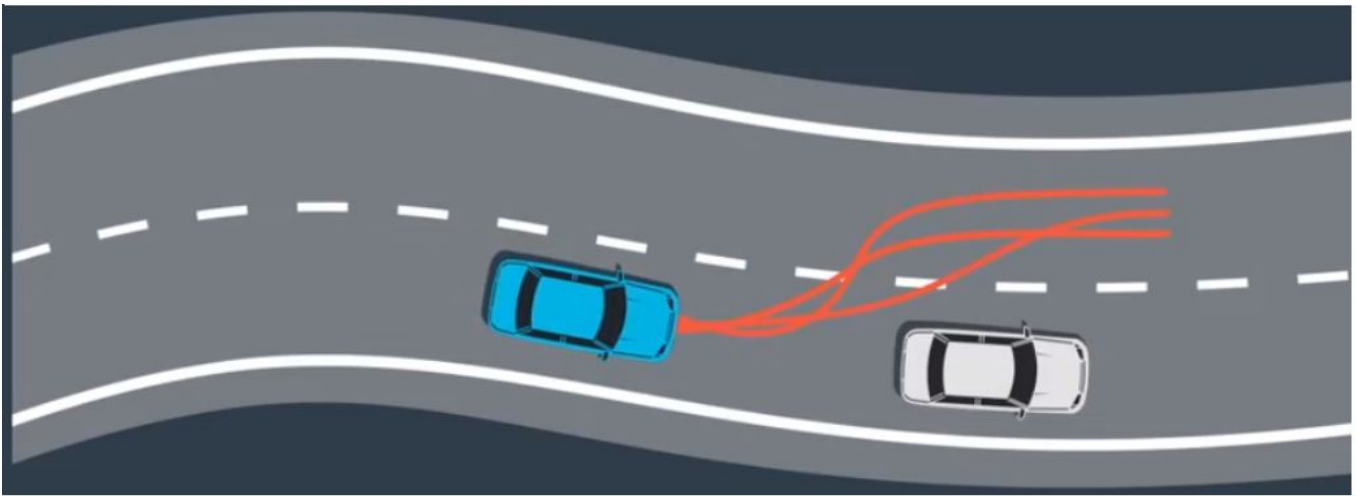


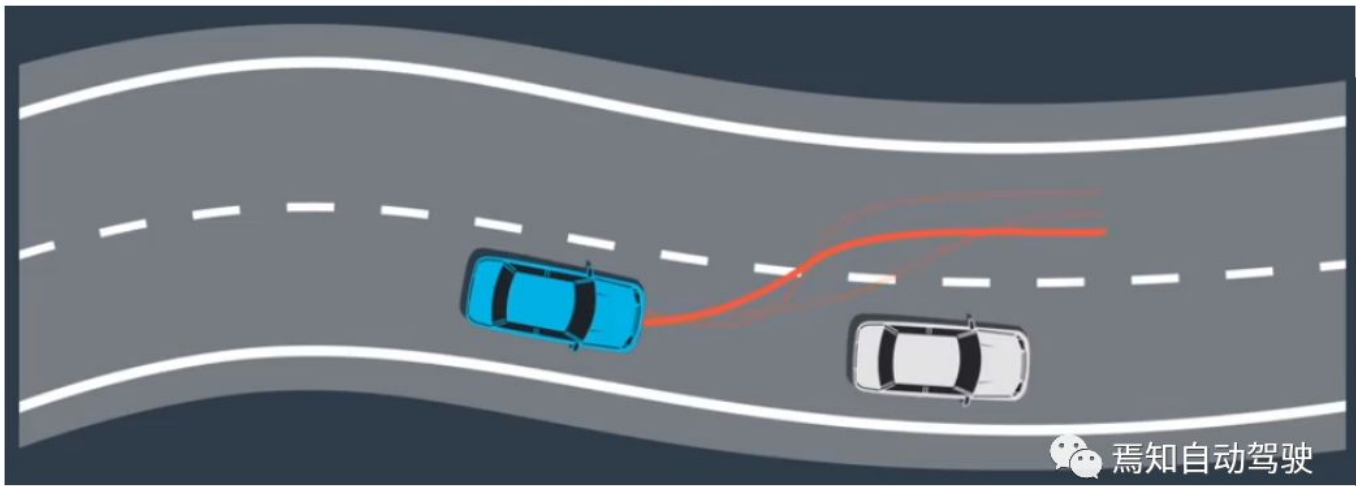
ADAS算法设计中的“运动规划” 详解（二）

前文充分讲述了运动规划中的轨迹规划策略，其目的是在规划结果中输出一条路径，该路径是不带有速度/加速度信息的绝对轨迹模型。而本文所讲的速度规划模块则是接收候选路径和上游决策结果，然后负责对该路径分配相应的速度值，以确保自车以合适的速度穿越该路径。速度规划过程必须包含如下一些基本信息：

1) 输入信息：若干候选路径和上游行为决策结果。如前所述，速度规划的前提是需要上游的若干路径需要充分考虑相关因素（如安全系数、离道路中心的距离、车辆模型限制，例如转向最大角度等等、曲线的平滑度、速度和曲率的变化关系、与障碍物的距离、车辆自身的压力），从而选择一条最优路径。

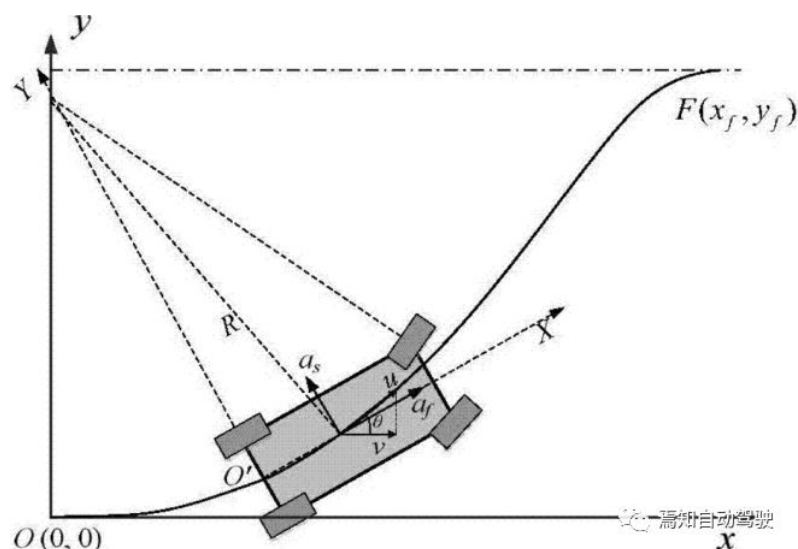


图a 若干候选路径



图b 最优路径（不带速度值）

2) 这里的候选路径可被表示为点序列，故速度规划则是将这些点与期望车辆信息（包含速度、加速度、曲率及其各自的变化率等信息）相关联；



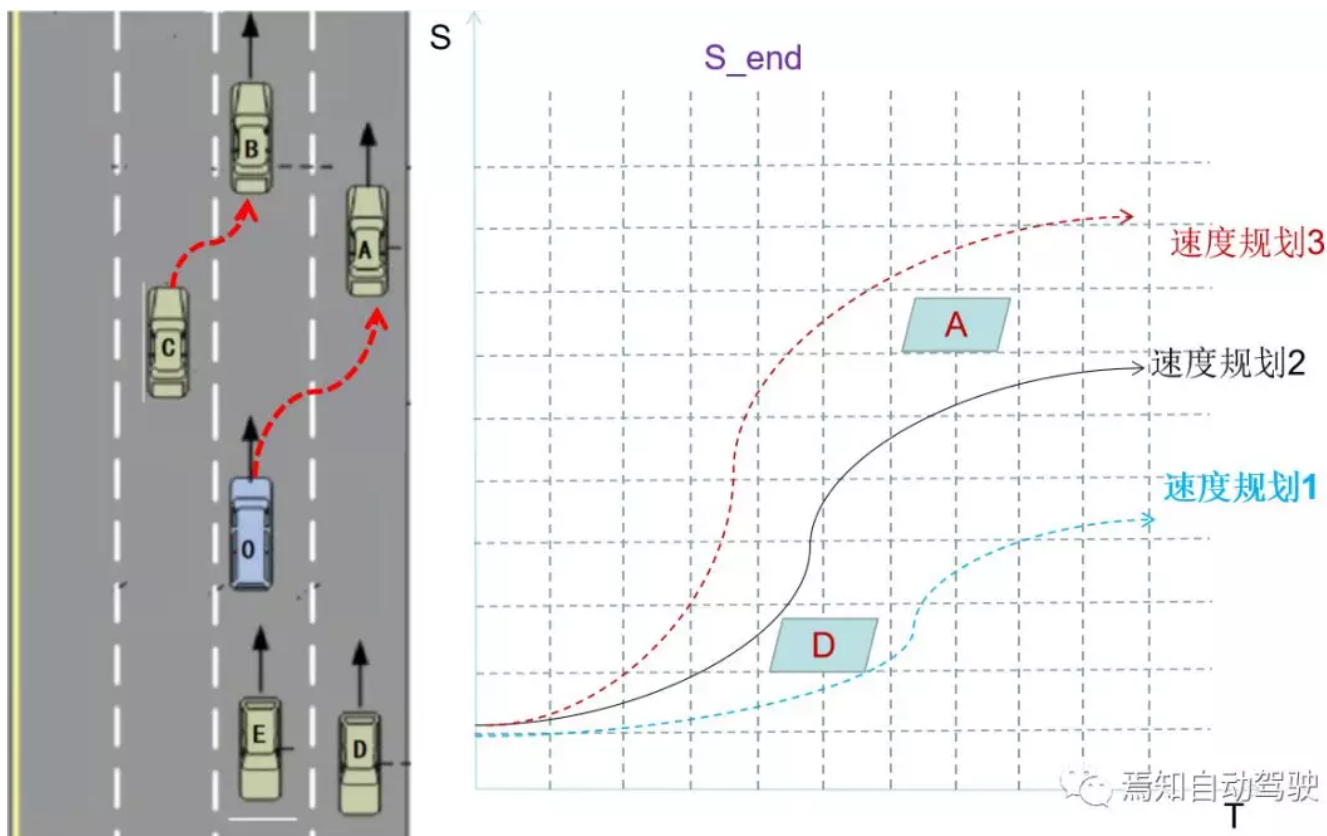
3) 约束条件：环境物理限制及驾驶舒适性。物理限制包含如加速度限制、速度限制等，舒适性可以体现在加速度、轮速、转角等信息的变化率上；

这里的速度规划模型采用ST（Speed Time）图模型进行规划，反应了速度和时间曲线的数学关系，当然其实间接也描述了速度、时间、位置三者的联系。该模型需要预先设定路径，预期轨迹与ST图上预设路径存在一定间隔，任意对象在ST图上会覆盖一定的区域。

● ○ 轨迹与速度规划策略

下面以自动换道来整体说明如何进行速度规划。

自动换道运动规划的结果是需要输出一条合适的换道轨迹线，如下图所示，表示了针对换道的典型的ST图，其中包含两个维度，其横轴表示时间，纵轴表示沿着给定路径的切向或纵向路径，轨迹规划结果会输出一条换道轨迹线，一般的，对该两辆车进行交通预测结果都是以恒定的速度跟随他们当前的车道继续行驶。A和D作为在目标车道有两个障碍物，且在任意时刻，ST图上的车辆A和D预测都是与S轴平行的线段。随着A和D沿着ST图上的轨迹移动，沿着T轴拖动这些平行于S轴的线段就形成了图中以浅绿色显示的阴影平行四边形区域。此外，与路径规划相似，ST图也可统一划分为小格子网格，每个网格均与代价函数相关。由此，速度规划则可以转化为ST图格子网络上最小化代价函数路径搜索问题。



我们在运动规划的目的是在 (t_0, S_0) 时，需要沿着最小代价函数的ST图最终达到 (t_{end}, S_{end}) 。其中有三个候选速度规划路径，分别如下：

1) 速度规划1：该路径表示了一条始终位于A车与D车之后的路径，故针对当前车辆位置而言，需要进行减速后确保A车和D车均通过后，自车方可变道进入右侧目标车道。

$$\begin{cases} V_{ego} \leq V_D \\ S_{ego} < S_D \end{cases}$$

2) 速度规划2：该路径表示了本车需要超越D车，跟随在A车之后进行变道，此时，在达到特定速度前，路径的斜率将保持持续增长，对应于路径S方向距离也会在同一点超过车辆a，故此时的速度规划应该在预测到D车速度之后，确保当前车辆的速度增长到与该D车平齐或大于该D车后，在确认换道可能性。此外，此过程中的速度规划还需要充分考虑与目标车道前车A的碰撞概率，最终确认换道可行性。

$$\begin{cases} V_{ego} > V_D, V_{ego} < V_A \\ S_{ego} > S_D, S_{ego} < S_A \end{cases}$$

3) 速度规划3：该路径是一种表示本车自动控制加速后将A车与D车均超越，在进行变道的逻辑，故该种情况下，本车的S距离比A和D均长。

$$\begin{cases} V_{ego} \geq V_A \\ S_{ego} > S_A \end{cases}$$

我们假设自车的行为决策是进行超过车辆D，对于A是让车行驶，这样的行为决策结果是针对D以上A以下的区域输出更少的代价值，该代价函数值是需要确保无碰撞的，这就使得路径选择函数中更多以D以上A以下的路径被选择为准，从而在搜索算法中避免更多无谓的算法选择。此外，速度规划曲线通常与加速度相关代价体现。如果速度规划曲线的两点之间过于陡峭，则说明可能产生了巨大

甚至不连续的加速度，而与之相关的代价函数将会变得很大，导致控制失败。此时便需要对网格中的速度规划点位置进行调整，对于给定每个网格的代价定义，ST图上的速度规划轨迹可以通过图形搜索算法如A*算法或Dijkstra算法进行计算，从而优化对整体速度规划轨迹的代价。在完成了ST图上速度规划曲线的计算后，我们就可以将每个轨迹点处的斜率视为速度，斜率导数视为加速度，此时针对运动规划阶段的轨迹就被相应的附上了速度/加速度信息。对于这种时空轨迹，我们统一的进行采样生成样本点后，附上相应的速度、加速度、曲率等信息一同发送到下层反馈控制模块，从而操控车辆进行自动驾驶。

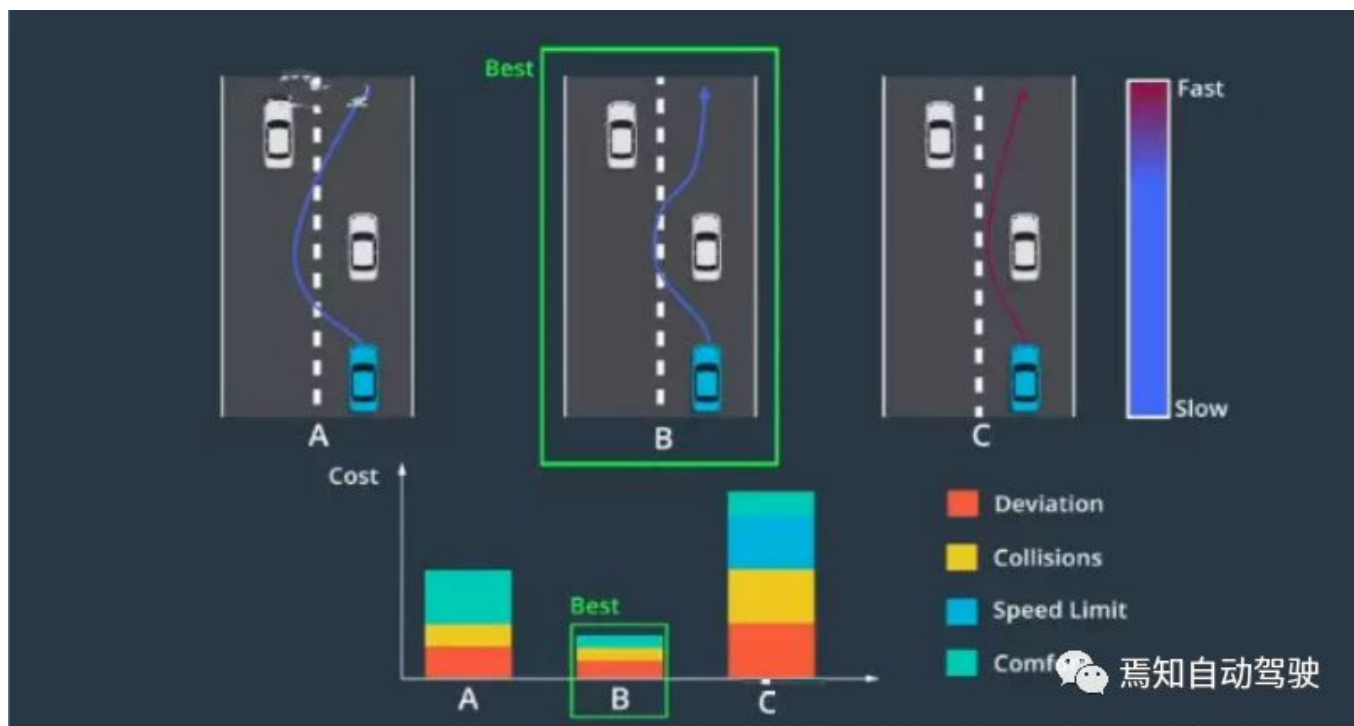
● 横向与纵向规划策略

如上在运动规划中使用路径-速度规划（建立ST图坐标）更适合于避障和服从上层模块行为决策，而对于期望得出时空轨迹的完整方法，该ST图就会显得不那么适宜。这时就可以进行横向-纵向规划（建立SL图坐标，其中S表示纵轴，L表示横轴），这里也可以简单的利用ST图解决问题，在建立时间与纵向S-t和时间与横向L-t维度上执行两个图搜索。一个期待的运动规划轨迹应当是平滑的，通常数学上表现为一阶导数连续（例如：位置、速度、加速度）。

不同的候选方案会生成不同的候选一维轨迹，当运动规划的纵向轨迹使得在某种状态下出现冲突/碰撞时，则终止对该纵向轨迹的继续规划，而剩余的轨迹则构成了候选横向轨迹的集合。一般的，纵向移动和横向移动是可以单独进行规划的，这样的运动规划的结果可以被更好的适用在一些高速场景或速度变化剧烈的场景中。而针对较低速度场景，由于车辆控制存在的不完整性，导致许多的横向运动规划轨迹无效，其真实地横向规划路径是受纵向规划路径影响的，横向移动可表示为纵向移动的函数 $M_{lat} = f(M_{long})$ 。相应的代价函数有如下的表示方式：

$$Cost = K_j J_t(f(M_{long})) + K_t \Delta S + K_l d_1^2,$$

其中纵向移动位置差 $\Delta S = s_{end} - s_{begin}$ ，三个影响因子 $K_j, K_t, K_l > 0$ ，d1表示执行到车端的候选补偿值，以上是基于纵向位置s的五次函数，且可以通过选择不同的组合 (T_j, d_j) 对方程式进行计算。利用最终计算代价函数的最小值就可选出最优轨迹，图示如下：



纵向规划与横向规划不同的地方是纵向规划的终止状态取决于一个移动潜在碰撞风险目标，而该潜在风险目标的确定取决于我们在路径规划中选择的所有备选路径。以跟随前车进行纵向规划为例，当本车跟随前车沿着s方向进行纵向移动，此时需要按照驾驶员设置的时距与前车保一定的跟车距离，同时为了防止碰撞，该距离不能小于最小跟车距离 D_{min} 。而此过程中，对于交通场景预测的重要性也体现出来，因为在时间帧内本车当前的预测目标轨迹 $S_{predict}(t)$ 将依赖于前方车辆的当前时刻预测轨迹 $S_{targetvehicle}$ 以及到达结束状态时预测道该目标的轨迹 $S'_{targetvehicle}$ ，其中 ρ 表示目标结束状态的预测轨迹对当前状态的影响因子。

$$S_{predict}(t) = S_{targetvehicle}(t) - D_{min} - \rho S'_{targetvehicle}(t)$$

对于自动换道场景中的纵向规划来说，通过让车换道到相邻车道与在纵向场景中跟随前车是相同的，如前面图所示，则可以有如下三种情况换道情况：

1) 通过让过车辆D换道，目标轨迹变为如下：

$$S_{predict}(t) = S_D(t) - D_{min} - \rho S'_D(t)$$

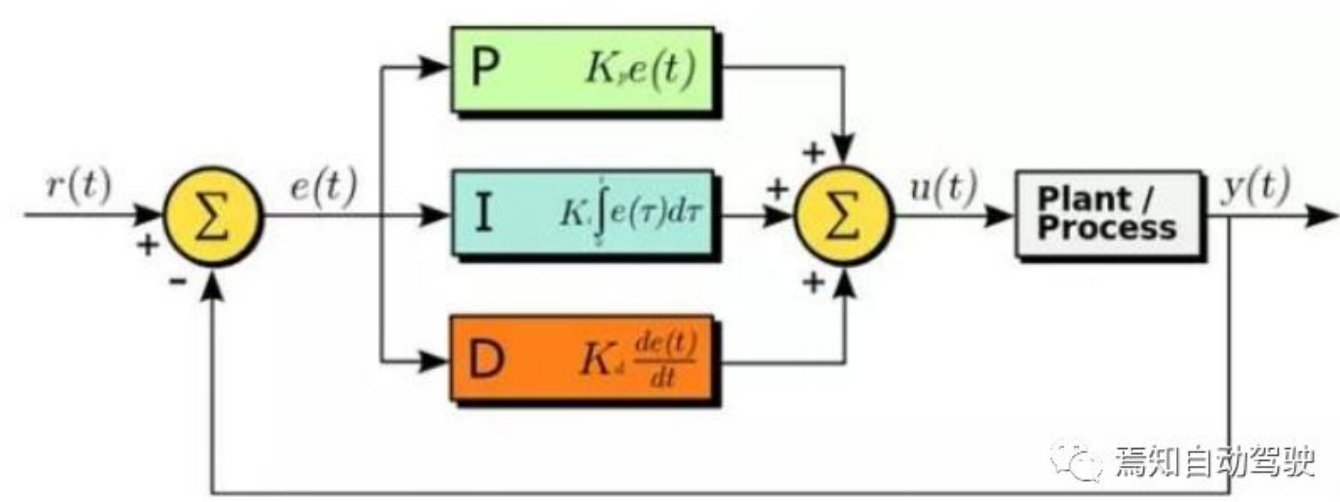
2) 通过超过车辆A换道，目标轨迹变为如下：

$$S_{predict}(t) = S_A(t) - D_{min} - \rho S'_A(t)$$

3) 通过插入车辆D与车辆A之间进行换道，目标轨迹变为：

$$S_{predict}(t) = \frac{1}{2} [S_A(t) + S_D(t)]$$

对于自动驾驶来说，其反馈控制与一般的机械控制相类似都是基于特定的预定义轨迹，并通过连续的反馈，追踪实际姿态与预定义轨迹上的姿态之间的差异，同时反馈系统需要引入附加的主动避障和路线优化。如下图表示了一种使用最为广泛的PID反馈控制系统。



其中e(t)表示当前期望姿态和实际姿态变量的跟踪误差，这种姿态变量有如下几种不同的定义。

- 1) 表示沿着轨迹的横/纵向距离差值 $e_s(t)=s_{pre}(t)-s_{cur}(t)$, $s=[x,y]$;
- 2) 表示各个轨迹点处的角度/曲率差值 $e_a(t)=\alpha_{pre}(t)-\alpha_{cur}(t)$, $\alpha=[angle,curvature]$;
- 3) 表示以上1) 2) 中所有变量的综合误差值 $e(t)=[e_s(t),e_a(t)]$;

P控制器表示当前跟踪误差的反馈，I和D控制器代表积分和微分项，Kp，Ki，Kd分别为其控制系数。

下面举例说明其中两种典型的自动驾驶PID控制器，

1) 自车横向规划PID控制器

该控制器利用了转角误差量 θ_e 和横向偏移误差量 l_e 进行综合控制，对于给定的时间帧,转向角度PID控制器如下：

$$\delta_n=K_1\theta_e+K_2l_e/V_s+K_3\dot{l_e}+K_4\sum_{i=1}^nl_e\Delta t$$

Vs表示前进速度，（K1，K2），K3，K4分别表示比例、积分、微分部分的增益，前馈P控制器参数 $K_p=[K_1,K_2]$ ，积分部分控制参数 $K_I=K_3$ ，微分部分控制参数 $K_D=K_4$ 。

2) 自车纵向规划PID控制器

对于纵向规划PID控制器来说，其更多的是关注s方向的前进速度Vs，从而控制加速/减速输出。由于不同道路曲率会影响纵向速度规划，高曲率下减速得更多，这里我们可以设计一个函数表示曲率与速度之间的关系 $V_{target}=f(\kappa_{current},\kappa_{predict})$ ，已有研究证明，速度和曲率之间存在一定关系为 $V_s=\frac{W_s}{\kappa_s}$ ，而目标速度的计算依赖于预测加速度的计算 $V_e=\alpha_{predict}t=V_{target}-V_{current}$ ，故以上公式可变换为如下情况表示：

$$V_{target} = f(\frac{W_{current}}{K_{current}}, \frac{W_{target}}{K_{target}})$$

那么对于一定时间帧 τ 内的加速/制动输出，通过PID控制的公式可以表示如下：

$$a_x = K_p V_e + K_I \sum V_e \tau + K_D \Delta V_e / \tau$$

这里讨论了典型的自动驾驶PID控制方法，实际上在应用过程中很多时候不止一次PID控制，比如对于加速控制来讲，为了达到控制的实时性和精准性，需要在整车级实现轮端PID控制，在系统级实现飞轮端PID控制。又如对于减速控制来讲，为了对实际加速度的输出进行精准调节，需要在制动器内部进行制动系统PID控制调节（比如制动压力回环反馈），而对于ADAS系统来说则更加关注整车PID控制调节（比如执行到减速度回环反馈）。

● ○ 总结

本章描述了自动驾驶车辆规划和控制的总体架构，包括路径规划、行为决策、交通预测、运动规划和反馈控制。在理论基础和实践中，这些方案均表现出一定的优势。自动驾驶重点不在于解决某一层面或模块的问题，而在于如何将一个很大的车辆规划和控制问题分为不同的层次，分而治之地解决各类局部问题。而在这一过程中每一个下层次均需要利用上层模块输出作为输入，并进一步通过数据流计算出实际控制信号，作为实际具体的解决方案。