

# 无人驾驶车辆路径跟踪模型预测方法研究

谢兼明

2018 年 6 月

中图分类号:

UDC 分类号:

## 无人驾驶车辆路径跟踪模型预测方法研究

作 者 姓 名	<u>谢兼明</u>
学 院 名 称	<u>机械与车辆学院</u>
指 导 教 师	<u>龚建伟教授</u>
答辩委员会主席	<u>闫清东教授</u>
申 请 学 位	<u>工学硕士</u>
学 科 专 业	<u>机械工程</u>
学位授予单位	<u>北京理工大学</u>
论文答辩日期	<u>2018 年 6 月</u>

# **Research on model predictive control for path tracking of Autonomous Vehicle**

Candidate Name:	<u>Jianming Xie</u>
School or Department:	<u>Mechanical Engineering</u>
Faculty Mentor:	<u>Prof. Jianwei Gong</u>
Chair, Thesis Committee:	<u>Prof. Qingdong Yan</u>
Degree Applied:	<u>Master of Science</u>
Major:	<u>Mechanical Engineering</u>
Degree by:	<u>Beijing Institute of Technology</u>
The Date of Defence:	<u>June, 2018</u>

# 无人驾驶车辆路径跟踪模型预测方法研究

北京理工大学

## 研究成果声明

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

特此申明。

签 名： 日期：

## 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： 日期：  
导师签名： 日期：

## 摘要

无人驾驶车辆运动控制是实现自动驾驶功能的基础，同时也是无人驾驶车辆平台的核心技术之一，有着重要的研究意义。近年来，以模型预测控制 (Model Predictive Control, MPC) 为代表的最优控制方法在无人驾驶车辆运动控制领域中得到广泛的研究和发展，已有许多仿真环境及实车试验验证了其作为轨迹跟踪算法的有效性。但目前很多研究未充分考虑算法对环境的适应性，以及控制对象车辆平台的执行机构及系统特点。针对实际应用需求，本文主要解决以下两个问题：一是环境适应性方面，提高 MPC 控制算法对道路轨迹曲率变化的适应性；二是车辆平台执行机构方面，主要解决车辆平台系统转向执行机构时延对 MPC 控制路径跟踪控制算法的影响。

针对路径曲率变化对跟踪算法的影响，本文提出一种考虑道路曲率变化的模型预测控制路径跟踪算法，此算法以后轴中心到参考路径的距离、车辆航向、轨迹曲率、参考路径航向和参考路径曲率为状态量，以前轮偏角作为控制量建立预测模型，在此基础上建立横向偏差和航向偏差限制的目标函数，同时增加道路几何约束和转向机构物理限制控制量约束，进而用二次规划算法求解带约束的目标函数。

针对车辆平台转向机构时延对控制效果的影响问题，本文先对通过下发一系列的期望前轮控制量，并对反馈的实际前轮偏角，获取时延大小的范围。由时延时间在实际转向过程中不是一个定值，这将给系统模型带来不确定性。针对这种带有不确定性的控制问题，其解决办法基本步骤如下：首先，构造参数时变多面体系统，其次求解参数时变多面体不确定系统的稳定性状态反馈控制率，然后结合原系统约束和反馈控制律构建鲁棒正向不变集作为参数不确定性系统的约束，最后求解时变多面体系统的滚动时域优化问题。

本文搭建 MATLAB-CarSim 联合仿真平台，设计双移线的仿真场景，完成给定参考轨迹下的轨迹跟踪控制测试，结果表明所设计算法的有效性和稳定性，能有效适用双移线这种曲率变化的道路。

最终，在无人驾驶车辆平台上和越野场景中验证了此算法的可行性、稳定性，取得了较好的实际效果。试验证明本文提出的算法能适应各种曲率的道路环境，并在增加时延补偿后，可以有效减少横向跟踪误差。

**关键词：** 无人驾驶车辆； 轨迹跟踪； 模型预测控制； 时延；

## Abstract

Motion control of intelligent vehicle is the basis of autonomous driving, and it is also one of the core technologies of intelligent vehicle platform. It has important research significance. In recent years, the optimal control method represented by model predictive control has been widely researched and developed in the field of intelligent vehicle's motion control. Many simulation environments and vehicle tests have verified its effectiveness as a trajectory tracking algorithm. However, many studies currently do not fully consider the adaptability of the algorithm to the environment, as well as the implementation mechanism and system characteristics of the control object vehicle platform. For practical application needs, this paper mainly addresses the following two issues: First, adaptability of the environment, improve the adaptability of the MPC control algorithm to changes in curvature of the road trajectory, and secondly, implement of the vehicle platform, mainly to solve the time delay of vehicle platform steering actuator system.

Aiming at the influence of path curvature variation on tracking algorithm, this paper proposes a model predictive control path tracking algorithm considering the change of road curvature. In this algorithm, the distance between the center axis and the reference path, vehicle pose, reference path heading and vehicle heading difference construct the state quantity, the steering angle is used as a control quantity to establish a prediction model, and on this basis, the objective function of lateral deviation and heading deviation limitation is established. At the same time, the geometric constraints of the road and the physical constraints of the steering mechanism are limited, and then the quadratic programming algorithm is used, then solve the constrained objective function.

Aiming at the influence of the delay of the steering mechanism of the vehicle platform, this paper first obtains the range of the delay size by sending a series of expected controls and comparing the actual front wheel deflection of the feedback. The delay time is not a fixed value in the actual steering process, which will bring uncertainty to the system model. In order to solve the uncertainty problem of the control, the basic steps of the solution are as follows. First, construct a time-varying polyhedral system. Secondly, achieve the stability state feedback control law of the time-varying polyhedral system, and then combine the original system constraints and feedback control laws to build robust positive invariant sets as the constraints of the parameter uncertainty system, and finally solve the rolling time domain optimization problem of time-varying polyhedral systems.

In this paper, a MATLAB-CarSim co-simulation platform is set up to design a double-



shift simulation scenario, and the trajectory tracking control test under a given reference trajectory is completed. The results show the effectiveness and stability of the designed algorithm and can be effectively applied to double-shift lines that the curvature of path is changing.

Finally, the feasibility and stability of this algorithm were verified on the smart vehicle platform and off-road scenarios, and good practical results were achieved. The experiment proves that the algorithm proposed in this paper can adapt to the road environment with various curvatures, and effectively reduce the lateral tracking error after adding delay compensation.

**Key words:** Intelligent vehicle; trajectory tracking; model predictive control; time-delay;

## 目录

第 1 章 绪论 .....	1
1.1 本论文研究背景和意义 .....	1
1.2 国内外研究现状 .....	2
1.2.1 无人驾驶技术的发展 .....	2
1.2.2 无人驾驶车辆运动控制研究现状 .....	5
1.2.3 模型预测控制研究现状 .....	6
1.3 课题研究内容 .....	8
第 2 章 模型预测控制理论基础 .....	9
2.1 模型预测控制理论 .....	9
2.2 非线性模型预测控制算法 .....	10
2.2.1 问题描述 .....	10
2.2.2 非线性模型预测控制的数值解法 .....	11
2.3 线性模型预测控制 .....	13
2.4 本章小结 .....	18
第 3 章 自适应采样周期的模型预测路径跟踪 .....	19
3.1 未考虑道路参数的模型预测控制算法 .....	19
3.1.1 未考虑道路参数的模型预测建模 .....	19
3.1.2 固定采样周期 .....	22
3.2 基于运动学的预测模型 .....	24
3.2.1 参考路径拟合 .....	24
3.2.2 预测模型 .....	25
3.2.3 模型线性化 .....	26
3.2.4 系统输出 .....	27
3.2.5 系统模型离散化 .....	28
3.3 系统约束 .....	30
3.4 代价函数 .....	31
3.5 本章小结 .....	32
第 4 章 模型预测控制时延处理 .....	34
4.1 问题描述 .....	34
4.2 加入延时的增广预测模型 .....	36
4.3 带延时的控制策略 .....	37
4.3.1 无约束情况下的稳定控制律 .....	37
4.3.2 最大输出允许集合 .....	38
4.3.3 控制策略 .....	39
4.4 本章小结 .....	39

---

第 5 章 仿真与实车验证 .....	40
5.1 MPC 仿真实验 .....	40
5.1.1 MPC 仿真实验 .....	40
5.1.2 仿真实验结果分析 .....	52
5.2 实车实验 .....	53
5.2.1 实验内容 .....	53
5.2.2 试验方法和步骤 .....	54
5.2.3 误差计算 .....	55
5.2.4 实验结果 .....	56
5.3 实验结论 .....	62
总结与展望 .....	63
参考文献 .....	65
附录 .....	69
攻读学位期间发表论文与研究成果清单 .....	71
致谢 .....	72

## 第1章 绪论

### 1.1 本论文研究背景和意义

随着社会的进步和国民经济的高速发展,世界各国汽车保有量在不断增加。汽车工业在给全世界人民带来了巨大便利的同时也带来了诸如交通拥堵、交通事故频发等问题。据世界卫生组织统计,每年有约 127 万人死于道路交通碰撞事故,2000 万至 5000 万人因被碰撞受到非致命伤害<sup>[1]</sup>。而导致交通事故的重要原因就是驾驶员注意力不集中或者驾驶经验不足等人为因素<sup>[2]</sup>。为解决这一问题,根本的方法就是从传统的人-车-路闭环控制系统中将“人”的因素淡化,发展具有辅助驾驶或者自动驾驶功能的无人驾驶车辆。相对于驾驶员操纵汽车,无人驾驶车辆的驾驶行为具有可预测性,同时对环境的反应也更加迅速。这将提高交通系统的安全性,同时改善交通系统的运输效率。无人驾驶车辆技术将是现代交通系统以及未来智能交通系统的核心技术之一。

无人驾驶车辆运动控制是实现自动驾驶功能的基础,同时也是无人驾驶车辆平台的核心技术之一,有着重要的研究意义。近年来,以模型预测控制(Model Predictive Control, MPC)为代表的最优控制方法在无人驾驶车辆运动控制领域中得到广泛的研究和发展,已有许多仿真环境及实车试验验证了其作为轨迹跟踪算法的有效性。但目前很多研究未充分考虑算法对环境的适应性,以及控制对象车辆平台的执行机构及系统特点。针对实际应用需求,本文主要解决以下两个问题:一是环境适应性方面,提高 MPC 控制算法对道路轨迹曲率变化的适应性;二是车辆平台执行机构方面,主要解决车辆平台系统转向执行机构时延对 MPC 控制路径跟踪控制算法的影响。

常用的运动学模型预测算法都是基于车辆模型,但建模过程中并未考虑道路的曲率形状,从而导致较大跟踪误差。因此,本文针对此问题提出一种适用各种道路环境的模型预测控制算法。首先,为了获取道路参数,我们首先对参考路径进行多项式拟合,以后轴中心到参考路径的距离、车辆位姿、车辆与参考轨迹的航向偏差,其中航向偏差和横向偏差可以表示成参考路径相关参数的形式,以前轮偏角作为控制量建立预测模型,在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束,进而用二次规划算法求解带约束的目标函数。此外,单一的采样周期很难适应所有工况,采样周期与预测时域和速度的乘积类似于前向预测距离,在曲率较大的情况,前向预测距离比较短,即前视距离比较短,而在直线情况下,预测距离应该比较长,即前视距离比较长,针对此规律,本文设计一种随道路曲率的自适应采样周

期的方法。

由于跟踪系统受客观条件时延的制约,使得所建立系统模型的参数跟实际系统参数之间存在着一定的偏差,这势必会影响系统的稳定性等问题。针对车辆平台转向机构时延对控制效果的影响问题,本文先对通过下发一系列的期望前轮控制量,并对比反馈的实际前轮偏角,获取时延大小的范围。由时延时间在实际转向过程中不是一个定值,这将给系统模型带来不确定性。针对这种带有不确定性的控制问题,其解决办法基本步骤如下:首先,构造参数时变多面体系统,其次求解参数时变多面体不确定系统的稳定性状态反馈控制率,然后结合原系统约束和反馈控制律构建鲁棒正向不变集作为参数不确定性系统的约束,最后求解时变多面体系统的滚动时域优化问题。

## 1.2 国内外研究现状

### 1.2.1 无人驾驶技术的发展

无人驾驶技术的发展主要分为三个阶段:起步阶段、发展阶段、百花齐放阶段。同时,为了更好的发展无人驾驶技术,各国对该技术的发展阶段制定了相应的等级划分规则。下面将从国外与国内两部分介绍各个阶段无人驾驶技术的发展与革新。

#### 1.2.1.1 国外无人驾驶技术的发展

国外无人驾驶车辆的研究具有悠久的历史,比较有影响力的有美国的 PATH (Partners for Advanced Transit and Highways) 计划,美国国防部主导的 DEMO 系列无人车<sup>[3]</sup>以及欧洲的 Eureka-PROMETHEUS 项目<sup>[4]</sup>等。80 年代,梅赛德斯奔驰公司和德国慕尼黑联邦国防军大学合作研制了 VaMoRs 无人驾驶汽车<sup>[5]</sup>,该实验车配备了 2 个单目摄像头和相关计算机设备,不需要路面电线和磁条等引导信号,仅依靠视觉感知信息,可在实际街道上实现低速无人驾驶。

DARPA (Defense Advanced Research Projects Agency) 举办的无人驾驶系列挑战赛,极大地推动了无人驾驶技术的发展<sup>[6,7]</sup>。其中,卡耐基梅隆大学 Tartan Racing 车队的 BOSS<sup>[8]</sup>与斯坦福大学 Stanford Racing 车队的 Junior<sup>[9]</sup>无人驾驶车辆表现尤为突出,分别在两次比赛中夺冠。BOSS 和 Junior 装配的硬件部件与软件模块有所差别,但其采用的无人驾驶框架较为相似,这套框架广泛运用于无人驾驶系统研发中。该框架主要包括规划模块(行为决策和路径选择),地图模块(利用传感器信息进行环境建模),定位模块(结合 GPS 信息和环境信息进行高精度定位)和控制模块(执行上层横纵向控制指令)。DARPA 举办的无人驾驶系列比赛使无人驾驶技术进入公众视野,并为无人驾驶技术的发展积累了大量的人才与经验。



图 1.1 DARPA 比赛中夺冠的无人车

2009 年，谷歌公司旗下的 Google X 实验室开始研发全无人驾驶汽车。该无人驾驶汽车搭载的传感器主要包括激光雷达、超声波雷达、单目摄像头、红外摄像以及 GPS/INS 系统<sup>[10]</sup>。使用这些传感器，谷歌无人车实现了对障碍物、行人、车道线交通标志等物体的有效识别，并进一步完成了躲避障碍物、车道线保持以及车辆定位等功能。Google 公司研发的无人车从 2010 年开始上路测试，至今已经行驶了 200 多万公里，遍布美国的四个城市。2014 年 10 月 Google 发布了自己的新车型<sup>[11]</sup>，如图 1.2 所示，该原型车没有刹车和方向盘，甚至没有油门，只有一个用于启动汽车的按键，启动后所有驾驶行为完全由车辆自主完成。该车型于 2015 年在山景城开始进行路测。截至目前，Google 已经有 24 辆雷克萨斯 SUV 和 34 辆原型车在路上进行测试，每周测试里程最多可达到 2.6 万英里。



图 1.2 谷歌无人车

特斯拉公司是在无人车商业化进程中走在最前列的公司之一。与 Google 无人车不同的是，特斯拉无人车不是真正意义上的无人驾驶，而是高级自动驾驶（辅助驾驶），更多地依靠摄像头、毫米波雷达等传感器进行车道保持、变道等功能<sup>[12]</sup>。

2013 年，戴姆勒奔驰公司采用接近量产的传感器配置沿着 Bertha Benz 的奔驰开创之路无人驾驶了 103 千米，路途中包含了城市，城郊和乡村道路，环境复杂多变，

虽然并非全程无人驾驶，但其较为低价的传感器配置（不包含激光雷达）为以后的无人驾驶传感器方案提供了很好的思路<sup>[13]</sup>。

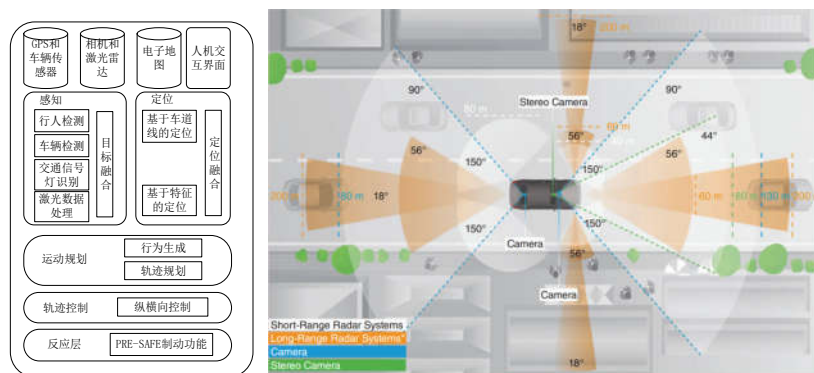


图 1.3 奔驰自动驾驶软件架构及传感器布置

### 1.2.1.2 国内无人驾驶技术的发展

我国的无人驾驶技术研究起步较晚，但发展势头较为迅猛。我国无人驾驶技术的研究起始于 1986 年的高技术研究发展计划，该计划制定了以智能机器人为主题的总体战略指向。1994 年 10 月清华大学研制了我国第一台“室外智能移动机器人 THMR-III”<sup>[14]</sup>，同期，国防科技大学，北京理工大学，南京理工大学等六所国内高校联合承担了“地面军用智能机器人”的研究项目，选用国产跃进客货车进行无人驾驶改造<sup>[15]</sup>。国防科学技术大学从上世纪八十年代开始先后研制无人驾驶车辆自动驾驶系统。

国家自然科学基金委于 2009 年至 2017 年期间举办了 8 届“中国智能车未来挑战赛”。国防科技大学<sup>[16]</sup>、北京理工大学<sup>[17]</sup>、清华大学<sup>[18]</sup>、上海交通大学<sup>[19]</sup>等多家院校的相关研究所均参加了该系列比赛，极大地促进了国内无人驾驶的发展，为国内相关企业和研究所培养了大量的相关人才。中国军方也极为重视无人驾驶技术的发展，分别于 2014 年和 2016 年举行了两届“跨越险阻”地面无人系统挑战赛。

除此之外，国内的互联网企业如百度、华为、腾讯等也相继成立无人驾驶部门。汽车企业如北汽、上汽、二汽、吉利、广汽等也积极与高校研究。百度公司于 2013 年启动无人驾驶汽车项目，其技术核心是包括高精度地图、定位、感知、智能决策与控制四个主要模块的“百度汽车大脑”。百度公司于 2015 年 12 月宣布，其无人驾驶汽车实现国内首次城市道路、郊区道路及高速道路混合路况下的全无人驾驶。2016 年百度世界大会无人车分论坛上，百度无人驾驶事业部负责人宣布，美国加州政府为百度发了全球第 15 张无人驾驶汽车测试牌照。





图 1.4 百度无人车

### 1.2.2 无人驾驶车辆运动控制研究现状

运动控制是无人驾驶车辆研究领域的核心技术之一，高效的运动控制系统是无人驾驶车辆实现智能行为的重要保障。轨迹跟踪控制是无人驾驶车辆运动控制的主要研究内容，即根据路径规划系统给出的参考路径及定位定向系统获得的车辆实时位置、车速等信息按照一定的控制算法计算获得油门、刹车、方向盘等执行机构的控制量，使得车辆可以快速地跟踪规划轨迹，同时保证车辆行驶的稳定性和舒适性。无人驾驶车辆的轨迹跟踪控制也是目前国内外相关研究机构的研究热点。在轨迹跟踪控制研究领域，目前应用比较成功的控制算法有：比例积分微分控制（Proportion Integration Differentiation, PID）、模糊控制（Fuzzy Control）、滑模控制（Sliding Mode Control, SMC）、纯跟踪控制（Pure Pursuit）及神经网络控制（NeuralNetworks Control）等。

2004 年，日产公司应用 PID 控制算法搭建了前馈与反馈结合的横向控制器，实现了车速连续变化的情况下保证车辆快速平稳地跟踪参考路径，有效地解决了无人驾驶车辆横向控制的实时性问题。

2006 年，法国的 Netto 等人了解决大曲率转向工况下无人驾驶车辆路径跟踪问题，构建了基于最优路径检测数据的反馈 PID 控制器<sup>[20]</sup>，使得无人驾驶车辆横向控制器能够更好地应对路径曲率变化。

2011 年，西班牙的 J.Perez 等人构建了一种新的无人驾驶车辆横向控制分层结构<sup>[21]</sup>。上层结构为具有四个输入变量和两个输出变量的模糊逻辑控制器，用以产生控制量期望值，下层为经典 PID 控制器，用以实现期望控制量。该结构具有比较好的路径跟踪性能。

2002 年，M.Tomizuka 等人针对无人驾驶车辆路径跟踪问题，分别应用线性反馈控制算法、滑模控制算法及基于前馈补偿的线性反馈控制算法搭建了横向控制器<sup>[22]</sup>。



设计相应实验对三种控制器进行对比分析,实验表明,非线性的滑模控制器具有最好的控制效果。

上述几种控制方法对参数和环境的依赖程度较高,当环境发生较大变化时无法很好地适应新状态下的轨迹跟踪。另外,无人驾驶车辆是一个非完整约束系统,在低速运动过程中主要受到运动学约束及执行机构的约束,高速运动过程中还需要考虑车辆动力学约束,上述几种控制方法无法有效处理各种约束条件。

### 1.2.3 模型预测控制研究现状

模型预测控制(Model Predictive Control, MPC),又称为滚动时域控制(Receding Horizon Control, RHC)<sup>[23]</sup>,是近年来被广泛研究讨论的一种反馈控制策略。模型预测控制最明显的优点是在控制过程中可以增加多种约束。

从上世纪 80 年代初,Cutler 和 Ramaker<sup>[24]</sup>提出模型预测控制的雏形开始,模型预测控制算法在过程控制等相关领域得到了有效的应用。80 年代后期,随着工业预测控制理论的发展,优化问题的可行性得到关注。进入 90 年代,模型预测控制迭代可行性和闭环稳定性等问题得到证明,为其蓬勃发展奠定了基础。本世纪以来,各种类型的模型预测控制算法得到了广泛的研究,主要有鲁棒模型预测控制算法<sup>[25]</sup>、非线性模型预测控制算法<sup>[26]</sup>及混合模型预测控制算法<sup>[27]</sup>。2010 年以来,相关学者纷纷提出了随机模型预测控制算法<sup>[28]</sup>、自适应模型预测控制算法<sup>[29-31]</sup>、分布式模型预测控制算法<sup>[32-35]</sup>以及经济模型预测控制算法<sup>[36]</sup>等新型模型预测控制算法,极大地丰富了模型预测控制算法研究内容。但是这些新型模型预测控制算法大多处于研究初期,实际应用还有一定的困难。

目前在模型预测控制领域,针对线性系统的线性模型预测控制算法是最为成熟也是最为完备的,在工业控制领域得到了比较广泛的应用,其主要优点在于可以满足工业控制实时性需求。无限时域无约束的线性模型预测控制算法等价于线性二次型优化问题(Linear Quadratic Regulation, LQR)<sup>[37]</sup>。对于有限时域无约束模型预测控制算法,可以采用动态规划方法求解优化目标的显示结构。而对于有限时域带约束的模型预测控制算法,可以通过求解李卡提方程,获得终端权重矩阵,用以保证控制系统的闭环稳定性。

随着模型预测控制算法的发展,非线性模型预测控制算法的可行性及稳定性也逐渐得到了证明<sup>[38]</sup>。对于非线性模型预测控制,最精确的模型是机理模型,通常采用对象的非线性微分方程进行描述,但其复杂度较高;另外一种应用比较广泛的模型是线

线性化模型，即将系统模型在平衡点附近进行近似线性化，以获得线性系统状态矩阵。

非线性模型预测控制算法在工业控制领域的应用远低于线性模型预测控制算法。主要原因在于精确的非线性模型较难建立，且现有的非线性优化方法较难满足工业控制实时性的需求。

将模型预测控制算法应用于无人驾驶车辆轨迹跟踪控制，利用其在线滚动优化特性及约束处理能力，能够有效地解决无人驾驶车辆在轨迹跟踪过程中反应迟缓或频繁转向的问题，并处理各种动力学约束条件。由于车辆动力学模型为非线性系统，因此模型预测控制算法在无人驾驶车辆轨迹跟踪控制中的应用主要有两种方式：基于非线性模型应用非线性模型预测控制算法；对非线性模型进行线性化并应用线性模型预测控制算法。

在文献[39]中，Falcone 提出了一种非线性模型预测控制算法，使车辆在满足物理约束的基础上，稳定地沿着既定轨迹行驶，较高的算法复杂度导致该算法在高速运动车辆平台上的应用受到了限制。在文献[40]中，Kuhne 通过对差速轮转向移动机器人的运动学模型的近似线性化，解决了线性模型预测控制问题，但是由于前轮主动转向车辆动力学具有更强的非线性特性，直接将该非线性系统近似线性化为一个线性时不变系统，将大大降低模型的描述精度，进而对控制精度造成较大的影响。但是以上并未考虑参考路径特别是道路曲率对模型预测控制起的影响，也未分析执行时延的影响。

小结：基于模型预测控制的轨迹跟踪算法在国外已经有了广泛而深入的研究，对车辆的建模已经比较成熟，仿真和实验都验证了算法的可行性与可靠性。但还存在主要难点为：（1）根据不同的道路环境，特别是曲率的影响，目前很多方法都难以适应曲率变化剧烈的道路。（2）对执行机构时延问题分析。

针对实际应用需求，本文主要解决以下两个问题：一是环境适应性方面，提高 MPC 控制算法对道路轨迹曲率变化的适应性；二是车辆平台执行机构方面，主要解决车辆平台系统转向执行机构时延对 MPC 控制路径跟踪控制算法的影响。

针对路径曲率变化对跟踪算法的影响，本文提出一种考虑道路曲率变化的模型预测控制路径跟踪算法，此算法以后轴中心到参考路径的距离、车辆航向、参考路径航向和参考路径曲率为状态量，以前轮偏角作为控制量建立预测模型，在此基础上建立横向偏差和航向偏差限制的目标函数，同时增加道路几何约束和转向机构物理限制控制量约束，进而用二次规划算法求解带约束的目标函数。

针对车辆平台转向机构时延对控制效果的影响问题，本文先对通过下发一系列的期望前轮控制量，并对比反馈的实际前轮偏角，获取时延大小的范围。由时延时间在

实际转向过程中不是一个定值，这将给系统模型带来不确定性。针对这种带有不确定性的控制问题，其解决办法基本步骤如下：首先，构造参数时变多面体系统，其次求解参数时变多面体不确定系统的稳定性状态反馈控制率，然后结合原系统约束和反馈控制律构建鲁棒正向不变集作为参数不确定性系统的约束，最后求解时变多面体系统的滚动时域优化问题。

最后通过仿真和实车试验验证本文提出的方法。

### 1.3 课题研究内容

课题具体工作内容包含 5 个章节，具体内容如下：

第 1 章：绪论。介绍了基于模型预测控制的无人驾驶车辆运动控制的研究目的和意义、国内外相关技术的研究现状和存在的问题，最后介绍了本课题主要的研究工作和论文结构。

第 2 章：模型预测控制理论基础。分析了两种模型预测控制算法的应用范围，分别推导了非线性时变和线性时变模型预测控制的数值解法。

第 3 章：分析一般模型预测控制算法不能适应所有道路工况的原因，并针对此问题提出一种考虑道路特性的模型预测控制算法。首先，为了获取道路参数，我们首先对参考路径进行多项式拟合，以后轴中心到参考路径的距离、车辆位姿、车辆与参考轨迹的航向偏差作为状态量，以前轮偏角作为控制量建立预测模型，在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束，进而用二次规划算法求解带约束的目标函数。此外，单一的采样周期很难适应所有工况，本文设计一种随道路曲率的自适应采样周期的方法。

第 4 章：针对车辆平台转向机构时延对控制效果的影响问题，本文先对通过下发一系列的期望前轮控制量，并对比反馈的实际前轮偏角，获取时延大小的范围。由时延时间在实际转向过程中不是一个定值，这将给系统模型带来不确定性。针对这种带有不确定性的控制问题，其解决办法基本步骤如下：首先，构造参数时变多面体系统，其次求解参数时变多面体不确定系统的稳定性状态反馈控制率，然后结合原系统约束和反馈控制律构建鲁棒正向不变集作为参数不确定性系统的约束，最后求解时变多面体系统的滚动时域优化问题。

第 5 章：本文提出的一种基于运动学模型的非线性模型预测控制算法进行仿真验证和实车验证。

## 第2章 模型预测控制理论基础

模型预测控制算法最早应用于工业过程控制领域，近些年来，由于其理论研究逐渐成熟，应用范围逐渐扩大，无人驾驶车辆运动控制就是其中的一个热点。但应用于智能车辆的模型预测控制算法相比于其他领域，有以下特点：（1）多数采用状态空间模型；（2）需要更高的实时性；（3）计算能力有限，无法进行非常复杂的运算。基于以上特点和需求，有必要对模型预测算法进行深入的分析，解决模型预测控制算法计算量大，实时性不高的问题，保证无人驾驶车辆在运动过程中的稳定与安全。同时，也需要考虑到计算机的计算特性，将模型预测控制问题转换为计算机易于求解的数学问题。

### 2.1 模型预测控制理论

模型预测控制是一类通过滚动求解带约束优化问题的控制方法，有文献也将其称为滚动优化控制<sup>[41]</sup>。根据所采用模型的不同，主要包括动态矩阵控（Dynamic Matrix Control, DMC）、模型算法控制（Model Algorithm Control, MAC）和广义预测控制（Generalized Predictive Control, GPC）等。同时，在现代控制理论中所广泛使用的状态空间模型，同样可以应用于模型预测控制中。模型预测控制的一个显著特点就是能方便的处理控制过程中的各种约<sup>[42]</sup>，包括系统状态约束、输出约束以及控制量约束等。同时，它还具备鲁棒性好，易于处理非线性系统等优点。

虽然模型预测控制的种类多样，实现方式也有所区别，但都是建立在预测模型、滚动优化和反馈校正三项基本原理基础上的。

（1）预测模型：预测模型是模型预测控制的基础，主要功能是根据对象的历史信息和未来输入预测系统未来的输出。预测模型的形式没有严格的限定，状态方程、传递函数这类传统的模型都可以作为预测模型。对于线性稳定系统，阶跃响应、脉冲响应这类非参数模型，也可以直接作为预测模型使用。

（2）滚动优化：模型预测控制通过某一性能指标的最优来确定控制作用，但优化不是一次离线进行，而是反复在线进行的，这就是滚动优化的含义，也是模型预测控制区别于传统最优控制的根本点。

（3）反馈校正：为了防止模型失配或者环境干扰引起控制对理想状态的偏离，在新的采样时刻，首先检测对象的实际输出，并利用这一实时信息对基于模型的预测进行修正，然后再进行新的优化。

基于这三个要素，模型预测控制的基本原理可以用图 2.1 来表示。控制过程中，始终存在一条期望轨迹，如图中红色曲线所示。以时刻  $k$  作为当前时刻，控制器结合当前的测量值和预测模型，预测系统未来一段时域内  $[k, k + N_p]$ （也称为预测时域）系统的输出，如图中黄色曲线所示。通过求解满足目标函数以及各种约束的优化问题，得到在控制时域内  $[k, k + N_c]$  一系列的控制序列，如图中的矩形波所示，将该控制序列的第一个元素作为受控对象的实际控制量。当来到下一个时刻  $k+1$  时，重复上述过程，如此滚动的完成一个个带约束的优化问题，实现对被控对象的持续控制。

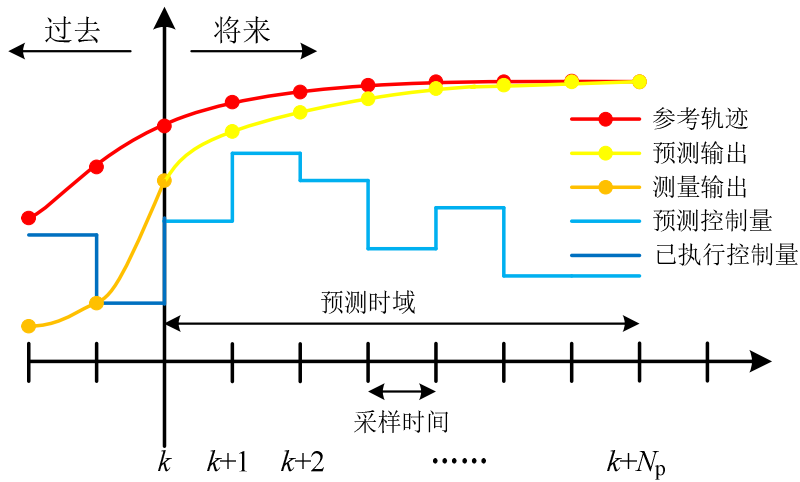


图 2.1 模型预测控制原理示意图

## 2.2 非线性模型预测控制算法

根据第二章的内容可知，车辆的运动学及动力学模型都是非线性的，因此，非线性模型预测控制算法对于无人驾驶车辆的运动控制来说是一种简单直接的控制方式。

### 2.2.1 问题描述

对于一个非线性系统，考虑如下一般形式的离散模型：

$$\begin{aligned} x(t+1) &= f(x(t), u(t)) \\ x(t) &\in \chi, u(t) \in \Gamma \end{aligned} \quad (2.1)$$

式中： $f(\cdot)$  为系统的状态转移函数， $x$  为  $n_s$  维状态变量， $u$  为  $m_c$  维控制变量， $\chi$  为状态变量约束， $\Gamma$  为控制变量约束。

设定  $f(0,0)=0$  为系统的一个稳定点，同时也是系统的控制目标。对于任意的时域  $N$ ，考虑如下的优化目标函数  $J_N(\cdot, \cdot)$ ：

$$J_N(x(t), U(t)) = \sum_{k=t}^{t+N-1} l(x(k), u(k)) + P(x(t+N)) \quad (2.2)$$

式中：  $U(t)=[u(t), \dots, u(t+N-1)]$  是在时域  $N$  内的控制量输入序列，  $\xi(t)$  是在输入向量序列  $U(t)$  作用于系统下的状态向量轨迹，优化目标函数中的第一项  $l(\cdot, \cdot)$  表征对期望输出的跟踪能力，第二项  $P(\cdot, \cdot)$  表征终端约束。

结合式 (2.1) 和式 (2.2)，非线性模型预测控制就是要在每一个步长内求解以下带约束的有限时域优化问题：

$$\min_{U_t, \xi_{t+1}, \dots, \xi_{t+N-1}} J_N(x_t, U_t) \quad (2.3a)$$

$$\text{subject to} \quad x_{k+1,t} = f(x_{k,t}, u_{k,t}), k = t, \dots, N-1 \quad (2.3b)$$

$$x_{k,t} \in \mathcal{X} \quad k = t+1, \dots, t+N-1 \quad (2.3c)$$

$$u_{k,t} \in \mathcal{U} \quad k = t, \dots, t+N-1 \quad (2.3d)$$

$$x_{t,t} = x(t) \quad (2.3e)$$

$$x_{N,t} \in \mathcal{X}_{\text{fin}} \quad (2.3f)$$

其中，式 (2.3b) 为系统所决定的状态约束，式 (2.3c) 和式 (2.3d) 分别为状态向量和控制输入向量约束，式 (2.3e) 为初始状态约束，式 (2.3f) 为终端状态约束。

假定上述优化问题存在可行解，通过求解该问题，可以得到最优控制序列  $U_t^*(t) = [u_{t,t}^*, \dots, u_{t+N-1,t}^*]$ 。根据模型预测控制的原理，只是将该控制序列中的第一个元素作为受控对象的实际控制输入，即：

$$u(x(t)) = u_{t,t}^* \quad (2.4)$$

在下一个采样时刻，系统重新以新的采样时刻为初始状态求解式 (2.3)，并将继续将控制序列的第一个元素施加给受控对象，如此循环，直至完成整个控制过程。对于任意的非线性模型预测问题，在方程求解的整个过程中包含了  $N(n+m)$  个最优变量， $n$  乘以  $N$  个非线性状态约束。同时，也包含了由控制输入约束和状态向量约束决定的一系列线性约束。因此，对于非线性模型预测控制，其求解难度会随着系统状态维度的增加而迅速增加。对于阶次较低的非线性系统，如车辆三自由度运动学模型，非线性模型预测控制的优势依然存在。

## 2.2.2 非线性模型预测控制的数值解法

在非线性模型预测控制中，通过非线性模型、当前的状态量和控制时域内的控制量序列对未来的状态量进行预测。本文介绍用内点法求解方式。

在面对无约束的优化命题时，我们可以采用牛顿法等方法来求解。而面对有约束

的命题时，我们往往需要更高级的算法。单纯形法（Simplex Method）<sup>[43]</sup>可以用来求解带约束的线性规划命题（LP），与之类似的有效集法（Active Set Method）<sup>[44]</sup>可以用来求解带约束的二次规划（QP），而内点法（Interior Point Method）<sup>[45]</sup>则是另一种用于求解带约束的优化命题的方法。常用的内点法有为障碍函数法（Barrier Method）。

对于含不等式约束的凸优化问题：

$$\begin{aligned} & \text{minimize } f_0(x) \\ & \text{subject to } f_i(x) \leq 0, i = 1, \dots, m \\ & \quad Ax = b \end{aligned} \quad (2.5)$$

其中  $f_0, \dots, f_m: \mathbb{R}^n \rightarrow \mathbb{R}$  是二次可微的凸函数， $A \in \mathbb{R}^{p \times n}$ ,  $\text{rank } A = p < n$ 。假设该问题可解，即存在最优的  $x^*$ ，即存在  $x \in D$  满足  $Ax = b$  和  $f_i(x) < 0, i = 1, \dots, m$ 。这意味着 Slater 约束品性成立，因此存在最优对偶  $\lambda^* \in \mathbb{R}^m$ ,  $v^* \in \mathbb{R}^p$ ，它们和  $x^*$  一起满足 KKT 条件：

$$\begin{aligned} & Ax^* = b, f_i(x^*) \leq 0, i = 1, \dots, m \\ & \lambda^* \succeq 0 \\ & \nabla f_0(x^*) + \sum_{i=1}^m \lambda_i^* \nabla f_i(x^*) + A^T v^* = 0 \\ & \lambda_i^* f_i(x^*) = 0, i = 1, \dots, m \end{aligned} \quad (2.6)$$

KKT 条件中的不等式使得对 KKT 系统的求解难以为继，因此 Barrier Method 的思想就是通过在原始的目标函数中添加一个障碍函数（也可以理解成惩罚函数）来代替约束条件中的不等式约束。也就是说，把式(2.5)转换成如下形式：

$$\begin{aligned} & \min f_0(x) + \sum_{i=1}^m I_-(f_i(x)) \\ & \text{subject to } Ax = b \end{aligned} \quad (2.7)$$

其中式(2.7)右边函数可以一个近似的函数的表达：

$$\hat{I}_-(u) = -(1/t) \log(-u) \quad (2.8)$$

其中  $t$  是用于调整近似程度的参数，从上图可以看出， $t$  越大近似效果越好。将上面的近似函数式(2.8)代入到式(2.7)中，可以得到如下的一个近似的等式：

$$\begin{aligned} & \min f_0(x) + \sum_{i=1}^m -(1/t) \log(-f_i(x)) \\ & \text{subject to } Ax = b \end{aligned} \quad (2.9)$$

定义如下的对数障碍（logarithmic barrier）：

$$\phi(x) = -\sum_{i=1}^m \log(-f_i(x)) \quad (2.10)$$

则式(2.9)的求解问题可转化为下式的求解：

$$\begin{aligned} \min \quad & tf_0(x) + \phi(x) \\ \text{subject to} \quad & Ax = b \end{aligned} \quad (2.11)$$

中心路径：对(2.11)中不同的  $t(t>0)$  值，我们定义  $x^*(t)$  为相应优化命题的解。那么，中心路径就是指所有点  $x^*(t)$  ,  $t>0$  的集合，其中的点被称为中心点。中心路径上的点满足如下的充分必要条件，首先  $x^*(t)$  都是严格可行的，即

$$Ax^*(t) = b, f_i(x^*(t)) < 0, \quad i = 1, \dots, m$$

此外还存在对偶变量  $\hat{v}$  使得下面的等式成立（拉格朗日函数的导数为 0）：

$$\begin{aligned} 0 &= t\nabla f_0(x^*(t)) + \nabla \phi(x^*(t)) + A^T \hat{v} \\ &= t\nabla f_0(x^*(t)) + \sum_{i=1}^m \frac{1}{-f_i(x^*(t))} \nabla f_i(x^*(t)) + A^T \hat{v} \end{aligned} \quad (2.12)$$

给等号两边都乘上  $1/t$  得：

$$0 = \nabla f_0(x^*(t)) + \sum_{i=1}^m \frac{1}{-tf_i(x^*(t))} \nabla f_i(x^*(t)) + \frac{A^T \hat{v}}{t} \quad (2.13)$$

令  $\lambda_i^*(t) = \frac{1}{-tf_i(x^*(t))}$ ,  $v^* = \frac{\hat{v}}{t}$ ，就取得了与(2.5)式中的第一个等式基本一致的结果，即存在  $x^*(t)$  能最小化拉格朗日函数：

$$L(x, \lambda, v) = f_0(x) + \sum_{i=1}^m \lambda_i^* f_i(x) + v^T (Ax - b) \quad (2.14)$$

即， $\lambda^*(t)$  和  $v^*$  是原命题(1)的一组可行的对偶变量，那么此时对偶目标函数值为：

$$\begin{aligned} g(\lambda^*(t), v^*(t)) &= f_0(x^*(t)) + \sum_{i=1}^m \lambda_i^*(t) f_i(x^*(t)) + v^{*T}(t) (Ax^*(t) - b) \\ &= f_0(x^*(t)) - \frac{m}{t} \end{aligned} \quad (2.15)$$

随着  $t$  取值增大， $x^*(t)$  可以最终收敛到最优解。

## 2.3 线性模型预测控制

将一个非线性系统近似为线性时变系统有很多方法，大体可以分为近似线性化与精确线性化两类。近似线性化方法简单，适用性较强，缺点是在控制精度要求非常高的场合难以适用，精确线性化一般不具备普遍性<sup>[46]</sup>，往往是需要针对单个系统具体分析。在模型预测控制中，一般采用近似的线性化方法。以下介绍常用的线性化方法。



该方法的主要思想是假设参考系统已经在期望路径上完全通过，得到了路径上每个时刻的状态量和控制量<sup>[47]</sup>。通过对参考系统和当前系统之间的偏差设计模型预测控制器来跟踪期望路径。参考系统的任意时刻的状态和控制量满足如下关系：

$$\dot{x}_r = f(x_r, u_r) \quad (2.16)$$

在任意  $(\xi_r, u_r)$  点进行泰勒展开，只保留一阶项，忽略高阶项，得到：

$$\dot{x} = f(x_r, u_r) + \left. \frac{\partial f}{\partial \xi} \right|_{\substack{x=x_r \\ u=u_r}} (x - x_r) + \left. \frac{\partial f}{\partial u} \right|_{\substack{x=x_r \\ u=u_r}} (u - u_r) \quad (2.17)$$

也可以写为：

$$\dot{x} = f(x_r, u_r) + f_{x,r} (x - x_r) + f_{u,r} (u - u_r) \quad (2.18)$$

式中： $f_{x,r}$  为  $f$  相对于  $x$  的雅克比矩阵， $f_{u,r}$  为  $f$  相对于  $u$  的雅克比矩阵

将公式 (2.16) 与公式 (2.18) 相减可以得到：

$$\dot{\tilde{\xi}} = A(t)\tilde{\xi} + B(t)\tilde{u} \quad (2.19)$$

式中： $\tilde{\xi} = x - x_r$ ， $\tilde{u} = u - u_r$ ， $A(t) = f_{x,r}$ ， $B(t) = f_{u,r}$

这样，就得到新的状态方程。而该状态方程是连续的，不能直接用于模型预测控制器的设计，有必要对其进行离散化处理，此处采用近似的离散化，即：

$$\begin{aligned} A_{k,t} &= I + TA(t) \\ B_{k,t} &= TB(t) \end{aligned} \quad (2.20)$$

结合公式 (2.19) 和公式 (2.20)，可以得到：

$$\tilde{\xi}(k+1) = A_{k,t}\tilde{\xi}(k) + B_{k,t}\tilde{u}(k) \quad (2.21)$$

至此，得到了非线性系统在任意一个参考点处线性化后的系统，该系统是设计线性模型预测控制算法的基础。

我们已经将一个非线性系统近似为线性时变系统，本节主要介绍如何根据线性时变系统设计模型预测控制器，以及如何将模型预测控制算法转化为便于计算机求解的二次规划问题。

首先，考虑以下的线性化模型：

$$x(k+1) = A_{k,t}x(k) + B_{k,t}u(k) \quad (2.22)$$

设定

$$\xi(k|t) = \begin{bmatrix} x(k|t) \\ u(k-1|t) \end{bmatrix} \quad (2.23)$$

可以得到一个新的状态空间表达式：

$$\begin{aligned}\xi(k+1|t) &= \tilde{A}_{k,t}\xi(k|t) + \tilde{B}_{k,t}\Delta u(k|t) \\ \eta(k|t) &= \tilde{C}_{k,t}\xi(k|t)\end{aligned}\quad (2.24)$$

式中：

$$\begin{aligned}\tilde{A}_{k,t} &= \begin{bmatrix} A_{k,t} & B_{k,t} \\ 0_{m \times n} & I_m \end{bmatrix}, \quad \tilde{B}_{k,t} = \begin{bmatrix} B_{k,t} \\ I_m \end{bmatrix} \\ \tilde{C}_{k,t} &= [C_{k,t} \quad 0]\end{aligned}\quad (2.25)$$

为了进一步简化计算，做出如下假设：

$$\begin{aligned}\tilde{A}_{k,t} &= \tilde{A}_t, k=1, \dots, t+N-1 \\ \tilde{B}_{k,t} &= \tilde{B}_t, k=1, \dots, t+N-1\end{aligned}\quad (2.26)$$

如果系统的预测时域为  $N_p$ ，控制时域为  $N_c$ ，那么，预测时域内的状态量和系统输出量可用以下公式计算：

$$\xi(t+N_p|t) = \tilde{A}_t^{N_p}\xi(t|t) + \tilde{A}_t^{N_p-1}\tilde{B}_t\Delta u(t|t) + \dots + \tilde{A}_t^{N_p-N_c-1}\tilde{B}_t\Delta u(t+N_c|t) \quad (2.27)$$

$$\eta(t+N_p|t) = \tilde{C}_t\tilde{A}_t^{N_p}\xi(t|t) + \tilde{C}_t\tilde{A}_t^{N_p-1}\tilde{B}_t\Delta u(t|t) + \dots + \tilde{C}_t\tilde{A}_t^{N_p-N_c-1}\tilde{B}_t\Delta u(t+N_c|t) \quad (2.28)$$

为了使整个关系更加明确，将系统未来时刻的输出以矩阵的形式表达：

$$Y(t) = \Psi_t \xi(t|t) + \Theta_t \Delta U(t) \quad (2.29)$$

式中：

$$Y(t) = \begin{bmatrix} \eta(t+1|t) \\ \eta(t+2|t) \\ \dots \\ \eta(t+N_c|t) \\ \dots \\ \eta(t+N_p|t) \end{bmatrix} \quad \Psi_t = \begin{bmatrix} \tilde{C}_t\tilde{A}_t \\ \tilde{C}_t\tilde{A}_t^2 \\ \dots \\ \tilde{C}_t\tilde{A}_t^{N_c} \\ \dots \\ \tilde{C}_t\tilde{A}_t^{N_p} \end{bmatrix} \quad \Delta U(t) = \begin{bmatrix} \Delta u(t|t) \\ \Delta u(t+1|t) \\ \dots \\ \Delta u(t+N_c|t) \end{bmatrix}$$

$$\Theta_t = \begin{bmatrix} \tilde{C}_t \tilde{B}_t & 0 & 0 & 0 \\ \tilde{C}_t \tilde{A}_t \tilde{B}_t & \tilde{C}_t \tilde{B}_t & 0 & 0 \\ \dots & \dots & \ddots & \dots \\ \tilde{C}_t \tilde{A}_t^{N_c-1} \tilde{B}_t & \tilde{C}_t \tilde{A}_t^{N_c-2} \tilde{B}_t & \dots & \tilde{C}_t \tilde{B}_t \\ \tilde{C}_t \tilde{A}_t^{N_c} \tilde{B}_t & \tilde{C}_t \tilde{A}_t^{N_c-1} \tilde{B}_t & \dots & \tilde{C}_t \tilde{A}_t \tilde{B}_t \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{C}_t \tilde{A}_t^{N_p-1} \tilde{B}_t & \tilde{C}_t \tilde{A}_t^{N_p-2} \tilde{B}_t & \dots & \tilde{C}_t \tilde{A}_t^{N_p-N_c-1} \tilde{B}_t \end{bmatrix}$$

通过观察式 (2.29)，我们就可以清楚的看到，在预测时域内的状态量和输出量都可以通过系统当前的状态量  $\xi(t|t)$  和控制时域内的控制增量  $\Delta U(t)$  计算得到，这也就是模型预测控制算法中“预测”功能的实现。

而实际上，系统的控制增量是未知的，只有通过设定合适的优化目标，并对其求解，才能得到控制时域内的控制序列。一般的优化目标函数可以设为如下形式：

$$J(\xi(t), u(t-1), \Delta U(t)) = \sum_{i=1}^{N_p} \left\| \eta(t+i|t) - \eta_{\text{ref}}(t+i|t) \right\|_Q^2 + \sum_{i=1}^{N_c-1} \left\| \Delta u(t+i|t) \right\|_R^2 \quad (2.30)$$

其中，第一项反映了系统对参考轨线的跟随能力，第二项反映了对控制量平稳变化的要求。 $Q$  和  $R$  为权重矩阵，整个表达式的功能是使系统能够尽快且平稳的跟踪上期望的轨迹。同时，在实际的控制系统中，往往需要满足系统状态量以及控制量的一些约束条件，一般如下：

控制量约束：

$$\mathbf{u}_{\min}(t+k) \leq \mathbf{u}(t+k) \leq \mathbf{u}_{\max}(t+k), k=0,1,\dots,N_c-1 \quad (2.31)$$

控制增量约束：

$$\Delta \mathbf{u}_{\min}(t+k) \leq \Delta \mathbf{u}(t+k) \leq \Delta \mathbf{u}_{\max}(t+k), k=0,1,\dots,N_c-1 \quad (2.32)$$

输出约束：

$$\mathbf{y}_{\min}(t+k) \leq \mathbf{y}(t+k) \leq \mathbf{y}_{\max}(t+k), k=0,1,\dots,N_c-1 \quad (2.33)$$

式 (2.30) ~ (2.33) 就形成了一个完整的优化目标表达式，通过求解这个带约束条件的优化目标，就能得到未来一段时刻的控制序列。然而，由于系统的模型是实时改变的，并不能保证每个时刻该优化目标都能得到可行解。因此，有必要对优化目标进行相应的处理，比较普遍并且证明有效的方式是在优化目标中加入松弛因子，如式 (2.34) 所示：

$$J(\xi(t), u(t-1), \Delta U(t)) = \sum_{i=1}^{N_p} \left\| \eta(t+i|t) - \eta_{\text{ref}}(t+i|t) \right\|_Q^2 + \sum_{i=1}^{N_c-1} \left\| \Delta u(t+i|t) \right\|_R^2 + \rho \varepsilon^2 \quad (2.34)$$

式中： $\rho$  为权重系数， $\varepsilon$  为松弛因子。

对于以上形式的优化目标，可以通过适当的处理将其转换为二次规划（Quadratic Programming, QP）问题。二次规划是一个典型的数学优化问题，它的优化目标是二次实函数，带有线性或者非线性约束，常用的解法为有效集法（Active Set Method, ASM）和内点法（Interior point method, IPM）。有效集法适用于解决只有不等式约束的二次规划问题，而内点法适用于任何形式的二次规划问题。

将式（2.29）代入优化目标式（2.30），并且将预测时域内的输出量偏差表示为：

$$E(t) = \Psi_t \tilde{\xi}(t|t) - Y_{\text{ref}}(t), Y_{\text{ref}} = [\eta_{\text{ref}}(t+1|t), \dots, \eta_{\text{ref}}(t+N_p|t)]^T \quad (2.35)$$

经过相应的矩阵计算，可以将优化目标调整为：

$$J(\xi(t), u(t-1), \Delta U(t)) = [\Delta U(t)^T, \varepsilon]^T H_t [\Delta U(t)^T, \varepsilon] + G_t [\Delta U(t)^T, \varepsilon] + P_t \quad (2.36)$$

$$\text{式中： } H_t = \begin{bmatrix} \Theta_t^T Q_e \Theta_t + R_e & 0 \\ 0 & \rho \end{bmatrix}, \quad G_t = \begin{bmatrix} 2E(t)^T Q_e \Theta_t & 0 \end{bmatrix}, \quad P_t = E(t)^T Q_e E(t)$$

在式（2.35）中， $P_t$  为常量，因此模型预测控制在每一步的带约束优化求解问题等价于求解如下的二次规划问题：

$$J(\xi(t), u(t-1), \Delta U(t)) = [\Delta U(t)^T, \varepsilon]^T H_t [\Delta U(t)^T, \varepsilon] + G_t [\Delta U(t)^T, \varepsilon] \quad (2.37)$$

$$\text{s.t. } \Delta U_{\min} \leq \Delta U(k) \leq \Delta U_{\max}, k = t, \dots, t + H_c - 1$$

$$U_{\min} \leq u(t-1) + \sum_{i=t}^k \Delta U(i) \leq U_{\max}, k = t, \dots, t + H_c - 1$$

$$Y_{\min} - \varepsilon \leq \Psi_t \tilde{\xi}(t|t) + \Theta_t \Delta U(t) \leq Y_{\max} + \varepsilon$$

$$\varepsilon > 0$$

在每一控制周期内完成对式（2.36）的求解后，得到了控制时域内的一系列控制输入增量：

$$\Delta U_t^* = [\Delta u_t^*, \Delta u_{t+1}^*, \dots, \Delta u_{t+N_c-1}^*]^T \quad (2.38)$$

根据模型预测控制的基本原理，将该控制序列中第一个元素作为实际的控制输入增量作用于系统，即：

$$u(t) = u(t-1) + \Delta u_t^* \quad (2.39)$$

## 2.4 本章小结

在本章中，首先介绍了模型预测控制算法的基本理论，在此基础上依次描述了非线性和线性模型预测控制问题。对于非线性模型预测控制，我们本文使用内点法求解。传统方法是在非线性模型的基础上，通过将参考系统和当前系统之间的偏差作为状态量，并对状态量进行泰勒展开，保留一阶项，忽略高阶项，从而将非线性模型转化成线性模型。近似线性化方法简单，适用性较强，缺点是在控制精度要求非常高的场合难以适用，精确线性化一般不具备普遍性，往往是需要针对单个系统具体分析。

### 第3章 自适应采样周期的模型预测路径跟踪

前一章介绍了模型预测控制的基本原理，并介绍了线性模型预测控制和非线性模型预测控制的解法。本章首先分析基于自行车模型算法原理和缺陷，并分析固定采样周期的影响。针对这些问题，本文提出一种考虑道路特性的模型预测控制算法。首先，为了获取道路参数，我们首先对参考路径进行多项式拟合，以后轴中心到参考路径的距离、车辆位姿、车辆与参考轨迹的航向偏差作为状态量，以前轮偏角作为控制量建立预测模型，在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束，进而用二次规划算法求解带约束的目标函数。此外，单一的采样周期很难适应所有工况，本文设计一种随道路曲率的自适应采样周期的方法。

#### 3.1 未考虑道路参数的模型预测控制算法

##### 3.1.1 未考虑道路参数的模型预测建模

车辆转向运动模型如图 3.1 所示，在惯性坐标系  $OXY$  下， $(X_r, Y_r)$  和  $(X_f, Y_f)$  分别为车辆后轮和前轮轴心的坐标； $\varphi$  为车体的横摆角（航向角）； $\delta_f$  为前轮偏角； $v_r$  为车辆后轴中心速度； $v_f$  为车辆前轴中心速度； $l$  为轴距。

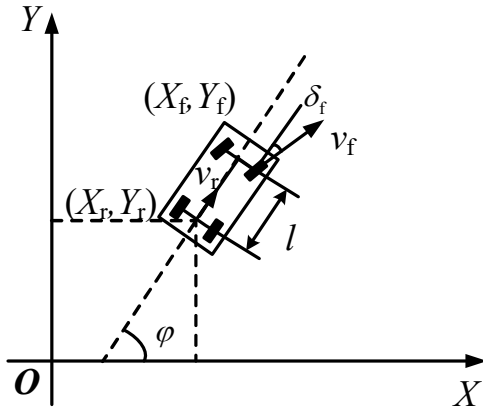


图 3.1 车辆运动模型

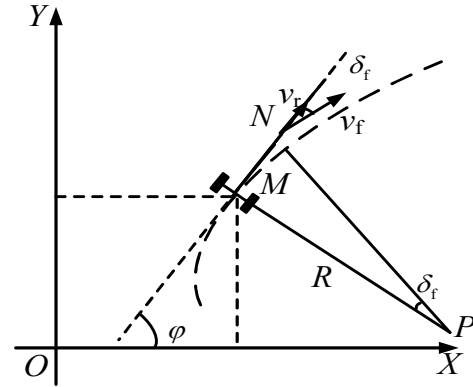


图 3.2 车辆转向示意图

图 3.2 为车辆转向过程示意图， $R$  为后轮转向半径， $P$  为车辆的瞬时转动中心， $M$  为车辆后轮轴心， $N$  为前轮轴心。此处假设转向过程中车辆质心侧偏角保持不变，即车辆瞬时转向半径与道路曲率半径相同。

在后轮行驶轴心  $(X_r, Y_r)$  处，速度为：

$$v_r = \dot{X}_r \cos \varphi + \dot{Y}_r \sin \varphi \quad (3.1)$$

前、后轮的运动学约束为：

$$\begin{cases} \dot{X}_f \sin(\varphi + \delta_f) - \dot{Y}_f \cos(\varphi + \delta_f) = 0 \\ \dot{X}_r \sin \varphi - \dot{Y}_r \cos \varphi = 0 \end{cases} \quad (3.2)$$

由式 (3.1) 和式 (3.2) 联合可得：

$$\begin{cases} \dot{X}_r = v_r \cos \varphi \\ \dot{Y}_r = v_r \sin \varphi \end{cases} \quad (3.3)$$

根据前后轮的几何关系可得：

$$\begin{cases} X_f = X_r + l \cos \varphi \\ Y_f = Y_r + l \sin \varphi \end{cases} \quad (3.4)$$

将式 (3.3) 和式 (3.4) 代入式 (3.2)，可以解得横摆角速度为：

$$\omega = \frac{v_r}{l} \tan \delta_f \quad (3.5)$$

式中： $\omega$  为车辆横摆角速度。

同时，由车辆的横摆角速度  $\omega$  和车速  $v_r$  可得到转向半径和前轮偏角：

$$\begin{cases} R = v_r / \omega \\ \delta_f = \arctan(l / R) \end{cases} \quad (3.6)$$

由式 (3.3) 和式 (3.5) 可以得到车辆运动学模型为：

$$\begin{bmatrix} \dot{X}_r \\ \dot{Y}_r \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi \\ \sin \varphi \\ \tan \delta_f / l \end{bmatrix} v_r \quad (3.7)$$

该模型可以进一步表示为更为一般的形式：

$$\dot{\xi}_{\text{kin}} = f_{\text{kin}}(\xi_{\text{kin}}, u_{\text{kin}}) \quad (3.8)$$

其中，状态量  $\xi_{\text{kin}} = [X_r, Y_r, \varphi]$ ，控制量  $u_{\text{kin}} = [v_r, \delta_f]$ 。

在无人驾驶车辆的路径跟踪控制过程中，往往希望以  $[v_r, \omega]$  作为控制量，将 (3.5) 代入式 (3.7) 中，该车辆运动学模型可以转换为如下形式：

$$\begin{bmatrix} \dot{X}_r \\ \dot{Y}_r \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi \\ \sin \varphi \\ 0 \end{bmatrix} v_r + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (3.9)$$

为了验证所建立的车辆运动学模型，在 Matlab/Simulink 环境中搭建该运动学模型，在相同的输入条件下与 CarSim 中所建立的整车模型进行对比分析。其中，相同的输入条件是指前轮偏角与车速随时间的变化历程相同，输出均为车辆位置和航向。

车辆基本参数设置为：轴距  $l=2.7\text{m}$ ，初始状态  $\xi_{\text{kin}}=[0,0,0]$ 。用于模型验证的输入信号随时间的变化历程如图 3.3 所示，其中图 3.3 (a) 为车速随时间的变化历程，图 3.3 (b) 为前轮偏角随时间变化历程。本文所建立的运动学模型和 CarSim 整车模型的输出信号如图 3.4 所示，其中图 3.4 (a) 为两者在给定的输入条件下输出的位置信息对比，图 3.4 (b) 为两者的航向角信息对比。

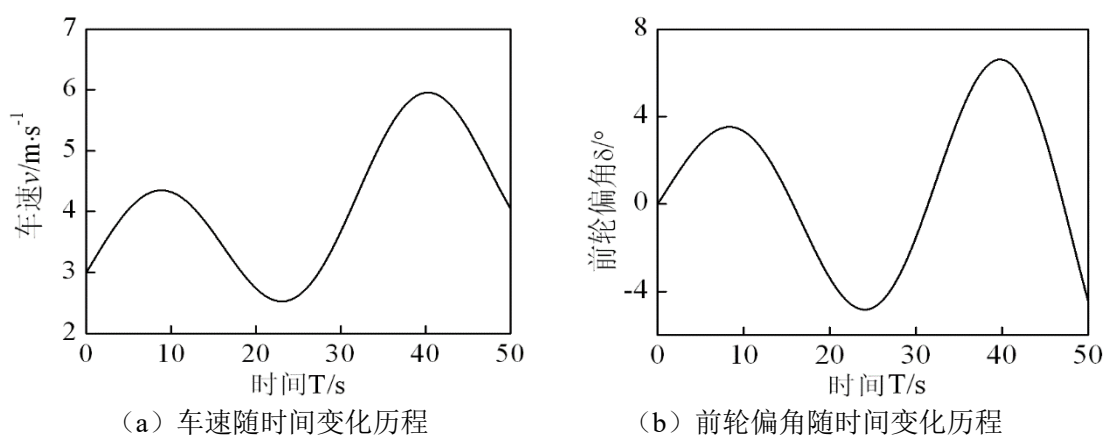


图 3.3 模型验证的输入信号

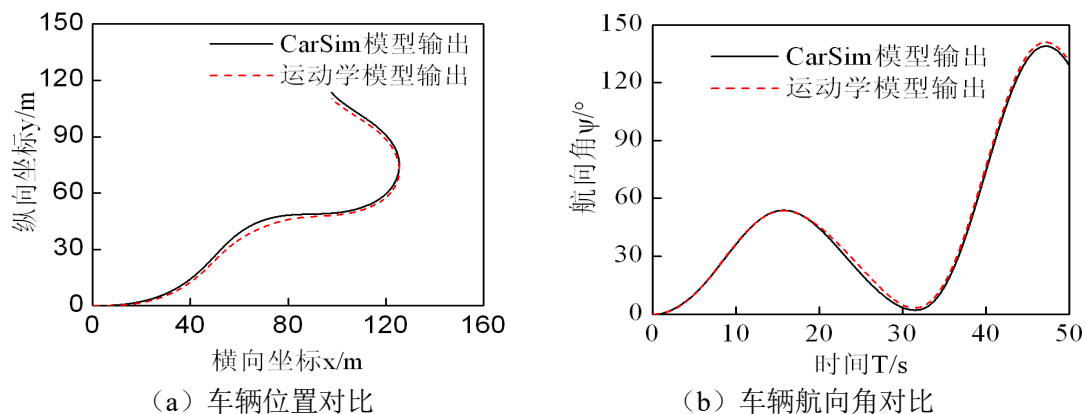


图 3.4 模型输出结果对比

从图 3.4(a)中对比结果可以看出，在相同的速度和前轮偏角输入下，运动学模型的车辆位置与 Carsim 参考轨迹存在较大偏差，特别是在道路曲率变化处较明显，并未考虑道路的本身特性，从而导致较大误差。说明使用简单的二自由度模型存在较大的跟踪误差。



### 3.1.2 固定采样周期

采样周期  $h_k$  与预测时域  $N_p$  和速度的乘积类似于前向预测距离，在曲率较大的情况，前向预测距离比较短，即前视距离比较短，而在直线情况下，预测距离应该比较长，即前视距离比较长。因此单一的采样周期难以实现多种道路曲率的跟踪。

对此我们通过仿真验证，单一的采样周期难以适应曲率变化的道路工况。

仿真实验介绍：给定参考路径为  $y = a \sin wx$ ，车辆起始位姿为  $(0,0,0)$ ，车速为  $10\text{m/s}$  匀速，在时延为  $0\text{s}$  情况下，分析不同采样时间  $h_k=0.05\text{s}$ ， $0.1\text{s}$ ， $0.2\text{s}$  和对模型预测轨迹跟踪器的影响。

(1) 采样周期为  $0.05\text{s}$  时的跟踪结果：

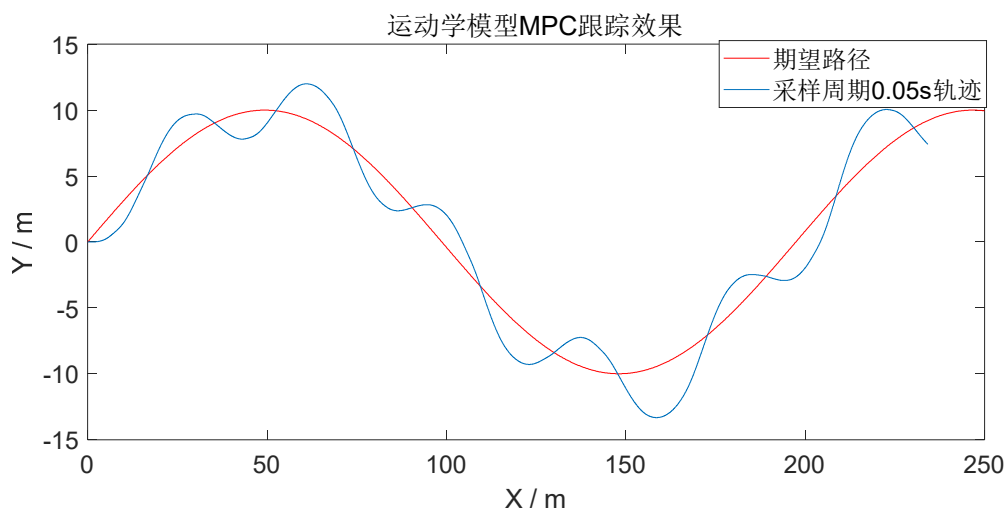


图 3.5 采样周期为  $0.05\text{s}$  时的跟踪结果

采样周期为  $0.05\text{s}$  时的跟踪结果分析：当以采样周期为  $0.05\text{s}$  进行模型预测跟踪时，跟踪结果震荡幅度大，跟踪的横向最大偏差大于  $3\text{m}$ ，此时模型预测控制器对误差响应强烈。

(2) 采样周期为  $0.15\text{s}$  时的跟踪结果：

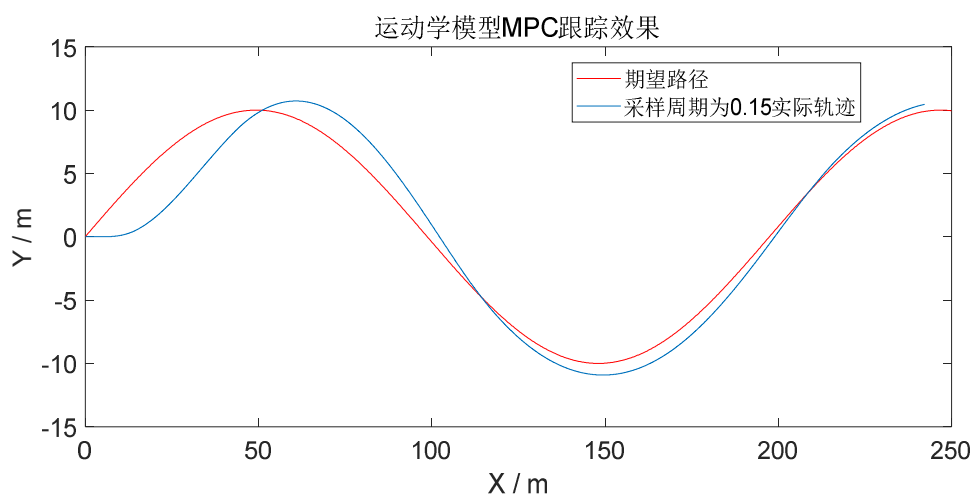


图 3.6 采样周期为 0.15s 时的跟踪结果

采样周期为 0.15s 时的跟踪结果分析：当以采样周期为 0.15s 进行模型预测跟踪时，跟踪结果如图 3.6 所示，可以发现在跟踪的开始阶段，模型预测控制器的响应滞后，从而造成较大偏差，但相比于 0.05s 的采样周期没有出现震荡现象，最终误差也逐渐收敛。

(3) 采样周期为 0.2s 时的跟踪结果：

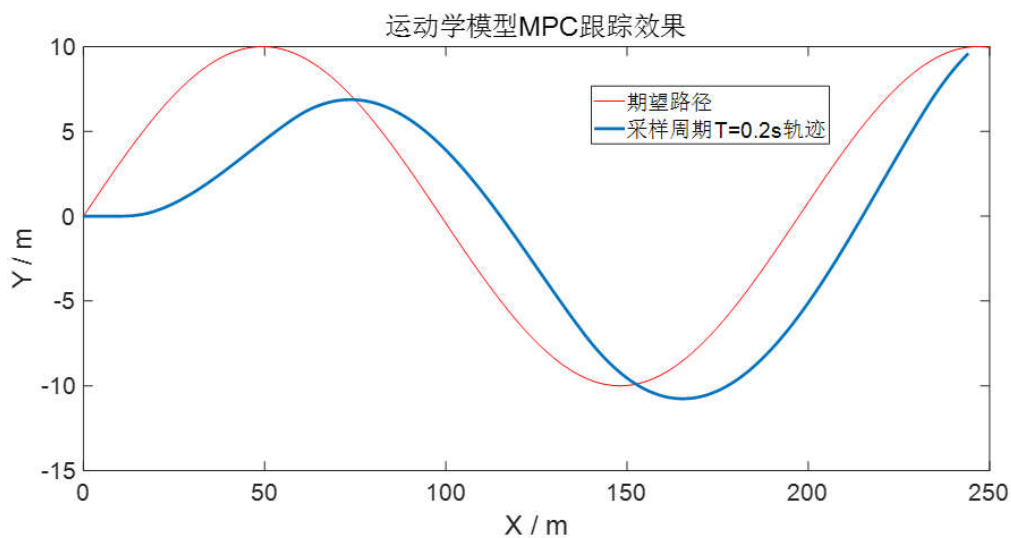


图 3.7 采样周期为 0.2s 时的跟踪结果

采样周期为 0.2s 时的跟踪结果分析：当以采样周期为 0.2s 进行模型预测跟踪时，跟踪结果如图 3.7 所示，可以发现在跟踪的开始阶段，模型预测控制器的响应严重滞后，从而造成大于 5m 的横向偏差，虽然相比于 0.05s 的采样周期没有出现震荡现象，但相对于 0.15s 采样周期结果，最终误差也没有逐渐收敛至一个较小范围。

结果分析：采样周期  $h_k$  与预测时域  $N_p$  和速度的乘积类似于前向预测距离，在曲率较大的情况，前向预测距离比较短，即前视距离比较短，而在直线情况下，预测距离应该比较长，即前视距离比较长。过小的采样周期在跟随曲率变化的道路将引起强烈的震荡，过大的采样周期将引起跟踪滞后。

### 3.2 基于运动学的预测模型

为了解决 3.1 节中模型预测控制未考虑道路参数和定采样周期带来跟踪误差较大的问题，本文采用一种适应各种道路曲率的变采样周期模型预测控制路径跟踪算法。首先，我们对参考路径离散的点拟合获取曲线的参数形式，再以后轴中心到参考路径的距离、车辆位姿、车辆与参考轨迹的航向偏差，以前轮偏角作为控制量建立预测模型，在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束，进而用二次规划算法求解带约束的目标函数。

#### 3.2.1 参考路径拟合

为了获取道路的相关参数，考虑到道路的普遍性和拟合的实时性，本文采用多项式拟合参考路径。为了获得多项式的各参数，本文适用最小二乘法。

给定  $m$  个数据点  $p_i(x_i, y_i), i=1, 2, \dots, m$ ，求近似曲线  $y = f(x)$ ，并且使得近似曲线与实际值的偏差最小。近似曲线在点  $p_i$  处的偏差  $e_i = f(x_i) - y_i, i=1, 2, \dots, m$ 。对此，我们使用最小二乘法拟合，即按偏差平方和最小的原则选取拟合曲线，并且采取二项式方程为拟合曲线的方法，称为最小二乘法。

推导过程：

设拟合多项式为：

$$y = a_0 + a_1x + \dots + a_kx^k \quad (3.10)$$

各点到这条曲线的距离之和，即偏差平方和如下：

$$R^2 = \sum_{i=1}^n [(y_i - (a_0 + a_1x_i + \dots + a_kx_i^k))]^2 \quad (3.11)$$

为了求得符合条件  $a$  的值，等式右边对  $a_i (i=0, \dots, k)$  求偏导数，得到了：

$$\begin{aligned}
& -2 \sum_{i=1}^n [y_i - (a_0 + a_1 x_i + \dots + a_k x_i^k)] = 0 \\
& -2 \sum_{i=1}^n [(y_i - (a_0 + a_1 x_i + \dots + a_k x_i^k)) x_i] = 0 \\
& \dots\dots\dots \\
& -2 \sum_{i=1}^n [(y_i - (a_0 + a_1 x_i + \dots + a_k x_i^k)) x_i^k] = 0
\end{aligned} \tag{3.12}$$

把这些等式表示成矩阵的形式，就可以得到下面的矩阵等式：

$$\begin{bmatrix}
n & \sum_{i=1}^n x_i & \dots & \sum_{i=1}^n x_i^k \\
\sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \dots & \sum_{i=1}^n x_i^{k+1} \\
\vdots & \vdots & \ddots & \vdots \\
\sum_{i=1}^n x_i^k & \sum_{i=1}^n x_i^{k+1} & \dots & \sum_{i=1}^n x_i^{2k}
\end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
\vdots \\
a_k
\end{bmatrix}
=
\begin{bmatrix}
\sum_{i=1}^n y_i \\
\sum_{i=1}^n x_i y_i \\
\vdots \\
\sum_{i=1}^n x_i^k y_i
\end{bmatrix} \tag{3.13}$$

将这个范德蒙得矩阵化简后可得到：

$$\begin{bmatrix}
1 & x_1 & \dots & x_1^k \\
1 & x_2 & \dots & x_2^k \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_n & \dots & x_n^k
\end{bmatrix}
\begin{bmatrix}
a_0 \\
a_1 \\
\vdots \\
a_k
\end{bmatrix}
=
\begin{bmatrix}
y_1 \\
y_2 \\
\vdots \\
y_n
\end{bmatrix} \tag{3.14}$$

也就是说  $XA=Y$ ，那么  $A=(X'X)^{-1}X'Y$ ，便得到了系数矩阵  $A$ ，同时，我们也就得到了拟合曲线。为了减少运算量和保证拟合的路径逼近真实路径，我们选用三次曲线拟合。

### 3.2.2 预测模型

我们在经典运动学模型的基础上，以车辆后轴中心为坐标系原点，将车辆后轴位置  $(X_r, Y_r)$  到参考曲线  $f(x) = ax^3 + bx^2 + Cx + d$  距离  $Cte$  添加为新的系统状态变量，另外考虑参考路径和车辆的航向偏差  $Epsi$ ，其中参考路径的航向由车体坐标系下的曲线获得，即  $\arctan(f'(X_r))$ 。以前轮偏角为控制量建立系统模型。这些量的几何关系如下图所示：

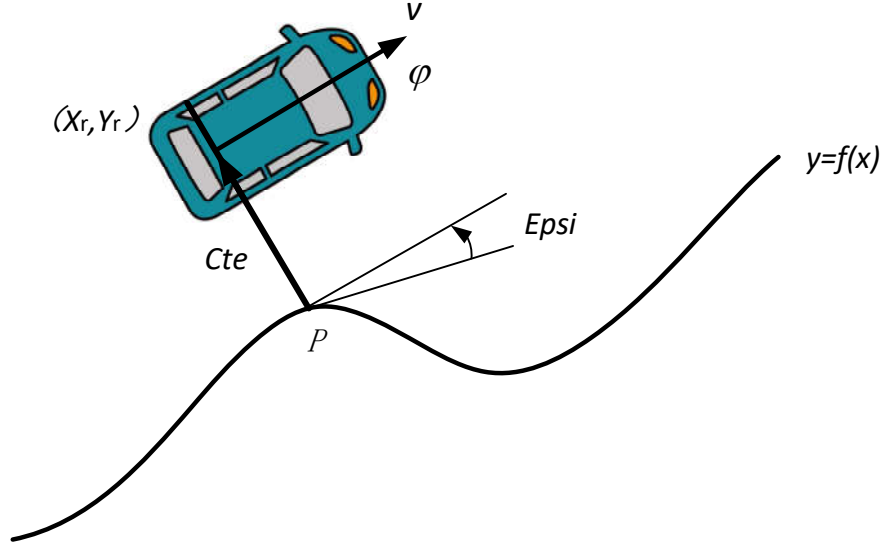


图 3.8 预测模型示意图

根据上图我们可知，我们定义横向偏差和航向偏差分别为  $Cte = Y_r - f(X_r)$ ， $Epsi = \varphi - \arctan(f'(X_r))$ ，将上式关系用状态方程表示为如下：

$$\dot{X}_r = v(t) \cos(\varphi) \quad (3.15-a)$$

$$\dot{Y}_r = v(t) \sin(\varphi) \quad (3.15-b)$$

$$\dot{\varphi} = -v(t) \delta / l \quad (3.15-c)$$

$$\dot{Cte} = v(t) \sin(epsi) \quad (3.15-d)$$

$$\dot{Epsi} = -v(t) \delta / l - 2bv(t)z(t) \quad (3.15-e)$$

式中： $v(t)$ 为车辆的速度，输入 $\delta$ 为车辆的前轮偏角， $l$ 为轴距，其中 $b$ 为多项式曲线的二次项系数，式中 $z(t) = k / (1 + k^2)$ ， $k = f'(v(t)dt)$ ， $k$ 为图中参考点 $P$ 的斜率。

### 3.2.3 模型线性化

对于上述非线性运动学模型，在车体坐标系下车辆的航向在单个周期内没有较大变化，即 $\varphi$ 值很小，假设车辆沿参考路径行驶时航向与参考路径航向不会相差太远，则航向的差值 $Epsi$ 在任何时刻通常小于15度，则我们对模型可以进行简化。

由于 $\varphi$ 很小，则根据小角度假设有：

$$\dot{X}_r = v(t), \dot{Y}_r = v(t)\varphi \quad (3.16)$$

由于当  $Epsi$  非常小时,  $\sin(Epsi) \approx Epsi$ , 则式(3.15-d)可简化为

$$Cte = v(t)Epsi \quad (3.17)$$

因此可以得到如下线性化的模型:

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) + E(t)z(t) \quad (3.18)$$

其中:

$$A(t) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & v(t) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & v(t) \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.19)$$

$$B(t) = [0 \ 0 \ -v(t)/l \ 0 \ -v(t)/l]^T \quad (3.20)$$

$$E(t) = [0 \ 0 \ 0 \ 0 \ -2bv(t)]^T \quad (3.21)$$

系统的状态矢量为  $x^T = [X_r \ Y_r \ \varphi \ Cte \ Epsi]$ , 并且在单个周期跟踪过程中, 假设在预测时域内速度为常值。

#### 3.2.4 系统输出

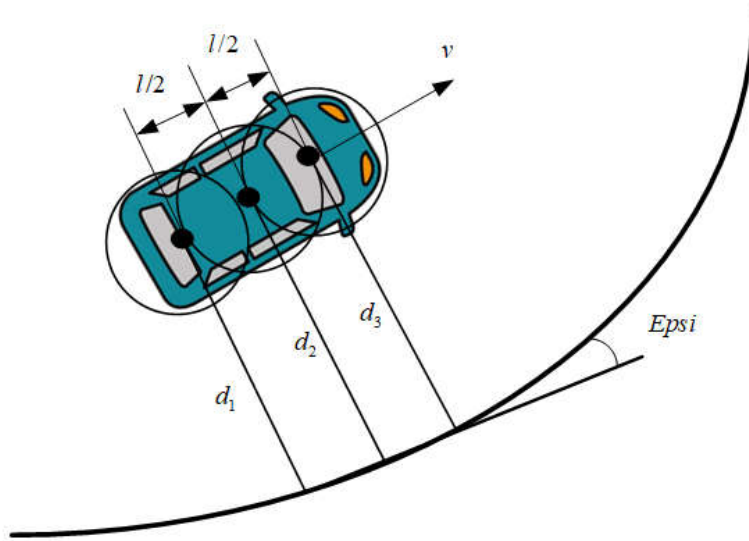


图 3.9 系统输出

为了接下来有效地表达系统约束, 本文定义了一系列的系统输出。为了制定相关的防碰撞的约束, 本文用几个圆近似代替车辆的形状。本文使用在文献中所提及的方法, 沿着车辆中心线放置 3 个圆, 分别放置在后轴中心、前轴中心和前后轴中心之间

的中点。定义该 3 中心到参考曲线的距离  $d_i$ ，定义  $d_i$  如下：

$$d_i = d_r + l_i \sin(Epsi) \approx d_r + l_i Epsi, \quad i = 1, 2, 3 \quad (3.22)$$

式中：  $l_1 = 0, l_2 = \frac{l}{2}, l_3 = l$ 。

上述关系可由图描述，则系统的输出可以定义为：

$$y = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & l/2 \\ 0 & 0 & 0 & 1 & l \end{bmatrix} \begin{bmatrix} X_r \\ Y_r \\ \varphi \\ Cte \\ Epsi \end{bmatrix} \quad (3.23)$$

### 3.2.5 系统模型离散化

线性系统状态空间模型的离散化，实际上是指在采样周期  $h_k$  下将状态空间模型

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + B(t)u(t) \\ y(t) &= C(t)x(t) \end{aligned} \quad (3.24)$$

变成离散系统的如下状态空间模型：

$$\begin{aligned} x((k+1)h_k) &= Gx(kh_k) + Hu(kh_k) \\ y(kh_k) &= Cx(kh_k) \\ G &= e^{Ah_k} \\ H &= B \int_0^{h_k} e^{At} dt \end{aligned} \quad (3.25)$$

- (1) 由于离散化主要是描述系统动态特性的状态方程而言，输出方程为静态的代数方程，其离散化后应保持不变；
- (2) 离散化主要针对连续状态方程  $\sum (A_c, B_c)$  如何通过采样周期  $T_s$  变换成离散系统状态方程  $\sum (G, H)$ 。

离散化方法有两种：

#### (1) 精确法

利用状态方程的求解公式以保证状态在采样时刻连续状态方程和离散化状态方程有相同的解来进行离散化。

连续系统的状态方程的求解公式如下：

$$x(t) = \Phi(t-t_0)x(t_0) + \int_{t_0}^t \Phi(t-\tau)Bu(\tau)d\tau \quad (3.26)$$

对于上式只考虑在采样时刻  $t = kh_k$  和  $t = (k+1)h_k$  时刻之间的状态响应，即对上式

$t_0 = kh_k, t = (k+1)h_k$ ，于是有：

$$x((k+1)h_k) = \Phi(h_k)x(kh_k) + \int_{kh_k}^{(k+1)h_k} \Phi((k+1)h_k - \tau)Bu(\tau)d\tau \quad (3.27)$$

假定  $u(t)$  在采样周期内保持不变，所以有：

$$x((k+1)h_k) = \Phi(h_k)x(kh_k) + \int_{kh_k}^{(k+1)h_k} \Phi((k+1)h_k - \tau)d\tau Bu(kh_k) \quad (3.28)$$

对上式做变量代换，令  $t = (k+1)h_k - \tau$ ，则上式可记为：

$$x((k+1)h_k) = \Phi(h_k)x(kh_k) + \int_0^{h_k} \Phi(t)dt Bu(kh_k) \quad (3.29)$$

由上两式可知：

$$G = \Phi(h_k) = e^{Ah_k} \quad (3.30)$$

$$H = \int_0^{h_k} \Phi(t)dt B = \int_0^{h_k} e^{At} dt B \quad (3.31)$$

## (2) 近似法

近似离散化方法是指在采样周期较小且离散化精度要求不高的情况下，用状态变量的差商代替微商来求得近似的差分方程，即：

$$\dot{x}(kh_k) = \lim_{h_k \rightarrow 0} [x((k+1)h_k) - x(kh_k)] / h_k \quad (3.32)$$

采样周期较小时，有：

$$\dot{x}(kh_k) = [x((k+1)h_k) - x(kh_k)] / h_k \quad (3.33)$$

将上式带入连续系统的状态方程，得到：

$$[x((k+1)h_k) - x(kh_k)] / h_k = Ax(kh_k) + Bu(kh_k) \quad (3.34)$$

由上式得：

$$\begin{aligned} G &= I + Ah_k \\ H &= Bh_k \end{aligned} \quad (3.35)$$

将上述近似离散法和精确离散法比较可知：

(1) 由于  $I + Ah_k$  和  $Bh_k$  分别是  $e^{Ah_k}$  和  $\int_0^{h_k} e^{At} dt B$  的 Taylor 展开式中一次近似，因此离

散近似化方法其实是取精确离散化方法的相应计算式的一次 Taylor 近似展开；

(2) 一般来说，采样周期越小，离散化精度越高。

为了获得更好的控制精度，本文采用精确离散法。

由于采样周期影响离散化的精度，所以本文对采样周期对控制的影响做数据上的分析，采样周期过小，离散的模型就越接近连续的模型，但是容易出现震荡，导致车辆侧向稳定性降低，而采样周期过大虽然可以保证侧向稳定性但容易导致收敛较慢。



影响路径跟踪的一个关键因素是道路曲率，鉴于此，本文采用一种基于道路曲率变采样时间的方法<sup>[48]</sup>：

$$h_k = \text{round}(10 + 10e^{-wPGC}) / 100 \quad (3.36)$$

其中： $PGC$  参考路径平均曲率， $w$  为收缩因子（下文取 20）。同样跟踪 3.1 节中的场景，采用自适应的采样周期对正弦曲线进行跟踪。

采样周期为  $h_k = \text{round}(10 + 10e^{-wPGC}) / 100$  的跟踪结果：

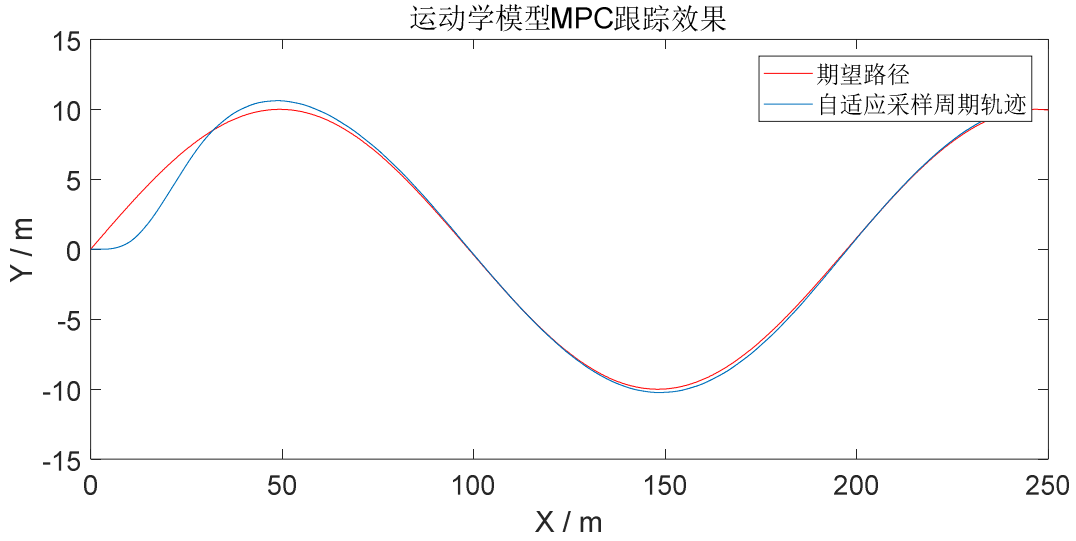


图 3.10 采样周期为  $\text{round}(10 + 10e^{-wPGC}) / 100$  的跟踪结果

采样周期为  $\text{round}(10 + 10e^{-wPGC}) / 100$  的跟踪结果分析：用采样周期自适应的方法，对跟踪误差的响应迅速，且不会出现跟踪结果滞后和震荡的现象。

此时，原模型变成：

$$\begin{aligned} x(k+1) &= Gx(k) + Hu(k) + Nz(k) \\ y(k) &= Cx(k) \end{aligned} \quad (3.37)$$

其中：

$$G = e^{A(\text{round}(10 + 10e^{-wPGC})/100)}, \quad H = B \int_0^{\text{round}(10 + 10e^{-wPGC})/100} e^{At} dt, \quad N = E \int_0^{\text{round}(10 + 10e^{-wPGC})/100} e^{At} dt$$

### 3.3 系统约束

当在真实场景中跟踪参考路径时，其他交通要素如车道边缘和障碍物都需要考虑，以防止碰撞为出发点制定系统的约束。因此本文根据防止碰撞的条件对式中的输出  $d_i$  限制上界和下界，这些限制关系如图(3.11)所示。在图中， $d_{i,\max}$  和  $d_{i,\min}$  分别代表车辆

运动时在时刻  $t_j$  近似圆移动的最大的左侧允许偏差和右侧允许偏差。由  $t_j$  时刻经时间预测时间间隔  $\Delta t$  到达  $t_\Delta$  时刻，本车和运动障碍物目标相遇，此时左侧的最大允许偏差为  $d_{i,\max}(s_r(t_\Delta), t_\Delta)$ ,  $i=1,2,3$ 。综上本文定义线性时间离散的系统输出约束如下：

$$d_{i,\min}(s_r(k), k) \leq d_i(k) \leq d_{i,\max}(s_r(k), k) \quad (3.38)$$

此外，在路径跟踪时我们也考虑车辆本身的特性对控制量的限制。这种限制来自转向机构的限制。考虑到车辆的预测模型，这个限制可以通过最大前轮偏角来限制车辆控制量的边界和限制控制量的增量：

$$u_{\min} \leq u(k) \leq u_{\max} \quad (3.39)$$

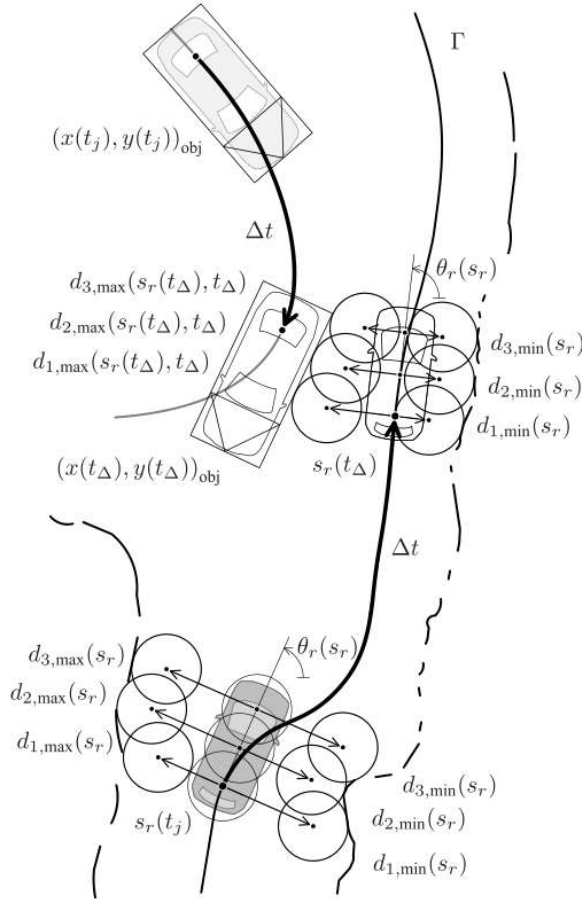


图 3.11 系统的状态约束

### 3.4 代价函数

遵循 LMPC 的基本思想，优化的过程是制定二次最小化运行成本函数  $l(x(k), u(k))$ 。

根据路径跟踪目标——稳定精确跟踪参考路径。因此设定目标函数设定的目的是产生最优的控制量，在控制器执行控制量之后能与参考路径的偏差缩小并收敛到某个微小的范围，且车辆本身保持平稳。基于上文提出的运动学模型，此节提出如下运行成本函数：

$$l(x(k), u(k)) = w_d Cte^2 + w_\theta Epsi^2 + w_k u^2 \quad (3.41)$$

前两项是惩罚偏离参考曲线的代价，后一项是限制车辆运行时的前轮偏角以保证车辆在转弯时的横向稳定性。 $w_d$ 、 $w_\theta$ 、 $w_k$ 为各项代价中的权重因子。将上式表示成矩阵的形式如下：

$$l(x(k), u(k)) = \sum_{k=1}^{N_p} x(k)^T Q x(k) + u(k) V u(k) \quad (3.42)$$

其中：

$$Q(k) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & w_d & 0 \\ 0 & 0 & 0 & 0 & w_\theta \end{bmatrix} \quad (3.43)$$

$$V(k) = w_k \quad (3.44)$$

因此，该模型预测控制的最优化问题可表示为：

$$\begin{aligned} \min \quad & l(x(k), u(k)) = \sum_{k=1}^{N_p} x(k)^T Q x(k) + u(k) V u(k) \\ \text{subject to:} \quad & d_{i,\min}(s_r(k), k) \leq d_i(k) \leq d_{i,\max}(s_r(k), k) \\ & u_{\min} \leq u(k) \leq u_{\max} \end{aligned} \quad (3.45)$$

加权矩阵  $Q$  和  $V$  的选取基本原则如下：

- (1) 当需要提高系统的快速响应性能，则可以通过减小  $V$  (或者增大  $Q$ ) 来实现；
- (2) 为了有效抑制控制量，降低控制成本，提升系统的稳态性能，则可以通过增加  $V$  (或者减小  $Q$ ) 来实现；
- (3) 加权矩阵  $Q$  和  $V$  的作用是相互制约的。

### 3.5 本章小结

常用的运动学模型预测算法都是基于车辆模型，但建模过程中并未考虑道路的曲率形状，从而导致较大跟踪误差。因此，本文针对此问题提出一种适用各种道路环境的模型预测控制算法。首先，为了获取道路参数，我们首先对参考路径进行多项式拟合，以后轴中心到参考路径的距离、车辆位姿、车辆与参考轨迹的航向偏差，其中航向偏差和横向偏差可以表示成参考路径相关参数的形式，以前轮偏角作为控制量建立预测模型，在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束，进而用二次规划算法求解带约束的目标函数。此外，单一的采样周期很难适应所有工况，采样周期与预测时域和速度的乘积类似于前向预测距离，在曲率较大的情况，前向预测距离比较短，即前视距离比较短，而在直线情况下，预测距离应该比较长，即前视距离比较长，针对此规律，本文设计一种随道路曲率的自适应采样周期的方法。

## 第4章 模型预测控制时延处理

由于执行机构执行控制量时会发生时延，而且时延时间越大，系统越不稳定。并且在实际对系统进行建模时，受客观条件约束，比如转向机构在执行前轮偏角时，由于存在系统时延，使得所建立系统模型的参数跟实际系统参数之间存在着一定的偏差，这势必会影响系统的稳定性等问题<sup>[49]</sup>。在建模时将不确定性引入到系统模型是可以人为有效地减少此类影响。

针对车辆平台转向机构时延对控制效果的影响问题，本文先对通过下发一系列的期望前轮控制量，并对比反馈的实际前轮偏角，获取时延大小的范围。由时延时间在实际转向过程中不是一个定值，这将给系统模型带来不确定性。针对这种带有不确定性的控制问题，其解决办法基本步骤如下：首先，构造参数时变多面体系统，其次求解参数时变多面体不确定系统的稳定性状态反馈控制律，然后结合原系统约束和反馈控制律构建鲁棒正向不变集作为参数不确定性系统的约束，最后求解时变多面体系统的滚动时域优化问题。

### 4.1 问题描述

对于一个时延系统，我们测出系统的时延  $\tau$ ，进而对预测的状态进行延时补偿，即将  $u(t)$  替换成  $u(t-\tau)$ ，其物理意义为实际执行的控制量为  $\tau$  时间前的。

时延的标定方法为，通过模拟下发一系列的控制量（前轮偏角），并结合反馈的前轮偏角，结合其两者的变化图，分析两者最大相差多少个运算周期和最小相差多少个运算周期，确定延时  $\tau$  所属的区间。

对于存在时延的控制系统，其连续时间状态方程为：

$$\dot{x}(t) = Ax(t) + Bu(t-\tau) \quad (4.1)$$

其中， $x(t)$  为状态矢量， $A$  为系统矩阵维度为  $n \times n$ ， $u(t-\tau)$  为具有输入时延  $\tau$  的输入矢量， $B$  为输入矩阵维度为  $n \times m$ 。输入的时延可以用如下图表示：

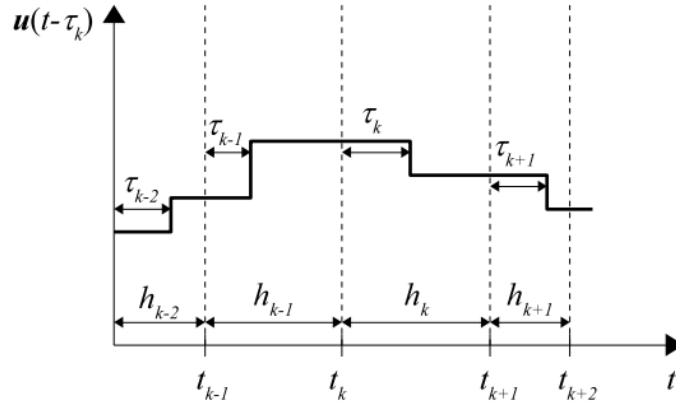


图 4.1 控制时延图

通常延时时间  $\tau$  都不是采样周期的整数倍，因此将延时时间表示成如下形式：

$$\tau = dh_k - \varepsilon \quad (4.2)$$

一般  $\varepsilon$  是随时间变化的，但可以选择不同的  $d$  保证它是有界的：

$$0 < \varepsilon \leq \bar{\varepsilon} \ll h_k \quad (4.3)$$

其中  $\bar{\varepsilon}$  为  $\varepsilon$  的最大值。

记在  $t$  时刻的状态  $x_k = x(t_k)$ ，因此式(4-1)可表示为：

$$x_{k+1} = Gx_k + Hu_{k-d} - F(u_{k-d} - u_{k-d+1}) \quad (4.4)$$

其中矩阵  $G$ 、 $H$  和  $F$  定义如下：

$$G = e^{Ah_k} \quad (4.5)$$

$$H = \int_0^{h_k} e^{A(h_k-s)} B ds \quad (4.6)$$

$$F = \int_{-|\varepsilon|}^0 e^{-As} B ds \quad (4.7)$$

离散时间模型的极端实现为：

对于  $\varepsilon=0$ ：

$$x_{k+1} = Gx_k + Bu_{k-d} \quad (4.8)$$

对于  $\varepsilon=\bar{\varepsilon}$ ：

$$x_{k+1} = Gx_k + (H - F)u_{k-d} + Fu_{k-d+1} \quad (4.9)$$

对于任意的变量  $\varepsilon$  满足  $0 \leq \varepsilon \leq \bar{\varepsilon} < h_k$  都满足如下约束条件：

$$Lx_k + Du_k \leq W \quad (4.10)$$

其中，该约束方程为第三章式(3.38)和(3.39)的组合。

## 4.2 加入延时的增广预测模型

将式(4.8)和(4.9)写成一个通用的模型如下：

$$\xi_{k+1} = G_f \xi_k + H_f u_k \quad (4.11)$$

其中：

$$\xi_k^T = [x_k^T \ u_{k-d}^T \ \dots \ u_{k-1}^T \ u_k^T] \quad (4.12)$$

$$G_f = \begin{bmatrix} G & H-F & F & \dots & 0 \\ 0 & 0 & I_m & \dots & 0 \\ \dots & \dots & \dots & \ddots & \dots \\ 0 & 0 & 0 & \dots & I_m \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad (4.13)$$

$$H_f = [0 \ 0 \ \dots \ 0 \ I_m]^T \quad (4.14)$$

如果式(4.7)的积分可写成：

$$F = G_f^{-1}(e^{G_f \varepsilon} - I_n)H_f \quad (4.15)$$

对于  $\varepsilon$  的极限值情况，我们可以获得如下：

$$F_0 = 0_{n \times m} \quad (4.16)$$

$$F_i = G_f^{-1}V(e^{\Lambda_i \varepsilon} - I_n)V^{-1}H_f, \forall i = 1, \dots, n \quad (4.17)$$

其中  $V$  和  $\Lambda_i$  由  $G$  求约旦标准型获得。则式(4-11)矩阵  $G_f$  对于  $0 \leq \varepsilon \leq \bar{\varepsilon} < h_k$  都满足：

$$G_f \in Co\{G_{f0}, G_{f1}, \dots, G_{fn}\} \quad (4.18)$$

即  $G_f$  可以表示为多面体的凸组合，其中， $Co\{\cdot\}$  为多面体，该多面体中的元素表示为：

$$G_{f0} = \begin{bmatrix} G_f & H_f & 0 & \dots & 0 \\ 0 & 0 & I_m & \dots & 0 \\ \dots & \dots & \dots & \ddots & I_m \\ 0 & 0 & 0 & \dots & I_m \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad (4.19)$$

$$G_{fi} = \begin{bmatrix} G_f & H_f - nF_i & nF_i & \dots & 0 \\ 0 & 0 & I_m & \dots & 0 \\ \dots & \dots & \dots & \ddots & I_m \\ 0 & 0 & 0 & \dots & I_m \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}, i = 1, \dots, n \quad (4.20)$$

*Proof:*  $G_f$  对于  $0 \leq \varepsilon \leq \bar{\varepsilon} < h_k$  都满足  $G_f \in Co\{G_{f0}, G_{f1}, \dots, G_{fn}\}$

$$\begin{aligned}
 F &= V(e^{\Lambda \varepsilon} - I)V^{-1}G^{-1}H \\
 &= \sum_{i=1}^n V(e^{\Lambda_i \varepsilon} - I)V^{-1}G^{-1}H \\
 &= \sum_{i=1}^n V(e^{\Lambda_i \varepsilon} - I)V^{-1}G^{-1}H \\
 &= \sum_{i=1}^n V(\beta_i(e^{\Lambda_i \varepsilon} - I) + (1 - \beta_i)(e^{\Lambda_i 0} - I))V^{-1}G^{-1}H \\
 &= \sum_{i=1}^n \beta_i F_i + (1 - \beta_i)F_0 \\
 &= (n - \sum_{i=1}^n \beta_i)F_0 + \sum_{i=1}^n \beta_i F_i \\
 &= \frac{n - \sum_{i=1}^n \beta_i}{n} nF_0 + \sum_{i=1}^n \frac{\beta_i}{n} nF_i \\
 G_f &= \sum_{i=0}^n \alpha_i G_{fi}, \alpha_i = \frac{\beta_i}{n}, \alpha_0 = \frac{n - \sum_{i=1}^n \beta_i}{n}
 \end{aligned}$$

带有不确定性(4.18)的模型(4.11)~(4.14)用来当作考虑时延的预测模型。在解决每次运算周期内的最优化问题之前，考虑系统的稳定性，本文接下来先详细介绍如何构造鲁棒正不变集，以进一步作为预测的最终约束。

### 4.3 带延时的控制策略

假设每对  $(G_{fi}, H_f)$ ,  $i=0, \dots, n$  是可控的。在第一阶段，在无约束情况下找到了多面体模型的稳定反馈控制律，其次通过考虑约束条件(4-10)构造了一个正不变集。

#### 4.3.1 无约束情况下的稳定控制律

考虑线性系统(4.11)写成如下形式：

$$\begin{aligned}
 \xi_{k+1} &= G_f \xi_k + H_f u_k \\
 G_f &\in \Omega \\
 \Omega &= Co\{G_{f0}, \dots, G_{fn}\}
 \end{aligned} \tag{4.21}$$

假定稳定性控制律  $u_k = K \xi_k$  存在，且可以由线性矩阵不等式获得。

考虑一个无限时域的 min-max 控制问题：



$$\min_{u_k, u_{k+1}, u_{k+2} \dots} \max_{G_f \in \Omega} J_{\infty} \quad (4.22)$$

其中:

$$J_{\infty} = \sum_{i=0}^{\infty} \xi_{k+i}^T Q \xi_{k+i} + u_{k+1}^T V u_{k+1} \quad (4.23)$$

$$u_k = K \xi_k \quad (4.24)$$

$Q$ 、 $V$  为权重矩阵。

定义一个二次型函数代表  $J_{\infty}$  的上界:

$$V(\xi) = \xi^T P \xi, P \succ 0 \quad (4.25)$$

该二次型函数满足:

$$V(\xi_{k+i+1}) - V(\xi_{k+i}) \leq -[\xi_{k+i}^T Q \xi_{k+i} + u_{k+1}^T V u_{k+1}] \quad (4.26)$$

将式(4.24)和(4.25)代入(4.26)可获得:

$$\xi_{k+i}^T [(G_f + H_f K)^T P (G_f + H_f K) - P + K^T V K + Q] \xi_{k+i} \leq 0 \quad (4.27)$$

或者(4.27)可表示为:

$$(G_f + H_f K)^T P (G_f + H_f K) - P + K^T V K + Q \leq 0 \quad (4.28)$$

记  $P = \gamma S^{-1}$ ,  $S \succ I$ ,  $Y = KS$ , 则可构造如下矩阵不等式:

$$\begin{bmatrix} S & (G_f S + H_f Y)^T & S Q^{1/2} & Y^T R^{1/2} \\ G_f S + H_f Y & S & 0 & 0 \\ Q^{1/2} S & 0 & \gamma I & 0 \\ R^{1/2} Y & 0 & 0 & \gamma I \end{bmatrix} \succ 0 \quad (4.29)$$

并使用  $LMI$  工具求解如下最优化问题:

$$\min_{\gamma, S, Y} \gamma$$

$$\begin{bmatrix} S & (G_f S + H_f Y)^T & S Q^{1/2} & Y^T R^{1/2} \\ G_f S + H_f Y & S & 0 & 0 \\ Q^{1/2} S & 0 & \gamma I & 0 \\ R^{1/2} Y & 0 & 0 & \gamma I \end{bmatrix} \succ 0, i = 0, \dots, n \quad (4.30)$$

并最终求得  $K = YS^{-1}$ 。

#### 4.3.2 最大输出允许集合

针对于带约束问题的情况, 首先将约束条件(4.10)表示成:

$$\Gamma \xi_k + D u_k \leq W \quad (4.31)$$

使用式(4.30)获取稳定控制律  $u_k = K \xi_k = YS^{-1} \xi_k$ , 则存在可以用增广状态表示的多面体

域:

$$Zone = \{\xi \in R^{(n+d \cdot m)} | (\Gamma + DK)\xi \leq W\} \quad (4.32)$$

结合(4.21)和(4.32)式得最大输出允许集合可定义为:

$$O_\infty^\Omega = \{\xi_0 | \Phi^k \xi_0 \in Zone, \forall \Phi \in \Omega_K, \forall k \in N\} \quad (4.33)$$

其中:  $\Omega_K = Co\{(G_{f_0} + H_f K); \dots; (G_{f_n} + H_f K)\}$ 。

### 4.3.3 控制策略

模型预测控制得最终目的是为了获得最优控制序列  $k_u = \{u_k, \dots, u_{k+N-d}\}$ 。针对有延时的问题, 式(3.45)表述成如下形式:

$$\begin{aligned} k_u^* = \arg \min_{k_u} \{ & \max_i \{ \xi_{k+N_p}^{(i)T} P \xi_{k+N}^{(i)} + \sum_{j=0}^{N_p-1} \xi_{k+j}^{(i)T} Q \xi_{k+j}^{(i)} + u_{k+j}^T V u_{k+j} \} \} \\ \text{subject to: } & \xi_{k+j+1}^{(i)} = G_{f_i} \xi_{k+j}^{(i)} + H_f u_{k+j} \\ & \Gamma \xi_{k+j}^{(i)} + D u_{k+j} \leq W, i=1, \dots, n; j=1, \dots, N_p \\ & \xi_{k+j}^{(i)} \in O_\infty^\Omega, i=1, \dots, n \end{aligned} \quad (4.34)$$

其实际意义为优化系统最坏情况下的最优性能。

## 4.4 本章小结

执行机构执行控制量时会发生时延, 而且时延时间越大, 系统越不稳定。并且在实际对系统进行建模时, 受客观条件约束, 比如转向机构在执行前轮偏角时, 由于存在系统时延, 使得所建立系统模型的参数跟实际系统参数之间存在着一定的偏差, 这势必会影响系统的稳定性等问题。针对车辆平台转向机构时延对控制效果的影响问题, 本文先对通过下发一系列的期望前轮控制量, 并对比反馈的实际前轮偏角, 获取时延大小的范围。由时延时间在实际转向过程中不是一个定值, 这将给系统模型带来不确定性。针对这种带有不确定性的控制问题, 其解决办法基本步骤如下: 首先, 构造参数时变多面体系统, 其次求解参数时变多面体不确定系统的稳定性状态反馈控制率, 然后结合原系统约束和反馈控制律构建鲁棒正向不变集作为参数不确定性系统的约束, 最后求解时变多面体系统的滚动时域优化问题。

## 第5章 仿真与实车验证

MPC 算法是基于最优控制理论的一类方法，它依靠建立优化目标函数，以目标函数值最小为目标，综合考虑多种约束条件，获得未来一段时间的最优控制量序列。为此，需要基于给定的车辆参考模型，结合车辆当前状态对未来一段时域内的轨迹进行预测，与对应的参考路段对比得到偏差，作为目标函数的优化项之一；同时，控制量的变化率、控制量大小等也都可以设置为目标函数的优化项；不同优化项通过设置权重来确定优化的优先级。

先通过不同仿真实验，不同的仿真实验对应仿真目的如下：

1) 设定参考轨迹为定曲率的期望轨迹，即圆弧，给定相同采样周期  $h_k=0.12s$ ，取不同时延  $0.5 h_k$ 、 $0.7 h_k$ 、 $1 h_k$  的情况，分析不同时延情况对模型预测控制跟踪器的影响；

2) 为了验证模型预测算法的可靠性，设定参考轨迹为双移线，给定初始航向偏差和航向偏差，对比纯跟踪和模型预测控制的在定速工况下的仿真结果来验证 MPC 算法，验证本文提出的算法适应曲率变化的双移线；

然后验证模型预测控制器在实车上运行的可行性，对此我们在不加任何时延处理对比实车测试期望前轮偏角和实际前轮偏角曲线，然后在算法里补偿，对比时延前后的效果，并使用比亚迪唐无人驾驶平台在越野环境实车测试。

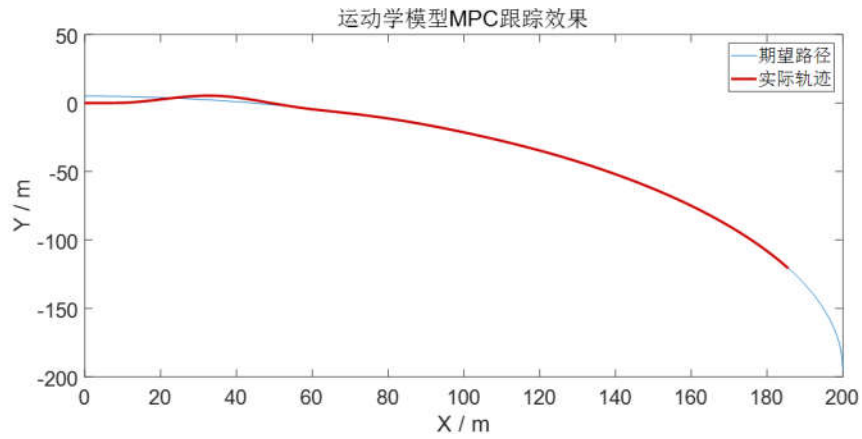
### 5.1 MPC 仿真实验

#### 5.1.1 MPC 仿真实验

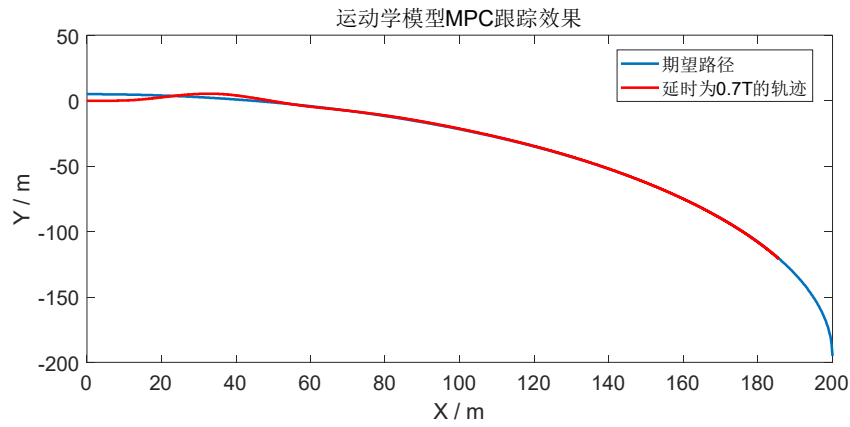
1) 仿真实验 1: 时延对模型预测轨迹跟踪器的影响及时延补偿的效果

仿真实验介绍：车辆起始位姿为  $x=[X,Y,\theta]^T=[0,0,0]$ ，跟踪期望轨迹为： $Y=\sqrt{200^2-X^2}-195$ ，车速为 10m/s 匀速逆时针跟踪，采样周期  $h_k$ ，分析不同时延  $0.5h_k$ 、 $0.7 h_k$  和  $1h_k$  对模型预测轨迹跟踪器的影响，并对比加入时延补偿后的效果。

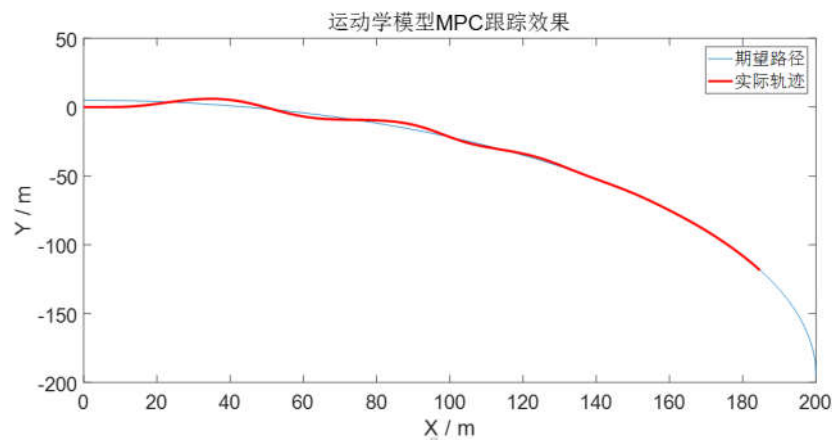
时延为  $0.5h_k$  跟踪结果：



时延为  $0.7 h_k$  跟踪结果:



时延为  $1 h_k$  跟踪结果:



$0.5-1 h_k$  时延补偿后的跟踪结果:

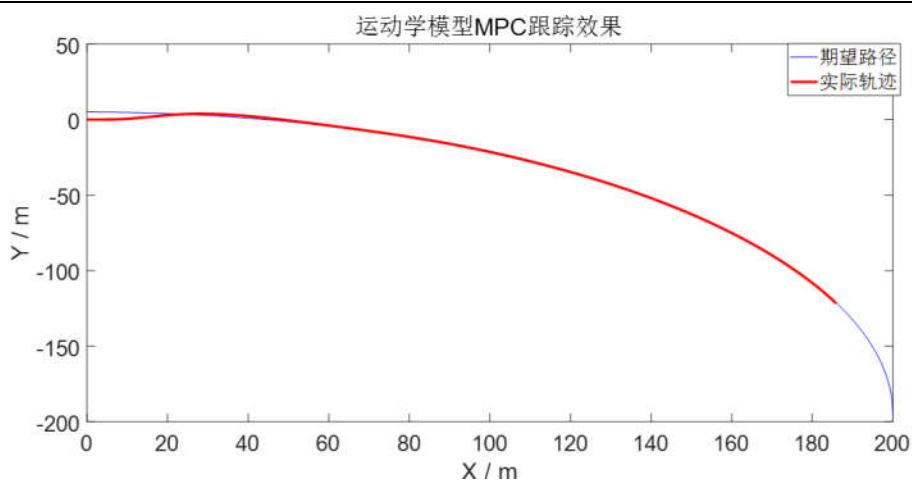
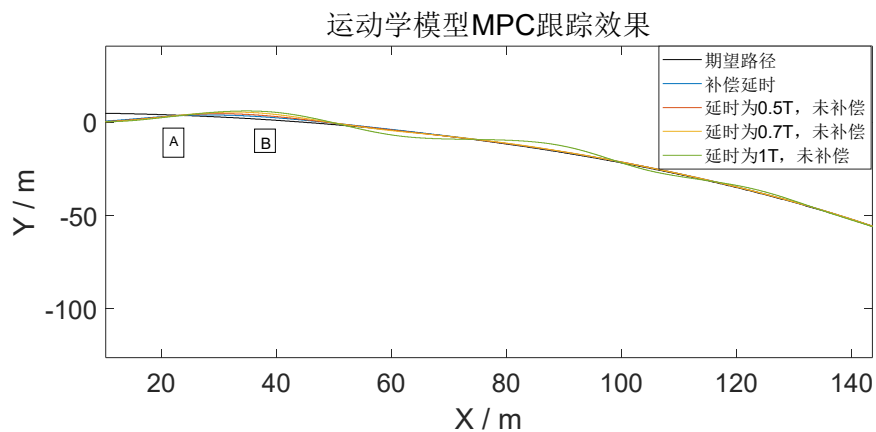


图 5.4 时延补偿后跟踪结果

时延  $0.5 h_k$ ,  $0.7 h_k$ ,  $1 h_k$  和时延补偿对比结果如下:

图 5.5 时延  $0.5 h_k$ ,  $0.7 h_k$ ,  $1 h_k$  和时延补偿对比结果

对上图 A、B 两处局部放大:

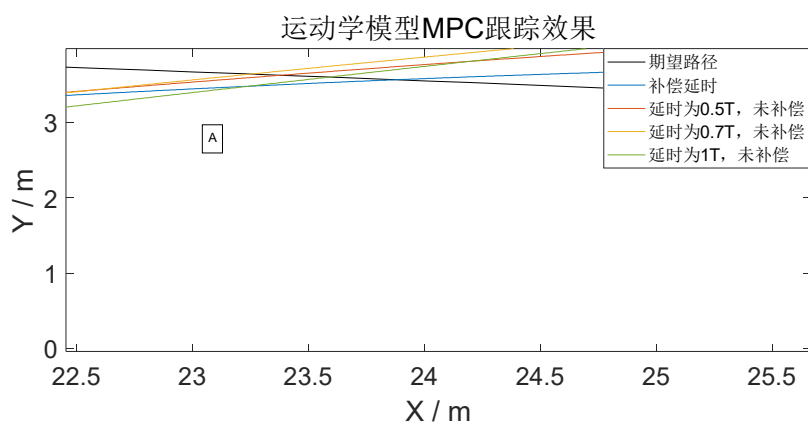


图 5.6(a) A 处局部放大

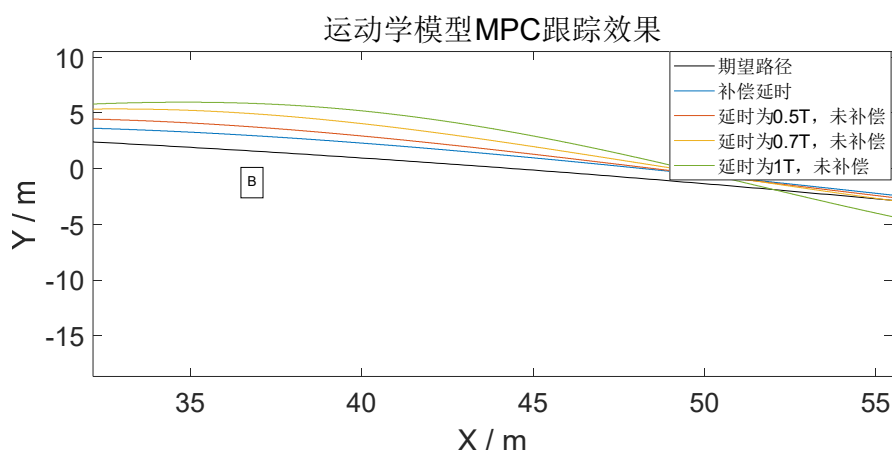


图 5.6(b) B 处局部放大

结果分析：对比执行时延  $0.5 h_k$ ,  $0.7 h_k$ ,  $1 h_k$  仿真实验结果发现执行时延对跟踪结果也有直接影响，时延越短，跟踪结果越好，时延过大，跟踪结果趋于发散。为了减少执行时延的影响，对此我们采用第 4 章方法，仿真结果表明该方法可以很好地适用于时延的情况。

## 2) 仿真实验 2: MATLAB 和 CarSim 联合仿真平台下 MPC 算法的相对于基于几何方法对比

在次仿真实验之前，本文介绍仿真平台相关工具，其中包括 MATLAB MPC 工具箱和 CarSim 仿真平台。

### (1) MPC 常用工具箱，其中包含

#### ① MATLAB 中自带的 MPC 工具箱

MATLAB 是由美国 MathWorks 公司出品的商业数学软件，用于算法开发、数据可视化、数据分析以及数值计算的高级技术计算语言和交互式环境，目前最新版本为 R2018a。MATLAB 中的模型预测控制工具箱（Model Predictive Control Toolbox）提供了一系列用于模型预测控制的分析、设计和仿真的函数。这些函数的类型主要有：

系统模型辨识函数。主要功能包括通过多变量线性回归方法计算系统的脉冲响应模型、阶跃响应模型等。

模型建立和转换函数。主要功能包括建立状态空间模型以及状态空间模型和脉冲响应模型、阶跃响应模型之间的转换等。

模型预测控制器设计和仿真工具。主要功能包括面向各类模型的预测控制器设计与仿真函数。

系统分析工具。主要功能包括计算系统频率响应、极点和奇异值。

以上每一类函数中，都有多种函数可供选择，在进行模型预测控制器的设计过程中，应根据系统自身的特点以及控制器的要求，有针对性的进行选用。详细的功能定义可以通过阅读 MATLAB 的帮助文件获得。

为了方便用户，从 MATLAB 7.01 版本开始，MPC 工具箱中增加了一个基于模型预测控制设计工具（Model Predictive Control Design Tool）的图形用户界面（GUI），启动界面如图所示。使用者可以根据菜单提示要求，一步步进行，非常适合初学者。对于设计好的模型预测控制器，可以利用 Simulink 库浏览窗口中 MPC 工具箱所提供的模型预测控制器模块 MPC Controller 进行调用，然后在 Simulink 环境下对复杂的控制系统进行仿真分析。

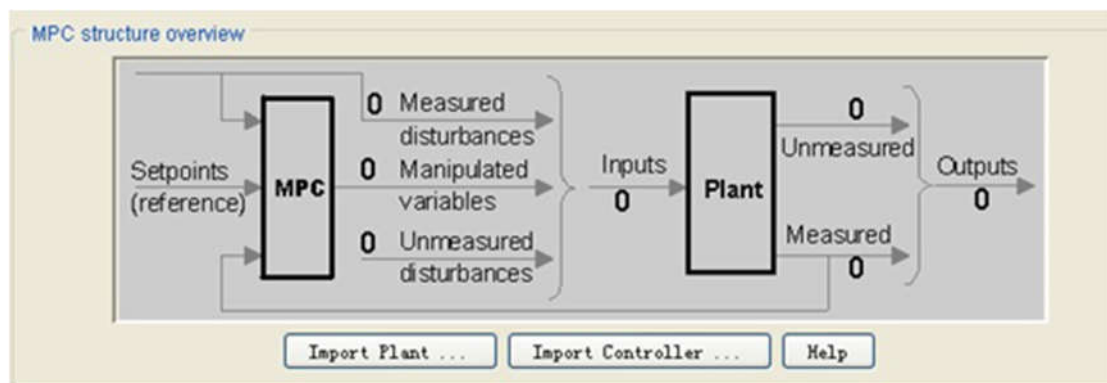


图 5.7 模型预测控制设计工具的图形界面

## ② MPCtools 工具箱

MPCtools 是由瑞典隆德理工学院的 Johan Akesson 教授等人开发的模型预测控制器设计与开发工具箱，目前版本为 1.0。和 MATLAB 自带的 MPC 工具箱类似，MPCtools 需要 MATLAB 环境的支持，同时也是通过调用相应的功能函数实现模型预测控制器的设计。用户只需要在第一次使用的时候将工具箱所在的地址加载到 MATLAB 的工作目录中，就可以正常使用。

MPCtools 工具箱通过 5 个函数完成整个模型预测控制器的设计，分别是 MPCInit、MPCSim、MPCfrsp、MPCOptimizeSol 和 MPCController。详细功能如表 5.1 所示：

表 5.1 MPCtools 主要函数及功能

序号	函数名称	功能简介
1	MPCInit	模型预测控制器的初始化
2	MPCSim	MATLAB 环境下对线性系统的控制过程仿真
3	MPCfrsp	计算模型预测控制控制器的频率响应
4	MPCOptimizeSol	优化目标函数的最优化求解
5	MPCController	Simulink 环境下控制器的搭建

MPCtools 自发布以来,受到了全世界自动控制领域学者的欢迎,它主要具有以下优点:(1)免费。该工具箱目前可以从互联网免费下载和使用,并且可以与世界各地的用户进行交流。(2)函数功能定义简洁明了,易于上手。整个工具箱主要的功能函数就只有 5 个并且各自的功能非常清晰,只要按照规定的调用格式就可以轻松完成模型预测控制器的设计。(3)支持底层函数的改写。该工具箱所有代码都是开放的,用户可以根据自己的使用需求对底层调用的函数进行改写,例如最优化函数的求解方式,用户可以选择有效集法、内点法或者其他解法。但 MPCtools 也有一些缺点,主要体现在:(1)只能处理线性模型,对于非线性系统,需要线性化后使用。(2)优化目标函数和约束条件的格式固定不变,开发复杂的控制器比较困难。

## (2) CarSim 软件介绍

CarSim 是由美国 MSC 公司(Mechanical Simulation Corporation)开发的车辆动力学仿真软件。它的动力学模型基础来源于密歇根大学交通运输研究所(University of Michigan Transportation Research Institute, UMTRI)多年理论和实践经验的积累。CarSim 可以方便灵活的定义试验环境和试验过程,准确预测和仿真汽车整车的操纵稳定性、动力性、平顺性等,适用于轿车、轻型货车等车型的建模仿真,其主要的功能界面如图 4.4 所示。从图中可以看出,CarSim 包含主要包含三大功能模块:①模型和工况参数设定②数学模型求解③输出和后处理。

### ① 模型和工况参数设定

该功能模块主要包括车辆参数设定与车辆测试工况设定两部分。其中车辆参数设定需要从已有的整车模型数据库选择合适的车型,并对动力传动系统、悬架系统等进



行设置。车辆测试工况设定主要是对车辆行驶环境及相关因素进行设置，包括路面信息、风阻等外部环境信息。

## ② 数学模型求解

该功能模块是整个软件求解运算的内核，需要设置求解器类型、仿真步长和仿真时间等信息。同时，该模块也是与其他所有外部环境的接口，包括 Simulink、Labview 和 dSPACE 等。

## ③ 输出和后处理部分

该功能模型功能包括仿真结果的 3D 动画显示和数据绘图两部分。用户可以通过仿真动画窗口直观地观察汽车动力学响应，还可以在数据绘图部分有选择地输出指定参数的曲线，进行定量分析。



图 5.8 CarSim 仿真软件

## (3) 仿真场景搭建与试验

仿真工况设定主要包括初始速度、驾驶员制动模式、转向模式、档位输入和路面参数等。由于所设计的轨迹跟踪控制器实时向车辆提供车轮转角信息和速度信息，并且加入了速度控制模块，因此制动模式和转向模式均选择为无，即不需要模型自身施加制动力和转向力，档位控制选择自动升档。路面参数设定需要设定道路曲线、宽度、摩擦系数等信息。为了验证模型预测算法的可靠性，设定参考轨迹为双移线。搭建的仿真模型、仿真参数和仿真道路场景如下：

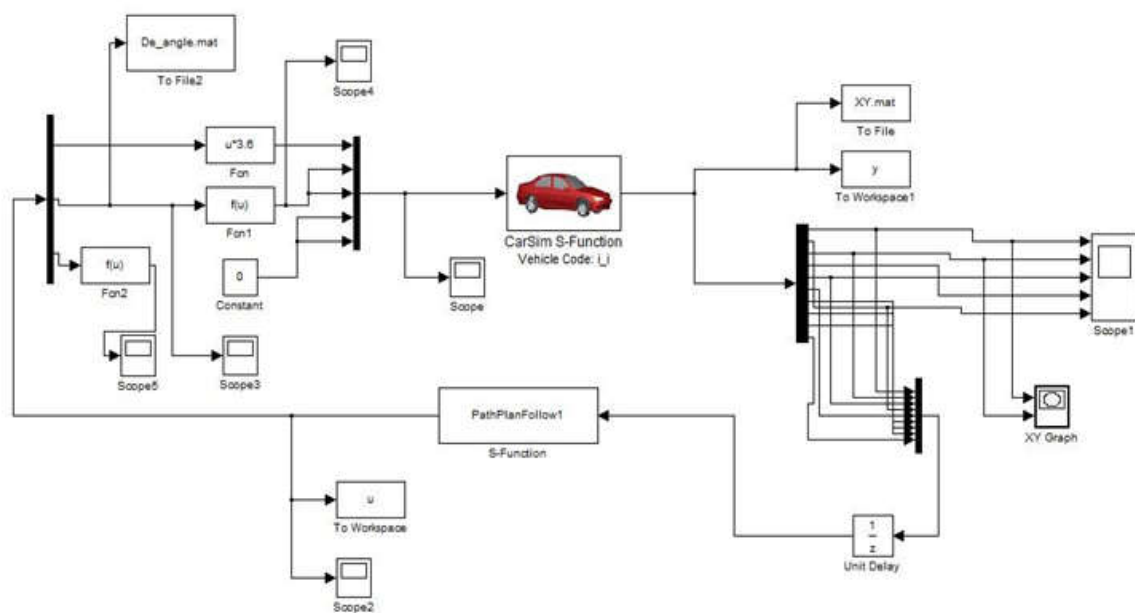


图 5.9 仿真模型

控制系统的建模与开发大多是在 MATLAB/Simulink 环境中完成的。在该环境中，开发人员可以利用图形建模功能编写复杂的控制逻辑。同时，世界上主流动力学仿真与分析软件都提供了与 Simulink 仿真的接口，所以本文选择 Simulink 环境进行控制系统建模。

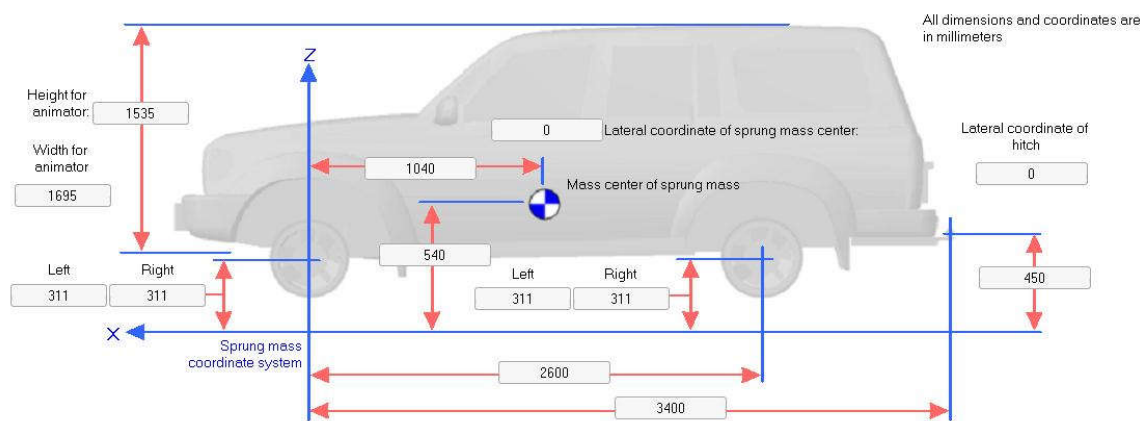


图 5.10 仿真参数设置

仿真中车辆参数如图 5.10 所示，仿真时车辆速度保持 10m/s 不变 Carsim 模块接收的输入为车速、车轮转向角，其中左右后轮转向角为 0，前轮偏角为 pathfollow (MPC) 模块计算出。输出为车辆在全局坐标中的左右后轮位置、车辆航向、左右前轮的实时转向角。

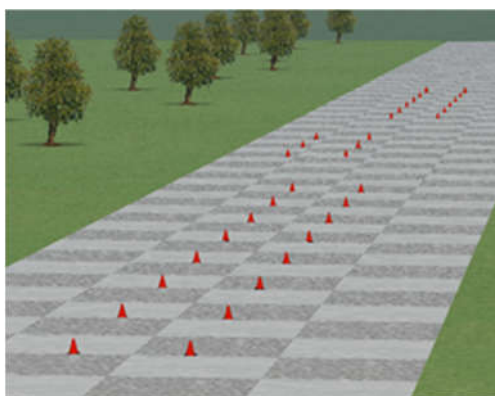
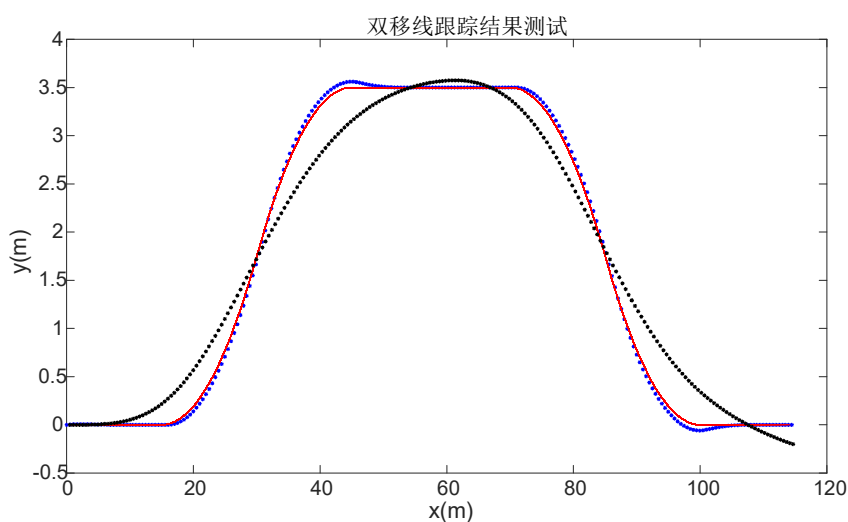


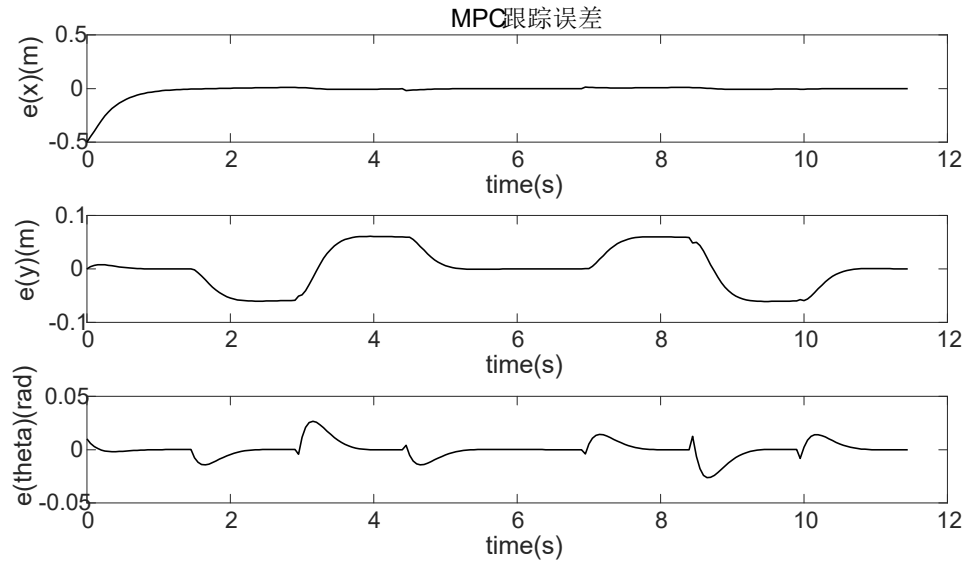
图 5.11 仿真场景

给定初始航向偏差和航向偏差，对比纯跟踪和模型预测控制的在定速工况下的仿真结果来验证 MPC 算法。

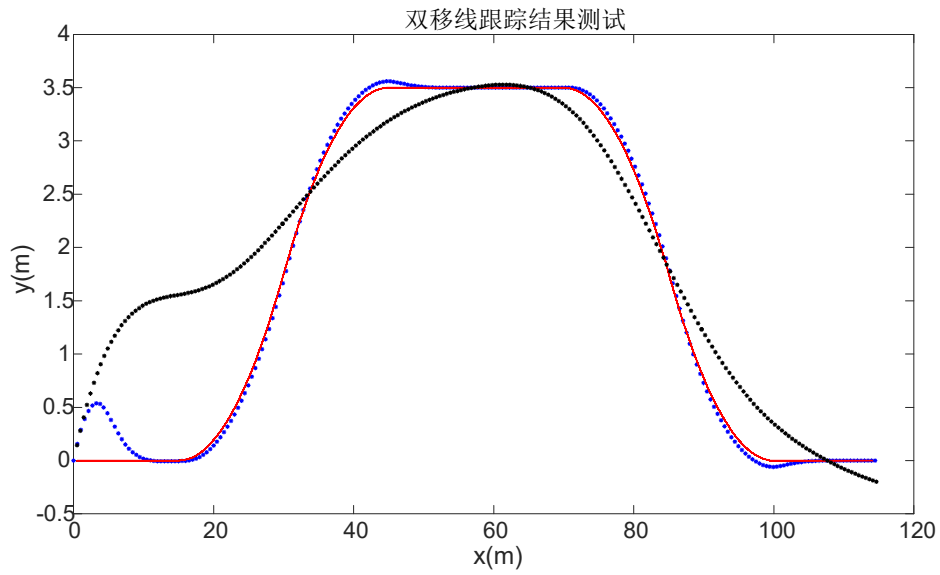
仿真实验介绍：给定参考轨迹为双移线，分别采用纯跟踪算法和模型预测控制算法在不同初始位姿  $(x, y, \varphi)$  对参考轨迹跟踪。在无定位误差情况下给定初始位姿  $(0, 0, 0)$ 、 $(0, 0, \pi/10)$ 、 $(0, 1, 0)$  和  $(0, 1, \pi/10)$ ，车速为 10m/s 跟踪该参考曲线。纯跟踪的参数预瞄距离为 15m，模型预测控制参数为目标函数里的权重配置为： $w_d = 1, w_\theta = 10, w_k = 5$ （以下仿真实验参数相同）。

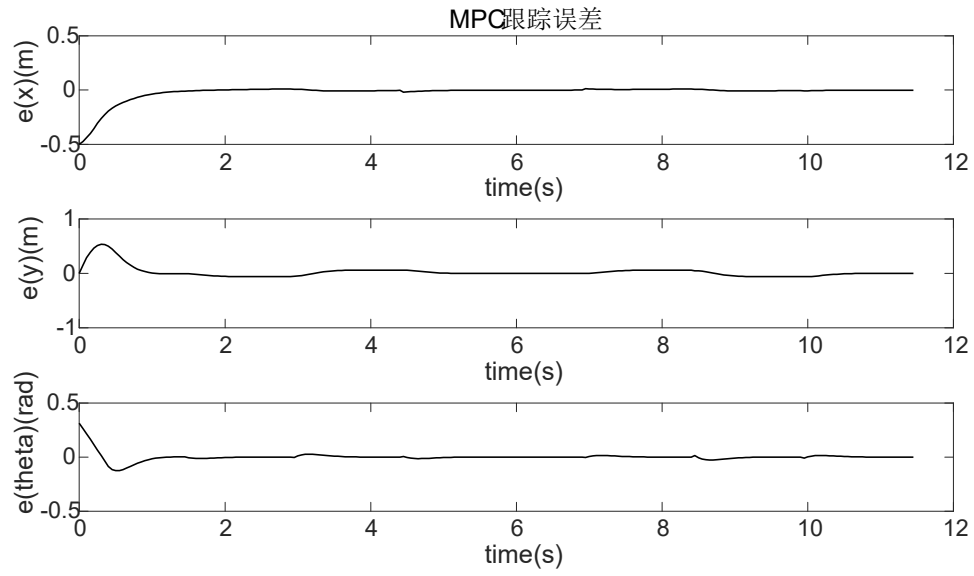
初始位姿为  $(0, 0, 0)$  跟踪结果如图 5.12 所示，其中红线为参考路径，蓝色点为 MPC 跟踪轨迹，黑色点为纯跟踪的轨迹（下同）。在初始位姿  $(0, 0, 0)$  情况下的跟踪误差如图 5.13 所示，图中三项误差分别代表  $x$  方向的偏差、 $y$  方向的偏差和航向偏差（下同）。

图 5.12 起始位姿为  $(0, 0, 0)$ ，双移线跟踪测试宏观对比

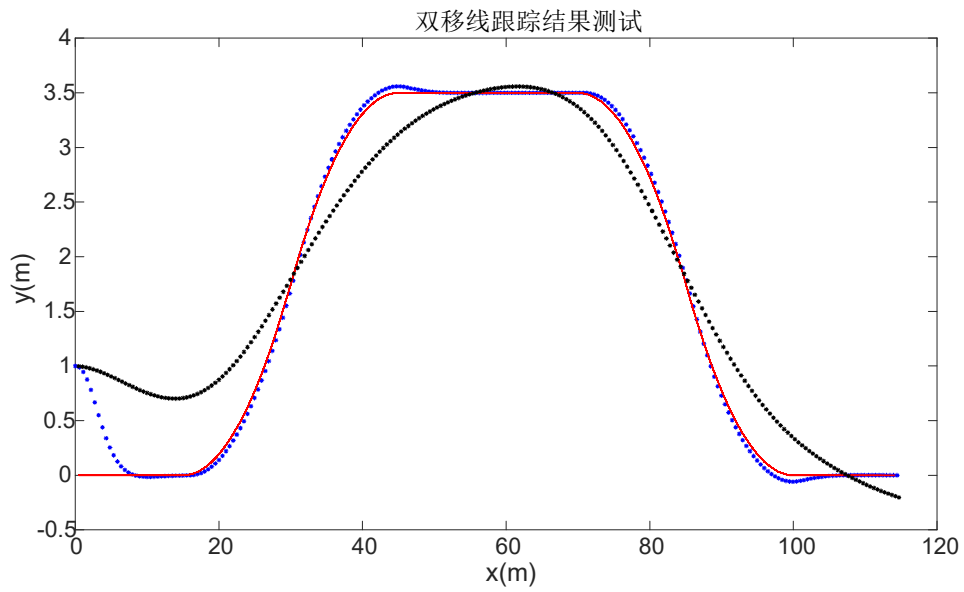
图 5.13 起始位姿为  $(0,0,0)$ ，MPC 跟踪误差

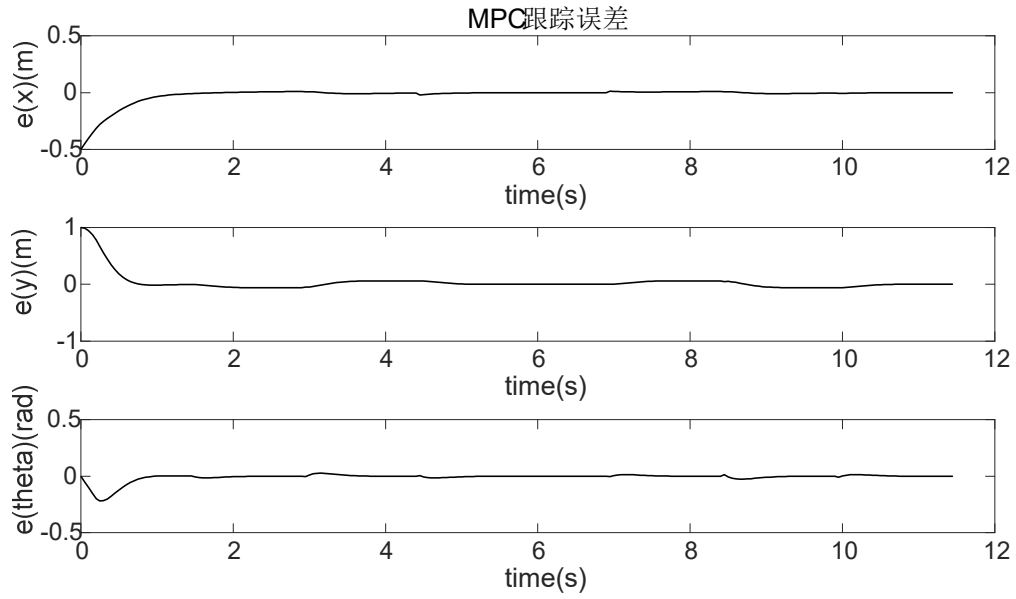
初始位姿为  $(0,0,\pi/10)$  跟踪结果如图 5.14 所示，MPC 的跟踪误差如图 5.15 所示。

图 5.14 起始位姿为  $(0,0,\pi/10)$ ，双移线跟踪测试宏观对比

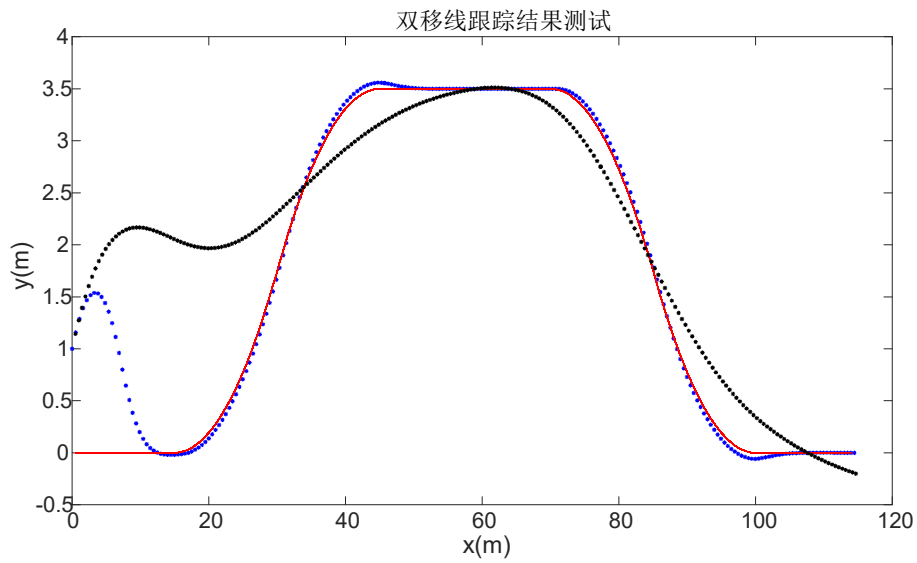
图 5.15 起始位姿为 $(0,0,\pi/10)$ ，MPC 跟踪误差

初始位姿为 $(0,1,0)$ 跟踪结果如图 5.16 所示，MPC 的跟踪误差如图 5.17 所示。

图 5.16 初始位姿为 $(0,1,0)$ 跟踪结果

图 5.17 初始位姿为 $(0,1,0)$ ，MPC 跟踪误差

初始位姿为 $(0,1,\pi/10)$ 跟踪结果如图 5.18 所示，MPC 的跟踪误差如图 5.19 所示。

图 5.18 初始位姿为 $(0,1,\pi/10)$ 跟踪结果

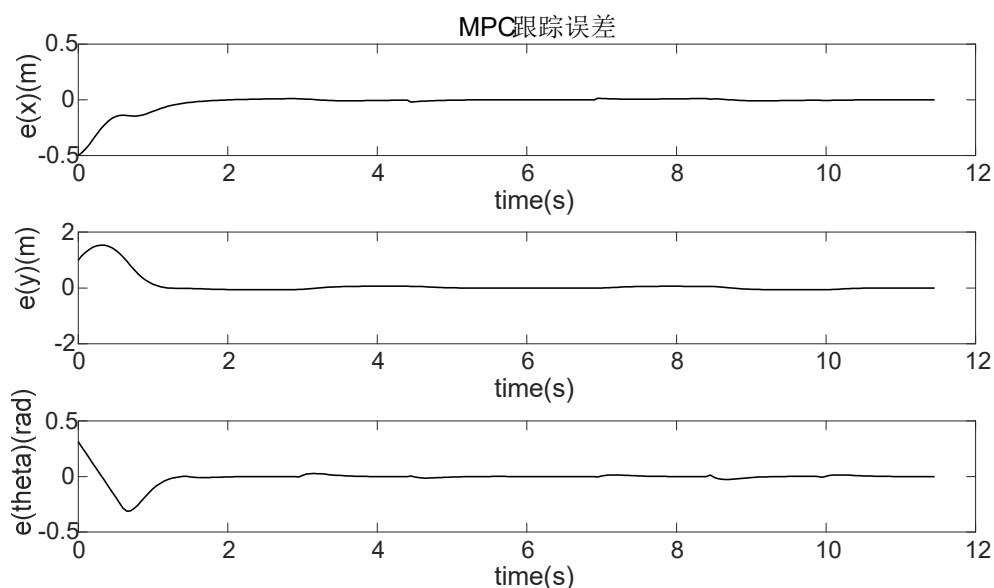


图 5.19 初始位姿为 $(0,1,\pi/10)$ ，MPC 跟踪误差结果

实验结果分析： 初始位姿为 $(0,0,0)$ 时，即初始位姿与参考路径没有横向偏差和航向偏差，使用 MPC 跟踪期望路径，跟踪结果如图 5.12 和 5.13 所示，MPC 跟踪轨迹基本与参考路径重合，横向偏差在正负 0.1m 以内，而纯跟踪效果较差。初始位姿为 $(0,0,\pi/10)$ 时，起始位姿与参考路径航向偏差较大，使用本文提出的跟踪算法，跟踪结果如图 5.14 和 5.15 所示，即使在起始点有较大的航向偏差情况下，最大横向偏差的绝对值小于 0.5，说明该算法依然能很好地跟踪上期望路径。初始位姿为 $(0,1,0)$ 时，起始位姿与参考路径横向偏差较大，使用 MPC 跟踪期望路径，跟踪结果如图 5.16 和 5.17 所示，实验结果表示该算法能让横向偏差快速收敛至 0。初始位姿为 $(0,1,\pi/10)$ 时，起始位姿与参考路径之间有较大的横向偏差和航向偏差，使用本文提出的跟踪算法，跟踪结果如图 5.18 和 5.19 所示，车辆与参考路径的横向偏差和航向偏差可快速收敛至 0，前横向偏差的最大值为 1.5，航向偏差的最大值小于 0.5 弧度。说明本文提出的算法很好地适应双移线这种曲率变化的道路。

### 5.1.2 仿真实验结果分析

道路曲率是影响运动控制的一大因素。本文提出一种考虑道路特性的模型预测控制算法。首先，为了获取道路参数，我们首先对参考路径进行多项式拟合，以后轴中心到参考路径的距离、车辆位姿、轨迹曲率、车辆与参考轨迹的航向偏差，以前轮偏角作为控制量建立预测模型，在此基础上建立横向偏差和航向偏差限制的目标函数并

添加状态约束和控制量约束，进而用二次规划算法求解带约束的目标函数。并在此基础上采用变采样周期的方法，使得跟踪器可以很好地跟踪期望路径。

## 5.2 实车实验

### 5.2.1 实验内容

前面验证了 MPC 算法在仿真环境下的可行性和稳定性。为了进一步验证算法在实车上的运行的可行性和稳定性，本文采用丰田无人驾驶平台如图 5.20 所示。为了尽可能减少定位误差的干扰，无人驾驶平台搭载 GPS 及千寻网络差分设备，车辆定位在无遮挡区域可达到厘米级精度，且底层控制平台为全线控（即车辆状态可通过 can 总线往外发送，车辆的油门、制动、挡位和方向盘可根据 can 上接收到的指令经行精确控制）。

为了验证算法针对不同道路曲率的适应性，我们设计不同道路场景，其中包括蛇形弯场景、平坦复杂道路和越野复杂道路。

本平台的控制部分，软件框架如图 5.21 所示。



图 5.20 丰田无人驾驶平台



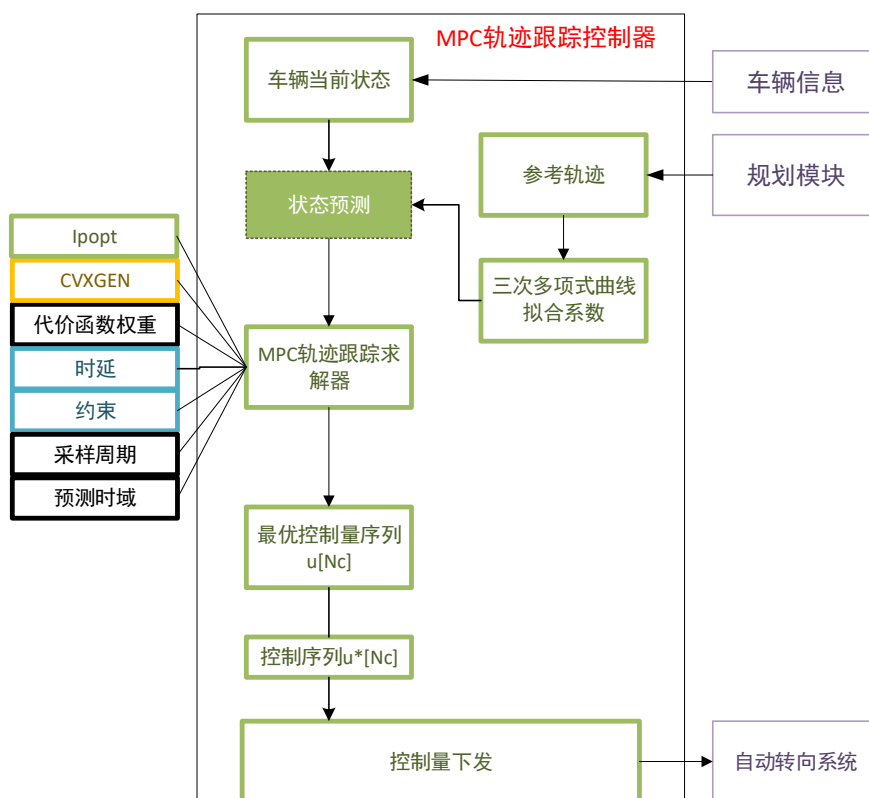


图 5.21 无人驾驶平台运动控制框架

### 5.2.2 试验方法和步骤

为了实车验证本文提出的方法的效果，本文采用循迹的方式即跟踪固定路径，具体实验步骤：

(1)人为沿某一路线驾驶，按固定周期（一般为 0.1s）将车辆的运行轨迹中的经纬度保存成路网的格式，并将路网当作参考路径；

(2)将期望路径转换成米制；

(3)启动程序，载入路网文件，对车辆实时位置也转换成米制，将参考路径转换至车体坐标系，适用模型预测控制器对参考路径进行跟踪。在局部路径规划中需要将全局坐标系下的全局期望路径  $PG$  转成局部坐标下的全局期望路径  $PL$ 。路径是由一系列的点组成，下面以全局期望路径上一点  $pi$  的坐标转化来说明。平面直角坐标系中通过横纵坐标和航向即可确定一点的位置，那么  $pi$  在全局坐标系中的位置为  $(x_G, y_G, \varphi_G)$ ，在局部坐标系  $pi$  位置  $(x_L, y_L, \varphi_L)$ ，其转换关系可以由式(5.1)计算得出。

$$\begin{aligned}
x_L &= (x_G - x_V) \sin \varphi - (y_G - y_V) \cos \varphi_V \\
y_L &= (x_G - x_V) \cos \varphi + (y_G - y_V) \sin \varphi_V \\
\varphi_L &= \frac{\pi}{2} + \varphi_G - \varphi_V
\end{aligned} \tag{5.1}$$

式中,  $(x_V, y_V, \varphi_V)$  为组合导航系统测量的当前车辆在惯性坐标系下的位姿。

### 5.2.3 误差计算

为了分析横向偏差, 我们需将参考轨迹和实际轨迹转化成米制坐标, 即经纬度转化成 WGS84 坐标<sup>[50]</sup>。WGS 参考系统是一个多椭球坐标系定义模型地球, 它被用作 GPS 坐标的参考系统。WGS 84 是一个大地坐标系: 它是参照一个 椭球模型与实际地球关系的大地测量基准。椭球坐标是三维的, 由纬度、经度和椭球高度组成。纬度是一个角度的赤道平面向北或向南测量; 经度是一个角度测量本初子午线的东或西; 椭球高指定椭球以上的高度, 垂直于其表面的方向, 通常用米来计量。为地球椭球模型是由两个参数定义: 半长轴、扁平率。半长轴从椭球中心到赤道的距离。扁平率为半长轴和半短轴之差比上长轴。从这两个参数中, 推导出包括半短轴与离心率在内的其他常数。设第一点 A 的经纬度为  $(longA, latA)$ , 第二点 B 的经纬度为  $(longB, latB)$ , 按照 0 度经线的基准, 东经取经度的正值(*longitude*), 西经取经度负值(*-longitude*), 北纬取 90-纬度值( $90-latititude$ ), 南纬取 90+纬度值( $90+latititude$ ), 则经过上述处理过后的两点被计为  $(mlongA, mlatA)$  和  $(mlongB, mlatB)$ 。令  $(\lambda, \phi) = (mlongA, mlatA), (mlongB, mlatB) = (0, 0)$ , 则所有位置的经纬度可转化为平面坐标系  $(x, y)$ , 转化公式如下:

$$x = k_0 N [A + (1 - T + C) A^3 / 6 + (5 - 18T + T^2 + 72C - 58e'^2) A^5 / 120] \tag{5.2}$$

$$\begin{aligned}
y = k_0 [M - M_0 + N \tan \phi [A^2 / 2 + (5 - T + 9C + 4C^2) A^4 / 24 + \\
(61 - 58T + T^2 + 600C - 330e'^2) A^6 / 720]]
\end{aligned} \tag{5.3}$$

$$\begin{aligned}
k_0 &= 0.9996 \\
e'^2 &= e^2(1 - e^2) \\
N &= a / (1 - e^2 \sin^2 \phi)^{1/2} \\
T &= \tan^2 \phi \\
C &= e'^2 \cos^2 \phi \\
A &= (\lambda - \lambda_0) \cos \phi \\
M &= a[(1 - e^2 / 4 - 3e^4 / 54 - 5e^6 / 256 - \dots)\phi \\
&\quad - (3e^2 / 8 + 3e^4 / 32 + 45e^6 / 1024 + \dots)\sin^2 \phi + \\
&\quad (15e^4 / 256 + 45e^6 / 1024 + \dots)\sin^4 \phi - \\
&\quad (35e^6 / 3072 + \dots)\sin^6 \phi + \dots]
\end{aligned} \tag{5.4}$$

表 1 WGS84 参数

<i>WGS 84 Parameter</i>	<i>value</i>
<i>Semi-major axis, a</i>	6387137 m
<i>Semi-minor axis, b</i>	6356752 m
<i>flattening, f</i>	1 / 298.257223563
<i>Eccentricity, e</i>	0.0818192

其中  $\lambda_0$  为所在区域的中央子午线。

获取了米制坐标后，与参考路径进行对比分析，判断实际轨迹点属于哪两个参考点之间，计算实际轨迹点到两参考点之间线段的距离，即为横向偏差。

#### 5.2.4 实验结果

本小节分析该算法在不同道路场景下的跟踪效果。

##### (1) 蛇形弯情况



图 5.22 蛇形弯场景

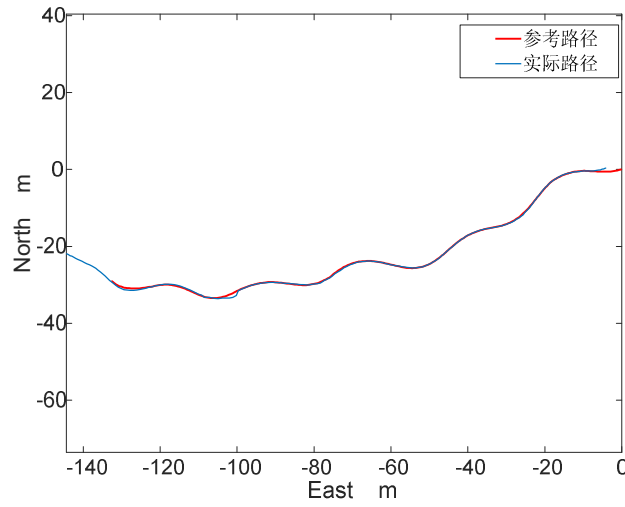


图 5.23 蛇形弯跟踪结果

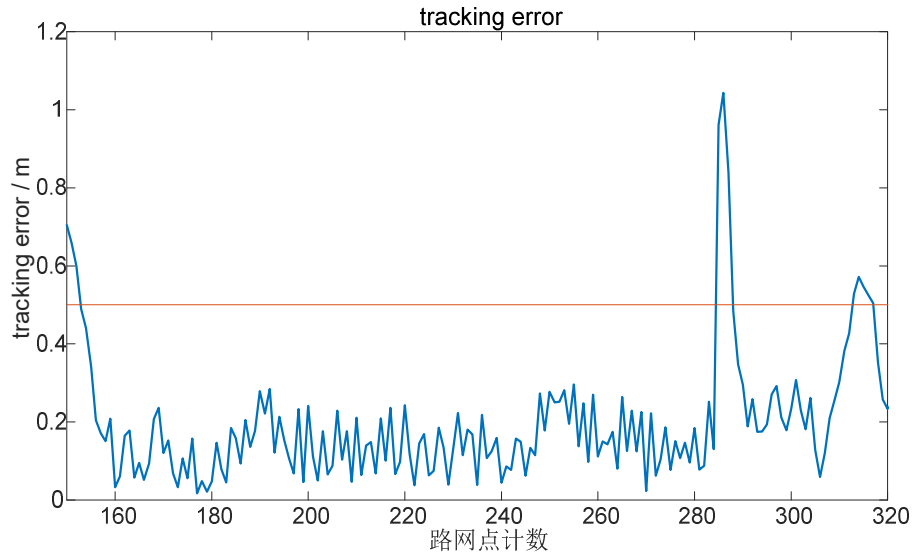


图 5.24 蛇形弯跟踪误差

图 5.24 说明：由于实际采集路径点时，启动程序车辆未启动到参考路径起点，保存了一些无用的点，故取起点处（路网计数点 150 处）的实际路网点开始分析。

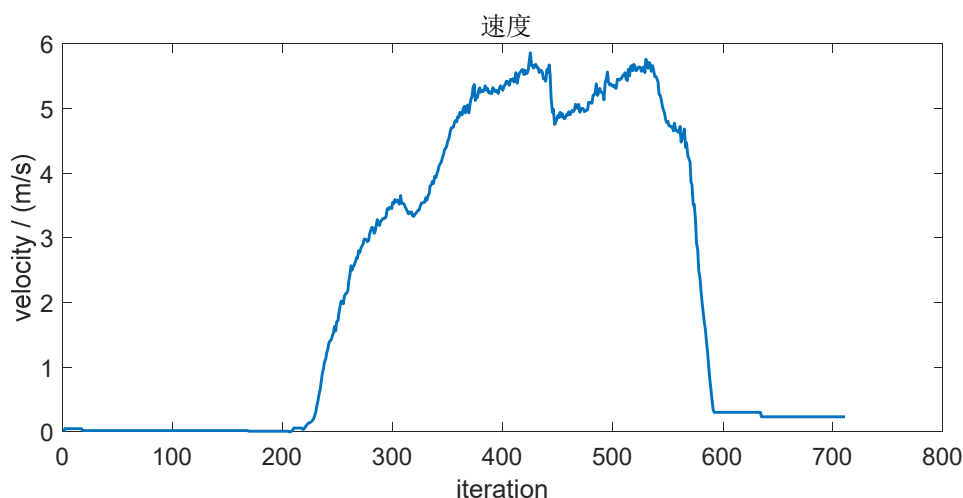


图 5.25 跟踪过程中速度

实验结果分析：在较为平坦的路面（如图 5.22 所示）以人为采集的路网为参考路径，如图 5.23 所示的红色路径，由图可知，参考路径为含有多个连续弯道的蛇形弯，弯道中，相邻波峰与波谷之间的距离小于 20m，其道路曲率一直在变化，给予模型预测控制较大难度。

由图 5.24 和图 5.25 可知，在行驶过程中，由于参考路径为蛇形弯，参考路径的曲率一直在变化，但车速稳定在 20km/h 左右，但跟踪误差基本稳定在 0.5m 以内，说明本文提出的算法很好地适用于道路曲率变化的工况。

## (2) 越野颠簸复杂道路



图 5.26 颠簸越野路面

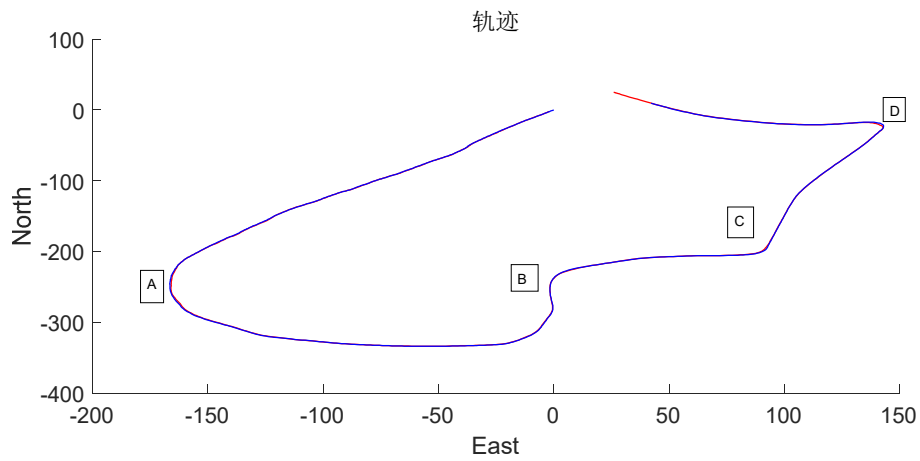


图 5.27 越野颠簸复杂道路跟踪结果

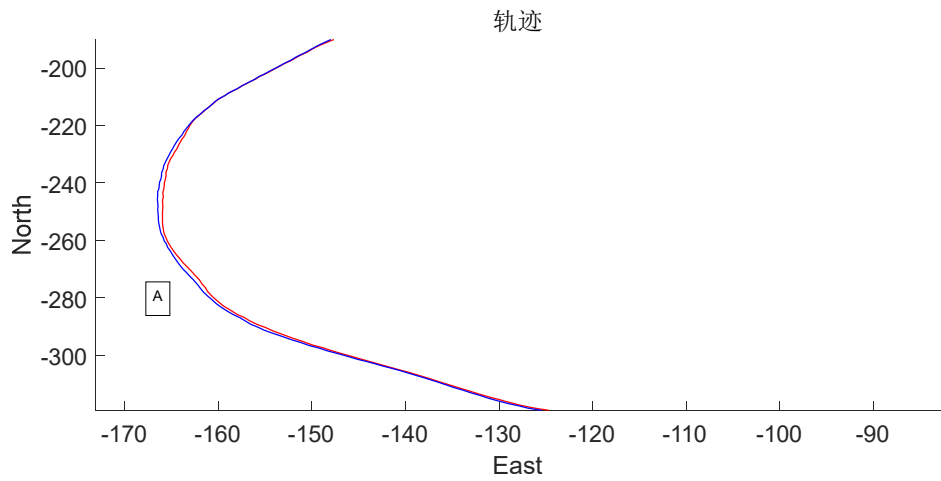


图 5.28 越野颠簸复杂道路跟踪结果局部放大 1

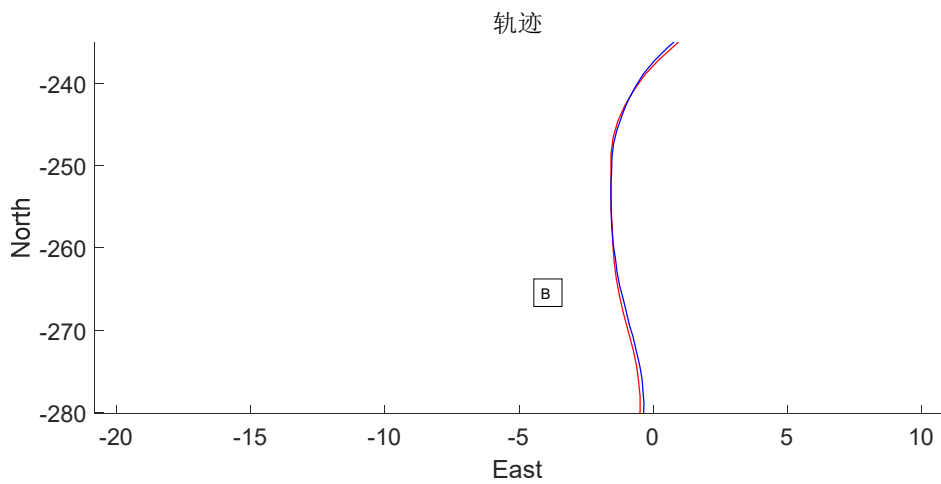


图 5.29 越野颠簸复杂道路跟踪结果局部放大 2

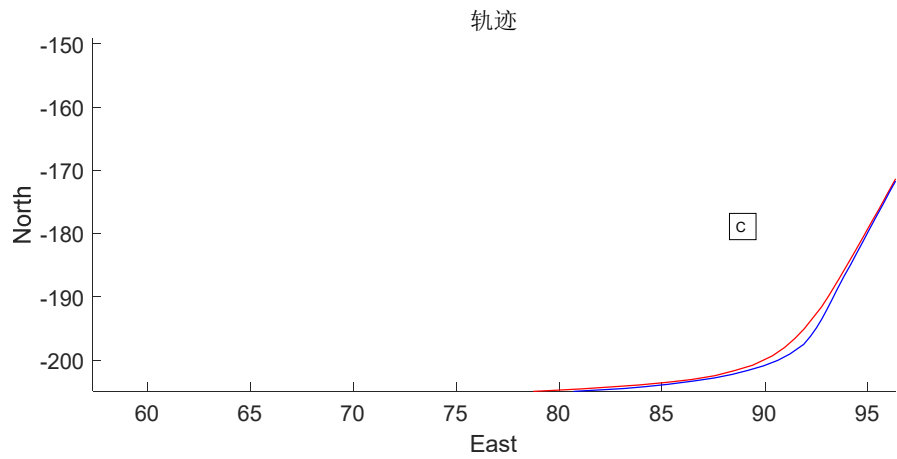


图 5.30 越野颠簸复杂道路跟踪结果局部放大 3

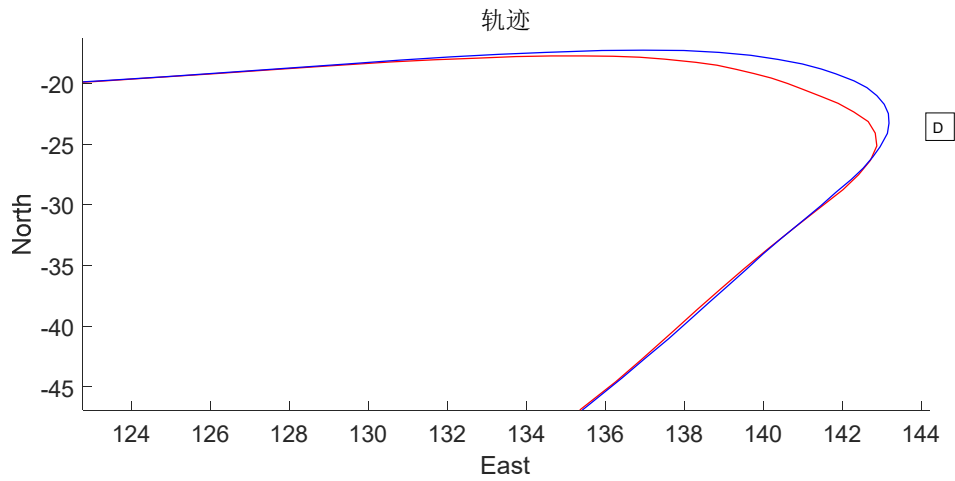


图 5.31 越野颠簸复杂道路跟踪结果局部放大 4

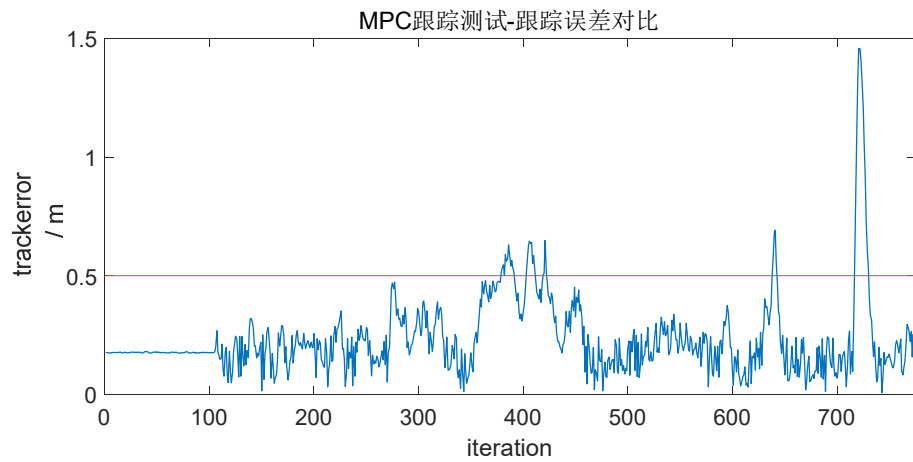


图 5.32 越野颠簸复杂道路跟踪误差

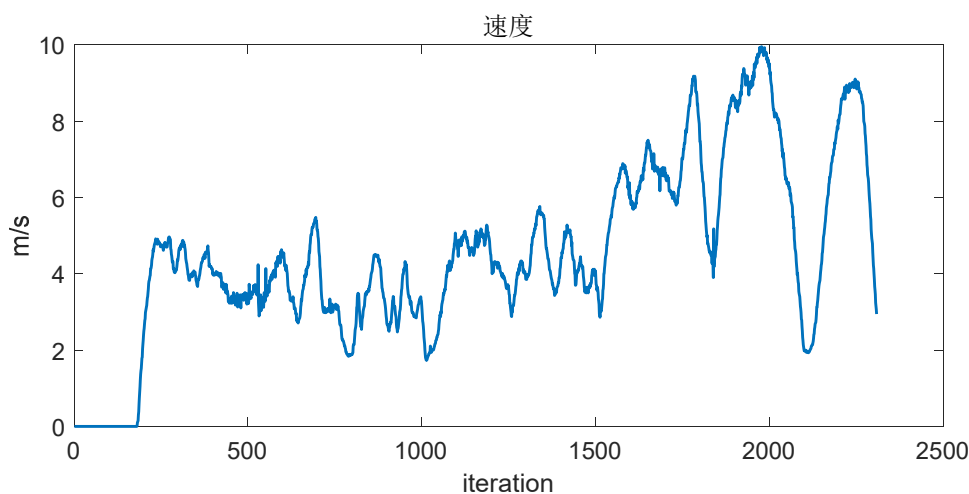


图 5.33 越野颠簸复杂道路跟踪速度

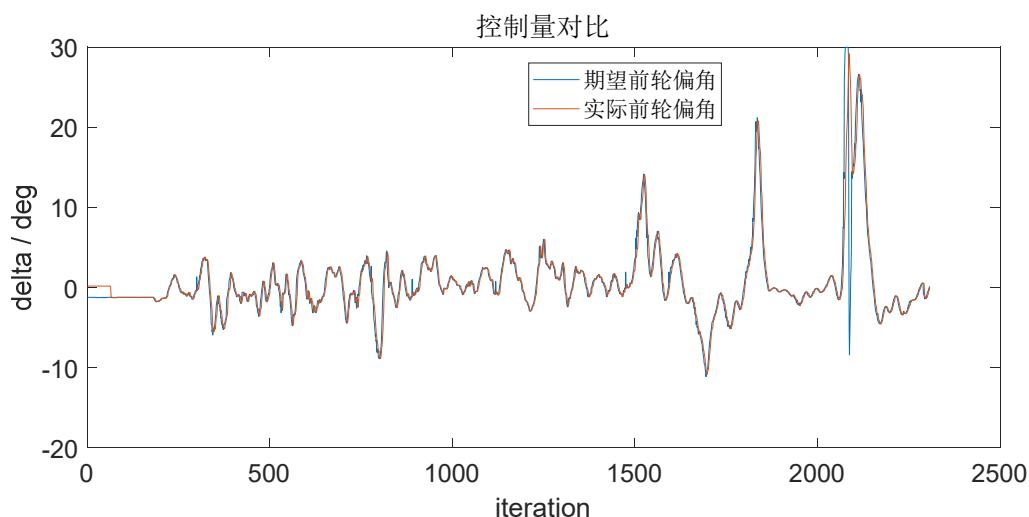


图 5.34 越野颠簸复杂道路跟踪前轮偏角变化

实验结果分析：图 5.27-5.31 横纵轴单位均为 m，图 5.32 的横轴为保存轨迹点的计数，图 5.33-5.34 横轴为程序周期(50ms)计数。在图 5.27 中我们可以看到 A、B、C、D 四处弯道，分别为小曲率弯、S 弯、直角弯和 U 形弯，除了在 D 处横向偏差大于 1m，其余全程跟踪误差基本小于 0.5m，D 处接近转弯极限半径，导致误差大于 1m。而且在行驶过程中，越野情况下的车速最大值也达到了 10m/s。从图 5.27-5.31 中可以看出，A 处转弯半径为 15m 左右，B 处为转弯半径 20m 左右的 S 弯，而 D 处转弯半径接近车辆极限转弯半径，基于横向偏差和航向偏差的变采样周期模型预测控制算法对不同曲率的弯道适应性很强，而且在接近极限转弯半径的弯道下，通行能力依然很好。



### 5.3 实验结论

(1) 由仿真和实车验证发现,采用 MPC 控制算法将优于基于道路几何的方法:针对这些问题,本文提出一种考虑道路特性的模型预测控制算法。首先,为了获取道路参数,我们首先对参考路径进行多项式拟合,以后轴中心到参考路径的距离、车辆位姿、车辆与参考轨迹的航向偏差作为状态量,以前轮偏角作为控制量建立预测模型,在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束,进而用二次规划算法求解带约束的目标函数。此外,单一的采样周期很难适应所有工况,设计一种随道路曲率的自适应采样周期的方法。使用本文提出的运动学模型将很好地适应各种曲率的弯道,而且跟踪误差基本限制在 0.5m 以内,可以达到实际运用的效果。

(2) 时延将导致模型预测控制器跟踪不稳定,而且时延越大,如果不加时延处理,跟踪效果越差。由于跟踪系统受客观条件时延的制约,使得所建立系统模型的参数跟实际系统参数之间存在着一定的偏差,这势必会影响系统的稳定性等问题。针对时延影响,考虑时延时间在实际转向过程中不是一个定值,这将给系统模型带来不确定性。为了解决模型的不确定性问题,本文将模型表示成增广矩阵,并将系统矩阵表示成凸集的凸组合。为了保证系统的稳定性,构建鲁棒正向不变集,并用来约束终端状态。最后在满足系统稳定性约束的基础上求解滚动时域优化问题。仿真试验证明,本方法有减少免时延带来的影响。

## 总结与展望

### 主要工作贡献

(1) 为了实现适应各种道路曲率的跟踪, 本文提出一种采样周期随道路曲率变化的模型预测控制方法, 在该方法中, 首先, 我们对参考路径离散的点拟合获取曲线的方程, 再以后轴中心到参考路径的距离、车辆位姿、轨迹曲率、车辆与参考轨迹的航向偏差作为状态量, 以前轮偏角作为控制量建立预测模型, 在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束, 进而用二次规划算法求解带约束的目标函数。

(2) 针对车辆平台转向机构时延对控制效果的影响问题, 本文先对通过下发一系列的期望前轮控制量, 并对比反馈的实际前轮偏角, 获取时延大小的范围。由时延时间在实际转向过程中不是一个定值, 这将给系统模型带来不确定性。针对这种带有不确定性的控制问题, 其解决办法基本步骤如下: 首先, 构造参数时变多面体系统, 其次求解参数时变多面体不确定系统的稳定性状态反馈控制律, 然后结合原系统约束和反馈控制律构建鲁棒正向不变集作为参数不确定性系统的约束, 最后求解时变多面体系统的滚动时域优化问题。

(3) 针对上述想法进行仿真和实车验证, 得出本文提出的方法能有效跟踪各种道路曲率的路径, 而且跟踪的横向偏差基本稳定在 0.5m 以内。

### 本文创新点

(1) 本文提出一种考虑道路特性的模型预测控制算法。本文首先对参考路径进行拟合, 以后轴中心到参考路径的距离、车辆位姿、车辆与参考轨迹的航向偏差作为状态量, 以前轮偏角作为控制量建立预测模型, 并采用自适应采样周期对模型离散化, 在此基础上建立横向偏差和航向偏差限制的目标函数并添加状态约束和控制量约束, 进而用二次规划算法求解带约束的目标函数。

(2) 针对实际过程中的不定常时延, 本文采用多面体系统描述系统的不确定性, 最终转化为 min-max 的鲁棒控制问题。

### 未来工作展望

(1) 本文提出的方法仅考虑了执行时延这一种干扰控制器的影响因素, 但在实际过程中, 网络传输也会带来时延定位误差会间接给跟踪造成影响;

(2) 使用三次曲线拟合参考路径获取道路参数，有时后存在较大误差，因此更有效的曲线拟合方法是未来研究点；

(3) 在速度较低的情况下，本文提出的算法可以很好地适应，但在高速（速度大于 80km/h）情况下，本文提出的算法在跟踪过程中将会失稳，因为在高速下，汽车的动力学特性将起主导因素，因此基于动力学模型预测控制的研究是未来的另一个研究方向。

## 参考文献

- [1] Who W, Mathers C, Fat D M, et al. The global burden of disease: 2004 update [J]. Irish Medical Journal, 2008, 106(1):4.
- [2] Dagan E, Mano O, Stein G P, et al. Forward collision warning with a single camera[C].IEEE Intelligent Vehicles Symposium, Parma ,2004: 37-42
- [3] Kurtz J. Demo III Experimental Unmanned Vehicles Autonomous Mobility System Overview[C]. IEEE ISIC/IRA/ISAS Joint Conf, Gaithersburg, 1998:640-643.
- [4] E. Dickmanns. Vision for Grand Vehicles: History and prospects [J].International Journal of Vehicle Autonomous System, 2002, 1(1):1-44.
- [5] Dickmanns E D, Behringer R, Dickmanns D, et al. The seeing passenger car 'VaMoRs-P'[C]. Intelligent Vehicles Symposium, Paris, 1994:68-73.
- [6] Grnter H, Sheng H. The 2005 DARPA Grand Challenge – The Great Robot Race [M]. Berlin :Springer Publishing Company, 2007.
- [7] Buehler M, Iagnemma K, Singh S. The DARPA Urban Challenge: Autonomous Vehicles in City Traffic [M]. Berlin: Springer Publishing Company, 2009.
- [8] Urmson C, Anhalt J, Bagnell D, et al. Autonomous driving in urban environments, Boss and the Urban Challenge [J]. Journal of Field Robotics, 2008, 25(8):425–466.
- [9] Montemerlo M, Becker J, Bhat S, et al. Junior. The Stanford entry in the Urban Challenge [J]. Journal of Field Robotics, 2008, 25(9):569–597.
- [10] Poczter S L, Jankovic L M. The Google Car: driving toward a better future [J]. Journal of Business Case Studies, 2014, 10(1): 7-12.
- [11] Schoettle B, Sivak M. A survey of public opinion about autonomous and self-driving vehicles in the US, the UK, and Australia [J]. Transportation Research Institute, 2014, 25(3):18-25.
- [12] Palin R, Johnston V, Johnson S, et al. The Aerodynamic Development of the Tesla Model S - Part 1: Overview [J]. Journal of Shijiazhuang Vocational Technology Institute, 2012, 105(B1):795–820.
- [13] Ziegler J, Bender P, Schreiber M, et al. Making Bertha Drive—An Autonomous Journey on a Historic Route[J].Intelligent Transportation Systems Magazine IEEE, 2014, 6(2):8-20.

- [14] Li W, Jiang X, Wang Y. Road recognition for navigation of an autonomous vehicle by fuzzy reasoning[C]. IEEE International Conference on Fuzzy Systems, New Orleans, 1996: 246-250.
- [15]陈慧岩, 张玉. 军用地面无人机动平台技术发展综述[J]. 兵工学报, 2014, 35(10):1696-1706.
- [16]孙振平. 自动驾驶汽车智能控制系统[D]. 长沙:国防科学技术大学,2004.
- [17] Gong jian-wei, Jin xiao-long. An Autonomous Driving Vehicle for Urban Environment [J]. Communications of CAA, 2011, 163(32):43-52.
- [18]张鹏飞. 智能汽车视觉导航技术的研究[D]. 北京:清华大学,2004.
- [19]王晨. 无人驾驶智能车控制与规划系统的设计与实现[D]. 上海:上海交通大学, 2009.
- [20] Netto M, Blosseville J M, Lusetti B, et al. A New Robust Control System with Optimized Use of the Lane Detection Data for Vehicle Full Lateral Control under Strong Curvatures[C]. IEEE Intelligent Transportation Systems Conference , Toronto, 2006:1382-1387.
- [21] Pe, Rez J, Milane&, et al. Cascade Architecture for Lateral Control in Autonomous Vehicles [J]. IEEE Transactions on Intelligent Transportation Systems, 2011, 12(1):73-82.
- [22] Tai M. Experimental Study of Lateral Control of Heavy Vehicles for Automated Highway Systems (AHS) [C]. Proceedings of the American Control Conference. American Automatic Control Council, Anchorage, 2002:851 - 856.
- [23]陈虹. 模型预测控制[M]. 北京:科学出版社, 2013.
- [24] Song J. Dynamic Matrix Control-A Novel Computer Control Algorithm [J]. Control & Instruments in Chemical Industry, 1984:57-64.
- [25] Wang Y Y. Robust model predictive control [D]. Madison: University of Wisconsin Madison, 2002.
- [26] Grune L, Pannek J. Nonlinear Model Predictive Control [M]. London: Springer, 2011.
- [27] Borrelli F, Bemporad A, Morari M. Predictive Control for Linear and Hybrid Systems [M]. London: Cambridge University Press, 2013.
- [28] Wahab N A, Katebi R, Balderud J, et al. Data-driven Adaptive Model-based Predictive Control with Application in Wastewater Systems [J]. IET Control Theory & Applications, 2011, 5(6): 803-812.
- [29] McKinley T L, Alleyne A G. Adaptive Model Predictive Control of an SCR Catalytic Converter System for Automotive Applications [J]. IEEE Transactions on Control Systems Technology, 2012, 20(6):1533-1547.
- [30] Akpan V A, Hassapis G D. Nonlinear Model Identification and Adaptive Model Predictive Control

Using Neural Networks [J]. ISA transactions, 2011, 50(2): 177-194.

[31] Camacho E F, Bordons C. Distributed Model Predictive Control [J]. Optimal Control Applications and Methods, 2015, 36(3): 269-271.

[32] Giovanini L. Game Approach to Distributed Model Predictive Control [J]. IET Control Theory & Applications, 2011, 5(15): 1729-1739.

[33] Li H P, Shi Y. Distributed Receding Horizon Control of Large-scale Nonlinear Systems: Handling Communication Delays and Disturbances [J]. Automatica, 2014, 50(4): 1264-1271.

[34] Wang C, Ong C J. Distributed Model Predictive Control of Dynamically Decoupled Systems with Coupled Cost [J]. Automatica, 2010, 46(12): 2053-2058.

[35] Ding B C, Xie L H, Cai W J. Distributed Model Predictive Control for Constrained Linear Systems [J]. International Journal of Robust and Nonlinear Control, 2010, 20(11): 1285-1298.

[36] Ma J R, Qin J, Salsbury T, et al. Demand Reduction in Building Energy Systems Based on Economic Model Predictive Control [J]. Chemical Engineering Science, 2012, 67(1): 92-100.

[37] Callier F M, Desoer C A. Linear System Theory [M]. New York: Springer, 2012.

[38] Bemporad A, Morari M. Robust Model Predictive Control: A Survey. In Robustness in Identification and Control [M]. London: Springer, 1999: 207-226.

[39] Borrelli F. MPC-based approach to active steering for autonomous vehicle systems [J]. Int. J. of Vehicle Autonomous Systems, 2005:265-291.

[40] Kühne F, Fetter W, João L, et al. Model Predictive Control of a Mobile Robot Using Linearization [J]. Proceedings of Mechatronics & Robotics, 2004:273-280.

[41] Xie F. Model Predictive Control of Nonholonomic Mobile Robots [D]. Stillwater: Oklahoma State University, 2007.

[42] 席裕庚, 李德伟, 林姝. 模型预测控制现状与挑战 [J]. 自动化学报, 2013, 39(3): 222-236.

[43] R. Madhumitha et al. Optimum Coordination of Overcurrent Relays Using Dual Simplex and Genetic Algorithms [C]. IEEE Computational Intelligence and Communication Networks (CICN) 2015 International Conference, Jabalpur, 2015:1544-1547.

[44] Vinko Lešić, Mario Vašak, Anita Martinčević, Hrvoje Novak. Nonlinear hierarchical building zone and microgrid control based on sensitivity analysis [C]. IEEE Process Control (PC) 2017 21st International Conference, Strbske Pleso, 2017:321-326.

- [45] Keshava Nirmal, John F Mustard. Spectral unmixing [J]. Signal Processing Magazine IEEE, 2002, 19(1): 44-57.
- [46]陆辉. 轻型车弱附着地面转向制动/驱动稳定性模型预测控制[D]. 吉林:吉林大学, 2012.
- [47]F. K"uhne, W. Lages, J. da Silva Jr. Model Predictive Control of a Mobile Robot Using Linearization[C]. Proceedings of Mechanics and Robotics, Heidelberg, 2004:537-542.
- [48]B. C. Chen, C. T. Tsai and K. Lee .Path-Following Steering Controller of Automated Lane Change System with Adaptive Preview Time[C]. IEEE International Conference on Systems, Man, and Cybernetics, Kowloon, 2015: 2522-2526.
- [49]Y. Su, K. K. Tan and T. H. Lee. Computation delay compensation for real time implementation of robust model predictive control[C]. IEEE 10th International Conference on Industrial Informatics, Beijing ,2012: 242-247.
- [50] 程鹏飞,文汉江,成英燕,王华.2000 国家大地坐标系椭球参数与 GRS 80 和 WGS 84 的比较[J]. 测绘学报,2009,38(03):189-194.

## 附录

$\varphi$ ：车体的横摆角（航向角）

$\delta$ ：轮胎偏转角

$\delta_f$ ：前轮偏转角

$\delta_r$ ：后轮偏转角

$\Delta\delta_f$ ：每个控制周期的前轮偏角增量

$x$ ：状态变量

$(X_r, Y_r)$ ：后轴中心

$Cte$ ：与参考路径横向偏差

$Epsi$ ：与参考路径航向偏差

$u$ ：控制变量

$y$ ：系统输出

$h_k$ ：采样周期

$N$ ：时域（包含预测时域和控制时域）

$N_p$ ：预测时域

$N_c$ ：控制时域

$u^*(t)$ ： $t$ 时刻最优控制量

$U^*(t)$ ： $t$ 时刻最优控制序列

$Y(t)$ ： $t$ 时刻预测时域内系统输出序列

$Y_{\min}$ 、 $Y_{\max}$ ：系统输出序列约束

$\xi(t+k|t)$ ：在 $t$ 时刻对 $t+k$ 时刻的预测状态量

$x_r$ ：参考系统的状态变量

$u_r$ ：参考系统的控制变量

$\Delta u$ ：控制增量

$u_{\min}$ 、 $u_{\max}$ ：控制量约束

$u_{\text{kin},\min}$ 、 $u_{\text{kin},\max}$ ：运动学模型中控制量约束

$u_{\text{dyn},\min}$ 、 $u_{\text{dyn},\max}$ ：动力学模型中控制量约束

$\Delta u_{\min}$ 、 $\Delta u_{\max}$ ：控制增量极约束



$\Delta \mathbf{u}_{\text{kin},\min}$ 、 $\Delta \mathbf{u}_{\text{kin},\max}$ ：运动学模型中控制增量约束

$\Delta \mathbf{u}_{\text{dyn},\min}$ 、 $\Delta \mathbf{u}_{\text{dyn},\max}$ ：动力学模型中控制增量约束

$n_s$ ：状态变量维数

$m_c$ ：控制变量维数

$\chi$ ：状态变量约束

$\Gamma$ ：控制变量约束

$\eta_{\text{ref}}$ ：系统参考输出

$\eta_{\text{ref},\text{local}}$ ：局部参考输出

$\mathbf{y}_{\text{hc}}$ ：硬约束输出

$\mathbf{y}_{\text{sc}}$ ：软约束输出

$\mathbf{y}_{\text{hc},\min}$ 和 $\mathbf{y}_{\text{hc},\max}$ ：硬约束输出极限值

$\mathbf{y}_{\text{sc},\min}$ 和 $\mathbf{y}_{\text{sc},\max}$ ：软约束输出极限值

$\mathbf{A}_t$ 、 $\mathbf{B}_t$ ： $t$ 时刻的转移矩阵

$\mathbf{A}_{k,t}$ 、 $\mathbf{B}_{k,t}$ ： $t$ 时刻对 $k$ 时刻预测的转移矩阵

$G(k)$ 、 $H(k)$ ：线性运动学模型转移矩阵的离散形式

$\varepsilon$ ：松弛因子

$\rho$ ：松弛因子的权重系数

## 攻读学位期间发表论文与研究成果清单

### 攻读学位期间发表论文

- [1] Xie J, Gong J, Wu S, et al. A personalized curve driving model for intelligent vehicle[C].IEEE International Conference on Unmanned Systems, Beijing, 2018:301-306.

### 项目成果清单

- [1]. 2017.10-2017.11 深圳无人驾驶大巴横向控制调试；
- [2]. 2016.9-2016.10 某企业项目合作，底层平台代码书写，调试；
- [3]. 2016.03~2016.09 北理慧动 1 号猛士无人驾驶平台参赛：制动系统机械结构、管路系统及制动压强控制算法改进；底层平台恢复性重建，参数标定，机构调试，ECU 程序编写；配合规划感知系统调试等；并于 2016 年 9 月参加跨越险阻 2016 地面无系统挑战赛；
- [4]. 2015.12~2016.03 某企业合作项目，基于比亚迪速锐的无人驾驶平台搭建及规划控制系统测试，其中，搭建整车无人驾驶平台，并进行规划算法与控制算法联调测试；并于陕西杨凌出差，进行供电系统的设计协商及传感器、计算平台、通信设备等布置协商；
- [5]. 2015.11~2015.12 参与天津某学院“红睿”无人驾驶平台改造及交接工作，负责底层平台 ECU 升级测试及无人驾驶平台相关技术文档汇总整理；
- [6]. 2015.07~2015.08 参与“速锐”无人驾驶平台纵向速度规划控制算法改进，参与三环无人驾驶系统联调测试；

## 致谢

本论文的工作是在导师龚建伟教授的悉心指导下完成的。

在长达两年半的工作学习中，龚老师在学习、科研和为人处世方面给了我极大的启发，使我在各方面有了较为显著的提高。龚老师平易近人的优点自从考研复试联系时就给我留下了深刻的印象，日常的科研工作中，龚老师亲力亲为的习惯更是深深地影响了我。在毕业设计过程中，龚老师及时了解我课题进展与实验进程，提纲挈领，给予我正确的规划与引导。在此，对龚老师在学习、科研、生活上的帮助表示由衷的感谢。

感谢陈慧岩老师严格要求每一位学生，使得实验室的学术氛围浓厚，陈老师在跨越险阻无人地面系统挑战赛期间所表现出克服一切困难的精神使我受益良多。感谢熊老师在学术方面给予我们积极的指导，组织大家查阅文献，并完成项目申报稿的书写，既锻炼了我们写作能力，又让我们开阔了视野。

在课题研究与科研工作中，感谢博士师兄朱敏从我进入研究生阶段开始，就悉心指导我了解智能驾驶的方方面面，耐心地指导我如何看 can 协议，如何使用毫米波雷达等等，感谢何潇师兄、朱敏师兄在纵向速度规划控制方面的指导。感谢杨磊和王威让我学习到了如何分析问题和解决问题的能力，感谢丁峰同学在生活和学习上的关心，感谢张玉师兄和刘凯师兄在学术和技术上给予的指导。感谢李昊、梅维杰、程文、韩雨、周陆杰、刘一荻、宋佳妮等同学两年的相伴，与你们三年的共同学习进步十分快乐。感谢周文凯，丁勇强师弟冒着酷暑一起攻克北理慧动一号车的底层问题，与你们一起参加比赛的过程收获很大。感谢在实验和比赛过程中给予我帮助的翟涌老师、刘晓刚工程师、于林工程师、董佳杰工程师和潘竹梅工程师，谢谢你们的大力支持和耐心指导。

最特别的感谢留给我的父母和亲人，感谢你们多年毫无保留的支持与无怨无悔的付出。

最后，衷心感谢在百忙之中抽空评阅本文并提出宝贵意见的各位专家学者！

谢兼明

二〇一八年六月于北京理工大



