

## 【自动驾驶】运动规划 | 轨迹规划 | Frenet坐标转换

### 技术背景

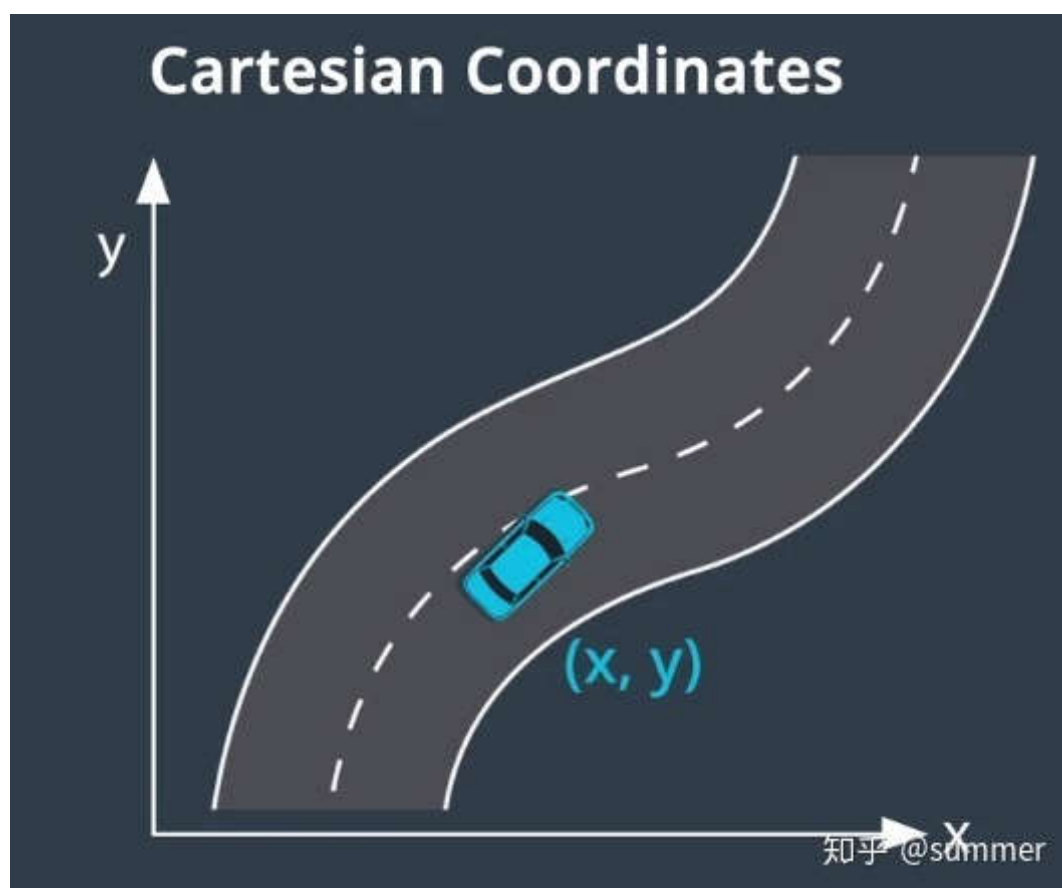
轨迹规划作为一种具有多个非线性约束的高维优化问题，需要消耗大量的计算资源，并且由于场景的复杂性和实时性，并不是总能快捷简单的得到最优轨迹，这对规划算法的鲁棒性和实用性有着较高的要求。因此，在自动驾驶开发中，如何提高算法的计算效率是一个关键点。

### 1、笛卡尔坐标系

通常使用笛卡尔坐标系描述物体的位置，但笛卡尔坐标系对车辆来说并不是最佳选择，即使给出了车辆位置  $(x, y)$ ，但也**不知道道路在哪里，也很难知道**



辆行驶了多远，也难以确定它是否偏离车道中心。



笛卡尔坐标系

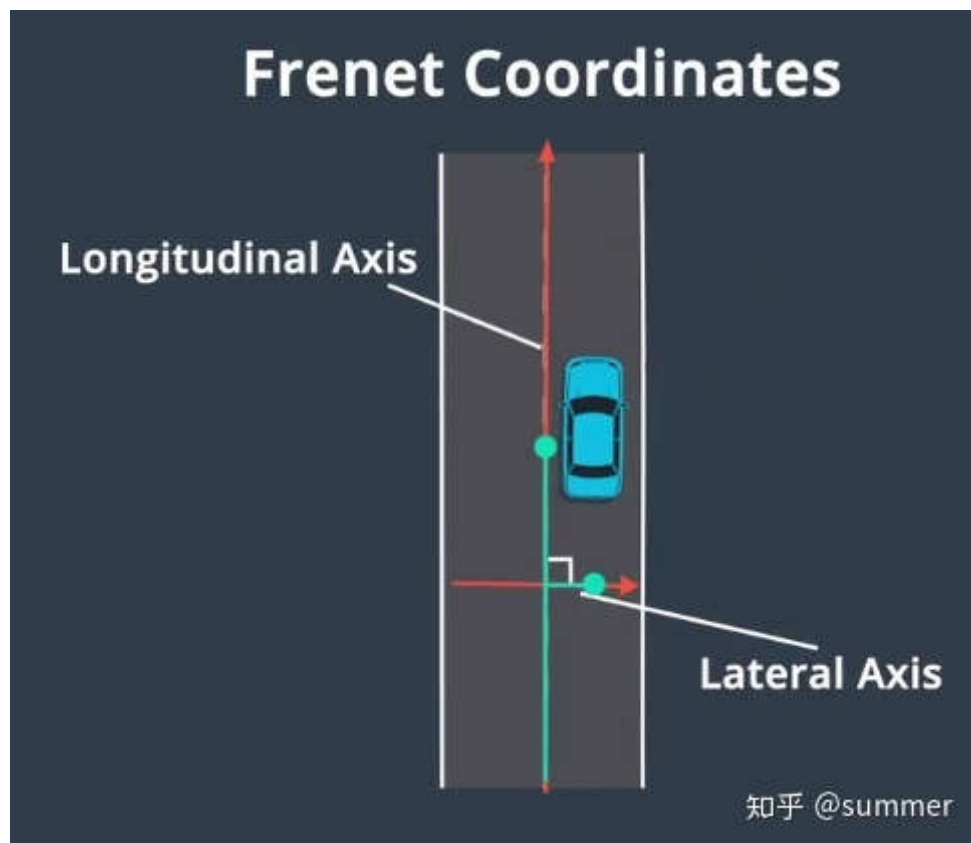
### 技术缺点：

- 1、**道路条件：**由于大部分的道路并不是笔直的，而是具有一定弯曲度的弧线；
- 2、**数据处理：**运动规划需要对HDmap获取的坐标点进行处理，数据量比较大；
- 3、**控制需求：**车辆只需知道所在车道位置以及与周边障碍物的横纵向距离进行控制。

## 2、Frenet 坐标系

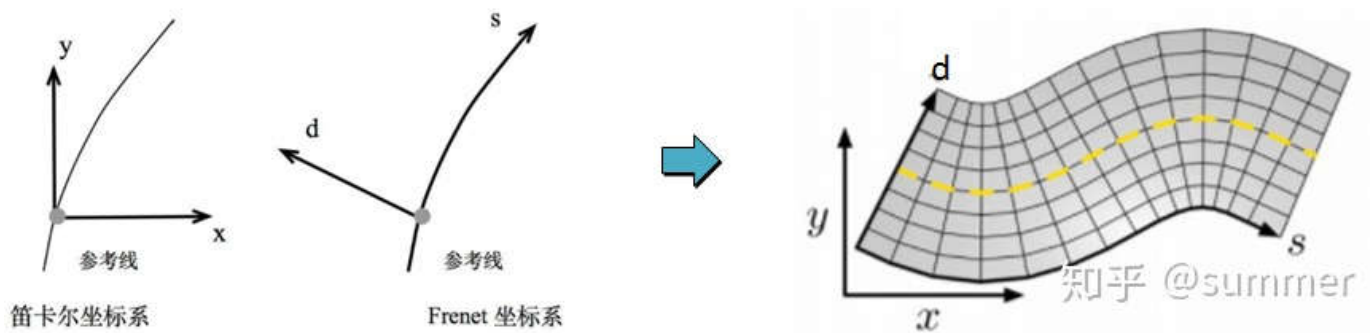
笛卡尔坐标系的替代解决方案为 Frenet 坐标系，Frenet 坐标系描述了汽车相对于道路的位置，在 Frenet 坐标系中， $s$  代表沿道路的距离称为纵坐标， $d$  表示与纵向线的位移称为横坐标。

这样保证了在道路的每个点上，横轴和纵轴都是垂直的，纵坐标表示在道路中的行驶距离，横坐标表示汽车偏离中心线的距离。



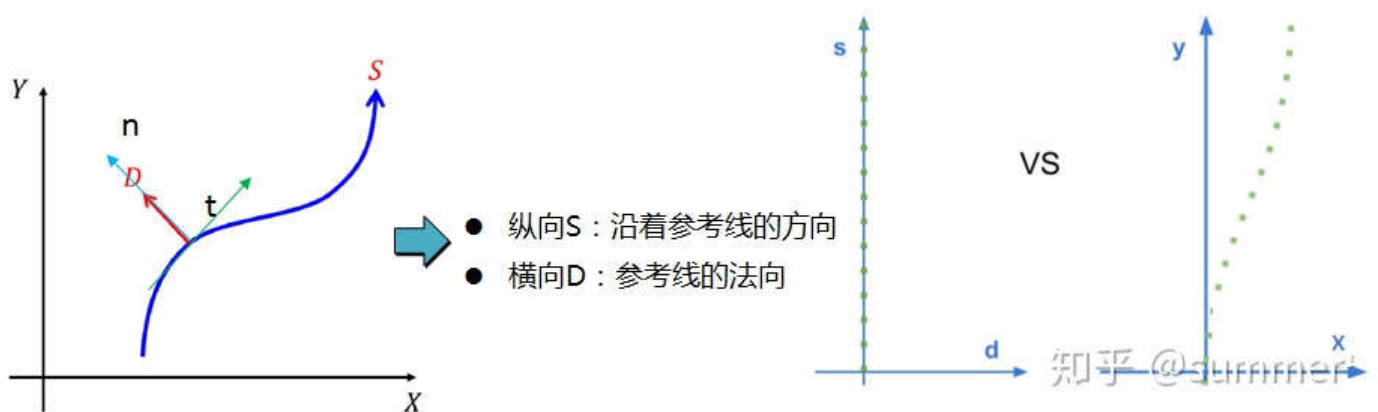
Frenet 坐标系

在Frenet坐标系中，使用道路的中心线作为参考线，使用参考线的切线向量  $t$  和法线向量  $n$  建立一个坐标系。以车辆自身为原点，坐标轴相互垂直，分为 $s$  方向（即沿着参考线的方向，通常被称为纵向，Longitudinal）和  $d$  方向（即参考线当前的法向，被称为横向，Lateral）。



通常，我们习惯使用笛卡尔坐标系来定义空间点的位置。但在现实中，道路往往不是“笔直”的，因此对于人类是非常简单的操作（如判断车辆在哪条车道），但计算机的笛卡尔坐标系中，往往是难以准确定义的。

在新的Frenet坐标系下车辆向前行驶并保持在车道内的轨迹就计算成一条直线，这会大大简化轨迹规划的难度。



S-D坐标系即为Frenet坐标系 Frenet（左）vs 笛卡尔坐标系（右）

## 技术优点：

- 1、**道路条件：**适用于自动驾驶弯道的路线规划；
- 2、**数据处理：**将地图数据降维处理，计算量减小，提高效率；
- 3、**控制需求：**Frenet坐标系明显地简化了问题，因为在车辆行驶中，道路的参考线（中心线）很容易找到；输出横纵向距离等参数给控制层。

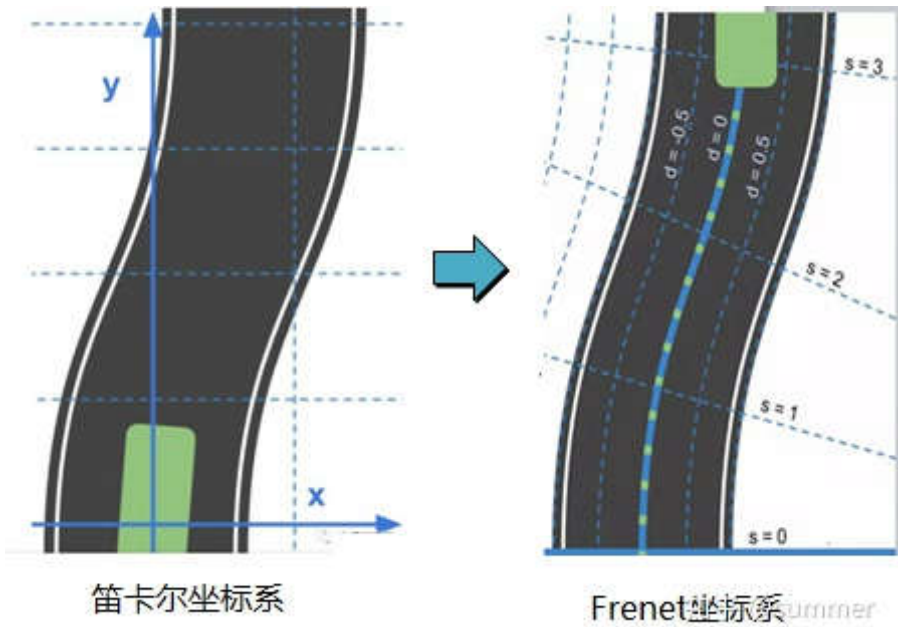
## 总结：

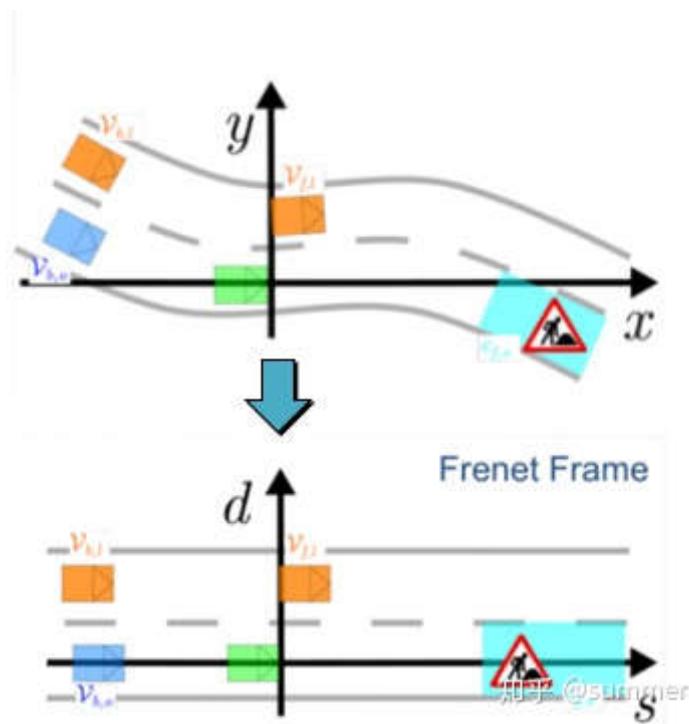


相比于笛卡尔坐标系，Frenet坐标系明显地简化了问题，因为在车辆行驶中，我们总是能够简单的找到道路的参考线（即道路的中心线），那么基于参考线的位置的表示就可以简单的使用纵向距离（即沿着道路方向的距离）和横向距离（即偏离参考线的距离）来描述。同样的，两个方向的速度( $s'$ 、 $s$ 和  $d'$ 、 $d$ )的计算也相对简单。同时纵向和横向的速度、加速度、加加速度等信息也更便于计算。

### 3、Frenet坐标系转换

在Frenet坐标系中，地面上的点的坐标位置可以由纵轴和横轴定位，分别记为S和D。其背后的数学原理非常复杂，但可以简单认为穿过道路中心的曲线决定了S轴，并指出了车辆在路上走了多远；D轴则映射为车辆的侧向位移。下图表示在弯曲道路上，笛卡尔坐标系和Frenet坐标系的转换。





## 降维处理：

$s$  沿着道路中线 (reference line) 的方向， $d$  则是垂直于道路中线的方向。于是，问题就转化成了求解  $s(t)$  和  $l(s)$  的问题，我们希望得出，自动驾驶车在  $t$  时刻应该走到  $s$  上的哪个位置；以及走在这个位置时，该车距离  $s$  的左右偏离是多大。

最终，一个复杂问题就转化成三个子优化问题：一个横向优化、纵向优化以及中线问题 ( $s$  方向是车辆的纵向规划， $d$  方向为横向规划)。具体内容会在横纵向规划中介绍。

Apollo问答 | Apollo笛卡尔坐标系转Frenet坐标系的函数如何理解？

[mp.weixin.qq.com](https://mp.weixin.qq.com)

