

知荐 | 规划控制论文综述精选一

Abstract

论文选自arXiv，概述城市工况规划控制技术的状态。初期的公众号规划是精选50篇自动驾驶论文进行学习记录，为了提升学习效率和质量，通过这种方式进行自我督促。

Contents

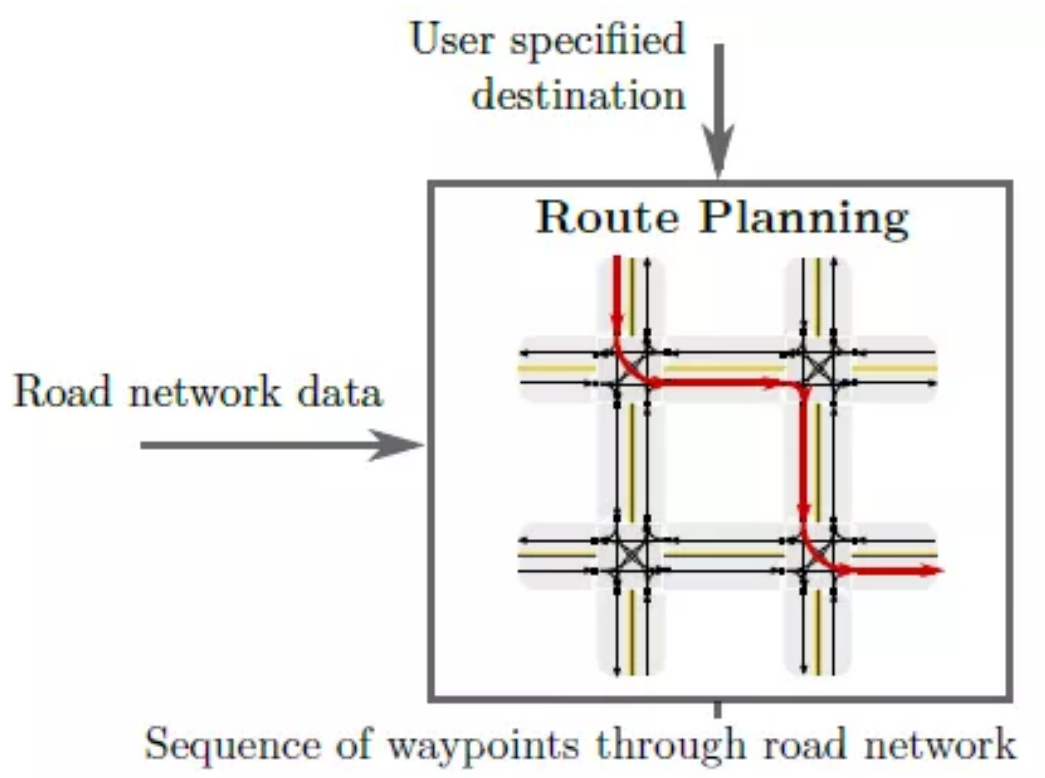
本篇论文主要从以下几个方面进行讲解：

- (1) 无人驾驶决策层技术的概述
- (2) 规划控制建模
- (3) 运动规划
- (4) 车辆控制
- (5) 总结

无人驾驶决策层概述

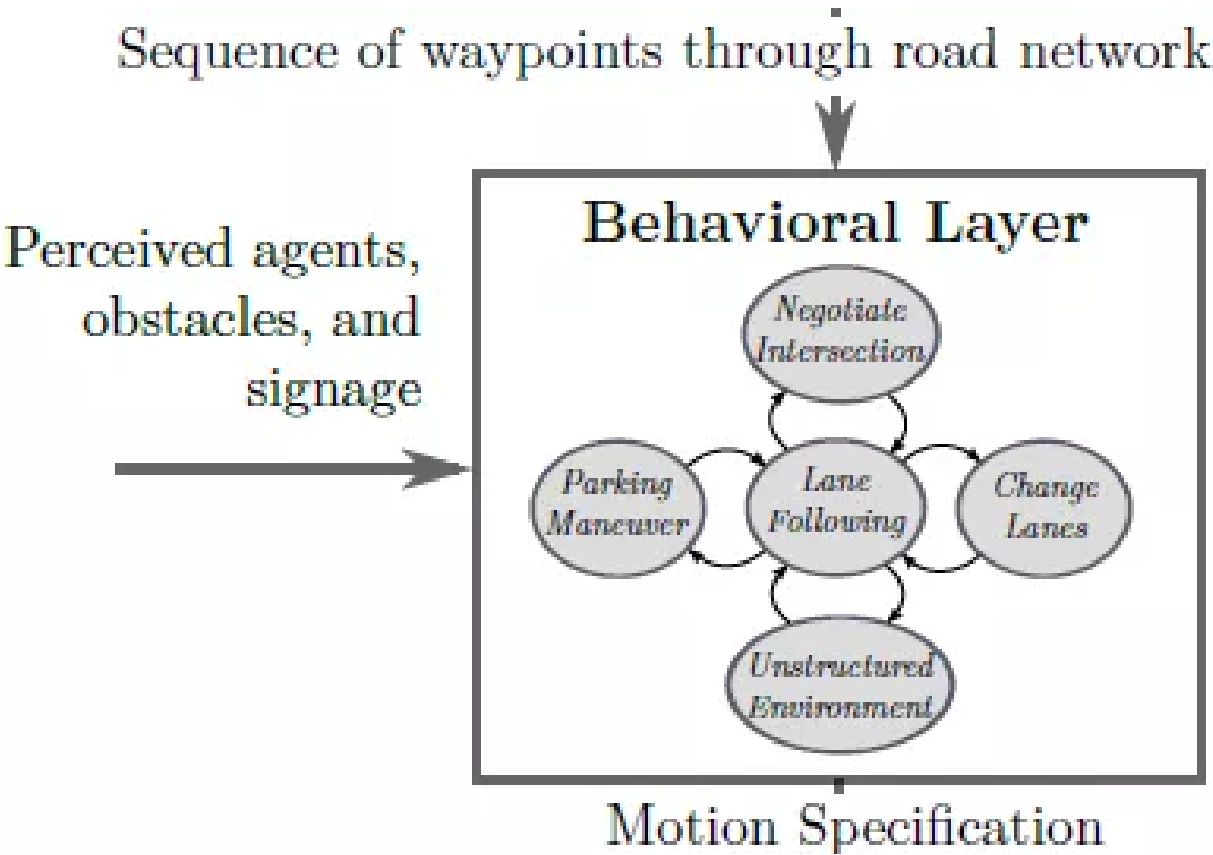
决策任务分为四个等级：

第一层级：路线规划



通过将道路网络表示为一个有向图，其边权值对应于穿越路段的代价，这条路线可以表述为在道路网络图上寻找最小代价路径的问题。常用算法有Dijkstra和A_Star算法。

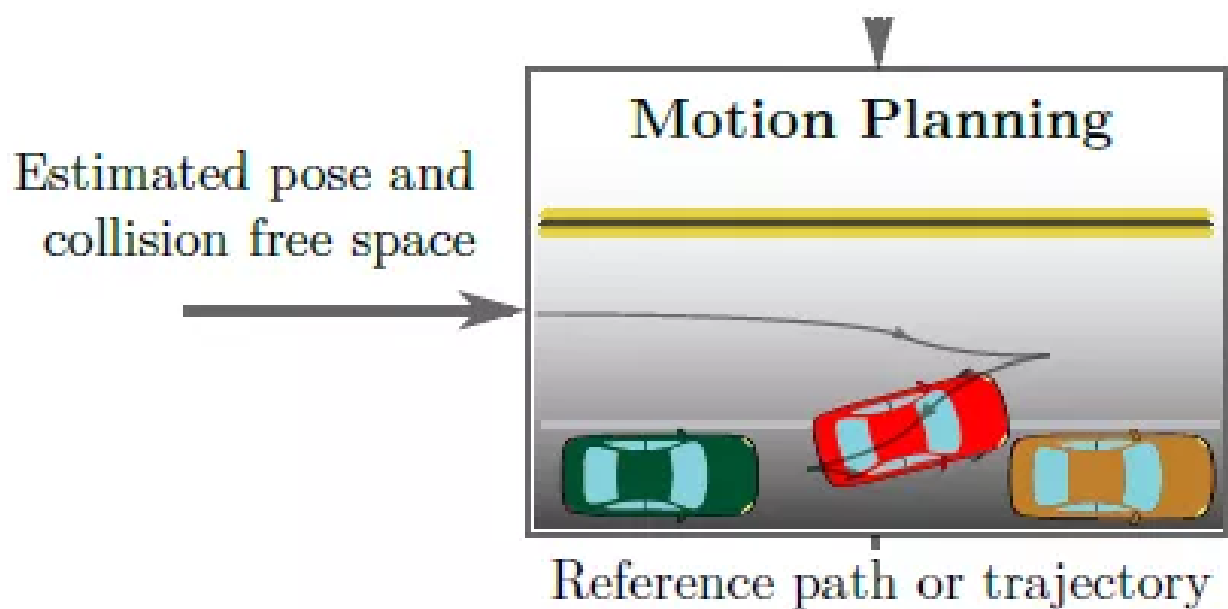
第二层级：行为决策



接收到全局规划的路线时，自动驾驶车辆必须能够选定的路线，并根据驾驶惯例和道路规则与其他交通参与者互动。

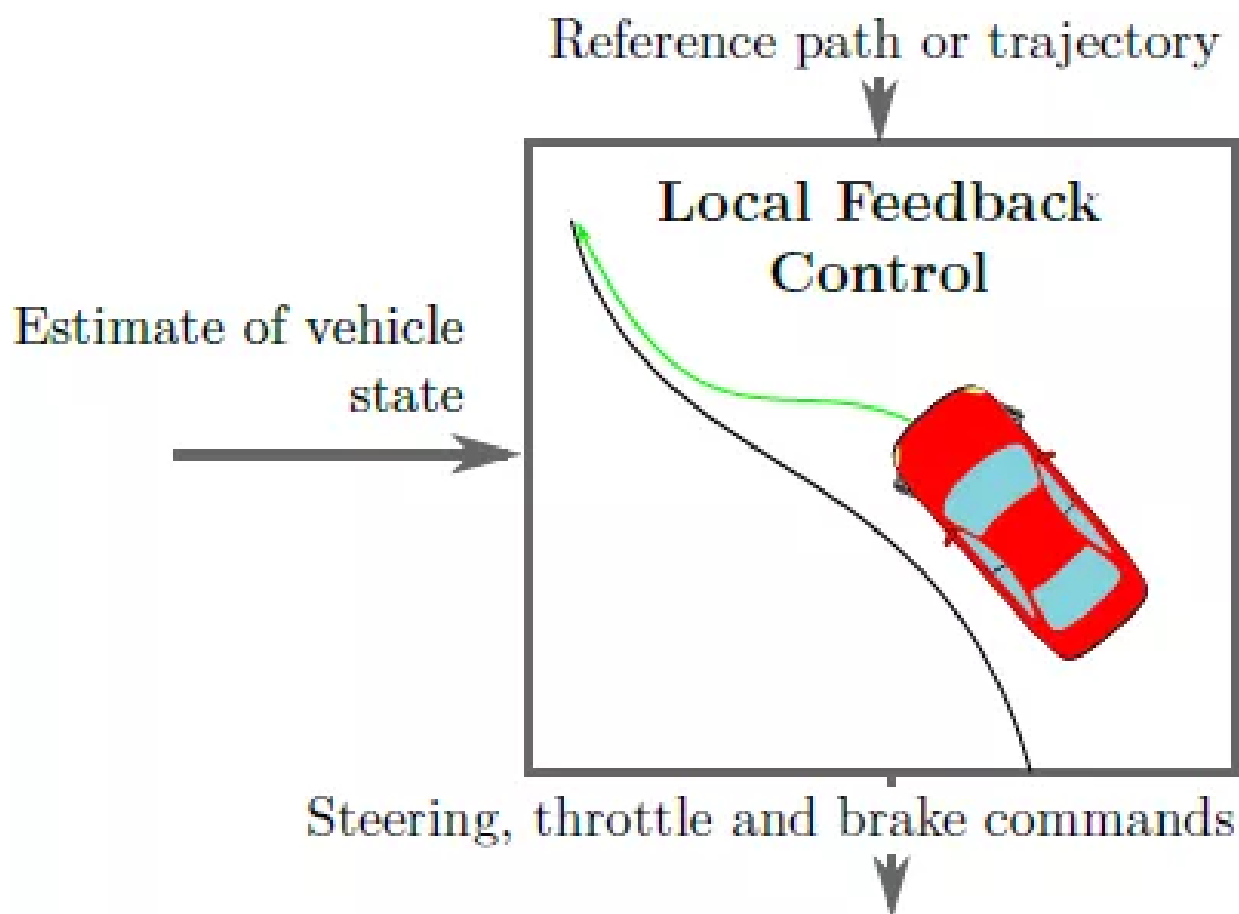
该层级较大的难点是，不同交通参与者的特点充满不确定性，需要对其他车辆、自行车和行人未来轨迹的意图进行预测和估计，解决方法主要有基于机器学习的技术如：高斯混合模型和高斯过程回归，谷歌的用于意图的学习技术，基于模型的方法即从传感器中直接估计意图。

第三层级：运动规划



当行为层决定在当前环境下执行的驾驶行为时，例如在车道内巡航、变车道或向右转弯，所选择的行为必须转换为可由低级反馈控制器跟踪的路径或轨迹。产生的路径或轨迹必须是动态可行的车辆，舒适的乘客，并避免与障碍物的碰撞检测到的车载传感器。寻找这样的路径或轨迹的任务是运动规划系统的责任。

第四层级：控制



反馈控制是主要方法，保证控制鲁棒性和稳定性是核心评价指标。

主要介绍路径规划与轨迹规划。

4.1 路径规划

有关路径规划专业性解释为：从初始状态开始，在满足全局和局部约束的情况下到达目标区域。根据是否考虑了解决路径的质量，使用可行和最优术语来描述该路径。可行路径规划是指在不关注解的质量的情况下，确定满足某些给定问题约束的路径的问题；而最优路径规划是指在给定的约束条件下，找到一条最优质量标准的路径的问题。

最常用的数值解法，通常不寻找精确解，而是尝试寻找一个满意解或一系列可行解，并收敛于最优解。这些方法的效用和性能通常由它们适用的问题类别以及它们保证收敛到一个最优解来量化。路径规划的数值方法大致可分为三类：

(1) 变分迭代法(Variational methods)

将路径表示为由有限维向量参数化的函数，通过使用非线性连续优化技术对向量参数进行优化来寻求最优路径。这些方法具有快速收敛到局部最优解的优点；然而，它们通常缺乏找到全局最优解的能力，除非提供了适当的初始猜测。

(2) 图搜索方法(Graph-search methods)

图搜索方法将车辆的配置空间离散为一个图，其中顶点表示车辆配置的有限集合，边表示顶点之间的转换。通过在这样的图中搜索最小代价路径，可以找到所需的路径。

图搜索方法不容易陷入局部极小值，但是，它们被限制为只优化有限的路径集，即那些可以由图中的原子运动原语构造的路径。

(3) 增量搜索方法(Incremental search methods)

增量搜索方法对配置空间采样，并逐步构建可达图(通常是树)，以维护可达配置的离散集合和它们之间可行的转换。一旦图足够大，至少有一个节点在目标区域内，就可以通过跟踪从开始配置到该节点的边来获得所需的路径。

与图搜索方法相比，该方法是递增图的大小直到在图中找到满意解。

4.2 轨迹规划

对于动态环境或带有动态约束的运动规划问题，更适合在轨迹规划框架中表述。

轨迹规划问题可以直接在时域上用一些变分方法进行数值求解，也可以将轨迹规划问题转化为具有附加时间维的构型空间上的路径规划。

	Model assumptions	Completeness	Optimality	Time Complexity	Anytime
Geometric Methods					
Visibility graph [33]	2-D polyg. conf. space, no diff. constraints	Yes	Yes ^a	$O(n^2)$ [68] ^b	No
Cyl. algebr. decomp. [76]	No diff. constraints	Yes	No	Exp. in dimension. [65]	No
Variational Methods					
Variational methods (Sec IV-C)	Lipschitz-continuous Jacobian	No	Locally optimal	$O(1/\epsilon)$ [77] ^{k,l}	Yes
Graph-search Methods					
Road lane graph + Dijkstra (Sec IV-D1)	Arbitrary	No ^c	No ^d	$O(n + m \log m)$ [78] ^{e,f}	No
Lattice/tree of motion prim. + Dijkstra (Sec IV-E)	Arbitrary	No ^c	No ^d	$O(n + m \log m)$ [78] ^{e,f}	No
PRM [79] ^g + Dijkstra	Exact steering procedure available	Probabilistically complete [80] [*]	Asymptotically optimal [*] [80]	$O(n^2)$ [80] ^{h,i,j,*}	No
PRM [*] [80], [81] + Dijkstra	Exact steering procedure available	Probabilistically complete [80]–[82] ^{*,†}	Asymptotically optimal [80]–[82] ^{*,†}	$O(n \log n)$ [80], [82] ^{h,i,j,*,†}	No
RRG [80] + Dijkstra	Exact steering procedure available	Probabilistically complete [80] [*]	Asymptotically optimal [80] [*]	$O(n \log n)$ [80] ^{h,i,j,*}	Yes
Incremental Search					
RRT [83]	Arbitrary	Probabilistically complete [83] ^{‡,*}	Suboptimal [80] [*]	$O(n \log n)$ [80] ^{h,i,j,*}	Yes
RRT [*] [80]	Exact steering procedure available	Probabilistically complete [80], [82] ^{*,†}	Asymptotically optimal [80], [82] ^{*,†}	$O(n \log n)$ [80], [82] ^{h,i,j,*,†}	Yes
SST [*] [84]	Lipschitz-continuous dynamics	Probabilistically complete [84] [†]	Asymptotically optimal [84] [†]	N/A ^j	Yes

路径规划方法分类

4.3 变分迭代法

解决轨迹优化问题。

(1) 直接变分法

直接变分方法背后的一般原则是限制一个有限维子空间的近似解，许多数值逼近方案已被证明在轨迹优化问题中十分有效。常见的两种方案：配置数值积分器和伪谱方法。

(2) 间接变分法

通过寻找满足这些最优性条件的解来解决问题。这些最优性条件被描述为控制状态和一组协态的常微分方程（ODE）的增广系统。

然而，这个常微分方程系统会导致一个两点边值问题，很难用数值方法解决。一种方法是改变问题的自由初始条件，并将系统向前整合以寻找初始条件，从而得到期望的终态。间接方法的优点，就像射击方法那样，是将优化问题的维数降低到状态空间的维数。

4.4 图搜索方法

尽管变分方法在许多情况下是有用的，但其适用性受到其收敛于局部极小值的限制。图搜索方法是通过在离散化的路径空间中执行全局搜索来减轻问题的方法。图搜索方法离散了配置空间并将

其表示成图，然后在图上寻找最小代价路径。

如何构建图？三种方法：手工制作的线图，从几何表方法和取样法。

图搜索算法，迪杰斯特拉算法：搜寻最短路径应用最广泛的算法，该算法执行最佳的第一次搜索，以构建图的方式来表示从给定顶点到其他顶点的最短路径，当只需要一条到单个顶点的路径时可以用启发式来指导搜索过程；

A_Star：是一种最突出的启发式算法，在给定启发式函数的条件下，A_star效率最优并且可以返回最优解。

D_Star：是一种实时的重规划算法

4.5 增量搜索方法

对固定图离散化进行搜索的技术的一个缺点是，它们只搜索可由图离散化中的原语构造的路径集。因此，这些技术可能无法返回可行路径，或者返回明显的次优路径。增量可行运动规划器致力于解决这一问题，并为任何运动规划问题实例提供可行路径，如果存在一个实例，只要有足够的计算时间。通常，这些方法会逐步构建配置空间越来越细的离散化，同时尝试确定在每一步的离散化中是否存在从初始配置到目标区域的路径。

一种重要的增量路径规划方法是基于这样一种思想:将根植于车辆初始配置的树向外递增，以探索可达配置空间。“探索性”行为是通过迭代地从树中选择一个随机的顶点，并通过应用该顶点的转向功能来扩展选中的顶点来实现的。一旦树长得足够大，可以到达目标区域，通过跟踪目标区域顶点到初始配置的链接，得到的路径将被恢复。

车辆控制

当前应用最广泛的是反馈控制器，可以确保在存在建模误差以及其他不确定因素带来的误差情况下将实际轨迹逐渐稳定到参考轨迹中。

控制目标可以是路径稳定和轨迹稳定。

Controller		Model	Stability	Time Complexity	Comments/Assumptions
Pure Pursuit	(V-A1)	Kinematic	LES* to ref. path	$O(n)^*$	No path curvature
Rear wheel based feedback	(V-A2)	Kinematic	LES* to ref. path	$O(n)^*$	$C^2(\mathbb{R}^n)$ ref. paths
Front wheel based feedback	(V-A3)	Kinematic	LES* to ref. path	$O(n)^*$	$C^1(\mathbb{R}^n)$ ref. paths; Forward driving only
Feedback linearization	(V-B2)	Steering rate controlled kinematic	LES* to ref. traj.	$O(1)$	$C^1(\mathbb{R}^n)$ ref. traj.; Forward driving only
Control Lyapunov design	(V-B1)	Kinematic	LES* to ref. traj.	$O(1)$	Stable for constant path curvature and velocity
Linear MPC	(V-C)	$C^1(\mathbb{R}^n \times \mathbb{R}^m)$ model [#]	LES* to ref. or path	$O\left(\sqrt{N} \ln\left(\frac{N}{\varepsilon}\right)\right)^\dagger$	Stability depends on horizon length
Nonlinear MPC	(V-C)	$C^1(\mathbb{R}^n \times \mathbb{R}^m)$ model [#]	Not guaranteed	$O(\frac{1}{\varepsilon})^\ddagger$	Works well in practice

车辆控制方法类别

写在结尾

本文综述了无人驾驶车辆决策问题的各个方面，重点是运动规划和反馈控制。对各种运动规划和控制技术的性能和计算需求的调查，为评估系统级设计的各种选择之间的兼容性和计算权衡提供了参考。