

Turtlebot3 SLAM与导航虚拟仿真实验-课前准备

注意

- 此次实验内容相对较多，为避免实验环境的配置和调试占用宝贵的课上时间，请同学们在实验课之前完成下面的各项准备工作，预计耗时20~45分钟，**不要留到实验课上才开始配置环境。**
- 本次实验课适用ROS系统版本为 `melodic` 和 `kinetic`。如果有同学使用其他版本ROS系统而在实验过程中出现问题，恐怕无法给予帮助。
- 仿真实验和实际环境实验有较大不同，本实验仿真运行效果也许不错，但实际环境中要差很多。
- 从pdf文件复制终端命令的时候会将行末的回车换行符一并复制，可先将复制内容粘贴到一文本文件中，删除行尾换行符后再复制、粘贴到终端中执行。

目录

Turtlebot3 SLAM与导航虚拟仿真实验-课前准备

- 一、Gazebo模型文件
- 二、依赖项安装
 - 2.1 Turtlebot3 机器人依赖项
 - 2.2 激光SLAM算法依赖项
 - 2.3 安装ROS导航包和DWA局部路径规划器
 - 2.4 安装 Octomap 功能包和 rviz 插件
 - 2.5 安装性能监视工具
 - 2.6 安装代码版本管理工具Git
- 三、验证实验环境
 - 3.1 验证 Turtlebot3 功能包
 - 3.2 验证SLAM功能包
 - 3.3 验证 Octomap 功能包
 - 3.4 关于计算机性能
- 四、实验内容预览

一、Gazebo模型文件

本次实验课中需要用到Gazebo仿真器的一些事先构建好的模型，默认情况下Gazebo将会联网加载；但由于服务器在国外，国内访问速度不佳，通常我们提前将常用的模型文件下载下来，减少Gazebo启动时联网搜索的等待时间。有网络条件的同学可以直接单击[这里](#)下载模型文件。如果无法下载或下载速度过慢，我已经将模型文件的压缩包上传到“奶牛快传”网站，由于单个链接下载次数有限制（40个），我多设置了几个链接，请大家随意点击下面三个链接中的任何一个进行下载，不要重复下载。三个链接将于2020年6月12日过期，如果届时同学有需要，或者三个链接都已经无法下载了，可以直接私聊我。

- [下载链接1](#)
- [下载链接2](#)
- [下载链接3](#)

下载完成后文件 `osrf-gazebo_models-cda67a315660.zip` 解压至任意目录（如 `~/model_file`），然后执行如下操作将模型文件安装到Gazebo的默认搜索路径中：

```
$ cd ~/model_file/osrf-gazebo_models-e6d645674e8a
$ mkdir build && cd build
$ cmake -DCMAKE_INSTALL_PREFIX=~/.gazebo ..
$ make install
```

这个模型文件包中已经包含了绝大多数常用的Gazebo模型，实验时为了节省时间，我们将禁止Gazebo运行期间链接模型服务器，执行如下命令打开Gazebo配置文件：

```
gedit ~/.ignition/fuel/config.yaml
```

将文件修改为如下内容并保存：

```
---
# The list of servers.
# servers:
-
  # name: osrf
  # url: https://api.ignitionrobotics.org

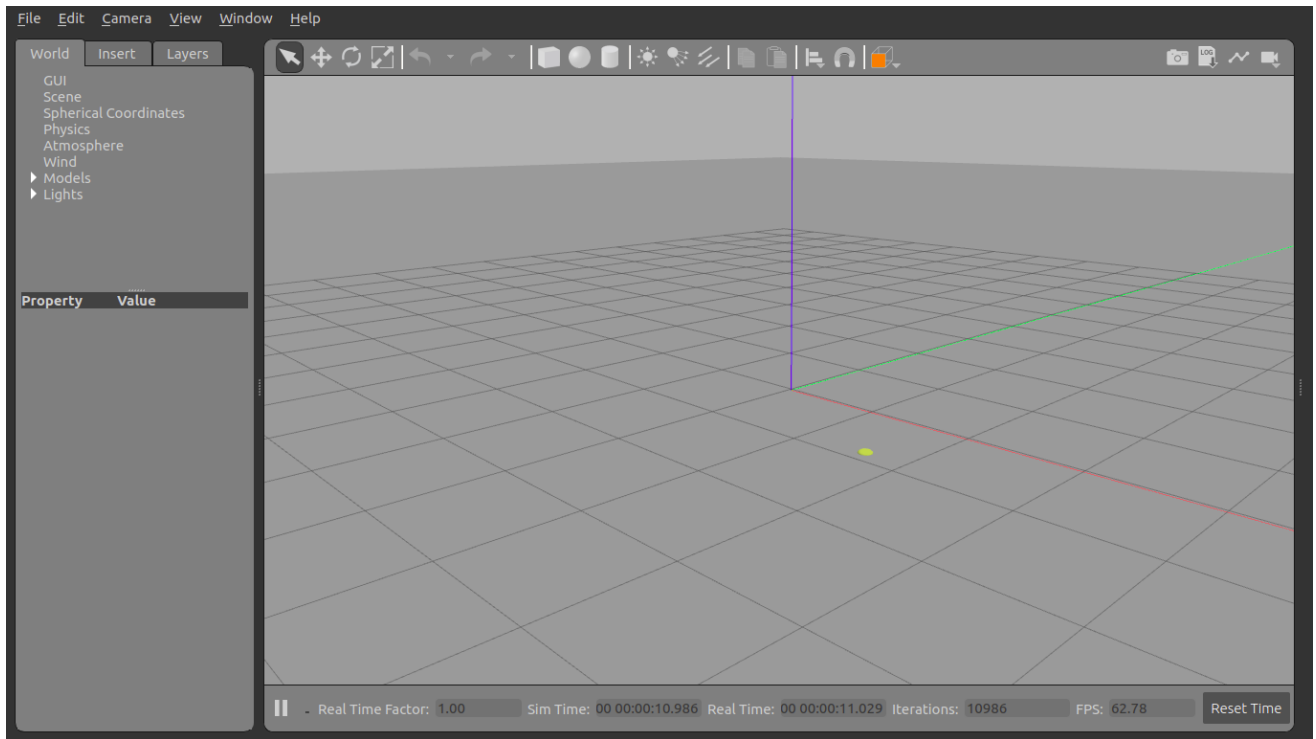
# -
#   # name: another_server
#   # url: https://myserver

# Where are the assets stored in disk.
# cache:
#   path: /tmp/ignition/fuel
```

接着运行Gazebo进行测试：

```
gazebo
```

打开过程可能需要耗时几分钟，期间gazebo可能黑屏显示；如果十多分钟后依旧没有反应，可重新启动几次。如果gazebo呈现下图所示画面，则说明gazebo模型设置成功。



二、依赖项安装

2.1 Turtlebot3 机器人依赖项

本次仿真实验需要安装Turtlebot3机器人的功能包：

```
$ sudo apt-get install ros-${ROS_DISTRO}-turtlebot3 ros-${ROS_DISTRO}-turtlebot3-  
description ros-${ROS_DISTRO}-turtlebot3-gazebo ros-${ROS_DISTRO}-turtlebot3-msgs  
ros-${ROS_DISTRO}-turtlebot3-slam ros-${ROS_DISTRO}-turtlebot3-teleop
```

命令中的 `${ROS_DISTRO}` 在执行时将会自动替换成电脑安装的ROS版本号 `melodic` 或 `kinetic`。各个功能包的大致作用如下：

- turtlebot3：提供turtlebot3机器人的基本控制逻辑
- turtlebot3-description：提供turtlebot3机器人的描述文件，用于Gazebo仿真和rviz可视化
- turtlebot3-gazebo：提供和Gazebo仿真支持
- turtlebot3-msgs：提供turtlebot3机器人的自定义消息类型
- turtlebot3-slam：提供turtlebot3机器人的SLAM算法支持
- turtlebot3-teleop：提供turtlebot3机器人的键盘控制支持

上述功能包在本次实验课中都讲使用到，同学们可以提前了解一下。

接着通过设置环境变量指定使用的Turtlebot3机器人模型：

```
$ echo export TURTLEBOT3_MODEL=waffle >> ~/.bashrc
```

Turtlebot3 系列共有三款机器人，本次实验中使用 `waffle`，感兴趣的同学可以单击[这里](#)查看TurtleBot3机器人的简介。

2.2 激光SLAM算法依赖项

```
$ sudo apt-get install ros-${ROS_DISTRO}-slam-karto
```

和课本中使用的 gmapping 建图方法不同，为了实现较好的建图效果、尽可能保证同学们实验过程中心情舒畅，本次实验课使用效果更好的 karto SLAM算法。

2.3 安装ROS导航包和DWA局部路径规划器

```
$ sudo apt-get install ros-${ROS_DISTRO}-navigation ros-${ROS_DISTRO}-dwa-local-planner
```

2.4 安装 Octomap 功能包和 rviz 插件

```
$ sudo apt-get install ros-${ROS_DISTRO}-octomap-ros ros-${ROS_DISTRO}-octomap-msgs  
ros-${ROS_DISTRO}-octomap-server ros-${ROS_DISTRO}-octomap-rviz-plugins
```

Octomap用于构建三维的八叉树地图，在拓展实验二中将用到。

2.5 安装性能监视工具

```
$ sudo apt-get install htop
```

用于接下来的实验环境验证。

2.6 安装代码版本管理工具Git

```
$ sudo apt-get install git
```

本次实验中许多代码需要在上一步的基础上进行修改，为了方便代码管理，在撰写实验报告时方便切换不同版本代码实现不同阶段的实验效果，建议安装git。

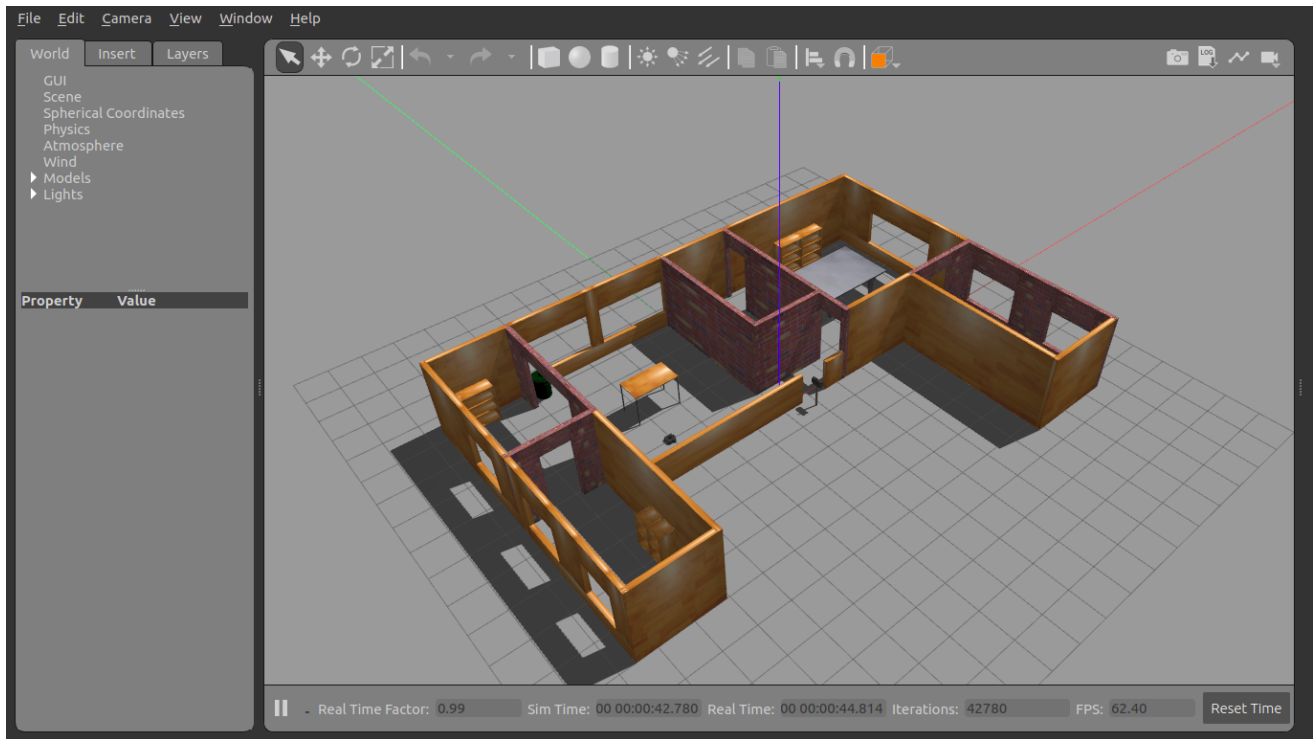
三、验证实验环境

3.1 验证 Turtlebot3 功能包

执行：

```
$ roslaunch turtlebot3_gazebo turtlebot3_house.launch
```

稍等片刻，这会启动事先搭建好的虚拟环境。如果终端无报错、gazebo可以正确打开并显示如下图所示的模型，则说明Turtlebot3功能包工作正常。

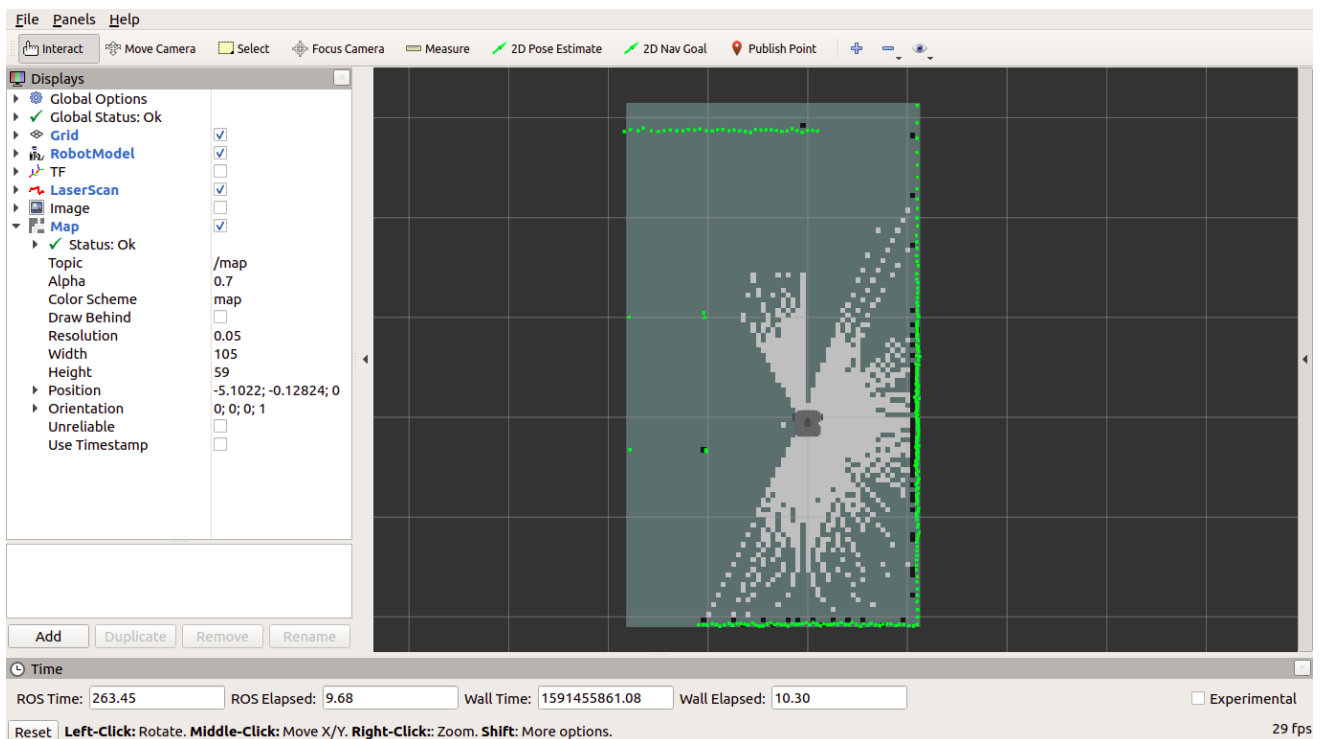


3.2 验证SLAM功能包

暂时不要关闭gazebo，新开终端，输入以下命令：

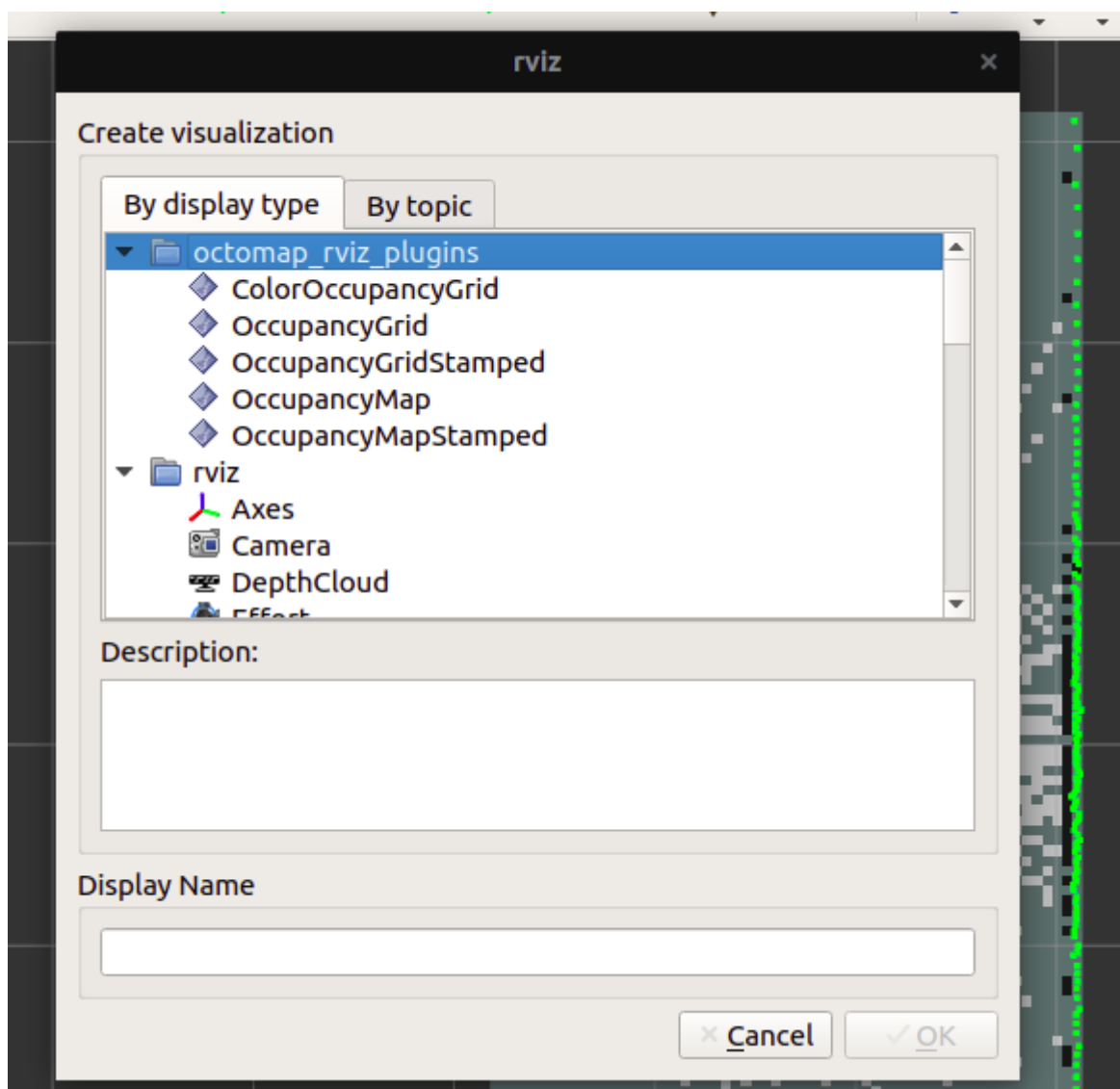
```
$ roslaunch turtlebot3_slam turtlebot3_slam.launch slam_methods:=karto
```

稍等片刻，这将会打开rviz。如果显示为如下的画面，则SLAM功能包工作正常。



3.3 验证 Octomap 功能包

保持上一步的rviz窗口不关闭，单击左下方的 `Add` 按钮，在弹出的对话框中如果能够看到带有octomap字样的插件，则说明Octomap功能包正确安装。



3.4 关于计算机性能

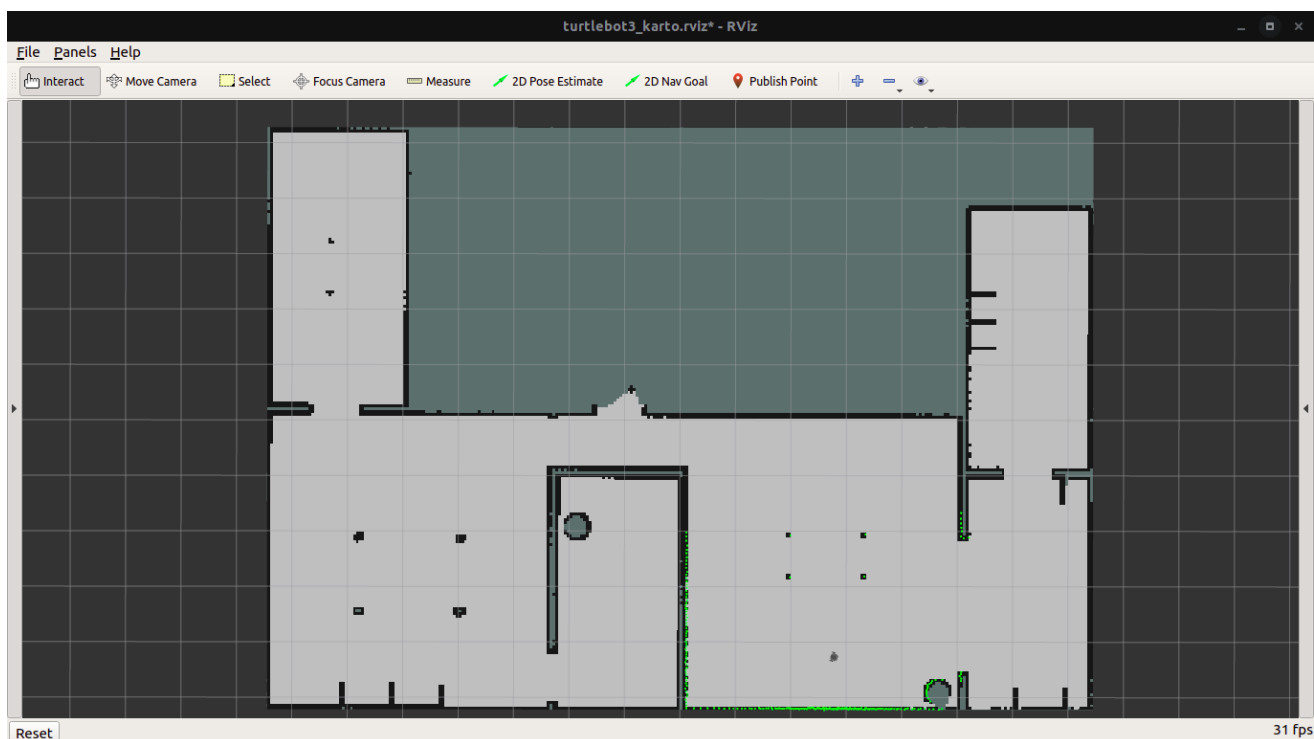
由于Gazebo计算需求较大，对使用虚拟机的同学不太友好。在仅开启上述窗口时，拖动Gazebo中的环境模型和rviz中的图像应当没有卡顿，建议各组至少有一名同学的计算机可以满足以上条件。此外，在终端中直接运行 `htop` 可以查看当前系统资源占用情况：

```
$ htop
```

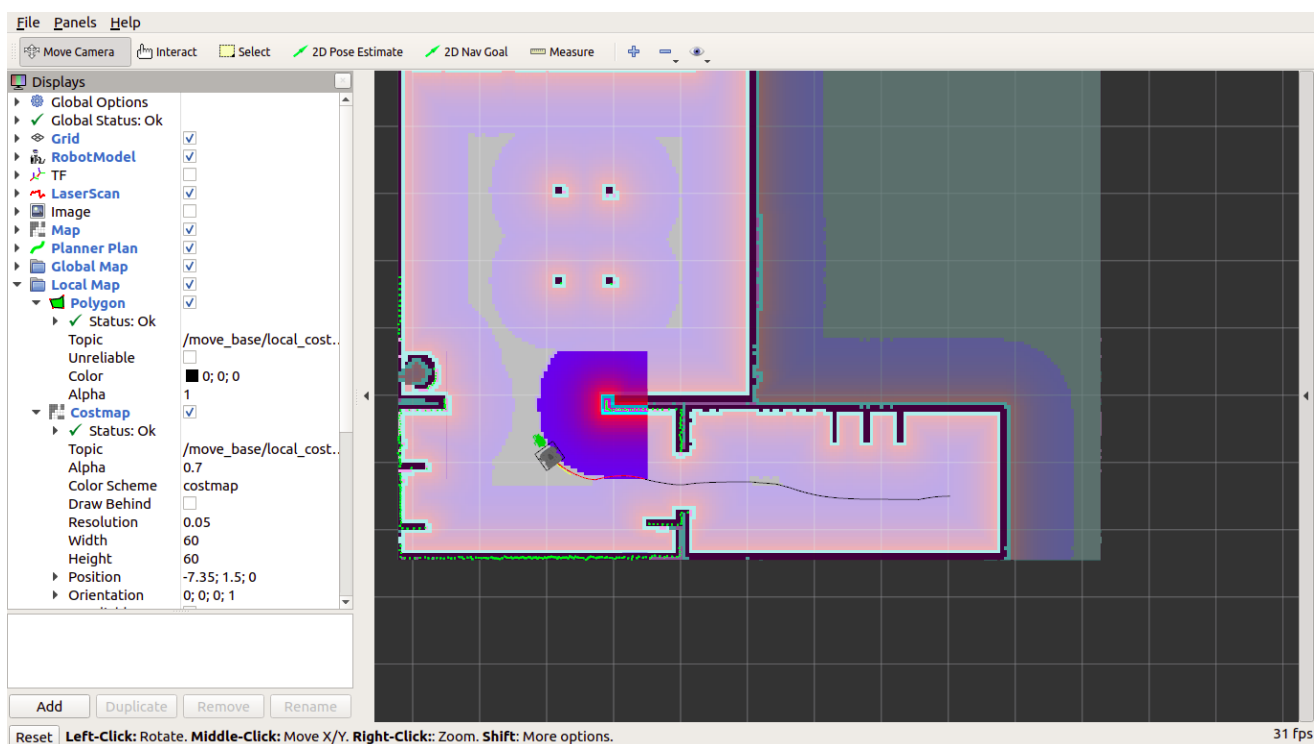
若所有CPU逻辑核心占用率均达95%以上，或内存剩余少于1GB，则只能完成基本实验和拓展实验一，拓展实验二可能无法流畅进行。

四、实验内容预览

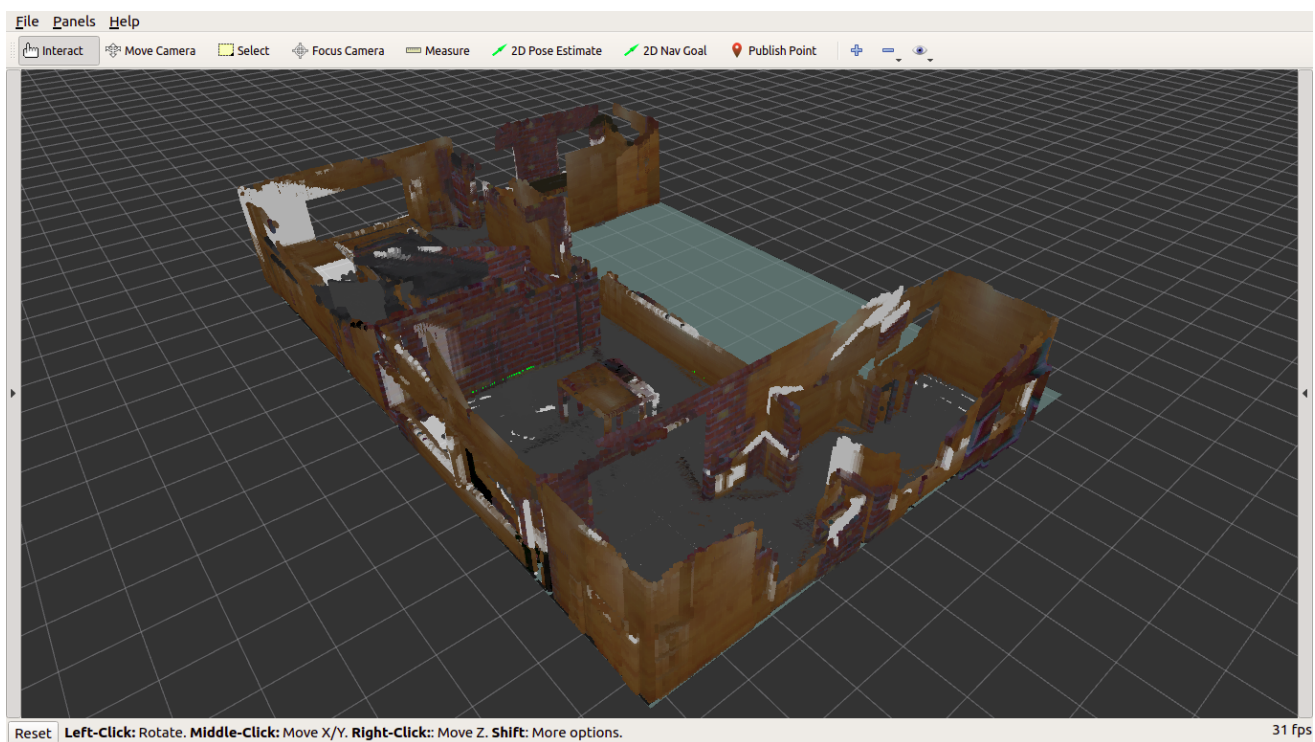
本次实验主要由基本实验、拓展实验一和拓展实验二组成，难度递增。基本实验的内容与课本中的内容类似，通过控制机器人在虚拟环境中运动，利用激光雷达数据构建房间地图，并利用该地图进行导航，实验过程通过调用现有的功能包实现。下图为构建地图的效果。



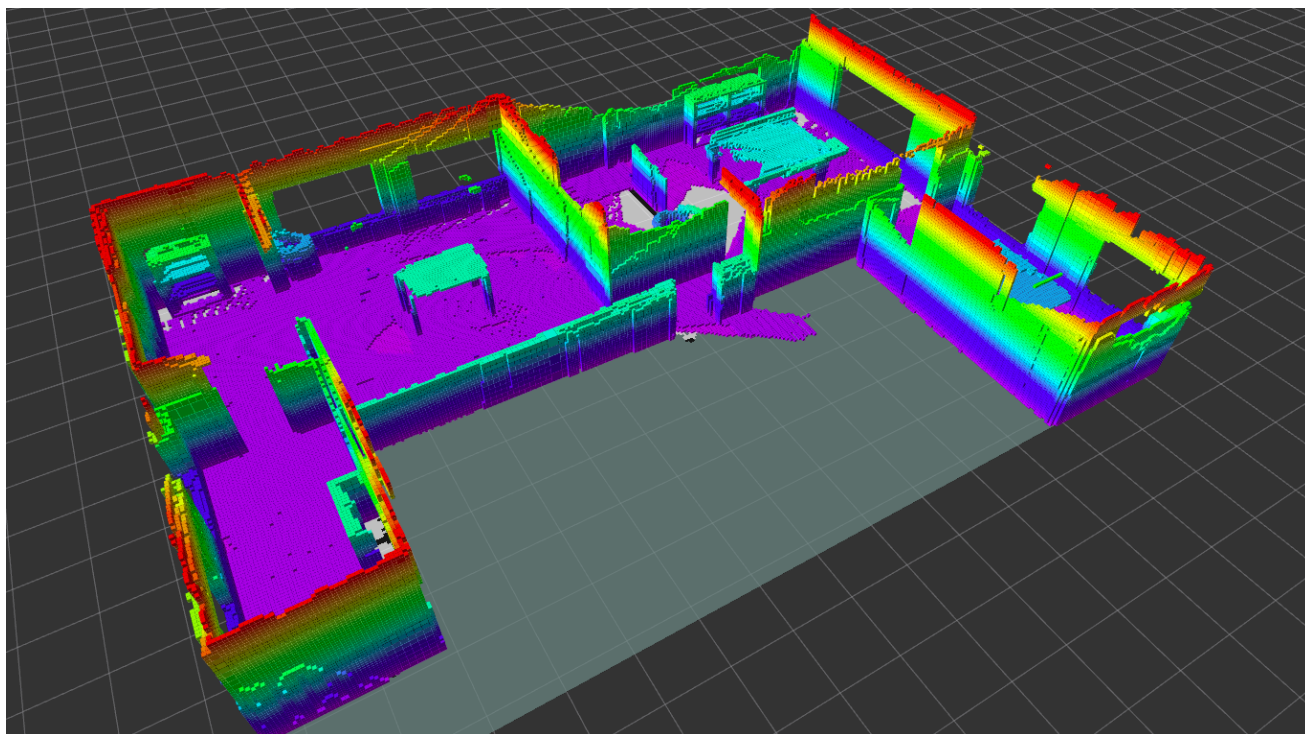
拓展实验一为高级导航实验，实验需要同学们补充编写代码，实现使用程序设置机器人初始位姿、设定导航目标点、获取导航任务执行状态等，最终实现从指定文件中读取一系列的路径点，使机器人顺序到达的任务，如下图所示。



拓展实验二为三维建图实验，实验需要同学们了解Turtlebot3机器人的TF坐标变换组成和针孔相机成像原理，通过补充编写代码，实现通过topic获取和发布图像、计算点云，最终构建点云地图和八叉树地图，最终效果如下图所示。



点云地图构建效果



八叉树地图构建效果

实验过程中有问题在群里讨论

刘国庆 2020.06.06