Optimal Kinodynamic Motion Planning using Incremental Sampling-based Methods

(by Sertac Karaman & Emilio Frazzoli)

★ Problem Definition

⇒ Let $X \subset \mathbb{R}^n$ & $U \subset \mathbb{R}^m$ be compact set and consider the dynamical system

$$\boxed{\dot{x}(t) = f(x(t), u(t))} \quad \text{—— (1)}$$

$$x(0) = x_0$$

where, $x(t) \in X \subseteq \mathbb{R}^d$ $\forall t$,
$u(t) \in U \subseteq \mathbb{R}^m$

and f is a smooth (continuously differentiable) function of its variables.

⇒ Let us denote the set of all essentially bounded measurable functions defined from $[0,T]$ to X, for any $T \in \mathbb{R}_{>0}$ by $\mathcal{X}$.

↳ We define $\mathcal{U}$ similarly.

⇒ The functions in $\mathcal{X}$ and $\mathcal{U}$ are called trajectories and controls respectively.

⇒ Let X_{obs} and X_{goal}, called the ~~obje~~ obstacle region and the goal region, respectively, be open subsets of X.

⇒ Let X_{free} also called the free space, denote the set defined as $X \setminus X_{obs}$.

Problem: Optimal Kinodynamic motion planning

Given the domain X, obstacle region X_{obs}, goal region X_{goal}, and a smooth function f that describes the system dynamics, find a control $u \in U$ with domain $[0,T]$ $\forall$ $T \in \mathbb{R}_{>0}$ such that the unique corresponding trajectory $x \in X$ with $\dot{x}(t) = f(x(t), u(t))$ $\forall$ $t \in [0,T]$

- → Avoids the obstacles i.e. $x(t) \in X_{free}$ $\forall$ $t \in [0,T]$
- → Reaches the goal region i.e. $x(T) \in X_{goal}$
- → and minimizes the cost functional

$$\boxed{J(x) = \int_0^T g(x(t))\,dt}$$

★ RRT* Algorithm (Dubin's Vehicle)

1. System dynamics

$$\dot{x}_D = V_D \cos(\theta_D)$$

$$\dot{y}_D = V_D \sin(\theta_D)$$

$$\dot{\theta}_D = u_D \quad |u_D| \leq \frac{V_D}{\rho}$$

2.

⇒

2. Steering procedure

⇒ Given two states $z_1, z_2 \in X$ for the Dubins vehicle, it is well known that the optimal path(s) to drive the system from z_1 to z_2 can be parameterized by six families of canonical paths.

- RSL
- LSR
- RSR
- LSL
- RLR
- LRL

{ L ⇒ Left
R ⇒ Right
S ⇒ Straight }