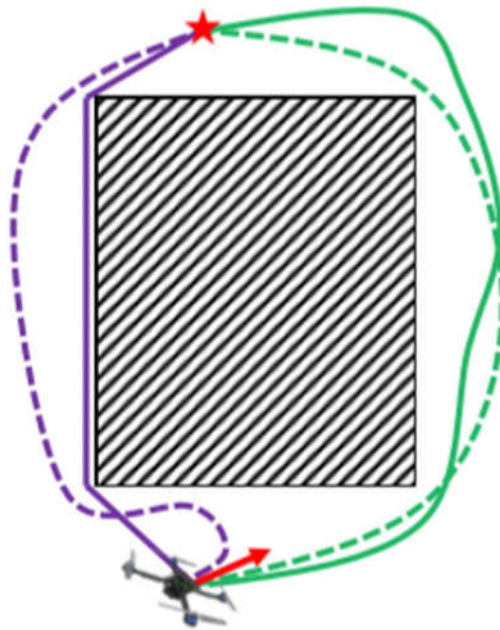


环境感知与规划专题（十一）——状态栅格规划器（State Lattice Planner）

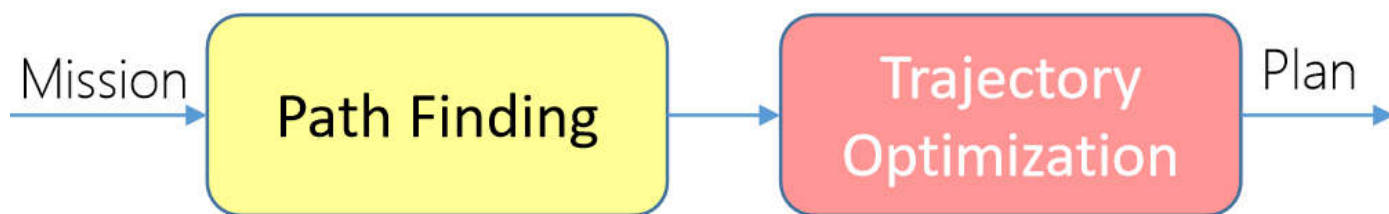
前言

前几篇中介绍了几种路径搜索算法，其中既有**基于图搜索**的路径搜索算法（如 **A***），也有**基于采样**的路径搜索算法（如**RRT**、**RRT***）。通过路径搜索算法，我们能够得到一条可通行的路径，然而，对于这条路径中的路标点（**Waypoint**）之间的路径往往是不符合机器人（无人机\无人车）动力学模型的。

因此，对于每一个路标点（**Waypoint**）之间的连接路径，我们需要进行进一步的优化，从而得到一条符合机器人（无人机\无人车）动力学约束的可通行的运动学连接（**feasible motion connections**）。



如上图所示，实线为前端路径搜索得到的可通行路径，而虚线为后端轨迹优化得到的符合动力学模型的可通行路径。经典的运动规划算法框架，往往由前端路径搜索与后端轨迹优化组成，在得到满足运动学的可通行的运动连接后，才能够将这条优化后的轨迹指令下达至控制器执行。

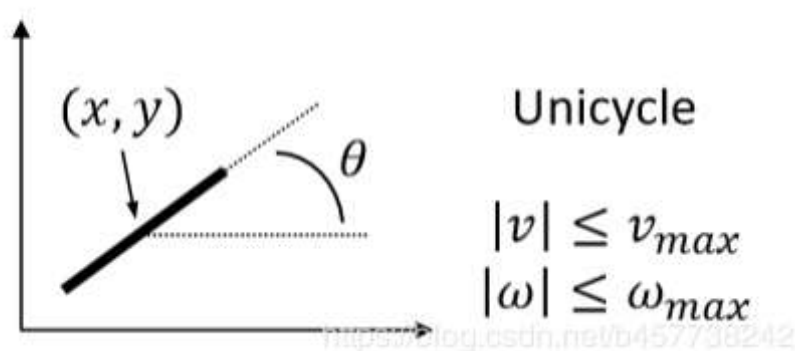


这也是本文重点讨论的基于采样的路径规划算法——状态栅格规划器（**State Lattice Planner**）。

状态栅格搜索（State Lattice Search）

在构建状态栅格之前，我们需要对被控对象进行运动学分析，即**建立对应的运动学模型**，这里以简单的无人车模型为例，如下图所示，我们可以将其简化为自行车模型：

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix} \cdot v + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \cdot \omega$$



运动学模型主要描述了位置、速度、加速度、转向角度等等状态量在时间上的状态转移，以及对于我们所关心的这些状态量的边界条件进行约束。

在得到被控对象的运动学方程后，我们可以开始正式构建状态栅格，对于**State Lattice Search**，我们通常有两种采样方式：

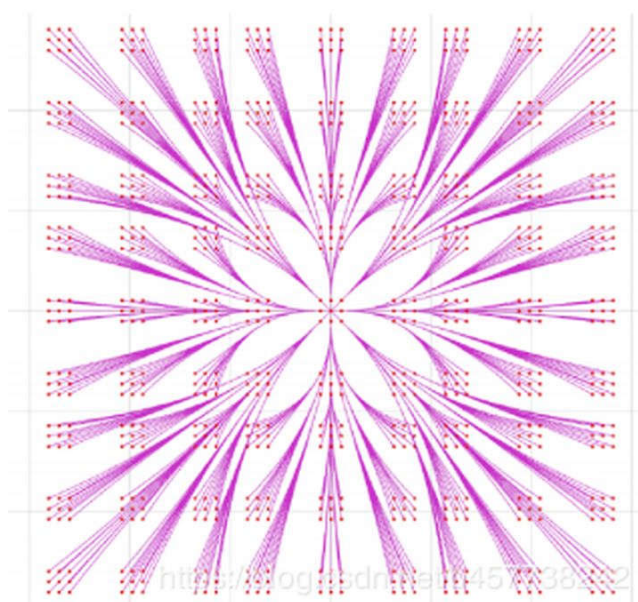
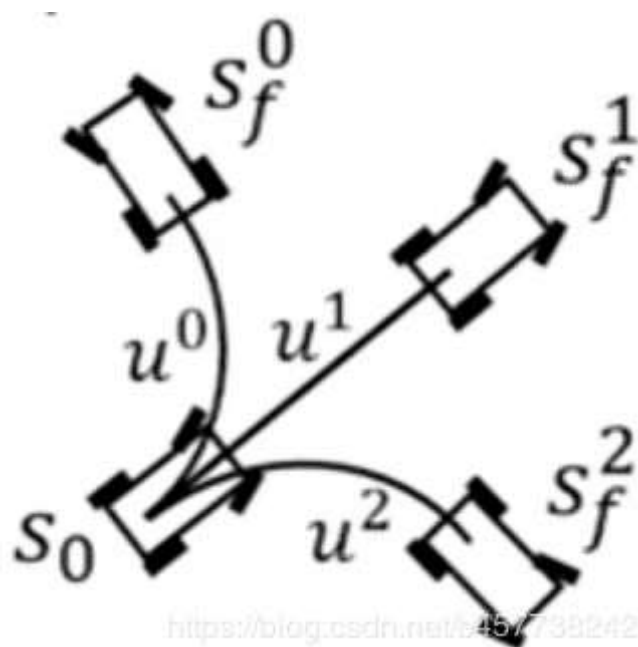
- 基于控制空间（Control Space）的采样



在**RRT***算法中，我们对空间中的位置点进行随机采样，那么，在控制空间中，我们应该如何进行采样呢？

很简单，在已知被控对象的情况下，我们固定输入量 u （控制量）与积分时间 T ，由此进行前向积分即可得到被控对象从任意初始状态 s_0 过度到的终止状态 s_f ，前向积分的方式能够简单粗暴地对被控对象在任意时间 T 之后的状态进行预测，从而得到一系列状态集合，这就是在控制空间中的采样。

由于输入量 u 与积分时间 T 都是任意给定的，因此，基于控制空间的采样往往是没有明确的目的性的，它的采样结果只能分布在某些状态附近，但不能精确地采样到某一个给定状态。



• 基于状态空间 (State Space)的采样

而另一种采样方式则是对状态空间的采样。这种采样方式则是在给定被控对象的初始以及目标状态后，反向计算中间过程，相对于前者，该采样方式在工程上较难实施，我们需要对其进行两点边界值问题的求解，才能求得两个状态之间轨迹。



两点边界值问题 (BVP)

经典的两点边界值问题，其实就是多项式方程求系数的过程，以多旋翼无人机为例，我们将无人机的状态方程表示为五阶多项式形式，那么，对其状态方程求各阶导数，即可求得位置、速度、加速度等状态量关于时间的表达式，那么，我们只需要将两点边界状态信息代入多项式方程，即可联立求得多项式系数的值，如此一来，一条可通行的轨迹就被解析化了。




- 5th order polynomial trajectory:
 - $x(t) = c_5 t^5 + c_4 t^4 + c_3 t^3 + c_2 t^2 + c_1 t + c_0$
- Boundary conditions

	Position	Velocity	Acceleration
t = 0	a	0	0
t = T	b	0	0

- Solve:

$$\begin{bmatrix} a \\ b \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ T^5 & T^4 & T^3 & T^2 & T & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 5T^4 & 4T^3 & 3T^2 & 2T & 1 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 20T^3 & 12T^2 & 6T & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} c_5 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix}$$

然而，求解**BVP**问题能够求解一条可通行轨迹，却无法求得一条最优轨迹。或者，当其边界条件部分确定时（如位置确定，速度、加速度状态不确定），需要求解**OBVP**问题。

OBVP问题可以认为是一个最优控制问题，以多旋翼无人机为例，假设，我们需要最小化无人机在飞行过程中的 *jerk*，即加加速度，那么可以有以下代价函数  [公式]：

Objective, minimize the integral of squared jerk:

$$J_{\Sigma} = \sum_{k=1}^3 J_k, \quad J_k = \frac{1}{T} \int_0^T j_k(t)^2 dt.$$

State: $s_k = (p_k, v_k, a_k)$ Input: $u_k = j_k$

System model: $\dot{s}_k = f_s(s_k, u_k) = (v_k, a_k, j_k)$

根据**Pontryain's minimum principle**，我们可以得到**Hamiltonian**函数：



$$\begin{aligned}
 H(s, u, \lambda) &= \frac{1}{T} j^2 + \lambda^T f_s(s, u) \\
 &= \frac{1}{T} j^2 + \lambda_1 v + \lambda_2 a + \lambda_3 j \\
 \dot{\lambda} &= -\nabla_s H(\boxed{s^*}, \boxed{u^*}, \lambda) = (0, -\lambda_1, -\lambda_2)
 \end{aligned}$$

Optimal state
Optimal input

式中 λ 为中间变量，根据极小值原理，可以得到在取得最优输入量 u 时的最优状态 s ：

The costate is solved as:

$$\lambda(t) = \frac{1}{T} \begin{bmatrix} -2\alpha \\ 2\alpha t + 2\beta \\ -\alpha t^2 - 2\beta t - 2\gamma \end{bmatrix}$$

The optimal input is solved as:

$$\begin{aligned}
 u^*(t) &= j^*(t) = \arg \min_{j(t)} H(s^*(t), j(t), \lambda(t)) \\
 &= \frac{1}{2} \alpha t^2 + \beta t + \gamma
 \end{aligned}$$

The optimal state trajectory is solved as:

$$s^*(t) = \begin{bmatrix} \frac{\alpha}{120} t^5 + \frac{\beta}{24} t^4 + \frac{\gamma}{6} t^3 + \frac{a_0}{2} t^2 + v_0 t + p_0 \\ \frac{\alpha}{24} t^4 + \frac{\beta}{6} t^3 + \frac{\gamma}{2} t^2 + a_0 t + v_0 \\ \frac{\alpha}{6} t^3 + \frac{\beta}{2} t^2 + \gamma t + a_0 \end{bmatrix}$$

Initial state: $s(0) = (p_0, v_0, a_0)$

同时，也能够将代价函数 J 化为关于时间 T 的函数，那么，我们对于代价函数求极小值即可求得最优的时间 T 。

$$J = \gamma^2 + \beta\gamma T + \frac{1}{3} \beta^2 T^2 + \frac{1}{3} \alpha\gamma T^2 + \frac{1}{4} \alpha\beta T^3 + \frac{1}{20} \alpha^2 T^4$$

然后，将两点边界状态量代入状态方程，即可得到 α 、 β 、 γ 的值，至此，**OBVP**问题求解完成。



$$\begin{bmatrix} \frac{1}{120}T^5 & \frac{1}{24}T^4 & \frac{1}{6}T^3 \\ \frac{1}{24}T^4 & \frac{1}{6}T^3 & \frac{1}{2}T^2 \\ \frac{1}{6}T^3 & \frac{1}{2}T^2 & T \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} \Delta p \\ \Delta v \\ \Delta a \end{bmatrix}$$

$$\begin{bmatrix} \Delta p \\ \Delta v \\ \Delta a \end{bmatrix} = \begin{bmatrix} p_f - p_0 - v_0 T - \frac{1}{2}a_0 T^2 \\ v_f - v_0 - a_0 T \\ a_f - a_0 \end{bmatrix}$$

对基于状态空间的采样方法而言，求解**OBVP**问题即可求得任意两点之间符合运动学方程的最优轨迹。

总结

本篇详细阐述了另一种基于采样的且符合被控对象运动学模型的路径规划算法——**State Lattice Planner**。

对**基于控制空间的采样方式**，我们可以很容易地在建立运动学模型的基础上，根据控制输入，对其初始状态进行一定时间 T 内的前向积分，从而得到一系列符合运动学模型的轨迹；对**基于状态空间的采样方式**，我们需要求解**OBVP**问题，有时候，甚至需要对整段轨迹的最优时间 T 进行优化。

下一篇，我们将针对**A***算法的改进型——**Hybrid A***进行讨论。

GitHub: [github.com/DistantUtopi...](https://github.com/DistantUtopia)

