
全自主泊车系统控制策略和算法研究

徐大陆

2014 年 6 月

中图分类号：TP242.6

UDC 分类号：621

全自主泊车系统控制策略和算法研究

作者姓名	<u>徐大陆</u>
学院名称	<u>机械与车辆学院</u>
指导教师	<u>龚建伟</u>
答辩委员会主席	<u>陈慧岩教授</u>
申请学位	<u>工程硕士</u>
学科专业	<u>车辆工程</u>
学位授予单位	<u>北京理工大学</u>
论文答辩日期	<u>2014 年 6 月</u>

Control Strategy And Algorithm Research in Autonomous Parking

Candidate Name:	<u>Dalu Xu</u>
School or Department:	<u>School of Mechanical Engineering</u>
Faculty Mentor:	<u>Associate Prof. Gong Jianwei</u>
Chair, Thesis Committee:	<u>Prof. Chen Huiyan</u>
Degree Applied:	<u>Master of Engineering</u>
Major:	<u>Mechanical Engineering</u>
Degree by:	<u>Beijing Institute of Technology</u>
The Date of Defence:	<u>June, 2014</u>

全自主泊车系统控制策略和算法研究

北京理工大学

研究成果声明

本人郑重声明：所提交的学位论文是我本人在指导教师的指导下进行的研究工作获得的研究成果。尽我所知，文中除特别标注和致谢的地方外，学位论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京理工大学或其它教育机构的学位或证书所使用过的材料。与我一同工作的合作者对此研究工作所做的任何贡献均已在学位论文中作了明确的说明并表示了谢意。

特此申明。

签 名：徐大陆 日期：20140618

关于学位论文使用权的说明

本人完全了解北京理工大学有关保管、使用学位论文的规定，其中包括：①学校有权保管、并向有关部门送交学位论文的原件与复印件；②学校可以采用影印、缩印或其它复制手段复制并保存学位论文；③学校可允许学位论文被查阅或借阅；④学校可以学术交流为目的，复制赠送和交换学位论文；⑤学校可以公布学位论文的全部或部分内容（保密学位论文在解密后遵守此规定）。

签 名：徐大陆 日期：20140618

导师签名：龚建伟 日期：20140618

摘要

随着汽车智能化的发展,对于自动泊车的研究逐渐引起了人们的关注。分析总结得知,现有的自动泊车研究大多停留在半自动的层面上,系统仅能够完成垂直或平行等特定类型的自动泊车,对车辆起始位置有限制,泊车过程中需要驾驶员控制车辆的加减速和换挡,因此与全自主泊车还存在一定差距。

本文旨在设计一种全自主泊车系统,即在驾驶员不参与的情况下车辆能够完成任意指定位置的自动泊车,具体实现以下目标:设计合理的规划算法,搜索到一条从泊车起始位置到任意终点位置的期望路径,路径的生成考虑车辆运动学的约束;设计路径跟踪算法,满足泊车工况下车辆在狭小空间行驶时对误差精度的要求;同时,提出高精度的车辆定位方法;当期望路径被障碍物占据时,系统能够重新规划出新的可行路径;对车体控制系统进行改造,实现车辆纵横向的自动控制。

为实现全自主泊车系统,本文首先利用激光雷达传感器获取障碍物信息,建立了车辆周围的二维环境地图,地图采用占据栅格的方式,能够很好地表征车辆周围的实际环境信息;基于二维栅格地图,选定合适的停泊位置及朝向后,利用启发式搜索算法在 Lattice 状态空间中规划出可行的泊车路径,规划中考虑车辆运动学约束,确保了路径规划的可行性;为了满足自主泊车工况下对精确跟踪控制的要求,本文设计了基于前馈-反馈方法的路径跟踪算法,算法在反馈控制的基础上引入前馈补偿,一定程度上提高了跟踪精度,并通过仿真验证了算法的可行性。在跟踪过程中,根据车载传感器信息和地图的匹配实现定位,通过直方图滤波获得车辆的最大概率位置,满足了在地下停车场等没有或 GPS 信号较弱的场景下对车辆定位的需求。当跟踪失败时能够重新规划,搜索新的可行路径。最后本文搭建了系统所需的硬件平台,包括传感器、工控机、执行机构以及各控制器的设计,该平台能够实现车辆纵横向的自动控制。

本系统在北京理工大学无人驾驶车辆 Ray 上完成测试,该试验车是国内第一辆实现转向、档位以及油门系统线控的无人驾驶车辆,实验结果表明该系统能较好地完成平行、垂直以及任意位姿的自动泊车,同时验证了路径规划、路径跟踪、车辆定位等系统各个算法的有效性。

关键词：自动泊车；路径规划；路径跟踪；地图创建与车辆定位；智能车辆

Abstract

With the development of intelligent vehicle, automatic parking study gradually attracts people's attention. Today's high-end models are also equipped with this feature. Through analysis and summary, we get that existing automatic parking study stays on semi-automatic level and there is still a gap between the autonomous parking.

This paper aims to design a fully autonomous parking system, and to achieve the following specific objectives: design a path planning algorithm to search a desired path from the start position to any parking plot, so that the system can complete the parallel, vertical and any type of parking; Design a path tracking algorithm to meet the accuracy requirements for vehicle when tracking along the desired path in parking condition; Besides, propose a high-accuracy positioning method for the vehicle; When the desired path is occupied by obstacles, the system can re-planning with a new feasible path, and the vehicle's longitudinal control such as acceleration and deceleration, and shift operations can be controlled automatically.

In order to achieve a full self-parking system, the paper has done some researches. Relying on sensors, the system can complete any kind of parking, and the initial and final position and orientation of the vehicle is arbitrary. In the parking process the vehicle's vertical and horizontal movement is controlled all by the system automatically. The system firstly established a two-dimensional environment map around the vehicle. When an appropriate parking location and orientation on the map is selected, the collision detection system will judge that whether the selected parking spaces meets the requirement of the vehicle size. Then the system plans a workable parking path on the basis of application of heuristic search algorithms in the lattice state space. In order to meet the accurate tracking control requirements in the process of self-parking, based on LQR, we design a feed-forward and feedback path tracking algorithm which is verified reliable by simulation. In the tracking process the vehicle achieves precise positioning by the map matching method based on in-vehicle sensors, which meets the requirements of positioning in the underground parking lot scene where GPS signals are poor or none. We add collision analysis to the parking system, when tracking fails it can re-plan to get a new feasible path. Finally, we designed the system including hardware platforms, sensors, industrial machines, actuators, and controllers, and finally set up the automatic parking experiment platform.

The system test is completed based on the unmanned vehicle of Beijing Institute of Technology. The test vehicle is the first one in China whose gear, throttle and

braking systems are all controlled automatically. Experimental results show that the system can accomplish parallel, perpendicular and arbitrary pose automatic parking perfectly.

Key Words: Automatic parking; Path planning; Path tracking; Map creation and Vehicle location; Intelligent Vehicle

目录

摘要	1
Abstract	III
第 1 章 绪论	1
1.1 课题研究的背景和意义	1
1.2 国内外研究现状及发展趋势	2
1.2.1 自动泊车研究现状	2
1.2.2 自动泊车应用现状	4
1.3 课题研究内容	5
1.4 论文结构	7
第 2 章 地图创建与车辆定位	9
2.1 基于激光雷达的地图创建	9
2.1.1 激光雷达	9
2.1.2 栅格地图创建	10
2.2 车辆定位方法	12
2.3 SLAM 同步定位与地图创建	13
2.3.1 传感器测量模型	13
2.3.2 车辆的定位	14
2.3.3 地图的更新	17
2.4 SLAM 定位结果验证	18
2.5 本章小结	21
第 3 章 全自主泊车路径规划及算法研究	23
3.1 路径规划概述	23
3.2 全局路径规划	23
3.2.1 Dijkstra 算法	23
3.2.2 A* 算法	26
3.2.3 算法比较	29
3.3 算法验证	30
3.3.1 仿真分析	31
3.3.2 实验结果	32

3.4 本章小结.....	32
第 4 章 全自主泊车路径跟踪算法研究	33
4.1 路径跟踪算法概述.....	33
4.1.1 常用路径跟踪算法.....	33
4.1.2 路径跟踪算法总结.....	37
4.2 智能车辆运动学模型.....	37
4.2.1 车辆几何模型.....	38
4.2.2 车辆运动学模型.....	39
4.3 前馈-反馈跟踪控制算法	41
4.3.1 反馈控制.....	41
4.3.2 前馈控制.....	43
4.3.3 前馈-反馈控制的融合	45
4.4 前馈-反馈路径跟踪算法仿真实验分析	46
4.4.1 曲线跟踪场景.....	46
4.4.2 直线-曲线跟踪场景	50
4.5 本章小结.....	51
第 5 章 实车实验	52
5.1 全自主泊车试验平台开发.....	52
5.1.1 全自主泊车实验平台设计.....	52
5.1.2 全自主泊车实车实验平台实现.....	53
5.1.3 智能车辆实验平台集成.....	55
5.2 全自主泊车系统实验设计及结果.....	56
5.2.1 平行泊车.....	56
5.2.2 垂直泊车.....	58
5.2.3 任意姿态泊车.....	59
5.3 本章小结.....	61
第 6 章 总结与展望.....	62
6.1 工作总结.....	62
6.2 课题展望.....	62
主要符号对照表.....	64
攻读学位期间发表论文与研究成果清单	65

致谢	66
参考文献	67

第1章 绪论

1.1 课题研究的背景和意义

随着经济的发展和人民生活水平的不断提高,一方面汽车拥有量越来越多,停车场、街道、公路、居民小区等拥挤不堪,可以泊车的空间越来越少;另一方面,驾车新手逐年增多,由于不熟练导致的各种问题也逐渐增多。美国密歇根大学交通研究所的Paul Green的研究表明,通过分析交通事故数据库统计资料和保险公司事故统计资料,由泊车导致的事故占到各类事故的44%,其中大约1/2到3/4的泊车碰撞是倒车造成的^[1]。这表明泊车是驾驶员容易出问题而导致交通事故的一个重要原因,如何改善汽车的操控性,尤其是泊车过程中的不便利,消除安全隐患,迅速、准确、安全地将汽车停靠到合适的位置,逐渐引起了人们的关注。

汽车在泊车过程中的困难主要有三方面:一是驾驶员的视野有限,驾车者在驾驶座上很难完全看清楚后方的情况,只能通过后视镜来观察车尾部情况,然而由于位置、天气等因素,后视镜又往往难以起到良好的效果;二是对于经验较少的驾驶员,通过后视镜来观察车尾情况,需要推理反向视角的变化,又需要频繁扭头观察真实情况,同时还需要控制方向盘、油门和制动等,容易造成操作失误;三是对于不熟悉的环境或者车位狭窄的情况下,驾车新手由于缺乏经验技巧,或者对车型部件的灵敏程度不熟悉,往往难以很好地控制汽车进行快速准确的入位。

如今汽车已经不仅仅作为一项机械产品,更承载了许多先进的电子设备,往智能汽车^[2]的方向发展,即致力于改善人们的驾驶乐趣和汽车的安全性,把人们从简单枯燥的驾驶行为中解放出来。因此,各大汽车厂商对于汽车电子^[3]都给予了很大关注,作为汽车辅助驾驶^[4]的一个领域,自动泊车技术^[5]将有着良好的应用价值和市场前景。

目前对于自动泊车系统的研究^{[6][7][8]},国外和港台地区的研究成果较为先进,国内由于起步较晚的原因,相关技术较为落后。国内外大部分研究者都以无障式平行或垂直泊车类型^{[9][10]}为研究重点,或采用路径规划的方式,或基于经验的方式完成泊车功能,但当车位为任意类型即非平行或垂直车位的场景时,系统将不能完成泊车,这就造成了可以停泊空间的未充分利用。另外现今的研究都假设车辆在泊车过程中没有障碍物,这与实际的情况不相符。

因此,设计一种能够实现任意位姿停车的全自主泊车系统将具有重要意义,

即系统能够按照驾驶员的意图完成任意位姿的停车,泊车过程中车辆的纵横向控制均由系统接管,并采用较高精度的定位和跟踪方法,同时跟踪失败时,判断是否需要重启规划,搜索一条能够避开障碍物的新路径。

1.2 国内外研究现状及发展趋势

1.2.1 自动泊车研究现状

针对自动泊车问题,国外和我国港台地区的研发工作起步较早,并已有汽车制造商将其应用到实际车型中,目前自动泊车技术主要有两大类研究方法:一是路径规划。即事先规划一个可行的几何路径,考虑环境约束、汽车运动学或动力学模型等;二是基于经验的控制算法。根据泊车熟练的司机的经验,模拟司机的驾车行为,实时产生控制命令。这种方法与车辆相对于泊车位的方向和位置有关,没有参考路径可循。在对自动泊车问题的研究中,由于汽车系统的复杂性和实验的不便利,许多研究者采用与四轮式移动机器人相类似的方法。

(1) 基于路径规划方法的研究

路径规划方法通过生成既定路径控制车辆的运动轨迹。这种方法依赖于控制器的精确性,控制效果并非完全可靠。在泊车过程中,即使位置误差能够被后来的运动所补偿,也会增加相应的时间成本,从而影响自动泊车的效果。

芝加哥大学的 Paromtchik 提出了非完整约束条件下平行泊车的方法^[11]。在泊车前,先扫描出泊车空间,接着小车按照正弦曲线的轨迹向后运动。在此过程中,通过控制转向角变化和速度来生成相应的一条运动轨迹。为了保证车辆在倒车过程中不会碰撞到其他车辆或墙壁,根据泊车位的长宽以及车辆距离障碍物的距离进行离线查表,生成一条无碰撞的轨迹路线。

英国伯明翰大学的 K-Jian 研究了基于传感器引导的自动泊车系统^[12]。他提出自动泊车的过程包括三个阶段:扫描空间、定位、调整,最终生成了一条无碰撞的泊车路径。考虑到泊车空间可能比较狭小,或者转向角范围较窄,车体会做出一定的向前或向后的调整。考虑到速度变化对泊车控制策略带来的影响,在泊车过程中也进行了速度的控制。泊车系统安装在移动机器人上进行了实验,并最终证明了探测到空间的有效性及生成轨迹的可靠性。

北京航空航天大学的刘亮^[13]等提出了一种自动泊车方法,文章采用双目相机检测停车位,并进行图像三维重建构建泊车环境约束地图,在此基础上,采用

双向路径规划的方法得到泊车期望路径，在低约束区采用正向道路规划，高约束区域采用逆向路径规划，路径由圆弧和直线组合而成。然而该系统只适用于垂直泊车位的停车，泊车过程中没有考虑障碍物等环境信息。

中国科学院合肥智能机械研究所的高航等借助于中国科学院合肥物质研究所^[14]的“智能先锋”号无人车完成了自动泊车任务。泊车位的位置由路网信息给出，车辆行驶到泊车位附近，通过激光雷达和车载摄像机检测车位的类型。根据阿克曼转向原理通过圆弧拼接的方式规划出了泊车路径，并通过触角法选择一条无碰撞且最接近期望路径的路径行驶。但该方法仅适用于平行、垂直以及单段泊车，泊车的起始位置受到限制。

Kyoungwook Min^[15]等提出了一种车辆行驶和泊车的路径生成方法，车辆在行驶模式时采用 A*算法规划出可行的道路，创建了栅格地图，对车辆进行 4 维运动约束。在泊车位附近则采用相切圆的方式进行泊车期望路径规划。

（2）基于经验方法的研究

基于经验的方法采用智能控制算法对自动泊车问题进行研究。以模糊控制方法为例，在获得小车的方向角后，基于小车相对泊车位置的纵向和横向距离，对不同的泊车位置，不同的泊车阶段会产生不同的模糊规则。模糊控制会在每一个采样周期，根据小车相对于泊车位置的相对位姿，产生一个控制命令，可以有效补偿系统误差。

美国佛罗里达州大学的 Zhao 等^[16]提出了在紧凑空间中基于模糊逻辑的自动平行泊车鲁棒控制算法。作者将小车看作是刚体，前轮运动方向与车身运动方向相同，建立了小车的运动学方程，设计了相应的模糊控制器并进行了仿真，验证了其算法的可靠性。并以四轮移动小车 ATRV-Jr 为对象进行了自动泊车实验。实验证明，在定位误差和泊车空间探测存在误差的情况下，小车仍然能够在紧凑空间内平行泊车。

希腊的 G-Ggato 提出了一种智能反应控制器^[17]，可以在紧凑空间内实现自动平行泊车。文章结合了模糊逻辑控制和滑模变结构控制，设计了一种降维滑模变结构模糊逻辑控制器。控制器通过强化学习机制和自动反应机制，能够对系统参数进行自主调整，在运动中实时调节，达到良好的控制效果。文章还对系统鲁棒性和噪声问题进行了探讨。文章认为系统干扰主要来源于传感器信息的误差，但由于其设计的自动模糊学习机制已经考虑到这种误差的存在，所以能够及时修正小车前进的方向角。

西班牙的 F.Gomez-Bravo 等^[18]针对非完整约束下平行泊车 and 斜行泊车问题,设计了一种模糊控制器。控制器根据环境信息做出最优路径选择并考虑避障。文章对自动泊车问题做了大量的数学分析,针对泊车的不同情形,分别设计了相应的控制算法。作者通过与实车比例类似的轮式移动机器人 ROMEO-3R 的泊车实验,证明了控制算法的有效性。

北京师范大学的杨昔阳等^[19]作了基于变论域理论的自动倒车问题研究。基于变论域理论,作者提出了一种自适应模糊控制器。结合卡车倒车的控制模型,进行了仿真实验,并与普通模糊控制器的控制效果进行了对比。仿真效果显示,其设计的自适应模糊控制器效果更好。

新加坡国立大学^[20]的研究者提出了一种全自动泊车方案,该方案由两部分组成,分别是车辆本身和控制中心,泊车工作独立自主完成,驾驶员只需把车辆停在停车场入口即可。泊车开始前车辆接收控制中心的指令进入停车场,该停车场具有一定的假设,停车位均设在车辆的左侧,车辆的右侧为墙体,车辆首先沿着墙体直行,并通过自身的传感器检测可行的停车位,并把传感器信息实时发送给控制中心,控制中心通过分析判断是垂直车位还是平行车位。寻找到可行车位时,命令车辆停车,并根据自身相对于车位位置做出泊车策略,完成泊车。其中车位检测通过红外传感器检测车位的宽度,车辆行驶的距离检测车位长度,通过模糊逻辑分析得到泊车位的类型,基于模糊逻辑控制器控制前轮偏角完成泊车任务。该方案在模型车辆上完成了测试,停车场过于理想化,且只适用于垂直和平行泊车,对于任意合适的空间则不能实现。

1.2.2 自动泊车应用现状

自动泊车系统已经被看作是下一代汽车的热门应用产品。早在 2003 年,丰田推出了全面改进后的油电混合车普瑞斯(Prius)^[21]。该车型具有一个能够辅助纵向泊车和车库泊车的智能泊车助手。该功能是通过采用电动操纵系统和基于 CCD 摄像机的后视监视器而实现的。只需将汽车开到泊车位置的前方,然后在车载导航系统画面上根据提示指定泊车位置后,电动操纵系统将会自动工作,引导车辆到泊车位置,驾驶者只需操作刹车即可。安装在尾部的 CCD 摄像机可利用捕捉到的图像数据计算出与目标间的距离。距离计算的基点是后轮车轴的中间位置。首先计算出连接当前位置与目标位置的最佳路径曲线,然后将信息传输至操纵系统,再根据此信息启动操纵系统。当进行制动操作时,辅助动作就会暂时

关闭。

2005 年,雪铁龙 C3 就展示了车辆辅助泊车入位系统,城市泊车系统(City Park)^[22],它使车辆在拥挤的市内泊车入位变得轻松自如。2007 年春,雪铁龙 C4 毕加索装配了博世泊车入位测量系统而成为全世界第一款配备有完善的泊车入位辅助系统的汽车。司机需要泊车时,只需按下中控台上的操控按钮,泊车入位辅助系统就会开启,车速需保持在 25km/h 以下,系统会依靠位于车辆前、后保险杠上的 4 个测距传感器以及安装于前保险杠两侧 2 个额外的测距传感器,保证精确测量泊车位。驾驶辅助系统能够及时通知汽车驾驶者预期的泊车位长度是否适合该车辆,在泊车过程中遇到障碍物会有提示信号。

2007 年,丰田也在旗下豪华品牌雷克萨斯系列的顶级轿车 LS 系列上安装了智能泊车助手(Intelligent Parking Assist, IPA)^[23]。这套系统使用了电动助力转向(EPS),在汽车倒车泊车时,方向盘可以自动操作,完成泊车入位的动作。这套 IRA 智能泊车系统使用超声波传感器检测泊车位置,并能结合摄像头识别泊车位线。当汽车经过或靠近泊车位位置时,汽车自动检测到泊车位位置和距离,并在显示屏上用线条标识出泊车位位置,此时只需驾驶员确认并换成倒车档,启动自动泊车系统,汽车就能够自动控制方向盘进行倒车入位。此过程中驾驶者只需确认安全和调整速度,超声波会一直检测四周直到泊车完毕,避免碰撞的发生。

2013 款福特蒙迪欧车型也装配了自动泊车系统^[24]。该系统由人机交互系统、环境数据采集系统、中央处理器和控制策略执行系统四部分组成。首先,驾驶员开启泊车功能,并以时速小于 35km/h 的速度行驶过可以停泊的区域,此时车载传感系统开始工作。系统探测到合适的车位后,会提示驾驶员继续前行,行进到系统认为合适的位置后提示驾驶员停车、挂入倒车挡,然后中央处理器将采集到的环境信息数据分析处理后,得出汽车的当前位置、目标位置以及周围的环境参数,依据以上参数做出自动泊车策略,生成相应的控制命令。执行机构接收控制命令控制车辆转向,并结合驾驶员的纵向控制完成泊车。目前此款车型只支持平行泊车,还不能完成倒车入库以及任意泊车位停车。

1.3 课题研究内容

本课题对自主泊车的路径规划与跟踪控制算法进行研究,结合车辆运动学模型,实现车辆在已知环境下的指定位置行驶。全自主泊车系统研究内容如图 1.1 所示,主要分为地图创建、车辆定位、路径规划与跟踪控制四个部分。其中地图

创建采用 SLAM 的方法，轨迹规划方面，采用 A*规划算法，在跟踪控制方面，设计基于前馈-反馈算法的轨迹跟踪控制器。主要功能包括以下几个方面：

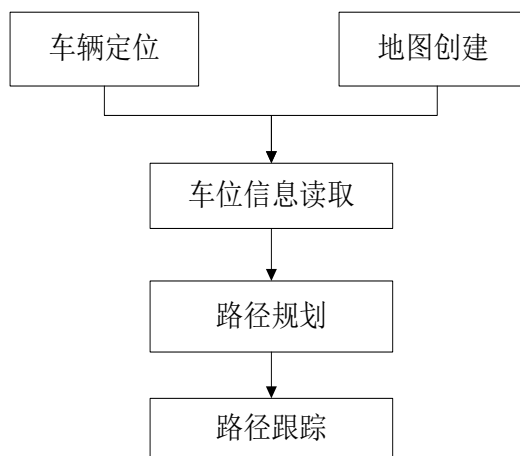


图 1.1 中央控制器功能框图

（1）地图创建与车辆定位

为真实反映车辆周围障碍物以及可通行区域的信息，利用激光雷达传感器对环境进行建模，创建了泊车环境地图，地图由占据栅格的方式来表达，把栅格被障碍物占据与概率关联起来，该地图较完整地表达了车辆所在的环境参数，驾驶员根据地图信息选择合适的停车位；同时通过传感器信息和地图的匹配实现车辆的精确定位，为后续的路径跟踪算法提供车辆的位置信息。

（2）路径规划算法研究

主要包含全局路径规划。路径规划首先在全局路径的引导下，规划出一条无碰撞的几何路线，规划的同时考虑到车辆的运动学约束，避免车辆无法跟踪期望路径的现象出现。

（3）路径跟踪算法研究

主要分为两部分内容，一是车辆的运动学建模。根据车辆几何尺寸和转向特征，建立合理的车辆运动学模型。同时考虑车辆前进与后退的异同之处，为下一步跟踪算法的设计做好准备。

二是轨迹跟踪控制器设计，该部分主要根据实际跟踪要求，选择合理的跟踪方法，确定输入输出量，设计一种具备可靠性与实时性的轨迹跟踪控制器。同时，确定较优的控制参数。

（4）仿真与实验

针对上述的算法，进行仿真与实验研究，以验证算法的可行性。其中仿真采用 P3-AT 轮式移动机器人验证算法的可行性，实车实验在北京理工大学无人驾

驶车辆 Ray 上进行，以验证系统的可行性。

1.4 论文结构

论文针对全自主泊车系统及相关技术展开了研究，具体内容包括泊车栅格地图创建、基于地图匹配的车辆定位，在 Lattice^[25]状态空间中的路径规划，并且建立了无人驾驶车辆的运动学模型，对前馈-反馈路径跟踪控制器进行设计，以及实验平台的搭建和车辆纵横向的自动控制，同时对每个模块的算法进行仿真验证，最后在实际车辆中完成自主泊车实验，本论文的结构如下。

第一章：主要阐述自动泊车技术的研究背景、国内外研究现状及发展趋势。具体包括自动泊车研究的必要性、研究机理及国内外的研究现状。通过分析现今自动泊车系统存在的问题，表明开展全自主泊车的必要性，强调开展本研究的意义。

第二章：分析了地图创建以及车辆定位的意义，针对这两个部分，首先介绍了泊车环境地图的创建，该方法的实现基于激光雷达传感器，介绍了其工作原理，进而阐述了占据栅格地图创建的原理和方法；然后围绕目前常用的车辆定位方法做了比较，分析对比各种方法存在的不足，提出了较精确的车辆定位方法，即同步定位与地图创建方法，文中详细阐述了该方法的原理，并通过实验对比证明了其优越性。

第三章：主要阐述了全局路径规划方法。首先介绍了路径规划的意义和目的，分析了其在全自主泊车系统中的作用。重点介绍了两种全局规划算法，即 Dijkstra 算法和 A* 规划算法，分别对这两种算法的原理做了详细的说明，并分析对比了两种算法的优劣，最终选取 A* 方法作为本文的全局路径规划算法。最后通过仿真实验，在泊车环境地图中对该方法做了可行性验证。

第四章：本章主要介绍了自动泊车系统中用到的路径跟踪算法。首先介绍了国内外无人驾驶车辆常用的路径跟踪算法，并对各算法的优缺点以及适用场合做了对比；然后根据该系统适用的乘用车特点，建立了车辆运动学模型，提出了适用于低速、狭小空间内泊车场景的前馈-反馈跟踪算法，详细说明了该算法的原理，以及算法流程和输入输出量等；最后为验证算法的可行性，进行了仿真和实车试验分别让实验车辆跟踪直线和曲线，跟踪误差较小，满足自动泊车的要求。

第五章：进行实车实验。为了对本文设计的全自主泊车系统进行实车验证，利用了北京理工大学无人车辆 Ray 试验平台，首先对其纵横向控制系统进行改

造，实现车辆纵横向的自动控制；然后利用该平台进行了平行、垂直以及任意位姿的自动泊车，通过判断车辆最终是否停泊在选定的停车位内，来验证该系统的可行性和有效性。

文章结论部分：总结归纳了本论文在全自主泊车领域取得的研究成果，特别是该系统所用到的路径规划、路径跟踪以及车辆定位所用到的方法。总结了本论文研究的创新点及其应用前景和价值，同时分析了该系统存在的不足之处，对该方向今后的研究做了展望与设想。

第 2 章 地图创建与车辆定位

为了让驾驶员能够观察并选定合适的停车位,本方案在泊车过程中首先创建了基于概率占据栅格的环境地图。该环境地图的创建利用了激光雷达传感器,通过对传感器采集数据的处理,创建了车辆周围的环境地图,该地图能够比较真实地反映车辆周围的可通行区域、障碍物区域以及可以停泊的空间。地图的创建为系统的路径规划提供了环境信息,使得规划算法能够在泊车场景中搜索到可行的期望路径。为了实现车辆的精确定位,地图的创建也使得基于地图匹配的定位方法得以实现。

2.1 基于激光雷达的地图创建

2.1.1 激光雷达

本文地图创建所用传感器为 SICK LMS511 单线激光雷达^[26]。激光雷达是一种精度相对较高、方向性好而且基本不受环境中可见光变化影响的测距传感器,无论在室内还是室外环境下的移动机器人导航研究中都得到了广泛应用。激光测距雷达可以直接获取距离数据,为机器人的导航提供了便捷有效的环境描述。根据扫描结构的不同,激光测距雷达有二维和三维两种。它们大部分都是靠一个旋转的反射镜将激光发射出去并通过测量发射光发出和接收的时间差计算距离。SICK LMS511 雷达是一种二维雷达,它可以扫描某一区域,并根据区域内各个点与扫描仪的相对位置,返回其测量值,如图 2.1 所示。LMS511 的测量数据为极坐标形式,返回的是测量物体与扫描仪中心之间的距离和相对扫描仪中心的角度。



图 2.1 SICK LMS511 激光雷达

LMS511 雷达可以设置多种角度分辨率和扫描频率组合，它输出每束光束的测量距离的表达方式与所设置的角度分辨率和扫描频率有关，如表 2.1 所示。本文使用的角度分辨率与扫描频率的配置为表 2.1 第二行所示加粗的角度分辨率与扫描频率组合，故测量距离值应为 LMS 输出数据的两倍。

表 2.1 SICK LMS511 的配置及输出性能

扫描频率(Hz)	角度分辨率(度)	Scaling factor for data output	最大距离(m)
25	0.1667	×1	65
25	0.25	×2	80
35	0.25	×1	65
35	0.5	×2	80
50	0.3333	×1	65
50	0.5	×2	80
75	0.5	×1	65
75	1	×2	80
100	0.6667	×1	65
100	1	×2	80

2.1.2 栅格地图创建

对于环境的描述有多种方法，大致可以分为三类^[27]：基于特征的几何信息表示法、拓扑地图表示法以及栅格地图表示法（Occupancy Grid Map, OGM）。基于特征的几何信息表示法将机器人采集的环境感知信息提取为更为抽象的几何特征，例如线段或曲线，使用这些几何信息描述环境。拓扑地图表示法将环境表示为一张拓扑意义中的图(graph)，图中的节点对应于环境中的一个特征状态、地点，如果节点间存在直接连接的路径则相当于图中连接节点的弧。用拓扑地图表示的环境简单，存储和搜索空间小，匹配速度快，缺点在于对拓扑图的使用是建立在对拓扑节点的识别匹配基础上的，如当环境中存在两个很相似的地方时，拓扑图方法将很难确定这是否为同一点。栅格地图表示法是将整个环境分为若干相同大小的栅格，每个栅格关联一个取值在[0,1]之间的实数，用来表征栅格被障碍物占据的可能性。其优点是较完整的表示了环境信息，缺点是当地图尺寸较大

时计算量大。

由于本课题中环境的规模较小，地图复杂性不大，基于栅格地图^[28]易于创建、维护方便且能保留几乎全部的环境状态的特点，在本课题中对于泊车车辆周围环境的描述采用这种方法。

概率占据栅格地图表示方法，如图 2.2 所示，占据栅格地图将环境划分为 n 个具有固定大小的网格 $m_i (i=1,2,\dots,n)$ ，在本文中需要表达的环境尺寸规模较小，为了提高车辆定位及环境地图创建的精度，选择了较小的网格分辨率，每个网格的固定大小为 0.2m ，预设的栅格数目为 100×100 ，因此可以表达的环境的大小为 $20\text{m} \times 20\text{m}$ 。占据栅格地图既可以容纳定位信息中的噪声，同时也可以容纳传感器信号中的噪声。对于每个网格 m_i 的赋值表示了该网格被占据的可能性，其大小为闭区间 $[0,1]$ 上的任意浮点值， m_i 值越大，该网格被占据的可能性就越大。其中 0 表示网格完全没有被占据，1 表示该网格确认被占据。初始化时所有网格 m_i 的值均被设为 0.5，表示区域为未知状态。当有测量点落在网格 m_i 中时， m_i 被占据的概率就相应地增加。由于将环境切割成了小块的网格，以概率表示了网格被占据的可能性，概率占据栅格地图较完整地保留了环境的空间状态。

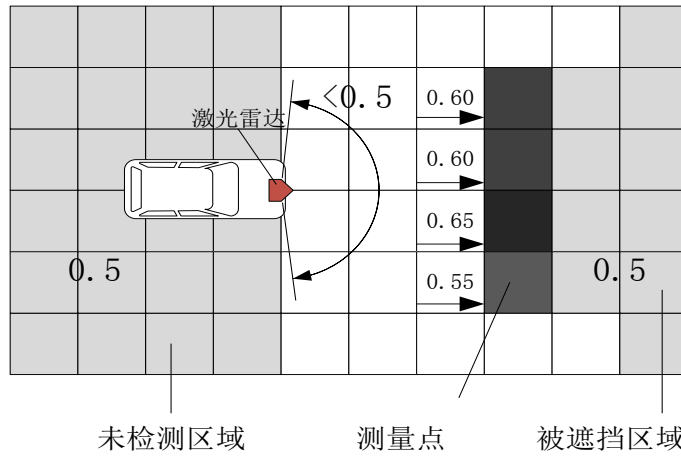


图 2.2 占据栅格地图

环境信息用栅格地图表达后，栅格节点间需要建立一定的连接关系才能保证能从起始点搜索到目标点的有效路径。以 $(0,0,0)$ 作为起始状态为例说明车辆在某一初始航向角下的基本运动单元的生成方法，参考文献[25]，本文中轨迹生成算法共生成了 8 条基本的运动单元，包括 4 条前向运动单元和 4 条后向运动单元，如图 2.3 所示。这里是对初始航向角为 0 度时生成的基本运动单元，如果对所有航向角 $\psi \in [0, 2\pi)$ 都离线生成这样的基本运动单元，则这些所有的基本运动单元便构成了车辆的状态空间，它表征了考虑车辆运动学约束的节点连接关系，文

献[25]中定义这种节点连接关系为 Lattice 连接。

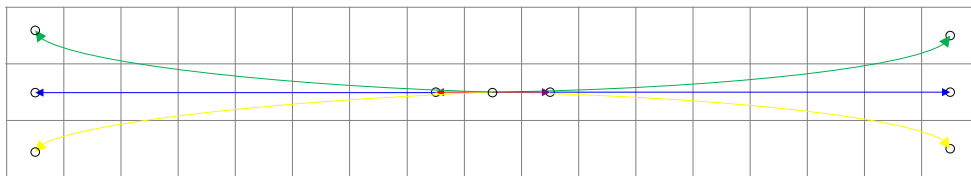


图 2.3 栅格连接关系

2.2 车辆定位方法

对于自动泊车的车辆来说,为了能够高效的导航即从环境中穿行并到达目的地,就必须找到一种方法,使它能够根据环境中已知的一些特征来判断出它自身的位置,也就是车辆的定位问题。智能车辆要想在未知环境中自主运动,一个基本问题就是确定自身在环境中的位置。无论是在结构化或非结构化环境中,机器人都必须了解自身与周围环境的位置关系才能正确选择和做出决策。目前,应用于机器人的定位方法有很多,这些方法也同样适用于自动泊车的实际车辆上,较为典型的有以下几种^[29]:

(1) 内部传感器定位

主要是借助于里程计、方向计等内部传感器通过累积测量的推算定位(Dead reckoning)^{[30][31][32]}来确定智能移动机器人相对于初始时刻的位置和方向,实现机器人的定位。该方法有一个严重的缺点,即误差具有累积性,所以过一段时间必须通过其他定位方式进行修正。

(2) 主动传感器定位

主要是借助超声波、激光测距仪等主动传感器定位。在结构化环境中,一般是在环境中预先建立电磁、红外等信标(Beacon),机器人通过检测信标来确定自身位置,而与具体的外部环境无关。激光测距仪用来建立机器人在环境中的绝对坐标,超声波传感器则是建立机器人在环境中的相对坐标。这种定位方式具有较高的定位精度和抗干扰能力。

(3) 视觉传感器定位

视觉定位是近年来发展起来的一种定位方法。其原理是通过摄像机获取的周围景物的图像,利用景物中一些自然的或者是人造的特征,通过图像处理方法得到周围环境模型来实现位置确定。这种定位方法累积误差小,但计算量很大,对外界条件(特别是光照条件)和路标有一定的要求。传感器视场的大小是这类定位

系统需要解决的一个关键问题。

(4) 激光扫描定位

激光扫描定位也是新近发展起来的一种先进的定位方法。其原理是利用旋转镜面机构向外发射并检测由路标反射回来的激光束,通过测量路标在参考坐标系中的几何关系,根据已知的路标关系地图和实测的各路标的方位角来计算传感器系统在参考坐标系中的位置和方向,实时给出传感器系统的精确位姿信息。这种定位方法定位精度高、抗干扰性好、无累积误差,而且定位周期短,适合于机器人的实时应用。

(5) 全球卫星定位

主要是借助于全球卫星定位系统 (Global Positioning System, GPS) ^{[33][34]},利用接收机对经过导航卫星发射的信号进行分析而实现机器人的定位。GPS 是美国国防部研制的卫星导航系统,但由于选择可用性技术 (Selective Availability, SA) 的引入和天气的影响,往往不能精确和稳定的定位。为了解决这个问题,可采用差分全球定位系统 (Differential Global Positioning System, DGPS) 来消除 SA 的影响, GPS 定位主要应用于室外作业的移动机器人系统。缺点是该方法的定位精度较低,在泊车这种狭小环境中并不适用,另外在地下停车场等 GPS 信号较弱或没有信号的场景时,该定位方法将会失效。

总结上述定位方法的原理以及其使用的场合和优缺点,本文将采用第四种方法,即基于激光雷达传感器的车辆定位方法,定位的同时创建泊车环境地图,即同步定位与地图创建 (Simultaneous Localization and Mapping, SLAM)。

2.3 SLAM 同步定位与地图创建

泊车过程中,需要获得车辆在局部环境中的位置信息,同步定位与地图创建 ^{[35][36]}被认为是提高定位准确性和地图精度的关键技术。在未知或非结构化环境移动过程中,根据位姿估计与传感器数据进行自身定位,同时是递增创建环境地图的过程。本节主要介绍课题中实现同步定位与地图创建坐标系统的建立、传感器测量模型以及机器人定位和地图的更新方法。

2.3.1 传感器测量模型

由于环境传感器同样有噪声,使得其测量值存在不确定性。因此需要对传感

器建立含噪声的模型。本研究中主要使用激光雷达传感器，其返回值为测量中心与某方向上与其距离最近的物体之间的距离。设激光雷达返回的观测值为 $z = (z_1, z_2, \dots, z_n)$ ，表示 n 个方向上激光雷达传感器返回的距离值。对于 $z_i (i = 1, 2, \dots, n)$ ，由于激光存在光斑扩散效应，且相邻激光束间存在夹角。以 SICK LMS291 为例，相邻激光束间最小夹角为 0.5° ，在距离激光发射原点 50m 时，相邻激光束间的距离已经大于 0.4m，激光雷达在这个距离上的检测结果不能完全表示在这个方向附近是否存在物体。因此，激光雷达测量的可信度应与测量距离相关，距离越远其可信度越低。在本课题中，简单考虑将激光最大最小检测范围内的概率设为 0.8。课题中采用的激光雷达测量模型如下。

如果 m_j 为激光束测量值落在栅格地图中的那个网格，则在单次测量下网格 m_j 被占据的概率如式 (2.1) 所示。

$$p(m_j, d_1) = \begin{cases} 0.4 & 0 \leq d_1 \leq d_{\min} \\ 0.8 & d_{\min} \leq d_1 \leq d_{\max} \\ 0.5 & d_1 > d_{\max} \end{cases} \quad (2.1)$$

其中 $d_{\max} = 40\text{m}$ ， $d_{\min} = 0.03\text{m}$ 。 m_j 为占据栅格地图中的网格， d_1 为激光束的测量值。

如果 m_i 为激光束路径上除去该测量落在栅格地图上的那个网格后的其他网格，则在该次测量下网格 m_i 被占据的概率如式 2.2 所示。

$$p(m_i, d_1) = \begin{cases} 0.4 & 0 \leq \rho < d_1 \\ 0.5 & \rho > d_1 \end{cases} \quad (2.2)$$

其中 d_1 为该次激光束的测量值， ρ 为网格 m_i 到激光雷达扫描中心的距离。

2.3.2 车辆的定位

在对移动机器人定位的研究中，国内外的研究者大多采用基于概率模型的定位方法^[37]，通过比较机器人位于采样点的可能性，获取可能性最高的采样点，用来估计机器人的位姿。目前较为通用的方法有基于卡尔曼滤波的定位方法^{[38][39][40]}，基于粒子滤波的定位方法以及基于直方图滤波的定位方法。基于卡尔曼滤波的定位方法在机器人定位中常用的做法是，将机器人与各环境特征点的距离作为系统状态量，机器人获取的对特征点的观测作为系统观测，利用卡尔曼滤波方法对观测值进行滤波，减小观测误差，利用机器人与特征点间的几何关系估

计机器人的位姿。当环境较为复杂，特征点数量增加时，卡尔曼滤波器维护的状态矩阵维数增加，计算复杂度增大。基于粒子滤波的定位方法是通过随机算法随机的散布一定数量的粒子，将粒子的位姿作为机器人位姿的一种估计，与环境模型匹配，匹配度越高的粒子权重越大，越接近机器人位姿真值，然后利用重采样技术重新划分权重大的粒子，舍弃权重小的粒子，使得粒子分布向机器人真值靠近。以此递归，粒子将越来越接近机器人位姿真值。

在本课题中，综合考虑环境的复杂度、计算的简便性等因素，采用的是基于直方图滤波的定位方法，同时，为了减小定位结果的抖动，利用扩展卡尔曼滤波器(EKF)对定位结果进行滤波，滤波后的结果作为最终的定位结果。

试验在 P3-AT 机器人上进行，该机器人能够提供自身大概的位置和航向信息。为了减小里程计的累积误差，通过前后时刻机器人运动和航向偏差加上前一时时刻获得的机器人后验定位位姿的方法获得机器人当前时刻的理想位置、航向，同时利用该偏差可以估算当前时刻的机器人速度，如式（2.3）所示。

$$\begin{cases} \dot{x} = \Delta x / t \\ \dot{y} = \Delta y / t \\ \dot{\theta} = \Delta \theta / t \end{cases} \quad (2.3)$$

上式中， $\Delta x, \Delta y, \Delta \theta$ 为里程计前后帧的定位偏差， t 为前后帧的时间偏差。

机器人在平面二维空间的运动为连续运动，其在状态空间的位置不可能突变，由于机器人运动速度、角速度已知，可以估计一个机器人在状态空间内的偏差范围，这里称这个偏差范围为状态空间偏差。则机器人可能的位置为理想位置加状态空间偏差。将状态空间偏差离散化，如图 2.4 所示，得到机器人状态空间偏差表，加上机器人理想位置，则可以得到机器人下一时刻的先验位姿估计。

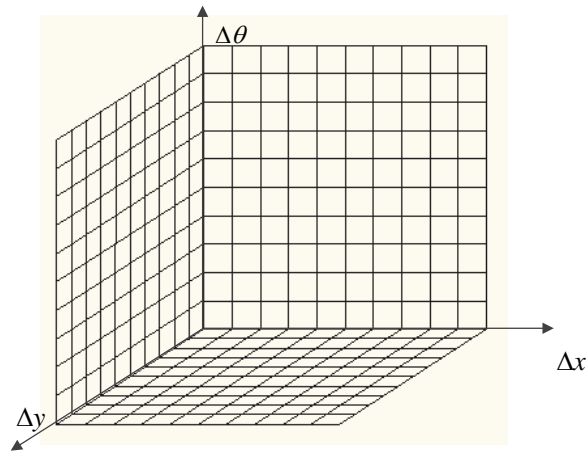


图 2.4 离散化的状态空间偏差

遍历状态空间偏差,将当前的状态空间偏差值加上机器人理想位置得到一个当前的预测状态位置,结合当前时刻激光雷达传感器测量的环境距离数据与之前时刻建立的环境地图,进行环境地图的扫描匹配,利用扫描匹配的评价函数,得到每一个可能位姿的评价函数得分,分数越高的说明与地图的匹配性越好,越接近机器人真实的位姿。扫描匹配评价函数如下。

假设激光雷达的一个测量在传感器坐标下为 (x_s, y_s) ,表明在传感器坐标下坐标为 (x_s, y_s) 的位置可能被障碍物占据。为计算机器人在位置 X_t 和与地图 M_{t-1} 的匹配率,课题中采用了一种投票的机制。首先,从机器人所在位置 X_t 出发,独立一束激光的测量 z_t^k 被投射到全局地图 M_{t-1} 中相应的位置上,对于投射的每一束激光的终点对应的栅格称为 g_t^k 。如果这个栅格被占据,与该栅格被占据概率值成比例的一个值会加入到投票中。然后最终投票出来的数值决定了机器人在该位置下的匹配率。

$$P_{score} = \sum Z_t^k p(M_{t-1}^{g_t^k}) / 10 \quad (2.4)$$

考虑到距离激光较近的测量的误差对于定位结果及地图创建结果影响较大,特别对于机器人航向的确定,因此在扫描匹配的评价函数中加入了测量的距离,认为越远的测量即存在误差对结果影响小,越可靠。其中, z_t^k 表示当前时刻激光雷达第 k 个测量的距离, $M_{t-1}^{g_t^k}$ 表示第 k 束激光终点对应在全局地图中的栅格, $P(M_{t-1}^{g_t^k})$ 表明该栅格被占据的概率。

经过投票可以得到一个评价函数取得最大值的机器人位姿,评价函数取得最大值表明在该位姿下激光雷达当前的观测与由历史信息建立的环境地图的匹配性最好,该位姿为机器人的最可能位姿。

由于前后帧时间间隔较短(200ms),且机器人是在人工使用手柄控制下行驶,因此其运动状态可以认为满足线性条件,可以利用扩展卡尔曼滤波对定位结果进行滤波。以机器人的位置航向及速度作为卡尔曼滤波器的状态矩阵,即 $X = [x \ y \ \theta \ \dot{x} \ \dot{y} \ \dot{\theta}]^T$ 故从 $t-1$ 时刻到 t 时刻的状态转移可以用下式描述

$$X_t = AX_{t-1} + w \quad (2.5)$$

其中 A 为状态转移矩阵, w 为过程噪声。

$$A = \begin{bmatrix} 1 & 0 & 0 & t & 0 & 0 \\ 0 & 1 & 0 & 0 & t & 0 \\ 0 & 0 & 1 & 0 & 0 & t \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

其中, t 为前后帧之间的时间差。观测为直方图滤波得到的机器人的当前姿态及速度, 即 $Z_t = [x' \ y' \ \theta' \ \dot{x}' \ \dot{y}' \ \dot{\theta}']^T$, 测量方程为:

$$X_t = EZ_t + u \quad (2.6)$$

上式中 E 为测量矩阵, 它是一个 6 阶单位矩阵, u 为测量噪声。

2.3.3 地图的更新

假设栅格与栅格间的占据状态独立, 栅格地图的创建(更新)的目标即基于 1 至 t 时刻的测量 $z_{1:t} = \{z_1, \dots, z_t\}$, 为栅格地图的每一个栅格 m_i 估计后验条件概率 $p(m_i | z_1, \dots, z_t)$ 。假设对于每个网格 m_i , 其基于 $1:t-1$ 时刻所有测量的概率值为 $p(m_i | z_1, \dots, z_{t-1})$, 基于 t 时刻测量得到的网格概率为 $p(m_i | z_t)$ 。

根据贝叶斯公式, 则基于 $1:t$ 时刻所有测量的概率 $p(m_i | z_1, \dots, z_t)$ 为

$$p(m_i | z_1, \dots, z_t) = \frac{p(z_t | z_1, \dots, z_{t-1}, m_i) \cdot p(m_i | z_1, \dots, z_{t-1})}{p(z_t | z_1, \dots, z_{t-1})} \quad (2.7)$$

假设过程符合马尔科夫假设, 对于网格 m_i , 其 t 时刻的测量与 $1:t-1$ 时刻的测量无关, 因此

$$p(z_t | z_1, \dots, z_{t-1}, m_i) = p(z_t | m_i) \quad (2.8)$$

由贝叶斯理论 $p(A|B) = \frac{p(B|A) \cdot p(A)}{p(B)}$, 式(2.7)可转化为

$$p(m_i | z_1, \dots, z_t) = \frac{p(m_i | z_t) \cdot p(z_t) \cdot p(m_i | z_1, \dots, z_{t-1})}{p(m_i) \cdot p(z_t | z_1, \dots, z_{t-1})} \quad (2.9)$$

式(2.9)表示了被占据网格的概率, 则未被占据网格的概率为

$$p(\bar{m}_i | z_1, \dots, z_t) = \frac{p(\bar{m}_i | z_t) \cdot p(z_t) \cdot p(\bar{m}_i | z_1, \dots, z_{t-1})}{p(\bar{m}_i) \cdot p(z_t | z_1, \dots, z_{t-1})} \quad (2.10)$$

将式(2.9)比上式(2.10), 可以得到

$$\frac{p(m_i | z_1, \dots, z_t)}{p(\bar{m}_i | z_1, \dots, z_t)} = \frac{p(m_i | z_t)}{p(\bar{m}_i | z_t)} \cdot \frac{p(\bar{m}_i)}{p(m_i)} \cdot \frac{p(m_i | z_1, \dots, z_{t-1})}{p(\bar{m}_i | z_1, \dots, z_{t-1})} \quad (2.11)$$

由 $p(\bar{A})=1-p(A)$ 和 $p(\bar{A}|B)=1-p(A|B)$ ，上式(2.11)可变化为

$$\frac{p(m_i | z_1, \dots, z_t)}{1-p(m_i | z_1, \dots, z_t)} = \frac{p(m_i | z_t)}{1-p(m_i | z_t)} \cdot \frac{1-p(m_i)}{p(m_i)} \cdot \frac{p(m_i | z_1, \dots, z_{t-1})}{1-p(m_i | z_1, \dots, z_{t-1})} \quad (2.12)$$

其中 $p(m_i)$ 表示没有被测量到的网格，仍为初始化值 0.5，所以上式(2.12)进一步化简为

$$\frac{p(m_i | z_1, \dots, z_t)}{1-p(m_i | z_1, \dots, z_t)} = \frac{p(m_i | z_t)}{1-p(m_i | z_t)} \cdot \frac{p(m_i | z_1, \dots, z_{t-1})}{1-p(m_i | z_1, \dots, z_{t-1})} \quad (2.13)$$

令 $S = \frac{p(m_i | z_t)}{1-p(m_i | z_t)} \cdot \frac{p(m_i | z_1, \dots, z_{t-1})}{1-p(m_i | z_1, \dots, z_{t-1})}$ ，则每个网格 m_i 在加入 t 时刻环境观测后的概率为

$$p(m_i | z_1, \dots, z_t) = \frac{S}{1+S} \quad (2.14)$$

通过式 (2.14)，将 t 时刻的测量与之前时刻的测量结合起来，得到更新过后的地图。

2.4 SLAM 定位结果验证

为了验证 SLAM 算法的可行性，利用实验室现有的 P3-AT 机器人进行了算法验证。实验分为两个部分：实验一是事先测量出一定长路段，控制机器人在该路段行驶，得到机器人里程计定位及本课题的 SLAM 程序给出的定位结果，比较里程计和 SLAM 的定位误差；实验二是控制 P3-AT 机器人绕花坛行走一圈，绘制环境地图并分析里程计和 SLAM 的定位结果。



图 2.5 实验一

实验一场景如图 2.5 (a) 所示，定长路段长为 4.78m，手柄控制机器人尽量保持同一航向行驶，机器人从起点运动到终点。完成该路段的结果如图 2.5 (b)

所示。

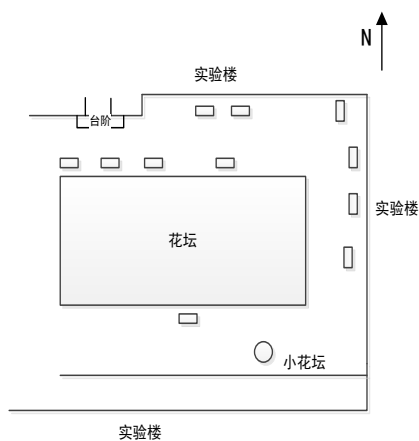
机器人里程计给出的结果为 $x = 5.371\text{m}$, $y = -0.017\text{m}$, $\theta = -3.25^\circ$, SLAM 的定位结果为 $x = 4.742\text{m}$, $y = -0.026\text{m}$, $\theta = -3.20^\circ$ 。全局坐标系为以机器人起始航向为 x 轴正向的笛卡尔坐标系。机器人到达终点后测量机器人 x, y 及航向的变化, 得到机器人理想的定位结果为 $x = 4.78\text{m}$, $y = -0.03\text{m}$, $\theta = 0^\circ$ 。由此得到里程计和 SLAM 的定位偏差如表 2.2 所示。

表 2.2 实验一定位结果比较

类别 内容	理想定位结果	里程计		SLAM	
		定位结果	定位偏差	定位结果	定位偏差
X(m)	4.78	5.371	0.591	4.742	-0.038
Y(m)	-0.03	-0.017	-0.013	-0.026	-0.004
航向(度)	-3	-3.25	0.25	-3.20	0.20

由表 2.2 的数据可知, SLAM 的定位偏差远远小于里程计的定位偏差, SLAM 的定位结果较里程计的定位结果有较大的改善。

实验二中, 利用手柄控制机器人顺时针绕场地中间花坛行驶一圈回到原点, 在机器人行驶过程中获得机器人的实时位姿, 并且建立环境的栅格地图。作为实验场地的北京理工大学车辆楼下场地是一片约 $70\text{m} \times 50\text{m}$ 的空地, 中间是一个花坛, 道路沿着花坛环形分布, 在道路两边停靠有车。实验场地如图 2.6 所示。



(a)实验场地示意图



(b)Google 地图上实验场地

图 2.6 实验二场地

机器人回到原点后建立的环境地图如图 2.7 所示, 由图可以看出, 建立的栅格地图较完整的描述了环境中的障碍物信息, 以栅格概率的形式给出了环境中的可通行区域及被占据区域, 为机器人导航提供了便利。全局坐标系为以机器人起

始航向为 x 轴正向的笛卡尔坐标系。由于控制的原因，机器人绕花坛行驶一圈后的终点并未与起点完全重合，经测量在全局坐标系下，机器人的定位结果应该为 $x = -0.02\text{m}, y = -0.01\text{m}, \theta = -1^\circ$ 。而当机器人完成一圈回到起点时，机器人里程计给出的结果为 $x = -0.729\text{m}, y = -1.418\text{m}, \theta = 2.896^\circ$ ，SLAM 给出的定位结果为 $x = -0.113\text{m}, y = -0.819\text{m}, \theta = -1.235^\circ$ 。从图中可以看出，在起点处前后两次扫描获得的地图重合度较高，在 x 方向约有 0.1m 的误差，在 y 方向误差稍大为 0.8m 。由此可知，SLAM 的定位结果与机器人里程计定位结果比较具有相对较高的精确度。表 2.3 为具体实验二定位结果比较。

表 2.3 实验二定位结果比较

类别 内容	理想定位结果	里程计		SLAM	
		定位结果	定位偏差	定位结果	定位偏差
X(m)	-0.02	-0.729	-0.709	-0.113	-0.093
Y(m)	-0.01	-1.418	-1.408	-0.819	-0.809
航向(度)	-1	-2.896	-1.896	-1.235	-0.235



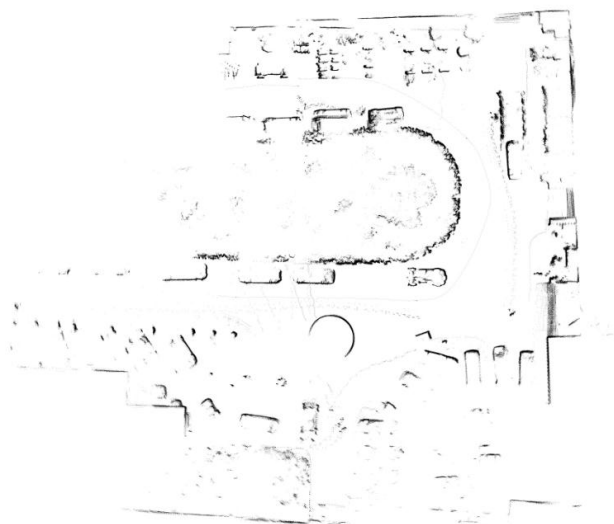
图 2.7 实验二建立的环境栅格地图

图 2.8 (a)为根据原始里程计数据以及同时刻激光测量数据重建的点云地图，其中环境中路边停放的车辆、花坛边缘等存在较多重影。图 2.8(b)则是根据 SLAM 定位结果以及同时刻激光测量数据重建的点云地图，在该点云地图中可以清楚看出路边停放的车辆、建筑物外墙、花坛边缘轮廓等。比较图 2.8(a)(b)两张点云地图，可以明显看出，依据原始里程计数据重建的点云地图中同一物体分布比较杂乱、重合度较低，依据 SLAM 定位结果重建的点云地图中物体分布较统

一、相同物体的重合度较高。由此，也可以验证 SLAM 的定位结果相较机器人里程计定位结果具有较高精度，创建的环境栅格地图具有较高的确信度。



(a) 基于原始里程计的点云地图重建



(b) 基于 SLAM 定位的点云地图重建

图 2.8 点云地图重建

2.5 本章小结

车辆在泊车过程中，由于泊车位的空间狭小，必须要求车辆能够精确地跟踪所规划出来的期望路径，否则可能导致刮擦碰撞等交通事故。为实现精确的路径跟踪，车辆需获得自身的精确位置，传统的方法多采用传感器信息根据车辆与障碍物的相对位置来得到，这种方法得到的位置精度不高。因此，可以在泊车中引

入精度更高的定位方法，即同步定位与地图创建，该方法在创建地图的同时能够把传感器信息与地图信息进行匹配得到车辆的最可能位置，同时利用获得的精确位置进行地图更新。实验证明，该方法对车辆的定位有较高的精度，满足泊车工况对定位精度的要求。

第3章 全自主泊车路径规划及算法研究

第二章主要介绍了泊车环境地图的创建和车辆定位，地图创建完毕，需要基于此环境模型为待停泊的车辆规划出一条从起始位置到停车位的行车路径^[41]，路径的搜索主要应用了全局路径规划方法。本章介绍了路径规划^[42]的意义和目的，分析了其在全自主泊车系统中的作用；然后重点介绍了两种全局规划算法，Dijkstra 算法^[43]和 A*规划算法^[44]，分别对这两种算法的原理做了详细的说明，并分析对比了两种算法的优劣，最终选取 A*方法作为本文的全局路径规划算法。最后通过仿真实验，在泊车环境地图中对该方法做了验证。

3.1 路径规划概述

路径规划（Path Planning）是指在具有障碍物的环境中，按照一定的评价标准，比如路径长度最短或能量消耗最少等，寻找一条从起始状态到目标状态的无碰撞路径^[45]。从路径规划的定义可以看出，路径规划与障碍物环境信息息息相关。因此，基于环境地图的路径搜索方法是研究中的一个关键问题。

3.2 全局路径规划

3.2.1 Dijkstra 算法

Dijkstra 算法是一种基于贪心策略的最短路径算法，它可以找出从起点到其他各节点间的最短路径。主要思想是首先从源点求出长度最短的一条路径，然后通过对路径长度迭代得到从源点到其他各节点的最短路径。

算法的具体做法是：设集合 Q 存放已经求出的最短路径的终点，集合 T 存放所有未标记的节点。初始状态时，集合 Q 中只有一个源点 s ，以后每求得的一条最短路径 (s, \dots, V_k) ，就将 V_k 加入到集合 Q 中，直到 T 中全部节点都加入 Q 中为止。无人驾驶车辆的全局路径规划，只需要计算从源点到终点的一条最短路径，当发现最短路径的终点为目标路径的终点时算法即终止。

设带权有向图 $G=(V,E)$ ，其中 V 是包含 n 个节点的节点集， E 是包含 m 条弧的弧集， (v,w) 是 E 中从 v 到 w 的弧， $Cost(v,w)$ 是弧 (v,w) 的非负权值，如果从 v 到 w 不通，则 $Cost[v,w]=\infty$ 。每个分量 $Dist[i]$ 表示已找到的从起始点 V_0 到每个终

点 V_i 的最小权值，则该向量的初始值为： $Dist[i] = Cost[s, i]$ ， $V_i \in V$ 。

设 s 为 V 中的节点， t 为 V 中可由 s 到达的节点，则求解从 s 到 t 的弧权值和最短路径搜索过程描述如下：

(1) 对每个节点 v 分配三个信息， $k(v)$ ， $Dist(v)$ 和 $p(v)$ ，其中 $k(v)$ 是一个布尔型变量，表明源点 s 的最短路径是否已经求出； $Dist(v)$ 是从 s 到当前节点 v 的最小路径的权值； $p(v)$ 是 v 的后向节点指针，并分别初始化为

$$Dist(v) = \begin{cases} 0(v=s) \\ \infty(v \neq s) \end{cases}; k(v) = \text{false}; p(v) = \{ \}; \quad (3.1)$$

(2) 扫描 $k(v)=\text{false}$ 的节点，从中选择一个具有最小路径长度的节点 v ，并令

$$k(v)=\text{true}, Dist(v) = \min \{ Dist(v_i) | k(v_i)=\text{false}, v_i \in V \} \quad (3.2)$$

(3) 检测每一个 $k(w)=\text{false}$ 且邻接于节点 v 的节点 w ，若满足

$$Dist(w) > Dist(v) + Cost(v, w) \quad (3.3)$$

则令

$$Dist(w) = Dist(v) + Cost(v, w) \quad (3.4)$$

(4) 重复执行扫描操作 2) 和 3)，直到 $k(t)=\text{true}$ 为止

(5) 由 t 开始遍历后向节点指针 p 直至源点 s ，即获得最短路径解。

$$P_{st} = \{v_0 = s, v_1, \dots, v_k = t\} \quad (3.5)$$

其中 $v_i = p(v_{i+1})$ ($i=0, 1, \dots, k-1$)。算法的流程图如图 3.1 所示。

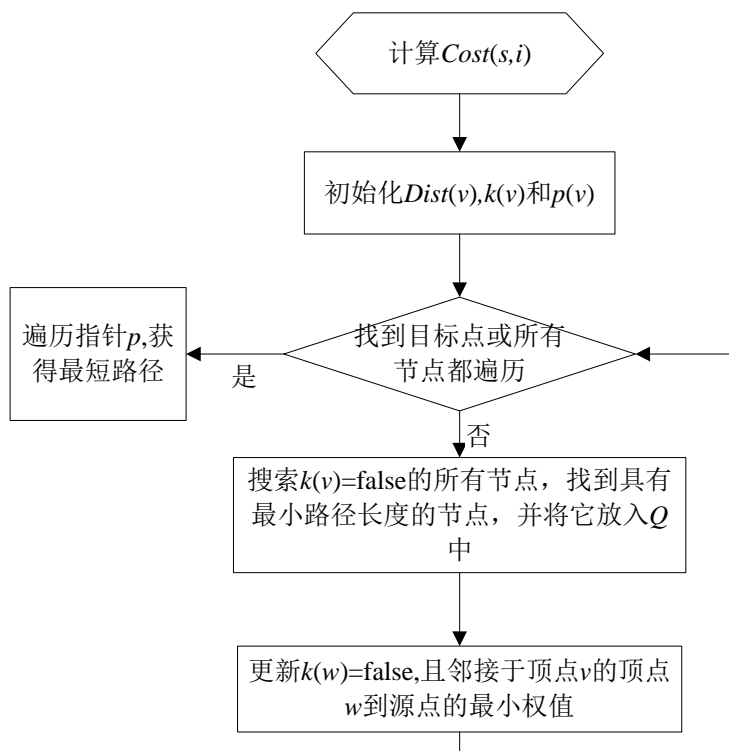


图 3.1 Dijkstra 算法流程图

评价一个算法的效率比较客观的方法是时间复杂度,它与具体的计算机软硬件因素无关,而只依赖于问题的规模 n 。如果图中有 n 个节点,那么 Dijkstra 算法的核心代码用 C 语言实现如表 3.1 所示:

表 3.1 Dijkstra 算法的核心代码实现

```

Dist[v]=0;k[v]=false; //初始化, v 是源点
1): for(int i=0;i<n; i++) { //主循环, 每次计算一条最短路径
    double min=MAXNUM; int u=v;
2): for(int j=0;j<n;j++) { //选择当前在集合 T 中的最短路径的节点 u
    if(!k[j]&&Dist[j]<min) {u=j; min=Dist[j];}
    k[u]=true;}
3): for(int w=0;w<n; w++) { //更新当前最短路径长度
    if(!k[w]&&Dist[u]+Cost[u][w]<Dist[w])
        Dist[w]=Dist[u]+c[u][w];
    }
}
  
```

由上述代码可知, 基于邻接矩阵的 Dijkstra 算法包含了二重 for 循环: 主循

环 1)和子循环 2)、3)，算法的时间复杂度为 $O(n^2)$ ， n 为节点数。该算法存在以下两点不足：

(1) 从未标记节点 T 中选择一个节点 k 作为转接节点后，需要扫描集合 T 中的所有节点 i ，并更新其 $Dist[i]$ ，但集合 T 中包含大量与转接点 k 不直接相连的节点。

(2) 在未标记集和 T 中选择一个 $Dist$ 值最小的节点作为下一个转接节点时，需要扫描集合 T 中的所有节点，但下一个标记节点往往是和标记节点集合 Q 中的节点直接相连的。基于上述两点不足，提出只对最短路径上节点的邻节点做处理而不涉及其他节点的优化方法。

3.2.2 A*算法

目前在路径优化领域最流行的启发式搜索算法是 A*算法，它对实现道路网络的最佳优先搜索极为有效。该算法在选择下一个被检查节点时，对当前节点距离终点的长度进行估计，评价其处于最优路径上的可能性的量度，从而先搜索可能性较大的节点，提高搜索效率。

A*算法的核心部分是它对每个道路节点均设计了一个估价函数：

$$f(s) = g(s) + h(s) \quad (3.6)$$

$f(s)$ 表示从起始节点经过节点 s 到目标节点的估计长度； $g(s)$ 表示从起始节点到当前节点的路径长度，它是已知的，大小为：

$$\sum_{i=start}^{k-1} cost(s_i, s_{i+1}) (k \leq goal) \quad (3.7)$$

$h(s)$ 为启发函数，是当前节点到目标节点的估计值，A*算法一定能搜索到最优路径的前提条件是：

$$h(s) \leq cost^*(s, s_{goal}) \quad (3.8)$$

$cost^*(s, s_{goal})$ 为当前节点到目标节点的最优距离，满足式(3.8)的 $h(s)$ 值越大，扩展的节点越少。为了保证搜索路径的最优性，通常选择启发函数 $h(s)$ 选择为曼哈顿距离、对角线距离或者欧基里德距离作为启发函数。对于给定的两个位置坐标 (x_i, y_i) 、 (x_j, y_j) ，它们的曼哈顿距离 d_m 、对角线距离 d_d 和欧基里德距离 d_e 分别为：

$$d_m = |x_i - x_j| + |y_i - y_j| \quad (3.9)$$

$$d_d = \max(|x_i - x_j|, |y_i - y_j|) \quad (3.10)$$

$$d_e = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (3.11)$$

A*算法用 *OPEN* 和 *CLOSED* 两个集合来管理道路节点，*OPEN* 存放扩展过的道路节点的子节点，它们属于待扩展节点；*CLOSED* 存放扩展过的节点。算法伪代码如表 3.1 所示，初始条件下，*OPEN* 只存放 s_{start} 节点且：

$$f(s_{\text{start}}) = g(s_{\text{start}}) + h(s_{\text{start}}) \quad (3.12)$$

$$g(s_{\text{start}}) = 0 \quad (3.13)$$

CLOSED 为空，另外除 s_{start} 所有节点的 g 值都初始化为无穷大。算法搜索时，每次从 *OPEN* 中选择 $f(s)$ 值最小的节点 s 进行扩展，节点 s 扩展到的子节点存放于 *OPEN*，节点 s 扩展完成后，会从 *OPEN* 被移到 *CLOSED*。如此循环，直到扩展到目标节点或者 *OPEN* 为空时，算法才终止。如果 *OPEN* 为空，则表明没有可行路径，规划失败。如果存在可行路径，A*算法一定能搜索到路径。

表 3.1 A*算法伪代码

The pseudocode of A* algorithm:

(1) $g(s_{\text{start}}) \leftarrow 0$ and g -values of the rest of the states are set to ∞ .

(2) $OPEN \leftarrow \{s_{\text{start}}\}$, $CLOSED \leftarrow \emptyset$.

1 **ComputePath()**

2 **while** (s_{goal} is not expanded)

3 remove s with the smallest $f(s)$ from *OPEN*;

4 for each successor s' of s

5 **if** $g(s') > g(s) + c(s, s')$

6 $g(s') \leftarrow g(s) + c(s, s')$;

7 insert/update s' in *OPEN* with $f(s') \leftarrow g(s') + h(s')$;

图 3.2 是 A*算法搜索流程图，流程图体现了表伪代码内容，当前节点是 BEST，为 *OPEN* 中 f 值最小的节点，SUC 为 BEST 的子节点，OLD 为 *OPEN* 中已经计算过 f 值的节点。算法搜索到目标点，则搜索成功，否则如果 *OPEN* 为空，则搜索失败。

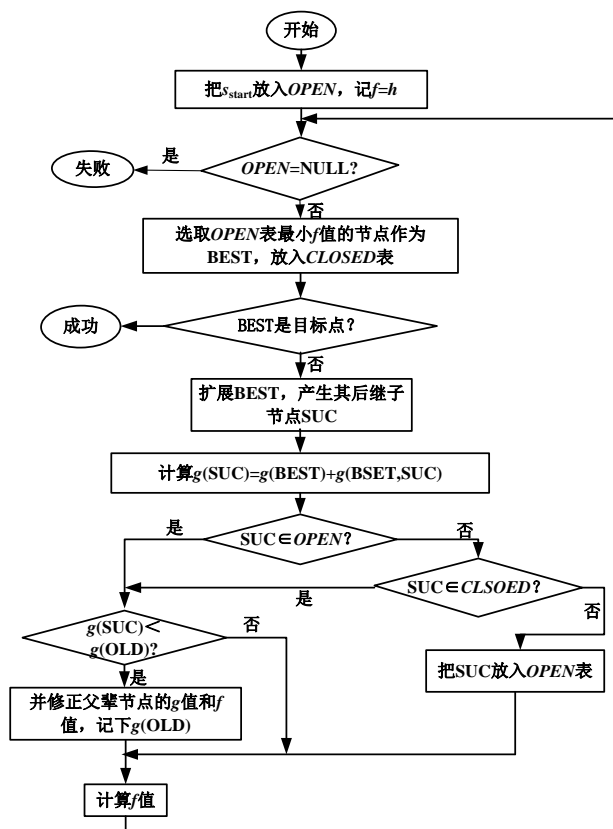


图 3.2 A*算法流程图

图 3.3 为 A*算法在拓扑地图内的搜索示意图, 起始节点为 S , 目标节点为 G , 其它节点均相当于道路节点, 节点与节点间的连线近似于道路。每个节点均有一个启发函数的值 $h(s)$, 并且 $h(s)$ 函数值满足式(3.8), 因此算法搜索到的路径一定具有最优性。

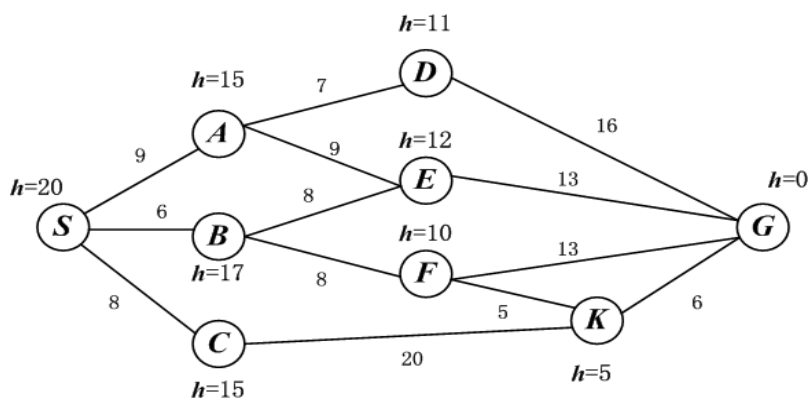


图 3.3 A*搜索示意图

表 3.2 为 A*算法搜索过程中, $OPEN$ 和 $CLOSED$ 分别表示各集合所存放的节点, $S(20)$ 表示节点 S 的 f 值为 20, 算法每次都会从 $OPEN$ 中选择 f 值最小的节点进行扩展, 到第 7 次扩展时, 算法搜索到最优路径, $S(20) \rightarrow B(23) \rightarrow F(24) \rightarrow K(24)$

$G(24)$ ，如图 3.4 所示的加粗线条。

表 3.2 A*算法搜索过程

	OPEN 集合	CLOSED 集合
1	$S(20)$	\emptyset
2	$A(24) B(23) C(23)$	$S(20)$
3	$K(33) A(24) B(23)$	$S(20) C(23)$
4	$K(33) E(26) A(24) F(24)$	$S(20) C(23) B(23)$
5	$E(26) A(24) G(24) K(24)$	$S(20) C(23) B(23) F(24)$
6	$E(26) A(24) G(24)$	$S(20) C(23) B(23) F(24) K(24)$
7	$E(26) A(24)$	$S(20) C(23) B(23) F(24) K(24) G(25)$
	最优路径:	$S(20) B(23) F(24) K(24) G(25)$

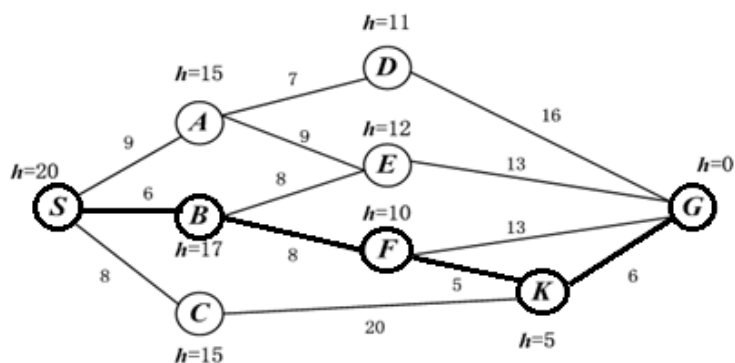


图 3.4 最优路径

值得说明的是，对于 A*算法有如下推论：

推论 1：如果启发函数满足式(3.8)，则所有节点不会被重复扩展。

如果函数满足条件(3.8)，则 A*扩展到节点 s 后，就已经找到了到达 s 的最短路径。因此找到某节点后，不会再出现比它的 $f(s)$ 值更小的节点了，由表可以看出没有节点被重复扩展。

推论 2：如果启发函数 $h(s)$ 满足式(3.8)，则 $h(s)$ 值越大，搜索到最优路径扩展的节点数越少。

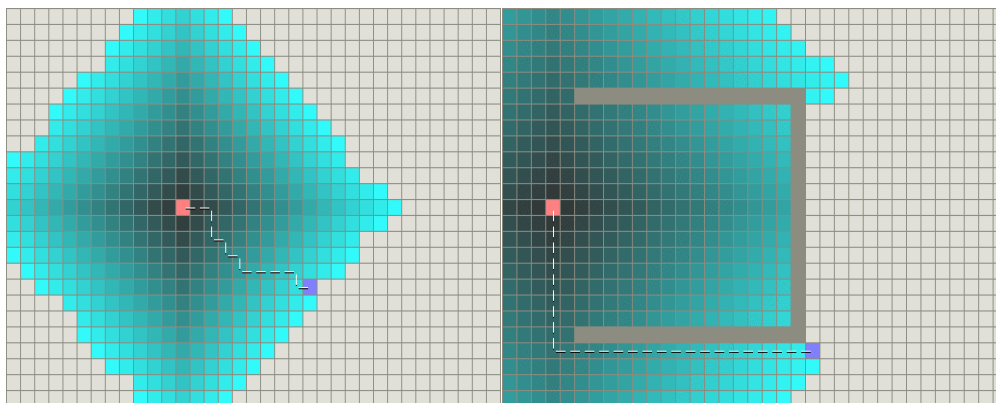
如果有两个启发函数 $h_1(s)$ 与 $h_2(s)$ 都满足启发函数(3.8)，如果 $h_1(s) > h_2(s)$ ，则启发函数为 $h_1(s)$ 的 A*算法扩展过的所有节点都会被启发函数为 $h_2(s)$ 的 A*算法所扩展。因此，启发函数 $h(s)$ 的选择关系到 A*算法的效率。

3.2.3 算法比较

在实际搜索过程中希望能将搜索的范围限制在源点和目标点连线的附近，减少与最短路径节点无关的计算，即算法的收敛性要好。通过启发式搜索，利用估

价函数，在每一步搜索过程中寻找估价函数值最高的节点作为下一个搜索节点，从而缩小了搜索范围，达到提高算法收敛性和搜索效率的目的。

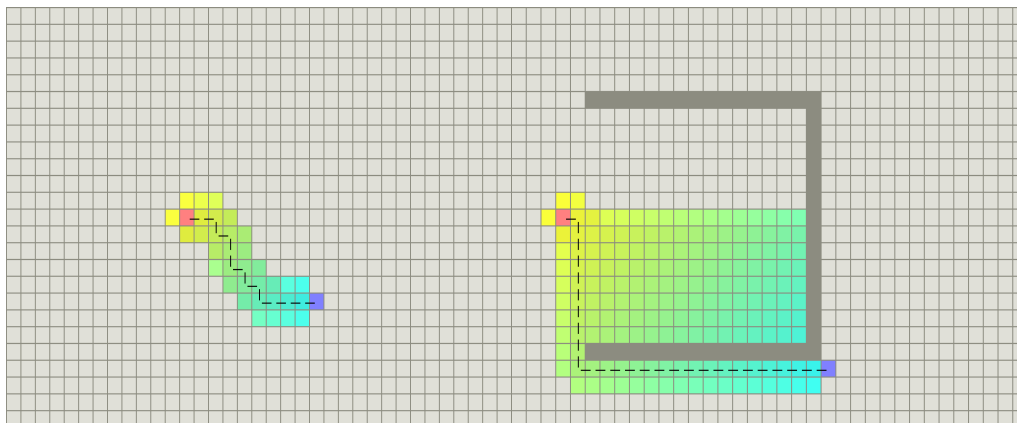
设计二维网格，以粉红色表示起点，紫色表示终点，蓝色表示搜索过的区域，颜色深浅表示权值的大小。Dijkstra 算法在无障碍物和有障碍物的情况下路径规划效果如图 3.5 所示。



(a) 无障碍物时 Dijkstra 算法搜索 (b) 有障碍物时 Dijkstra 算法搜索

图 3.5 Dijkstra 算法搜索效果

针对相同的场景，使用 A*算法进行路径规划，其效果如图 3.6 所示。由图 3.5 和图 3.6 可知，A*启发式路径规划算法的比 Dijkstra 算法搜索的节点较少，效率更高，因此在本课题的路径规划中选用此算法。



(a) 无障碍物时 A*算法搜索 (b) 有障碍物时 A*算法搜索

图 3.6 A*算法搜索效果

3.3 算法验证

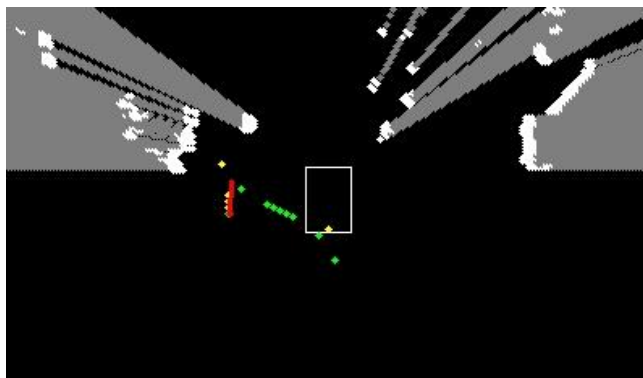
为了验证算法的可行性，分别设计了不同泊车场景，在场景中设置障碍物，

在可通行区域确定泊车位置后，测试系统能否规划出一条从起始点到终点无碰撞的路径。

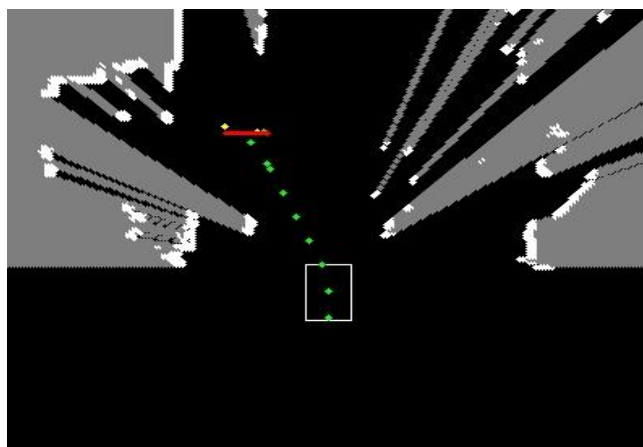
3.3.1 仿真分析

试验场景设置为三种，分别为平行泊车、前方垂直泊车以及后方垂直泊车。这三种方式是城市中最常见的泊车类型，泊车场景在北京理工大学校园内路边停车位中采集，真实的模拟实际泊车环境。

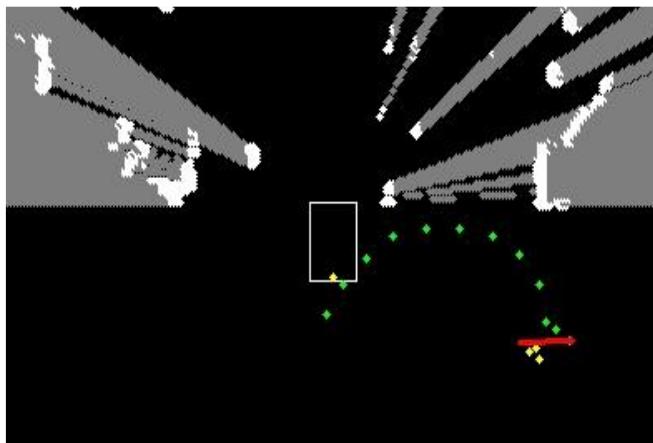
试验结果如图 3.7 所示，白色边框代表待泊车辆，其所在位置为车辆的起始位置。地图中黑色区域代表可通行区，白色点状或区域代表障碍物。在可通行区域确定车辆最终停泊的位置，具体选定步骤为在显示的地图中鼠标点击第一点确定为车辆中心的位置，同时点击第二个点，前后两点的连线即图中红色直线确定为车辆的航向，此时车辆最终位姿已经确定，系统启动规划，三种场景下的路径规划结果如图 3.7 所示。



(a) 平行泊车



(b) 前方垂直泊车



(c) 后方垂直泊车

图 3.7 路径规划结果

3.3.2 实验结果

规划过程均在较短的时间内完成,由上述结果分析该算法的可行性。图 3.7(a)为平行泊车规划结果,泊车位在车辆的左侧,车头最终朝向选为与初始状态一致,系统将规划出泊车期望路径,由图中绿色和黄色的点组成,绿色点表示车辆在该点之后的一段路径中为正向行驶,黄色的点则表示在该点之后将倒车。可见车辆先倒车行驶,然后直行到指定泊车位,最后车辆倒车调整姿态使得车头方向与预选的一致。图 3.7(b)为前方垂直泊车,车辆首先进入到指定位置附近,然后倒车调整车头朝向,满足预设位姿。图 3.7(c)为后方垂直泊车,车辆经过多次的位姿调整,最终能搜索到一条可行的路径。证明了本文所用算法的可行性。

3.4 本章小结

路径规划的方法有很多,包括上面提到的Dijkstra算法、A*算法,现有的泊车应用中路径规划方法多采用直线与圆弧拼接的方式,这种规划方法对于任意终点位置及任意姿态的泊车并不适用,系统不能规划出合适的路径。实验证明,采用上述提到的A*路径规划方法能够结合障碍物的信息得到可行的泊车路径,试验验证了该算法的可行性和快速性。

第4章 全自主泊车路径跟踪算法研究

本章主要介绍了自动泊车系统中用到的路径跟踪算法。首先介绍了国内外无人驾驶车辆常用的路径跟踪算法，并对各算法的优缺点以及适用场合做了对比；然后根据该系统适用的乘用车特点，建立了车辆运动学模型，提出了适用于低速、狭小空间内泊车场景下的前馈-反馈跟踪算法，详细说明了该算法的原理，以及算法流程和输入输出量等；最后为验证算法的可行性，进行了仿真和实车实验，分别让车辆跟踪直线和曲线，测试跟踪效果。

4.1 路径跟踪算法概述

路径跟踪控制算法是指无人驾驶车辆沿着预定的参考路径，安全、稳定、准确行驶的控制方法^[46]，其性能直接影响无人驾驶车辆自主行驶的执行能力。

早期的路径跟踪方法，如几何路径规划法^[47]、旋量法^[48]、滚动路径法^[49]等，适用于室内移动的机器人，但由于无人驾驶车辆的约束是非完整约束，且车辆体积尺寸比较大，受到如最小转弯半径、最大角速度等动力学条件的约束，对于无人驾驶车辆而言是不适用的。路径跟踪算法的本质是消除无人驾驶车辆在行驶过程中产生的跟踪偏差。跟踪偏差由距离偏差和角度偏差两部分组成，消除偏差可以将两种偏差分别消除，还可以将距离偏差或角度偏差转换为其中一种形式，再将其消除。根据消除偏差的方法不同产生了多种路径跟踪算法。

4.1.1 常用路径跟踪算法

2007 DAPRA Urban Challenge中宾夕法尼亚大学研制的无人驾驶车辆ben^[49]上采用了将角度偏差和距离偏差利用评价函数统一的方法，如图4.1所示。

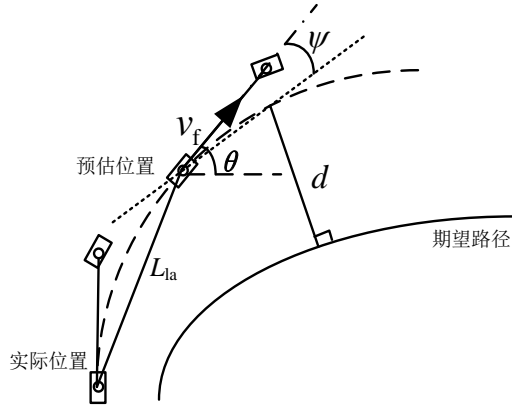


图 4.1 宾夕法尼亚大学-Ben

其中 d 为距离偏差, ψ 为角度偏差, L_{la} 为预瞄距离。评价函数如式(4.1)所示。

$$C(\delta) = d^2 + [L_{la} * \sin(\frac{\psi}{2})]^2 \quad (4.1)$$

该函数能够将角度偏差通过预瞄距离与距离偏差关联起来, 在应用中可以调节。对 $C(\delta)$ 进行比例积分调节得到前轮偏角控制量。这种方法对车辆动力学特性有较好的适应性, 能够满足车辆对精确跟踪的需要或者在泊车工况下有很好的适应性。

基于距离偏差, 俄克拉何马州立大学 (Ohio State University, OSU) 也提出了一种跟踪偏差计算方法: 环形预瞄法(Circular Look-Ahead, CLA)^[50]。以车辆当前的后轴中心位置为圆心, 以预瞄距离作为半径作圆, 将此圆被当前轨迹与期望路径所截取的圆弧的长度作为跟踪偏差, 其实质是根据几何关系把角度偏差转换为距离偏差, 如图4.2所示。

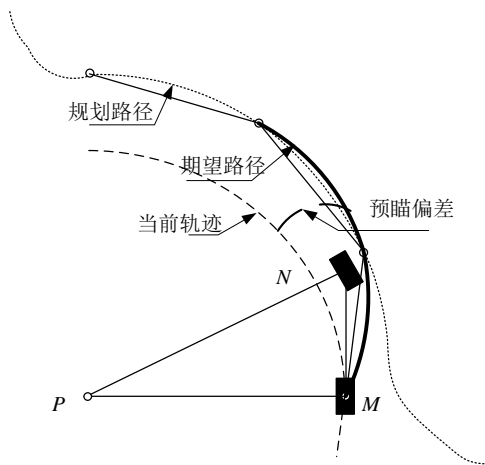


图 4.2 CLA 示意图

09年于西安举行的“智能车未来挑战赛”中北京理工大学^{[51][52]}无人驾驶车

辆采用的跟踪方法是follow-the-carrot，该方法以跟踪偏差 α 为控制目标，以道路信息和车辆位姿等作为输入，以前轮偏角为输出量，通过PID控制得到期望的前轮偏角，以达到消除跟踪偏差的目的。这种方法简单可行，在台北国立交通大学^[53]等众多大学的研究中都有应用。该方法的参考点的选择分为三种情况，正常情况下参考点的选择如图4.3 (a)所示，遇到车辆转弯场景时的情况如图4.3 (b)，当目标点在视野范围内，则直接以目标点为跟踪点，如图4.3 (c)所示。

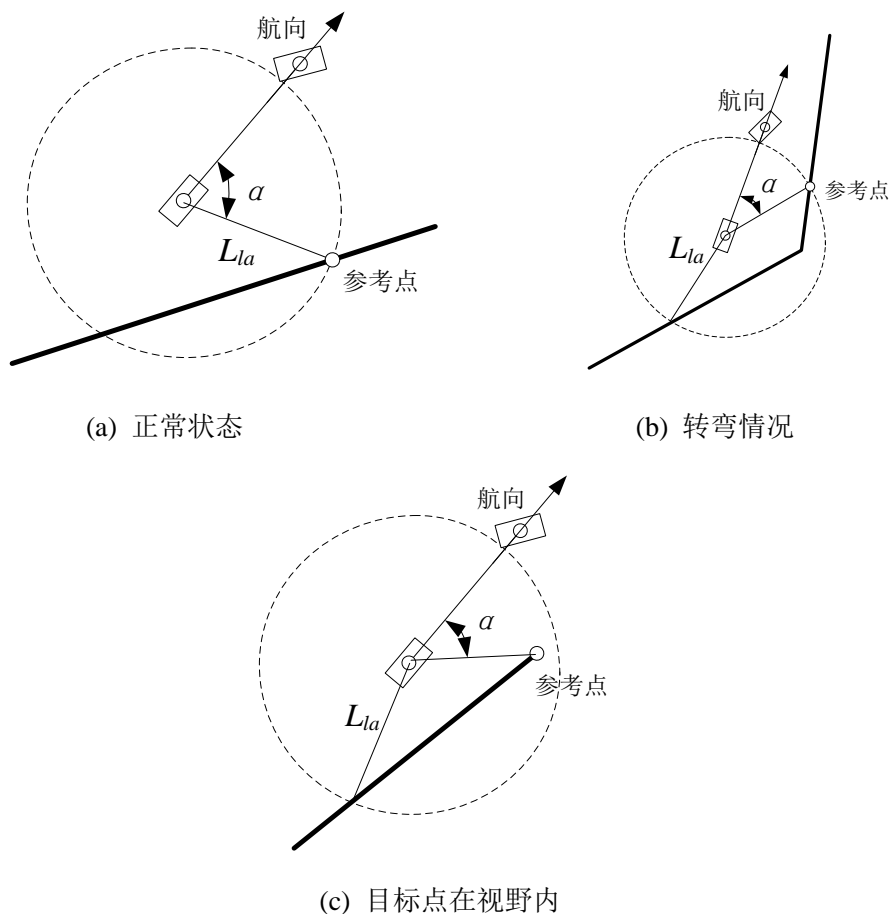


图 4.3 参考点的选择

该方法将车辆后轴中心到参考点之间的连线与车辆航向之间的夹角 α 作为跟踪偏差，如图4.4所示。

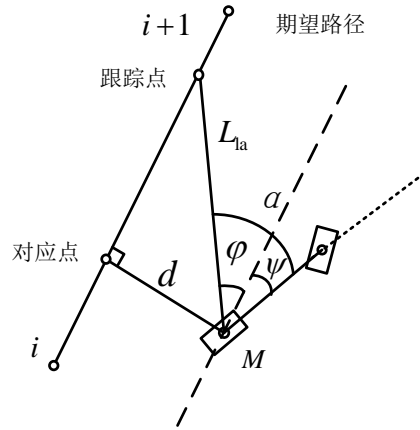


图 4.4 偏差确定

图中， d 为距离偏差， ψ 为角度偏差， L_{la} 为预瞄距离， φ 为由距离偏差转化得到的角度值， α 为由车辆位置和目标点组成的直线与车辆当前航向之间的夹角，由几何关系知：

$$\varphi = \arcsin(d / L_{la}) \quad (4.2)$$

$$\alpha = \psi + \varphi \quad (4.3)$$

由式 (4.2) 和式 (4.3) 可知，该跟踪偏差计算方法的本质是将距离偏差转换为角度值，进而控制车辆减小跟踪误差。

在 2013 “中国智能车挑战赛” 中获得冠军的北京理工大学车队在路径跟踪方面采用了 Pure-pursuit^{[54][55][56]} 跟踪算法。算法原理即假设车辆经过一个圆弧到达目标点，根据车辆模型与运动学关系得到前轮偏角控制量，如图 4.5 所示。

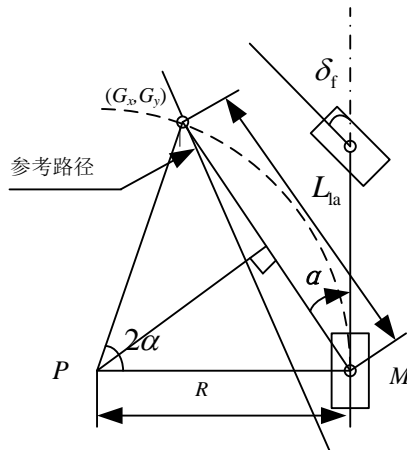


图 4.5 Pure-pursuit 路径跟踪算法

其中， (G_x, G_y) 为目标点， α 为跟踪偏差， R 为车辆行驶圆弧的半径， L 为常数，表示车辆的轴距。由几何关系得：

$$\frac{L_{la}}{\sin(2\alpha)} = \frac{R}{\sin(\frac{\pi}{2} - \alpha)} \quad (4.4)$$

$$\frac{1}{R} = \frac{2\sin(\alpha)}{L_{la}} \quad (4.5)$$

前轮偏角控制量为：

$$\delta_f = \arctan(2L\sin(\alpha) / L_{la}) \quad (4.6)$$

根据消除跟踪偏差的原理不同将路径跟踪方法分成两大类^[46]：第一类方法是通过参考路径产生描述车辆运动的动力学物理量，并根据实际跟踪误差进行反馈控制^[57]；第二种方法是基于车辆当前的位姿（位置和姿态）和参考路径之间跟踪偏差的反馈控制方法^[58]。后者根据参考路径通过车辆运动学和几何学模型计算出描述车辆运动的物理量，如前轮偏角^[56]，车辆横摆角速度^[55]，并设计反馈控制系统进行跟踪。后者以车辆前方或当前的道路信息如位置坐标、曲率等作为输入，以车辆与参考路径之间的横向偏差和航向偏差为控制目标，通过各种反馈控制方法如PID控制法、增益调度法、 H_∞ 控制法等，设计使车辆控制具有鲁棒性的反馈控制系统。这种方法可以得到较高精度的路径跟踪效果，但是需要精确的道路信息，并且可适应性差，在不同的道路环境或车辆行驶速度下的跟踪效果差别较大。

4.1.2 路径跟踪算法总结

许多基于无人驾驶车辆研究的路径跟踪算法具有一定的局限性，这些算法在特定的环境中如较高的速度、结构化道路环境等有较好的表现，但对于狭小空间、较低车速行驶的自动泊车工况，这些方法还存在一些不足。大多基于预瞄的跟踪算法^[59]，车辆实际跟踪的点是距离车辆前方一定预瞄距离之外的点，在泊车环境下车辆很难行驶到距离自身最近的期望位置，同时算法没有考虑到车辆的倒车行驶；再如基于航向和距离偏差的反馈控制算法，该算法是在车辆已经存在横向或航向偏差时采用反馈控制来消除误差，不能预知误差发生的可能进而提前采取控制措施以提高跟踪精度。

4.2 智能车辆运动学模型

车辆自动泊车的过程是一个低速运行的过程，车辆在缓慢行驶时不出现侧向力，因而没有车轮侧偏角，在这种情况下车轮滚动时没有侧向滑动。所以可以用车辆的运动学模型代替动力学模型，假设车辆的前轮速度恒定，建立车辆的运动学模型。

4.2.1 车辆几何模型

忽略车辆悬架的作用，认为无人驾驶车辆只做平行于地面的平面运动，即汽车沿 Z 轴的位移，绕 Y 轴的俯仰角与绕 X 轴的侧倾角均为零。无人驾驶车辆由前轮引导，后轮驱动，且车辆在任意时刻做直线运动或者绕某个点做圆周运动。根据无人驾驶车辆几何参数，得到车体的几何模型如图4.6所示。其中， δ_{fl} 、 δ_{fr} 分别为左前轮偏角和右前轮偏角， L 为车辆的轴距。

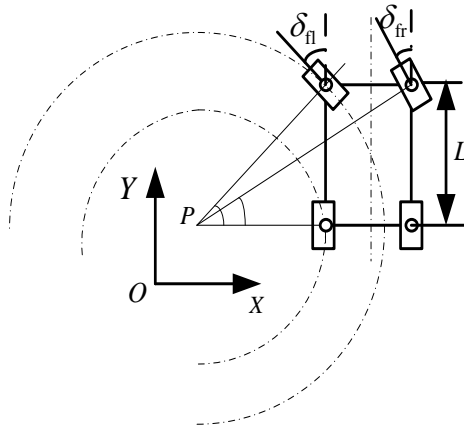


图 4.6 车辆模型

以前轮转角和速度作为输入，在每个周期内认为无人驾驶车辆速度为恒定值，因此无人驾驶车辆只有绕 Z 轴的横摆运动。则将无人驾驶车辆简化为一个二轮自行车模型，如图4.7所示，这是一个二自由度汽车模型^[60]。

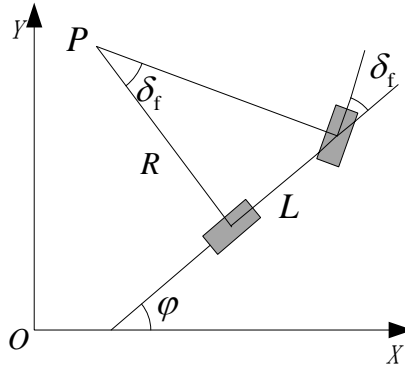


图 4.7 车辆简化模型

其中, δ_f 为前轮偏角, L 为车辆的轴距, R 为后轮在当前前轮偏角下行驶圆弧的半径。由几何关系得:

$$\delta_f = (\delta_{f1} + \delta_{f2}) / 2 \quad (4.7)$$

$$\tan \delta_f = \frac{L}{R} \quad (4.8)$$

由此可知, 汽车在低速运动时, 车身运动轨迹与方向盘输入转角有关, 与车速无关, 可以认为汽车低速时在一定转角输入时做固定半径的圆周运动。

4.2.2 车辆运动学模型

自动泊车的过程是一个车辆低速运行的过程, 车辆在缓慢行驶时不出现侧向力, 因而没有车轮侧偏角, 在这种情况下车轮滚动时没有侧向滑动^[61]。此外假设: 没有空气动力的作用, 忽略左、右轮胎由于载荷变化而引起的轮胎特性的变化以及轮胎回正力矩的作用。所以可以用车辆的运动学模型代替动力学模型, 假设车辆的前轮速度恒定, 建立车辆的运动学模型^[62]。

在惯性坐标系 OXY 下, (X_r, Y_r) 和 (X_f, Y_f) 分别为车辆后轴和前轴轴心的坐标。车辆转向运动模型, 如图 4.8 所示, θ 为车体的横摆角(航向角), δ_f 为前轮偏角, v_r 为车辆后轴中心速度, v_f 为车辆前轴中心速度, L 为轴距。

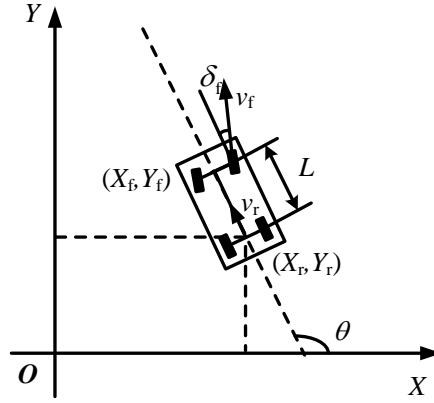


图 4.8 车辆运动模型

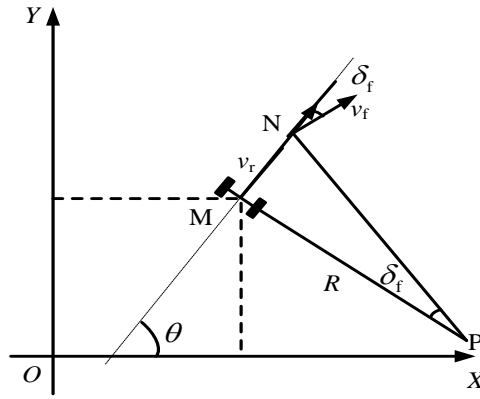


图 4.9 车辆前轮转向示意图

图 4.9 为车辆转向过程示意图， R 为后轮转向半径， P 为车辆的瞬时转动中心， M 为车辆后轴轴心， N 为前轴轴心。此处假设转向过程中车辆质心侧偏角保持不变，即车辆瞬时转向半径与道路曲率半径相同。

在后轴行驶轴心 (X_r, Y_r) 处，速度为：

$$v_r = \dot{X}_r \cos \theta + \dot{Y}_r \sin \theta \quad (4.9)$$

前、后轴的运动学约束为：

$$\begin{cases} \dot{X}_f \sin(\theta + \delta_f) - \dot{Y}_f \cos(\theta + \delta_f) = 0 \\ \dot{X}_r \sin \theta - \dot{Y}_r \cos \theta = 0 \end{cases} \quad (4.10)$$

由式(4.9)和式(4.10)联合可得：

$$\begin{cases} \dot{X}_r = v_r \cos \theta \\ \dot{Y}_r = v_r \sin \theta \end{cases} \quad (4.11)$$

根据前后轮的几何关系可得：

$$\begin{cases} X_f = X_r + L \cos \theta \\ Y_f = Y_r + L \sin \theta \end{cases} \quad (4.12)$$

将式(4.11)和式(4.12)代入式(4.10)，可以解得横摆角速度为：

$$\omega = \frac{v_r}{L} \tan \delta_f \quad (4.13)$$

式中， ω 为车辆横摆角速度。同时，由 ω 和车速 v_r 可得到转向半径 R 和前轮偏角 δ_f ：

$$\begin{cases} R = v_r / \omega \\ \delta_f = \arctan(L / R) \end{cases} \quad (4.14)$$

由式(4.11)和式(4.13)可以得到车辆运动学模型为：

$$\begin{bmatrix} \dot{X}_r \\ \dot{Y}_r \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ \tan \delta_f / L \end{bmatrix} v_r \quad (4.15)$$

该模型可以进一步表示为更为一般的形式：

$$\dot{\xi}_{kin} = f_{kin}(\xi_{kin}, u_{kin}) \quad (4.16)$$

中，状态量 $\xi_{kin} = [X_r, Y_r, \theta]^T$ ，控制量 $u_{kin} = [v_r, \delta_f]^T$ 。在无人驾驶车辆的路径跟踪控制过程中，往往希望以 $[v_r, \omega]$ 作为控制量，将式(4.13)代入式(4.15)中，该车辆运动学模型^[63]可以转换为如下形式：

$$\begin{bmatrix} \dot{X}_r \\ \dot{Y}_r \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ 0 \end{bmatrix} v_r + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (4.17)$$

本文中所用到的智能车辆平台的参数如表 4.1 所示。

表 4.1 比亚迪智能车辆平台几何参数

车长(Lc)	4.68m	轴距(L)	2.66m
接近角	31°	离去角	23°
车宽	1.765m	车高	1.49m

4.3 前馈-反馈跟踪控制算法

4.3.1 反馈控制

反馈控制^[64]是自动控制系统中的一种控制方式。在自动控制系统中，被控对象的输出量，即被控量是要求严格加以控制的物理量，它可以要求保持为某一恒定值，例如温度、压力、流量等，也可以要求按照某个给定规律运行，例如飞行航迹、记录曲线等；而控制装置则是对被控对象施加控制作用的机构的总体，

它可以采用不同的原理和方式对被控对象进行控制,但最基本的一种是基于反馈控制原理组成的反馈控制系统。在反馈控制系统中,控制装置对被控对象施加的控制作用,是取自被控量的反馈信息,用来不断修正被控量的偏差,从而实现对被控对象进行控制的任务,这就是反馈控制的原理。在电磁学的反馈电路中,我们把取出输出量送回到输入端,并与输入信号相比较产生偏差信号的过程,称为反馈。若反馈的信号与输入信号相减,使产生的偏差越来越小,则称为负反馈;反之,则称为正反馈。反馈控制就是采用负反馈并利用偏差进行控制的过程。在实际过程中,往往采用反馈手段对系统进行调节,这种方法称为反馈控制方法。使用反馈控制方法,要由控制器、执行机构、控制对象、反馈装置等四个主要部分构成一个系统,并采用信息技术对系统进行不断的调整,从而使系统达到某种特定的状态,或按规律运行。反馈控制方法的一般原理,可用图 4.10 表示,其原理为:原信号输入到控制器,经控制器转换成控制信号,送到执行机构并对控制对象产生控制作用。在外界对控制对象产生扰动的情况下,将输出信号由反馈装置送回控制器进行循环调整。实际上,所有的反馈控制方法,都是按这个过程实现的,无论是人机系统还是生命系统,都是按这一过程完成其行为的。在技术中,从恒温设施、给排水系统、工业生产线的自动控制,到鱼雷、导弹、现代航天运载工具的自动控制,都采用这种方法。

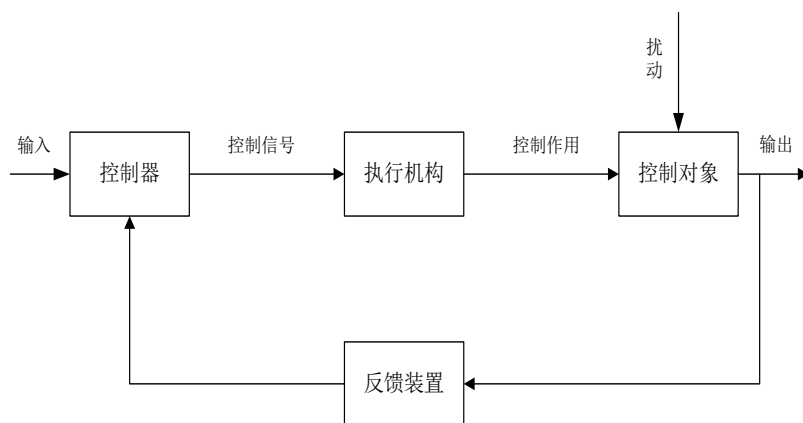


图 4.10 反馈控制图

车辆在跟踪某段路径时,会存在一定的偏差,包括航向偏差 ψ (车辆的实际航向与该路段期望航向的偏差)和距离偏差 d (车辆后轴中心到该路段的距离),如图 4.11 所示。

本文以车辆的后轴中心为研究目标,车辆在行进过程中后轴中心 M 在期望路径上的投影点为 P_i ,后轴中心到投影点的距离定义为车辆的横向偏差 d ;过投

影点作期望路径在该点的切线，切向方向代表车辆在改点的期望航向，与车辆实际的航向之间的差 ψ 定义为航向偏差。

为了消除这两方面的偏差，首先采用反馈控制，反馈输入量为航向偏差 ψ 和距离偏差 d ，输出量为车辆前轮偏角 δ_1 ，则

$$\delta_1 = \psi + \arctan \frac{kd}{v_r} \quad (4.18)$$

其中 k 为比例系数， v_r 为当前车速。由式（4.18）可知，车辆的前轮偏角是航向偏差和横向偏差以及车速的函数，航向偏差和前轮偏角成正相关，偏差越大对应的前轮偏角越大；距离偏差和前轮偏角也成正相关，这样当横向距离较大时可以快速的缩小偏差；车速和前轮偏角成负相关，根据车辆动力学特性可知，当车速较高时，进行较大幅度的转向将会造成车辆侧滑等危险情况发生，因此在消除距离偏差时引入了速度变量，防止较高车速时大角度的前轮偏角造成的车辆不稳定情况的发生。

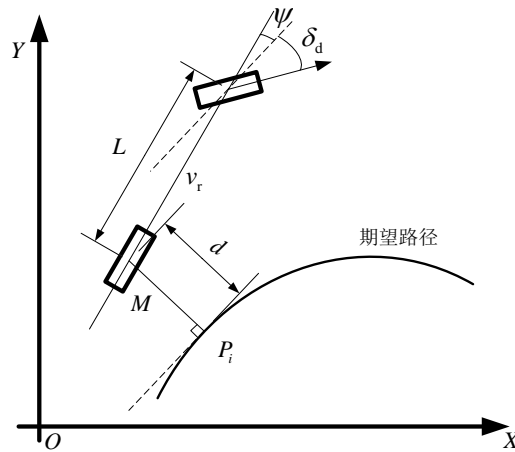


图 4.11 反馈控制

4.3.2 前馈控制

单回路控制系统和串级控制系统都是反馈控制系统，它主要是根据被控量和给定值的偏差信号来进行控制的。反馈控制的最大优点是可以克服所有引起被控量发生变化的扰动信号，但它本身也存在很大的缺点，那就是调节不及时，调节总是滞后于扰动，即只有扰动作用于系统引起被控量发生变化，导致调节器输入端的偏差信号发生变化后，调节器才改变输出的调节信号，克服扰动，对被控量进行调节。

与反馈控制相比较，前馈控制^[65]直接根据扰动信号对被控量进行调节，调节快速性很好。

前馈控制的概念包括以下四点：

(1)在前馈控制系统中，送给控制器的测量信号是扰动信号，而不是被控量，这和反馈控制存在很大的差别。反馈控制是将被控量作为测量信号，调节器是根据被控量的测量值与给定值的偏差对被控量进行调节；而前馈控制是直接根据某个扰动信号的变化来对被控量进行调节的。

(2)前馈控制系统中的控制器通常叫做前馈控制器，或者前馈补偿器，该控制器只接收某个扰动作用的测量信号，无给定值输入信号。因而从严格意义上讲，它不是一个控制器，而是一个补偿器，其作用是补偿扰动信号对被控量所造成的影响。

(3)一个前馈控制器只能对某一个扰动信号进行补偿。若系统中存在多个扰动信号，则需要设计多个前馈控制器，分别去对多个扰动信号进行补偿。

(4)前馈控制系统主要由前馈控制器、测量变送器、执行器、调节机构和被控对象组成。其中，测量变送器是对扰动信号进行测量，而前馈控制器的输入信号只有一个，即扰动量的测量信号，无被控量的定值输入信号。因而前馈控制器实际是一个前馈补偿器，补偿扰动对被控量的影响。下面结合泊车过程中的路径跟踪进行分析。

在路径规划中可以计算得到每个路径点的曲率，路径的曲率作为扰动信号，曲率的变化影响了路径跟踪效果，如果知道了扰动信号与跟踪误差的某种关系，则可以提前采取措施对被控制量进行补偿。如图 4.12 所示，为了实现车辆在小范围内对路径的精确跟踪，提前减小偏差发生的可能性，本文加入了前馈控制。车辆的前馈输入为该点处路径的曲率 ρ' ，该曲率由上述规划算法得到，根据上面所述的车辆运动学模型，得到前轮偏角控制量 δ_2 为：

$$\delta_2 = \arctan L \cdot \rho' \quad (4.19)$$

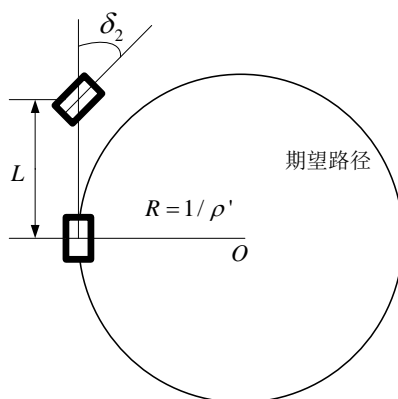


图 4.12 前馈控制

4.3.3 前馈-反馈控制的融合

在控制系统中，前馈控制和反馈控制是经常用到的两种控制方式。这两种控制方式具有各自的特点，如前所述，反馈控制能够克服所有扰动信号对被控量产生的影响，但调节滞后于扰动，不及时；前馈控制根据扰动信号进行调节，调节及时，但一个前馈控制器只能补偿某一种扰动，无法克服所有的扰动，要对所有的扰动信号进行补偿，就必须针对每个扰动信号分别设计专门的前馈控制器，这样就使得系统结构复杂。通常，我们只对主要的扰动信号采用前馈控制加以补偿，如本文中主要扰动为路径曲率的变化，而对其他的扰动信号通过反馈控制进行克服，这样，在一个控制系统中同时采用了前馈控制和反馈控制两种控制方式，这样的控制系统称为前馈-反馈复合控制系统。

(1) 前馈-反馈复合控制系统的基本概念

前馈-反馈复合控制系统中既有针对主要扰动信号进行补偿的前馈控制，又存在为克服其他扰动影响而采用的被控量反馈控制，这样的控制系统就是前馈-反馈复合控制系统。

(2) 概念的理解

复合控制是指系统中存在两种不同的控制方式，即前馈控制和反馈控制。前馈控制的作用是对主要的扰动信号进行完全补偿，可以针对主要的扰动信号来设计相应的前馈控制器。引入反馈控制是为了使系统能克服所有扰动信号对被控量产生的影响。除了已知的主要的扰动信号以外，系统中还存在其他的扰动信号，这些扰动信号对被控量的影响比较小，有的是能够考虑到的，有的根本考虑不到

或者无法测量，但都可通过反馈控制加以克服。系统中需要测量的信号既有被控量，又有扰动信号。

（3）前馈-反馈复合控制系统的特点

系统综合了反馈控制和前馈控制的优点，弥补了它们的缺点，因而前馈-反馈复合控制系统得到了广泛的应用；引入前馈补偿并不影响系统的稳定性，前馈无论放在什么位置，都不构成闭合回路，系统的特征式保持不变，因而不会影响系统的稳定性。

鉴于前馈-反馈控制系统具有上述优点，本文的路径跟踪算法把前馈控制引入到反馈中，并把最终控制量作为系统的输出。综合前馈-反馈的情况，车辆最终的前轮偏角控制量 δ_f 为：

$$\delta_f = \delta_1 + \delta_2 \quad (4.20)$$

通过发送控制量到转向电机控制器，使得实时前轮偏角为 δ_f ，实现车辆对期望路径的跟踪。

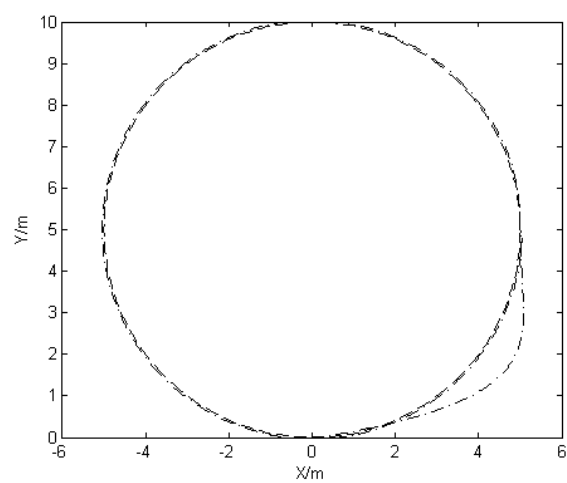
4.4 前馈-反馈路径跟踪算法仿真实验分析

为了验证前馈-反馈跟踪算法的优越性，分别通过两种场景的仿真实验对改进前后的结果进行分析。仿真实验在 MobileSim 中进行，把车辆模型设置为第 2 节中建立的运动学模型，比例参数 k 经多次调整取 0.5，平均车速设置为 0.5m/s。设计试验场景，让小车跟踪曲线和直线-曲线路径，通过对跟踪误差的分析对比，验证前馈-反馈控制算法较之反馈控制算法的优越性。

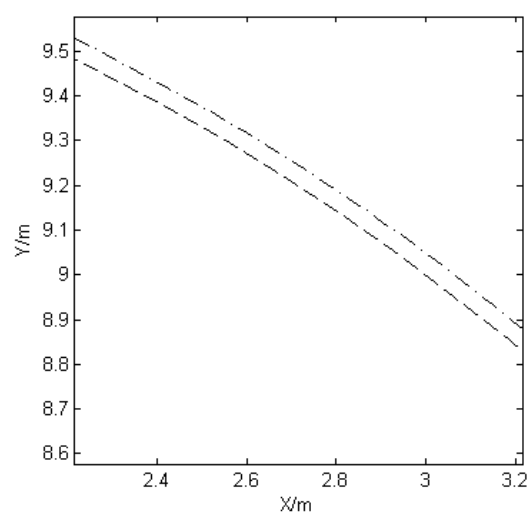
4.4.1 曲线跟踪场景

车辆在泊车过程中的期望路径由直线和曲线组成，在直线段时路径的曲率保持不变为 0，前馈补偿为 0，因此前馈-反馈算法的效果和反馈算法一致，这里不做分析对比。针对曲线情况，分别设置了圆形和椭圆形期望路径，由于曲率不为 0，因此前馈控制能根据扰动量的变化提前采取补偿措施，减小了误差出现的可能，并与纯反馈算法做了对比分析。

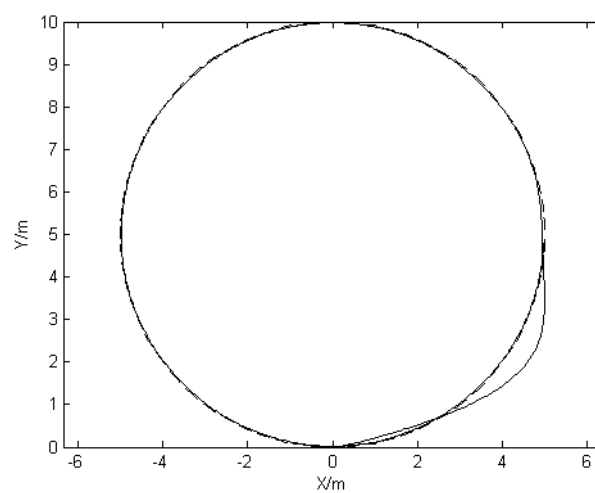
实验一：半径 $r=5m$ 的圆



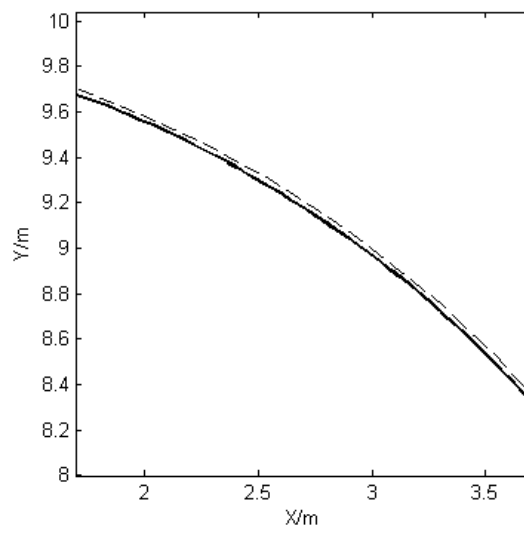
(a) 反馈跟踪结果



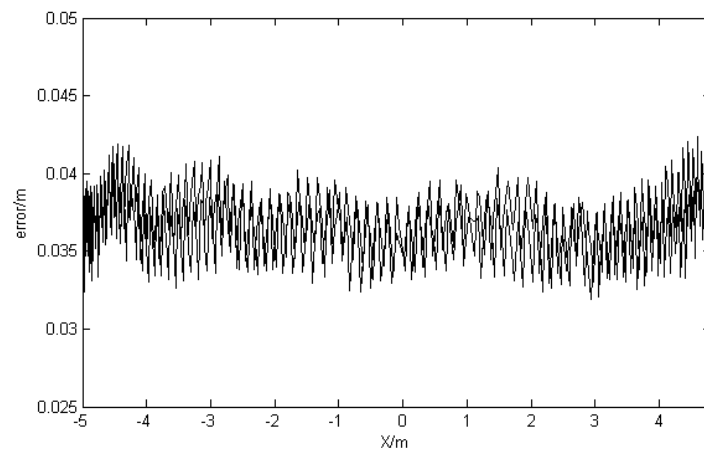
(b) 反馈跟踪结果局部放大图



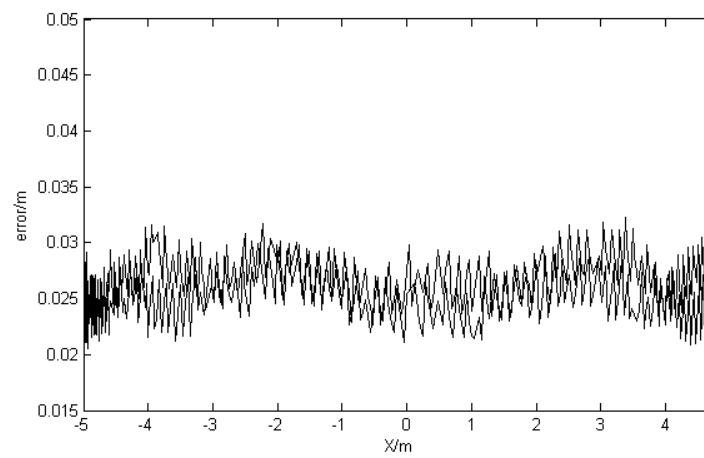
(c) 前馈-反馈跟踪结果



(d) 前馈-反馈跟踪结果局部放大图



(e) 反馈控制横向偏差

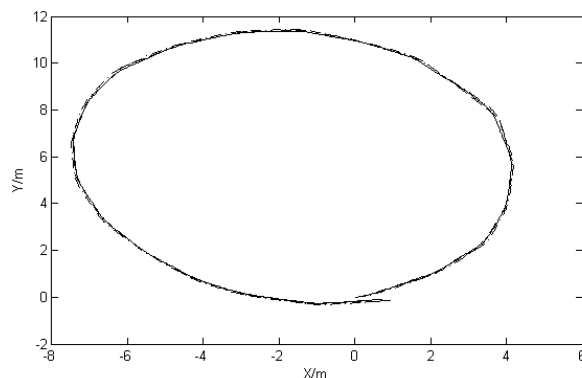


(f) 前馈-反馈控制横向偏差

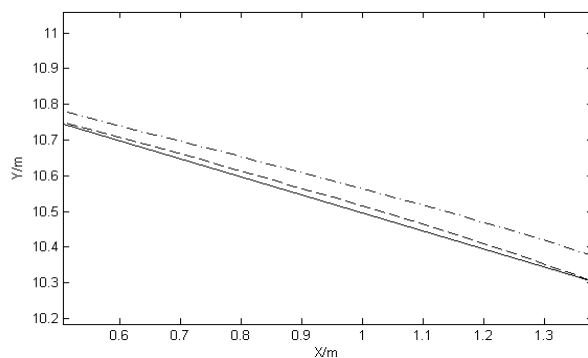
图 4.13 跟踪 $r=5$ 的圆

跟踪结果如图 4.13 所示，其中虚线代表期望路径，细实线代表应用前馈-反馈方法获得的车辆实际行驶轨迹，点画线表示应用反馈方法得到的车辆实际行驶轨迹。由图 4.13 (e) 和图 4.13 (f) 可知，反馈控制的横向跟踪误差在 0.04m 上下波动，前馈-反馈的跟踪误差在 0.025m 上下波动，结果表明在跟踪圆形曲线时，前馈-反馈算法具有更高的跟踪精度。

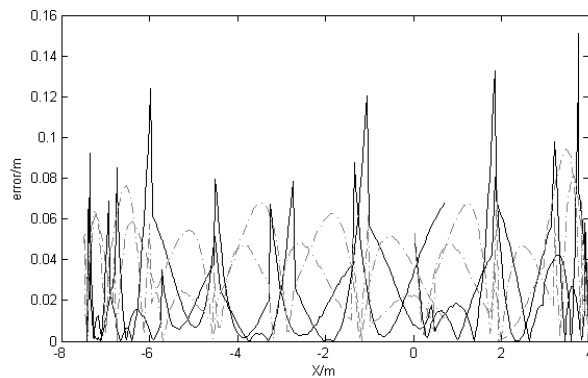
实验二：椭圆



(a) 椭圆跟踪结果



(b) 椭圆跟踪结果局部放大图



(c) 椭圆跟踪误差

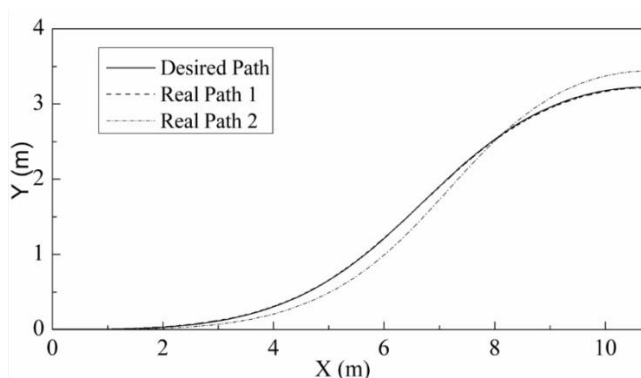
图 4.14 跟踪椭圆

跟踪椭圆的仿真结果如图 4. 14 所示，图 4. 14 (a) (b) 中黑色虚线为参考路径，点画线为反馈控制得到的车辆实际轨迹，细实线代表前馈-反馈控制方法得到的车辆实际轨迹。由于人工给定的参考路径点较为稀疏，因此参考轨迹实际上是由一些直线段连接成的多边形。从图 4. 14 (c) 中可以看出，多边形的顶点处，采用反馈控制算法的车辆轨迹的曲率会出现较大的变化，即在该点处车辆前轮偏角变化较大；与此同时，采用前馈-反馈控制算法的车辆行驶轨迹变化较为平缓，即前轮偏角不会出现较大的变化。证明了前馈-反馈算法在跟踪曲线时具有较好的平顺性，冲击度较小。

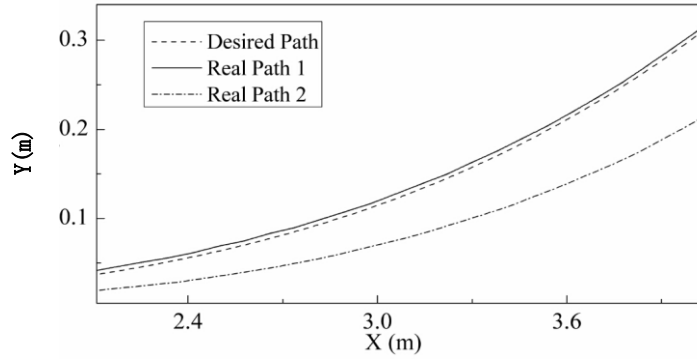
4.4.2 直线-曲线跟踪场景

泊车过程中，期望路径一般由直线段和曲线组成，在直线到曲线过渡的位置，如果采用纯反馈的跟踪算法，则车辆在行进过程中保持前轮偏角不变跟踪直线，当直线段结束，出现横向和航向偏差时，车辆才会根据反馈系统来消除偏差。与之不同，前馈-反馈控制算法能够根据期望路径曲率的变化，提前预测车辆下一路段的情况，当遇到曲线路段时，提前给予补偿，如提前改变前轮偏角来减小误差发生的可能性。试验时车辆平均速度为 0.5m/s，对比结果如图 4. 15 所示。

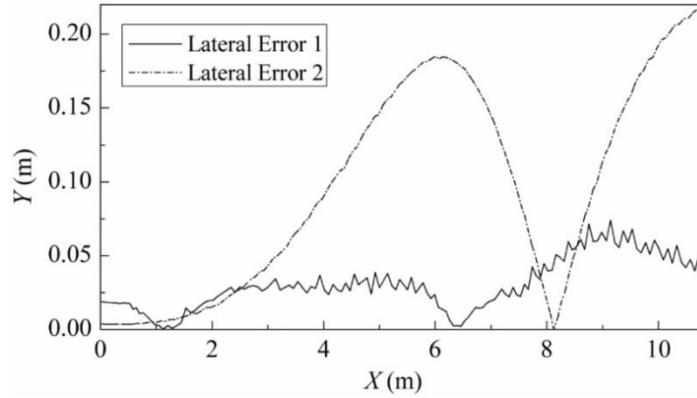
图 4. 15(a)中虚线表示平行泊车期望路径，它是由反正切函数和多项式函数组成；实线表示采用前馈-反馈跟踪算法得到的车辆实际行驶轨迹；点画线则表示采用反馈跟踪算法获得的车辆实际行驶轨迹；图 4. 15(b)为跟踪结果的局部放大图；图 4. 15(c)为采用前馈-反馈算法和反馈算法跟踪的横向误差对比，其中点画线代表反馈方法跟踪误差，实现则为前馈-反馈方法跟踪误差。



(a) 期望路径与实际路



(b) 局部放大视图



(c) 跟踪误差对比

图 4.15 试验结果对比

从图 4.15 (c) 中可以看出车辆在刚开始直线段行驶时, 这两种跟踪方法的误差非常接近, 近乎于零。当结束直线段行驶开始曲线段跟踪时, 反馈跟踪误差逐渐增大, 而前馈-反馈跟踪方法的横向偏差则基本保持不变, 因此采用前馈控制能够根据期望路径曲率的变化趋势提前采取措施, 减小误差发生的可能性。由图可以看出, 整个跟踪过程中车辆的横向偏差小于 6cm, 满足了车辆在平行泊车工况下对路径跟踪的精确性要求, 证明了该方法在泊车路径跟踪中的可行性。

4.5 本章小结

总结现有的路径跟踪方法存在的不足, 本文提出了前馈-反馈路径跟踪方法, 即在反馈跟踪算法的基础上引入前馈控制, 预测误差发生的可能并采取补偿措施, 实现路径的精确跟踪, 实验通过车辆对直线和曲线的跟踪效果证明了该方法的可行性与优越性。

第 5 章 实车实验

为验证全自主泊车系统在实际应用中的可行性,利用实验室的智能车辆实验平台,开展了大量的实车实验工作。实验研究结合智能车辆的环境感知、路径规划、地图重建以及车辆定位等模块,在停车场等环境下测试系统整体功能,并通过系统的性能来验证前几章提出的方法的正确性和可行性。

5.1 全自主泊车试验平台开发

5.1.1 全自主泊车实验平台设计

全自主泊车系统实车实验平台是智能车辆实验平台的一个组成部分。因此,车辆纵横向的自动控制开发应基于智能车辆体系结构来设计,以满足智能车辆的整体需求。下面从智能车辆体系结构^[66]出发,开发自动泊车系统实验平台。本文的自动泊车系统主要由四个方面组成,分别是操作面板和显示屏、中央控制器、激光雷达传感器以及底层执行模块,如图 5.1 所示,各部分功能说明如下:

(1) 环境采集传感器包括设置在车身上的激光雷达设备,用来采集车辆周围的环境信息,进行障碍物检测,并为后面的地图创建、车辆定位以及车位检测提供环境信息。所述环境采集传感器为安装在车头正中位置的激光雷达。

(2) 车载显示器和控制面板模块是用于驾驶员观测车辆周围环境信息并与系统进行交互的界面。地图中显示的区域为激光雷达传感器进行信息融合后得到的车辆周围环境地图。

(3) 中央控制器是泊车系统的核心部分,它用来处理外部传感器传送过来的数据,同时接收控制面板发出的信号,经过一系列的信息处理,最终输出控制量给底层控制模块,控制车辆按照预定的轨迹行驶并完成泊车。中央处理器主要由地图创建模块、车辆定位模块、路径规划单元和路径跟踪单元组成。

(4) 底层控制模块是实现泊车功能的最终执行者,它通过 CAN 总线接收中央控制器发送过来的数据,分别控制车辆的转向、加减速、制动以及档位的自动切换,使得车辆沿着期望路径行驶。

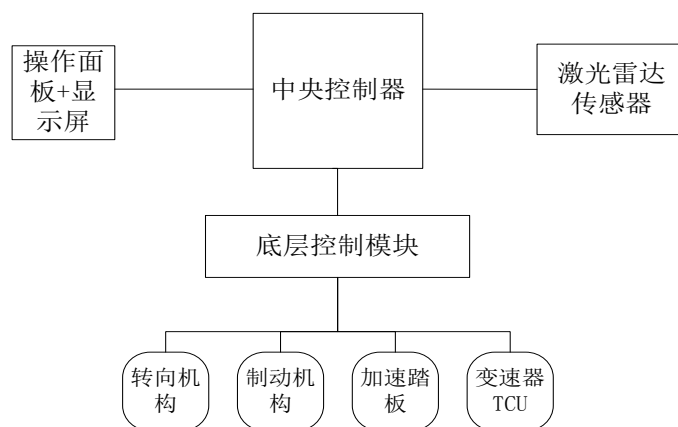


图 5.1 智能车辆体系结构

5.1.2 全自主泊车实车实验平台实现

根据全自主泊车系统所需的软硬件条件，课题基于北京理工大学无人驾驶车辆 Ray 开发了全自主泊车实验平台。

该智能车辆系统主要由 4 个部分组成，传感器、感知计算机、规划控制计算机和车辆等，系统结构如图 5.2 所示。感知计算机与规划计算机之间通过以太网通信方式进行数据交换；规划控制计算机和车辆控制 ECU 之间通过串口进行通信；车辆执行机构控制部分则通过 CAN 总线进行数据传输和共享。

本文的车辆实验平台采用国产某型轿车，具有电子节气门，可通过 CAN 总线与发动机 ECU 通信，对电子节气门开度进行直接控制；而制动由于无法实现线控，需外加执行机构，通过有刷直流电机控制制动踏板模仿驾驶员踩制动踏板的动作。系统包括以下几个方面：

（1）车辆传感器信号

传感器信号包括车速、档位信息等。以上信息均来自于车体 CAN 网络，分别由 ABS 和 TCU 控制器提供。

（2）控制器

规划控制计算机采用 Advantech 原装工业控制器 ARK-3440F-U4A2E，实现定位定向、任务规划、路径规划、纵向规划与控制等功能。通过串口与车辆控制 ECU 进行通信，下达控制命令并接收车辆状态信息。

（3）执行机构

在所采用的智能车辆实验平台中,电子油门和转向均采用线控方式,上层控制器将控制命令发送到车载 CAN 网络中,发动机 ECU 和 EPS 的 ECU 控制其系统原装执行机构实现对电子油门和方向盘的控制。但纵向控制中需要对电子油门和制动踏板进行控制,而在该平台中无法通过 CAN 总线实现对制动系统(或 ESP)的干预,因此为了实现对制动踏板的控制,需要外加执行机构。在驾驶室安装一台有刷直流电机,通过拉线的方式对制动踏板进行控制,模仿人驾驶车辆时的操作。电机采用 FAULHABER 3863H024C,其控制器采用 MCDC3006,车辆控制 ECU 通过串口与电机控制器进行通信,实现对电机的控制。

感知计算机根据摄像头、激光雷达等环境感知传感器提供的数据进行检测、识别,得到车辆前方道路和障碍物信息等。规划控制计算机首先根据感知计算机提供的环境信息判断智能车辆可行驶区域,并根据自身行驶状态对智能车辆的运动进行规划,包括路径规划和速度规划;然后根据车辆自身状态和规划计算机提供的运动规划结果计算车辆控制所需的控制量,并通过串口传输给车辆控制 ECU。车辆控制 ECU 作为车辆控制系统和上层(规划控制计算机、感知计算机等)之间的纽带,完成数据的上下传输功能。车辆控制 ECU 直接从原车的 CAN 总线上获取所需车辆行驶信息,如车速、前轮偏角等,发送给控制计算机。同时,车辆控制 ECU 从控制计算机接收期望控制指令,如期望车速、期望前轮偏角、期望档位等,并根据从原车 CAN 总线上获取的车辆当前信息,按照车辆控制系统(发动机电子喷射控制系统、变速器、电动助力转向系统、车身电子控制系统等) ECU 的需求对指令分解、再加工,按照通信协议格式发送到车体 CAN 总线上。转向控制 ECU、发动机电子喷射控制系统 ECU 和档位控制 ECU 分别从各自对应的 ID 中获取期望控制指令和当前实际状态,通过反馈对转向执行机构、电子节气门、换档执行机构进行控制,实现车辆的期望运动。CAN 总线的搭建方便了模块的加入,如紧急制动系统等模块可以方便地作为一个 CAN 节点添加到 CAN 总线中。

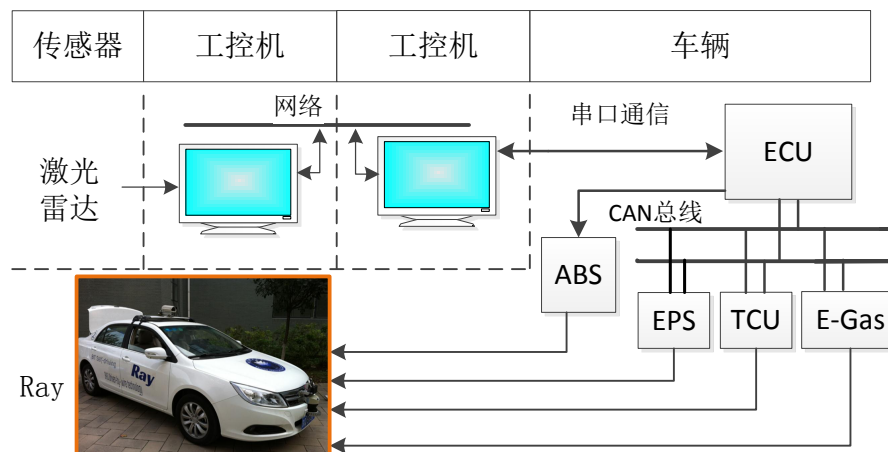


图 5.2 智能车辆体系结构

5.1.3 智能车辆实验平台集成

图 5.3 表示以上述智能车辆实验平台的开发工作为基础所搭建的智能车辆实验平台总体布置与实物。图中看出，除传感器以及工控机外，还设计并安装了供电系统，以便通过开关操作传感器、控制系统和执行器的供电情况。

平台中传感器的布置紧凑、车辆控制执行机构的安装与改造合理、供电系统与控制系统的操作方便。这大大提高了平台在试验中的可操作性和可靠性。



图 5.3 智能车辆实验平台

5.2 全自主泊车系统实验设计及结果

试验在北京理工大学校园内进行，全自主泊车试验的操作流程如图 5.4 所示。实验操作流程如下：

(1) 驾驶员在准备泊车时启动程序，接着，车辆在人工驾驶行进中创建车辆周围环境地图并寻找可以停泊的空间。

(2) 当存在合适泊车位时，驾驶员停车，把档位切换至 N 挡，此时驾驶员在显示屏上选择泊车位，并在泊车位边框中设定车辆的位置和车头朝向，点击确认泊车按钮。具体做法为在显示屏的地图中点击第一点确定为车辆中心的位置，同时点击第二个点，前后两点的连线为车辆的航向，由此确定车辆最终位姿。

(3) 系统根据车辆当前以及最终位姿规划出一条可行的泊车路径，车辆将按照设定的算法沿着该路径行驶。

(4) 如果跟踪失败，则车辆停止行驶，进行重新规划，如果道路确定不可通行，则反馈跟踪失败信号给控制面板，此时由驾驶员重新选定泊车位，接下来车辆继续执行步骤 3 和 4。

(5) 最终车辆行驶到预定的位置，系统发送泊车成功信号给控制面板，泊车结束。

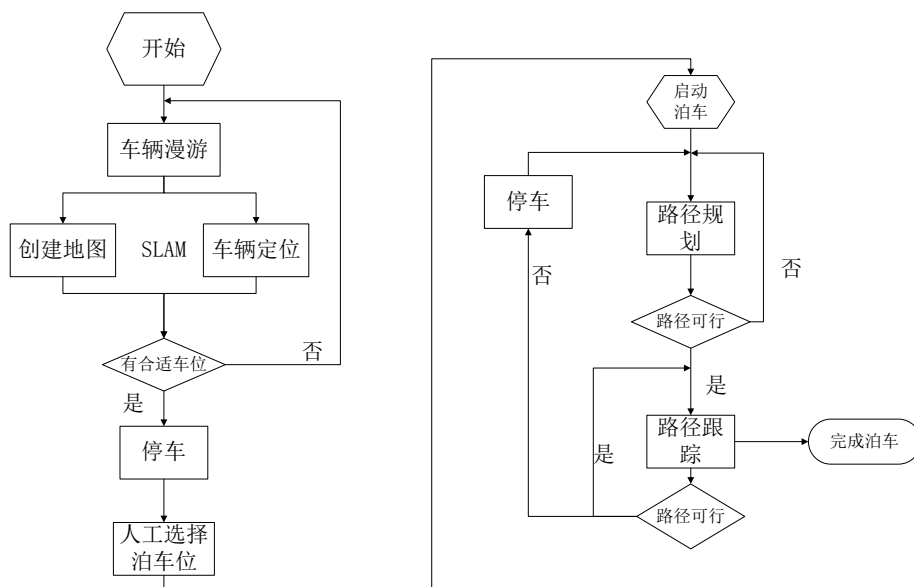


图 5.4 全自主泊车工作流程

5.2.1 平行泊车

平行停车位是城市环境中最常见的类型,这种车位的泊车对驾驶员的考验也最大。实验设置了车辆左前方平行停车位。试验场景的设置如图 5.5 所示,图 5.6 为实际泊车环境。泊车过程中车辆的平均行驶速度为 0.3m/s 。

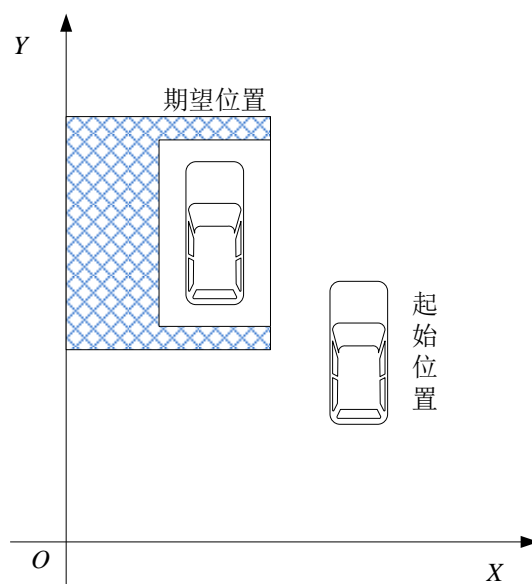


图 5.5 车辆平行泊车示意图



图 5.6 车辆平行泊车实验场景

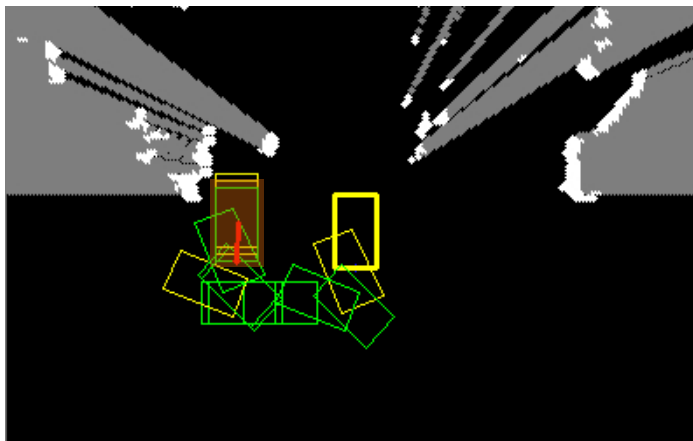


图 5.7 车辆自动平行泊车行驶轨迹

目前对于自动泊车系统的性能还没有统一的评价指标,本文单纯从泊车成功与否的角度定义了泊车的效果,认为满足以下两点要求,即认为系统具有较好的自动泊车性能。一是车辆最终停靠在了车位以内,并且车辆四周距车位边框或障碍物的最小距离大于 10 厘米;二是车辆最终的航向与车位纵向的夹角在 $[-10, 10]$ 度范围内。

图 5.7 为车辆的几何形状在平行泊车过程中的运行轨迹随时间的变化情况。停车空间的大小为 $l=6\text{m}$, $h=2.4\text{m}$, 停车空间的长度约为车长的 1.3 倍, 停车空间的宽度约为车宽的 1.36 倍, 如图中棕色矩形所示。车辆到达泊车位大致经过五个步骤, 倒车-前行-倒车-前行-倒车, 经过多次的运动和姿态调整最终泊车入位。如图 5.7 所示, 车辆最终停在了棕色矩形之中, 通过测量, 车辆终点航向与车库纵向轴线夹角约为 4 度, 控制效果较好。

5.2.2 垂直泊车

车辆垂直泊车即自动入库倒车, 在实际应用中有广泛的应用背景, 其控制任务如图 5.8 所示, 车辆从起始位置开始泊车, 由于倒车空间较小, 车辆首先进一段距离, 然后开始倒车。当车辆的尾部到达停车空间的右侧时, 车辆的前轮以一定的角度进行转向, 随着车辆与停车空间相对位置的变化车辆的前轮转向也随之发生变化。车辆航向由 90 度左右开始逐渐增大(车辆航向为逆时针逐渐增大), 当接近于 180 度时, 即车辆主轴与 Y 轴垂直时, 车辆前轮的转向角变为 0 度。此时, 车辆垂直入库。当车辆的尾部接近停车空间的底部时, 到达期望位置并停车, 整个自动垂直泊车任务完成。泊车过程中车辆的平均行驶速度为 0.4m/s。

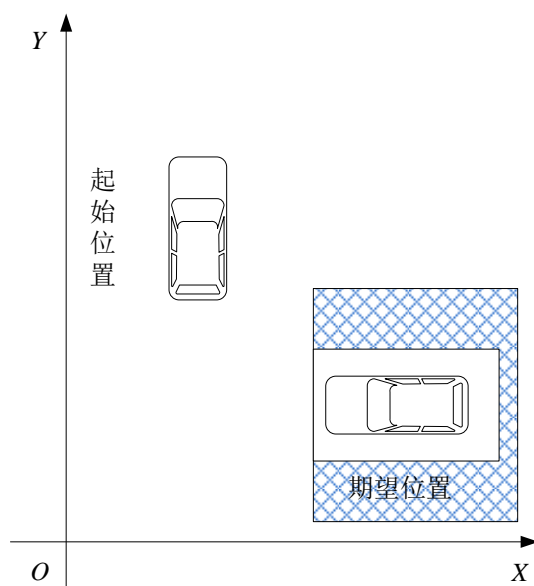


图 5.8 车辆垂直泊车示意图

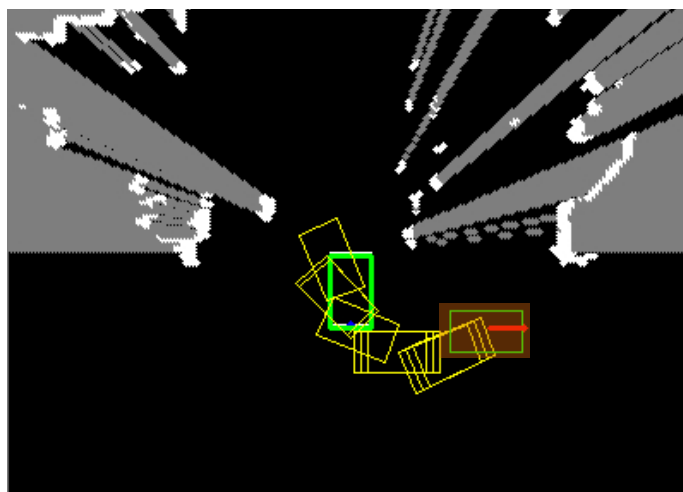


图 5.9 车辆自动垂直泊车行驶轨迹

图 5.9 为车辆的几何形状在垂直泊车过程中的运行轨迹随时间的变化情况。停车空间的大小为 $l=5.5\text{m}$, $h=2.5\text{m}$ 。测量显示, 车辆最终停靠在棕色区域之中, 车辆到达停车空间底部时车辆航向与车位纵向轴的夹角约为 3 度, 控制效果较好, 设计的自动泊车系统较好的完成了车辆垂直泊车的任务。

5.2.3 任意姿态泊车

针对任意位姿的泊车, 设计了车辆斜形泊车场景, 该泊车形式在实际中也有广泛的应用, 主要适用于斜形停车位。试验场景设置如图 5.10 所示, 实际泊车环境如图 5.11 所示。车辆从预备位置前行, 在中间位置遇到障碍物倒车, 之后

按照规划路径避开障碍物到达停车位,最后经过一次倒车和前行调整位姿完成泊车。实验过程中车辆平均行驶速度为 0.4m/s。

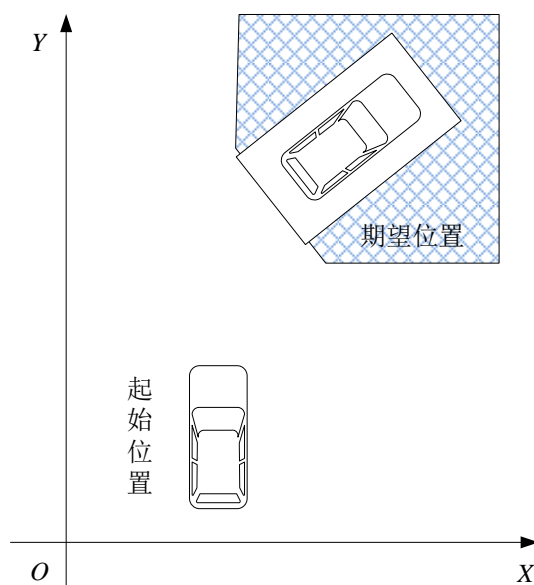


图 5.10 任意位姿泊车示意图



图 5.11 任意位姿泊车实验场景

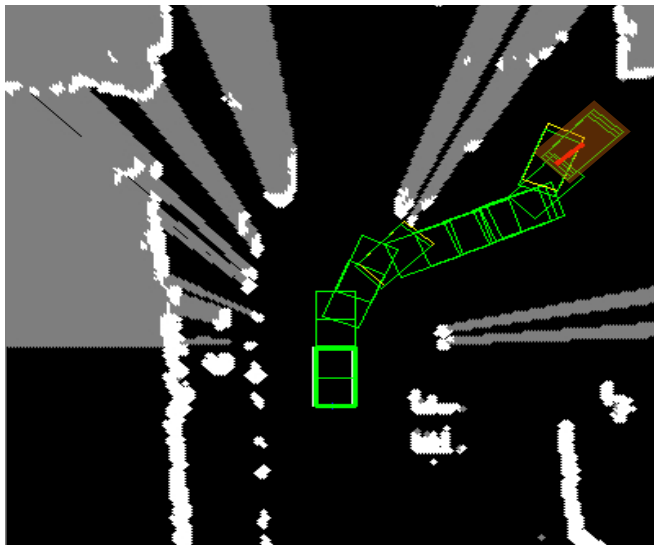


图 5.12 车辆任意位姿泊车行驶轨迹

图 5.12 为车辆的几何形状在任意位姿泊车过程中的运行轨迹随时间的变化情况。停车空间的大小为 $l=6\text{m}$ ， $h=2.5\text{m}$ ，停车空间的长度约为车长的 1.3 倍，停车空间的宽度约为车宽的 1.4 倍，停车空间的倾角为 45 度。行车过程中未与障碍物发生碰撞，车辆到达停车空间底部时车辆航向与车位纵向轴的夹角约为 5 度，航向和位置误差均满足要求，控制效果较好。

5.3 本章小结

为了对本文设计的全自主泊车系统进行实车验证，在北京理工大学无人驾驶车辆 Ray 的基础上，开发了全自主泊车实验平台。首先对车体控制系统进行改造，实现车辆纵横向的自动控制。在实际的停车场环境中，分别选择了平行、垂直以及任意车位类型的泊车进行试验，车辆完成泊车后，测量车辆的实际位置与选定的车位之间的误差来判定泊车的性能，经过多次试验，车辆都成功泊车入位，车辆航向偏差也在合理的范围内，即验证了本系统的可行性，同时也证明了系统所涉及算法的有效性。

第 6 章 总结与展望

6.1 工作总结

本文设计了一种全自主泊车系统。总结前人研究的成果,本系统在通常的自动泊车系统上做出了重大改进,实现了车辆任意位姿的自动泊车。这主要表现在该系统不仅能够完成平行或垂直泊车,对于任意选定的位置也能够规划出一条从起点到终点的无碰撞路径,即在一定范围内对泊车起始和终点位置不做限定。该系统能够同时控制车辆的纵横向运动,泊车过程中无需驾驶员的参与。泊车过程中跟踪失败时,系统能够重新启动规划,搜索一条新的无碰撞路径,当存在障碍物导致规划失败时,系统能够提示驾驶员人工操作完成泊车。

本文首先利用激光雷达传感器创建了车辆周围的二维环境地图,基于此地图,利用启发式搜索算法在 Lattice 状态空间中规划出可行的泊车路径;建立了车辆的几何和运动学模型,设计基于 LQR 的前馈-反馈路径跟踪控制算法,并通过试验验证了算法的可行性。车辆行驶过程中,根据车载传感器与地图匹配实现定位,通过滤波获取精确地车辆位置,定位精度高于航迹推算方法,满足在地下停车场等没有 GPS 信号的场景下车辆精确定位的需求。当跟踪失败时能够重新规划,搜索新的可行路径。最后本文设计了系统所需的硬件平台,包括传感器、工控机、执行机构以及各控制器,搭建了自动泊车实验平台。

系统在北京理工大学无人驾驶车辆上完成测试,在实际泊车环境中,分别设置了平行、垂直和斜行停车位,实验前由驾驶员在地图中选定所设停车位,测试车辆能否顺利驶入指定的泊车位置。通过多次实验,结果表明该系统具有较好的泊车性能,在完全自主的情况下完成了任意位姿的自动泊车。

6.2 课题展望

本文综合了地图创建与定位、路径规划以及路径跟踪等算法,设计了全自主泊车系统,并在北京理工大学无人驾驶车辆上得到了实验验证。但是该系统还存在一定的缺陷,需要下一步继续的研究和完善,具体有以下几点不足:

(1) 采用的激光雷达传感器为单线雷达,只能采集某一固定高度所在的平面有无障碍物,在该高度以上或以下的障碍物则不能检测,这将导致在泊车过程

中较低或者较高的障碍物被漏检，造成车辆与之发生碰撞或刮擦。

(2) 本文建立的栅格地图单元格大小为 $0.2\text{m} \times 0.2\text{m}$ ，这种栅格的尺寸在泊车狭窄的环境中较大，容易导致规划出来的路径不是最优路径。但是随着单元格尺寸的变小，相同大小地图中单元格的数量将会增加，这将导致基于栅格地图的路径搜索时间增加，满足不了高效率泊车的要求。因此，如何做到最优化与效率之间的平衡将是未来工作研究的一个重点。

(3) 本系统中车辆最终的停泊位姿由驾驶员人工确定，即在栅格地图中，驾驶员需要在所要停泊的位置选取两个点，第一个点确定车辆中心最终的期望位置，第一点指向第二点的矢量方向作为车辆最终的车头朝向，这样就确定了车辆终点的位姿。这种方法虽然可以按照驾驶员的意图在任意位置停车，但是由于驾驶员在地图中选择点的时候具有不确定性，驾驶员选择的最终泊车位置有可能和障碍物的距离较近或者与障碍物发生干涉，因此该系统需要增加车位自动识别模块，检测空白区域能否泊车以及泊车位的类型。同时在驾驶员手动选择之后，应该增加车位是否合适的检测环节，如果与周围的障碍物距离较近或者发生干涉，系统应能够判别并对驾驶员进行提示。

(4) 在自动泊车过程中车辆的平均行驶速度较低，随着速度的增大跟踪效果也随之恶化，因此下一步还需要对跟踪算法进行优化，同时把车辆横向规划与纵向规划结合起来，使得两者有效的耦合，保证车辆精准的跟踪性能。

(5) 泊车过程中没有加入障碍物碰撞分析，如果中途有障碍物出现，将会导致泊车无法进行。

总之，本方案设计的全自主泊车系统还存在一些不足，需要进一步的改进和完善。

主要符号对照表

m_j	网格
$p(m_i, d)$	占据概率
P_{score}	匹配率
A	状态转移矩阵
w	测量矩阵
$h(s)$	启发函数
z_i	激光雷达观测值
d_m	曼哈顿距离
d_d	对角线距离
d_e	欧基里德距离
d	距离偏差
ψ	航向偏差
L_{la}	预瞄距离
R	车辆行驶圆弧的半径
L	车辆的轴距
δ_f	前轮偏角
δ_{fl}	左前轮偏角
δ_{fr}	右前轮偏角
P	瞬时转向中心
M	车辆后轴轴心
v_r	车辆后轴中心速度
δ_1	反馈计算前轮偏角
δ_2	前馈计算前轮偏角
ρ'	路径的曲率
θ	车体的航向
ω	车辆横摆角速度
k	距离偏差比例系数
h	车位宽度
l	车位长度

攻读学位期间发表论文与研究成果清单

[1] XuDalu. An Improved Path Tracking Algorithm for Intelligent Vehicle in Automatic Parking Conditions.

[2] 全自主泊车发明专利

致谢

本论文的工作是在导师龚建伟副教授的悉心指导下完成的。感谢龚老师两年来在学习和生活上对我的启发和教导，这令我终生难忘。他以渊博的专业知识、丰富的科研经验和细致严谨的科研态度，以及和蔼可亲的笑容，使我深受启迪和感动。

两年前我从一个近乎一无所知的本科生来到智能车辆实验室，两年的时间使我见识了很多，开阔了眼界，看到周围老师以及同学们对科研的态度和勤奋好学的品质，也让我对自己的要求提高了很多。老师们对待学生如同自己的孩子，不会因为自己的底子差而忽略自己，相反的是给予自己耐心的指导，有些事情做得不够好，老师们给的更多的是鼓励和包容。如今即将离开学校，两年的学习使自己受益匪浅，虽然科研上没有多大的成就，但在实验室的每一天都学到了自己未知的东西。非常感激研究生的生涯，感激龚老师的谆谆教诲，这里收获的一切我都会铭记在心。

课题及实验工作的顺利完成更受到了姜岩老师的指导和帮助，姜老师对待科研专注的精神和孜孜不倦的态度深深地感染了我，每个周的工作汇报虽然受到了不少批评，但这些指正都十分中肯，让我对要研究的内容更加清晰明朗，也让自己错误的做事方法得到了改正。

同时也要感谢总是带给同学们笑声的熊光明老师和严肃认真的陈慧岩老师，感谢你们的悉心指导和耐心帮助。

感谢实验室的胡玉文、江燕华、张浩杰、刘凯、张玉、权苗苗、王金航、孙银健、李勇、肖强、李宁、周帅、张瑞琳、刘璐、王诗源、蒋键等同学们对我的课题给予的帮助，三人行，必有我师，跟你们相处的这段时间我收获了知识还有友情，在此向你们表示衷心的感谢。

最后，我还要感谢我的父母、家人以及朋友给予我学业上的不懈支持。

参考文献

- [1] World Health Organization. The Global Burden of Disease: 2004 Update [R]. WHO Scientific Group. Geneva: WHO, 2008.
- [2] 杨殿阁等. 智能交通系统中的汽车技术[J], 汽车工程, 2003, 25(3): 223- 226.
- [3] 庄继德. 汽车电子控制系统工程[J]. 北京: 北京理工大学出版社, 1998.
- [4] 柳洁. 基于模糊控制的汽车倒车辅助决策系统 [D]. 广东工业大学硕士学位论文, 2005.4
- [5] 徐中明. 智能交通系统 (ITS) 中的智能汽车技术[J]. 重庆大学学报, 2005, 28(8):17-21.
- [6] 宋金泽. 自主泊车系统关键技术研究[D]. 国防科技大学博士学位论文,2009.10.
- [7] 吴锴. 智能自动泊车系统研究[D]. 南京理工大学硕士学位论文,2008.6.
- [8] 蒋志文, 曾祎. 基于双目视觉和路径规划的车辆自动泊车系统[J]. 公路与汽运. 2008, 第 4 期, 69-72.
- [9] 李占江. 车辆自动泊车的模糊控制方法研究[D]. 吉林大学硕士学位论文, 2007.5.
- [10] 柳洁. 基于模糊控制的汽车倒车辅助决策系统[D]. 广东工业大学硕士学位论文, 2005.4.
- [11] Tomoaki Kobayashi, Sumiko Majima. Real-Time Optimization Control for Parking a Vehicle Automatically[C]. Proceedings of IEEE International Conference on Vehicle Electronics, 2001, 97-102.
- [12] K.Jiang, L.D. Seneviratne. A sensor guided Autonomous parking system for nonholonomic mobile robots[C]. Processings of the 1999 IEEE International Conference on Robotics and Automation, Detroit, MI, May,1999, 311-316.
- [13] Liu Liang, Zhang Lei. The Simulation of an Auto-parking System[C]. 2011 6th IEEE Conference on Industrial Electronics and Applications.2011.
- [14] 高 航, 梁华为. 基于多传感器多路径规划自动泊车系统仿真及实车验证[J]. 《自动化与仪器仪表》.2011 年第 4 期 (总第 156 期) .
- [15] Kyoungwook Min, Jeongdan. Design and Implementation of Path Generation Algorithm for Controlling Autonomous Driving and Parking[C].2012 12th International Conference on Control, Automation and Systems. Jeju Island, Korea. 2012:17-21.
- [16] Roberto Carbrera-Cosetl, Migual Zeus Mora-Alvarez, Ruben Alejos-Palomares. Self-Parking System Based in a Fuzzy Logic Approach[C]. 2009 International Conference on Electrical,

- Communications, and Computers. 2009: 119-124.
- [17] G.G.Rigatos, S.G.Tzafestas, G.J.Evangelidis. Reactive Parking Control of Nonholonomic Vehicles via a Fuzzy Learning Automation[C]. Proceedings of the IEEE Control Theory Application. 2001: 148:169-179.
- [18] F.Gomez-Bravo, F.Cuesta, A.Ollero. Parallel and diagonal parking in nonholonomic autonomous vehicles[C]. Engineering Applications of Artificial,Intelligence, 2001, 14:419-434.
- [19] 杨昔阳, 尤晴曦, 李洪兴. 基于变论域理论的自动倒车控制[J]. 北京师范大学学报. 2005,第 41 卷第 4 期,348-350.
- [20] Jin Xu, G.Chen, Ming Xie. Vision-guided automatic parking for smat car[C].Proceedings of the IEEE Interlligent Vehicles Symosium, Dearbon, MI, 2000: 700-725.
- [21] [EB/OL] http://zh.wikipedia.org/wiki/Toyota_Prius
- [22] [EB/OL]<http://www.autohome.com.cn/329/>
- [23] [EB/OL]http://car.bitauto.com/leikesasils/?WT.srch=1&WT.mc_id=aladdin
- [24] [EB/OL] http://car.bitauto.com/mengdiou/?WT.srch=1&WT.mc_id=aladdin
- [25] 张浩杰. 不确定环境下基于启发式搜索的智能车辆路径规划研究[D].北京理工大学博士学位论文, 2013.3.
- [26] 关超华, 陈泳丹, 陈慧岩等. 基于改进 DBSCAN 算法的激光雷达车辆探测方法[J]. 北京理工大学学报. 2010,30(006):732-736.
- [27] Montemerlo M, Thrun S, Koller D, et al. FastSLAM: A factored solution to the simultaneous localization and mapping problem, 2002[C].
- [28] 杨丽莉. 基于激光雷达的智能车辆避障与导航技术[D]. 北京: 北京理工大学, 2008.
- [29] 富立, 范耀祖. 车辆定位导航系统[M]. 北京:中国铁道出版社, 2003.
- [30] 常青, 杨东凯. 车辆导航定位方法及应用[M]. 北京: 机械工业出版社, 2004.
- [31] Zhang L, Choi S, Park S. Robust ICP registration using biunique correspondence[C]. 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, Hangzhou, China, May 16, 2011 - May 19, 2011, 2011. IEEE Computer Society.
- [32] Guo R, Sun F, Yuan J. ICP based on Polar Point Matching with application to Graph-SLAM[C].
- [33] Grewal M S, Andrews A P, Bartone C G. Global Navigation Satellite Systems, Inertial

- Navigation, and Integration[M]. John Wiley & Sons, 2013.
- [34] Hofmann-Wellenhof B, Lichtenegger H, Wasle E. GNSS – global navigation satellite systems: GPS, GLONASS, Galileo, and more[M]. Springer, 2007.
- [35] Durrant-Whyte H, Bailey T. Simultaneous localization and mapping (SLAM): Part I[J]. IEEE Robotics and Automation Magazine, 2006,13(2):99-108.
- [36] Bailey T, Durrant-Whyte H. Simultaneous localization and mapping (SLAM): part II[J]. IEEE Robotics & Automation Magazine, 2006,13(3):108-117.
- [37] Brooks R A and Tomas L P. A subdivision algorithm in configuration space for find path with rotation[J]. IEEE Transactions on Systems, Man and Cybernetics. SMC-15(2): 225-233, Mar/Apr. 1985.
- [38] Teslic L, Krjanc I, Klancar G. Using a LRF sensor in the Kalman-filtering-based localization of a mobile robot[J]. ISA Transactions, 2010,49(1):145-153.
- [39] Teslic L, V S Krjanc I, Klancar G. EKF-based localization of a wheeled mobile robot in structured environments[J]. Journal of Intelligent and Robotic Systems, 2010,62(2):187-203.
- [40] Zhou W, Zhao C, Guo J. The study of improving Kalman filters family for nonlinear SLAM[J]. Journal of Intelligent and Robotic Systems: Theory and Applications, 2009,56(5):543-564.
- [41] Massimo Bertozzi, Alberto Broggi. GOLD: A Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection [J].IEEE Trans. On Image Processing, 1998, 7(1): 62-82.
- [42] 张勤, 曲宁玺, 李艳, 神谷好承. 自动泊车轨迹生成方法的研究[J]. 计算机仿真, 2010, 第 27 卷第 10 期, 148-151.
- [43] 周培云. 自主式智能车辆导航与路径规划技术研究[D]. 北京: 北京理工大学, 2009.
- [44] M. Likhachev, D. Ferguson, D. Gordon, A. Stentz and S. Thrun, Anytime Dynamic A*: An Anytime, Replanning Algorithm[C]. Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS), 262-271, 2005.
- [45] Mat B. 游戏人工智能编程案例精粹[M]. 北京: 人民邮电出版社, 2008.
- [46] Jonathan B, Tully F, Jim K, et al. Little Ben: The Ben Franklin Racing Team's Entry in the 2007 DARPA Urban Challenge[J].Journal of Field Robotics. 2008,25(9):598-614.
- [47] Chris U, Joshua A, Drew B, et al. Autonomous Driving in Urban Environments: Boss and the Urban Challenge [J]. Journal of Field Robotics. 2008,25(8):425-466.

- [48] Sebastian T, Mike M, Hendrik D, et al. Stanley The Robot that Won the DARPA Grand Challenge[J]. Journal of Field Robotics. 2006,23(9):661-692.
- [49] Stefan F. Campbell. Steering Control of an Autonomous Grand Vehicle with Application to the DARPA Urban Challenge[D].USA: Massachusetts Institute of Technology,2007.
- [50] Ming Feng H, Umit O, A path Following Control Algorithm for Urban Driving. IEEE Intelligent Vehicles Symposium [C]. Netherlands:IEEE.2008:227-231.
- [51] 齐建永. 智能车辆导航定位与自动引导技术研究[D]. 北京:北京理工大学,2008.
- [52] 满益明, 陈慧岩. 无人地面车辆的研究概况[C]. 中国汽车工程协会越野车分会. 中国西安, 2008: 71- 75.
- [53] Broggi A. Automatic Vehicle Guidance [M]. USA: World Scientific Publishing Company, 1999.
- [54] Ming Feng H, Umit O, A path Following Control Algorithm for Urban Driving[C]. IEEE Intelligent Vehicles Symposium. Netherlands:IEEE.2008:227-231.
- [55] Cem H, Ozguner U, Keith A. Redmill. Automated Lane Change Controller Design[C].IEEE Transactions on Intelligent Transportation System,2003,4(1):13-22.
- [56] Kristijan M, Roland P, Roland S. Path Following for Autonomous Vehicle Navigation with Inherent Safety and Dynamic Margin[C]. IEEE Intelligent Vehicle Symposium.USA:IEEE.2008:108-113.
- [57] Chen Q, Ozuguner U, Redill K. Ohio State University at the 2004 DARPA Grand Challenge: Developing a Completely Autonomous Vehicle, IEEE Intelligent Systems, Vol.19, No.5, pp.8-11, Oct.2004.
- [58] 李庆中, 顾伟康, 叶秀清等. 移动机器人路径跟踪的智能预瞄控制方法研究[J].机器人, 2002,24(3).
- [59] 孙振平, 安向京, 贺汉根. CITAVT-IV 视觉导航的自主车[J]. 机器人. 2002, 24(2): 115-121.
- [60] 王德平. 汽车驱动防滑控制与动力学稳定性控制的控制逻辑与算法研究[D]. 长春: 吉林工业大学, 1998.
- [61] 赵治国. 车辆动力学及其非线性控制理论技术的研究[D]. 西北工业大学, 2002.12.
- [62] 余志生. 汽车理论[M]. 北京: 机械工业出版社, 1999.
- [63] 王德平. 汽车驱动防滑控制与动力学稳定性控制的控制逻辑与算法研究[D].长春: 吉林工业大学, 1998.

- [64] 孙振平. 自动驾驶汽车智能控制系统[D]. 长沙:国防科学技术大学, 2004.
- [65] 张辉. 非完整性移动机器人体系结构设计与轨迹跟踪控制研究[D]. 长沙: 湖南大学,2007.