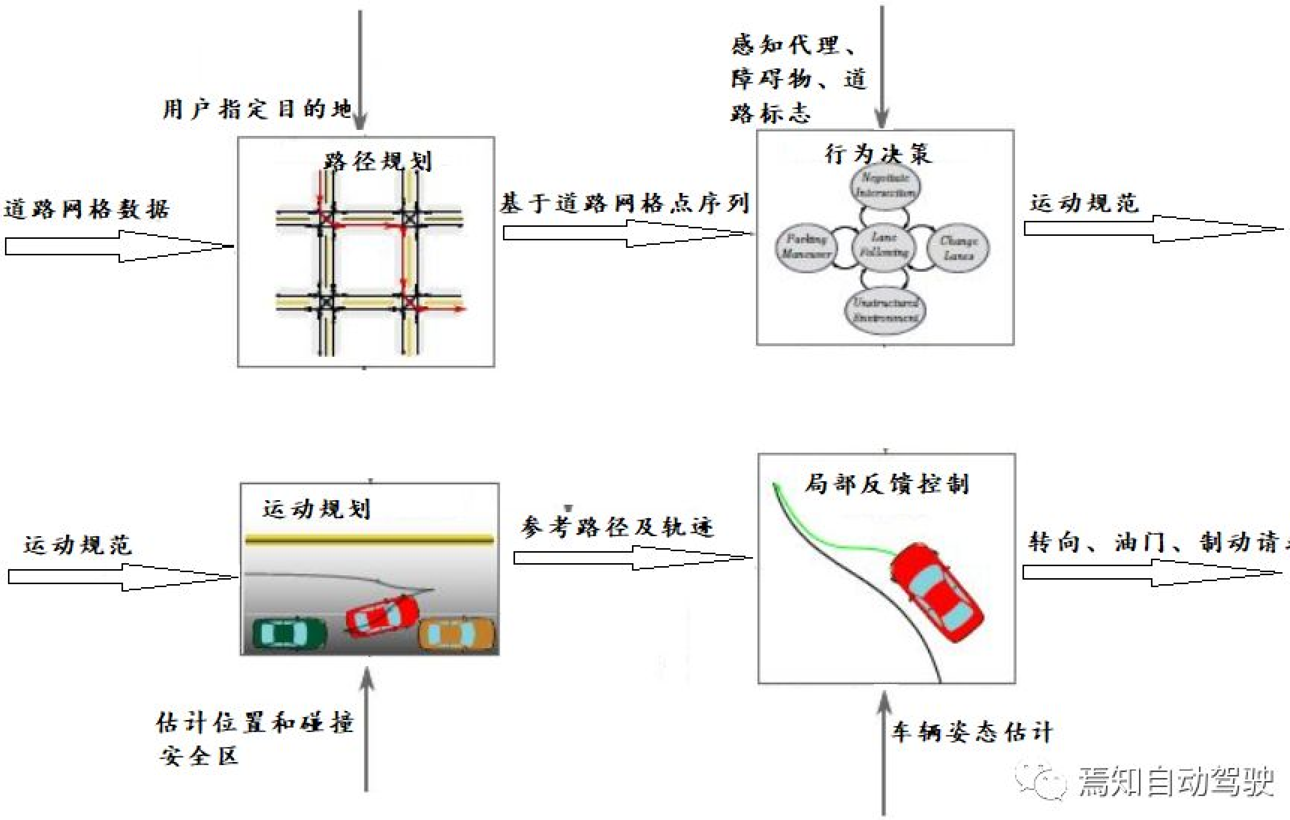
知荐 | ADAS算法设计中的“运动规划”详解（一）

前面文章具体讲解了路径规划与行为决策模块相关内容，包括如何预测轨迹、如何进行行为决策，本文将就行为决策的下层模块“运动规划”模块进行详细阐述。

运动规划的任务是将前端生成的轨迹发送到车端进行实际控制及反馈调节，而规划并不是一次性的全局规划，而是将轨迹规划的内容进行细分后的轨迹点序列，且每个点包含了位置、时间、速度、曲率等属性。自动驾驶中的运动规划可以比拟为机器人运动规划的特例，但是相比而言却更为简单，因为车辆运动往往遵循一定的道路规则且仅在二维平面上移动。而规划的结果即是一些诸如扭矩（节气门开度）、制动、驱动甚至换挡控制信号等。由于车辆控制不是一个理想化的协调系统，实际车辆轨迹表现出了样条轨迹特征。因此，运动规划可以形式化为有确定特性/约束的轨迹优化问题。



在实际响应行为预测的轨迹过程中，这一特性/约束可表示为诸如平滑度和曲率的约束，这才是“运动规划”真正的意义所在，即通过优化使路径规划和行为决策的结果能够真正应用到车辆控制执行端。

代价函数

优化问题中的两个关键要素是优化目标和约束，其原理是为各个候选解决方案选择一定的代价函数Func（i），并在参数执行过程中使得代价函数值cost最小。

代价函数的设计必须包含三个关键影响要素：

1）作为行为决策的下端模块，代价函数必须服从上层行为决策输出。以跟车为例，对于前车的跟随结果是需要与前车保持一定的车距进行跟随控制，那么运动规划的过程就是在防止碰撞的前提下，使得局部决策结果在特定的区域中保持车辆的稳定行驶。

2）运动规划也要考虑道路行驶规则，规划的结果应该与道路形状保持一致。也即在代价函数设计中，应该充分考虑到运动规划结果应该遵循道路形状进行自然移动。

3）运动规划因充分考虑反馈调节。优化问题中的约束更多的是强调整车表现上，这就需要代价函数设计过程中不仅要强调从上层行为决策输出和遵循道路方向，也要在下层执行中充分汲取执行反馈数据进行实时调节。比如，在进行转向控制过程中，需要利用曲率和曲率的二次倒数进行约束并输出至轮端，且在执行过度情况下需要重新进行约束调整，这一过程有时也被称作约束滤波。又如，在进行加速控制过程中，我们也要充分考虑发动机的响应性能（如最大最小节气门开度或扭矩限值）后规划对于发送端的扭矩/加速度发送限制问题。

2

运动过程中的时空规划

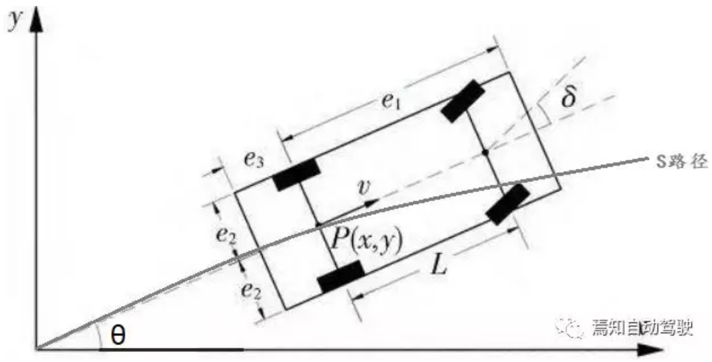
运动规划过程实际是对于车辆的时空规划问题，目前有两种基本的处理机制。

其一是一种简单的版本，它可以被分解为两个问题的有序处理：即接收路径规划数据和实行速度/加速度规划。由于前段生成的路径规划只是生成相应的轨迹形状，并未在该路径上产生任何速度/加速度信息，加入了速度/加速度规划后，路径规划结果可以解决自动驾驶中车辆跟随轨迹进行变化的实时性问题。这种方案是在实际执行过程中将运动规划进行局部化处理，分而治之。然而这种方法在轨迹规划时刻并为你考虑相应的速度，故可能出现规划的轨迹并不能很好的适配速度曲线。故，该方法比较适合做城市道路的低速自动驾驶规划控制。

其二是一种将运动规划的时空问题分解为从正交方向上进行考虑的实时规划方案，这两个正交方向分别是以纵向X及横向Y为坐标系进行考虑的。这种规划方法在轨迹规划的时候就考虑了速度，故在实际应用中，可以很好的适应速度变化较大的轨迹规划情况。故，该方法更适合于高速公路场景下的高速自动驾驶规划控制。

1、运动规划中的车辆模型

车辆模型是基于车辆姿态和道路模型进行定义的，车辆姿态可以通过如下元素组合进行定义：



(x,y)表示道路二维平面上的位置，θ表示航向角，k表示曲率（也即θ的变化率），v代表与轨迹相切的速度值。单位时间内的横纵向坐标变化值表示如下：

$$\Delta x = v\cos\theta$$
$$\Delta y = v\sin\theta$$

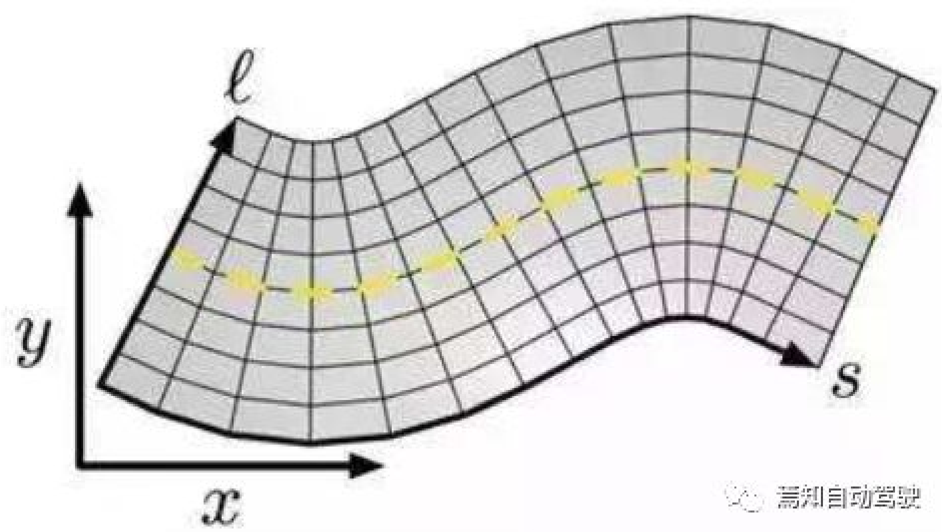
在考虑了车辆产生连续路径情况下，其路径方向s与车辆姿态变量之间应该满足如下微分方程关系：

$$dx/ds = \cos(\theta(s))$$
$$dy/ds = \sin(\theta(s))$$
$$d\theta/ds = k(s)$$

以上变量中，系统需要提前输入曲率k及航向角θ的约束限值。但是，对于k与θ之间的关系也需要根据实际情况做相应的约束限制，有研究证明，在高速段自动驾驶过程中，曲率k与θ满足一定的线性关系：$\theta = vk$。而低速段，则不再满足线性关系，而是多次曲线方程的非线性关系。

2、运动规划中的道路模型

车辆运动规划模型部分严重依赖于道路模型，特别是在车辆沿着道路行走过程中，需要定义车道中心线作为轨迹基准线。如下图标表示了一段道路曲线坐标表示方式：



原点x轴附近，横向距离l随着y增加而增加。假设车道的宽度保持为定值，那么整个车道可以表示为车道中心参考线（黄色曲线表示）纵向方向的一组点：$\{p(s,l) \in R\}$ 其中我们将s表示沿着路径切线方向的距离，l表示为垂直于路径s方向对的横向距离，则该轨迹基准线则可以采用一定微距进行采样的方式获得。每个采样函数都包含如下坐标信息 $f(s)=f[r_x(s),r_y(s),r_\theta(s),r_k(s)]$ 这里我们定义两种坐标系来分别表示车辆姿态，一种是相对坐标系/道路坐标系 (s,l) ，另一种是绝对坐标系/世界坐标系 $p(s,l)=[x_f(s,l),y_f(s,l),\theta_f(s,l),k_f(s,l)]$ 两种坐标系之间的关系满足如下：

$$
\begin{aligned}
x_f(s,l) &= r_x(s)+l\cos(r_\theta(s),\pi/2)\\
y_f(s,l) &= r_y(s)+l\sin(r_\theta(s),\pi/2)\\
\theta_f(s,l) &= r_\theta(s)
\end{aligned}
$$

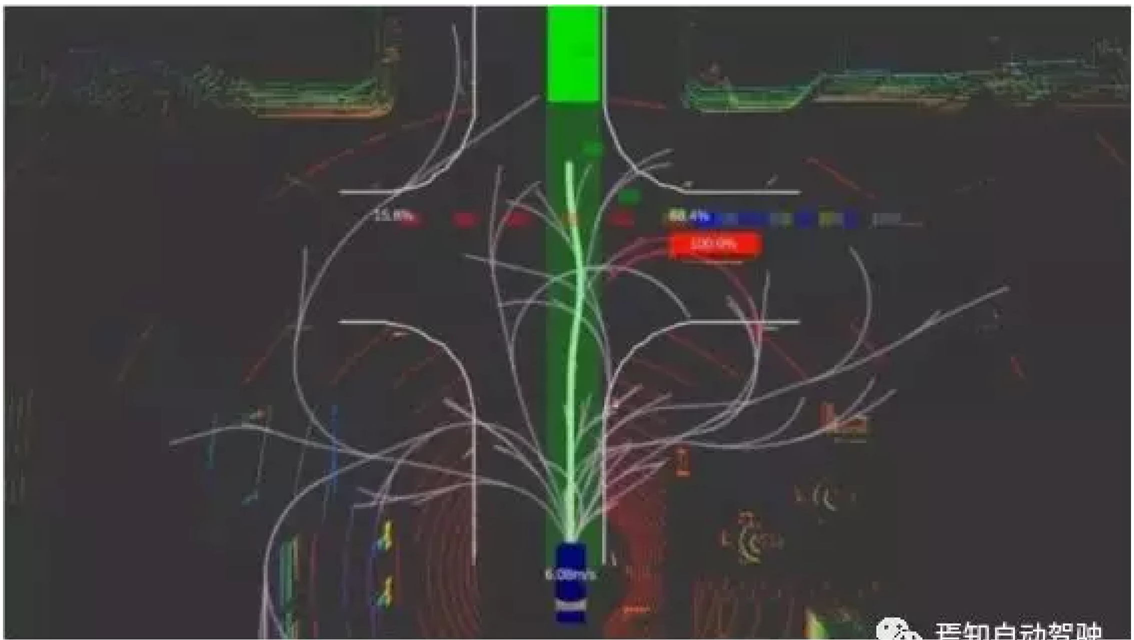
对于一定弯道曲率情况下的坐标表示方式如下：

$$
k_f(s,l)=\frac{1}{(r_k(s)^{-1}-l)}
$$

在自动驾驶中为了防止在过弯过程中出现甩尾或向弯道外侧甩出弯道的情况，从如上公式可以看出，在s大小不变的情况下，该曲率 $k_f(s,l)$ 值需要随着横向距离l向弯道内侧增加。

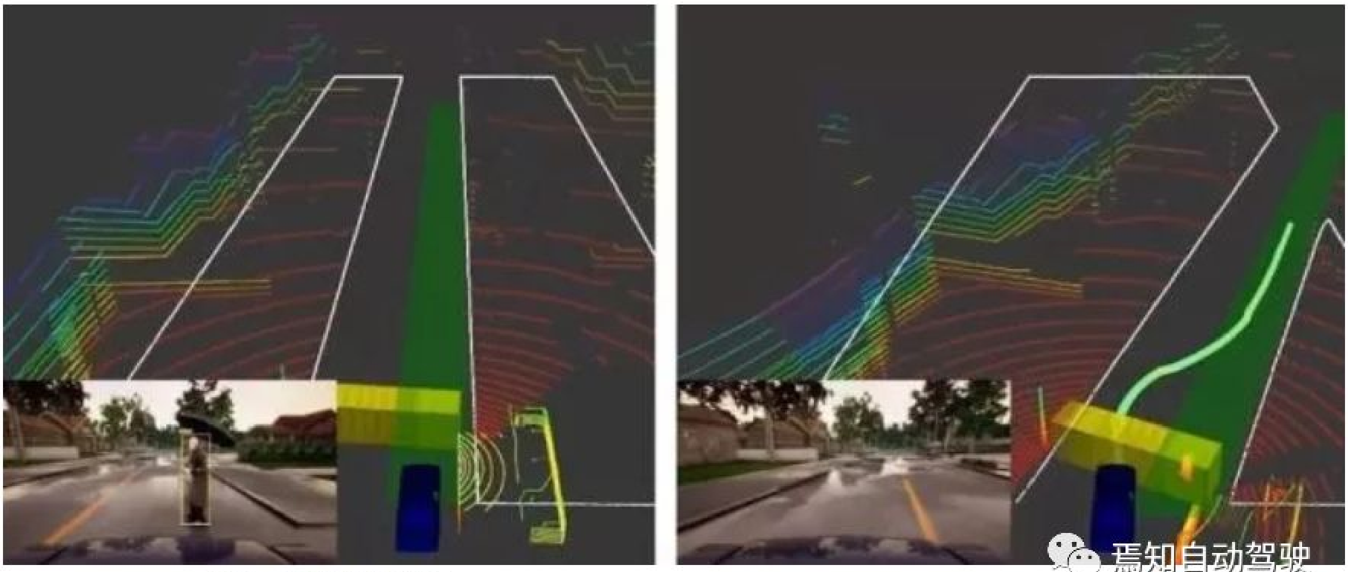
3、运动规划中寻找最优路径

如前所述，运动路径规划可被归结为一个利用最小化代价函数值找到最优路径问题。即我们将车辆路径定义为从原始点连续映射到车辆姿态集合 $\rho:[0,1]\rightarrow C$ 。如下图a表示了车辆运动规划系统从一个网格开始，网格是一个巨大的预先计算的图形，包含了车辆在无障碍物环境中可以采取的所有不同轨迹。



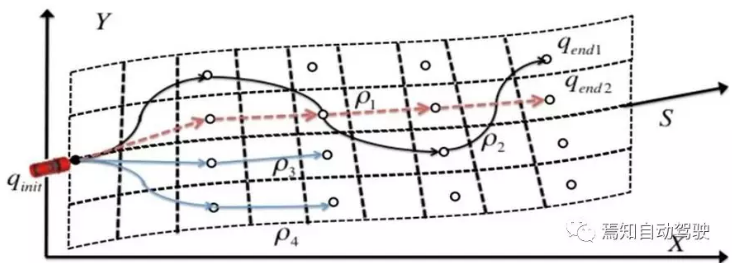
图a 原始搜索路径总和

我们搜索代价函数的方式是找到一条路径，从最初的姿态出发，到达一个期望的最终姿态，这一过程必须保证在满足一定约束条件下，代价函数最小化，寻找这一最优路径过程中，我们需要将样本点放置在可能穿越的潜在区域。如下图c表示最终搜索的最优路径。



图b 有障碍物路径　　　图c 无障碍物且代价函数最小路径

如下图，我们均匀的将车道划分为s长度和l长度相等的网格，每个网格的中心点(s_i, l_j)作为轨迹样本点。如下图，所示的网格和样本轨迹点共有16个轨迹点（4行在s方向，4列在l方向），由此可以规划出全部候选路径总计256条，规划问题中需要在这256条中寻找最优路径。



样本点的连接方式可采用多项式回旋曲线模型进行链接，如上的表示方式可设置为弧长多项式（对应了s方向），其回旋次数可表示为3阶曲线。

$$k(s) = k_0 + k_1 s + k_2 s^2 + k_3 s^3$$

如上图，当考虑以连续曲率连接初始位置姿态$q_{init} = (x_1, y_1, \theta_1, k_1)$与终止位置姿态$q_{endi} = (x_i, y_i, \theta_i, k_i)$时，曲率的一阶和二阶导数需要满足的初始约束如下：

$$\begin{aligned} k_0 &= k_1 \\ k_1 &= dk(0)/ds \\ k_2 &= d^2k(0)/ds^2 \end{aligned}$$

这样的方式对于求解未知参数(k_3, s_G),通过梯度下降法便可以很快求解得到。

当然，除了采用3次回旋曲线外，在一些复杂的自动驾驶工况下，也使用5阶曲线作为轨迹曲线，表示如下

$$k(s)=k_0+k_1s+k_2s^2+k_3s^3+k_4s^4+k_5s^5$$

两者区别在如下工况中体现得较为明显：

在满足边界约束条件时，对应于车轮转速的曲率二阶导数dk^2/ds^2在三阶曲线路径中是不连续的，而在5阶曲线中则可使得dk/ds和dk^2/ds^2连续；在低速情况下，三阶引入的不连续性对于下层反馈控制的影响不是十分显著，而在高速情况下，这种不连续性则不可忽略。故针对如纠偏曲线，自动换道回退曲线等情况，其回旋曲线模型要求的采样点越多越能够模拟出曲线形状的真实性，利用5次曲线能够更好的表示出轨迹预测情况。此外，此类功能针对均是针对高速工况，采用5次曲线也可更加有效的解决3次曲线对下层反馈控制的不连续性。

那么是否具备这样一种方案，它既能规避低阶次项带来的不连续性问题所造成的冲击，又能规避高阶次项所带来的较大计算量呢？

答案是肯定的。还是以自动换道为例，当换道曲线采用的方程式里变量以轨迹路径单位长度s表示时，我们需要5次方才能真正意义上的表示出最优的回旋曲线模型，这一反馈控制输入也是以车轮端移动的单位距离作为反馈调节，而这种距离的探测完全依赖于诸如雷达、摄像头等环境传感器来进行，既然是环境传感器，其受环境的影响因素也就不能忽略，在某些极端环境下可能造成探测不精准等不利影响。而我知道相对距离和加速度有着二次方的关系，表示如下：

$$\Delta s_i=\frac{1}{2}at^2,\quad s=\sum_{i=1}^{1-N}\Delta s_i$$

故一段较长的距离可看成每一小段较小直线距离的叠加，这样如上的公式中关于距离的可变因子就可以被分解为加速度表示的可变因子，因为加速度是可以通过自身加速度传感器进行精准探测的，这里不用再受制于环境。故这样的探测结果对于路径规划控制也是有利的。

如上过程规划出几条可能的路径后，就需要采用一定的优化算法搜索最优路径。对于路径q_i连接了从初始点至终点$n_0,n_1,\ldots,n_k$的轨迹，该轨迹既定的代价函数包含两个部分：

$$\mathrm{cost}(q_i)=c(q_i)+u(q_i)$$

$c(q_i)$表示为沿预测路径的累积代价，$u(q_i)$表示结束当前路径终点n_k而引入的新的代价，也称为增量代价。对于每一次迭代过程可表示如下：

$$\mathrm{cost}[q(n_0,n_1,\ldots n_k)]=c_{\min}[q(n_k)]+u[q(n_{k-1},n_k)]$$

如上方程中$c_{\min}[q(n_k)]$表示为达到节点n_k的最小代价函数，$u[q(n_{k-1},n_k)]$表示为倒数第二个点到当前最后一个点的增量代价。最后选出的最优轨迹曲线应该是累积代价函数和最小的路径。

以上我们所设计的代价函数应充分考虑前面所提到的几个要素，其一，我们的回旋曲线接近车道中心线；其二，我们所规划的路径是无碰撞的；其三，我们所规划的回旋路径形状足够光滑，保证了舒适性和控制的可行性。

3

总结

本文详细阐述了运动规划中的路径规划过程，包括车辆模型、道路模型、轨迹设计等，同时，也详细说明了关于如何利用代价函数进行轨迹预测的过程，对于工程设计中需要考虑的要素我们也做了充分说明，这些轨迹预测策略都可以被很好的利用在低速或速度变化不大的简单场景工况下，下文我们将引入速度规划对运动规划的整体影响来充分阐述相应的预测过程策略。