

# 智能车辆轨迹跟踪鲁棒模型预测控制策略研究

2024 年 6 月

智能车辆轨迹跟踪鲁棒模型预测控制策略研究

北京理工大学

中图分类号: U461

UDC 分类号: 629

# 智能车辆轨迹跟踪鲁棒模型预测控制策略研究

作 者 姓 名 李岩  
学 院 名 称 机械与车辆学院  
指 导 教 师 任宏斌副教授  
答 辩 委 员 会 主 席 马彪教授  
申 请 学 位 硕士专业学位  
学 科/类 别 车辆工程  
学 位 授 予 单 位 北京理工大学  
论 文 答 辩 日期 2024 年 6 月

# **Research on Robust Model Predictive Control Strategy for Intelligent Vehicle Trajectory Tracking**

Candidate Name: Li Yan  
School or Department: School of Mechanical Engineering  
Faculty Mentor: Assoc. Prof. Ren Hongbin  
Chair, Thesis Committee: Prof. Ma Biao  
Degree Applied: Professional Degree of Master  
Major: Vehicle Engineering  
Degree by: Beijing Institute of Technology  
The Date of Defence: June, 2024

## 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独立完成的研究成果。文中所撰写内容符合以下学术规范（请勾选）：

论文综述遵循“适当引用”的规范，全部引用的内容不超过 50%。

论文中的研究数据及结果不存在篡改、剽窃、抄袭、伪造等学术不端行为，并愿意承担因学术不端行为所带来的一切后果和法律责任。

文中依法引用他人的成果，均已做出明确标注或得到许可。

论文内容未包含法律意义上已属于他人的任何形式的研究成果，也不包含本人已用于其他学位申请的论文或成果。

与本人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。

特此声明。

签 名： 李岩      日期： 2024.6.8

## 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： 李岩    日期： 2024.6.8

导师签名： 任宏斌    日期： 2024.6.8

## 摘要

智能车辆的轨迹跟踪控制是实现自动驾驶功能的关键技术，如何应对轨迹跟踪过程中的外部干扰因素和参数不确定性因素，提升控制策略的鲁棒性和跟踪精度是其中的研究重点。本文以智能车辆为研究对象，以存在外部干扰和参数不确定性工况下的车辆轨迹跟踪控制为研究内容，提出了基于管道模型预测控制（Tube Model Predictive Control, Tube MPC）的轨迹跟踪分层鲁棒控制框架，该框架包括基于 Tube MPC 的上层轨迹跟踪鲁棒控制方法和基于前馈反馈结合的下层加速度跟踪控制方法，实现了智能车辆准确、鲁棒的轨迹跟踪控制。

首先，针对轨迹跟踪分层鲁棒控制架构的特点，构建了以车辆点质量模型为基础的 Frenet 坐标系下的跟踪误差模型。引入横、纵向校正加速度作为控制输入量并在模型中引入横、纵向动力学附加扰动信号，完成了状态空间方程以及约束条件线性化的推导。随后设计了基于归一化最小均方算法和递归分位数估计算法相结合的扰动边界在线辨识方法，用于在轨迹跟踪过程中对横、纵向动力学附加扰动信号进行在线辨识，确定扰动信号范围边界。

其次，针对轨迹跟踪过程中的外部干扰和参数不确定性问题，本文提出了基于 Tube MPC 算法的上层轨迹跟踪鲁棒控制策略，采用了基于在线可达集的方法对 Tube 鲁棒不变集进行设计计算，通过选择椭圆不变集描述方式提升在线可达集的计算效率，针对 Tube MPC 算法的闭环反馈增益进行设计和求解，构建优化问题并对约束条件进行收紧，设计了终端代价函数和终端约束集，完成了对 Tube MPC 轨迹跟踪鲁棒控制器的设计。

然后，针对期望加速度信号的跟踪控制问题，本文提出了基于前馈和反馈结合的下层加速度跟踪控制策略，对上层鲁棒控制算法输出的期望加速度信号进行跟踪，提出了基于误差模型在线估计的前馈修正优化策略，缩小前馈项与期望响应的偏差，并针对控制系统中执行器延迟问题，提出了基于模型参考自适应控制算法的优化策略。

最后，本文在 Simulink 和 Carsim 上进行了轨迹跟踪鲁棒控制策略的联合仿真实验研究。仿真实验结果表明，所提出的基于 Tube MPC 算法轨迹跟踪控制器与常规 MPC 算法相比，在存在外部干扰或参数不确定性时能够输出更为平滑且易于跟踪的期望加速度信号，提升了轨迹跟踪精度；所提出的基于前馈和反馈的下层加速度跟踪控制策略能够有效跟踪上层输出的期望加速度，基于误差模型在线估计的前馈修正优

化策略能有效提升系统的响应速度和跟踪精度，基于模型参考自适应控制算法的执行器延迟优化能有效地消除信号延迟所导致的超调振荡现象，提升加速度跟踪精度。

**关键词：**智能车辆；轨迹跟踪；不确定性辨识；鲁棒模型预测控制

## Abstract

The trajectory tracking control of intelligent vehicles is a key technology for realizing autonomous driving functions. Dealing with external disturbance factors and parameter uncertainties during the trajectory tracking process, and improving the robustness of the control strategy and tracking accuracy are the focus of research. In this paper, intelligent vehicles with Ackermann steering are taken as the research object, and the research content focuses on vehicle trajectory tracking control under conditions of external disturbances and parameter uncertainties. A trajectory tracking hierarchical robust control framework based on Tube Model Predictive Control (Tube MPC) is proposed. This framework includes an upper-level robust control method for trajectory tracking based on Tube MPC and a lower-level acceleration tracking control method based on combined feedforward and feedback. This framework achieves accurate and robust trajectory tracking control for intelligent vehicles.

Firstly, based on the characteristics of the trajectory tracking hierarchical robust control architecture, a tracking error model in Frenet coordinates with a vehicle point mass model as the basis was constructed. Lateral and longitudinal correction accelerations were introduced as control inputs, and lateral and longitudinal dynamic additional disturbance signals were incorporated into the model to linearize the state-space equations and constraint conditions. Subsequently, a disturbance boundary online identification method combining normalized least squares algorithm and recursive quantile estimation algorithm was designed to identify online the lateral and longitudinal dynamic additional disturbance signals during the trajectory tracking process, determining the boundary range of disturbance signals.

Secondly, to address the external disturbances and parameter uncertainties during the trajectory tracking process, an upper-level trajectory tracking robust control strategy based on Tube MPC algorithm was proposed. The idea of separating nominal system from actual system was utilized to construct a state error system, designing a constraint tightening method based on robust invariant sets, calculating the Tube robust invariant set based on online reachable sets methodology. The design efficiency of online reachable sets was

improved by describing the invariant set using elliptical forms, designing and solving the closed-loop feedback gains for Tube MPC algorithm, constructing optimization problems, tightening the constraint conditions, designing terminal cost function and terminal constraint set, completing the design of Tube MPC trajectory tracking robust controller.

Then, addressing the problem of tracking the desired acceleration signal, a lower-level acceleration tracking control strategy based on combined feedforward and feedback was proposed. Tracking the desired acceleration signal output by the upper-level robust control algorithm, an optimized strategy of feedforward correction based on online estimation of error model was proposed to reduce deviations between the feedforward term and expected response. To tackle actuator delay issues in the control system, an optimization strategy based on model reference adaptive control algorithm was proposed.

Finally, joint simulation experiments of trajectory tracking robust control strategy were conducted on Simulink and Carsim. Simulation results show that the proposed trajectory tracking controller based on Tube MPC algorithm can output smoother and easier-to-track desired acceleration signals in the presence of external disturbances or parameter uncertainties compared to conventional MPC algorithm, thus improving tracking accuracy. The proposed lower-level acceleration tracking control strategy based on combined feedforward and feedback can effectively track the desired acceleration output from the upper level. The feedforward correction optimization strategy based on online estimation of error model can effectively enhance system response speed and tracking accuracy. The actuator delay optimization based on model reference adaptive control algorithm can effectively eliminate overshoot oscillations caused by signal delay, thus improving acceleration tracking accuracy.

**Key Words:** intelligent vehicles; trajectory tracking; uncertainty identification; robust model predictive control

## 目录

第 1 章 绪论 .....	1
1.1 研究背景和意义 .....	1
1.2 国内外研究现状和发展趋势 .....	2
1.2.1 智能车辆轨迹跟踪控制研究现状 .....	2
1.2.2 鲁棒模型预测控制研究现状 .....	4
1.3 本文主要研究内容及技术路线 .....	5
1.4 论文整体框架 .....	6
第 2 章 车辆模型及扰动边界在线辨识 .....	8
2.1 轨迹跟踪控制系统结构分析及建模分析 .....	8
2.1.1 轨迹跟踪控制系统结构分析 .....	8
2.1.2 轨迹跟踪鲁棒控制车辆模型分析 .....	9
2.2 考虑扰动的轨迹跟踪误差模型 .....	10
2.2.1 Frenet 坐标系与 Cartesian 坐标系转换 .....	10
2.2.2 轨迹跟踪误差模型建立 .....	13
2.2.3 加速度约束线性化推导 .....	15
2.3 基于 NLMS 和 RQE 的扰动边界在线辨识 .....	18
2.3.1 扰动边界辨识问题描述 .....	18
2.3.2 归一化最小均方算法 .....	19
2.3.3 递归分位数估计算法 .....	22
2.3.4 扰动边界在线辨识方法 .....	23
2.3.5 在线辨识方法仿真对比分析 .....	24
2.3.6 基于 RUMI 的扰动边界在线辨识仿真分析 .....	26
2.4 本章小结 .....	28
第 3 章 基于 TUBE MPC 的上层轨迹跟踪鲁棒控制策略 .....	30
3.1 Tube MPC 算法分析 .....	30
3.1.1 状态误差系统构建 .....	30
3.1.2 基于鲁棒不变集的约束条件收紧 .....	32
3.2 Tube 不变集优化设计 .....	34
3.2.1 Tube 不变集计算方法分析 .....	34

---

3.2.2 Tube 不变集描述方式分析 .....	37
3.3 Tube MPC 轨迹跟踪鲁棒控制器设计 .....	38
3.3.1 反馈增益设计及求解 .....	39
3.3.2 标称 MPC 优化问题构建 .....	41
3.3.3 约束条件构造及收紧 .....	42
3.3.4 终端代价函数及终端约束集设计 .....	46
3.4 本章小结 .....	47
<b>第 4 章 下层加速度跟踪控制及优化策略 .....</b>	<b>49</b>
4.1 基于前馈和反馈结合的加速度跟踪控制策略 .....	49
4.1.1 横向加速度跟踪控制策略设计 .....	50
4.1.2 纵向加速度跟踪控制策略设计 .....	52
4.2 基于误差模型在线估计的前馈修正策略 .....	54
4.2.1 基于归一化最小均方算法的纵向前馈修正 .....	54
4.2.2 基于递归最小二乘算法的横向前馈修正 .....	55
4.3 基于模型参考自适应控制的执行器延迟优化策略 .....	57
4.3.1 模型参考自适应控制原理 .....	57
4.3.2 模型参考自适应控制器设计 .....	59
4.4 本章小结 .....	62
<b>第 5 章 智能车辆轨迹跟踪控制仿真实验 .....</b>	<b>63</b>
5.1 智能车辆轨迹跟踪控制仿真平台简介 .....	63
5.1.1 基于 Simulink 和 Carsim 的联合仿真平台简介 .....	63
5.1.2 仿真实验技术参数及行驶环境设置 .....	64
5.2 Simulink 控制器仿真模型搭建 .....	65
5.2.1 参考轨迹信息预处理模块 .....	65
5.2.2 上层 Tube MPC 轨迹跟踪鲁棒控制器 .....	65
5.2.3 下层前馈反馈结合加速度跟踪控制器 .....	67
5.3 环形赛道工况仿真实验 .....	68
5.3.1 仿真实验条件设置 .....	68
5.3.2 附加外部干扰仿真实验 .....	69
5.3.3 参数不确定性的仿真实验 .....	78

5.4 蛇形工况仿真实验设置 .....	83
5.4.1 仿真实验条件设置 .....	83
5.4.2 前馈修正策略仿真实验 .....	84
5.4.3 执行器延迟优化策略仿真实验 .....	88
5.5 本章小结 .....	91
结论 .....	93
参考文献 .....	96
攻读学位期间发表论文与研究成果清单 .....	102
致谢 .....	103

## 第1章 绪论

### 1.1 研究背景和意义

随着社会进步和经济的不断发展，汽车保有量逐年提升，汽车技术也越发先进，人们对于出行的安全性、舒适性等方面要求逐步提高。融合了智能化、网联化和电动化等先进技术的智能车辆成为汽车领域被广泛研究关注的焦点，智能车辆技术的发展对于交通拥堵，环境污染，能耗过大，交通事故等问题的解决有重要的意义<sup>[1]</sup>。

根据相关人员的统计，交通事故造成了全世界每年近八成的死亡事故<sup>[2]</sup>，而其中接近九成的交通事故是由于人为操作不当所造成的，比如驾驶员疲劳、超速行驶以及转向或制动不及时等等，不仅会造成相当严重的经济财产损失，而且会严重威胁人类生命安全<sup>[3]</sup>。而智能驾驶技术能够感知周围环境信息，预测行人与其他车辆的行动轨迹，并做出最优的行为决策和控制动作，这能够避免由驾驶员操作不当引起的安全隐患<sup>[4]</sup>。在《中国制造 2025》中，国务院在汽车重点发展领域做出指示：将掌握汽车低碳化、信息化以及智能化等核心技术作为首要目标<sup>[5]</sup>。针对智能网联车辆，《中国制造 2025》提出我国将在 2025 年建立完善的自主研发生产体系，掌握智能驾驶总体和各部分关键技术，实现产业转型升级。由此可见，无论是实际需求还是国家政策导向，都表明智能驾驶技术对于当今汽车产业的重要性<sup>[6]</sup>。

智能驾驶系统通常由定位导航、环境感知、行为决策、轨迹规划、轨迹跟踪等模块构成<sup>[7]</sup>。其中，定位导航模块的作用主要是获取智能车辆当前的位置信息并为智能车辆提供全局路径导航。当给定目的地理位置信息后，定位导航模块将依据电子地图及实时路网交通流信息，规划出最优全局路径。环境感知模块的主要作用是采用多种传感器，如激光雷达、毫米波雷达、摄像头等，对车辆当前的周围环境信息进行数据采集和处理，获取有效的行驶环境信息<sup>[8-11]</sup>。行为决策和轨迹规划相当于智能驾驶系统的“大脑”，通过接收感知层传来的环境信息并结合自车信息，在满足交通法规和车辆动力学约束的前提下，为智能车辆决定下一步行动并规划出合理的参考轨迹<sup>[12-16]</sup>。轨迹跟踪模块的主要作用是计算出合理的控制指令控制智能车辆准确跟踪参考轨迹，目前的主流方法是将其解耦为纵向车速控制和横向路径跟踪控制<sup>[17]</sup>，根据自车状态信息和参考轨迹信息，采用合理的控制方法计算出期望的纵向力矩和前轮转角，以操控车辆跟踪参考轨迹。

在智能车辆轨迹跟踪控制过程中可能会遇到随机的外部干扰或是车辆内部的参

数不确定性, 如何在控制过程中考虑这些外部干扰和不确定性的影响, 提升控制算法的鲁棒性, 精确地跟踪参考轨迹, 是保证智能车辆安全行驶的关键。本文针对智能驾驶中的轨迹跟踪控制技术以及考虑不确定性的鲁棒模型预测控制算法, 开展了大量文献调研, 并对研究现状进行总结, 相关调研情况如下。

## 1.2 国内外研究现状和发展趋势

### 1.2.1 智能车辆轨迹跟踪控制研究现状

智能车辆的轨迹跟踪控制模块通常被解耦为纵向车速跟踪控制和横向路径跟踪控制。其中, 纵向车速跟踪的控制量为制动压力和油门开度, 或是纵向力矩, 目前车速控制技术较为成熟且控制系统结构较为简单, 常用的控制算法有 PID、滑模控制等。横向路径跟踪控制的控制量通常为前轮转角或方向盘转角, 相比与纵向控制更为复杂, 也是目前轨迹跟踪控制研究的重点和难点, 常用的控制算法有基于几何模型的控制方法、线性二次型调节器、模型预测控制等等。

基于几何模型的控制方法主要包括纯跟踪 (Pure Pursuit, PP) 和 Stanley 算法。PP 算法通过提前设定预瞄距离, 计算车辆的后轴中心点与参考轨迹上的预瞄点之间的横向偏差, 并以比例控制的方式转化为前轮转角控制量, 控制车辆以圆弧路径的形式通过参考轨迹的预瞄点。该算法简单有效, 对参考轨迹曲率要求不高, 适用于低速场景, 但预瞄距离的设定对跟踪效果影响显著。刘一帆<sup>[18]</sup>等人适用 PP 算法进行 AGV 路径跟踪控制, 并利用模糊控制器自适应确定预瞄距离, 提升了路径跟踪精度。付景枝<sup>[19]</sup>等人根据实时车速以及 GPS 轨迹与自车位置之间最短距离, 设计了基于二次多项式预估前视距离的方法, 提升了路径跟踪精度。Stanley 算法<sup>[20]</sup>是由斯坦福大学在 DARPA 挑战赛中提出的方法, 通过横向位移偏差和航向偏差共同决定前轮转角, 但该算法同样只适用于低速场景, 且不适用于非连续曲率路径, 对参考轨迹要求较高。王鑫<sup>[21]</sup>等人利用 PP 算法对 Stanley 算法的前轮转角计算方式进行改进, 实时计算当前车速下的前轮转角, 提升了 Stanley 算法路径跟踪的精确性和平滑性。

线性二次型调节器 (Linear Quadratic Regulator, LQR) 是一种已经应用较为成熟的控制算法, 是基于最优化原理建立的反馈控制方法, 一般通过求解黎卡提方程获得反馈控制增益。Levinson<sup>[22]</sup>等人在 2007 年的 DARPA 挑战赛中将 LQR 算法应用于斯坦福大学参赛车辆 Junior 上, 实现了对油门、制动以及转向的综合控制, 提升舒适度的同时最大限度减小跟踪误差, 完成了数百英里的轨迹跟踪控制。Zhang<sup>[23]</sup>等人将 LQR

算法与滑模控制算法结合，通过横向位移偏差和航向偏差计算期望前轮转角，结合车道线检测实现了对期望轨迹的精确跟踪。Park<sup>[24]</sup>等人将 LQR 算法应用于车辆漂移技术并开发 1:10 汽车实验平台进行了方法可行性的验证，提升车辆在湿滑路面等极端条件下的安全性。Meng<sup>[25]</sup>等人为提高铰接自卸车的路径跟踪性能，设计了一种基于 LQR 和遗传算法（GA）相结合的新型控制器，设计的控制器用于控制前车架中点的线速度和角速度，仿真实验结果以及实车实验结果表明，在低速下，LQR-GA 控制器比现有方法具有更好的跟踪性能。

模型预测控制（Model Predictive Control, MPC）是目前轨迹跟踪领域应用最广泛的算法之一。其基本思想是利用预先建立好的数学预测模型、系统当前状态量和未来控制输入序列去预测控制系统未来一段时间内的状态轨迹，然后通过求解带约束的最优控制问题获得一系列控制输入，并将第一个控制输入作用于系统，使控制系统的预测状态轨迹不断逼近目标状态轨迹，缩小跟踪误差。且相比于前述的轨迹跟踪方法能灵活有效地处理约束条件，能够对车辆的动力学约束考虑的更加全面和完善，也因此受到了广泛的研究和关注。Falcone<sup>[26-29]</sup>等人首次将 MPC 算法应用于轨迹跟踪控制，设计了自主转向系统，实现对期望路径的稳定跟踪。Chu<sup>[30]</sup>等人针对简化的车辆模型所导致的稳态跟踪误差，设计了一种基于 PID 反馈控制与 MPC 算法相结合的轨迹跟踪控制器，有效提升了 MPC 算法的跟踪精度和车辆转向平稳性。Jeong<sup>[31]</sup>等人针对四轮独立转向车辆提出了一种基于线性时变车辆模型的 MPC 车辆运动控制器，经仿真实验验证该算法能有效将转向角和驱动力矩分配给每个独立的执行机构，提升了跟踪性能。Yoon<sup>[32]</sup>等人针对 UGVs 障碍物避撞的最优轨迹跟踪问题提出了一种结合轮胎模型的非线性模型预测控制器，通过仿真实验验证了方法的有效性。Li<sup>[33]</sup>等人针对四轮毂电机独立驱动的电动汽车在高速行驶时的轨迹跟踪和操纵稳定性问题，提出了一种综合控制策略，上层由 MPC 算法计算前轮转角和附加横摆力矩，下层采用转矩最优控制和滑膜控制结合的方法进行转矩分配，通过实验验证该方法能有效降低跟踪误差。

关于轨迹跟踪控制的研究还有许多，上述针对轨迹跟踪控制算法的阐述仅代表一部分主流思想，除此之外，还有诸如 PID 的经典控制方法，以及诸如强化学习的新兴人工智能控制算法，都能被应用于轨迹跟踪控制。综合上述研究现状调查，MPC 算法综合控制效果更好，且适应性更强，能够处理不同情况下的约束条件。但常规的 MPC 算法缺乏对不确定性和外部干扰的考虑，而智能车辆轨迹跟踪控制中通常存在

着许多不确定性因素会对控制器性能造成影响<sup>[34-35]</sup>，所以在存在较大程度的不确定性和外部干扰影响时，MPC 算法的跟踪效果会变差，甚至难以维持车辆系统的稳定，因此一些改进的鲁棒模型预测控制方法被提出。

### 1.2.2 鲁棒改进的模型预测控制研究现状

为解决轨迹跟踪控制过程中可能遭遇的外部干扰和不确定性问题，一些改进的用于处理不确定性系统的模型预测控制算法被提出，其主要分为两类：随机模型预测控制和鲁棒模型预测控制。

随机模型预测控制（Stochastic MPC, SMPC）算法是一种结合随机优化理论的改进 MPC 算法，它将不确定性视为具有期望和协方差的噪声并利用噪声的统计特性来实现控制目标。王秋<sup>[36]</sup>等人基于 SMPC 算法设计了转向控制器，对车辆建模简化过程中可能产生影响的未建模动态进行考虑，从而提高了主动转向控制的控制性能。方培元<sup>[37]</sup>等人提出了一种基于 SMPC 的运动规划方法，采用质点运动模型和高斯分布对周围动态车辆的预测轨迹进行不确定性表征，并使用 SMPC 中的机会约束进行描述，以此建立空间位置约束，通过仿真验证了方法的有效性。Carvalho<sup>[38]</sup>等人针对动态城市环境中自车外其他物体的不确定性问题，设计了一种结合环境模型和 SMPC 算法的集成控制框架，在 MPC 预测中嵌入机会约束实现避撞和控制功能，同时引入风险因子对算法保守性进行调控，通过仿真实验验证了该方法在不确定城市环境下的有效性。但在智能驾驶系统中，对噪声的完整概率描述通常难以获得，因此 SMPC 算法在实际应用中仍存在困难。而在 RMPC 中，不确定性被描述为有界的扰动集，这使得其在智能驾驶系统中应用更为合理和容易。

鲁棒模型预测控制（Robust MPC, RMPC）算法主要有两种，分别是 Min-max MPC 和 Tube MPC。Min-max MPC 考虑由不确定性所能引起的最坏的情况，保证系统在最坏情况下仍能稳定运行，即在任意一种不确定性的影响下系统仍能稳定运行<sup>[39]</sup>。Peng<sup>[40]</sup>等人针对由时变、不确定车辆动力学特性和外部扰动引起的模型失配问题，以四轮独立驱动电动汽车为研究对象，提出了一种基于 Min-max MPC 的协调路径跟踪和横摆力矩控制方法，通过仿真实验证实该方法能有效增强系统鲁棒性，提升路径跟踪精度。虽然 Min-max MPC 可以使约束能被所有不确定性所满足，但此方法极其保守，可能会导致控制问题没有可行解，因此应用较少。

近年来，Tube MPC 算法在车辆控制领域被越来越广泛地使用。Tube MPC 将理想

标称系统和实际系统相分离。通过设计状态反馈增益矩阵，构建状态误差系统。并基于不变集理论将实际系统状态轨迹约束在以标称系统状态轨迹为中心的容许区域内，确保在存在不确定性的情况下，实际系统的状态轨迹尽可能贴近标称系统的理想状态轨迹。Mata<sup>[41]</sup>等人基于线性时不变的二自由度自行车模型和 Tube MPC 解决了在横向路径跟踪过程中车速不确定性问题，并考虑了严格的控制信号和横向路径跟踪误差的约束，通过仿真实验验证了该方法能在较大车速范围内保证路径跟踪精度和车辆的行驶稳定性。Rathai<sup>[42]</sup>等人以车道保持系统为研究对象，为保证车辆在存在扰动情况下的稳定性，提出了一种基于线性时变模型的鲁棒 Tube MPC 控制器，使用 LMI 和 SDP 预先计算鲁棒反馈增益，仿真实验结果表明与所提出控制器与 CLQR 控制器相比具有更好的车道保持控制效果和行驶稳定性。Wischnewski<sup>[43]</sup>等人将 Tube MPC 算法应用于自动驾驶赛车操纵极限下的控制，用于抑制控制过程中非线性效应和外部干扰的影响，在硬件在环平台上的实验结果表明，其相比于 MPC 算法和 LQR 算法能在较少违法加速度约束的情况下保持相当的圈速，因此被认为具有实际应用意义。Alcalá<sup>[44]</sup>等人针对高速和外部干扰场景，提出了一种在线 T-MPC 算法，基于线性时变车辆模型设计了由标称控制器和校正局部控制器组成的解决方案，使用  $H_\infty$  控制设计得到局部控制器的反馈增益，并采用 zonotope 替代 polytope 计算不变集，提高了计算效率，通过仿真实验与 LQR 算法对比，该算法能有效抑制外部干扰，提升跟踪精度。相比于 SMPC 和 Min-max MPC，Tube MPC 不需要扰动的完整概率描述，只需要其上下边界，且具有较小的保守性。

### 1.3 本文主要研究内容及技术路线

本文以阿克曼转向的智能车辆为研究对象，本文的主要研究内容是考虑外部干扰和参数不确定性情况下的智能车辆轨迹跟踪控制问题，提出了一种基于鲁棒模型预测控制的轨迹跟踪控制架构。研究内容具体包括车辆动力学模型和扰动边界在线辨识方法；基于 Tube MPC 算法上层轨迹跟踪鲁棒控制策略；基于前馈和反馈结合的下层加速度跟踪控制策略；基于 MATLAB/Simulink 和 Carsim 的轨迹跟踪鲁棒控制联合仿真实验验证。

如图 1.1 所示为全文技术路线架构图，图中说明了框架中各部分研究内容的输入输出关系。首先针对本文控制架构进行车辆模型的构建，在车辆点质量动力学模型基础上推导了 Frenet 坐标系下的跟踪误差模型，针对外部干扰及参数不确定性问题，通

过引入横、纵向动力学附加扰动构建了控制系统模型，针对横、纵向动力学扰动信号设计了 RUMI 扰动边界在线辨识方法，实时辨识得到的横、纵向动力学扰动边界将输出给基于 Tube MPC 的上层轨迹跟踪鲁棒控制器，用于构建 Tube 不变集并收紧约束条件，随后 Tube MPC 控制将依据车辆状态及参考轨迹信息求解输出期望的横、纵向加速度信号给到下层基于前馈和反馈结合的加速度跟踪控制器，下层加速度跟踪控制对期望加速度信号进行跟踪输出期望的车辆控制动作，即前轮转角和纵向力，通过前馈修正以及延迟优化策略对控制动作进行调整后输出给被控车辆完成一次控制循环。

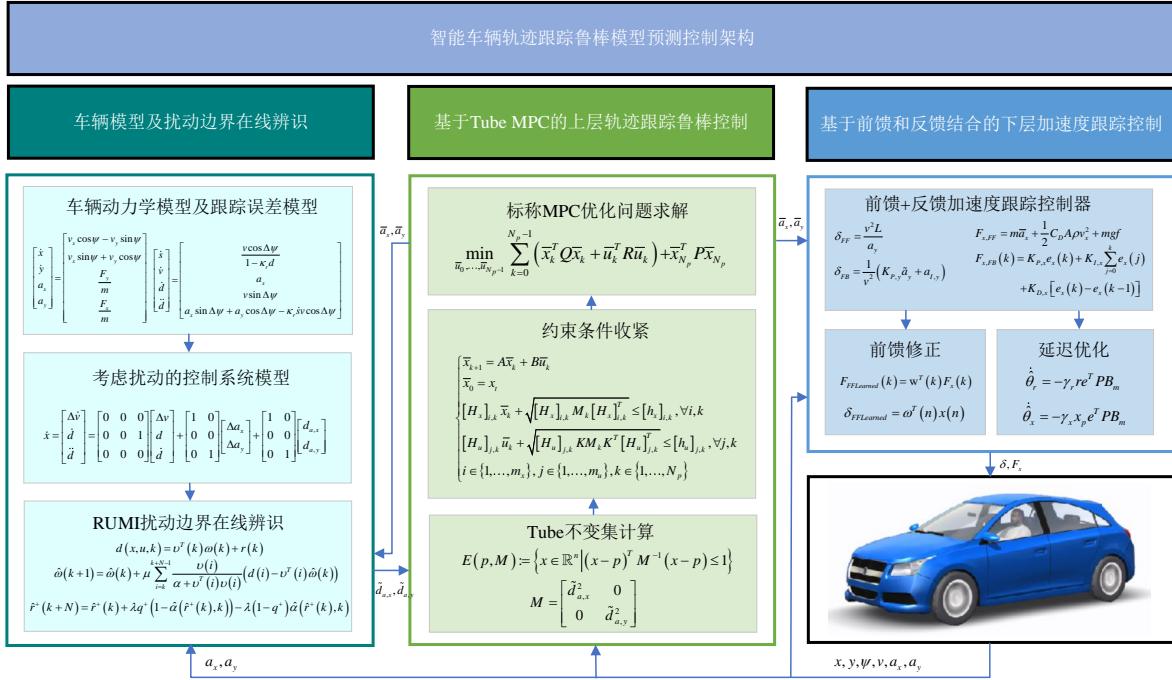


图 1.1 全文技术路线架构图

## 1.4 论文整体框架

论文整体框架如图 1.2 所示，论文的章节安排如下：

在第一章中，概述了本文选题的目的和意义，对智能车辆轨迹跟踪控制技术和鲁棒模型预测控制技术的研究现状和发展趋势进行阐述，对本文的主要研究内容及全文技术路线架构进行介绍，最后对论文整体框架和章节安排进行介绍。

在第二章中，建立上层分层控制架构，针对其控制结构特点，建立车辆点质量动力学模型，并推导了 Frenet 坐标系下跟踪误差模型，引入校正加速度和扰动信号构建控制系统模型的状态空间方程，针对扰动边界，设计了基于 NLMS 和 QE 结合的 RUMI 扰动边界在线辨识方法，并对该方法辨识效果进行仿真验证。

在第三章中，设计了基于 Tube MPC 算法的上层轨迹跟踪鲁棒控制策略，构建了状态误差系统，提出了基于鲁棒不变集的约束收紧方法，设计了 Tube 不变集的计算方法和描述方式，针对轨迹跟踪鲁棒控制问题分别进行了反馈增益求解和计算、优化问题构建、约束条件收紧以及终端代价函数和终端约束集的设计。

在第四章中，针对上层控制器输出的期望加速度信号，设计了基于前馈和反馈结合的下层加速度跟踪控制策略，针对固定输入输出关系所造成的前馈偏差问题，设计了基于误差模型在线估计的前馈修正策略，针对执行器延迟问题，设计了基于模型参考自适应控制算法的延迟优化策略。

在第五章中，利用 Simulink 和 Carsim 建立联合仿真平台，设置外部干扰及参数不确定性工况，对所提出 Tube MPC 轨迹跟踪鲁棒控制策略与常规 MPC 算法进行跟踪精度和鲁棒性的对比实验，对下层加速度跟踪控制及其优化策略有效性进行验证。

在第六章中，对全文内容进行总结，归纳本文的主要贡献和创新点，对未来工作内容进行展望和设想。

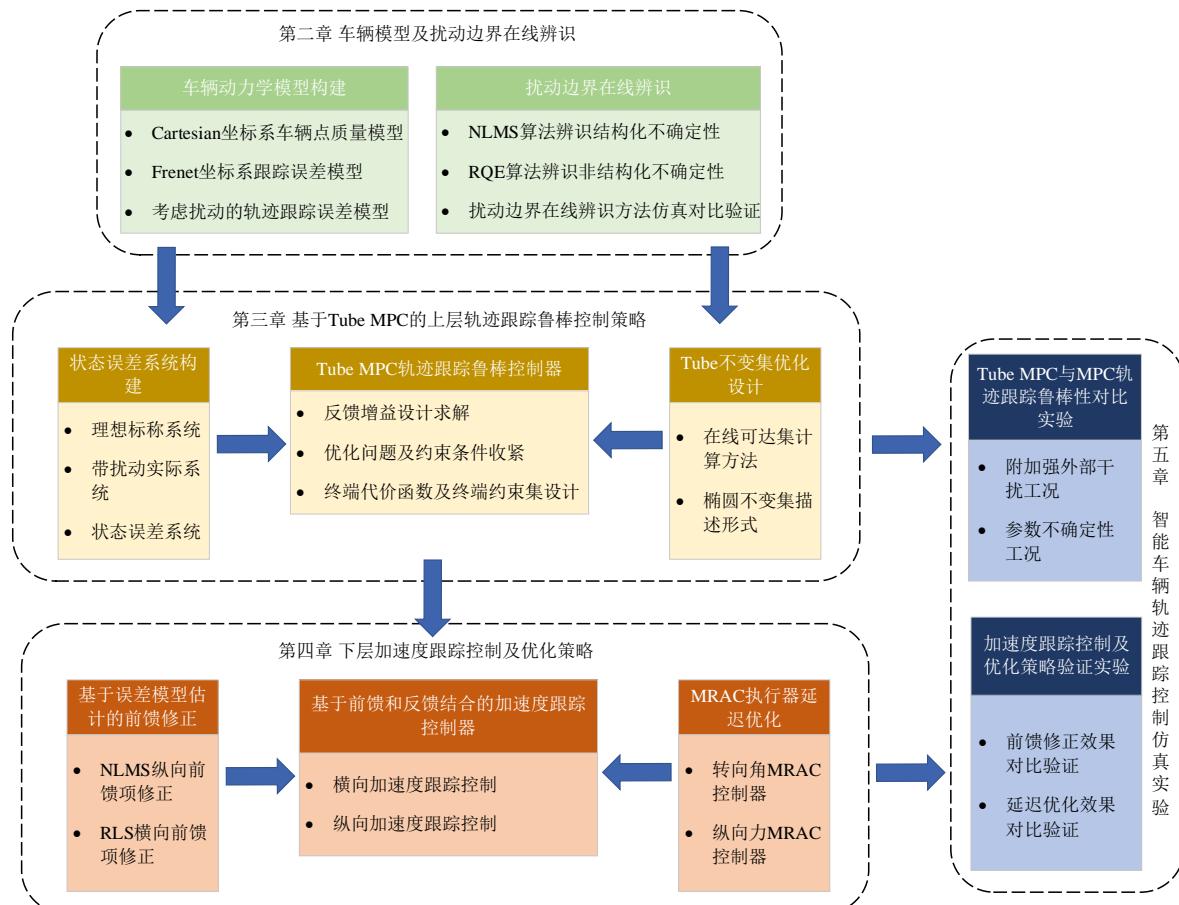


图 1.2 论文整体框架图

## 第2章 车辆模型及扰动边界在线辨识

建立能够准确描述轨迹跟踪控制系统动态响应特性的数学模型是设计轨迹跟踪控制器的基础。跟踪控制器的实时性和控制准确性对保证智能车辆沿参考轨迹安全行驶有着重要影响，在建模时应尽可能减小计算复杂度，同时在明确控制需求的情况下，能精确描述系统特征。针对不确定性系统的控制模型，还需在建模时将不确定性信号纳入考虑。综合上述要求，本文设计了以车辆点质量模型为基础的跟踪误差模型，同时引入校正加速度作为控制输入，添加横、纵向动力学附加扰动，构建了控制系统的状态空间方程，并针对其中扰动信号边界的辨识问题，设计了一种基于归一化最小均方算法（Normalized Least Mean Square, NLMS）与递归分位数估计算法（Recursive Quantile Estimation, RQE）相结合的扰动边界在线辨识方法。

### 2.1 轨迹跟踪控制系统结构分析及建模分析

#### 2.1.1 轨迹跟踪控制系统结构分析

应用于轨迹跟踪控制中的 MPC 算法通常以二自由度单轨车辆模型作为其预测模型，通过对非线性的预测模型进行线性和离散化，可将最优控制问题转化为易于求解的二次规划问题，最终直接对车辆的转向角和纵向车速进行控制。这种控制策略通常忽略了跟踪过程中的不确定性和外部干扰，当存在较大的扰动信号时，控制系统的跟踪精度会下降，甚至失稳。为解决该问题，本文将不确定性和外部干扰视为车辆横、纵向动力学上的附加扰动，提出了基于 Tube MPC 的轨迹跟踪鲁棒控制方法用于抑制扰动对控制效果的影响。同时，Tube MPC 相比于 MPC 进一步提升了计算量，针对此问题，本文提出了如图 2.1 所示的上下分层控制结构，将计算量合理分配，上层为考虑参数不确定性和外部干扰的 Tube MPC 轨迹跟踪鲁棒控制器，结合参考轨迹、车辆状态以及扰动范围信息，计算出期望横、纵向加速度信号。下层则为加速度跟踪控制器，对期望加速度进行跟踪，输出对车辆的控制指令，即前轮转角和纵向力。该控制结构设计具有一定优势，因为它可以通过简单高效的下层控制算法来减少剩余的执行机构相关的参数不确定性，而不会增加 Tube MPC 中预测模型的复杂性。

针对本文控制系统结构，为后文建模和控制器构建做出如下假设：本文的被控对象为采用阿克曼转向结构的前轮转向乘用车辆；参考轨迹信息将由智能驾驶系统上游的轨迹规划模块得到，包括纵向的轨迹信息，即车速、纵向加速度，以及横向的轨迹

信息，即曲率、行驶距离、航向角、位置坐标等。车辆状态信息和定位信息将由智能驾驶系统的状态估计模块得到，利用合理传感器、估计和滤波算法获取。

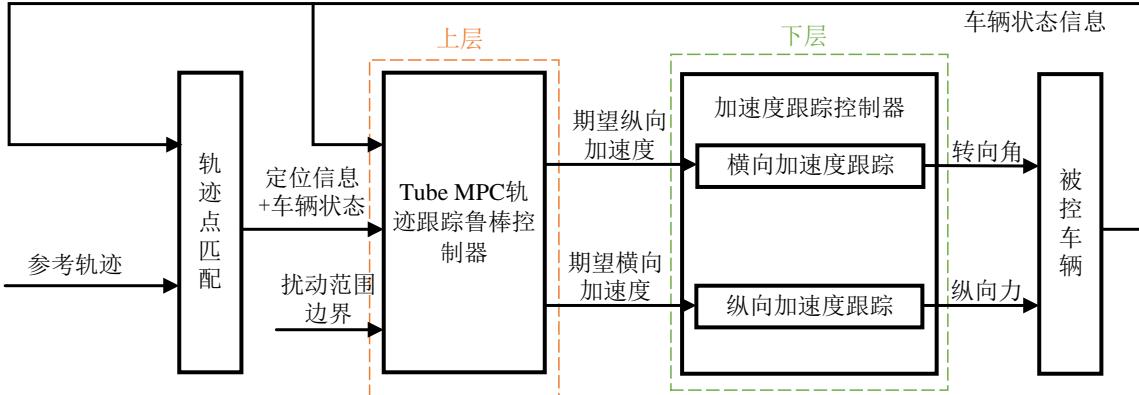


图 2.1 轨迹跟踪控制系统结构图

### 2.1.2 轨迹跟踪鲁棒控制车辆模型分析

本文的车辆模型主要服务于上层 Tube MPC 轨迹跟踪鲁棒控制器，选择合适的车辆模型对于提高控制算法的控制性能和计算效率有很大的帮助。目前针对车辆的建模方法主要有几何模型、运动学模型以及动力学模型三种<sup>[45]</sup>。

车辆几何模型通常用于描述自车位置、尺寸信息与参考轨迹之间的几何关系，该模型最常见于纯跟踪算法和 Stanley 算法，两种算法均是基于车辆和参考轨迹之间的几何偏差信息，计算出前轮转角。该模型的主要优势为简单高效，但缺点也较为明显，因为没有考虑车辆自身的速度、加速度、横摆角速度信息，导致其仅适用于低速工况。车辆运动学模型是用于描述车辆运动状态和运动规律的数学模型，主要考虑了车辆的位置、速度、加速度及横摆角速度信息，而不考虑主要与轮胎力相关的横、纵向动力学特性影响，默认车轮为刚性物体，其速度方向即为轮毂纵轴方向。应用于跟踪控制时，通常使用在低速场景，因为在高速场景中，由于轮胎侧偏角的存在，轮胎力对车辆运动状态的影响将无法再被忽略。车辆动力学模型则是通过牛顿力学关系建立的。需要考虑车辆与地面之间力的影响，考虑了轮胎特性即轮胎的侧偏，在高速和低速工况下均适用，但在高速情况下通常需要进行前轮转角的小角度假设，相比于运动学模型在更为精准，也相对更复杂。因此成为轨迹跟踪控制领域应用最为广泛的车辆模型。

在本文提出的控制系统结构中，上层轨迹跟踪鲁棒控制器需要处理横、纵向动力学上的附加扰动，即扰动主要对横、纵向加速度造成影响，因此本文选择对动力学模型进行建模。车辆动力学建模时需要对复杂度和精确度进行权衡<sup>[46]</sup>，基于文献[47-49]

中已有的良好结果，本文选择以加速度约束的点质量模型作为上层轨迹跟踪鲁棒控制器的车辆模型，假设横向和纵向加速度能够被下层加速度控制器完全跟踪，点质量模型就能准确地表示车辆动力学关系<sup>[43]</sup>。该模型与其他基于线性或非线性轮胎模型的车辆模型相比，能充分降低模型复杂度，因为后者需要更多的参数辨识工作，并导致复杂的数值优化问题。轮胎模型方面的参数不确定性通过下层控制器去抑制。

## 2.2 考虑扰动的轨迹跟踪误差模型

### 2.2.1 Frenet 坐标系与 Cartesian 坐标系转换

车辆动力学模型通常所描述的是车辆受力与运动之间的关系，并不能直接应用于轨迹跟踪控制。为实现智能车辆的轨迹跟踪控制，需要将参考轨迹信息与车辆动力学模型相结合，构造轨迹跟踪误差模型，通过最小化车辆自身位置与参考轨迹之间的偏差信息计算相应的控制指令，驱使车辆跟踪参考轨迹<sup>[50]</sup>。

车辆动力学模型通常在 Cartesian 坐标系下建立，也称为笛卡尔坐标系。虽然在 Cartesian 坐标系下能较为直观的反映参考轨迹与车辆的位置信息，但对于计算车辆与参考轨迹间的相对位置偏差则较为复杂。而 Frenet 坐标系则是一种基于道路中心线的移动坐标系，Frenet 坐标系重点表述了车辆与参考轨迹的相对位置信息，可使车辆跟踪偏差有更简单的表述方式。如图 2.2 所示为 Frenet 坐标系与 Cartesian 坐标系之间的关系，Frenet 坐标系使用变量  $s$  和  $d$  描述车辆与参考轨迹的相对位置信息，其中， $s$  表示车辆沿参考轨迹的行驶距离，即 Frenet 坐标系下的纵坐标， $d$  表示车辆质心与参考轨迹的横向偏差，即 Frenet 坐标系下的横坐标。综上所述，本文选择在 Frenet 坐标系下构建轨迹跟踪误差模型，下文将对两种坐标系的转换关系进行推导。

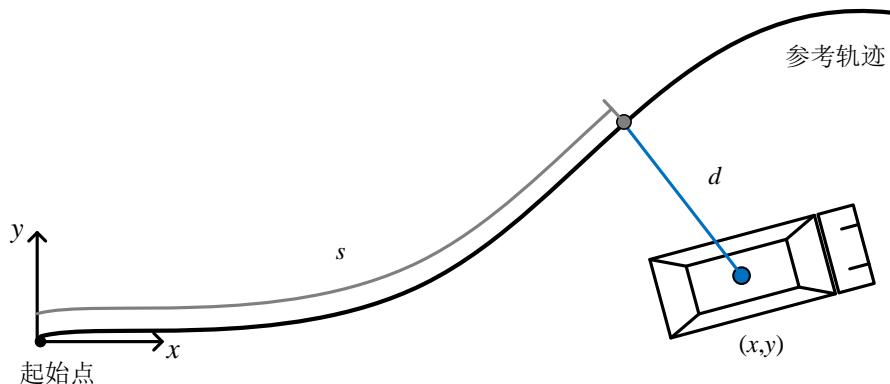


图 2.2 Frenet 坐标系与 Cartesian 坐标系关系示意图

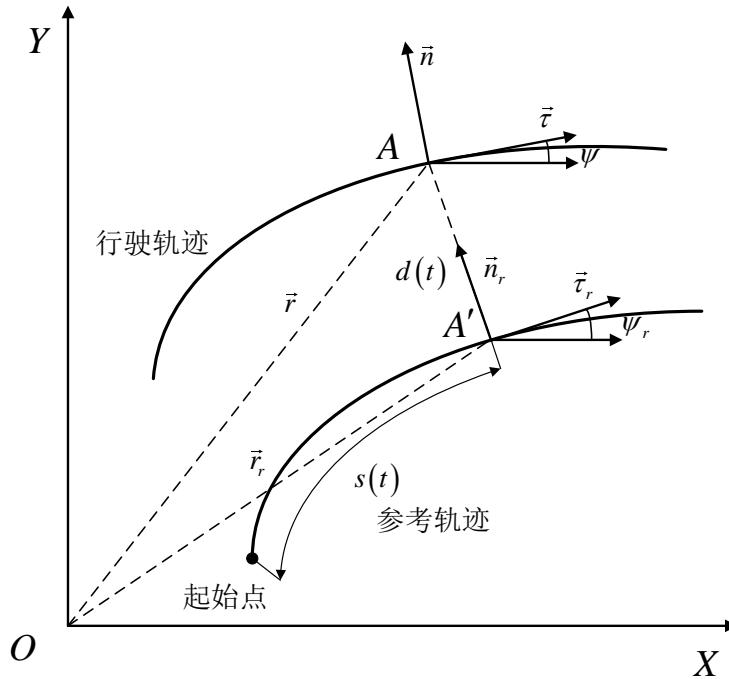


图 2.3 Frenet 坐标系与 Cartesian 坐标系转换关系

如图 2.3 所示, 车辆质心位于 A 点处, 单位切向向量为  $\vec{\tau}$ , 易知  $\vec{\tau} = [\cos(\psi), \sin(\psi)]$ , 单位法向量为  $\vec{n}$ , 易知  $\vec{n} = [-\sin(\psi), \cos(\psi)]$ , A 点在 Cartesian 坐标系下的坐标为  $(x, y)$ , 记为  $\vec{r}$ , 车速记为  $\vec{v}$ , 加速度为  $\vec{a}$ , 航向角为  $\psi$ , 曲率为  $\kappa$ , 在 Cartesian 坐标系下, 车辆运动状态描述为  $[\vec{r}, \vec{v}, \vec{a}, \psi, \kappa]$ , 而在 Frenet 坐标系下, 车辆运动状态需要被描述为  $[s, \dot{s}, \ddot{s}, d, \dot{d}, \ddot{d}, d', d'']$ , 其中,  $\dot{s} = ds/dt$  表示 Frenet 纵坐标对时间的导数, 也即车辆沿参考轨迹的纵向速度;  $\ddot{s} = d\dot{s}/dt$  表示沿参考轨迹的纵向加速度;  $\dot{d} = dd/dt$  表示 Frenet 横坐标对时间的导数, 也即沿参考轨迹的横向速度;  $\ddot{d} = d\dot{d}/dt$  表示沿参考轨迹的横向加速度;  $d' = dd/ds$  表示 Frenet 横坐标对纵坐标的一阶导;  $d'' = dd'/ds$  表示 Frenet 横坐标对纵坐标的一阶导。

匹配点的寻找方法: 遍历当前位置到参考轨迹所有采样点的距离, 寻找到距离最短的采样点即为匹配点。

$$[d_{\min}, \text{index}] = \min \{d_1, d_2, d_3, \dots\} \quad (2.1)$$

图中 A 点在参考轨迹上的匹配点为 A', 投影点在参考轨迹上的方位角为  $\psi_r$ , 曲率为  $\kappa_r$ , A' 的单位切向量为  $\vec{\tau}_r$ , 易知  $\vec{\tau}_r = [\cos(\psi_r), \sin(\psi_r)]$ , 单位法向量为  $\vec{n}_r$ , 易知  $\vec{n}_r = [-\sin(\psi_r), \cos(\psi_r)]$ , Cartesian 坐标系下该点的坐标为  $(x_r, y_r)$ , 记为  $\vec{r}_r$ 。

由平面几何可知, 三角形 AOA' 满足,

$$\vec{r} = \vec{r}_r + d \cdot \vec{n}_r \quad (2.2)$$

将式(2.2)改写为  $d \cdot \vec{n}_r = \vec{r} - \vec{r}_r$ ，两边点乘  $\vec{n}_r$  可得，

$$d = (\vec{r} - \vec{r}_r) \cdot \vec{n}_r \quad (2.3)$$

对式(2.3)两边对时间求导可得，

$$\dot{d} = (\vec{r} - \vec{r}_r) \cdot \vec{n}_r + (\vec{r} - \vec{r}_r) \cdot \vec{\dot{n}}_r \quad (2.4)$$

在 Frenet 坐标系中，易得如下辅助公式，

$$\begin{cases} \vec{r} = \vec{v} = |\vec{v}| \vec{\tau} \\ \vec{r}_r = \dot{s} \vec{\tau}_r \\ \vec{\tau}_r = \kappa_r \dot{s} \vec{n}_r \\ \vec{n}_r = -\kappa_r \dot{s} \vec{\tau}_r \end{cases} \quad (2.5)$$

将(2.5)中各式代入到式(2.4)中整理可得车辆沿参考轨迹纵向速度为，

$$\dot{s} = \frac{v \cos \Delta \psi}{1 - \kappa_r d} \quad (2.6)$$

式中， $\Delta \psi = \psi - \psi_r$ 。

同时可得车辆沿参考轨迹的横向车速为，

$$\dot{d} = v \sin \Delta \psi \quad (2.7)$$

对式(2.7)两端对时间求导可得车辆沿参考轨迹的横向加速度为，

$$\begin{aligned} \ddot{d} &= \dot{v} \sin \Delta \psi + v (\sin \Delta \psi)' \\ &= a_x \sin \Delta \psi + v \Delta \dot{\psi} \cos \Delta \psi \end{aligned} \quad (2.8)$$

式中， $\Delta \dot{\psi} = \dot{\psi} - \dot{\psi}_r = \kappa v - \kappa_r \dot{s}$ ，且有  $a_y = \kappa v^2$ ，因此上式也可表达为，

$$\ddot{d} = a_x \sin \Delta \psi + a_y \cos \Delta \psi - \kappa_r \dot{s} v \cos \Delta \psi \quad (2.9)$$

因此，综上可得下文车辆动力学建模所需 Cartesian 坐标系转 Frenet 坐标系的部分公式如下，剩余部分本文不再进行推导。

$$\dot{s} = \frac{v \cos \Delta \psi}{1 - \kappa_r d} \quad (2.10)$$

$$\dot{d} = v \sin \Delta \psi \quad (2.11)$$

$$\ddot{d} = a_x \sin \Delta \psi + a_y \cos \Delta \psi - \kappa_r \dot{s} v \cos \Delta \psi \quad (2.12)$$

## 2.2.2 轨迹跟踪误差模型建立

在本文提出的控制系统结构中, 上层轨迹跟踪鲁棒控制器需要处理横、纵向动力学上的附加扰动, 即扰动主要对横、纵向加速度造成影响。综合考虑控制结构特点和计算复杂度, 本文选择选择以加速度约束的点质量模型作为上层轨迹跟踪鲁棒控制器的车辆模型, 既满足上层控制器的控制需求, 又能降低预测模型的复杂度。车辆点质量模型忽略了车辆自身尺寸信息, 将车辆假定为一个带质量的点来描述车辆运动, 是对车辆单轨模型的进一步简化, 适合与 Tube MPC 算法结合使用<sup>[51]</sup>。

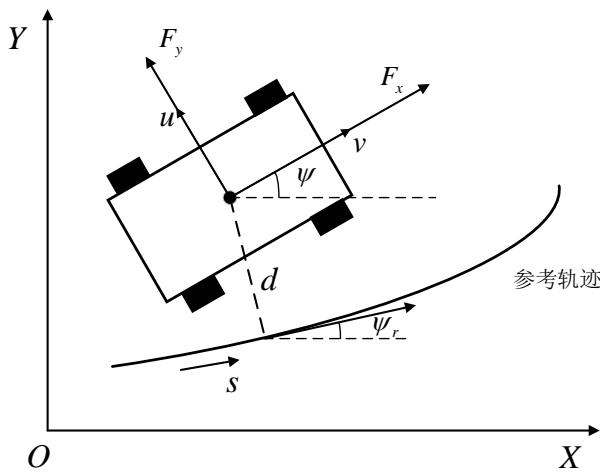


图 2.4 车辆点质量模型

在 Cartesian 坐标系下的车辆点质量模型为,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ a_x \\ a_y \end{bmatrix} = \begin{bmatrix} v \cos \psi - u \sin \psi \\ v \sin \psi + u \cos \psi \\ \frac{F_y}{m} \\ \frac{F_x}{m} \end{bmatrix} \quad (2.13)$$

式中,  $m$  表示车辆质量,  $F_x$  表示车辆前后轴上轮胎纵向力合力,  $F_y$  表示车辆前后轴上轮胎侧向力合力,  $v$  和  $a_x$  分别表示车身坐标系下车辆的纵向速度和纵向加速度,  $u$  和  $a_y$  分别表示车身坐标系下车辆的侧向速度和侧向加速度。

为实现轨迹跟踪控制的目的, 需将模型与参考轨迹信息结合, 如图 2.4 所示, 利用 2.2.1 节所推导的 Cartesian 坐标系与 Frenet 坐标系转换关系, 将参考轨迹信息与车辆点质量模型结合, 构建轨迹跟踪控制需要的跟踪误差模型, 用于描述该模型的微分方程组如下:

$$\begin{bmatrix} \dot{s} \\ \dot{v} \\ \dot{d} \\ \Delta\psi \end{bmatrix} = \begin{bmatrix} \frac{v \cos \Delta\psi}{1 - \kappa_r d} \\ \frac{F_x}{m} \\ v \sin \Delta\psi \\ \kappa v - \kappa_r \dot{s} \end{bmatrix} \quad (2.14)$$

模型的状态变量包括：车辆沿参考路径的行驶距离  $s$ ，车辆质心与参考路径间的横向跟踪偏差  $d$ ，车辆航向与参考路径航向间的航向跟踪偏差  $\Delta\psi$  以及车辆的纵向速度  $v$ 。模型的控制输入为：车辆纵向驱动力  $F_x$  和实际行驶曲率  $\kappa$ ，其中后者通常用中性转向假设来表示，即  $\kappa = \delta/l$ ，式中  $\delta$  表示前轮转角， $l$  表示车辆轴距。

车辆的动力学性能需要考虑横向加速度和纵向加速度相结合的约束条件，其通常被近似为椭圆形式约束，但高性能驾驶实验表明<sup>[52]</sup>，菱形约束更能反映真实情况下的动力学约束，因此点质量模型应满足如下加速度约束，

$$\left| \frac{a_x}{a_{x,\max}} \right| + \left| \frac{a_y}{a_{y,\max}} \right| \leq 1 \quad (2.15)$$

式中， $a_{x,\max}$  和  $a_{y,\max}$  分别表示车身坐标系下横向和纵向加速度极限。

此外为保证车辆行驶在车道界限范围内，需要满足如下约束，

$$d^- \leq d \leq d^+ \quad (2.16)$$

式中， $d^-$  和  $d^+$  分别表示参考轨迹左侧边界极限和右侧边界极限。

为便于将上层轨迹跟踪控制问题转化为易于求解的二次规划问题，需将上述动力学模型重新表述为一个线性系统，其中核心思想在于引入校正加速度作为控制输入。

首先，引入控制输入转换信息  $F_x = ma_x$  和  $\kappa v^2 = a_y$ ，以及式(2.9)，将动力学模型改写成加速度的显性表达形式，

$$\begin{bmatrix} \dot{s} \\ \dot{v} \\ \dot{d} \\ \ddot{d} \end{bmatrix} = \begin{bmatrix} \frac{v \cos \Delta\psi}{1 - \kappa_r d} \\ a_x \\ v \sin \Delta\psi \\ a_x \sin \Delta\psi + a_y \cos \Delta\psi - \kappa_r \dot{s} v \cos \Delta\psi \end{bmatrix} \quad (2.17)$$

其次，引入校正加速度  $\Delta a_x$  和  $\Delta a_y$  将上述非线性模型构造为线性系统。其中，定义横向校正加速度为  $\Delta a_y = \ddot{d}$ ，将其作为控制输入用于降低轨迹跟踪的横向偏移误差，定义纵向校正速度为  $\Delta a_x = \Delta \dot{v}$ ，式中  $\Delta v = v_r - v$ ，其中  $v_r$  表示纵向速度规划得到的期望速度，将其作为控制输入用于降低纵向速度跟踪控制的跟踪误差。从而可以将式

(2.17)构造为线性状态空间方程的形式用于构建后文的轨迹跟踪控制优化问题, 同时考虑扰动信号, 得到最终的控制系统模型如下:

$$\dot{x} = \begin{bmatrix} \Delta \dot{v} \\ \dot{d} \\ \ddot{d} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta v \\ d \\ \dot{d} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta a_x \\ \Delta a_y \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} d_{a,x} \\ d_{a,y} \end{bmatrix} \quad (2.18)$$

其中, 状态量为  $x = [\Delta v, d, \dot{d}]^T$ , 控制输入为  $u = [\Delta a_x, \Delta a_y]^T$ 。本文将轨迹跟踪过程所存在的外部干扰或参数不确定性综合考虑为横、纵向动力学附加扰动  $d_{a,x}$  和  $d_{a,y}$ , 将其引入作为不确定性控制系统的扰动信号, 其作用形式与  $\Delta a_x$  和  $\Delta a_y$  相同, 可通过参数辨识的方式确定扰动边界。上述线性系统的形式即为考虑不确定性的轨迹跟踪控制系统模型, 可用于后文 Tube MPC 优化问题的构建。

### 2.2.3 加速度约束线性化推导

为了求解, 还需要将横、纵向加速度约束式(2.15)进行线性化, 其中纵向加速度表达式仍然可由上述动力学模型得到,

$$a_x = \frac{\partial v_r}{\partial s} \frac{v \cos \Delta \psi}{1 - \kappa_r d} - \Delta a_x \quad (2.19)$$

纵向加速度表达式可由式(2.17)和式(2.19)得到, 将式(2.19)代入式(2.17)的第四项可得到,

$$\begin{aligned} \ddot{d} &= \Delta a_y = a_x \sin \Delta \psi + a_y \cos \Delta \psi - \kappa_r \dot{s} v \cos \Delta \psi \\ &= \left( \frac{\partial v_r}{\partial s} \frac{v \cos \Delta \psi}{1 - \kappa_r d} - \Delta a_x \right) \sin \Delta \psi + a_y \cos \Delta \psi - \kappa_r \frac{(v \cos \Delta \psi)^2}{1 - \kappa_r d} \end{aligned} \quad (2.20)$$

整理可得,

$$a_y = \frac{\Delta a_y}{\cos \Delta \psi} + \kappa_r \frac{\cos \Delta \psi}{1 - \kappa_r d} v^2 - \left( \frac{\partial v_r}{\partial s} \frac{v \cos \Delta \psi}{1 - \kappa_r d} - \Delta a_x \right) \tan \Delta \psi \quad (2.21)$$

由于最优化问题的短时域性质, 在进行线性化处理时将参考速度  $v_r$  和参考曲率  $\kappa_r$  在线性化参考点附近一段纵向范围  $s$  内视为定值。因此可以将式(2.19)改写为与状态量和控制输入相关的形式,

$$a_x(d, \dot{d}, \Delta v, \Delta a_x) = a_{x,r} \frac{(v_r - \Delta v) \cos \left( \arcsin \left( \frac{\dot{d}}{v_r - \Delta v} \right) \right)}{1 - d \kappa_r} - \Delta a_x \quad (2.22)$$

对各项求偏导可得,

$$\frac{\partial a_x}{\partial d} = a_{x,r} \kappa_r \frac{(v_r - \Delta v) \sqrt{1 - \frac{\dot{d}^2}{(v_r - \Delta v)^2}}}{(1 - d \kappa_r)^2} \quad (2.23)$$

$$\frac{\partial a_x}{\partial \dot{d}} = -a_{x,r} \frac{\dot{d}}{(v_r - \Delta v)(1 - d \kappa_r) \sqrt{1 - \frac{\dot{d}^2}{(v_r - \Delta v)^2}}} \quad (2.24)$$

$$\frac{\partial a_x}{\partial \Delta v} = -a_{x,r} \frac{1}{(1 - d \kappa_r) \sqrt{1 - \frac{\dot{d}^2}{(v_r - \Delta v)^2}}} \quad (2.25)$$

$$\frac{\partial a_x}{\partial \Delta a_x} = -1 \quad (2.26)$$

选择路径( $d_s = 0, \dot{d}_s = 0$ ), 零纵向校正加速度( $\Delta a_{x,s} = 0$ )以及上步迭代得到的速度曲线 $v = v_r - \Delta v = v_p$ 作为线性化参考点, 并在参考点处进行泰勒展开, 忽略高阶项, 可得到纵向加速度与状态量和控制量相关的线性形式,

$$a_x \approx a_{x,r} \kappa_r v_p d - a_{x,r} \Delta v - \Delta a_x + a_{x,r} v_p \quad (2.27)$$

同理将式(2.21)改写为与状态量和控制输入相关的形式,

$$\begin{aligned} a_y(d, \dot{d}, \Delta v, \Delta a_x, \Delta a_y) &= \frac{\Delta a_y}{\cos \left( \arcsin \left( \frac{\dot{d}}{(v_r - \Delta v)} \right) \right)} \\ &+ \kappa_r \frac{\cos \left( \arcsin \left( \frac{\dot{d}}{(v_r - \Delta v)} \right) \right)}{1 - d \kappa_r} (v_r - \Delta v)^2 \\ &- a_x(d, \dot{d}, \Delta v, \Delta a_x) \tan \left( \arcsin \left( \frac{\dot{d}}{(v_r - \Delta v)} \right) \right) \end{aligned} \quad (2.28)$$

对各项求偏导可得,

$$\frac{\partial a_y}{\partial d} = \kappa_r^2 \frac{\cos \left( \arcsin \left( \frac{\dot{d}}{(v_r - \Delta v)} \right) \right)}{(1 - d \kappa_r)^2} (v_r - \Delta v)^2 - \frac{\partial a_x}{\partial d} \tan \left( \arcsin \left( \frac{\dot{d}}{(v_r - \Delta v)} \right) \right) \quad (2.29)$$

$$\frac{\partial a_y}{\partial d} = \frac{\Delta a_y \dot{d}}{\left(v_r - \Delta v\right)^2 \left(1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}\right)^{3/2}}$$

$$-\kappa_r \frac{\dot{d}}{\left(1 - d\kappa_r\right) \sqrt{1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}}} \quad (2.30)$$

$$-\frac{\partial a_x}{\partial d} \left(d, \dot{d}, \Delta v, \Delta a_x\right) \tan\left(\arcsin\left(\frac{\dot{d}}{\left(v_r - \Delta v\right)}\right)\right)$$

$$-a_x \left(d, \dot{d}, \Delta v, \Delta a_x\right) \left(\frac{\dot{d}^2}{\left(v_r - \Delta v\right)^3 \left(1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}\right)^{3/2}} + \frac{1}{\left(v_r - \Delta v\right) \sqrt{1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}}}\right)$$

$$\frac{\partial a_y}{\partial \Delta v} = \frac{\Delta a_y \dot{d}^2}{\left(v_r - \Delta v\right)^3 \left(1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}\right)^{3/2}}$$

$$+\frac{\kappa_r}{1 - d\kappa_r} \left(\frac{\dot{d}^2}{\left(v_r - \Delta v\right) \sqrt{1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}}} - 2\left(v_r - \Delta v\right) \sqrt{1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}}\right) \quad (2.31)$$

$$-\frac{\partial a_x}{\partial \Delta v} \left(d, \dot{d}, \Delta v, \Delta a_x\right) \tan\left(\arcsin\left(\frac{\dot{d}}{\left(v_r - \Delta v\right)}\right)\right)$$

$$-a_x \left(d, \dot{d}, \Delta v, \Delta a_x\right) \left(\frac{\dot{d}}{\left(v_r - \Delta v\right)^2} \sqrt{1 - \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^2}} + \frac{\dot{d}^2}{\left(v_r - \Delta v\right)^3 \left(1 - \frac{\dot{d}^3}{\left(v_r - \Delta v\right)^4}\right)^{3/2}}\right)$$

$$\frac{\partial a_y}{\partial \Delta a_x} = \tan\left(\arcsin\left(\frac{\dot{d}}{\left(v_r - \Delta v\right)}\right)\right) \quad (2.32)$$

$$\frac{\partial a_y}{\partial \Delta a_y} = \frac{1}{\cos\left(\arcsin\left(\frac{\dot{d}}{\left(v_r - \Delta v\right)}\right)\right)} \quad (2.33)$$

选择路径 $(d_s = 0, \dot{d}_s = 0)$ ，零横、纵向校正加速度 $(\Delta a_{x,s} = 0, \Delta a_{y,s} = 0)$ 以及上步迭代得到的速度曲线 $v = v_r - \Delta v = v_p$ 作为线性化参考点，并在参考点处进行泰勒展开，忽

略高阶项, 可得到纵向加速度与状态量和控制量相关的线性形式,

$$a_y \approx \kappa_r^2 v_p^2 d - \frac{a_x(d, \dot{d}, \Delta v, \Delta a_x)}{v_p} \dot{d} - 2\kappa_r v_p \Delta v + \Delta a_y + \kappa_r v_p^2 \quad (2.34)$$

将加速度约束式(2.15)改写为如下形式,

$$\pm a_x a_{y,\max} \pm a_y a_{x,\max} \leq a_{x,\max} a_{y,\max} \quad (2.35)$$

将前文推导结果式(2.27)和式(2.34)代入式(2.35)中即可得到横、纵向加速度关于状态量和控制输入的线性方程形式,

$$\begin{aligned} & \pm (a_{x,r} \kappa_r v_p d - a_{x,r} \Delta v - \Delta a_x + a_{x,r} v_p) a_{y,\max} \\ & \pm \left( \kappa_r^2 v_p^2 d - \frac{a_x(d, \dot{d}, \Delta v, \Delta a_x)}{v_p} \dot{d} - 2\kappa_r v_p \Delta v + \Delta a_y + \kappa_r v_p^2 \right) a_{x,\max} \leq a_{x,\max} a_{y,\max} \end{aligned} \quad (2.36)$$

## 2.3 基于 NLMS 和 RQE 的扰动边界在线辨识

在前文控制系统建模中本文引入了横、纵向动力学附加扰动  $d_{a,x}$  和  $d_{a,y}$ , 将车辆模型构建为一个不确定性系统, 为后文鲁棒控制策略的设计做准备。一个鲁棒控制策略设计的关键要素是确定最坏情况下的系统不确定性。目前多数研究<sup>[41-43]</sup>仍采用预先假设好固定的扰动范围边界值的方式进行鲁棒控制策略设计, 这种假设为确保系统稳定性通常存在一定的保守性。由于在系统运行之前进行保守的不确定性假设, 鲁棒控制器通常会受到性能限制。越来越多需要在安全、关键环境中进行高性能操作的系统(如智能驾驶车辆)促使研究新的设计方法。最近, 参数在线辨识方法已经成为一种受到广泛关注的方式, 其可以基于系统运行期间收集的数据在线实时辨识扰动边界值, 从而减少鲁棒控制策略的保守性。因此本文拟采用一种归一化最小均方算法(NLMS)与递归分位数估计算法(RQE)相结合的方法, 以辨识扰动信号  $d_{a,x}$  和  $d_{a,y}$  的边界值, 确定扰动信号的精确边界可用于计算后文 Tube MPC 算法的 Tube 不变集, 降低鲁棒控制策略的保守性, 提升控制系统的控制性能。

### 2.3.1 扰动边界辨识问题描述

设计轨迹跟踪鲁棒控制策略需要确定扰动信号范围边界, 本文的扰动信号即为式(2.18)中的横、纵向动力学附加扰动  $d_{a,x}$  和  $d_{a,y}$ , 下文将对扰动边界辨识问题进行描述。2.2 节所推导的考虑扰动的控制系统模型可描述为如下离散形式,

$$x(k+1) = f(x(k), u(k)) + d(x(k), u(k), k) \quad (2.37)$$

式中,  $x(k)$  表示在  $k$  时刻的状态量,  $u(k)$  表示在  $k$  时刻的控制输入,  $d$  表示扰动信号, 通常与状态量和控制输入相关。

将扰动信号建模为如下形式,

$$d(x, u, k) = v^T(k) \omega(k) + r(k) \quad (2.38)$$

$$v^T(k) = [\phi_1(x(k), u(k)), \dots, \phi_f(x(k), u(k))] \quad (2.39)$$

如式(2.38)所示, 扰动信号被分解成结构化不确定性和非结构化不确定性,  $v^T(k) \omega(k)$  为结构化不确定性部分, 也可视为线性回归模型,  $r(k)$  为非结构化不确定性部分。式中,  $v(k) \in \mathbb{R}^f$  为基函数  $\phi(\cdot)$  的向量,  $\omega(k)$  为权重系数向量。对扰动信号采用这种分解方式可以使本文提出的估计方法适用于任意形式的有界加性扰动信号, 同时可以通过对基函数的合理选择尽可能地减小非结构化不确定性  $r(k)$  的值, 从而提高扰动边界的精度, 降低鲁棒控制的保守性<sup>[53]</sup>。

扰动边界在线辨识的目的是实时更新横、纵向动力学附加扰动  $d_{a,x}$  和  $d_{a,y}$  的边界值, 用于后文 Tube MPC 轨迹跟踪鲁棒控制器在计算 Tube 不变集时使用。综上, 需要解决的问题如下, 本文将进行如下式所示的扰动边界辨识:

$$d^-(x, u) \leq d(x, u, k) \leq d^+(x, u) \quad (2.40)$$

为确保扰动边界辨识的可靠性, 本文将采用批处理的形式进行辨识<sup>[53]</sup>, 针对单批次数量为  $N$  的数据, 确保达到数据覆盖率为  $q$ , 数据覆盖率为表示数据处于辨识出的上下界范围的数量占总数据量的比值, 通常会设定为较高的值以确保绝大多数正常采样点位于所辨识的边界范围内, 剔除少数超出边界的异常点, 即确保式(2.40)对总量为  $Nq$  的数据成立。

由此, 上述扰动边界辨识问题也可改写为如下形式,

$$\begin{cases} d(x, u, k) \leq d^+(x, u) \\ d(x, u, k) \leq d^-(x, u) \end{cases} \quad (2.41)$$

式(2.41)中的两式分别对数据量为  $Nq^+$  和  $Nq^-$  的数据成立, 其中  $q^+$  和  $q^-$  分别表示上下两式对应的数据覆盖率, 其可以被视为对上、下分位数概率解释的等效。 $q^+$  和  $q^-$  的值通常被确定为能覆盖较大范围的数值, 例如,  $q^+ = 0.99$  和  $q^- = 0.01$ , 这意味着满足式(2.40)的整体数据覆盖率为  $q = q^+ - q^- = 0.98$ 。

### 2.3.2 归一化最小均方算法

针对扰动信号模型式(2.38)中结构化不确定部分性  $v^T(k)\omega(k)$ ，需要选择合适的辨识方法实时更新权重系数  $\omega(k)$ 。最小均方算法（Least Mean Square, LMS）是一种常用的自适应滤波算法，也可用于线性模型的参数辨识，LMS 算法的核心思想是通过迭代更新权值，使得误差信号的均方差最小化。它是一种基于梯度下降的优化算法，根据当前权值对目标函数的梯度进行估计，然后调整权值以使误差最小化。其具有原理简单、收敛速度较快而且适用范围广等优点。然而，LMS 的步长也即学习率需要预先设定，在滤波过程中无法调整，因此在设计上需要对稳态误差和收敛速度做出权衡，两者不可兼得，步长大则收敛速度快，但误差较大；步长小，可提升精度减小误差，但收敛速度较慢。为解决此问题一种变步长的自适应滤波算法被提出，即归一化最小均方（Normalized Least Mean Square, NLMS）算法<sup>[54]</sup>。该算法在滤波过程中能够实时调整步长，刚开始步长大，收敛快；收敛后，减小步长，保证较高收敛精度。本文将采用 NLMS 算法用于估计扰动信号模型中的结构化不确定性部分。

对于如下线性回归模型，

$$y(\mathbf{x}) = w_0 + w_1 x_1 + \cdots + w_n x_n \quad (2.42)$$

写成矢量形式如下，

$$y = \mathbf{w}^T \mathbf{x} \quad (2.43)$$

传统 LMS 算法流程如下，

- 1) 给定  $\mathbf{w}(0)$
- 2) 计算输出值：  $y(k) = \mathbf{w}^T(k) \mathbf{x}(k)$
- 3) 计算估计误差：  $e(k) = d(k) - y(k)$
- 4) 权重系数更新：  $\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k) \mathbf{x}(k)$

其中， $d(k)$  表示期望响应， $y(k)$  称为对期望响应的估计， $\mathbf{x}(k)$  表示输入信号的向量， $\mathbf{w}(k)$  表示权重系数向量， $\mu$  表示步长，也称为学习率。

LMS 算法的核心思想是通过梯度下降法进行学习，迭代训练，计算更新权重系数，直至达到收敛条件。但由于 LMS 算法中步长是人为设定的常量，权重系数的调整量与输入向量  $\mathbf{x}(k)$  成正比，而  $\mathbf{x}(k)$  是含有噪声的信号，当  $\mathbf{x}(k)$  较大时，LMS 算法会出现梯度噪声放大的问题，为了克服这个问题，可以采用 NLMS 算法，因为其使用输入向量的平方欧式范数对权重系数调整量进行了归一化。

NLMS 算法的设计准则可以表述为带约束的优化问题：给定输入向量  $\mathbf{x}(k)$  和期望响应  $d(k)$ ，确定更新后的权重系数向量  $\mathbf{w}(k+1)$ ，以使下式的平方欧式范数最小，

$$\Delta w(k+1) = w(k+1) - w(k) \quad (2.44)$$

并受制于以下约束条件,

$$w^T(k+1)x(k) = d(k) \quad (2.45)$$

可以使用拉格朗日乘子法解决该带约束的优化问题。根据此方法, 上述问题的代价函数可以描述为,

$$J(k) = \|\Delta w(k+1)\|^2 + \lambda(d(k) - w^T(k+1)x(k)) \quad (2.46)$$

其中,  $\lambda$  为拉格朗日乘子。

为寻找使  $J(k)$  最小化的  $w(k+1)$ , 令  $J(k)$  对  $w^T(k+1)$  求导, 并令其值为 0,

$$\frac{\partial J(k)}{\partial w^T(k+1)} = 2(w(k+1) - w(k)) - \lambda x(k) = 0 \quad (2.47)$$

可得,

$$w(k+1) = w(k) + \frac{1}{2}\lambda x(k) \quad (2.48)$$

将式(2.48)代入(2.45), 以求解  $\lambda$ ,

$$\begin{aligned} d(k) &= \left( w(k) + \frac{1}{2}\lambda x(k) \right)^T x(k) \\ &= w^T(k)x(k) + \frac{1}{2}\lambda \|x(k)\|^2 \end{aligned} \quad (2.49)$$

由此可得,

$$\begin{aligned} \lambda &= \frac{2(d(k) - w^T(k)x(k))}{\|x(k)\|^2} \\ &= \frac{2e(k)}{\|x(k)\|^2} \end{aligned} \quad (2.50)$$

结合上述结果, 可以推导出  $\Delta w(k+1)$  的最优值, 即由式(2.48)和式(2.50)可得,

$$\Delta w(k+1) = \frac{1}{\|x(k)\|^2} e(k) x(k) \quad (2.51)$$

为了对权重系数向量的增量变化进行控制而不改变向量的方向, 引入一个正的实数标量因子  $\mu$  作为学习率, 与 LMS 算法中的学习率一致, 即有,

$$\Delta w(k+1) = \frac{\mu}{\|x(k)\|^2} e(k) x(k) \quad (2.52)$$

为了保证分母不为零, 令  $\|x(k)\|^2$  加上一个常数  $\alpha (0 < \alpha \leq 1)$ , 从而得到权重系数更新公式,

$$\Delta w(k+1) = \frac{\mu}{\alpha + \|x(k)\|^2} e(k) x(k) \quad (2.53)$$

综上所述, NLMS 算法流程如下:

- 1) 给定  $w(0)$
- 2) 计算输出值:  $y(k) = w(k)^T x(k)$
- 3) 计算估计误差:  $e(k) = d(k) - y(k)$
- 4) 权重系数更新:  $w(k+1) = w(k) + \frac{\mu}{\alpha + \|x(k)\|^2} e(k) x(k)$

### 2.3.3 递归分位数估计算法

2.3.2 节中的 NLMS 算法完成了对扰动信号模型式(2.38)中结构化不确定部分性  $v^T(k)\omega(k)$  的辨识, 扰动模型中剩余的非结构化不确定性  $r(k)$  的辨识则可以视为对残差  $d(x, u, k) - v^T(k)\omega(k)$  上下边界的辨识。用于估计残差上下界的一种最基本的想法就是将所有数据存储起来并进行排序从而寻找指定数据比例的分位数来确定。然而, 这种方式无法适用于连续的数据流, 由于受到存储空间和计算力的限制, 本文拟采用递归分位数估计算法 (Recursive Quantile Estimation, RQE) 对扰动信号模型中的非结构部分的上下边界进行估计<sup>[55]</sup>。下文将进行递归分位数估计算法更新公式的推导。

根据分位数的定义, 我们可以通过最小化如下函数确定分位数,

$$J = \arg \min_{\hat{\theta}_\tau \in \mathbb{R}} \left( \sum_{i: \theta_i > \hat{\theta}_\tau} \tau |\theta_i - \hat{\theta}_\tau| + \sum_{i: \theta_i \leq \hat{\theta}_\tau} (1 - \tau) |\theta_i - \hat{\theta}_\tau| \right) \quad (2.54)$$

式中,  $\theta_i$  为实际输出值,  $\hat{\theta}_\tau$  为分位数估计值, 而  $\tau$  为其对应的分位比例值。

为了便于求解, 将式(2.54)改写为加权最小二乘的形式,

$$J = \frac{1}{2n} \left( \sum_{i: \theta_i > \hat{\theta}_\tau} \tau (\theta_i - \hat{\theta}_\tau)^2 + \sum_{i: \theta_i \leq \hat{\theta}_\tau} (1 - \tau) (\theta_i - \hat{\theta}_\tau)^2 \right) \quad (2.55)$$

式中,  $n$  为数据量, 该式即可视为回归模型的损失函数, 通过最小化损失函数可求得分位数估计公式。

定义分位数的经验估计式为,

$$\hat{\alpha}(\hat{\theta}_\tau) = \frac{1}{n} \sum_{i=1}^n I(\theta_i \leq \hat{\theta}_\tau) \quad (2.56)$$

式中,  $I(\cdot)$  为示性函数, 即当括号内条件为真时, 取值为 1, 当括号内条件为假时, 取值为 0。

则可以得到分位数估计的更新公式,

$$\begin{aligned} \hat{\theta}_\tau(k+1) &= \hat{\theta}_\tau(k) + \zeta \frac{\partial J}{\partial \hat{\theta}_\tau(k)} \\ &= \hat{\theta}_\tau(k) + \zeta \frac{1}{n} \left( \tau \sum_{i:\theta_i > \hat{\theta}_\tau} (\hat{\theta}_\tau(k) - \theta_i) + (1-\tau) \sum_{i:\theta_i < \hat{\theta}_\tau} (\hat{\theta}_\tau(k) - \theta_i) \right) \\ &= \hat{\theta}_\tau(k) + \zeta \tau (1 - \hat{\alpha}(\hat{\theta}_\tau(k))) - \lambda (1-\tau) \hat{\alpha}(\hat{\theta}_\tau(k)) \end{aligned} \quad (2.57)$$

式中,  $\zeta > 0$  为分位数估计的学习率。

### 2.3.4 扰动边界在线辨识方法

如图 2.5 所示为本文所提出的归一化最小均方算法和递归分位数估计算法相结合的扰动边界在线辨识方法结构示意图, 在后文中将其统称为递归不确定性模型辨识 (Recursive Uncertainty Model Identification, RUMI)。

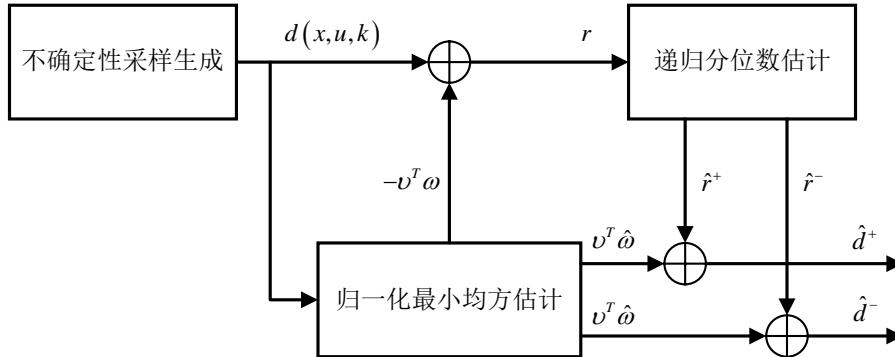


图 2.5 RUMI 递归不确定性在线辨识方法示意图

不确定性采样得到的扰动信号数据  $d(x, u, k)$ , 采用归一化最小均方估计算法可以为结构化不确定性部分的权重系数提供一个 L2 最优估计  $\hat{\omega} \in \mathbb{R}^f$  [56], 随后非结构化不确定性  $r(k) = d(x, u, k) - v^T \hat{\omega}(k)$  可由递归分位数估计算法估计得出其上界  $\hat{r}^+(k)$  和下界  $\hat{r}^-(k)$ , 结合两种估计算法的结果可以最终得出扰动信号模型上下界的估计结果。

$$\begin{cases} \hat{d}^+(v(k)) = v^T \hat{\omega}(k) + \hat{r}^+(k) \\ \hat{d}^-(v(k)) = v^T \hat{\omega}(k) + \hat{r}^-(k) \end{cases} \quad (2.58)$$

根据 2.3.2 节中的 NLMS 算法原理并结合本文的扰动信号模型, 针对扰动模型中的结构化不确定性部分  $v^T(k)\omega(k)$ , 将  $N$  步采样扰动信号数据作为一批次, 可得到其权重系数估计结果批处理形式的更新公式为:

$$\hat{\omega}(k+1) = \hat{\omega}(k) + \mu \sum_{i=k}^{k+N-1} \frac{v(i)}{\alpha + v^T(i)v(i)} (d(i) - v^T(i)\hat{\omega}(k)) \quad (2.59)$$

式中,  $\mu > 0$  为学习率,  $\alpha > 0$  为较小的常数量。

根据 2.3.3 节中 RQE 算法原理, 设定批处理形式的分位数的经验估计为如下形式:

$$\hat{\alpha}(\hat{r}^+, k) = \frac{1}{N} \sum_{i=k}^{k+N-1} I(r(i) \leq \hat{r}^+) \quad (2.60)$$

由于本文以加速度偏差的绝对值作为扰动采样信号, 因此则针对扰动模型中非结构化不确定性部分  $r(k)$  仅需要辨识上界  $\hat{r}^+(k)$ , 其批处理形式的递归估计公式如下:

$$\begin{aligned} \hat{r}^+(k+N) = & \hat{r}^+(k) + \lambda q^+ (1 - \hat{\alpha}(\hat{r}^+(k), k)) \\ & - \lambda (1 - q^+) \hat{\alpha}(\hat{r}^+(k), k) \end{aligned} \quad (2.61)$$

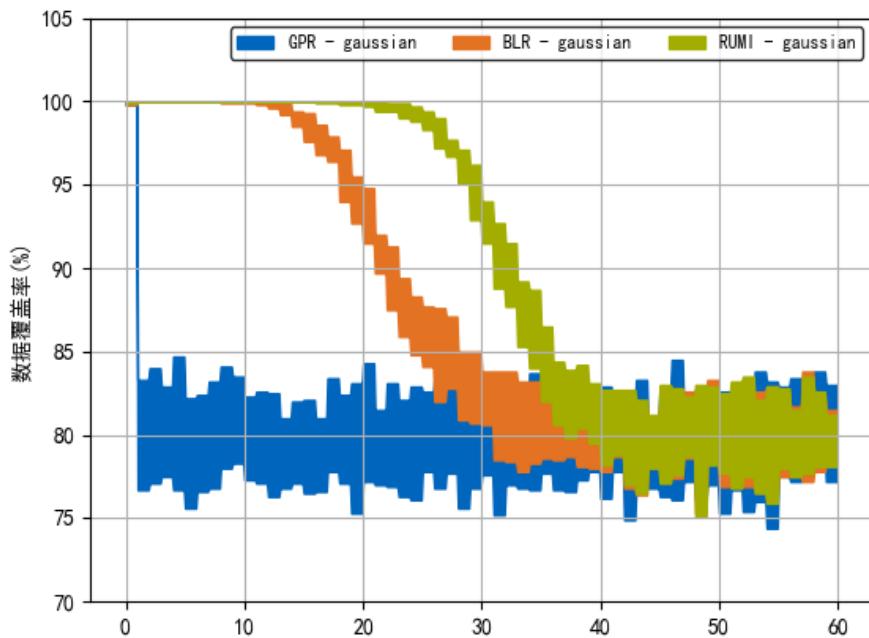
### 2.3.5 在线辨识方法仿真对比分析

为验证本文所提出 RUMI 方法的有效性, 选择与高斯过程回归 (Gaussian Process Regression, GPR) 算法和贝叶斯线性回归 (Bayesian Linear Regression, BLR) 方法进行仿真对比验证, 这两种方法目前均在基于学习的控制领域得到广泛应用<sup>[57-60]</sup>。

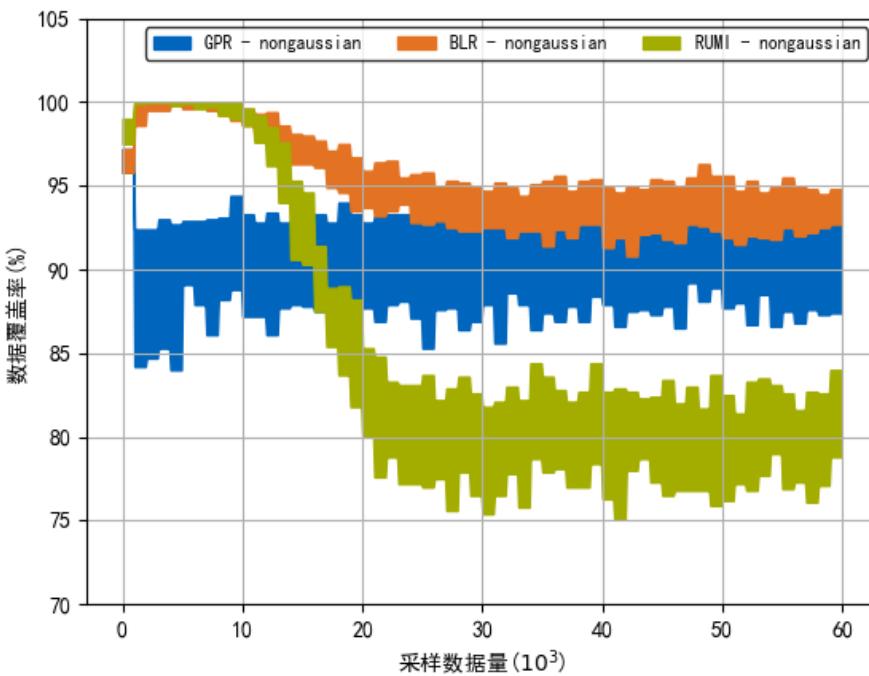
在 Python 中分别编写了 GPR, BLR 以及 RUMI 代码, 其中 BLR 的实现是基于文献[56]和[58]所提出的方法, GPR 的实现是基于文献[61]所提出的方法。在仿真中设置如下基于径向基函数的基准测试模型所生成的数据流作为扰动信号:

$$d(k) = v^T(k)\omega + \varepsilon(k) \quad (2.62)$$

式中,  $v^T(k)$  为径向基函数的向量,  $\varepsilon(k)$  为噪声信号, 分别构建了高斯噪声和非高斯噪声信号。其中, 高斯噪声由均值为 0, 标准差为 0.3 的高斯分布生成, 非高斯噪声则由两个移位的高斯分布构建组成。在仿真过程中, 共生成了 20 批次采样信号, 每批次信号共包含 60000 个采样信号, 并将目标数据覆盖率设定为 80%, 仿真结果如图 2.6 所示, 分析如下:



(a) 高斯噪声辨识效果



(b) 非高斯噪声辨识效果

图 2.6 不确定性辨识效果对比

如图 2.6 (a)所示为三种算法对高斯噪声扰动信号边界的辨识效果,由图可见,三种算法均可以将数据覆盖率达到目标数据覆盖率达到左右,即意味着所辨识得到的扰动边界内包含了 80%左右的采样数据量,三种算法均成功辨识到正确的扰动边界。而图 2.6 (b)所示的情况则有所不同,在对非高斯噪声扰动信号的辨识结果中,只有本文提出的 RUMI 算法的数据覆盖率达到目标数据覆盖率达到,即能够成功地辨识出正确的扰动边界值, GPR 和 BLR 算法均无法收敛到目标数据覆盖率达到,这在实际的扰动边界辨识时可能会出现低估或高估的情况,从而造成鲁棒控制器的性能不稳定。

综合上述仿真结果分析,相比于 GPR 算法和 BLR 回归算法,本文提出的 RUMI 算法因为不需要进行先验的概率假设,无论对高斯噪声和非高斯噪声都具备有效的回归结果,适用于更广泛的扰动信号类型。此外,如表 2-1 所示,由于算法结构简单高效, RUMI 的计算时间相比于其他两种算法能够有效缩减,有利于与 Tube MPC 算法相结合。

表 2-1 每批次数据平均计算时间对比

GPR	BLR	RUMI
1321.4 ms	16.2 ms	9.1 ms

### 2.3.6 基于 RUMI 的扰动边界在线辨识仿真分析

RUMI 算法在后文中将与 Tube MPC 轨迹跟踪鲁棒控制策略相结合,用于辨识控制系统模型(2.18)中的横、纵向动力学附加扰动  $d_{a,x}$  和  $d_{a,x}$  的边界值。在每一时刻采样实际加速度信号与期望加速度信号的偏差值作为横、纵向动力学附加扰动的采样信号。在 Simulink 中搭建 RUMI 算法模型与 Tube MPC 算法相结合优化控制策略的保守性,其关系结构如图 2.7 所示,在轨迹跟踪控制过程中, RUMI 算法对横、纵向加速度偏差信号进行实时采样,在线辨识其边界值作为横、纵向动力学附加扰动的边界值,传递给 Tube MPC 控制器用于 Tube 不变集的构建。

在轨迹跟踪仿真实验中,利用 RUMI 算法进行横、纵向动力学附加扰动边界在线辨识的效果如图 2.8、图 2.9 以及图 2.10 所示。其中,图 2.8 所示为在线表示过程中数据覆盖率的变化曲线,红色虚线表示目标数据覆盖率为 90%,即要求所辨识的扰动边界内包含 90%的采样数据。由图可见,无论是横向扰动还是纵向扰动, RUMI 算法均能够将数据覆盖率达到目标覆盖率达到附近,即辨识出了正确的扰动边界值。具体的加速度偏差采样信号点以及附加扰动边界在线辨识的结果如图

2.9 以及图 2.10 所示。

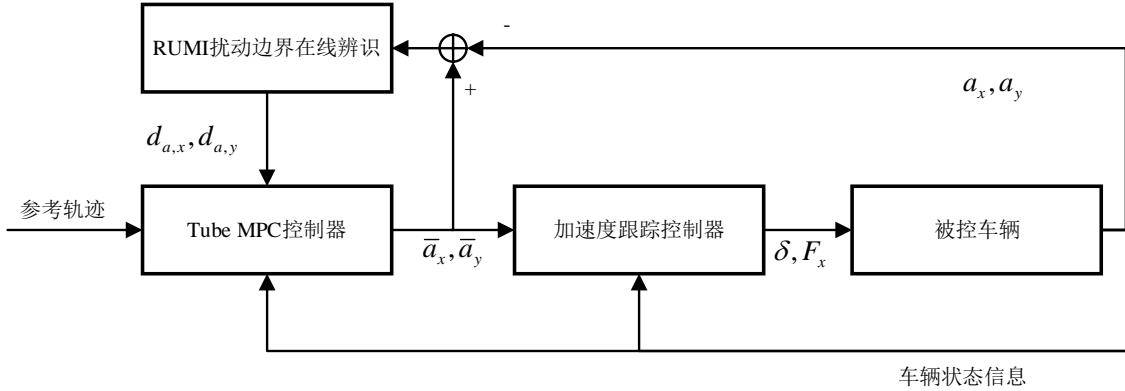


图 2.7 RUMI 扰动边界辨识与 Tube MPC 控制器关系结构图

综上所述，本文所提出的 RUMI 算法能够在轨迹跟踪鲁棒控制过程中有效地辨识出正确的扰动边界值，并在控制过程中实时更新。将 RUMI 算法与 Tube MPC 算法相结合，可以有效避免因提前预设扰动范围而造成的控制策略保守问题，精确地辨识出扰动的边界值能够优化鲁棒控制策略对扰动的预估准度，降低保守性，提升控制性能。

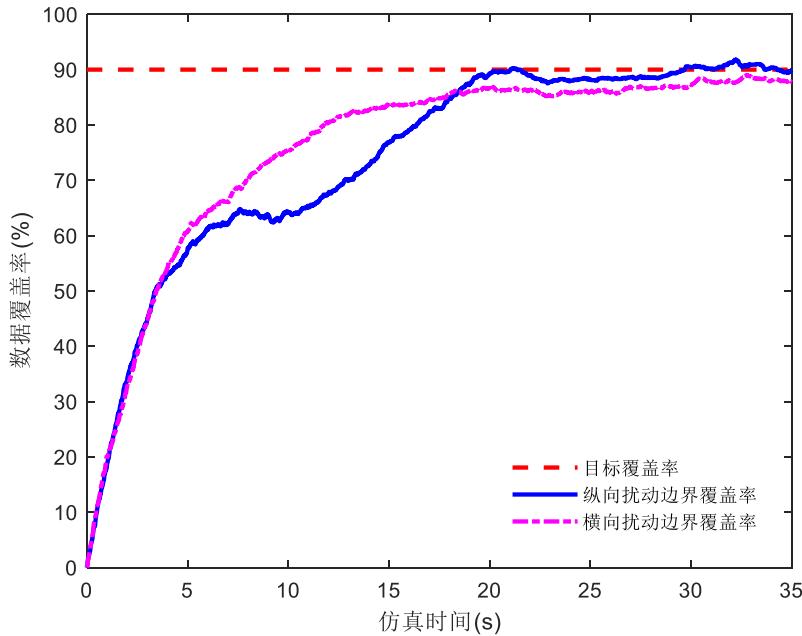


图 2.8 横、纵向扰动边界辨识数据覆盖率

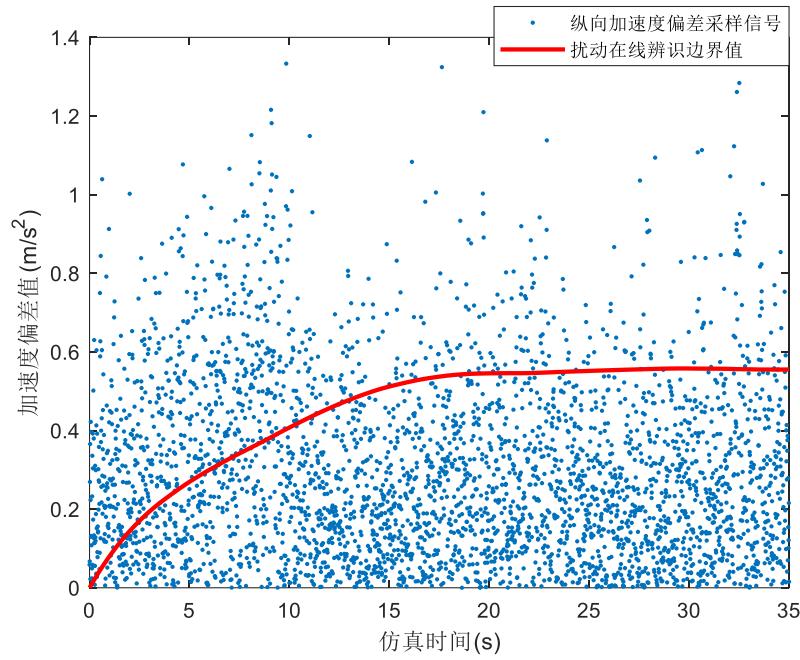


图 2.9 纵向加速度偏差采样信号及边界辨识效果

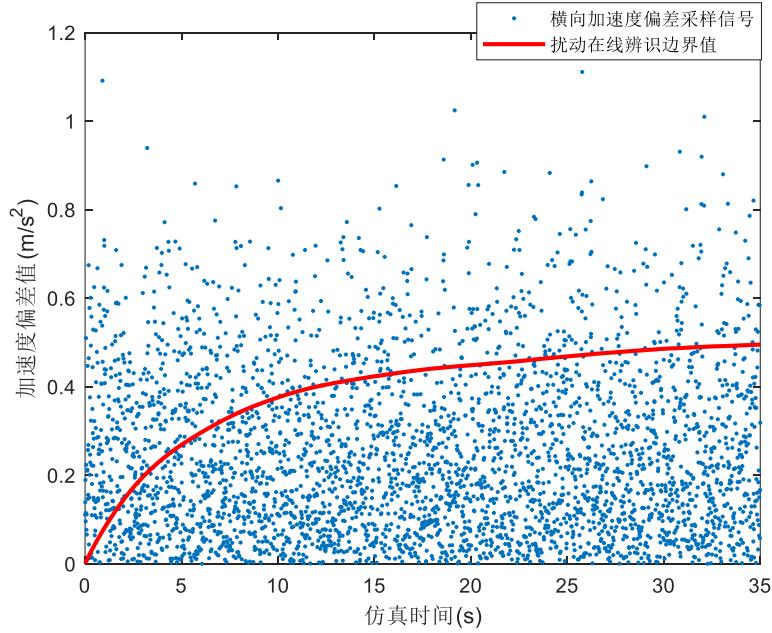


图 2.10 横向加速度偏差采样信号及边界辨识效果

## 2.4 本章小结

本章提出了基于上下分层控制的轨迹跟踪鲁棒控制策略，分析上层 Tube MPC 鲁棒控制器的控制需求及特点，选择以点质量模型为基础的跟踪误差模型作为车辆模

型；推导了 Frenet 坐标系与 Cartesian 坐标系的转换关系公式；推导了 Frenet 坐标系下的跟踪误差模型；引入校正加速度作为控制输入进行线性化，同时添加横、纵向动力学附加扰动，确定了系统的状态空间方程，推导了加速度约束的线性化形式；针对模型中的扰动估计问题，设计了基于 NLMS 和 QE 相结合的 RUMI 扰动边界在线辨识方法，在 Python 中通过仿真实验设置了高斯噪声和非高斯噪声，分别与高斯过程回归和贝叶斯线性回归两种算法进行扰动边界辨识效果对比验证，结果表明 RUMI 算法能有效辨识两种噪声扰动信号，并且拥有较短的计算时间；在 Simulink 中搭建了 RUMI 模块，并在轨迹跟踪控制仿真实验中验证了该方法对横、纵向动力学扰动边界在线辨识的有效性。

## 第3章 基于 Tube MPC 的上层轨迹跟踪鲁棒控制策略

智能车辆在轨迹跟踪过程中存在着相当多的不确定性因素，这其中可能有车辆外部的干扰因素，如路面状况、风力因素等，也可能有车辆内部的参数不确定性，如质量参数变化。当存在外部干扰或参数不确定性时，会直观地影响车辆的横、纵向动力学性能，常规的 MPC 算法将难以维持系统的稳定性，导致跟踪误差大幅增加甚至车辆失稳的现象。针对这个问题，本章将前一章车辆模型状态空间方程的基础上，设计基于 Tube MPC 算法的上层轨迹跟踪鲁棒控制策略，将外部干扰和参数不确定性视为横、纵向动力学上的附加扰动纳入算法考虑，以提高轨迹跟踪控制策略的抑制干扰能力，提高系统鲁棒性。

### 3. 1 Tube MPC 算法分析

Tube MPC 算法是将理想标称系统，即不考虑附加扰动的系统，从实际系统中分离出来，构建两种系统之间的状态误差系统，并设计反馈控制增益，把对实际系统的控制转化为对标称系统的控制的一种方法。本质上仍是对标称 MPC 优化问题进行求解得到最优控制输入，再由 Tube MPC 算法中的反馈控制律将其转化为实际系统的控制输入，由此确保实际系统的状态轨迹尽可能贴近标称系统的理想状态轨迹，并利用约束条件将其限制在以理想状态轨迹为中心的邻域内，从而实现闭环系统的渐进稳定，该邻域即为 Tube。下文将对 Tube MPC 算法原理进行介绍。

#### 3. 1. 1 状态误差系统构建

Tube MPC 的核心思想之一就是将理想的标称系统与实际系统相分离，构建状态误差系统，把对实际系统的控制转化为对标称系统的控制，从而更易于求解，因此首先需要构建实际系统与标称系统之间的状态误差系统，下文将对此进行介绍。

一个考虑扰动信号的线性离散系统，即实际系统如下，

$$x_{k+1} = Ax_k + Bu_k + w_k \quad (3.1)$$

其中包括状态量  $x \in \mathbb{R}^n$ ，控制输入  $u \in \mathbb{R}^m$  以及外部扰动信号  $w \in \mathbb{R}^o$ ，系统矩阵  $A \in \mathbb{R}^{n \times n}$ ， $B \in \mathbb{R}^{n \times m}$  都有着相对应的维度。状态量、控制输入以及外部扰动信号分别满足如下约束条件，

$$x_k \in X = \{A_x x_k \leq b_x\} \quad (3.2)$$

$$u_k \in U = \{A_u u_k \leq b_u\} \quad (3.3)$$

$$w_k \in W = \{A_w w_k \leq b_w\} \quad (3.4)$$

而将外部扰动信号置零，则可以从考虑扰动的实际系统中分离出不考虑扰动的理想标称系统如下，

$$\bar{x}_{k+1} = A\bar{x}_k + B\bar{u}_k \quad (3.5)$$

式中， $\bar{x} \in \mathbb{R}^n$  和  $\bar{u} \in \mathbb{R}^m$  分别为标称系统的状态量和控制输入。

通过求解如下目标函数，可以得到标称 MPC 系统的控制输入量，

$$\min_{\bar{u}_0, \dots, \bar{u}_{N_c-1}} \sum_{i=0}^{N_p-1} \bar{x}_i^T Q \bar{x}_i + \sum_{j=0}^{N_c-1} \bar{u}_j^T R \bar{u}_j \quad (3.6)$$

式中， $N_p$  为预测时域， $N_c$  为控制时域。求解得到的标称系统控制时域内的最优控制输入序列为  $\bar{u} = [\bar{u}_0, \bar{u}_1, \dots, \bar{u}_{N_c-1}]^T$ 。假定标称系统状态向量的初值为  $\bar{x}_0$ ，则将最优控制输入序列  $\bar{u}$  与状态量初值共同代入标称系统状态空间方程，可迭代计算得到预测时域内的状态向量序列  $\bar{x} = [\bar{x}_0, \bar{x}_1, \dots, \bar{x}_{N_p-1}]^T$ ，如图 3.1 所示，预测时域内的标称系统状态轨迹  $\bar{x}$  由最优控制输入序列  $\bar{u}$  作用得到，预测时域内的实际系统状态轨迹  $x$  由最优控制输入序列  $\bar{u}$  和外部扰动信号  $w$  共同作用得到。由于外部扰动信号的存在，导致实际系统状态预测轨迹偏离理想的标称系统状态预测轨迹，从而可能会造成轨迹跟踪效果变差，甚至车辆失稳的危险情况。

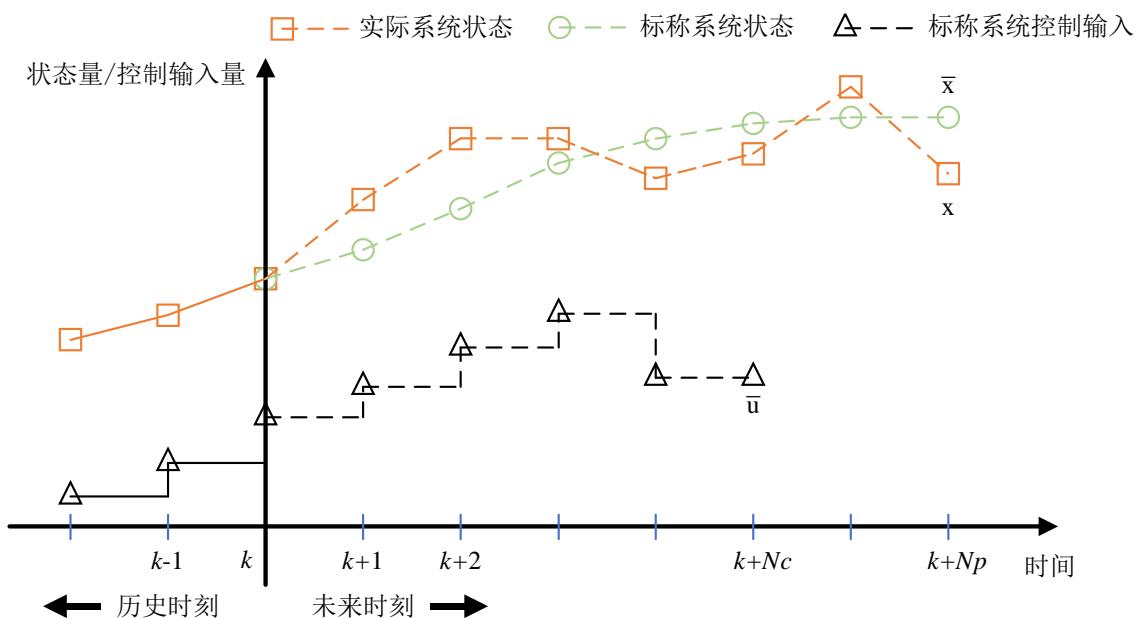


图 3.1 标称系统与实际系统状态轨迹对比图

相比于 MPC 求解一个开环的最优控制输入序列, Tube MPC 采用预稳定的控制策略, 为消除外部扰动导致的偏差, 使实际系统的状态轨迹尽可能地接近标称系统的理想状态轨迹, 有必要构建一个误差负反馈控制律以补偿控制变量。补偿控制变量的本质是在标称状态轨迹附近设定一个局部反馈控制器, 通过闭环反馈增益  $K$  去消除实际系统与标称系统之间状态轨迹的偏差<sup>[62]</sup>。则实际系统的控制律如下,

$$u_k = \bar{u}_k + K(x_k - \bar{x}_k) \quad (3.7)$$

由实际系统式(3.1)减去标称系统式(3.5), 可以构建实际系统与标称系统之间的状态误差系统如下,

$$\tilde{x}_{k+1} = A\tilde{x}_k + B\tilde{u}_k + w_k \quad (3.8)$$

$$\tilde{x}_k = x_k - \bar{x}_k \quad (3.9)$$

$$\tilde{u}_k = u_k - \bar{u}_k \quad (3.10)$$

式中,  $\tilde{x}_k \in \mathbb{R}^n$  为  $n$  维状态量误差, 表示由外部扰动引起的实际系统与标称系统之间的状态轨迹偏差量,  $\tilde{u}_k \in \mathbb{R}^m$  为  $m$  维控制量偏差, 表示为了弥补外部扰动所需采取的控制输入补偿量。

由式(3.7)和式(3.10)可得, 状态误差系统的控制律为,

$$\tilde{u}_k = K\tilde{x}_k \quad (3.11)$$

则状态误差系统的闭环状态方程可写为如下形式,

$$\tilde{x}_{k+1} = A_K\tilde{x}_k + w_k \quad (3.12)$$

$$A_K = A + BK \quad (3.13)$$

通过调整反馈增益矩阵  $K$ , 可以使得状态误差系统的系统矩阵  $A_K$  成为赫尔维茨矩阵并使得控制系统具有渐进稳定性。通常可以采用连续线性最优二次型的反馈律来构造上述局部反馈控制器系统<sup>[59]</sup>。

### 3.1.2 基于鲁棒不变集的约束条件收紧

上文利用了 Tube MPC 将标称系统从实际系统中分离的思想, 设计了预稳定的控制策略, 构建了闭环状态误差系统, 以确保实际系统的状态轨迹能尽可能贴近标称系统。本节将介绍 Tube MPC 的另一核心思想: 利用不变集理论对标称系统的约束条件进行收紧, 从而保证实际系统的约束条件在存在外部扰动或参数不确定性的情况下仍然被满足。

约束条件的收紧需要引入鲁棒不变集<sup>[63]</sup>, 其定义如下: 鲁棒不变集 (Robust

Invariant Set) 是指在系统控制理论中, 为了保证系统的稳定性和性能, 在给定的干扰、噪声等不确定因素下, 系统状态所能保持的一组状态集合。这个集合的特点是, 无论系统受到何种干扰或噪声的影响, 系统的状态都能够保持在鲁棒不变集内, 从而保证系统的稳定性和性能满足设计要求。

对于如式(3.1)所示的不确定性线性离散系统, 假设存在非空集合  $Z \in \mathbb{R}^n$ , 当且仅当任意  $k$  时刻,  $\forall x_k \in Z, Z \subseteq X$ , 对于  $\forall w_k \in W$  均满足,

$$x_{k+1} = Ax_k + Bu_k + w_k \in Z \quad (3.14)$$

则称  $Z$  为该系统的鲁棒不变集。

针对约束条件收紧的计算, 后文将涉及到两种集合运算操作, 故在此引入。

闵可夫斯基和:

$$X \oplus Y \triangleq (x + y \mid x \in X, y \in Y) \quad (3.15)$$

庞特里亚金差:

$$X \ominus Y \triangleq (x \mid x + y \in X, \forall y \in Y) \quad (3.16)$$

假设  $Z$  为状态误差系统(3.12)的鲁棒不变集, 则由鲁棒不变集的定义可知对于任意的状态误差量初始值  $\tilde{x}_k = x_k - \bar{x}_k \in Z$  和任意的外部扰动量  $w_k \in W$ , 都可以保证  $\tilde{x}_{k+i} \in Z, i = 0, \dots, N_p - 1$ 。由此可知, 实际系统的状态轨迹  $x_k$  在任意外部扰动  $w_k \in W$  的影响下, 都将被控制律(3.7)限制在以标称系统状态轨迹  $\bar{x}_k$  为中心的鲁棒不变集  $Z$  内, 即  $x_{k+i} \in \bar{x}_{k+i} \oplus Z, i = 0, \dots, N_p - 1$ 。而鲁棒不变集  $Z$  也即为 Tube 不变集<sup>[64]</sup>。如图 3.2 所示为 Tube 不变集作用原理。

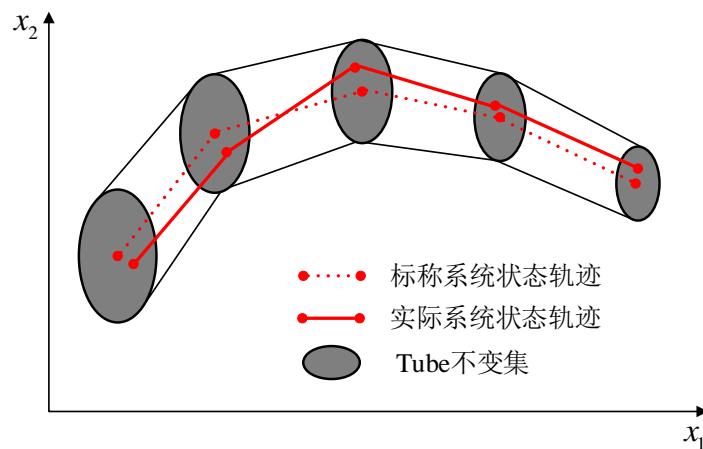


图 3.2 Tube 不变集作用原理示意图

由实际系统的控制律式(3.7)易知, 实际系统的控制输入量包含标称系统的最优控

制输入量  $\bar{u}_k$  和由状态误差所得的反馈控制输入补偿量  $K(x_k - \bar{x}_k)$  两部分。已知在控制过程中, 需保证实际系统满足相应的状态量约束  $x_{k+1} \in X$  和控制输入量约束  $u_k \in U$ , 则标称系统的最优求解过程中必须满足的收紧约束条件如下,

$$\begin{cases} \bar{x}_{k+1} \in \bar{X} = X \ominus Z \\ \bar{u}_k \in \bar{U} = U \ominus KZ \end{cases} \quad (3.17)$$

式中,  $\bar{X}$  表示标称系统的状态量容许集,  $\bar{U}$  表示标称系统的控制输入量容许集。

### 3.2 Tube 不变集优化设计

Tube MPC 算法是通过将实际系统的状态轨迹限制在以标称系统的理想状态轨迹为中心的 Tube 不变集范围内, 从而抑制外部干扰或参数不确定性对控制性能的影响。由此可知, Tube 不变集的设计优劣将对 Tube MPC 算法的控制效果。Tube 不变集的大小决定了标称系统状态量约束和控制输入量约束收紧的程度, Tube 不变集越大则约束收紧程度越紧, 意味着算法在应对扰动时将采取更为保守的控制动作, 反之, 当 Tube 不变集越小则约束收紧程度越松, 虽然控制动作不会过于保守, 但当存在超出扰动范围的扰动信号时, 可能会造成控制性能下降。下文将分别对 Tube 不变集的计算方法和描述方式进行分析和选择, 确定适合本文轨迹跟踪鲁棒控制策略的 Tube 不变集设计方法。

#### 3.2.1 Tube 不变集计算方法分析

目前 Tube 不变集的计算方法主要有两种, 分别为最小鲁棒不变集方法和在线可达集方法<sup>[65]</sup>。如图 3.3 所示, 第一个概念是采用离线计算得到的最小鲁棒不变集去描述 Tube, 可以保证 Tube 包含在任意扰动信号和容许控制输入作用下的所有可能状态轨迹, 是获得 Tube 的一种非常直观的方法。第二个概念是采用一系列在线计算得到的可达集去描述 Tube, 由于考虑了不断累积的误差模型和外部干扰所造成的不断增大的偏差, Tube 的尺寸将在控制过程中不断更新, 随着预测范围的增大而增大, 确保在控制过程的初始阶段保守性尽可能小。下面将对两种方法展开更详细的分析。

由 3.1.2 节中鲁棒不变集的定义可知, 如果给定一个区域, 在该区域内的任何一个状态, 在扰动和控制器的作用下, 都已知呆在这个区域内, 则称这个区域为扰动不变集, 可以想象, 只要这个区域给的足够大, 即把扰动考虑的足够大, 则一定可以实现这个效果, 因此要找的是一个最小扰动不变集, 这样可以把对扰动考虑的保守性降

到最低。因此，最小鲁棒不变集  $F_\infty$  是不确定性系统所有可能状态轨迹的最小边界<sup>[63]</sup>，也是一种获得 Tube 不变集最直接的方法。最小鲁棒不变集被证明存在且唯一<sup>[66]</sup>，其计算公式如下，

$$F_\infty = \bigoplus_{i=0}^{\infty} A_K^i W \quad (3.18)$$

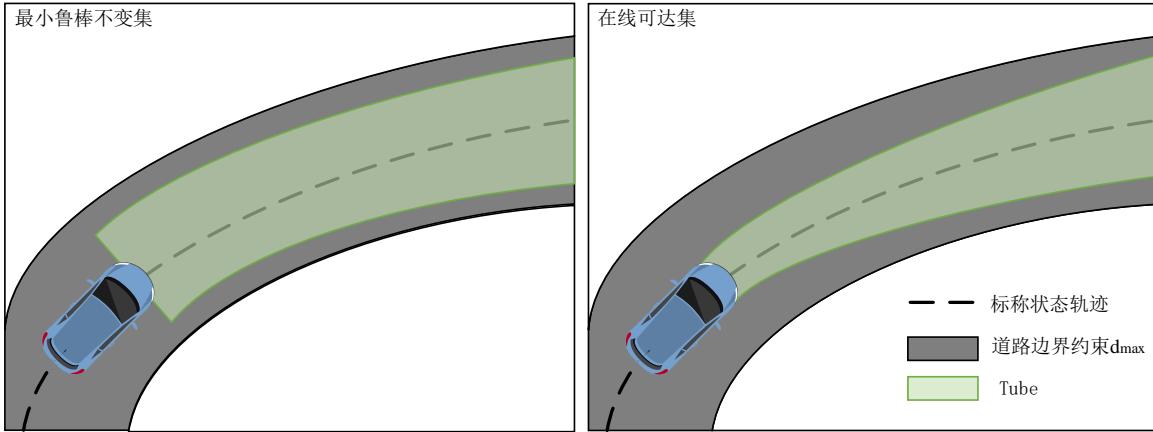


图 3.3 基于最小鲁棒不变集和基于在线可达集的 Tube MPC 方法对比

由上式易知最小鲁棒不变集无法直接计算得到，通常需要近似计算。Rakovic 等人<sup>[66]</sup>提出了一种  $\varepsilon$  外逼近的方法近似计算最小鲁棒不变集，其采用有限次迭代寻找一个数组  $(\alpha, s)$  满足如下条件，

$$\begin{cases} A^s W \subseteq \alpha W \\ \alpha(1-\alpha)^{-1} F_s \subseteq \mathbb{B}_p^n(\varepsilon) \end{cases} \quad (3.19)$$

$$F_s = \bigoplus_{i=0}^{s-1} A_K^i W \quad (3.20)$$

式中， $\mathbb{B}_p^n(\varepsilon)$  表示  $p$  范数球。

基于数组  $(\alpha, s)$  的最小鲁棒不变集的近似计算公式为，

$$F(\alpha, s) = (1-\alpha)^{-1} F_s \quad (3.21)$$

基于最小鲁棒不变集的标称系统状态容许集  $\bar{X}$  和控制输入容许集  $\bar{U}$  分别为，

$$\bar{X} = X \ominus F(\alpha, s) \quad (3.22)$$

$$\bar{U} = U \ominus K F(\alpha, s) \quad (3.23)$$

由公式(3.18)可知，虽然最小鲁棒不变集是状态轨迹的最小边界，但由于是预先离线计算得到，因此需要预先确定扰动范围，同时需要考虑无限时域内所有在扰动范围内的可能取值，因此在控制的过程中，约束收紧的程度将从始至终保持相同，这使

得在控制初期, 扰动信号还未累积, 扰动量较小时, 控制动作就比较保守, 导致控制算法的性能下降。为解决此问题, 降低控制算法的保守性, 需要结合扰动边界在线辨识, 在线实时更新 Tube 不变集的大小, 因此可以利用在线可达集方法计算 Tube 不变集。

在线可达集是指从初始状态开始, 通过一系列的动作或决策, 在有限的时间内能够到达的状态的集合。在线可达集通常用于描述在某个任务或问题中, 能够通过一系列步骤来实现目标的状态集合。通常可由在线计算  $N$  步可达集实现, 而  $N$  步可达集可以通过多次迭代一步可达集得到, 首先引入一步可达集的定义<sup>[63]</sup>:

对于考虑加性有界不确定性的线性离散系统, 任意给定状态量  $x_k \in \Omega$ , 在控制输入量  $u_k \in U$  的作用下, 下一时刻系统状态所能达到的集合  $R(\Omega)$  成为集合  $\Omega$  的一步可达集, 即

$$R(\Omega) := \left\{ x_{k+1} \in X \mid \exists x_k \in \Omega, \exists u_k \in U, \forall w_k \in W : x_{k+1} = Ax_k + Bu_k + w_k \right\} \quad (3.24)$$

在每一采样时刻  $k$ , 将一步可达集在预测时域内向前迭代, 获得误差系统的  $N$  步可达集为<sup>[67]</sup>,

$$R_{k+i+1} = (A + BK)R_{k+i} \oplus W, i = 0, \dots, N_p - 1 \quad (3.25)$$

式中,  $R_{k+i}, i = 0, \dots, N_p - 1$  为误差系统的  $i$  步可达集, 且  $R_0 = W$ 。

则基于  $N$  步可达集的标称系统状态容许集  $\bar{X}_{k+i}$  和控制输入容许集  $\bar{U}_{k+i}$  分别为,

$$\bar{X}_{k+i} = X \ominus R_{k+i}, i = 0, \dots, N_p - 1 \quad (3.26)$$

$$\bar{U}_{k+i} = U \ominus KR_{k+i}, i = 0, \dots, N_p - 1 \quad (3.27)$$

与最小鲁棒不变集的计算公式相比,  $N$  步可达集的迭代公式(3.25)仅考虑了预测时域内的外部干扰, 因此该方法获得的可达集  $R_{k+i}$  远小于最小鲁棒不变集  $F(\alpha, s)$ , 并且有  $R_{k+i} \subseteq F(\alpha, s), \forall i = 0, \dots, N_p - 1$ 。由此可知, 基于  $N$  步可达集的标称系统状态容许集  $\bar{X}_{k+i}$  和控制输入容许集  $\bar{U}_{k+i}$  将远大于基于最小鲁棒不变集的标称系统状态容许集  $\bar{X}$  和控制输入容许集  $\bar{U}$ 。与此同时, 可达集  $R_{k+i}$  会随迭代次数  $i$  的增加而逐渐增大, 标称系统的状态容许集  $\bar{X}_{k+i}$  则会随之减小, 能够确保系统在初始时刻附近的保守性较小, 同时在预测范围后续不断增大的过程中考虑累积的外部扰动, 保证系统的稳定性。

综上所述, 基于在线可达集的 Tube 计算方法相比于基于最小鲁棒不变集的 Tube 计算方法, 在保证控制系统稳定性的同时能够降低控制算法的保守性。因此本文将采用在线可达集方法进行 Tube 不变集计算。

### 3.2.2 Tube 不变集描述方式分析

在确定 Tube 不变集计算方法后, 还需确定 Tube 不变集的描述方式。不变集的描述方式主要分为多面体不变集和椭圆不变集两种形式。其中, 多面体描述方式为一种较为通用的集合描述方式, 能够更为全面地处理多种形式的干扰和不等式约束, 尤其是对非对称性的干扰处理更加准确, 但应用于 Tube 不变集的描述时数学形式较为复杂, 实时计算量较大。而椭圆不变集的描述形式则是通过假设系统中所有的干扰均为对称干扰而建立的, 因此该不变集是对系统状态向量的加权平方和进行约束。椭圆不变集约束在优化过程中具有较高的计算效率。

由于在线计算可达集的原因,  $N$  步可达集必须在每一采样时刻重新计算, 这极大地降低了 Tube 不变集的利用效率。研究表明, 随着多面体顶点数量的增加, 闵可夫斯基和的运算时间复杂度是  $O(n^6)$ <sup>[68,69]</sup>, 因此多面体集合运算会给控制算法带来巨大的实时计算量。因此本文采用椭圆不变集的描述方式, 在确保低保守性的同时, 进一步降低控制算法的实时计算效率。

椭圆不变集的描述方式如下,

$$E(p, M) := \left\{ x \in \mathbb{R}^n \mid (x - p)^T M^{-1} (x - p) \leq 1 \right\} \quad (3.28)$$

式中,  $p$  为椭圆域的中心向量, 通常可以选择标称系统的状态轨迹作为椭圆域的中心轨迹。 $M$  表示椭圆域的形状矩阵。同时, 约束集可以描述为如下多面体形式,

$$X = \left\{ x \in \mathbb{R}^n \mid H_x x \leq h_x, h_x \in \mathbb{R}^{m_x} \right\} \quad (3.29)$$

此外, 椭圆不变集相比于多面体不变集在约束条件收紧计算时计算量更小。为便于后文约束条件的收紧计算, 在此引入椭圆不变集的相关计算操作。两个椭圆不变集的闵可夫斯基和并不一定是椭圆形式的集合, 因此, Kurzhanski<sup>[70]</sup>等人提出了一种过近似的方法来计算两个椭圆形式集合的闵可夫斯基和, 其公式如下,

$$E(p_1, M_1) \oplus E(p_2, M_2) \subset E\left(p_1 + p_2, (1+c^{-1})M_1 + (1+c)M_2\right) \quad (3.30)$$

式中,  $E(p_1, M_1)$  和  $E(p_2, M_2)$  分别表示两个任意的椭圆不变集,  $c = \sqrt{\text{Tr}(M_1)/\text{Tr}(M_2)}$ ,  $\text{Tr}(M)$  表示矩阵  $M$  的迹。

此外, 椭圆形式集合的仿射变换公式为,

$$A \cdot E(p, M) + b = E\left(Ap + b, A M A^T\right) \quad (3.31)$$

下文将基于椭圆不变集进行 Tube MPC 优化问题及约束条件构建, 针对如下考虑扰动的线性离散系统,

$$x_{k+1} = Ax_k + Bu_k + Dw_k \quad (3.32)$$

分离出的不带扰动理想标称系统为,

$$\bar{x}_{k+1} = A\bar{x}_k + B\bar{u}_k \quad (3.33)$$

Tube MPC 预稳定的反馈控制律为,

$$u_k = \bar{u}_k + K(x_k - \bar{x}_k) \quad (3.34)$$

根据前文引入的椭圆不变集运算操作, 在此可以获得在  $k$  时刻的椭圆形式可达集  $\chi_k = E(\bar{x}_k, M_k)$

$$M_{k+1} = (1 + c_k^{-1})(A + BK)M_k(A + BK)^T + (1 + c_k)D\tilde{M}D^T \quad (3.35)$$

$$c_k = \sqrt{\text{Tr}((A + BK)M_k(A + BK)^T) / \text{Tr}(D\tilde{M}D^T)} \quad (3.36)$$

椭圆域的形状矩阵的取值并不依赖于初始的状态量和控制输入序列, 因此可以在优化问题公式构建之前通过公式(3.35)和公式(3.36)预先求解得到。

上述公式允许我们构建如下最优控制问题,

$$\min_{\bar{u}_0, \dots, \bar{u}_{N_p-1}} \sum_{k=0}^{N_p-1} (\bar{x}_k^T Q \bar{x}_k + \bar{u}_k^T R \bar{u}_k) + \bar{x}_{N_p}^T P \bar{x}_{N_p} \quad (3.37)$$

约束条件如下,

$$\begin{cases} \bar{x}_{k+1} = A\bar{x}_k + B\bar{u}_k \\ \bar{x}_0 = x_t \\ \begin{aligned} & [H_x]_{i,k} \bar{x}_k + \sqrt{[H_x]_{i,k} M_k [H_x]_{i,k}^T} \leq [h_x]_{i,k}, \forall i, k \\ & [H_u]_{j,k} \bar{u}_k + \sqrt{[H_u]_{j,k} K M_k K^T [H_u]_{j,k}^T} \leq [h_u]_{j,k}, \forall j, k \\ & i \in \{1, \dots, m_x\}, j \in \{1, \dots, m_u\}, k \in \{1, \dots, N_p\} \end{aligned} \end{cases} \quad (3.38)$$

根据 Hessem<sup>[71]</sup>等人提出的方法, 对于实际系统的约束条件  $x_k \in X_k$  和  $u_k \in U_k$  可以分别被改写为(3.38)中的不等式形式, 从而作为标称系统的收紧约束条件。

### 3. 3 Tube MPC 轨迹跟踪鲁棒控制器设计

上一节完成了对 Tube 不变集计算方法和描述方式的优化设计, 本节将进行 Tube MPC 轨迹跟踪鲁棒控制器的设计, 其主要结构如图 3.4 中绿色虚线内所示。其中, 闭环反馈增益将基于系统状态空间方程矩阵离线设计和计算得到, 实际系统的约束集信息将由上层的规划信息计算得到。控制算法将依据车辆的状态信息实时地进行标称

MPC 优化问题的构建与求解, 最终输出期望的横、纵向加速度给到下层的加速度跟踪控制器, 并最终输出期望的纵向力和转向角给到车辆系统以实现实时的轨迹跟踪功能。

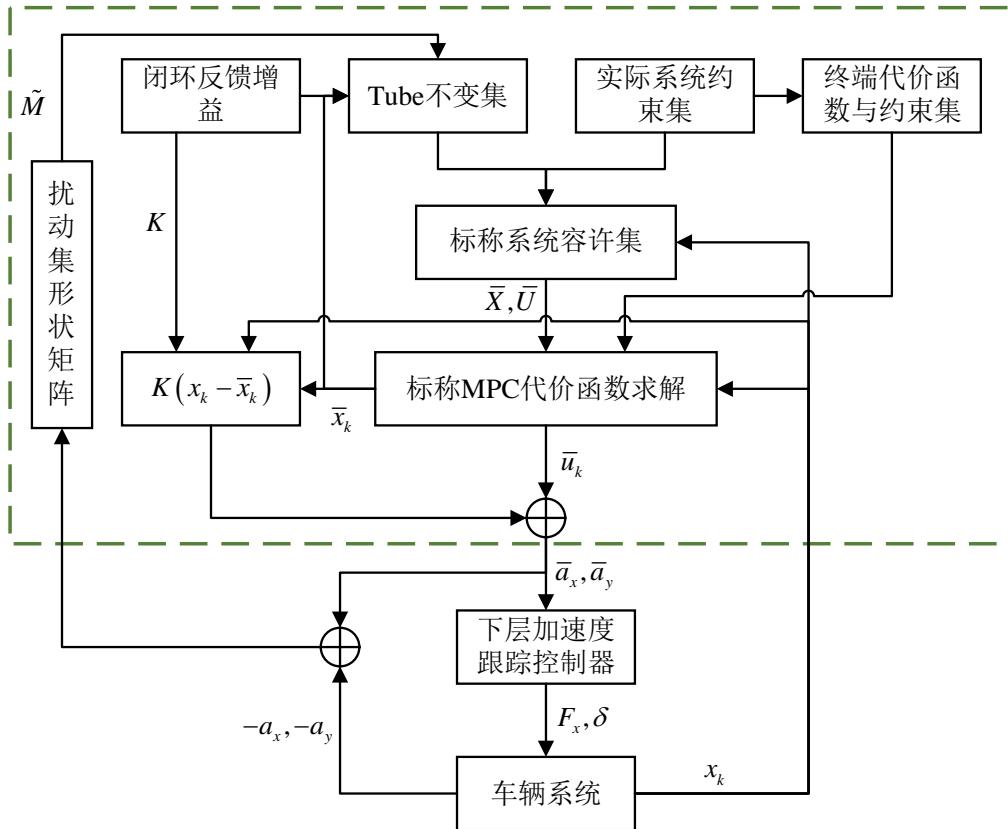


图 3.4 Tube MPC 上层轨迹跟踪鲁棒控制策略结构示意图

### 3.3.1 反馈增益设计及求解

为减小实际系统状态轨迹与标称系统状态轨迹之间的偏差, 本文设计了状态误差系统(3.12)代价函数, 其中第一项与状态量误差  $\tilde{x}_{k+i}$  相关, 确保状态估计偏差尽可能小, 第二项与控制输入  $\tilde{u}_k$  相关, 确保较小的控制能量损耗, 将该代价函数作为控制性能指标, 其表达式如下:

$$J_{\infty}(k) = \sum_{i=0}^{\infty} \tilde{x}_{k+i}^T Q_e \tilde{x}_{k+i} + \tilde{u}_{k+i}^T R_e \tilde{u}_{k+i}, i = \{0, 1, \dots, \infty\} \quad (3.39)$$

式中,  $Q_e$  和  $R_e$  分别表示状态误差系统中状态量和控制输入量的权重系数矩阵, 状态误差系统的反馈控制律为  $\tilde{u}_{k+i} = K \tilde{x}_{k+i}, i = \{0, 1, \dots, \infty\}$ ,  $K$  即为 Tube MPC 所需要设计的反馈增益矩阵。

为保证状态误差系统的稳定性, 系统的李雅普诺夫函数  $V(x)$  应该满足  $V(0)=0$ , 同时又满足,

$$V(\tilde{x}_{k+i+1})-V(\tilde{x}_{k+i})\leq 0, \forall \tilde{x}\neq 0 \quad (3.40)$$

由式(3.39)可知, 所设计控制性能指标  $J_\infty(k)$  由于是无限时域无法直接计算, 因此需要获得其上界, 需对  $V(x)$  进行进一步约束, 形式如下,

$$V(\tilde{x}_{k+i+1})-V(\tilde{x}_{k+i})\leq -\left[\tilde{x}_{k+i}^T Q_e \tilde{x}_{k+i} + \tilde{u}_{k+i}^T R_e \tilde{u}_{k+i}\right], i=\{0, 1, \dots, \infty\} \quad (3.41)$$

叠加公式(3.41)可得,

$$J_\infty(k)\leq V(\tilde{x}_k) \quad (3.42)$$

由式(3.42)可得  $J_\infty(k)$  的上界为  $V(\tilde{x}_k)$ , 最小化  $V(\tilde{x}_k)$  即可最小化  $J_\infty(k)$ , 可以将状态误差系统的控制性能指标转换为  $V(\tilde{x}_k)$  以便后续计算。

综上所述, 反馈增益矩阵设计应满足:

- 1) 使状态误差系统的控制性能最优, 即  $V(\tilde{x}_k)$  最小;
- 2) 使状态误差系统稳定, 即  $V(\tilde{x}_k)$  满足约束(3.41)。

确定反馈增益矩阵的设计需求之后, 下文将针对反馈增益矩阵的求解进行推导。

首先引入 Schur 补定理, 其表述如下:

对于如下的线性矩阵不等式,

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} > 0 \quad (3.43)$$

等价于如下形式,

$$R > 0, Q - SR^{-1}S^T > 0 \quad (3.44)$$

或者为如下形式,

$$Q > 0, R - S^T Q^{-1} S > 0 \quad (3.45)$$

式中,  $Q = Q^T$ ,  $R = R^T$ ,  $S$  为适当维数的矩阵。

根据反馈增益矩阵设计需求, 本文将采用经典的二次型函数  $V(x) = x^T P x$  来定义李雅普诺夫函数, 其中  $P$  为正定矩阵。同时令  $S = \gamma P^{-1}$ , 那么有关控制性能指标函数  $V(\tilde{x}_k) = \tilde{x}_k^T P \tilde{x}_k$  的最小化问题就可以通过 Schur 补定理转化为如下线性矩阵不等式的特征值问题:

$$\min_{\gamma, S} \gamma \quad (3.46)$$

约束为,

$$\begin{bmatrix} I & \tilde{x}_k^T \\ \tilde{x}_k & S \end{bmatrix} \geq 0 \quad (3.47)$$

为在最小化控制性能指标函数  $V(\tilde{x}_k)$  的同时保证系统稳定, 应确保满足反馈增益矩阵设计的第二条, 即  $V(\tilde{x}_k)$  应满足式(3.41), 将状态误差系统的控制性能指标函数  $V(\tilde{x}_k) = \tilde{x}_k^T P \tilde{x}_k$  代入式(3.41)可以得到,

$$\tilde{x}_{k+1}^T P \tilde{x}_{k+1} - \tilde{x}_k^T P \tilde{x}_k \leq -[\tilde{x}_k^T Q_e \tilde{x}_k + \tilde{u}_k^T R_e \tilde{u}_k] \quad (3.48)$$

综合状态误差系统的状态方程(3.12)以及反馈控制律(3.11), 约束条件式(3.47)可简化为如下形式,

$$(A + BK)^T P (A + BK) - P + K^T R K + Q_e \leq 0 \quad (3.49)$$

通过 Schur 补定理将式(3.49)整理为线性矩阵不等式形式, 从而可以将其与线性矩阵不等式特征值问题(3.46)进行联立求解。令  $P = \gamma S^{-1}$ , 式(3.49)即可转化为如下约束条件形式:

$$\begin{bmatrix} S & SA^T + (KS)^T B^T & S Q_e^{\frac{1}{2}} & (KS)^T R_e^{\frac{1}{2}} \\ AS + BKS & S & 0 & 0 \\ Q_e^{\frac{1}{2}} S & 0 & I & 0 \\ R_e^{\frac{1}{2}} KS & 0 & 0 & I \end{bmatrix} \geq 0 \quad (3.50)$$

(3.46)、(3.47)以及(3.50)联立组成了一个标准的线性矩阵不等式特征值问题。利用 MATLAB 内部的 LMI Toolbox 并结合车辆参数建立和求解 LMI 系统模型, 即可完成了对上述问题的离线求解并得到状态误差系统的反馈增益矩阵  $K$ 。

### 3.3.2 标称 MPC 优化问题构建

在获得 Tube 不变集的形状矩阵信息之后就可以开始对标称 MPC 的优化问题进行构建和求解。

此外, 为了使最优化问题求解得到的控制输入序列, 即横、纵向的校正加速度  $\Delta a_x$  和  $\Delta a_y$ , 变化更加平滑, 从而得到更利于跟踪的期望加速度轨迹, 同时也是为了动力学模型能更好地匹配实际系统的行为, 本文将横向和纵向加速度的变化率作为二次正则项添加到标称 MPC 优化问题的代价函数中。同时, 为了简化计算, 本文将基于线

性化参考点的横、纵向加速度值与校正加速度之和进行线性化，可得横、纵向加速度的正则项如下，

$$\tilde{a}_x = a_{x,op} + \Delta a_x \quad (3.51)$$

$$\tilde{a}_y = a_{y,op} + \Delta a_y \quad (3.52)$$

式中， $a_{x,op}$  和  $a_{y,op}$  分别表示横、纵向加速度在线性化参考点处的取值。

尽管本文采用了一种鲁棒模型预测控制策略去构建优化问题，但违反扰动假设的情况仍然可能会导致优化问题没有可行解。因此，依据精确惩罚函数的思想<sup>[43]</sup>，本文将引入松弛因子  $\varepsilon$  来软化约束。在优化问题有可行解的情况下，使用一个一范数作为松弛成本以及一个二范数作为一个小的正则项以确保获得硬约束解。同时，作为额外的安全措施，本文将松弛因子限制在最大违规范围的 5%，并在无法在该容许范围内求得可行解的情况下触发安全紧急操作。

最终得到标称 MPC 优化问题的代价函数如下，

$$\sum_{k=0}^{N_p-1} \left( \bar{x}_k^T Q \bar{x}_k + \bar{u}_k^T R \bar{u}_k + \gamma_x \tilde{a}'_x^2 + \gamma_y \tilde{a}'_y^2 \right) + \bar{x}_{N_p}^T P \bar{x}_{N_p} + \rho_1 \|\varepsilon\|_1 + \rho_2 \|\varepsilon\|_2 \quad (3.53)$$

式中， $\tilde{a}'_x$  和  $\tilde{a}'_y$  为横、纵向加速度正则项的变化率，由连续两个阶段的近似横、纵向加速度之差计算得到， $\gamma_x$  和  $\gamma_y$  分别为其相对应的代价函数权重系数。松弛因子相对应的权重系数分别为  $\rho_1$  和  $\rho_2$ 。

### 3.3.3 约束条件构造及收紧

上文完成了标称 MPC 优化问题的构建，下文将进行约束条件的构造。

标称 MPC 优化问题的约束条件包括等式约束和不等式约束。其中，等式约束部分主要用于根据初始时刻状态量和控制输入序列去推导预测时域的状态量，主要为以下两式，

$$\bar{x}(k+1) = A\bar{x}(k) + B\bar{u}(k) \quad (3.54)$$

$$\bar{x}(0) = x_i \quad (3.55)$$

其中，式(3.54)为状态预测方程，式(3.55)中  $x_i$  表示测量得到的当前时刻状态量，作为状态量初值。等式约束部分主要实现 MPC 算法中的预测功能，后续将会被代入到不等式约束中用以构建求解器易于求解的约束形式。

根据 2.2.2 节的式(2.15)和式(2.16)可知，标称 MPC 的优化问题的不等式约束条件包括两部分，分别为对横向距离偏差的约束和对横、纵向加速度极限的约束。其中，

前者体现为对状态量的约束，后者体现为对控制输入量的约束。下文将对两种约束的收紧形式进行推导。

下文对第一部分约束进行分析，横向距离偏差的约束为如下形式，

$$d^-(k) \leq d(k) \leq d^+(k) \quad (3.56)$$

其中， $k = 0, 1, \dots, N_p - 1$  表示预测时域内的第  $k$  时刻， $d^+(k)$  和  $d^-(k)$  分别表示跟踪过程中自车和参考轨迹的横向距离偏差的左右极限边界，是由上层规划信息针对道路宽度规划获取得到的序列信息。为便于约束收紧运算以及二次规划求解器求解，将实际系统需满足的横向跟踪偏差约束写为如式(3.29)一样的多面体形式如下，

$$X = \left\{ x(k) \in \mathbb{R}^n \mid H_x \bar{x}(k) \leq h_x, h_x \in \mathbb{R}^{m_x} \right\} \quad (3.57)$$

根据 3.2.2 节中所介绍的约束收紧方法，依据椭圆 Tube 不变集的形状矩阵  $M_k$  将约束式(3.57)收紧为如下形式，即为标称系统需满足的状态量约束：

$$[H_x]_{i,k} \bar{x}(k) + \sqrt{[H_x]_{i,k} M_k [H_x]_{i,k}^T} \leq [h_x]_{i,k}, \forall i \in \{1, \dots, m_x\}, k \in \{1, \dots, N_p\} \quad (3.58)$$

由控制系统的车辆动力学模型(2.18)可知，状态向量由速度偏差  $\Delta v$ 、横向距离偏差  $d$  和横向距离偏差变化率  $\dot{d}$  三部分组成，即  $n = 3$ 。关于横向距离偏差的约束包括上、下界两部分，即  $m_x = 2$ 。由式(3.56)易推导得，

$$[H_x]_{1,k} = [0 \ 1 \ 0] \quad (3.59)$$

$$[H_x]_{2,k} = [0 \ -1 \ 0] \quad (3.60)$$

$$[h_x]_{1,k} = d^+(k) - \sqrt{[H_x]_{1,k} M_k [H_x]_{1,k}^T} \quad (3.61)$$

$$[h_x]_{2,k} = -d^-(k) - \sqrt{[H_x]_{2,k} M_k [H_x]_{2,k}^T} \quad (3.62)$$

下文对第二部分约束进行分析，横、纵向加速度的约束为如下形式，

$$\pm a_x a_{y,\max} \pm a_y a_{x,\max} \leq a_{x,\max} a_{y,\max} \quad (3.63)$$

其中，横、纵向加速度最大极限  $a_{x,\max}$  和  $a_{y,\max}$  是由上层轨迹规划得到的序列信息。易知式(3.63)共包含以下四种情况，

$$a_x a_{y,\max} + a_y a_{x,\max} \leq a_{x,\max} a_{y,\max} \quad (3.64)$$

$$-a_x a_{y,\max} - a_y a_{x,\max} \leq a_{x,\max} a_{y,\max} \quad (3.65)$$

$$-a_x a_{y,\max} + a_y a_{x,\max} \leq a_{x,\max} a_{y,\max} \quad (3.66)$$

$$a_x a_{y,\max} - a_y a_{x,\max} \leq a_{x,\max} a_{y,\max} \quad (3.67)$$

由上式易知, 横、纵向加速度极限的约束为非线性约束, 为了便于将优化问题转化为易于求解的二次规划的形式, 需要对横、纵向加速度极限约束进行线性化, 其中线性化的推导过程可见本文 2.2.2 节, 为便于理解, 将推导结果符号化, 改写为如下形式,

$$\begin{aligned} & (a_{x,grad,states}(k) + a_{y,grad,states}(k))\bar{x}(k) + (a_{x,grad,inputs}(k) + a_{y,grad,inputs}(k))\bar{u}(k) \\ & + a_{x,op}a_{y,max} + a_{y,op}a_{x,max} \leq a_{x,max}a_{y,max} \end{aligned} \quad (3.68)$$

$$\begin{aligned} & -(a_{x,grad,states}(k) + a_{y,grad,states}(k))\bar{x}(k) - (a_{x,grad,inputs}(k) + a_{y,grad,inputs}(k))\bar{u}(k) \\ & - a_{x,op}a_{y,max} - a_{y,op}a_{x,max} \leq a_{x,max}a_{y,max} \end{aligned} \quad (3.69)$$

$$\begin{aligned} & (-a_{x,grad,states}(k) + a_{y,grad,states}(k))\bar{x}(k) + (-a_{x,grad,inputs}(k) + a_{y,grad,inputs}(k))\bar{u}(k) \\ & - a_{x,op}a_{y,max} + a_{y,op}a_{x,max} \leq a_{x,max}a_{y,max} \end{aligned} \quad (3.70)$$

$$\begin{aligned} & (a_{x,grad,states}(k) - a_{y,grad,states}(k))\bar{x}(k) + (a_{x,grad,inputs}(k) - a_{y,grad,inputs}(k))\bar{u}(k) \\ & + a_{x,op}a_{y,max} - a_{y,op}a_{x,max} \leq a_{x,max}a_{y,max} \end{aligned} \quad (3.71)$$

式中,  $a_{x,grad,states}(k) = \left[ \frac{\partial a_x}{\partial \Delta v}(k), \frac{\partial a_x}{\partial d}(k), \frac{\partial a_x}{\partial \dot{d}}(k) \right]$  表示在预测时域的第  $k$  时刻纵向加速度  $a_x$  对各状态量的偏导项,  $a_{y,grad,states}(k) = \left[ \frac{\partial a_y}{\partial \Delta v}(k), \frac{\partial a_y}{\partial d}(k), \frac{\partial a_y}{\partial \dot{d}}(k) \right]$  表示在预测时域第  $k$  时刻横向加速度  $a_y$  对各状态量的偏导项,  $a_{x,grad,inputs}(k) = \left[ \frac{\partial a_x}{\partial \Delta a_x}(k), \frac{\partial a_x}{\partial \Delta a_y}(k) \right]$  表示在预测时域的第  $k$  时刻纵向加速度  $a_x$  对各控制输入量的偏导项,  $a_{y,grad,inputs}(k) = \left[ \frac{\partial a_y}{\partial \Delta a_x}(k), \frac{\partial a_y}{\partial \Delta a_y}(k) \right]$  表示在预测时域的第  $k$  时刻横向加速度  $a_y$  对各控制输入量的偏导项, 其中各个偏导项的推导结果由 2.2.2 节可知。 $a_{x,op} = \bar{a}_x v_p$  和  $a_{y,op} = \bar{k} v_p^2$  分别表示横、纵向加速度在线性化参考点  $(d=0, \dot{d}=0, \Delta a_x=0, \Delta a_y=0)$  处的取值。

将实际系统需满足的控制输入量约束也写成如式(3.29)一样的多面体形式,

$$U = \left\{ u(k) \in \mathbb{R}^m \mid H_u \bar{u}(k) \leq h_u, h_u \in \mathbb{R}^{m_u} \right\} \quad (3.72)$$

根据 3.2.2 节中约束收紧方法, 依据椭圆 Tube 不变集的形状矩阵  $M_k$  将约束式(3.72)收紧为如下形式, 即为标称系统的控制输入约束:

$$[H_u]_{j,k} \bar{u}(k) + \sqrt{[H_u]_{j,k} K M_k K^T [H_u]_{j,k}^T} \leq [h_u]_{j,k}, \forall j \in \{1, \dots, m_u\}, k \in \{1, \dots, N_p\} \quad (3.73)$$

由控制系统的车辆动力学模型(2.18)可知, 控制输入量由横、纵向加速度校正量两部分组成, 即  $m=2$ , 且加速度极限约束集在预测时域内的每个时刻包含 4 个不等式约束, 即  $m_u=4$ 。由横、纵向加速度极限约束的线性化推导公式可得,

$$[H_u]_{1,k} = a_{x,grad,inputs}(k) + a_{y,grad,inputs}(k) = [-a_{y,max}(k), a_{x,max}(k)] \quad (3.74)$$

$$[H_u]_{2,k} = -a_{x,grad,inputs}(k) - a_{y,grad,inputs}(k) = [a_{y,max}(k), -a_{x,max}(k)] \quad (3.75)$$

$$[H_u]_{3,k} = -a_{x,grad,inputs}(k) + a_{y,grad,inputs}(k) = [a_{y,max}(k), a_{x,max}(k)] \quad (3.76)$$

$$[H_u]_{4,k} = a_{x,grad,inputs}(k) - a_{y,grad,inputs}(k) = [-a_{y,max}(k), -a_{x,max}(k)] \quad (3.77)$$

$$\begin{aligned} [h_u]_{1,k} = & a_{x,max} a_{y,max} - (a_{x,grad,states}(k) + a_{y,grad,states}(k)) \bar{x}(k) \\ & - a_{x,op} a_{y,max} - a_{y,op} a_{x,max} - \sqrt{[H_u]_{1,k} K M_k K^T [H_u]_{1,k}^T} \end{aligned} \quad (3.78)$$

$$\begin{aligned} [h_u]_{2,k} = & a_{x,max} a_{y,max} + (a_{x,grad,states}(k) + a_{y,grad,states}(k)) \bar{x}(k) \\ & + a_{x,op} a_{y,max} + a_{y,op} a_{x,max} - \sqrt{[H_u]_{2,k} K M_k K^T [H_u]_{2,k}^T} \end{aligned} \quad (3.79)$$

$$\begin{aligned} [h_u]_{3,k} = & a_{x,max} a_{y,max} - (-a_{x,grad,states}(k) + a_{y,grad,states}(k)) \bar{x}(k) \\ & + a_{x,op} a_{y,max} - a_{y,op} a_{x,max} - \sqrt{[H_u]_{3,k} K M_k K^T [H_u]_{3,k}^T} \end{aligned} \quad (3.80)$$

$$\begin{aligned} [h_u]_{4,k} = & a_{x,max} a_{y,max} - (a_{x,grad,states}(k) - a_{y,grad,states}(k)) \bar{x}(k) \\ & - a_{x,op} a_{y,max} + a_{y,op} a_{x,max} - \sqrt{[H_u]_{4,k} K M_k K^T [H_u]_{4,k}^T} \end{aligned} \quad (3.81)$$

在完成约束条件的构建后，易发现两个约束条件式(3.58)和式(3.73)都需要 Tube 椭圆不变集在每一时刻更新计算出其在预测时域内每一步的形状矩阵  $M_k$ ，从而能够收紧预测时域内的每一约束条件。形状矩阵  $M_k$  在预测时域内的更新计算需要已知当前时刻的形状矩阵初始值  $M_0$  以及当前时刻的扰动集形状矩阵  $\tilde{M}$ 。其中，由于预测时域的初始时刻外部干扰引起的偏差累积可近似视为零，因此形状矩阵初始值被设定为接近零矩阵的形式，即，

$$M_0 = \begin{bmatrix} 1 \times 10^{-8} & 0 & 0 \\ 0 & 1 \times 10^{-8} & 0 \\ 0 & 0 & 1 \times 10^{-8} \end{bmatrix} \quad (3.82)$$

扰动集形状矩阵  $\tilde{M}$  则由扰动边界在线辨识得到的横、纵向加速度扰动边界  $\tilde{d}_{a,x}$  和  $\tilde{d}_{a,y}$  得到，扰动边界在线辨识所采用的方法为 2.3.4 节中所介绍的方法。扰动集形状矩阵  $\tilde{M}$  的形式如下，

$$\tilde{M} = \begin{bmatrix} \tilde{d}_{a,x}^2 & 0 \\ 0 & \tilde{d}_{a,y}^2 \end{bmatrix} \quad (3.83)$$

同时,由于外部干扰信号的边界值只在控制循环的每一周期进行一次不确定性辨识,因此需要在预测时域内考虑扰动信号累积的情况。在此本文将采用以和每一时刻加速度极限成正相关的线性增量的方式对扰动信号累积情况进行模拟,从而进一步扰动集形状矩阵  $\tilde{M}$  的精准度以及控制系统的稳定性,则可以将扰动边界在线辨识得到的横、纵向加速度扰动边界  $\tilde{d}_{a,x}$  和  $\tilde{d}_{a,y}$  以及扰动集形状矩阵  $\tilde{M}$  重写为如下形式,

$$\tilde{d}_{a,x}(k+1) = \tilde{d}_{a,x}(k) + \alpha \cdot a_{x,\max}(k) \quad (3.84)$$

$$\tilde{d}_{a,y}(k+1) = \tilde{d}_{a,y}(k) + \alpha \cdot a_{y,\max}(k) \quad (3.85)$$

$$\tilde{M}_k = \begin{bmatrix} \tilde{d}_{a,x}^2(k) & 0 \\ 0 & \tilde{d}_{a,y}^2(k) \end{bmatrix} \quad (3.86)$$

式中,  $\alpha$  表示增量系数,本文将其设定为  $\alpha = 0.08\%$ ,即预测时域每向前推一步,扰动信号边界值累积增加量为加速度极限  $0.08\%$ 。

### 3.3.4 终端代价函数及终端约束集设计

终端约束集和终端代价函数是两种保证 MPC 控制稳定性的常用手段。终端约束集是指在 MPC 控制中对系统状态和控制输入施加的最终约束条件。这些约束条件通常与系统在给定时间内达到一种期望行为或满足某种性能指标有关。终端约束集的目的是将系统的状态限制在一定范围内,在设计时要考虑系统的可行域,并确保最终控制输入和状态处于合理的范围内,以维持系统的稳定性。终端约束集的定义通常基于系统的物理限制和设计需求。

终端代价函数是在 MPC 控制优化问题中引入的一种目标函数。它量化了对系统状态和控制输入进行调节的目标或成本。终端代价函数的设计旨在使系统在给定时间内达到期望的行为,并优化控制输入的选择,以实现所需的性能。终端代价函数一般由状态误差、控制输入变化率、控制输入大小等组成,并根据具体需求和问题进行合理的权衡。通过最小化终端代价函数,可以在每个时间步选择最优的控制输入,以实现稳定和最优的控制。

同时使用终端约束集和终端代价函数可以确保系统的稳定性、提高控制的鲁棒性和性能优化,并使 MPC 控制系统能够在不同应用场景中灵活适应和应对各种挑战。因此本文将采用两种方法相结合的方式来维持控制系统的稳定性。

首先本文将采用二次型李雅普诺夫函数作为终端代价函数,因此有,

$$F(\tilde{x}_{i+1}) - F(\tilde{x}_i) = \tilde{x}_{i+1}^T P \tilde{x}_{i+1} - \tilde{x}_i^T P \tilde{x}_i \leq 0 \quad (3.87)$$

对权重矩阵  $P$  求解问题可转化为前文建立的 LMI 特征值问题(3.46)，因此终端代价函数的权重矩阵可表示为，

$$P = \gamma S^{-1} \quad (3.88)$$

本文采用鲁棒一步集与最大鲁棒不变集来计算标称系统 MPC 优化问题的终端约束集，其中鲁棒一步集的定义如下：

对于包含不确定性的有界线性离散系统，有  $x_k \in \Xi(\Omega)$ ，有界扰动  $w_k \in W$ ，如果控制输入量与状态量满足相应的约束条件，并存在容许控制输入使得  $x_{k+1} \in \Omega$ ，则称  $\Xi(\Omega)$  是集合  $\Omega$  的鲁棒一步集，其形式如下，

$$\Xi(\Omega) := \{x_k \in X \mid \exists u_k \in U : Ax_k + Bu_k + w_k \in \Omega, \forall w_k \in W\} \quad (3.89)$$

最大鲁棒不变集的定义如下：

对于鲁棒不变集  $O_\infty \subseteq X$ ，当且仅当  $O_\infty$  包含  $X$  内的所有鲁棒不变集时，则称  $O_\infty$  是不确定性有界线性离散系统的最大鲁棒不变集。

由最大鲁棒不变集的定义可知，若  $O_\infty$  为控制系统的最大鲁棒不变集，那么当控制系统的初始状态满足  $x_0 \in O_\infty$  时，控制系统的状态方程在经过有限次迭代后得到的预测时域内的系统状态仍然满足  $x_k \in X$ 。因此采用最大鲁棒不变集  $O_\infty$  作为轨迹跟踪控制系统的终端约束集，能够保证控制系统状态在有限的预测时域内有界。

最大鲁棒不变集  $O_\infty$  的计算步骤如下：

首先进行初始化，设定  $\Omega_0 = X$ ，其中  $X$  为实际控制系统需要满足的状态量约束，其形式如式(3.57)所示。

其次需要进行迭代计算，在第  $k$  次 ( $k \geq 0$ ) 迭代计算中，首先根据式(3.89)进行鲁棒一步集的计算，得到  $\Xi(\Omega_k)$ ，然后将鲁棒一步集  $\Xi(\Omega_k)$  与集合  $\Omega_k$  取交集即可得到集合  $\Omega_{k+1}$ ，即  $\Omega_{k+1} = \Omega_k \cap \Xi(\Omega_k)$ 。

最后需要确定终止条件，若  $\Omega_{k+1} = \Omega_k$ ，则表明鲁棒不变集已经完成收敛， $\Omega_{k+1}$  即为控制系统的最大鲁棒不变集，迭代过程结束；若  $\Omega_{k+1} \neq \Omega_k$ ，则返回上一步，继续进行迭代计算过程。

### 3.4 本章小结

本章通过对 Tube MPC 原理进行分析，通过引入反馈增益矩阵，构建了理想标称

系统与实际系统之间的状态误差系统；设计了基于鲁棒不变集的约束条件收紧方法；通过对 Tube 不变集与控制保守性之间关系的分析，确定了基于在线可达集的 Tube 不变集计算方法；通过权衡在线可达集计算效率和保守性，选择了以椭圆不变集的形式描述 Tube 不变集；设计了基于 Tube MPC 的上层轨迹跟踪鲁棒控制策略，完成了算法中局部控制器反馈增益矩阵的计算，构建了轨迹跟踪控制优化问题，并推导了加速度约束条件及道路边界约束条件的收紧形式，设计了终端代价函数和终端约束集以保证系统的渐进稳定。

## 第4章 下层加速度跟踪控制及优化策略

由上层轨迹跟踪控制器得到的期望横、纵向加速度是操纵车辆按规划参考轨迹行驶的关键信息，需要交由下层加速度跟踪控制器进行跟踪控制，输出期望的前轮转向角和纵向力给到车辆的执行机构层，从而最终实现对车辆的合理操纵。本文将采用基于前馈补偿的比例积分微分（Proportional-Integral-Derivative, PID）反馈控制方法分别进行横、纵向加速度跟踪控制器的设计。针对加速跟踪控制过程中前馈与期望输出偏差，设计了基于误差模型估计的前馈修正策略。针对执行器信号延迟问题，设计了基于模型参考自适应控制的控制调整策略。

### 4.1 基于前馈和反馈结合的加速度跟踪控制策略

本文所提出的基于上下分层思想的轨迹跟踪控制策略，上层 Tube MPC 轨迹跟踪鲁棒控制器是基于考虑加速度约束的点质量车辆模型进行构建的，考虑了外部干扰和参数不确定性，输出期望的横、纵向加速度，下层加速度跟踪控制器则需要对期望加速度进行跟踪控制并输出对车辆的控制指令，即前轮转角和纵向力。本文的控制策略不同于其他文献中常见的基于复杂车辆模型并直接求解对车辆操纵指令的 MPC 方法，消除了对复杂轮胎模型的需求，但车辆对加速度的响应能力是否匹配模型预测控制器的期望将变得更加重要，因此下层加速度控制器将采用基于前馈和 PID 反馈结合的高效控制方法。

前馈控制和反馈控制都是控制理论中常见的控制方法。前馈控制是一种基于预期输出的控制方式，它通过提前计算出输入与输出之间的关系，将理想的输入信号直接应用于系统中。其原理是根据系统的数学模型，通过输入-输出关系的逆运算来生成合适的控制信号。前馈控制可以快速响应外部干扰或变化，并减小这些干扰对系统性能的影响。然而，前馈控制只能根据模型的准确性来预测系统的输出，当系统存在不确定性或模型误差时，前馈控制可能无法产生理想的控制效果。

反馈控制则是一种基于系统的实际输出与期望输出之间的差异进行调节的控制方式。其原理是将系统的实际输出通过传感器获得后，与期望输出进行比较，得到误差信号，再根据该误差信号进行控制。反馈控制能够根据实际情况及时调整控制输入信号，以减小误差，并保持系统的稳定性和鲁棒性。然而，反馈控制在响应外部干扰或变化时可能存在较长的调整时间，且对系统的稳定性和鲁棒性要求较高。

前馈控制和反馈控制结合的优势在于兼具二者的优点，并弥补各自的不足。前馈控制可以提前预测系统的输出，并快速响应外部干扰或变化，从而减小这些干扰对系统的影响。反馈控制能够根据实际输出与期望输出之间的差异进行修正，以确保系统稳定性和鲁棒性。通过将前馈控制和反馈控制相结合，可以实现更精确、更快速的控制效果。如图所示为前馈和反馈控制结合的结构示意图。

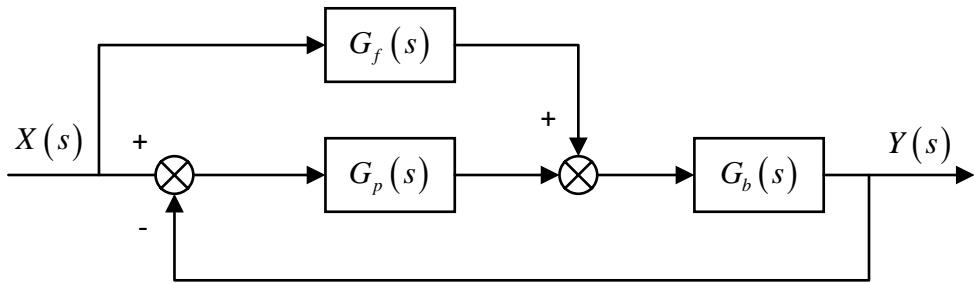


图 4.1 前馈和反馈结合的控制系统结构图

则传递函数为：

$$G_0(s) = \frac{Y(s)}{X(s)} = \frac{G_p(s)G_b(s) + G_f(s)G_b(s)}{1 + G_p(s)G_b(s)} \quad (4.1)$$

系统误差  $E(s) = X(s) - Y(s)$ ，代入式(4.1)可得系统误差对输入的传递函数为：

$$G(s) = \frac{E(s)}{X(s)} = \frac{1 - G_f(s)G_b(s)}{1 + G_p(s)G_b(s)} \quad (4.2)$$

若使  $1 - G_f(s)G_b(s) = 0$  即  $G_f(s) = 1/G_b(s)$ ，则可以使系统误差  $E(s)$  为零。

系统增加前馈和未增加前馈时，传递函数的分母是相同的，即两种情况下传递函数的极点是相同的，因此增加前馈控制不会影响系统的稳定性，却可以在不改变原系统参数和结构的情况下，大大提高系统的稳态精度，动态性能也比较容易得到保证。

#### 4.1.1 横向加速度跟踪控制策略设计

横向加速度  $a_y$  的跟踪控制策略结构示意图如图 4.2 所示，由图易知，前馈和 PID 反馈模块分别接收上层轨迹跟踪控制传递来的横向期望加速度信号  $\bar{a}_y$  和横向加速度极限值  $a_{y,max}$  以及车辆系统的当前的状态信息，包括横向加速度  $a_y$  和纵向速度  $v_x$  等，由前馈和 PID 反馈模块分别计算出相应的期望转角值  $\delta_{FF}$  和  $\delta_{FB}$  输出给车辆的转向执行机构实现对横向加速度的跟踪控制。

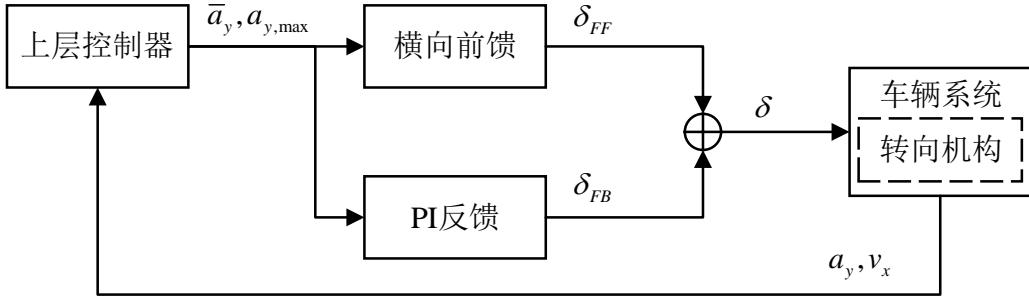


图 4.2 横向加速度跟踪控制策略结构示意图

前馈控制是一种基于预期输出的控制方式，它通过提前计算出输入与输出之间的关系，将理想的输入信号直接应用于系统中。前馈的期望转向角  $\delta_{FF}$  主要由车辆运动学关系得到<sup>[43]</sup>，

$$\delta_{FF} = \frac{L}{R} \quad (4.3)$$

式中， $L$  为车辆轴距， $R$  为车辆的转向半径。其中，车辆转向半径  $R$  可以由横向加速度  $a_y$  和纵向速度  $v_x$  计算得到，即  $R = v^2/a_y$ ，将该式代入到式(4.3)中得到：

$$\delta_{FF} = \frac{a_y L}{v^2} \quad (4.4)$$

反馈控制模块则采用经典的 PI (Proportional-Integral) 控制方法，即由比例控制和积分控制组成。其中，比例控制是基于系统反馈误差的比例关系来实现控制的，本控制系统的反馈误差为加速度跟踪误差，即  $\tilde{a}_y = a_y - \bar{a}_y$ 。比例控制能够快速响应系统的变化，减小稳态误差，提高系统的动态性能，但是比例控制无法消除系统的稳态误差，因此需要结合积分控制。积分控制是基于系统反馈误差的累计关系来实现控制的，它通过对反馈误差进行积分，并将积分结果作为控制器的一部分，以消除系统的稳态误差。经典的 PI 控制方法将比例控制与积分控制相结合，不仅能够快速响应系统的变化，还能消除稳态误差，提高加速度跟踪控制系统的精确度和稳定性。同时，在实际应用中也更易于调试和优化。

反馈模块的计算公式如下：

$$\delta_{FB} = \frac{1}{v^2} (K_{P,y} \tilde{a}_y + a_{I,y}) \quad (4.5)$$

$$\dot{a}_{I,y} = K_{I,y} \tilde{a}_y - \left( \frac{a_{I,y}}{c_1 a_{y,max}} \right)^3 \quad (4.6)$$

式中， $K_{P,y}$  和  $K_{I,y}$  分别为横向加速度反馈控制的比例系数和积分系数。式(4.6)中的第

一项表示对加速度跟踪误差的积分累计, 第二项为对积分控制器的软约束, 防止积分饱和的情况。所谓的积分饱和就是指控制系统长时间存在一个方向的偏差, 比如车辆需要长时间的向左转向且存在一定的转向不足时, PID 控制器的输出由于积分器的作用将不断累积而扩大, 从而导致控制器的输出不断增大超出正常范围进入饱和区, 即控制器输出转向角值超出了车辆转向执行机构的限制范围<sup>[72]</sup>。当控制系统出现反向的跟踪偏差时, 首先需要从饱和区退出, 从而不能对反向的跟踪偏差进行快速的响应。因此, 在积分器中添加抗积分饱和项, 该项主要由横向加速度极限值  $a_{y,\max}$  和调节系数  $c_1$  所确定, 其基本思想是当积分项值  $a_{I,y}$  超过横向加速度极限  $a_{y,\max}$  时, 对积分项施加负反馈, 使其尽快退出饱和, 调节系数  $c_1$  用于调节积分项饱和边界。

#### 4.1.2 纵向加速度跟踪控制策略设计

纵向加速度  $a_y$  的跟踪控制策略如图 4.3 所示, 与横向加速度跟踪控制策略相类似, 接收上层轨迹跟踪控制器传递来的期望纵向加速度  $\bar{a}_x$  以及车辆系统反馈的当前纵向加速度  $a_x$  和纵向速度  $v_x$  等状态信息, 根据相应的系统误差分别计算前馈的期望纵向驱动力  $F_{x,FF}$  和反馈的期望纵向力  $F_{x,FB}$ , 从而输出合理的纵向驱动力  $F_x$  给车辆的驱动机构实现对纵向加速度的跟踪控制。

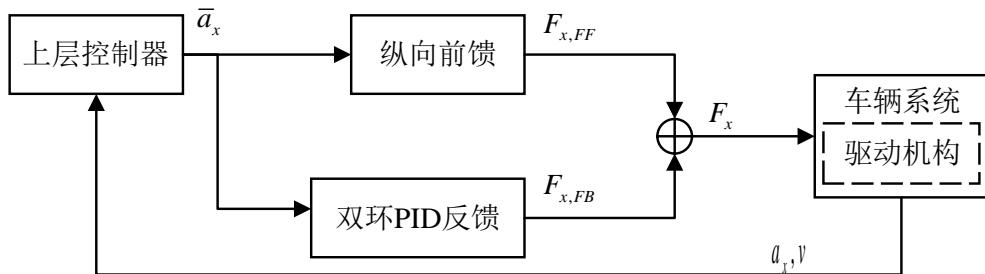


图 4.3 纵向加速度跟踪控制策略结构示意图

前馈模块计算主要由车辆的纵向行驶方程计算得到, 其计算公式如下:

$$F_{x,FF} = F_f + F_w + F_i + F_j \quad (4.7)$$

式中,  $F_f$  表示滚动阻力,  $F_w$  表示空气阻力,  $F_i$  表示坡度阻力,  $F_j$  表示加速阻力。为简化计算, 本文主要研究水平路况, 故假设路面坡度为零, 即  $F_i = 0$ 。

滚动阻力  $F_f$  的计算公式如下:

$$F_f = mgf \quad (4.8)$$

式中,  $m$  为车辆质量,  $g$  为重力加速度,  $f$  为滚动阻力系数。

空气阻力  $F_w$  的计算公式如下：

$$F_w = \frac{1}{2} C_D A \rho v_x^2 \quad (4.9)$$

式中， $C_D$  为空气阻力系数， $A$  为车辆的迎风面积，即车辆行驶方向的投影面积， $\rho$  为空气密度。

加速阻力  $F_j$  的计算公式如下：

$$F_j = m \bar{a}_x \quad (4.10)$$

综上所述即可得到前馈的期望纵向驱动力计算公式，即：

$$F_{x,FF} = m \bar{a}_x + \frac{1}{2} C_D A \rho v_x^2 + m g f \quad (4.11)$$

反馈模块采用速度和加速度的双环 PID 反馈控制方法，以确保在纵向加速度跟踪控制的同时，纵向速度也能对规划速度有较好的跟踪效果。如图 4.4 所示为双环 PID 方法的控制结构图，外环为纵向速度的 PID 反馈控制，内环为纵向加速度的 PID 反馈控制。纵向速度控制部分依据实际纵向速度  $v_x$  和规划纵向速度  $\bar{v}_x$  的偏差，计算纵向加速度补偿值  $a_{com}$  输出到纵向加速度偏差计算部分。纵向加速度控制部分则依据由上层轨迹跟踪控制器得到的期望纵向加速度  $\bar{a}_x$  加上纵向加速度补偿量  $a_{com}$  后与实际纵向加速度  $a_x$  的偏差，计算最终的反馈期望纵向驱动力  $F_{x,FB}$ 。

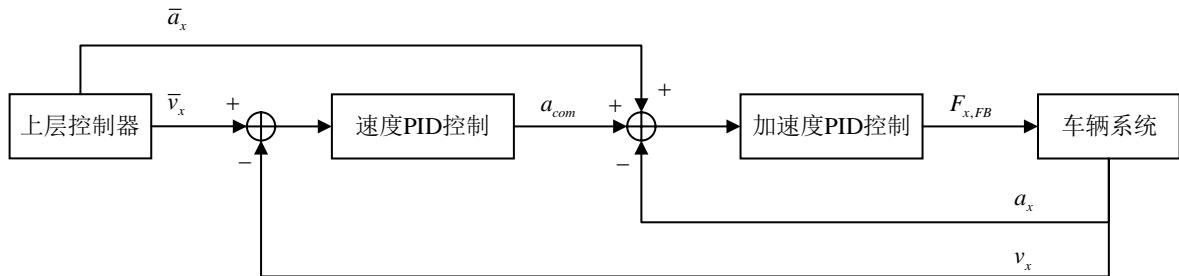


图 4.4 纵向速度和加速度双环 PID 反馈控制结构示意图

纵向加速度补偿量  $a_{com}$  的计算公式为：

$$a_{com}(k) = K_{P,v} e_v(k) + K_{I,v} \sum_{i=0}^k e_v(i) + K_{D,v} [e_v(k) - e_v(k-1)] \quad (4.12)$$

$$e_v(k) = \bar{v}_x(k) - v_x(k) \quad (4.13)$$

式中， $K_{P,v}$ ， $K_{I,v}$  和  $K_{D,v}$  分别为外环速度 PID 控制的比例系数、积分系数和微分系数。

反馈期望纵向驱动力的计算公式为：

$$F_{x,FB}(k) = K_{P,x}e_x(k) + K_{I,x} \sum_{j=0}^k e_x(j) + K_{D,x} [e_x(k) - e_x(k-1)] \quad (4.14)$$

$$e_x(k) = \bar{a}_x(k) + a_{com}(k) - a_x(k) \quad (4.15)$$

式中,  $K_{P,x}$ ,  $K_{I,x}$  和  $K_{D,x}$  分别为内环加速度 PID 控制的比例系数、积分系数和微分系数。

## 4.2 基于误差模型在线估计的前馈修正策略

加速度跟踪控制算法中的前馈项均是基于固定的动力学关系式计算得到, 与反馈环节相结合可以提高控制系统的响应速度。而前馈项计算所采用的动力学关系式通常是经过一系列简化, 与真实情况存在一定的偏差, 因此本节将针对跟踪偏差值进行参数估计并对前馈项做出修正, 从而进一步提升加速度跟踪控制系统的响应速度和跟踪精度。

### 4.2.1 基于归一化最小均方算法的纵向前馈修正

归一化最小均方算法(NLMS)的原理于 2.3.2 节有过介绍, 本节将直接介绍 NLMS 算法在纵向前馈修正项计算方面的应用方法。

如图 4.5 所示为纵向加速度跟踪的前馈修正项计算流程图, 由图可以看出, 误差模型的输入信号  $x(k)$  为每一时刻纵向加速度跟踪控制系统输出的期望纵向力  $F_x$ , 它由三部分组成, 分别为反馈期望纵向力  $F_{x,FB}$ , 前馈期望纵向力  $F_{x,FF}$  以及前馈修正项  $F_{FFLearned}$ 。其中, 前馈修正项  $F_{FFLearned}$  即需要 NLMS 算法在每一时刻循环更新计算, 用于实时调整控制系统输出的期望纵向力  $F_x$ 。纵向误差模型在  $k$  时刻的期望响应  $d(k)$ , 也可称为真实误差值, 是当前纵向力测量值  $ma_x(k)$  和期望纵向力  $F_x(k)$  的偏差, 同时纵向跟踪误差模型也可近似为与输入信号相关的线性回归模型, 其权重系数为  $w(k)$ , 对误差模型期望响应的估计值为  $y(k)$ 。在计算得到  $d(k)$  和  $y(k)$  后即可计算得到估计误差  $e(k)$ , 由估计误差  $e(k)$  和输入信号  $x(k)$  即可对权重系数进行更新计算得到下一时刻的误差模型权重系数  $w(k+1)$ , 并根据回归模型  $y(k) = w^T(k)x(k)$  进一步计算得到下一时刻的误差估计值作为期望纵向力的前馈补偿项  $F_{FFLearned}$ , 从而结束当前时刻的循环, 进入下一时刻的循环。

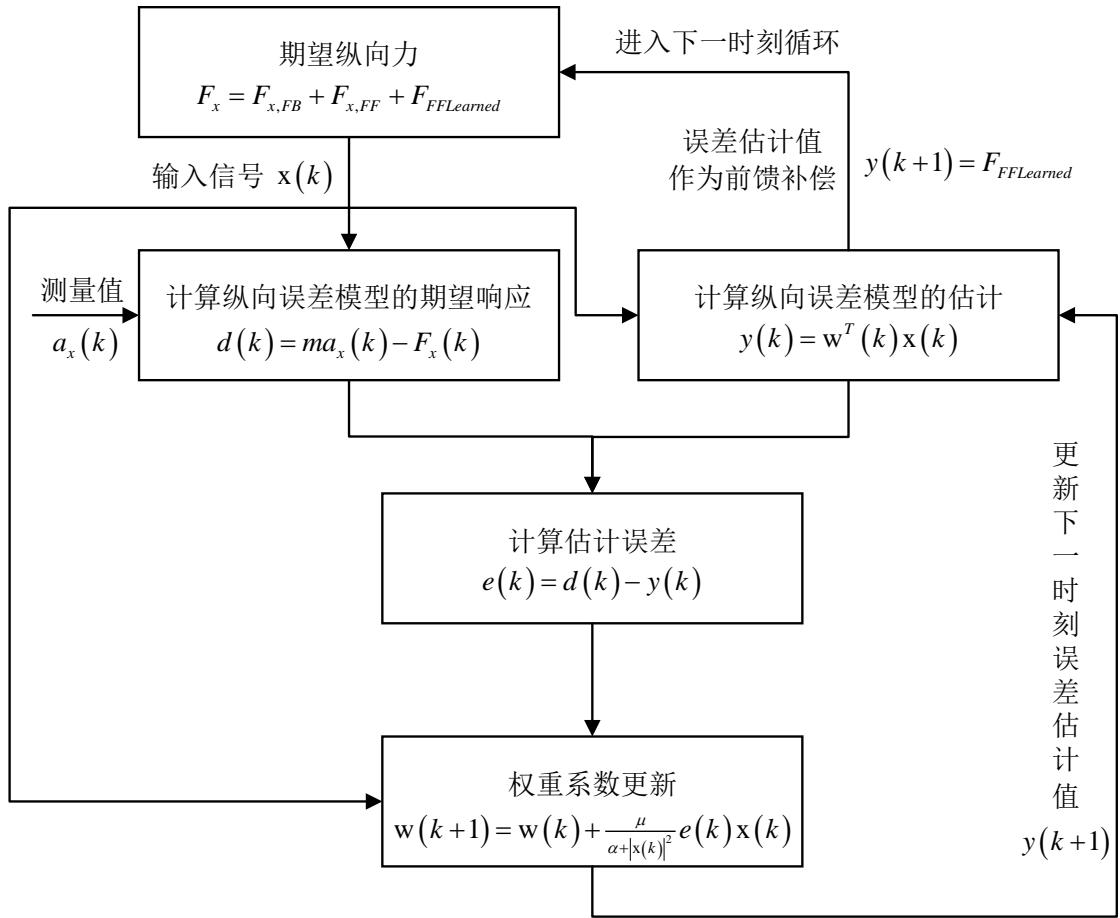


图 4.5 归一化最小均方算法计算前馈补偿项流程图

#### 4.2.2 基于递归最小二乘算法的横向前馈修正

智能车辆轨迹跟踪横向控制相比于纵向控制复杂度更高，所需要应对的外部环境信息也较为复杂，因此对于横向加速度跟踪控制的误差模型估计选择采用递归最小二乘法（Recursive Least Squares, RLS）。RLS 算法和 NLMS 算法都是经典的自适应滤波算法，RLS 算法是一种基于最小二乘法的自适应滤波算法，通过最小化实际输出与期望输出之间的均方误差来更新滤波器的权重，它使用矩阵计算方法进行权重更新，并保持对过去观测数据的所有信息进行追踪。而 NLMS 算法则是基于梯度下降法，通过不断调整权重以减小实际输出与期望输出之间的误差来更新滤波器的权重。它使用单个样本数据进行权重更新，并只保留有限的历史信息。RLS 算法由于利用了全部过去观测数据的信息，对于快速变化的系统响应和噪声环境下的自适应性能较好。NLMS 算法由于只利用有限历史信息和单个样本数据进行权重更新，对于慢速变化的系统响应和较稳定环境下的自适应性能较好。因此，本文选择在横向加速度跟踪误差模型估

计时采用 RLS 算法，在纵向加速度跟踪误差模型估计时采用 NLMS 算法。

如图 4.6 所示，RLS 算法的计算流程为：

步骤一：初始化。设置初始滤波器权重向量  $\omega(0)$ ，以及初始协方差矩阵  $P(0)$ 。协方差矩阵  $P(0)$  通常初始化为一个乘以较小常量的单位矩阵，表示对初始权重没有太多先验信息。

步骤二：接收输入信号并预测输出。输入观测信号  $x(n)$  ( $n$  代表当前时间步)，计算滤波器的预测输出  $y(n) = \omega^T(n)x(n)$ 。

步骤三：计算估计误差。根据期望输出  $d(n)$  和预测输出  $y(n)$ ，计算估计误差  $e(n) = d(n) - y(n)$ 。

步骤四：计算增益向量  $K(n)$ 。利用输入信号  $x(n)$  和上一时刻的协方差矩阵  $P(n-1)$ ，计算增益向量  $K(n) = P(n-1)x(n)/[\lambda + x^T(n)P(n-1)x(n)]$ ，其中  $0 < \lambda \leq 1$  为遗忘因子。

步骤五：更新协方差矩阵  $P(n)$ 。利用递推公式更新协方差矩阵， $P(n) = \lambda^{-1}[P(n-1) - K(n)x^T(n)P(n-1)]$ 。

步骤六：更新权重向量  $\omega(n)$ 。利用增益向量  $K(n)$  和预测误差  $e(n)$ ，更新权重向量  $\omega(n) = \omega(n-1) + K(n)e(n)$ 。

参数递推估计，每取得一次新的观测数据后，就在前次估计结果的基础上，利用新引入的观测数据对前次估计的结果，根据递推算法进行修正，减少估计误差，从而递推地得出新的参数估计值。这样，随着新观测数据的逐次引入，不断地进行参数估计，更新误差模型直至收敛。

横向加速度跟踪控制的误差模型相比于纵向的误差模型要更为复杂，难以用线性模型近似，为线性不可分问题。线性不可分问题是指在某个特征空间中，无法通过一条直线或一个超平面将不同类别的数据完全分开的问题。在这种情况下，无法使用线性分类器（如线性回归）来对数据进行准确分类。解决线性不可分问题的一种方法是使用非线性转换技术，将原始特征空间映射到一个更高维的特征空间中，使得数据变得线性可分。常用的非线性转换技术包括多项式特征、高斯核函数、径向基函数等。通过将数据映射到高维空间，线性不可分的问题可以转化为在新的特征空间中线性可分的问题，进而使用线性分类器进行分类。针对横向加速度跟踪误差模型的估计问题，选择采用构造关于横向加速度和纵向速度的径向基函数，将其作为 RLS 算法的输入信号。

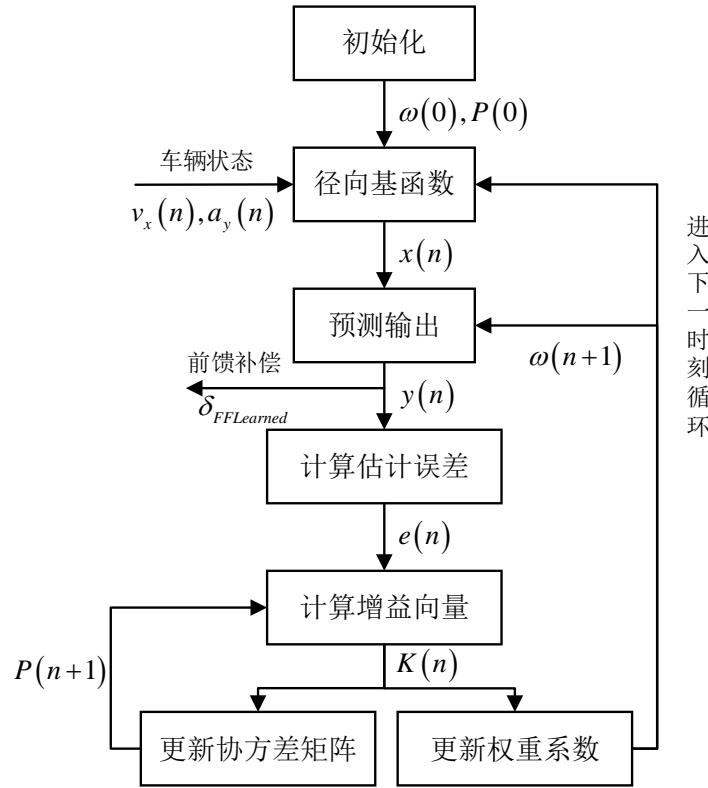


图 4.6 RLS 算法计算横向前馈补偿流程图

### 4.3 基于模型参考自适应控制的执行器延迟优化策略

在横、纵向加速度跟踪控制器设计的过程中，我们通常都假设车辆的执行机构，如转向机构、驱动机构以及制动机构等，可以完美地执行控制算法所计算出的控制指令。但是现实中车辆执行器是存在跟踪延迟的，执行器的延迟问题会导致控制的超调和震荡问题，因此本文决定采用模型参考自适应控制方法对执行器进行辅助调节，以解决延迟问题。

#### 4.3.1 模型参考自适应控制原理

自适应控制的核心思想是在面对系统内在或外在的未知变动，能以一种自我调整的机制来确保系统始终维持在预定的理想状态。进一步而言，自适应控制系统实时运作的过程中，需要持续性地检测和获取系统的输入、当前状态、输出乃至性能参数的动态变化。随之，根据这些实时获取的信息，遵循合理的设计逻辑，做出相应的控制决策，用以修改控制器的结构、参数乃至控制力度，最终目标是实现控制效果的优化或近乎最优。而模型参考自适应控制（Model Reference Adaptive Control, MRAC）就是自适应控制中一种较为成熟的方法。

在模型参考自适应控制中，我们需要设计一个期望输出模型，即参考模型。该模型描述了我们希望系统实现的期望响应。然后，我们将系统的输出与参考模型的输出进行比较，得到误差信号。模型参考自适应控制系统的基本结构如图所示，其主要由两部分组成：其一为内环，主要由控制器与被控对象组成，这一环节可称之为可调系统；其二为外环，则由参考模型与自适应机制构成。实际上，这种结构是在传统反馈控制回路的基础上，巧妙地增加了一个参考模型和控制器参数的自动调节回路，从而赋予了系统更高的灵活性和适应性。

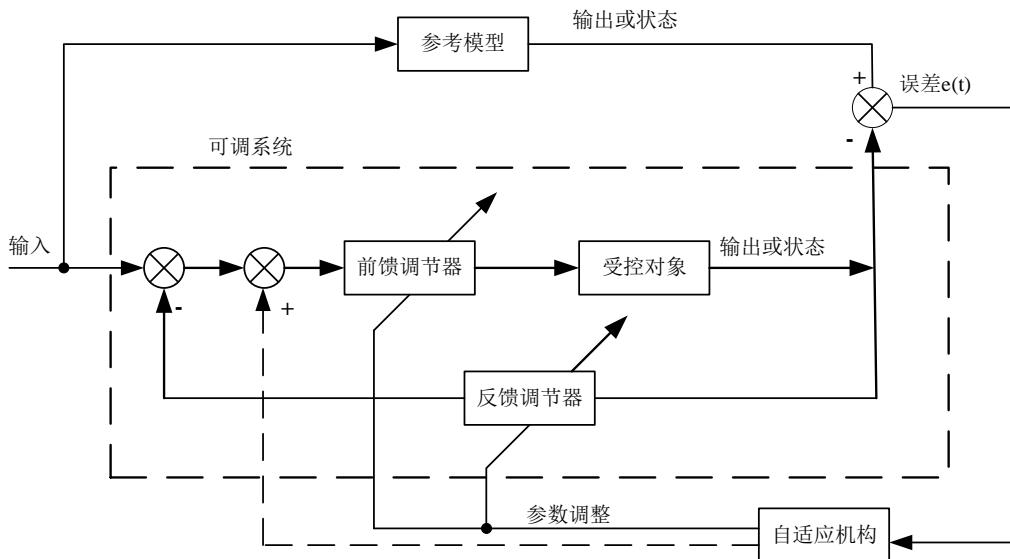


图 4.7 模型参考自适应控制系统基本结构

在模型参考自适应控制系统中，参考模型的输出扮演着极其重要的角色，它充当了期望动态性能指标的基准。输入的目标信号会同时作用于实际模型和参考模型，将系统的输出与参考模型的输出进行比较，得到误差信号。这一误差信号将依据特定的自适应变化律，被用以调整控制器的参数，最终引导受控制对象的输出尽可能地趋同于参考模型的输出。

由上述原理可知，模型参考自适应控制的关键是确定自适应变化律，自适应变化律的优劣将决定实际模型的输出是否能够很好的贴近参考模型的输出。目前，该领域的研究主要聚焦于两大方向：其一为参数最优化方法，即通过最小化系统的某个性能指标函数，如使误差信号的平方和达到最小，即通过最优化方法计算自适应控制器的最优调节参数。其二为基于稳定性理论的方法，其基本思想是保证控制器参数自适应

调节过程是稳定的，如基于 Lyapunov 稳定性理论的设计方法和基于 Popov 超稳定理论的方法。

#### 4.3.2 模型参考自适应控制器设计

模型参考自适应控制方法在智能车辆控制中，通常需要与上层控制算法联合起来使用，起到辅助调节的作用。在本文中，上层控制算法即为加速度跟踪控制器，所输出的控制量即为转向角和纵向力，而模型参考自适应控制器的作用则是使实际模型的输出跟随参考模型的输出，而参考模型的截止频率和阻尼系数等参数通常由设计者给定，从而使得实际模型对期望转向角或期望纵向力的响应接近设计者所期望得到的参考模型的响应。该方法的实际意义是为了让同一套轨迹跟踪控制算法在面对不同的执行机构时，不受硬件参数差异的影响，使实际模型的响应接近参考模型进行响应，同时，通过对参考模型参数的合理设计，削弱执行机构对控制效果的延迟影响，如超调、震荡现象。由于原理相似，下文将以前轮转角为例，推导模型参考自适应控制的应用原理。

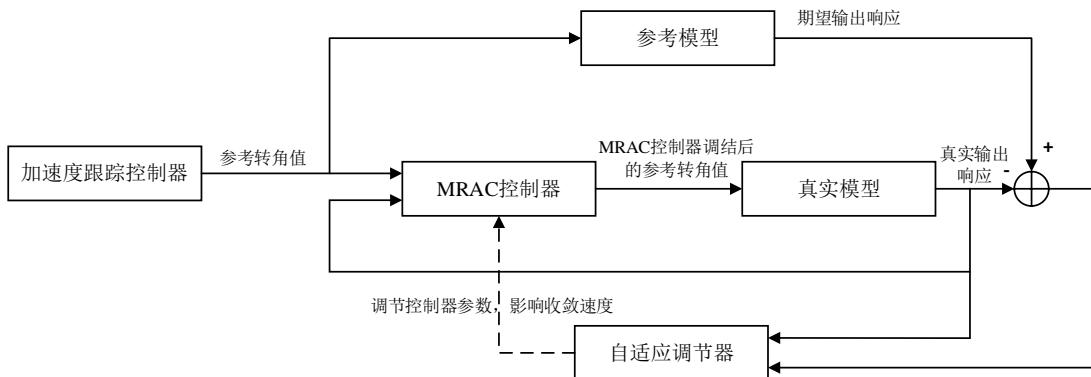


图 4.8 模型参考自适应控制算法的结构框架图

对于转向角信号，可以使用一阶系统或二阶系统进行建模，为模拟更加真实的延迟效应，本文采用二阶系统建模，状态量包括两个，一个是转向角  $\delta$ ，一个是转向角的变化率  $\dot{\delta}$ ，假设二阶转向角参考模型的微分方程为：

$$\ddot{\delta}_m + 2\zeta_m \omega_m \dot{\delta}_m + \omega_m^2 \delta_m = \omega_m^2 r \quad (4.16)$$

式中， $r$  表示加速度跟踪控制算法输出的期望转向角， $\delta_m$  表示参考模型对期望转向角的响应输出， $\zeta_m$  和  $\omega_m$  分别表示二阶转向角参考模型的阻尼系数和截止频率。

假设二阶转向角实际模型的微分方程为：

$$\ddot{\delta}_p + 2\zeta_p \omega_p \dot{\delta}_p + \omega_p^2 \delta_p = \omega_p^2 u \quad (4.17)$$

式中,  $u$  表示经过模型参考自适应控制器调整的转向角输出值,  $\delta_p$  表示实际模型对期望转向角的响应输出,  $\zeta_p$  和  $\omega_p$  分别表示二阶转向角实际模型的阻尼系数和截止频率。

为便于后续推导, 现将参考模型和实际模型的微分方程转换为状态空间方程, 状态量包括转向角和转向角变化率, 定义参考模型的状态量为  $x_m = [\delta_m \quad \dot{\delta}_m]^T$ , 真实模型的状态量为  $x_p = [\delta_p \quad \dot{\delta}_p]^T$ 。

参考模型的状态空间方程为:

$$\dot{x}_m = A_m x_m + B_m r \quad (4.18)$$

$$A_m = \begin{bmatrix} 0 & 1 \\ -\omega_m^2 & -2\zeta_m \omega_m^2 \end{bmatrix}, B_m = \begin{bmatrix} 0 \\ \omega_m^2 \end{bmatrix} \quad (4.19)$$

真实模型的状态空间方程为:

$$\dot{x}_p = A_p x_p + B_p u \quad (4.20)$$

$$A_p = \begin{bmatrix} 0 & 1 \\ -\omega_p^2 & -2\zeta_p \omega_p^2 \end{bmatrix}, B_p = \begin{bmatrix} 0 \\ \omega_p^2 \end{bmatrix} \quad (4.21)$$

如图 4.8 所示为模型参考自适应控制算法的结构框架图, 由图易知, MRAC 控制器的自适应控制律为:

$$u = \theta_x^T x_p + \theta_r^T r \quad (4.22)$$

式中,  $\theta_x$  和  $\theta_r$  分别为实际模型状态量  $x_p$  和期望转向角  $r$  对应的自适应控制律系数, 即是 MRAC 控制器需要确定的两个重要参数, 分别称为状态反馈向量和输入反馈系数。

真实模型的响应特性可以完全跟随参考模型是 MRAC 控制器想要达到的理想状态, 则意味着当两模型当前状态量相同的时候, 即  $x_m = x_p$ , 我们希望两模型的状态量变化率也相同, 即  $\dot{x}_m = \dot{x}_p$ 。在此假设符合前述 MRAC 控制器理想状态的自适应控制律为  $u = \theta_x^{*T} x_p + \theta_r^{*T} r$ , 将该控制律代入真实模型的状态空间方程可以得到:

$$\dot{x}_p = (A_p + B_p \theta_x^{*T}) x_p + B_p \theta_r^{*T} r \quad (4.23)$$

由  $\dot{x}_m = \dot{x}_p$  对比式(4.23)和式(4.18)的系数可以得到:

$$B_p \theta_x^{*T} = A_m - A_p \quad (4.24)$$

$$B_p \theta_r^{*T} = B_m \quad (4.25)$$

由式(4.24)和式(4.25)可以计算得到理想状态下的状态反馈向量  $\theta_x^*$  和输入反馈系数  $\theta_r^*$ ，但是由于真实模型是未知的，即  $A_p$  和  $B_p$  都是未知的，因此上述的数值分析方法实际上是不可用的，需要采用递归迭代的方式去计算，假设实际输入的控制律为：

$$\hat{u} = \hat{\theta}_x^T x_p + \hat{\theta}_r^T r \quad (4.26)$$

将式(4.26)的控制输入代入真实模型的状态方程后可以整理为如下形式：

$$\dot{x}_p = A_m x_p + B_m r + B_p (\tilde{\theta}_x^T x_p + \tilde{\theta}_r^T r) \quad (4.27)$$

将式(4.27)与式(4.18)作差可以构建状态误差系统，其表达式如下：

$$\dot{e} = A_m e + B_p (\tilde{\theta}_x^T x_p + \tilde{\theta}_r^T r) \quad (4.28)$$

式中， $\tilde{\theta}_x^T = \hat{\theta}_x^T - \theta_x^{*T}$ ， $\tilde{\theta}_r^T = \hat{\theta}_r^T - \theta_r^{*T}$ ，误差变量为  $e = x_p - x_m$ 。

随后基于 Lyapunov 稳定性理论的方法进行自适应控制律的设计，选取 Lyapunov 函数的形式为：

$$V(e, \tilde{\theta}) = e^T P e + \gamma_x^{-1} \tilde{\theta}_x^T \tilde{\theta}_x + \gamma_r^{-1} \tilde{\theta}_r^T \tilde{\theta}_r \quad (4.29)$$

式中， $P$  为正定的实对称矩阵， $\gamma_x$  和  $\gamma_r$  为自适应控制律参数向量的收敛率调整系数。

对 Lyapunov 函数求导可得：

$$\begin{aligned} \dot{V} &= e^T P \left( A_m e + B_p (\tilde{\theta}_x^T x_p + \tilde{\theta}_r^T r) \right) + \left( e^T A_m^T + x_p^T \tilde{\theta}_x B_p^T + r^T \tilde{\theta}_r B_p^T \right) P e \\ &\quad + 2\gamma_x^{-1} \tilde{\theta}_x^T \dot{\tilde{\theta}}_x + 2\gamma_r^{-1} \tilde{\theta}_r^T \dot{\tilde{\theta}}_r \end{aligned} \quad (4.30)$$

由矩阵运算易知，存在关系式  $e^T P B_p \tilde{\theta}_x^T x_p = x_p^T \tilde{\theta}_x B_p^T P e = \tilde{\theta}_x^T x_p e^T P b_p$  以及关系式  $e^T P B_p \tilde{\theta}_r^T r = r^T \tilde{\theta}_r B_p^T P e = \tilde{\theta}_r^T r e^T P B_p$ ，因此，选取自适应参数变化率分别为  $\dot{\tilde{\theta}}_x = -\gamma_x x_p^T e^T P B_p$  和  $\dot{\tilde{\theta}}_r = -\gamma_r r e^T P B_p$ ，可以得到：

$$\dot{V} = e^T (P A_m + A_m^T P) e \quad (4.31)$$

由式(4.31)易知，通过对  $P$  矩阵进行合理的设计，可以确保  $P A_m + A_m^T P$  为负定，即确保  $\dot{V} \leq 0$ ，使系统是稳定的。在实际应用过程中可用  $B_m$  近似代替  $B_p$  并且可将  $\theta_x^{*T}$  和  $\theta_r^{*T}$  视为理想状态下的定值，最终确定 MRAC 控制器的自适应参数变化率为：

$$\dot{\tilde{\theta}}_x = -\gamma_x x_p^T e^T P B_m \quad (4.32)$$

$$\dot{\tilde{\theta}}_r = -\gamma_r r e^T P B_m \quad (4.33)$$

在得到自适应参数变化率之后，即可在每一控制时刻迭代计算更新状态反馈向量  $\hat{\theta}_x$  和输入反馈系数  $\hat{\theta}_r$ ，进而通过式(4.26)更新该时刻实际输入的控制律  $\hat{u}$ ，最终经过不断的迭代计算状态反馈向量  $\hat{\theta}_x$  和输入反馈系数  $\hat{\theta}_r$  会收敛到理想状态下的状态反馈向量  $\theta_x^*$  和输入反馈系数  $\hat{\theta}_r$ ，使得实际输入的控制律收敛到理想状态下的控制律，进而保证真实模型的响应输出与参考模型保持一致。

#### 4.4 本章小结

本章针对上层轨迹跟踪鲁棒控制器输出的期望加速度信号，设计了基于前馈和反馈结合的下层加速度跟踪策略，输出对车辆的控制指令，分别为前轮转角和纵向力；分别推导了横向加速度跟踪策略和纵向加速度跟踪策略，其中横向加速度前馈由阿克曼转向原理推导得到，反馈由 PI 控制实现，并添加了抗积分饱和项；纵向加速度前馈则由车辆的行驶方程推导得到，反馈由双环 PID 控制实现，其中外环为速度控制，内环为加速度控制；针对实际测量得到的跟踪偏差问题，设计了基于误差模型在线估计的方法对前馈项进行修正，其中根据各自控制特点，纵向部分采用 NLMS 算法，横向部分采用 RLS 算法；针对执行器延迟问题，设计了基于模型参考自适应控制的延迟处理方法，使得实际模型的输出尽可能贴近参考模型输出。

## 第5章 智能车辆轨迹跟踪控制仿真实验

在智能汽车领域，仿真实验具有极其重要的作用。智能汽车的开发和测试过程中，需要面对大量的复杂场景和驾驶情况，通过传统的实地测试显然是耗时耗力且成本高昂的。而仿真实验则提供了一种高效、安全和可控的手段，帮助工程师评估和改善智能汽车的性能和安全性。本文基于 MATLAB/Simulink 和 Carsim 搭建了联合仿真平台，并针对前文提出的控制策略以及优化算法进行性能的测试和分析。

### 5.1 智能车辆轨迹跟踪控制仿真平台简介

#### 5.1.1 基于 Simulink 和 Carsim 的联合仿真平台简介

Carsim 是一款广泛应用于汽车工程领域的仿真软件，它能够模拟车辆在各种道路条件下的运动行为和性能。该软件由 Mechanical Simulation Corporation 开发，已经成为全球汽车制造商、研究机构和大学的重要工具之一。Carsim 以多体动力学为基础，可以准确地模拟车辆在加速、制动、转向等各种驾驶情况下的运动特性。通过对车辆的悬挂系统、车轮和轮胎特性等进行建模，Carsim 可以提供精确的车辆动力学分析结果，包括车辆的加速度、速度、横向加速度、横摆角等。除了车辆本身的动力学特性，Carsim 还可以模拟不同的道路条件，如平坦路面、不均匀路面、曲线道路等，以及不同的驾驶操作，如转向、刹车等。这使得工程师可以根据实际需要进行车辆性能评估、悬挂系统调校、动力学优化等。Carsim 还具有强大的参数化能力，用户可以根据需要对车辆的各种参数进行调整，如车辆质量、惯性特性、悬挂系统刚度等，从而对车辆的性能进行优化和改进。此外，Carsim 还支持与其他软件的集成，如 MATLAB/Simulink，使得用户可以更加灵活地进行复杂系统的建模和仿真。

如图 5.1 所示为本文所搭建的基于 MATLAB/Simulink 和 Carsim 的联合仿真平台，在 MATLAB/Simulink 中搭建了轨迹跟踪控制器，所有功能模块均由 Simulink 基本模块与 S 函数编写实现，假设已从轨迹规划模块获取了参考轨迹信息，本文将主要针对控制算法模块进行验证分析。Simulink 中的参考轨迹信息预处理模块接收参考轨迹信息以及车辆状态信息，用于计算匹配点坐标以及预测范围内的参考轨迹插值信息，计算完成后将相应信息传输给上层轨迹跟踪鲁棒控制器，计算出期望的横、纵向加速度信号，输出给下层加速度跟踪控制器，计算出车辆的控制指令即前轮转角和纵向力，其中纵向力经标定查表的方式转化为油门开度和制动压力。控制指令将由 Simulink 发

送至 Carsim 进行车辆运动模拟，完成一个控制循环。

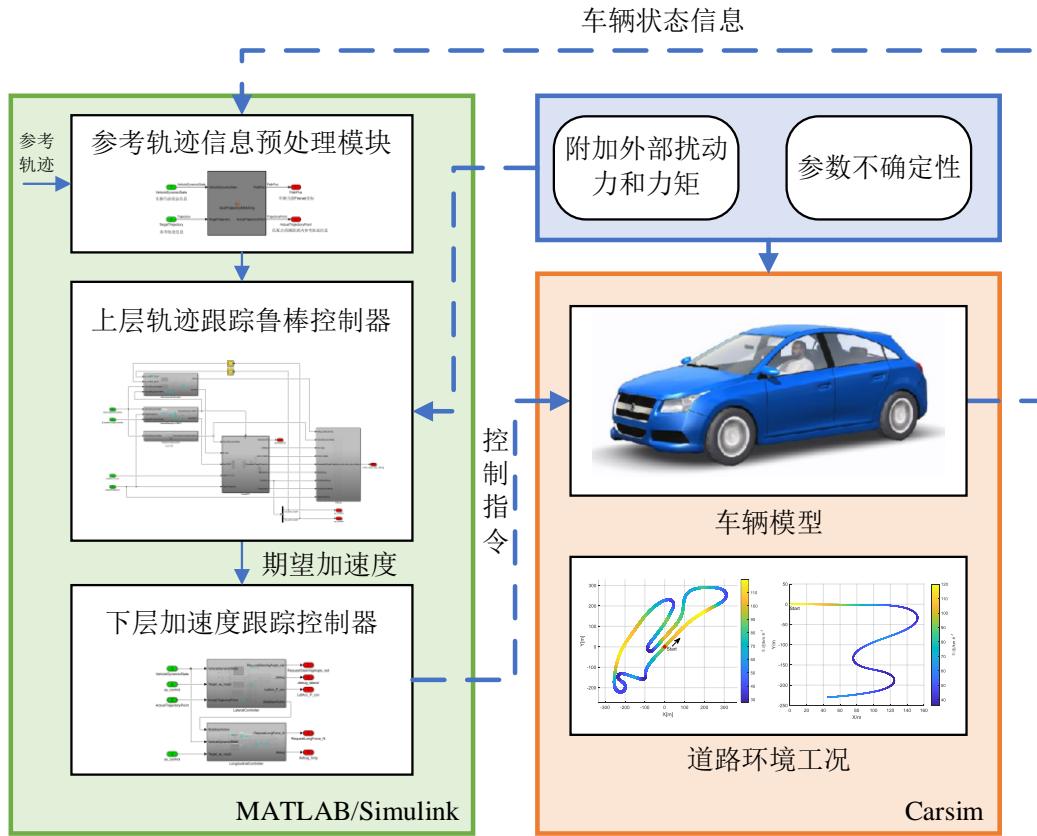


图 5.1 MATLAB/Simulink 和 Carsim 联合仿真平台架构

### 5.1.2 仿真实验技术参数及行驶环境设置

本文选用 Carsim 中的 C-Class/Hatchback 车型为被控车辆，并采用传动比为 24 的前轮转向系统，轮胎型号均为 215/55 R17，仿真车辆的部分技术参数如表 5-1 所示。

表 5-1 车辆技术参数

参数名称	符号	数值
整车质量	$m$	1270 kg
横摆转动惯量	$I_z$	$1536.7 \text{ kg} \cdot \text{m}^2$
质心距前轴距离	$L_f$	1.215 m
质心距后轴距离	$L_r$	1.705 m
轮距	$B$	1.675m
车身宽度	$W$	1.79 m
车身长度	$L$	3.80 m

依据前文假设条件, 本文设置智能汽车的行驶道路宽度为 3.8 m, 并设置道路的路面坡角、路面倾斜角等参数均为 0。此外, 将所有行驶道路均设置为干燥的沥青路面, 其路面附着系数设置为  $\mu=0.8$ 。

## 5. 2 Simulink 控制器仿真模型搭建

### 5. 2. 1 参考轨迹信息预处理模块

在 Simulink 中搭建参考轨迹信息预处理模块如图 5.2 所示, 该模块接收车辆当前状态信息和参考轨迹信息, 首先通过遍历参考轨迹点寻找距当前车辆位置最近的点作为匹配点, 利用 Cartesian 坐标系与 Frenet 坐标系转换关系, 得到车辆当前位置的 Frenet 坐标, 包括行驶距离  $s$ , 横向偏差  $d$  以及航向偏差  $\varphi$ 。在确定匹配点后, 根据预测时域长度确定当前匹配点以及预测范围内的参考轨迹点信息, 最终输出给上层 Tube MPC 轨迹跟踪鲁棒控制器。

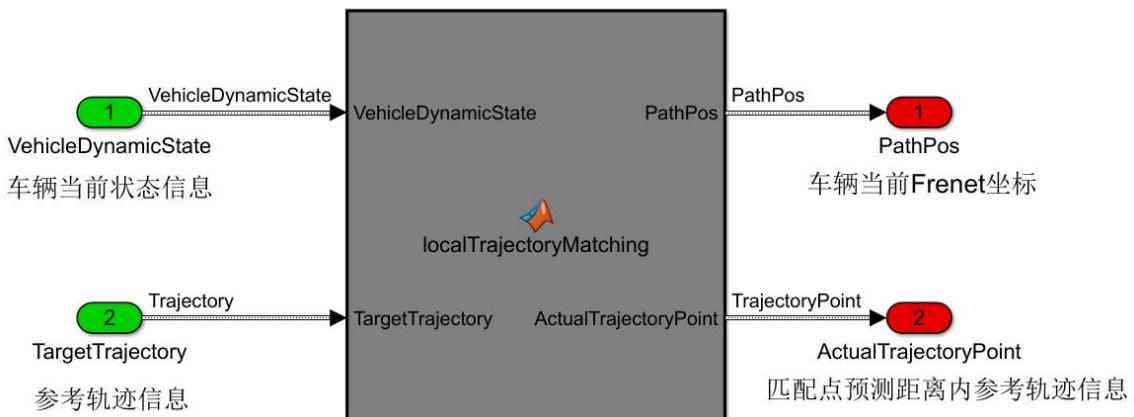


图 5.2 参考轨迹信息预处理模块 Simulink 模型结构示意图

### 5. 2. 2 上层 Tube MPC 轨迹跟踪鲁棒控制器

在 Simulink 中搭建上层 Tube MPC 轨迹跟踪鲁棒控制器模型如图 5.3 所示, 该模块主要接收经参考轨迹预处理模块传递来的相应信息, 进行上层控制指令信息的计算, 并输出给下层控制器。如图 5.3 所示, 上层 Tube MPC 轨迹跟踪鲁棒控制器的实现主要由两部分构成, 分别为图中红框标注的 1 号部分以及 2 号部分, 其中 1 号部分为 2.3 节所介绍的扰动边界在线辨识模块, 也可称为扰动信号估计模块, 2 号部分为 Tube MPC 算法模块。不确定性参数辨识模块主要接收车辆实际的横、纵向加速度状态信息以及 Tube MPC 控制算法求解得到的期望横、纵向加速度信息, 并将两者的差

值分别作为车辆横、纵向所遭受的外部扰动信号，通过 2.3 节介绍的 RUMI 扰动边界辨识方法，进行横、纵向动力学扰动信号的边界在线辨识，并最终输出辨识结果给 Tube MPC 控制算法模块，用于 3.3.2 节中所介绍的扰动集形状矩阵  $\tilde{M}$  构建。Tube MPC 控制算法模块主要接收车辆当前状态信息、规划参考轨迹信息以及扰动信号边界信息等，最终经过 Tube MPC 控制算法求解输出得到图中 3 号部分所示的期望横、纵向加速度信息，并传递给下层加速度跟踪控制器控制器进行跟踪

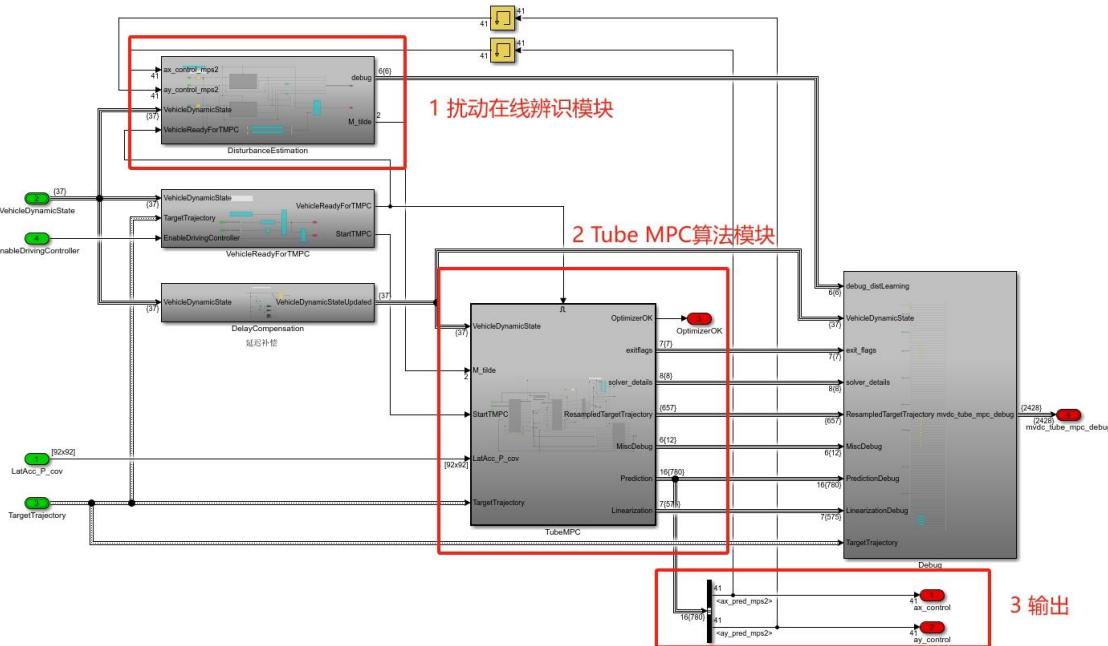


图 5.3 上层轨迹跟踪控制器 Simulink 模型结构示意图

Tube MPC 控制算法模块具体构成如图 5.4 所示，主要由 4 部分构成，其中第 1 部分为线性化预处理模块，主要是为后续优化问题构建做相应的预处理，其中包括对当前路点进行匹配，基于当前点对预测范围内的参考轨迹进行插值采样，由扰动信号边界信息计算预测时域内每个时刻的 Tube 不变集形状矩阵  $M_k$  以及对参考轨迹速度和曲率信息进行线性化处理等等功能。第 2 部分为优化问题及约束条件构建模块，主要用于构建优化问题和约束条件矩阵，接收经过预处理得到的预测时域内车辆各状态信息和参考轨迹的插值目标点信息，并进行相应的计算将其构建为所需的二次规划问题，并将相应的矩阵信息传递给后续的求解器模块进行求解。第 3 部分即为 OSQP 求解器模块，将其转换为 S-Function 形式以便在 Simulink 中进行调用，使用 OSQP 求解即可对二次规划问题进行求解，得到最优的控制输入序列。第 4 部分为状态预测模块，用于预测信息推导，该模块接收求解器输出的最优控制序列信息和预处理模块输出的

车辆当前状态信息以及参考轨迹插值信息，进行预测时域内各状态量信息的推导，其中包括车辆的位置信息、速度信息以及加速度信息等等，并将其中的期望横、纵向加速度信息传递给下层加速度跟踪控制器用以进行后续的控制工作。

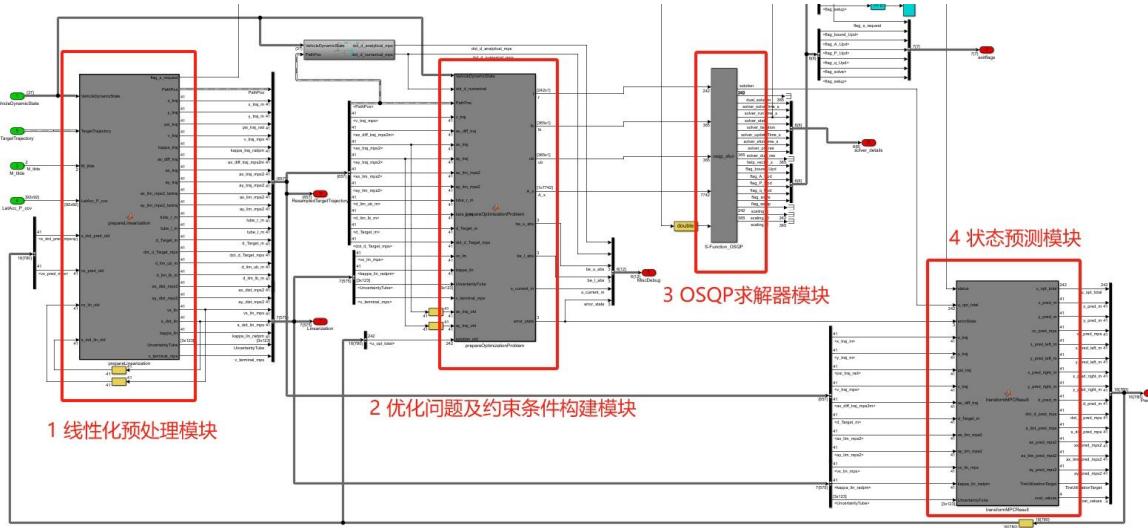


图 5.4 Tube MPC 控制算法模块 Simulink 模型结构示意图

### 5.2.3 下层前馈反馈结合加速度跟踪控制器

在 Simulink 中搭建基于前馈和反馈结合的下层加速度跟踪控制器如图所示，该模块的主要作用为接收上层控制器传递来的期望加速度信号，结合车辆当前状态信息以及参考轨迹点信息，通过前馈控制和反馈控制结合的方法计算出车辆的控制指令，即前轮转角和纵向力，并将相应的控制指令输出给 Carsim，用于操纵模拟车辆跟踪相应的参考轨迹。

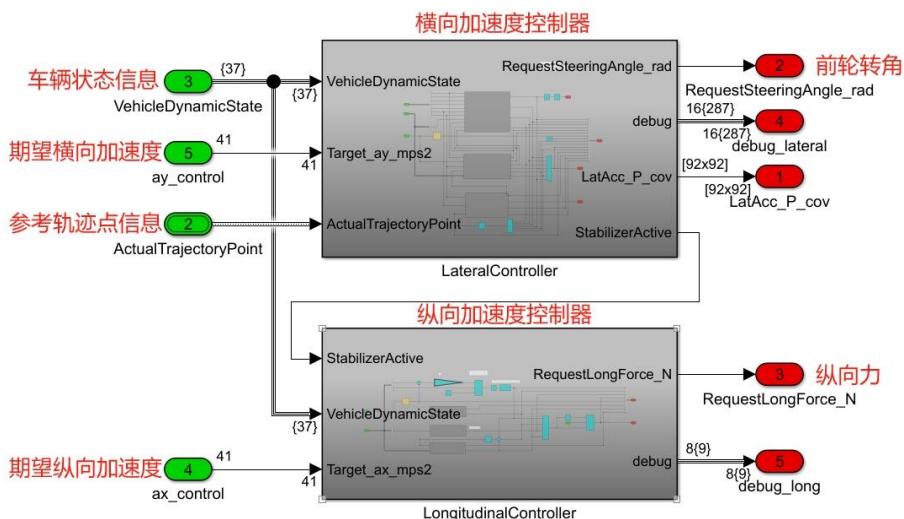


图 5.5 下层加速度跟踪控制器 Simulink 模型结构示意图

## 5.3 环形赛道工况仿真实验

### 5.3.1 仿真实验条件设置

为验证所提出的轨迹跟踪控制策略的鲁棒性和跟踪效果,本文设置轨迹跟踪仿真实验分别采用基于常规 MPC 的轨迹跟踪策略和基于 Tube MPC 的轨迹跟踪鲁棒控制策略去跟踪同一参考轨迹,该参考轨迹如图 5.6 所示为长距离环形赛道工况,赛道总长度为 2313 m,车速变化范围为[28, 120] km/h,其中综合了直道、大曲率弯道、小曲率弯道等多种路况以及加速、减速等情况,并分别设置存在外部干扰和参数不确定性两种实验工况。同时设定由规划层得到的规划信息均保持相同,其中包括规划纵向速度  $v_{x,r}$ 、规划纵向加速度  $a_{x,r}$  以及规划曲率  $\kappa_r$  等,而横向加速度的规划参考值可由  $a_{y,r} = \kappa_r v_{x,r}^2$  得到。针对相同的实验条件用两种算法分别进行一圈的轨迹跟踪实验,用于对比验证两种控制策略的鲁棒性和跟踪性能。

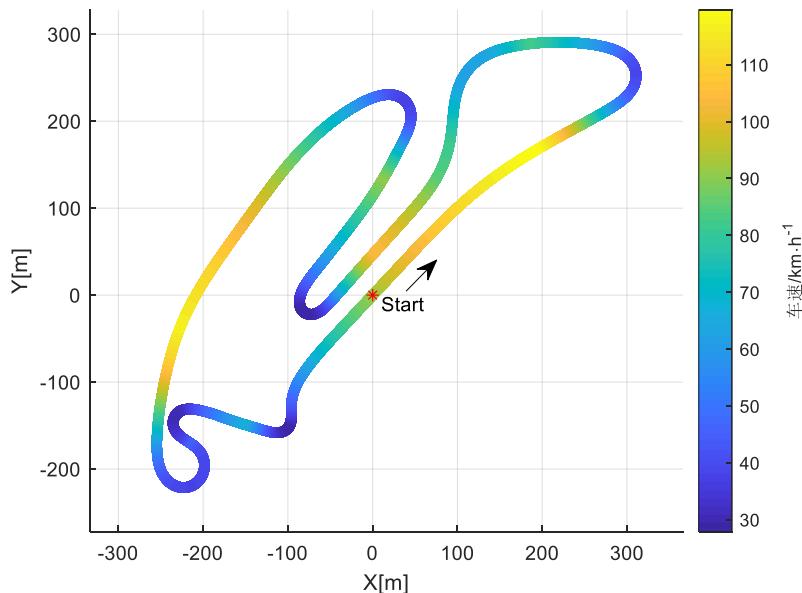


图 5.6 参考轨迹示意图

车辆的初始速度设定为 25 m/s,控制系统的离散时间步长设定为 40ms,同时这也是 Tube MPC 算法的执行频率。预测时域设定为 1.6s,即预测时域包括 40 步。根据道路宽度、路面附着系数等信息,在确保行车安全的前提下,选择将上层 Tube MPC 轨迹跟踪鲁棒控制器的横、纵向加速度极限设定为  $a_{x,\max} = a_{y,\max} = 8 \text{m} \cdot \text{s}^{-2}$ ,横向距离偏差极限为  $d_{\max} = 1 \text{m}$ 。同时,Tube MPC 控制器优化问题中代价函数相应的权重系数矩阵分别设定为,

$$Q = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 40 \end{bmatrix}, R = \begin{bmatrix} 10 & 0 \\ 0 & 5 \end{bmatrix} \quad (5.1)$$

除此之外，横、纵向加速度正则项的权重系数设定为  $\gamma_x = 0.1$  和  $\gamma_y = 200$ ，松弛变量的权重系数分别设定为  $\rho_1 = 1000$  和  $\rho_2 = 100$ 。

### 5.3.2 附加强外部干扰仿真实验

本文对侧向风与路面附着系数不确定性等典型外部环境干扰进行建模分析，确定其对轨迹跟踪控制系统的影响，以进一步确定实验条件。

首先是侧向风的扰动分析，在智能车辆高速行驶时，侧向风会使车辆侧向力发生变化，使车辆偏离参考轨迹，影响乘坐舒适性与车辆安全性。而侧向风对车辆的影响直观体现为其对车辆横向力及横摆力矩的影响，计算公式如下：

$$F_w = \frac{C_y \rho A_w}{2} V_w^2 \quad (5.2)$$

$$M_w = \frac{C_M \rho A_w l}{2} V_w^2 \quad (5.3)$$

$$\beta_w = \arctan\left(\frac{V_w}{V_x}\right) \quad (5.4)$$

式中， $F_w$  为由侧向风产生的车辆侧向力， $M_w$  为由侧向风产生的车辆横摆力矩， $C_y$  为侧向力系数， $C_M$  为横摆力矩系数， $\rho$  为空气质量密度， $A_w$  为汽车的侧向迎风面积， $V_w$  为侧向风速， $\beta_w$  为侧向风来流侧偏角。其中， $C_y$  和  $C_M$  随  $\beta_w$  增大而增大，且与车辆外部形状有关，横摆力矩的作用方向也受到车辆外部形状的影响。

其次是不确定路面附着系数的扰动分析，与侧向风不同，路面附着系数不确定性对车辆轨迹跟踪系统的扰动是通过使轮胎侧向力模型失配产生的。在路面附着系数确定的情况下，名义路面附着系数  $\mu_0$  近似等于实际路面附着系数  $\mu$ ，因此车辆的名义轮胎侧向力为  $F_{y[f,r]}$  也近似等于实际车辆轮胎侧向力  $F_{real,y[f,r]}$ 。但是若是存在路面附着系数不确定性，名义路面附着系数将存在较大的观测误差，那么实际轮胎侧向力与名义轮胎侧向力也将存在较大的计算误差，该误差为：

$$F_{\mu,y[f,r]} = F_{real,y[f,r]} - F_{y[f,r]} \quad (5.5)$$

在控制系统模型中，由不确定路面附着系数引起的车辆侧向力与横摆力矩的扰动

分别为：

$$F_\mu = 2(F_{\mu,yf} + F_{\mu,yr}) \quad (5.6)$$

$$M_\mu = 2(F_{\mu,yf}a - F_{\mu,yr}b) \quad (5.7)$$

式中， $F_\mu$  和  $M_\mu$  分别表示由路面附着系数不确定性引起的车辆横向力扰动与横摆力扰动。

由上述对典型外部环境扰动的建模分析可知，外部干扰对于车辆的影响体现为对车辆施加横向扰动力和横摆扰动力矩，而在实际的车辆行驶过程中，存在着侧向风和不确定路面附着系数之外的诸多扰动，且难以进行一一建模分析，因此需要通过附加随机扰动的方式对复杂的外部环境扰动进行模拟。

为了验证所提出控制器在强外部干扰条件下轨迹跟踪控制的鲁棒性，本节设计了附加扰动下 Tube MPC 和 MPC 两种控制算法的对比实验，跟踪轨迹如 5.3.1 节所介绍。在 Carsim 中加入了幅值分别为 1000 N 和 1000 Nm 的随机扰动横向力和横摆力矩。横向扰动力和横摆扰动力矩如图 5.7 和图 5.8 所示，在随机扰动下两种控制算法的仿真实验结果及分析如下。

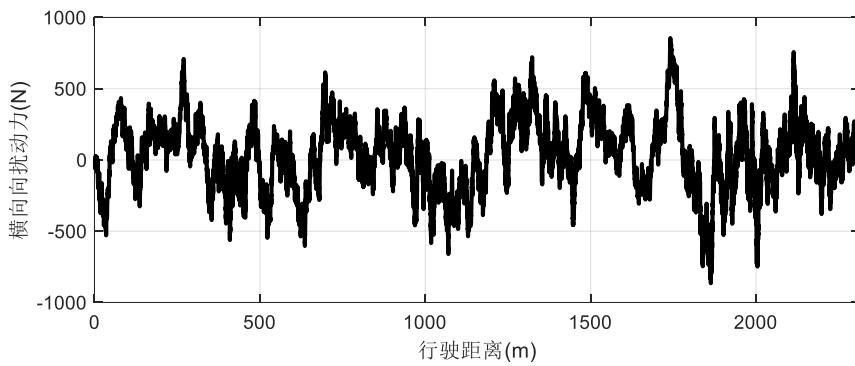


图 5.7 附加横向扰动力

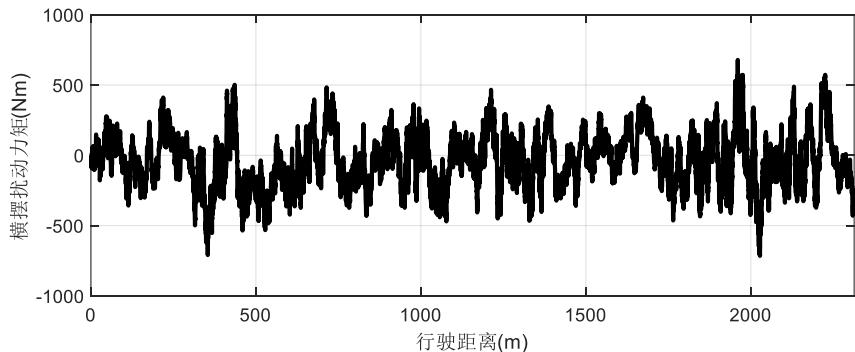


图 5.8 附加横摆扰动力矩

### 5.3.2.1 上层 Tube MPC 轨迹跟踪鲁棒控制策略的对比分析

如图 5.9 所示为两种算法的横向加速度对比图, 其中黑色实线表示由轨迹规划得到的横向加速度参考值, 红色点划线表示不考虑外部干扰的常规 MPC 算法输出得到的横向加速度期望值, 蓝色虚线表示由 Tube MPC 算法输出得到的横向加速度期望值。由图 5.9 可知, 在存在横向外部干扰信号的情况下, 两种算法都能输出与规划参考值相接近的横向加速度期望值, 这说明常规的 MPC 算法与 Tube MPC 算法均具有一定的鲁棒性, 可以抑制一定范围内的外部干扰的影响, 保持轨迹跟踪控制系统的稳定性, 这是因为 MPC 算法本质上也是一种基于前馈和反馈相结合的控制算法, MPC 算法能够依据实际控制效果对控制输入进行调整, 因此常规的 MPC 算法虽没有考虑外部扰动, 但也拥有一定的鲁棒性。但是常规的 MPC 算法输出的横向加速度期望值相比于 Tube MPC 算法存在较为明显的振荡现象, 即 Tube MPC 算法输出的横向加速度期望曲线更加贴近于规划的参考加速度曲线, 且在较大的转向弯角处, 存在着更大的横向加速度峰值。这表明虽然常规的 MPC 算法存在着一定的鲁棒性, 能够维持轨迹跟踪的稳定性, 但其对外部干扰的抑制能力要明显弱于 Tube MPC 算法。

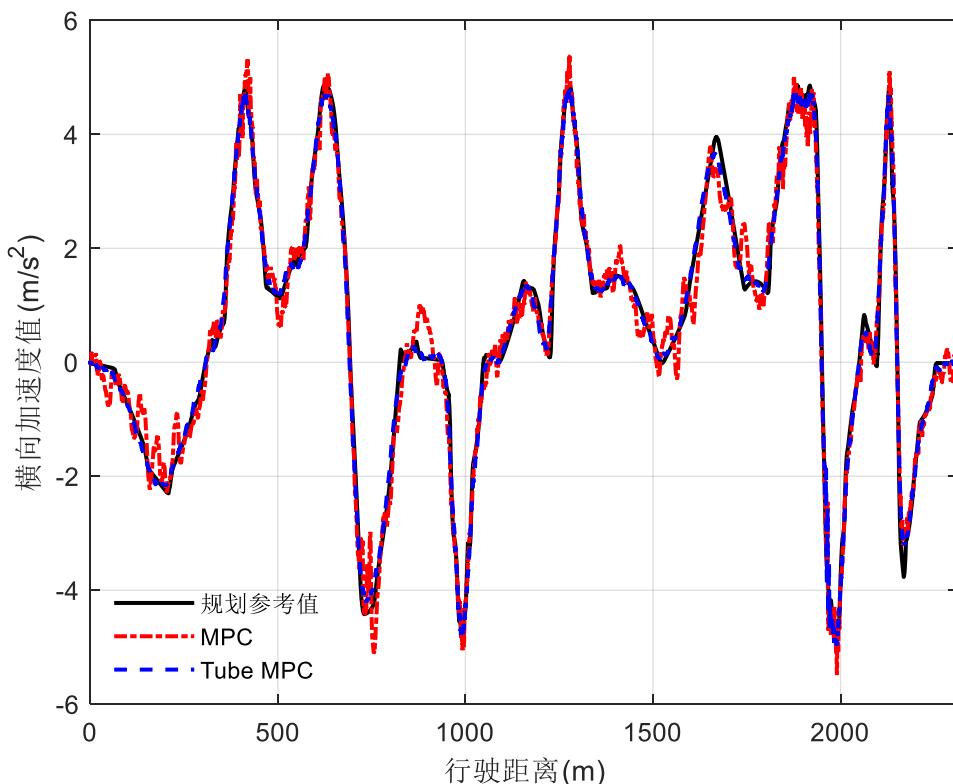


图 5.9 期望横向加速度对比图

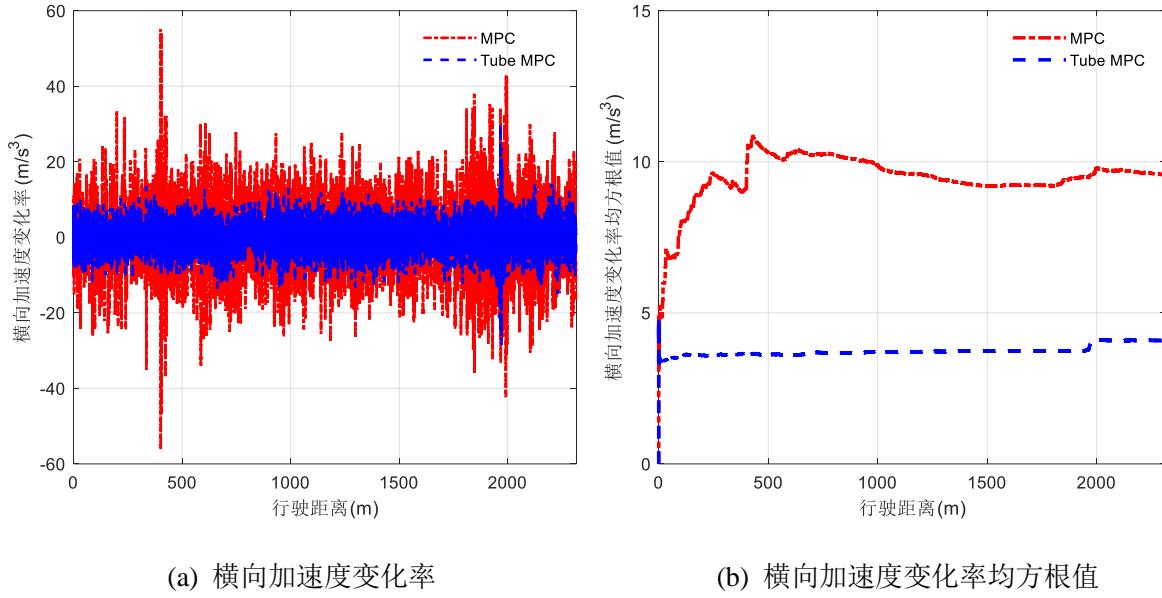


图 5.10 横向加速度变化率及其均方根值对比图

如图 5.10 所示, 左图为两种算法的横向加速度变化率对比图, 右图为横向加速度变化率均方根值对比图。加速度变化率能够反映加速度曲线的平滑程度, 由图 5.10 可知, 常规的 MPC 算法将横向加速度变化率稳定在  $10 \text{ m/s}^3$  左右, 而 Tube MPC 算法则将其降低至  $4 \text{ m/s}^3$  左右, 这表明 Tube MPC 算法输出的横向加速度曲线更加平滑, 对于下层加速度跟踪控制器将更易于跟踪, 有利于避免最终输出的车辆前轮转角出现反复调整振荡的现象。

如图 5.11 所示为两种算法所输出的纵向加速度对比图, 由图可知, 与横向加速度所展示的趋势类似, 在存在附加扰动的情况下, Tube MPC 算法能有效抑制外部干扰影响, 输出与轨迹规划得到的参考加速度相贴近的纵向加速度信号。而常规的 MPC 算法纵向加速度曲线因受到白噪声干扰信号的影响产生了明显的振荡现象, 稳定性较差, 这说明 Tube MPC 算法抑制外部干扰的能力更强。

图 5.12 (a) 为两种算法的纵向加速度变化率对比图, 图 5.12 (b) 为纵向加速度变化率均方根值对比图。由图可知, 常规的 MPC 算法将纵向加速度变化率稳定在  $8 \text{ m/s}^3$  左右, 而 Tube MPC 算法则将其降低至  $4 \text{ m/s}^3$  左右, 与图 5.10 相似, Tube MPC 算法输出了更加平滑且易于跟踪的纵向加速度信号, 有利于提升纵向车速的跟踪精度, 以及保证乘员的舒适性。

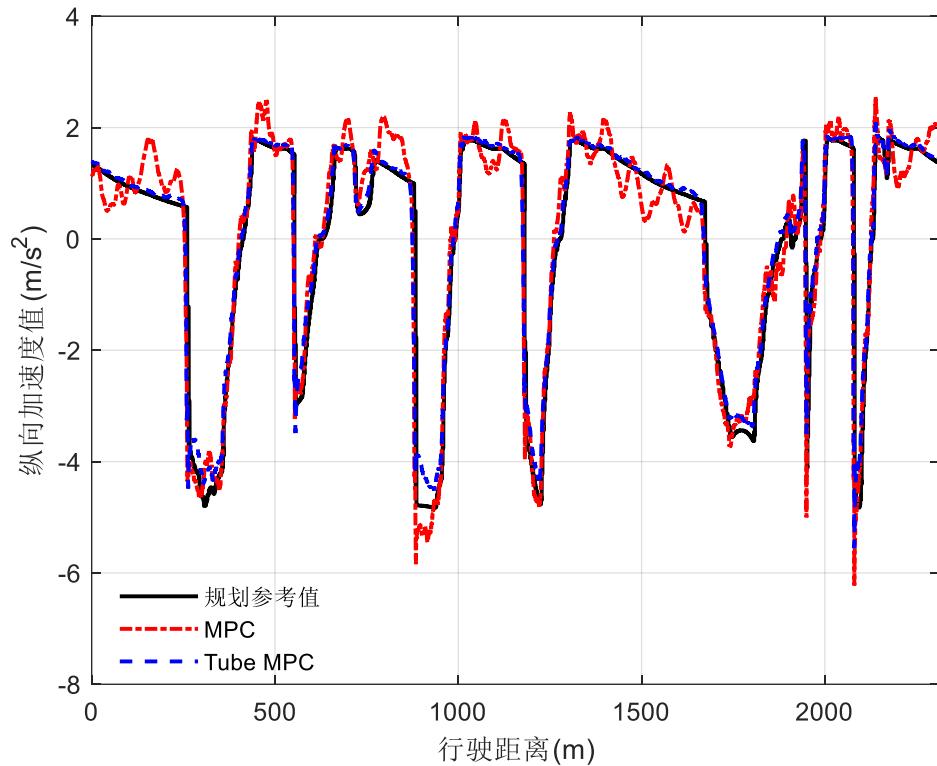
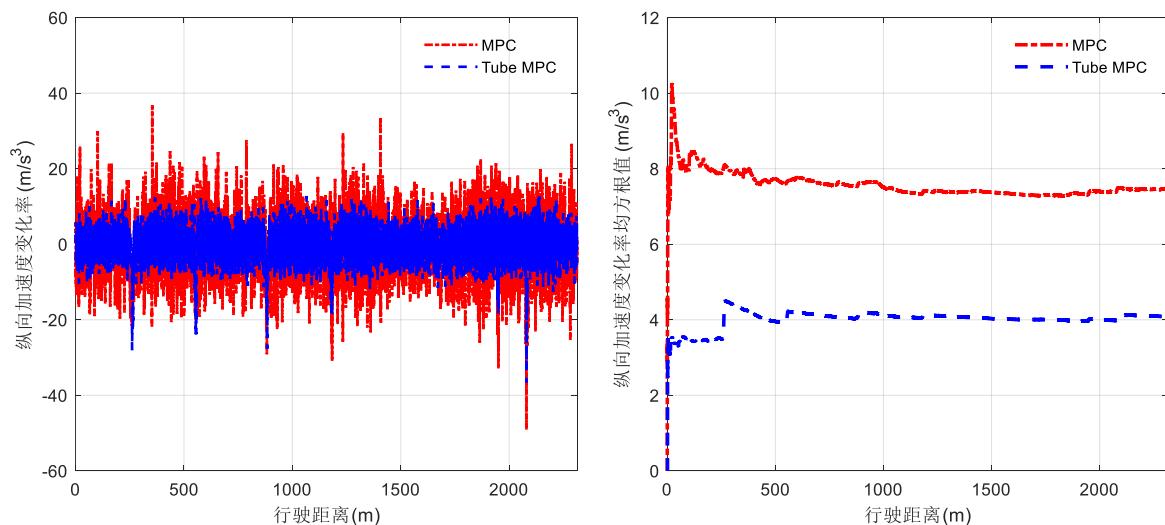


图 5.11 期望纵向加速度对比图



(a) 纵向加速度变化率

(b) 纵向加速度变化率均方根值

图 5.12 纵向加速度变化率及其均方根值对比图

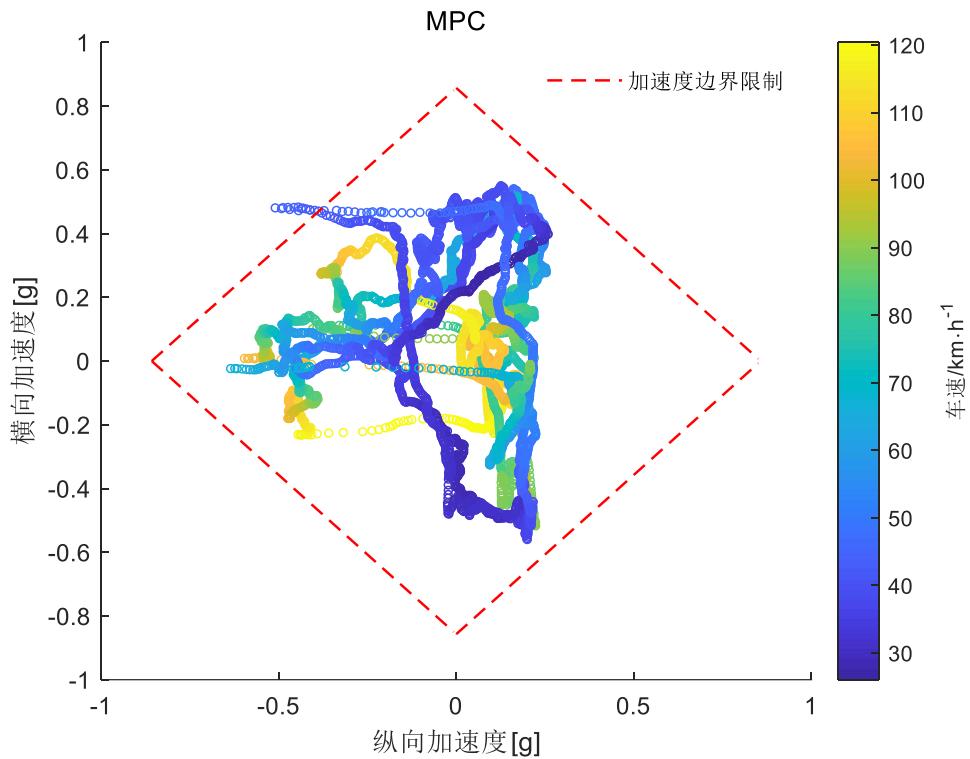


图 5.13 MPC 算法 g-g 图

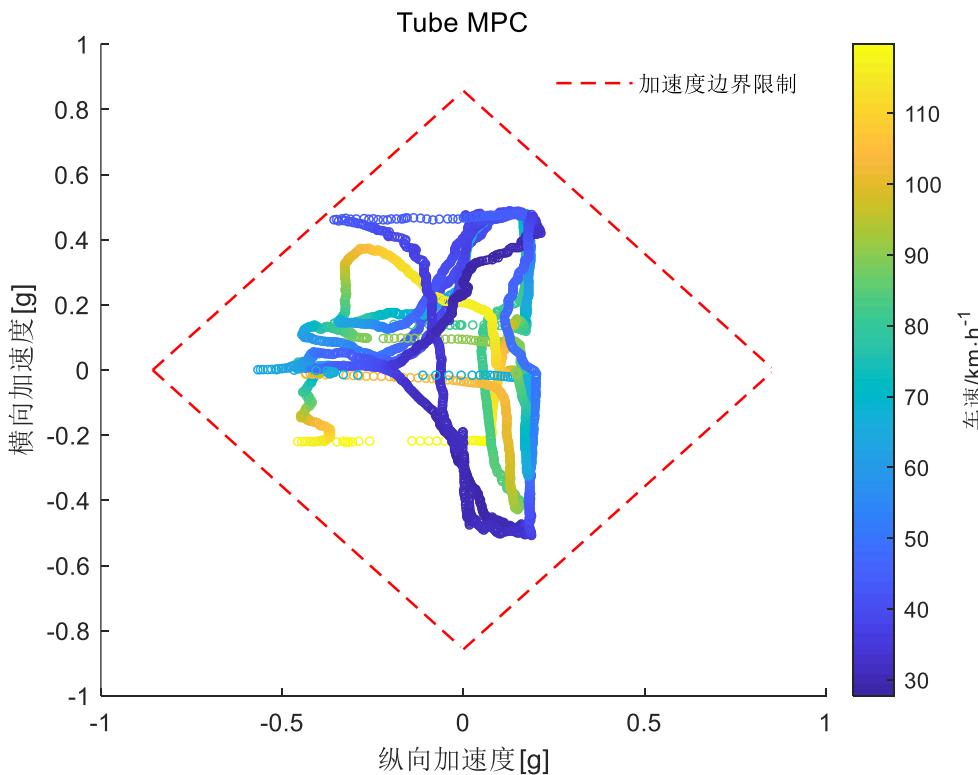


图 5.14 Tube MPC 算法 g-g 图

如图 5.13 和图 5.14 所示分别为 MPC 算法和 Tube MPC 算法的  $g$ - $g$  图像, 该图横纵坐标分别表示纵向加速度值与横向加速度值, 单位为重力加速度  $g$ , 图中红色虚线所表示的即为 2.2.2 节中所提出的菱形加速度极限约束式(2.15), 当  $g$ - $g$  图中的加速度点位于菱形约束范围内时, 说明控制算法输出的期望加速度信号始终遵守加速度边界约束, 能够保证车辆行驶的安全性和稳定性; 而当  $g$ - $g$  图中的加速度点超出菱形约束范围时, 则说明车辆将运行在危险范围内, 可能造成失稳的危险情况。

由图 5.13 和图 5.14 可以看出, MPC 算法所输出的加速度信号有超出加速度边界约束的情况, 而 Tube MPC 算法所输出的加速度信号能够始终维持在加速度边界约束范围内。且 MPC 算法所输出的  $g$ - $g$  曲线相比于 Tube MPC 算法, 变化趋势抖动较为剧烈, 这也从侧面再次说明 Tube MPC 算法能够输出变化更为平滑的加速度曲线, 更易于下层控制器的跟踪, 以及更利于提升舒适性。由此可见, 在存在横、纵向动力学外部干扰的情况下, Tube MPC 算法能够更好的抑制扰动的影响, 保证约束条件始终被满足, 维持控制系统的稳定性。

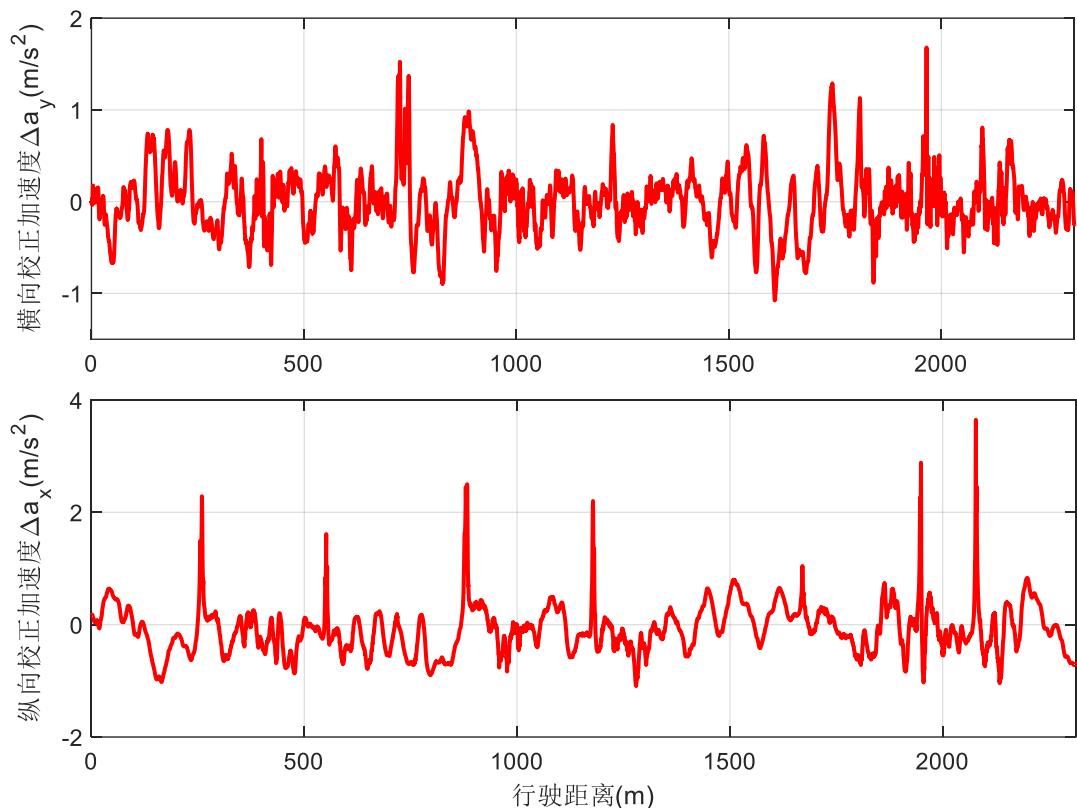


图 5.15 Tube MPC 算法校正加速度  $\Delta a_x$  和  $\Delta a_y$

Tube MPC 算法相比于 MPC 算法能实现上述这些优势的主要原因是其在 MPC 算法基础上在控制系统模型中引入了校正加速度  $\Delta a_x$  和  $\Delta a_y$  作为控制输入量, 同时考虑了横、纵向动力学附加扰动  $d_{a,x}$  和  $d_{a,y}$ , 并通过引入反馈增益矩阵和构建 Tube 不变集, 将不确定性系统的状态轨迹始终约束在以理想标称系统状态轨迹为中心的邻域内。在本节实验中 Tube MPC 求解得出的校正加速度  $\Delta a_x$  和  $\Delta a_y$  如图 5.15 所示, 其可以对输出的加速度信号  $a_x$  和  $a_y$  进行校正, 使其在存在外部干扰的情况下仍旧能够尽可能地贴近轨迹规划得到参考加速度信号, 并且能够确保加速度边界约束始终被满足。

综上所述, 在上层轨迹跟踪控制器中, Tube MPC 算法相比于常规的 MPC 算法能够通过引入校正加速度  $\Delta a_x$  和  $\Delta a_y$  对输出结果进行调整, 进而得到更贴近轨迹规划参考值、更平滑且易于跟踪的期望加速度信号, 同时保证控制系统始终满足加速度边界约束, 进而能够更好地抑制外部干扰的影响, 维持系统的稳定性, 显著地提高控制策略的鲁棒性。

### 5.3.2.2 下层加速度跟踪控制效果对比验证

在完成对上层控制器输出的期望加速度仿真结果分析后, 下文将对下层加速度跟踪控制器仿真结果进行分析。下层加速度跟踪控制器将对上层控制器中 MPC 和 Tube MPC 两种算法输出的期望加速度信号进行跟踪, 从而输出车辆控制指令, 即前轮转角和纵向力, 控制车辆跟踪参考轨迹。下文将对比分析两种算法在附加外部干扰影响下的横、纵向轨迹跟踪效果。

如图 5.16 所示为两种算法对参考轨迹的横向跟踪效果对比, 其中左图为横向位移偏差图, 右图为航向偏差图。分析图中数据可知, 在 MPC 算法的跟踪结果中, 横向位移偏差的最大值和均方根值分别为 0.312 m 和 0.135 m, 而在 Tube MPC 算法的跟踪结果中, 可以将横向位移偏差的最大值和均方根值分别降低至 0.208 m 和 0.088 m, 分别降低了 33.3% 和 34.8%; MPC 算法的航向偏差最大值和均方根值分别为 0.0778 rad 和 0.0229 rad, 而 Tube MPC 算法则将其分别降低至 0.0663 rad 和 0.0194 rad, 分别降低了 14.8% 和 15.3%。图中数据分析结果表明, 在外部干扰下, Tube MPC 算法相比于 MPC 算法所输出的期望加速度信号经下层加速度跟踪控制器跟踪后能降低横向位移跟踪偏差和航向偏差, 提升横向控制的跟踪精度。

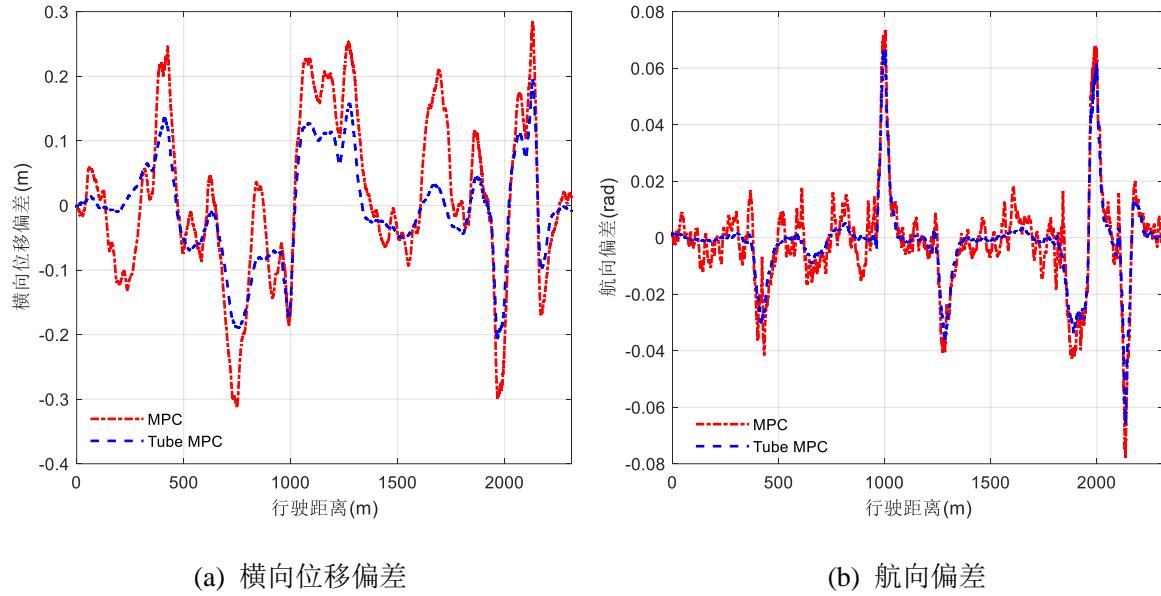


图 5.16 横向轨迹跟踪效果对比

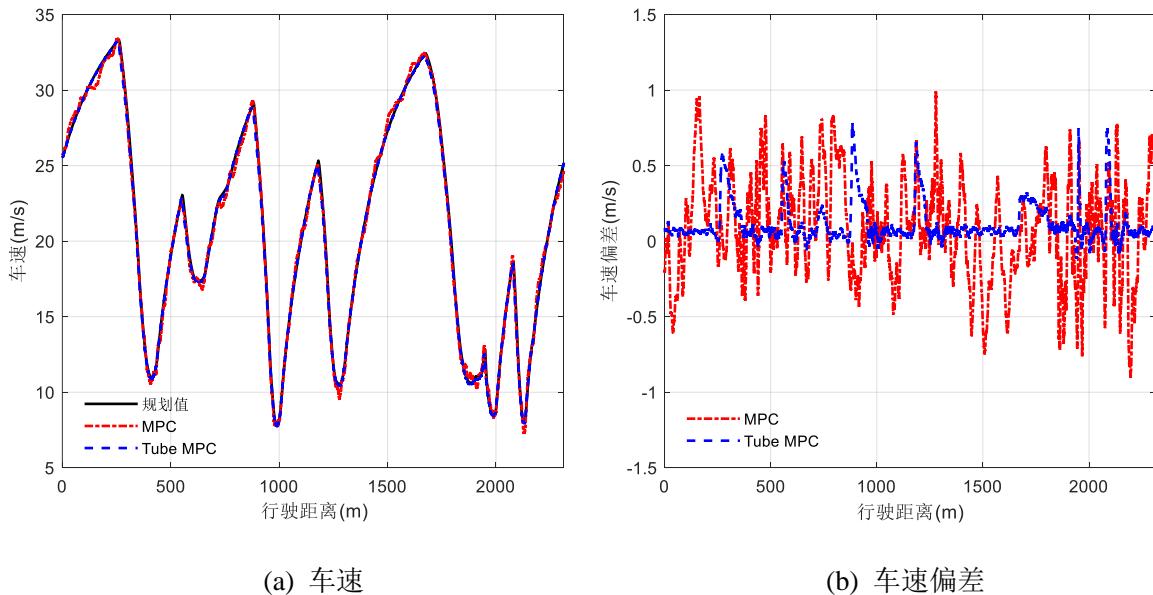


图 5.17 纵向轨迹跟踪效果对比

如图 5.17 所示为两种算法对参考轨迹的纵向跟踪效果对比，其中左图为车速图，右图为车速偏差图。分析图中数据可知，在 MPC 算法的跟踪结果中，车速偏差的最大值和均方根值分别为  $0.992 \text{ m/s}$  和  $0.349 \text{ m/s}$ ，而在 Tube MPC 算法的跟踪结果中，可以将车速偏差的最大值和均方根值分别降低至  $0.782 \text{ m/s}$  和  $0.164 \text{ m/s}$ ，分别降低了  $21.2\%$  和  $53.0\%$ 。图中数据分析结果表明，在外部干扰下，Tube MPC 算法相比于 MPC 算法能降低纵向车速偏差，提升纵向控制的跟踪精度。

分析图 5.18 中的控制指令信号可知，在外部干扰的影响下，MPC 算法得到的前轮转角和纵向力均存在一定程度的振荡和更大的幅值，相比之下，Tube MPC 算法则能够输出较为平稳的控制指令，保证智能车辆轨迹跟踪控制的稳定和跟踪精度。

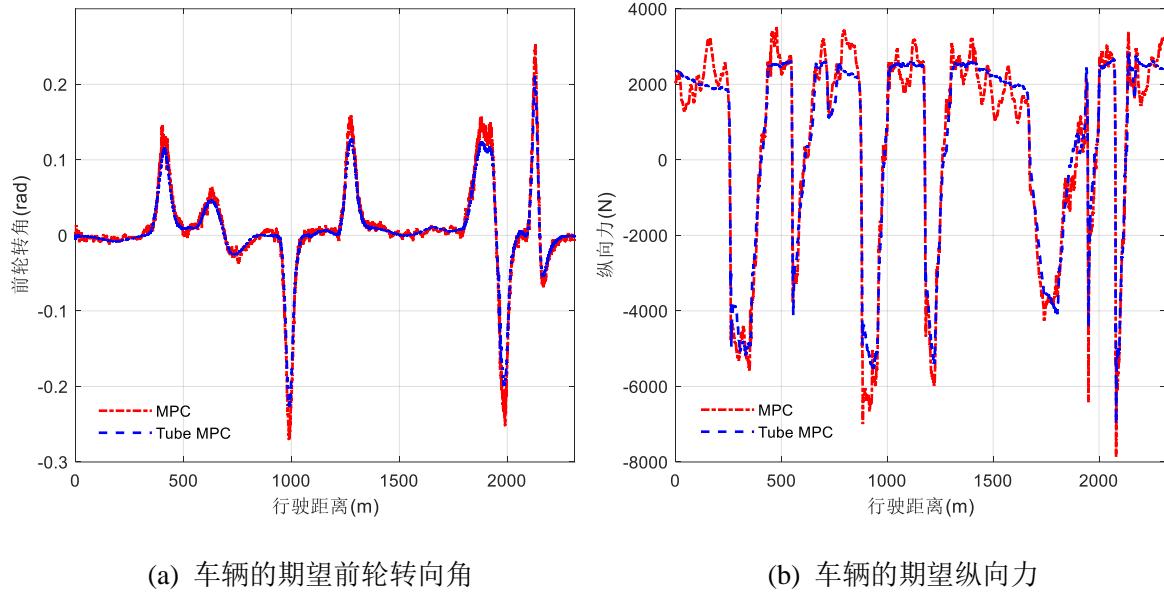


图 5.18 车辆控制动作对比

综上所述，本文所提出的基于 Tube MPC 算法的上层轨迹跟踪鲁棒控制器，对于存在外部干扰的情况有着更好的鲁棒优化效果，进一步提升了常规的 MPC 算法抑制外部干扰的能力，维持轨迹跟踪控制系统的稳定性，输出更接近规划参考值且更平滑更易于跟踪的横、纵向期望加速度，进而实现更好的横、纵向轨迹跟踪效果。

### 5.3.3 参数不确定性的仿真实验

本节将针对参数不确定性对 Tube MPC 和 MPC 算法的鲁棒性和控制性能进行对比验证。本节仿真实验仍对环形赛道工况进行跟踪，将针对控制系统模型中的车辆质量参数进行设置，如表所示将实际车辆质量下调 10%，模拟模型参数与实际失配的情况。仿真结果及分析如下。

表 5-2 车辆质量参数设置

模型质量参数	实际质量参数
1270 kg	1143 kg

如图 5.19 所示为 Tube MPC 和 MPC 两种算法所输出的期望纵向加速度信号，以及 Tube MPC 算法的控制输入量之一纵向校正加速度  $\Delta a_x$ 。由图可知，Tube MPC 算法

输出的期望加速度信号略小于 MPC 算法, 这是由于纵向校正加速度  $\Delta a_x$  对 Tube MPC 算法的期望加速度信号做出了修正。如图 5.20 所示为两种算法输出的期望横向加速度信号以及横向校正加速度  $\Delta a_y$ , 由图可知, 两种算法的期望横向加速度信号偏差较小, 基本保持一致, 且横向校正加速度  $\Delta a_y$  对于 Tube MPC 算法期望横向加速度的修正较小, 这说明实际车辆质量参数下调 10% 主要对车辆的纵向性能造成影响, 对横向性能影响较小。本文所提出的基于 Tube MPC 算法的轨迹跟踪鲁棒控制器能够将横、纵向的加速度偏差视为扰动信号, 控制过程中通过对 Tube MPC 优化问题进行求解可以得到校正加速度  $\Delta a_x$  和  $\Delta a_y$ , 针对这种模型参数不确定性所造成的加速度偏差, 校正加速度将对上层轨迹跟踪鲁棒控制器输出的期望加速度信号做出调整, 以缩小实际的加速度跟踪偏差, 下文将分析调整后的期望加速度信号对于下层加速度跟踪效果的影响。

两种算法输出的期望加速度信号经下层加速度跟踪控制器跟踪后所得到的实际纵向加速度信号如图 5.21 所示, 实际横向加速度信号如图 5.22 所示, 由图可知, 质量参数的变化对于横向加速度的影响较小, 易于看出两种算法输出的横向加速度信号经跟踪后得到的实际横向加速度均与规划参考值较为接近, 偏差较小。由图 5.21 可知, 质量参数不确定性对两种算法控制性能的影响主要体现在纵向加速度方面, Tube MPC 算法输出的期望纵向加速度经跟踪控制后与规划的参考加速度信号更为贴近, 而 MPC 算法的纵向加速度跟踪效果则与规划参考值存在一定的偏差, 整体呈现大于规划参考值的趋势。这是由于实际车辆质量小于模型中的质量参数, 导致了 MPC 算法输出的期望纵向力相比于实际需要的纵向力偏大, 车辆执行了偏大的纵向力控制动作, 所以 MPC 算法输出的期望加速度信号经下层加速度跟踪控制器跟踪后得到的实际纵向加速度跟踪结果也偏大, 而 Tube MPC 可以将纵向加速度偏差视为扰动信号, 在控制过程中计算出校正加速度  $\Delta a_x$  对期望加速度信号做出实时调整, 从而保证最终更好的加速度跟踪效果。

如图 5.23 所示为下层加速度跟踪控制器对两种算法输出的期望加速度信号进行跟踪后输出的车辆控制动作, 图 5.23 (a) 所示为期望的前轮转角, 图 5.23 (b) 所示为期望的纵向力。与前文分析一致, 两种算法最终得到的横向控制动作, 即前轮转角基本一致, 而在纵向控制动作, 即纵向力方面, Tube MPC 算法对期望纵向加速度的调整得到体现, 输出得到相比 MPC 算法略小的纵向力, 从而得到图 5.21 中与轨迹规划的参考加速度值更为接近的跟踪效果。

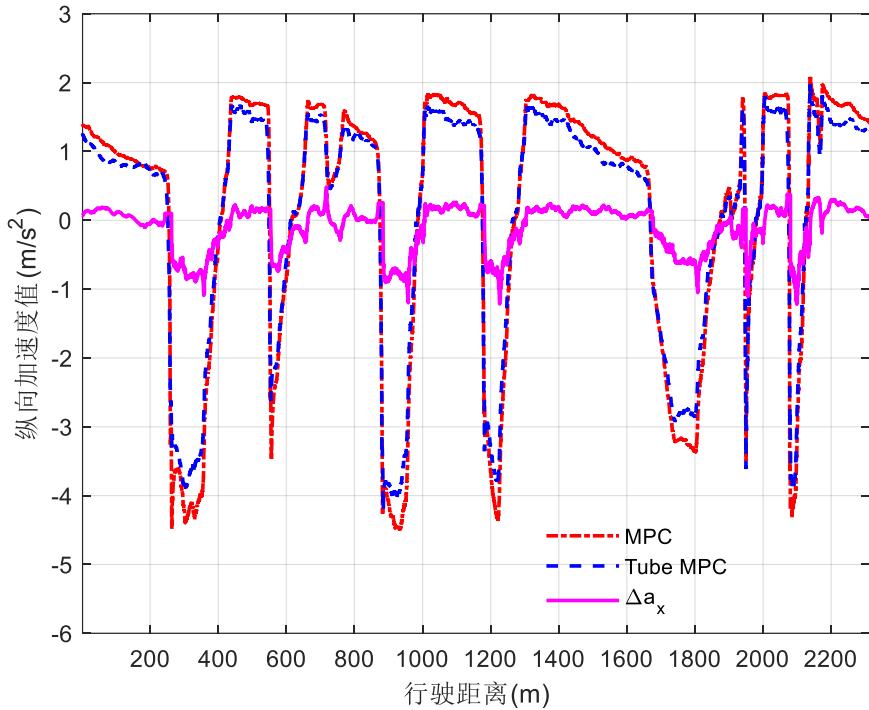


图 5.19 期望纵向加速度及纵向校正加速度  $\Delta a_x$

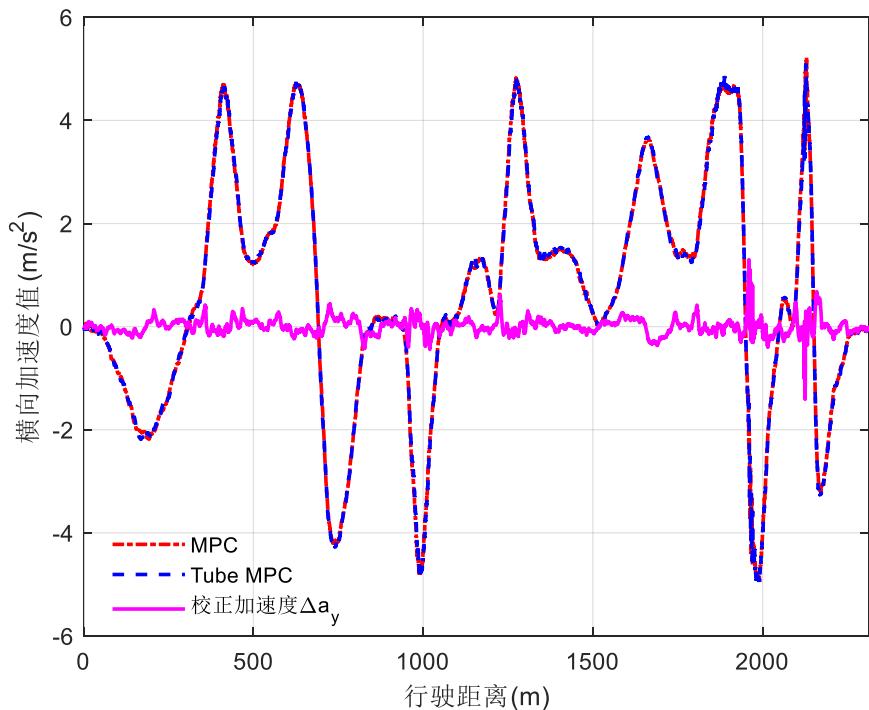


图 5.20 期望纵向加速度及纵向校正加速度  $\Delta a_y$

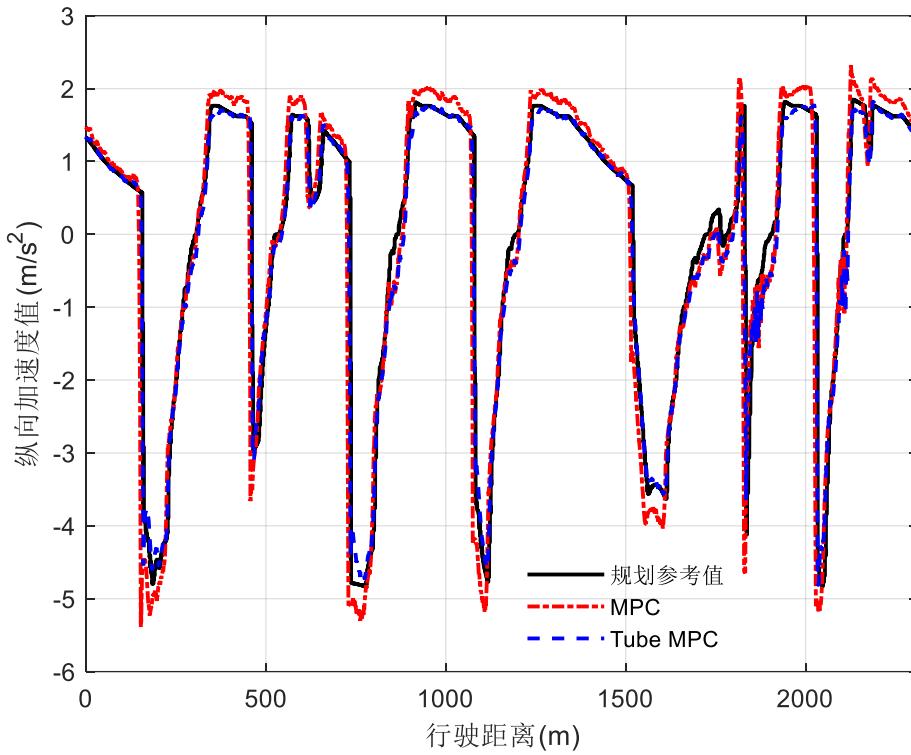


图 5.21 实际纵向加速度跟踪结果

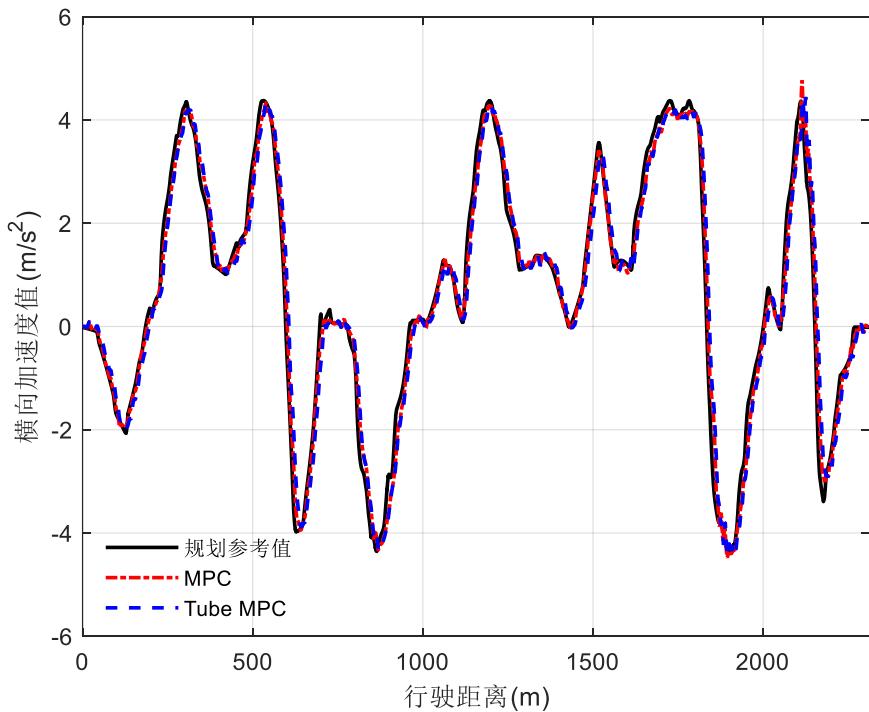


图 5.22 实际横向加速度跟踪结果

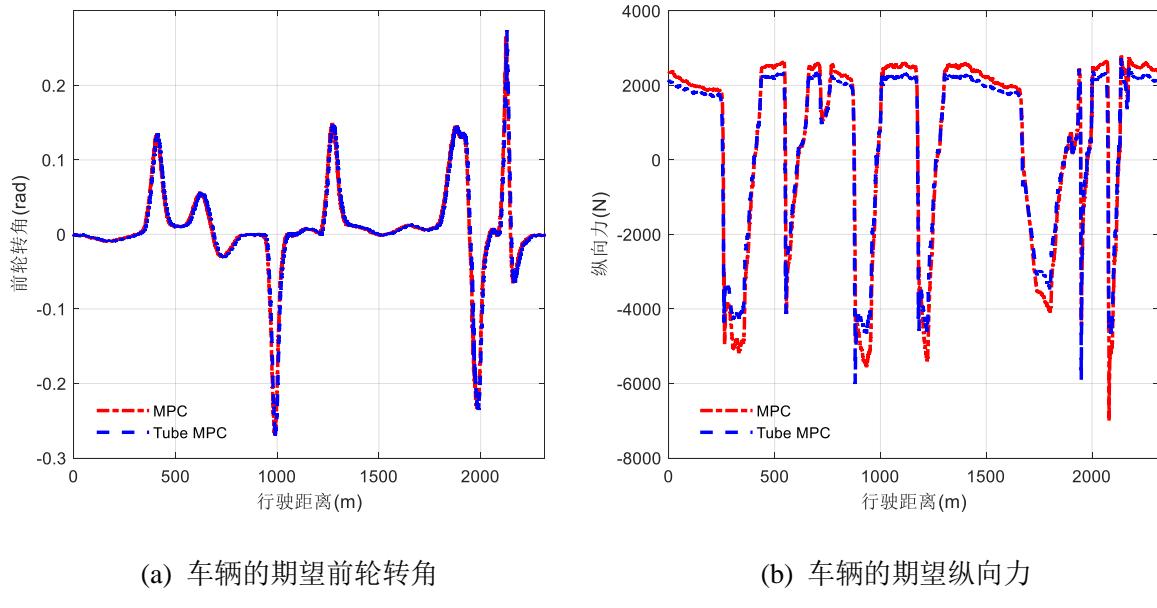


图 5.23 车辆控制动作对比

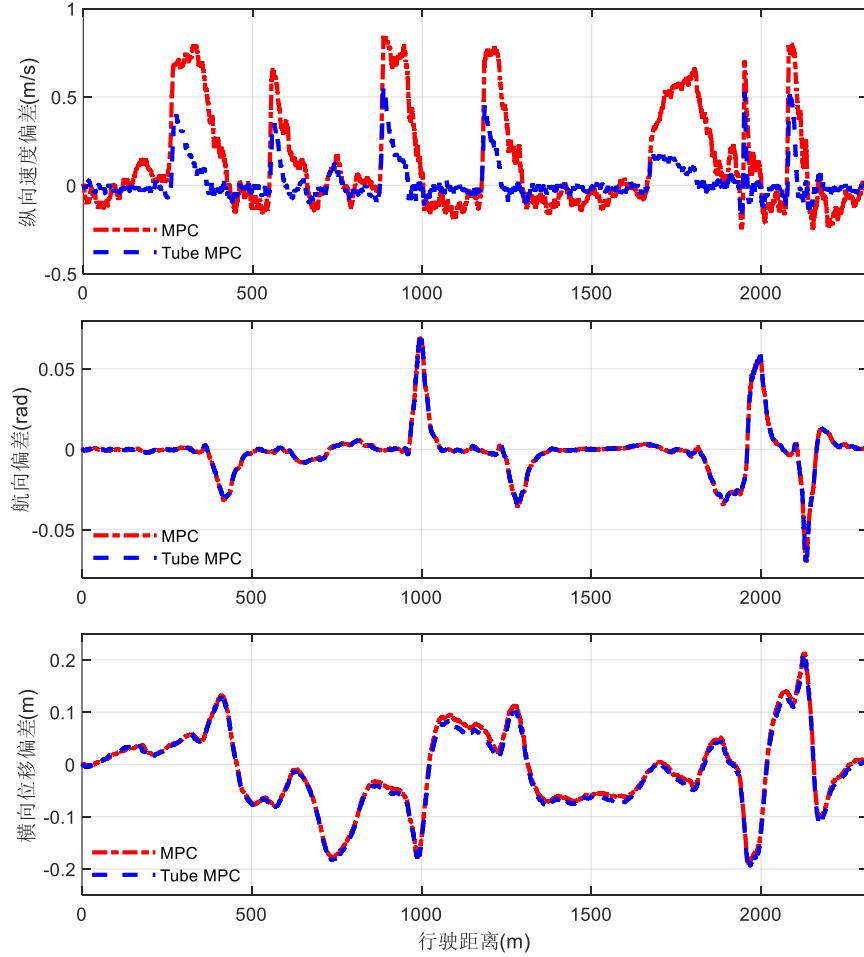


图 5.24 横纵向轨迹跟踪效果对比

如图 5.24 所示为最终得到的横纵向轨迹跟踪效果，由上到下依次为纵向速度偏差、横向位移偏差以及航向偏差。与前文分析一致，质量参数不确定性对于横向控制影响较小，因此两种算法最终的横向位移偏差和航向偏差较为接近。而在纵向速度偏差方面，分析图中数据可知，MPC 算法纵向速度偏差的最大值和均方根值分别为 0.846 m/s 和 0.304 m/s，而 Tube MPC 算法将其分别降低至 0.555 m/s 和 0.102 m/s，分别降低了 34.4% 和 66.4%，提升了纵向控制的跟踪精度。

综上所述，Tube MPC 算法相比于 MPC 算法对于参数不确定性有较好的抑制作用，通过校正加速度对期望加速度信号进行调整，从而能够提升控制策略的鲁棒性，进而提升轨迹跟踪精度。

## 5.4 蛇形工况仿真实验设置

### 5.4.1 仿真实验条件设置

本节将利用联合仿真实验平台对智能车辆轨迹跟踪控制器中的下层加速度跟踪控制器进行性能验证，主要对 4.2 节和 4.3 节中提出的优化策略效果进行验证。由于本节仿真实验主要分析下层加速度跟踪控制器优化策略效果，为便于分析，本节实验将采用如图 5.25 所示的短距离蛇形工况参考轨迹进行跟踪并分析跟踪结果，该参考轨迹总长度为 458 m，规划车速变化范围为 [36, 120] km/h，在实验开始前将初始车速设定为 33 m/s，该工况在横向跟踪方面包含前段和后段的直道路段，以及中段的连续左右转向蛇形路径，在纵向跟踪方面包含了驶入弯道前的减速工况和驶出弯道后加速，因此适合用于本文的下层加速度跟踪控制器性能验证，假设上层 Tube MPC 轨迹跟踪鲁棒控制器已经由轨迹规划模块获得所需的参考轨迹信息，能够输出期望横、纵向加速度信号，仿真实验将基于上层控制器输出的期望加速度信号验证下层加速度跟踪控制器的控制性能。

下层加速度跟踪控制器的主要参数设置如下：

横向加速度反馈 PI 控制的比例系数和积分系数  $K_{P,y}$  和  $K_{I,y}$  分别设定为 1.2 和 0.3；纵向加速度双闭环 PID 反馈控制中，外环速度 PID 控制的比例系数、积分系数和微分系数  $K_{P,v}$ ， $K_{I,v}$  和  $K_{D,v}$  分别设定为 3，0.001 和 0.1，内环加速度 PID 控制的比例系数、积分系数和微分系数  $K_{P,x}$ ， $K_{I,x}$  和  $K_{D,x}$  分别设定为 1.2，0 和 0.1。

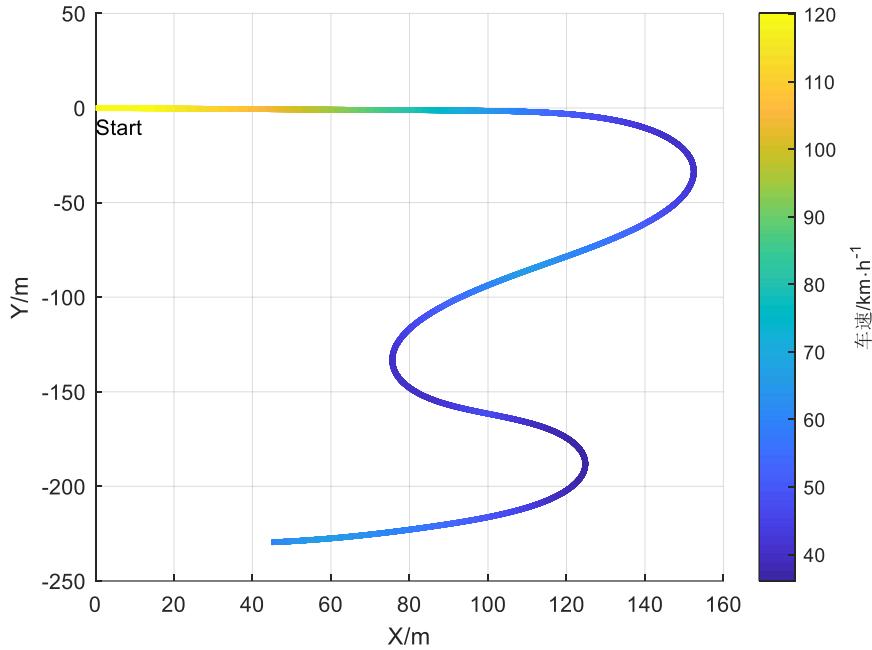


图 5.25 蛇形工况参考轨迹

#### 5.4.2 前馈修正策略有效性仿真验证

本节实验的设置用于对 4.1 节所设计的加速度跟踪控制策略和 4.2 节中提出的前馈修正优化策略进行性能验证和效果对比。利用仿真实验平台分别搭建了 4.1 节中设计的无前馈修正项的常规加速度跟踪控制器以及 4.2 节中设计的带有前馈修正优化策略的加速度跟踪控制器，分别对蛇形工况参考轨迹进行跟踪，主要对下层加速度跟踪控制器的输出信号进行分析。

前馈修正策略中的学习率  $\mu$ ，调节因子  $\alpha$  以及遗忘因子  $\lambda$  分别设定为 0.1, 0.05 和 1，仿真实验结果和分析如下。

如图 5.26 和图 5.27 所示分别为横向和纵向的加速度跟踪结果图，图中黑色实线为由上层轨迹跟踪鲁棒控制器输出的期望加速度信号，红色点划线表示无前馈修正的常规加速度跟踪控制策略的跟踪结果，蓝色虚线表示带有前馈修正优化策略的加速度跟踪控制结果。由图 5.26 和图 5.27 可知，基于前馈和反馈结合的加速度跟踪控制策略能对期望的加速度信号有较为稳定的跟踪效果，但是无前馈修正的加速度跟踪结果相比于带有前馈修正优化策略的加速度跟踪结果响应速度较慢且跟踪精度较差。这是由于加速度跟踪控制策略中前馈项是基于预先设定好的输入输出关系进行计算，而输入输出的关系式与车辆动力学相关，通常带有一定的简化和假设，因此与实际模型有

所差异，从而造成了前馈项与期望响应存在偏差，在加速度跟踪控制的初期需要反馈控制环节对该偏差进行调整，导致加速度跟踪响应速度较慢，精度较差。而基于误差模型在线估计的前馈校正通过对前馈项和期望输出信号的偏差进行在线估计进而得到前馈修正项添加到前馈信号中，从而使得前馈项在控制初期即与期望响应更加接近，提高了加速度跟踪控制的响应速度和跟踪精度。

如图 5.28 所示分别为车辆控制动作的对比图，其中，图 5.28 (a)为期望的前轮转角，图 5.28 (b)为期望的纵向力。车辆的控制动作是其横、纵向加速度变化的原因，因此控制量的变化趋势与加速度的变化趋势相接近，未添加前馈校正的控制器因前馈项与期望输出响应存在偏差，所以加速度跟踪控制器所计算得到车辆控制动作存在滞后和超调的情况。而添加了前馈修正的控制器，经过前馈校正的优化后，在原先计算出的控制动作基础上添加了前馈修正项，使得前馈项更加接近期望响应，因此相比于未添加前馈修正的加速度跟踪控制器，其控制动作响应速度更快且更精准，有利于提升加速度跟踪精度。

图 5.29 所展示的为横、纵向轨迹跟踪控制效果。由上到下三张图分别展示的是横向位移偏差、航向偏差以及纵向速度偏差，横向位移偏差和航向偏差用于反映横向的轨迹跟踪控制效果，纵向速度偏差用于反映纵向的轨迹跟踪控制效果。分析横向位移偏差图中的数据可知，添加前馈修正可以将横向位移偏差的最大值由 0.25 m 降低到 0.19 m，均方根值由 0.15 m 降低到 0.10 m，分别降低了 24.2% 和 33.3%。分析航向偏差图中数据可知，添加前馈修正可以将航向偏差的最大值由 0.048 rad 降低到 0.032 rad，均方根值由 0.021 rad 降低到 0.015 rad，分别降低了 33.3% 和 28.6%。分析纵向速度偏差图中的数据可知，添加前馈修正可以将纵向速度偏差的最大值由 1.25 m/s 降低到 0.75 m/s，均方根值由 0.31 m/s 降低到 0.20 m/s，分别降低了 40.0% 和 35.5%。由此可见，添加前馈修正的加速度跟踪控制策略能够提升控制动作的响应速度和加速度的跟踪精度，从而能够提升智能车辆在横、纵向参考轨迹跟踪上的跟踪精度。

综合上述实验结果分析可知，本文于 4.1 节所提出的基于前馈和反馈结合的下层加速度跟踪策略能够基本稳定地跟踪期望加速度信号，符号加速度跟踪需求。在 4.2 节提出的基于误差模型在线估计的前馈修正优化策略则可以进一步提升前馈项的精准度，进而有效提升加速度跟踪的响应速度，提升了对期望加速度信号的跟踪精度，进一步降低了轨迹跟踪的横向位移偏差、航向偏差以及纵向车速偏差，提升了轨迹跟踪精度。

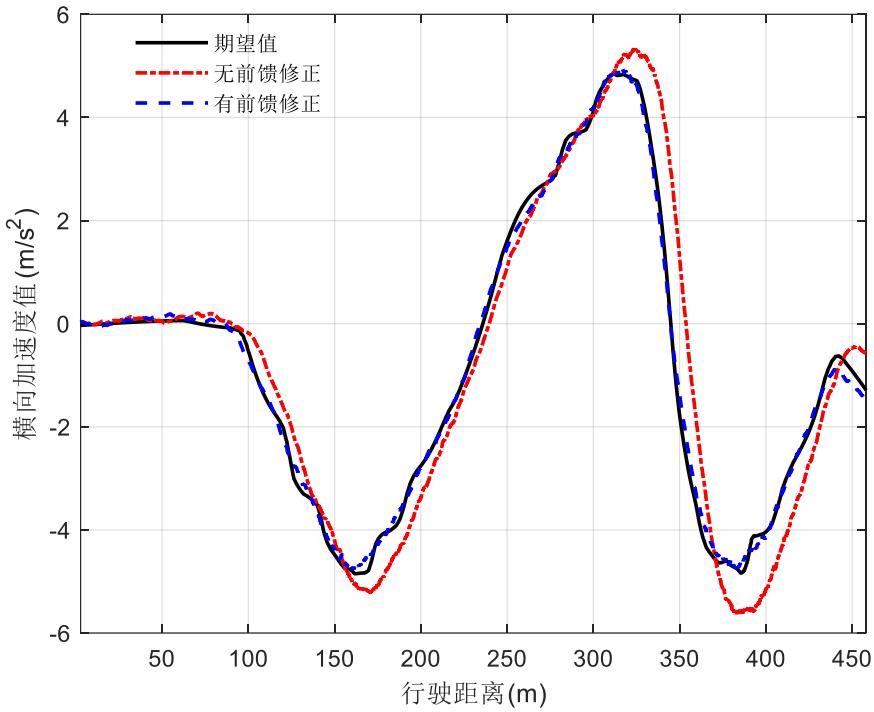


图 5.26 横向加速度跟踪结果

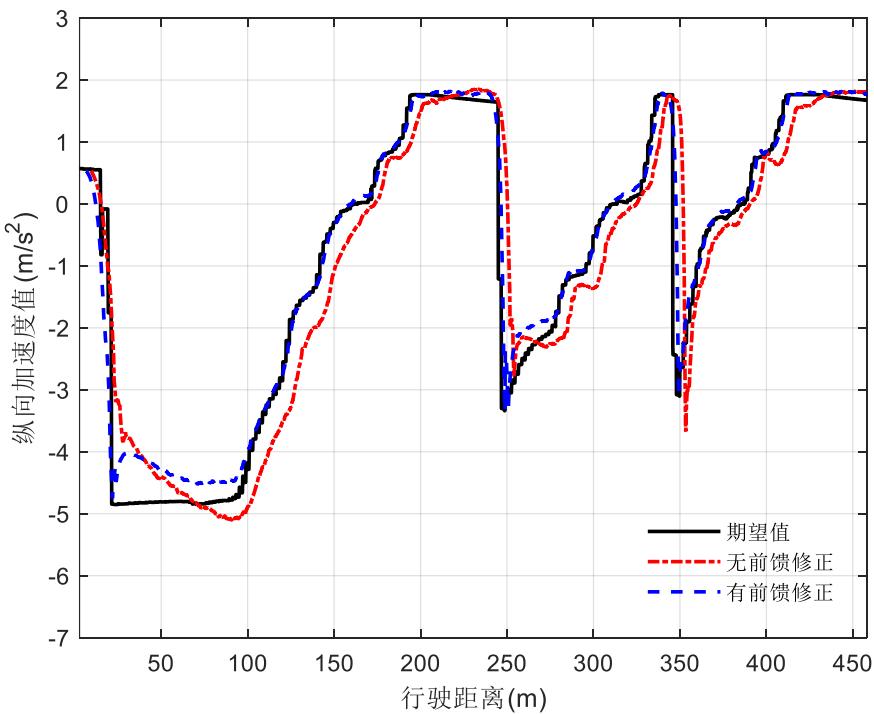


图 5.27 纵向加速度跟踪结果

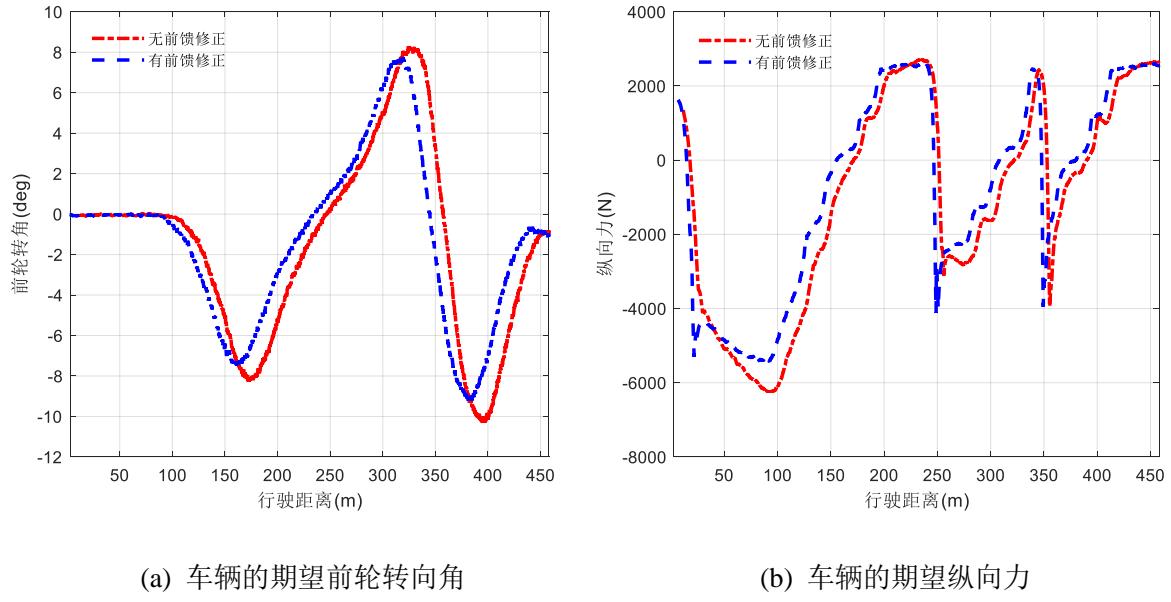


图 5.28 车辆控制动作对比

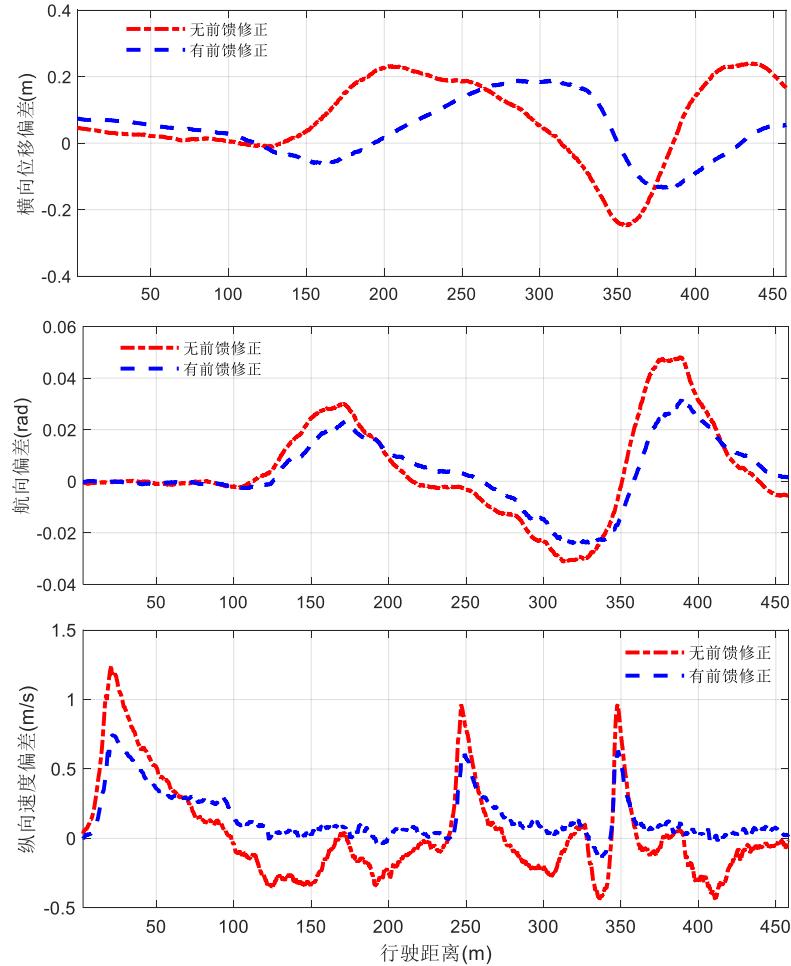


图 5.29 横、纵向轨迹跟踪结果对比

### 5.4.3 执行器延迟优化策略有效性仿真验证

在实际的车辆控制中，执行器信号延迟是一个常见的问题。由于各种因素的影响，包括信号传输时间、传感器采样时间以及执行器响应时间等等，都会导致车辆控制系统中的执行器信号延迟。当存在执行器延迟时，控制系统的反馈信号与实际的控制指令之间存在时间差。这意味着当控制系统检测到需要进行调节时，执行器并不能立即响应并执行相应的指令而是会延迟一段时间，这种延迟会导致控制系统的输出会因为执行器无法及时响应控制指令而产生过冲现象，导致系统反复振荡，车辆的响应会受到影响，会导致智能车辆轨迹跟踪性能下降，甚至可能会引发危险情况。

针对上述问题，本文在仿真实验平台中利用 Simulink 中的延迟模块设计了如图 5.30 所示信号延迟环节用于模拟现实中的执行器延迟情况，其中从控制器到车辆执行机构的信号传输延迟设定为 10 ms，车辆执行机构接收到控制指令并执行的延迟设定为 200 ms，在此基础上对本文提出的模型参考自适应算法进行效果验证，实验结果及分析如下。



图 5.30 仿真实验信号延迟设置示意图

如图 5.31 所示为纵向加速度跟踪效果对比图，图 5.32 所示为横向加速度跟踪效果对比图，图中黑色实线表示由上层轨迹跟踪控制算法得到的期望横、纵向加速度信号，红色点划线表示没有添加延迟信号处理的下层加速度跟踪控制效果，蓝色虚线表示添加了 MRAC 延迟信号处理的下层加速度跟踪控制效果。由图 5.31 和图 5.32 中无延迟信号处理的加速度跟踪效果可知，执行器的延迟将会造成加速度跟踪控制过程中的超调、振荡现象，导致横、纵向的加速度跟踪偏差较大。而添加了基于 MRAC 算法的延迟信号处理后，可以明显抑制超调、振荡现象，降低了加速度跟踪偏差，使得车辆实际的加速度曲线贴近期望的加速度曲线。这是由于 MRAC 算法能够依据实际模型与参考模型的输出偏差实时调整自适应控制律，从而在控制过程中实时调整控制动作，即前轮转角和纵向力，从而确保实际模型所输出的控制动作与考虑了延迟影响的参考模型控制动作相接近，提升加速度跟踪精度。

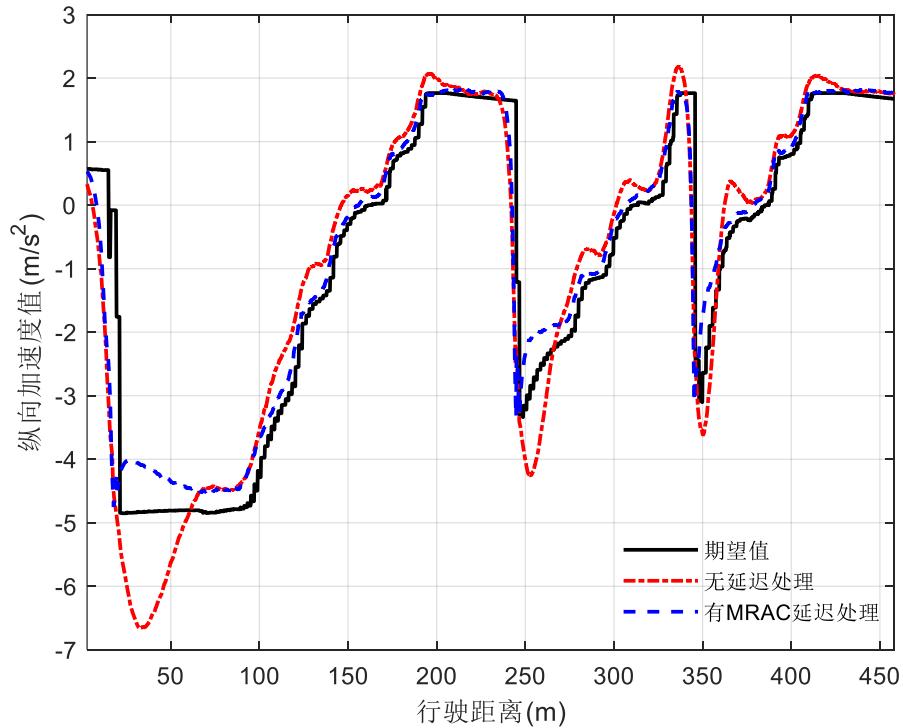


图 5.31 纵向加速度跟踪效果对比

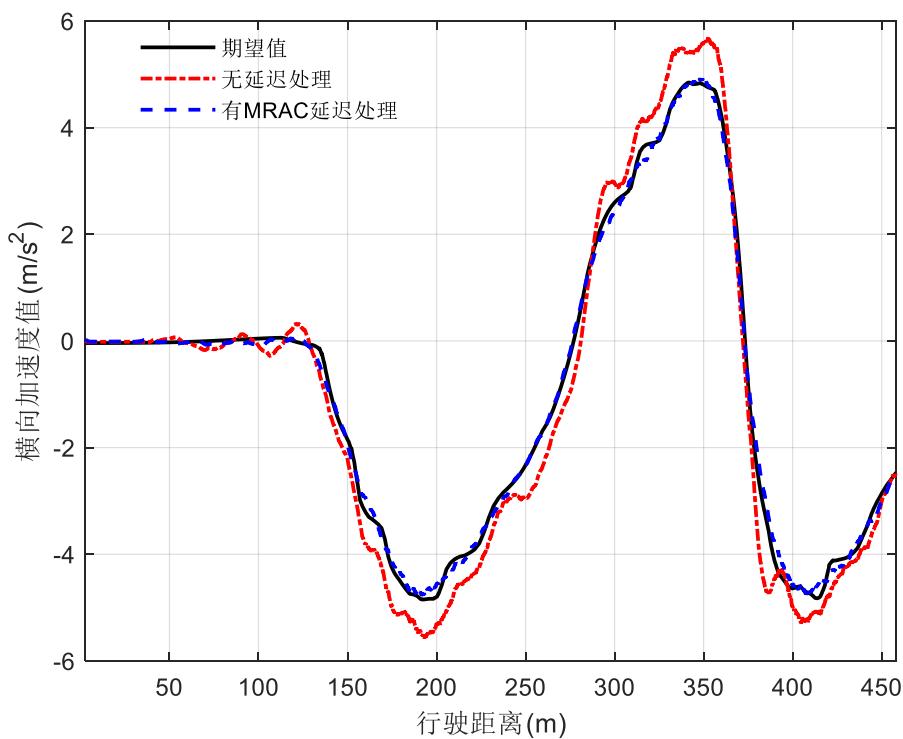


图 5.32 横向加速度跟踪效果对比

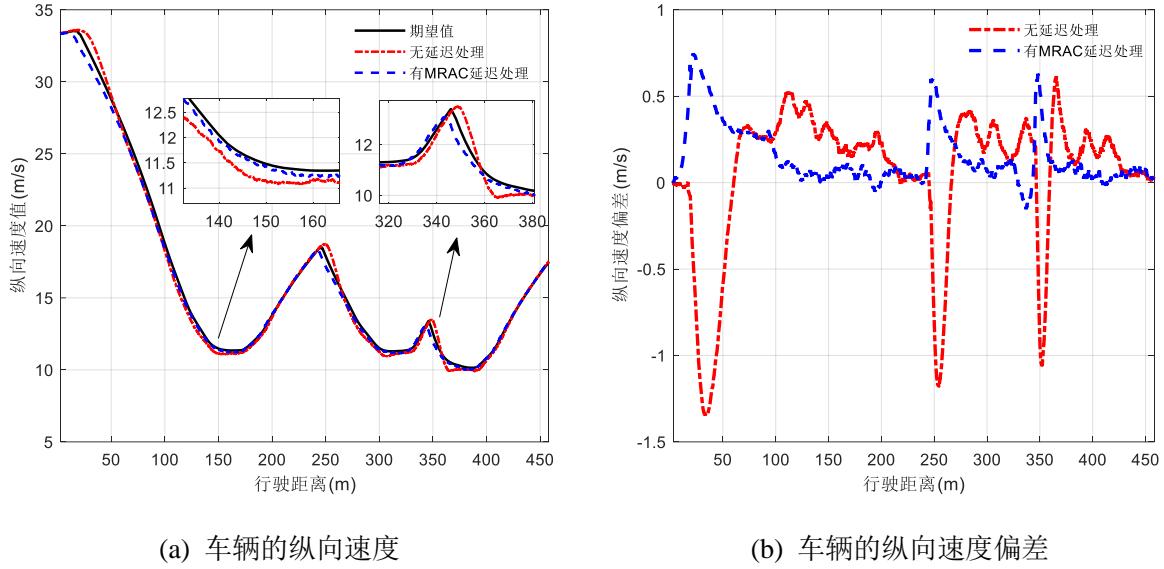


图 5.33 纵向轨迹跟踪控制效果对比

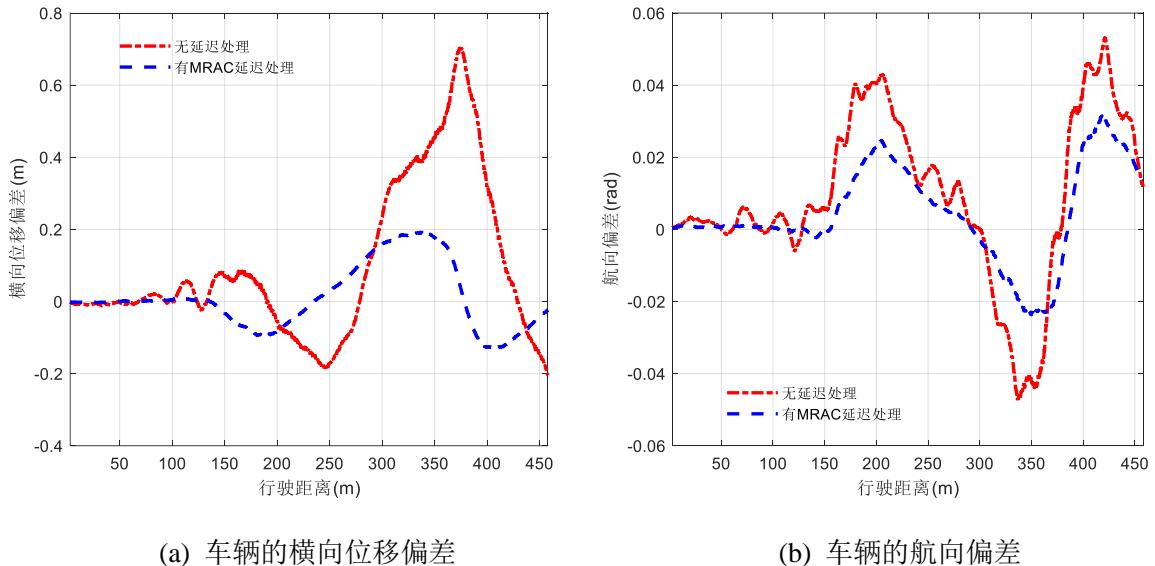


图 5.34 横向轨迹跟踪控制效果对比

图 5.33 和图 5.34 所展示的分别为纵向跟踪控制效果和横向跟踪控制效果。分析图 5.33 (b)中的数据可知,添加基于 MRAC 算法的延迟信号处理可以将纵向速度偏差的最大值由  $1.35 \text{ m/s}$  降低到  $0.74 \text{ m/s}$ , 均方根值由  $0.37 \text{ m/s}$  降低到  $0.19 \text{ m/s}$ , 分别降低了 45.2% 和 48.6%, 提升了智能车辆在纵向跟踪控制上的精度。分析图 5.34 (a)中数据可知,添加基于 MRAC 算法的延迟信号处理可以将横向位移偏差的最大值由  $0.69 \text{ m}$  降低到  $0.20 \text{ m}$ , 均方根值由  $0.26 \text{ m}$  降低到  $0.10 \text{ m}$ , 分别降低了 71.0% 和 61.5%, 同时可以将车辆的航向偏差的最大值由  $0.053 \text{ rad}$  降低到  $0.031 \text{ rad}$ , 均方根值由  $0.026 \text{ rad}$

降低到  $0.015 \text{ rad}$ , 分别降低了 41.5% 和 42.3%, 提升了智能车辆在横向轨迹跟踪控制上的精度。

图 5.35 所示为下层加速度跟踪控制器所输出的车辆控制指令, 分别为期望的前轮转角和期望的纵向力。由控制指令图对控制过程进行分析可知, 面对未知的延迟信号影响, MRAC 算法通过设置理想的参考模型并在控制过程中依据实际模型和参考模型的偏差自适应调整输出的控制指令, 使其与理想的输出响应所接近, 从而削弱信号延迟所造成的超调、振荡影响, 达到更好的控制效果。

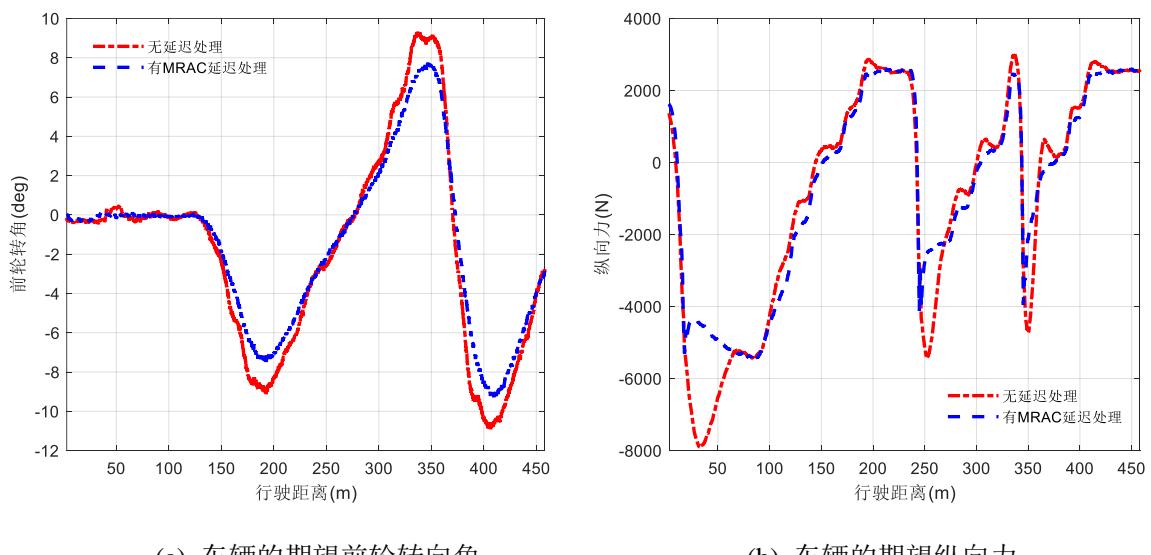


图 5.35 车辆控制动作对比

综上所述, 本文所设计的基于 MRAC 算法的执行器信号延迟处理策略能有效抑制信号延迟所带来的影响, 避免超调、振荡现象, 确保跟踪得到的实际加速度与期望加速度尽可能贴近, 从而提升轨迹跟踪控制在横向和纵向上的跟踪精度。

## 5.5 本章小结

本章搭建了基于 Simulink 和 Carsim 的联合仿真实验平台, 确定了仿真实验架构, 基于 Simulink 和 S 函数编写了轨迹跟踪控制器, 设定了被控车辆的技术参数、行驶环境; 设计了长距离复杂环形赛道工况进行实验, 分别设置了附加外部干扰和参数不确定性两种实验工况, 对 MPC 和 Tube MPC 两种算法的鲁棒性和轨迹跟踪效果进行验证, 仿真结果表明, Tube MPC 算法相比于 MPC 算法能有效抑制外部干扰和参数不确定性的影响, 在应对附加纵向扰动力和横摆扰动力矩时, 能输出更为平滑且易于跟踪

的期望加速度曲线，保证加速度边界始终被满足，维持系统的稳定性，并最终提升了轨迹跟踪精度，在应对质量参数不确定性时，能根据模型失配导致的加速度偏差实时调整期望加速度信号，使得最终实际的加速度跟踪效果与规划参考值更为贴近；针对下层加速度跟踪控制优化策略，设计了短距离蛇形工况参考轨迹，对基于误差模型在线估计的前馈修正策略效果以及基于 MRAC 的执行器延迟处理策略的效果进行验证。仿真结果表明，基于误差模型估计的前馈修正策略能有效提升控制器的响应速度，显著降低了横、纵向的跟踪偏差，基于 MRAC 的执行器延迟处理策略能有效削弱延迟信号造成的超调振荡现象，提升轨迹跟踪精度。

## 结论

### 全文总结

本文以智能车辆轨迹跟踪控制为研究内容,针对轨迹跟踪过程中的外部干扰和参数不确定性问题,提出了一种基于 Tube MPC 算法和 RUMI 扰动边界辨识的上下分层鲁棒控制策略,在存在外部干扰及参数不确定性的情况下,有效提升了轨迹跟踪控制算法的鲁棒性以及跟踪精度。具体研究了基于点质量模型的 Frenet 坐标系跟踪误差模型以及考虑扰动的状态空间方程;设计了基于归一化最小均方算法和递归分位数估计算法相结合的 RUMI 扰动边界在线辨识方法;基于 Tube MPC 算法设计了上层轨迹跟踪鲁棒控制器,输出期望的加速度;基于前馈和反馈结合的控制方法设计了下层加速度跟踪控制器以及前馈修正和延迟优化策略;基于 Simulink 和 Carsim 搭建联合仿真平台,通过仿真实验验证本文所提出控制策略的鲁棒性和跟踪效果。具体工作内容如下:

(1) 车辆模型及横纵向动力学扰动边界在线辨识。首先提出了本文轨迹跟踪鲁棒控制系统的上下层结构,对控制结构特点进行分析,并选择以加速度约束的车辆点质量模型作为上层轨迹跟踪鲁棒控制系统的模型;推导了 Frenet 坐标系和 Cartesian 坐标系的转换关系,并将 Cartesian 坐标系下的点质量模型转化为适用于轨迹跟踪的 Frenet 坐标系跟踪误差模型;通过引入校正加速度作为控制输入,引入横、纵向动力学附加扰动,建立了不确定性系统的线性状态空间方程,并对加速度约束条件做了线性化推导;为精确辨识扰动信号范围,提出了基于归一化最小均方算法和递归分位数估计算法相结合的 RUMI 扰动边界在线辨识方法,在仿真中与高斯过程回归和贝叶斯回归分别针对高斯噪声和非高斯噪声的辨识效果进行对比验证,仿真结果表明所提出 RUMI 算法对两种噪声扰动均能有效辨识,且计算速度更快。

(2) 基于 Tube MPC 算法的上层轨迹跟踪鲁棒控制策略。首先结合 Tube MPC 算法原理将标称系统与实际系统分离,构建了状态误差系统,推导了基于鲁棒不变集的约束条件收紧方法;基于在线可达集设计了 Tube 不变集计算方法;基于椭圆不变集的 Tube 不变集描述方式优化了约束条件的收紧计算;基于二次型李雅普诺夫函数和线性矩阵不等式设计了状态误差系统的反馈增益矩阵;构建了标称 MPC 优化问题,并推导了状态量以及控制输入量约束条件的收紧形式;基于二次型李雅普诺夫函数设计了标称系统 MPC 优化问题终端代价函数,并采用鲁棒一步集与最大鲁棒不变集来

计算终端约束集，最终完成 Tube MPC 轨迹跟踪鲁棒控制器的构建，输出期望加速度信号。

(3) 下层加速度跟踪控制以及前馈修正优化策略和执行器延迟优化策略。首先设计了基于前馈和反馈相结合的下层加速度跟踪控制器，输出对车辆的控制指令，分别为前轮转角和纵向力；分别推导了横向加速度跟踪策略和纵向加速度跟踪策略，其中横向加速度前馈由阿克曼转向原理推导得到，反馈由 PI 控制实现，并添加了抗积分饱和项；纵向加速度前馈则由车辆的行驶方程推导得到，反馈由双环 PID 控制实现，其中外环为速度控制，内环为加速度控制；针对实际测量得到的跟踪偏差问题，设计了基于误差模型在线估计的方法对前馈项进行修正，其中根据各自控制特点，纵向部分采用 NLMS 算法，横向部分采用 RLS 算法；针对执行器延迟问题，设计了基于模型参考自适应控制的延迟处理方法，使得实际模型的输出尽可能贴近参考模型输出，消除超调和振荡现象。

(4) 本章搭建了基于 Simulink 和 Carsim 的联合仿真实验平台，确定了仿真实验架构，基于 Simulink 和 S 函数编写了轨迹跟踪控制器，设定了被控车辆的技术参数、行驶环境；设计了长距离复杂环形赛道工况进行实验，分别设置了附加外部干扰和参数不确定性两种实验工况，对 MPC 和 Tube MPC 两种算法的鲁棒性和轨迹跟踪效果进行验证，仿真结果表明，Tube MPC 算法相比于 MPC 算法能有效抑制外部干扰和参数不确定性的影响，在应对附加横向扰动力和横摆扰动力矩时，能输出更为平滑且易于跟踪的期望加速度曲线，保证加速度边界始终被满足，维持系统的稳定性，并最终提升了轨迹跟踪精度，分别将横向、航向及车速偏差均方根值降低了 34.8%，15.3% 和 53.0%；在应对质量参数不确定性时，能根据模型失配导致的加速度偏差实时调整期望加速度信号，使得最终实际的加速度跟踪效果与规划参考值更为贴近，提升了轨迹跟踪精度，分别将车速偏差最大值和均方根值降低了 34.4% 和 66.4%；针对下层加速度跟踪控制优化策略，设计了短距离蛇形工况参考轨迹，对基于误差模型在线估计的前馈修正策略效果以及基于 MRAC 的执行器延迟处理策略的效果进行验证。仿真结果表明，基于误差模型估计的前馈修正策略能有效提升控制器的响应速度，降低了横、纵向的跟踪偏差，分别将横向、航向及车速偏差均方根值降低了 34.8%，15.3% 和 53.0%；基于 MRAC 的执行器延迟处理策略能有效削弱延迟信号造成的超调振荡现象，提升轨迹跟踪精度，分别将横向、航向及车速偏差均方根值降低了 33.3%，28.6% 和 35.5%。

## 创新点

(1) 针对轨迹跟踪过程中采用常规 MPC 算法在外部干扰或参数不确定性影响下控制性能较差的问题, 提出了基于 Tube MPC 算法的上下分层协同控制策略, 在上层 Tube MPC 控制器中, 将外部干扰和部分参数不确定性考虑为横、纵向动力学的附加扰动, 结合加速度约束的点质量模型构建状态空间方程, 最终输出考虑扰动后的期望加速度信号。该控制策略能够有效的抑制扰动影响。

(2) 针对不变集构建过程中因对扰动边界预估不准确而存在的保守性问题, 本文提出的基于 NLMS 和 QE 结合的 RUMI 不确定性在线辨识方法, 通过仿真实验表明, 该方法能精确辨识扰动边界值, 相比于高斯过程回归和贝叶斯线性回归, 对高斯噪声和非高斯噪声两种形式的扰动信号均有较好的辨识效果, 且计算速度更快。

## 研究展望

(1) 本文所提出的 RUMI 扰动边界在线辨识方法聚焦于提升对不同类型扰动信号辨识的通用性以及计算效率, 却没有兼顾到收敛速度, 导致收敛速度相比于高斯过程回归和贝叶斯线性回归偏慢, 未来可针对该问题进一步深入研究。

(2) 由于高速行驶的实车实验具有较大的危险性, 可能造成人力和物力的极大损失, 因此本文仅通过联合仿真实验平台对所提出的上层轨迹跟踪鲁棒模型预测控制策略以及下层的加速度跟踪控制策略进行了其有效性的初步验证。但智能驾驶技术必须经过实车实验的考验才能真正地应用于智能汽车的开发, 因此后续需要逐步与硬件结合验证算法的可靠性。

## 参考文献

- [1] 张涛, 李清, 张长水, 等. 智能无人自主系统的发展趋势[J]. 无人系统技术, 2018, 1(01): 11-22.
- [2] 何慧玲. 智能车辆局部轨迹规划与跟踪控制算法研究[D]. 哈尔滨: 哈尔滨工业大学, 2021.
- [3] 曾宇. 基于车联网的高速公路安全预警系统设计[D]. 西安: 长安大学, 2017.
- [4] Indri M, Grau A, Ruderman M. Guest editorial special section on recent trends and developments in industry 4.0 motivated robotic solutions[J]. IEEE Transactions on Industrial Informatics, 2018, 14(4): 1677-1680.
- [5] 李克强, 戴一凡, 李升波, 等. 智能网联汽车 (ICV) 技术的发展现状及趋势[J]. 汽车安全与节能学报, 2017, 8(01): 1.
- [6] 朱宏任. 实施中国制造 2025 战略促进汽车智能化发展[J]. 时代汽车, 2015 (10): 27-30.
- [7] 张智能. 复杂动态环境下智能汽车局部路径规划与跟踪[D]. 重庆: 重庆大学, 2019.
- [8] 吕品, 李凯, 许嘉, 等. 无人驾驶汽车协同感知信息传输负载优化技术[J]. 计算机学报, 2021, 44(10).
- [9] Liu F, Lu Z, Lin X. Vision-based environmental perception for autonomous driving[J]. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 2022: 09544070231203059.
- [10] Chen Q, Xie Y, Guo S, et al. Sensing system of environmental perception technologies for driverless vehicle: A review of state of the art and challenges[J]. Sensors and Actuators A: Physical, 2021, 319: 112566.
- [11] Velasco-Hernandez G, Barry J, Walsh J. Autonomous driving architectures, perception and data fusion: A review[C]//2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP). IEEE, 2020: 315-321.
- [12] Wang W, Qie T, Yang C, et al. An intelligent lane-changing behavior prediction and decision-making strategy for an autonomous vehicle[J]. IEEE transactions on industrial electronics, 2021, 69(3): 2927-2937.
- [13] Hang P, Lv C, Huang C, et al. An integrated framework of decision making and motion planning for autonomous vehicles considering social behaviors[J]. IEEE transactions on vehicular technology, 2020, 69(12): 14458-14469.
- [14] Huang C, Huang H, Hang P, et al. Personalized trajectory planning and control of lane-change

- maneuvers for autonomous driving[J]. IEEE Transactions on Vehicular Technology, 2021, 70(6): 5511-5523.
- [15] Song H, Ding W, Chen Y, et al. Pip: Planning-informed trajectory prediction for autonomous driving[C]//Computer Vision - ECCV 2020: 16th European Conference, Glasgow, UK, August 23 - 28, 2020, Proceedings, Part XXI 16. Springer International Publishing, 2020: 598-614.
- [16] 陈慧, 王洁新. 基于驾驶人不满度的高速公路自动驾驶换道决策[J]. 中国公路学报, 2019, 32(12): 1-9.
- [17] 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(1): 110-117.
- [18] 刘一帆, 施光林, 陈耀峰, 等. 基于纯跟踪模型的模糊路径跟踪控制方法[J]. 机械设计与研究, 2022.
- [19] 付景枝, 尹泽凡, 刘云平, 等. 基于改进纯跟踪算法的无人车路径跟踪研究[J]. 机械设计, 2022, 39(S2): 41-45.
- [20] Thrun S, Montemerlo M, Dahlkamp H, et al. Stanley: The robot that won the DARPA Grand Challenge[J]. Journal of field Robotics, 2006, 23(9): 661-692.
- [21] 王鑫, 凌铭, 饶启鹏, 等. 基于改进 Stanley 算法的无人车路径跟踪融合算法研究[J]. 汽车技术, 2022, (07): 25-31.
- [22] Levinson J, Askeland J, Becker J, et al. Towards fully autonomous driving: Systems and algorithms[C]//2011 IEEE intelligent vehicles symposium (IV). IEEE, 2011: 163-168.
- [23] Zhang X, Zhu X. Autonomous path tracking control of intelligent electric vehicles based on lane detection and optimal preview method[J]. Expert Systems with Applications, 2019, 121: 38-48.
- [24] Park M, Kang Y. Experimental verification of a drift controller for autonomous vehicle tracking: A circular trajectory using LQR method[J]. International Journal of Control, Automation and Systems, 2021, 19(1): 404-416.
- [25] Meng Y, Gan X, Wang Y, et al. LQR-GA controller for articulated dump truck path tracking system[J]. Journal of Shanghai Jiaotong University (Science), 2019, 24: 78-85.
- [26] 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(2-4): 265-291.
- [27] Falcone P, Borrelli F, Asgari J, et al. Predictive active steering control for autonomous vehicle systems[J]. IEEE Transactions on control systems technology, 2007, 15(3): 566-580.

- [28] Falcone P, Eric Tseng H, Borrelli F, et al. MPC-based yaw and lateral stabilisation via active front steering and braking[J]. *Vehicle System Dynamics*, 2008, 46(S1): 611-628.
- [29] Falcone P, Borrelli F, Tseng H E, et al. Linear time - varying model predictive control and its application to active steering systems: Stability analysis and experimental validation[J]. *International Journal of Robust and Nonlinear Control: IFAC - Affiliated Journal*, 2008, 18(8): 862-875.
- [30] Chu D, Li H, Zhao C, et al. Trajectory tracking of autonomous vehicle based on model predictive control with PID feedback[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2022, 24(2): 2239-2250.
- [31] Jeong Y, Yim S. Model predictive control-based integrated path tracking and velocity control for autonomous vehicle with four-wheel independent steering and driving[J]. *Electronics*, 2021, 10(22): 2812.
- [32] Yoon Y, Shin J, Kim H J, et al. Model-predictive active steering and obstacle avoidance for autonomous ground vehicles[J]. *Control Engineering Practice*, 2009, 17(7): 741-750.
- [33] Zhai L, Wang C, Hou Y, et al. MPC-based integrated control of trajectory tracking and handling stability for intelligent driving vehicle driven by four hub motor[J]. *IEEE transactions on vehicular technology*, 2022, 71(3): 2668-2680.
- [34] 史先鹏, 刘士荣. 机械臂轨迹跟踪控制研究进展[J]. *控制工程*, 2011, 18(1): 116-122.
- [35] 曹凯, 于善义, 于少伟. 基于多隐马尔可夫模型的车辆机动行为识别与预测[J]. *信息与控制*, 2014, 43(4): 506-512.
- [36] 王秋, 曲婷, 陈虹. 基于随机模型预测控制的自主车辆转向控制[J]. *信息与控制*, 2015, 44(04): 499-506.
- [37] 方培元, 熊璐, 冷搏, 等. 基于多约束随机模型预测控制的无人车运动规划与控制[J]. *同济大学学报(自然科学版)*, 2022, 50(S1): 128-134.
- [38] Carvalho A, Gao Y, Lefevre S, et al. Stochastic predictive control of autonomous vehicles in uncertain environments[C]//12th international symposium on advanced vehicle control. 2014, 9.
- [39] 孙中奇. 轮式机器人跟踪控制研究: 模型预测控制方法[D]. 北京: 北京理工大学, 2018.
- [40] Peng H, Wang W, An Q, et al. Path tracking and direct yaw moment coordinated control based on robust MPC with the finite time horizon for autonomous independent-drive vehicles[J]. *IEEE Transactions on Vehicular Technology*, 2020, 69(6): 6053-6066.

- [41] Mata S, Zubizarreta A, Pinto C. Robust tube-based model predictive control for lateral path tracking[J]. IEEE Transactions on Intelligent Vehicles, 2019, 4(4): 569-577.
- [42] Rathai K M M, Amirthalingam J, Jayaraman B. Robust tube-MPC based lane keeping system for autonomous driving vehicles[C]//Proceedings of the 2017 3rd International Conference on Advances in Robotics. 2017: 1-6.
- [43] Wischnewski A, Euler M, Gümüs S, et al. Tube model predictive control for an autonomous race car[J]. Vehicle System Dynamics, 2022, 60(9): 3151-3173.
- [44] Alcalá E, Puig V, Quevedo J, et al. Fast zonotope - tube - based LPV - MPC for autonomous vehicles[J]. IET Control Theory & Applications, 2020, 14(20): 3676-3685.
- [45] 李松. 智能汽车轨迹跟踪控制算法研究综述[J]. Automotive Digest, 2023 (9).
- [46] Polack P, Altché F, d'Andréa-Novel B, et al. The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?[C]//2017 IEEE intelligent vehicles symposium (IV). IEEE, 2017: 812-818.
- [47] Heilmeier A, Wischnewski A, Hermansdorfer L, et al. Minimum curvature trajectory planning and control for an autonomous race car[J]. Vehicle System Dynamics, 2019.
- [48] Betz J, Wischnewski A, Heilmeier A, et al. A software architecture for the dynamic path planning of an autonomous racecar at the limits of handling[C]//2019 IEEE international conference on connected vehicles and expo (ICCVE). IEEE, 2019: 1-8.
- [49] Subosits J K, Gerdés J C. From the racetrack to the road: Real-time trajectory replanning for autonomous driving[J]. IEEE Transactions on Intelligent Vehicles, 2019, 4(2): 309-320.
- [50] 滕飞. 紧急避让工况智能车轨迹跟踪控制算法研究[D]. 长春: 吉林大学, 2022.
- [51] Yu J, Guo X, Pei X, et al. Path tracking control based on tube MPC and time delay motion prediction[J]. IET Intelligent Transport Systems, 2020, 14(1): 1-12.
- [52] Betz J, Heilmeier A, Wischnewski A, et al. Autonomous driving—a crash explained in detail[J]. Applied Sciences, 2019, 9(23): 5126.
- [53] Wischnewski A, Betz J, Lohmann B. Real-time learning of non-Gaussian uncertainty models for autonomous racing[C]//2020 59th IEEE Conference on Decision and Control (CDC). IEEE, 2020: 609-615.
- [54] Ciochină S, Paleologu C, Benesty J. An optimized NLMS algorithm for system identification[J]. Signal Processing, 2016, 118: 115-121.

- [55] Yazidi A, Hammer H. Multiplicative update methods for incremental quantile estimation[J]. IEEE transactions on cybernetics, 2017, 49(3): 746-756.
- [56] Murphy K P. Machine learning: a probabilistic perspective[M]. MIT press, 2012.
- [57] Koller T, Berkenkamp F, Turchetta M, et al. Learning-based model predictive control for safe exploration[C]//2018 IEEE conference on decision and control (CDC). IEEE, 2018: 6059-6066.
- [58] McKinnon C D, Schoellig A P. Learning probabilistic models for safe predictive control in unknown environments[C]//2019 18th European Control Conference (ECC). IEEE, 2019: 2472-2479.
- [59] 张昊, 魏超, 胡纪滨, 等. 基于 Ribbon 模型的鲁棒 MPC 路径跟踪控制方法[C]//中国汽车工程学会 (China Society of Automotive Engineers) . 2023 中国汽车工程学会年会论文集 (1) . 北京理工大学机械与车辆学院; 北方车辆研究所; ,2023:9.
- [60] Kabzan J, Hewing L, Liniger A, et al. Learning-based model predictive control for autonomous racing[J]. IEEE Robotics and Automation Letters, 2019, 4(4): 3363-3370.
- [61] Williams C K I, Rasmussen C E. Gaussian processes for machine learning[M]. Cambridge, MA: MIT press, 2006.
- [62] Yu S, Maier C, Chen H, et al. Tube MPC scheme based on robust control invariant set with application to Lipschitz nonlinear systems[J]. Systems & Control Letters, 2013, 62(2): 194-200.
- [63] Blanchini F. Set invariance in control[J]. Automatica, 1999, 35(11): 1747-1767.
- [64] Mayne D Q, Langson W. Robustifying model predictive control of constrained linear systems[J]. Electronics letters, 2001, 37(23): 1422-1423.
- [65] 孙传扬. 智能汽车紧急避撞轨迹规划与路径跟踪控制策略研究[D]. 北京: 北京交通大学, 2021.
- [66] Rakovic S V, Kerrigan E C, Kouramas K I, et al. Invariant approximations of the minimal robust positively invariant set[J]. IEEE Transactions on automatic control, 2005, 50(3): 406-410.
- [67] Gonzalez R, Fiacchini M, Alamo T, et al. Online robust tube-based MPC for time-varying systems: A practical approach[J]. International Journal of Control, 2011, 84(6): 1157-1170.
- [68] Gritzmann P, Sturmfels B. Minkowski addition of polytopes: computational complexity and applications to Gröbner bases[J]. SIAM Journal on Discrete Mathematics, 1993, 6(2): 246-269.
- [69] Bekker H, Brink A. Reducing the time complexity of Minkowski-sum based similarity calculations by using geometric inequalities[C]//Computational Science and Its Applications – ICCSA 2004:

- International Conference, Assisi, Italy, May 14-17, 2004, Proceedings, Part III 4. Springer Berlin Heidelberg, 2004: 32-41.
- [70] Kurzhanski A B. Ellipsoidal calculus for estimation and feedback control[M]//Systems and Control in the Twenty-first Century. Boston, MA: Birkhäuser Boston, 1997: 229-243.
- [71] van Hessem D H, Bosgra O H. Closed-loop stochastic dynamic process optimization under input and state constraints[C]//Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301). IEEE, 2002, 3: 2023-2028.
- [72] Wischnewski A, Herrmann T, Werner F, et al. A tube-MPC approach to autonomous multi-vehicle racing on high-speed ovals[J]. IEEE Transactions on Intelligent Vehicles, 2022, 8(1): 368-378.

## 攻读学位期间发表论文与研究成果清单

### 发表论文

- [1] H. Ren, **Y. Li**, L. Wang, L. Yang. Lateral Path Tracking Control Based on Tube MPC. 2024 Ineternational Symposium on Intelligent Unmanned System and Artificial Intelligence, SIUSAI2024.  
(EI 检索, 已录用, 导师一作, 本人二作)

### 参与项目

- [1] 2021.01-2023.12 国家自然青年基金项目: 复杂环境下驾驶行为预测机理与无人车运动控制方法研究。主要参与工作: CAN 通信低速无人小车的运动控制模块开发与维护。
- [2] 2022.01-2023.12 校际合作项目: 基于在线学习的无人车快速避障与运动控制研究。主要参与工作: 基于 Tube MPC 及扰动在线辨识的轨迹跟踪控制算法研究。
- [3] 2022.10-2025.10 军科委基础加强项目: 高敏捷运动规划与控制。主要参与工作: 基于模型参考自适应控制的延迟优化算法研究。

## 致谢

三年的硕士研究生生涯即将结束，首先由衷感谢导师任宏斌老师，本论文在选题、理论和实验研究等各个工作阶段均得到了导师极具价值的指导和建议，任老师对论文的撰写和修改提出了许多宝贵的建议，给予了我莫大的帮助；任老师渊博的学术知识、宽广的学术视野、一针见血的学术见地和一丝不苟的学术态度使我受益匪浅；此外，在科研学习工作之外，任老师平易近人、风趣幽默，在生活方面也给予学生很多实用的建议，对学生无微不至的关怀令人感到十分温暖。在此，谨向任老师表示最诚挚的谢意。

在三年研究生课程学习和科研工作期间，401 实验室就像一个大家庭。实验室在吴志成老师、赵玉壮老师、齐志权老师、杨林老师以及任宏斌老师等各位老师的带领下蒸蒸日上，他们对 401 的每一位同学都有着耐心的学术指导和严格的要求。实验室在各位老师的指导下营造了优良的学术氛围，经常能够看到实验室里老师与同学们进行学术探讨，即使是在深夜也经常能够看到各位老师和同学们忙碌的身影，在如此优秀的学术氛围中，我也收获了许多。在此，由衷地对 401 实验室的各位老师表示感谢。

科研探索的日子同样离不开实验室的各位同学的支持和帮助。感谢周孝添师兄、周高立师兄、吴世南师兄、郭超然师兄、王捍铭师兄、朱晶师兄等各位师兄在科研工作和学习生活中对我的指导。感谢同级的孙纪禹、赵嘉豪、刘罗航、马小涵、赵成祥等同学，在学术和生活中我们互相帮助共同成长。感谢赵师源、李雨浓、谢盼盼、李锦、牛雅琪、侯龙波等各位师弟师妹的帮助。感谢温暖的 401 实验室全体同学。感谢舍友任校辰、黄自晓、马俊玖，良好的宿舍氛围让我有家的感觉，与你们一起度过的大学生活是我宝贵的回忆。

多年的求学生涯离不开父母的支持和理解，感谢我的父母在学业和生活中一如既往地支持我，鼓励我求学，让我没有后顾之忧。感谢我的女友王奇在我的生活和学习中对我的帮助和鼓励，无论高峰还是低谷，无论喜悦还是失落，虽然身处异地，但永远能感受到你的温暖和陪伴，是我积极面对生活的最大动力。

最后，衷心感谢百忙之中评阅本文的各位专家学者。