configuration.

3.1.2 RRT*

A form of optimal motion planning. Differs from RRT in two ways.

- Parent selection, using cost-to-come
- Rewiring step

Pros/cons:

- (+) Asymptotically optimal
- (-) Sacrifices speed, considers optimal path to everything

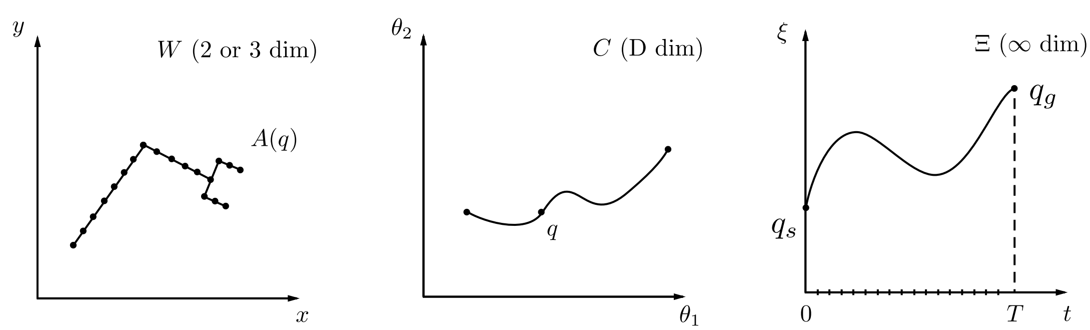
3.2 Trajectory Optimization Outline

1. Problem Statement
2. (Functional) Gradient Descent
3. CHOMP [Next lecture] – See reading for a comparison between CHOMP, BiRRT, and RRT*.

3.3 Problem Statement

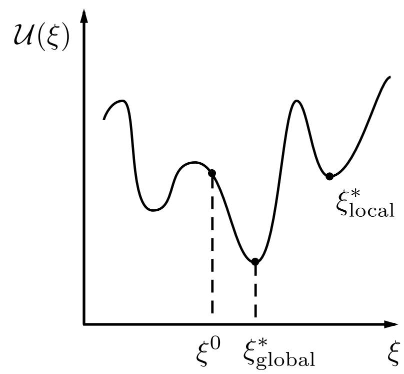
Notation:

- trajectory $\xi : [0, T] \to C$ - maps time to configurations, infinite dimensional
- $\mathcal{U} : \Xi \to \mathbb{R}^+$, functional that maps functions to scalars
- objective $\mathcal{U}$ can encode aspects like path length, efficiency, obstacle avoidance, legibility, uncertainty reduction (achieve goal with high probability), comfort of human



Optimization Problem:

$$
\begin{aligned}
\xi^* = \underset{\xi \in \Xi}{\operatorname{argmin}} \quad & \mathcal{U}(\xi) \\
\text{subject to} \quad & \xi(0) = q_s, \\
& \xi(T) = q_g
\end{aligned}
$$



We can solve this problem when it is convex, however it is most often non-convex (e.g. due to obstacles). At local minima, the trajectory might be inefficient, have collisions, scare people, etc.

Often: find ξ^* s.t. $\mathcal{U}(\xi^*) \leq \beta$.

3.4 Functional Gradient Descent

$$\xi_{i+1} = \xi_i - \frac{1}{\alpha}\nabla\xi_i(U)$$

Thm: If

$$\mathcal{U}[\xi] = \int_0^T F(t, \xi(t), \xi'(t))dt$$

and using the euclidean inner product (see following section), then you can prove that:

$$\nabla_\xi \mathcal{U} = \frac{\partial F}{\partial \xi(t)} - \frac{d}{dt}\frac{\partial F}{\partial \xi'(t)}$$

This has a similar derivation to the Euler-Lagrange formula from calculus of variations. We will go through the derivation in the next lecture.

Example: Consider the example where you minimize the squared norm of velocity subject to starting at q_s and ending at q_g:

$$\mathcal{U}[\xi] = \frac{1}{2}\int_0^T \|\xi'(t)\|^2 dt$$

Since this objective is quadratic, we can show that the optimal solution (when the gradient is 0) corresponds to a straight line path of constant velocity. Using the above theorem:

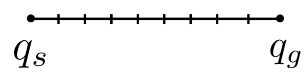
$$\nabla_\xi \mathcal{U} = 0 - \frac{d}{dt}\xi'(t) \tag{3.1}$$

$$= -\xi''(t) = 0 \tag{3.2}$$

$$\xi'(t) = a \tag{3.3}$$

$$\xi(t) = at + b \tag{3.4}$$

To solve for a and b, one can simply plug in the boundary conditions that $\xi(0) = q_s$ and $\xi(T) = q_g$. The resulting trajectory is the following, a straight line with constant velocity:



3.4.1 Hilbert space & inner products

Ξ is a Hilbert space, a complete vector space with an inner product.

An inner product $\langle \xi_1, \xi_2 \rangle$ is defined to have the following properties

- **symmetry**: $\langle \xi_1, \xi_2 \rangle = \langle \xi_2, \xi_1 \rangle$
- **positive definite**: $\langle \xi_1, \xi_1 \rangle \geq 0$, and $\langle \xi_1, \xi_1 \rangle = 0 \iff \xi_1 = 0$
- **linearity** in the first argument: $\langle a\xi_1, \xi_2 \rangle = a\langle \xi_1, \xi_2 \rangle$ and $\langle \xi_1 + \xi_2, \xi_3 \rangle = \langle \xi_1, \xi_3 \rangle + \langle \xi_2, \xi_3 \rangle$ (the same holds for the second argument by symmetry)

The Euclidean inner product is defined as

$$\langle \xi_1, \xi_2 \rangle = \int_0^T \xi_1(t)^T \xi_2(t) dt$$

In the more familiar vector form, this becomes $\xi_1^T \xi_2$.