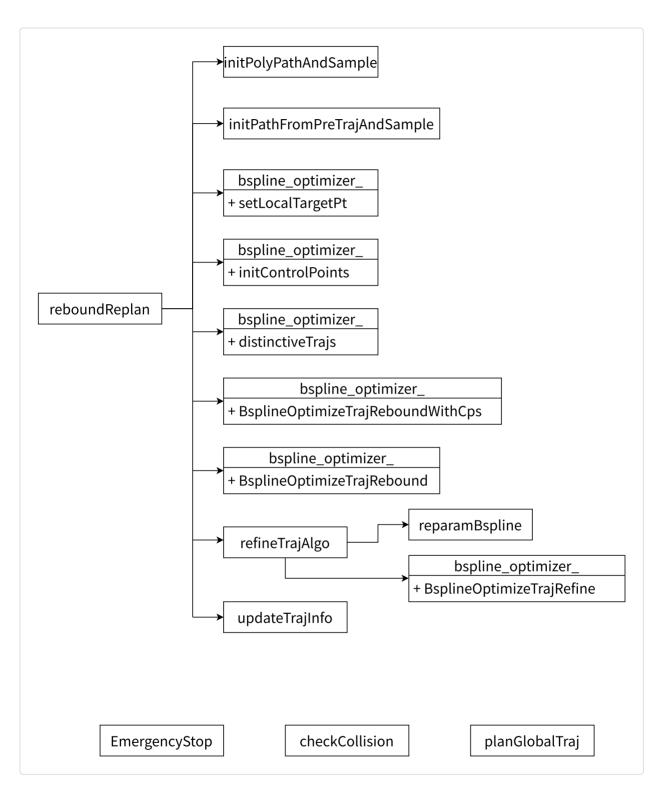
EGOPlannerManager



重构后的代码位于https://gitee.com/tiemuhua/ego_refactored.git

全局规划

把从出发点到目标点的直线当成是global trajectory。

局部规划

- 1. 从这条直线上挑一个距离当前点为给定值的点,当成是local trajectory的终点。终点可能位于障碍物里面,这个问题稍后再解决。
- 2. 通过minimum snap算法得到一条初始轨迹,并对轨迹采样得到初始控制点。具体有以下三种方式
 - a. flag_randomPolyTraj == true: 在start_pt和local_target_pt中点附近随机采一个点,然后用这三个点做最小snap多项式拟合得到初始轨迹。
 - b. flag_randomPolyTraj == false: 只用start_pt和local_target_pt做minimum snap多项式拟合得到初始轨迹
 - c. 在上一次规划得到的轨迹中采样,然后对采样得到的点进行minimum snap多项式拟合得到初始轨迹。
- 3. 在初始轨迹上以ts的间隔时间进行采样,不断缩短采样时间,直到相邻采样点间距小于 ctrl_pt_dist * 1.5,并且采样点数不少于7个。
- 4. 用采样得到的点作为B样条曲线的控制点,初始化BsplineOptimizer。返回轨迹前三分之二段所有位于障碍物当中的部分。
- 5. 对于前三段位于障碍物中的轨迹,分别考虑是从A star找到的轨迹方面绕过还是从相反的方面绕过,得到八条候选轨迹。
- 6. BsplineOptimizer对候选的B样条曲线进行优化,找到不经过障碍物的轨迹。
- 7. BsplineOptimizer对优化后的B样条曲线进行第二次优化,重新确定每段轨迹所需要的飞行时间,找到动力学代价最小的轨迹。