

SL和ST

这里感觉必须得看代码才能理解，有很多细节问题不知道具体是怎么处理的，比如当前规划出一条无碰撞的路径后下一帧虚拟障碍物在哪呢？还有不是说同一障碍物会打标签用相同决策吗，那如果它的运动状态变了呢？情况会不会不一样呢？还有在下面这个例子中，如果SL决定向右转，显然是想加速避开，那此时ST是不是应该采取相同策略呢？会不会ST的dp采取了减速的策略导致做出无效操作了呢？

(老王说即使目标运动状态不变，当前规划出的路径已经可以避开障碍物了，虚拟障碍物依然要存在，因为不存在的话SL又会规划出一条直线了，但虚拟障碍物应该存在什么位置呢？如果障碍物运动状态有微小变化，虚拟障碍物在之后的规划中又会以什么位置存在呢？)

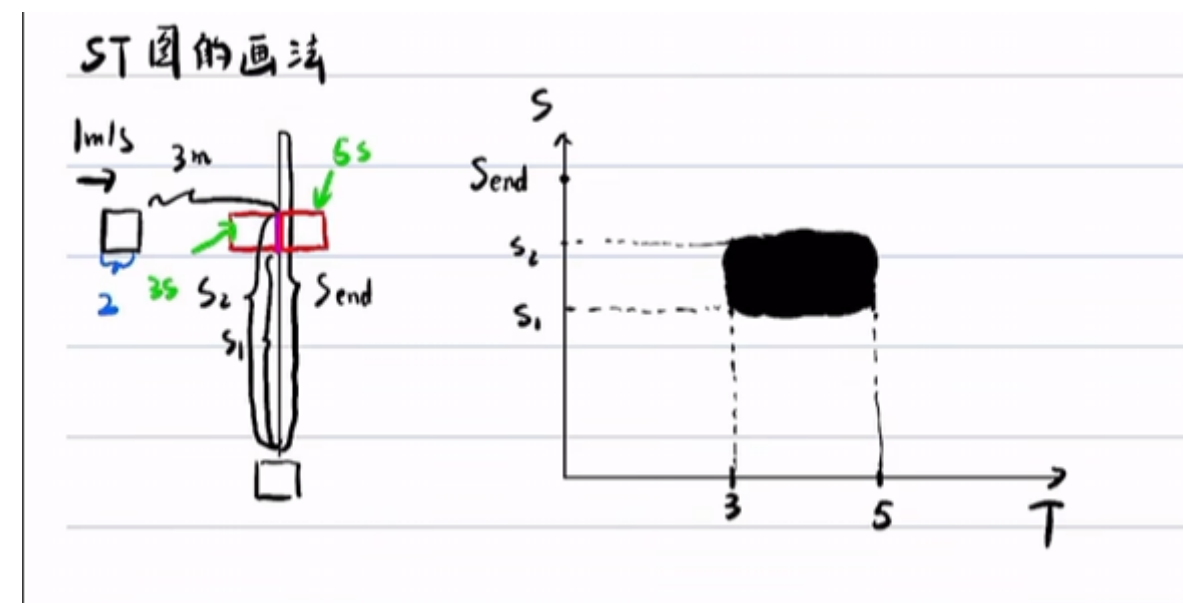
在SL图中，S以参考线为轴，L以垂直参考线为轴；而在ST图中，S是当前帧SL规划出的路径，T是时间，因此二者的S其实并不能算是同一个S。

SL图其实可以看成是一个笛卡尔坐标系下的xy图，图内的曲线就代表实际的轨迹，曲线的L坐标与O的偏移量代表车与参考线的偏移量。

ST规划又称为速度规划，因为它是基于SL的位置规划轨迹，对什么时间到达什么位置进行规划，这个本质上就是在对速度进行规划。

Apollo 1.5	EM planner	引入SL, ST曲线规划, 凸空间, 先决策后规划, 迭代优化
Apollo 2.5	加入lattice planner	
Apollo 3.5	大幅修改EM planner, 变成了public road planner	(取消了SL, ST迭代机制, SL只管静态ST只管动态, 加入scenario模块)
Apollo 5.0	再次大幅修改public road planner	SL的dp被取消
Apollo 6.0-7.0	改动不大	

在下面这个例子中，SL规划的直线轨迹将会在未来3s的时候与障碍物产生碰撞。在这种情况下，SL规划首先要根据预测的障碍物运动状态标记出它会与上一帧生成的路径产生碰撞的区域，称为虚拟障碍物。但仅靠SL规划并不能让车知道是该加速通过这段轨迹还是减速慢行，因为他没有时间维度，这就是ST规划要做的事。ST规划下的虚拟障碍物是带有时间戳的，其生成的曲线应该避开虚拟障碍物的区域，这样才能使SL规划的轨迹得到完美执行。



但ST规划也面临两个割裂的凸空间，即是应该加速超过还是减速？这时候又需要dp决策模块去开辟一个凸空间了。

