移动机器人项目组项目总结报告

# 项目概述

本项目主要是实现移动机器人多站点循环导航和自主探索建图功能，并拓展加入UI可视化界面增加导航目标和寻墙检测建图。在多站点循环导航任务中，首先要完成多点导航函数的撰写，实现单次导航任务，然后在函数中加入里程计清零环节和循环次数设定环节，最后使用UI界面实现循环次数和导航点的设定。在自主探索建图任务中，首先配置好自主避障参数，然后实现自主建图探索功能，接下来根据建图效果确立停止条件，最后完成停止条件函数的撰写。

# 项目组成员简介

**项目负责人：**

李超，攀枝花学院15级自动化

**项目成员：**

李博，攀枝花学院15级自动化

钟浩，攀枝花学院15级自动化

# 项目主要目标任务、技术指标

**（1）多站点循环导航**

·在地图中设置多个导航点，并使用UI界面显示在地图中

·实现UI界面设置循环导航次数功能，并实现自动循环导航

·实现单次循环导航结束，里程计位姿清零

**（2）自主探索建图**

·实现机器人以自由避障方式在环境中自由移动，并同时构图

·实现以机器人为中心，探索方圆XXX米范围，并同时构图

**（3）UI界面设计**

·编写rviz可视化插件，通过UI界面设置并显示导航点

·编写rviz可视化插件，通过UI界面设置循环导航次数并开启导航

**（4）寻墙检测建图**

·基于rrt（快速探索随机树算法）实现寻墙检测建图

# 项目相关资源及使用方法、项目进度情况

**（1）项目相关资源及使用方法**

·项目文件已打包，分为真实机器人和仿真机器人，具体使用方法请参照项目文件中的readme.txt。

**（2）项目进度情况**

**·Day 01**

上午

1.gmapping参数配置

2.总体launch文件的编写

3.机器人tf，状态，滤波器launch文件的编写

4.移动机器人调试

5.完成gmapping建图修图

下午

1.move\_base参数配置

2.amcl参数配置

3.调试机器人导航参数

4.在rviz中完成机器人单点导航

5.记录多个导航目标点

6.查阅随机循环导航函数的资料

7.完成随机循环导航功能节点函数

8.完成随机循环导航功能测试

**·Day 02**

上午

1.完成循环导航代码的编写

2.调试循环导航功能

下午

1.完成自主探索建图代码的编写

2.调试自主探索功能

3.优化自主探索功能代码

**·Day 03**

上午

1.完成初始化位姿功能

2.完成里程计清零功能

下午

1.完成可设置循环次数导航功能

2.调试初始化位姿，里程计清零，循环导航功能

**·Day 04**

上午

1.完成单点设定导航插件

下午

1.完善单点设定导航插件

2.调试单点设定导航插件功能

**·Day 05**

上午

1.完成多导航点记录插件

2.完成多点循环导航节点

下午

1.完成多点循环导航插件

2.调试多点循环导航插件功能

**·Day 06**

上午

1.查找关于巡墙算法的解决方案

2.完成rrt\_exploration(快速随机搜索树)的下载和demo测试

下午

1.修改rrt\_exploration接口

**·Day 07**

上午

1.修改rrt\_exploration接口

下午

1.修改rrt\_exploration接口

**·Day 08**

上午

1.在仿真机器人上完成快速随机搜索树自主探索建图功能

2.优化导航UI界面

下午

1.在真实机器人上完成快速随机搜索树自主探索建图功能

2.优化导航UI界面

**·Day 09**

上午

1.分别完成导航和自主建图的move\_base参数的配置

下午

1.完成rviz中marker标记

**·Day 10**

上午

1.将marker功能添加到真实机器人上，并完成各项参数优化。

下午

1.完成代码整理，说明文档撰写。

# 5.项目所遇到问题，原因分析，处理措施、经验总结

**问题1：如何设计人机交互界面接口**

·原因分析：没有设计人机交互界面的知识，工作无法展开

·处理措施：参照《ROS机器人开发实践》第12章12.2plugin，学习并了解UI界面设计流程，并设计出一个简单的单点导航插件。参照rviz源码，学习并了解tools的设计流程，并设计出一个符合导航需求的tool。插件是通过继承父类创建一个类的方式来生成，一个插件需要一个头文件和一个CPP文件，本项目中通过继承rviz::panel和rviz::tool父类的方式生成一个tool插件和panel插件，函数中的操作均在类的构造函数中完成，整个CPP文件中不能出现int main()主函数，在头文件中需要声明出变量和函数，在CPP中定义函数，在构造函数中每触发一次信号，将会执行与信号连接的函数，如果函数执行时间较长，会使构造函数无法结束，从而导致rviz无响应，直到函数执行结束为止，编写完成后需要写一个XML文件用以加载插件，并在package.xml中的export中加载该XML文件，在启动rviz时，插件便会加载成功，只需要在使用时添加。

**问题2：如何完成寻墙检测建图**

·原因分析：没有寻墙检测可行方案

·处理措施：通过查询资料，找到基于rrt的寻墙检测建图的实施方案，对该方案进行修改，使本项目机器人能够实现寻墙检测建图功能。首先运行rrt的演示demo，通过查看topic，发现该功能包是用于多机器人的寻墙检测，发现该功能包接口并不是通用型，我们需要根据自身机器人接口去修改rrt源码接口来实现rrt与机器人通信，rrt是通过订阅map通过规划，过滤等一系列数据处理得到可执行导航点并发布到move\_base，使机器人完成寻墙检测建图。当rrt无法过滤出有效点时，便会停止探索，完成建图。

**项目要点：**

**1.导航点的确定和发布，分为以下三种方式：**

（1）通过监听2D NAV GOAL的话题，可得到导航点的位姿信息，将位姿信息写入函数当中进行导航

（2）通过yaml配置文件将导航点位姿信息导入导航函数中进行导航

（3）通过编写一个rviz tool，发布导航点位姿信息，再编写一个函数用以接收导航点位姿信息进行导航。

**2.初始化位姿**

由于建图时机器人的起点与导航时机器人的起点不符，导致每次开始导航时，都需要重新设定机器人起始位姿。因此我们将起始点位姿记录下来，发布到initialpose话题中，实现自动初始化位姿。

**3.rviz中实时显示导航点**

rviz地图中无法显示导航点，我们无法清晰了解到机器人导航任务的导航点位置，所以需要将地图中的导航点以标记的形式实时显示在地图当中。编写一个函数向marker发布导航点位置标记，从而在地图中实时显示导航点位置。

**项目经验总结：**

团队成员在工作中要紧密结合，畅通信息，切实做到精诚合作，在整个项目工作中各成员确保以完成任务为目标，根据各自职责，兼顾各个环节，加强信息沟通，形成工作联动，在完成必选任务外，积极探索可选任务，在完成任务目标的同时，也要确保任务完成质量，在团队成员均无法解决某一问题时，开会讨论，通过多种途径，分工查询相关资料并汇总，商讨出解决方案。

# 6.展望和不足之处

**不足之处：**

（1）.由于机器人导航参数无法适应每一个环境，在不同环境中机器人的导航和探索的性能有优劣之分。在不同环境中要配置不同的导航参数。

（2）.ui界面的优化不足，很多地方还要加强优化，使操作更加简洁明了。

（3）.由于数据量太大，会导致远程界面卡死，无法完成远程操作控制，需要加强数据通信的速度。

（4）.在机器人上加入深度相机用以加强避障能力时，需要处理的数据量过大，CPU达不到需要的运算处理能力。

**展望：**

（1）.可以在机器人的控制方式上加入语音控制，以便于加强人机交互。

（2）.可以在机器人的功能中加入视觉能力，以便于更多功能的开发。

（3）.可以在机器人中加入机器学习，加强机器人自适应能力。

（4）.可以在机器人上加入机械臂，来丰富机器人的功能。