

分类号: U461

密 级: 公开

学校代码: 10363

学 号: 2210120172



安徽工程大学

Anhui Polytechnic University

硕士学位论文

题 目 基于 MPC 自动驾驶轨迹跟踪
控制研究

论 文 作 者 臧豫徽
校 内 导 师 杨爱喜
校 外 导 师 李兰友
专业学位类别 机械
研究方向(领域) 自动驾驶轨迹跟踪

论文提交日期: 2024 年 6 月 1 日

分类号：U461

单位代码：10363

密 级：公开

学 号：2210120172

基于 MPC 自动驾驶轨迹跟踪控制研究

RESEARCH ON MPC-BASED AUTONOMOUS DRIVING TRAJECTORY TRACKING CONTROL

学生姓名： 臧豫徽
校内导师： 杨爱喜
校外导师： 李兰友
专 业： 机械
研究方向： 自动驾驶轨迹跟踪

论文答辩日期：2024 年 05 月 25 日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名： 臧豫徽

日期：2024 年 06 月 01 日

安徽工程大学学位论文版权使用授权书

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，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 ☐，在_____年解密后适用本版权书

本学位论文属于

不保密 ☒。

学位论文作者签名：

臧豫徽

指导教师签名：

杨松

日期：2024 年 06 月 01 日

日期：2024 年 06 月 01 日

摘 要

现如今，汽车行业正经历前所未有的变革，智能化和网联化成为产业的重要发展趋势。自动驾驶技术作为汽车智能化的研究热点，对于提高交通安全、缓解交通压力以及提升出行体验具有重大意义。轨迹跟踪控制技术是衔接上层路径决策规划结果和底盘控制执行的桥梁，是实现自动驾驶的关键技术之一。模型预测控制(MPC)作为一种先进的控制策略，通过预测车辆未来一段时间内的动态行为，并结合预设的轨迹和约束条件，优化计算出当前时刻的最优控制输入。基于此，本文的研究工作将围绕 MPC 轨迹跟踪控制开展以下研究：

(1) 车辆运动学和动力学模型搭建。基于车辆三自由度“单车模型”建立车辆运动学和动力学模型。在动力学模型基础上构建以横向速度、纵向速度、横摆角、横摆角速度、横向位移和纵向位移为状态量，前轮转向角为控制量的车辆状态空间方程，为后续 MPC 轨迹跟踪控制器的设计提供了控制主体。

(2) LTV-MPC 轨迹跟踪控制器构建。在车辆三自由度动力学模型的基础之上，经过车辆非线性系统的线性化及离散化后，以控制增量作为控制变量来构建状态空间方程。最后添加关于车辆系统状态量及前轮转角控制增量的目标函数并添加相关约束，完成 LTV-MPC 轨迹跟踪控制器构建。

(3) 道路曲率自适应 LTV-MPC 轨迹跟踪控制器设计。基于目标函数权重矩阵对轨迹跟踪控制性能影响的考虑，分别对比分析目标函数权重矩阵 Q 、 R 因道路曲率变化而产生轨迹跟踪效果的差异，设计出道路曲率自适应函数。通过与传统 MPC 和 LTV-MPC 轨迹跟踪控制器进行对比，验证了改进后控制器在车辆跟踪精度、行驶稳定性上的提升。

(4) 基于 RLS 优化下的变预测时域控制器设计。首先，基于最小二乘法设计了侧偏刚度实时估计法，将实时计算的轮胎侧偏刚度输入至车辆动力学模型，从而避免车辆在工作在轮胎“非线性区域”时车辆稳定性和操控性的降低。其次，为进一步提高控制器在各速度工况下的适应能力，试验首先分析了车辆从低速到高速的阶梯速度下，分别在四组从低到高的预测时域下的轨迹跟踪效果。通过其控制变化规律拟合出变预测时域控制函数并验证其有效性。

(5) 实车验证 LTV-MPC 轨迹跟踪算法的可行性。详细阐述了实车验证实验相关

软硬件及其环境搭建准备过程。通过对比实车轨迹与参考轨迹的偏差以及车辆在行驶过程中的姿态变化，包括俯仰角、偏航角和翻滚角等，验证了 LTV-MPC 轨迹跟踪控制器在实车应用中的跟踪稳定性和控制精准度。

关键词：轨迹跟踪，MPC，曲率自适应，最小二乘法，变预测时域

Abstract

With the continuous advancement of electrification and intelligence in automobiles, along with the sustained growth of automobile ownership, the automotive industry is undergoing unprecedented transformation. Intelligentization and connectivity have become important trends in the development of the automotive industry. Autonomous driving technology, as a research hotspot, holds significant implications for enhancing traffic safety, alleviating traffic congestion, and improving travel experiences. Trajectory tracking control technology serves as a crucial link between upper-level path decision-making and chassis control execution, making it one of the key technologies for realizing autonomous driving. Model Predictive Control (MPC), as an advanced control strategy, predicts the dynamic behavior of the vehicle over a future time horizon, and optimizes the current optimal control input based on predefined trajectories and constraints. Therefore, the research in this paper mainly focuses on MPC trajectory tracking control, and the following research aspects are conducted:

(1) The vehicle kinematic and dynamic models were constructed by comprehensively considering factors such as model accuracy and controller complexity. Based on the three-degree-of-freedom "single-track model" of the vehicle, kinematic and dynamic models were established. Building upon the dynamic model, the vehicle state space equations were formulated with lateral velocity, longitudinal velocity, yaw angle, yaw rate, lateral displacement, and longitudinal displacement as state variables, and front wheel steering angle as the control variable. This provides the foundation for the subsequent design of the MPC trajectory tracking controller.

(2) The construction of the MPC trajectory tracking controller begins with the establishment of the vehicle's three-degree-of-freedom dynamic model. After linearizing and discretizing the nonlinear vehicle system, considering the controllability of the controller and the stability of vehicle operation, the control variable of the controller is transformed into the form of control increments for the construction of state space equations. Finally, the objective function concerning the vehicle system's state variables

and the control increments of the front wheel steering angle is formulated, along with the addition of relevant constraints, completing the construction of the MPC trajectory tracking controller.

(3) The design of a road curvature adaptive MPC trajectory tracking controller begins with a detailed description of the construction process of the joint simulation platform using Simulink and CarSim. Considering the impact of the weighting matrices of the objective function on the trajectory tracking performance, the effects of the weighting matrices Q and R on trajectory tracking due to road curvature variations are analyzed separately. Subsequently, a road curvature adaptive function is designed. Finally, through comparison with traditional MPC and LTV-MPC trajectory tracking controllers, the improved controller's enhancements in vehicle tracking accuracy and driving stability are validated.

(4) Based on the optimized RLS, a variable prediction horizon controller is designed. Firstly, considering the significant impact of tire linear lateral stiffness on the accuracy of vehicle dynamic model construction, especially in complex conditions such as high speed, a real-time estimation method for lateral stiffness is designed based on the least squares method. The real-time calculated tire lateral stiffness is then inputted into the vehicle dynamic model, thereby avoiding the reduction in vehicle stability and controllability when the vehicle operates in the "nonlinear region" of the tire. Secondly, to improve the adaptability of the MPC trajectory tracking controller under various speed conditions, experiments analyze the trajectory tracking effects under step speeds from low to high speeds. The control variations are then fitted to derive the function for variable prediction horizon control. Finally, the effectiveness of the designed variable prediction horizon controller is analyzed and validated by comparing it with the fixed prediction horizon controller using indicators such as lateral tracking error, yaw angle error, yaw rate, and center of mass lateral deviation angle.

(5) Feasibility verification of the MPC trajectory tracking algorithm was conducted through real vehicle experiments. The detailed preparation process involved the setup of vehicle-related hardware and software as well as the environment for the experiments. In the ROS/Autoware framework, parameters and information related to the MPC trajectory

tracking algorithm, such as prediction horizon, control interval, and vehicle dynamic model, were configured. The `mpc_follower` node was utilized to compute the optimal vehicle control commands, which were then transmitted to the `/vehicle_cmd` topic. Subsequently, the control signals were sent to the chassis via PCAN to control the vehicle. Based on this setup, road trajectory data was collected using the real vehicle, and the tracking performance of the MPC trajectory tracking controller was validated using this trajectory.

Keywords: Trajectory tracking, MPC, Curvature adaptive, Least squares algorithm, Time-Varying prediction horizon

目 录

摘 要	I
Abstract	III
目 录	VI
第 1 章 绪 论	1
1.1 课题背景及研究意义	1
1.2 自动驾驶国内外研究现状	2
1.2.1 国外发展现状	2
1.2.2 国内发展现状	5
1.3 轨迹跟踪技术研究现状	8
1.4 本文主要研究内容	13
1.4.1 课题面临的问题分析	13
1.4.2 课题研究章节安排	14
第 2 章 车辆运动学与动力学模型	16
2.1 车辆运动学模型	16
2.2 车辆动力学模型	18
2.3 本章小结	20
第 3 章 曲率自适应 LTV-MPC 轨迹跟踪研究	21
3.1 模型预测控制原理	21
3.2 LTV-MPC 轨迹跟踪控制器设计	22
3.2.1 车辆非线性系统线性化	22
3.2.2 车辆系统离散化	23
3.2.3 预测状态方程与输出模型的构建	24
3.2.4 目标函数设计	25
3.2.5 反馈矫正	26
3.3 曲率自适应控制器设计	27
3.3.1 联合仿真平台介绍	27
3.3.2 仿真实验设计及分析	28
3.3.3 权重矩阵对跟踪效果的影响分析	31
3.3.4 自适应权重系数矩阵的拟合	32
3.4 曲率自适应 LTV-MPC 控制器对比仿真实验	33
3.5 本章小结	36
第 4 章 基于 RLS 的变预测时域 LTV-MPC 的算法研究	37

目 录

4.1 轮胎侧偏刚度估计	37
4.1.1 递推最小二乘法公式	37
4.1.2 实时轮胎侧偏刚度的估计	38
4.1.3 轮胎侧偏刚度的仿真验证	40
4.2 变预测时域 LTV-MPC 跟踪控制器设计	43
4.2.1 探究车速与预测时域 N_p 对跟踪效果的影响	43
4.2.2 变预测时域控制器函数的拟合	48
4.3 实验结果及分析	49
4.5 本章小结	51
第 5 章 自动驾驶轨迹跟踪实车实验与分析	52
5.1 实验平台搭建	52
5.1.1 硬件平台搭建	52
5.1.2 软件系统平台	58
5.2 轨迹跟踪实车实验	59
5.2.1 点云地图建立	59
5.2.2 实车实验	60
5.3 车辆实验结果	63
5.4 本章小结	65
第 6 章 结论与展望	66
6.1 研究成果总结	66
6.2 未来研究工作展望	67
参考文献	68
攻读硕士学位期间承担的科研任务与主要成果	73
致 谢	74

第1章 绪论

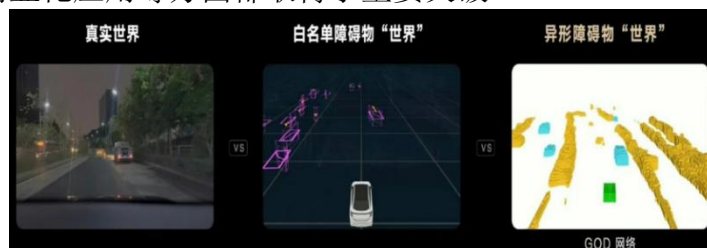
1.1 课题背景及研究意义

随着现如今科技的飞速发展以及汽车保有量的不断增多，汽车行业正处于百年未有之大变局，汽车产业正逐渐朝着智能化和网联化不断迈进^[1,2]。其中自动驾驶技术已成为全球交通领域的研究热点。其发展背景深厚，现状亦日新月异，对于提高交通安全、缓解交通压力以及提升出行体验具有重大意义。自动驾驶技术的兴起主要得益于多个领域的科技进步。首先，计算机视觉和传感器技术的突破为自动驾驶提供了强大的感知能力^[3]，通过激光雷达、摄像头、毫米波雷达等传感器，自动驾驶模块能够实现对采集到的传感器信息进行融合处理以实现对周围环境的精准感知和识别。其次，大数据和人工智能技术的崛起为自动驾驶提供了强大的决策和学习能力。通过对海量数据的分析和学习，自动驾驶系统能够不断优化自身的驾驶策略，提高行驶的安全性和舒适性。此外，政策法规的逐步完善也为自动驾驶技术的商业化应用提供了有力保障^[4]。

目前自动驾驶技术已经取得了显著的进展。以特斯拉为例，其自动驾驶系统 FSD 已经实现在高速公路和部分城市道路上的自动驾驶功能，如图 1-1(a)所示。该系统通过高精度地图、传感器融合以及深度学习等技术，实现对车辆周围环境的实时感知和判断，完成对车辆的自主控制。华为推出的自动驾驶解决方案 ADS2.0，通过深度融合包括云原生、数字孪生、大型模型以及意图驱动等，成功构建端到端的全面自动化和高度智能化能力，如图 1-1(b)所示。此外，谷歌的 Waymo、百度的 Apollo 等也在技术研发、实测验证以及商业化应用等方面都取得了重要突破。



(a) Tesla 自动辅助驾驶



(b) 华为 ADS2.0

图 1-1 Tesla、华为自动驾驶技术

轨迹跟踪技术作为其核心技术之一，在连接车辆底盘和上下层信息发挥着至关

重要的作用^[5]。在现代自动驾驶系统中，轨迹跟踪技术是实现车辆精确控制、保障行驶安全以及提升行驶效率的关键环节，而车辆底盘作为执行机构，直接负责车辆转向、驱动和制动等操作^[6]。因此，轨迹跟踪技术与车辆底盘之间的紧密配合至关重要。首先，轨迹跟踪技术通过实时感知和计算车辆的实际运动状态与预期运动轨迹之间的偏差，生成相应的控制指令，这些指令需要准确地传递给车辆底盘执行机构，以实现车辆的精确控制^[7]。以上要求轨迹跟踪技术与车辆底盘之间必须建立高效、准确的信息传输机制，确保控制指令的实时性和准确性。其次，车辆底盘在执行控制指令的过程中，会产生大量实时数据，如车速、转向角度、加速度等，这些数据对于上层决策系统来说具有重要参考价值^[8]。轨迹跟踪技术将实时获取并处理这些数据，对车辆的实际运动状态进行准确评估，根据评估结果调整控制策略。因此，轨迹跟踪技术不仅需要将控制指令传递给底盘，还需要从底盘获取实时反馈数据，实现双向的信息交互。

模型预测控制作为一种先进的控制策略，基于预测车辆未来一段时间内的动态行为，并结合预设的轨迹和约束条件，优化计算出当前时刻的最优控制输入。这种基于模型预测和优化求解的控制方式，使得 MPC 控制器在处理复杂动态环境和多变驾驶任务时具有独特的优势^[9]。基于此，本文的研究工作便主要围绕 MPC 轨迹跟踪控制研究展开。

1.2 自动驾驶国内外研究现状

1.2.1 国外发展现状

自动驾驶汽车在国外的发展历程可以追溯到 20 世纪初，随着计算能力、传感器技术、人工智能和机器学习领域的突破，自动驾驶技术的研究和开发飞速进步^[10]。

在 1989 年，CMU 的研究人员开发了一种名为 ALVINN (Autonomous Land Vehicle In a Neural Network) 的神经网络系统，它能够通过学习驾驶员的行为来控制车辆。ALVINN 是最早将机器学习技术应用于无人驾驶汽车的例子之一。

2002 年美国国防高级研究计划局 (DARPA) 开始牵头举办 DARPA 挑战赛，极大地推动了无人驾驶汽车技术的发展。来自各个研究机构和私人企业的团队参与了比赛，展示了各种自动驾驶技术，如图 1-2 所示。



图 1-2 DARPA 无人驾驶挑战赛车辆

2015 年，特斯拉推出了全球首款商业化辅助驾驶系统 Autopilot，每一辆特斯拉汽车，都搭载 1 个前向的毫米波雷达、1 个前视摄像头、12 个超声波雷达、1 套高精度电子辅助制动系统（线控刹车）以及一块搭载 Mobileye EyeQ3 芯片的主板，这套硬件被称为 Autopilot Hardware 1.0。在特斯拉 2.0 Autopilot 中，包含 8 个摄像头，1 个毫米波雷达，12 个超声波雷达，以及一块搭载英伟达 DRIVE PX2 AutoCruise 定制版的计算主板。2017 年，特斯拉发布的 Model X, Model S 都已经配备了 Autopilot2.0 系统，2019 年 4 月 FSD 芯片问世，特斯拉进入 Autopilot 3.0 时代，从功能角度来看，除了原有功能，Autopilot 3.0 新增了增强型可视化，特斯拉的完全自动驾驶是基于视觉摄像头和 AI 的神经网络算法来实现的。目前，特斯拉引入了先进的 Bird's Eye View(BEV)算法。其实现方法是将原本摄像头 2D 视角通过算法校正和改变，形成基于鸟瞰视角的俯视图，图 1-3 为 Tesla 采用的 BEV+Transformer 基本流程概念。

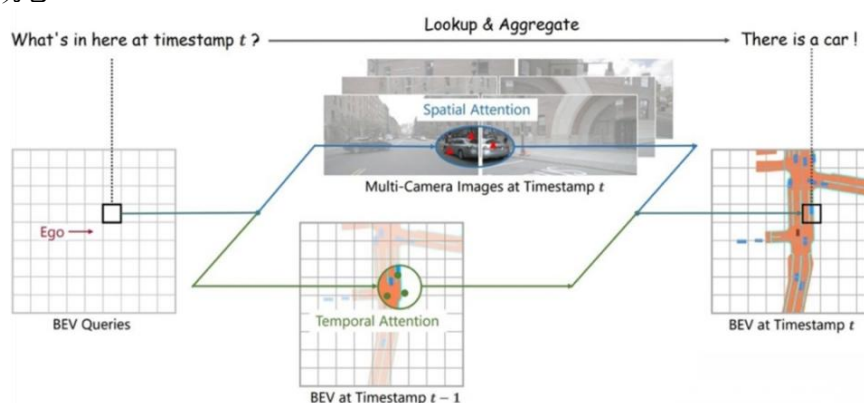


图 1-3 BEV+Transformer 基本概念

Waymo 的自动驾驶系统采用包括视觉相机、激光雷达和超声波传感器相互融合，

能够为车辆提供 360 度的环境感知能力。此外，Waymo 利用先进的深度学习算法和大数据，使车辆能够识别和预测周围物体行为。落地应用方面，Waymo 自动驾驶车辆已在超过 25 座城市的公共道路上累计超过 2000 万英里的自动驾驶实际测试，并辅以高达 100 亿英里的模拟测试里程。进入 2022 年，Waymo 与 Cruise 获自动驾驶客运服务许可，这标志着它们在自动驾驶技术的商业化探索道路上迈出了重要一步。这一系列的进展充分展现了自动驾驶技术在公共出行领域的广阔应用前景和巨大潜力。下图 1-4 为 Waymo 自动驾驶汽车。



图 1-4 Waymo 自动驾驶汽车

瑞典在无人公交、无人卡车货运、无人清扫等自动驾驶应用领域成果突出。2023 年 6 月，瑞典自动驾驶卡车企业 Einride 与挪威邮政服务机构 PostNord 签订了合作运营协议，当月部署 6 辆 Einride 卡车至 2024 年 6 月部署 35 辆电动卡车。同时英国、德国、芬兰、苏格兰等地都已在城市试点无人公交车，目前多家企业正在联合开发“欧洲地平线 ULTIMO”自动驾驶电动小巴服务，预计到 2025 年，45 辆无人驾驶电动小巴将在瑞士日内瓦、德国克罗纳赫和挪威奥斯陆这三个欧洲城市运行一年。在自动驾驶应用方面，日本是明确将卡车编队行驶列入国家规划并推进开展的国家之一。自 2022 年发布《报告书草案 Ver.6.0》中提出将构建高级干线物流系统，目标是实现卡车编队行驶、L4 级卡车与物流站点无缝协作的高级干线运输。2023 年 5 月，日本开始制定最新的“mobility roadmap”，计划至 2024 年在新东名高速高速设置 100km 以上自动驾驶车辆专用车道，并且计划 2026 年将允许 L4 级完全自动驾驶卡车商业运营。

自动驾驶法规方面，2023 年 4 月 1 日，日本《道路交通法》修正案正式实施，开始解禁在路线和车速等特定条件下的 L4 级自动驾驶。一是在自动驾驶公交车方面，允许在人口稀少的区域，将 L4 级自动驾驶系统应用于公共交通的特定路线，未来还可以应用于高速公路的私家车和卡车。二是在无人配送方面，允许通过远程操控的

自动送货机器人按照行人规则以低于 6 公里每小时的速度在公共道路上行驶，日本政府计划在 2025 年将此修正案适应的地区扩大到全国 40 个地方。2023 年 11 月 7 日，英国正在制定一项名为《自动驾驶汽车法案》的新法案，旨在明确自动驾驶汽车事故责任应由车企承担，该法案只适用于 L3 级以上的自动驾驶车型。

1.2.2 国内发展现状

我国在自动驾驶技术方面的研究起步稍晚，于 20 世纪 80 年代开始起步。在国防科工委和国家 863 计划的资助下，清华大学自 1988 年开始研究开发 THMR 系列智能车如图 1-5 所示，THMR-5 智能车能够实现结构化环境下的车道线自动跟踪^[11]。



图 1-5 清华大学智能车 THMR-5

国防科技大学从 20 世纪 80 年代末开始先后研制出基于视觉的 CITAVT 系列智能车辆。直到 1992 年，中国首辆真正意义上的无人驾驶汽车才在国防科技大学的努力下成功问世。2011 年 7 月，一汽集团与国防科技大学携手，共同研发的红旗 HQ3 无人驾驶汽车，在高速公路上进行了全长 286 公里的无人驾驶测试，如图 1-6 所示。这一创举标志着中国自主研发无人驾驶汽车技术的重大突破，彰显了国家在复杂交通环境下自动驾驶技术的新高度^[12]。据实验当天的详细数据显示，尽管面临复杂多变的路况和交通环境，但在整个驾驶过程中，人工干预的次数仅为 10 次，累计里程也仅为 2.24 公里。这一数据充分表明，人工干预在这次自动驾驶实验中所占的比例极低，不到总里程的百分之一，从而进一步验证了该无人驾驶系统的高度自主性和可靠性。



图 1-6 红旗 HQ3

在 2012 年，军事交通学院成功研发“军交猛狮III号”，该车辆能够在自动驾驶模式下行驶 114 公里，且最高时速可达 105 公里/小时。随后几年，我国在自动驾驶领域的研究取得了显著的进展，在 2015 年 8 月，宇通生产的大型客车在郑开大道城铁贾鲁河站启动，于完全开放的公路环境中圆满进行了自动驾驶的测试。此次实验标志着国内首次大型客车在高速公路上的自动驾驶实践。2017 年底，百度携手雄安新区共同推进自动驾驶智能试点项目。进而在 2018 年 5 月，百度在雄安开展了 L4 级别的自动驾驶路测活动，进一步推动了自动驾驶技术在我国的发展与应用。2019 年，百度在北京举办的开发者大会上发布了 Apollo 3.5 版本，该版本包括了全新的感知、规划、控制等模块，为自动驾驶技术的发展提供了更多的可能性^[13]。



图 1-7 Apollo Go Robotaxi

2020 年 3 月，百度 Apollo 计划宣布正式开放 Apollo Go Robotaxi，成为中国首个商用自动驾驶出租车服务。2021 年，小鹏汽车推出车型 P7 搭载高阶驾驶辅助功能，并在 3000 公里自动驾驶类功能测试中实现平均百公里接管次数 0.71 次，这些测试不仅包括在公路上的行驶状况，还有在复杂的城市交通环境中的应对能力。小鹏自主研发了一套名为 XNGP 的高级驾驶辅助系统，随着版本迭代，XNGP 系统不断升级，增加了更多高阶辅助驾驶功能，例如自动变道、自动泊车、红绿灯识别等。通过不断地数据收集和机器学习，XNGP 的性能在持续提升，如图 1-8 为小鹏车型运行时 XNGP 运行界面。



图 1-8 小鹏 XNGP 运行界面

2022 年，华为宣布推出自主研发的智能驾驶计划，将人工智能技术应用于智能

驾驶系统中，通过深度学习、机器学习等方法，使车辆能够实现环境感知、决策和控制等功能的部署。其包括使用摄像头、雷达、激光雷达等传感器获取环境信息，并通过人工智能算法进行处理和分析，以实现智能驾驶的目标。华为 5G 技术可以提供高速、低延迟的通信连接，为智能驾驶系统提供必要的数据传输支持。这包括车辆之间的通信（V2V）和车辆与基础设施的通信（V2I），以及车辆与云端的通信（V2C）等。通过整合人工智能、5G 通信等先进技术，为智能汽车行业提供全面的解决方案，推动智能驾驶技术的发展和应用，下图 1-9 展示了自动驾驶系统框架。

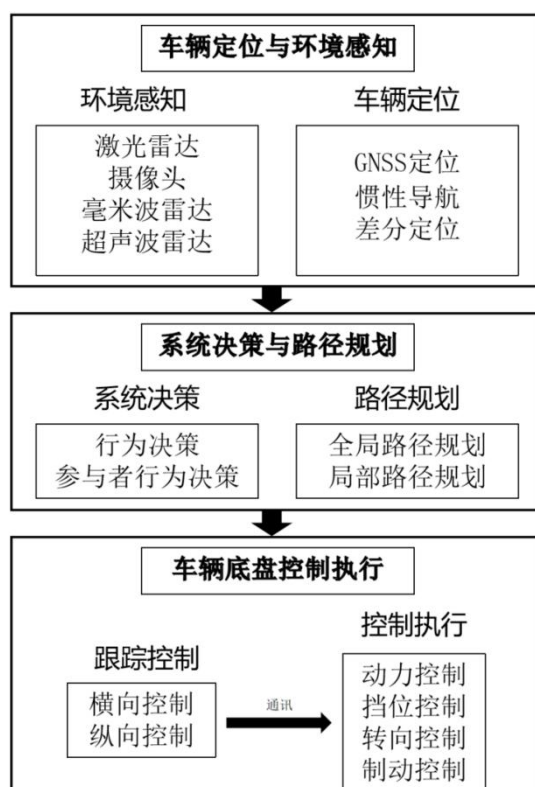


图 1-9 自动驾驶系统框架

2023 年 11 月，由工业和信息化部，公安部，住房和城乡建设部，交通运输部四部门联合发布的《关于开展智能网联汽车准入和上路通行试点工作的通知》将在智能网联汽车道路测试与示范应用工作基础上，遴选具备量产条件的搭载自动驾驶功能的智能网联汽车产品开展准入试点，而取得准入的智能网联汽车产品可在限定区域内开展上路通行试点，通知中的智能网联汽车，指的就是配备国家标准《汽车驾驶自动化分级》定义的 3 级驾驶自动化（有条件自动驾驶）和 4 级驾驶自动化（高度自动驾驶）功能的车辆^[14]。意味着 L3 级和 L4 级自动驾驶汽车可以在试点公共开放路段合法上路进行测试和运营并行。为相关自动驾驶企业投入无人驾驶技术的研

究与开发提供了法律和制度的保障。

1.3 轨迹跟踪技术研究现状

自动驾驶技术包括环境自主感知、规划决策和底盘控制三大核心模块。轨迹跟踪控制是衔接上层路径决策规划结果和底盘控制执行的桥梁，是实现自动驾驶的关键技术之一。轨迹跟踪系统的主要功能是接收并执行感知和决策规划模块提供的指令信息，使自动驾驶车辆沿着预规划的轨迹行驶，并且在行驶过程中保证轨迹跟踪精度和车身的控制稳定性^[15]。目前，轨迹跟踪算法在国内外均得到广泛的研究，但仍然存在一些挑战和开放性问题。

对于自动驾驶的轨迹跟踪运动控制来说，在实现过程具有以下难点。首先由于车辆是一个高度耦合的复杂系统且在各种工况下行驶时动力学模型往往非常复杂且存在不确定性，如车辆质量、空气动力学效应、轮胎与地面之间的摩擦等。这些因素都会影响车辆行驶性能和轨迹跟踪精度，所以如何建立准确且适应性强的车辆模型是轨迹跟踪控制的核心难点之一^[16]。其次自动驾驶车辆需要实时感知周围环境的信息，包括道路状况、交通信号、障碍物等，并对这些信息进行处理和预测。然而，由于传感器噪声、数据融合误差以及复杂多变的交通环境等因素，环境感知与预测的准确性往往受到限制。在跟踪的实时性方面，自动驾驶系统需要在毫秒级的时间内完成环境感知、决策规划和轨迹跟踪等任务^[17]。然而，由于车载计算资源的有限性，故控制器最应在保证实时性的前提下实现高性能的轨迹跟踪控制。

目前关于自动驾驶轨迹跟踪的理论主要有模型预测控制(MPC)控制方法^[18-22]、深度强化学习控制方法^[23]、PID 控制技术、线性二次型(LQR)控制技术^[24]、纯跟踪(Pure Pursuit)控制技术^[25]、滑膜控制技术^[26]等。其中模型预测控制器是一种基于系统模型的控制方法，通过综合预测模型、系统当前时刻的状态量以及未来时刻的控制量构建代价函数，同时考虑车辆的动力学约束进行在线滚动优化，对预测结果进行修正，为自动驾驶车辆提供更加精确和稳定的轨迹跟踪控制^[27,28]。

(1) PID 控制器

PID 控制器通过对当前车辆状态误差（目标轨迹与实际轨迹之间的差距）、过去车辆状态误差的积分和误差变化率的微分进行加权求和，来计算控制输出。对于 PID 轨迹跟踪控制器的实现过程，首先，重点为确定车辆轨迹跟踪的目标，即车辆如何行驶以跟踪给定的轨迹^[29]。然后，需要设计 PID 控制器的架构。PID 控制器由比例

(P)、积分(I)和微分(D)三个部分组成。比例部分通过测量当前误差量以产生控制输出；积分部分通过积累历史误差来消除系统的稳态误差^[30]；微分部分通过测量误差变化率来预测未来的误差。接下来，确定PID控制器的参数，包括比例系数、积分系数和微分系数^[31]。这些参数根据车辆的动态特性和轨迹跟踪性能的要求进行调整。控制器构建过程是一个迭代的过程，需不断地调整PID控制器的参数和逻辑，以使其能够准确地跟踪预设车辆轨迹。通过合理地设计和调整PID控制器，实现对车辆轨迹的跟踪和平稳性控制^[32]。

Zhao^[33]等将PID算法与神经网络进行融合，应用于无人驾驶车辆横向控制，相对于单PID控制器，融合算法能够更好的改善系统的动态特性。Nie^[34]等对车身电子稳定系统的控制算法进行改进，提出一种基于模糊PID控制器，在高速行驶工况下，能更好的提高车辆行驶的稳定性。Zhang^[35]采用基于BP神经网络优化的PID控制算法进行纵向控制，能够自动调整PID控制参数，达到最优的控制效果。

(2) 线性二次型(LQR)

LQR控制器在实现轨迹跟踪控制过程中，首先要对车辆进行动力学建模，包括建立车辆的状态空间模型，描述车辆在不同控制输入下的运动行为。状态空间模型包括位置、速度、方向等车辆的状态变量和前轮转向角控制输入。随后，设计二次型目标函数来评价轨迹跟踪的性能，目标函数中包括状态偏差的加权和控制输入的加权。目标是 최소화目标函数，以使车辆能够在跟踪期望轨迹的同时保持稳定和平滑的行驶。接着，利用LQR控制器设计状态反馈控制规律，通过求解Riccati方程计算最优的状态反馈增益矩阵。这个增益矩阵将用于计算控制输入，以使系统状态朝着期望轨迹方向演变。计算得到的最优状态反馈增益矩阵将输入至状态反馈控制规律之中。此过程需要在实时系统中实现，以确保车辆能够在每个控制时刻持续地跟踪期望轨迹，将状态反馈控制器与车辆的动力学模型相结合，同时车辆状态量将被测量并反馈给控制器，形成闭环控制系统。最终，使用CarSim仿真工具或实际车辆进行测试和调试。通过车辆的状态反馈验证控制器的性能，并对参数进行调整，以确保系统达到最佳的轨迹跟踪性能。

Zhou^[36]等将多点预瞄驾驶员模型与LQR算法进行结合，在不同仿真工况下进行验证算法的可行性，仿真结果表明该控制算法能够实现较好的跟踪效果。Gu^[37]等采用遗传算法优化LQR算法，得到最优Q和R矩阵，优化后的LQR算法能够获得更好的控制效果。

(3) 滑模控制

滑模控制是一种用于实现车辆轨迹跟踪的控制方法，其基本思想是设计一个滑模面，使得系统状态能够快速进入并保持在滑模面上^[38]，通过控制输入使系统状态沿着滑模面快速运动，实现对期望轨迹的跟踪^[39]。对于模糊控制的实现，首先需要建立车辆关于位置、速度、方向状态变量和控制转向角输入。接着，设计一个超平面滑模面，将系统状态空间分成两个区域：滑模面上方和滑模面下方。滑模面的选择将基于车辆的状态变量和期望轨迹，以确保在滑模面上的状态变量能够实现期望轨迹的跟踪。随后设计滑模控制律，滑模控制律是一种通过调整控制输入使系统状态始终保持在滑模面上的控制策略。这个过程选择合适的控制策略，以实现滑模面上系统状态的控制。最后，将设计的滑模控制律应用于车辆的控制系统中。这需要在实时系统中实现滑模控制器，以确保车辆能够持续地跟踪期望轨迹。

在控制过程中，需考虑系统的动态特性和实际操作的限制。实施闭环控制，将滑模控制器与车辆的动力学模型相结合，形成闭环控制系统。这样，车辆的状态量将被测量并反馈给控制器，实现对期望轨迹的跟踪。这个过程需要确保控制系统的稳定性和鲁棒性，以应对外部干扰和系统参数变化。

Zhang^[40]等在强鲁棒性滑模控制算法的基础上，设计了一种用于车辆横向控制的轨迹跟踪控制器，提出以增益切换为主要目标的模糊控制器概念以消除系统的抖动现象，并通过定曲率和双移线两种试验工况在联合仿真平台上验证了控制器的性能。Gilles Tagne^[41]等在基于高阶滑膜控制算法的基础上设计了一种可以分别针对车辆横向误差和航向误差的路径跟踪控制器，并且分别在在联合仿真试验平台和实车试验平台上进行了验证。但是滑膜控制的系统易发生抖振，这使其广泛应用受到限制。

(4) 纯跟踪控制(Pure Pursuit)

纯跟踪控制的核心思想是通过在车辆前方设定一个预瞄点，根据该点与车辆当前位置的关系计算出一个合适的转向角，使车辆能够沿着预定的轨迹行驶。对于纯跟踪控制在实现轨迹跟踪控制过程，纯跟踪控制的实现需要实时获取车辆的运动状态信息，包括位置、速度和航向角等^[42]。接下来，根据获取的车辆状态信息和预定的目标轨迹，确定一个预瞄点。预瞄点通常选择在车辆前方一定距离处的目标轨迹上，距离的大小可以根据车辆的行驶速度和转弯半径等因素进行调整。预瞄点的选择对于纯跟踪控制的性能具有重要影响，过近的预瞄点可能导致车辆反应过于敏感，而过远的预瞄点则可能导致车辆跟踪轨迹时产生较大的偏差。确定预瞄点后，需计

算车辆当前位置到预瞄点的向量与车辆当前航向间的预瞄偏差角。预瞄偏差角反映了车辆当前航向与目标轨迹之间的差异。然后,根据预瞄偏差角和车辆的行驶速度,利用纯跟踪控制算法计算出最优转向角。求解过程中,纯跟踪控制通过不断获取车辆的运动状态信息、确定预瞄点、计算预瞄偏差角、利用算法计算转向角以及实施控制输入等步骤,实现对车辆轨迹跟踪运动的精确控制。这种控制方法具有实现简单、计算量小的优点,并且在一定条件下能够实现较好的轨迹跟踪效果。但存在高速或复杂环境下的轨迹跟踪性能可能会受到限制的局限性。因此,在实际应用中需要根据具体场景和需求选择合适的轨迹跟踪控制方法。

Xu^[43]利用模糊控制器对纯跟踪算法进行改进,使其能够自适应调节前视距离,极大提升了轨迹跟踪效果。Hu^[44]等采用纯跟踪算法对不同曲率路径跟踪时,出现道路适应能力弱和跟踪精度差的问题,提出一种基于代价的滚动预瞄模型(rolling preview model, RPM),以提高纯跟踪算法跟踪精度与鲁棒性。

(5) 强化学习

强化学习在实现轨迹跟踪控制过程中的构建过程是连续且动态的过程。首先建立环境模型,模型能够准确地模拟车辆的动力学行为、感知系统、控制系统以及外部环境的影响。随后确定车辆轨迹跟踪任务的状态空间(车辆位置、速度、方向)和动作空间(转向角度、加速度)。然后,设计奖励函数,用于评估车辆每个时间步的行为。奖励函数可以有效地引导车辆学习到正确的轨迹跟踪策略^[45]。同时奖励函数会鼓励车辆尽可能接近期望轨迹、保持稳定行驶以及避免不良行为。根据任务的特性和要求,选择合适的强化学习算法。常用的算法包括 Q-learning、Deep Q-Network (DQN) 和 Policy Gradient 等。利用选择的强化学习算法,在环境模型中训练一个智能体,使其学习到最优的轨迹跟踪策略。在训练过程中,智能体通过与环境的交互来不断调整策略,实现最大化累积奖励。对训练过程进行优化,包括调整学习率、探索策略、奖励函数等参数,以提高训练效率和性能^[46]。

在训练结束后,评估训练得到的智能体在轨迹跟踪任务中的性能。以检验智能体学到的策略在实际场景中的适用性和鲁棒性。根据评估结果,对训练过程和策略进行改进。这包括重新设计奖励函数、调整参数、增加状态特征,以进一步提高轨迹跟踪性能。通过以上流程,可构建一个完整的基于强化学习的轨迹跟踪控制系统,实现车辆对期望轨迹的准确跟踪。这个系统将车辆的运动状态与强化学习算法相互联系起来,使车辆能够通过与环境交互来不断优化轨迹跟踪策略^[47]。

Wang^[48]等人基于强化学习的变参数模型预测控制(MPC)算法多目标控制策略,实现智能车辆路径跟踪控制系统的参数自适应整定并取得较好的控制精度。Li^[49]等人提出了一种基于蒙特卡洛-深度确定性策略梯度(MC-DDPG)的无人驾驶汽车避障跟踪控制算法。该算法基于深度强化学习网络搭建控制系统模型,在策略学习采样过程中采用优秀的训练样本,使用蒙特卡洛方法优化网络训练梯度,对算法的训练样本采取优劣区分,使用优异的样本通过梯度算法寻找最优的网络参数,从而增强网络算法的学习能力,实现无人驾驶汽车的更优连续控制。

(6) 模型预测控制

模型预测控制(Model Predictive Control, MPC)是一种广泛应用于轨迹跟踪技术中的先进控制方法。在当前的研究现状中,MPC技术在车辆轨迹跟踪领域得到了广泛应用和深入研究^[50]。MPC的研究现状涵盖了多个方面,这些研究主要集中在如何建立更准确的车辆模型、如何设计更有效的优化算法以及如何处理复杂环境下的轨迹跟踪问题等方面。

在车辆模型方面,学者们提出了单轮模型、双轮模型、三自由度模型等不同自由度的车辆模型,以适应不同精度和复杂度的轨迹跟踪需求。其中,车辆三自由度模型考虑了车辆在横向、纵向和车身旋转三个方向上的运动或受力。这种模型更加全面地描述了车辆的运动行为,能够更准确地预测车辆的行驶轨迹和动态响应。在需要更高精度的轨迹跟踪任务中得到广泛应用^[51]。对于更高速或更复杂的行驶环境,需要采用更高自由度的模型。

在优化算法方面,学者们致力于提高MPC的计算效率和实时性。包括改进优化求解算法、采用更高效的数值计算方法以及利用并行计算等技术手段。此外,还有一些研究关注如何将机器学习和人工智能方法与MPC相结合,以实现更智能的轨迹跟踪控制。在复杂环境下的轨迹跟踪问题方面,学者们面临着诸多挑战。例如,如何处理道路曲率变化、行驶工况时变、车辆模型构建精度较低、交通流影响以及传感器噪声等问题。为了解决这些问题,研究者们提出了一系列改进方法^[52],如引入道路预瞄信息、融合多传感器数据以及设计鲁棒性更强的MPC控制器等^[53]。

Song^[54]等基于二自由度车辆运动学模型,设计了时变模型预测控制器,探究了该控制器在不同路况下的轨迹跟踪性能。Ji^[55]等采用了多约束模型预测控制系统,同时利用二次规划算法求解控制器目标函数,实现了车辆在换道工况下的跟踪性能与车身稳定性控制。Sun^[56]将行驶稳定性因素作为控制目标,通过将前轮转角作为输入

仿真系统的控制量设计 MPC 控制器实现车辆的目标轨迹跟踪。Guo^[57]等将车身横摆角速度误差作为控制器的状态量，同时将车身边界约束和路况信息引入控制器的设计来提升 MPC 轨迹跟踪控制器的精度。Luo^[58]采用 BP 神经网络对基于 MPC 算法的横向跟踪控制策略进行优化，使其能够自适应选取时域参数，提高了横向跟踪精度和跟踪稳定性。Fan^[59]等基于 MPC 算法设计轨迹跟踪控制器，利用模糊控制算法优化预测时域和控制时域，在中高速行驶工况下所设计的控制器能够较好的进行轨迹跟随。Bai^[60]等人针对换道轨迹规划场景，基于微分平坦理论建立轨迹规划模型，利用 MPC 算法进行轨迹跟踪，取得不错的换道效果。

总体而言，MPC 技术在轨迹跟踪控制方面已经取得显著进展，并在实际应用中展现出良好的性能。然而，随着自动驾驶技术的不断发展和应用场景的不断拓展，MPC 在轨迹跟踪控制中仍面临许多新的挑战和问题，仍需要研究者们继续深入探索和研究。

1.4 本文主要研究内容

1.4.1 课题面临的问题分析

基于 MPC 轨迹跟踪控制器的研究现状，现将部分亟待解决的问题总结如下：

(1) 道路曲率变化适应性问题

当道路曲率变化较大时，车辆的动态行为可能会发生显著变化，这将导致控制器所使用的动力学模型与实际车辆行为之间出现失配。这种失配会导致控制器的性能下降，甚至导致跟踪失败。其次，在目标函数的状态量和控制量矩阵方面，状态量和控制量矩阵将直接关系到控制输入的幅值和变化率，其次目标函数计算结果对状态量和控制量矩阵权重大小较为敏感。

(2) 轮胎侧偏刚度精度问题

在车辆的三自由度动力学模型建立过程中，为简化计算流程，会做出线性轮胎的假设。然而固定的轮胎侧偏刚度无法准确反映车辆在不同路况和驾驶条件下的实际动态行为。轮胎的侧偏刚度会受到道路曲率和车辆速度等多种因素的影响。因此，使用固定值将导致模型与实际车辆行为之间存在失配，进而影响控制器的跟踪性能。其次，这种简化可能导致控制器在应对道路曲率变化较大时的适应性降低。当道路曲率变化较大时，车辆需要更大的转向角和加速度来跟踪道路，而固定的轮胎侧偏

刚度可能无法提供足够的灵活性来适应这些变化，导致跟踪误差增大。

(3) MPC 预测时域对速度工况的适应性问题

固定预测时域无法适应不同的驾驶场景和需求。在高速驾驶或道路曲率变化较大的情况下，较长的预测时域更为合适，以便控制器能够提前预测并应对未来的轨迹变化。然而，在低速或道路曲率变化较小的情况下，较短的预测时域可能更为有效，过长的预测时域会引入不必要的计算负担和干扰，导致控制器的响应速度变慢。此外，固定的预测时域会限制控制器的性能和优化能力。预测时域的选择直接影响控制器对未来轨迹的预测精度和控制输入的优化范围。如果预测时域设置得过大或过小，将导致跟踪误差增大、控制器性能下降或无法找到最优的控制策略。

1.4.2 课题研究章节安排

针对章节 1.4.1 中课题困难点分析，本文以曲率自适应控制器、轮胎侧偏刚度实时估计和速度时变预测时域控制器三个主题作为本文的研究重点进行展开。如图 1-10 为本篇论文的主体架构示意图。

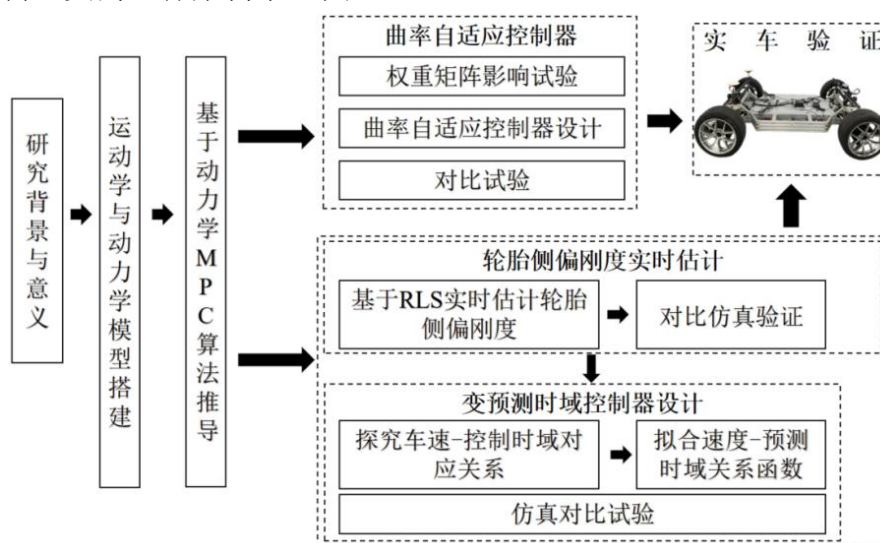


图 1-10 论文的主体结构及关系示意图

第一章围绕自动驾驶技术背景、研究意义及轨迹跟踪技术的研究现状展开论述。随着汽车行业的智能化和网联化进程加速，自动驾驶技术已成为全球交通领域的研究热点。通过计算机视觉、传感器、大数据和人工智能等技术的融合，为交通安全、效率和舒适性的提升提供了解决方案。其中特斯拉、华为等企业的显著进展和成功案例验证了自动驾驶技术的可行性和前景。轨迹跟踪技术作为自动驾驶的核心技术

之一，对实现车辆精确控制、保障行驶安全至关重要。而后针对轨迹跟踪控制技术，详细阐述了 PID 控制器、模糊控制、线性二次型 LQR、滑模控制、纯跟踪控制 Pure Pursuit、强化学习和模型预测控制等多种方法的跟踪原理，同时以上方法被广泛应用于轨迹跟踪技术的研究中。最后引出本文旨在探讨 MPC 轨迹跟踪技术的研究。

第二章首先完成了基于多种假设前提下的三自由度运动学模型的推导，这类模型在低速、稳态或平坦道路上的轨迹跟踪中表现出色，引出车辆处于高速行驶或复杂驾驶环境下时，仅依赖运动学模型可能无法满足精确控制的需求。随后分别基于车身 X 轴、Y 轴和绕 Z 轴建立了平衡方程，完成车辆的动力学方程的推导，分析了车辆动力学可以反映各种工况下车辆的动态响应和行驶特性的可行性。第二章动力学模型构建为后文 MPC 轨迹跟踪控制器的设计提供了控制主体。

第三章首先介绍了包括系统建模、控制目标设定、预测未来行为、优化控制策略、执行控制策略和实时更新的模型预测控制原理。重点分析了目标函数权重矩阵对控制器的影响规律并完成了 LTV-MPC 轨迹跟踪控制器设计和曲率自适应控制器的设计，最后通过仿真实验验证了控制器的有效性。最终实验对比验证了曲率自适应 LTV-MPC 控制器在轨迹跟踪精度和车辆稳定性方面可行性。

第四章提出了基于递归最小二乘法（RLS）的变预测时域模型预测控制（MPC）算法，以提升轨迹跟踪控制器的性能。首先，针对轮胎侧偏刚度这一关键参数，采用 RLS 方法进行实时估计，提高了车辆动力学模型的准确性以提高跟踪精度。其次，通过分析车速与预测时域对跟踪效果的影响，设计了变预测时域 MPC 跟踪控制器，以优化控制性能。通过仿真验证，证明所提算法的有效性。实验结果表明，变预测时域 MPC 控制器在不同车速下均能实现良好的轨迹跟踪效果，且实时估计的轮胎侧偏刚度提高了控制器的鲁棒性。

第五章描述了 MPC 轨迹跟踪算法的实车实验与分析过程。首先，实验平台基于全线控底盘进行实车硬件搭建，包括传感器选型和安装部署。上层软件方面，在 Linux 系统下部署了 ROS/Autoware 自动驾驶平台和涉及算法，完成了实验平台的整体搭建。通过实验，验证了 LTV-MPC 轨迹跟踪控制器在实车应用的有效性。

第六章为结论和展望。对全文工作及创新点进行总结，并讨论未来的研究方向。

第2章 车辆运动学与动力学模型

车辆模型可分为两大类：一类是车辆运动学模型，另一类是车辆动力学模型，对 MPC 轨迹跟踪至关重要。因为动力学和运动学建模为 MPC 提供了关于车辆行为的准确描述。动力学建模考虑了车辆的质量、惯性、摩擦等物理因素，以及车辆的加速度、转向率等动态特性。这种模型能够更准确地预测车辆的行为，包括车辆在不同工况下的动态响应和行驶特性^[61]。在 MPC 中，使用动力学建模可以使控制器更好地理解并预测车辆的行为，进而生成更加精确和有效的控制指令，实现对车辆的精确控制。运动学建模更多地关注车辆的几何特性和基本运动规律，如位置、速度和转向角等。虽然运动学模型忽略了一些物理因素，但在低速、稳态或平坦道路工况下，它可以提供足够准确的描述，用于快速规划和跟踪车辆轨迹^[62]。本章综合控制器的计算复杂度与模型精度的考虑，建立了简化后的三自由度车辆运动学和动力学方程。

2.1 车辆运动学模型

车辆横向运动的数学模型中，当模型不考虑车辆的受力情况时，可将车辆模型简化成单车模型(bicycle model)，如下图 2-1 所示。

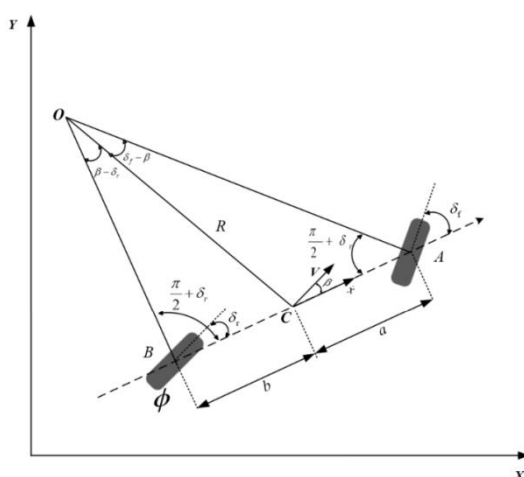


图 2-1 单车模型图

在单车模型中：

- (1) 左右前轮合并为单个轮胎，其中心点为 A 点，同样后轮等效后的中心点为 B 点。
- (2) 前后轮的转向角用 δ_f 和 δ_r 表示，模型中前后轮都可以转向，对于只有前轮转向的

系统，后轮转向角 δ_r 可设置为0。

(3) C 点代表车辆的重心， A 点和 B 点到重心的距离分别用 a 和 b 表示，轴距表示为 $L = a + b$ 。

(4) 车辆重心的速度用 V 表示，与车辆纵向轴的夹角为 β ，该角度为车辆质心侧偏角。

(5) 假设车辆平动，车辆运动状态可以用三个坐标量描述： X 、 Y 和 ϕ 。其中 X 、 Y 分别代表车辆重心的横、纵向位置， ϕ 为车身横摆角。

(6) O 点代表车辆的瞬时旋转中心，线段 AO 与 BO 分别与前后轮方向垂直，其交点为 O 点，线段 OC 的长度代表车辆的轨迹半径 R 。

(7) 车辆重心处的速度垂直于 OC ，车辆速度矢量与车辆纵轴的夹角为 β 。

(8) 假设速度矢量 V 的方向在 A 点和 B 点的转向角的方向相同，即在 A 点的速度矢量与车辆纵轴的夹角为 δ_f ，同样 B 点的速度矢量与车辆纵轴的夹角为 δ_r 。即前后轮的滑移角都为0。

基于以上假设，在图2-1的三角形 OCA 、 OCB 中，由根据正弦定理得：

$$\frac{\sin(\delta_f - \beta)}{a} = \frac{\sin\left(\frac{\pi}{2} - b\right)}{R} \quad (2-1)$$

$$\frac{\sin(\beta - \delta_f)}{b} = \frac{\sin\left(\frac{\pi}{2} - \delta_r\right)}{R} \quad (2-2)$$

由几何知识及条件假设，化简整理姿态信息：

$$\dot{\phi} = \frac{V \cos(\beta)}{a+b} \{ \tan(\delta_f) - \tan(\delta_r) \} \quad (2-3)$$

由于轴距 $L = a + b$ 远远小于轨迹半径 R ，滑移角 β 接近于0。一般车辆模型后轴为固定轴，故 δ_r 为0，所以等式(2-3)可以近似为：

$$\dot{\phi} = \frac{V}{L} \tan(\delta_f) \quad (2-4)$$

又根据自行车模型的位置信息得：

$$\begin{cases} \dot{X} = V \cos(\beta + \phi) \\ \dot{Y} = V \sin(\beta + \phi) \end{cases} \quad (2-5)$$

假设车辆重心点位于后轴，则 $\beta \approx 0$ ，则由上式(2-1)、(2-2)、(2-4)和(2-5)可得如下其微分方程形式：

$$\begin{cases} \dot{X} = V \cos(\phi) \\ \dot{Y} = V \sin(\phi) \\ \dot{\phi} = \frac{V \tan(\delta_f)}{L} \end{cases} \quad (2-6)$$

至此车辆三自由度运动学模型搭建完成。

2.2 车辆动力学模型

由于运动学模型忽略了车身质量及车辆受力等因素，因此仅适于在轨迹曲率以及侧向加速度较小的场景下。车辆动力学模型与运动学模型相比，主要用于分析研究车辆的平顺性和操纵稳定性，在车辆高速行驶下有着更高的可靠性，并考虑到了轮胎的受力情况及侧偏特性。在车辆动力学建模方面由最早的二自由度“单轨模型”发展了后续的二自由度、七自由度、十四自由度等更高自由度的动力学模型^[63]。高自由度动力学模型优势在于能够更加准确的描述车辆的动力学特性，提供更加准确数学模型，但复杂被控对象在运动控制算法求解过程中往往要求更加庞大的计算能力，所以实际表现会导致控制系统延时，导致控制效果达不到预期。

为兼顾车辆的行驶环境、转向稳定性以及求解复杂度等因素，本文选取以车辆三自由度动力学模型为基础。当车辆在采用简化的三自由度动力学模型时，要做出以下理想性假设：(1)忽略轮胎的横纵向耦合关系且轮胎的侧偏角和纵向滑移率较小；(2)忽略车辆的侧倾、垂向运动；(3)忽略车身悬架特性；(4)忽略空气阻力。根据上述简化条件可得车辆的动力学模型，如图 2-2 所示。其中 OXY 为大地坐标系； oxy 为车身坐标系； a 和 b 分别为车辆质心到前、后轴间的距离； F_{cf} 、 F_{cr} 分别为车辆轮胎受到的侧向力； F_{lf} 、 F_{lr} 分别为前、后轮胎受到的纵向力； $\dot{\phi}$ 为横摆角速度。

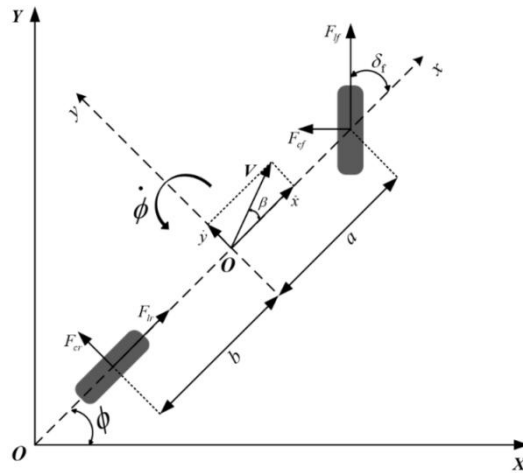


图 2-2 车辆动力学模型

根据牛顿第二定律，分别建立车身 X 轴、Y 轴、绕 Z 轴三个方向的平衡方程：

$$\begin{cases} m\ddot{x} = m\dot{y}\dot{\phi} + 2(F_{lr} + F_{lf}\cos\delta_f - F_{cf}\sin\delta_f) \\ m\ddot{y} = -m\dot{x}\dot{\phi} + 2(F_{cr} + F_{lf}\sin\delta_f + F_{cf}\cos\delta_f) \\ I_z\ddot{\phi} = 2a(F_{lf}\sin\delta_f + F_{cf}\cos\delta_f) - 2bF_{cr} \end{cases} \quad (2-7)$$

式(2-7)中： m 为整车整备质量； I_z 为车身绕 Z 轴转动时的转动惯量； δ_f 为车辆前轮转角； ϕ 为车体的横摆角； $\ddot{\phi}$ 为车辆的横摆加速度； $\ddot{x}$ 、 $\ddot{y}$ 分别为车辆纵向和侧向的加速度。

当轮胎的侧偏角较小且在侧向加速度小于等于 $0.4g$ 时，前后轮胎所受的横向力，纵向力与侧偏角的变化关系可近似描述为线性函数同时拟合精度较高。前后轮的侧向力分别为：

$$\begin{cases} F_{cf} = C_{cf} \left(\delta_f - \frac{\dot{y} + a\dot{\phi}}{\dot{x}} \right) \\ F_{cr} = C_{cr} \frac{b\dot{\phi} - \dot{y}}{\dot{x}} \end{cases} \quad (2-8)$$

式(2-8)中： C_{cf} 为前轮侧偏刚度、 C_{cr} 为后轮侧偏刚度。

车辆前轮、后轮的纵向力分别为：

$$\begin{cases} F_{lf} = C_{lf}S_f \\ F_{lr} = C_{lr}S_r \end{cases} \quad (2-9)$$

式(2-9)中： C_{lf} 分别为前轮纵向刚度， C_{lr} 分别为后轮纵向刚度； S_r 为前轮的滑移率； S_f 为后轮的滑移率。

通过车身坐标系和大地坐标系之间的转换，可在大地坐标系下对车辆运动情况进行描述：

$$\begin{cases} \dot{Y} = \dot{x}\sin\phi + \dot{y}\cos\phi \\ \dot{X} = \dot{x}\cos\phi - \dot{y}\sin\phi \end{cases} \quad (2-10)$$

式(2-10)中： $\dot{X}$ 、 $\dot{Y}$ 为车辆在大地惯性坐标系下的纵向速度和侧向速度。

综合上述式(2-7)、(2-8)、(2-9)和(2-10)，可建立式(2-11)：

$$\begin{cases} m\ddot{y} = -m\dot{x}\dot{\phi} + 2 \left[C_{cf} \left(\delta_f - \frac{\dot{y} + a\dot{\phi}}{\dot{x}} \right) + C_{cr} \left(\frac{b\dot{\phi} - \dot{y}}{\dot{x}} \right) \right] \\ m\ddot{x} = m\dot{y}\dot{\phi} + 2 \left[C_{lf}S_f - C_{cf} \left(\delta_f - \frac{\dot{y} + a\dot{\phi}}{\dot{x}} \right) \delta_f + C_{lr}S_r \right] \\ I_z\ddot{\phi} = 2 \left[aC_{cf} \left(\delta_f - \frac{\dot{y} + a\dot{\phi}}{\dot{x}} \right) - bC_{cr} \left(\frac{b\dot{\phi} - \dot{y}}{\dot{x}} \right) \right] \\ \dot{Y} = \dot{x}\sin\phi + \dot{y}\cos\phi \\ \dot{X} = \dot{x}\cos\phi - \dot{y}\sin\phi \end{cases} \quad (2-11)$$

将式(2-11)转换成状态空间方程：

$$\begin{cases} \dot{\xi}_{(t)} = f(\xi_{(t)}, u_{(t)}) \\ \eta_{(t)} = C\xi_{(t)} \end{cases} \quad (2-12)$$

式(2-12)中系统状态量 $\xi_{(t)} = [\dot{y}, \dot{x}, \phi, \dot{\phi}, Y, X]$ ；系统控制量 $u_{(t)} = [\delta_f]$ ；系统输出量为 $\eta_{(t)} = [\phi, Y]$ ；输出矩阵 $C = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$ 。以上所建立的车辆动力学模型是第三章及第四章所设计的 LTV-MPC 轨迹跟踪控制器的基础。

2.3 本章小结

本章首先完成了基于多种假设前提下的三自由度运动学模型。由运动学的推导过程可以知道，其主要关注车辆的几何特性和基本运动规律，如位置、速度和转向角等。这类模型在低速、稳态或平坦道路上的轨迹跟踪中表现出色。然而由于车辆处于高速行驶或复杂驾驶环境下时，仅依赖运动学模型无法满足精确控制的需求。这时，车辆动力学模型就显得尤为重要。为兼顾车辆的行驶环境、转向稳定性以及求解复杂度等因素，本文分别基于车身 X 轴、Y 轴和绕 Z 轴建立了平衡方程，而后经过推导将车辆的动力学方程转化成了状态空间的形式。动力学模型考虑了车辆如质量、惯性、轮胎与地面间的摩擦等以及车辆的加速度、转向率等动态特性相关动力学因素。车辆动力学可以反映各种工况下车辆的动态响应和行驶特性，从而生成更加精确有效的控制指令。本文的动力学模型构建，也为后文 MPC 轨迹跟踪控制器的设计提供了控制主体。

第3章 曲率自适应 LTV-MPC 轨迹跟踪研究

3.1 模型预测控制原理

模型预测控制基本原理是通过在每个采样时刻对系统未来行为进行预测，并基于这些预测结果来生成最优控制策略，以实现求解所需的控制目标。MPC 的主要步骤包括建立系统模型、设定控制目标、预测未来行为、优化控制策略和执行控制操作。首先，需要建立描述系统动态行为的数学模型，然后确定控制目标，包括期望的系统输出或状态，以及约束条件。随后，基于系统模型和当前状态，使用优化算法对未来一段时间内系统的行为进行预测，通过求解优化问题生成最优的控制策略。最后，根据生成的控制策略执行控制操作，在每个控制周期结束后重新计算和更新控制策略。MPC 能够处理多变量、多约束和非线性系统，并且可以灵活地应用于不同类型如工业过程控制、自动驾驶汽车和能源管理等领域的控制问题。其优点包括能考虑系统动态特性、约束条件和不确定性，提高控制性能和鲁棒性，同时允许灵活地调整控制策略以适应不同的工况和需求，图 3-1 展示了基础 MPC 控制算法原理。该控制算法可分为以下主要步骤：

(1) 系统建模

首先需要建立描述系统动态行为的数学模型。所建模型是基于物理原理的动力学模型或经验模型。动力学模型基于系统的动力方程，描述系统状态随时间的演化。模型的选择取决于系统复杂性、控制要求等因素。

(2) 控制目标设定

确定包括期望的系统输出和状态，以及约束条件下的控制目标。控制目标为系统轨迹跟踪状态。约束条件包括对输入变量的限制、状态变量的边界等。

(3) 预测未来行为

基于系统模型和当前状态，使用优化算法对未来一段时间内系统的行为进行预测。预测涉及对系统状态、输出以及其它重要变量的估计。预测时间范围和精度取决于控制系统的要求和性能。

(4) 优化控制策略

在预测的基础上，通过使用线性规划或二次规划优化求解最小化控制误差和满

足系统约束条件问题来生成最优的控制策略。

(5) 执行控制策略

根据生成的控制策略，通过调节系统的输入变量实施对系统的控制操作，只执行生成控制策略的第一个控制动作，在下一个控制周期重新计算和执行控制策略。

(6) 实时更新

在每个控制周期结束后，根据系统的实际响应和测量值，重新开始整个预测和优化过程，更新控制策略并实施新的控制操作。这种实时更新策略可以使系统对外部扰动和变化做出及时的响应，保持控制系统的稳定性和鲁棒性。

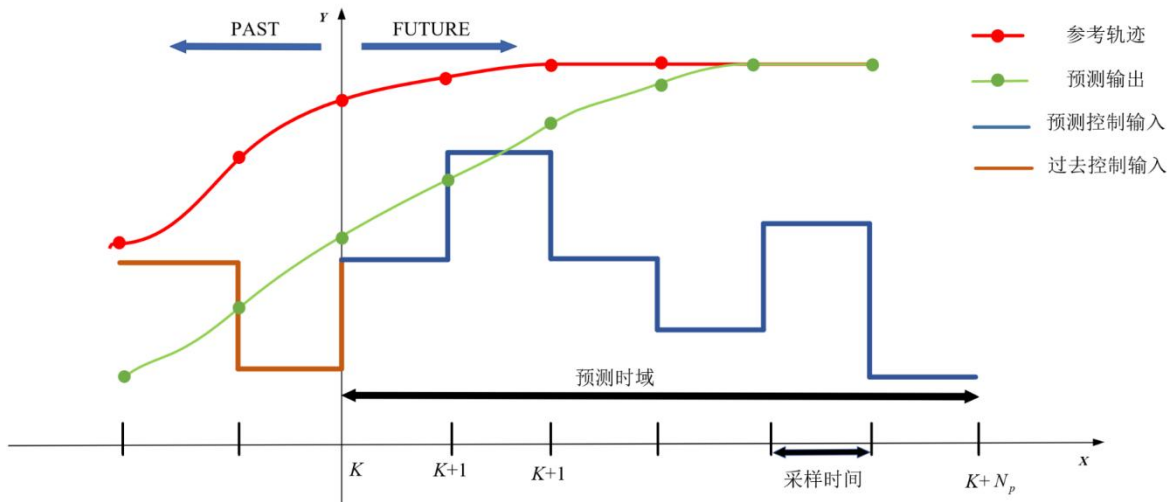


图 3-1 MPC 算法原理图

3.2 LTV-MPC 轨迹跟踪控制器设计

3.2.1 车辆非线性系统线性化

由于自动驾驶车辆在行驶过程中对跟踪控制器的实时性要求较高，而非线性 MPC 轨迹跟踪控制器求解计算过程较为复杂无法满足该要求^[64]，线性化处理后的 MPC 轨迹跟踪控制器在进行预测过程中的优化问题通常更易于求解，可以在较短的时间内得到快速的控制响应且保持系统动态稳定的局部特性，故需要将式(2-11)进行线性化处理。

设系统的任意工作点位为 $[\xi_{(0)}, u_{(0)}]$ ，对系统施加控制量 $u_{(0)}$ 后，得到状态量：

$$\dot{\xi}_{(0)} = f(\xi_{(0)}, u_{(0)}) \quad (3-1)$$

在 $[\xi_{(0)}, u_{(0)}]$ 处进行保留一阶项的泰勒展开可得：

$$\dot{\xi}_{(t)} = f(\xi_{(0)}, u_{(0)}) + \left. \frac{\partial f}{\partial \xi} \right|_{\substack{\xi=\xi_{(0)} \\ u=u_{(0)}}} (\xi_{(t)} - \xi_{(0)}) + \left. \frac{\partial f}{\partial u} \right|_{\substack{\xi=\xi_{(0)} \\ u=u_{(0)}}} (u_{(t)} - u_{(0)}) \quad (3-2)$$

式(3-2)中

$$\left. \frac{\partial f}{\partial \xi} \right|_{\substack{\xi=\xi_{(0)} \\ u=u_{(0)}}} = \begin{bmatrix} \frac{\partial \ddot{x}}{\partial \dot{x}} & \frac{\partial \ddot{x}}{\partial \dot{y}} & 0 & \frac{\partial \ddot{x}}{\partial \dot{\phi}} & 0 & 0 \\ \frac{\partial \ddot{y}}{\partial \dot{x}} & \frac{\partial \ddot{y}}{\partial \dot{y}} & 0 & \frac{\partial \ddot{y}}{\partial \dot{\phi}} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{\partial \ddot{\phi}}{\partial \dot{x}} & \frac{\partial \ddot{\phi}}{\partial \dot{y}} & 0 & \frac{\partial \ddot{\phi}}{\partial \dot{\phi}} & 0 & 0 \\ \cos \phi & -\sin \phi & \frac{\partial f_{\ddot{y}}}{\partial \phi} & 0 & 0 & 0 \\ \sin \phi & \cos \phi & \frac{\partial f_{\ddot{x}}}{\partial \phi} & 0 & 0 & 0 \end{bmatrix}$$

$$\left. \frac{\partial f}{\partial u} \right|_{\substack{\xi=\xi_{(0)} \\ u=u_{(0)}}} = \left[\frac{2C_{cf}}{m}, \frac{2C_{cf}}{m} \left(2\delta_{f,t-1} - \frac{\dot{y}_t + a\dot{\phi}_t}{\dot{x}_t} \right), 0, \frac{2aC_{cf}}{I_z}, 0, 0 \right]^T$$

将式(3-2)和(3-1)式进行做差得：

$$\dot{\tilde{\xi}}_{(t)} = A_{(t)}\tilde{\xi}_{(t)} - B_{(t)}\tilde{u}_{(t)} \quad (3-3)$$

式(3-3)中：

$$\dot{\tilde{\xi}}_{(t)} = \dot{\xi}_{(t)} - \dot{\xi}_{(0)}; \tilde{\xi}_{(t)} = \xi_{(t)} - \xi_{(0)}; \tilde{u}_{(t)} = u_{(t)} - u_0; \tilde{A}_{(t)} = I + TA_{(t)}; \tilde{B}_{(t)} = TB_{(t)}$$

3.2.2 车辆系统离散化

本采用向前欧拉法对式(3-3)进行离散化，得：

$$\begin{cases} \tilde{\xi}_{(k+1|t)} = \tilde{A}_{(k|t)}\tilde{\xi}_{(k|t)} + \tilde{B}_{(k|t)}\tilde{u}_{(k|t)} + d_{(k|t)} \\ d_{(k|t)} = \tilde{\xi}_{(k+1|t)} - \tilde{A}_{(k|t)}\tilde{\xi}_{(k|t)} - \tilde{B}_{(k|t)}\tilde{u}_{(k|t)} \end{cases} \quad (3-4)$$

式(3-4)中：

$$\tilde{A}_{(k|t)} = \begin{bmatrix} 1 + \frac{2\delta_f C_{cf}(\dot{y} + a\dot{\phi})}{m\ddot{x}^2} T & \left(\dot{\phi} - \frac{2\delta_f C_{cf}}{m\dot{x}}\right) T & 0 & \left(\dot{y} - \frac{2a\delta_f C_{cf}}{m\dot{x}}\right) T & 0 & 0 \\ \left(-\dot{\phi} + \frac{2[C_{cf}(\dot{y} + a\dot{\phi}) + C_{cr}(\dot{y} - b\dot{\phi})]}{m\ddot{x}^2}\right) T & 1 - \frac{2C_c + 2C_{cr}}{m\dot{x}} T & 0 & \left(-\dot{x} + \frac{2(aC_{cf} - bC_{cr})}{m\dot{x}}\right) T & 0 & 0 \\ 0 & 0 & 1 & T & 0 & 0 \\ \frac{2aC_{cf}(\dot{y} + a\dot{\phi}) + 2bC_{cr}(\dot{y} - b\dot{\phi})}{I_z \dot{x}^2} T & \frac{-2aC_{cf} + 2bC_{cr}}{I_z \dot{x}} T & 0 & 1 + \frac{-2(a^2 C_{cf} + 2b^2 C_{cr})}{I_z \dot{x}} T & 0 & 0 \\ T \cos \phi & -T \sin \phi & -T(\dot{x} \sin \phi + \dot{y} \cos \phi) & 0 & 1 & 0 \\ T \sin \phi & T \cos \phi & T(\dot{x} \cos \phi - \dot{y} \sin \phi) & 0 & 0 & 1 \end{bmatrix}$$

$$\tilde{B}_{(k|t)} = \left[\frac{2C_{cf}}{m}, \frac{2C_{cf}}{m} \left(2\delta_f - \frac{\dot{y}_t + a\dot{\phi}_t}{\dot{x}_t} \right), 0, \frac{2aC_{cf}}{I_z}, 0, 0 \right]^T$$

$$\begin{cases} \bar{\xi}_{(k+1|t)} = f(\bar{\xi}_{(k|t)}, \bar{u}_{(k|t)}), k = t, t+1, \dots, t+N_p-1 \\ \bar{\xi}_{(t|t)} = \tilde{\xi}_{(t)} \\ \bar{u}_{(k|t)} = u_{(t-1)} \\ \tilde{A}_{(k|t)} = (I + TA_{(t)}) \\ \tilde{B}_{(k|t)} = TB_{(t)} \end{cases}$$

为降低控制器的计算复杂度，做式(3-5)假设：

$$\begin{cases} \tilde{A}_{(k|t)} = \tilde{A}_{(t|t)}, k = t, t+1, \dots, t+N_p-1 \\ \tilde{B}_{(k|t)} = \tilde{B}_{(t|t)}, k = t, t+1, \dots, t+N_p-1 \end{cases} \quad (3-5)$$

在上述线性离散化后得到的空间状态方程中，为实现对控制增量的有效约束，需将线性离散化后的空间状态方程进行进一步转换为关于车辆控制增量的状态空间方程：

$$\check{\xi}_{(k|t)} = \begin{pmatrix} \tilde{\xi}_{(k|t)} \\ \tilde{u}_{(k-1|t)} \end{pmatrix} \quad (3-6)$$

$$\begin{cases} \check{\xi}_{(k+1|t)} = \tilde{A}_{(k|t)} \check{\xi}_{(k|t)} + \tilde{B}_{(k|t)} \Delta u_{(k|t)} + \tilde{d}_{(k|t)} \\ \check{\eta}_{(k|t)} = \tilde{C} * \check{\xi}_{(k|t)} \end{cases} \quad (3-7)$$

式(3-7)中 $\tilde{A}_{(k|t)} = \begin{bmatrix} \tilde{A}_{(k|t)} & \tilde{B}_{(k|t)} \\ 0_{m \times n} & I_m \end{bmatrix}$; $\tilde{B}_{(k|t)} = \begin{bmatrix} \tilde{B}_{(k|t)} \\ I_m \end{bmatrix}$; $\Delta u_{(k|t)} = u_{(k|t)} - u_{(k-1|t)}$; $\tilde{d}_{(k|t)} = \begin{bmatrix} d_{(k|t)} \\ 0 \end{bmatrix}$; $\tilde{C} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$; m 、 n 分别表示系统控制量和状态量的个数。

3.2.3 预测状态方程与输出模型的构建

定义 k 时刻系统输出和输入分别为：

$$\begin{cases} Y_{(k+1|t)} = [\check{\eta}_{(k+1|t)}, \check{\eta}_{(k+2|t)}, \dots, \check{\eta}_{(k+N_p|t)}]^T \\ \Delta U_{(k|t)} = [\Delta u_{(k|t)}, \Delta u_{(k+1|t)}, \dots, \Delta u_{(k+N_c|t)}]^T \end{cases} \quad (3-8)$$

其中 N_p 、 N_c 分别为系统的预测时域和控制时域，且 $N_p > N_c$ 。

对进行整个预测时域内的迭代，得到系统未来 k 时刻预测方程矩阵表达式：

$$Y_{(k|t)} = \psi_{(k|t)} \tilde{\xi}_{(k|t)} + \theta_{(k|t)} \Delta U_{(k|t)} + \Gamma_{(k|t)} \phi_{(k|t)} \quad (3-9)$$

式(3-9)中:

$$\begin{aligned} \Psi_{(k|t)} &= \begin{bmatrix} \tilde{C}\tilde{A}_{(k|t)} \\ \tilde{C}\tilde{A}_{(k|t)}^2 \\ \vdots \\ \tilde{C}\tilde{A}_{(k|t)}^{N_p} \end{bmatrix} \\ \Theta_{(k|t)} &= \begin{bmatrix} \tilde{C}\tilde{B}_{(k|t)} & 0 & \dots & 0 \\ \tilde{C}\tilde{A}_{(k|t)}\tilde{B}_{(k|t)} & \tilde{C}\tilde{B}_{(k|t)} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ \tilde{C}\tilde{A}_{(k|t)}^{N_p-1}\tilde{B}_{(k|t)} & \tilde{C}\tilde{A}_{(k|t)}^{N_p-2}\tilde{B}_{(k|t)} & \dots & \tilde{C}\tilde{A}_{(k|t)}^{N_p-N_c}\tilde{B}_{(k|t)} \end{bmatrix} \\ \Gamma_{(k|t)} &= \begin{bmatrix} \tilde{C} & 0 & 0 & 0 \\ \tilde{C}\tilde{A}_{(k|t)} & \tilde{C} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{C}\tilde{A}_{(k|t)}^{N_p-1} & \tilde{C}\tilde{A}_{(k|t)}^{N_p-2} & \dots & \tilde{C}\tilde{A}_{(k|t)}^0 \end{bmatrix} \\ \Delta U_{(k|t)} &= \begin{bmatrix} \Delta u_{(k|t)} \\ \Delta u_{(k+1|t)} \\ \vdots \\ \Delta u_{(k+N_c-1|t)} \end{bmatrix} \\ \phi_{(k|t)} &= \begin{bmatrix} \tilde{d}_{(k|t)} \\ \tilde{d}_{(k+1|t)} \\ \vdots \\ \tilde{d}_{(k+N_p-1|t)} \end{bmatrix} \end{aligned}$$

3.2.4 目标函数设计

目标函数设计对 MPC 轨迹跟踪控制器至关重要, 决定了控制器如何权衡不同的控制目标以及如何对系统行为进行优化。目标函数设计的意义在于指导控制器使车辆沿着期望轨迹行驶, 通过将系统状态量误差纳入目标函数, 使控制器在当下控制时域内不断调整控制策略以最小化误差, 从而实现精确的轨迹跟踪。此外, 目标函数的设计同时考虑了多个控制目标, 在本章目标函数的设计之中, 将车辆的横摆角误差、轨迹跟踪横向误差以及车辆前轮转角的变化量作为考量因素, 同时分别施加代价权重矩阵纳入目标函数^[65]。为防止由于系统的复杂性和动力学约束而导致的无法求解情况, 加入松弛因子, 设计的目标函数的表达式如下所示:

$$J(\xi_{(k|t)}, u_{(k-1|t)}, \Delta U_{(k|t)}) = \sum_{i=1}^{N_p} \|\tilde{\eta}_{(k+i|t)} - \eta_{ref_{(k+i|t)}}\|_Q^2 + \sum_{i=0}^{N_c-1} \|\Delta U_{(k+i|t)}\|_R^2 + \rho \varepsilon^2 \quad (3-10)$$

式(3-10)中: Q 为控制量权重矩阵, R 为状态量权重矩阵; ρ 为松弛因子权重系数; η_{ref} 为输出量的参考值。

对于约束部分, 在保证了该控制器可以使系统在每个控制周期内尽可能接近期望轨迹, 实现良好的轨迹跟踪性能的条件下, 将控制量和控制增量的大小加入目标函数中, 以最小化控制输入的变化量。通过优化此项, 控制器可减小控制输入的变化幅度, 实现更加平滑的控制操作, 减少系统的振荡和能耗。此外将系统输出约束条件加入目标函数中, 确保了系统状态量和输出在允许范围内^[66,67]。通过优化这些约束条件, 控制器可以确保系统在运行过程中不会超出安全范围, 避免产生不稳定或不安全的行为。

控制器条件约束部分:

控制量约束:

$$U_{min} \leq U_t + A_I \times \Delta U \leq U_{max} \quad (3-11)$$

控制增量约束:

$$lb \leq X \leq ub \quad (3-12)$$

输出量约束:

$$Y_{min} \leq Y(k) \leq Y_{max} \quad (3-13)$$

在上式(3-11)、(3-12)和(3-13)中, U_{max} 为控制量的最大值; U_{min} 为控制量的最小值; lb 为状态量的最大值; ub 为状态量的最小值; ΔU_{max} 为控制增量的最大值; ΔU_{min} 分别为控制增量的最小值。

3.2.5 反馈矫正

在 Carsim-Simulink 联合仿真系统的每一个控制周期中, 对式(3-10)进行求解计算, 可得到控制增量序列:

$$\Delta U_{(k|t)}^* = [\Delta u_{(k|t)}^*, \Delta u_{(k+1|t)}^*, \dots, \Delta u_{(k+N_c-1|t)}^*]^T \quad (3-14)$$

将该控制增量序列中的第一项作为前轮转角控制变化量输入 CarSim 进行下一个控制周期内的仿真。

$$u_{(k|t)} = u_{(k-1|t)} + \Delta u_t^* \quad (3-15)$$

系统根据当前 CarSim 仿真得到的状态量预测下一时刻的输出，再通过式(3-15)得到前轮转角输入至 CarSim，如此不断循环向前来实现对参考轨迹的跟踪。

3.3 曲率自适应控制器设计

3.3.1 联合仿真平台介绍

(1) CarSim 仿真环境介绍

CarSim 是一款高精度汽车动力学仿真软件，用于模拟和分析车辆在各种条件下的动态行为和性能。它提供了精确的物理模型，支持多体动力学仿真，能够考虑悬挂系统、轮胎特性、空气动力学等因素。软件还能模拟不同工况下的环境条件，如道路状况和天气变化，以及驾驶员行为和车载传感器等。CarSim 通过提供与其他软件的接口，如 MATLAB/Simulink，方便与其它工具集成，其广泛应用于车辆设计、性能测试和自动驾驶系统开发等领域。整个仿真流程包括模型建立、参数设置、环境配置、仿真执行、结果分析和参数调整等步骤。

CarSim 仿真软件主界面如下图 3-2 所示，该软件分为四大功能模块：

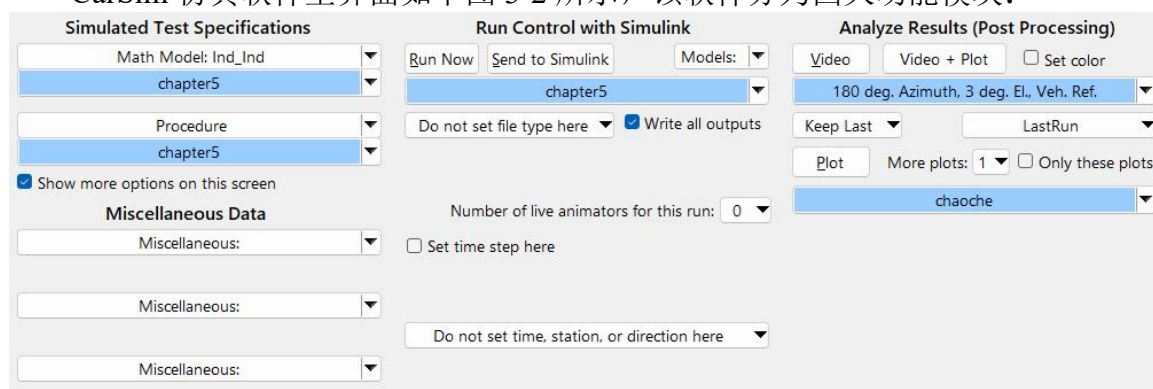


图 3-2 CarSim 主界面

1) 仿真车辆模型设定模块：车辆参数配置是仿真的基础，决定了车辆的基本行驶特性。此区域，可以定义和配置车身尺寸、重量、悬挂系统、轮胎特性等车辆的基本参数。

2) 车辆环境工况及驾驶条件模块：在这个区域，用户可以设定道路类型、路面状况、天气条件等仿真环境条件，同时也可以设置驾驶员输入、车速、加速度等驾驶条件。

3) 模型求解模块：此模块主要负责控制仿真运行过程，允许用户设定仿真开始和结束时间，仿真步长，以及控制仿真的暂停、继续和停止。通过此模块，可执行

精确地控制仿真，确保仿真结果的准确性和可靠性。此外，该模块还提供了一些高级功能，如仿真的自动重复运行，以及与外部控制系统的接口，用于更复杂的仿真场景。

4) 仿真运行和结果分析模块：此区域可以运行仿真模型，并查看仿真结果。CarSim 提供了多种数据分析和可视化工具，帮助深入理解车辆的动态行为和性能特点。

(2) Matlab/Simulink 控制模块介绍

Simulink 是一个基于 Matlab 的图形化编程环境，专用于模拟、分析和设计动态系统。其提供了一个交互式的界面，允许用户通过拖放预定义的块和连接线来构建系统模型。Simulink 支持多种域的模拟，包括控制系统、信号处理、通信系统、物理建模等。用户可以在 Simulink 中设计复杂的系统模型，进行仿真实验，分析系统性能，并生成可用于实际硬件部署的代码。Simulink 广泛应用于汽车、航空航天、电子和工业自动化等工程和科学领域，是一个用于验证设计理念、优化系统参数和加速产品开发过程的强大工具。

Simulink 提供了丰富的库和工具，支持自定义模块的开发，同时也提供了与其他软件和硬件的接口，方便与外部系统集成。构建 Simulink 与 CarSim 联合仿真平台可以同时结合 Simulink 提供丰富的控制算法库和设计工具以及 CarSim 的高精度车辆动力学模型，使得开发和测试车辆控制策略变得更加高效，精准的模拟车辆在各种条件下的行为；此外联合仿真平台可灵活的配置仿真参数及环境，支持多种仿真场景的设计和测试，快速验证 MPC 轨迹跟踪控制策略的有效性，一定程度上减少实车测试的需要。

3.3.2 仿真实验设计及分析

(1) CarSim 整车参数的设置

在 CarSim 整车建模时，首先对目标车型进行建模，本文选择 B-Class,Hatchback Aero 车型作为车辆建模的基础。该车型基本参数如下表 3-1 所示，随后根据表中数据在 Sprung Mass 模块中进行建模，完成目标车辆的基本参数的设定，车辆模型相 CarSim 软件界面设置结果如下图 3-3 所示。

表 3-1 车辆模型参数

参数	定义	数值
m	车辆质量	1723 kg
I_z	绕 z 轴横摆转动惯量	4175 kg·m ²
a	前轴到车辆质心的距离	1232 mm
b	后轴到车辆质心的距离	1468 mm
h_{CG}	质心高度	500 mm
C_{cf}	前轮侧偏刚度	66895 N/rad
C_{cr}	前轮纵向刚度	66900 N/rad
d	轮距	225 mm

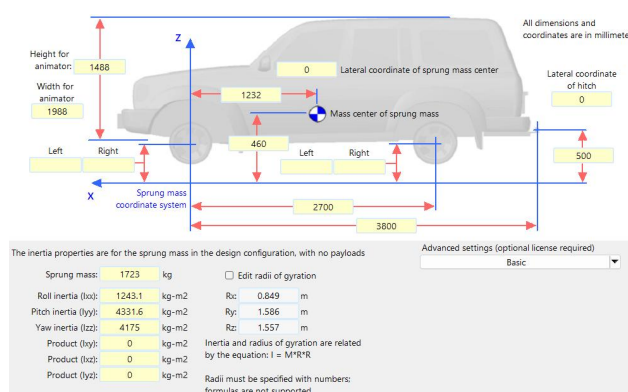


图 3-3 CarSim 车辆部分基础参数设置

(2) Matlab/Simulink 联合仿真模型搭建

Matlab/Simulink 为 CarSim 提供了联合仿真接口，根据 MPC 轨迹跟踪控制器所需输入量与输出量确定该端口变量，输入输出端口设置如下图 3-4 所示。本章仿真实验的控制对象为前轮转向车辆，控制目标为前轮转向角度，故 S-Function 的计算输出结果为前轮转向角度，经单位匹配转化后输入至仿真车辆的输入端口。在 CarSim 中经仿真步长内的控制仿真，向 Simulink 的 S-Function 模块输出控制时间段内 CarSim 输出端口所设定的车辆实时状态。搭建完成的 Simulink-CarSim 联合仿真平台如下图 3-5 所示。

Variables Activated for Import			Variables Activated for Export		
Name	Mode	Initial Value			
1 IMP_STEER_L1	Add	0.0	1. Vy_SM		
2 IMP_STEER_R1	Add	0.0	2. Vx_SM		
3 IMP_STEER_L2	Add	0.0	3. Yaw		
4 IMP_STEER_R2	Add	0.0	4. AV_Y		
			5. Yo		
			6. Xo		
			7. Kappa_L1		
			8. Kappa_R1		
			9. Beta		

(a)输入端口

(b)输出端口

图 3-4 CariSim 输入输出端口设定

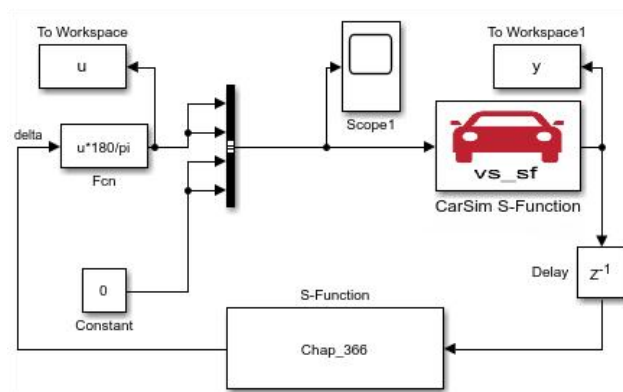


图 3-5 Simulink 联合仿真系统

(3) 仿真实验思路设计

在基于动力学的 MPC 轨迹跟踪控制器中，目标函数有反映对轨迹跟踪精度、控制平滑性和其他性能指标的要求。具体而言，目标函数包括衡量轨迹偏差的平方项、控制增量的平方项。通过选择合适的权重矩阵和项的形式，调整控制器对不同性能指标的重视程度，从而实现在轨迹跟踪精度、控制平滑性和系统稳定性等方面的平衡。目标函数的设计使得基于动力学 MPC 轨迹跟踪控制器能够适应不同的应用场景和性能要求，是其灵活性和强大性的关键所在。

在目标函数设计之中，目标函数权重矩阵 Q 和 R 起着非常重要的作用。状态量权重矩阵 Q 用于调整位置、速度、姿态等状态量的重要程度。在轨迹跟踪问题中，控制器能可使车辆尽可能地接近预定轨迹，因此会给与轨迹偏差较大的状态量更高的权重。通过调整 Q 矩阵中的元素，可调节控制器对不同状态量的敏感程度，进而影响控制器的行为；控制增量权重矩阵 R 矩阵用于调整控制前轮转向角输入变化的重要性。在实际应用中，控制器能够平滑地调整控制输入，避免过度的操作，以提高车辆的行驶舒适性和系统的稳定性。通过增加 R 矩阵中的权重，可以减小控制增量的幅度，使控制器倾向于采取更平滑的控制策略。

为协调改善 LTV-MPC 在轨迹跟踪时跟踪的自适应跟踪能力的精确性和稳定性，在控制器中引入权重自适应算法以实现车辆根据跟踪道路曲率实时变化权重系数来优化轨迹跟踪效果。

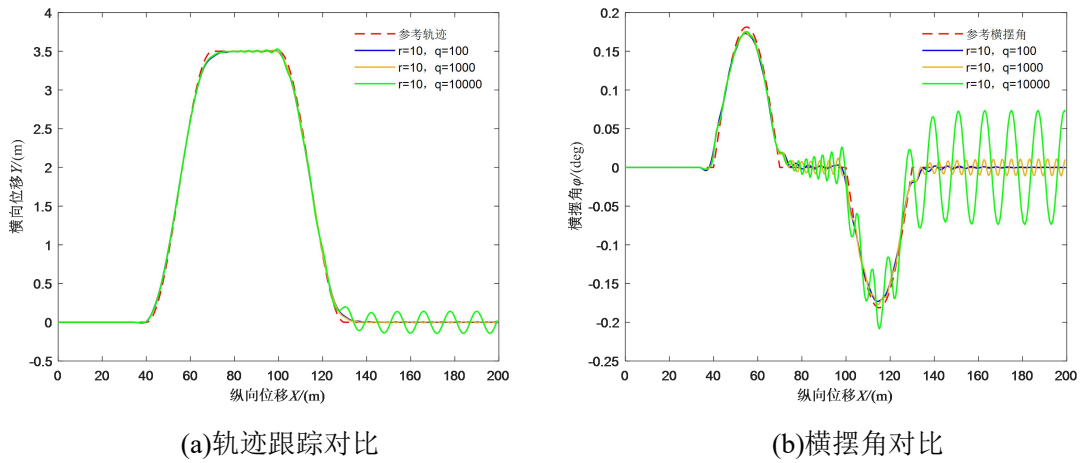
本章实验采用 Carsim 与 Simulink 联合仿真，表 3-2 为 S-function 模块中 MPC 控制器的部分参数。

表 3-2 控制器参数

参数	定义	数值
N_p	预测时域	12
N_c	控制时域	5
T	采样时间	0.05 s
u	前轮转角控制量约束范围	$-0.1744 \leq \delta_f \leq 0.1744$
Δu	前轮转角控制增量约束范围	$-0.0148 \leq \Delta \delta_f \leq 0.0148$
ρ	松弛因子	1000

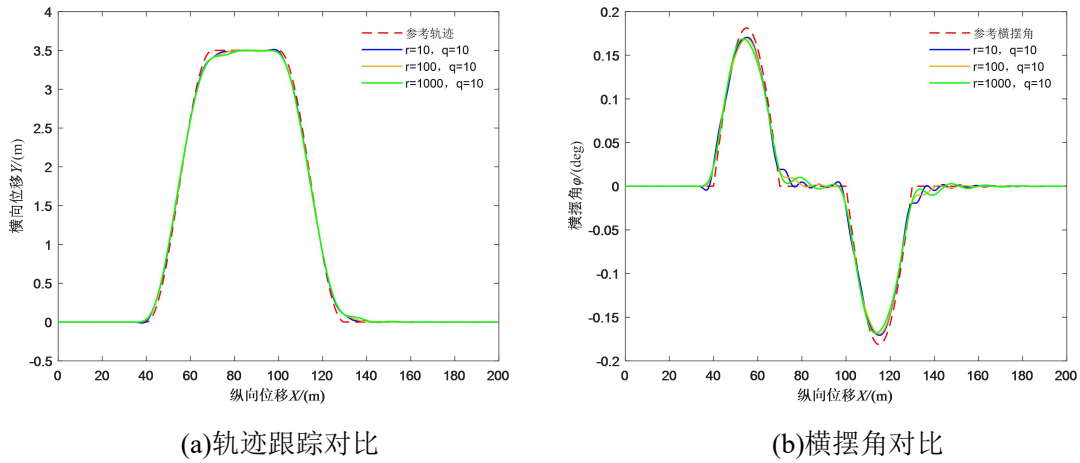
3.3.3 权重矩阵对跟踪效果的影响分析

(1) 探究矩阵 Q 的变化对跟踪效果的影响

图 3-6 不同 Q 矩阵轨迹跟踪结果对比图

由图 3-6 的(a)和图 3-6(b)可得较大比例的 q 在弯道处可以较快的跟踪上目标轨迹。但是如果比例过大, 则会导致进入直线轨迹时产生震荡, 对于轨迹曲率的变化反应过度灵敏; Q 的比例相对较小时, 参考轨迹曲率变化时, 其轨迹跟踪的速度相对较慢, 但趋近轨迹相对稳定, 且震荡幅度也相对较小。

(2) 探究矩阵 R 的变化对跟踪效果的影响

图 3-7 不同 R 矩阵轨迹跟踪结果对比图

由图 3-7(a)和图 3-7(b)可以看出，在车辆沿着直线轨迹行驶时，不同 r 值都能取得不错的跟踪效果。当直线行驶的车辆进入曲线行驶时，较大的 r 值，出现了相对于较小的 r 值的跟踪滞后性。相对较小的 r 值在面对曲线曲率变化时，跟踪至目标轨迹的速度更快。

3.3.4 自适应权重系数矩阵的拟合

(1) 参考轨迹曲率的计算

道路参考轨迹曲率为：

$$K_{ref} = \frac{|Y(X_{ref}(t))''|}{(1+Y(X_{ref}(t))^2)^{3/2}} \quad (3-16)$$

在实验过程中，为便于对自变量 K 的函数的拟合，故对 K_{ref} 进行归一化处理：

$$\bar{K} = \frac{K_{ref} - K_{min}}{K_{max} - K_{min}} \quad (3-17)$$

式(3-17)中， X 为纵向位置； K 为参考轨迹的曲率； K_{min} 为参考轨迹曲率的最小值； K_{max} 为参考轨迹曲率的最大值。

(2) 曲率自适应函数拟合

综合上述对比实验可得，由图 3-6 的(a)和(b)可知，当车辆延直线行驶时，此时参考轨迹的行驶曲率变化较小，不同的 r 值和较小的 q 值都可以相对稳定的跟踪目标轨迹。若 q 值过大，即使在进行直线轨迹行驶时，也会发生车身的震荡导致轨迹跟踪失败。由图 3-7(a)和(b)可知，当车辆跟踪的参考轨迹曲率较大且发生较大变化时，

较大的 r 在轨迹曲率发生变化的时候, 轨迹跟踪出现了明显的滞后性。而相对较小的 r 值, 面对轨迹曲率发生显著变化时的跟踪精度更佳, 图 3-8 展示了道路曲率自适应 LTV-MPC 轨迹跟踪控制器和 CarSim 间的求解控制流程示意图。综合上述车辆行驶在不同道路曲率时不同权重系数对轨迹跟踪效果的影响规律, 拟合出以下道路曲率自适应函数。

$$\begin{cases} q = q_0 - a_1 \cdot (1 - \bar{K}) \\ r = r_0 \end{cases} \quad 0 \leq \bar{K} \leq \frac{1}{2} \quad (3-18)$$

$$\begin{cases} q = q_0 \\ r = r_0 - a_2 \bar{K} \end{cases} \quad \frac{1}{2} \leq \bar{K} \leq 1 \quad (3-19)$$

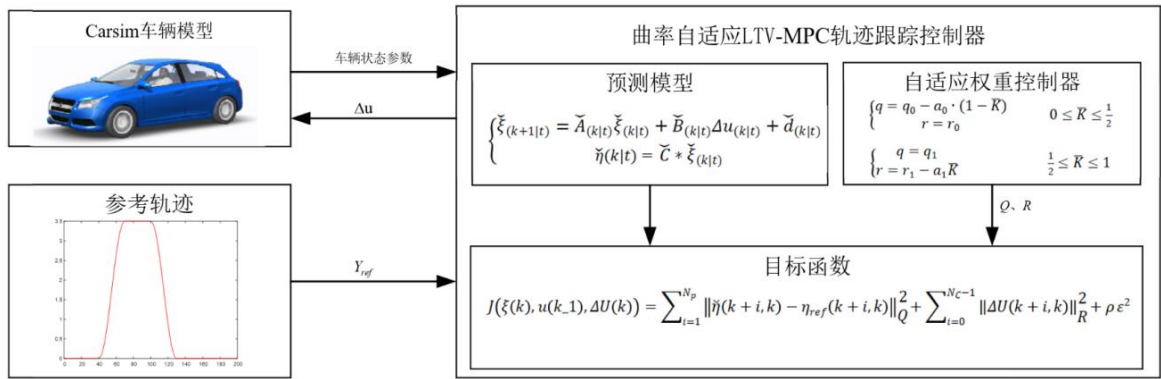


图 3-8 道路曲率自适应 LTV-MPC 轨迹跟踪控制器示意图

3.4 曲率自适应 LTV-MPC 控制器对比仿真实验

为验证该轨迹跟踪控制器的跟踪性能, 仿真实验采取车辆以 36km/h 的速度在超车轨迹曲线工况下与非线性 MPC 控制器、LTV-MPC 控制器进行对比。上述两种对比控制器均采用跟踪效果较好的一组权重矩阵参数, 且同时联合仿真相关参数设定与表 3-1 和表 3-2 保持一致。

联合仿真实验在相同控制时域、预测时域及其他可变因素相同的情况下, 对比了传统非线性 MPC 以及控制器优化前的 LTV-MPC 轨迹跟踪控制器, 对比仿真实验轨迹跟踪结果、跟踪误差、车辆横摆角变化如图 3-9 所示。

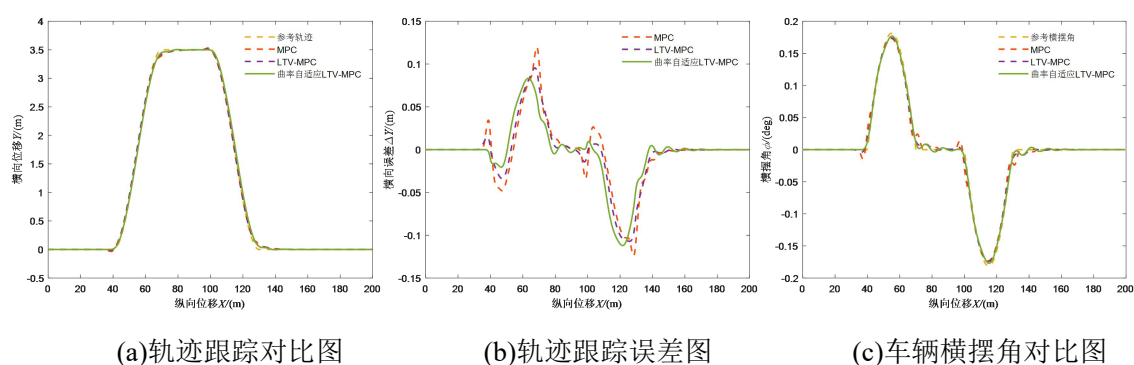


图 3-9 曲率自适应 MPC 轨迹跟踪结果对比图

基于上述实验结果，表 3-3 统计了 LTV-MPC 与曲率自适应 LTV-MPC 轨迹跟踪控制器在纵向位置为 40m、67m、100m 和 127m 共四处转向中心位置的横向误差均值，横向误差均值计算方式是分别在每个转向中心位置的前后两米每 1 米取一点，共五个点测量求得其横向误差均值。

表 3-3 四处轨迹曲率变化较大处附近横向误差平均值

纵向位置	40 m	67 m	100 m	127 m
LTV-MPC	0.01027	0.07953	0.007887	0.08426
曲率自适应 LTV-MPC	0.009007	0.05041	0.004956	0.05515
优化效果 (%)	12.27%	36.61%	37.19%	34.53%

综合图 3-9 和表 3-3 可知车辆在纵向位置为 40m、67m、100m 和 127m 共四处跟踪轨迹曲率变化较大处，曲率自适应 LTV-MPC 轨迹跟踪控制器相较于 LTV-MPC 在跟踪过程中跟踪的效果较好。将轨迹的行驶曲率作为影响因素来确认权重系数的大小，进而改善了轨迹在曲率变化时的跟踪性能，一定程度抑制了车辆通过转向在轨迹跟踪过程中的转向不足或者超调现象。

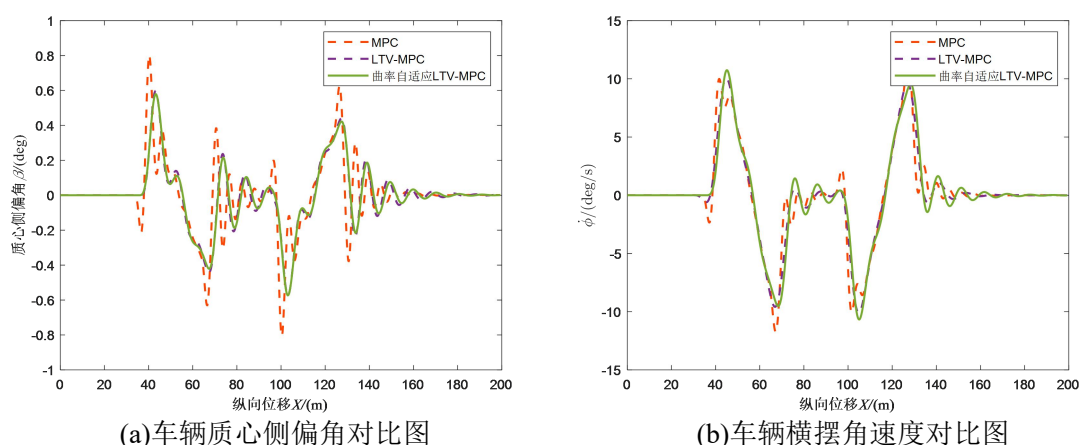


图 3-10 曲率自适应 MPC 轨迹跟踪结果对比图

表 3-4 仿真实验跟踪情况

控制器类型	最大横向误差/(m)	最大横摆角误差/($^{\circ}$)	最大质心侧偏角/($^{\circ}$)	最大横摆角速度/($^{\circ}$ /s)
MPC	0.120	0.0185	0.0801	11.7376
LTV-MPC	0.0955	0.0163	0.5973	10.059
曲率自适应 LTV-MPC	0.0832	0.0154	0.5800	10.740
优化效果 (%)	12.88%	5.52%	2.90%	\

如图 3-10 和表 3-4 所示,实验结果对比了传统 MPC 与 LTV-MPC 轨迹跟踪控制器同曲率自适应 LTV-MPC 轨迹跟踪控制器上的轨迹跟踪效果。从上面实验结果可以看出优化后的控制器的轨迹跟踪效果要优于 LTV-MPC 控制器。

仿真实验结果显示线性 MPC 轨迹跟踪控制器跟踪效果在轨迹跟踪精准度以及控制稳定性上均由于传统非线性 MPC 控制器。此外,对比将曲率自适应方法改进于 LTV-MPC 控制器前后可得: LTV-MPC 的最大横向误差为 0.955m,改进后的轨迹跟踪控制器最大横向误差为 0.832m,减小了 12.88%。尤其在参考轨迹曲率变化较大处可以更好的目标轨迹,跟踪的轨迹精度更高。LTV-MPC 的最大横摆角误差为 0.0163° ,改进后的控制器最大横摆角误差为 0.0154° ,减小了 5.52%; LTV-MPC 的最大质心侧偏角为 0.5973m,改进后的控制器最大横摆角误差为 0.5800m,最大质心侧偏角减小 2.90%。改进后的控制器在跟踪超车轨迹过程中的最大横摆角速度为 10.74,比较 LTV-MPC 增加了 $0.681^{\circ}/s$ 。改进后的控制器根据参考道路曲率求解出对应目标函数的权重系数,然后通过二次规划函数求解出在约束范围内的前轮转角增量,该策略使得轨迹跟踪过程更加稳定,进而车辆行驶过程中更加平稳。车辆在轨迹曲率变化较大处时轨迹跟踪精度提升是由于改进后的控制器在参考轨迹曲率较大时,将实时

降低控制量权重矩阵来增强车辆的转向灵活性。当车辆行驶至曲率较小处时，控制器则会降低状态量权重矩阵。因此提高了车辆轨迹跟踪时的稳定性与精度。

3.5 本章小结

本章研究内容旨在提升 MPC 轨迹跟踪控制器的跟踪精确性和稳定性。首先通过泰勒公式线性展开以及向前欧拉法离散化处理完成了 LTV-MPC 的理论部分的推导。为确保轨迹跟踪精度和车辆控制稳定性，目标函数包含了系统跟踪误差和前轮转角的增长量两项指标，同时在目标函数中加入松弛因子，避免出现在多动力学约束情况下的无解情况。随后搭建 CarSim 和 Matlab/Simulink 联合仿真平台，分别分析目标函数的权重矩阵 Q 、 R 因道路曲率变化而对轨迹跟踪的效果，设计出道路曲率自适应函数。

最后在相同工况下与传统 MPC 和 LTV-MPC 轨迹跟踪控制器进行对比，通过分析其轨迹跟踪横向误差、横摆角、横摆角速度、质心侧偏角等参数，发现曲率自适应轨迹跟踪控制器在双移线工况下在四处轨迹曲率变化较大时，相较于为改进的 LTV-MPC，轨迹跟踪误差平均减小 30%，车身最大质心侧偏角降低了 2.9%，车身横摆角及横摆角速度均在可控范围内，验证了优化后的控制器相比与 LTV-MPC 在轨迹跟踪精度和车辆行驶稳定性上都有了较大的提升。

第4章 基于 RLS 的变预测时域 LTV-MPC 的算法研究

由于 MPC 轨迹跟踪控制器的预测时域参数可看作驾驶员基于当前车辆状态向前预瞄未来一段时间内的预测距离。当车辆在变速度下，若采用固定的预测时域将影响轨迹跟踪控制效果，无法满足 MPC 轨迹跟踪控制器对多工况下的适应性要求。此外轮胎侧偏刚度是对车辆模型进行建模的重要参数，反映了车辆在轮胎线性区的操纵稳定性。在第三章建立 LTV-MPC 轨迹跟踪控制器的过程中，做出了轮胎横向侧偏角与侧偏刚度的函数处于线性区域的假设，即假设轮胎的侧偏刚度为定值，而在实际的车辆运行过程中，轮胎的侧偏刚度会随着路面条件、行驶车速等工况的不同而发生变化，并对跟踪结果产生一定的影响，本章将围绕轮胎侧偏刚度实时估计与变模型预测时域来优化 LTV-MPC 轨迹跟踪控制器。

4.1 轮胎侧偏刚度估计

4.1.1 递推最小二乘法公式

基于车辆动力学模型进行模型预测控制需要用到车辆的状态信息、惯性参数、参考道路信息以及轮胎线性侧偏刚度等。其中，车辆的状态信息包括纵向速度、侧向速度和前轮偏角等，车辆惯性参数包括车身质量、车身转动惯量和质心到前后轴的距离位置等，参考道路信息包括路面倾角和道路参考曲率等。车辆的状态信息、惯性参数以及参考道路信息可以在仿真环境中直接获得。

轮胎侧偏刚度线性化是对车辆模型进行线简化的重要过程，反映了车辆在轮胎线性区域内的操纵稳定特性，而且随着行驶工况、路面条件的不同，此参数往往发生变化。轮胎侧偏刚度进行实时估计可以提高自动驾驶车辆对不同道路工况的适应性，下面根据车辆的状态信息和惯性参数的历史数据，采用递归最小二乘法估计轮胎的线性侧偏刚度。

假设所研究的历史数据窗口长度为 N ，采用缓冲数列方法将 N 组数据绑定为一个数组，可得到轮胎侧偏力与轮胎线性侧偏刚度的线性回归方程：

$$Y_m(k) = \varphi^T(k)\Phi(k) + e(k) \quad (4-1)$$

式(4-1)中， $\varphi(k)$ 为回归向量。

当估计前轴线性侧偏刚度时：

$$\varphi(k) = \frac{v_y(k) + l_f \dot{\psi}(k)}{v_x(k)} - \delta_f(k) \quad (4-2)$$

式(4-2)中, $\varphi(k)$ 表示前轴处的轮胎侧偏角。

待估计的模型参数为: $\Phi(k) = \bar{C}_{af}(k)$ 时

$$Y_m(k) = [F_{yf}(k) \quad F_{yf}(k-1) \quad \cdots \quad F_{yf}(k-N+1)]^T \quad (4-3)$$

当估计后轴的线性侧偏刚度时, $\varphi(k)$ 为后轴处的轮胎侧偏角, 且待估计的模型参数 $\Phi(k) = \bar{C}_{ar}(k)$ 时:

$$Y_m(k) = [F_{yr}(k) \quad F_{yr}(k-1) \quad \cdots \quad F_{yr}(k-N+1)]^T \quad (4-4)$$

基于递推最小二乘法的模型参数估计原理可以表述为: 在所研究的历史数据窗口内, 使实际值与估计值的平方差最小, 其代价函数可表示为:

$$J = \sum_{i=1}^N \lambda_m^{N-i} [Y_m(i) - \varphi^T(i) \hat{\Phi}(i)]^2 \quad (4-5)$$

式(4-5)中, λ_m 为遗忘因子, N 为时间窗口的长度, $\Phi(k)$ 表示模型参数的估计值。

经化简可递推最小二乘法的递推公式:

$$\begin{cases} P(k) = \lambda_m^{-1} [P(k-1) - K(k) \varphi^T(k) P(k-1)] \\ \hat{\Phi}(k) = \hat{\Phi}(k-1) + K(k) [Y_m(k) - \varphi^T(k) \hat{\Phi}(k-1)] \\ K(k) = P(k-1) \varphi(k) [\lambda_m + \varphi^T(k) P(k-1) \varphi(k)]^{-1} \end{cases} \quad (4-6)$$

至此完成对最小二乘法的推导。

4.1.2 实时轮胎侧偏刚度的估计

根据第二章内容, 建立的如下三自由度动力学模型:

$$\begin{cases} \dot{v}_x = \frac{F_x^r}{m} + v_y \gamma \\ \dot{v}_y = \frac{F_y^f + F_y^r}{m} - v_x \gamma \\ \dot{\phi} = \frac{l_f F_y^f - l_r F_y^r}{I_z} \end{cases} \quad (4-7)$$

可以求解得到前轮侧向力:

$$\begin{cases} F_y^f(k) = \frac{l_r m a_y(k)}{L} + \frac{I_z \dot{\phi}(k)}{L} \\ F_y^r(k) = \frac{l_f m a_y(k)}{L} - \frac{I_z \dot{\phi}(k)}{L} \end{cases} \quad (4-8)$$

当车辆处于高速曲线行驶时, 轮胎侧偏力与侧偏角将不再呈现线性关系, 故传统高速工况下的线性轮胎模型将不再适用, 需要引入轮胎非线性模型。下列 flala 轮

胎侧偏特性提出了一种能够近似表达轮胎非线性特征的模型：

$$\begin{cases} 2F_y^f = 2C_f\alpha_f - \frac{LC_f^2}{\mu m g a}\alpha_f^2 \\ 2F_y^r = 2C_r\alpha_r - \frac{LC_r^2}{\mu m g b}\alpha_r^2 \end{cases} \quad (4-9)$$

前后轮胎侧偏角计算公式如下：

$$\begin{cases} \alpha_f = \delta_f - \frac{u+l_f\gamma}{v} = \delta_f - \beta - \frac{l_f\gamma}{v} \\ \alpha_r = \delta_r - \frac{u-l_r\gamma}{v} = \delta_r - \beta + \frac{l_r\gamma}{v} \end{cases} \quad (4-10)$$

根据上述非线性轮胎模型计算公式，协同上前后轮胎侧偏角计算公式(4-10)采用递归最小二乘算法在线估计前后轮胎的转弯刚度。结合递推最小二乘算法的系统方程可写为下式(4-11)，式中相关符号含义如下表 4-1 所示：

$$\begin{cases} \hat{C}_{f,y}(k) = \hat{C}_{f,y}(k-1) + K_y(k) \cdot (F_y^f(k) - \hat{C}_{f,y}(k-1)\alpha_f(k)) \\ K_y(k) = P_y(k-1)\alpha_f(k) \cdot (\rho_y + P_y(k-1)\alpha_f(k)^2)^{-1} \\ P_x(k) = (1 - K_y(k)\alpha_f(k)) \cdot P_y(k-1)/\rho_y \end{cases} \quad (4-11)$$

表 4-1 最小二乘参数估计符号注释表

符号	含义
v_x/v_y	纵向/横向速度
l_f/l_r	前/后轴到质心的距离
K_x/K_y	递归最小二乘法的估计器增益 估价员
P_x/P_y	递归性的误差协方差矩阵 最 小二乘估计量
I_z	车身转动惯量
α_f/α_r	前/后轮侧偏角
ρ_x/ρ_y	递归性最小的遗忘因子 平方 估计量
$\hat{C}_{f,y}$	轮胎前轴侧偏刚度估计值
$\hat{C}_{r,x}$	轮胎后轴侧偏刚度估计值

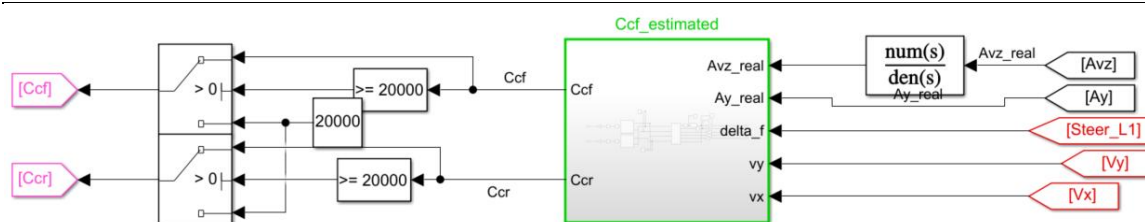


图 4-1 RLS 控制求解原理图

图 4-1 为仿真实验中, 利用 Simulink 搭建的基于 RLS 估计轮胎侧偏刚度求解图。由图 4-1 可知, 将车辆的横摆角速度、横向加速度、前轮转角、横向速度和纵向速度输入至推导的 RLS 算法求解流程之中, 便可得到估计的前后轮胎侧偏刚度。而后将估计到的时变轮胎侧偏刚度结果输入至预设计的 LTV-MPC 轨迹跟踪控制器之中, 以此避免使用固定的轮胎侧偏刚度。加入轮胎侧偏刚度实时估计到基于动力学的 LTV-MPC 轨迹跟踪控制器中理论可以带来多方面的改进。首先, 它能更准确地捕捉轮胎的动态行为从而提高轨迹跟踪的精度。其次, 它增强了控制器在不同路面条件和车辆状态下的稳定性适应能力。此外, 侧偏刚度实时估计还使控制器能够更有效地利用轮胎潜在的操控性能, 从而优化车辆的整体操控性能。同时, 这种估计减少了模型不确定性和参数变化对控制器性能的影响, 提升了控制器的鲁棒性。侧偏刚度实时估计增强了控制策略的适应性和灵活性, 使得控制器能够自适应不同的轮胎特性和工作条件, 这对提高车辆的安全性和操控性至关重要。

4.1.3 轮胎侧偏刚度的仿真验证

在 4.1.2 中提出了基于 RLS 方法实时估计轮胎侧偏刚度的方法利用估计的轮胎刚度来实时更新车辆动力学模型中的该参数, 来提高车辆三自由度动力学模型的模型精度的方法。而后 LTV-MPC 轨迹跟踪控制器再根据数据计算输出当前时刻的控制变量至 Carsim 车辆仿真模块进行控制。

在 CarSim 软件中设置路面附着系数为 0.85, 速度从 0-100m 处做线性匀加速运动至 200km/h, 在 100-200m 时, 做匀减速运动, 速度曲线如下图 4-2 所示。

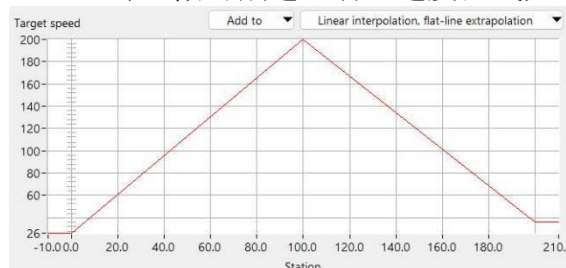


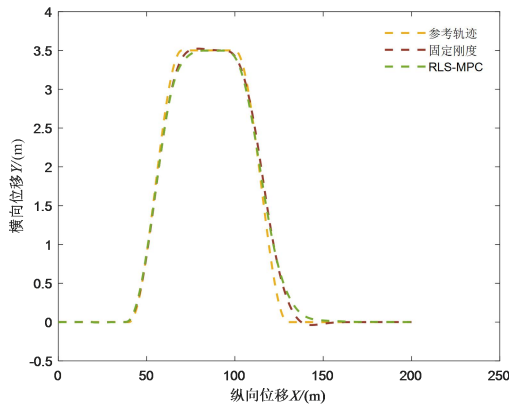
图 4-2 对比实验 CarSim 速度曲线示意图

为验证该方法对 LTV-MPC 轨迹跟踪控制器提升的有效性，仿真实验同样基于 MATLAB/Simulink 与 CarSim 展开。其中 CarSim 提供的仿真车辆模型与第三章车辆基本参数保持一致，其他相关控制器参数如表 4-2 所示。本节将基于 RLS 方法实时估计轮胎侧偏刚度的 LTV-MPC 控制器同固定前后侧偏刚度的控制器进行对比。

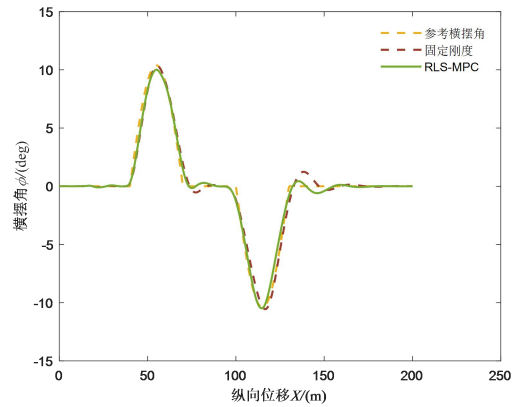
表 4-2 控制器参数

参数	定义	数值
N_p	预测时域	12
N_c	控制时域	5
T	采样时间	0.02 s
Q	状态量矩阵	$[1000 \ 0; 0 \ 600]$
R	控制量矩阵	$[10]$
u	前轮转角控制量约束范围	$-0.1744 \leq \delta_f \leq 0.1744$
Δu	前轮转角控制增量约束范围	$-0.0148 \leq \Delta \delta_f \leq 0.0148$
ρ	松弛因子	1000
ϕ	航向角约束	$-3^\circ \leq \phi \leq 5^\circ$

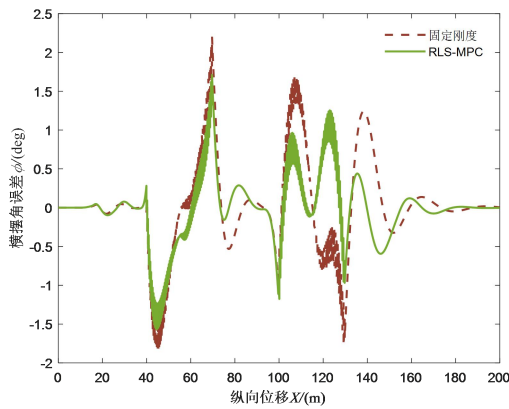
仿真实验在基于超车轨迹工况下进行，实验结果分析了轨迹跟踪情况、横摆角变化、横摆角偏差、前轮转角四组数据如下图 4-3 所示。



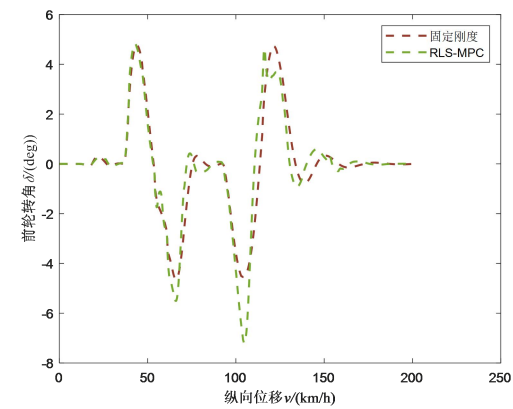
(a) 轨迹跟踪对比图



(b) 横摆角对比图



(c) 横摆角误差对比图



(d) 前轮转角对比图

图 4-3 加入 RLS 下对比固定刚度仿真结果

由图 4-3 (a)和 4-3(b)可知, 基于固定刚度下, 由于车辆横摆角跟踪的出现较大震荡, 导致车辆在 70m 及 140m 左右时的轨迹跟踪出现明显的超调现象。此外由图 4-3(c)可知, 固定刚度下控制器的最大横摆角误差为 2.1deg, 时变刚度下控制器的最大横摆角误差为 1.695deg, 大约减小了 20%。此外在车身的四处转向位置, 采用固定刚度下的控制器横摆角误差均大于基于实时 RLS 估计的控制器, 这是由于轮胎的非线性特性, 轮胎的侧偏刚度实际上会随着车辆实时的工况而改变, 而固定参数的轮胎侧偏角忽略了这种非线性变化, 导致控制器无法适应这些变化, 限制了控制器调整横摆动态响应的能力, 导致跟踪性能下降。而基于 RLS 实时轮胎侧偏刚度估计的控制器会综合车辆的横摆角速度、横向加速度、前轮转角以及横向和纵向速度来实时估计当前车辆状态进而调整车辆轮胎侧偏刚度。所以从转向稳定性的结果来看要显然优于使用固定轮胎侧偏刚度下轨迹跟踪控制器。

4.2 变预测时域 LTV-MPC 跟踪控制器设计

预测时域 N_p 在 MPC 控制器中起到了至关重要的作用，它代表了控制器对未来系统状态的预测程度，即预测未来多少个时间步长。其作用主要体现在以下几个方面：

(1) 预测时域决定了控制器对未来系统行为的预测范围。当预测时域较大时，控制器能够预测更远的未来状态，从而可以提前做出更全面的决策，有助于控制器更好地应对系统的不确定性和干扰，提高系统的鲁棒性。

(2) 预测时域的大小直接影响到控制器的性能和精度。过大的预测时域可能导致控制器在计算过程中产生较大的误差，降低轨迹跟踪的精度。而过小的预测时域则可能使车辆因无法及时转向而造成轨迹跟踪失败。因此，选择合适的预测时域对于实现精确的轨迹跟踪和稳定控制至关重要。

(3) 预测时域还影响到控制器的计算复杂度和实时性。较长的预测时域意味着需要对未来时刻计算更多的预测模型状态量，这需要使用更长的优化时间间隔，从而增加每个周期控制器的计算量。因此，在满足系统性能要求的同时，还需要考虑预测时域对计算资源和实时性的影响。在 MPC 算法应用至轨迹跟踪中时，其类似于驾驶员对未来时刻对目标轨迹的预瞄，故其在不同速度下预瞄距离的长短对轨迹跟踪过程的稳定性以及精度具有重大影响。当车辆处于较低车速下时，应采用较小的 N_p 即“较短的预瞄距离”以此达到提高轨迹跟踪精度的目的；当车辆处于较高车速下时，控制器应配合较大的 N_p 即“较长的预瞄距离”来达到良好的控制稳定性以及跟踪精度。

本节将搭建 CarSim 于 Matlab/Simulink 联合仿真平台，在 4.1 加入 RLS 轮胎侧偏刚度实时估计方法基础上分别在低、中、高速下分析不同预测时域在双移线轨迹下的跟踪控制效果，根据不同预测时域反馈出的效果来拟合出适合不同车速工况下的预测时域。最后在联合仿真平台下搭建车辆变速工况，分析对比变预测时域 LTV-MPC 轨迹跟踪控制器同若干固定预测时域在轨迹跟踪精度以及稳定性上的差异。

4.2.1 探究车速与预测时域 N_p 对跟踪效果的影响

为探究车速与对应预测时域对跟踪效果的影响，仿真实验分析了车辆分别在 36km/h、72km/h、90km/h、115km/h 下，试验四组预测时域($N_p=8$ 、 $N_p=14$ 、 $N_p=20$ 、 $N_p=26$)下的轨迹跟踪精度与行驶稳定性。

(1) 当车速为 36km/h 时，不同预测时域的仿真结果对如图 4-4 和表 4-3 所示。

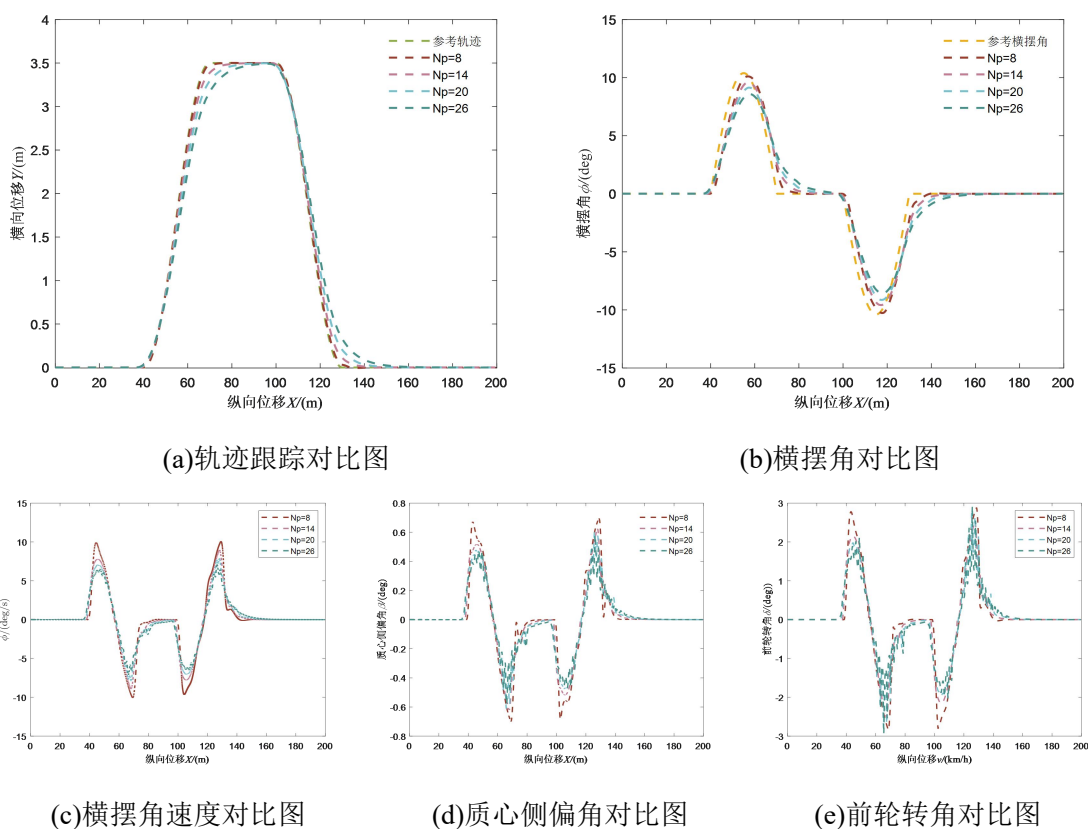


图 4-4 36km/h 多固定预测时域跟踪效果对比图

表 4-3 36km/h 跟踪数据对比表

预测时域 N_p	有效性判定	最大横向误差/(m)	最大横摆角误差/($^{\circ}$)	最大质心侧偏角/($^{\circ}$)	最大横摆角速度/($^{\circ}$ /s)
8	有效	0.03	0.28	0.68	9.94
14	有效	0.13	0.80	0.52	7.81
20	有效	0.22	1.25	0.47	7.02
26	有效	0.32	8.58	0.47	6.48

当车辆在 36km/h 的速度下行驶时，由仿真对比结果不难看出由于车速较低，控制器在四组预测时域下均能完成正常的目标轨迹跟踪，其横摆角速度、质心侧偏角、前轮转角的变化范围均在可控且合理的范围内。但在车辆转向时，随着 N_p 的增大其真实行驶轨迹呈现出逐渐远离目标轨迹的现象，当 $N_p=8$ 时，其轨迹跟踪的横向偏差最小且与参考横摆角之间的插值最小。其中，当 $N_p=26$ 时，其与目标轨迹的横向偏移量最大，跟踪精度最差。

(2) 当车速为 72km/h 时，不同预测时域的仿真结果对比如图 4-5 所示。

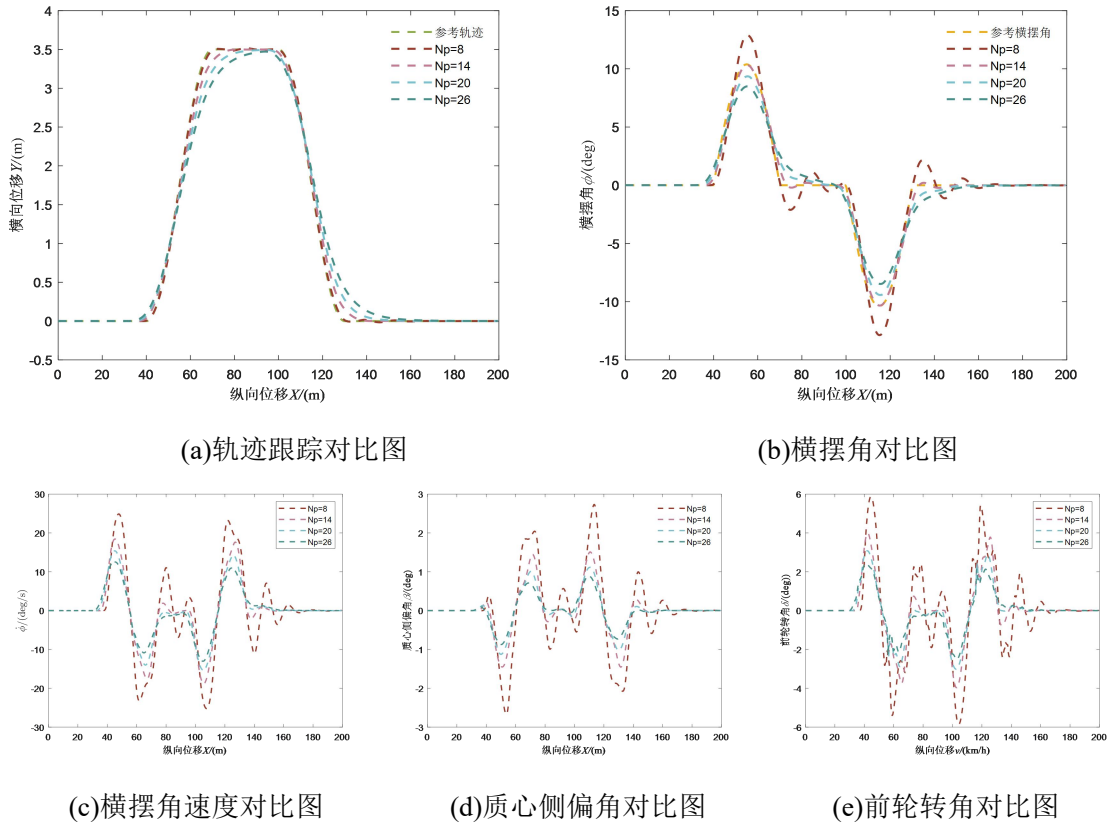


图 4-5 72km/h 多固定预测时域跟踪效果对比图

表 4-4 72km/h 跟踪数据对比表

预测时域 N_p	有效性判定	最大横向误差/(m)	最大横摆角误差/($^{\circ}$)	最大质心侧偏角/($^{\circ}$)	最大横摆角速度/($^{\circ}$ /s)
8	有效	0.02	0.51	2.71	24.80
14	有效	0.16	0.06	1.51	18.45
20	有效	0.27	1.03	1.11	15.40
26	有效	0.40	1.88	0.89	12.55

当车辆在 72km/h 的速度下行驶时，由仿真对比结果可知当 $N_p=8$ 时，车辆在纵向位移为 75 米以及 130 米处时出现了明显的车身横摆角、横摆角速度、质心侧偏角和前轮转角的波动，车辆行驶时的稳定性较差。当 $N_p=14$ 时，在转向时，车辆轨迹跟踪过程以及横摆角变化较为自然平滑，同时横摆角速度、质心侧偏角也均在可控范围内，跟踪精度与车身稳定性于本组属最优。

(3) 当车速为 90km/h 时，不同预测时域的仿真结果对比如图 4-6 所示。

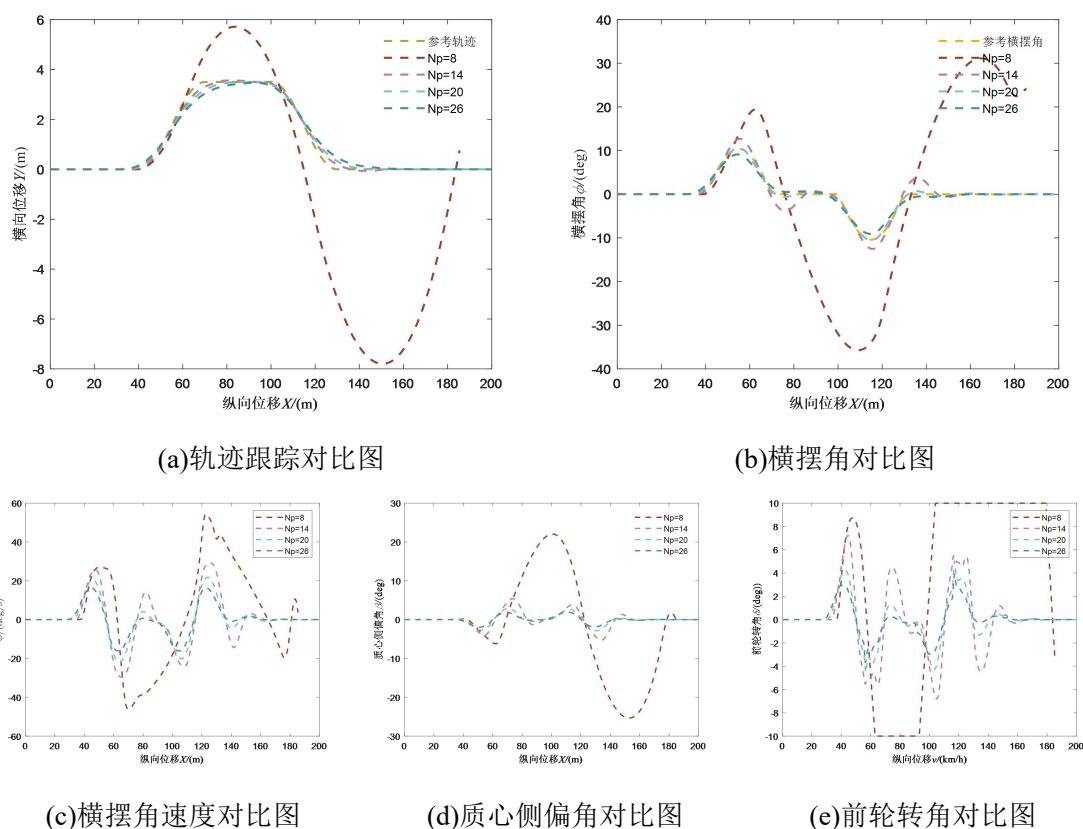


图 4-6 90km/h 多固定预测时域跟踪效果对比图

表 4-5 90km/h 跟踪数据对比表

预测时域 N_p	有效性判定	最大横向误差/(m)	最大横摆角误差/(°)	最大质心侧偏角/(°)	最大横摆角速度/(°/s)
8	失败	—	—	—	—
14	有效	0.23	2.30	5.40	24.52
20	有效	0.37	0.23	2.85	20.49
26	有效	0.49	1.27	1.87	16.27

由图 4-6 和表 4-5 可知在 90km/h 下, 在 $N_p=8$ 时, 车辆出现了完全无法跟踪目标轨迹的情况且车身稳定性差, 行驶过程危险。 $N_p=14$ 时, 其在纵向位移为 75 米以及 130 米处时同样出现了横摆角、横摆角速度、质心侧偏角和前轮转角的波动。由此可知在相对高速的情况下, 较小的预测时域难以满足车辆正常的“预瞄距离”。当 $N_p=20$ 时, 在转向时, 车辆轨迹跟踪过程以及横摆角变化较为自然平滑, 同时横摆角速度、质心侧偏角也均在可控范围内, 跟踪精度与车身稳定性于本组属最优。

(4) 当车速为 115km/h 时, 不同预测时域的仿真结果对比如图-7 所示。

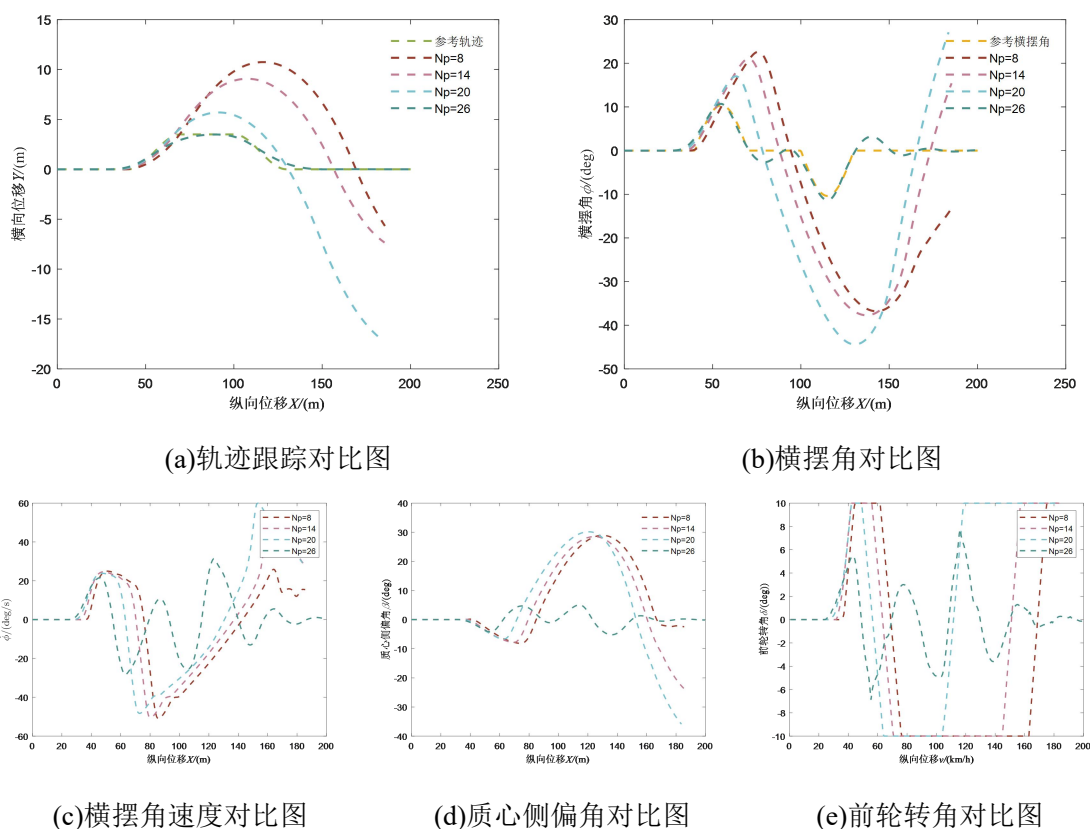


图 4-7 115km/h 多固定预测时域跟踪效果对比图

表 4-6 115km/h 跟踪数据对比表

预测时域 N_p	有效性判定	最大横向误差/(m)	最大横摆角误差/($^{\circ}$)	最大质心侧偏角/($^{\circ}$)	最大横摆角速度/($^{\circ}$ /s)
8	失败	—	—	—	—
14	失败	—	—	—	—
20	失败	—	—	—	—
26	有效	0.61	0.32	4.60	31.28

由图 4-7 和表 4-6 可知。当车速为 115km/h 时，由于高速工况下相关的车辆动力学特性发生巨大变化，在较小的预测时域下，车辆的横摆角速度、质心侧偏角出现大幅度的波动，超出了车辆稳定性控制的界限，其跟踪的精度以及行驶的稳定性的均难以保证，故较小的三组预测时域均导致了跟踪过程的失败。当 $N_p=26$ 时，由于车辆具有较长的预瞄距离，控制器考虑了将来更多的轨迹变化趋势，故其基本可以实现目标轨迹的跟踪，但精度同样有所下降。

综合上述对比实验分析可知，由于实际车辆在行驶过程中时常会因为各种因素进行加减速，若 MPC 轨迹跟踪控制器采用恒定不变的预测时域，其可能仅适用于固定速度下的跟踪而无法保证各类加减速工况下的跟踪精度以及行驶稳定性。在车辆

处于低速时，控制器采用较小的预测时域类似较短的“预瞄距离”可以达到较高的轨迹跟踪精度以及行驶稳定性。而车辆在高速行驶时应当采取更远的“预瞄距离”即更大的预测时域来考虑更多的未来轨迹的变化趋势以及时应对道路轨迹变化。

4.2.2 变预测时域控制器函数的拟合

经过对 4.2.1 节的对比仿真实验分析，为了提高自动驾驶车辆在不同速度工况下的适应能力，LTV-MPC 轨迹跟踪控制器应灵活的根据当前车辆的低、中、高行驶车速动态更新控制器的预测时域的大小来实现灵活控制跟踪目标轨迹。

以 4.2.1 中的四组速度与预测时域的对应组作为数据拟合来源，如下表所示。

表 4-7 行驶速度与预测时域对应表

速度/(km/h)	预测时域 N_p
36	8
72	14
90	20
115	26

将表 4-7 中四组数据来源在 matlab 进行多项式拟合，拟合结果如下函数所示。

图 4-8 为预测时域与车速的拟合函数关系曲线。

$$N_p = \begin{cases} 8, & v_x < 36 \\ \text{int}(-0.000042866v_x^3 + 0.01116v_x^2 - 0.6944v_x + 20), & 36 \leq v_x \leq 120 \\ 30, & v_x > 120 \end{cases} \quad (4-12)$$

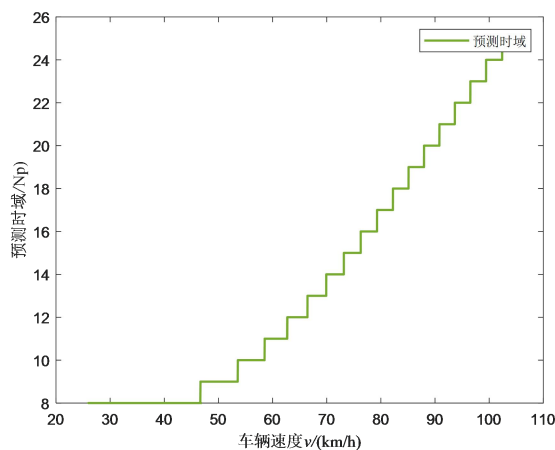


图 4-8 预测时域与车速的拟合函数关系曲线

4.3 实验结果及分析

在 4.1 节中验证了在 LTV-MPC 轨迹跟踪控制器之上添加 RLS 算法实时轮胎刚度估计的方法在轨迹跟踪精度以及行驶稳定性的提升。4.2 节分析了阶梯车速下控制器预测时域取值不同对轨迹跟踪控制的影响，最终拟合出变预测时域随速度控制函数。为验证上述两种方法结合对轨迹跟踪效果提升的有效性，下面在 RLS 轮胎刚度实时估计的方法之上结合预测时域自适应方法进行实验。基于 RLS 变预测时域 LTV-MPC 轨迹跟踪控制原理图如 4-9 所示。同时在超车轨迹下车辆变速开展实验，车辆速度变化图像如图 4-10 所示。

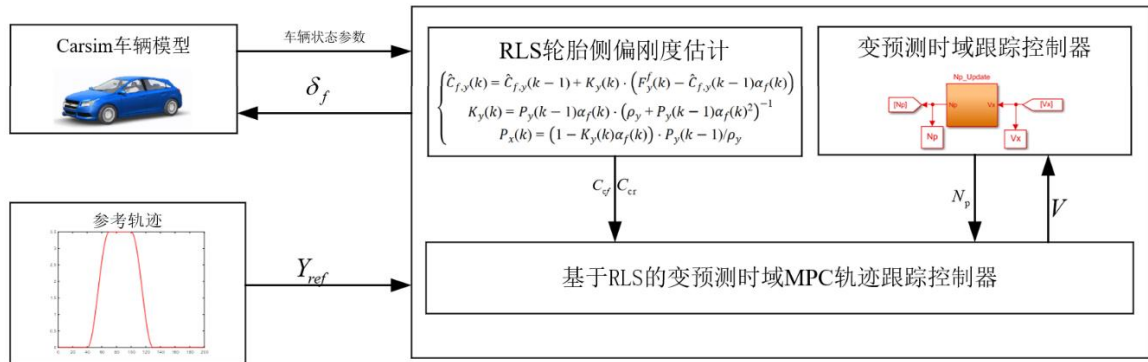


图 4-9 基于 RLS 变预测时域 LTV-MPC 轨迹跟踪控制原理图

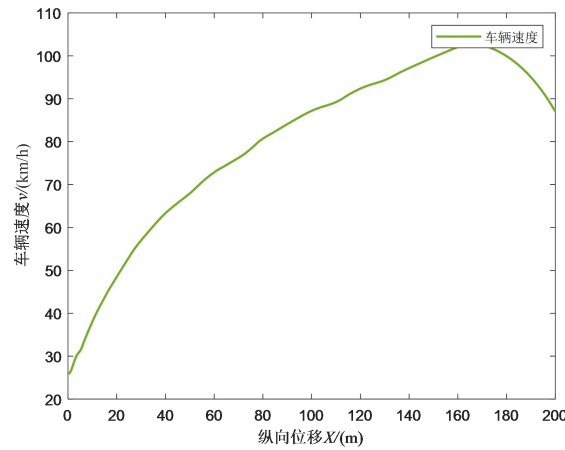
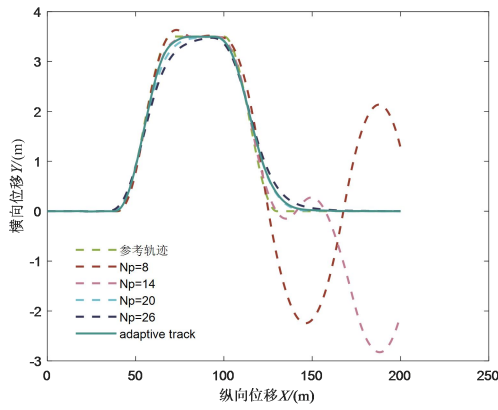
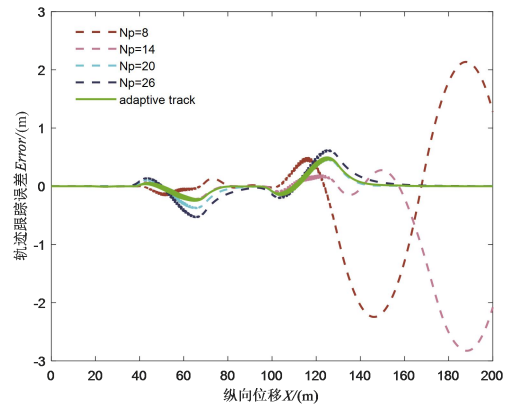


图 4-10 仿真中车辆速度变化

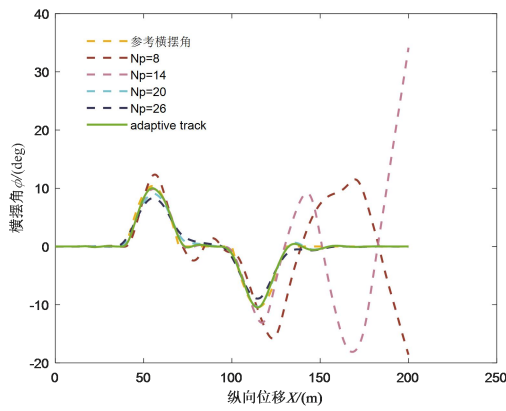
仿真实验对比了自适应控制器与四组分别在固定时域($N_p=8$ 、 $N_p=16$ 、 $N_p=20$ 、 $N_p=26$)下的控制器，试验在变速工况下进行。对应的超车轨迹曲线跟踪结果如图 4-11 所示。



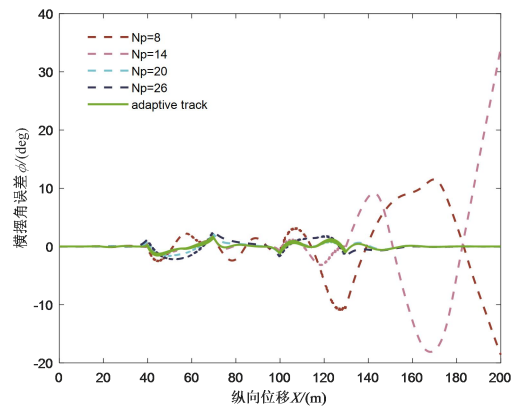
(a) 轨迹跟踪对比图



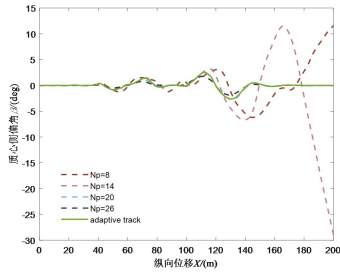
(b) 轨迹跟踪横向误差对比图



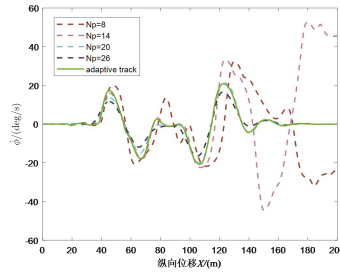
(c) 跟踪横摆角对比图



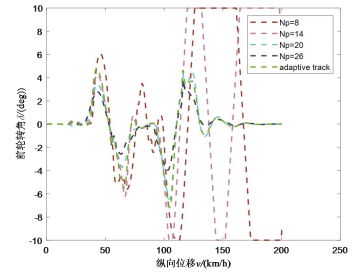
(d) 横摆角误差对比图



(e) 质心侧偏角对比图



(f) 横摆角速度对比图



(g) 前轮转角对比图

图 4-11 变预测时域轨迹跟踪结果对比图

由上图 4-11 仿真结果可知,在该工况下当预测时域 $N_p=8、14$ 时会因为在纵向距离为 130m 后车速过快但“预测距离过短”产生较大的前轮转角,使得车辆质心侧偏角发生急剧变化,车辆行驶稳定性快速降低,使得车辆不能响应期望的转角,造成轨迹跟踪失败,且在 140m 后其质心侧偏角已经超出车辆行驶的安全范围,导致了车辆失稳。从轨迹跟踪的精度来对比,自适应预测时域轨迹跟踪控制器的最大横向误差为 0.45m, $N_p=20、26$ 的固定预测时域控制器的最大横向误差为 0.48m 和 0.63m, 相较之具有更好的轨迹跟踪精度。其次改自适应预测时域轨迹跟踪控制器的最大横

摆角速度为 20.82deg/s，最大质心侧偏角为 2.62deg，故其车辆的行驶稳定性均在可控范围之内。所以在兼顾车辆的控制精度以及行驶稳定性的情况下，自适应预测时域轨迹跟踪控制器相较于多种固定预测时域轨迹跟踪控制器具有更好的速度普适性以及控制稳定性。

4.5 本章小结

本章首先通过对轮胎侧偏刚度的实时估计，提高了车辆动力学模型的精度，进而优化了无人驾驶车辆的轨迹跟踪性能。同时，通过探究车速与预测时域对跟踪效果的影响，设计了变预测时域的 LTV-MPC 跟踪控制器，以适应不同行驶工况下的控制需求。

在轮胎侧偏刚度估计方面，本章首先介绍了轮胎侧偏刚度对车辆模型线性化的重要性，以及其在变速工况下的变化特性。然后，基于车辆状态信息和惯性参数的历史数据，采用递推最小二乘法对轮胎侧偏刚度进行实时估计。此外，本章还通过实验对比了基于 RLS 方法实时估计轮胎侧偏刚度的 LTV-MPC 控制器与固定前后侧偏刚度控制器的性能。实验结果表明，采用实时估计轮胎侧偏刚度的 LTV-MPC 控制器在轨迹跟踪效果上优于固定侧偏刚度控制器，通过仿真试验验证该方法能够有效地提高车辆动力学模型的精度，并改善轨迹跟踪效果。在变预测时域 LTV-MPC 跟踪控制器设计方面，本章分析了此控制器受预测时域的影响，特别是在不同车辆速度下采用合适预瞄距离的重要性。通过探究车速与预测时域对跟踪效果的影响，设计了能够根据车速实时调整预测时域的 LTV-MPC 跟踪控制器。这种设计提高了车辆在不同速度下的轨迹跟踪性能和适应性。

综上所述，本章通过实时估计轮胎侧偏刚度和设计变预测时域 LTV-MPC 跟踪控制器，优化了自动驾驶车辆的轨迹跟踪性能。这些方法对于提高自动驾驶车辆的操纵稳定性和行驶安全性具有重要意义，为自动驾驶技术的实际应用提供了有力的技术支持。

第 5 章 自动驾驶轨迹跟踪实车实验与分析

5.1 实验平台搭建

本小节将搭建实车实验平台，实车硬件方面，基于线控底盘进行搭建，对传感器的选型进行介绍，并对传感器进行实车安装部署与调试。上层软件方面，在 Linux 系统下，部署 ROS 系统、Autoware 自动驾驶平台和本文所涉及的算法功能包，以及其他环境依赖部署到工控机上，完成实验平台的搭建，实车实验硬件组成如图 5-1 所示。



图 5-1 实车实验硬件组成示意图

5.1.1 硬件平台搭建

(1) 线控底盘

近年来，汽车逐渐向网联化、智能化、电动化和共享化的方向转型发展，同时汽车底盘也开始趋向集成化。线控底盘技术是汽车实现转型发展的重要技术，能够在技术层面为汽车行业开发智能座舱和推动智能汽车实现自动驾驶提供强有力的支持^[68]。与传统汽车相比，融合了底盘线控技术的智能网联汽车具有可控性强、控制精度高、机械结构紧凑、液压结构简单、运动状态监测能力强、信息反馈效率高等优势，能够利用整车控制器（Vehicle Control Unit, VCU）来集中处理车辆信息，并将经过处理的信息传输到底盘系统当中，以便底盘系统根据指令控制汽车行驶，精准感知车辆的行驶状态，确保车辆能够安全稳定运行，底盘线控系统控制如图 5-2

所示。

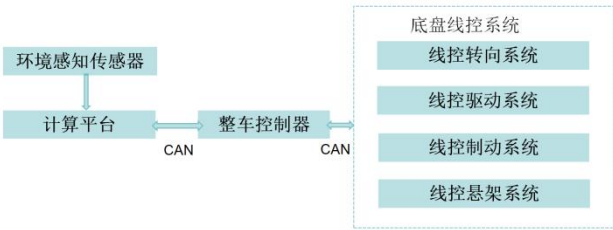


图 5-2 底盘线控系统控制框图

本文选择的是纯电动线控底盘，如图 5-3 所示。该底盘采用四轮毂电机进行驱动，制动采用电液压盘式驱动，悬挂为前后双叉臂独立悬挂，车架材料为高强度铝合金。融合了线控及分布式驱动技术，加强稳定性的同时也提高了车辆的应用空间和运载能力，线控底盘主要参数如下表 5-1，其转向电机主要参数如表 5-2 所示，轮毂电机参数如表 5-3 所示。



图 5-3 PIX 线控底盘

表 5-1 PIX 线控底盘主要参数

底盘参数	数值
轮胎规格	205/45R17
轮距	1355mm
最高车速	40km/h
最大负载	1000kg
续航里程	70km
最大爬坡度	20°
最小转弯半径	3m
轮胎最大转弯角度控制	外轮 22° 内轮 32°

表 5-2 PIX 线控转向电机主要参数

参数名称	数值
最大输出功率	720W
额定电压	12V DC
额定转速	1050rpm
额定扭矩	3.2N · m
工作温度	-40℃~+85℃
转动范围	± 124℃
通信速度	500 k/s

表 5-3 PIX 线控轮毂电机主要参数

参数名称	数值
最大输出功率	4424*4W
额定电压	72V
最大转矩电流	58.1A
控速精度	1 rpm/bit
最大转速	811.9rpm
最大转矩	165.3*4N · m
车速范围	± 60Km/h
电机尺寸	10 英寸

(2) 激光雷达

激光雷达是一种利用激光技术进行探测和测距的设备，用于三维环境感知和地图创建，通过激光光束扫描周围环境，从而生成高分辨率的点云数据，广泛应用于自动驾驶、机器人导航等领域。激光雷达的主要特点包括测量精度高、速度快、抗干扰能力强等。在无人驾驶轨迹跟踪系统中，激光雷达通常与其他传感器（如毫米波雷达、摄像头、超声波传感器等）一起使用，以提供全方位的环境感知能力。通过融合多种传感器的数据，系统可以更加准确地识别道路标线、障碍物、行人等，从而实现可靠的轨迹跟踪和导航。

本文选用的是由 Velodyne 公司生产的 VLP-16 线激光雷达，具体参数如表 5-4 所示。

表 5-4 VLP-16 线激光雷达

激光雷达参数	数值
扫描频率 (HZ)	5~20
最大水平可视范围 (°)	360
最大垂直可视范围 (°)	30
最远检测距离 (m)	100
角度分辨率 (°)	0.1~0.4
精度误差 (°)	-0.03 ~ +0.03

激光雷达通过螺栓固定安装在车辆前部，如图 5-4 所示



图 5-4 VLP-16 线激光雷达

(3) 相机和惯性导航系统

在无人驾驶技术中，相机通常被称为视觉传感器，通过捕捉周围环境的图像，为无人驾驶车辆提供了丰富的视觉信息，帮助车辆进行导航、障碍物检测、交通标志识别等任务。本文考虑到成本选用 FLIR 公司生产的 BFS-U3-04S2C-CS 单目相机作为实验相机如图 5-5 所示，此单目相机基本参数如表 5-5 所示。



图 5-5 FLIR 单目相机

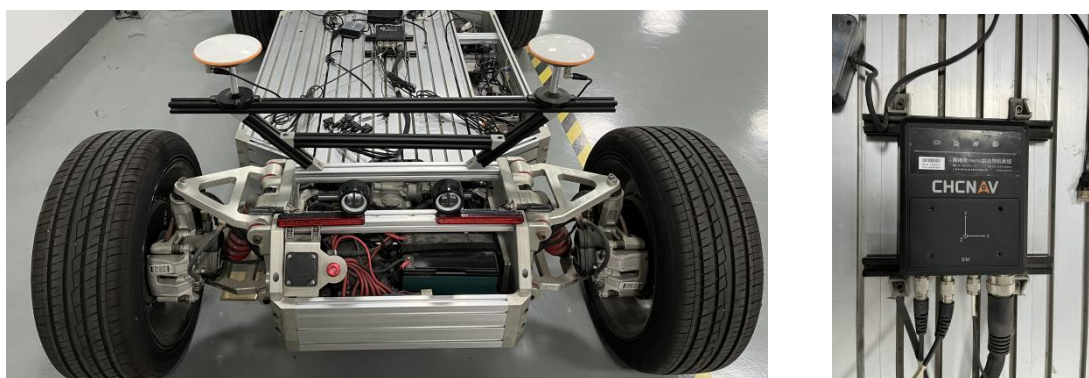
表 5-5 FLIR 单目相机基本参数

单目相机基本参数	数值
尺寸 (mm)	29×29×30
最大每秒传输帧数 (FPS)	750×540
最大分辨率	522
像元尺寸 (μm)	6.9
成像缓存 (MB)	240
数据接口	USB3.1Gen1

组合惯导 (GNSS/IMU) 系统是一种将全球导航卫星系统 (GNSS) 和惯性导航系统 (IMU) 进行集成的导航系统。该系统结合了 GNSS 的高精度定位能力和 IMU 的独立工作能力, 以提供更为准确、稳定和可靠的导航和定位服务。华测公司生产的 CGI-410 厘米级组合导航系统是一款高性能的导航设备, 它结合了全球导航卫星系统 (GNSS) 和惯性导航系统 (IMU) 的优点, 提供了厘米级的定位精度。该系统内置了高精度 MEMS 陀螺仪和加速计, 能够实时测量载体的角速度和加速度, 并通过积分运算得到载体的速度、位移和姿态等信息, 在线控底盘上惯导的天线及其主机安装位置如图 5-6(a)和 5-6(b)所示。CGI-410 还支持外接里程计信息进行辅助, 这有助于进一步提高导航的精度和稳定性。此外, 该系统支持 WIFI 无线接入和网页访问, 其基本参数如表 5-6 所示。

表 5-6 华测 CGI-410 组合惯导基本参数

组合惯导基本参数	数值
姿态精度	0.1° (基线长度≥2m) 单点 L1/L2:1.2m
定位精度	DGPS:0.4m RTK:1cm+1ppm
数据更新频率	100HZ
物理尺寸	162×120×53mm
重量	0.5kg
	3 × RS232 1 × RS422 1 × CAN 1 × Micro USB 接口
外部接口	2 × GNSS 天线接口 1 × 4G 天线接口 1 × 电源接口



(a) 惯导天线

(b) 惯导主机

图 5-6 华测 CGI-410 组合惯导 GNSS

(4) CAN 总线

CAN 总线（Controller Area Network）是一种广泛应用于汽车和工业自动化领域的串行通信网络。CAN 网络由含有 CAN 控制器和 CAN 收发器的单元、2 个终端电阻、2 条传输数据线(CAN-H、CAN-L)组成。CAN 控制器用来接收微处理器传来的信息,对这些信息进行处理并传给 CAN 收发器,同时 CAN 控制器也接收来自 CAN 收发器传来的数据,对这些数据进行处理,并传给控制单元的微处理器。CAN 收发器接收控制器送来的数据,并将其发送到 CAN 数据传输总线上,同时 CAN 收发器也接收 CAN 数据总线上的数据,并将其传给 CAN 控制器。终端电阻的作用是防止数据在到达线路终端后像回声一样返回,并因此干扰原始数据,从而保证了数据的正确传送,终端电阻通常安装在控制单元内。传输数据线是两条双向数据线,分为高位(CAN-H)和低位(CAN-L)数据线,主要作用是进行网络数据传输。

PCAN 是 PEAK-System Technik 公司的一个产品系列如下图 5-7 所示,一种通过 USB 接口连接 PC 和 CAN 网络的接口,能够通过 CAN 网络进行通信和数据交换。如图所示,一端为 DB9 引脚用于连接 CAN 总线,另一端 USB 连接工控机。



图 5-7 PCAN

5.1.2 软件系统平台

(1) ROS 操作系统

Ubuntu 是一种基于 Linux 内核的开源操作系统，是一款功能强大、稳定可靠、易于使用且开源的操作系统。本文也是基于上述特点选取该操作系统作为控制器的运行环境，为了实现软件与硬件之间的高效通讯，采用 ROS(Robot Operating System) 作为内部通讯框架，以确保通讯的便捷性、可靠性和实时性。ROS 系统也叫机器人操作系统，是一个灵活的框架和工具集，用于构建机器人软件。ROS 是一个在 Linux 上运行的集成开发环境（IDE）和中间件，使用基于消息的通信机制，允许不同的模块之间通过发布和订阅消息进行通信。ROS 采用的是进程（也称为 Nodes）的分布式框架，ROS 中的进程可分布于不同主机，不同主机协同工作，从而分散计算压力。ROS 语言是独立的，包括 Java，C++，Python 等，为了支持更多应用开发和移植。ROS 可采用组件化方式集成一些工具和软件到系统中并作为一个组件直接使用，如 RVIZ（3D 可视化工具），可以根据 ROS 定义的接口在其中显示机器人模型等，组件还包括仿真环境和消息查看工具等。在 ROS 系统中，通讯方式主要分为四类：话题（Topic）、服务（Service）、参数服务器（Parameter Server）以及动作（Action）通讯。基本通信机制主要有如下三种实现策略：话题通信(发布订阅模式)、服务通信(请求响应模式)、参数服务器(参数共享模式)，其中话题通讯是 ROS 中使用频率最高的一种通信模式。

3. Autoware 是一个开源的自动驾驶技术平台，它提供了一套完整的软件栈，用于实现自动驾驶汽车的各个方面，包括但不限于感知、决策规划和控制。Autoware 旨在支持多种车辆和硬件平台，提高了其通用性和灵活性。Autoware 主体框架如图 5-8 所示。

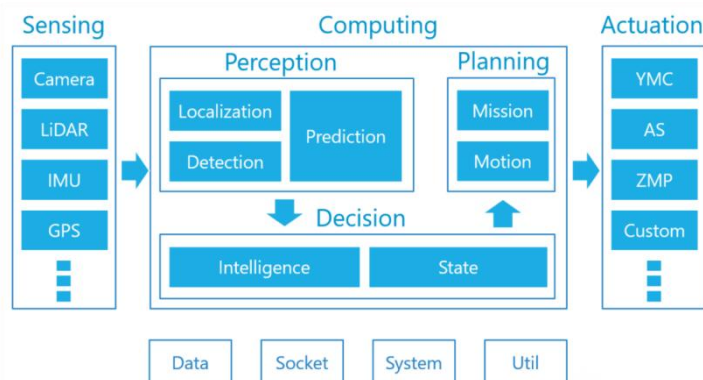


图 5-8 Autoware 主体框架图

其中 sensing 模块对应的是各类传感器对真实世界中各类数据的采样，例如 camera 采样图像、LiDAR 采样激光点云等，采样数据属于未处理的原始数据，需要输入到 computing 模块进行计算处理，computing 模块主要是为了对传感器采样的原始数据进行加工处理，最后以为实现安全高效的导航为目的，将规划结果输出给 actuation 模块。其中 computing 模块主要分为三个小模块，perception 感知模块，这部分要处理 localization 通过车辆当前采集传感器数据和已有地图进行自身定位，若无地图需要通过 SLAM 构建地图，然后 detection 模块负责检测周围与车辆有场景交互的车辆、行人等，prediction 模块会对检测初得物体进行未来预测估计，以便提前规划防止碰撞。decision 决策模块，根据之前感知的结果，Autoware 决策一个由有限状态机表示的驾驶行为，以便可以选择适当的计划功能，当前的决策方法是基于规则的系统。Planning 规划模块，主要是根据决策和起始点和目标点，采用 mission 和 motion 模块可以计算出一条无障碍的路径。actuation 模块表示驱动器模块，如 YMC 驱动器，是接收 planning 模块出来的规划结果，使驱动器实现驱动控制。

5.2 轨迹跟踪实车实验

5.2.1 点云地图建立

本文实车实验拟采用 PIX 线控底盘于亚太机电公司厂区内开展，由于点云地图为自动驾驶车辆提供了一个精确的环境模型。通过激光雷达等传感器采集的点云数据能够详细地反映道路结构、障碍物位置以及周边环境的三维信息。其次，点云地图可以用于定位。在轨迹跟踪过程中，车辆需要实时地知道自己在地图中的准确位置。通过将实时采集的点云数据与预先构建的点云地图进行匹配，可以实现高精度的定位，这是进行高精度轨迹跟踪的前提条件，故在进行轨迹跟踪实车实验之前，本文先采集并构建了厂区内激光点云地图。

在 ROS 框架下，使用 Autoware 建立点云地图的过程涉及到多个步骤，包括启动仿真环境、录制点云数据、修改激光雷达的 launch 文件、确保 IP 地址在同一网段、启动 Autoware 和激光雷达、回放和录制点云数据、以及最后进行建图。从启动仿真环境开始，通过 `roslaunch vehicle_sim_launcher world_test.launch` 指令激活 Gazebo 并加载测试世界。随着仿真环境的加载，Rviz 视觉工具和 `rqt_robot_steering` 控制插件也被启动，允许用户通过界面控制车辆模型，并进行行驶。在车辆行驶过程中，需

要确保激光雷达传感器的坐标系与 Autoware 设定的 velodyne 坐标系相匹配,并监听正确设置的话题/`points_raw` 以接收原始点云数据。

在车辆走完厂区内所需路径后,通过按键 `Ctrl+C` 来停止数据的记录。如果激光雷达设备需要特别配置,比如修改 IP 地址或者调整扫描参数,必须修改相应的 launch 文件以确保硬件正确启动并与 ROS 系统通信。为了顺利进行通讯,网络 IP 和设备 IP 应该配置在同一网段内。

接下来,通过进入 Autoware 的 `ros/` 目录并执行 `./run` 命令来启动 Autoware 的各个组件,包括感知、定位、规划等模块。随后,可以在 Autoware 的 Sensing 界面下选择对应的激光雷达配置文件,确保点云数据能够发布到/`points_raw` 话题。在地图构建过程中使用 Autoware 提供的可视化工具 `rviz` 来实时查看点云数据和构建的地图,在 `RViz` 界面中添加适当的显示类型(如 `PointCloud2`)并订阅相应的点云话题。最终在完成地图构建后,将地图命名并保存为 `PCD` 文件,实验采集厂区内点云地图如下图所示。

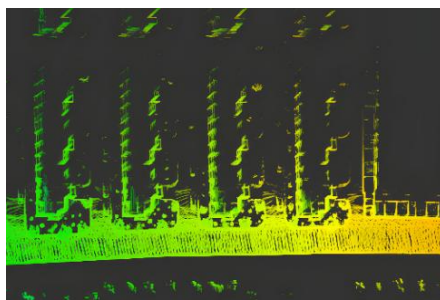


图 5-9 实验区点云地图

5.2.2 实车实验

此参考轨迹为预先手动遥控并记录保存下的轨迹,实车实验拟进行跟踪的轨迹如下图所示。

(1) 手动遥控车辆记录参考目标轨迹

在确保 ROS 环境配置成功后,启动 Autoware,同时加载激光雷达、相机、IMU 的传感器驱动。在 Autoware 运行后,打开终端启动 `rviz`,添加点云以及轨迹以实时察看车辆位置和姿态信息,如下图所示。随后使用遥控装置平均以 `1m/s` 的速度手动驾驶车辆并沿着预设轨迹(如图 5-10)行驶。最后使用 `roslaunch record` 命令记录车辆行驶过程中的轨迹数据,图 5-11 为打开 `rviz` 后的可视化跟踪过程图。

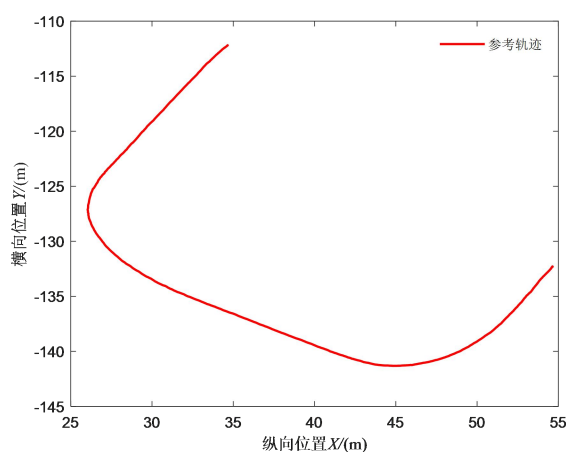


图 5-10 实车跟踪目标轨迹

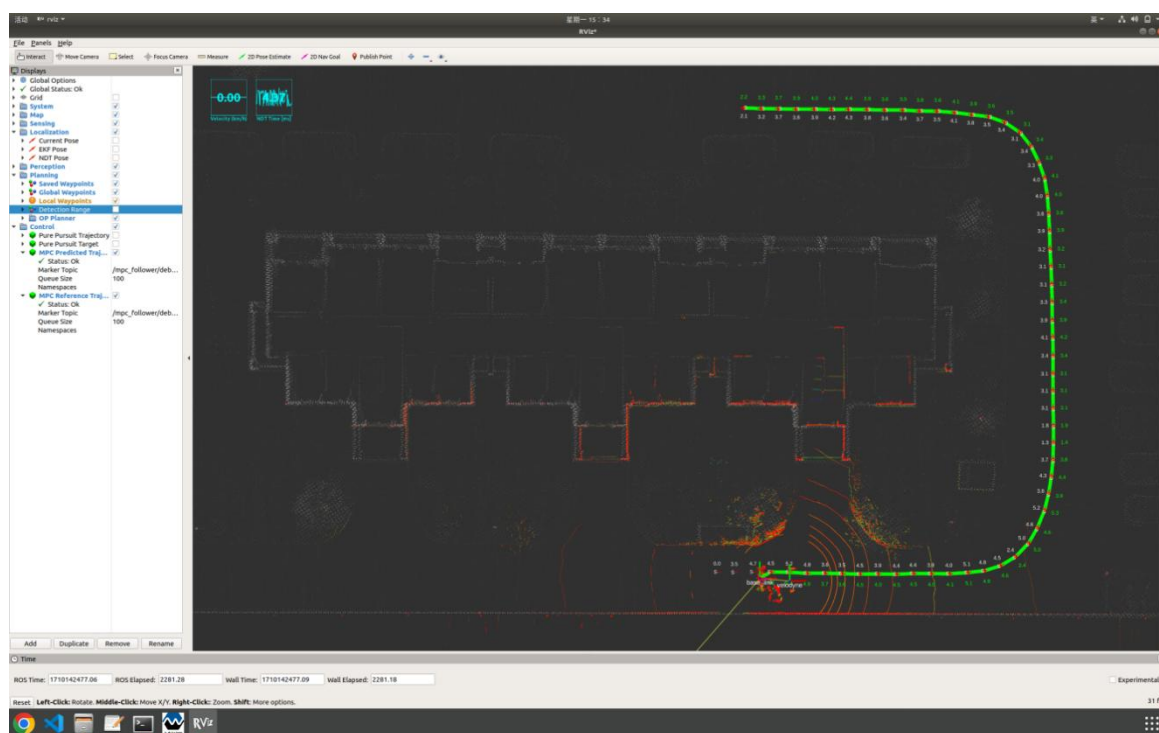


图 5-11 rviz 可视化跟踪轨迹图

(2) 设置 LTV-MPC 轨迹跟踪算法

在 Autoware 中，设置同第四章相同内容的预测时域、控制间隔、车辆动力学模型等信息。随后将之前保存的参考轨迹文件加载至 Autoware 之中，作为 LTV-MPC 轨迹跟踪算法的目标轨迹。其中 LTV-MPC 算法主要实现在 `mpc_follower` 节点中。该节点接收 `/vehicle_status`、`/vehicle_cmd` 和 `/trajectory` 等消息，其中 `/vehicle_status` 消息包括车辆状态信息（例如位置、速度、方向等），`/vehicle_cmd` 消息包括车辆控制指

令（例如方向盘转角、油门踏板位置、刹车踏板位置等），/trajectory 消息包括规划的车辆轨迹。通过对这些消息的处理，mpc_follower 节点可以计算出最优的车辆控制指令，并将其发送给/vehicle_cmd 话题，从而实现对车辆的控制。

```

/* 控制参数设置*/
double ctrl_period_;
double steering_lpf_cutoff_hz_;
double admissible_position_error_;
double admissible_yaw_error_deg_;
double steer_lim_deg_;
double wheelbase_;

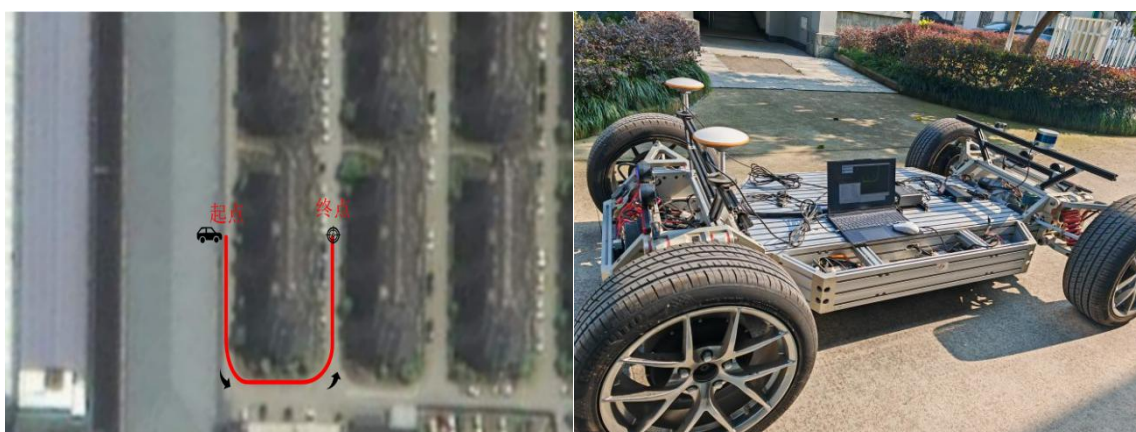
bool enable_path_smoothing_;
bool enable_yaw_recalculation_;
int path_filter_moving_ave_num_;
int path_smoothing_times_;
int curvature_smoothing_num_;
double traj_resample_dist_;
struct MPCParam
{
    int prediction_horizon;
    double prediction_sampling_time;
    double weight_lat_error;
    double weight_heading_error;
    double weight_heading_error_squared_vel_coeff;
    double weight_steering_input;
    double weight_steering_input_squared_vel_coeff;
    double weight_lat_jerk;
    double weight_terminal_lat_error;
    double weight_terminal_heading_error;
    double zero_ff_steer_deg;
    double delay_compensation_time;
};

```

图 5-12 部分控制参数设置

(3) 自动驾驶轨迹跟踪

将车辆至于目标跟踪轨迹起点位置附近，如下图所示，在确认所有配置无误后按下遥控装置的“mode”按钮将车辆切换至自动驾驶模式，此时遥控装置屏幕显示“Self Driving”表示模型切换成功。为实车考虑实验的安全性、车辆状况以及环境条件等安全因素考虑，车辆开始沿着目标轨迹以 5m/s 自动驾驶，车辆在两个转向处位置如下图所示。在自动驾驶过程中，通过 rviz 或其他监控工具实时查看车辆的行驶状态和轨迹跟踪情况。必要时，准备随时接管车辆控制以确保安全。在实验结束后，收集自动驾驶过程中的形式数据，包括车辆的位置和姿态信息，用于后续实验数据对比，其跟踪过程如下图 5-13 所示。



(a)目标轨迹卫星图

(b) 车辆起始状态



(c) 车辆第一处转向实况

(d) 车辆第二处转向实况

图 5-13 实车轨迹跟踪过程

5.3 车辆实验结果

在实车实验结束后,将录制好的参考轨迹数据包以及跟踪轨迹数据包利用 `rosvbag` 进行解析播放,订阅提取两者 `ndt_pose` 和 `twist_cmd` 话题。将订阅到的实验结果导出并绘制图像如下图 5-14 所示。

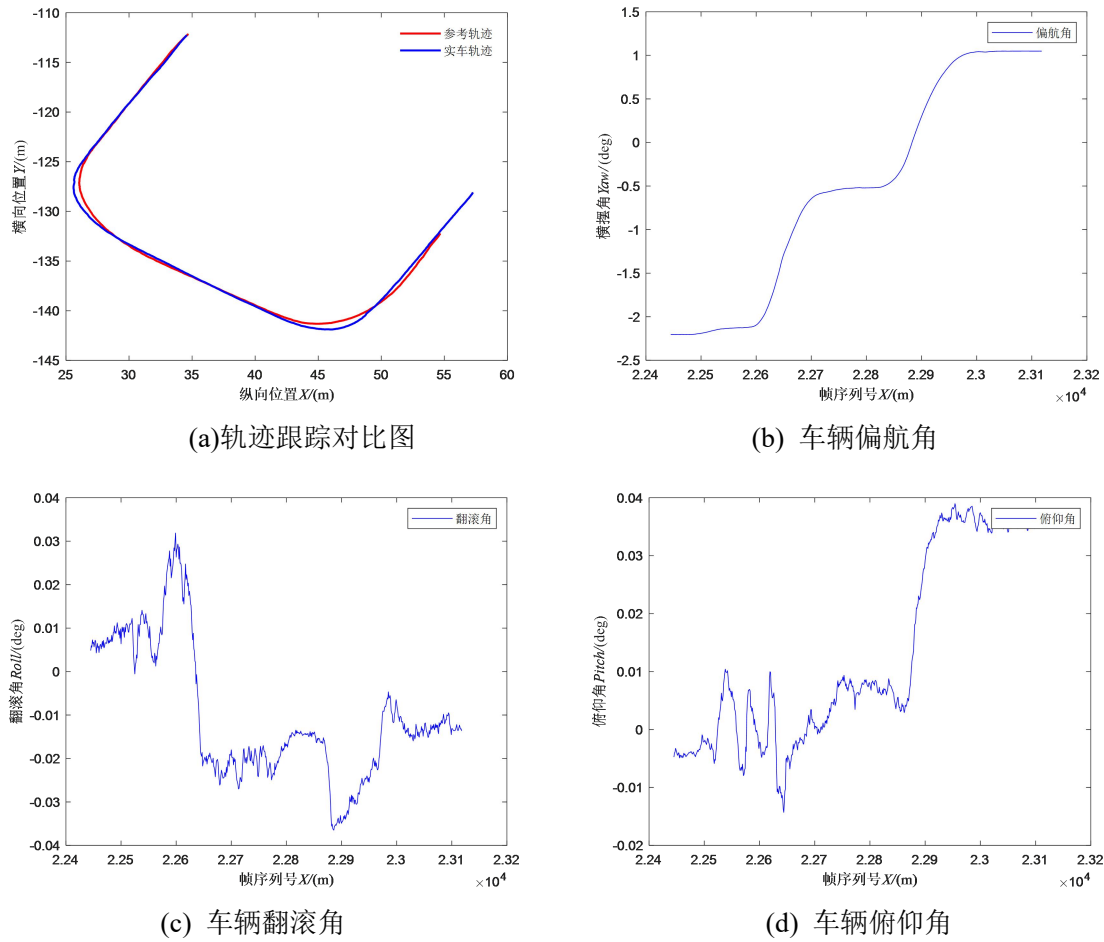


图 5-14 实车轨迹跟踪结果图

图 5-14(a)对比了实车采集的参考轨迹和实车轨迹之间的差异。可以看到，实车轨迹与参考轨迹在大部分时间内都保持高度重合，这表明所设计的 LTV-MPC 轨迹跟踪控制器在跟踪方面的良好性能。此外，在车辆转向时，实车轨迹能够迅速响应并贴近参考轨迹，体现了控制器的实时性与准确性。

图 5-14(c)图 5-14(d)分别呈现了随着仿真帧序列的车辆翻滚角以及俯仰角变化。其中车辆俯仰角在-0.01 到 0.03 弧度范围内波动，这反映了车辆在行驶过程中对于道路起伏或加速减速时的姿态调整；车辆翻滚角的数据点分布在-0.04 到 0.03 弧度之间，表明车辆在行驶过程中翻滚角的变化相对较小，进而说明车辆在跟踪过程中侧倾现象较弱，稳定性较好。图像(b)记录了偏航角随帧序列号的变化情况。偏航角的变化范围在指定的 0.5 到 1.5 弧度之间，这代表了车辆在行驶过程中方向的变化。当车辆需要转弯或跟随曲线道路时，偏航角会相应调整产生较大变化，以确保车辆能够按照预定轨迹行驶。

从控制器跟踪性能的角度分析，LTV-MPC 控制器在车辆动力学轨迹跟踪中表现出色，实车轨迹与参考轨迹的高度重合证明了控制器的准确性和有效性。这表明 LTV-MPC 算法能够实时计算并优化车辆的控制输入，以实现精确的轨迹跟踪；从车辆响应方面分析，从俯仰角、偏航角和翻滚角的变化图中可以看出，车辆在行驶过程中能够根据道路条件和控制器指令做出相应的姿态调整。这种响应能力对于实现稳定的轨迹跟踪至关重要。道路环境适应性分析方面，设计的 LTV-MPC 控制器在无论是直线道路还是曲线道路，控制器都能够迅速调整车辆姿态和行驶方向，以保持与参考轨迹的高度一致性，显示了 LTV-MPC 算法在处理复杂驾驶环境中的潜力。

5.4 本章小结

本章详细探讨了实验平台的搭建过程，并基于搭建的平台进行了轨迹跟踪实车实验，最终对实验结果进行了深入分析。

在硬件平台搭建部分，本章首先介绍了实验平台所需的硬件设备及其功能。特别强调了软硬件连接框架的重要性，介绍了 CAN 总线负责车辆内部各个控制单元之间的数据交换，是实现车辆自主控制的关键，详细描述了传感器和执行器的选型依据和安装方法，本车的软硬件设备是车辆完成环境感知和底盘执行的基础。

在轨迹跟踪实车实验部分，首先介绍了点云地图的建立过程，这是实现轨迹跟踪的基础。通过激光雷达等传感器采集的环境数据，构建出车辆行驶环境的精确模型。随后详细描述了实车实验的过程，包括实验场景的选择、实验参数的设置以及实验数据的收集等。这为我们验证轨迹跟踪算法的有效性实验提供了重要的数据支持。

在车辆实验结果部分，对收集到的实验数据进行了深入分析。通过对比实车轨迹与参考轨迹的偏差以及车辆在行驶过程中的姿态变化，包括俯仰角、偏航角和翻滚角等，验证了 LTV-MPC 轨迹跟踪控制器在实车应用中的跟踪稳定性和控制精度。

第 6 章 结论与展望

6.1 研究成果总结

自动驾驶作为未来交通领域的核心发展方向，基于模型预测控制的轨迹跟踪控制方法在精准度、稳定性和实时性上的提升对自动驾驶的发展具有至关重要的推动作用。对于全文的主要研究工作与结论如下：

(1) 完成了 LTV-MPC 轨迹跟踪控制器的构建。在车辆三自由度动力学模型的基础之上，经过车辆非线性系统的线性化及离散化后，考虑到控制器的可控性以及车辆行驶稳定性，将控制器的控制量转化为控制增量的形式。最后构建了关于车辆系统状态量以及前轮转角控制增量的目标函数并添加相关约束。针对道路曲率自适应 LTV-MPC 轨迹跟踪控制器设计。基于对目标函数权重矩阵对轨迹跟踪控制性能的影响的考虑，详细描述了 Simulink 与 CarSim 联合仿真平台的构建过程。分别分析了目标函数的权重矩阵 Q 、 R 因道路曲率变化而对轨迹跟踪的影响效果，设计出道路曲率自适应函数。最后经过与传统 MPC 和 LTV-MPC 轨迹跟踪控制器进行对比，验证了改进后控制器在精度以及稳定性上的提升。

(2) 针对基于 RLS 优化下的变预测时域控制器设计。考虑到车辆轮胎线性侧偏刚度对于车辆动力学模型构建的在高速等复杂工况的精度影响较大，设计了基于最小二乘法的侧偏刚度实时估计法，将实时计算的轮胎刚度输入至车辆动力学模型，从而避免车辆在工作在轮胎“非线性区域”时车辆稳定性和操控性的降低。其次，为了提高 LTV-MPC 轨迹跟踪控制器在变速工况下的适应能力，试验首先分析了车辆从低速到高速的阶梯速度下，四组从低到高的固定预测时域下的轨迹跟踪效果，通过其控制变化规律总结拟合出变预测时域控制函数以及时应对工况变化。最后，将所设计的变预测时域控制器同固定预测时域控制器通过横向跟踪误差、横摆角误差、横摆角速度和质心侧偏角等指标的对比，分析验证其有效性。

(3) 验证了实车状态下 LTV-MPC 轨迹跟踪算法的有效性。详细阐述了实车验证实验的车辆相关软硬件及其环境搭建准备过程。在 ROS/Autoware 中，设置预测时域、控制间隔、车辆动力学模型等相关参数和信息。通过 `mpc_follower` 节点可以计算出最优的车辆控制指令，并将其发送给 `/vehicle_cmd` 话题，通过 PCAN 将控制信号发送

至底盘实现对车辆的控制。通过对比实车轨迹与参考轨迹的偏差，验证了轨迹跟踪算法的跟踪精度。同时分析了车辆在行驶过程中的姿态变化，包括俯仰角、偏航角和翻滚角等，证实了在 LTV-MPC 轨迹跟踪控制器下车辆轨迹跟踪过程中的行驶稳定性。

6.2 未来研究工作展望

本文是基于 MPC 自动驾驶轨迹跟踪控制展开研究，主要集中在算法层面的改进与创新。基于 MPC 自动驾驶轨迹跟踪控制技术的热点问题，从目前的技术状态来看，有必要在以下几个方面继续深入研究：

(1) 计算效率

由于 MPC 的计算过程较为繁琐复杂，在实际车辆部署时需要消耗大量的计算资源，对相关硬件提出了较高的要求。但对算法本身来说，可以通过保证精度和控制效果的前提下继续优化算法或并行计算的方式来提升算法求解效率。

(2) 参数优化

MPC 方法在实际应用中对于控制参数的选择和调整较为敏感。通常需要通过经验或者试错的方法来调整 MPC 控制器的参数，以使其在特定的应用场景中达到最佳性能，可以探索利用机器学习的数据驱动技术和自适应控制技术等方法来自动调整 MPC 控制器的参数，以减少对于人工经验的依赖同时提升该算法在各场景及其工况下的适应性。

(3) 实车验证相关问题

现如今，汽车逐步进入线控时代，全线控车辆相较于传统燃油车具有更高的数据传输效率以及电气化特征。在后续的工作中，有必要将 MPC 轨迹跟踪控制器尝试部署在全线控车辆之上。

参考文献

- [1] 孔大为. 全国机动车保有量 3.9 亿辆驾驶人 4.76 亿人[N]. 人民公安报交通安全周刊, 2021-10-15(001).
- [2] 李克强. 智能电动汽车的感知、决策与控制关键基础问题及对策研究[J]. 科技导报, 2017, 35(14): 85-88.
- [3] 王佳荣. 面向自动驾驶的多传感器三维环境感知系统关键技术研究[D].中国科学院大学(中国科学院长春光学精密机械与物理研究所),2020.DOI:10.27522/d.cnki.gkcg.2020.000142.
- [4] Guan Y, Ren Y, Li S E, et al. Centralized cooperation for connected and automated vehicles at intersections by proximal policy optimization[J]. IEEE Transactions on Vehicular Technology, 2020, 69(11): 12597-12608.
- [5] 龚建伟, 姜岩, 徐威. 无人驾驶车辆模型预测控制[M]. 北京: 北京理工大学出版社, 2014.
- [6] 熊璐, 杨兴, 卓桂荣等. 无人驾驶车辆的运动控制发展现状综述 [J]. 机械工程学报, 2020, 56(10): 127-143.
- [7] 丁代林. 无人驾驶车辆轨迹跟踪控制及其稳定性研究[D].重庆理工大学, 2023.
- [8] Zhang R, Zhang C N. Combined Control Allocation and Sliding Mode Control in the Dynamic Control of a Vehicle with Eight In-Wheel Motors[J]. Journal of Beijing Institute of Technology, 2017, 26(1): 58-66.
- [9] 陈虹著. 系统与控制丛书 模型预测控制[M]. 北京: 科学出版社. 2013.
- [10] 陈慧岩, 熊光明, 龚建伟主编. 无人驾驶车辆理论与设计[M]. 北京: 北京理工大学出版社. 2018.
- [11] 《中国公路学报》编辑部. 中国汽车工程学术研究成果综述 • 2017[J]. 中国公路学报, 2017, 30(06): 1-197
- [12] 杜明博. 基于人类驾驶行为的无人驾驶车辆行为决策与运动规划方法研究[D]. 中国科学技术大学, 2016.
- [13] 曹晓昂. 2017 年智能网联汽车产业盘点[J]. 汽车纵横, 2018, 8(1): 30-32.
- [14] 《智能汽车创新发展战略》(发改产业(2020)202 号)[R]. 北京: 中华人民共和国国家发展和改革委员会, 2020.
- [15] Stano P, Montanaro U, Tavernini D, Tufo M, Fiengo G, Novella L, & Sorniotti A. Model predictive path tracking control for automated road vehicles: A review[J]. Annual Reviews in

- Control,2023, 55(4):194 – 236.
- [16] 安林芳. 智能车辆自动驾驶路径规划研究[D]. 长沙: 湖南大学, 2017.
- [17] K. Kim, B. Kim, K. Lee, B. Ko, and K. Yi. Design of integrated risk management-based dynamic driving control of automated vehicles. IEEE Intelligent Transportation Systems Magazine, 2017, 9(1): 57-73.
- [18] 梁忠超,张欢,赵晶等.基于自适应 MPC 的无人驾驶车辆轨迹跟踪控制[J].东北大学学报(自然科学版),2020,41(06):835-840.
- [19] 吴海东,司振立.基于线性矩阵不等式的智能车轨迹跟踪控制[J].浙江大学学报(工学版),2020,54(01):110-117.
- [20] Rokonzaman M, Mohajer N, & Nahavandi S. Effective adoption of vehicle models for autonomous vehicle path tracking: a switched MPC approach[J]. Vehicle System Dynamics, 2023,61(5):1236 – 1259.
- [21] 刘溯奇,王刚,安伟彪等.基于变预测时域的电动汽车轨迹跟踪控制[J].科学技术与工程,2021,21(17):7348-7354.
- [22] CHENG S, LI L, CHEN X, et al. Model Predictive-Control-Based path tracking controller of autonomous vehicle considering parametric uncertainties and velocity-varying[J]. IEEE Transactions on Industrial Electronics, 2021, 68(9): 8698-8707.
- [23] 贺伊琳,宋若旸,马建.基于强化学习 DDPG 的智能车辆轨迹跟踪控制[J].中国公路学报,2021,34(11):335-348.
- [24] 张栩源,李军.基于 LQR 双 PID 的智能电动汽车轨迹跟踪横纵向协同控制[J].汽车安全与节能学报,2021,12(03):346-354.
- [25] 汤新华,成宇庆,潘树国等.基于时滞补偿的纯追踪控制预瞄距离优化方法[J].中国惯性技术学报,2023,31(09):876-882.
- [26] 姜立标,吴中伟.基于趋近律滑模控制的智能车辆轨迹跟踪研究[J].农业机械学报,2018,49(03):381-386.
- [27] FENTON R E, SELIM I. On the optimal design of an automotive lateral controller[J]. IEEE Transactions on Vehicular Technology, 1988, 37(2): 108 – 113.
- [28] 王银,张灏琦,孙前来等.基于自适应 MPC 算法的轨迹跟踪控制研究[J].计算机工程与应用,2021,57(14):251-258.
- [29] MARINO R, SCALZI S, NETTO M. Nested PID steering control for lane keeping in autonomous

- vehicles[J]. Control Engineering Practice, 2011, 19(12): 1459 – 1467.
- [30] 徐明泽,刘清河.基于 LQR 和 PID 的智能车轨迹跟踪控制算法设计与仿真[J].太原理工大学学报,2022,53(05):877-885.
- [31] KHODAYARI A, GHAFFARI A, AMELI S, et al. A historical review on lateral and longitudinal control of autonomous vehicle motions[C]. 2010 International Conference on Mechanical and Electrical Technology, 2010: 421 – 429.
- [32] 章仁燮,熊璐,余卓平,等. 基于条件积分算法的无人驾驶车辆轨迹跟踪鲁棒控制方法[J]. 机械工程学报, 2018, 54(18): 129 – 139.
- [33] 赵盼. 城市环境下无人驾驶车辆运动控制方法的研究[D]. 合肥: 中国科学技术大学, 2012.
- [34] 聂小博,熊玥,潘勇军.基于模糊 PID 算法的车身稳定控制策略与多工况联合仿真[J].动力学与控制学报,2021,19(03):46-52.
- [35] 张正华. 基于神经网络的无人驾驶车辆轨迹跟踪控制[D].燕山大学,2020.
- [36] 周东昇. 基于预瞄滚动时域 LQR 的无人驾驶汽车路径跟踪研究[D].重庆交通大学,2018.
- [37] Gu J,Fang D.Genetic Algorithm Based LQR Control for AGV Path Tracking Problem[J].Journal of Physics:Conference Series,2021,1952(3).
- [38] 赵熙俊, 刘海鸥, 熊光明, 等. 自动转向滑模变结构控制参数选取方法[J]. 北京理工大学学报, 2011, 31(10): 1174 – 1178.
- [39] 江绍康,江浩斌,马世典.基于反演滑模自适应策略的自动泊车路径跟踪控制[J].重庆理工大学学报(自然科学),2020,34(10):9-16.
- [40] 张杰, 方涛, 郑平平, 等. 基于滑模理论的轨迹跟踪横向控制研究[J]. 农业装备与车辆工程, 2019, 57(03): 13-16.
- [41] Tagne G, Talj R, Charara A. Higher-Order sliding mode control for lateral dynamics of autonomous vehicles, with experimental validation. In: IEEE Intelligent Vehicles Symposium. Gold Coast: IEEE, 2013, 678-683.
- [42] COULTER R C. Implementation of the pure pursuit path tracking algorithm[R]. Pittsburgh: Carnegie Mellon University Pittsburgh PA Robotics Institute, CMU-RI-TR-92-01, 1992.
- [43] 徐豪迪. 基于线控底盘的无人驾驶路径规划与跟踪控制研究[D].浙江科技学院,2022.
- [44] 胡丹丹,尹鹏飞,牛国臣等.基于滚动预瞄模型的纯跟踪算法[J].科学技术与工程,2023,23(09):3787-3793.
- [45] JI X, HE X, LV C, et al. Adaptive-neural-network-based robust lateral motion control for

- autonomous vehicle at driving limits[J]. Control Engineering Practice, 2018, 76: 41 – 53.
- [46] Taghavifar H. Neural network autoregressive with exogenous input assisted multi-constraint nonlinear predictive control of autonomous vehicles [J]. IEEE Transactions on Vehicular Technology, 2019, 68(7): 6293-6304.
- [47] Huang Z, Xu X, He H, et al. Parameterized batch reinforcement learning for longitudinal control of autonomous land vehicles [J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017, 49(4):730-741
- [48] 汪洪波,王春阳,赵林峰等.基于强化学习的智能车辆路径跟踪变参数 MPC 多目标控制[J/OL]. 中国公路学报:1-19[2024-03-19].
- [49] 李新凯,虎晓诚,马萍等.基于改进 DDPG 的无人驾驶避障跟踪控制[J].华南理工大学学报(自然科学版),2023,51(11):44-55.
- [50] KAYACAN E, RAMON H, SAEYS W. Robust trajectory tracking error model-based predictive control for unmanned ground vehicles[J]. IEEE/ASME Transactions on Mechatronics, 2016, 21(2): 806 – 814.
- [51] 葛林鹤. 自动驾驶车辆可重构模型预测运动控制器研究[D]. 吉林大学, 2022.
- [52] ZHU M, CHEN H, XIONG G. A model predictive speed tracking control approach for autonomous ground vehicles[J]. Mechanical Systems and Signal Processing, 2017, 87: 138 – 152.
- [53] BORRELLI F, FALCONE P, KEVICZKY T, et al. MPC-based approach to active steering for autonomous vehicle systems[J]. International Journal of Vehicle Autonomous Systems, 2005, 3(2/3/4): 265
- [54] SONG X, SHAO Y, QU Z. A vehicle trajectory tracking method with a time-varying model based on the model predictive control[J]. IEEE Access, 2019, 8: 16573-16583.
- [55] 冀杰, 唐志荣, 吴明阳,等.面向车道变换的路径规划及模型预测轨迹跟踪[J].中国公路学报,2018, 31(4): 172-179.
- [56] 孙银健 .基于模型预测控制的无人驾驶车辆轨迹跟踪控制算法研究[D].北京:北京理工大学, 2015.
- [57] GUO H, CAO D, ChEN H, et al. Model predictive path following control for autonomous cars considering a measurable disturbance: Implementation, testing, and verification[J]. Mechanical Systems and Signal Processing, 2019, 118: 41-60.

- [58] 罗婷. 无人驾驶车辆路径规划及跟踪控制研究[D]. 山东理工大学, 2022.
- [59] 范贤波, 彭育辉, 钟聪. 基于自适应 MPC 的自动驾驶汽车轨迹跟踪控制[J]. 福州大学学报(自然科学版), 2021, 49(04): 500-507.
- [60] 白成盼, 惠飞, 景首才. 基于微分平坦与 MPC 的智能车换道控制算法[J]. 计算机技术与发展, 2020, 30(05): 16-20.
- [61] HESSBURG T, PENG H, TOMIZUKA M, et al. An experimental study on lateral control of a vehicle[C]. 1991 American Control Conference, 1991: 3084 – 3089.
- [62] 朱兴建. 基于模糊自适应 MPC 的无人驾驶车辆轨迹跟踪控制研究[D]. 西华大学, 2022.
- [63] OZGUNER U, UNYELIOGLU K A, HATIPOGLU C. An analytical study of vehicle steering control[C]. Proceedings of International Conference on Control Applications, 1995: 125 – 130
- [64] 钟思祺. 基于模型预测控制的自动驾驶汽车跟踪控制方法研究[D]. 华南理工大学, 2021.
- [65] RAFAILA R C, LIVINT G. Nonlinear model predictive control of autonomous vehicle steering[C]. 2015 19th International Conference on System Theory, Control and Computing (ICSTCC), 2015: 466 – 471.
- [66] MAYNE D Q, RAWLINGS J B, RAO C V, et al. Constrained model predictive control: Stability and optimality[J]. Automatica, 2000, 36(6): 789 – 814.
- [67] WALZ F, HOHMANN S. Model predictive longitudinal motion control for low velocities on known road profiles[J]. Vehicle System Dynamics, 2020, 58(8): 1310 – 1328.
- [68] Yang A, Zang Y, Xu L, Li L, Tan D. A Systematic Review and Future Development of Automotive Chassis Control Technology. Applied Sciences. 2023; 13(21):11859.

攻读硕士学位期间承担的科研任务与主要成果

(一) 参与的科研项目

- [1] 轮毂电机全向轮控制滑板底盘, 企业自选研发项目, 项目编号: SZC30003.
- [2] 无人驾驶自卸充电车, 企业自选研发项目, 项目编号: SZC300010.
- [3] 中型教具车规划项目, 企业自选研发项目, 项目编号: SX30005.

(二) 发表的学术论文

- [1] A Systematic Review and Future Development of Automotive Chassis Control Technology. Applied Sciences. 2023; 13(21):11859. (除导师外一作)
- [2] 线控底盘多系统协同与控制策略应用研究[J].时代汽车,2023(12):144-146.(除导师外一作)
- [3] 基于曲率自适应 LTV-MPC 轨迹跟踪控制研究[J].安徽工程大学学报.(一作)

(三) 申请及获得的发明专利

- [1] 一种无人物流车激光雷达地面点云的去除方法及应用[P]. 浙江省,CN117196975A,2023-12-08.(实质审查)

(四) 申请及获得的软著

- [1] 2024SR0254929. 车辆驾驶自动路线规划软件 V1.0 (2023.11.20).国家版权局.

(五) 相关标准的参与建设

- [1] DB 3301/T—2024,智能网联车辆开放道路环境评估与分类规范[S].

(六) 科学技术成果

- [1] DJ101002024Y0132,通用智能化线控底盘关键技术开发与应用,浙江省科技厅.

致 谢

首先，我要向我的导师杨爱喜教授表示最诚挚的感谢。在整个研究生阶段，您不仅是我学术道路以及工作道路上的引路人，更是我人生路上的导师。您严谨的学术态度、深厚的行业造诣和敏锐的洞察力，使我深受启发。您对我的论文从选题、开题、撰写到修改完善，都给予了耐心细致的指导。您的教诲不仅让我在学术上和工作上取得了进步，更让我在人生道路上受益匪浅。

感谢车辆系的时培成教授、肖平教授对我的帮助和支持，祝老师们身体健康、阖家幸福、工作顺利！

感谢浙江省智能网联汽车创新中心的黄瑞钦、叶昭芳、冯成泽、程杰、张绍丹、杨阳、胡双全老师、李兰友老师、管远华老师、孙怡彭老师以及亚太机电的于海涛。

感谢硕士期间课题组的同门许柳柳、靳凯、李屹、张建国、张庆、毛飞、江彤以及室友刘艺、宫国庆、张志涛。

最终，我要感谢我的老爸、老妈、姐姐和若寒。在我求学的道路上，你们始终是我坚实的后盾。你们的支持和鼓励是我能够不断前行的动力。

再次感谢所有给予我帮助和支持的人。在未来的道路上，我将继续努力，深耕汽车领域，不辜负导师以及家人的期望。

尚德敏学 唯实惟新

