

# 智能机器人实验

——SLAM与自主导航实践

李华龙

## 1. 初识ROS

- ROS简介
- ROS的安装
- 第1个ROS例程
- ROS总体架构

## 2. ROS基础

- ROS工作空间
- ROS通信编程
- ROS分布式通信
- ROS关键组件

## 3. 建模与仿真

- 机器人系统简介
- 机器人URDF建模
- 机器人模型优化
- Gazebo物理仿真

## 4. SLAM与自主导航仿真

- 相关基础概念介绍
- SLAM功能包的应用
- ROS中的导航框架
- 综合仿真实现

## 5. SLAM与自主导航实践

- 机器人实验平台了解与使用
- 真实机器人SLAM测试与验证
- 真实机器人自主导航实现
- 综合测试与验证

# 内容概览

1

机器人实验平台了解与使用

2

真实机器人SLAM测试与验证

3

真实机器人自主导航实现

4

综合测试与验证

# 1 机器人实验平台了解与使用

## Turtlebot 4 Lite 版<sup>1</sup>

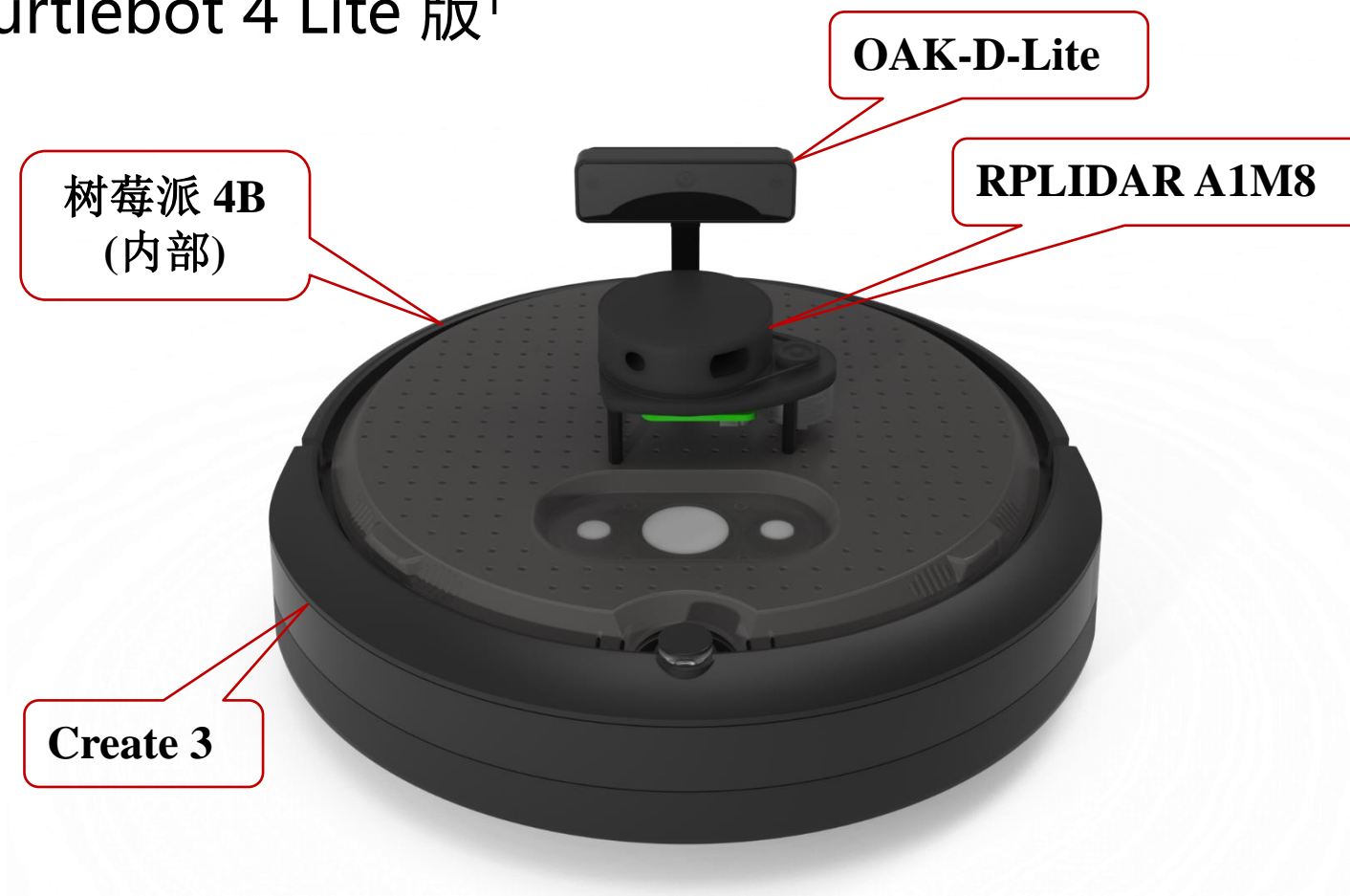


1. <https://turtlebot.github.io/turtlebot4-user-manual/overview/features.html>

## Turtlebot 4 Lite 版

- ◆ 一款基于 ROS2 的移动机器人，主要用于教育和科研
- ◆ 可对周围环境建图，自主导航，相机上可运行 AI 模型
- ◆ 基于 Create3 平台<sup>1,2</sup>,
- ◆ 主要组成部分
  - 激光雷达：RPLIDAR A1M8<sup>3</sup>
  - 深度相机：OAK-D-Lite<sup>4</sup>
  - 处理单元：树莓派 4B
  - 底盘平台：Create 3

## Turtlebot 4 Lite 版<sup>1</sup>



## Create 3<sup>1, 2</sup>





## Create 3<sup>1</sup>, 2



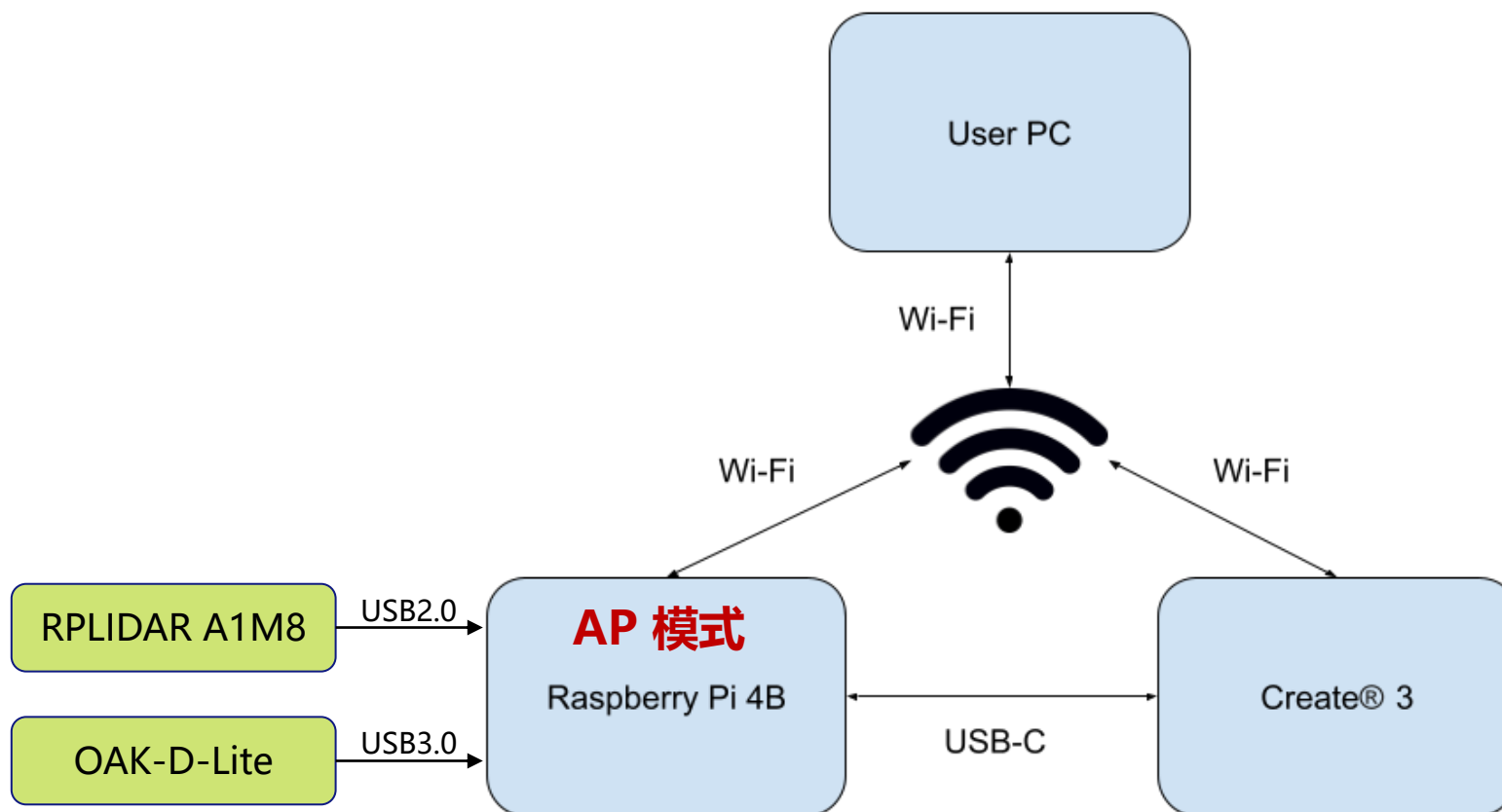
1. <https://edu.irobot.com/what-we-offer/create3>
2. [https://iroboteducation.github.io/create3\\_docs/](https://iroboteducation.github.io/create3_docs/)

## Create 3<sup>1, 2</sup>



1. <https://edu.irobot.com/what-we-offer/create3>
2. [https://iroboteducation.github.io/create3\\_docs/](https://iroboteducation.github.io/create3_docs/)

## Turtlebot 4 Lite 版<sup>1</sup>



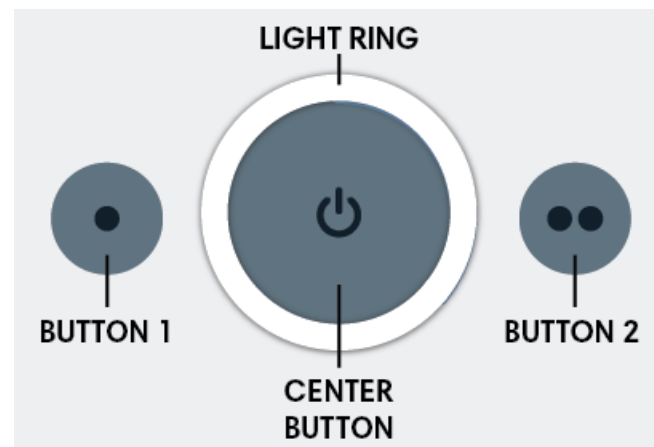
1. <https://turtlebot.github.io/turtlebot4-user-manual/overview/features.html>

## Turtlebot 4 Lite<sup>1</sup> 使用

**开机：**将机器人放置在充电座上，**耐心**等待光环停止闪烁，并播放提示音即可

**关机：**将机器人从充电座上移开，按住电源按钮至灯环闪烁，并播放提示音

**提示：**随时、及时充电



## Turtlebot 4 Lite 使用

◆ 将机器人放置在充电座上等待开机

◆ 笔记本电脑配置

□ 先设置 root 密码，若已设置过，则跳过此步

```
$ sudo passwd
```

□ 切换至 root 用户，按提示输入密码

```
$ su root
```

□ 修改 /etc/hostname 文件，将原主机名改为：ubt

```
$ gedit /etc/hostname
```

□ 修改 /etc/hosts 文件，将原主机名改为：ubt，并添加：10.42.0.1 ubuntu

```
$ gedit /etc/hosts
```

□ 将电脑 WiFi 连至 TB4 的热点 Turtlebot4-xxxx，设置固定 IP，重启电脑

IP: 10.42.0.100    Netmask: 255.255.255.0    Gateway: 10.42.0.1

注：密码与SSID相同，其中的“xxxx”为机器人序列号后4位，不要弄错

## Turtlebot 4 Lite 使用

Cancel Wired Apply

Details Identity **IPv4** IPv6 Security

**IPv4 Method**

☐ Automatic (DHCP) ☐ Link-Local Only

☒ Manual ☐ Disable

**Addresses**

Address	Netmask	Gateway	
10.42.0.100	255.255.255.0	10.42.0.1	✕
			✕

**DNS** Automatic **ON**

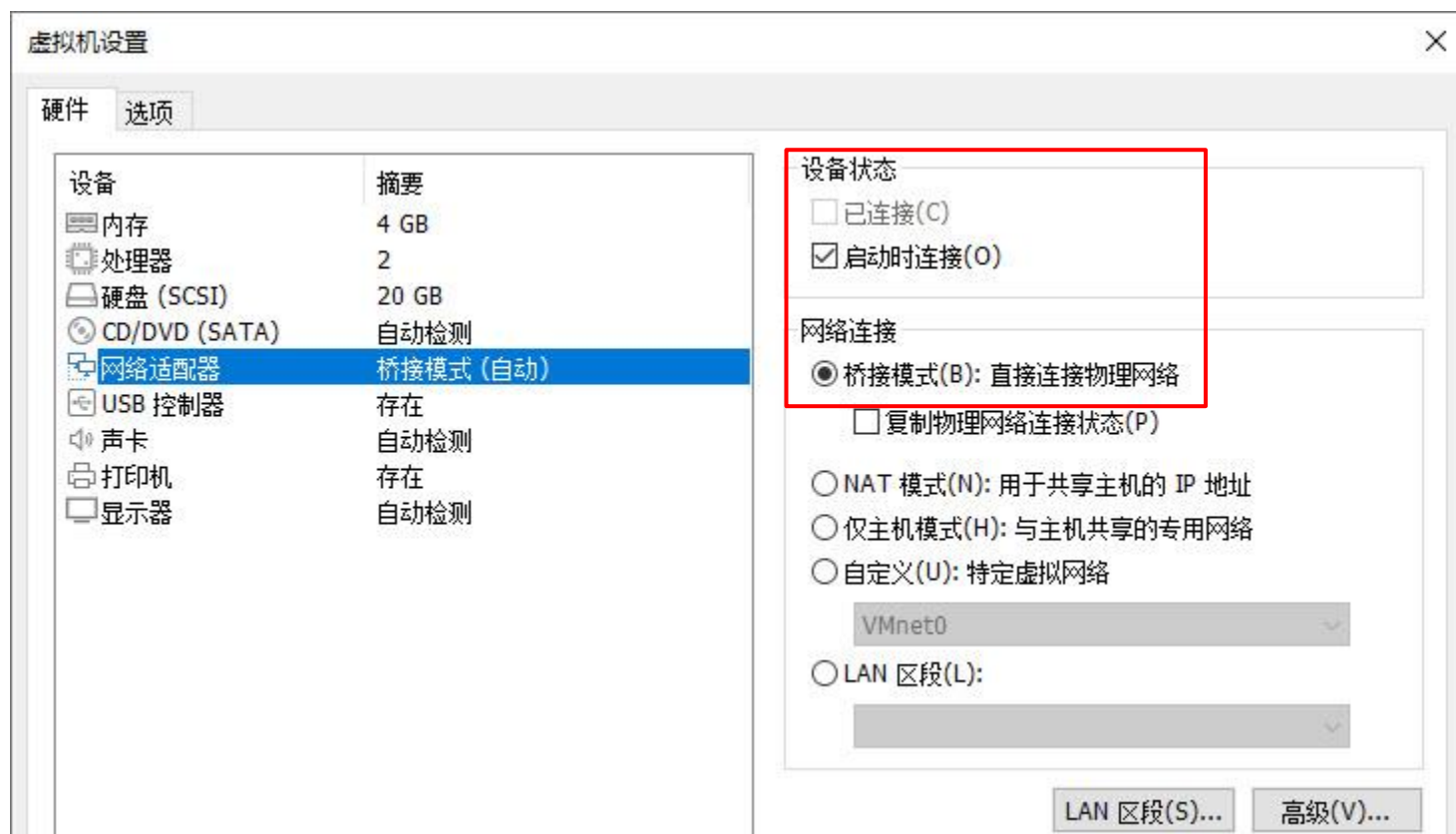
Separate IP addresses with commas

**Routes** Automatic **ON**

Address	Netmask	Gateway	Metric	
				✕

## Turtlebot 4 Lite 使用

### ◆ 虚拟机的配置（如用）



## roscore

```
roscore http://ubt:11311/

File Edit View Search Terminal Help
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://ubt:43751/
ros_comm version 1.14.13

SUMMARY
=====

PARAMETERS
* /rostdistro: melodic
* /rosversion: 1.14.13

NODES

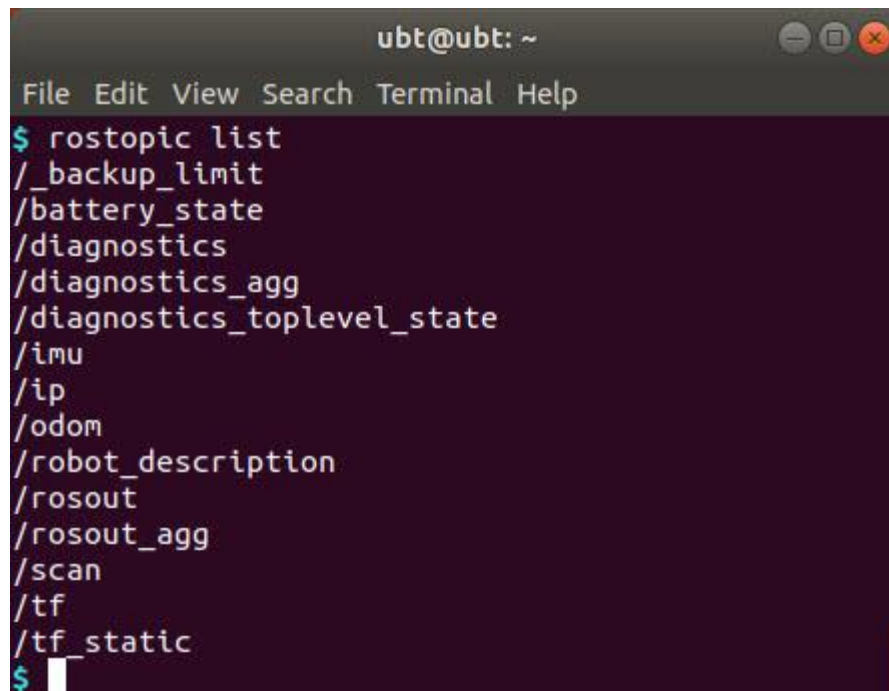
auto-starting new master
process[master]: started with pid [2950]
ROS_MASTER_URI=http://ubt:11311/

setting /run_id to b7bb4c4c-6581-11ed-8fe0-000c2979b29d
process[rosout-1]: started with pid [2961]
started core service [/rosout]
```

提示：为避免程序停止运行后，需重启机器人，可一直维持 roscore 运行



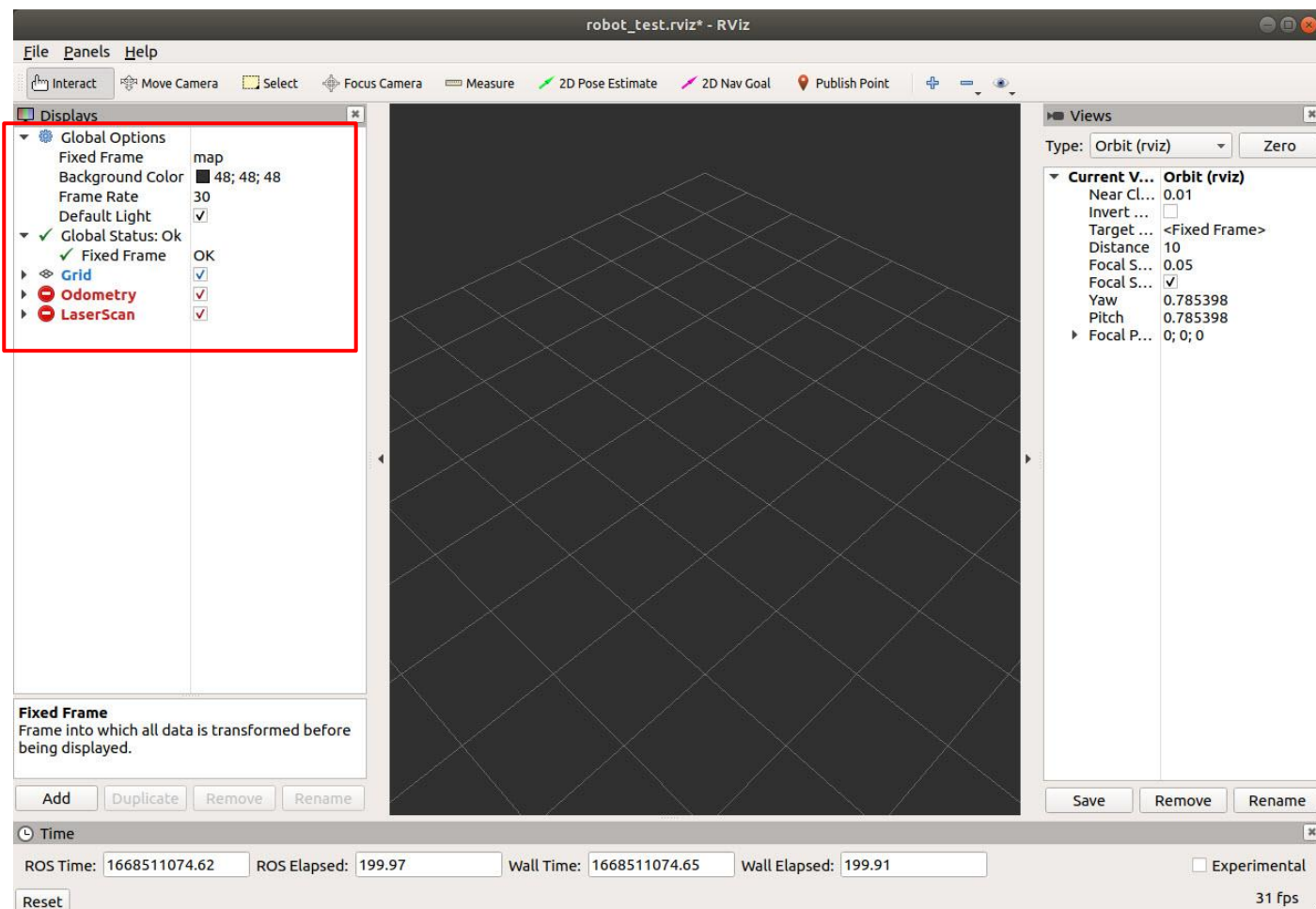
话题列表（roscore 后直接查看）



```
ubt@ubt: ~  
File Edit View Search Terminal Help  
$ rostopic list  
/_backup_limit  
/battery_state  
/diagnostics  
/diagnostics_agg  
/diagnostics_toplevel_state  
/imu  
/ip  
/odom  
/robot_description  
/rosout  
/rosout_agg  
/scan  
/tf  
/tf_static  
$
```

注：只显示了机器人发布的

## 查看里程计和激光雷达数据 (RViz)



## 添加缺少的静态坐标变换 (launch文件)

```
<launch>
```

```
<!--static_transform_publisher x y z yaw pitch roll frame_id child_frame_id-->  
<node pkg="tf2_ros" type="static_transform_publisher" name="base_link_rplidar_link_broadcaster"  
      args="0.0 0.0 0.1 1.5707963267949 0.0 0.0 base_link rplidar_link"/>
```

```
<!-- 设置launch文件的参数 -->
```

```
<arg name="rvizconfig" default="$(find ros_2dnv_learning)/config/robot_test.rviz"/>
```

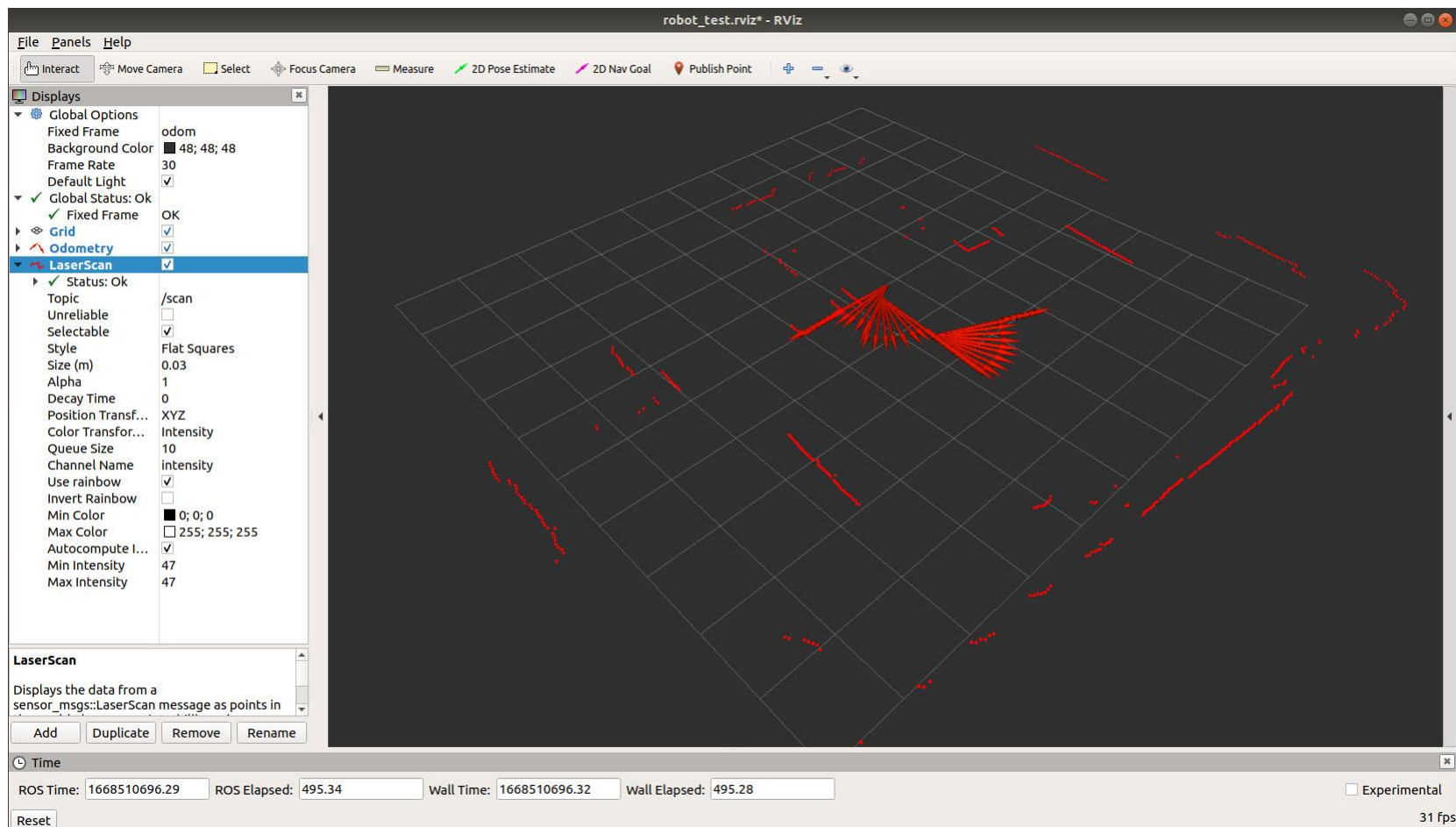
```
<node name="rviz" pkg="rviz" type="rviz" args="-d $(arg rvizconfig)"/>
```

```
<node name="teleop_twist_keyboard" pkg="teleop_twist_keyboard" type="teleop_twist_keyboard.py"/>
```

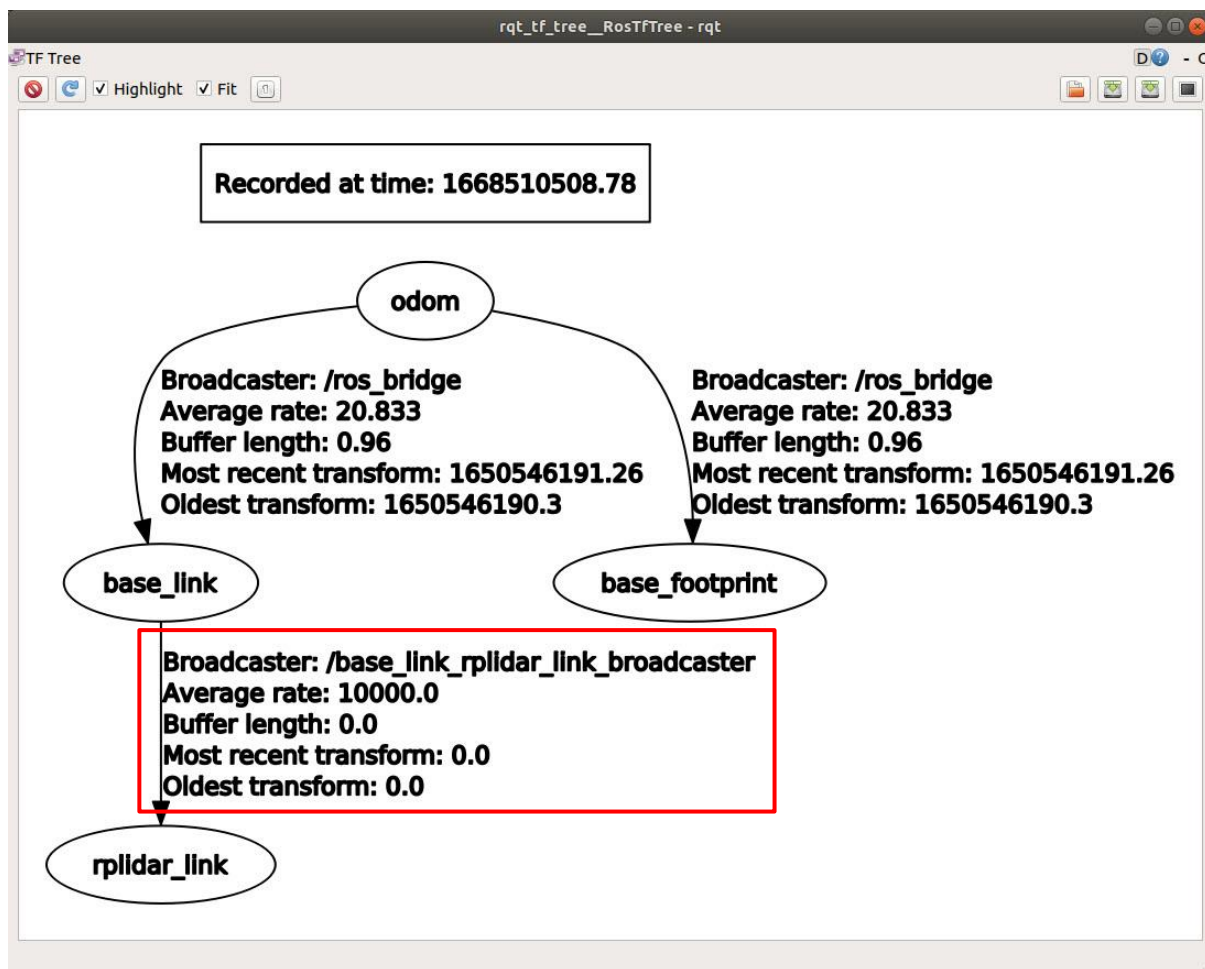
```
</launch>
```

robot\_test.launch

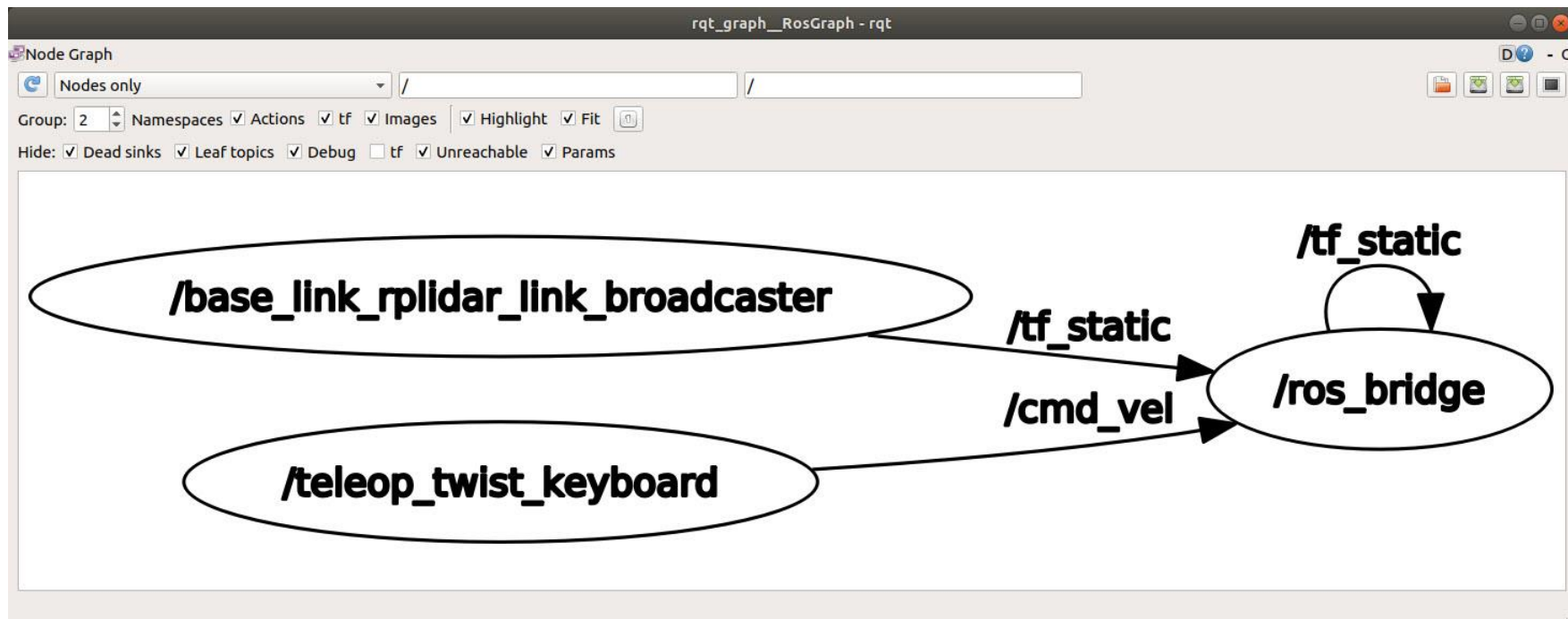
## 控制机器人运动



## 查看 TF 树



## 查看计算图



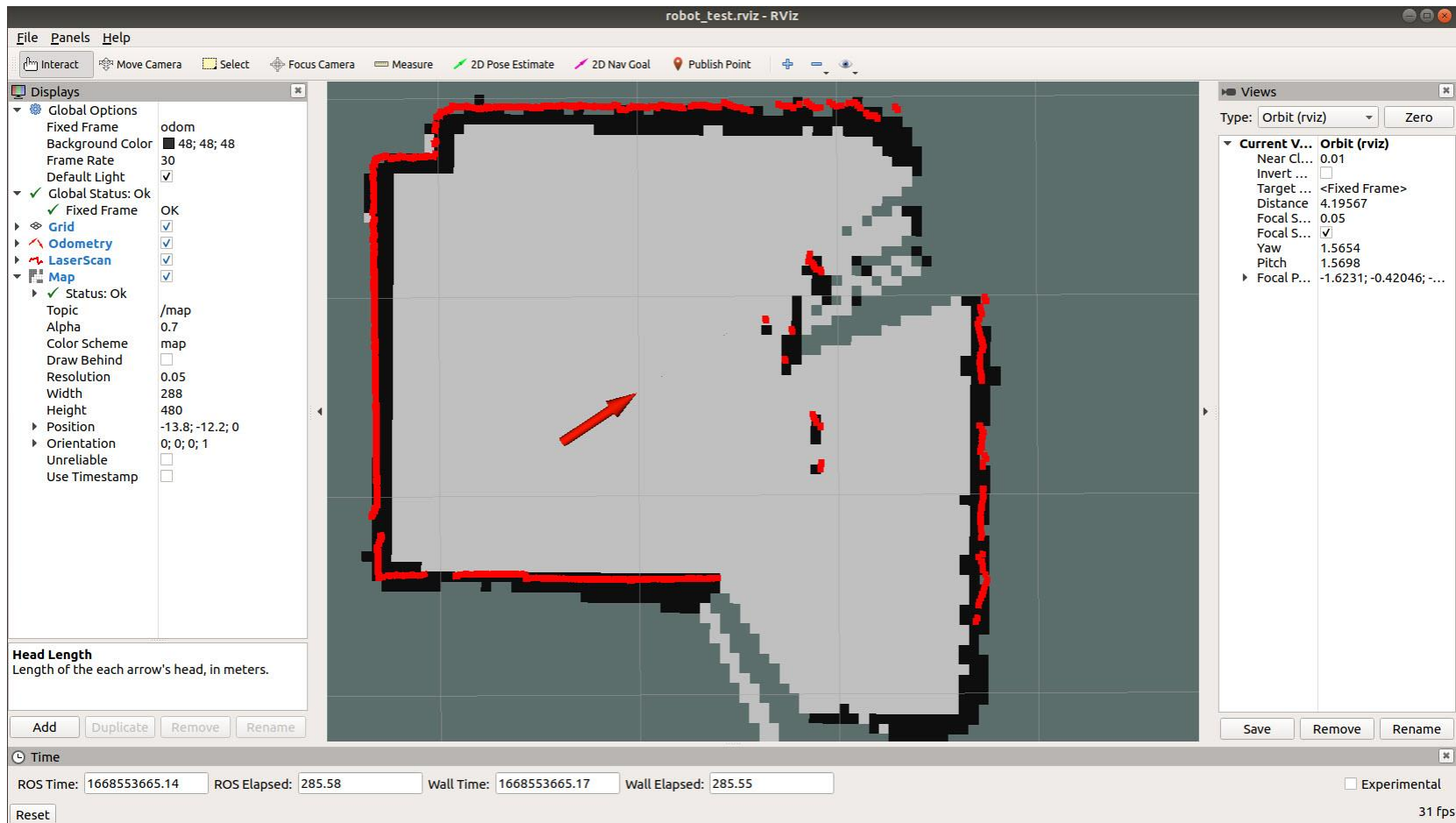
## 2 机器人SLAM测试与验证

## SLAM实测

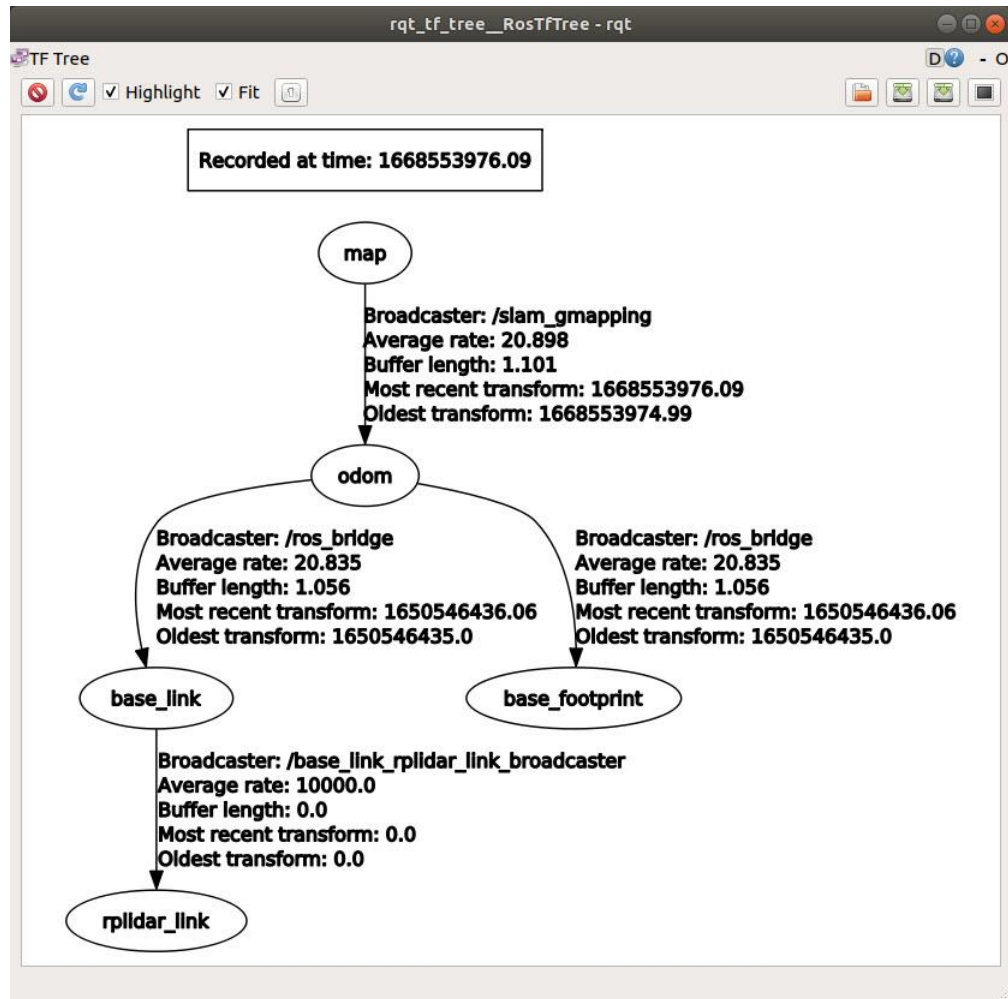
- ◆ 综合前面的仿真实验和机器人平台使用方法，编写相关 launch 文件
- ◆ 注意根据仿真、及实测结果，对**相关参数进行优化**
- ◆ 可对多种 SLAM 功能包分别进行实测对比，再与仿真效果比对
- ◆ 实测得到测试场地的二维地图，用于后面的导航测试
- ◆ 机器人不能与围挡和障碍物发生碰撞



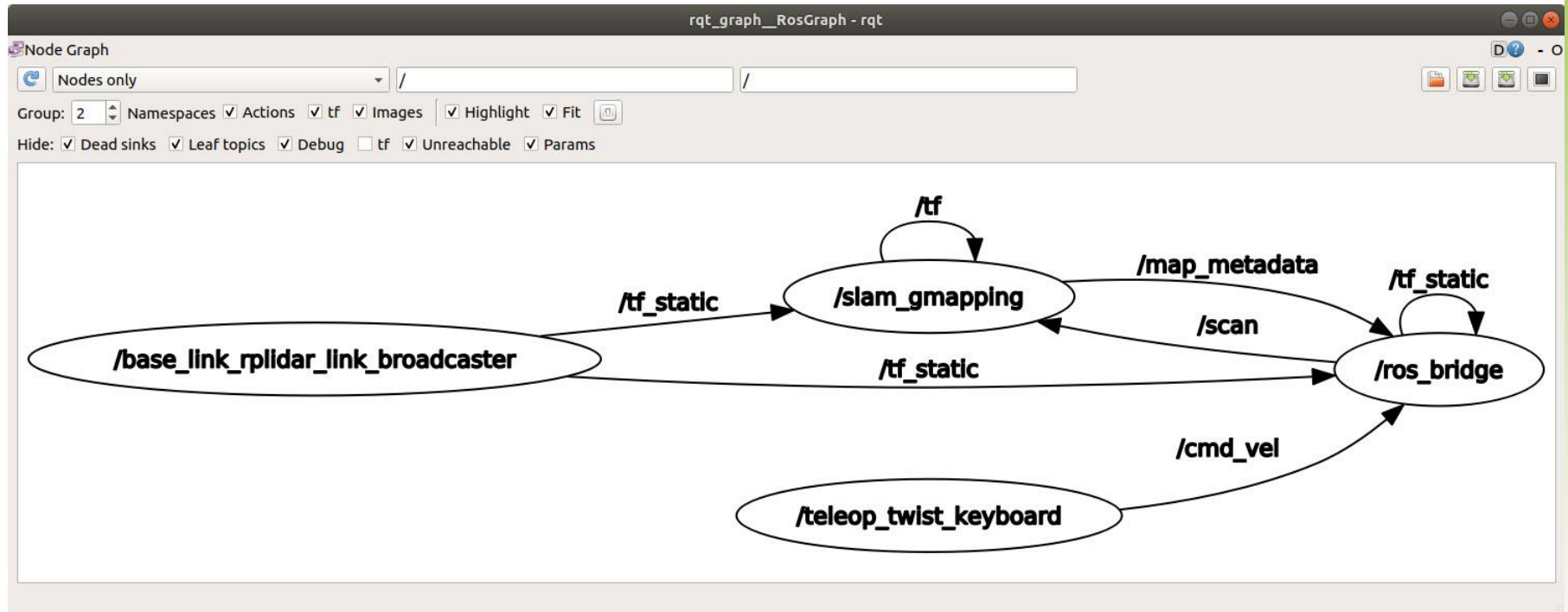
## SLAM示例 (RViz)



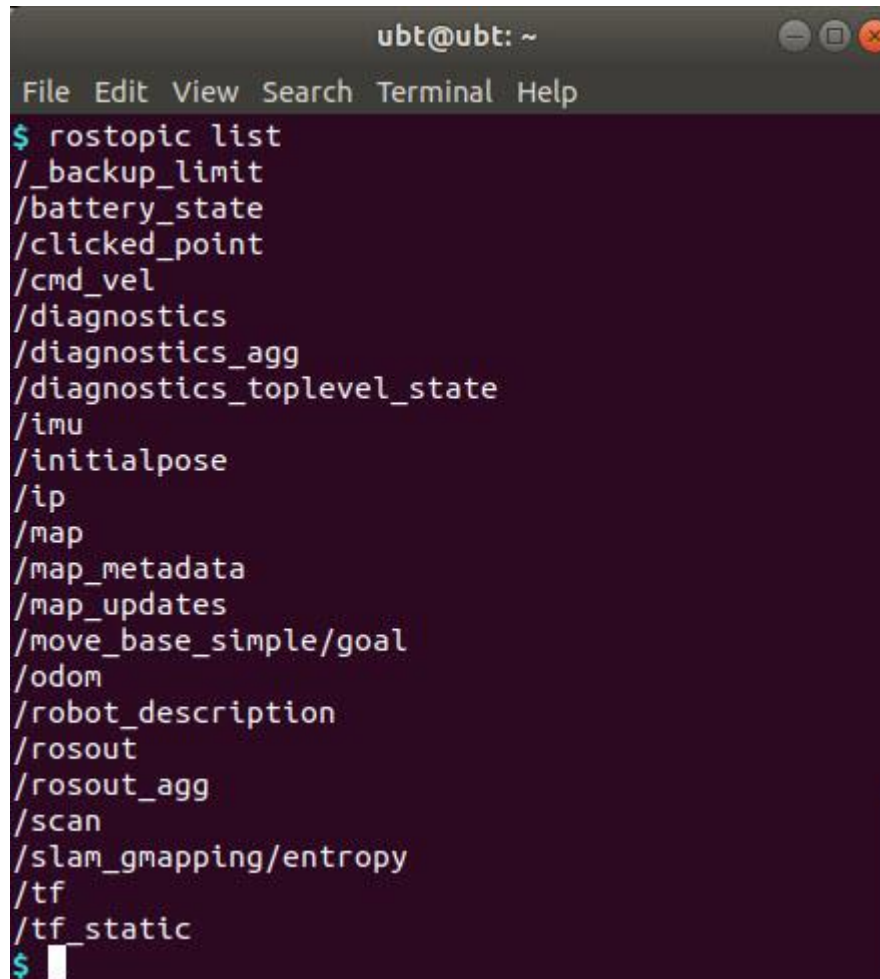
## SLAM示例 (TF 树)



## SLAM示例 (计算图)



## SLAM示例（话题列表）



```
ubt@ubt: ~  
File Edit View Search Terminal Help  
$ rostopic list  
/_backup_limit  
/battery_state  
/clicked_point  
/cmd_vel  
/diagnostics  
/diagnostics_agg  
/diagnostics_toplevel_state  
/imu  
/initialpose  
/ip  
/map  
/map_metadata  
/map_updates  
/move_base_simple/goal  
/odom  
/robot_description  
/rosout  
/rosout_agg  
/scan  
/slam_gmapping/entropy  
/tf  
/tf_static  
$
```

# 3

## 自主导航实现

## 导航实测

- ◆ 综合前面的仿真实验和机器人平台使用方法，编写相关 launch 文件
- ◆ 应用前面 SLAM 实测得到的测试场地地图
- ◆ 根据仿真结果选择合适的路径规划算法(全局，本地)进行**实测和参数优化**
- ◆ 也可对多种路径规划算法分别进行实测对比，再与仿真效果比对
- ◆ 鼓励结合机器人和测试环境，尝试对开源的路径规划算法代码进行优化
- ◆ 机器人不能与围挡和障碍物发生碰撞

# 4

## 综合测试与验证

## 测试任务（选做，占 10%）

- ◆ 实测场地+机器人实验平台+个人笔记本电脑
- ◆ 实测场地内障碍物位置、大小、数量均随机
- ◆ 机器人在**未提供地图**的情况下，从**任意位姿**出发，完成整个场地的二维建图后，**自主导航回至初始位姿**
- ◆ 整个过程中不能人为干预
- ◆ 主要评价维度：建图质量，最终位姿偏差，完成时长
- ◆ 机器人不能与围挡和障碍物发生碰撞
- ◆ 2 人一组，每次测试限时 5 分钟，每组 2 次机会



# 小结

## 1. 机器人实验平台了解与使用

- ◆ Turtlebot 4 Lite 移动机器人

## 2. 机器人SLAM测试与验证

- ◆ 编写 launch 文件，SLAM功能包的选择与使用，参数优化，算法优化

## 3. 机器人自主导航实现

- ◆ 编写 launch 文件，路径规划算法的选择与使用，参数优化，算法优化

## 4. 综合测试与验证（选做，占 10%）

- ◆ SLAM 建图 + 自主导航实测
-

## 1. 机器人实验平台使用范围

- ◆ C0105, C区1楼其他空地

## 2. 测试场地使用

- ◆ 避免拥挤, 及时清理场地和车轮

## 3. 机器人平台的使用

- ◆ 注意安全, 注意爱惜, 规范操作, 按约定时间轮流使用, 及时充电

## 4. 提前充分利用机器人建模与仿真

- ◆ 测试场地容纳数量、机器人实验平台数量、续航时间有限

## 5. 若有意见、建议及时提出

扶上马

送一程

后面看大家的了！

---

谢谢!