



华南理工大学  
South China University of Technology

# 硕士学位论文

面向狭窄环境的移动机器人室内自主导航

研究及实现

作者姓名 谭皓禹

学科专业 计算机技术

指导教师 董敏 副教授

覃争鸣 高级工程师

所在学院 计算机科学与工程学院

论文提交日期 2025 年 4 月



# **Research and implementation of indoor autonomous navigation for mobile robots in narrow environment**

A Dissertation Submitted for the Degree of Master

**Candidate: Haoyu Tan**

**Supervisor: Associate Prof. Min Dong**

South China University of Technology

Guangzhou, China



分类号：TP3

学校代号：10561

学 号：202221043901

## 华南理工大学硕士学位论文

# 面向狭窄环境的移动机器人室内自主导航研究 及实现

作者姓名：谭皓禹

指导教师姓名、职称：董敏 副教授

申请学位级别：工程硕士

学科专业名称：计算机技术

研究方向： 室内机器人导航

论文提交日期：2025 年 6 月 06 日

论文答辩日期：2025 年 6 月 06 日

学位授予单位：华南理工大学

学位授予日期： 年 月 日

答辩委员会成员：

主席：\_\_\_\_\_

委员：\_\_\_\_\_



# 华南理工大学

## 学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律后果由本人承担。

作者签名： 日期： 年 月 日

## 学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属华南理工大学。学校有权保存并向国家有关部门或机构送交论文的复印件和电子版，允许学位论文被查阅（除在保密期内的保密论文外）；学校可以公布学位论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存、汇编学位论文。本人电子文档的内容和纸质论文的内容相一致。

本学位论文属于：

保密（校保密委员会审定为涉密学位论文时间：\_\_\_\_年\_\_\_\_月\_\_\_\_日），于\_\_\_\_年\_\_\_\_月\_\_\_\_日解密后适用本授权书。

不保密，同意在校园网上发布，供校内师生和与学校有共享协议的单位浏览；同意将本人学位论文编入有关数据库进行检索，传播学位论文的全部或部分内容。

(请在以上相应方框内打“√”)

作者签名： 日期：

指导教师签名： 日期：

作者联系电话： 电子邮箱：

联系地址(含邮编)：广东省广州市天河区华南理工大学（五山校区）3号楼



## 摘要

我国政府在《“十四五”机器人产业发展规划》中指出，面向家庭服务、公共服务、养老助残等领域需求，重点推进服务机器人的研制及应用，开发控制软件、核心算法等，提高机器人控制系统的功能和智能化水平。其中移动服务机器人以室内为主要工作场景，以自主导航作为执行任务的基础功能，然而室内环境的错综复杂性以及不可预测的障碍物对机器人的导航技术构成了严峻的挑战。当前主流商用机器人多采用激光雷达结合低功耗计算平台的导航架构，该方案在常规场景下表现稳定，但导航系统在狭窄环境等受限空间内对误差源的容忍性下降，容易出现轨迹规划失效、运动控制失稳等问题，这种空间适应性缺陷导致机器人难以满足全场景自主作业的可靠性要求，成为了服务机器人实现完全自主可靠地工作的障碍之一。对此本文针对硬件方案成本受限的移动机器人如何安全稳定地通过狭窄环境进行研究，提出了一种基于自适应环境与示教重复算法的面向狭窄环境的室内自主导航系统，主要工作如下：

- 1) 本文提出了自适应狭窄环境的室内导航系统。该系统在导航过程中根据地图中障碍物与全局路径的几何关系标记狭窄环境并生成合适通行位姿对；机器人出入被标记的狭窄环境时自动切换相应导航策略，以自适应环境；在狭窄环境导航策略中，全局成本地图膨胀化以规划更安全的全局路径，机器人根据合适通行位姿分段规划全局路径，目的为提前调整位姿以减少在狭窄环境中的转向需求，并通过最优控制问题转化为最小二乘问题的思想优化 MPC 路径跟踪方法，用以代替局部轨迹规划方法计算轨迹，防止局部轨迹碰撞误判导致导航失败。
- 2) 本文提出了基于豪斯多夫距离的点云匹配示教重复算法，作为自适应狭窄环境的室内导航系统中的补充策略。由于导航系统中如定位、地图等各环节的误差源过多，因此抛弃传统导航框架，直接利用传感器信息控制机器人。示教阶段利用三维激光雷达扫描狭窄环境三维特征形成激光点云管道，重复阶段机器人根据当前环境点云与点云管道进行基于豪斯多夫距离的点云匹配，并根据李雅普诺夫稳定性理论构造反馈控制器，以推理当前点云状态与目标点云状态之间转换的控制速度。
- 3) 本文结合上述方法设计并实现了面向狭窄环境的机器人室内导航系统并在仿真环境和真实机器人上部署，在部署平台上完成了一系列可行性与性能测试。仿真及真实环境实验结果表明，该系统能够有效识别环境狭窄与否并切换相应的导航策略，提升了机器人面对狭窄环境时的通过率。

**关键词：**移动机器人；室内导航系统；狭窄环境；自适应环境；示教重复

## Abstract

The Chinese government pointed out in the "14th Five-Year Plan" robot industry development Plan that for the needs of family services, public services, elderly care and other fields, it focuses on promoting the development and application of service robots, developing control software, core algorithms, etc., and improving the function and intelligence level of the robot control system. Among them, mobile service robots take indoor as the main working scene and autonomous navigation as the basic function of performing tasks. However, the complexity of indoor environment and unpredictable obstacles pose a severe challenge to the navigation technology of robots. At present, the navigation architecture of LiDAR combined with low-power computing platform is mostly adopted by mainstream commercial robots. This scheme is stable in conventional scenarios, but the tolerance of the navigation system to error sources in confined Spaces such as narrow environments is reduced, and problems such as trajectory planning failure and motion control instability are prone to occur. This spatial adaptability defect makes it difficult for robots to meet the reliability requirements of full-scene autonomous operation, and becomes one of the obstacles for service robots to achieve complete autonomous and reliable work. In this paper, aiming at studying how mobile robots with limited hardware scheme cost can safely and stably pass through narrow environment, an indoor autonomous navigation system for narrow environment based on adaptive environment and teaching repetition algorithm is proposed. The main work is as follows:

- 1) An automatic adaptive narrow-environment navigation system is proposed. During navigation, the system marks narrow environments based on geometric relationships between obstacles and global paths, generating suitable passage pose pairs. It automatically switches navigation strategies when entering or exiting marked narrow environments. For confined spaces, the global cost map is inflated to plan safer paths. The robot segments global paths using optimal pose pairs to reduce steering demands and optimizes the Model Predictive Control (MPC) path-tracking method by transforming optimal control problems into least-squares optimizations, replacing local trajectory planning to avoid collision misjudgments.
- 2) A Hausdorff distance-based point cloud matching teaching-repeat algorithm is introduced

as a supplementary strategy. To mitigate cumulative errors from localization and mapping in traditional frameworks, this method directly utilizes sensor data. During the teaching phase, a 3D LiDAR scans narrow environments to construct a laser point cloud pipeline. In the repeat phase, the robot matches current environmental point clouds with the pipeline using Hausdorff distance and designs a feedback controller based on Lyapunov stability theory to derive control velocities for transitioning between current and target point cloud states.

- 3) An integrated navigation system is designed and deployed in both simulated and real-world environments. Feasibility and performance tests on physical robots validate the system's effectiveness in enhancing reliability for narrow-space navigation. Simulations and real-world experimental results demonstrate a significant improvement in the robot's passage rate through confined spaces, highlighting the system's robustness in complex indoor scenarios.

**Keywords:** Mobile robot; Indoor navigation system; Narrow environment; Automatic adaptation environment; Teach and repeat

# 目 录

<b>摘 要</b> .....	I
<b>Abstract</b> .....	III
<b>插图目录</b> .....	VII
<b>表格目录</b> .....	IX
<b>第一章 绪论</b> .....	1
1.1 研究背景和意义 .....	1
1.2 国内外研究现状 .....	3
1.3 研究内容 .....	6
1.4 组织结构 .....	7
<b>第二章 相关工作</b> .....	9
2.1 机器人操作系统 ROS .....	9
2.2 同时定位建图 (SLAM) .....	10
2.3 传统室内自主导航框架 .....	10
2.3.1 地图服务 .....	11
2.3.2 自主定位 .....	11
2.3.3 全局路径规划 .....	13
2.3.4 局部轨迹规划 .....	13
2.3.5 环境感知模块 .....	14
2.4 本章小结 .....	16
<b>第三章 自适应狭窄环境的室内导航系统</b> .....	17
3.1 系统整体框架 .....	17
3.2 导航策略信息生成环节 .....	19
3.2.1 导航策略信息的描述 .....	19
3.2.2 导航策略信息生成环节的步骤 .....	20
3.2.3 全局成本地图膨胀化 .....	20
3.2.4 合适位姿通行对的生成方法 .....	23
3.3 导航策略切换环节 .....	25
3.3.1 导航策略切换环节的步骤 .....	25

3.3.2 判断进入狭窄环境的方法 .....	26
3.3.3 改进路径跟踪算法 .....	28
3.4 本章小结 .....	31
<b>第四章 基于豪斯多夫距离的点云匹配示教重复算法 .....</b>	<b>33</b>
4.1 引言 .....	33
4.2 算法整体框架 .....	34
4.3 基于豪斯多夫距离的点云匹配定位 .....	35
4.4 点云管道跟踪控制 .....	38
4.5 本章小结 .....	42
<b>第五章 实验设计与结果分析 .....</b>	<b>43</b>
5.1 自适应狭窄环境的室内导航系统的相关实验 .....	43
5.1.1 仿真环境实验 .....	43
5.1.2 真实环境实验 .....	51
5.2 点云匹配示教重复算法的相关实验 .....	56
5.2.1 前期部署 .....	56
5.2.2 豪斯多夫距离计算优化验证 .....	56
5.2.3 空旷场地的示教重复实验 .....	60
5.2.4 狹窄环境的示教重复实验 .....	63
5.3 本章小结 .....	64
<b>总结与展望 .....</b>	<b>65</b>
<b>参考文献 .....</b>	<b>67</b>
<b>攻读博士/硕士学位期间取得的研究成果 .....</b>	<b>74</b>
<b>致 谢 .....</b>	<b>75</b>

## 插图目录

图 1-1 室内移动服务机器人 . . . . .	2
图 1-2 机器人穿越门时的三种情况示意图 . . . . .	3
图 1-3 面向狭窄环境的机器人室内导航系统的框架 . . . . .	8
图 2-1 传统室内自主导航系统的框架 . . . . .	11
图 2-2 单线激光雷达生成的二维激光点云 . . . . .	15
图 2-3 三维激光雷达生成的三维激光点云 . . . . .	15
图 3-1 传统室内导航系统的框架 . . . . .	17
图 3-2 自适应狭窄环境的室内导航系统的框架 . . . . .	18
图 3-3 本文提出的系统框架下导航的工作流程 . . . . .	19
图 3-4 全局成本地图示意图 . . . . .	21
图 3-5 膨胀公式示意图 . . . . .	22
图 3-6 全局成本地图用于规划全局路径的效果对比 . . . . .	23
图 3-7 合适通行位姿对生成方法的示意图 . . . . .	25
图 3-8 狹窄环境进入判断方法的流程图 . . . . .	27
图 3-9 非完整运动学 . . . . .	31
图 3-10 超图的简化示例结构 . . . . .	32
图 4-1 Livox Mid-360 三维激光雷达 . . . . .	34
图 4-2 基于豪斯多夫距离的点云匹配示教重复算法的框架 . . . . .	35
图 4-3 激光点云数据过滤前后对比 . . . . .	38
图 4-4 差速驱动模型 . . . . .	42
图 5-1 仿真机器人模型 . . . . .	44
图 5-2 仿真环境 . . . . .	44
图 5-3 在仿真环境中使用 SLAM 算法创建的地图 . . . . .	45
图 5-4 STPs 在 Rviz 中的可视化视图 . . . . .	45
图 5-5 各狭窄环境导航方法的轨迹对比 . . . . .	46
图 5-6 第 8 次实验中各方法轨迹的可视图 . . . . .	48
图 5-7 第 8 次实验轨迹误差图 . . . . .	48
图 5-8 四个不同的仿真环境 . . . . .	50

图 5-9 四个不同的仿真环境的地图和导航任务 . . . . .	50
图 5-10 用于真实环境实验的机器人 . . . . .	51
图 5-11 机器人控制模块的结构 . . . . .	52
图 5-12 真实环境中的实验场景 . . . . .	53
图 5-13 在真实环境中创建的地图 . . . . .	53
图 5-14 STPs 在 Rviz 中的可视化视图 . . . . .	53
图 5-15 STPs 在 Rviz 中的可视化视图 . . . . .	54
图 5-16 计算耗时对比 . . . . .	55
图 5-17 真实环境实验中路径跟踪算法的轨迹对比 . . . . .	56
图 5-18 安装三维激光雷达的机器人 . . . . .	57
图 5-19 Livox Mid-360 三维激光雷达的有效 FOV 范围 . . . . .	57
图 5-20 不同下采样体素参数下的激光点云图 . . . . .	58
图 5-21 空旷场地示教重复实验的场地 . . . . .	60
图 5-22 空旷场地的二维俯视图 . . . . .	61
图 5-23 狹窄环境的示教动作点展示 . . . . .	63

## 表格目录

表 5-1 仿真环境实验的软硬件配置 . . . . .	43
表 5-2 导航任务阶段一各方法效果对比 . . . . .	46
表 5-3 导航任务阶段二各方法效果对比 . . . . .	46
表 5-4 路径跟踪算法对比 . . . . .	48
表 5-5 各方法在不同环境中执行导航任务的成功次数 . . . . .	49
表 5-6 RockChip RK3399 具体参数 . . . . .	52
表 5-7 单线激光雷达的具体参数 . . . . .	52
表 5-8 真实环境实验中各方法效果对比 . . . . .	54
表 5-9 真实环境实验中路径跟踪算法对比 . . . . .	55
表 5-10 Livox Mid-360 三维激光雷达的具体参数 . . . . .	57
表 5-11 不同体素参数与线程数下的点云匹配计算耗时/毫秒 . . . . .	59
表 5-12 示教重复轨迹跟踪偏差 . . . . .	62
表 5-13 不同初始角度下的示教重复轨迹跟踪偏差 . . . . .	62
表 5-14 示教重复轨迹跟踪偏差 . . . . .	64

X

# 第一章 绪论

随着机器人技术的迅速发展，机器人从实验室扩展到了工厂、家庭服务、医疗看护、教育娱乐等各行业，为人类带来了许多便利。移动机器人以室内为主要工作场景，以自主导航作为执行任务的基础功能，然而室内环境的错综复杂性以及不可预测的障碍物对机器人的导航技术提出了较高的要求。目前大部分广泛应用于室内服务的机器人搭载基于激光雷达与低功耗控制平台的传统导航方案，其在面对狭窄环境时存在无法稳定安全地通过的问题，该问题成为了服务机器人实现完全自主可靠地工作的障碍之一。

因此，本文针对硬件方案成本受限的移动机器人如何安全稳定地通过狭窄环境进行研究，提出了一种基于自适应环境与示教重复算法的面向狭窄环境的室内自主导航系统，并将其部署在真实机器人上。本章主要有以下内容：阐述狭窄环境下的移动机器人室内导航技术的研究背景和意义、调研相关领域的国内外研究现状、介绍本文的研究内容和主要贡献、梳理本文的组织结构。

## 1.1 研究背景和意义

《“十四五”机器人产业发展规划》中提到，面向家庭服务、公共服务、医疗健康、养老助残等领域需求，重点推进服务机器人的研制及应用，开发控制软件、核心算法等，提高机器人控制系统的功能和智能化水平<sup>[1]</sup>。而第十四届全国人大二次会议通过的政府工作报告明确提出深化人工智能研发应用、实施“人工智能+”行动<sup>[2]</sup>。在多重政策背景下，室内机器人自主导航技术作为新一代信息技术融合创新的重要载体正迎来重大发展机遇，它为智慧医疗、商业综合体、家庭服务等场景提供厘米级空间的感知能力，发展深度契合国家打造数字产业集群的战略导向。近年来，优秀的服务机器人研究不断涌现，如机器人能够引导视障人士在未知的室内环境中行动<sup>[3]</sup>，作为家庭物理辅助提高人们的生活质量<sup>[4]</sup>，在移动过程中定位跟踪实验室内的设备以监控管理实验室<sup>[5]</sup>等。

在工业生产环境中，由于机器人执行的任务类型较为单一且运行环境具有固定性和可控性，对机器人成本的敏感度相对较低，因此可通过改造环境或基于高性能计算平台与高精度传感器的导航方案以实现任务需求。然而当机器人应用扩展到商用服务领域时，其运行环境呈现出显著的差异性特征：家用及商用场景通常具有空间尺度大、场景复杂度高且动态变化性强等特点，同时受限于公共空间改造的可行性约束，难以通过增设专用辅助设备来支持机器人运行。这种环境特征的转变对定位导航算法的环境适应性和动态场景处理能力提出了更高要求。与之相悖的是，商用及消费级产品存在严格的成

本约束，通常无法配置高精度传感器与高性能计算单元，普遍采用基于单线激光雷达、单目视觉传感器及低功耗嵌入式 SoC 平台运行传统室内导航系统的轮式移动底盘架构。在此类硬件配置条件下，如何在有限的硬件资源配置下实现室内完全自主可靠地工作，成为移动机器人软件系统开发的核心技术挑战。而需要克服的实际障碍之一是，室内移动所依赖的导航技术无法令机器人稳定通过狭窄环境。



图 1-1 室内移动服务机器人

在现有导航研究体系中，狭窄环境的概念尚未形成统一明确的界定标准。本研究基于几何约束特性对该术语进行量化定义：当移动机器人所处环境的通行截面宽度小于其本体宽度的 2 倍且需进行复杂的姿态调整与路径规划才可能通行，则该环境被视为狭窄环境。

理想情况下，当狭窄环境中存在满足运动约束条件的全局路径时，机器人应具备生成可行的局部轨迹以穿越该环境的能力。然而在工程实践中发现，机器人在遵循全局路径以及生成局部轨迹等方面存在问题：在狭窄环境中，传统导航框架中的局部轨迹规划受激光雷达测距误差、地图分辨率量化误差、定位漂移与当前位姿等多源因素影响，无法感知前方存在的可通行区域导致局部轨迹生成失败；环境感知的不确定性可能导致轨迹频繁发生变化，引发运动振荡现象；更严重时在通行过程中与环境产生碰撞，最终导致导航任务失败。

为了说明该问题，图1-2演示机器人（机身宽度 20cm）穿越狭窄门框（宽度 35cm）的场景，其中绿色线条为机器人计算的可行路径，粉红色区域为可行区域，蓝色区域为致命区域，红色与橙色点为激光点云。在该场景中展示了三种不同的情况：(a) 期望的

情况，机器人起始位于通道中间且正对门，并几乎沿一条直线路径穿越；(b) 更具挑战性的情况，机器人位置靠墙，传感器噪声、定位漂移等多源误差的组合使得感知门框处的可导航区域减少，但仍可计算出一条可行轨迹；(c) 机器人位姿倾斜对门，无法感知可通行区域，机器人认为没有足够的空间通过门，因此会中止导航。所有这些情况下的关键问题是，机器人规划和执行导航的操作空间相对较小。以门框通过场景为例，理论上存在约 15 厘米的可行通道空间，但传感器数据构建的代价地图显示可行区域宽度仅剩数厘米。这种感知空间与理论空间的显著差异导致导航成功率大幅降低，严重制约了机器人在室内环境中的移动能力。当前代价图虽已反映传感器噪声、栅格地图量化误差、定位漂移等因素，但仍需综合考量导航算法在轨迹跟随全局路径过程中产生的偏差，以及机械系统运动执行误差对最终导航效果的叠加影响。

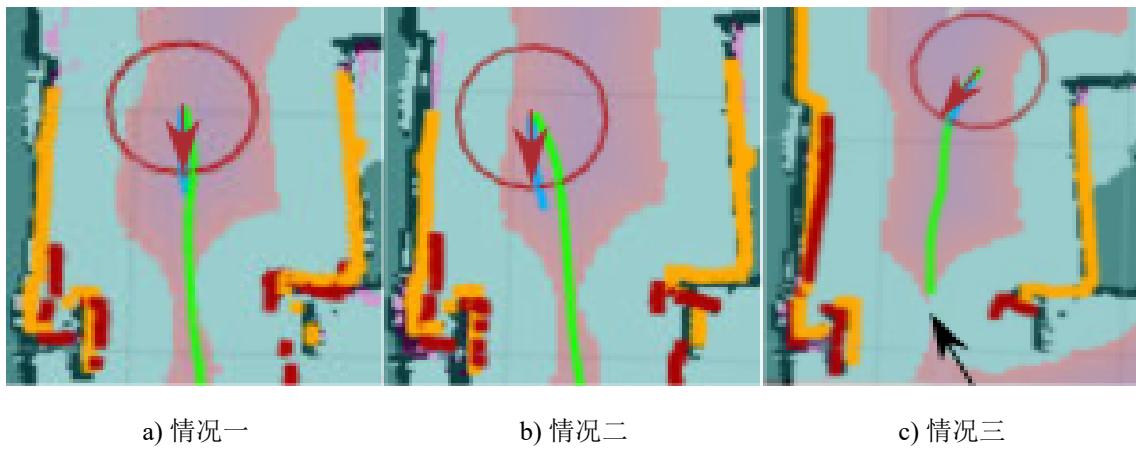


图 1-2 机器人穿越门时的三种情况示意图

## 1.2 国内外研究现状

许多学者对机器人在狭窄环境中无法安全稳定通行的问题展开了研究，Yuan 等<sup>[6]</sup>提出对 TEB<sup>[7]</sup>、DWA<sup>[8]</sup>等主流局部轨迹规划算法实施参数优化策略，通过降低避障相关的权重参数以弱化机器人的碰撞检测条件，使机器人能够在狭窄环境前感知到更宽的可行区域，以此生成轨迹并通行，然而该方法在提升通行能力的同时降低了系统安全裕度，且可能引发其他场景下导航性能退化的问题。代婉玉等<sup>[9]</sup>认为 TEB 计划复杂的动态环境中的轨迹时会发生由非平滑速度控制量引起的轨迹振动，因此提出划分可行区域，并将障碍距离添加至算法的速度约束的方案，有效地减少了振动振幅和在机器人接近障碍物后速度跳跃过程中机器人的影响。栾添添等<sup>[10]</sup>提出 PAP-TEB 算法，采用速度插值补偿技术改良 TEB 算法的速度输出特性，结合人工预设狭窄通道两侧辅助路径点的方式，

通过定点规划算法提升狭窄区域通过率。然而，该方法需要人工预先设置辅助路径点，严重制约了系统的自主决策能力，机器人无法做到完全自主地工作。Moreno 等<sup>[11]</sup>设计了基于环境特征识别的自动路径点生成机制，通过智能检测地图中导航困难的区域并创建辅助导航节点，有效改善了狭窄通道的轨迹规划效果。但该方案仅适用于门框、转角等短距离狭窄结构，在连续性狭长走廊场景中仍存在路径规划失效的局限性。

一些工作将焦点集中在导航的全局路径规划部分，专门设计用于狭窄可导航区域的规划器，对于类似汽车的机器人，它们提供了能够通过多种机动生成高质量路径的几何路径规划器<sup>[12][13][14]</sup>。Wang 等<sup>[15]</sup>提出了一种全新的两阶段路径规划器，该规划器结合了高效采样桥接测试<sup>[16]</sup>算法，用于识别关键区域（即狭窄区域）和快速探索随机树（RRTs），在多自由度机器人路径规划中展现了良好的性能。遵循这种方法，Lai 等<sup>[17]</sup>提出了 RRTs 的一种变体 RRDT\*，其使用多个分离树来通过 Markov 链随机采样来利用空格的局部连接性，用于平衡本地和全局信息，在高度受限的环境中获得了更好的结果。

关于在狭窄区域中通行的问题，一些方法通过改变机器人的形态或运动方式解决。Zhong 等<sup>[18]</sup>提出一种单腿磁控软跳机器人，通过无线磁场驱动实现连续快速跳跃和运动方向自主调整，使其能够在狭窄的空间内灵活调整运动方向；Liu 等<sup>[19]</sup>设计了一种基于正交连杆的具有可重组模块的蛇形机器人，能够在狭窄管道中以行波或螺旋的形式向前移动；Barati 等<sup>[20]</sup>基于依次连接的关节执行器组成的蛇形机器人提出蛇运动步态策略，通过深度强化学习实现了蛇形机器人更快的运动与更节能的运动步态，进一步强化了其在狭窄环境中的运动能力。尽管上述机器人在狭窄环境中通行的表现良好，但普遍依赖外部遥控操作，尚未建立完整的自主导航体系。

检测并穿越门的任务作为机器人在狭窄环境中的特殊挑战之一被广泛研究。Salaris 等<sup>[21]</sup>提出了一种基于共焦曲线的几何方法，用于仅使用单目相机导航的机器人穿过窄门。Aude 等<sup>[22]</sup>实现了基于单目相机检测长直线与踢脚板特征的识别门状态（关闭、部分打开、全开等）的系统，以允许机器人判断开门幅度是否足以通行。Kim 等<sup>[23]</sup>在带有立体相机的机器人上部署基于上下文的目标识别方法，适用于检测不同类型的门；Dai 等<sup>[24]</sup>提出利用 Kinect 相机采集的 RGB-D 数据完成对门的检测和定位的方法，并设计非线性自适应控制器使机器人垂直通过门。Kim 等<sup>[25]</sup>提出搭载云台变焦相机的轮椅机器人系统，能够实现高精度视觉标记门并穿越。然而，上述方法依赖特定特征的检测来识别门或走廊，因此具有一定的场景限制，无法在其他狭窄环境中广泛应用。

采用基于地标的导航技术是目前仓储物流、餐厅等场景中的移动机器人的主流导航

解决方案，通过在环境中布置辅助设备引导机器人精准运动即可安全稳定地通过狭窄过道。磁条导航技术<sup>[26]</sup>在地面铺设磁条，机器人通过测量路径上的磁感应信号获取精准定位信息，控制器分析道路信息并引导机器人沿磁条路径向目的地驶去。卢光旭<sup>[27]</sup>提出一种基于波峰寻找法的 AGV 色带导航技术，利用色带作为导航路径的人工标志物，采用可见光摄像机对色带的饱和度和色度信息进行识别，有效降低了环境布置的成本。在环境中布置二维码的二维码导航已在天猫超市、海康威视等公司仓库中使用<sup>[28]</sup>，机器人通过相机扫描地面二维码，解析信息获取当前位置，依次沿导航路径中二维码标签位置行走到达目标点，其最早应用于亚马逊 KIVA 物流机器人，而后国内电商企业纷纷采用，是目前最主流的物流仓储室内导航方法。分析发现，其共同点为需要布置大量地标以及根据特定环境构建配置相应的导航系统，部署维护成本高与环境变化适应性低的缺点成为其无法在其他场景中广泛应用的原因之一。

在成功生成全局路径的状态下，若局部轨迹规划算法失败，我们可以考虑使用路径跟踪算法生成轨迹以对全局路径进行严格跟踪来代替局部轨迹规划的功能。局部轨迹规划算法与路径跟踪算法在功能和应用上存在显著差异：局部轨迹规划算法侧重于在动态环境中实时规划机器人的运动路径，避免碰撞并保持高效的轨迹规划；而路径跟踪算法主要关注如何使机器人沿预定的轨迹平滑、准确地行驶，这类算法通过优化控制输入来最小化跟踪误差，旨在提高路径跟踪的精度和稳定性。目前主流的路径跟踪算法有 Pure Pursuit<sup>[29]</sup>、PID<sup>[30]</sup>、LQR<sup>[31]</sup>、MPC<sup>[32]</sup>等算法。Pure Pursuit 算法是一种基于几何模型的跟踪算法。其核心思想是模拟人类驾驶行为，在车辆前方设定一个预瞄点，通过计算车辆当前位置到该点的圆弧路径来确定前轮转向角。算法根据车辆轴距和预瞄点横向偏差来求解转向曲率，使得车辆沿圆弧逼近目标路径，在这个过程中预瞄距离的选择较为关键，过短易导致轨迹震荡，过长则跟踪滞后。该算法简单高效，但对复杂路径适应性有限。PID 算法根据比例（P）、积分（I）、微分（D）三个环节调节横向偏差，其中比例项快速响应当前路径偏移，积分项消除像持续侧风干扰等稳态误差，微分项抑制超调与振荡，其结构简单易实现，但参数调优依赖经验且在高速或大曲率场景下稳定性受限。LQR 算法基于最优控制理论，以状态空间模型为基础设计控制器最小化二次型代价函数，通过求解 Riccati 方程得到最优反馈增益以实现多状态变量的平衡控制，其擅长处理线性系统，但对非线性及约束条件支持有限。MPC 采用滚动时域优化策略，在每个控制周期内基于动力学模型预测未来多步状态，求解带约束的最优控制序列并执行首步控制，其核心优势在于显式处理多约束条件和非线性模型，适应复杂场景但实时性依赖

高效求解器，计算成本较高，通常需硬件加速。

示教重复（TR）技术在机器人通过狭窄环境中的应用具有重要意义，操作者通过示教可手动或半自动地让机器人记录一条安全的路径，重复则确保每次通过狭窄区域时的轨迹一致性。作为最早的研究之一，Furgale 等<sup>[33]</sup>在机器人沿着路线驾驶时构建了一个重叠子地图的流形图，机器人根据该图定位并自主重复路线。Gridseth 等<sup>[34]</sup>通过使用本地图像特征开发了一种称为多体验定位的方法，其通过在多次重复中发现的局部图像特征之间桥接来提高性能。上述方法对初始视角敏感：由于局部图像特征的位置会根据视点发生巨大变化，机器人可能会丢失对路径的跟踪。Camara 等<sup>[35]</sup>在示教阶段使用 CNN 模型作为特征提取器以构建特征数据库；在重复阶段，将 CNN 模型提取的特征与特征数据库进行比较以进行视觉位姿识别，同时水平偏移估计器还用于查询机器人需要移动的方向。然而该算法对视点更改仍不可靠，当机器人从远离实际示教路径的位姿开始时也可能会丢失对路径的跟踪。Dall’Osto 等<sup>[36]</sup>提出的 TR 算法利用里程计信息与由计算轻量级视觉处理模块驱动的校正信号配对，以完成位姿估算，但该算法无法处理极端的初始位姿变化。为缓解位姿依赖问题，Toudeshki 等<sup>[37]</sup>提议利用环境对象，它可以是独立于视点的环境的稳健语义特征，该算法记录在示教阶段发现的语义对象，在重复阶段中使用 SEQ-SLAM-like 重新定位来识别最相似的学习场景，并根据 funnel lane 理论使用运动控制算法使机器人沿示教轨迹行驶。该工作使用二维图像特征而非三维地图作为对象，将导致重复精度较差。为了解决 VTR 的重新定位部分，还有其他基于图像的算法可用，如 Superglue<sup>[38]</sup>和 RANSAC<sup>[39]</sup>。Superglue 是一种基于神经网络的图像特征匹配方法，通过联合查找对应关系并拒绝不匹配的点来匹配两组局部特征。Mahdavian 等<sup>[40]</sup>在示教阶段使用 ORB-SLAM<sup>[41]</sup>采集摄像头位姿与环境三维点云，使用 YOLOv3<sup>[42]</sup>检测环境中的物体，并将两者输出组合以构建三维语义图；在重复阶段，根据检测到的对象和存储的语义图，通过将 Superglue 与 RANSAC 结合找到示教与重复阶段初始姿势之间转换的方式重新定位机器人，并使机器人向示教路径移动。然而，部署上述方法均需要较大的算力支持才能使推理速度与机器人的控制频率相匹配。

### 1.3 研究内容

在室内服务场景中，移动机器人依赖自主导航技术实现任务执行，高度动态性和复杂的环境对导航系统提出了较高的精度要求。当前主流商用机器人多采用激光雷达结合低功耗计算平台的导航架构，该方案在常规场景下表现稳定，但在狭窄环境等受限空

间内导航系统对误差源的容忍性下降，容易出现轨迹规划失效、运动控制失稳等问题，这种空间适应性缺陷导致机器人难以满足全场景自主作业的可靠性要求，成为了移动机器人实现完全自主可靠地工作的障碍之一。

因此，本文针对硬件方案成本受限的移动机器人如何安全稳定地通过狭窄环境进行研究，主要贡献如下：

- 1) 本文提出了自适应狭窄环境的室内导航系统。该系统在导航过程中根据地图中障碍物与全局路径的几何关系标记狭窄环境并生成合适通行位姿对；机器人出入被标记的狭窄环境时自动切换相应导航策略，以自适应环境；在狭窄环境导航策略中，全局成本地图膨胀化以规划更安全的全局路径，机器人根据合适通行位姿分段规划全局路径，目的为提前调整位姿以减少在狭窄环境中的转向需求，并通过最优控制问题转化为最小二乘问题的思想优化 MPC 路径跟踪方法，用以代替局部轨迹规划方法计算轨迹，防止局部轨迹碰撞误判导致导航失败。
- 2) 本文提出了基于豪斯多夫距离的点云匹配示教重复算法，作为自适应狭窄环境的室内导航系统中的补充策略。由于导航系统中如定位、地图等各环节的误差源过多，因此抛弃传统导航框架，直接利用传感器信息控制机器人。示教阶段利用三维激光雷达扫描狭窄环境三维特征形成激光点云管道，重复阶段机器人根据当前环境点云与点云管道进行基于豪斯多夫距离的点云匹配，并根据李雅普诺夫稳定性理论构造反馈控制器，以推理当前点云状态与目标点云状态之间转换的控制速度。
- 3) 本文结合上述方法设计并实现了面向狭窄环境的机器人室内导航系统，如图1-3所示，在仿真环境和真实机器人上部署，并针对提出的方法完成了一系列可行性与性能测试。仿真及真实环境实验结果表明，该系统能够有效识别环境狭窄与否并切换相应的导航策略，提升了机器人面对狭窄环境时的通过率。

## 1.4 组织结构

本文共有五章，具体内容如下：

第一章为绪论，交代了本课题研究面向狭窄环境的移动机器人室内导航技术的研究背景和意义，对该课题所涉及到技术的国内外研究现状进行介绍与分析，然后说明本文的研究内容与贡献，最后介绍本文的组织结构。

第二章为相关工作，介绍了与本文研究内容相关的研究工作，包括机器人操作系统 ROS、同时定位建图（SLAM）技术和传统室内自主导航框架下的相关技术。

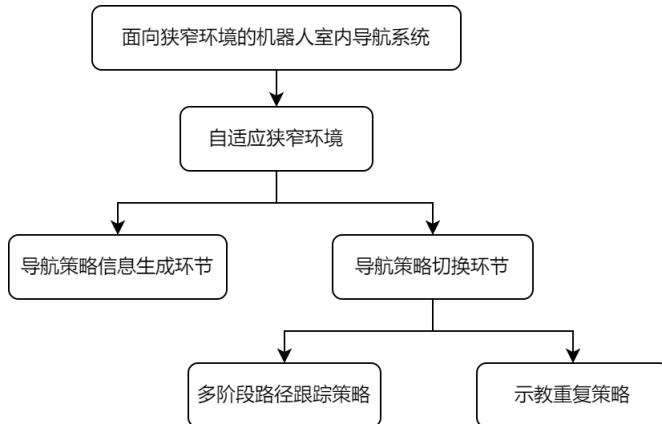


图 1-3 面向狭窄环境的机器人室内导航系统的框架

第三章详细说明本文提出的自适应狭窄环境的室内导航系统。在第三章中，首先介绍该系统的整体框架与工作流程，即在传统室内导航系统的基础上增加了导航策略信息生成环节与导航策略信息切换环节，并解释增加上述环节的理由；然后阐述系统中导航策略信息生成体实现方式，包括导航策略信息的描述、信息生成步骤、全局成本地图膨胀与通行对的生成方法；最后阐述系统中导航策略切换环节的实现方式，包括切换步骤、判断进入狭窄环境的方法与路径跟踪算法的改进思路。

第四章详细说明本文提出的基于豪斯多夫距离的点云匹配示教重复算法，作为自适应狭窄环境的室内导航系统中的补充策略。首先引入豪斯多夫距离作为处理点云之间的误差的数学工具；然后说明豪斯多夫距离的并行化计算方法，以加速其在嵌入式平台的效率；最后阐述基于李雅普诺夫稳定性理论构造的反馈控制器，以推理当前点云状态与目标点云状态之间转换的控制速度。

第五章为实验设计与结果分析，对第三章和第四章提出的方法进行可行性验证与性能分析。

最后是总结与展望，对本文所完成的工作及其实现效果进行概述，并针对系统设计中的不足之处，提出了未来研究的改进方向与可能的研究思路。

## 第二章 相关工作

本文研究的面向狭窄环境的机器人室内自主导航系统需要具体实现并部署于机器人上，借助机器人操作系统 ROS 使得开发更加高效；同时定位建图以地图的形式为后续的路径规划提供环境信息，是室内自主导航的前提；而研究内容基于传统室内自主导航系统改进。本章主要介绍上述相关技术，包括机器人操作系统 ROS、同时定位建图与传统室内自主导航框架。

### 2.1 机器人操作系统 ROS

ROS (Robot Operating System) 是一个运行于计算机操作系统上层的开放源代码的机器人元操作系统，其提供了一整套工具、库和框架，使得机器人软件的开发更加高效、模块化和可扩展，在社区活跃程度、库的丰富度、扩展性和开发便利性上都表现出色，因此是研究机器人的理想平台。ROS 具体包括以下两方面内容：

- **操作系统侧**: ROS 提供了标准的操作系统服务，如设备抽象与硬件支持、节点进程管理、进程间通信、计算资源管理以及标准化软件框架等。
- **用户侧**: ROS 提供支持多种语言的编程接口、强大的开发工具及仿真支持（如 Rviz、Gazebo、Webots 等）、包管理工具和丰富的开源功能包。

ROS 是一个松耦合的分布式系统，为异源计算集群提供了一个结构化的通信层<sup>[43]</sup>，实现了多种节点即进程通信机制供用户选择：

- **话题 (Topic)**: 基于发布/订阅模式，多个节点可同时发布消息至同一话题，所有订阅该话题的节点可实时同步获取话题中的消息更新。该机制采用非阻塞式传输方式，节点之间无直接依赖关系，适用于如传感器数据等实时数据流的分布式传播场景。
- **服务 (Service)**: 基于客户端/服务器的请求响应模式，客户节点向指定服务发起请求后等待，服务节点完成计算处理后返回应答数据。该机制采用阻塞式通信模型，要求服务调用方在获得响应或达到超时阈值前保持等待，适用于如路径规划等需要即时响应的指令型交互场景。
- **行为 (Action)**: 面向长周期任务设计的复合模式。行为通信融合了实时数据流与过程调用的双重特性。任务发起节点通过专用接口提交执行请求后，可实时获取任务进度反馈并保留对执行过程的控制权。该机制采用目标-反馈-结果的三段式交互设计，适用于机器人导航、复杂运算等需要过程监控的异步任务场景。

## 2.2 同时定位建图 (SLAM)

在移动机器人室内自主导航领域，同时定位建图（Simultaneous localization and mapping, SLAM）作为核心任务之一被定义为：移动机器人在未探索环境中实时构建环境模型并同步确定自身空间位姿的双重过程。单线激光雷达通过旋转式激光发射装置获取平面障碍物的精确距离信息，其经济性和测量可靠性使其成为 SLAM 的常用感知设备。以 Gmapping<sup>[44]</sup>为代表的粒子滤波算法采用 Rao-Blackwellized 优化策略构建平面环境模型，类似的方法还有 HectorSLAM<sup>[45]</sup>，其解决了前端扫描匹配问题，但不依赖里程计数据的特点对激光雷达的测量提出了更高的要求。

基于图优化的 SLAM 方法<sup>[46][47][48]</sup>通过图论建模解决定位问题，该理论框架可追溯至 Lu 和 Milios 的奠基性研究<sup>[49]</sup>。此类算法将传感器观测抽象为图节点，通过扫描匹配获得的位姿变换构成边约束，当检测到环境重复区域时引入闭环约束并触发图优化。相较于粒子滤波方法，这类算法通过稀疏化数据处理显著降低计算负荷，典型实现包括 Cartographer<sup>[50]</sup>和 Karto<sup>[51]</sup>等开源系统。SLAM-Toolbox<sup>[52]</sup>在继承 Karto 架构的基础上，创新性地通过冻结地图更新的方式实现了纯定位模式，拓展了算法的应用场景。

上述 SLAM 方法扫描环境后，由传统导航框架中的地图服务将环境信息转化并保存为栅格地图，即一种将环境表示为离散二维网格单元的数字化地图，每个栅格通过概率值或状态（如占用、空闲、未知）描述对应物理空间的状态。这种地图以矩阵形式存储，支持机器人进行路径规划、避障与导航。

尽管单线激光雷达具有成本优势，其感知特性仍存在局限性：稀疏的二维距离数据导致全局重定位困难，尤其在长廊等特征匮乏场景中易产生定位漂移<sup>[47]</sup>。视觉传感器通过丰富的纹理信息为全局定位提供可能，ORB-SLAM 系列算法<sup>[53][41][54]</sup>和 VINS-Mono<sup>[55][56]</sup>等视觉 SLAM 方案利用图像特征实现环境建模。然而单目视觉系统在深度估计方面存在固有缺陷，且持续图像处理带来的计算负荷也限制了其在嵌入式平台的部署可行性。

## 2.3 传统室内自主导航框架

传统室内自主导航框架是一套基于激光雷达 SLAM 的模块化系统，如图2-1所示，由地图服务、自主定位、全局路径规划、局部轨迹规划和环境感知等多个模块构成，其核心目标是通过各模块之间的协同工作，确保机器人在已知的动态环境中实现点到点

的、安全高效的自主移动。

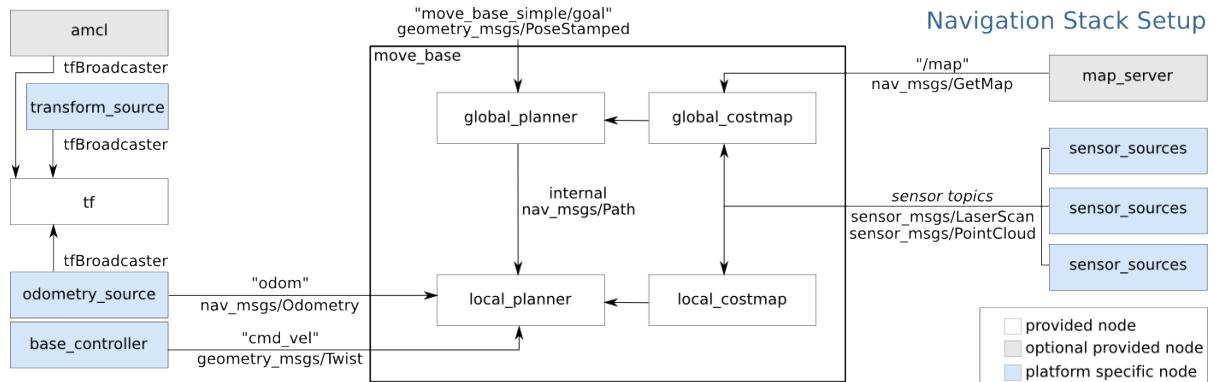


图 2-1 传统室内自主导航系统的框架

### 2.3.1 地图服务

地图服务模块的主要功能为存储和加载基于 SLAM 技术构建的静态全局地图，并转化为全局代价地图作为话题发布，供自主定位与全局路径规划模块订阅。同时，地图服务还可配合环境感知模块，对全局代价地图进行裁剪加工以生成局部代价地图，为局部轨迹规划提供地图支持。

### 2.3.2 自主定位

自主定位模块功能为通过传感器数据匹配地图，实时确定机器人在地图上的位姿。目前用于二维移动机器人的主流概率定位方法是自适应蒙特卡洛（AMCL）算法<sup>[57]</sup>。AMCL 是一种基于粒子滤波的概率定位算法如算法 1，用于在已知环境地图中估计移动机器人的位姿，核心思想是通过蒙特卡洛方法（Monte Carlo Method）近似机器人位姿的后验概率分布并引入自适应机制动态调整粒子数量和参数以平衡计算效率与定位精度。

- 粒子滤波框架

AMCL 基于粒子滤波（Particle Filter），通过一组带权重的粒子表示机器人位姿的概率分布。初始时粒子在环境地图中均匀分布。每个粒子包含位置坐标  $(x, y)$  和朝向  $\theta$  并赋予初始权重  $w_i$ 。

- 预测阶段（运动模型更新）

机器人通过里程计或运动控制器获取运动信息（如线速度  $v$  和角速度  $\omega$ ），利用运动模型预测粒子状态。AMCL 采用差分驱动模型或高斯噪声模型更新粒子位姿：

$$x_t^{(i)} = f(x_{t-1}^{(i)}, u_t) + \epsilon \quad (2-1)$$

其中  $f(\cdot)$  为运动模型函数， $u_t$  为控制输入， $\epsilon$  为噪声项，模拟里程计累积误差。

**Algorithm 1** 粒子滤波算法

**Input:**  $t - 1$  时刻的粒子群  $X_{t-1} = \{x_{t-1}^i\}_{i=1}^N$ ,  $t - 1$  时刻的系统状态  $x_{t-1}$ ,  $t$  时刻的过程噪声  $u_t$ ,  $t$  时刻的测量噪声  $z_t$

**Output:**  $t$  时刻的粒子群  $X_t$ ,  $t$  时刻的系统状态  $x_t$

```

1:  $\bar{X}_t = X_t = \emptyset$ 
2: for  $i = 1$  to  $N$  do
3:    $x_t^i = \text{motion\_update}(u_t, x_{t-1}^i)$  //根据运动模型采样;
4:    $w_t^i = \text{measurement\_update}(z_t, x_t^i)$  //根据测量模型确定粒子权重;
5:    $\bar{X}_t = \bar{X}_t + (x_t^i, w_t^i)$  //更新粒子群  $\bar{X}_t$ 
6: end for
7: for  $i = 1$  to  $N$  do
8:   draw  $x_t^m$  from  $\bar{X}_t$  with probability  $\propto w_t^m$  //从粒子群  $\bar{X}_t$  中根据权重采样;
9:    $X_t = X_t + x_t^m$  //更新  $t$  时刻的粒子群状态;
10: end for
11:  $x_t = \frac{1}{N} \sum_{i=1}^N x_t^i$  //状态估计;
12: return  $X_t, x_t$ 

```

- 测量更新（传感器融合）

通过激光雷达获取环境观测数据  $z_t$ , 结合已知地图计算每个粒子的权重。权重反映观测数据与地图的匹配程度, 计算方式为:

$$w_t^{(i)} = p(z_t | x_t^{(i)}, m) \quad (2-2)$$

其中  $p(\cdot)$  为传感器模型 (如似然场模型或波束模型),  $m$  为环境地图。高权重粒子表示其位姿更符合当前观测。

- 自适应重采样

为避免粒子退化即少数粒子占据大部分权重, AMCL 根据粒子权重进行重采样。其自适应特性体现在:

- **有效粒子数 ( $N_{\text{eff}}$ ) 判断:** 通过  $N_{\text{eff}} = 1 / \sum(w^{(i)})^2$  评估粒子集退化程度, 若低于阈值则触发重采样。
- **KLD (Kullback-Leibler Divergence) 自适应采样:** 根据粒子分布与目标分布的 KL 散度动态调整粒子数量, 减少冗余计算。

- 噪声参数自适应

AMCL 根据定位结果动态调整运动模型中的噪声参数（如平移噪声  $\alpha_1, \alpha_2$  和旋转噪声  $\alpha_3, \alpha_4$ ），在机器人静止或高速运动时优化定位鲁棒性。

AMCL 通过自适应机制使得粒子滤波的效率和鲁棒性得到显著提升，适用于动态变化环境下的实时定位。然而其全局定位能力依赖于初始粒子的分布范围，且在高度对称或特征稀疏的环境中可能出现定位发散的现象。

### 2.3.3 全局路径规划

全局路径规划模块基于全局地图计算起点到终点的最优可行路径（忽略动态障碍）。常用的全局路径规划算法有 Dijkstra 算法和 A\* 算法<sup>[58]</sup>。

Dijkstra 算法以广度优先搜索为核心思想，从起点层层向外根据评估函数  $G(n)$ （通常是起点至当前栅格  $n$  的行进距离）评估每个栅格，直到找到目标节点或没有路径可走。在找到目标后，算法通过回溯父节点构建出最短路径。Dijkstra 算法的计算结果为最优解，代价为搜索路径过程中时空复杂度较高，不适用于地图规模较大的场景。

A\* 算法在其基础上加入启发式函数  $H(n)$ ，优先扩展最有潜力，即  $F(n) = G(n) + H(n)$  最小的节点，以平衡路径长度与计算效率，计算结果虽不保证最优，但大大降低了时空复杂度，足够精确和高效。常用启发方法为当前栅格  $n$  至终点的欧氏距离或曼哈顿距离。

### 2.3.4 局部轨迹规划

局部轨迹规划模块的功能为实时发布速度控制话题“cmd\_vel”，其承担了动态避障的重要任务，使机器人在不发生碰撞的情况下尽可能沿全局路径行驶。目前主流算法有在约束速度窗口中采样、评价、取最优速度的动态窗口法（DWA）<sup>[8]</sup>、基于定时弹性带的实时轨迹修正的时间弹性带（TEB）方法<sup>[7][59]</sup>和模型预测路径积分控制算法等<sup>[60]</sup>。DWA 算法通过速度采样和轨迹模拟来生成一系列候选轨迹并评估最优路径，然而基于采样的方法无法保证找到全局最优轨迹。TEB 算法由经典的弹性带算法（EB）<sup>[61]</sup>改进而来。其继承了 EB 算法的思想，将初始路径看作一条可形变的弹性带，自定义的内力使路径收缩，外力拉伸路径使其远离障碍物，并加入时间序列信息用作控制优化过程中路径的变形，以满足路径的光滑性和速度可控性等要求。相比 DWA 算法而言，TEB 算法考虑了轨迹优化和动力学约束考虑，能够生成更加灵活、精确且可控的全局最优轨迹，代价是计算开销的增加。

### 2.3.5 环境感知模块

环境感知模块的功能直接依赖于机器人所携带的硬件设备如相机、激光雷达、编码器等。这些硬件传感器通过感知外部环境并为机器人提供必要的环境数据支持，从而实现如 SLAM、自主定位、局部轨迹规划等功能。

- **相机**

相机作为最常见的视觉传感器，通过捕捉周围环境的图像或视频流提供丰富的视觉信息。机器人能够根据图像处理算法来识别并理解物体、障碍物和环境的颜色、纹理、形状等特征。这些视觉数据可以用于物体检测、目标追踪，或者与其他传感器的数据融合以增强机器人对环境的理解。

- **单线激光雷达**

单线激光雷达是一种利用激光光束进行距离测量的传感器，能够提供高精度的二维深度数据，帮助机器人识别障碍物的位置及形态。本文使用基于飞行时间（Time of Flight, ToF）<sup>[62]</sup>的深圳市镭神智能系统有限公司生产的 LS01D 型单线激光雷达，其基本原理为：通过脉冲激光器发射近红外波段的高频调制激光脉冲，经目标物体反射后由高灵敏度光电探测器接收回波信号，系统通过精确计算发射与接收信号的时间差，结合光速常数解算目标距离信息，如公式2-3：

$$d = \frac{c \cdot \Delta t}{2} \quad (2-3)$$

其中， $d$  是物体的距离， $c$  是光速， $\Delta t$  是往返时间。

该激光雷达采用机电旋转扫描结构实现水平方向 360° 全景探测，最终提供的环境信息为带有强度和深度数据的二维点云数据，如图2-2所示。

单线激光雷达具有较为简单的结构与低成本，常用于低精度要求的应用场景。然而由于单线激光雷达只有一个激光发射线，它在空间覆盖范围和精度上存在一定的局限性。

- **三维激光雷达**

三维激光雷达是一种通过同时发射多束激光并接收其反射信号来实现三维环境感知的传感器。其核心原理同样基于时间飞行（Time-of-Flight, ToF）测距技术，结合多通道激光发射与接收系统并行获取多个方向的距离信息，最终生成高密度的三维点云数据。本文采用的 Livox Mid-360 激光雷达使用旋转棱镜扫描系统，它通过双楔形棱镜的差速旋转产生玫瑰曲线扫描轨迹，配合 850nm 波长激光器实现无

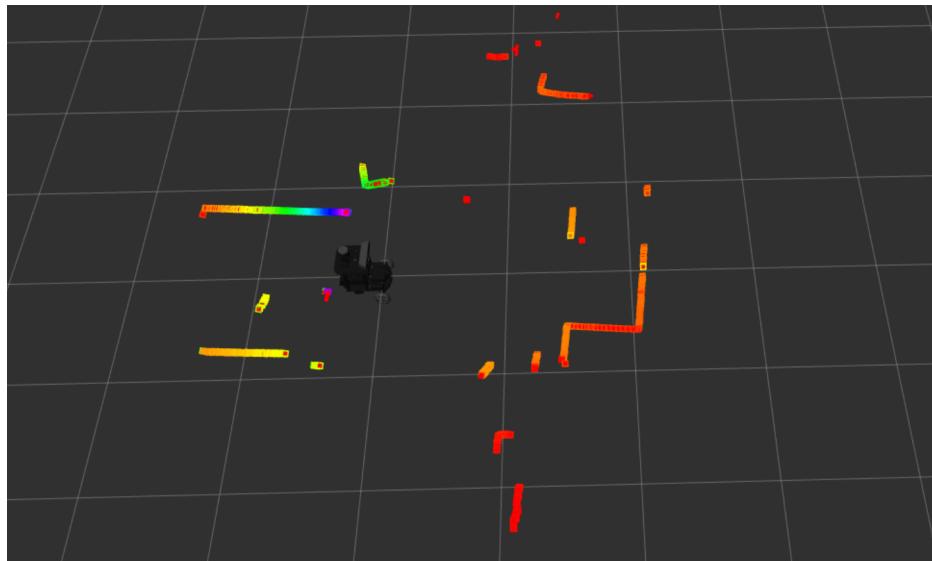


图 2-2 单线激光雷达生成的二维激光点云

重复区域扫描，采用动态噪声抑制算法、自适应阈值滤波与运动畸变补偿算法有效降低环境光干扰与运动伪影的影响并提升点云信噪比，最终提供的环境信息为带有强度和深度数据的三维点云数据，如图2-3所示。

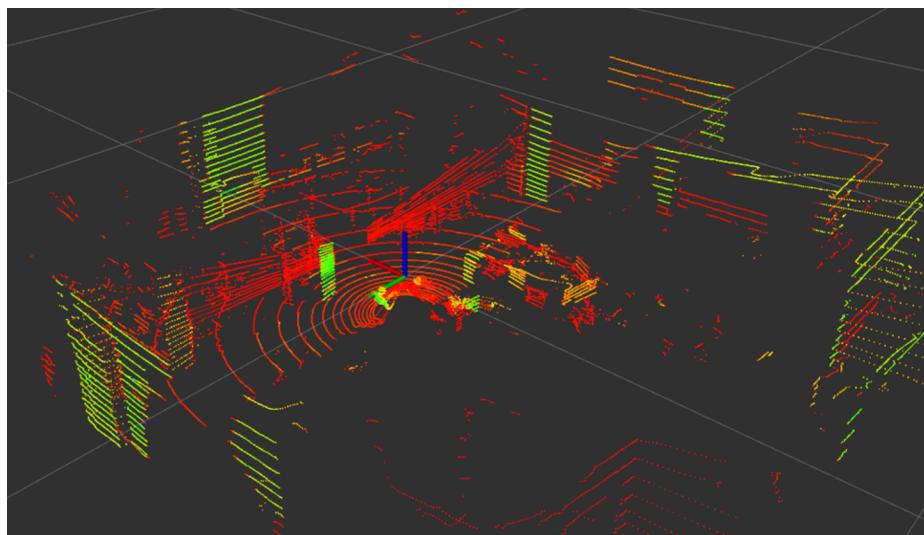


图 2-3 三维激光雷达生成的三维激光点云

三维激光雷达相比单线激光雷达具有更高的点云密度、更宽的扫描角度，这意味着更强的环境感知能力，结构复杂也使其成本相对更高。

#### • 编码器

编码器主要用于检测机器人的运动信息，通常与电机连接以监测电机的转速、旋转角度等信息。机器人能够根据编码器提供的数据计算自身的位移、速度、加速度等里程计信息，这些信息对于自主定位来说非常重要，特别是在 GPS 信号不可

用的室内环境中，编码器能够提供连续、可靠的运动反馈，确保机器人能够精确地跟踪行驶轨迹。

这些硬件传感器在 ROS 通信网络中发布各自的感知数据，使得机器人能够实时获取关于环境的全面信息。SLAM 系统将通过结合相机、激光雷达和编码器的数据进行环境建图；自主定位算法则利用这些信息对机器人的当前位姿进行实时估算；局部轨迹规划算法则在此基础上为机器人规划安全且高效的行进轨迹。

## 2.4 本章小结

本章介绍了机器人操作系统 ROS、同时定位建图（SLAM）以及传统室内自主导航框架的相关技术。ROS 为机器人开发提供了高效的工具和通信机制，SLAM 技术通过构建环境地图和确定机器人位姿支持自主导航，而传统的室内导航框架则通过各模块的协同工作，实现机器人在复杂环境中的自主移动。上述技术为本研究的室内导航系统提供了理论和实践基础。

## 第三章 自适应狭窄环境的室内导航系统

本章将详细介绍自适应狭窄环境的室内导航系统。首先介绍该系统的整体框架与工作流程，即在传统室内导航系统的基础上增加了导航策略信息生成环节与导航策略信息切换环节，以达到机器人自动判断并标记狭窄环境、根据环境切换不同导航策略的目的；然后阐述系统中导航策略信息生成环节的实现方式，包括导航策略信息的描述、信息生成步骤、全局成本地图膨胀与通行对的生成方法；最后阐述系统中导航策略切换环节的实现方式，包括切换步骤、判断进入狭窄环境的方法与路径跟踪算法的改进思路。仿真及真实实验结果能够证明该方法的可行性与高效性。

### 3.1 系统整体框架

传统的室内导航系统框架如图3-1所示。在非完整系统路径规划任务中，机器人运动特性需要同时满足几何学约束如障碍物避让和运动学约束如最小转弯半径的双重要求<sup>[63]</sup>，对此该领域普遍采用两阶段规划架构：全局路径规划阶段基于静态环境地图的几何约束条件，通过搜索算法计算得到连接起止点的最优路径；局部轨迹规划阶段则融合全局路径指引与激光雷达点云信息等实时环境感知数据，构建满足机器人动力学特性且路径跟踪误差可控的局部运动轨迹。这种分层规划机制在实际应用中会引发一些问题：其一，虽然全局路径规划可解算出理论可行的通行路径，但当机器人以非理想位姿接近通道入口即其与入口存在较大航向偏差时，运动学模型的限制将导致局部轨迹规划器无法生成满足路径跟踪要求的可行解；其二，如激光雷达的距离噪声和角度分辨率限制等感知传感器固有的测量误差会加剧局部环境建模的不确定性，使得规划器误判狭窄区域不可通行，最终引发导航任务失败。

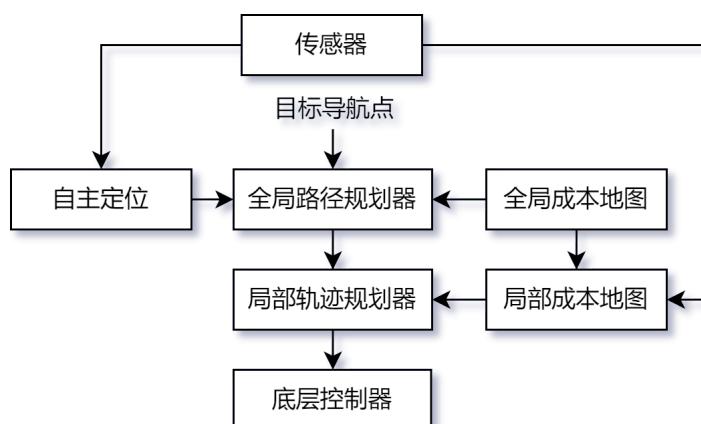


图 3-1 传统室内导航系统的框架

本文提出的自适应狭窄环境的室内导航系统的整体框架如图3-2所示。

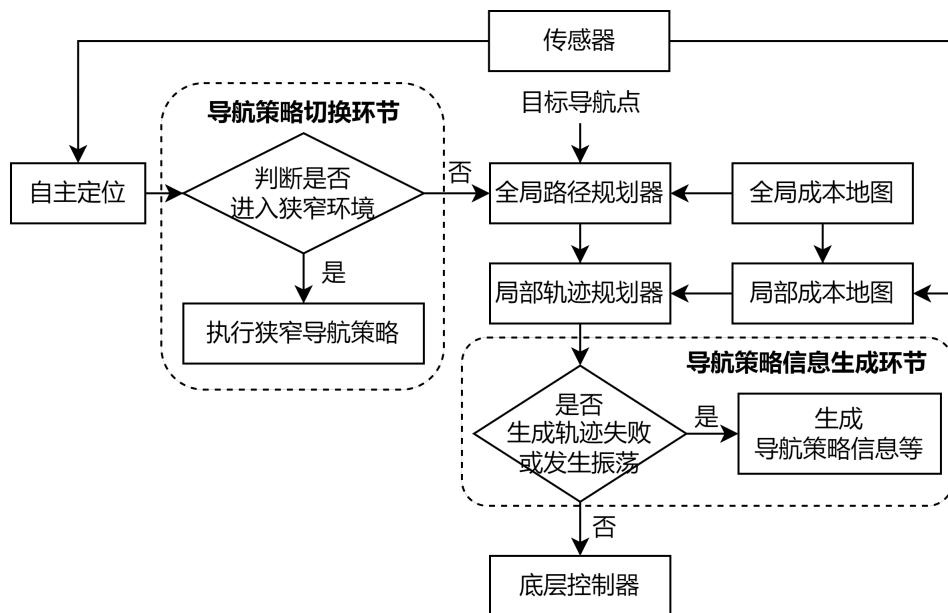


图 3-2 自适应狭窄环境的室内导航系统的框架

本研究所构建的导航框架在传统室内导航系统架构的基础上创新性地引入了双重自适应优化机制如图3-2虚线框所示。系统通过以下两个核心环节实现狭窄环境场景的智能应对：其一是基于环境拓扑分析的导航策略信息生成环节，该模块依托 SLAM 构建的全局地图提取狭窄通道的通道宽度、延伸方向及出入口空间坐标等几何特征参数，在通道出入口处生成合适通行位姿用于指导机器人通行；其二是导航策略切换环节，该环节基于上述生成的导航策略信息动态检测机器人运行轨迹与狭窄区域的相对位置关系，根据检测结果实现常规导航策略与狭窄环境导航策略的自适应切换机制，其中常规导航策略作为传统室内导航系统功能模块的技术代称，旨在与面向狭窄场景的特殊导航策略形成功能区分，其中前者主要继承传统系统的路径规划与避障功能，后者则专门处理机器人穿越狭窄通道时的运动控制问题。

如图3-3所示为自适应狭窄环境的室内导航系统的工作流程。其中，狭窄环境导航策略采用图3-3所示流程主要基于以下技术考量：对全局成本地图实施膨胀化处理能够有效提升路径规划的安全裕度；针对狭窄通道内大曲率轨迹易引发运问题，系统依据导航策略信息生成环节在通道出入口生成的合适通行位姿对人在进入前完成行进姿态校准，从而规避因突发转向需求导致轨迹曲率过大的安全隐患；路径规划采用分段处理机制确保机器人在狭窄区域内尽可能执行直线运动，降低运动控制系统的计算复杂度；最后采用路径跟踪算法替代传统局部轨迹规划方案，该算法严格遵循全

局路径生成局部运动轨迹，与传统方案依赖动态环境感知不同，该方式有效克服了激光雷达等传感器因测量误差导致局部轨迹规划失效的技术瓶颈，同时通过高精度路径追踪避免机器人位姿偏差超出安全阈值。下文将重点论述导航策略信息生成模块与策略切换模块的具体实现方案。

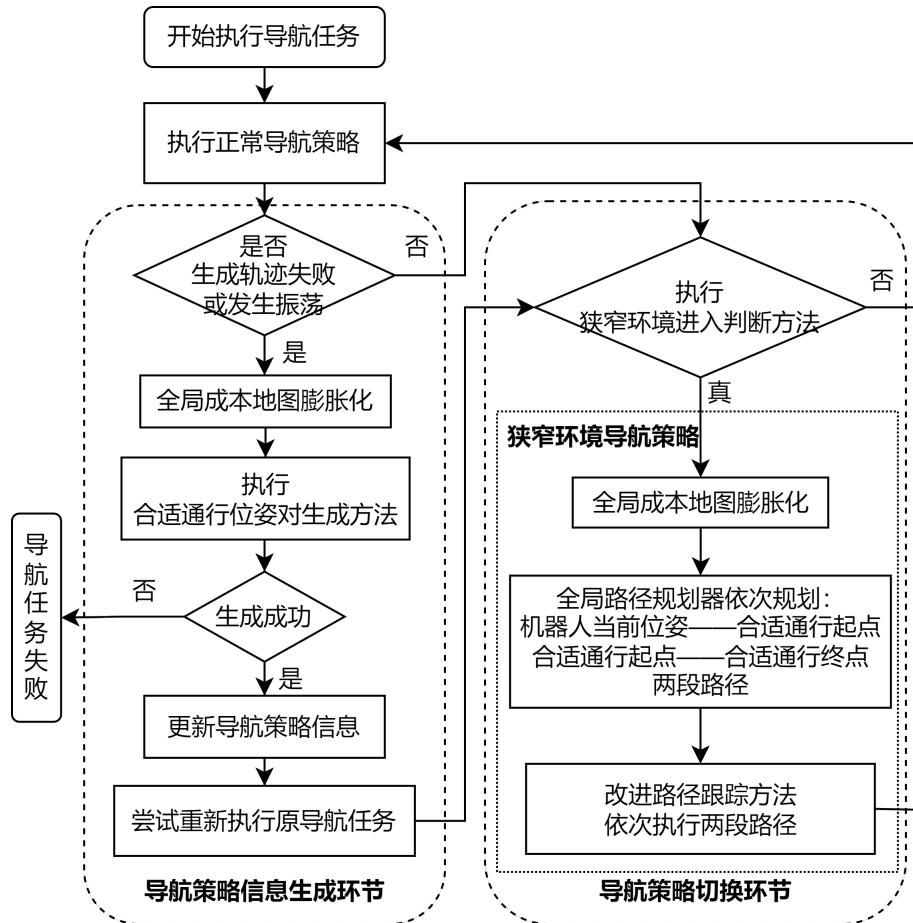


图 3-3 本文提出的系统框架下导航的工作流程

## 3.2 导航策略信息生成环节

### 3.2.1 导航策略信息的描述

导航策略信息  $I$  在数学上表征为包含  $n$  组导航失效位姿 (Navigation Failure Pose, NFP) 及其关联的合适通行位姿对 (Suitable Transit Poses, STPs) 的元组序列，其形式化表达为：

$$I = \{NFP_i, STPs_i\}, i = 1, 2, 3, \dots, n \quad (3-1)$$

式中  $NFP_i$  表示机器人在环境地图中记录的第  $i$  个导航失效位姿，其对应生成的合适通行位姿对记作  $STPs_i$ 。每个  $STPs$  由合适通行起点位姿  $STP_B$ (Suitable Transit

Pose-Begin) 与合适通行终点位姿  $STP_E$ (Suitable Transit Pose-End) 构成, 形式化表达为  $STPs = (STP_B, STP_E)$ 。每个位姿参数由三元组  $(x, y, \theta)$  定义, 其中  $x$  表示地图横坐标值,  $y$  表示地图纵坐标值,  $\theta$  表示机器人的行进方向。

### 3.2.2 导航策略信息生成环节的步骤

导航策略信息生成环节的具体步骤如下:

- (1) 当检测到机器人出现局部轨迹规划失败或运动振荡现象时, 进入步骤 (2);
- (2) 全局成本地图膨胀化, 进入步骤 (3);
- (3) 执行合适通行位姿对生成方法, 当成功生成  $STPs$  时, 认为环境狭窄导致机器人出现步骤 (1) 所述状况, 因此将当前位姿作为导航失效位姿  $NFP$ , 将该  $NFP$  与对应的  $STPs$  形成元组加入  $I$ , 进入步骤 (4); 当生成  $STPs$  失败时, 认为机器人出现步骤 (1) 所述情况由环境其他因素而非环境狭窄导致, 直接判断导航任务失败;
- (4) 将更新后的  $I$  加载至导航策略切换环节, 并尝试重新执行原导航任务。

### 3.2.3 全局成本地图膨胀化

由于本章方法的策略中将使用路径跟踪算法代替轨迹规划算法, 机器人将严格遵循规划出的全局路径行驶, 因此构建安全可行的全局路径是保证本章方法后续执行效果的必要一环。本章针对这一技术需求提出采用全局成本地图膨胀化方法作为解决方案, 该方法能够保证路径规划的可行性与安全性。

SLAM 算法生成的全局静态地图作为环境表征的基础数据来源, 其内在局限性主要体现在以下两个方面: 其一, 该地图通过离散栅格化处理仅能提供静态环境的空闲、占用及未知三元状态分类, 而无法对动态障碍物的时空分布特性进行有效建模, 这导致基于此类先验地图生成的路径规划方案在实际运行中面临动态障碍物规避失效的风险; 其二, 该地图表示方法在几何约束建模层面存在显著缺陷, 其栅格占用判定规则未将移动机器人本体的碰撞体积纳入考量, 即在路径搜索过程中未建立与机器人运动学参数相匹配的障碍物膨胀层计算机制, 从而导致规划算法输出的理论轨迹在机器人实际运动过程中可能引发机器人本体与静态障碍物之间的碰撞。综上, 此类仅包含静态环境信息的原始栅格地图无法直接满足移动机器人运动规划系统对动态环境适应性与本体安全性保障的双重要求。

如图3-4所示, 全局成本地图是基于栅格化表达的层叠环境模型。全局成本地图通

过数值化表征各栅格单元的可通行度，其成本范围为 0 到 255，为经典路径搜索算法（如 Dijkstra 算法、A\* 算法）提供关键的环境特征数据。最基本的模型为三层：

- (1) 静态层：为了做全局规划，机器人需要感知范围超过传感器的地图，因此使用 SLAM 算法生成，该层表示代价地图中大部分不会变化的区域，一般作用于全局成本地图的底层。
- (2) 障碍层：该层从机器人搭载的如激光雷达、超声波雷达等感知传感器中采集数据，并将其转化为栅格地图。
- (3) 膨胀层：前两层叠加后，在每个致命障碍周围插入缓冲区，为障碍物创建安全距离。

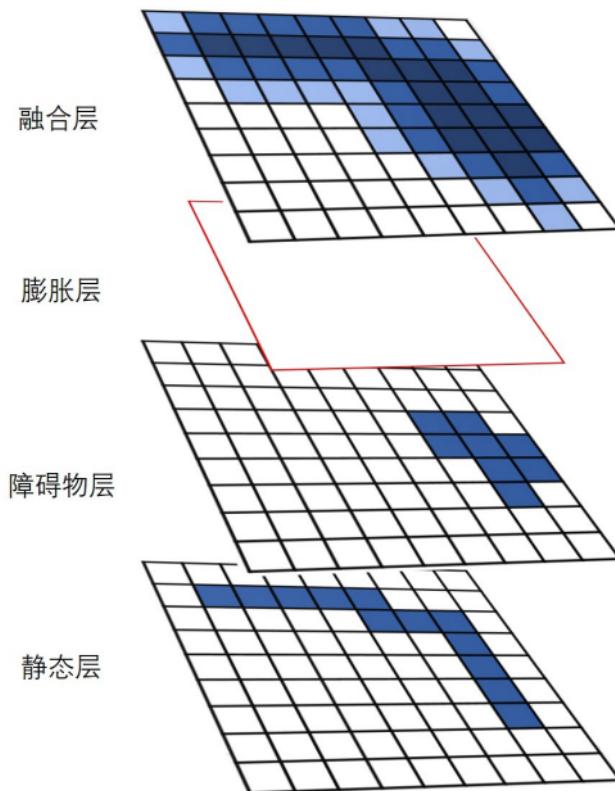


图 3-4 全局成本地图示意图



膨胀层的构建通常采用静态安全边界策略：在障碍物轮廓外延展形成宽度不小于机器人本体包络半径的不可通行区域即致命区域，该区域与原始障碍物具有相同的通行成本参数。该方案存在显著的双重局限性：其一，当安全边界宽度设置过小时，路径规划算法在最小化路径成本的目标函数驱动下倾向于生成紧贴障碍物边缘的优化路径，这将导致机器人实际运行时面临安全隐患；其二，过大的安全边界设置虽能提升路径安全性，但也会导致原本满足机器人通行的狭窄区域被错误标记为不可通行区域从而造成

路径规划失败。为有效解决上述问题，本文提出了动态可调的全局成本地图膨胀公式：

$$C = \begin{cases} C_o, d_o < r_r \\ 0, d_o > r_i \\ C_o * (1 - \frac{d_o - r_r}{r_i}), other \end{cases} \quad (3-2)$$

式3-2中， $C$  表示当前栅格的成本； $C_o$  表示障碍物栅格的成本； $r_r$  表示机器人的机身半径； $d_o$  表示与当前栅格距离最小的障碍物之间的距离； $r_i$  表示全局成本地图的膨胀半径，根据狭窄环境的判断阈值设定，一般设为机器人机身宽度的两倍。

该膨胀公式作图如图3-5所示，

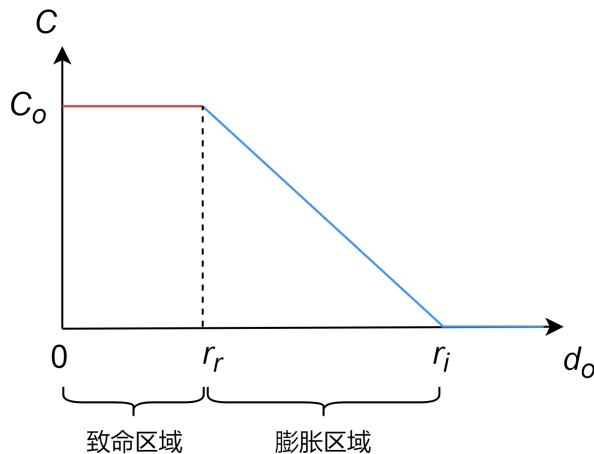


图 3-5 膨胀公式示意图

本方法采用分层膨胀策略优化环境表征：首先在障碍物占据的栅格单元外围构建宽度等于机器人本体包络半径  $r_r$  的致命区域，从几何约束层面规避机器人物理尺寸与规划路径的空间冲突风险；在此安全边界之外设置具有线性衰减特性的膨胀区域，该区域内各栅格的通行成本值随与邻近障碍物的欧氏距离的递增而减小，从而引导路径搜索过程优先选择具有更大安全裕度的栅格节点。该策略的优势在于通过膨胀区域的成本梯度引导机制，既能维持路径在几何空间中的可行性，又能促使规划算法生成具有最优安全间距的全局路径。本方法基于栅格地图的数据结构进行设计，可无缝集成于各类基于栅格化环境建模的导航系统中，具有良好的算法兼容性。

在全局路径规划算法中选择 A\* 算法作为演示案例，图3-6通过可视化对比实验验证了不同的全局成本地图对路径规划效果的影响机制。实验场景中，黑色栅格表示实体障

碍物区域，红色区域对应根据机器人机身半径设置的致命区域，蓝色标记分别表示路径的起始位置与导航目标位置，算法生成的全局路径以蓝色线条呈现，图3-6a采用热力图谱映射方式，通过渐变色阶直观展示栅格成本值与障碍物距离的反向关联特性。从图中可看出，相较于采用传统成本地图的规划结果，本研究所提方法使规划的路径与障碍物间安全间距大大提升，这种路径更有利于机器人行驶的安全性，为后续路径跟踪的执行创造了理想的条件。

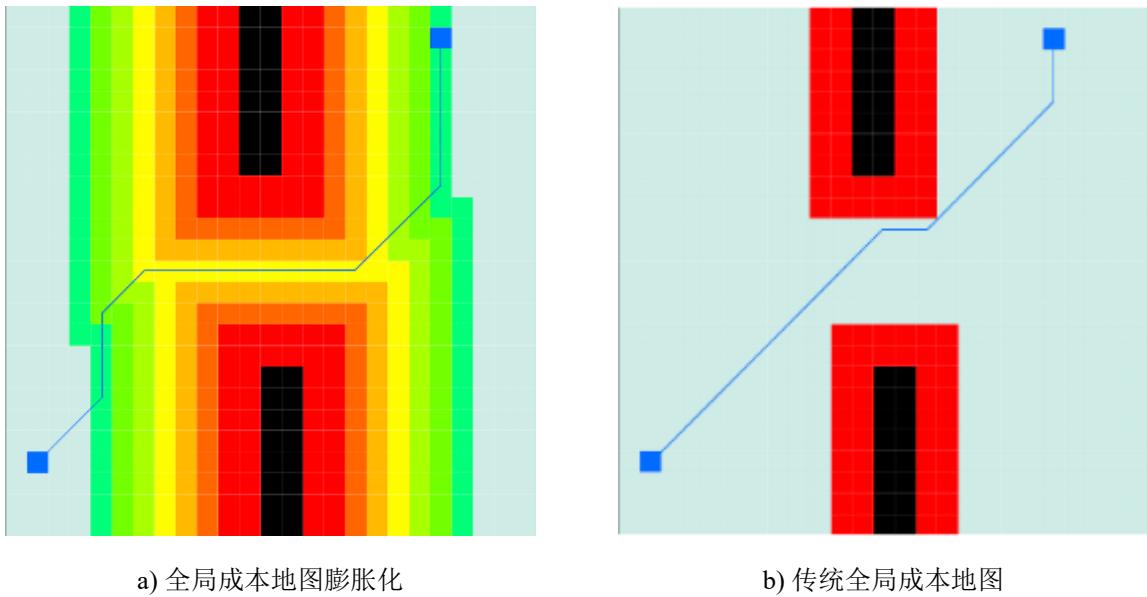


图 3-6 全局成本地图用于规划全局路径的效果对比

#### 3.2.4 合适位姿通行对的生成方法

合适通行位姿对能够引导机器人在进入狭窄环境前调整行进位姿，以安全裕度更大的方式通过。 $STPs$  的生成方法的示意图如图3-7所示，其依据机器人所处的位姿、全局成本地图与全局路径进行计算，尝试在即将行驶的路段中寻找可能存在的狭窄环境，并于入口处生成  $STP_B$ ，出口处生成  $STP_E$ 。具体步骤如下：

- (1) 调用全局路径规划算法重新生成一条由当前位姿至导航目标点的全局路径  $P = \{p_j\}_{j=1,2,3,\dots,m}$ ，其中  $p_j$  为全局路径中的第  $j$  个路径点的位姿，由于在**合适位姿通行对**生成之前已对全局成本地图进行膨胀化，因此路径  $P$  会在全局成本地图的优化作用下，向狭窄通道的几何中线贴近。
- (2) 计算路径  $P$  中各路径点  $p_j$  所处环境的宽度，该宽度序列表示为：

$$W = \{w_j\}_{j=1,2,3,\dots,m} \quad (3-3)$$



其中,  $w_j$  为  $p_j$  所处环境的宽度, 该宽度的计算方式为按一定条件搜索  $w_j$  的第一定义点  $o_{j1}$  和第二定义点  $o_{j2}$ , 然后计算两者之间的欧氏距离, 具体方式如下:

- i) 在全局成本地图中, 以  $p_j$  为圆心, 以栅格地图分辨率  $r_m$  为初始半径作为初始圆形搜索范围, 并同时以  $r_m$  为半径增长步长, 狹窄环境的判断阈值  $\gamma_n$  为最大搜索半径, 逐步扩展圆形范围以搜索距离  $p_j$  最近的障碍点;
- ii) 若在最大搜索半径范围内未能搜索到障碍点, 则计  $w_j$  为最大值  $M$ , 否则将搜索到的第一个障碍点作为第一定义点  $o_{j1}$  如图3-7a所示, 并以  $r_m$  为增长步长, 沿向量  $\overrightarrow{o_{j1}p_j}$  方向搜索第一个障碍点作为第二定义点  $o_{j2}$  如图3-7b所示;
- iii) 计算  $w_j$ :

$$w_j = \sqrt{(x_{o_{j1}} - x_{o_{j2}})^2 + (y_{o_{j1}} - y_{o_{j2}})^2} \quad (3-4)$$

式3-4中  $x_{o_{j1}}$  与  $y_{o_{j1}}$  分别表示  $o_{j1}$  的  $x$  坐标和  $y$  坐标,  $x_{o_{j2}}$  与  $y_{o_{j2}}$  同理。

(3) 按正序遍历  $W$  中的宽度  $w_j$ , 并进行以下操作:

- i) 当访问到第  $a$  个元素  $w_a$ , 且其在遍历过程中首次满足条件  $w_a < \gamma_n$  时, 将  $w_a$  的第一定义点  $o_{a1}$  和第二定义点  $o_{a2}$  位置的中点作为  $STP_B$  的位置, 垂直于向量  $\overrightarrow{o_{a1}o_{a2}}$  且朝向  $\overrightarrow{p_{a-1}p_a}$  的方向作为  $STP_B$  的方向;
- ii) 继续从  $W$  的第  $a+1$  个元素开始访问, 当访问到第  $b(b > a)$  个元素  $w_b$ , 且其首次满足  $w_b < \gamma_n$  时, 将  $p_b$  的位置作为  $STP_E$  的位置, 垂直于向量  $\overrightarrow{o_{b1}p_b}$  且朝向  $\overrightarrow{p_{b-1}p_b}$  的方向作为  $STP_E$  的方向, 其中  $o_{b1}$  是  $w_b$  的第一定义点, 生成的  $STPs$  如图3-7c所示; 若  $W$  遍历完毕后未能同时找到满足相应条件的  $w_a$  和  $w_b$ , 则  $STPs$  生成失败, 导航任务中止。

(4) 若  $STPs$  成功生成, 则根据全局成本地图对其位置进行局部的成本优化:

- i) 以  $STP_B$  为圆心,  $\overrightarrow{o_{a1}STP_B}$  所在直线为分界,  $STP_B$  的反方向为朝向, 机器人机身半径  $r_r$  为半径形成圆形搜索范围, 在该范围内寻找成本最小的点形成候选点集  $S_n = \{c_i\}_{i=1,2,3,\dots,n}$ , 并依次计算向量  $\overrightarrow{STP_Bc_i}$  的方向与角度  $|\theta_{STP_B} - \pi|$  之间的角度差, 将使得该角度差最小的点的位置作为  $STP_B$  优化后的位置, 其中  $\theta_{STP_B}$  表示位姿  $STP_B$  的方向;
- ii) 以  $STP_E$  为圆心,  $\overrightarrow{o_{b1}STP_E}$  所在直线为分界,  $STP_E$  为朝向, 机器人机身半径  $r_r$  为半径形成圆形搜索范围, 在该范围内寻找成本最小的点形成候选点集  $S'_n = \{c'_i\}_{i=1,2,3,\dots,n'}$ , 并依次计算向量  $\overrightarrow{STP_Ec'_i}$  的方向与  $\theta_{STP_E}$  之间的角度差,

将使得该角度差最小的点的位置作为  $STP_E$  优化后的位置，其中  $\theta_{STP_E}$  表示位姿  $STP_E$  的方向；  
优化后的  $STPs$  如图3-7d所示。

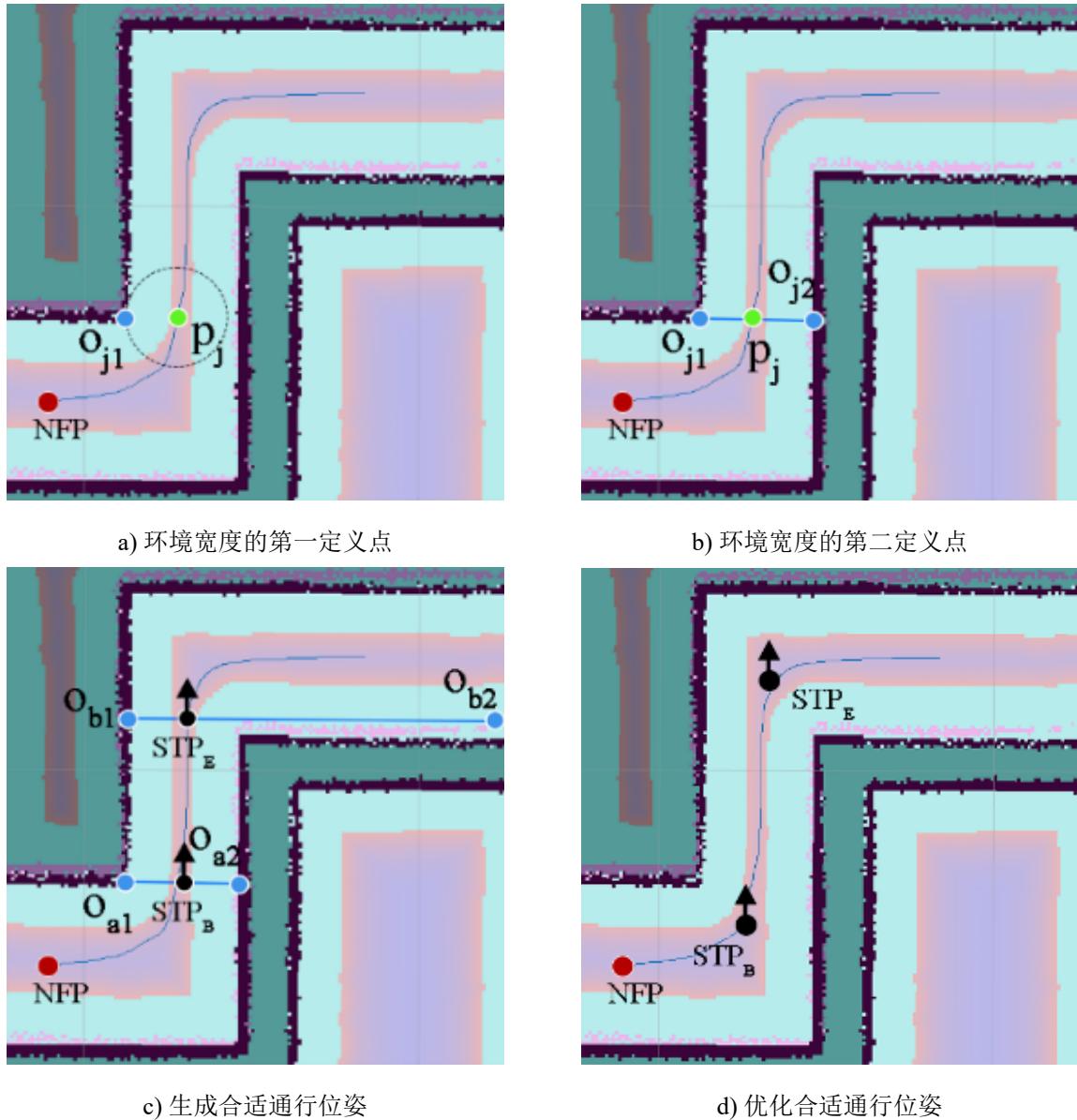


图 3-7 合适通行位姿对生成方法的示意图

### 3.3 导航策略切换环节

#### 3.3.1 导航策略切换环节的步骤

导航策略切换环节通过实时分析环境参数与先前生成的导航策略信息  $I$  进行比对，触发决策以实现导航策略的切换，其具体步骤如下：

- (1) 机器人在导航任务执行中基于预设周期触发狭窄环境进入判断方法，当检测结果

为真时，同步输出与当前判断结果相匹配的  $STPs$ ，并执行步骤（2）；

- (2) 机器人将导航策略由正常导航策略切换为狭窄环境导航策略并执行：首先进行全局成本地图的膨胀化，然后调用全局路径规划器分别规划出从当前位置到  $STP_B$ 、从  $STP_B$  到  $STP_E$  两段路径，最后使用改进的路径跟踪算法代替局部轨迹规划算法来生成轨迹，以完成这两段路径的导航过程；
- (3) 机器人导航至  $STP_E$  后，说明已离开狭窄环境，导航策略由狭窄环境导航策略切换至正常导航策略，继续执行从  $STP_E$  至导航目标点的导航任务。

### 3.3.2 判断进入狭窄环境的方法

狭窄环境进入判断方法如图3-8所示，其主要依据当前位置与  $I$  中的导航失效点  $NPF$  之间的距离和当前全局路径与距离机器人最近的  $NPF$  对应的  $STPs$  的匹配程度来判断机器人是否即将进入狭窄环境。具体步骤如下：

- (1) 遍历  $I$ ，计算  $NPF$  与当前位置的距离，并将该距离小于预设定阈值  $\gamma_d$  的  $NPF$  加入候选点集  $S_q = \{NPF_k\}_{k=1,2,3,\dots,q}$ ，其中  $\gamma_d$  越大，机器人将在距离狭窄环境越远的位置开始执行狭窄环境导航策略，用户可根据自身需求进行调整。
- (2) 依次计算  $S_q$  中的每个  $NPF_k$  所对应的合适通行位姿对  $STPs_k$  与当前全局路径  $P$  的匹配度  $D_k$ ，其具体计算方法如下：
  - i) 按正序在  $P$  中寻找首个满足以下条件的路径点  $p_a$ ：

$$\begin{cases} \Delta(p_a, STP_{B_k}) < \theta_d/2 \\ \Theta(p_a, STP_{B_k}) < \pi/2 \end{cases} \quad (3-5)$$

其中， $\Delta$  为计算两个路径点之间欧氏距离的算子， $\Theta$  为计算两个路径点之间的角度差的算子，并记：

$$\begin{cases} d_{B_k} = \Delta(p_a, STP_{B_k}) \\ \theta_{B_k} = \Theta(p_a, STP_{B_k}) \end{cases} \quad (3-6)$$

若  $P$  中没有符合式3-5条件的路径点，则令  $D_k = 0$ 。

- ii) 若  $p_a$  存在，则继续从  $P$  的第  $a+1$  个元素起寻找首个满足以下条件的路径点

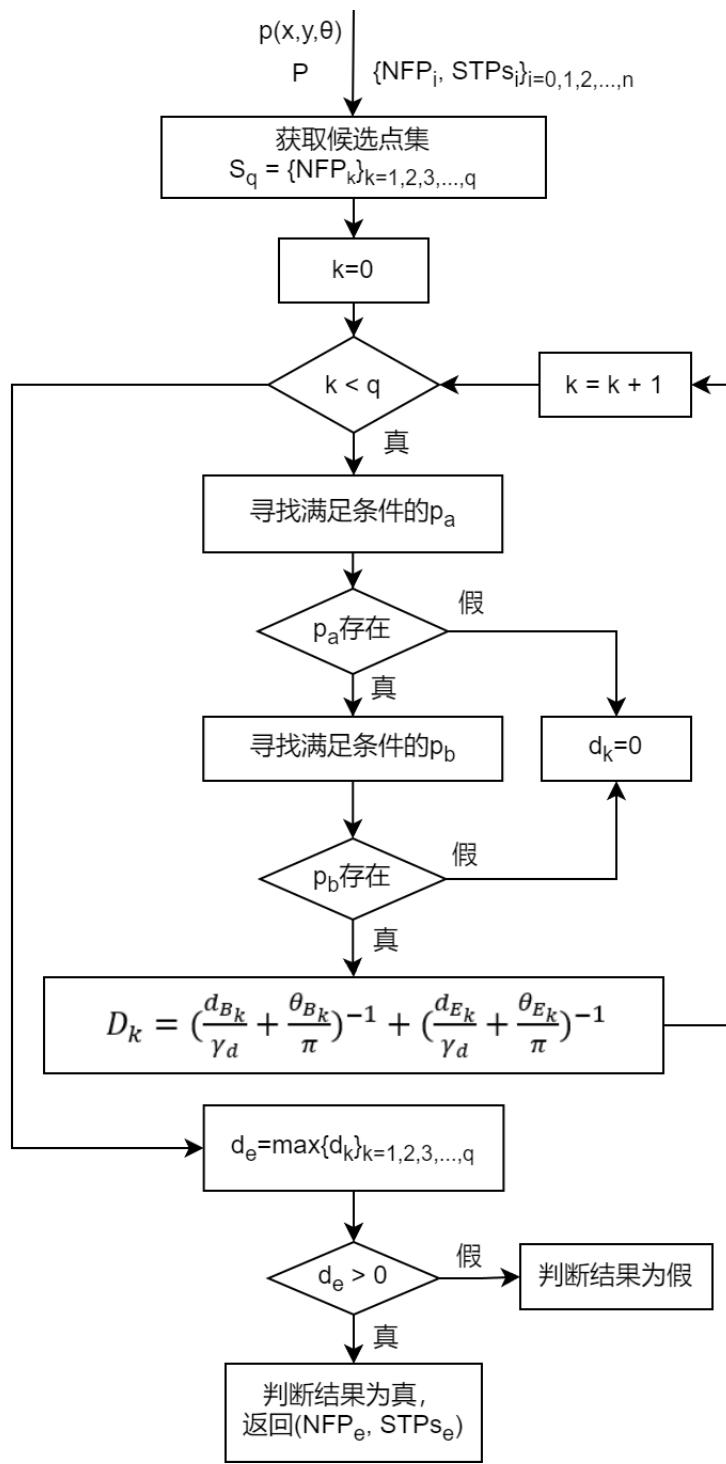


图 3-8 狹窄环境进入判断方法的流程图

 $p_b$ :

$$\begin{cases} \Delta(p_b, STP_{E_k}) < \theta_d/2 \\ \Theta(p_b, STP_{E_k}) < \pi/2 \end{cases} \quad (3-7)$$

并记：

$$\begin{cases} d_{E_k} = \Delta(p_b, STP_{E_k}) \\ \theta_{E_k} = \Theta(p_b, STP_{E_k}) \end{cases} \quad (3-8)$$

同样的，若  $P$  中没有符合式3-7条件的路径点，则令  $D_k = 0$ 。

iii) 当同时找到符合条件的  $p_a$  和  $p_b$  时，则可以计算  $D_k$ ，计算公式为：

$$D_k = \left( \frac{d_{B_k}}{\gamma_d} + \frac{\theta_{B_k}}{\pi} \right)^{-1} + \left( \frac{d_{E_k}}{\gamma_d} + \frac{\theta_{E_k}}{\pi} \right)^{-1} \quad (3-9)$$

(3) 令  $D_e = \max\{D_k\}_{k=1,2,3,\dots,q}$ ，当  $D_e > 0$  时，狭窄环境进入判断为真，并返回对应的导航策略信息  $(NFP_e, STPs_e)$ ；否则判断为假。

### 3.3.3 改进路径跟踪算法

路径跟踪算法与局部轨迹规划方法在系统设计层面具有显著的策略性差异，相较于需要实时生成局部轨迹并同步完成避障决策的局部轨迹规划方法，路径跟踪算法主要依托预生成的全局路径进行位姿调控，其核心在于建立高精度的误差反馈机制以提升路径跟随精度。这种设计范式通过规避复杂的动态避障计算，有效降低系统对计算资源的消耗且简化了运动控制难度，适用于存在严格运动学约束的机器人平台。

在狭窄场景的静态障碍物规避场景中，路径跟踪算法的安全性主要来源于全局路径的安全性。在全局成本地图膨胀化后，生成的全局路径处于通道中线区域，已满足机器人本体尺寸与通道边界的几何约束，能够确保在保持理想跟踪精度的情况下其运动包络线始终处于安全边界之内。对于动态障碍物的处理，系统通过激光点云数据实时检测运动障碍物的状态，其避障策略为：当行进方向出现障碍物时，触发紧急制动机制进入等待状态；待障碍物移出预设安全阈值后，自动恢复路径跟踪任务。

当前路径跟踪领域的技术体系已形成多种经典控制方法并存的格局，其中具有代表性的解决方案包括基于几何追踪模型的 Pure Pursuit 算法、采用误差反馈机制的 PID 控制器、基于最优控制理论的线性二次调节器（LQR），以及融合预测控制思想的模型预测控制（MPC）等。Varma 等<sup>[64]</sup>通过建立轨迹跟踪精度、系统鲁棒性、计算实时性等多维度评估指标体系对主流算法进行量化对比研究，结果表明：在理想情况下，MPC 凭借其滚动优化与反馈校正机制展现出最优的综合性能，LQR 因具备状态空间的最优调节能力次之，而传统 PID 控制受限于固定参数调节特性，其动态响应特性相对较弱。然而研究同时指出，MPC 在工程部署中存在显著瓶颈——其核心算法需要在线求解多步

预测时域内的非线性优化问题，这导致计算复杂度呈现指数级增长，即便在 RK3399 等具备 Cortex-A72 高性能核心的低功耗计算平台上，MPC 的迭代求解过程仍会产生超过 100ms 的运算延迟，致使系统难以满足移动机器人对控制周期（通常 <50ms）的实时性要求，在第五章中针对本章内容的真实实验中能够体现该问题。这种计算性能与理论优势间的矛盾严重制约了其实际应用价值。

在局部轨迹规划理论框架内，ROSMANN 等<sup>[65]</sup>通过理论建模揭示了时间弹力带算法与模型预测控制算法之间的内在关联性。研究指出，TEB 算法的本质可解构为基于 MPC 框架的特殊实现形式：该算法将机器人运动学约束抽象为位姿空间内的平面刚体系统模型，通过时间弹性带理论对状态序列进行离散化建模。其创新性在于采用有限差分法对连续位姿变化量进行数值近似，将复杂的非线性最优控制问题转化为离散状态点的几何约束优化问题。算法通过引入二次型惩罚函数构建复合目标函数，将带约束的非线性规划重构为稀疏矩阵表征的无约束最小二乘问题，这种数学重构显著降低了 Hessian 矩阵的求解维度，从而在保证优化精度的前提下实现计算效率的阶跃式提升。本文提出的改进路径跟踪算法基于 MPC 算法的优化思想对二轮差速驱动机器人的运动学特性进行模型适配，将机器人的运动学在位姿空间中以平面系统模型的形式进行几何表达，采用有限差分位姿的方式得到非线性规划，并利用二次惩罚函数将非线性规划转化为无约束最小二乘问题以优化整体计算过程。

改进的路径跟踪算法将生成的轨迹在位姿空间中有限差分为  $n$  个位姿  $P_i = (x_i, y_i, \theta_i)$  构成位姿序列  $Q$ ， $n - 1$  个由相邻连续位姿之间的时间间隔  $\Delta T_i$  构成的时间间隔序列  $\tau$ ，两者组合表示轨迹  $G$ ：

$$Q = \{P_i\}, i = 1, 2, 3, \dots, n \quad (3-10)$$

$$\tau = \{\Delta T_i\}, i = 1, 2, 3, \dots, n - 1 \quad (3-11)$$

$$G = (Q, \tau) \quad (3-12)$$

在 MPC 中，在非完整运动学、动力学等约束下，根据轨迹  $G$  与路径之间的误差能够构建出相应目标函数，从而形成非线性规划问题，可以通过若干二次惩罚函数  $f_j$  转换为无约束最小二乘问题：

$$G^* = \operatorname{argmin}_G \sum_j \gamma_j f_j(G), j \in \{p, a, l, n\} \quad (3-13)$$

在若干惩罚函数  $f_j$  中,  $f_p$  表示远离全局路径的惩罚函数, 如式3-14所示。其中  $d_{min,j}$  为机器人当前位姿与全局路径的路径点之间的最小距离,  $r$  表示机器人与路径点之间的距离阈值,  $\varepsilon$  为偏移因子,  $S$  为缩放比例。

$$f_p = \begin{cases} \left(\frac{d_{min,j} - (r - \varepsilon)}{S}\right)^2, & d_{min,j} > r - \varepsilon \\ 0, & d_{min,j} \leq r - \varepsilon \end{cases} \quad (3-14)$$

$f_l$  为线速度与线加速度的几何约束形式的惩罚函数,  $f_a$  为角速度与角加速度的几何约束形式的惩罚函数, 分别表示为式3-15和式3-16。

$$f_l = (v_i - (v_{i-1} + a_i * \Delta T_i))^2 \quad (3-15)$$

$$f_a = (\omega_i - (\omega_{i-1} + a_{\omega,i} * \Delta T_i))^2 \quad (3-16)$$

其中  $v_i$  和  $a_i$  分别是机器人在位姿  $P_i$  时的线速度和线加速度,  $\omega_i$  和  $a_{\omega,i}$  分别表示机器人在位姿  $P_i$  时的角速度和角加速度, 它们通过相邻位姿的平均速度计算近似值:

$$v_i = \left\| \begin{pmatrix} x_{i+1} - x_i \\ y_{i+1} - y_i \end{pmatrix} \right\| / \Delta T_i \quad (3-17)$$

$$\omega_i = (\theta_{i+1} - \theta_i) / \Delta T_i \quad (3-18)$$

$$a_i = 2(v_{i+1} - v_i) / (\Delta T_i + \Delta T_{i+1}) \quad (3-19)$$

$$a_{\omega,i} = 2(\omega_{i+1} - \omega_i) / (\Delta T_i + \Delta T_{i+1}) \quad (3-20)$$

$f_n$  是非完整运动学约束的惩罚函数。差速驱动机器人模型的运动学约束要求其只能在当前位姿的前进方向上运动, 这使得机器人的相邻位姿必须位于如图3-9所示的公共曲率弧线上。满足这一条件的数学表述为  $P_i$  方向与  $d_{i,i+1}$  方向的夹角  $\beta_i$  应和  $d_{i,i+1}$  与  $P_{i+1}$  方向的夹角  $\beta_{i+1}$  相等, 即:

$$\begin{pmatrix} \cos\theta_i \\ \sin\theta_i \end{pmatrix} \times d_{i,i+1} = d_{i,i+1} \times \begin{pmatrix} \cos\theta_{i+1} \\ \sin\theta_{i+1} \end{pmatrix} \quad (3-21)$$

$$d_{i,i+1} = \begin{pmatrix} x_{i+1} - x_i \\ y_{i+1} - y_i \end{pmatrix} \quad (3-22)$$

则  $f_n$  表示为:

$$f_n = \left\| \left[ \begin{pmatrix} \cos\theta_i \\ \sin\theta_i \end{pmatrix} + \begin{pmatrix} \cos\theta_{i+1} \\ \sin\theta_{i+1} \end{pmatrix} \right] \times d_{i,i+1} \right\|^2 \quad (3-23)$$

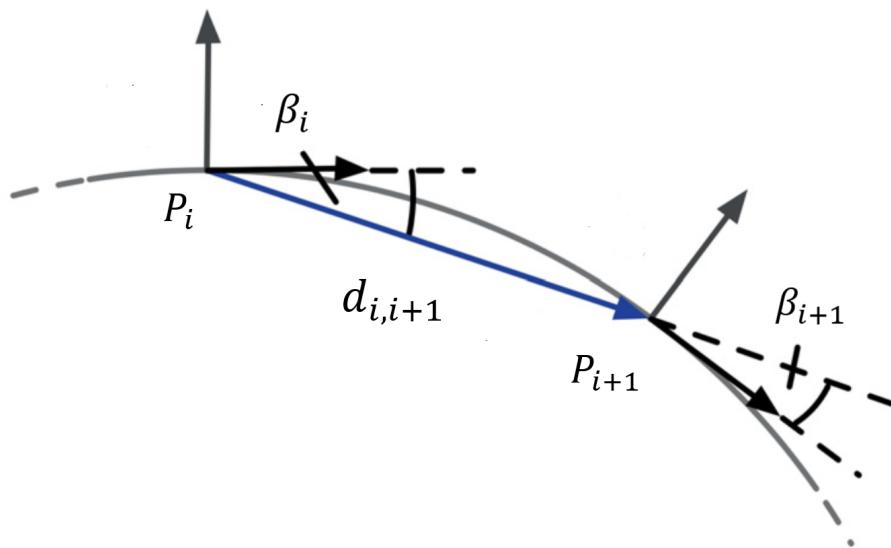


图 3-9 非完整运动学

根据式3-13所示的最小二乘优化问题的数学特性分析，其目标函数中的各组成项具有局部关联特性，即每个变量仅与轨迹  $G$  中相邻的有限位姿节点产生依赖关系。这种局部依赖性使得对应的系统矩阵呈现高度稀疏结构，从而能够运用高效的稀疏矩阵求解方法，如基于图论优化的 g2o 算法<sup>[66]</sup>进行快速求解。具体实现时，算法将优化问题建模为超图结构如图3-10所示：其中轨迹  $G$  的离散位姿节点及时间间隔参数定义为超图顶点，而表征运动学约束、路径跟踪误差等二次惩罚项则构建顶点间的拓扑连接关系。基于这种图模型表达，g2o 框架通过迭代执行顶点位姿调整与边权重更新，最终将收敛于满足多约束条件的最优轨迹解  $G^*$ 。

### 3.4 本章小结

本章主要阐述了自适应狭窄环境的室内导航系统的设计与实现。与传统的室内导航系统相比，本系统创新性地引入了导航策略信息生成环节与导航策略切换环节，实现了机器人对狭窄环境的自动识别与导航策略的动态调整。本章首先概括传统室内导航系统

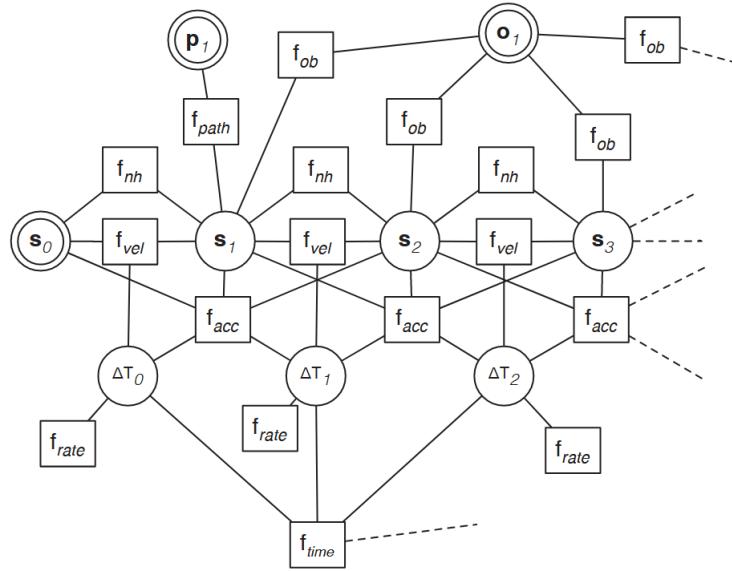


图 3-10 超图的简化示例结构

在路径与轨迹规划阶段暴露的问题，引入并分析了系统的整体架构以及工作流程，详细说明了流程中每个重要步骤对改善问题的作用。然后本章详细介绍了导航策略信息生成环节的实现方法，涵盖了导航策略信息的描述方式、该环节的整体步骤、全局成本地图的膨胀处理以及**合适位姿通行对**的生成方法。通过这一环节，系统能够有效识别并标记狭窄环境，还为后续的导航策略切换提供数据支持。最后是导航策略切换环节，本章重点阐述了策略切换的整体步骤、狭窄环境进入判断方法以及路径跟踪算法的优化思路。这些改进使得机器人能够根据环境变化灵活切换导航策略，确保在狭窄环境中安全高效地通行。综上所述，本章提出的自适应狭窄环境室内导航系统能够显著提升机器人在室内狭窄环境中的导航能力。

## 第四章 基于豪斯多夫距离的点云匹配示教重复算法

本章将提出基于豪斯多夫距离的点云匹配示教重复算法，作为自适应狭窄环境的室内导航系统中的补充策略。首先介绍提出该算法的背景；然后说明该算法的整体框架，包括算法设计与工作流程；接着详细阐述了用于定义点云帧间差异的豪斯多夫距离的计算、优化方法以及点云匹配定位的原理；最后描述根据李雅普诺夫稳定性理论构造的点云管道跟踪控制器以及最终生成运动控制信息的过程。

### 4.1 引言

基于 SLAM 建图的室内导航技术从传感器海量的感知信息中提取环境特征，将其与已知的地图数据进行匹配以实时获取机器人的位姿，并根据在地图中建模的实况环境规划轨迹，最终到达目标导航点。导航系统的过程复杂性引入了较多误差源，如激光雷达测距误差、地图分辨率量化误差、定位漂移等因素，定位不准确会直接影响后续运动规划和控制的有效性；系统中感知、定位、规划和控制需要同时高效处理也使得计算复杂性提高，一旦计算产生延迟就可能导致机器人控制失稳甚至任务失败。上述问题在狭窄环境中体现得尤为突出：激光雷达测距误差和机器人姿态的调整，可能会导致原本探测到的可行空间减小或消失；定位漂移则会严重影响后续的轨迹规划。

本文第三章内容的思路是提高机器人对误差源的容忍性，而另一种思路则是减少误差源，该思路对于机器人自主穿越狭窄环境有着更大的潜力，因此本章设想一种在狭窄环境中导航时不使用室内导航系统，而是利用传感器信息直接生成运动控制信息的示教重复方法。这种方法将传感器信息视为一个集合，并设计一个非向量空间的控制器来引导当前集合向所期望的集合转换，从而消除了向量空间方法中的定位误差和地图分辨率量化误差，提高了计算效率，并为传感器噪声和不准确性提供了更大的健壮性。

这种非向量空间方法与基于 SLAM 建图的室内导航系统这类向量空间方法的根本不同之处在于，它定义集合之间的误差，而非向量之间的误差。因此需要新的数学工具来处理非向量空间中的集合动力学。如 Aubin<sup>[67][68]</sup>和 Doyen<sup>[69][70]</sup>提出非向量空间的动力学基于突变分析，这种方法已经应用于视觉伺服控制<sup>[71]</sup>和纳米级操作<sup>[72]</sup>等领域。到目前为止，还没有一个全面的算法可以应用非向量空间理论来解决移动机器人导航中的感知、定位、运动规划和控制任务问题。本文提出基于豪斯多夫距离的点云匹配示教重复算法。该算法包括：(1) 基于豪斯多夫距离的点云匹配解决非向量空间定位问题；(2) 在重复阶段使用基于李雅普诺夫理论构造的控制器对示教阶段录制的点云管道进行跟

踪控制。

最后在传感器的选择上，由于运动控制完全依赖感知信息，因此需要更丰富的感知信息保证方法的安全性和可靠性。单线激光雷达提供的二维环境信息在空间完整性和深度感知等方面都存在局限，尤其在狭窄环境中平面特征较为单一，容易导致不同场景下的特征相似。感知到环境的改变。三维激光雷达能够全向立体地感知环境，目前 SLAM 导航级别的多线机械式激光雷达如镭神智能、禾赛、万集等品牌的成本均在一万元左右，受限于商业机器人的成本问题，本文使用价格不到四千元的大疆的 Livox Mid-360 三维激光雷达如图4-1所示，其不同于传统机械式的多线激光雷达，通过混合固态技术将水平视场角提升到 360° 的同时，垂直视场角也达到 59°，能够感知到球形范围的三维环境，提供更全面的点云数据。



图 4-1 Livox Mid-360 三维激光雷达

## 4.2 算法整体框架

基于豪斯多夫距离的点云匹配示教重复算法的框架如图4-2所示。该算法作为自适应狭窄环境的室内导航系统中的补充策略，在机器人识别到狭窄环境后，通过重复预先示教的轨迹的方式来穿越狭窄环境。具体地，在示教阶段手动操纵机器人在狭窄环境中通行，过程中按一定频率记录激光雷达观测的点云数据形成序列，定义该点云数据的序列为点云管道，表达如式4-1：

$$K(s) = \{P_s\}, s = 1, 2, \dots, n \quad (4-1)$$

其中  $s$  表示点云管道  $K$  中点云的每一帧从 1 到  $n$  的索引， $P_s$  是点云管道中的某一帧。在重复阶段，机器人根据当前观测的点云数据与点云管道进行点云匹配定位，取当前定位的点云帧的后若干帧作为目标点云帧，使用点云管道跟踪控制器生成由当前帧至目标

帧的运动控制信息，循环若干轮后，机器人完成重复轨迹的任务。

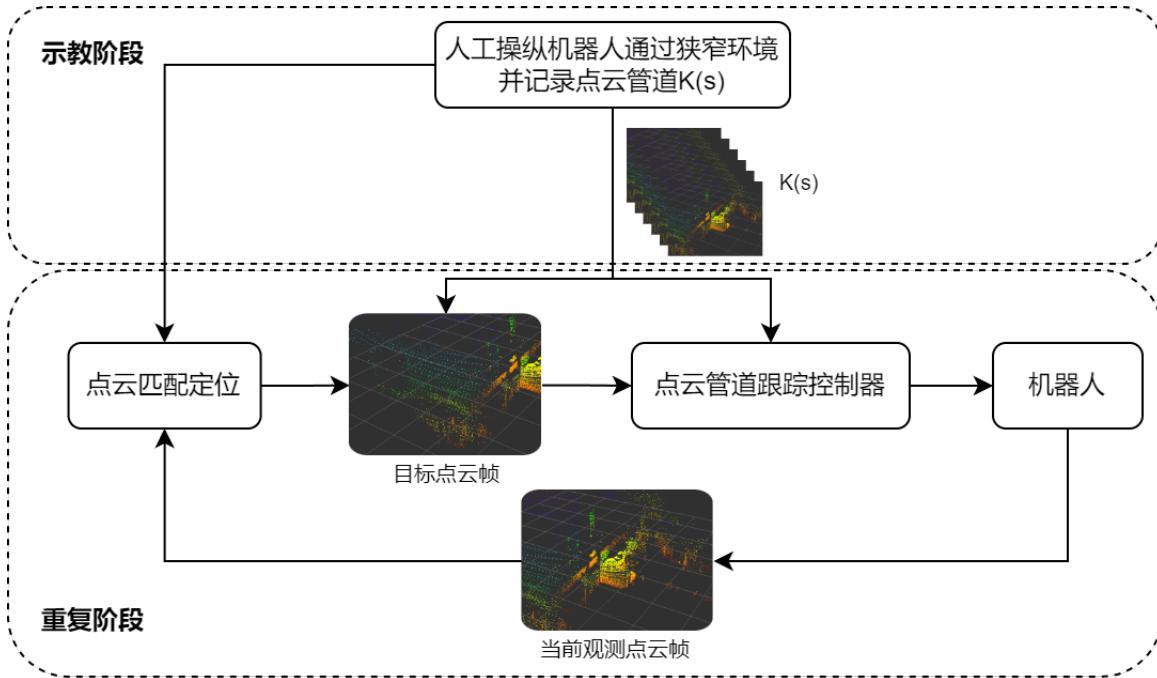


图 4-2 基于豪斯多夫距离的点云匹配示教重复算法的框架

### 4.3 基于豪斯多夫距离的点云匹配定位

在基于豪斯多夫距离的点云匹配示教重复算法中，机器人的姿态不再通过基于全局地图坐标系中的定位算法来确定，这将导致机器人无法确定其当前状态以完成导航任务。在基于事件的理论应用于基于非向量事件的方法之前，它已经成功地应用于远程操作<sup>[73]</sup>、移动机器人导航<sup>[74]</sup>等应用中，其核心思想是找到一个独立于时间的事件参考，该参考与传感器的实时测量有关而非时间相关。因此，本文引入了基于事件的理论，其中点云管道被描述为从事件参考到一组点云的映射。在本章提出的算法中，定位从确定机器人位姿转变为沿点云管道搜索与当前观测状态最相似的点云帧。

在非向量空间理论中，机器人的状态被描述为集合而非位姿向量。本章提出的算法中使用三维激光雷达输出的点云集合来描述机器人状态，通过匹配当前点云集合与点云管道中的点云集合来确定机器人当前的定位信息，因此需要为集合定义新的指标。豪斯多夫距离是一种用于衡量两个有限闭集合之间最大偏差的度量，其优势在于：(1) 能够有效地衡量两个点集之间的最大偏差，关注最坏情况的差异，这使得其在形状匹配和轮廓分析中非常有用；(2) 与切比雪夫距离、DTW 等其他度量方法相比，它无需点对点的配对，能够直接评估整个点集的差异；(3) 局部噪声点对其影响较小，适用于具有部分遮挡或缺失数据的情况，在处理复杂或存在遮挡的激光点云数据时拥有更强的健壮

性。目前豪斯多夫距离在计算机视觉、医学图像分析和三维物体匹配中表现尤为亮眼。综上，本章选择使用豪斯多夫距离描述两个激光点云集合之间的差异，其数学描述如下：

设空间中存在两个点云集合  $A = \{a_1, a_2, \dots, a_p\}$  和  $B = \{b_1, b_2, \dots, b_q\}$ ，则点云  $A$  和  $B$  之间的豪斯多夫距离定义为两者之间的极大极小距离，如公式4-2：

$$H(A, B) = \max[h(A, B), h(B, A)] \quad (4-2)$$

式中，

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\| \quad (4-3)$$

$$h(B, A) = \max_{b \in B} \min_{a \in A} \|b - a\| \quad (4-4)$$

其中  $A, B \in \mathbb{R}^n$ ,  $a, b \in \mathbb{R}^3$ ,  $\|\cdot\|$  是两点之间的欧氏距离。

算法2是计算豪斯多夫距离的伪代码，从中可看出，其时间复杂度为  $O(n_A n_B)$ ，其中  $n_A$  和  $n_B$  分别为点集  $A$  和  $B$  的元素数量，平方级的时间复杂度使得该算法的计算效率在面对数量庞大的点云数据时表现不佳。由于计算豪斯多夫距离是生成运动控制信息的其中一环，其计算效率对机器人最终的运动控制效果影响重大，因此本文从两个方向提高计算效率：其一，过滤点云数据；其二，并行化计算豪斯多夫距离。

本文过滤点云数据的步骤如下：

- (1) 下采样：体素滤波器将点云划分成小的立方体区域（体素），用每个体素内的点的均值或质心来表示该区域从而达到降低点云密度并保留点云主要特征的效果。
- (2) 限定点云范围：首先根据 Z 轴值过滤点云，保留 Z 值在 0 到 2 米范围内的点（更高的位置基本无有价值的点云特征），然后计算每个点到原点的距离，删除距离原点大于 2 米的点。
- (3) 过滤离群点：使用半径离群点滤波器过滤点云，即根据指定半径（0.3 米）和最小领域点数量（10 个）去除在该半径内领域点数量不足的离群点，避免点云噪声对后续计算的影响。

图4-3为点云过滤前后对比，可以看出过滤后的点云数据在保留特征的同时更加干净且数量有效减小。

豪斯多夫距离的计算过程可分解为两个方向的最大最小距离搜索：首先遍历点集  $A$  中的每个点，计算其到点集  $B$  中所有点的最小欧氏距离，并记录全局最大值；随后

---

**Algorithm 2** 计算点集 A 和 B 的豪斯多夫距离

---

**Input:** 点集 A, 点集 B**Initialization:** 点集 A 到 B 的单向豪斯多夫距离  $d_{AB} \leftarrow 0$ , 点集 B 到 A 的  $d_{BA} \leftarrow 0$ 

```
1: for each point  $a \in A$  do
2:    $d_a \leftarrow \infty$ 
3:   for each point  $b \in B$  do
4:      $d_{\text{current}} \leftarrow \text{CalculateDistance}(a, b)$  // 计算点 a 和点 b 之间的欧氏距离
5:     if  $d_{\text{current}} < d_a$  then
6:        $d_a \leftarrow d_{\text{current}}$ 
7:     end if
8:   end for
9:    $d_{AB} \leftarrow \max(d_{AB}, d_a)$ 
10: end for
11: for each point  $b \in B$  do
12:    $d_b \leftarrow \infty$ 
13:   for each point  $a \in A$  do
14:      $d_{\text{current}} \leftarrow \text{CalculateDistance}(b, a)$  // 计算点 a 和点 b 之间的欧氏距离
15:     if  $d_{\text{current}} < d_b$  then
16:        $d_b \leftarrow d_{\text{current}}$ 
17:     end if
18:   end for
19:    $d_{BA} \leftarrow \max(d_{BA}, d_b)$ 
20: end for
21: return  $\max(d_{AB}, d_{BA})$  // 输出点集 A 和 B 的豪斯多夫距离
```

---

对点集 B 到 A 执行对称操作，最终取两者最大值作为最终距离。针对该任务的计算密集型特性以及 CPU 的多核架构，本文引入广泛应用于多核 CPU 并行编程的 OpenMP (Open Multi-Processing) 多线程编程框架<sup>[75]</sup>帮助处理计算过程，其核心价值在于让开发者以极低的代码改动成本，高效发挥多核处理器的计算潜力。OpenMP 通过简单易懂的编译语句将传统的串行代码快速转换为并行程序且无需开发者手动管理线程的创

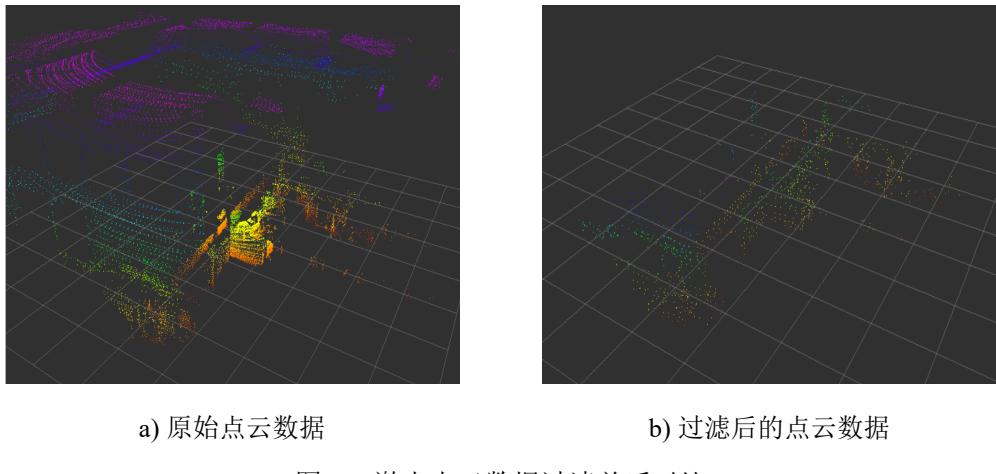


图 4-3 激光点云数据过滤前后对比

建与销毁，大幅降低了多核编程的门槛，且不同于 CUDA（面向 GPU 加速），OpenMP 专注于单机多核场景，成为 CPU 密集型任务轻量化并行的首选工具。因此，本文基于 OpenMP 多线程编程框架设计以下计算优化策略：

- (1) 任务分解与线程级并行：由于点集  $A$  和  $B$  中各点的最小距离计算相互独立，可将外层循环（遍历  $A$  和  $B$  中的每个点）分解为多个并行任务，即通过 OpenMP 的 `#pragma omp parallel for` 指令为每个线程分配一组点，线程间无数据依赖，实现计算任务的粗粒度并行。
- (2) 最小距离计算的向量化加速：在每个线程内部，针对单个点  $a \in A$  到点集  $B$  的全遍历过程，利用循环展开和编译器自动向量化优化欧氏距离的计算，降低内层循环的指令级延迟。
- (3) 全局最大值的归约同步：各线程在完成局部点的最小距离计算后，需将局部最大值同步至全局结果，通过 OpenMP 的 `reduction(max:variable)` 子句，隐式实现多线程结果的原子性归约，避免显式锁机制带来的性能损耗。

点云匹配算法根据豪斯多夫距离量化当前观测点云与点云管道中各点云帧之间的差异，并与差异最小的点云帧进行匹配，作为机器人的当前定位，为后续跟踪管道提供关键支持，其伪代码如算法3。

#### 4.4 点云管道跟踪控制

在重复阶段，机器人进行点云匹配定位后得到当前观测点云数据对应点云管道中的点云帧，若要令机器人向目标点云帧移动以达到点云管道跟踪的效果，就需要计算当前点云帧与目标点云帧之间转换的运动控制信息。基于非向量空间控制理论，本文设计

**Algorithm 3** 基于豪斯多夫距离的点云匹配定位算法

**Input:** 当前激光雷达观测的点云帧  $K_{Lidar}$ , 点云管道  $K(s)$

**Initialization:** 最小豪斯多夫距离  $d_{min}$ , 最相似的点云帧序号  $i_{min}$

```

1:  $d_{min} \leftarrow \infty$ ,  $i_{min} \leftarrow -1$ 
2: for  $s = 1$  to  $\text{Length}(K(s))$  do
3:    $d \leftarrow \text{CalculateDistance}(K_{Lidar}, K(s))$  // 计算当前点云帧与点云管道中的点云帧的
   豪斯多夫距离;
4:   if  $d < d_{min}$  then
5:      $d_{min} \leftarrow d$ ,  $i_{min} \leftarrow s$ 
6:   end if
7: end for
8: return  $i_{min}$ 

```

点云管道跟踪控制器，该稳定控制问题可以表述为：给定一个目标集合  $\hat{K}$  和当前集合  $K$ , 设计一个反馈控制器  $u = \gamma(K(t))$ , 其中集合  $K(t) \in \mathbb{R}^n$  中的  $K(\cdot)$  定义为  $\mathbb{R}^n \mapsto 2^{\mathbb{R}^n}$ ,  $2^{\mathbb{R}^n}$  表示  $\mathbb{R}^n$  的幂集包括  $\mathbb{R}^n$  的所有子集,  $\gamma(K(t))$  是反馈映射, 该映射应使得  $K(t)$  在  $t \rightarrow \infty$  时收敛于  $\hat{K}$ 。李雅普诺夫稳定性理论广泛应用于控制器设计任务中, 因此本文基于豪斯多夫距离构造以下李雅普诺夫函数  $V(K)$ :

$$V(K(t)) = \int_K d_{\hat{K}}^2(x) dx + \int_{\hat{K}} d_K^2(\hat{x}) d\hat{x} \quad (4-5)$$

设  $\xi : E \mapsto \mathbb{R}^n$ , 其中  $E \in \mathbb{R}^n$  为有界 Lipschitz 函数, 表示所有如  $BL(E, \mathbb{R}^n)$  的此类函数的集合。则  $\xi \in BL(E, \mathbb{R}^n)$  的过渡定义为:

$$T_\xi(t, K_0) = \{x(t) : x = \xi(x), x(0) \in K_0\} \quad (4-6)$$

其中  $K_0 \in \mathbb{R}^n$  可看作从初始集合  $K_0$  出发, 在  $\xi(x)$  下转化的一个管道。集合的突变定义如下:

$$\dot{K}(t) = \{\xi(x) | \xi \in BL(E, \mathbb{R}^n)\} \quad (4-7)$$

此时, 集合的动态可以用突变方程描述:

$$\xi(x) \in \dot{K}(t) \quad (4-8)$$

根据公式4-8，可以将控制输出  $u(t)$  加入到公式中，如下：

$$\xi(x(t), u(t)) \in \dot{K}(t) \text{ with } u(t) = \gamma(K(t)) \quad (4-9)$$

当  $\gamma$  在  $u(t)$  中呈线性时，动态系统可以描述为公式4-10：

$$\xi(x(t), u(t)) = L(x(t))u(t) \quad (4-10)$$

对于具有 3 个自由度来限制平面平移和旋转运动的机器人而言，交互矩阵  $L(\cdot)$  的具体形式表现为：

$$L(x(t)) = \begin{pmatrix} -1 & 0 & y \\ 0 & -1 & -x \\ 0 & 0 & 0 \end{pmatrix} \quad (4-11)$$

基于李雅普诺夫稳定性理论，下面控制器能够实现指数稳定性：

$$u(t) = -\alpha \frac{D^+(K)}{D(K)D^T(K)} V(K) \quad (4-12)$$

其中， $\alpha > 0$  作为一个控制增益， $D^+(K)$  是  $D(K)$  的伪逆函数， $D(K)$  表达式如下：

$$D(K) = \int_K (x - P_{\hat{K}}(x))^L(x) - \int_{\hat{K}} (x - P_K(x))^T L(P_K(x)) \quad (4-13)$$

其中  $P_{\hat{K}}(x)$  表示将点  $x \in K$  投影到集合  $\hat{K}$  上的映射操作。

上述控制器得到的运动控制输出为三轴线速度和三轴角速度，无法直接用于控制机器人，仍需要根据机器人的运动模型转换成相应的驱动轮转速。当前主流的运动控制模型包含全向移动、差速驱动、阿克曼转向和滑移转向等多种形式。本文研究两轮差速驱动机器人为主，其动力系统由两个结构完全对称的驱动轮通过差速配合实现转向、直行等运动控制。

对于具有非完整约束的差速驱动模型，由于其没有  $y$  方向上的运动自由度，控制器计算得到的控制输出  $u(t) = \{v_x, v_y, \omega\}$  中的  $y$  方向上的控制应该映射到角速度上，那么差速驱动机器人的控制输出  $u_{diff}(t) = \{v_x^{diff}, \omega^{diff}\}$  为：

$$\begin{cases} v_x^{diff} = v_x \\ \omega^{diff} = \omega + \beta * v_y \end{cases} \quad (4-14)$$

其中  $\beta$  是任意的映射增益，用于将横向速度反馈到角速度中，根据机器人实际情况进行

调节。

图4-4为差速驱动运动模型的示意图。其中  $ICC$  (Instantaneous Center of Curvature) 是机器人的旋转瞬时曲率中心,  $R$  是机器人的旋转半径,  $L$  为驱动轮之间的轮距,  $v_l$  和  $v_r$  分别表示机器人的左驱动轮和右驱动轮的线速度。此时线速度为  $V = v_x^{diff}$ , 角速度为  $\omega = \omega^{diff}$ , 根据线速度、角速度和旋转半径之间的关系有:

$$R = \frac{V}{\omega} \quad (4-15)$$

$$v_l = \omega R - \frac{\omega L}{2} \quad (4-16)$$

$$v_r = \omega R + \frac{\omega L}{2} \quad (4-17)$$

整理上述式子得:

$$v_l = V - \frac{\omega L}{2} \quad (4-18)$$

$$v_r = V + \frac{\omega L}{2} \quad (4-19)$$

设驱动轮半径为  $R_{wheel}$ , 左右驱动轮的角速度分别为  $\omega_l$  和  $\omega_r$ , 根据  $\omega = V/R$  可计算两轮的角速度:

$$\omega_l = \frac{2V - \omega L}{2R_{wheel}} \quad (4-20)$$

$$\omega_r = \frac{2V + \omega L}{2R_{wheel}} \quad (4-21)$$

得到两驱动轮的线速度和角速度后, 调用 PWM 算法控制电机转速即可完成对机器人的运动控制。

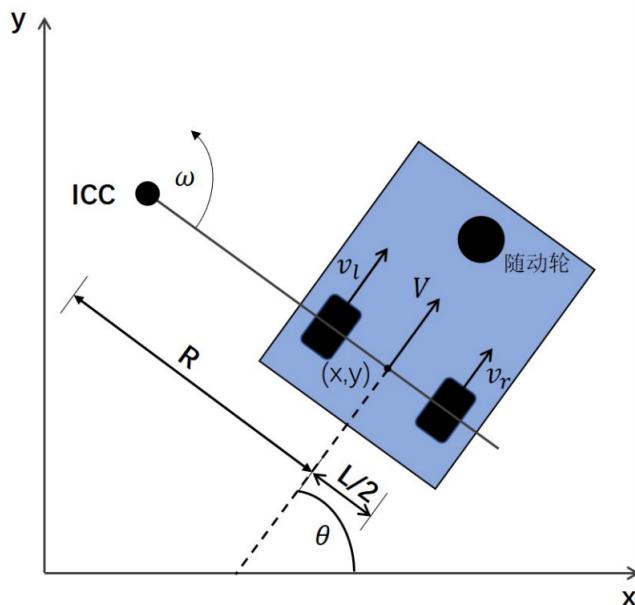


图 4-4 差速驱动模型

## 4.5 本章小结

本章介绍了基于豪斯多夫距离的点云匹配示教重复算法，首先阐述了提出该算法的背景；然后说明该算法的整体框架；接着详细叙述了用于定义点云帧间差异的豪斯多夫距离的计算方法以及点云匹配定位的原理；最后描述了根据李雅普诺夫稳定性理论构造的点云管道跟踪控制器以及最终生成运动控制信息的过程。

## 第五章 实验设计与结果分析

本章主要是对第三、四章提出的方法进行实验验证与分析。在自适应狭窄环境的室内导航系统的相关实验中，采用差速驱动模型作为机器人的运动模型，分别搭建了仿真环境和真实环境，在仿真与真实机器人平台中部署该导航系统，并在其中进行不同狭窄环境中的导航方法的对比和替换不同路径跟踪算法执行狭窄导航策略的效果比对；在点云匹配示教重复算法的相关实验中，验证了豪斯多夫距离计算优化方法的有效性，在机器人平台中部署该算法并分别在空旷场地与狭窄环境验证机器人跟踪示教轨迹的效果。实验表明，本文提出的方法具备可行性，且有效提高了机器人通过狭窄环境的成功率。

### 5.1 自适应狭窄环境的室内导航系统的相关实验

对于自适应狭窄环境的室内导航系统（下文简称为自适应方法）的验证工作，本文基于 ROS 机器人操作系统实现了该导航系统，通过仿真模拟与真实场景相结合的方式进行验证，实验过程中采用三维可视化监测工具 Rviz 对合适通行位姿对 STPs 和运动轨迹等核心数据进行可视化、采集与分析。

#### 5.1.1 仿真环境实验

##### 5.1.1.1 前期部署

仿真环境实验的软硬件配置如表5-1所示。

表 5-1 仿真环境实验的软硬件配置

项目名	设置
CPU	锐龙 R9-6900HS
GPU	NVIDIA GeForce RTX3050ti
运行内存	16GB
操作系统	Ubuntu20.04
ROS 版本	ROS-Noetic

仿真差速驱动机器人的建模如图5-1所示，底盘由半径为 10 厘米的圆柱状机身、两个宽度为 1.5 厘米的驱动轮和两个球状万向随动轮搭建而成，底盘上搭载一个扫描范围为 270 度的单线激光雷达，机身长度为 20 厘米，宽度为 23 厘米。

本文在三维动态模拟器 Gazebo 中搭建的仿真环境如图5-2所示，该仿真环境中存在

若干宽度为 35 厘米至 40 厘米的狭窄环境（经测试，机器人使用正常导航策略穿越这些区域时较为困难）。仿真机器人启用 SLAM 算法扫描该环境创建的全局地图如图 5-3 所示，在该地图上标注了狭窄环境的宽度以及导航点 A、B、C。本次实验的导航任务分为两个阶段：相对简单的一阶段从点 A 导航至点 B；相对困难的二阶段从点 B 导航至点 C。

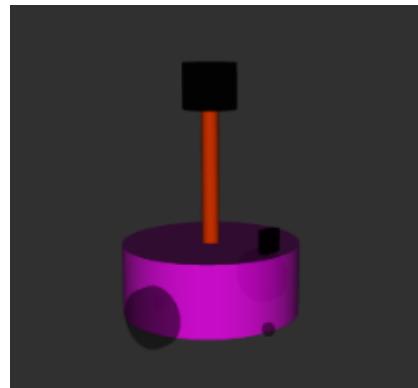


图 5-1 仿真机器人模型

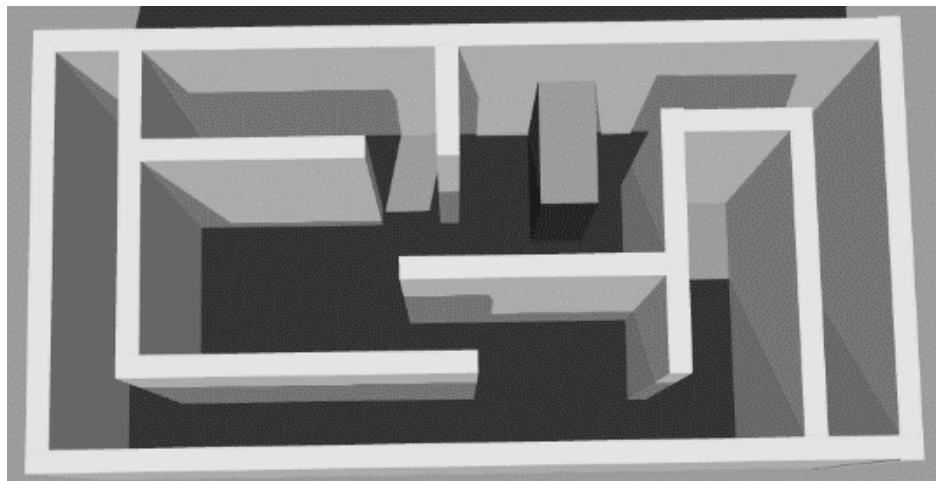


图 5-2 仿真环境

为了后续不同方法对比实验的公平性比较，首先令机器人使用自适应方法从点 A 导航至点 C，以收集该环境下的相关导航策略信息。该轮导航完成后，将  $STPs$  的位置发布至 ROS 通信网络中，并在 Rviz 上订阅查看，如图 5-4 所示。其中，每两个相同颜色的箭头分别表示合适通行起点位姿  $STP_B$  和合适通行终点位姿  $STP_E$ 。由此可看出，自适应方法能够正确地在狭窄环境的出入口生成  $STPs$ 。

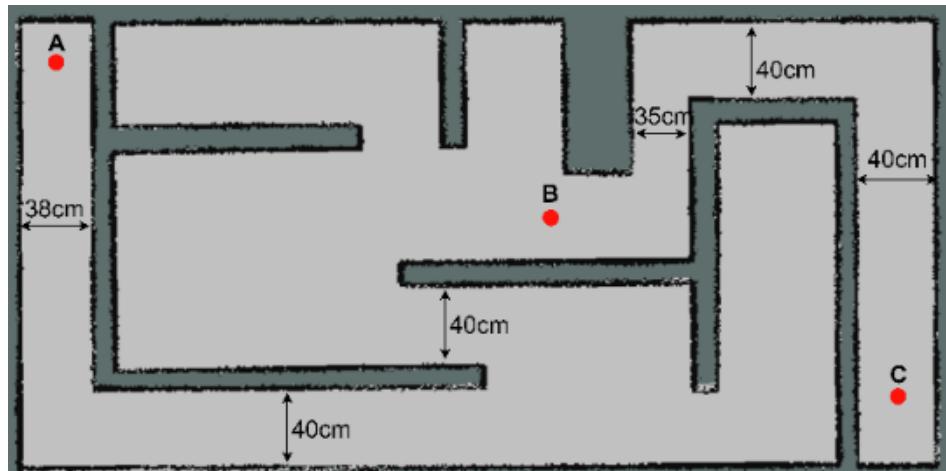


图 5-3 在仿真环境中使用 SLAM 算法创建的地图

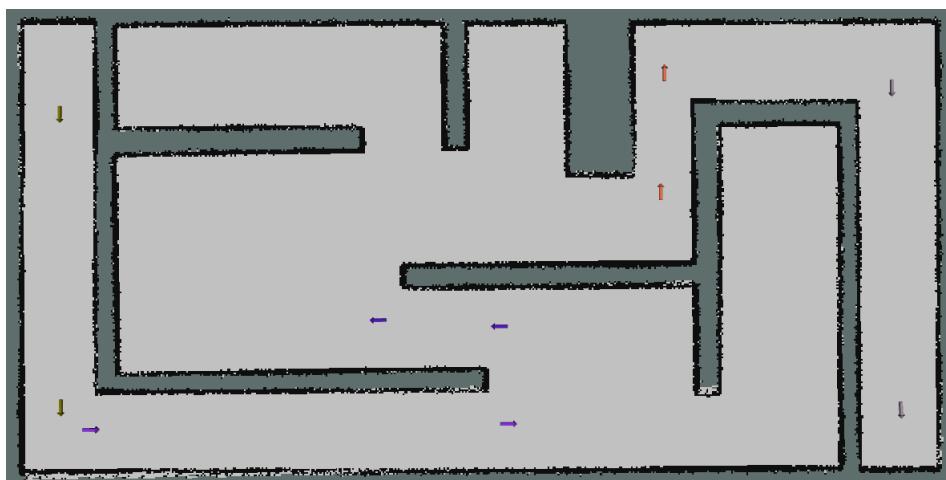


图 5-4 STPs 在 Rviz 中的可视化视图

### 5.1.1.2 狹窄环境中的导航方法对比

本文根据发布时间、室内导航系统相关度等因素选择了通过降低避障相关的权重参数以弱化机器人的碰撞检测条件的调参法<sup>[6]</sup>与采用速度插值补偿技术改良 TEB 算法的速度输出特性并结合人工预设狭窄通道两侧辅助路径点的 PAP-TEB 方法<sup>[10]</sup>进行复现。令两者与本文提出的自适应方法分别执行相同的导航任务，即从点 A 依次导航至点 B 和点 C，每个方法执行完一次导航任务算作一次对比实验。该对比实验共计执行 15 次，相关统计结果如表 5-2 和表 5-3 所示。注意，平均耗时、平均轨迹长度等统计结果中只纳入成功通过狭窄环境的实验数据，下文实验同理。图 5-5 记录了其中一次（第 5 次）对比实验的机器人行驶轨迹的可视图。

在各导航方法的性能评估中，实验分为两个阶段进行分析。阶段一测试结果显示，当狭窄环境的宽度处于 38 厘米至 40 厘米区间时，所有被测试的方法均能顺利完成导航

表 5-2 导航任务阶段一各方法效果对比

方法	通过次数（次）	平均耗时（秒）	平均轨迹长度（米）
调参法	15	57.26	7.579
PAP-TEB 方法	15	64.53	7.931
自适应方法	15	62.11	7.734

表 5-3 导航任务阶段二各方法效果对比

方法	通过次数（次）	平均耗时（秒）	平均轨迹长度（米）
调参法	7	38.62	4.589
PAP-TEB 方法	0	—	—
自适应方法	12	40.85	4.658

任务。通过对比平均耗时与轨迹长度指标，调参法以 15.3 秒平均耗时与 4.1 米轨迹长度表现最优，其技术优势源于放宽了碰撞判定阈值，使得规划的轨迹更接近障碍物边缘（图5-5中的橙色轨迹），从而缩短了实际运动距离。相较之下，PAP-TEB 方法因引入人工辅助定位点的约束，自适应方法受限于 *STPs*，两者均需维持更大的安全边界距离，导致轨迹曲率增加，平均耗时分别达到 64.53 秒与 62.11 秒，较调参法分别增加 12.7% 与 8.5%。

在导航任务的阶段二中，所有方法在宽度为 35 厘米的狭窄环境面前均出现了通行失败的问题，但失败方式存在显著差异。从实验数据可看出，PAP-TEB 方法表现最糟糕，在 15 组对比实验中通过狭窄环境的成功率为 0。从图5-5中的蓝色轨迹可得知，

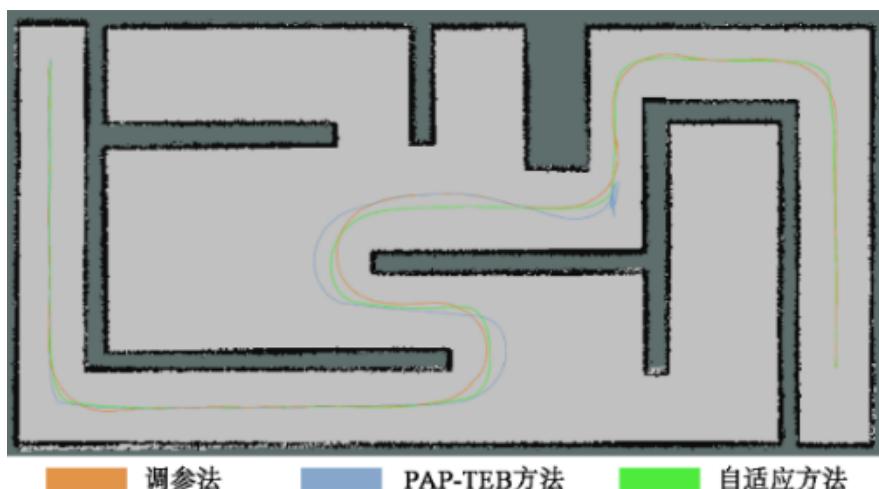


图 5-5 各狭窄环境导航方法的轨迹对比

PAP-TEB 方法因在通道入口生成轨迹失败而倒退重新寻找合适的角度，接近入口后再次生成轨迹失败，由此引发运动振荡的现象。本文认为该方法的失效源于多源误差耦合作用：当安全距离设置为 5 厘米时，若机器人行驶在通道中间，其两侧的安全裕度计算方式为：

$$\text{机器人两侧安全裕度} = \frac{\text{通道宽度} - \text{机器人机身宽度}}{2} \quad (5-1)$$

代入计算得两侧安全裕度为 6 厘米，大于安全距离，理论上局部轨迹规划器能够生成可行轨迹，然而由于激光传感器噪声、栅格地图分辨率等多误差源形成的定位偏差和可行区域范围的减小，导致局部轨迹规划器对障碍物距离的保守估计值超过实际安全阈值，致使所有候选轨迹被误判为碰撞轨迹。调参法虽然通过放宽碰撞检测阈值使机器人能够通过狭窄区域 7 次，但碰撞风险仍然不可忽视，从图 5-5 中可看出，代表调参法的橙色轨迹比较接近通道左侧拐角，其剩余 8 次失败的原因均为在通行过程中与环境产生碰撞，其中 5 次撞到入口左侧拐角。自适应方法通过动态切换狭窄导航策略，在进入狭窄区域前根据 STPs 调整自身位姿，使得机器人的前进方向正对通道入口，在狭窄区域中采用路径跟踪算法替代传统导航系统中的局部轨迹规划算法，严格遵循安全的全局路径行驶。尽管受到多源误差影响导致路径跟踪精度有所浮动，发生了 3 次碰撞，但通行成功率达到 80%，在三种方法中整体表现最佳。

### 5.1.1.3 路径跟踪算法对比

为了说明改进的路径跟踪算法的有效性，下文将分别使用目前主流的路径跟踪算法 Pure pursuit、LQR、MPC 方法替换自适应方法中狭窄导航策略使用的改进的路径跟踪算法，执行导航任务阶段二中的通过宽度为 35 厘米的狭窄环境的任务，以对比不同路径跟踪算法的面对狭窄环境的导航效果。每个方法执行完一次导航任务算作一次对比实验，对比实验共计执行 20 次，统计结果如表 5-4 所示。为更直观地感受各算法跟踪轨迹的精度，这里展示某一次（第 8 次）所有方法都成功通过的对比实验的轨迹误差和轨迹可视图，如图 5-6 和图 5-7 所示。

基于表 5-4 的定量评估与图 5-7 的轨迹误差分析，MPC 方法的成功率最高，改进路径跟踪算法以 75% 的成功率次之。从轨迹误差图中可看出，Pure pursuit 与 LQR 方法在转弯处跟踪路径的稳定性较差，轨迹误差最高达到 4 至 5 厘米。本文分析认为：Pure pursuit 算法在路径曲率突变的区域表现出典型的内切追踪现象，其非完整约束下的几何

表 5-4 路径跟踪算法对比

方法	通过次数(次)	轨迹平均误差(厘米)	计算平均耗时(毫秒)
Pure pursuit	11	1.703	0.19
LQR	9	1.736	0.42
MPC	17	0.684	26.12
改进的路径跟踪算法	15	0.751	9.49

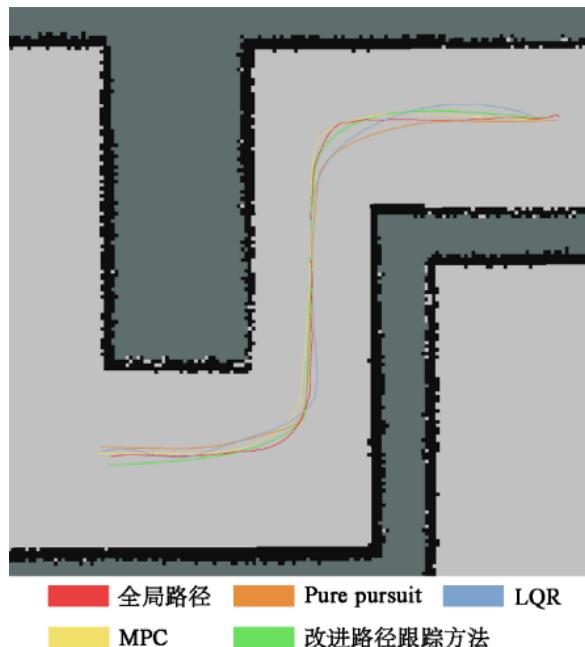


图 5-6 第 8 次实验中各方法轨迹的可视图

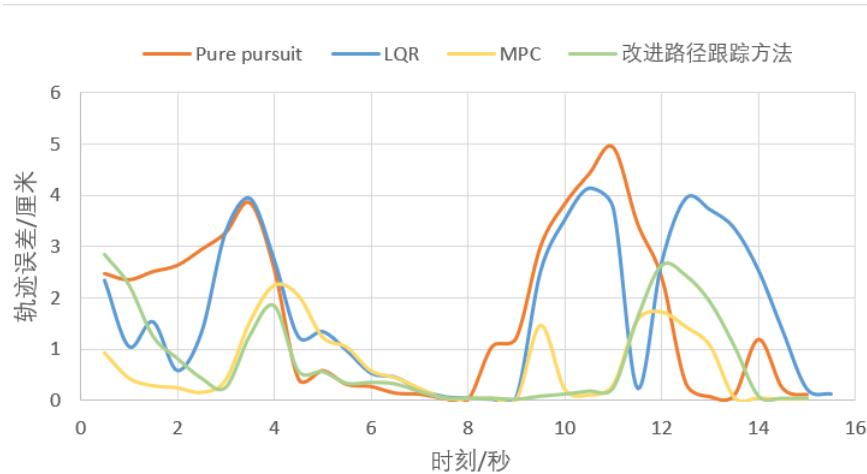


图 5-7 第 8 次实验轨迹误差图

追踪缺陷导致轨迹横向误差变大；LQR 方法的算法特性对路径曲率的连续性要求较高，然而全局路径在规划时并未考虑路径曲率连续的因素，因此跟踪非连续曲率路径将导致 LQR 超调，即系统响应中输出量短暂超过目标值的现象，在路径跟踪中表现为车辆在调整方向时横向误差超过期望路径另一侧，随后振荡收敛。这两种方法的最大轨迹偏移量均偏高，导致实际通过率下降至 55% 以下。

MPC 方法的轨迹误差小于等于 0.684 厘米，虽在跟踪精度上具有优势，但其产生的计算负荷使每次输出控制速度的平均计算耗时达到 26.12 毫秒，较改进的路径跟踪算法的 9.49 毫秒增加了约 175%，然而 MPC 方法多付出的平均计算耗时仅使得跟踪精度相比改进的路径跟踪算法上升 9.7%。在仿真环境试验中，仿真机器人的计算平台为高功耗高性能处理器，MPC 的平均计算耗时尚能跟上机器人的控制频率，在接下来的真实环境试验中，MPC 方法计算负荷过大的问题所带来的影响将会表现在路径跟踪精度上。综合评估，改进的路径跟踪算法对轨迹的跟踪效果仅次于 MPC 方法，计算平均耗时远小于 MPC 方法。

#### 5.1.1.4 其他仿真环境下的对比

为了更全面评估自适应方法在狭窄场景中的性能表现，本文搭建如图5-8所示的四类狭窄空间仿真场景，分别采用调参法、PAP-TEB 方法及自适应方法实施导航测试，通过对比分析不同方法在狭小空间穿越任务中的成功率差异进一步验证所提方法的环境适应能力。

操纵机器人分别在四个不同的仿真环境中使用 SLAM 算法创建地图如图5-9所示，图中红点标注的点 A 和点 B 表示机器人需要在该环境中执行从点 A 到点 B 的导航任务。

分别在机器人中使用调参法、PAP-TEB 方法和自适应方法执行上文描述的导航任务进行对比实验，每个方法在所有环境中完成导航任务算作一次对比实验，对比实验共计执行 10 次，各方法导航任务执行成功的次数统计结果如表5-5所示。

表 5-5 各方法在不同环境中执行导航任务的成功次数

方法	环境 1	环境 2	环境 3	环境 4
调参法	6	8	3	6
PAP-TEB	4	7	4	5
自适应方法	8	10	7	8

在序列式窄门（仿真环境 1）、狭长通道（仿真环境 2）以及具备非结构化地形特征

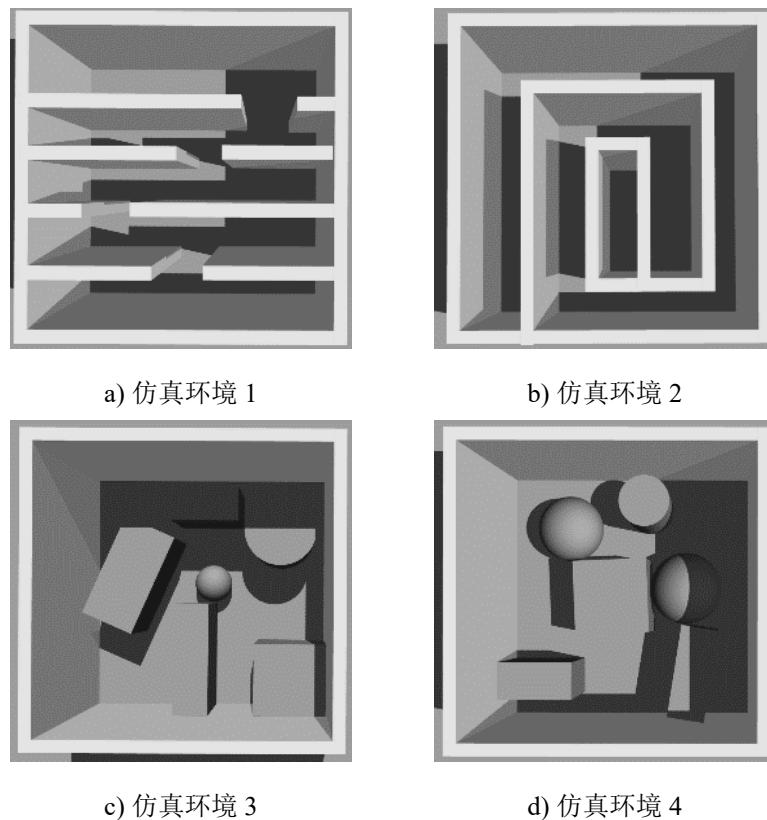


图 5-8 四个不同的仿真环境

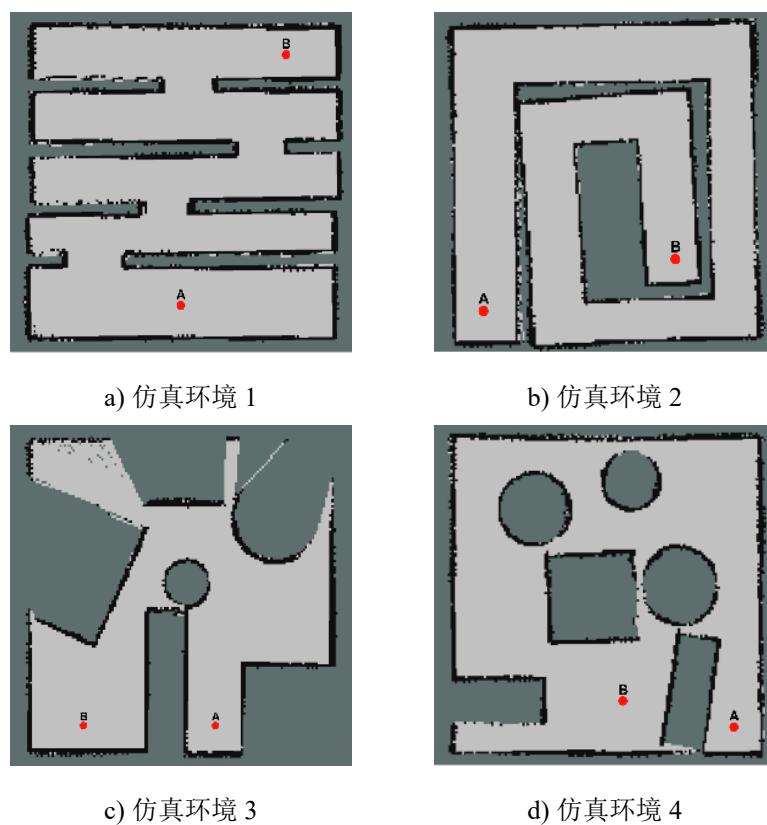


图 5-9 四个不同的仿真环境的地图和导航任务

的受限空间（仿真环境 3、4）测试中，自适应方法展现出对复杂几何约束的感知能力，均能够有效标记狭窄环境并生成 *STPs* 指导机器人通行。实验数据表明，该方法在不同类型狭窄场景中的导航成功率平均值为 82.5%，较调参法与 PAP-TEB 方法分别提升 65% 和 43.5%，验证了其应对不同狭窄环境的健壮性。

## 5.1.2 真实环境实验

由于仿真环境实验相对理想，不受过多传感器测量误差与其他外界因素干扰，且无法完全模拟真实机器人的计算性能，故需要真实环境实验进一步验证本文提出方法的有效性与健壮性。

### 5.1.2.1 前期部署

实验使用的移动机器人平台如图 5-10 所示，其车身长宽均为 20 厘米，控制模块主要由嵌入式微处理器平台、底层控制平台以及电机驱动板构成，结构关系如图 5-11 所示。负责运行 ROS 系统并执行导航任务的微处理器平台采用瑞芯微 RockChip RK3399，相关软硬件参数如表 5-6 所示。机器人前方部署的单线激光雷达为深圳市镭神智能系统有限公司生产的 LS01D 型单线激光雷达，具体参数如表 5-7 所示。



a) 机器人正面

b) 机器人侧面

图 5-10 用于真实环境实验的机器人

本文在实验室中搭建的真实环境实验场景如图 5-12 所示，其中包含若干宽度为 32 厘米至 37 厘米的狭窄通道。机器人利用 SLAM 对环境创建的地图如图 5-13 所示，上面标注了关键场景的宽度和导航点 A、B，本次实验的导航任务为从点 A 导航至点 B。

与仿真环境实验相似，首先令机器人使用自适应方法从点 A 导航至点 B，以收集该环境下的相关导航策略信息。该轮导航完成后，将 *STPs* 的位置发布至 ROS 通信网络中，并在 Rviz 上订阅查看，如图 5-14 所示。其中，每两个相同颜色的箭头分别表示合适通行起点位姿  $STP_B$  和合适通行终点位姿  $STP_E$ 。由此可看出，自适应方法在真实环境

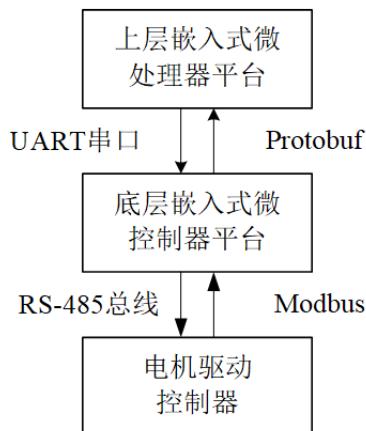


图 5-11 机器人控制模块的结构

表 5-6 RockChip RK3399 具体参数

项目名	设置
CPU	双 Cortex-A72+ 四 Cortex-A53
GPU	Mali-T860
运行内存	LPDDR3 4GB
通讯模块	正基科技 AP6356S WiFi
操作系统	Ubuntu18.04
ROS 版本	ROS-Melodic

中也能够正确地在狭窄环境的出入口生成 *STPs*。

表 5-7 单线激光雷达的具体参数

项目名	设置
扫描频率	3.11Hz
测点速率	3600 点/秒
测量范围	8 米
扫描角度	360°
角度分辨率	1°
光照强度	20000lux

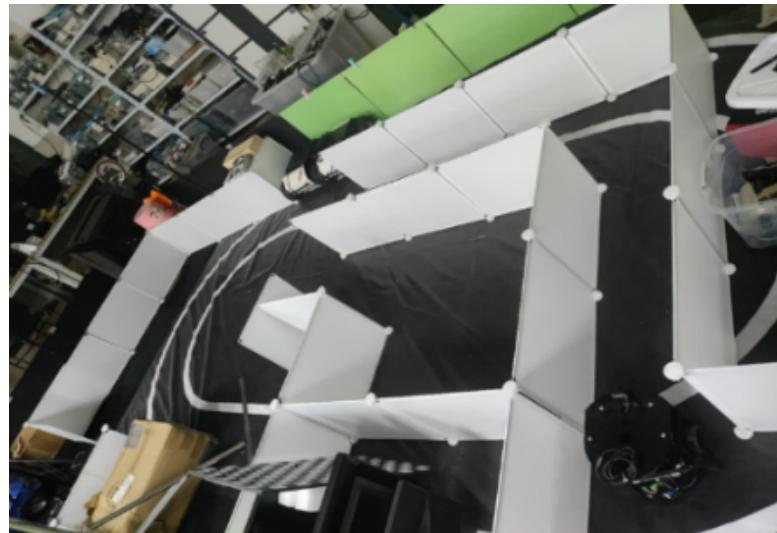


图 5-12 真实环境中的实验场景

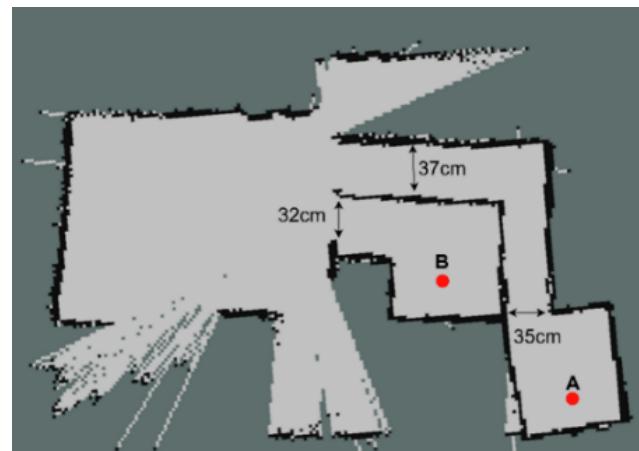


图 5-13 在真实环境中创建的地图



图 5-14 STPs 在 Rviz 中的可视化视图

### 5.1.2.2 狹窄环境中的导航方法对比

本文分别在机器人上使用调参法、PAP-TEB 与自适应方法执行导航任务进行对比实验，对比实验共计执行 15 次，统计结果如表 5-8 所示，图 5-15 为第 2 次实验的轨迹可视图。

表 5-8 真实环境实验中各方法效果对比

方法	通过次数（次）	平均耗时（秒）	平均轨迹长度（米）
调参法	7	47.76	5.066
PAP-TEB 方法	5	63.90	5.609
自适应方法	11	55.83	5.485

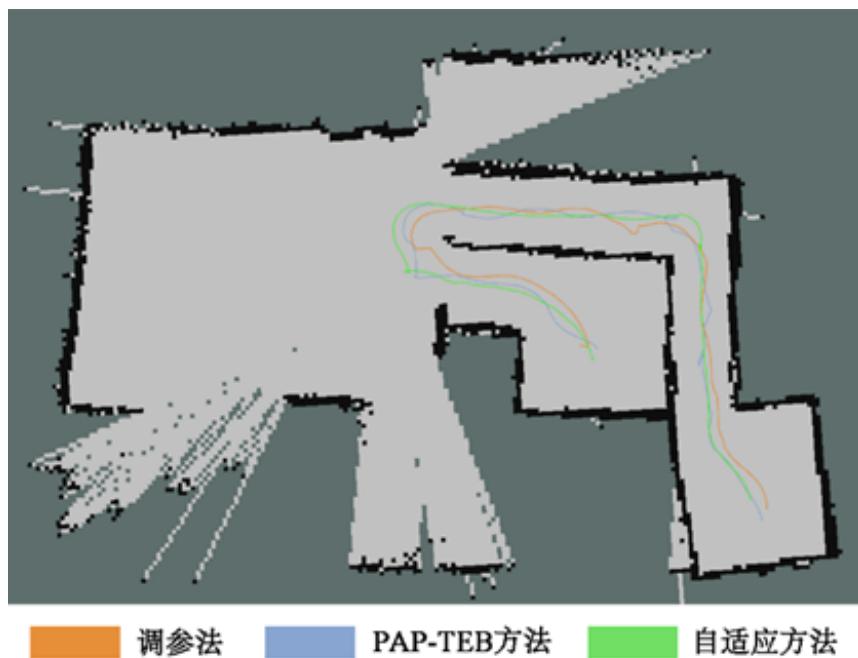


图 5-15 STPs 在 Rviz 中的可视化视图

实验结果表明，自适应方法在狭窄通道场景中展现出最优性能，其执行导航任务的成功率达到 73.3%，该结果与仿真环境实验的结论一致。通过对比图 5-15 中各方法的轨迹可得，调参法生成的轨迹紧贴障碍物边缘导致安全裕度不足，碰撞概率显著提升；PAP-TEB 方法则因轨迹碰撞检测环节的误判问题造成机器人行驶的连续性被打断甚至导航任务失败，最终导致其平均通行时间增加且成功率下降。而自适应方法使机器人始终沿通道中线稳定通行，主动保持与两侧障碍物的安全间距，有效降低了系统碰撞风险，拥有更安全可靠的狭窄环境通过能力。

### 5.1.2.3 路径跟踪算法对比

表 5-9 真实环境实验中路径跟踪算法对比

方法	通过次数（次）	轨迹平均误差（厘米）	计算平均耗时（毫秒）
MPC	11	3.63	61.48
改进的路径跟踪算法	16	2.47	36.34

由于 Pure pursuit 与 LQR 方法在仿真环境中表现不佳，因此下文仅使用 MPC 方法替换自适应方法中狭窄导航策略用到的改进路径跟踪算法，分别执行从点 A 导航至点 B 的导航任务以对比两者的效果差异。对比实验共执行 20 次，统计结果如表5-9所示，输出每个控制结果的计算耗时对比如图5-16，图5-17为第 10 次实验的轨迹可视图。

根据表5-9的实验数据对比可知，改进路径跟踪算法以 80% 的通过率展现出最优性能，而传统 MPC 算法的通过率仅为 55%，较仿真实验结果出现明显下降。本文分析 MPC 算法在真实场景中性能下降的主要原因为：嵌入式硬件平台的计算资源受限，难以支撑 MPC 算法的高实时性计算需求。在 RK3399 处理器上运行时，MPC 算法单次求解时间存在显著波动，其峰值计算延迟最高达到 184.53 毫秒如图5-16所示，远超机器人的控制周期（一般为 50 至 100 毫秒）。这种突发的控制超时直接导致控制器对当前位姿的控制响应滞后，使机器人在路径跟踪过程中出现周期性位姿修正延迟，表现为行进轨迹的横向振荡现象，进而引发跟踪误差累积与路径偏移加剧。实验结果表明，本文提出的改进路径跟踪算法有效缓解了 MPC 算法的计算瓶颈问题，也验证了计算优化的必要性。

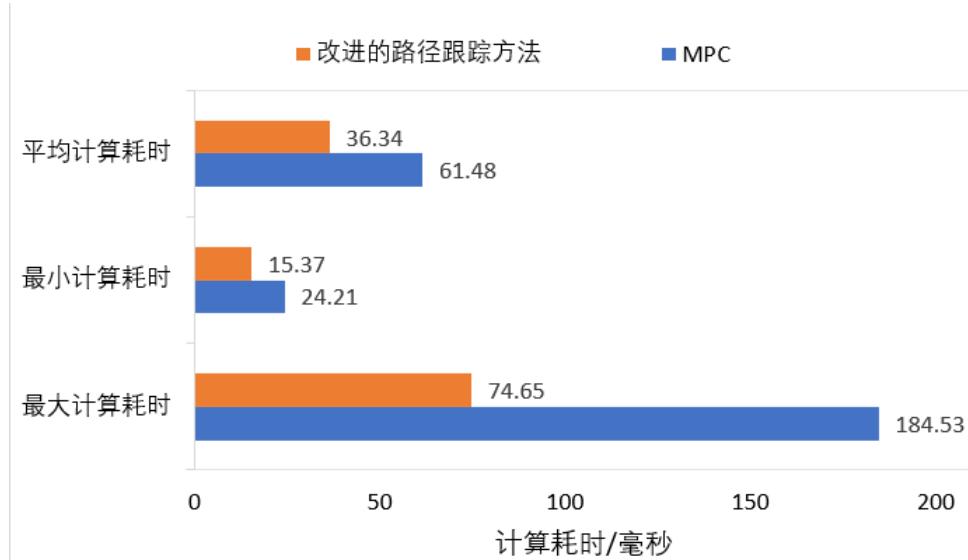


图 5-16 计算耗时对比

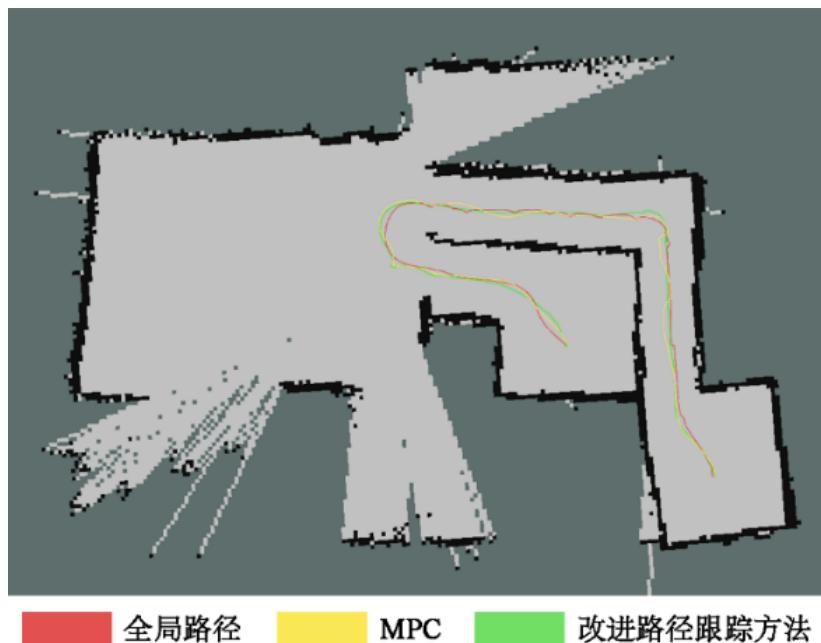


图 5-17 真实环境实验中路径跟踪算法的轨迹对比

## 5.2 点云匹配示教重复算法的相关实验

对于基于豪斯多夫的点云匹配示教重复算法的验证工作，本文在真实机器人上实现了该算法，并在自适应狭窄环境的室内导航系统的基础上将其作为备选狭窄导航策略，用户可根据自身需求选择使用第三章或第四章提出的狭窄导航策略。至此本文提出的面向狭窄环境的室内导航系统已完全实现。本文在真实场景中验证该算法，实验过程中通过采集轨迹的方式验证示教重复的效果。

### 5.2.1 前期部署

实验使用的移动机器人平台如图5-18所示，其在上文真实环境试验使用的移动机器人平台的前上方部署了大疆的 Livox Mid-360 三维激光雷达，其扫描范围如图5-19所示，具体参数如表5-10所示。

### 5.2.2 豪斯多夫距离计算优化验证

首先验证本文提出的豪斯多夫距离计算优化方法，即通过过滤点云数据以减少计算量和并行化以提升计算效率的方式降低计算耗时。其中过滤点云数据方法中影响计算耗时最大的是下采样体素参数，并行化方法中影响计算耗时最大的是线程数量，因此针对下采样体素参数和线程数量对计算耗时的影响进行实验，验证优化效果并寻找适合本文实验所用计算平台的参数。虽然下采样能够有效降低点云密度，但仍需注意过大的体素



a) 机器人正面      b) 机器人侧面

图 5-18 安装三维激光雷达的机器人

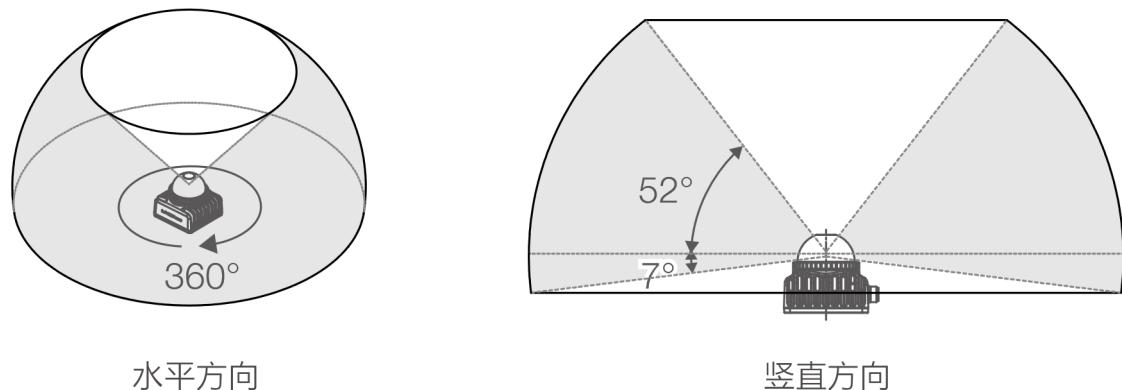


图 5-19 Livox Mid-360 三维激光雷达的有效 FOV 范围

参数即过小的下采样率会导致特征失真，可能增大点云匹配的误差，如图5-20b所示，当体素参数大至0.2时，相比于原始的激光点云数据丢失了过多的细节与特征。综上，设计实验探索体素参数、线程数与点云匹配计算耗时之间的关系，计算耗时如表5-11所示。

表 5-10 Livox Mid-360 三维激光雷达的具体参数

项目名	设置
激光波长	905nm
点云帧率	10Hz
测点速率	200000 点/秒
量程 (@100klx)	40m @ 10% 反射率
视场角	水平 360°, 坚直 -7° 至 52°

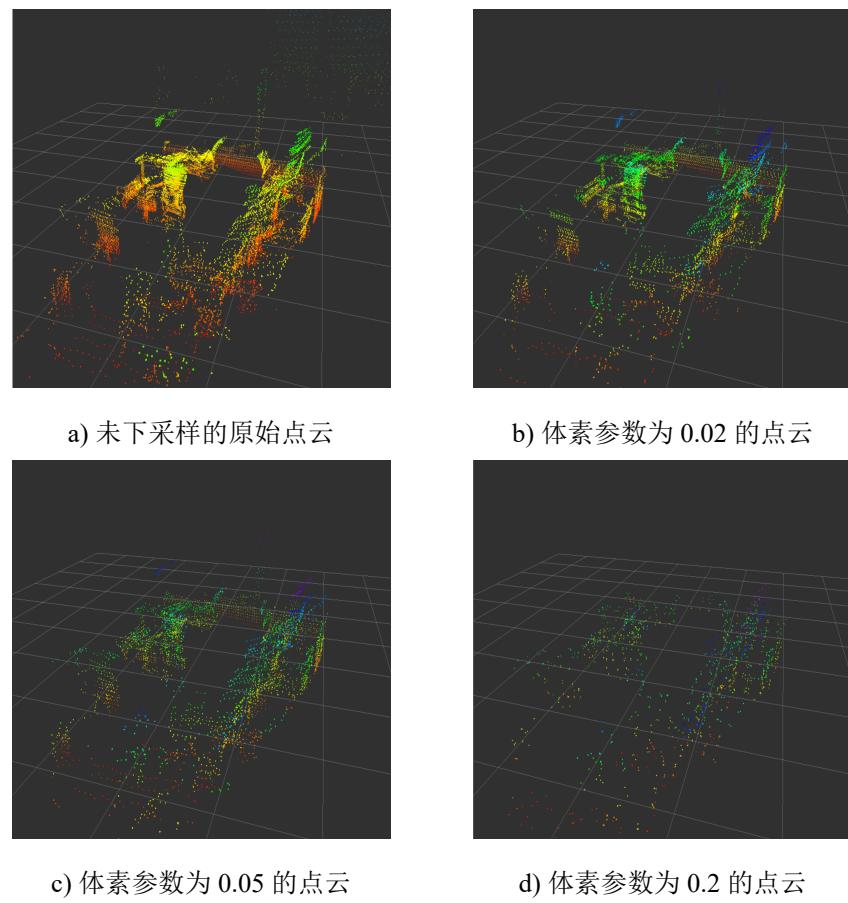


图 5-20 不同下采样体素参数下的激光点云图

表 5-11 不同体素参数与线程数下的点云匹配计算耗时/毫秒

线程数/个 \ 体素参数	0.2	0.1	0.05	0.02	不下采样
1	41.8	183.4	476.9	1131.4	1632.7
2	22.4	104.3	256.8	634.8	952.6
4	14.5	62.2	170.4	425.3	612.2
6	12.1	53.7	133.6	323.5	474.7
8	13.6	58.2	157.5	373.7	526.8
10	14.8	67.7	169.2	410.8	602.1

从点云过滤层面分析，当体素参数设置为 0.2 时单线程点云匹配的计算耗时降低至 41.8 毫秒，相比于不下采样的计算速度提升约 39 倍，证实了体素滤波对点云密度的指数组下采样的作用，即参数增大可使三维空间划分的体素数量以立方速率减少，从而大幅降低匹配算法的计算复杂度，使得算法仅需处理稀疏化的关键几何特征。然而体素参数并非越小越好，当体素参数小于场景特征尺寸时，继续缩小边长对几何细节的保留效果趋于饱和，但计算代价仍以超线性速度增长，如体素参数从 0.05 减至 0.02 时，耗时增幅达 137%。

从多线程计算层面分析，当线程数从 1 增至 6 时，各体素参数条件下的计算耗时均显著下降：在不采样的情况下，6 线程耗时较单线程降低了 70.9%，加速比达 3.5 倍。然而当线程数增至 8 到 10 时，计算耗时没有显著下降甚至出现反常回升：在不采样的情况下，8 线程耗时较 6 线程增加了约 11%，10 线程较 6 线程增加 26.8%。本文分析认为这与多线程的硬件依赖性有关，即 RK3399 仅 6 个物理核心且不支持超线程功能，因此在 6 线程并行化时 2 个 A72 核与 4 个 A53 核同时工作使计算效率最大化，而当线程数超过物理核心时需要进行线程分时处理，导致线程上下文切换频率上升，从而无法进一步提升并行化加速效果。

综上分析，下采样与并行化均能够有效优化点云匹配的计算耗时，考虑到移动机器人运动控制频率一般为 10 到 50 赫兹，因此点云匹配计算耗时不能高于 100 毫秒，且对于 RK3399 计算平台而言，6 线程并行化能够最大化计算加速效果，在满足计算耗时要求的同时体素参数越小越能够保留更多环境特征，因此后续实验中将采用体素参数 0.1 的体素滤波下采样与 6 线程并行化处理进行点云匹配的计算。

### 5.2.3 空旷场地的示教重复实验

为确保移动机器人的安全，进行狭窄环境实验前先在空旷场地验证点云匹配示教重复算法的基本功能以及在不同速度、初始角度偏差等不同因素影响下的轨迹跟踪精度和稳定性。

实验场地为实验室外的茶水区域如图5-21所示，为实验场地绘制二维俯视图并标记各个示教动作点如图5-22，机器人的任务为从A1出发，依次经过A2-3、B1-3、C1-3。

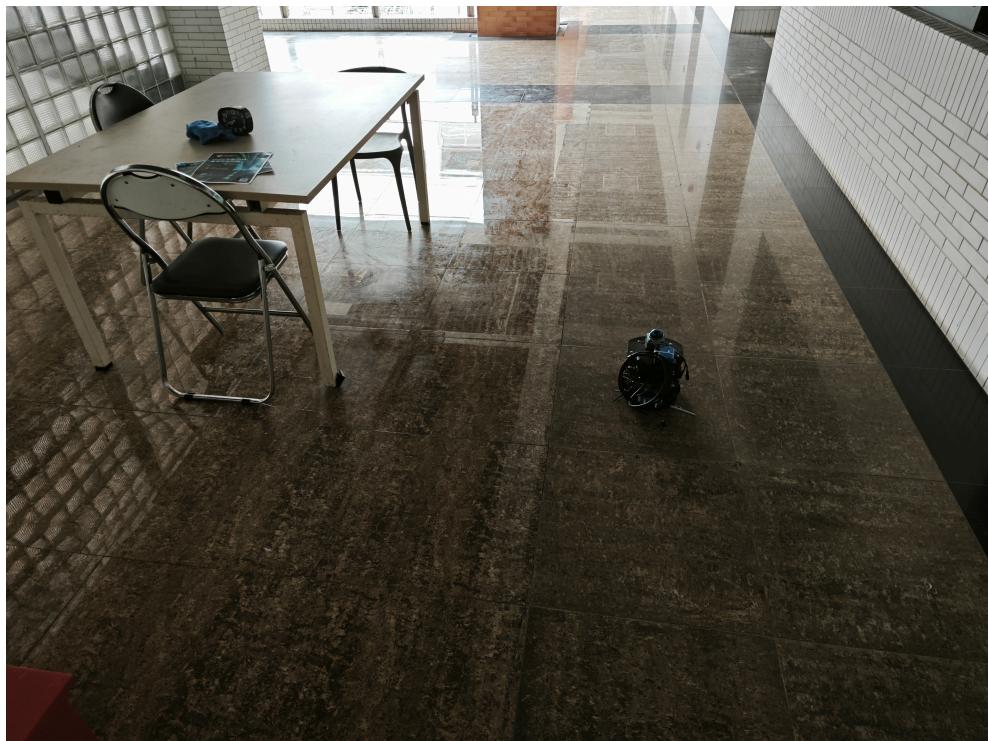


图 5-21 空旷场地示教重复实验的场地

首先手动操作机器人依次经过各个示教动作点，随后分别设置 0.1、0.2、0.3 米/秒的最大线速度执行算法，该对比实验重复 3 次，通过对比示教轨迹与重复轨迹在示教动作点上的位置及角度偏差评估算法的性能和稳定性，实验结果如表5-12所示。

由表可知，当最大线速度为 0.1 米/秒时，位置平均偏差稳定在 1.50 至 3.52 厘米之间，角度偏差为 2.46 至 6.45°。在 A1-A3、B2-B3、C2-C3 这类直线轨迹上的偏差较低，而转弯点 B1 和 C1 的偏差显著上升，本文认为这可能与转向时机器人动力学模型的瞬态误差积累相关，此时位置标准差和角度标准差均较小，分别小于等于 1.32 厘米和 1.86°，表明低速下算法稳定性良好；当最大线速度为 0.2 米/秒时，位置偏差平均增长 17.8%，角度偏差平均增长 31.4%。然而，速度翻倍并未导致误差成倍增加，如 A1 点位置偏差仅从 1.50 厘米增加到 1.85 厘米，而移动效率提升了 100%，转弯点偏差的显著上

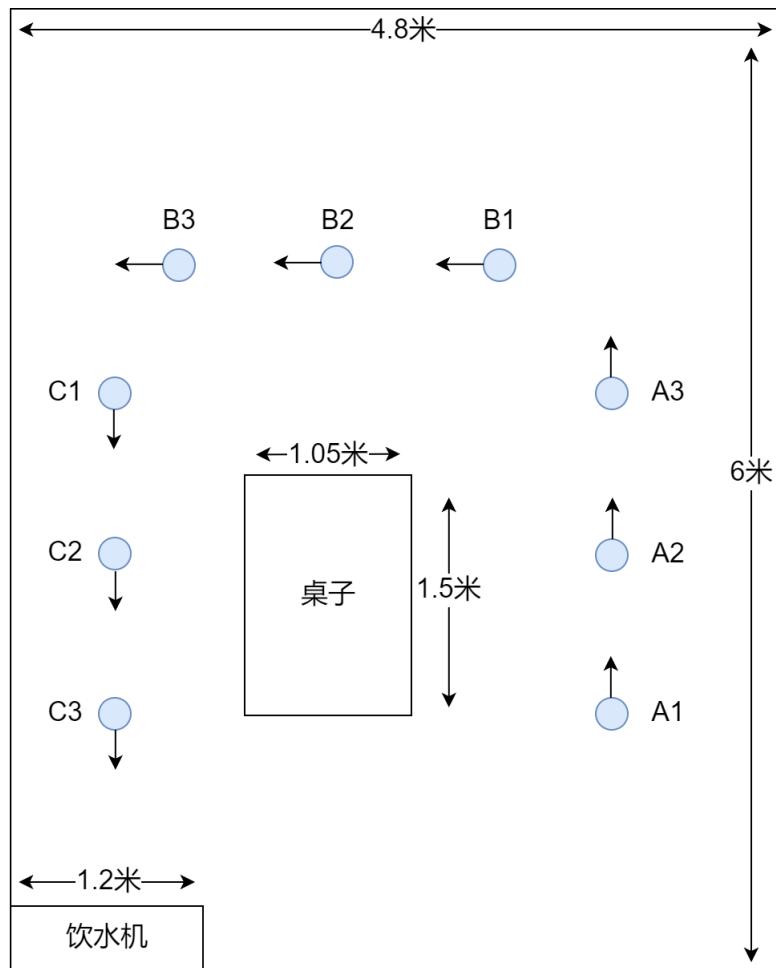


图 5-22 空旷场地的二维俯视图

升情况同 0.1 米/秒，有所上升但仍处于可控范围，标准差增幅低于 15%，说明算法仍能保持稳定；当速度提高到 0.3 米/秒时，相比于 0.1 米/秒时的位置偏差平均增长 53.0%，角度偏差平均增长 87.7%，且位置标准差增大至 2.12 厘米，角度标准差增加到 3.15°，分析认为在该速度下转弯时容易因惯性产生滞后误差，导致轨迹跟踪效果不佳。综上，点云匹配示教重复算法在 0.1-0.2 米/秒速度范围内的轨迹跟踪精度整体较高，尽管最大线速度的提升会导致偏差增大，但 0.2 米/秒下的偏差增幅显著低于速度提升幅度，因此后续实验中均选择 0.2 米/秒作为平衡效率与精度的最优参数。

为探索初始位姿的改变是否对轨迹跟踪有较大影响，令机器人以偏离初始角度 5、10、15° 的姿态，以 0.2 米/秒的最大线速度重复上述实验 3 次，实验结果如表 5-13 所示。

由表可知，初始角度偏差对点云匹配算法的轨迹跟踪精度的影响程度随偏差角度的增加而加剧。与无初始角度偏差且最大线速度为 0.2 米/秒下的数据相比，当初始角度偏差为 5° 时，A2、B2、C2 的位置偏差分别增至 3.05 厘米、3.68 厘米和 4.02 厘米，增幅

表 5-12 示教重复轨迹跟踪偏差

最大速度 米/秒	评价指标	示教动作点								
		A1	A2	A3	B1	B2	B3	C1	C2	C3
0.1	位置平均偏差/厘米	1.50	2.10	2.35	3.52	2.63	2.41	3.48	2.95	2.80
	位置标准差/厘米	0.45	0.72	0.68	1.25	0.95	0.81	1.32	0.88	0.75
	角度平均偏差/°	2.46	3.15	3.36	6.45	3.75	3.54	6.24	4.26	4.05
	角度标准差/°	0.66	1.05	0.84	1.86	1.23	1.14	1.74	1.32	1.08
0.2	位置平均偏差/厘米	1.85	2.45	2.68	4.13	3.02	2.95	4.25	3.45	3.20
	位置标准差/厘米	0.62	0.85	0.78	1.42	1.12	0.95	1.58	1.05	0.88
	角度平均偏差/°	3.15	3.96	4.35	8.55	5.04	4.86	8.25	5.64	5.16
	角度标准差/°	1.05	1.26	1.14	2.25	1.56	1.35	2.46	1.65	1.44
0.3	位置平均偏差/厘米	2.35	3.15	3.42	5.25	4.12	3.95	5.35	4.50	4.25
	位置标准差/厘米	0.85	1.12	1.05	1.85	1.42	1.25	2.12	1.58	1.35
	角度平均偏差/°	5.46	6.15	6.75	10.95	7.35	7.14	10.65	7.95	7.56
	角度标准差/°	1.74	1.95	1.86	2.85	2.25	2.04	3.15	2.55	2.34

表 5-13 不同初始角度下的示教重复轨迹跟踪偏差

初始角度偏差/°	示教动作点	位置平均偏差/厘米	角度平均偏差/°
5	A2	3.05	5.34
	B2	3.68	6.45
	C2	4.02	6.90
10	A2	4.25	7.86
	B2	5.15	10.05
	C2	5.75	11.64
15	A2	5.85	11.25
	B2	7.12	18.39
	C2	8.25	28.38

为 16-25%，角度偏差增幅为 22-35%；当初始偏差为  $15^\circ$  时，B2 和 C2 的位置偏差分别增长至 7.12 厘米和 8.25 厘米，角度偏差为  $18.39^\circ$  和  $28.38^\circ$ ，误差增速超线性增长趋势，最大角度偏差达到无初始角度偏差的 3.44 倍，基本无法跟踪示教轨迹。虽然单独使用该算法存在初始角度偏差的限制，但面向狭窄环境的室内导航系统在执行该算法前会根据合适通行位姿对调整机器人进入狭窄环境时的姿态，因此能够有效避免该算法对初始角度偏差敏感的问题。

#### 5.2.4 狹窄环境的示教重复实验

该实验沿用自适应狭窄环境的室内导航系统的相关实验中的准备的真实场景如图5-12所示。首先手动操控机器人依次录制从示教动作点 A1 到 A3、B1 到 B3、C1 到 C2 的激光点云管道共三个，其中每个轨迹的起点与终点都是面向狭窄环境的室内导航系统生成的一对合适通行位姿对 (STPs)，如图5-23所示。机器人的导航任务为从导航点 A 移动到导航点 B，过程中会执行狭窄环境进入判断方法、调整至合适通行起点位姿 ( $STP_B$ ) 并切换狭窄导航策略即点云匹配示教重复算法。实验执行 5 次上述机器人导航任务，其中导航任务执行成功 4 次，点云匹配示教重复算法的轨迹跟踪结果如表5-14所示。

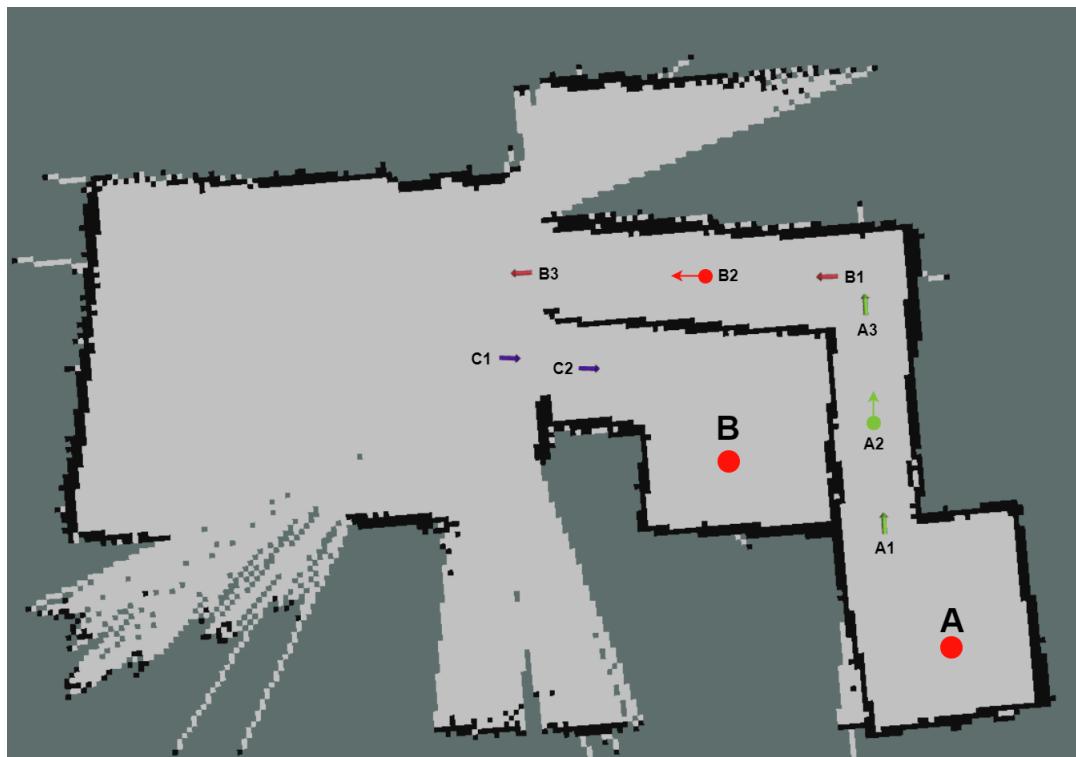


图 5-23 狹窄环境的示教动作点展示

表 5-14 示教重复轨迹跟踪偏差

评价指标	示教动作点							
	A1	A2	A3	B1	B2	B3	C1	C2
位置平均偏差/厘米	1.90	2.50	2.32	1.53	2.75	3.05	1.74	1.25
位置标准差/厘米	0.65	0.85	0.72	0.55	1.20	1.35	0.60	0.50
角度平均偏差/°	3.30	4.05	4.50	6.39	8.25	5.04	3.75	5.85
角度标准差/°	1.05	1.26	1.14	1.74	2.25	2.04	0.90	1.95

由表得知，点云匹配示教重复算法在狭窄环境中的表现与空旷场景接近，最大位置平均偏差为 3.05 厘米，远小于上文计算的该狭窄环境两侧 6 厘米的安全裕度，且最大角度平均偏差为 8.25°，整个跟踪过程没有出现较大的角度偏差，本文分析认为机器人是通过根据合适通行位姿对提前调整通行位姿的方式到达 A1、B1、C1 点，保证了初始角度不会产生大偏差，因此能够有效避免该算法对初始角度偏差敏感的问题，保证后续轨迹跟踪的稳定。该算法通过狭窄环境的成功率为 80%，与第三章提出的自适应方法的 73.3% 相比提升了约 7%，两者差距不大，均能够有效提升机器人面对狭窄环境时的通行性能。综上，本实验验证了该算法在狭窄环境下跟踪示教轨迹的精确性和稳定性。

### 5.3 本章小结

本章主要介绍了第三、四章提出的自适应狭窄环境导航系统与基于豪斯多夫距离的点云匹配示教重复算法的实验。在自适应狭窄环境导航系统的实验中，本章基于差速驱动模型部署算法并构建了仿真环境与真实环境，通过对不同导航方法在狭窄环境中的表现验证了所提方法的有效性；同时在系统中替换多种路径跟踪算法进行横向对比，证明了狭窄导航策略中的改进路径跟踪算法的必要性。在点云匹配示教算法的实验中，验证了豪斯多夫距离计算优化方法能够显著提升点云匹配的计算效率，并在空旷与狭窄场景下验证了机器人跟踪示教轨迹的精度和稳定性。实验结果表明，本文方法在保证机器人导航安全性的前提下有效解决了传统方法在狭窄场景中通过率低的问题。

## 总结与展望

在室内服务场景中，移动机器人依赖自主导航技术实现任务执行，但高度动态性和复杂性环境对导航系统的精度提出了较高的要求。当前主流商用机器人多采用激光雷达结合低功耗计算平台的导航架构，该方案在常规场景下表现稳定，但在狭窄环境等受限空间内导航系统对误差源的容忍性下降，容易出现轨迹规划失效、运动控制失稳等问题。这种空间适应性缺陷导致机器人难以满足全场景自主作业的可靠性要求，成为了移动机器人实现完全自主可靠地工作的障碍之一。因此，本文针对硬件方案成本受限的移动机器人如何安全稳定地通过狭窄环境进行研究，主要贡献如下：

- 1) 本文提出了自适应狭窄环境的室内导航系统。该系统在导航过程中根据地图中障碍物与全局路径的几何关系标记狭窄环境并生成合适通行位姿对；机器人出入被标记的狭窄环境时自动切换相应导航策略，以自适应环境；在狭窄环境导航策略中，全局成本地图膨胀化以规划更安全的全局路径，机器人根据合适通行位姿分段规划全局路径，目的为提前调整位姿以减少在狭窄环境中的转向需求，并通过最优控制问题转化为最小二乘问题的思想优化 MPC 路径跟踪方法，用以代替局部轨迹规划方法计算轨迹，防止局部轨迹碰撞误判导致导航失败。
- 2) 本文提出了基于豪斯多夫距离的点云匹配示教重复算法，作为自适应狭窄环境的室内导航系统中的补充策略。由于导航系统中如定位、地图等各环节的误差源过多，因此抛弃传统导航框架，直接利用传感器信息控制机器人。示教阶段利用三维激光雷达扫描狭窄环境三维特征形成激光点云管道，重复阶段机器人根据当前环境点云与点云管道进行基于豪斯多夫距离的点云匹配，并根据李雅普诺夫稳定性理论构造反馈控制器，以推理当前点云状态与目标点云状态之间转换的控制速度。
- 3) 本文结合上述方法设计并实现了面向狭窄环境的机器人室内导航系统，在仿真环境和真实机器人上部署，并针对提出的方法完成了一系列可行性与性能测试。仿真及真实环境实验结果表明，该系统能够有效识别环境狭窄与否并切换相应的导航策略，提升了机器人面对狭窄环境时的通过率。

本文针对狭窄环境下的移动机器人导航问题提出了一系列创新性解决方案并在仿真与真实场景中验证了其有效性。然而受限于研究周期，仍有若干问题值得在未来进一步探索与完善：

- 1) 多模态感知与动态环境适应性增强：当前系统主要依赖激光雷达构建环境表征，

未来可探索如视觉、深度相机、超声波雷达等多模态传感器进行深度融合机制，特别在人流量大、障碍物频繁移动的动态密集场景中，如何实时解耦动态/静态障碍物并基于多源信息优化导航策略是提升系统鲁棒性的关键方向。此外，边缘计算与轻量化模型部署的结合可能为低功耗硬件平台下的实时感知-决策闭环提供新思路。

- 2) 基于豪斯多夫距离的点云匹配示教算法在结构化狭窄环境中表现良好，但面对如弯曲管道、多分支路口等复杂非规则狭窄通道时，点云特征的泛化能力仍需提升。未来可研究引入深度学习辅助的点云关键特征提取方法，结合注意力机制增强匹配精度；针对高密度点云数据，可设计更高效的计算加速策略能够提升机器人控制频率从而达到更高的轨迹跟踪精确度。
- 3) 人机交互与自主性平衡机制：狭窄环境中的人机共存需求尚未充分研究。未来可探索基于社会力模型或意图预测的人机共融导航策略，在保证安全性的同时减少对人类活动的干扰，拿在机器人通过狭窄区域时举例，可结合语音提示、灯光交互等设计构建透明化、可解释的协同通行规则，进一步提升服务机器人在复杂社会环境中的接受度等。

## 参考文献

- [1] 中华人民共和国. “十四五”机器人产业发展规划[EB/OL]. 2021. [https://www.gov.cn/zhengce/zhengceku/2021-12/28/content\\_5664988.htm](https://www.gov.cn/zhengce/zhengceku/2021-12/28/content_5664988.htm).
- [2] 中华人民共和国. 政府工作报告[EB/OL]. 2024. [https://www.gov.cn/yaowen/liebiao/202403/content\\_6939153.htm](https://www.gov.cn/yaowen/liebiao/202403/content_6939153.htm).
- [3] Hsieh Y Z, Ku X L, Lin S S. The development of assisted-visually impaired people robot in the indoor environment based on deep learning[J]. Multimedia Tools and Applications, 2024, 83(3): 6555-6578.
- [4] Mora A, Prados A, Mendez A, et al. ADAM: a robotic companion for enhanced quality of life in aging populations[J]. Frontiers in Neurorobotics, 2024, 18: 1337608.
- [5] Huang J, Thurow K, Junginger S, et al. IoT based Labware Tracking during Mobile Robot Transportation[C]//2023 12th International Conference on Control, Automation and Information Sciences (ICCAIS). 2023: 405-410.
- [6] Yuan H, Li H, Zhang Y, et al. Comparison and improvement of local planners on ros for narrow passages[C]//2022 International Conference on High Performance Big Data and Intelligent Systems (HDIS). 2022: 125-130.
- [7] Rösmann C, Feiten W, Wösch T, et al. Efficient trajectory optimization using a sparse model[C]//2013 European Conference on Mobile Robots. 2013: 138-143.
- [8] Fox D, Burgard W, Thrun S. The dynamic window approach to collision avoidance[J]. IEEE Robotics & Automation Magazine, 1997, 4(1): 23-33.
- [9] Wanyu D, Lijuan Z, Jiafeng W, et al. Research on Local Path Planning Algorithm Based on Improved TEB Algorithm.[J]. Journal of Computer Engineering & Applications, 2022, 58(8): 283.
- [10] LUAN T, WANG H, YOU B, et al. TEB unmanned vehicle navigation method for position and attitude auxiliary points in narrow space[J]. Chinese Journal of Scientific Instrument, 2023, 44(4): 121-128.
- [11] Moreno F A, Monroy J, Ruiz-Sarmiento J R, et al. Automatic waypoint generation to improve robot navigation through narrow spaces[J]. Sensors, 2019, 20(1): 240.
- [12] Kiss D, Papp D. Effective navigation in narrow areas: A planning method for autonomous

- cars[C]//2017 IEEE 15th international symposium on applied machine intelligence and informatics (SAMI). 2017: 423-430.
- [13] Kim D, Chung W, Park S. Practical motion planning for car-parking control in narrow environment[J]. IET control theory & applications, 2010, 4(1): 129-139.
- [14] Banzhaf H, Palmieri L, Nienhäuser D, et al. Hybrid curvature steer: A novel extend function for sampling-based nonholonomic motion planning in tight environments[C]//2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC). 2017: 1-8.
- [15] Wang W, Xu X, Li Y, et al. Triple RRTs: an effective method for path planning in narrow passages[J]. Advanced Robotics, 2010, 24(7): 943-962.
- [16] Hsu D, Jiang T, Reif J, et al. The bridge test for sampling narrow passages with probabilistic roadmap planners[C]//2003 IEEE international conference on robotics and automation (cat. no. 03CH37422): vol. 3. 2003: 4420-4426.
- [17] Lai T, Ramos F, Francis G. Balancing global exploration and local-connectivity exploitation with rapidly-exploring random disjointed-trees[C]//2019 International Conference on Robotics and Automation (ICRA). 2019: 5537-5543.
- [18] Zhong T, Wei F, Zhai Z, et al. An untethered miniature soft jumping robot inspired by quadrupeds[J]. Journal of Bionic Engineering, 2023, 20(4): 1467-1480.
- [19] Liu J, Wang Y, Li M, et al. Joint Linkage and Motion System Design of Pipeline Detecting Snake Robot[C]//2023 35th Chinese Control and Decision Conference (CCDC). 2023: 1545-1550.
- [20] Liu Y, Barati Farimani A. An Energy-Saving Snake Locomotion Pattern Learned in a Physically Constrained Environment With Online Model-Based Policy Gradient Method [J]. Journal of Mechanisms and Robotics, 2023, 15(4): 041007.
- [21] Salaris P, Vassallo C, Soueres P, et al. The geometry of confocal curves for passing through a door[J]. IEEE Transactions on Robotics, 2015, 31(5): 1180-1193.
- [22] Aude E P, Lopes E P, Aguiar C S, et al. Door crossing and state identification using robotic vision[J]. IFAC Proceedings Volumes, 2006, 39(15): 659-664.
- [23] Kim S, Cheong H, Kim D H, et al. Context-based object recognition for door detection [C]//2011 15th International Conference on Advanced Robotics (ICAR). 2011: 155-160.

- [24] Dai D, Jiang G, Xin J, et al. Detecting, locating and crossing a door for a wide indoor surveillance robot[C]//2013 IEEE international conference on robotics and biomimetics (ROBIO). 2013: 1740-1746.
- [25] Kim B K, Tanaka H, Sumi Y. Robotic wheelchair using a high accuracy visual marker lensbar and its application to door crossing navigation[C]//2015 IEEE International Conference on Robotics and Automation (ICRA). 2015: 4478-4483.
- [26] 王振东, 郭晓军, 王海峰, 等. 基于 RFID 的 AGV 定位与磁导航[J]. 装备制造技术, 2021(07): 106-110.
- [27] 卢光旭. 一种基于波峰寻找法的 AGV 色带导航技术[J]. 中国新技术新产品, 2020(09): 33-35.
- [28] Wu Y, Ge D. Key technologies of warehousing robot for intelligent logistics[C]//The First International Symposium on Management and Social Sciences (ISMSS 2019). 2019: 79-82.
- [29] Samuel M, Hussein M, Mohamad M B. A review of some pure-pursuit based path tracking techniques for control of autonomous vehicle[J]. International Journal of Computer Applications, 2016, 135(1): 35-38.
- [30] Johnson M A, Moradi M H. PID control[M]. Springer, 2005.
- [31] Morales S, Magallanes J, Delgado C, et al. LQR trajectory tracking control of an omnidirectional wheeled mobile robot[C]//2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA). 2018: 1-5.
- [32] Bai G, Meng Y, Liu L, et al. Review and comparison of path tracking based on model predictive control[J]. Electronics, 2019, 8(10): 1077.
- [33] Furgale P, Barfoot T D. Visual teach and repeat for long-range rover autonomy[J]. Journal of field robotics, 2010, 27(5): 534-560.
- [34] Gridseth M, Barfoot T D. DeepMEL: Compiling visual multi-experience localization into a deep neural network[C]//2020 IEEE International conference on robotics and automation (ICRA). 2020: 1674-1681.
- [35] Camara L G, Pivoňka T, Jílek M, et al. Accurate and robust teach and repeat navigation by visual place recognition: A CNN approach[C]//2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2020: 6018-6024.

- [36] Dall' Osto D, Fischer T, Milford M. Fast and robust bio-inspired teach and repeat navigation[C]//2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2021: 500-507.
- [37] Toudehki A G, Shamshirdar F, Vaughan R. Robust uav visual teach and repeat using only sparse semantic object features[C]//2018 15th Conference on Computer and Robot Vision (CRV). 2018: 182-189.
- [38] Sarlin P E, DeTone D, Malisiewicz T, et al. Superglue: Learning feature matching with graph neural networks[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 4938-4947.
- [39] Fischler M A, Bolles R C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography[J]. Communications of the ACM, 1981, 24(6): 381-395.
- [40] Mahdavian M, Yin K, Chen M. Robust visual teach and repeat for ugvs using 3d semantic maps[J]. IEEE Robotics and Automation Letters, 2022, 7(4): 8590-8597.
- [41] Mur-Artal R, Tardós J D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras[J]. IEEE transactions on robotics, 2017, 33(5): 1255-1262.
- [42] Farhadi A, Redmon J. Yolov3: An incremental improvement[C]//Computer vision and pattern recognition: vol. 1804. 2018: 1-6.
- [43] Quigley M, Conley K, Gerkey B, et al. ROS: an open-source Robot Operating System[C] //ICRA workshop on open source software: vol. 3: 3.2. 2009: 5.
- [44] Grisetti G, Stachniss C, Burgard W. Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters[J]. IEEE Transactions on Robotics, 2007, 23(1): 34-46.
- [45] Kohlbrecher S, von Stryk O, Meyer J, et al. A flexible and scalable SLAM system with full 3D motion estimation[C]//2011 IEEE International Symposium on Safety, Security, and Rescue Robotics. 2011: 155-160.
- [46] 李梟凯, 李广云, 索世恒, 等. 激光 SLAM 技术进展[J]. 导航定位学报, 2023, 11(04): 8-17.
- [47] Khan M U, Zaidi S A A, Ishtiaq A, et al. A Comparative Survey of LiDAR-SLAM and LiDAR based Sensor Technologies[C]//2021 Mohammad Ali Jinnah University International Conference on Computing (MAJICC). 2021: 1-8.

- [48] Grisetti G, Kümmerle R, Stachniss C, et al. A tutorial on graph-based SLAM[J]. IEEE Intelligent Transportation Systems Magazine, 2010, 2(4): 31-43.
- [49] Lu F, Milios E. Globally consistent range scan alignment for environment mapping[J]. Autonomous robots, 1997, 4: 333-349.
- [50] Hess W, Kohler D, Rapp H, et al. Real-time loop closure in 2D LIDAR SLAM[C]//2016 IEEE international conference on robotics and automation (ICRA). 2016: 1271-1278.
- [51] Konolige K, Grisetti G, Kümmerle R, et al. Efficient sparse pose adjustment for 2D mapping[C]//2010 IEEE/RSJ International Conference on Intelligent Robots and Systems. 2010: 22-29.
- [52] Macenski S, Jambrecic I. SLAM Toolbox: SLAM for the dynamic world[J]. Journal of Open Source Software, 2021, 6(61): 2783.
- [53] Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: a versatile and accurate monocular SLAM system[J]. IEEE transactions on robotics, 2015, 31(5): 1147-1163.
- [54] Campos C, Elvira R, Rodríguez J J G, et al. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam[J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-1890.
- [55] Qin T, Shen S. Online Temporal Calibration for Monocular Visual-Inertial Systems[C] //2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2018: 3662-3669.
- [56] Qin T, Li P, Shen S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator[J]. IEEE Transactions on Robotics, 2018, 34(4): 1004-1020.
- [57] Thrun S. Probabilistic robotics[J]. Communications of the ACM, 2002, 45(3): 52-57.
- [58] Macenski S, Booker M, Wallace J. Open-Source, Cost-Aware Kinematically Feasible Planning for Mobile and Surface Robotics[J]. Arxiv, 2024.
- [59] Rösmann C, Hoffmann F, Bertram T. Kinodynamic trajectory optimization and control for car-like robots[C]//2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2017: 5681-5686.
- [60] Williams G, Drews P, Goldfain B, et al. Aggressive driving with model predictive path integral control[C]//2016 IEEE International Conference on Robotics and Automation (ICRA). 2016: 1433-1440.

- [61] Quinlan S, Khatib O. Elastic bands: Connecting path planning and control[C]// [1993] Proceedings IEEE International Conference on Robotics and Automation. 1993: 802-807.
- [62] Whyte R, Streeter L, Cree M J, et al. Application of lidar techniques to time-of-flight range imaging[J]. Applied optics, 2015, 54(33): 9654-9664.
- [63] Gasparetto A, Boscaroli P, Lanzutti A, et al. Path planning and trajectory planning algorithms: A general overview[J]. Motion and operation planning of robotic systems: Background and practical approaches, 2015: 3-27.
- [64] Varma B, Swamy N, Mukherjee S. Trajectory tracking of autonomous vehicles using different control techniques (pid vs lqr vs mpc)[C]// 2020 International conference on smart technologies in computing, electrical and electronics (ICSTCEE). 2020: 84-89.
- [65] Rösmann C, Makarow A, Bertram T. Online motion planning based on nonlinear model predictive control with non-euclidean rotation groups[C]// 2021 European Control Conference (ECC). 2021: 1583-1590.
- [66] Kümmerle R, Grisetti G, Strasdat H, et al. g 2 o: A general framework for graph optimization[C]// 2011 IEEE international conference on robotics and automation. 2011: 3607-3613.
- [67] Aubin J P, Aubin J P. Set-valued analysis[M]. Springer, 1999.
- [68] Aubin J P, Frankowska H, Aubin J P, et al. Differential inclusions[M]. Springer, 2009.
- [69] Doyen L. Shape Lyapunov functions and stabilization of reachable tubes of control problems[J]. Journal of mathematical analysis and applications, 1994, 184(2): 222-228.
- [70] Doyen L. Mutational equations for shapes and vision-based control[J]. Journal of Mathematical Imaging and Vision, 1995, 5: 99-109.
- [71] Li C, Wang S, Wang S, et al. Robot motion control with compressive feedback[C]// 2021 IEEE International Conference on Robotics and Automation (ICRA). 2021: 12925-12930.
- [72] Song B, Zhao J, Xi N, et al. Non-vector space control for nanomanipulations based on compressive feedbacks[C]// 2012 IEEE International Conference on Robotics and Automation. 2012: 2767-2772.
- [73] Liu X, So C H, Huang H, et al. 3D Stereoscopic Display in the Event-Based Real-Time Internet-Based Robotic Teleoperation System[C]// 2022 IEEE International Conference

- on Robotics and Biomimetics (ROBIO). 2022: 2299-2304.
- [74] Sheng Y, Ye J, Wang S, et al. Event-Potential Method for Mobile Robot Motion Planning and Control[C]//2023 WRC Symposium on Advanced Robotics and Automation (WRC SARA). 2023: 444-450.
- [75] Chapman B, Jost G, Van Der Pas R. Using OpenMP: portable shared memory parallel programming[M]. MIT press, 2007.

## 攻读博士/硕士学位期间取得的研究成果

一、已发表（包括已接受待发表）的论文，以及已投稿、或已成文打算投稿、或拟成文投稿的论文情况（只填写与学位论文内容相关的一部分）：

序号	作者（全体作者，按顺序排列）	题目	发表或投稿刊物名称、级别	发表的卷期、年月、页码	与学位论文哪一部分（章、节）相关	被索引收录情况
1	董敏，谭皓禹，杨礼铭，沈煜，陈章韶，毕盛	面向独居老人的智能家居监护系统	嵌入式技术与智能系统	2024 年 8 月	第三章	Hans
2	董敏，谭皓禹，毕盛	面向狭窄环境的机器人室内导航系统	计算机科学，北大核心	2024 年 11 月	第三章	CNKI

二、与学位内容相关的其它成果（包括专利、著作、获奖项目等）

- 1) 发明专利：董敏, 谭皓禹, 毕盛. 一种机器人室内导航系统自适应狭窄环境的方法 [P]. 广东省: CN202410028012.9, 2024-04-26. (实质审查生效中)
- 2) 获奖项目：2023 “兆易创新杯”第十八届中国研究生电子设计竞赛，全国总决赛二等奖、安谋科技专项企业奖一等奖
- 3) 获奖项目：2023 昇腾 AI 创新大赛 2023-开发者套件创新赛道，广州赛区铜奖

## 致 谢

时光荏苒，三年的研究生生涯即将结束，有幸得到诸多良师益友相伴，在此我想向所有帮助过我的人致以最真挚的谢意。

首先我要感谢父母二十多年的养育之恩，你们一直是我前行路上最坚实的后盾，用无私的爱与辛勤的付出让我始终心无旁骛地追逐梦想。惟愿未来我能成为你们的骄傲，亦如你们一直是我的港湾。

其次感谢我的导师董敏教授与毕盛教授。在机器人领域的探索中，两位老师以深厚的学术造诣与前瞻性视野为我点亮明灯。老师们在机器人软硬件上的支持、指引研究方向与用心提出论文修改建议都为我学术生涯中的成长提供了莫大的帮助。能够师从二位在机器人领域深耕，是我研究生生涯最珍贵的际遇。

实验室 B3351 是温暖的大家庭。感谢已毕业的师兄师姐们，从在校期间的代码调试与硬件测试，到毕业后留下的技术文档与经验分享都让我少走了无数弯路；感谢并肩作战的同窗伙伴，那些在实验室敲代码、调小车的日子因有你们的鼓励与帮助而充满斗志；感谢实验室的所有同学，每周打羽毛球成了我科研之余最期待的时光，球场上的酣畅淋漓驱散了科研的疲惫，也让我真正感受到了运动的魅力。以上种种都将成为 B3351 留给我最鲜活的记忆。

山水一程，三生有幸。愿此去经年，我们都能在各自的征途上乘风破浪，于更高处重逢。