 

**Linux系统基础**

**实验报告**

学 号:

姓 名:

院 系:

专 业:

指导教师:

日 期:

**目 录**

[1 ROS基础实验-虚拟环境中移动机器人控制 1](#_Toc181173344)

[1.1 实验内容 1](#_Toc181173345)

[1.2 实验要求 1](#_Toc181173346)

[2 ROS感知实验-使用激光雷达进行地图构建 2](#_Toc181173347)

[2.1 实验内容 2](#_Toc181173348)

[2.2 实验要求 2](#_Toc181173349)

[3 ROS机器人导航实验-自主导航与路径规划 3](#_Toc181173350)

[3.1 实验内容 3](#_Toc181173351)

[3.2 实验要求 3](#_Toc181173352)

[4 ROS网络通信实验-发布与订阅节点的数据传输 4](#_Toc181173353)

[4.1 实验内容 4](#_Toc181173354)

[4.2 实验步骤 4](#_Toc181173355)

[实验心得 6](#_Toc181173356)

# ROS基础实验-虚拟环境中移动机器人控制

## 实验内容

本实验的目的是学习ROS和Gazebo仿真平台的基本操作，加载并控制一个虚拟机器人在环境中移动。通过本实验，学生可以了解 ROS 中节点、话题、消息等基本概念，并能掌握使用Gazebo仿真器进行虚拟测试的流程。学生将使用turtlebot3仿真模型，在Gazebo中创建虚拟环境，并通过键盘或代码实现机器人移动，最后实现自动导航功能。

## 实验要求

（1）环境搭建

在ROS中安装并配置Gazebo仿真器，确保能加载turtlebot3模型。使用roslaunch turtlebot3\_gazebo turtlebot3\_empty\_world.launch启动一个空白的仿真环境。

（2）机器人控制

使用teleop工具，通过键盘控制机器人前进、后退、左转和右转，熟悉Gazebo仿真环境与ROS的基本操作。编写一个ROS节点，用于让机器人执行简单的运动序列（例如：沿矩形路径行走）。

（3）自动导航

使用激光雷达数据，编写简单的避障算法，使机器人能够自动检测前方障碍物并改变方向，避免碰撞。设置目标位置，让机器人沿着设定的路径从起点移动到终点，并能够避开静态障碍物。

# ROS感知实验-使用激光雷达进行地图构建

## 实验内容

本实验旨在让学生了解如何利用激光雷达传感器进行环境的三维建图。学生将使用turtlebot3机器人和SLAM算法在Gazebo中实现仿真地图构建。通过本实验，学生将掌握激光雷达与SLAM的工作原理，以及如何利用ROS的gmapping包进行实时地图构建。

## 实验要求

（1）SLAM 环境搭建

使用 roslaunch turtlebot3\_gazebo turtlebot3\_world.launch 启动一个包含障碍物的仿真环境。

安装并配置ROS的gmapping包，确保能够在Gazebo中实现SLAM建图。

（2）激光雷达传感器使用

了解 turtlebot3 机器人中激光雷达的配置，通过RViz查看激光雷达扫描的数据。使用roslaunch turtlebot3\_slam turtlebot3\_gmapping.launch启动SLAM节点，观察RViz中地图构建的过程。

（3）地图构建

使用teleop工具控制机器人移动，通过激光雷达扫描完成环境的探索和地图的构建。保存生成的地图到本地（例如，保存为map.yaml和map.pgm文件），以便后续实验使用。

（4）验证与优化

通过调整gmapping的参数（如最大激光范围、分辨率等），优化地图构建的效果。验证地图的完整性与精度，确保生成的地图与实际仿真环境一致。

# ROS机器人导航实验-自主导航与路径规划

## 实验内容

在本实验中，学生将使用ROS导航栈实现机器人在已知地图中的定位和路径规划。该实验将帮助学生掌握如何结合SLAM构建的地图进行自主导航，实现机器人在已知环境中的自主路径规划、避障及到达目标点。

## 实验要求

（1）AMCL定位

使用上一步实验保存的地图，启动amcl节点，进行自适应蒙特卡洛定位（roslaunch turtlebot3\_navigation turtlebot3\_navigation.launch map\_file:=<path to map.yaml>）。使用RViz确保机器人能够正确地在构建的地图中显示位置，并通过手动控制验证定位精度。

（2）目标设置与路径规划

使用ROS的导航栈move\_base，在RViz中设置目标位置，让机器人自动规划路径到达终点。验证机器人是否能够避开静态障碍物，沿最优路径到达目标位置。

（3）动态障碍物避让

在仿真环境中添加动态障碍物，测试机器人能否实时避开动态障碍物，并继续前往目标。通过调整局部规划器的参数，优化机器人的避障效果和路径规划速度。

# ROS网络通信实验-发布与订阅节点的数据传输

## 实验内容

本实验旨在帮助学生理解ROS中的发布-订阅机制，通过创建多个ROS节点，实现数据的发布与订阅，探索不同节点之间如何通过话题进行通信。学生将编写简单的发布者和订阅者节点，观察消息的传递过程，并使用工具可视化ROS系统的拓扑结构。

## 实验步骤

**（1）环境准备**

确保 ROS 系统（如 ROS Noetic）已成功安装，并已创建工作空间。

创建一个新的ROS包，用于存放本实验的节点代码。

**（2）编写发布者节点**

创建一个发布者节点，该节点定期发布一个整数计数器（从0开始，每秒增加1），将其发布到特定的话题上（自己命名）。

（3）**编写订阅者节点**

创建一个订阅者节点，订阅话题，接收来自发布者的整数消息，并将其打印到控制台。

（4）**运行ROS节点**

（5）**观察数据传输**

在订阅者的控制台中，观察到来自发布者的整数计数消息被成功接收并打印。

通过调整发布频率或改变发布的数据类型，观察对消息传递的影响。

**（6）可视化与调试**

使用rqt\_graph工具可视化ROS节点之间的关系。

**（7）扩展任务**

添加更多的发布者和订阅者节点，形成复杂的消息传递结构。例如，创建一个新的发布者节点发布字符串消息，观察订阅者如何接收不同类型的消息。

探索ROS的服务（Service）和动作（Action）机制，与发布-订阅模型进行对比，理解不同通信方式的适用场景。

# 实验心得