

第十五届全国大学生智能汽车竞赛
室外光电组（预赛）

技 术 报 告

学 校： 南京大学

队伍名称： 啊对对队

参赛队员： 刘子良 吴晓琦 李一洲 韩明浩

带队教师： 刘海涛

关于技术报告和学术论文使用授权的说明

本人完全了解第十五届全国大学生智能汽车竞赛关保留、使用技术报告和
研究论文的规定，即：参赛作品著作权归参赛者本人，比赛组委会和赞助公司可
以在相关主页上收录并公开参赛作品的设计方案、技术报告以及参赛模型车的视
频、图像资料，并将相关内容编纂收录在组委会出版论文集中。

参赛队员签名： 刘子良 韩明浩 吴晓琦 李一洲

带队教师签名： 刘海涛

日 期： 2022-1-4

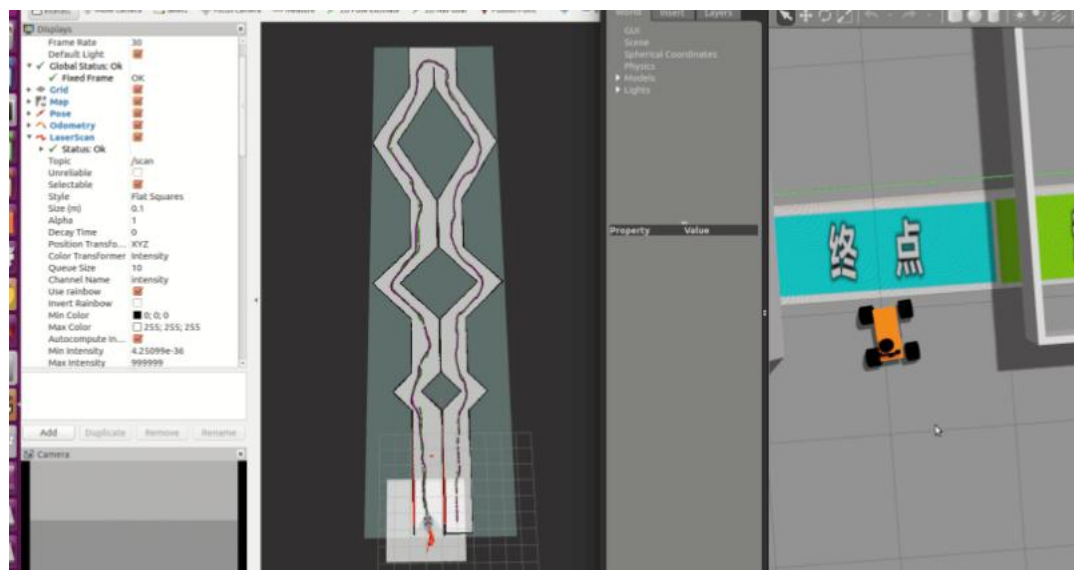
目 录

第 1 章 方案概述.....	1
第 2 章 问题描述.....	1
第 3 章 技术方案.....	1
第 4 章 方案实现.....	2
第 5 章 测试分析.....	5
第 6 章 作品总结.....	5
参考文献.....	5

第 1 章 方案概述

参考了基于 ros 组官方提供的代码包进行优化，重写了控制器和路径追踪算法，优化了 navigation 导航包中的全局路径规划和局部路径规划算法，创新点即添加权重值避免遇到道路中间的故障时进行反复规划路径导致出错或者变慢。

测试效果：正常情况下 1min10s 左右跑完带有路障的地图全程



第 2 章 问题描述

问题：进行小车定位的关键问题之一是比赛禁止使用 gazebo 中直接发布的小车位置坐标，只允许使用车模上有的相机、10HZ 激光雷达和 100HZ 的 IMU 进行定位。同时需要实现小车的自主导航和避障。

实现方案：订阅激光雷达/scan 话题进行一级定位，整合 rf2o 和 imu 进行更精确的定位，采用 amcl 修正定位误差，实现相对精准的定位，采用 navigation 实现小车自主导航和避障功能。

第 3 章 技术方案

这里使用 rf2o_laser_odometry 将小车的激光雷达数据转为能标识小车所在位置的里程计信息，同时为了提升精度，将此里程计信息通过 robot_pose_ekf 融

合 IMU 数据生成更精准的里程计信息。最后搭配 AMCL 解决小车的定位问题。成功转换好 tf 树后使用 navigation 实现小车自主导航和避障功能。

算法原理与改进思路：采用 RF20 算法和 robot_pose_ekf 功能包，同时重写了车辆轨迹跟踪算法，基于几何追踪的 pure-pursuit（纯追踪）算法。RF20 是一种基于平面激光扫描的 2D 里程表估算方法。对于没有精确里程表的移动机器人很有用。RF20 是一种快速精确的方法，可以从连续的范围扫描中估算激光雷达的平面运动。对于每个扫描点，其根据传感器速度制定范围流量约束方程，并最小化所得几何约束的鲁棒函数以获得运动估计。此算法优化的内容是，与传统方法相反，此方法不搜索对应关系，而是以密集 3D 视觉测距法为基础，基于扫描梯度执行密集扫描对齐，具有着极低的计算成本和较高的精确度。Robot_pose_ekf 订阅了编码器、imu、vo 三个主题，T 不需要三个主题同时有效，每个主题数据都会产生一个位置估计及协方差。为了解决 rf2o 对应包发出的主题不包含 covariance 这一问题，需要在 rf2o 包中提供一个静止和运动时的协方差矩阵以正常工作。之后可以联合 acml 建立移动机器人二维环境下的概率定位系统。ACML 它实现了自适应（或 kld 采样）的蒙特卡罗定位方法，其中针对已有的地图使用粒子滤波器跟踪一个机器人的姿态。建立好 tf 树后采用 Navigation 导航包进行路径规划（move_base）。Pure pursuit 纯追踪算法的原理部分以自行车模型为例，以车后轴为切点，车辆纵向车身为切线，通过控制前轮转角，使车辆可以沿着一条经过目标路点（goal point）的圆弧行驶，pure pursuit 需要给出预瞄距离 L_d ，因此后期当工作效果不好的时候可以对这个预瞄距离进行 tuning，但是过程繁琐，在某些情况下使用该模型效果不佳。改进思路为引入横向偏差、引入前馈控制等。

第 4 章 方案实现

mapping 功能包提供了基于激光的 SLAM，在 ROS 系统中使用 slam_gmapping 节点表示。通过该节点用户可以用机器人在移动过程中激光传感器获取的数据创建 2D 栅格地图。订阅主题 tf 用于激光器坐标系，基座坐标系，里程计坐标系之间转换和主题 scan 即激光器扫描数据。发布主题 map_metadata、map、~entropy，分别用于周期性发布地图 metadata 数据、周期性发布地图数据、发布机器人姿态分布熵的估计。之后进行定位，rf2o 功能包订阅主题 laser_scan (This topic

can be remapped via the `~laser_scan_topic` parameter) 和 `tf`，发布主题 `odom` (This topic can be remapped via the `~odom_frame_id` parameter) 和 `tf`。由于 `rf2o_laser_odometry` 包发出的 `/odom` 主题中的 `pose` 和 `twist` 是不包含 `covariance` 的，而之后使用的 `robot_pose_ekf` 需要用到协方差矩阵，因此这里需要提供一个静止和运动时的协方差矩阵才可以正常工作，以下的协方差矩阵可供参考

```
#define ODOM_POSE_COVARIANCE {1e-3, 0, 0, 0, 0, 0,
                                0, 1e-3, 0, 0, 0, 0,
                                0, 0, 1e6, 0, 0, 0,
                                0, 0, 0, 1e6, 0, 0,
                                0, 0, 0, 0, 1e6, 0,
                                0, 0, 0, 0, 0, 1e3}

#define ODOM_TWIST_COVARIANCE {1e-3, 0, 0, 0, 0, 0,
                                0, 1e-3, 0, 0, 0, 0,
                                0, 0, 1e6, 0, 0, 0,
                                0, 0, 0, 1e6, 0, 0,
                                0, 0, 0, 0, 1e6, 0,
                                0, 0, 0, 0, 0, 1e3}
```

有了协方差矩阵后只需要在特定位置将其付给 `odom.pose.covariance` 和 `odom.twist.covariance` 即可

```
odom.pose.pose.position.y = robot_pose.y();
odom.pose.pose.position.z = 0.0;
odom.pose.pose.orientation = tf::createQuaternionMsgFromYaw(robot_pose.yaw());

//set the velocity
odom.child_frame_id = base_frame_id;
odom.twist.twist.linear.x = lin_speed; //linear speed
odom.twist.twist.linear.y = 0.0;
odom.twist.twist.angular.z = ang_speed; //angular speed

if (lin_speed==0 && ang_speed==0)
{
    odom.pose.covariance = ODOM_POSE_COVARIANCE2;
    odom.twist.covariance = ODOM_TWIST_COVARIANCE2;
}
else
{
    odom.pose.covariance = ODOM_POSE_COVARIANCE;
    odom.twist.covariance = ODOM_TWIST_COVARIANCE;
}
```

`Robot_pose_ekf` 功能包订阅主题 `odom`、`vo`、`imt_data`，发布主题 `robot_pose_ekf/odom_combined`，其中有 `tf` 变换 `odom_combined` → `base_footprint`。可以使用 `roslaunch rqt_tf_tree rqt_tf_tree` 查看 `tf` 树，`roslaunch rqt_graph rqt_graph` 查看节点图。然后加入 `acml` 的配置来更新 `tf` 树，至此实现定位部分。接下来的部分应该算是整个智能车仿真比赛最为核心的部分——`move_base`（路径规划包）。`move_base` 提供了 ROS 导航的配

第 5 章 测试分析

通过在空白无路障地图上各种刁钻位置加入锥桶，测试小车跑完全程的时间和翻车概率。由于电脑性能有高有低，摄像头是否打开、局部规划器参数不足够摸透等问题，小组内测试结果有各种不同，包括但不限于成绩不同和翻车概率不同。最终找到比较满意的局部规划器参数，最终跑图成绩在 1min 左右。

第 6 章 作品总结

工作量相关：平等分工

首先本次项目配置和学习让我了解 ros 编程相关知识，虽然配置出错时一开始往往一头雾水和调参的过程十分痛苦，但学到的东西还是非常有用的。

参考文献

ROS Wiki: http://wiki.ros.org/robot_pose_ekf

ROS Wiki: [amcl](http://wiki.ros.org/amcl)

参数调整: <https://www.cnblogs.com/dyan1024/p/7825988.html>

ROS Wiki : <http://wiki.ros.org/tf/Tutorials>

ROS Wiki : [Gmapping - ROS Wiki](http://wiki.ros.org/Gmapping)

ROS Wiki : http://wiki.ros.org/hector_mapping

古月居 ACML 教程: <https://www.guyuehome.com/273>

Navigation 教程: <https://www.ncnynl.com/archives/201708/1880.html>

纯追踪算法: <https://zhuanlan.zhihu.com/p/48117381>