**环境地图**

ROS中的环境地图的概念和我们日常生活中的地图有所区别。环境地图在ROS中以2D的网格进行描述，每个网格包含了一个值，这个值为该网格被其他物体占据的可能性，而忽略了该点的其他信息，比如这个网格是一面墙还是一面桌子，是坚硬的还是柔软的等等。因为对于一般的机器人来说，它关注的是空间和物体的“存在性”，而不需要知道处于某个空间位置的物体是什么，忽略了物体的名称等“意义”。从某种意义上讲，环境地图更接近于我们在图论中讨论的抽象的图，这个和机器人使用环境地图的意义密不可分。

**创建环境地图的意义**

环境地图为机器人提供了可达性信息，告诉了机器人它周围的位置是否存在障碍物，当然这个判断是基于概率的。这些信息在机器人导航中起到了至关重要的作用，当然，这些信息产生作用还要结合其他知识和工具，比如ACML（Adaptive Monte Carlo Localization），TF（Transform）等等用于机器人自身定位的算法。

**环境地图的表示**

我们打开一个名为map.yaml的文件，该文件展示了机器人对已有的地图文件解析后的相关信息

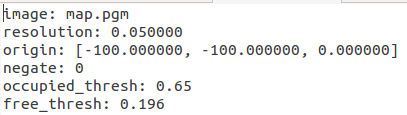


图1 map.yaml

其中origin表示原点的位置，occupied\_thresh和free\_thresh为判断每个方格是否被占据的阈值，此处的计算方法为比较灰度值和255 \* (1 – thresh)。

**使用rosbag记录数据**

使用rosbag工具可以将在某次运动中的传感器数据记录下来，并可以多次重放，以达到调试算法的目的。这里使用rosbag重放数据，以创建机器人在某次运动中的环境地图。

为了便于观察，使用机器人模拟器和可视化工具进行案例的演示。

首先启动一个机器人仿真环境，这里使用命令创建一个在Stage仿真工具中运动的机器人。

**roslaunch turtlebot\_stage turtlebot\_in\_stage.launch**

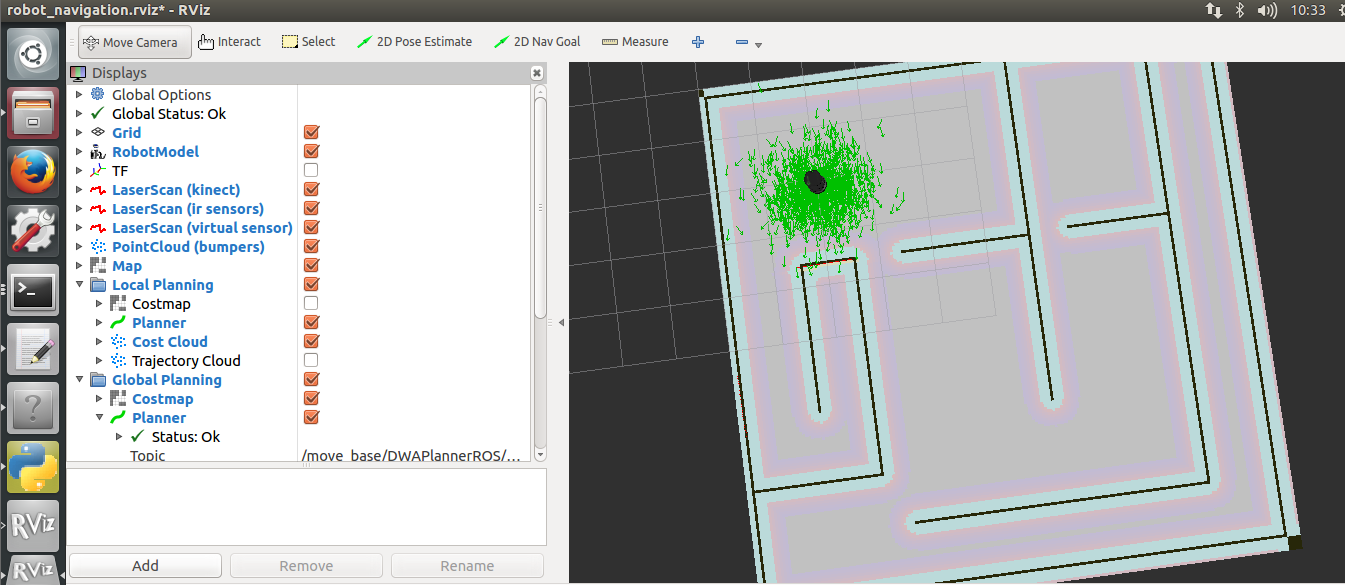


图2 使用Stage进行仿真

然后使用键盘控制机器人进行移动。

**roslaunch turtlebot\_teleop keyboard\_teleop.launch**

由于在turtlebot机器人上没有激光测距仪（仿真软件中也没有），不过该仿真软件的Kinect相机会使用深度数据提供相应的LaserScan消息（/scan话题即激光测距仪的信息）使用命令对/scan和/tf消息进行记录。

**rosbag record -O data.bag /scan /tf**

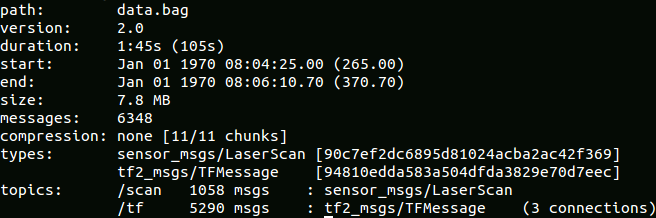


图3 data.bag信息

其间可以通过**rostopic**查看相应话题的订阅情况。

**使用gmapping建图**

在终端中运行**roscore**，在另一终端使用命令

**rosparam set usr\_sim\_time true**

**rosrun gmapping slam\_gmapping**

使用仿真记录下来的包的时间戳，并启动gmapping节点。

此时重放采集到的数据

**rosbag play –clock data.bag**

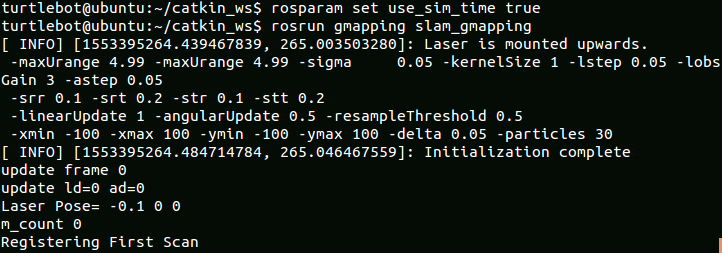


图4 slam\_gmapping的输出

直至rosbag完成消息的重放，且gmapping也不在输出诊断信息时，运行map\_saver节点保存地图

**rosrun map\_server map\_saver**

此时会存储两个文件，分别是map.pgm和map.yaml，在之前的介绍中已经对这两个文件进行了分析。

**启动地图服务器并查看地图**

此时运行命令将地图添加到ROS中，并使用rviz工具进行查看，过程中要添加/map话题信息的查看。

**rosrun map\_server map\_server map.yaml**

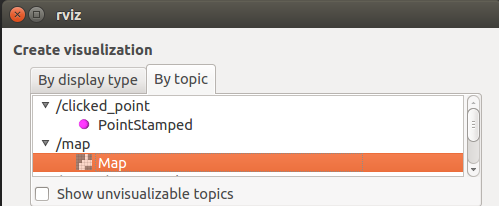


图5 在rviz中添加/map话题进行查看

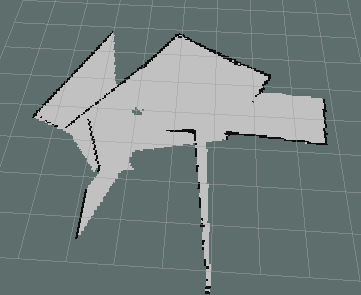


图6 重建图像

可以看出，地图重建的质量并不是很高，可以使用rosparam set对slam\_gmapping包中的参数进行重设。

此外我们还可以通过**rostopic echo /map\_metadata**查看关于/map\_metadata话题的相关数据

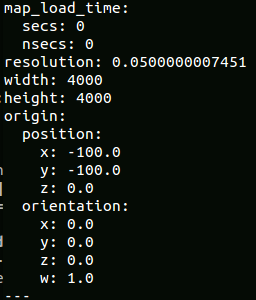


图7 /map\_metadata的信息