

硕士学位论文

(专业学位论文)

考虑动力学约束的

自动驾驶汽车运动规划研究

RESEARCH ON MOTION PLANNING OF
AUTONOMOUS DRIVING WITH DYNAMICS
CONSTRAINTS

由煜法

哈尔滨工业大学

2023 年 5 月

国内图书分类号：U469.72

国际图书分类号：629

学校代码：10213

密级：公开

硕士学位论文

考虑动力学约束的 自动驾驶汽车运动规划研究

硕士研究生：由煜法

导师：赵林辉 副教授

申请学位：专业学位

学科：电子信息

所在单位：航天学院

答辩日期：2023年5月

授予学位单位：哈尔滨工业大学

Classified Index: U469.72

U.D.C: 629

Dissertation for the Master's Degree

**RESEARCH ON MOTION PLANNING OF
AUTONOMOUS DRIVING WITH DYNAMICS
CONSTRAINTS**

Candidate :	You Yufa
Supervisor :	Assoc. Prof. Zhao Linhui
Professional Degree Applied for :	Master of Engineering
Speciality :	Electronic Information
Affiliation :	School of Astronautics
Date of Defence :	May, 2023
Degree-Conferring-Institution :	Harbin Institute of Technology

摘 要

运动规划是自主移动机器人实现安全行驶的关键因素。近十年来,随着自动驾驶技术的不断发展,专门针对于自动驾驶的运动规划技术引起越来越多研究者的关注。目前,自动驾驶的应用场景还非常有限,因为当前的运动规划并没有考虑车辆的动力学特性,使车辆在许多场景中存在滑动、旋转甚至侧倾的风险。在规划中引入动力学约束以克服上述问题,一直是研究的重点与难点。因此本课题针对考虑动力学约束的自动驾驶运动规划,给出相应的策略。

车辆动力学模型研究方面,论文首先建立并分析了 14 自由度车辆动力学模型,通过 veDYNA 仿真验证模型的准确性。为了使模型更适用于指导运动规划,本论文通过 Milliken 力矩法将模型转换为 Milliken 动力学模型。同时,定义滑动指数、旋转指数、侧倾指数,通过深度优先搜索在 Milliken 动力学模型中获得动力学安全边界,并定义安全距离。

针对有车道线信息的道路空间,本文首先建立了参考线模型,通过共线点删除和三次样条插值,获得平滑的规划参考线。其次,本文建立了风险势场模型,来描述道路空间中形状于运动较为规则的障碍物;最后,本文设计了基于 Frenet 坐标系的规划算法,在代价函数中引入了安全距离,从而提高规划轨迹的安全性。

针对无参考线、含多类障碍物的自由空间,本文设计了改进 Voronoi 势场,以描述不规则的地形和障碍物。为了降低运动规划对参考线的依赖,本文设计了基于启发式搜索的运动规划算法,通过加入动力学约束的启发式搜索实现避障,同时优化了规划轨迹的动力学特性。

最后,进行了系统架构设计,包含硬件平台、软件架构和输入输出接口,给出了算法的开发和实现方案。设计了硬件在环实验流程,并通过 Jetson Orin 平台、C++代码开发和硬件在环仿真,验证了算法的有效性。

关键词:自动驾驶; 运动规划; Milliken 动力学

Abstract

Motion planning is a critical factor for safe autonomous driving of unmanned vehicles, and motion planning techniques specifically designed for autonomous driving have received increasing attention in the past decade. However, the current motion planning does not consider the dynamic characteristics of the vehicle, which limits the application scenarios of autonomous driving due to safety concerns. Therefore, this study focuses on autonomous driving motion planning with dynamic constraints and proposes corresponding strategies.

To achieve this goal, the thesis first establishes and analyzes a 14-degree-of-freedom vehicle dynamic model and converts it into a Milliken dynamic model using the Milliken moment method. The thesis then defines the roll index, yaw index, and slip index and obtains a safe domain enveloping line in the Milliken dynamic model through depth-first search.

For the road space with lane information, the thesis establishes a reference line model, obtains a smooth planning reference line, establishes a risk field model to describe obstacles in the road space, and designs a planning algorithm based on the Frenet coordinate system. The thesis also defines a safety margin related to dynamics in the cost function to improve the safety of the planning trajectory.

For the free space without reference line information, the thesis designs a Voronoi field to describe irregular terrain and obstacles and a motion planning algorithm based on heuristic search. The study defines the augmented primitive and heuristic information related to dynamics and considers dynamic constraints when selecting search actions.

Finally, a systematic architectural design was conducted, encompassing hardware platform, software architecture, and input-output interfaces, providing the development and implementation plan for the algorithm. The development process for hardware-in-the-loop experimentation was designed, and the effectiveness of the algorithm was verified through simulation and hardware-in-the-loop simulation using the Jetson Orin platform.

Keywords: Autonomous driving, Motion planning, Milliken Dynamics

目 录

摘 要	I
ABSTRACT	II
第 1 章 绪 论	1
1.1 课题背景及研究的目的和意义	1
1.2 国内外研究现状	2
1.2.1 道路空间运动规划发展现状	2
1.2.2 自由空间运动规划发展现状	4
1.3 本课题的主要研究内容	5
第 2 章 车辆动力学建模与分析	8
2.1 引言	8
2.2 车辆坐标系建模	8
2.3 车辆模型	9
2.3.1 轮胎模型	10
2.3.2 整车动力学模型	12
2.4 Milliken 动力学约束	17
2.4.1 Milliken 力矩法	18
2.4.2 Milliken 动力学安全指标	20
2.4.3 安全距离	26
2.5 本章小结	28
第 3 章 考虑动力学约束的道路空间规划方法	29
3.1 引言	29
3.2 参考线建模	29
3.2.1 共线点删除	30
3.2.2 三次自然样条插值	31
3.3 风险势场建模	35
3.3.1 风险势场函数	35
3.3.2 风险等级集	37
3.4 Frenet 最优规划器	39
3.4.1 Frenet 坐标转换	39
3.4.2 轨迹生成	40

3.4.3 代价函数设计	44
3.5 道路空间规划器仿真验证	49
3.5.1 仿真场景选择	49
3.5.2 巡航场景验证	50
3.5.3 避障场景验证	52
3.5.4 弯道场景验证	53
3.6 本章小结	54
第 4 章 考虑动力学约束的自由空间规划方法	56
4.1 引言	56
4.2 改进 Voronoi 势场建模	56
4.2.1 Voronoi 图	57
4.2.2 广义 Voronoi 图	58
4.2.3 Voronoi 势场	59
4.2.4 改进 Voronoi 势场	60
4.3 启发式最优规划器	63
4.3.1 搜索动作	63
4.3.2 启发式搜索	65
4.3.3 代价函数设计	66
4.4 自由空间规划器仿真验证	67
4.4.1 仿真场景选择	67
4.4.2 越野场景验证	68
4.4.3 阻塞场景	69
4.5 本章小结	71
第 5 章 系统架构设计与硬件在环验证	72
5.1 引言	72
5.2 系统架构设计	72
5.2.1 系统的硬件平台	72
5.2.2 系统软件架构	73
5.2.3 系统输入输出接口	75
5.3 算法的嵌入式实现	77
5.3.1 基本数学类	77
5.3.2 规划器类	79
5.3.3 开发工具类	79

5.4 硬件在环实验验证	80
5.4.1 方案设计	80
5.4.2 实验验证及结果分析	80
5.5 本章小结	84
结 论	85
参考文献	86
攻读硕士学位期间取得创新性成果	90
哈尔滨工业大学学位论文原创性声明和使用权限	91
致 谢	92

第 1 章 绪 论

1.1 课题背景及研究的目的和意义

运动规划涉及人工智能、机器人学、控制等多个领域的技术，对于任何自主移动机器人都是必不可少的。运动规划的任务是求解由当前状态到目标状态的变换过程，或者求解为实现这个变换需要完成的动作。运动规划目前在各种地面移动机器人系统中的应用十分广泛，特别是自动驾驶系统。自动驾驶系统的运动规划模块接收感知、预测、地图等模块的输入信息，然后为控制模块输出参考轨迹。

近几十年来，汽车产业的发展和汽车保有量增加带来了一些社会问题，比如越来越多交通事故。除此之外，汽车造成的噪音污染、环境污染和交通拥堵等问题也愈发严重。汽车自动驾驶和辅助驾驶技术在减少事故、缓解拥堵、提高效率以及保证行驶安全性等领域带来发展，在未来该技术将会是新一轮科技革命中的一项关键性技术，全球各个国家正在积极的推进相关的研究与开发工作。运动规划技术的发展也得到了相应推动，并且专门针对于汽车的运动规划技术也得到了越来越多研究者的关注。

在 20 世纪 80 年代，就已经出现了自动驾驶汽车^[1]，此时的自动驾驶任务在于简单控制指令的生成，而运动规划技术还没有得到充分发展。在 21 世纪的第一个十年，城镇自动驾驶测试的成功使自动驾驶汽车的发展进程从可行性证明转向商业性应用^[1]，在这个过程中运动规划算法保证了车辆能够在给定环境下完成自主行驶。此时经典的运动规划框架初步确立，但其性能仍需进一步发展。随后许多自动驾驶公司相继成立，众多汽车公司、互联网公司也相继参与到无人驾驶的商用化进程中。这是自动驾驶技术从实验走向落地的过程，在这个过程中感知、决策规划、控制的技术都日趋完善，形成了一些较为完整的框架、算法和系统。在当前的发展阶段，针对自动驾驶汽车的运动规划算法基本确定根据场景分为两类：道路空间运动规划和自由空间运动规划。

道路空间是具有车道线等交通标记的结构化道路环境，在这个空间下车辆需要尽量沿车道中心线行驶，能保证遵守交通规则，并且能让其他交通参与者。在城市中，道路空间形成了巨大的交通网络，但自动驾驶目前只能应用交通网络的少数路段。这是因为目前的自动驾驶汽车规划技术并不考虑自身的动力学特性，因此在陌生环境下自动驾驶汽车无法确保自身远离动力学的不安全状态。比如一小块沙石、结冰、雨水路面，随时可能使车辆进入滑动、旋转等

危险状态，但不考虑动力学的运动规划却很难避免这种风险。

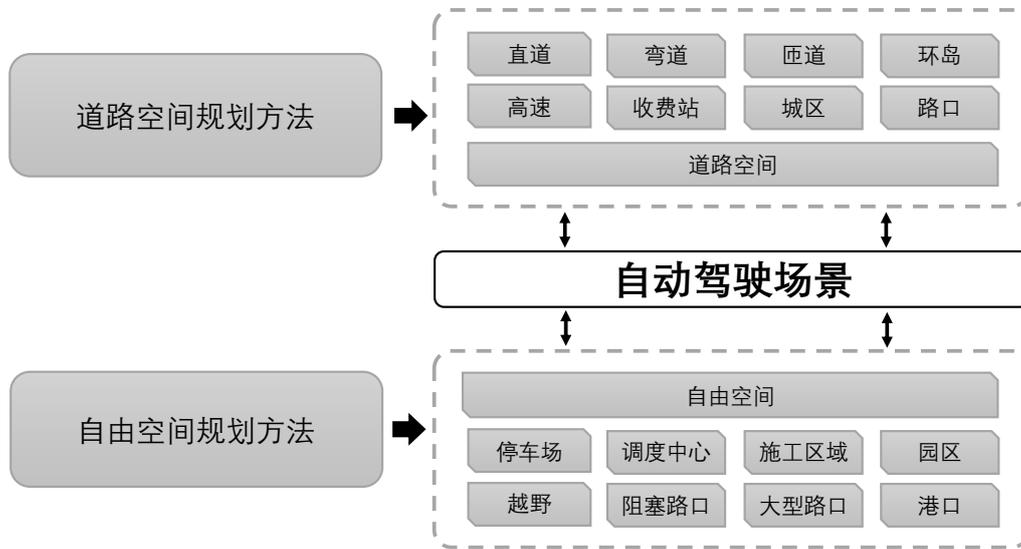


图1-1 自动驾驶场景分类

同时，在行驶过程中车辆也随时可能丢失车道线信息，进入自由空间。丢失车道线的原因可能是传感器遮挡、驶入开放地形等。在自由空间中，障碍物的形状往往是更加不规则的，而且没有车道线引导车辆的行驶方向。在这种环境下，缺少动力学分析同样会为自动驾驶带来危险，因此也需要在自由空间中探讨上述问题的解决方案。

综上所述进一步研究考虑动力学约束的自动驾驶运动规划方法十分具有意义。一方面可以提高自动驾驶汽车对日常行驶场景中极限工况的应对能力，优化轨迹的安全性，避免车辆行驶时进入动力学危险状态。另一方面可以扩大自动驾驶服务的应用范围，使自动驾驶算法应对更多的应用场景，满足更多的任务要求。

本课题得到国家自然科学基金面上项目（61973097）“多源信息下电动汽车底盘系统分布式估计与协同控制研究”的支持。

1.2 国内外研究现状

1.2.1 道路空间运动规划发展现状

在自动驾驶出现以前，很少有地面自主移动机器人是在道路空间中工作的，因此道路空间运动规划研究起步较晚。但随着近年来智能网联汽车和自动驾驶技术的兴起，该项技术得到了迅速发展。道路空间的规划问题需要充分考虑乘车人员的舒适性，以及乘车人员和其他交通参与者的人身安全，并且要遵守复

杂的交通规则。

2007 年的 DARPA Urban Challenge^[2]是较为早期的道路空间自动驾驶运动规划的探索。在比赛中，斯坦福大学和卡内基梅隆大学的车辆成为了优胜者。斯坦福大学的车队使用了样条优化的方法^[3]，使运动规划技术能够同时满足车辆运动学和道路边界的约束。而卡内基梅隆大学的车辆使用模型预测的方法^{[4][5]}，解决了运动规划的多约束多目标的优化问题。

在早期阶段中，道路空间运动规划的任务要求为不超出道路边界即可，随后的研究开始探索如何使轨迹更好的回归到车道中心线，并且提高轨迹的性能。2010 年，Werling 等人将 Frenet 坐标系应用于自动驾驶运动规划^[6]，能够实现车辆结构化道路中侧向规划和速度规划的解耦，降低了道路空间运动规划的问题维度。2016 年，Schlechtriemen 等人提出了将运动学约束转换到 Frenet 坐标系下的方法^[7]，能够充分考虑参考线曲率对车辆轨迹的影响，以应对各种几何形状的道路空间，比如高速匝道和环形交叉路口。同年，Wang 等人将模型预测控制用于高速场景下环道轨迹生成^[8]，能够基于侧向动力学模型生成避免与简单障碍物碰撞的轨迹，但将应用场景限制在允许将轮胎力进行线性化近似的场景。2018 年 Fan 等人构建了较为完整的决策规划框架^[9]，其本身是一个规划器，但能通过轻决策实现道路选择，通过动态规划和优化结合的方式生成最优轨迹，这个框架通过 Apollo 自动驾驶平台实现了代码开源和验证。同年，Fan 等人提出了数据驱动的自动调整框架，该框架包括一种新的基于秩的条件逆强化学习算法、一种离线训练策略和一种自动收集和标记数据的方法^[10]。李柏在博士论文中设计了复杂约束下自动驾驶车辆运动规划的计算最优控制方法^[11]，采用全联立正交配置有限元方法将最优控制命题离散化为非线性规划问题。2020 年，Zhang 等人研究了 Frenet 坐标系下的约束表示，以及车辆轨迹生成的二次规划建立与求解^[12]。

近年来也有一些研究者提出了在道路空间中考虑动力学模型的运动规划算法。2018 年，Guo 等人提出了同时考虑自车和目标动力车动力学模型的模型预测运动规划算法^[13]。2019 年，Huang 等人提出了侧向与纵向解耦的基于模型预测的运动规划算法^[14]，先通过模型预测控制跟踪参考速度，再进行考虑避障的侧向路径生成，在侧向规划中使用了二阶线性模型；2020 年，Zhou 等人提出了通过轮胎力模型分析轮胎力线性区和饱和区的边界，来指导运动规划^[15]。2021 年，Yang 等开发了学习人类驾驶行为的换道轨迹规划方法^[16]，通过模拟人类驾驶风格来使规划更符合车辆动力学约束。Liu 等人研究了换道轨迹规划中的动力学模型预测方法^[17]，使模型能适用于不同速度、终端状态约束的换道任务，在规划过程中保证轮胎力处于线性区间。Lim 等人在 Frenet 坐标系

下设计了一种基于模型预测的轨迹规划方法^[18]，能够根据不同的任务和性能要求选择车辆模型。2022年，张鹏等基于 Carsim 动力学模型进行的车辆路径规划的仿真测试^[19]。李荣黎提出了通过离线计算车辆动力学的映射关系^[20]，提高效率的自动驾驶赛车路径与车速协同规划方法。殷国栋提出了基于四轮稳定域的路径规划方法^[20]，通过 Dugoff 轮胎模型和最大质心侧偏角建立稳定域，并实现基于多项式的路径规划。

近年来人们在一直在研究在自动驾驶运动规划中加入动力学约束的方法。有一些方法是使用模型预测的方法提出的，龚建伟老师在他的著作中也总结了模型预测方法在自动驾驶中的应用^[22]。该方法存在的问题主要在于，目标函数的选择需要符合驾驶的安全性、舒适性等要求，但在目标函数较为复杂时优化问题的求解为题也会出现较高的难度和耗时。在最近的研究中，一些基于神经网络或者动力学稳定域的方法被人们提出，试图通过预训练和预计算的方法，来降低运动规划给硬件带来的计算负担，从而降低自动驾驶算法的硬件成本。但目前这些研究尚处于初步探索阶段，只能应用于少量的自动驾驶场景。

1.2.2 自由空间运动规划发展现状

自由空间运动规划算法相对于道路空间运动规划算法来说起步较早，因为在自动驾驶汽车出现在道路空间中之前，封闭自由空间的自动驾驶测试被人们更早开展^[23]。在这段时期，人们提出了一些运动规划方法能使车辆穿越荒漠、山谷等存在不规则障碍物的自由空间。2004年，Urmson 通过激光雷达有限的感知精度对地形进行评估，根据障碍分布设计了运动规划算法^[24]。该算法能够初步实现汽车的自动驾驶，但最终没能避免与岩石发生碰撞。2005年，Gutierrez 提出了通过螺旋线和直线拼接的运动规划方法^[25]，在结合预设的安全走廊完成了穿越沙漠的任务。2006年，Thrun^[26]、Trepagnier^[27]、Urmson^[28]等人通过启发式搜索和随机搜索算法，完成了自由空间中自动驾驶挑战的任务，通过实验证明了搜索类的算法可以初步解决自由空间中的自动驾驶运动规划问题。

在运动规划发展早期，通过搜索产生的轨迹具有较差的平滑性和舒适性，并且不能很好的反应车辆的运动学和动力学特性，因此后续的研究在这些方面做出了许多改进。在启发式搜索方面，Persson 在搜索动作中加入运动学约束，并且通过模拟退火表^[29]平衡探索的平滑性和效率。张浩在启发函数中加入关于父辈节点的约束^[30]，使其更好的向目标方向收敛且生成的路径可以更加平滑化。Cai 等人提出了规划中的安全走廊在线生成方法^[31]，并且通过螺旋线解决规划中的曲率不连续问题。冯欣阳提出了双层的搜索路径平滑策略^[32]，先通过 Dubins 曲线拟合规划路径，在接近障碍物时使用直线、圆弧线、螺旋线结

合的方法生成较为精准的规划路径；

随机搜索算法在维度较高的空间中具有较快的计算效率，但随机搜索产生的轨迹并不能保证最最优性，因此有许多研究致力于提升随机搜索算法的最优性。RRT*算法在扩展节点时尝试重新选择父节点^[33]，在搜索过程中以提高最优性为目标不断重建搜索树。Karaman 提出的 AnytimeRRT 算法在获得搜索结果后持续随机搜索^[34]，提高路径的最优性，同时能更好的应对动态环境。Gammell 等人提出的 Informed-RRT 约束了随机采样的状态范围^[35]，随机扩展的状态只能处于变化的区域内。也有一些算法致力于提高随机搜索算法的平滑性，并且使规划轨迹满足运动学和动力学约束。Perez 提出的 LQR-RRT 运动规划方法能使用线性二阶调节器^[36]，为状态的随机扩展提供启发式信息。王圣懿等人提出了基于先验概率分布改进的 RRT 车辆运动规划算法^[37]，杜卓洋提出了随机树的预处理策略^[38]，利用上一帧的搜索结果进行初始化，并设计了基于数值优化的路径平滑方法；

近年来也有一些基于优化的自由空间运动规划方法研究，以约束的形式提高轨迹的动力学性能。郑凯琳等人设计了基于改进 TEB 算法的阿克曼机器人运动规划系统^[39]，避免加速度的变化率过大使底盘电机输出的力矩突变，引起底盘受到冲击震荡；余冬冬在博士论文中提出了一种基于图优化的轨迹规划方法 TEB 的改进算法^[40]，采用轨迹跟踪的算法跟踪 TEB 优化后的轨迹替换原来的速度计算，使整个算法更加合理。Li 等人设计了基于车辆运动学模型预测的泊车规划算法^[41]，通过贝塞尔曲线生成大致轨迹，再使用模型预测控制进行归集优化。这些方法大多针对泊车、仓储等特殊自动驾驶场景，不能处理复杂的自由空间自动驾驶场景。除此之外，这些方法在考虑运动规划的车辆约束时引入车辆运动学，而不考虑车辆动力学约束，因为很难保证在线规划的计算效率。

1.3 本课题的主要研究内容

为了解决上述问题，本文主要针对考虑动力学约束的自动驾驶运动规划展开研究。首先设计了车辆模型和规划空间模型。车辆模型包括轮胎模型、整车模型和 Milliken 力矩法，能反映了车辆的动力学特性，并将动力学模型转换为 Milliken 动力学模型，使模型更适用于指导规划。规划空间模型包括道路空间和自由空间模型，能描述全部的自动驾驶场景。其次设计了考虑动力学约束的道路空间运动规划和自由空间运动规划，在不同的规划空间和规划模型中引入动力学约束，提高运动规划算法的动力学安全性。本文研究的主要内容如图 1-2 所示：

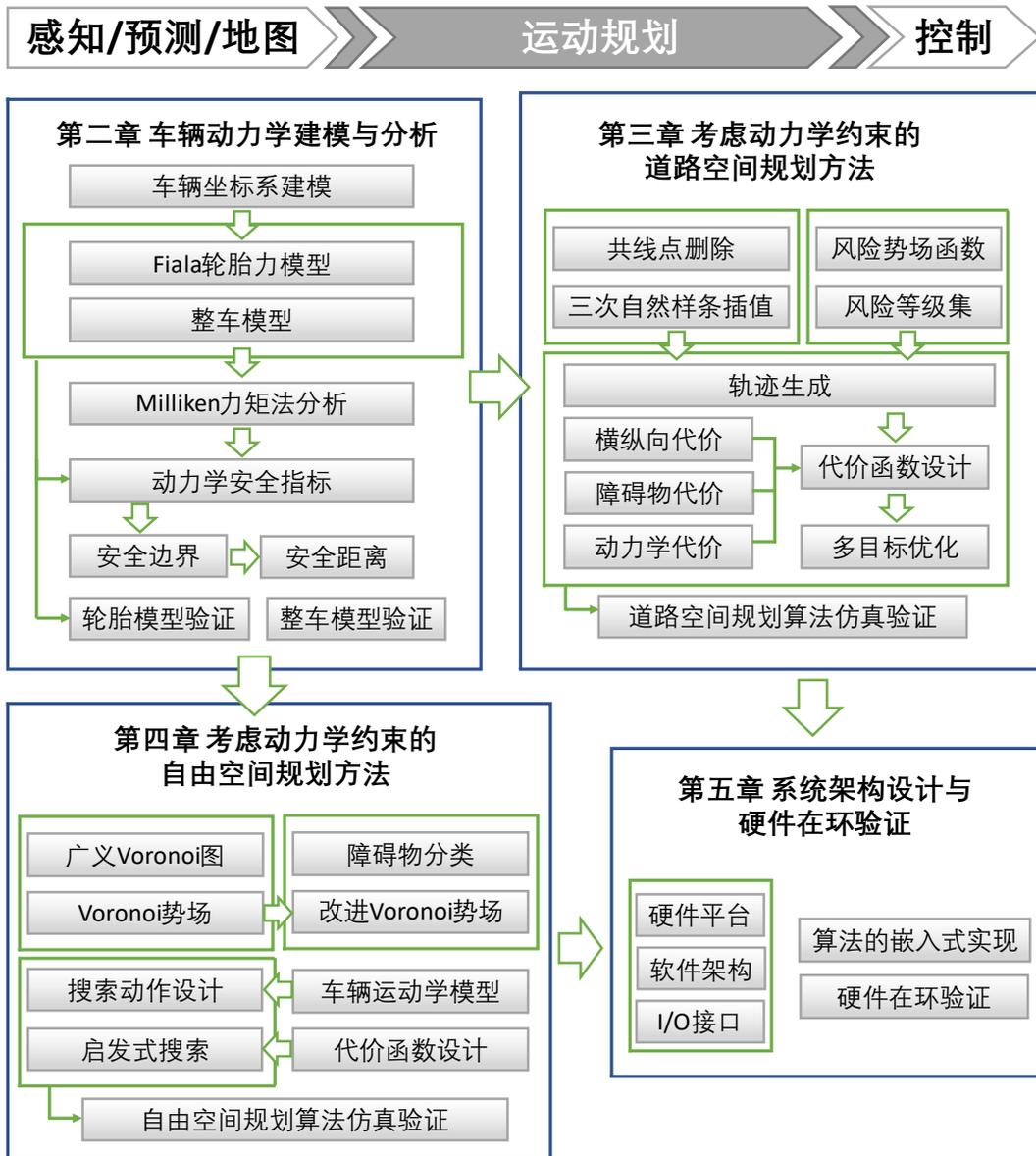


图1-2 主要研究内容

第一章介绍了本文的研究背景以及意义,分别介绍了道路空间和自由空间的轨迹规划与轨迹跟踪的研究现状,进而根据当前研究所存在的问题引出本文的主要研究内容。

第二章论文首先建立并分析了 14 自由度车辆动力学模型,包括轮胎模型和整车模型。为了使模型更适合于指导运动规划,本论文通过 Milliken 力矩法将模型转换为 Milliken 动力学模型。随后,本文通过与 veDYNA 模型比较验证模型的准确性。最后,本文定义滑动指数、旋转指数、侧倾指数,并通过深度优先搜索在 Milliken 动力学模型中获得动力学安全边界,并定义了安全距离。

第三章本文设计了考虑动力学约束的道路空间运动规划。针对有车道线信息的道路空间,本文首先建立了参考线模型,通过共线点删除和三次样条插值,获得平滑的规划参考线。其次,本文建立了风险势场模型,来描述道路空间中形状和运动较为规则的障碍物;最后,本文设计了基于 Frenet 坐标系的规划算法,在跟踪巡航速度的多种场景下规划最优轨迹。根据安全距离定义了动力学约束,从而提高规划轨迹的安全性。

第四章本文设计了考虑动力学约束的自由空间运动规划。针对障碍物不规则的自由空间,参考线模型和风险势场模型都会失效,本文设计了改进 Voronoi 势场,以描述不规则的地形和障碍物。为了降低运动规划对参考线的依赖,本文设计了基于启发式搜索的运动规划算法。本文在算法中定义了动力学相关的启发式信息,从而能够获取动力学安全的轨迹。

第五章本文设计并实现了算法的硬件在环验证。首先设计了运动规划系统软硬件架构,将运动规划系统通过 C++代码实现,并且在车规级计算平台上进行了部署。为了提高开发效率与算法效率,本文设计了系统的进程间通信接口和 CAN 总线接口。本文通过 CAN 总线实现车规级计算平台与场景仿真平台通信,通过硬件在环实验验证了算法的有效性。

第 2 章 车辆动力学建模与分析

2.1 引言

在本章中，首先建立了全局坐标系、车辆坐标系、轮胎坐标系和质心坐标系，并描述了坐标系之间的转换关系。全局坐标系用于表示车辆相对于环境的运动轨迹；车辆坐标系的原点通常被定义为车辆后轴中心，用来表示整车的状态；轮胎坐标系用于描述车轮的运动和受力。

接下来本文建立了 14 自由度车辆动力学模型。首先基于组合滑移的 Fiala 轮胎模型计算轮胎受力，考虑纵向力与侧向力的耦合。其次建立了整车的动力学模型，建立了滑移率、侧偏角，并且设计了考虑动态重量转移的垂向载荷计算方法，结合轮胎模型分析了车辆平移、横摆、俯仰、侧倾、前轮转角、车轮旋转以及重心位置变化。通过与 veDYNA 动力学模型的比较实验，验证了模型的准确性。

最后为了使模型更适合于指导运动规划，本论文通过 Milliken 力矩法将模型转换为 Milliken 动力学模型，通过引入离线预计算提高了模型的在线计算效率。同时，定义滑动指数、旋转指数、侧倾指数，在 Milliken 动力学模型中进行了安全状态筛选，并通过基于深度优先遍历的包络线搜索算法在 Milliken 动力学模型中获得动力学安全边界，并且定义了安全距离。

2.2 车辆坐标系建模

讨论车辆模型前首先定义了需要使用的坐标系：全局坐标系、车辆坐标系和轮胎坐标系。

全局坐标系通常使用垂直于地球表面向下的 Z 轴、指向地理东的 Y 轴和指向地理北的 X 轴来定义。

车辆坐标系能够相对于全局坐标系进行平移和旋转。本文中的车辆坐标系的定义如图 2-1 所示，x 轴指向前方与纵轴对齐，y 轴指向右侧，z 轴指向下方。旋转定义逆时针旋转为正，绕 x 轴旋转称为滚侧倾，绕 y 轴旋转称为俯仰，绕 z 轴旋转称为横摆。

轮胎坐标系位于由轮距和轴距定义的车辆坐标系相对位置，如图 2-2 所示。轮胎坐标系的 z 轴与车辆坐标系的 z 轴对齐，但车轮坐标系可随车轮绕 z 轴旋转。

图 2-1 为全局坐标系和车辆坐标系的关系，两个坐标系的 z 轴方向均符合

右螺旋坐标系定义。

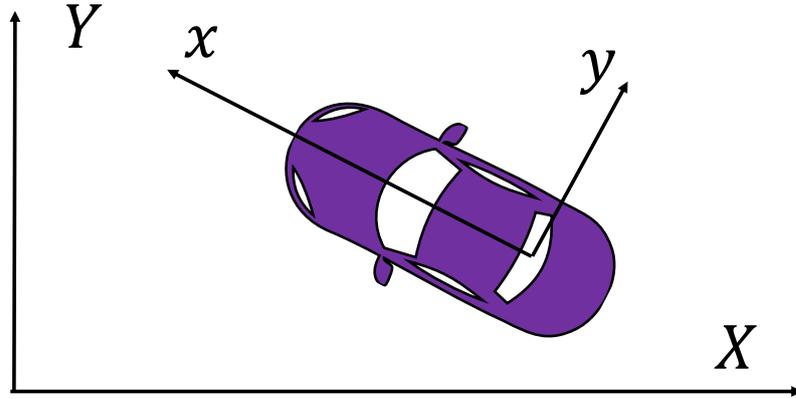


图2-1 车辆坐标系

图 2-2 为车辆坐标系和车轮坐标系的关系。

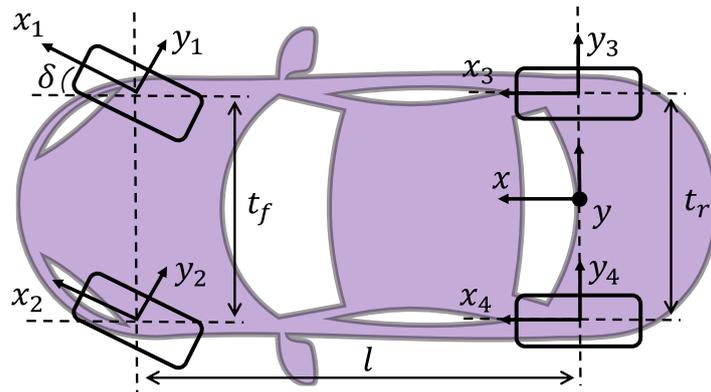


图2-2 车轮坐标系

2.3 车辆模型

在建立车辆模型前，需要做出一些关于车辆模型的基本假设。这些假设代表了标准四轮 Ackerman 转向汽车的特征。这样的假设同时能确保模型中的参数可以通过实际的车辆模型来测量，或者从汽车制造商获取。这些假设包括：

- (1) 车辆有四个车轮，在配置方面，车辆侧向对称于其 x 轴；
- (2) 车辆产生的所有力通过轮胎作用到地面；
- (3) 轮胎具有标准橡胶斜交轮胎或子午线轮胎的共同特征；
- (4) 车辆只能通过前轮转向；
- (5) 悬架和轮胎柔度的贡献远远大于车架刚度的贡献；
- (6) 车轮通过特性已知的悬架系统连接到弹簧负载；

(7) 空载质量和静态重量分布已知。

目前生产的大多数车辆在尺寸上是侧向对称的，但前轴和后轴的配置之间通常存在显著差异。这种纵向对称性的缺失会对机动特性产生很大影响，因此不能假设纵向对称性。

如果与悬架部件的刚度相比车架刚度足够高，则可以将车架建模为刚体。Sampo 等人针对车辆底盘的动力学特性研究结果表明，决定车架刚度对操控性影响的关键因素是车架刚度与悬架侧倾刚度的比值^[42]。对于车架/侧倾刚度比大于 5 的车辆，刚性车架假设是有效的^[43]。

除车辆模型假设外，本文还对车辆的运行环境做出如下假设：

- (1) 车辆将在组合滑移条件下运行；
- (2) 质量和分布都未知的载荷可动态应用于簧上质量；
- (3) 车辆将在通常局部平坦的地形上行驶；
- (4) 地形不会出现坡度的大幅度变化；
- (5) 时变地面-轮胎相互作用可通过动态摩擦系数表征；
- (6) 空气动力很小，可以忽略。

不同的运行环境假设会对车辆的建模过程产生影响。如果假设车辆在组合滑移条件下运行，那么在建立轮胎模型时需要考虑轮胎的纵向力和侧向力的耦合。在本工作中，假设地形剖面局部平坦，足以对于四个车轮都能保持地面和车轮之间的接触。时变地面-轮胎相互作用的精准建模问题是非常复杂的，需要考虑车轮下沉、土壤抗剪强度和土壤附着力等因素^[45]，超出了本文的研究范围。

该模型共有 14 个自由度： x 、 y 、 z 方向的平移， θ 角旋转，前轮转向 δ ，簧质量旋转 φ 和 γ ，车轮旋转 ω_n ，以及重心位置 x_{CG} 、 y_{CG} 、 z_{CG} 。模型自由度的选择除车辆的基本运动状态^[45]外，还考虑到所有与描述车辆的动力学安全性相关的特征，包括：可变重心位置、质量以及侧向和纵向重量传递。

2.3.1 轮胎模型

轮胎接触面处产生的力最终影响车辆的运动，因此需要选择能够准确计算这些力的轮胎模型，并能兼容车辆模型设计的总体目标。

本工作中选择的轮胎模型为 Hsu^[47]所述的组合滑移 Fiala 模型。该模型可以以刷子轮胎模型为基础推导出来，其完整推导由 Pacejka 在其著作中给出^[48]，此处不再重复。

由文献综述可知，除 Fiala 模型外，Magic 公式轮胎模型也是比较常用的轮胎模型，而且精确性高于 Fiala 模型^[49]。然而，Magic 公式模型包含了大量的经验参数，并且其中很多参数的物理意义并不明确。因此，它不适用于本文所涉

及的在线运动规划技术，除此之外，对于许多没有专用测试场景和工具的研究机构来说，通常无法获得准确的 Magic 公式轮胎模型参数值。

Fiala 模型根据垂直载荷 F_{zn} 、侧偏角 α_n 、滑移率 S_n 、侧偏刚度 C_α 、纵向刚度 C_s 和摩擦系数 μ 等便于获取的参数定义。本文 Fiala 模型的组合滑移版本，组合滑移表明模型适用于轮胎同时发生纵向和侧向滑移的情况。组合滑移分析对于分析车辆的实际动态特性来说不可或缺，因为轮胎能够产生的最大力是侧向力和纵向力的函数，这种关系通常被称为摩擦圆或摩擦椭圆。

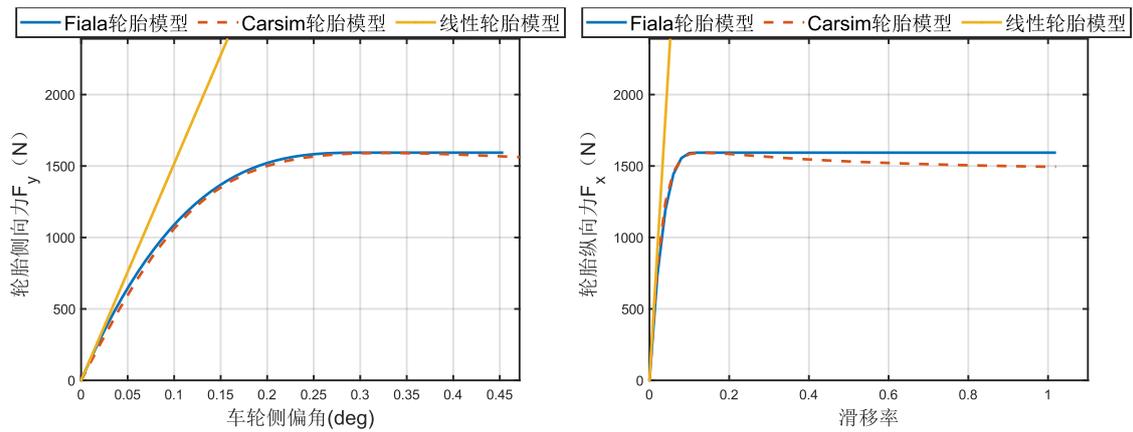
轮胎模型的两个关键输出是接触片处产生的侧向力 F_{yn} 和纵向力 F_{xn} 。纵向力方程增加了线性滚动阻力系数 C_{rr} ，以便能够模拟轮胎在滚动时由于变形而产生的阻力。

$$\begin{aligned} F_{xn} &= C_s \left(\frac{S_n}{1 + S_n} \right) \frac{F_n}{f_n} - C_{rr} F_{zn}, n = 1, 2, 3, 4 \\ F_{yn} &= C_\alpha \left(\frac{\tan \alpha_n}{1 + S_n} \right) \frac{F_n}{f_n}, n = 1, 2, 3, 4 \end{aligned} \quad (2-1)$$

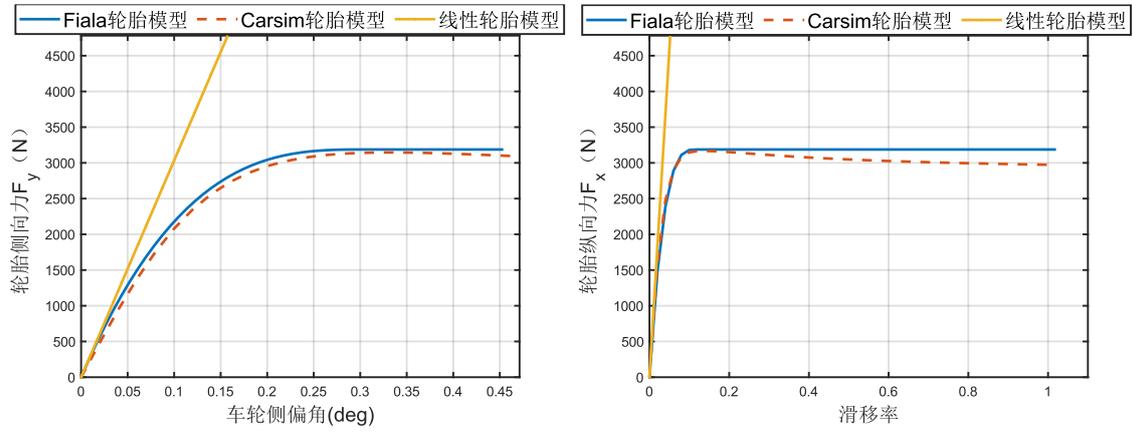
其中 F_{xn} 为第 n 个轮胎的纵向力， F_{yn} 为第 n 个轮胎的侧向力， S_n 为第 n 个轮胎的滑移率， α_n 为第 n 个轮胎的侧偏角， C_{rr} 为线性滚动阻力系数， C_s 和 C_α 分别为纵向刚度和侧偏刚度。 F_n 为第 n 个轮胎的基于刷子模型计算的轮胎力，而 f_n 为组合滑移的轮胎系数，如下计算：

$$\begin{aligned} F_n &= \begin{cases} f_n - \frac{1}{3\mu F_{zn}} f_n^2 + \frac{1}{27\mu^2 F_{zn}^2} f_n^3, f_n \leq 3\mu F_{zn}, n = 1, 2, 3, 4 \\ \mu F_{zn}, f_n > 3\mu F_{zn} \end{cases} \\ f_n &= \sqrt{C_s^2 \left(\frac{S_n}{1 + S_n} \right)^2 + C_\alpha^2 \left(\frac{\tan \alpha_n}{1 + S_n} \right)^2}, n = 1, 2, 3, 4 \end{aligned} \quad (2-2)$$

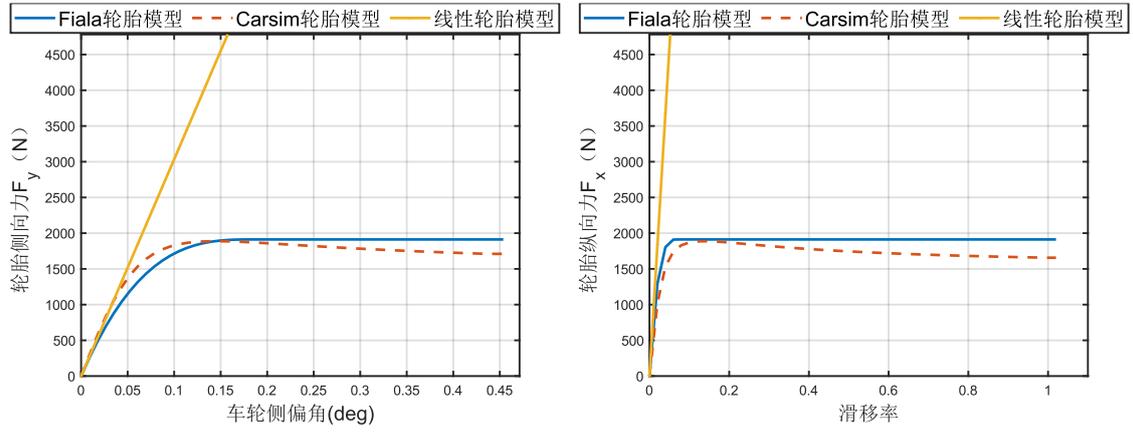
其中 μ 为路面附着系数。组合滑移轮胎系数 f_n 体现了因组合滑动条件而产生的摩擦圆，如果轮胎任意一个方向的刚度设置为零，这些方程将简化为标准单轴 Fiala 方程。下图为不同垂向载荷和路面附着系数下线性轮胎模型、veDYNA 中的 Magic 公式轮胎模型、线性轮胎模型的轮胎模型验证。在不同工况下，Fiala 轮胎模型与 Magic 公式相比并没有很高的失真，并且准确性远高于线性模型。



(a) 路面附着系数 1, 垂向载荷 1593.58N



(b) 路面附着系数 1, 垂向载荷 3187.16N



(c) 路面附着系数 0.5, 垂向载荷 3187.16N

图2-3 Fiala 轮胎模型验证

2.3.2 整车动力学模型

整车动力学模型对轮胎模型施加输入后获得轮胎力，然后通过四个轮胎的受力解算车辆的运动状态。图 2-2 为纵向运动、横向运动、横摆运动三个运动

自由度的车辆动力学模型:

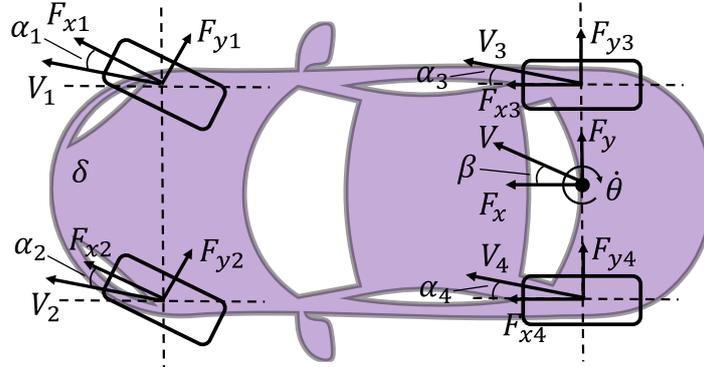


图2-4 纵侧向车辆动力学模型示意图

从轮胎模型中可以看出，轮胎力由轮胎的滑移率 S_n 、侧偏角 α_n 和垂向载荷 F_{zn} 决定。滑移率^[48]定义了轮胎实际旋转速度与自由滚动（即不产生力）轮胎速度之比，定义为：

$$S_n = \frac{r_{eff}\omega_n - V_x}{V_x}, n = 1,2,3,4 \quad (2-3)$$

其中 V_x 为车辆纵向速度， r_{eff} 为有效轮胎半径， ω_n 为车轮的旋转角速度。

车轮侧偏角^[48]可定义为轮胎方向和行驶方向之间的角度：

$$\beta = \text{atan}\left(\frac{V_x}{V_y}\right)$$

$$\alpha_1 = -\delta + \text{atan}\left(\frac{V_x\beta + (l - x_{CG})\dot{\theta}}{V_x - 0.5(t_F + y_{CG})\dot{\theta}}\right)$$

$$\alpha_2 = -\delta + \text{atan}\left(\frac{V_x\beta + (l - x_{CG})\dot{\theta}}{V_x + 0.5(t_F - y_{CG})\dot{\theta}}\right)$$

$$\alpha_3 = -\text{atan}\left(\frac{-V_x\beta + x_{CG}\dot{\theta}}{V_x - 0.5(t_R + y_{CG})\dot{\theta}}\right)$$

$$\alpha_4 = -\text{atan}\left(\frac{-V_x\beta + x_{CG}\dot{\theta}}{V_x + 0.5(t_R - y_{CG})\dot{\theta}}\right) \quad (2-4)$$

其中 V_y 是车辆侧向速度， β 为车身侧偏角， α_n 为第 n 个车轮的侧偏角， δ 为前轮转角， t_F 和 t_R 分别为前后车轴长度， l 为车辆轴距， $\dot{\theta}$ 为横摆角速度， x_{CG} 和 y_{CG} 为纵向和侧向质心位置。

由于本文设计的动力学模型的目标是为规划提供动力学约束，需要在各种工况下对各个轮胎的轮胎力做出准确的估计，因此考虑了重量转移对垂向载荷的影响。影响重量转移一个关键特性是滚动中心，它被定义为施加到弹簧质量上的侧向力不会产生滚动力矩的点。滚动中心可由前后悬架的运动学确定，并由弹簧质量围绕其滚动的虚拟滚动轴连接。应该注意的是，在实际车辆系统中

滚动中心实际上会随着悬架振动而轻微移动。然而，在本工作中，滚动中心将被视为相对于车架的静态位置。为了简化和概括悬架特性，簧载和非簧载质量之间的连接将建模为一系列旋转接头。每个旋转中心放置一个接头，其轴与旋转轴对齐。第三个旋转接头位于俯仰中心，其轴平行于车辆 y 轴。内部车辆动力学的最显著影响是四个车轮之间的重量转移。这种转移可分为两个正交部分：纵向和侧向。

纵向重量转移是由油门和制动输入引起的加速度引起的，如下计算：

$$\begin{aligned}\Delta W_{xF} &= \frac{-(z_{CG} + r_e)m_s \dot{V}_x}{l} \\ \Delta W_{xR} &= \frac{(z_{CG} + r_e)m_s \dot{V}_x}{l}\end{aligned}\quad (2-5)$$

其中 ΔW_{xF} 和 ΔW_{xR} 分别为前轮和后轮的纵向重量转移，为 m_s 为簧上质量， r_e 为滚动中心关于质心的相对高度， g 为重力加速度。

重心的侧向偏移将影响纵向重量转移的分布。侧向重心偏移的影响可以通过计算每个车轮的纵向重量转移系数来补偿：

$$\begin{aligned}C_{x1} &= \frac{1}{2} - \frac{y_{CG}}{0.5t_F} \\ C_{x2} &= \frac{1}{2} + \frac{y_{CG}}{0.5t_F} \\ C_{x3} &= \frac{1}{2} - \frac{y_{CG}}{0.5t_R} \\ C_{x4} &= \frac{1}{2} + \frac{y_{CG}}{0.5t_R}\end{aligned}\quad (2-6)$$

其中是 C_{xn} 为第 n 个轮胎的纵向载荷转移系数。

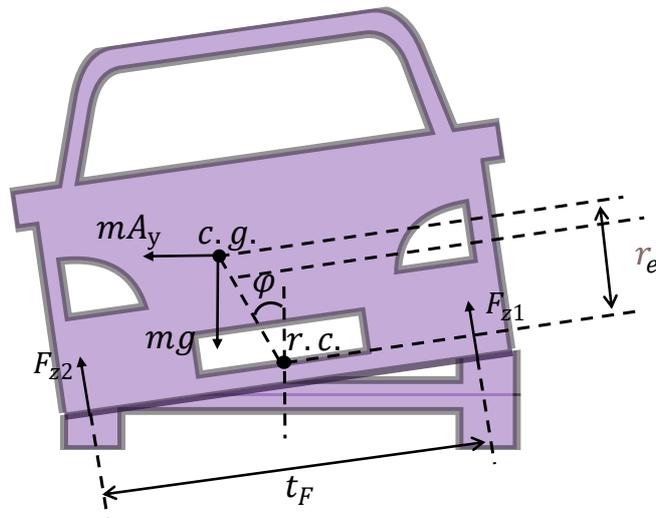


图2-5 侧倾车辆动力学模型示意图

侧向重量转移是由于作用在重心上的侧向加速度引起的，并且受到重心高度、滚动中心位置、悬架柔度和重心侧向偏移的影响，通过车辆的侧倾动力学模型来分析，如图 2-5。滚动中心的影响导致侧向载荷转移由一个特征高度控制，该特征高度是质心和在质心纵向位置的滚动轴之间的距离：

$$H = z_{CG} + r_e - \frac{z_{RC_F}x_{CG} + z_{RC_R}(l - x_{CG})}{l} \quad (2-7)$$

其中 H 是横向滚动特征高度， z_{RC_F} 和 z_{RC_R} 为前后侧向滚动中心相对于车辆坐标系原点的高度。

侧向加速度的影响可以通过计算车辆一侧的力矩得出：

$$\begin{aligned} \Delta W_{yF} &= \frac{k_{\varphi_F} m_s g A_y H}{t_F (k_{\varphi_F} + k_{\varphi_R} - m_s g A_y y_{CG} - m_s g H)} + m_s g A_y \frac{x_{CG} x_{RC_F}}{l t_F} \\ \Delta W_{yR} &= \frac{k_{\varphi_R} m_s g A_y H}{t_F (k_{\varphi_F} + k_{\varphi_R} - m_s g A_y y_{CG} - m_s g H)} + m_s g A_y \frac{(l - x_{CG}) x_{RC_R}}{l t_R} \end{aligned} \quad (2-8)$$

其中 ΔW_{yF} 和 ΔW_{yR} 分别为前轮和后轮的侧向重量转移， k_{φ_F} 和 k_{φ_R} 是前后侧向侧倾摩擦系数， x_{RC_F} 和 x_{RC_R} 为前后侧向滚动中心相对于车辆坐标系原点的纵向位置。

通过计算静态重量分布，可以找到考虑质心位置变化的重量分布：

$$\begin{aligned} F_{z1s} &= \frac{C_{x1} m_s g x_{CG}}{l} + m_u g \\ F_{z2s} &= \frac{C_{x2} m_s g x_{CG}}{l} + m_u g \\ F_{z3s} &= \frac{C_{x3} m_s g (l - x_{CG})}{l} + m_u g \\ F_{z4s} &= \frac{C_{x4} m_s g (l - x_{CG})}{l} + m_u g \end{aligned} \quad (2-9)$$

其中 F_{zn_s} 是第 n 个轮胎的静态垂向载荷， m_u 是簧下质量。

然后可以应用动态重量转移的效果：

$$\begin{aligned} F_{z1} &= F_{z1s} - \Delta W_{yF} - C_{x1} \Delta W_{xF} \\ F_{z2} &= F_{z2s} + \Delta W_{yF} - C_{x2} \Delta W_{xF} \\ F_{z3} &= F_{z3s} - \Delta W_{yR} + C_{x3} \Delta W_{xR} \\ F_{z4} &= F_{z4s} + \Delta W_{yR} + C_{x4} \Delta W_{xR} \end{aligned} \quad (2-10)$$

在横向和纵向加速度条件下，这些值在给定质心位置的情况下产生每个车轮上的垂向载荷。

在得到滑移率、侧偏角和垂向载荷后，可以使用轮胎模型计算轮胎力，然后求各个轮胎力的矢量和，以计算作用在车上的合力：

$$\begin{aligned}
 F_x &= \sum_{n=0}^4 F_{xn} \cos \delta_n + F_{yn} \sin \delta_n \\
 F_y &= \sum_{n=0}^4 F_{xn} \sin \delta_n + F_{yn} \cos \delta_n
 \end{aligned} \tag{2-11}$$

其中 δ_n 为第 n 个轮胎的转角，因此该模型可以应用于前轮转向、前后轴转向和四轮转向的汽车。本文将以前轮转向的汽车为例进行研究，但该研究可以扩展为其他转向方式的汽车。前轮转向汽车存在如下关系：

$$\begin{aligned}
 \delta_1 &= \delta_2 = \delta \\
 \delta_3 &= \delta_4 = 0
 \end{aligned} \tag{2-12}$$

此外，由于力产生的横摆力矩 N 可通过力及其力矩臂的乘积计算：

$$\begin{aligned}
 N &= [0.5t_F \cos \delta + (l - x_{CG}) \sin \delta] F_{x1} \\
 &\quad - [0.5t_F \cos \delta - (l - x_{CG}) \sin \delta] F_{x2} \\
 &\quad - 0.5t_R F_{x3} \\
 &\quad - 0.5t_R F_{x4} \\
 &\quad + [(l - x_{CG}) \sin \delta - 0.5t_F \cos \delta] F_{y1} \\
 &\quad + [(l - x_{CG}) \sin \delta + 0.5t_F \cos \delta] F_{y2} \\
 &\quad - x_{CG} F_{y3} \\
 &\quad - x_{CG} F_{y4}
 \end{aligned} \tag{2-13}$$

为了根据横摆力矩确定所需的横摆项，有必要计算横摆轴的转动惯量。由于该值在没有专用设备的情况下很难测量，因此有必要对该值进行近似。执行此操作的一种方法是使用定义如下的动态指数^[48]：

$$DI = \frac{r_z^2}{x_{CG}(l - x_{CG})} \tag{2-14}$$

其中 r_z 为车辆质量到 z 轴的等效半径，大多数车辆的动态指数约等于 1，高性能车辆通常略低。根据动力指数，转动惯量的合理近似值可计算为：

$$I_z = m_s x_{CG} (l - x_{CG}) DI \tag{2-15}$$

在 x - y 平面上应用牛顿第二定律可以计算由于力引起的速度变化：

$$\begin{aligned}
 \dot{V}_x &= \frac{F_x}{M} + V_y \dot{\theta} \\
 \dot{V}_y &= \frac{F_y}{M} - V_x \dot{\theta} \\
 \dot{\theta} &= \frac{N}{I_z}
 \end{aligned} \tag{2-16}$$

然后通过积分确定每个方向的速度，并通过双积分确定位置和横摆角：

$$\begin{aligned}
 V_x &= \int \dot{V}_x dt + V_{x0} \\
 V_y &= \int \dot{V}_y dt + V_{y0} \\
 \dot{\theta} &= \int \ddot{\theta} dt + \dot{\theta}_0 \\
 X &= \int V_x dt + X_0 \\
 Y &= \int V_y dt + Y_0 \\
 \theta &= \int \dot{\theta} dt + \theta_0
 \end{aligned} \tag{2-17}$$

最后，给出上述车辆模型的参数值选择：

表2-1 整车模型参数

整车参数	参数值	整车参数	参数值
整车质量 m	1370kg	侧倾摩擦系数 k_{ϕ_F}	48000N/rad
簧上质量 m_s	1170kg	纵向质心位置 x_{CG}	1.373m
簧下质量 m_u	200kg	侧向质心位置 y_{CG}	0.0m
侧偏刚度 C_α	16000	垂向质心位置 z_{CG}	0.463m
纵向刚度 C_s	28000	旋转中心高度 z_{RC}	0.3m
横摆转动惯量 I_z	1150kg·m ²	侧倾刚度 C_ϕ	28000
车辆轴距 l	2.7m	车轮有效半径 r_{eff}	0.316m
后轴长度 t_F	1.2m	前轴长度 t_F	1.2m

2.4 Milliken 动力学约束

本章所建立模型的最终目标是生成一个可以实时更新的动力学约束，为运动规划器提供指导，该约束应以一种操纵性能限制以及中间状态信息的形式表示。非线性方程组表示的车辆动力学模型用于描述在相关输入变量的单个值下的车辆状态，这使得获取动力学边界需要在线遍历所有的状态。此外，由于车辆动力学模型的复杂性，这些方程通常必须使用数值迭代技术进行求解。这些难点使得直接使用非线性动力学方程组实时求解规划所需的动力学边界不可行。为了解决这一难题，将开发一种包含离线预计算的力学模型表示方法。

2.4.1 Milliken 力矩法

Milliken 力矩法^[43]是一种表示由于一组输入变量作用在车辆上产生的稳态力和力矩的方法。Milliken 力矩法可以将非线性方程组形式的车辆动力学模型转换为 Milliken 动力学模型，用于表示多个前轮转角 δ 和车身侧偏角 β 下的横向力和横摆力矩。

Milliken 动力学模型的建立方法分为约束实验法和计算机求解法^[43]。约束实验将汽车支撑在一条可以以真实汽车速度移动的传送带上，假设驱动力和制动力可通过车辆的发动机和制动器（动力系统）施加。约束系统由三条线组成：一条用于纵向约束车辆以反应牵引力或制动力，两条横向约束，一条位于车辆前部，另一条位于重心。两条横向线上的力之和为汽车上的总侧向力；前横向线上的力乘以到重心的距离，得到车辆上的横摆力矩。

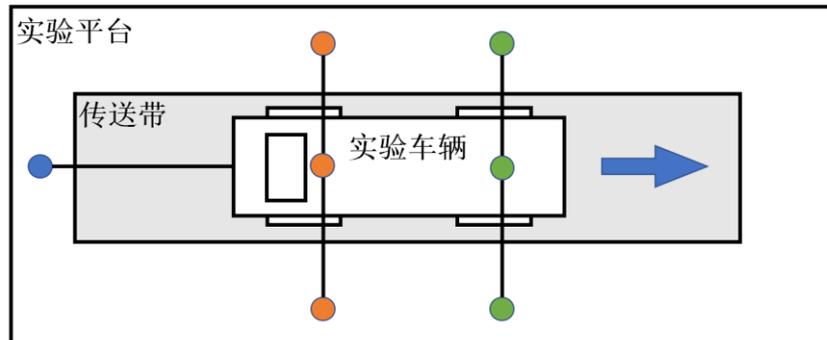


图2-6 Milliken 动力学模型的建立方法分为约束实验

本文采取计算机求解法，在当前车辆的纵向速度 V_x 、纵向加速度 \dot{V}_x 下，通过为上文中的车辆模型施加不同的前轮转角 δ 、车身侧偏角 β ，得到多组车辆的横摆力矩和侧向力的变化曲线，如图 2-7(a)所示：

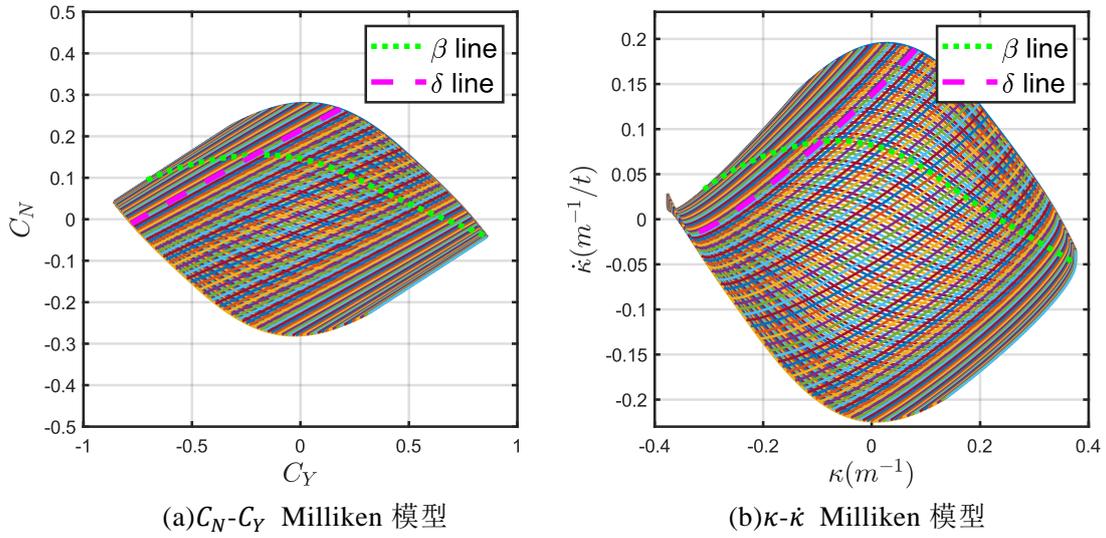


图2-7 Milliken 动力学模型

其中 C_N 和 C_Y 分别为横摆力矩和侧向力的归一化形式，归一化方法如下：

$$\begin{aligned} C_N &= \frac{N}{Mgl} \\ C_Y &= \frac{F_y}{Mg} \end{aligned} \quad (2-18)$$

其中 M 为车辆总质量。Milliken 动力学模型采用预计算的形式表示车辆动力学特性，比微分方程组更适合提取动力学安全边界。假设所有侧向力都作为向心力，结合当前时刻的状态，将力-力矩的 Milliken 动力学模型转化为 κ - $\dot{\kappa}$ 的 Milliken 动力学模型，使其更适用于运动规划算法，曲率可以根据向心力公式计算：

$$\kappa = \frac{F_y}{V_x^2 M} \quad (2-19)$$

其中 κ 为当前车辆运动的曲率。根据牛顿第二定律：

$$\frac{N}{I_z} = \dot{\theta} = \frac{d}{dt}(V_x \kappa) = \dot{V}_x \kappa + V_x \dot{\kappa} \quad (2-20)$$

将 κ 带入上式可得：

$$\dot{\kappa} = \left(\frac{N}{I_z} - \dot{V}_x \kappa \right) / V_x \quad (2-21)$$

转化后的模型如图 2-7(b)所示。

为证明 Milliken 动力学模型具有可接受的精度，设计实验将 Milliken 模型与 VeDYNA 中较准确的多体动力学模型进行对比。在 VeDYNA 中截取双移线

动作的数据，包括：车辆纵向加速度 \dot{V}_x 、车辆纵向速度 V_x 、车身侧偏角 β 、车前轮转向角 δ 、各个轮的横纵向受力。取 VeDYNA 中摩擦系数为 0.9。将 \dot{V}_x 、 V_x 、 β 、 δ 、 μ 作为输入数据，利用 Milliken 动力学模型计算出各个轮胎的横纵向受力，比较其与 VeDYNA 输出值的差别。

下图是在车速 100km/h 双移线工况下，Milliken 动力学模型与多体动力学模型的精度比较，可以看出 Milliken 动力学模型与 veDYNA 模型输出在测试工况下基本保持一致。

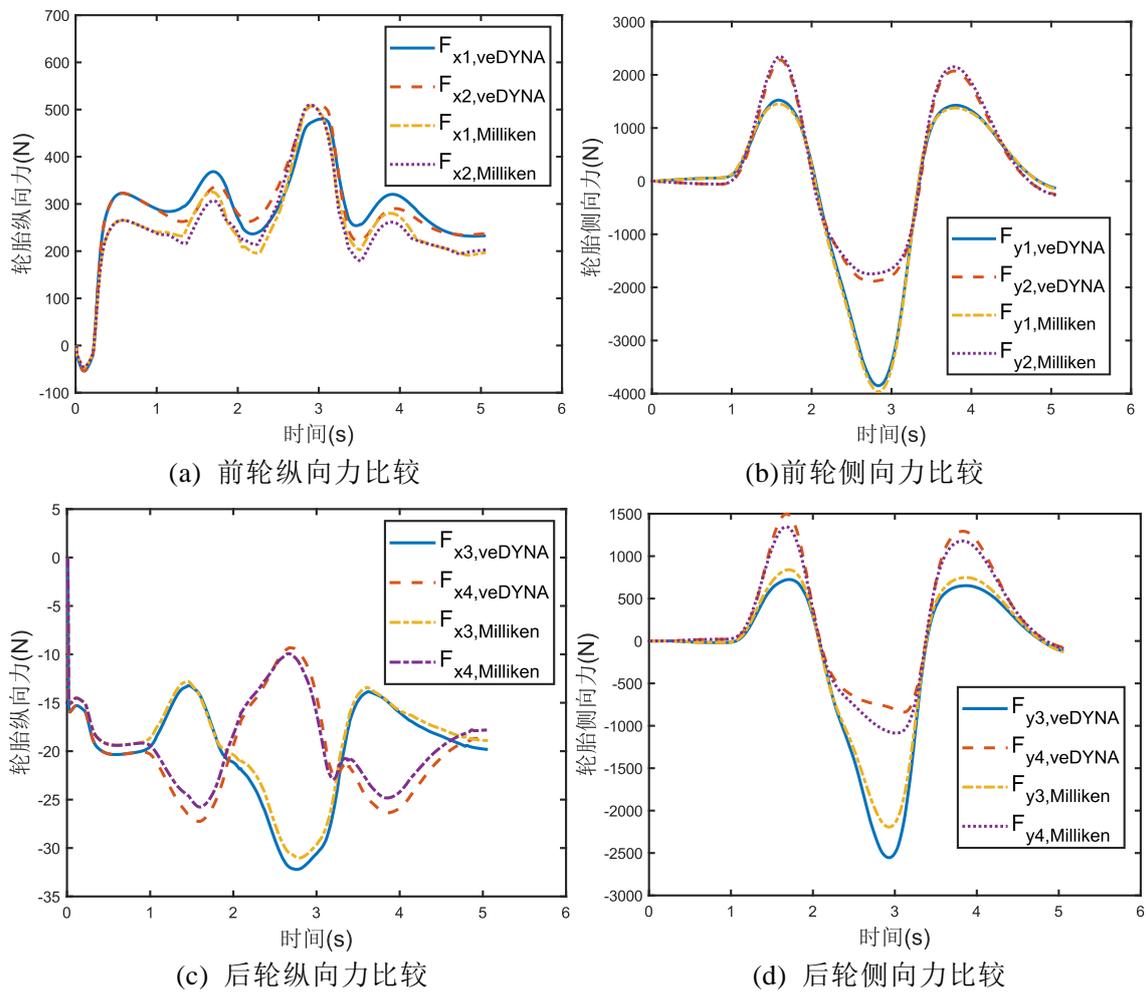


图2-8 Milliken 动力学模型精度验证

2.4.2 Milliken 动力学安全指标

在完成 Milliken 动力学模型建立后，需要设计指标对动力学风险进行量化，从而定义安全边界。根据 Carrier 所述^[44]，车辆在三种主要模式下会失去驾驶安

全性：滑动、旋转、侧倾，通过定义滑动指数(SI, slipping index)、旋转指标(DI, drifting index)和侧倾指标(RI, rolling index)来描述每一种模式，这三个指标将联合约束车辆的动力学安全运行范围。

在大多数情况下，车辆侧倾比旋转更加危险，而车辆旋转比滑动更加危险。车辆侧倾会导致一边甚至两边轮胎离开地面，车辆可能连续翻滚，从而极大的威胁驾驶员的生命安全；车辆旋转会导致车辆的速度和方向完全失控，从而大概率会引发碰撞；车辆滑动会导致转向系统失效，使车辆的行驶方向变得失去控制。同时，在大多数情况下，风险越大的危险行驶模式对应了较难触及的动力学状态边界，因为车辆只有在非常极限的行驶状态下才会引发侧倾或旋转。车辆在行驶状态下需要满足滑动指数、旋转指标、侧倾指标均小于某一安全阈值，这样才能保证车辆在行驶过程中不进入任何一种危险状态。

2.4.2.1 侧倾指标

侧倾指标 $RI^{[44]}$ 是用来预测发生侧翻的阈值。这种方法使用能量公式来确定车辆一侧的两个轮子与地面失去接触的临界阈值的近似值。这种能量 E_{roll} 有两个分量，一个是由弹簧压缩、重力决定的势能 U_{roll} ，另一个是由簧载质量的运动决定的动能 T_{roll} ，可以表示为：

$$\begin{aligned} U_{roll} &= \frac{1}{2} \varphi^2 C_\varphi - m_s g (z_{CG} - z_{RC}) (1 - \cos \varphi) \\ T_{roll} &= \frac{1}{2} (I_x + m_s (z_{CG} - z_{RC})^2) \dot{\varphi}^2 \\ E_{roll} &= U_{roll} + T_{roll} \end{aligned} \quad (2-22)$$

其中 U_{roll} 为侧倾势能， T_{roll} 为侧倾动能， E_{roll} 为侧倾能量， C_φ 为悬架刚度。动能项依赖于角速度，在稳态时为零，而势能项与瞬态效应无关。这种分离允许 RI 计算分两部分进行。可以先根据当前角速度瞬时值计算旋转能量，然后通过 Milliken 动力学模型计算势能，最终的能量计算是两个值的线性叠加。

临界翻转能量 E_{crit} 是在力矩平衡的一般情况下诱发车轮离地的最小能量，这个能量可以通过假设横向力的最大值 $F_y = \mu F_z$ 并利用力矩平衡来计算。临界能量可定义为满足下列条件的最小能量：

$$(l - \mu z_{RC}) M g = \varphi C_\varphi + \dot{\varphi} k_\varphi \quad (2-23)$$

其中 k_φ 为悬架阻尼系数。

在整车模型中通过重量转移计算了各个轮胎的垂向载荷，可以通过下式来根据垂向载荷计算车身侧偏角 φ ：

$$\varphi = (F_{z1} + F_{z3} - F_{z2} - F_{z4})/C_{\varphi} \quad (2-24)$$

由临界条件和侧倾能量，可以获得临界能量，从而定义侧倾指标 RI：

$$RI = \frac{E_{crit} - E_{roll}}{E_{crit}} \quad (2-25)$$

需要注意 RI 值小于 0 并不一定意味着车辆会发生侧翻，它预测了车轮离地时的不稳定侧倾状态，比实际侧翻的预测更为保守。还需注意的是，此度量仅适用于一般的侧倾场景。如果车辆与路肩的横向碰撞而失稳，车辆的状态将很难被预测，本文提出的模型无法适用于这种情况。

2.4.2.2 旋转指标

旋转指标 DI^[44]是用来预测在车辆发生旋转之前方向稳定性丧失的阈值。当车辆后轮接近力饱和时，会出现横摆不稳定性。因此，DI 定义为总后轴力与最大可用后轴力的比。饱和轮胎力定义为路面摩擦系数乘以垂向载荷的值。为了与 DI 保持一致，将该比值的值归一化，如下式所示，值为 0 表示预测车辆状态即将旋转。

$$DSM = \frac{\mu F_{zF} - F_{yF}}{\mu F_{zF}} \quad (2-26)$$

2.4.2.3 滑动指标

滑动指标 SI^[44]的设计是为了预测由于车辆进入滑动状态而失去方向控制的阈值。滑动状态实际上可能是稳定的，但这是不可取的，因为它代表控制输入和车辆路径之间关系的退化或丧失。对于本研究中考虑的这种类型的车辆，方向可控性在很大程度上取决于前轴的力饱和。因此，SI 将被定义为总前轴力与最大可用前轴力的比值。与 DI 相似，这个值被归一化，值为 0 表示预测车辆状态即将滑动。

$$SI = \frac{\mu F_{zR} - F_{yR}}{\mu F_{zR}} \quad (2-27)$$

2.4.2.4 约束的 Milliken 动力学模型

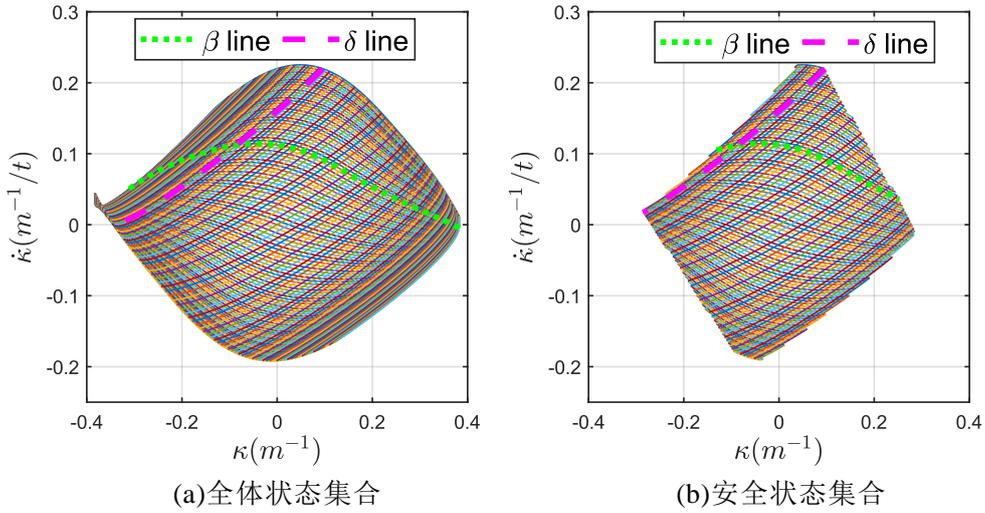


图2-9 Milliken 动力学安全状态

在 Milliken 动力学模型生成的过程中，可以对于某个状态定义一个表示是否安全的逻辑值 $f_s(\dot{V}_x, V_x, \delta, \beta)$ ，称为安全指数。安全指数如下取值：

$$f_s(\dot{V}_x, V_x, \delta, \beta) = \begin{cases} 1, & RI \cdot DI \cdot SI > 0 \\ 0, & else \end{cases} \quad (2-28)$$

安全指数等于 1 的状态定义为安全状态，图 2-9 为在确定速度、加速度、路面附着系数下 Milliken 动力学模型中的全体状态和安全状态。

需要注意的是，由于安全状态是被非安全状态所包围，而且在 Milliken 动力学模型定义的 κ - κ 平面中是一个连通的区域，这使得安全状态集合的包络线等价于安全状态和非安全状态的分界线，可以将其定义为安全边界 (SE, safe edge)。安全状态是通过检验状态的侧倾指标、旋转指标、指标定义的。因此上述过程能够将 Milliken 动力学模型中的状态分为安全状态和非安全状态，但不能确定 SE。

通过深度优先遍历算法遍历安全状态的集合，可以获得安全状态集合的包络线，即 Milliken 动力学模型的 SE。如图 2-10 所示，SE 搜索算法遍历了确定速度、加速度、路面附着系数下的动力学安全区域，并且成功将安全边界进行了标记和存储。Milliken 动力学中的安全边界可以用于指导车辆驾驶员的行为，或者指导自动驾驶运动规划算法。安全边界从动力学分析出发，为车辆安全形势提供了约束条件，避免车辆处于可能引发侧倾、旋转、滑动的危险状态。

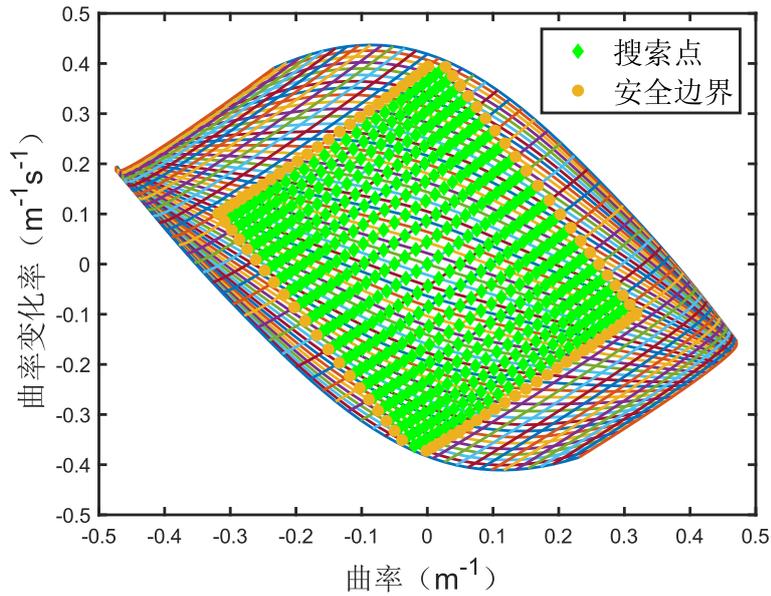
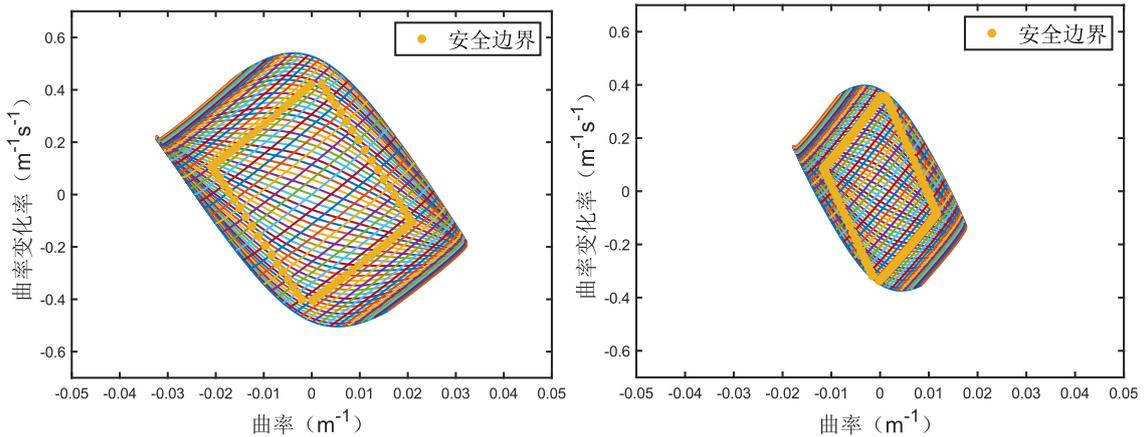


图2-10 Milliken 动力学安全边界

通过下图可以看出，Milliken 动力学模型以及安全边界会随着路面附着系数、当前车辆纵向速度 V_x 、当前车辆加速度 \dot{V}_x 变化。图 2-11(a)为对照组，从 2-11(b)与 2-11(c)中可以看出速度提升和路面附着系数降低都会导致安全边界收缩，图 2-11(d)与 2-11(e)说明加减速同样会导致安全边界收缩。最后，2-11(f)说明在速度非常低时，安全边界的约束接近于车辆的运动学约束，即最小转向半径的约束。



(a) $V_x = 60\text{km} \cdot \text{h}^{-1}, \dot{V}_x = 0, \mu = 0.9$

(b) $V_x = 100\text{km} \cdot \text{h}^{-1}, \dot{V}_x = 0, \mu = 0.9$

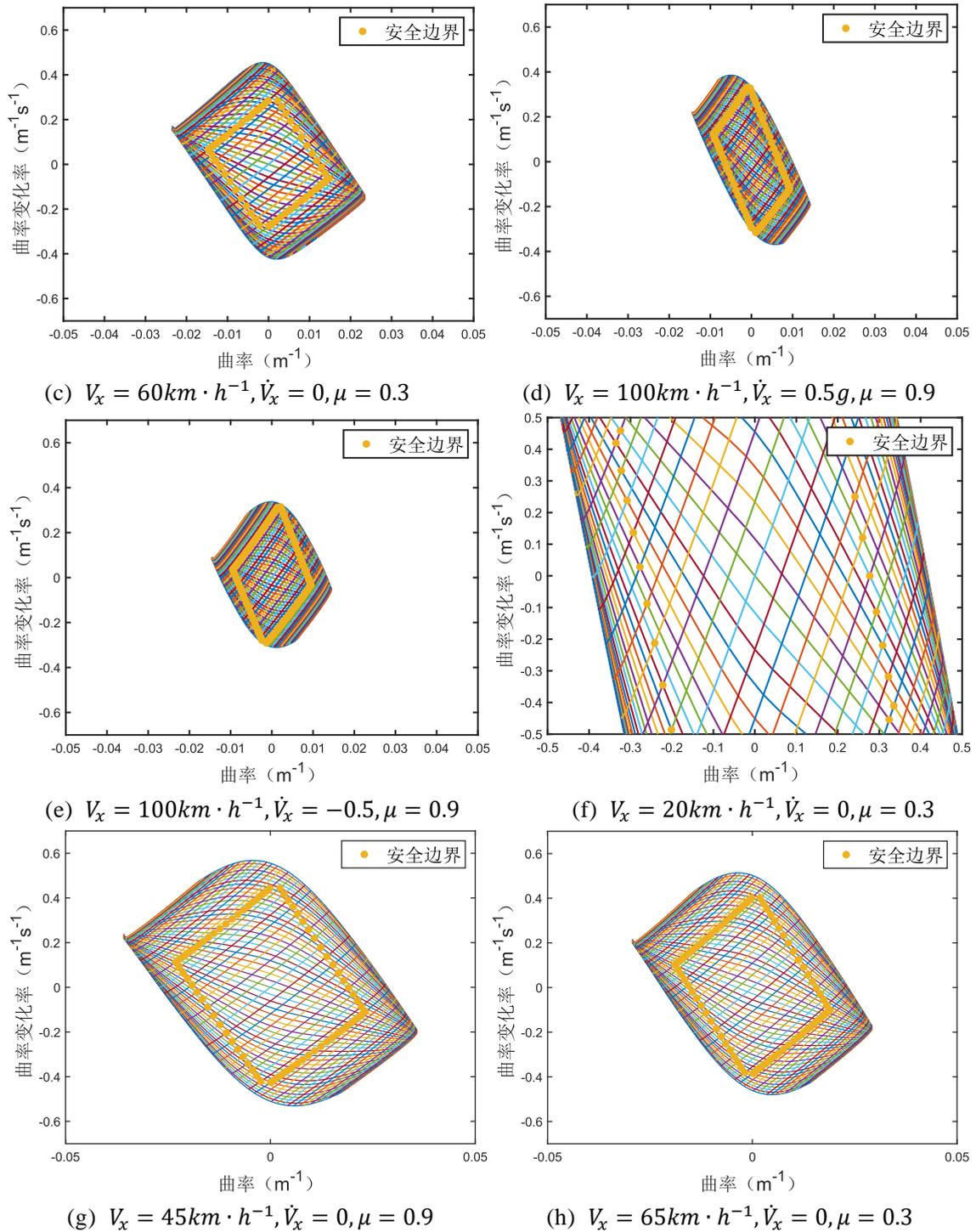


图2-11 不同状态下的 Milliken 动力学安全边界

需要注意当速度差达到 20km/h 以上时, 不同速度下的安全边界才会相差较为明显, 而速度差小于 20km/h 的两个场景之间的安全边界是非常接近的, 如图 2-11(g)和 2-11(h)所示。在具有设定巡航速度的运动规划场景中, 规划速度变化一般不超过 20km/h, 此时规划过程中的安全边界变化较小, 因此可以通过规划

初始状态估计轨迹安全边界。

2.4.3 安全距离

对于每一条轨迹都可以映射到 Milliken 动力学模型中，如下图所示：

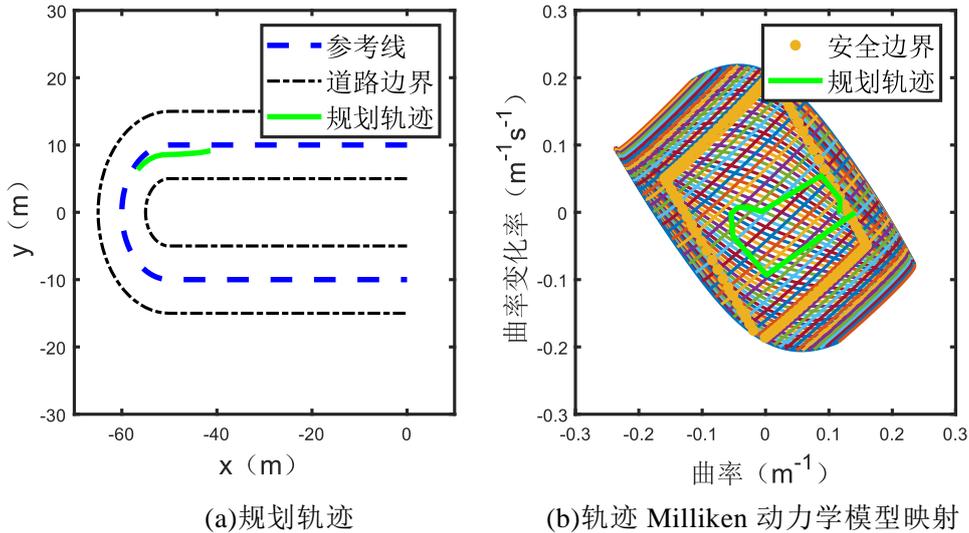


图2-12 规划轨迹的 Milliken 动力学模型映射

图中左图为一地图坐标系下的轨迹，而右图为该轨迹在 Milliken 动力学模型中的映射。在 2.4.4 中，本文根据侧倾指标、旋转指标、滑动指标生成了 Milliken 动力学模型中的安全边界 SE，进一步定义了约束的 Milliken 动力学模型。从上图中可以看出，在一些风险较大的驾驶场景，比如道路湿滑条件下通过 U 型弯道，规划轨迹在 Milliken 空间中的映射（绿色曲线）存在超出安全边界部分。超出安全边界的状态代表了车辆出现了以下三种危险状况中的至少一种：

侧倾指标 RI 超过了发生侧翻的阈值。此时车辆某一侧的两个轮子与地面失去接触的临界点，这是由弹簧压缩势能和簧载质量动能之和过大导致的；

旋转指标 DI 超出阈值，代表了对方向稳定性丧失和车辆发生旋转的边界预测。此时车辆后轮接近力饱和时，会出现横摆不稳定性，总后轴力与路面摩擦系数和垂向载荷乘积的比超过 1。

滑动指标 SI 超出阈值，代表了车辆进入滑动状态而失去方向控制。这是一种可能稳定的但不可取的状态，代表控制输入和车辆路径之间关系的退化或丧

失，在动力学上反应为前轴的力饱和。

因此在定义动力学代价 C_{mil} 前，定义 Milliken 动力学模型中状态到安全边界 SE 的最小距离 d_{mil} ，称为安全距离。如下图所示：

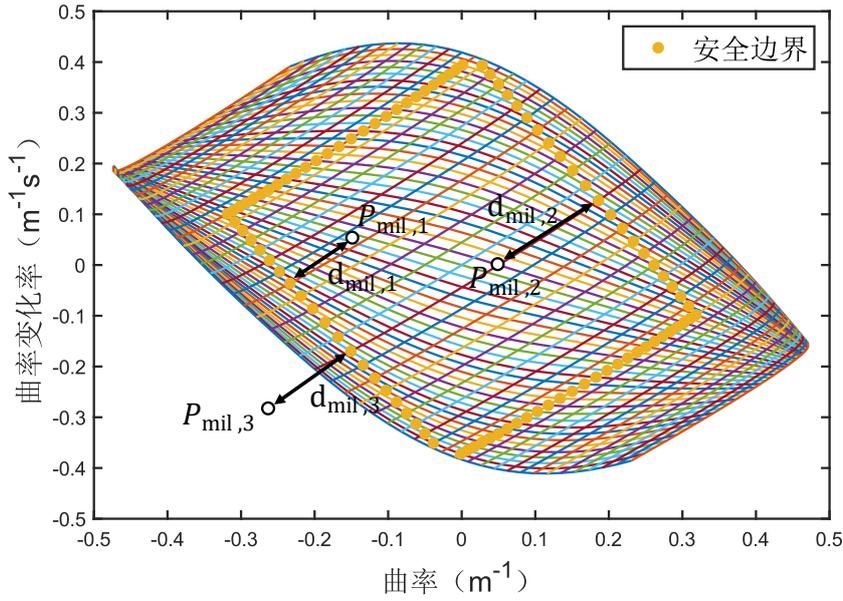


图2-13 安全距离

图中 $P_{mil,1}$ 、 $P_{mil,2}$ 、 $P_{mil,3}$ 为 Milliken 空间的状态采样点，其中 $P_{mil,1}$ 、 $P_{mil,2}$ 在 SE 内，因此状态的安全距离定义为：

$$d_{mil}(t) = (\kappa_e(t) - \kappa(t))^2 + (\dot{\kappa}_e(t) - \dot{\kappa}(t))^2 \quad (2-29)$$

$\kappa(t)$ 和 $\dot{\kappa}(t)$ 为 t 时刻轨迹状态的曲率和曲率导数， $\kappa_e(t)$ 和 $\dot{\kappa}_e(t)$ 为 t 时刻轨迹状态对应的 SE 上的最近状态的曲率和曲率导数。当状态接近 SE 时，说明即将进入侧倾、旋转或者滑移的危险状态，因此安全距离会减小。对于在 SE 外的状态 $P_{mil,3}$ 来说，如果沿用上述定义，会导致 $d_{mil,3} > d_{mil,1}$ ，这显然是错误的，因此在定义轨迹的安全距离 D_{mil} 时对 SE 内侧和外侧的状态做不同处理：

$$D_{mil} = \min_{t < t_f} d_{mil}(t)$$

$$d_{mil}(t) = \begin{cases} (\kappa_e(t) - \kappa(t))^2 + (\dot{\kappa}_e(t) - \dot{\kappa}(t))^2, & \kappa_e(t)^2 + \dot{\kappa}_e(t)^2 > \kappa(t)^2 + \dot{\kappa}(t)^2 \\ -(\kappa_e(t) - \kappa(t))^2 + (\dot{\kappa}_e(t) - \dot{\kappa}(t))^2, & \kappa_e(t)^2 + \dot{\kappa}_e(t)^2 \leq \kappa(t)^2 + \dot{\kappa}(t)^2 \end{cases} \quad (2-30)$$

其中 $d_{mil}(t)$ 为轨迹 t 时刻状态的安全距离，在 SE 内侧为正值，否则为负值，

保证 SE 内侧安全距离大于 SE 外侧； D_{mil} 为轨迹的最小安全距离，如果 $D_{mil} < 0$ ，说明轨迹中存在 Milliken 动力学模型 SE 外的状态，轨迹存在侧倾、旋转或者滑移风险。

2.5 本章小结

本章建立并分析了轮胎模型和整车模型，通过 VeDYNA 仿真验证模型的准确性。为了使模型更适合于指导运动规划，本论文通过 Milliken 力矩法将模型转换为 Milliken 动力学模型。同时，定义滑动指数、旋转指数、侧倾指数，并通过深度优先搜索在 Milliken 动力学模型中获得动力学安全边界。通过对比不同车辆状态下的安全边界，分析了不同路面附着系数、不同纵向速度、车辆加减速对安全边界的影响。最后，根据安全边界定理了安全距离，量化车辆状态的 dynamics 安全风险，用于在后续章节中指导运动规划算法。

第 3 章 考虑动力学约束的道路空间规划方法

3.1 引言

本章设计了考虑动力学约束的道路空间运动规划。自动驾驶运动规划模块的输出包括高精地图、车辆底盘信息和环境感知信息，运动规划模块需要先对这些信息进行预处理，从而构造出规划空间。针对有车道线信息的道路空间，本文首先建立了参考线模型，将从感知或者地图获取的车道信息通过共线点删除和三次样条插值，获得平滑的规划参考线。

其次，本文建立了量化道路空间环境碰撞风险的风险势场模型，来描述道路空间中形状与运动较为规则的障碍物。以障碍物占用空间分布的估计值表示了障碍物的风险，并且满足估计值向速度方向倾斜。

最后，本文设计了基于 Frenet 坐标系的最优规划算法，在代价函数中定义与安全边界相关的动力学代价，从而将安全距离的动力学约束加入到规划器中，提高了道路空间运动规划的安全性。

3.2 参考线建模

自动驾驶的道路空间由结构化道路构成。在自动驾驶技术中，结构化道路是指符合一定规范和标准的道路，包括交通标志、车道线、车道宽度、路口等基本元素。在结构化道路中，道路元素之间有着明确的规划和划分，便于通过传感器获取道路信息并进行元素区分，并通过自身的定位和导航系统精确地了解车辆所处的位置和行驶方向。

道路空间中的车道信息可以从高精地图中获取，也可以通过传感器获取。大多数道路还是没有配套的高精地图的，而且想要为所有的道路都建立高精地图是一件非常困难的工作。因此在没有高精地图的环境下车辆需要通过车载传感器获取车道信息。常用的传感器包括摄像头、激光雷达、毫米波雷达和超声波传感器等。实时感知的效果会受到路面状况、天气条件、车速等因素对车道线检测的影响，因此与在不同环境下车道线的数据特征会存在一定区别。从高

精地图获取的车道线与感知得到的车道线也会存在差别。

因此下面将建立统一的参考线模型，主要包括下面两部分：

删除共线点：删除共线点是参考线预处理中的一个重要步骤，它可以减少参考线中不必要的冗余信息，降低计算复杂度，提高参考线的质量和精度。共线点指的是在一条直线上的点，它们对参考线的表示没有实际意义，但会增加计算和存储的负担。共线点也会影响运动规划效果。由于共线点不提供有用的信息，如果保留这些点，会使参考线变得复杂，增加路径规划的插值或者优化问题难度。而删除共线点可以使参考线更加简洁、清晰，更符合车辆的行驶特性，有助于提高路径规划的效率和性能。

三次自然样条插值：参考线形状和位置都会受到一定的噪声和误差的影响，因此需要对其进行插值和平滑处理，以获得更加准确和可靠的参考线模型。本文采用了三次自然样条曲线进行插值，通过定义多项式方程来逼近原始数据点，从而生成平滑的曲线。与其他插值方法相比，三次自然样条插值的优点在于可以减少振荡和过拟合的现象，同时保证生成的曲线在给定的区间上具有两阶导数的连续性和零值边界条件。

3.2.1 共线点删除

共线点删除算法是指在自动驾驶参考线生成过程中，对于一些几乎在同一直线上的点进行删除的算法。该算法的目的是减少参考线中不必要的冗余点，降低计算量，并提高自动驾驶系统的运行效率。

本文采用的基于夹角的共线点删除算法计算相邻三个点的夹角，若夹角小于设定的阈值，则将中间点删除，保留前后两个点，并继续对剩余点进行检查，直到所有点都被检查过。基于夹角的共线点删除算法则不容易产生失真，并且能够更好地保留原始数据中的拐点和锐角，因此在处理复杂曲线的情况下表现更好。但是，它需要更复杂的数学运算来计算夹角。在使用计算机进行计算时，可以通过向量运算简化计算过程，如下所示：

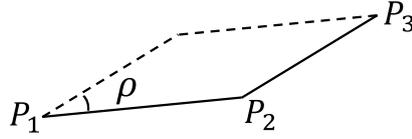


图3-1 共线点删除

判断图中 P_1 、 P_2 、 P_3 共线等价于向量 P_1P_2 和向量 P_2P_3 的夹角 $\rho = 0$ ，即：

$$\cos\rho = \frac{P_1P_2 \cdot P_2P_3}{|P_1P_2||P_2P_3|} = 1 \quad (3-1)$$

假设 $P_1 = (x_1, y_1)$ ， $P_2 = (x_2, y_2)$ ， $P_3 = (x_3, y_3)$ ， $P_1P_2 = (x_2 - x_1, y_2 - y_1) = (x_{21}, y_{21})$ ， $P_2P_3 = (x_3 - x_2, y_3 - y_2) = (x_{32}, y_{32})$ ，则上述共线条件可以如下表示：

$$x_{21}x_{32} + y_{21}y_{32} = \sqrt{x_{21}^2 + y_{21}^2}\sqrt{x_{32}^2 + y_{32}^2} \quad (3-2)$$

等式两边取平方并移相移项得：

$$(x_{21}x_{32} - y_{21}y_{32})^2 = 0 \quad (3-3)$$

即 $x_{21}x_{32} - y_{21}y_{32} = 0$ 。将 $x_{21} = x_2 - x_1$ ， $y_{21} = y_2 - y_1$ ， $x_{32} = x_3 - x_2$ ， $y_{32} = y_3 - y_2$ 代入上式得：

$$(x_2 - x_1)(y_3 - y_2) = (x_3 - x_2)(y_2 - y_1) \quad (3-4)$$

化简得：

$$x_1y_2 - x_2y_1 + x_3y_1 - x_1y_3 + x_2y_3 - x_3y_2 = 0 \quad (3-5)$$

因此基于夹角的共线点删除算法可以定义为：

$$x_1y_2 - x_2y_1 + x_3y_1 - x_1y_3 + x_2y_3 - x_3y_2 < \varepsilon \quad (3-6)$$

其中 ε 为共线阈值， ε 越大删除的点越多。改进的基于夹角的共线点删除算法不容易产生失真且避免了复杂的数学运算，因此本文使用这一方法进行共线点删除。

3.2.2 三次自然样条插值

共线点删除和插值的结合使用主要是为了得到一条光滑的曲线，避免出现过多的折线段，同时保留足够的参考点来准确描述车辆行驶的轨迹和姿态。共线点删除可以去除掉一些冗余的点，从而简化数据，降低存储和计算的开销。但是同时共线点删除也会引入一些问题，比如参考线上点的采样间隔不均匀和过长。插值则可以通过计算相邻点之间的曲线来填补被删除的点，得到一条连

续、光滑的曲线，更好地描述车辆的行驶轨迹。因此，本文在设计自动驾驶的参考线建模时，会先进行共线点删除，然后再对剩余的点进行插值，得到一条连续、光滑的参考线。

本文选择三次样条插值是因为该模型具有以下优点：

(1) 光滑性：三次自然样条插值曲线能够保证插值曲线的一阶和二阶导数在控制点处连续，因此插值曲线非常光滑，使得车辆行驶更加平稳，降低驾驶员和乘客的不适感。

(2) 可控性：通过调整控制点位置，可以控制插值曲线的形状，使得它能够更好地适应道路的曲率变化，从而提高行驶的安全性。

(3) 唯一性：三次自然样条插值曲线是唯一的，即在给定的一组控制点下，只有一条三次自然样条插值曲线，因此能够确保生成的参考线唯一性。

(4) 计算效率高：三次自然样条插值曲线的计算速度比较快，能够在实时性要求较高的自动驾驶系统中得到广泛应用。

将删除共线点后的曲线中的点定义为控制点，用参数化方程如下表示：

$$\begin{cases} x_c = x_c(s) \\ y_c = y_c(s) \end{cases} \quad (3-7)$$

其中 $s = \int_0^x \sqrt{1 + y'^2(x)} dx$ ，为曲线的弧长。 x_c 、 y_c 为控制点在全局坐标系中的坐标。

参考线模型用二维三次自然样条可以如下表示：

$$x_r(s) = \begin{bmatrix} x_r^0(s), s \in [s_0, s_1] \\ x_r^1(s), s \in [s_1, s_2] \\ \dots \\ \dots \\ x_r^{k-1}(s), s \in [s_{k-1}, s_k] \end{bmatrix}, y_r(s) = \begin{bmatrix} y_r^0(s), s \in [s_0, s_1] \\ y_r^1(s), s \in [s_1, s_2] \\ \dots \\ \dots \\ y_r^{k-1}(s), s \in [s_{k-1}, s_k] \end{bmatrix} \quad (3-8)$$

其中 k 为控制点的数量， x_r 、 y_r 为参考线在全局坐标系中的坐标。 $x_r^0 \dots x_r^{k-1}$ 、 $y_r^0 \dots y_r^{k-1}$ 为每个区间上的多项式。下面以 $x_r(s)$ 为例，求解 $x_r^0 \dots x_r^{k-1}$ 。取每个区间上的多项式的次数为 3，则有：

$$\begin{bmatrix} x_r^0(s) \\ x_r^1(s) \\ \dots \\ \dots \\ x_r^{k-1}(s) \end{bmatrix} = \begin{bmatrix} a_0s^3 + b_0s^2 + c_0s + d_0 \\ a_1s^3 + b_1s^2 + c_1s + d_1 \\ \dots \\ \dots \\ a_{k-1}s^3 + b_{k-1}s^2 + c_{k-1}s + d_{k-1} \end{bmatrix} \quad (3-9)$$

式中 $a_0 \dots a_{k-1}$ 、 $b \dots b_{k-1}$ 、 $c_0 \dots c_{k-1}$ 、 $d_0 \dots d_{k-1}$ 为多项式的系数，满足：

$$\left. \begin{aligned} x_r^{i-1}(s_i) &= x_r^i(s_i) \\ x_r'^{i-1}(s_i) &= x_r'^i(s_i) \\ x_r''^{i-1}(s_i) &= x_r''^i(s_i) \end{aligned} \right\}, i = 1, 2, \dots, k-1 \quad (3-10)$$

由式中的控制点要求系数需要满足：

$$\begin{aligned} x_r^0(s_0) &= x_r^0 \\ x_r^1(s_1) &= x_r^1 \\ \dots \\ x_r^n(s_k) &= x_r^k \end{aligned} \quad (3-11)$$

总共有 $4n$ 个系数需要求解。式 1 包含了了 $3n - 3$ 个方程，式 2 包含了了 $n + 1$ 个方程，共 $4n - 2$ 个方程。还需要两个等式才能使方程组具有唯一解。因此令

$$x_r''(s_0) = x_r''(s_k) = 0 \quad (3-12)$$

由于 $x_r^i(s_i) = a_i s^3 + b_i s^2 + c_i s + d_i, i = 1, 2, \dots, k$ ，对三次样条进行两次微分可得到线性样条。下面由直线的两点式方程可以求解其表达式：

$$\frac{x_r''^i(s) - x_r''^i}{s - s_i} = \frac{x_r''^i - x_r''^{i-1}}{s_i - s_{i-1}} \quad (3-13)$$

化简得：

$$x_r''^i(s) = x_r''^i \frac{s - s_{i-1}}{s_i - s_{i-1}} + x_r''^{i-1} \frac{s - s_i}{s_{i-1} - s_i} \quad (3-14)$$

令 $h_i = s_i - s_{i-1}$ 得：

$$x_r''^i(s) = x_r''^i \frac{s - s_{i-1}}{h} + x_r''^{i-1} \frac{s_i - s}{h} \quad (3-15)$$

将上式积分两次获得三次样条：

$$x_r^i(s) = \frac{x_r''^i}{6h_i} (s - s_{i-1})^3 + \frac{x_r''^{i-1}}{6h_i} (s_i - s)^3 + C_i (s - s_{i-1}) + D_i (s_i - s) \quad (3-16)$$

其中

$$\begin{aligned} C_i &= \frac{x_r^i}{h_i} - \frac{s_i h_i}{6} \\ D_i &= \frac{x_r^{i-1}}{h_i} - \frac{x_r''^{i-1} h_i}{6} \end{aligned} \quad (3-17)$$

“震荡”或“振铃”现象，从而提高曲线的实用性。

3.3 风险势场建模

接下来进行环境中碰撞风险的量化，定义某种表示风险的势场是一种量化环境风险的方法，它将环境中的障碍物看作是产生风险的源头，将每个障碍物都看作是一个产生势场的点源，从而构建了整个环境的势场模型。在这个模型中，每个点的势能表示了该点的安全级别，即该点到最近障碍物的距离越近，则势能越高，风险越大。

风险势场的优点是它能够有效地避免碰撞，并且可以通过势场的梯度方向快速规划路径。由于障碍物和车辆位置的变化会导致势场的变化，该方法可以自适应地适应不同的环境。然而，传统风险势场方法也存在一些缺点。如果使用势场的梯度变化直接指导规划，环境微小的变化可能导致期望的轨迹频繁改变。除此以外在复杂的环境中可能会存在局部最小值问题，导致规划路径质量下降。针对这些问题，本文用风险等级划分的方法代替梯度法，并且在规划器的设计中降低了对噪声的敏感程度。

3.3.1 风险势场函数

在道路空间中，其他交通参与者的行为可预测性也是比较强的，因此建立的风险势场函数需要能够反应其运动状态的影响，同时能够根据障碍物的形状变化。设计一个满足该要求的势场函数并进行充分的合理性验证，已经超出了本文的工作内容。已经有一些工作进行了风险量化方法的设计，并且得到了许多规划方法的采用。本文参考了 Pierson 所设计的风险势场函数^[50]量化空间中的障碍物风险：

$$H(\mathbf{q}, \mathbf{p}, \dot{\mathbf{p}}) = \sum_{i=1}^n \frac{e^{-(\mathbf{q}-\mathbf{p}_i)^T \Omega (\mathbf{q}-\mathbf{p}_i)}}{1 + e^{-\dot{\mathbf{p}}_i^T (\mathbf{q}-\mathbf{p}_i)}} \quad (3-19)$$

$$\Omega = \text{diag} \left\{ \frac{1}{\sigma_x^2}, \frac{1}{\sigma_y^2} \right\}$$

其中 H 为风险势场函数，整个环境的风险势场由自车附近所有障碍物的风险势场叠加而成。风险势场函数能适用于二维空间或高维空间，在二维空间中，

向量 q 、 p 和 \dot{p} 分别表示了自车与障碍物的位置，以及障碍物的速度。 σ_x 和 σ_y 分别为障碍物的长度和宽度尺寸信息。

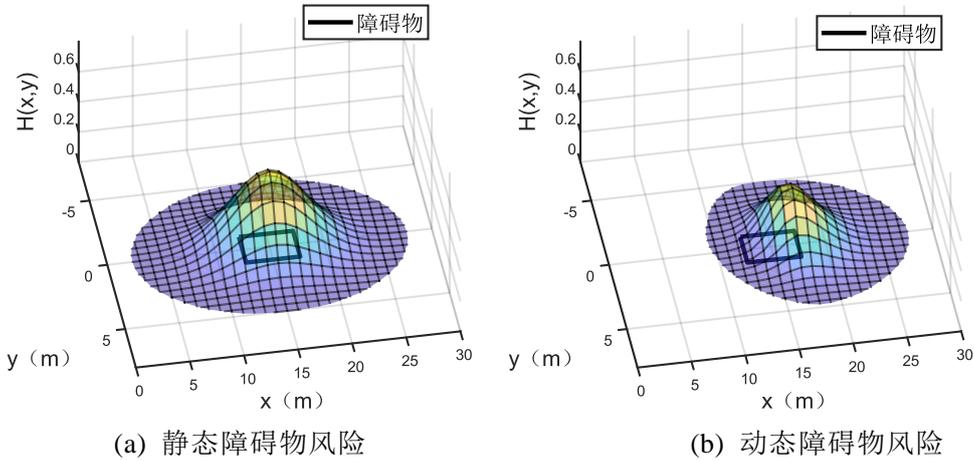


图3-3 风险势场函数

风险势场函数有两个显著的特点，可以通过下图进行更直观的展示。首先，障碍物的大小对于风险势场的形状具有重要的影响。当障碍物的 x 方向长度增加时，风险值也随之增加，并且在对应的方向上风险值更高。这说明了障碍物的大小和位置对于风险势场的形成和分布具有显著的影响。其次，障碍物的速度也会对于风险势场产生影响。在图 3-3 中可以看到，动态障碍物在速度方向上会导致风险值上升，而在速度反方向上则会导致风险值下降。因此，这也会导致风险势场的形状发生变化。这个特点说明了障碍物的运动状态对于风险势场的分布也具有非常重要的影响。

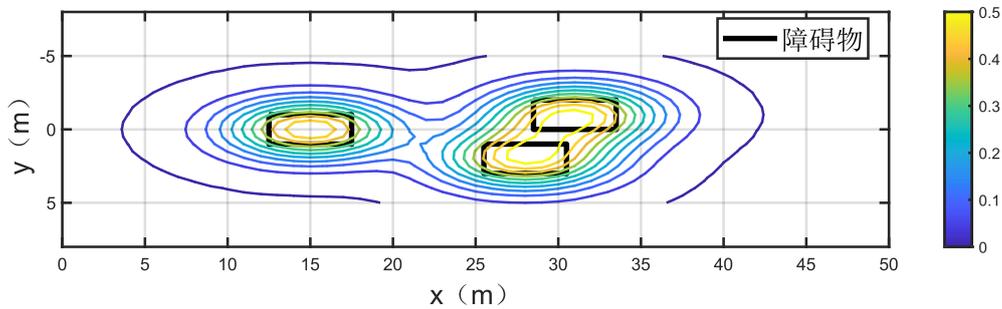


图3-4 多障碍物风险势场叠加

3.3.2 风险等级集

下面为了简化风险势场对规划器的输入形式，本小节定义了用于避障和指导规划的各种风险阈值，并根据风险阈值将风险势场转化为了风险等级集。

其中 H_p 作为规划器的规划阈值。由于风险势场是随着当前位置到障碍物的距离递减的，因此当 H 小于某个值的时候，说明对应的位置已经超出了传感器的感知范围或者规划器的规划范围。因此定义规划阈值 H_p ，对于与规划器来说只把 $H > H_p$ 的风险势场作为输入，减少了风险势场的相关计算量，提高了规划算法的效率。

考虑到系统的动力学，我们将 $H_t > H_p$ 定义为制动阈值，如果风险势场保持在此阈值范围内，自动驾驶汽车始终具有足够的停车距离以避免碰撞。当风险大于该阈值时，车辆应该采取更加谨慎的动作，因为当前状态可能已经不能只通过制动来防止与障碍物相碰撞。这个值可以根据当前的车辆速度和障碍物的距离进行调整。

碰撞阈值 $H_c > H_t$ 是用来定义碰撞参数。当风险势场高于该值时，车辆可能会遭受碰撞。在机器人规划算法中，碰撞阈值通常被设置为自车的安全半径，本文根据自动驾驶的场景进行了设计。

3.3.2.1 碰撞阈值

本文将通过单个障碍物定义阈值，然后推广到多障碍物的情景中，在单障碍物情景下：

$$H(\mathbf{d}, \dot{\mathbf{p}}_i) = \frac{e^{-\mathbf{d}^T \Omega \mathbf{d}}}{1 + e^{-\dot{\mathbf{p}}_i^T \mathbf{d}}} \quad (3-20)$$

其中 $\mathbf{d} = (\mathbf{p} - \mathbf{p}_i)$ 为障碍物的车辆坐标系下的位置坐标。为了计算碰撞前的最大风险 H_c ，假设碰撞半径为 r_c ，单位向量 $\mathbf{u} = \frac{\mathbf{d}}{\|\mathbf{d}\|}$ ，则满足 $\|\mathbf{d}\| = r_c$ 的风险势场可以如下表示：

$$H(r_c \mathbf{u}, \dot{\mathbf{p}}_i) = \frac{e^{-r_c^2 \mathbf{u}^T \Omega \mathbf{u}}}{1 + e^{-\alpha r_c \dot{\mathbf{p}}_i^T \mathbf{u}}} \quad (3-21)$$

其中 $\mathbf{u}^T \Omega \mathbf{u}$ 的值由单位向量的 \mathbf{u} 方向确定。定义 $\frac{1}{\sigma_m^2} = \min\left\{\frac{1}{\sigma_x^2}, \frac{1}{\sigma_y^2}\right\}$ ，则有：

$$H(r_c \mathbf{u}, \dot{\mathbf{p}}_i) \leq \frac{e^{-\frac{r_c^2}{\sigma_m^2}}}{1 + e^{-ar_c v_i}} \quad (3-22)$$

因此定义碰撞阈值为：

$$H_c = \frac{e^{-\frac{r_c^2}{\sigma_m^2}}}{1 + e^{-ar_c v_i}} \quad (3-23)$$

3.3.2.2 制动阈值

与碰撞阈值类似，假设自车到第*i*个障碍物的制动距离为 $r_{b,i}$ ，则第*i*个障碍物的制动阈值定义如下：

$$H(r_{b,i} \mathbf{u}, \dot{\mathbf{p}}_i) = \frac{e^{-r_{b,i}^2 \mathbf{u}^T \Omega \mathbf{u}}}{1 + e^{-ar_{b,i} \dot{\mathbf{p}}_i^T \mathbf{u}}} \leq \frac{e^{-\frac{r_{b,i}^2}{\sigma_m^2}}}{1 + e^{-ar_{b,i} v_i}} \leq \frac{e^{-\frac{r_{b,i}^2}{\sigma_m^2}}}{2} = H_t \quad (3-24)$$

需要注意的是， $r_{b,i}$ 是针对于每个障碍物定义的，在取其上限时风险需要高于相对各个最短制动距离定义的阈值。

3.3.2.3 规划阈值

为了满足 $H_t > H_p$ ，通过截断 H_t 的级数展开来定义规划阈值 H_p ，以产生一个较低和合理规划范围风险估计，如下所示：

$$H_p = \sum_{i=1}^6 e^{-\frac{r_{b,i}^2}{\sigma_m^2}} \quad (3-25)$$

3.3.2.4 风险等级集

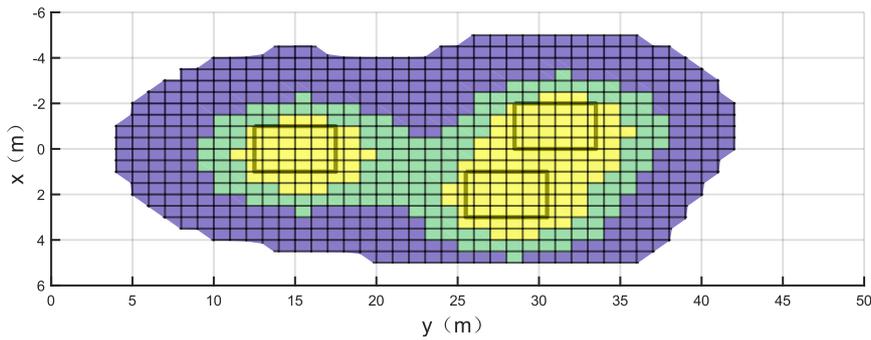


图3-5 风险等级集

通过上述风险阈值的定义，风险势场模型被划分成了三个风险等级集，如上图所示。根据风险由低到高，分别称为规划集、危险集和碰撞集。使用风险等级集来指导规划算法，相较于直接使用风险势场函数，计算效率更高。这种方法为自动驾驶车辆的行驶安全提供了一种有效的解决方案，并且更接近人类

驾驶员的行为。

3.4 Frenet 最优规划器

本文设计的考虑动力学约束的 Frenet 最优规划器模型相较于传统的 Frenet 最优规划器模型，在规划最优轨迹时，结合了 Milliken 动力学模型，考虑了最优轨迹的动力学安全性。传统的 Frenet 最优规划器模型通常只考虑了运动学约束和路径规划，没有考虑车辆动力学特性和安全性等因素。

相较于传统的 Frenet 最优规划器模型，本文提出的模型可以更好地满足实际道路的行驶要求，能够避免由于车速过快、弯道过急等原因造成的车辆失控和行驶不稳定等问题。

3.4.1 Frenet 坐标转换

Frenet 坐标系可以将横向、纵向、时间三个维度的优化问题解耦为两个二维空间的规划问题^[6]，如图 3-6 所示，降低了规划问题的维度。两个空间分别为横向的 SL 空间和纵向的 ST 空间，本文将分别在这两个空间进行路径规划和速度规划。

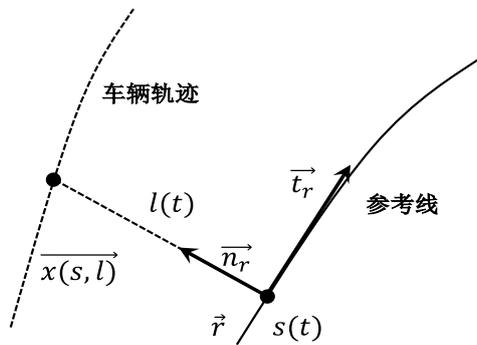


图3-6 Frenet 坐标转换

地图坐标系下的坐标 \vec{x} 可由 Frenet 坐标系 $s(t)$ 和 $l(t)$ 如下表示：

$$\vec{x}(s(t), l(t)) = \vec{r} + l(t)\vec{n}_r(s(t)) \quad (3-26)$$

其中 $s(t)$ 表示自动驾驶汽车沿参考线里程， $l(t)$ 表示对于参考线的侧向偏差，也经常表示为 $l(s)$ 。

选取地图坐标系中的状态变量为 $[\vec{x}, \theta, \kappa, v, a]$ ，选取 Frenet 坐标系中的状态

变量为 $[s, \dot{s}, \ddot{s}; l, l', l'']$ 。其中 \vec{x} 是地图坐标系下的坐标，是一个向量； θ 是在地图坐标下的方向； $\kappa = \frac{d\theta}{ds}$ 是在地图坐标下的曲率； v, a 分别为在地图坐标下的速度和加速度。 \dot{s}, \ddot{s} 分别为纵向坐标对时间的一阶导数和二阶导数，也就是纵向速度和加速度； l', l'' 分别为横向坐标对纵向坐标的一阶导数和二阶导数。

根据 Wang 的推导^[8]，Frenet-笛卡尔坐标转换如下：

$$\begin{aligned} \overrightarrow{x(s(t), l(t))} &= \vec{r} + l(t)\vec{n}_r(s(t)) \\ v &= \dot{s} \frac{1 - \kappa_r l}{\cos \Delta\theta} \\ \theta &= \theta_r + \arctan \frac{l'}{1 - \kappa_r l} \\ a &= \ddot{s} \frac{1 - \kappa_r l}{\cos(\Delta\theta)} + \frac{\dot{s}^2}{\cos(\Delta\theta)} (\tan(\Delta\theta) \Delta\theta' (1 - \kappa_r l) - (\kappa_r' l + \kappa_r l')) \\ \kappa &= \frac{\cos^3(\Delta\theta) (d'' + (\kappa_r' l + \kappa_r l')) + \tan(\Delta\theta)}{(1 - \kappa_r l)^2} + \frac{\kappa_r \cos \Delta\theta}{1 - \kappa_r l} \end{aligned} \quad (3-27)$$

笛卡尔-Frenet 坐标转换如下：

$$\begin{aligned} \dot{s} &= \frac{v_x \cos \Delta\theta}{1 - \kappa_r l} \\ \ddot{s} &= \frac{\cos \Delta\theta}{1 - \kappa_r l} \left[a - \frac{\dot{s}}{\cos \Delta\theta} (\tan \Delta\theta) \Delta\theta' (1 - \kappa_r l) - (\kappa_r' l + \kappa_r l') \right] \\ l &= [\vec{x} - \vec{r}]^T \vec{n}_r \\ l' &= [1 - \kappa_r d] \tan \Delta\theta \\ l'' &= -[\kappa_r' l + \kappa_r l'] \tan \Delta\theta + \frac{1 - \kappa_r l}{\cos^2 \Delta\theta} \left[\kappa \frac{1 - \kappa_r l}{\cos \Delta\theta} - \kappa_r \right] \end{aligned} \quad (3-28)$$

其中 κ_r 和 κ 分别为参考线的曲率和曲率变化率， θ_r 为参考线的切线方向， \vec{x} 、 \vec{r} 和 \vec{n}_r 分别为状态点向量、映射点向量和映射点法向量。

3.4.2 轨迹生成

3.4.2.1 纵向轨迹生成

针对道路空间场景的纵向轨迹生成将道路空间的驾驶场景分为三类场景考虑，包括巡航场景、跟车场景和超车场景。针对不同的场景，生成纵向轨迹时将选择三种不同的纵向状态。本文通过采样确定终端车辆状态，通过状态关于时间的多项式曲线表示纵向轨迹。在 Frenet 坐标系中的状态变量 $[s, \dot{s}, \ddot{s}; l, l', l'']$ 中， $[s, \dot{s}, \ddot{s}]$ 代表了纵向车辆状态，而终端纵向车辆状态用 $[s_f, \dot{s}_f, \ddot{s}_f]$ 。

首先分析巡航场景，此时由于车辆前方无障碍物，车辆的行驶目标为按照巡航速度进行车道保持。因此状态采样需要满足终端纵向速度 \dot{s}_f 与设定速度接近。为满足生成轨迹的舒适性，使终端加速度 \ddot{s}_f 为0。巡航场景的状态采样如下所示：

$$\begin{aligned} \mathbf{s}_c &= [s_f, \dot{s}_f, \ddot{s}_f] \\ \dot{s}_f &= \dot{s}_d + k_1 \cdot \Delta \dot{s}_f \\ \ddot{s}_f &= 0 \\ t_f &= k_2 \cdot \Delta t \end{aligned} \quad (3-29)$$

其中 \mathbf{s}_c 为巡航场景的终端纵向车辆状态， \dot{s}_d 为纵向速度采样间隔， Δt 为时间采样间隔。在此场景下，规划器可以确定三个初始状态和两个终端状态，因此可以使用四次多项式对纵向轨迹进行描述。上式中的 k_1 和 k_2 为采样系数，取 $k_1=1、0、1$ ， $k_2=2、3、4$ ， $\Delta \dot{s}_f = \Delta t = 1$ ，规划初始纵向速度为10m/s，目标纵向速度为15m/s，则采样时间为3s、4s、5s，采样速度为14s、15s、16s。巡航场景的终端纵向车辆状态如下图所示：

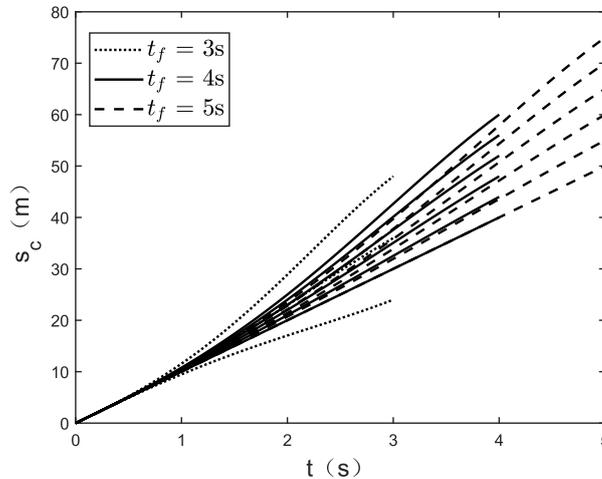


图3-7 巡航场景速度规划

超车场景与跟车场景涉及到与障碍车（或称为目标车）的交互。在超车场景中，需要将车辆的终端纵向位移设置于障碍物前方，以完成超车动作，终端速度可设为与障碍物速度相等，终端加速度设为0，如下所示：

$$\begin{aligned}
 \mathbf{s}_o &= [s_o, \dot{s}_o, \ddot{s}_o] \\
 s_o &= s_l + k_3 \cdot \Delta s \\
 \dot{s}_o &= v_l \\
 \ddot{s}_o &= 0 \\
 t_f &= k_2 \cdot \Delta t
 \end{aligned} \tag{3-30}$$

其中 \mathbf{s}_l 为障碍车的终端纵向车辆状态， Δs 为纵向距离采样间隔， Δt 为时间采样间隔。在此场景下，规划器可以确定三个初始状态和三个终端状态，因此使用五次多项式对纵向轨迹进行描述。上式中的 k_2 和 k_3 为采样系数，取 $k_1=-1$ 、 0 、 1 ， $k_2=3$ 、 4 、 5 ， $\Delta s = 2$ 、 $\Delta t = 0.5$ ，规划初始纵向速度为 10m/s ，目标纵向速度为 15m/s ，则采样时间为 3s 、 4s 、 5s ，采样速度为 15m/s ，采样位置为障碍车前方 3m 、 5m 、 7m 处。超车场景的终端纵向车辆状态如下图所示：

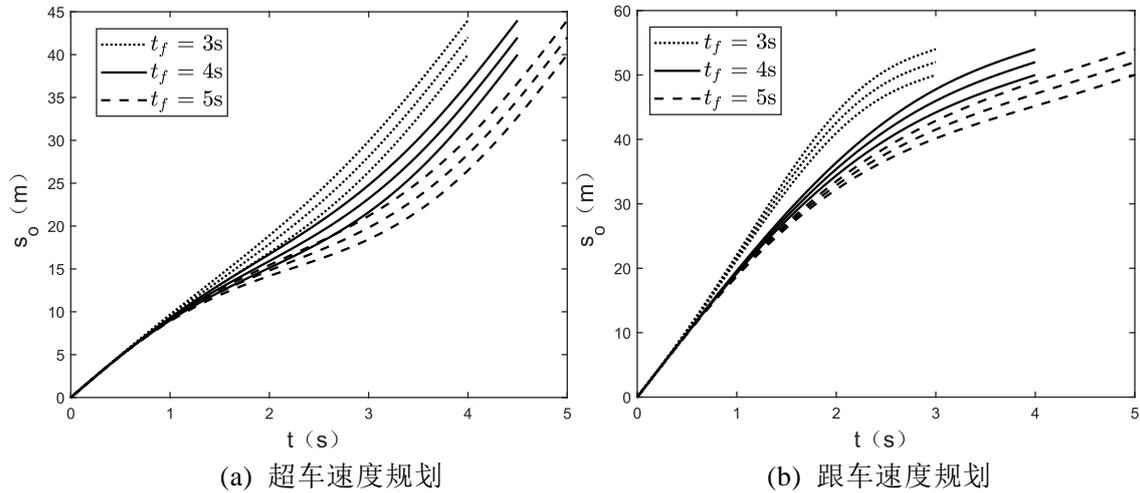


图3-8 变道场景速度规划

跟车场景与超车场景类似，终端速度可设为与障碍物速度相等，终端加速度设为 0 。区别是跟车场景需要将车辆的终端纵向位移设置于障碍物前后方。跟车场景的终端纵向车辆状态如图 3-8(b)所示。

在上述两个场景中，纵向轨迹的五次多项式如下表示：

$$s(t) = a_5 t^5 + a_4 t^4 + a_3 t^3 + a_2 t^2 + a_1 t + a_0 \tag{3-31}$$

其中 $a_5 \dots a_1$ 为多项式的系数，可以通过如下线性方程组求解：

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5 \\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^3 & 5t_f^4 \\ 0 & 0 & 2 & 6t_f & 12t_f^2 & 20t_f^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} s_0 \\ \dot{s}_0 \\ \ddot{s}_0 \\ s_f \\ \dot{s}_f \\ \ddot{s}_f \end{bmatrix} \quad (3-32)$$

巡航场景下的四次轨迹多项式可用同样的方法获取：

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 \\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} s_0 \\ \dot{s}_0 \\ \ddot{s}_0 \\ s_f \\ \dot{s}_f \end{bmatrix} \quad (3-33)$$

3.4.2.2 横向轨迹生成

与纵向轨迹相似，横向轨迹同样使用五次多项式表示。车辆的横向行为比纵向行为更加复杂，纵向行为可以根据上一小节中的三种场景进行划分，而不同场景中横向的采样行为需要更加密集才能满足自动驾驶的需求。为了使车辆能够沿参考线平行方向，需要将车辆的终端横向速度和加速度设为 0，而对横向位置进行采样，如下所示：

$$\begin{aligned} \mathbf{l} &= [l, \dot{l}, \ddot{l}] \\ l &= l_f + k_4 \cdot \Delta l \\ \dot{l} &= 0 \\ \ddot{l} &= 0 \\ s_f &= k_5 \cdot \Delta s \end{aligned} \quad (3-34)$$

其中 \mathbf{l} 为障碍车的终端横向车辆状态， l 为横向位置采样， Δl 为横向位置采样间隔， Δt 为时间采样间隔。上式中的 k_2 和 k_3 为采样系数，取 $k_4 = -8, -7.9, \dots, 7.9, 8$ ， $k_5 = 7, 8, 9$ ， $\Delta l = 1$ 、 $\Delta t = 5$ ，则采样纵向距离为 35m、40m、45m。多项式参数可以通过求解如下线性方程组获取：

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 1 & s_f & s_f^2 & s_f^3 & s_f^4 & s_f^5 \\ 0 & 1 & 2s_f & 3s_f^2 & 4s_f^3 & 5s_f^4 \\ 0 & 0 & 2 & 6s_f & 12s_f^2 & 20s_f^3 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} l_0 \\ \dot{l}_0 \\ \ddot{l}_0 \\ l_f \\ \dot{l}_f \\ \ddot{l}_f \end{bmatrix} \quad (3-35)$$

采样终端横向车辆状态生成的横向轨迹如图 3-9 所示：

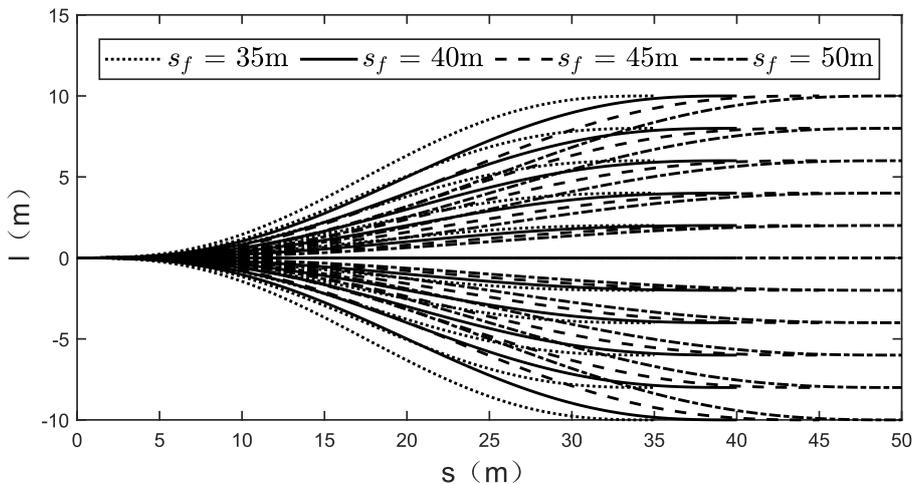


图3-9 横向轨迹规划

将 $s(t)$ 和 $l(s)$ 结合参考线模型，进行 Frenet 坐标系到地图坐标系的变换，对于每一组参数方程 $s(t)$ 和 $l(s)$ 可以得到对应的车辆轨迹，如图 3-10 所示：

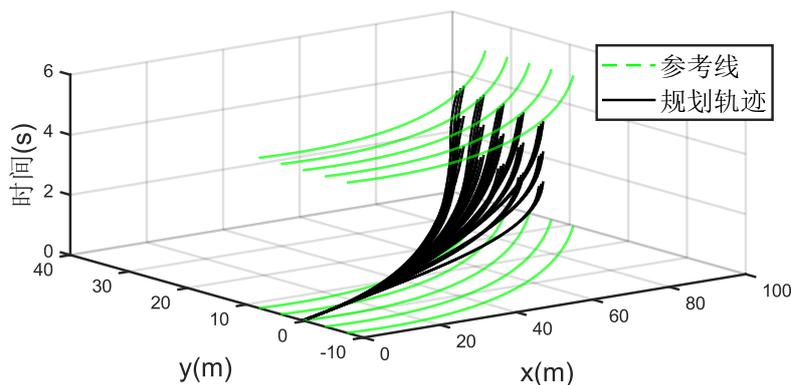


图3-10 笛卡尔空间规划轨迹

3.4.3 代价函数设计

在运动规划中，代价函数是用来评估不同轨迹的好坏程度的函数。代价函数将路径映射到一个实数值，用于比较不同路径之间的优劣，并最终选择最优路径。代价函数在运动规划中起着关键作用，设计合理的代价函数能够帮助优化路径规划，提高运动规划算法的效率和性能。

本文设计的道路空间规划器代价函数由四部分构成，如下所示：

$$C_r = C_{lat} + C_{lon} + C_{obs} + C_{mil} \quad (3-36)$$

其中 C_r 为道路空间规划器代价函数， C_{lat} 为侧向代价函数， C_{lon} 为纵向代价

函数， C_{obs} 为避障代价函数， C_{mil} 为动力学代价函数。

3.4.3.1 侧向代价

自动驾驶运动规划中的侧向代价函数用于评估车辆与参考路径的偏离程度，即侧向偏差。在设计侧向代价函数时，一般需要考虑的因素包括：车辆在侧向上与参考路径之间的距离，偏离距离越大，代价函数的值越大；车辆与参考路径的相对角度，表示车辆当前的行驶方向与参考路径的切线方向之间的夹角，如果车辆偏离参考路径的角度越大，则代价函数的值越大。车辆侧向位移的一阶、二阶、三阶导数，其绝对值累加值越大，说明车辆采取的侧向动作越剧烈，轨迹的舒适性会随之下降，则代价函数的值越大。

本文的侧向代价函数设计如下：

$$C_{lat} = C_{lat,d} + C_{lat,v} + C_{lat,a} + C_{lat,j} \quad (3-37)$$

其中 C_{lat} 为侧向代价函数， $C_{lat,d}$ 为侧向偏差代价， $C_{lat,v}$ 为侧向速度代价， $C_{lat,a}$ 为侧向加速度代价， $C_{lat,j}$ 为侧向加加速度代价。侧向代价函数的各个部分计算如下：

$$\begin{aligned} C_{lat,d} &= w_{lat,d} \int_0^{s_f} l(s)^2 s \\ C_{lat,v} &= w_{lat,v} \int_0^{s_f} \dot{l}(s)^2 s \\ C_{lat,a} &= w_{lat,a} \int_0^{s_f} \ddot{l}(s)^2 s \\ C_{lat,j} &= w_{lat,j} \int_0^{s_f} \ddot{\ddot{l}}(s)^2 s \end{aligned} \quad (3-38)$$

其中 $w_{lat,d}$ 、 $w_{lat,v}$ 、 $w_{lat,a}$ 、 $w_{lat,j}$ 为各项代价的权重系数。

3.4.3.2 纵向代价

在自动驾驶运动规划中，纵向代价函数的设计需要考虑以下因素：车辆终端速度的偏差，表示规划轨迹在末端状态下的速度与期望速度之间的差距，速度偏差越大，代价函数的值越大。车辆纵向位移的二阶、三阶导数，其绝对值累加值越大，说明车辆采取的纵向加减速动作越剧烈，轨迹的舒适性会随之下降，则代价函数的值越大。车辆完成规划轨迹消耗的时间，表示车辆需要提高行车效率。同时尽快完成规划轨迹也有助于避免违法行驶和提高驾驶安全性。

本文的纵向代价函数设计如下：

$$C_{lon} = C_{lon,v} + C_{lon,a} + C_{lon,j} + C_{lon,t} \quad (3-39)$$

其中 C_{lon} 为纵向代价函数， $C_{lon,v}$ 为纵向速度代价， $C_{lon,a}$ 为纵向加速度代价， $C_{lon,j}$ 为纵向加加速度代价， $C_{lon,t}$ 为纵向时间代价。纵向代价函数的各个部分计算如下：

$$\begin{aligned} C_{lon,v} &= w_{lon,v} (\dot{s}_d - \dot{s}(t_f))^2 \\ C_{lon,a} &= w_{lon,a} \int_0^{t_f} \ddot{s}(t)^2 t \\ C_{lon,j} &= w_{lon,j} \int_0^{t_f} \ddot{\ddot{s}}(t)^2 t \\ C_{lon,t} &= w_{lon,t} t_f \end{aligned} \quad (3-40)$$

其中 $w_{lon,d}$ 、 $w_{lon,v}$ 、 $w_{lon,a}$ 、 $w_{lon,j}$ 为各项代价的权重系数。

3.4.3.3 障碍物代价

在本章的 3.2 中定义了风险势场和风险等级集，在本小节本文将借助上述工具计算障碍物代价。首先在 SL 空间中对 s 进行采样，采样步长选为 20m，并且将采样点进行 Frenet 坐标系到地图坐标系下的转换，如下图所示：

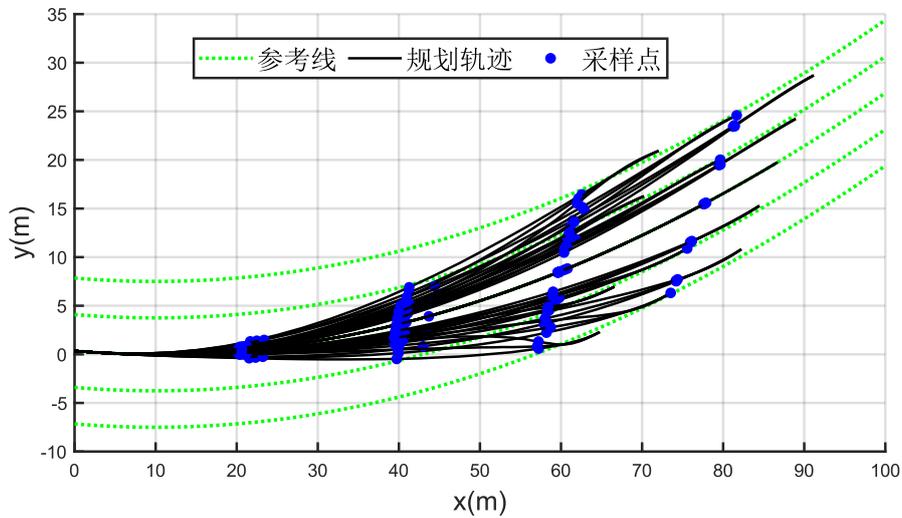


图3-11 障碍物代价采样点

检验每个轨迹上的采样点所处的风险等级集，并根据表 3-1 建立采样点风险等级状态到障碍物代价的映射。当采样点全部没有在规划集内或者有一个以上在碰撞集内，障碍物代价为 3；当采样点全部在规划集内，并且不存在危险集和碰撞集内的采样点时，障碍物代价为 0；当采样点部分在规划集内，部分在危

险集内时，障碍物代价为 1。通过图 3-12 可以看出通过采样点分布确定的不同轨迹的障碍物风险。

表3-1 障碍物代价映射表

规划集	采样点分布个数		障碍物代价 C_{obs}
	危险集	碰撞集	
0	-	-	3
>1	0	0	0
>1	>1	0	1
>1	>1	>1	3

根据上述映射关系，图中的自动驾驶场景中设置障碍物，并且根据风险等级集计算障碍物代价 C_{obs} ，计算结果如下图所示。若最优规划轨迹的 C_{obs} 为 3，则代表本次规划失败，通过交互系统提醒驾驶员接管驾驶。

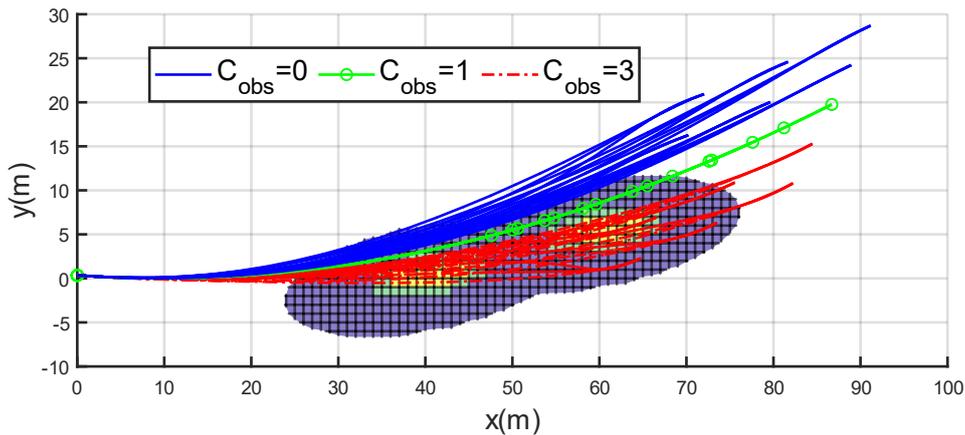


图3-12 轨迹的障碍物代价

3.4.3.4 动力学代价

规划算法的动力学代价 C_{mil} 根据安全距离如下定义：

$$C_{mil} = -w_{mil}D_{mil} \quad (3-40)$$

图 3-13 为轨迹 1 和轨迹 2 的安全距离计算。轨迹 1 选择了较为保守的转弯策略，逐渐逼近参考线，在 Milliken 动力学模型中没有超出安全边界的状态，轨迹安全距离 $D_{mil} = 0.0279$ ；轨迹 2 选择了较为激进的转弯策略，快速逼近参考线，在 Milliken 动力学模型中存在超出安全边界的状态，轨迹安全距离 $D_{mil} = -0.0732$ 。

其中 w_{mil} 为权重系数。

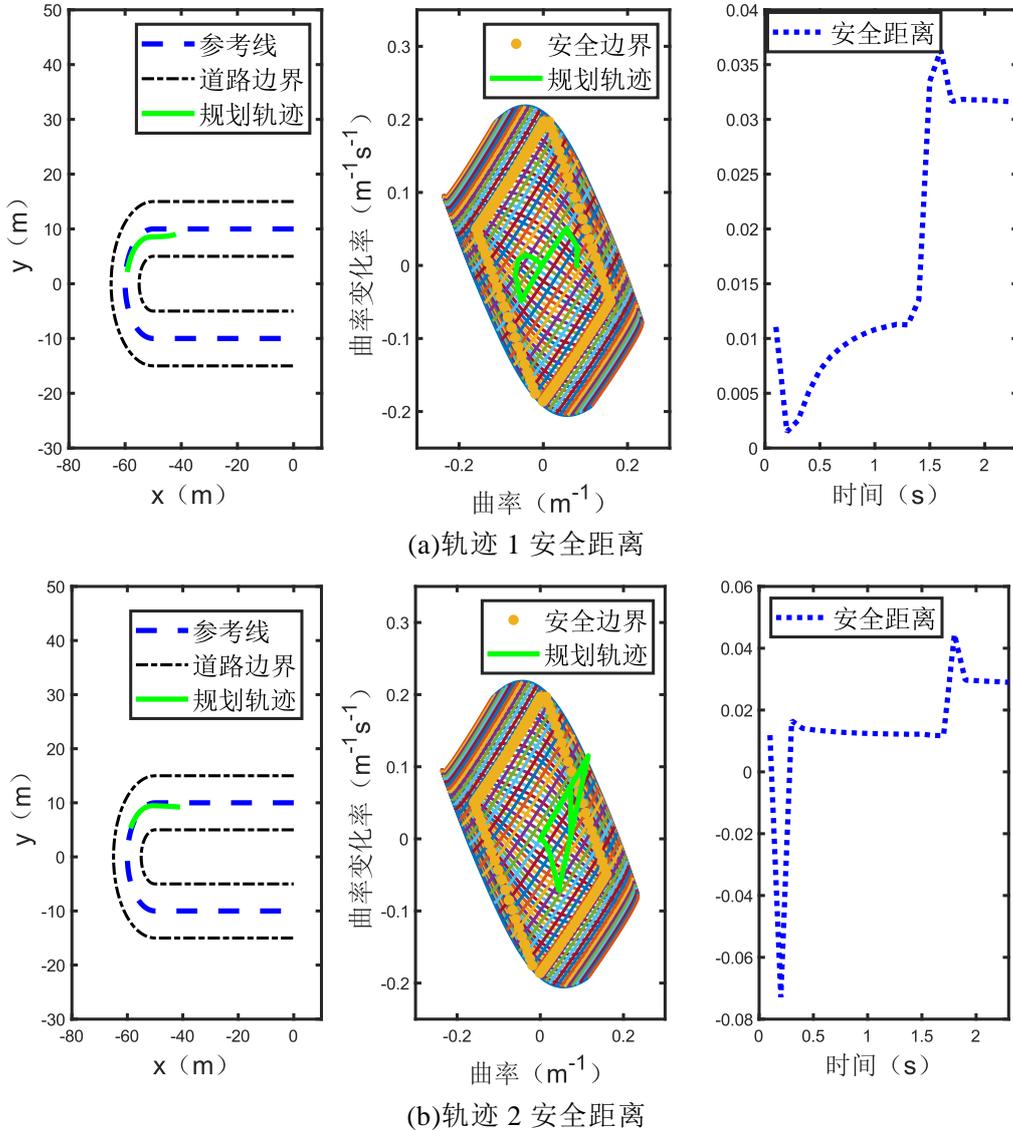


图3-13 轨迹的安全距离

若最优规划轨迹的 D_{mil} 为负值，即障碍物代价 C_{mil} 为正值，则代表规划器产生的最优轨迹仍无法满足动力学安全性，本次规划失败，通过交互系统提醒驾驶员接管驾驶。

3.4.3.5 代价函数权重

代价函数权重作为超参数，需要在实验过程中人工整定。模型的代价函数权重最终整定如下，能使规划算法在后续实验中具有较好的性能指标。由于模型中的代价函数权重较多，因此将代价函数权重中选择了主要权重，包括侧向位置代价权重 $w_{lat,d}$ 、纵向速度代价权重 $w_{lon,v}$ 、动力学代价权重 w_{mil} ，代表了规

划算法的主要目标为避障、回归车道参考线、跟踪巡航速度、满足动力学约束，次要目标为保证各阶导数尽量平滑并使规划轨迹耗时尽可能短。

表3-2 Frenet 最优规划器代价函数权重

权重	权重值	权重	权重值
$w_{lat,d}$	2	$w_{lon,v}$	1
$w_{lat,v}$	0.1	$w_{lon,a}$	0.1
$w_{lat,a}$	0.1	$w_{lon,j}$	0.1
$w_{lat,j}$	0.1	$w_{lon,t}$	0.1
w_{mil}	2		

3.5 道路空间规划器仿真验证

3.5.1 仿真场景选择

对于本章的模型，需要通过实验验证以下内容：

参考线生成算法能够生成平滑、形状连续的参考线，能够精确地逼近原始数据，描述曲线的形状和特征；对于多余的共线车道中心点，算法能够合理的删除一部分点，以降低算法的复杂度；基于参考线的 Frenet 坐标系变换能准确地计算 Frenet 轨迹在地图坐标系下的映射。

风险势场模型可以量化环境中其他交通参与者的风险，从而指导规划器实现避障。根据规划阈值、制动阈值、安全阈值，算法能够准确地划分风险等级集。

考虑动力学约束的 Frenet 最优规划器能包含多种驾驶决策生成横纵向轨迹，并且根据轨迹计算横向代价和纵向代价；根据当前环境的风险等级集和障碍物代价映射表，规划器能计算不同轨迹的障碍物代价；规划器还能根据当前车辆状态调用 Milliken 动力学模型，计算安全边界、安全距离和动力学代价，分析并改进规划轨迹的动力学安全性。

在上文中已经完成了算法各个子模块的验证，下面将针对巡航场景、避障场景和弯道场景，验证本文设计的考虑动力学约束的 Frenet 最优规划器能够在规划中及时发现潜在的动力学风险，并且实现对风险的抑制，同时兼顾避障、跟踪巡航速度、消除侧向偏差等基本目标。

3.5.2 巡航场景验证

在进行复杂的车道变换之前，需要先验证规划器在巡航中的车道保持功能，以确保车辆可以在巡航阶段安全平稳地行驶。因此，首先设置巡航场景对规划器进行验证。在巡航场景中车辆将实现速度保持与车道保持功能。给定车道线和道路边界，在初始状态，车辆的速度为 15m/s ，期望的巡航速度为 15m/s ，路面附着系数为 0.5 ，车辆的初始位置距参考线有 1.5m 的偏移。使用上文所设计的规划器，规划能够使车辆在 13s 内回归到车道中心线的轨迹，并且能够维持在巡航速度。

假设控制器可以完美跟踪规划轨迹，可以得到自动驾驶汽车未来的行驶轨迹。图 3-14 为车辆的规划轨迹，从图中可以看出车辆在行驶约 100m 后回归车道中心线上。

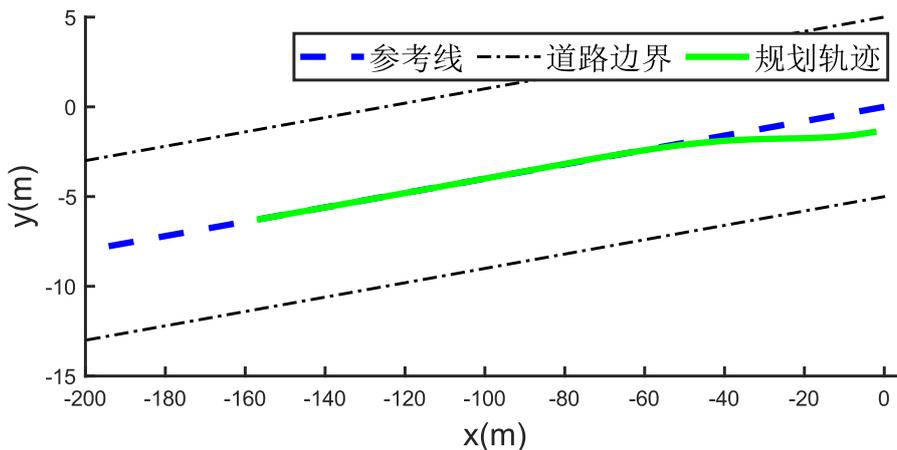


图3-14 巡航场景规划轨迹

图 3-15 为规划轨迹上的车辆状态随时间变化的曲线。车辆在 6s 时就将侧向偏差降到了 0.1m 以下，为了满足舒适性的要求，纵向的加速度和加速度变化率都被限制到较小的范围，因此轨迹状态变化平缓。轨迹的纵向速度可维持在巡航速度。

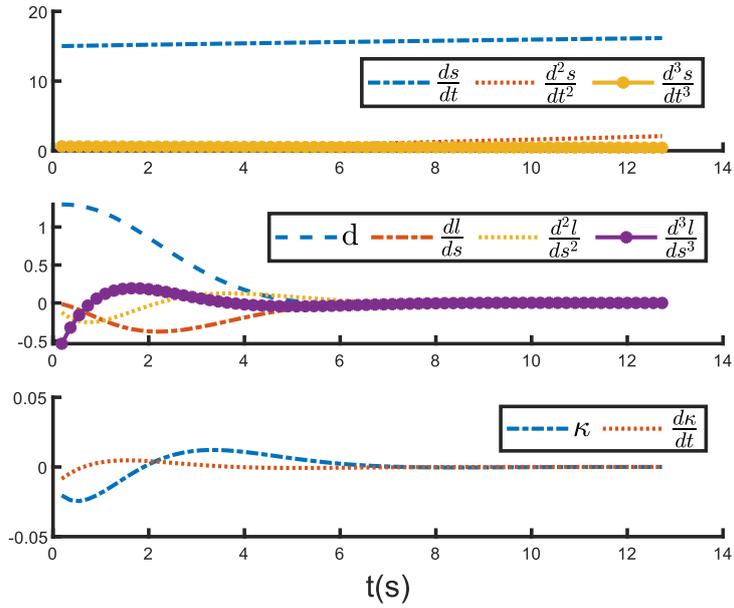


图3-15 巡航场景规划轨迹状态

从图 3-16 中在绿色实线为巡航过程中所有规划帧的最优规划轨迹，红色虚线为假设完美跟踪的车辆行驶轨迹。由于场景的路面附着系数不变，巡航过程中的纵向速度、纵向加速度变化也较小，因此巡航过程中的安全边界接近初始状态车辆安全。可以看出所有规划轨迹和模拟跟踪轨迹都没有超出 Milliken 动力学中的安全边界的状态。

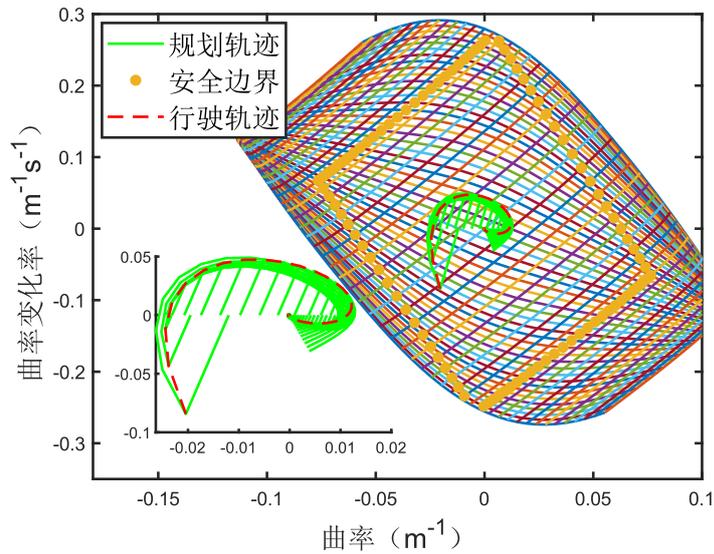


图3-16 巡航场景规划动力学安全性

3.5.3 避障场景验证

在避障场景下给定车道线和道路边界，初始状态车辆速度为 15m/s，期望巡航速度为 15m/s，路面附着系数为 0.5。车辆的初始位置据参考线无偏移。车辆前方设置速度为 10m/s 的障碍物。为模拟真实的障碍物行为，在车辆完成超车动作时，障碍车辆做出轻微的让道动作。使用上文所设计的规划器，能够使车辆在 13s 内避障并回归到车道中心线，同时使车速维持在巡航速度。

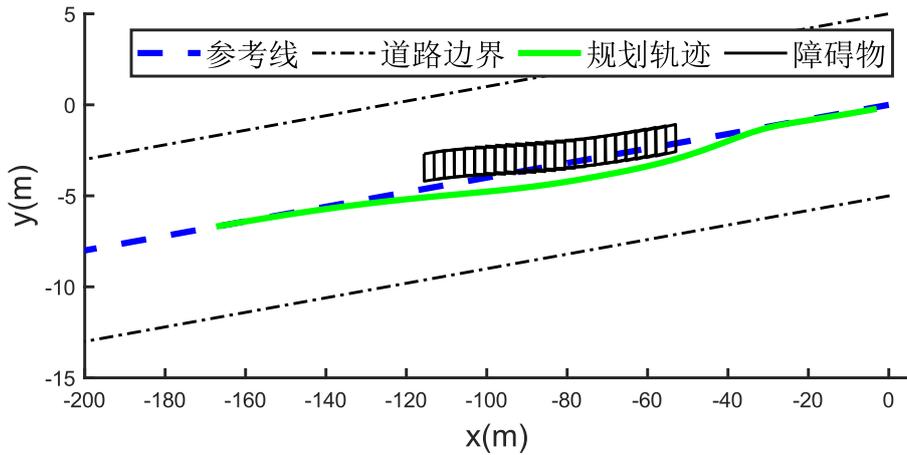


图3-17 避障场景规划轨迹

如图 3-18 所示，避障过程中规划速度维持在设定巡航速度。在开始避障和结束避障时都给出了平滑的期望车辆状态，并且在 11s 后车辆相对于参考线的侧向偏差可以降低到 0.1m 以下。Milliken 动力学中状态如图 3-19 所示。

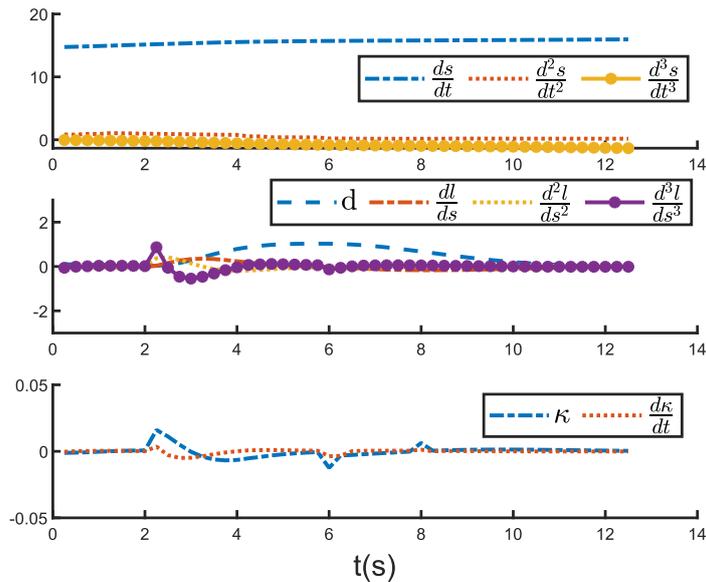


图3-18 避障场景规划轨迹状态

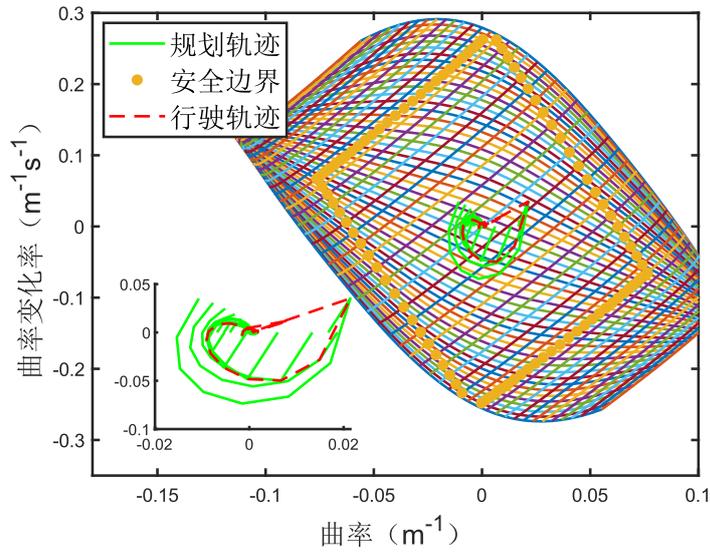


图3-19 避障场景规划动力学安全性

3.5.4 弯道场景验证

车辆初始状态车辆的速度为 15m/s，并以 15m/s 的期望巡航速度通过前方 U 型弯道。车辆的初始位置据参考线有 1.5m 偏移。规划能够使车辆在 10s 内通过弯道并回归到车道中心线。

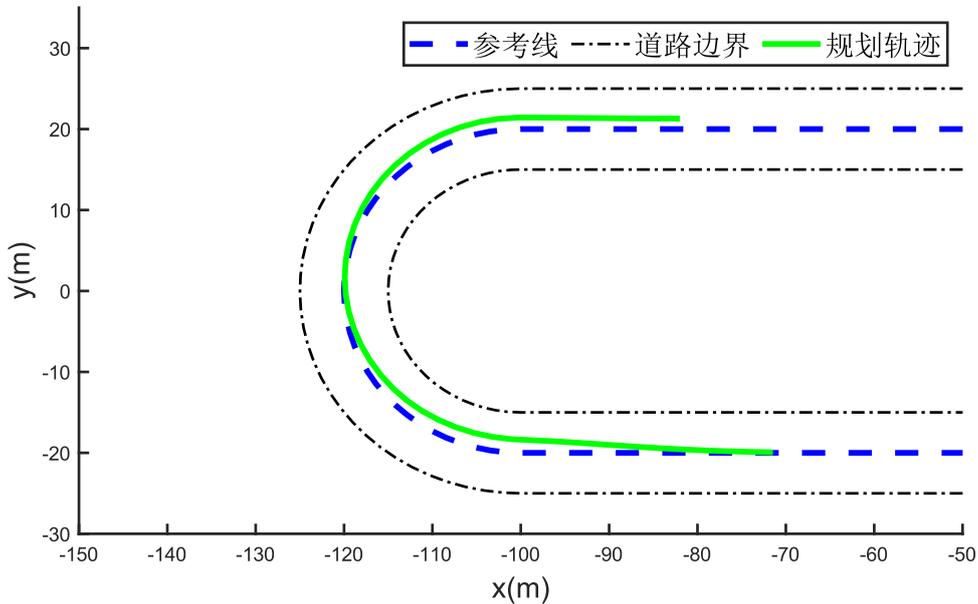


图3-20 弯道场景规划轨迹

规划中的车辆状态如图 3-21 所示，由于弯道参考线本身存在常数曲率，因此车辆在行驶时除自身的驱动力和转向力外，还需要额外提供转弯向心力。如图 3-22，在提供向心力时，轨迹在 Milliken 动力学模型中的状态也存在明显的

从安全区域向安全边界的偏移，但未超出安全边界。

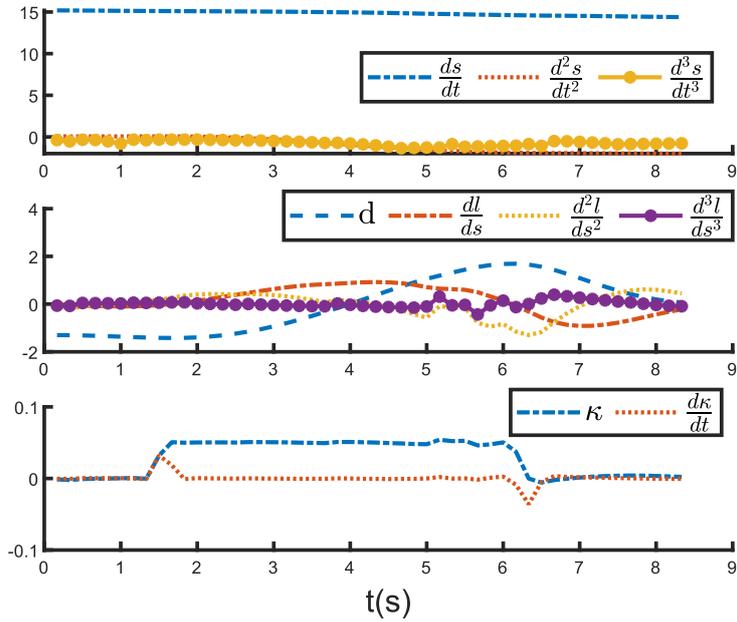


图3-21 弯道场景规划轨迹状态

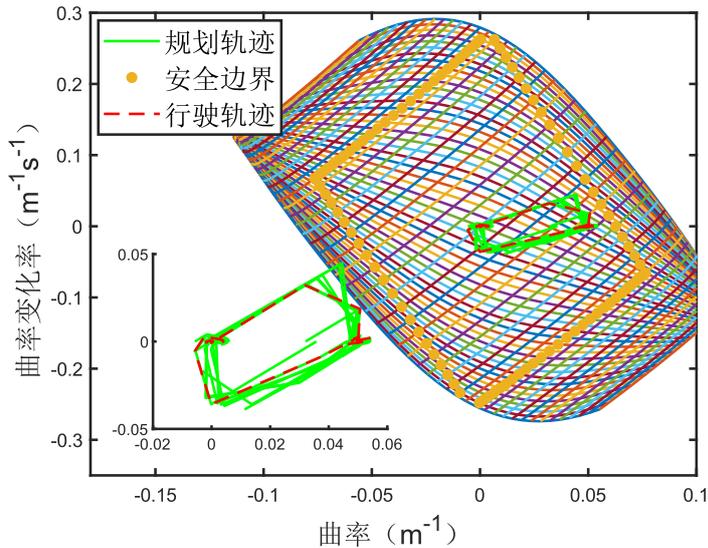


图3-22 弯道场景规划动力学安全性

3.6 本章小结

本章设计了考虑动力学约束的道路空间运动规划。针对有车道线信息的道路空间，本文首先建立了参考线模型，通过共线点删除和三次样条插值，获得平滑的规划参考线。其次，本文建立了风险势场模型，来描述道路空间中形状于运动较为规则的障碍物；最后，本文设计了基于 Frenet 坐标系的最优规划算法，在代价函数中定义与安全边界相关的动力学代价，从而提高规划轨迹的安

全性。通过巡航场景、避障场景、弯道场景仿真对规划算法进行了验证。在进行合理的参数整定后,算法能使规划轨迹在上述验证场景中实现维持巡航速度、跟踪参考线的目标,并且使规划轨迹中的状态满足由安全边界定义的动力学约束。

第4章 考虑动力学约束的自由空间规划方法

4.1 引言

当前自动驾驶的高精地图覆盖区域是有限的，因此要求运动规划算法在无高精地图的区域也能正常工作。在无高精地图时，车道信息要通过传感器数据来获取，由于障碍物遮挡和传感器范围限制，通过感知获得的车道信息会出现部分或者全部丢失的情况。如果要实现为从停车位到停车位的全过程提供自动驾驶服务，车辆可能会经过一些非结构化的空间，比如大型路口、停车场、开放空地等。这些空间本身并不存在车道结构和车道中心线，也很难通过其他方法生成如第三章类似的参考线模型，无法使用上文设计的 Frenet 最优规划器进行规划。

为了使运动规划算法适应更多的场景，本文将这类道路空间规划器无法处理的场景定义为自由场景，并在本章提出了考虑动力学约束的自由空间规划器，使得规划轨迹能同时满足障碍物约束和动力学约束。

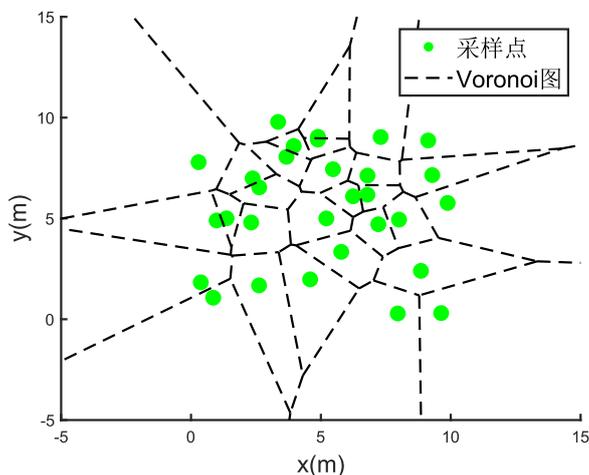


图4-1 Voronoi 图

4.2 改进 Voronoi 势场建模

在道路空间中，障碍物的形状和动作都是比较固定的，我们可以使用上一章设计的风险势场来描述。然而在自由空间中，由于参考线的缺失，障碍物的

位置分布、边缘形状、运动轨迹都会存在更大的不确定性。因此需要建立模型对规划的自由空间进行描述，量化环境中不规则障碍物的风险，为此本文引入了改进的 Voronoi 势场模型。通过 Delaunay 三角剖分建立 Voronoi 图，并且扩展了适用于自动驾驶自由空间的广义 Voronoi 图。

4.2.1 Voronoi 图

Voronoi 图^[28]是一种数学结构，通常用于在二维或三维空间中表示对象之间的接近程度。如图 4-1 所示，该图将空间划分为一组区域，每个区域都由最靠近该区域内的一个特定对象的点组成。本文将从 Voronoi 图出发，建立能够分割不规则障碍物的广义 Voronoi 图。

实现 Voronoi 图中的区域分割有很多种方法，包括 Fortune 算法、基于距离变换的分割法、Delaunay 三角剖分等。其中 Delaunay 三角剖分将 Voronoi 图的计算问题实现了分解，首先将点集剖分为外接圆不包含任何其他点的三角形，然后根据外接圆的圆心获得 Voronoi 图。Voronoi 图、Delaunay 三角剖分的关系如下图所示：

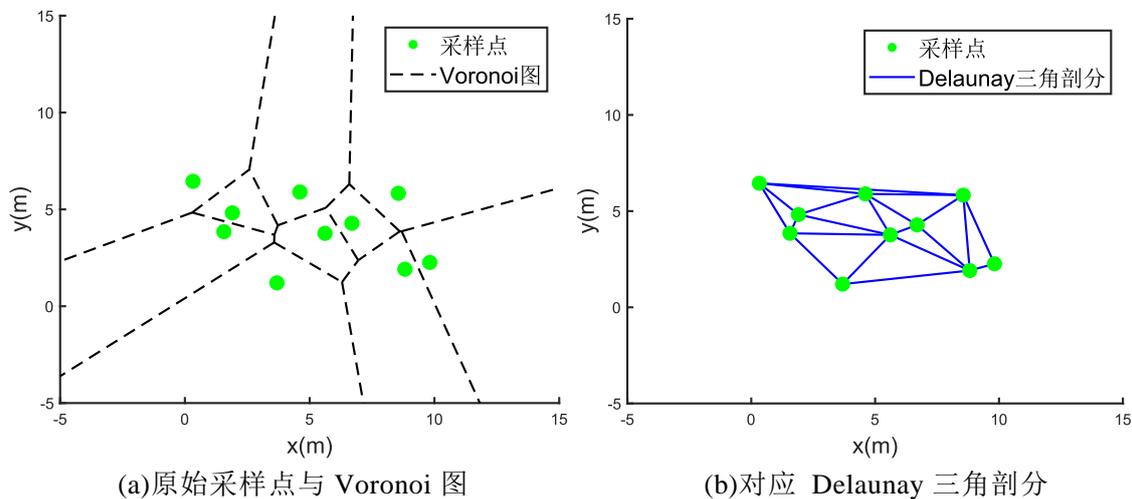


图4-2 Delaunay 三角剖分

图 4-2(a)为一组点集的 Voronoi 图，点集中包含了 10 个点。图 4-2(b)为点集对应的 Delaunay 三角剖分。图 4-3 中显示了由 Delaunay 三角剖分得到 Voronoi 图的过程。对于点集中每个点，找到所有以该点为顶点的三角形，连接这些三角形外接圆心，构成以该点为中心的多边形，连接所有相邻多边形的边，即可

得到完整的 Voronoi 图，如图 4-3 所示。

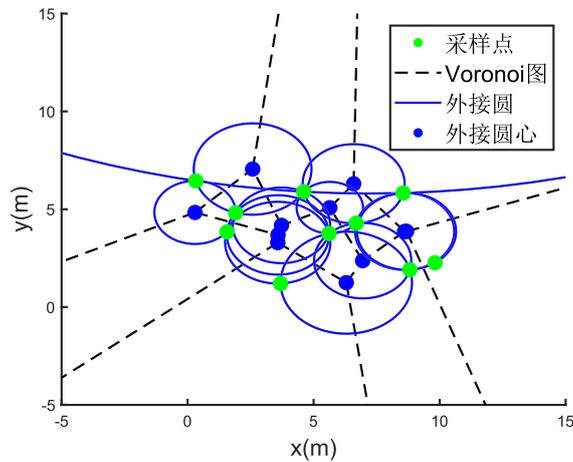


图4-3 Delaunay 三角剖分过程

4.2.2 广义 Voronoi 图

广义 Voronoi 图将点集的分割扩展为不规则障碍物的分割。首先需要定义障碍物的形状和位置，可以通过手工绘制、CAD 软件、地图数据、传感器感知等方式得到。通常，障碍物的形状可以是多边形、曲线、点集等不规则形状。

下图为软件绘制的障碍物数据，模拟自动驾驶自由空间中的障碍物，用障碍物的边界点来表示。如果直接用所有的点集进行 Delaunay 三角剖分和 Voronoi 图生成，效果如下。由于形成了许多多余的 Voronoi 边缘，因此还没有达到分割不规则障碍物的效果。

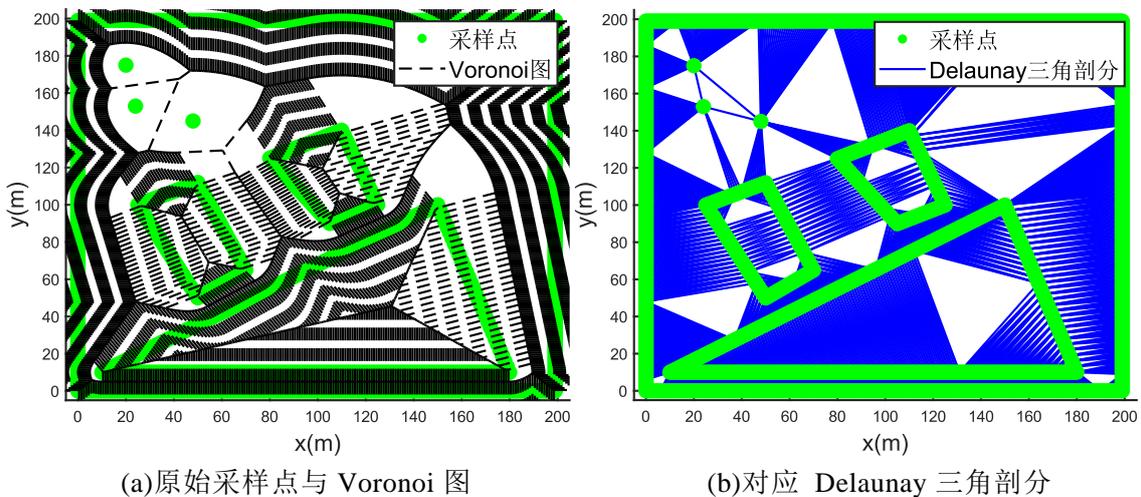


图4-4 不规则障碍物 Voronoi 图

图中多余的分割点主要指穿越障碍物内部的点。广义 Voronoi 图的作用是根据障碍物对可行驶区域进行划分，没有必要分割障碍物以内的区域，因此可以检查每一条轨迹有没有穿过障碍物。分割算法中障碍物是由障碍物边缘的采样点定义的，可以把障碍物边缘的采样点形式转换为采样线段形式表示，然后通过检验分割线与障碍物边界上的每一个采样线段是否相交，来决定是否删除该分割线。如果分割线与任意一个障碍物边缘上的任意一个采样线段存在相交，则删除该条分割线。

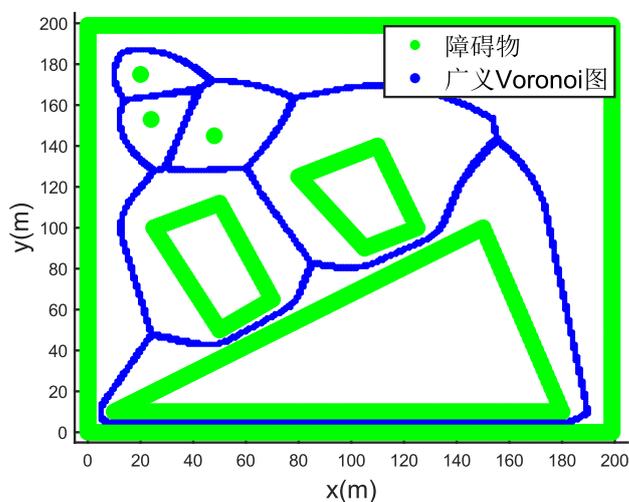


图4-5 广义 Voronoi 图

删除多余分割线后的广义 Voronoi 图如下图所示，实现了对自动驾驶环境中不规则障碍物的空间分割，因此可以用于对输入运动规划算法的环境信息进行建模。

4.2.3 Voronoi 势场

上文定义了自由空间中不规则障碍物的描述方法，并且通过广义 Voronoi 图实现了不规则障碍物的分割。下面将根据 Voronoi 分割线生成 Voronoi 势场，从而能引导自动驾驶运动规划尽量贴近障碍物的分割线，这样的轨迹具有更好的地形通过性。用 Voronoi 势场描述自由空间中的不规则障碍物，能使规划轨迹最大程度地远离障碍物的边缘，还能提高狭窄通道的通过概率。

Voronoi 势场是基于广义 Voronoi 图建立的，需要具有如下特征：在采样位置接近障碍物边缘或者远离 Voronoi 分割线时，Voronoi 势场值上升，代表了对

碰撞风险增加进行惩罚；在采样位置接近 Voronoi 分割线或者远离障碍物边缘，Voronoi 势场值下降，代表了对碰撞风险降低进行奖励。使用 Dolgov 提出的方法^[51]构造由引导线到障碍物边界递减的 Voronoi 势场，如下所示：

$$V(x, y) = \begin{cases} \left(\frac{1}{1 + d_o(x, y)} \right) \left(\frac{d_v(x, y)}{d_o(x, y) + d_v(x, y)} \right) \frac{(d_o - d_o^{max})^2}{(d_o^{max})^2}, & d_o < d_o^{max} \\ 0, & else \end{cases} \quad (4-1)$$

式中 $V(x, y)$ 为采样点处的 Voronoi 势场值， $d_o(x, y)$ 为采样位置到障碍物的最小距离， d_v 为采样位置到广义 Voronoi 图的分割线的最小距离， $d_o^{max} > 0$ 是控制 Voronoi 势场的最大有效范围的常数。

该势场具有以下性质：当 $d_o \geq d_o^{max}$ 时，势场值为零；势场值在 $[0, 1]$ 之间且并且在 (x, y) 上连续，因为不能同时有 $d_o = d_v = 0$ ；只有在障碍物内才能达到最大，且仅在广义 Voronoi 图的分割线处达到最小值。下图为三维视角下的 Voronoi 势场，可以看出上文描述的各项特征：

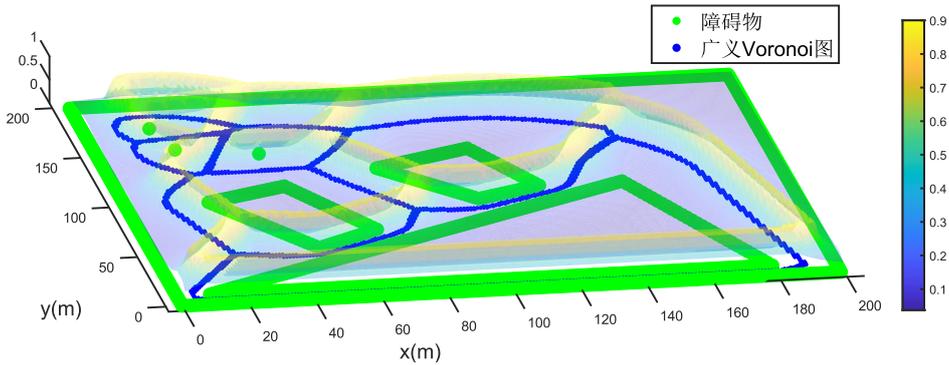


图4-6 Voronoi 势场

4.2.4 改进 Voronoi 势场

4.2.4.1 障碍物分类

自由空间中也有一些障碍物是具有规则的形状的，其运动状态的可预测性也比较强，这些障碍物可以用上一章建立的风险势场来描述。在多数场景下，场景中的障碍物的种类都是很多的。假设一辆自动驾驶汽车穿越一个拥堵的交通路口或者停车场空地，这是一个典型的无道路结构的自由空间场景。阻塞的多辆汽车可能会被传感器识别为大型的不规则障碍物，而其他未阻塞车辆的边缘形状和运动路径是规则的。同样越野场景是一种典型的自由空间场景，场景

中的障碍物包括不规则的地形障碍物，以及具有规则形状和运动的其他车辆、人员。

因此本文在自由空间规划方法设计中，做了如下障碍物分类：

静止不规则障碍物：建筑物、路障、栏杆、施工区域、多个相连阻塞交通参与者（机动车、非机动车、行人等）、岩石、不可通过地形（深坑）等；

动态规则障碍物：行人、自行车、汽车、电动车、摩托车等。

需要注意的是，在自由空间中，不规则的动态障碍物本文是不考虑的，因为不规则的障碍物一般是边缘不规则的地形，或者多辆阻塞无法行动的汽车，这些障碍物无法实现整体的运动；而规则障碍物虽然存在静止和运动状态，因为并不方便对其静止状态和运动状态进行分别建模，因此统一使用动态规则障碍物进行建模。

对于大型移动的不规则障碍物，这在一般的车辆行驶场景中是及其罕见的，本文将暂时不做考虑，视为后续相关研究中需要解决的问题。

4.2.4.2 Voronoi-风险势场叠加

对障碍物进行分类后，通过 Voronoi 势场和风险势场的叠加来描述自由空间中的所有障碍物，即：

$$F = V(x, y) - H(\mathbf{q}, \mathbf{p}, \dot{\mathbf{p}})$$

$$V(x, y) = \begin{cases} \left(\frac{1}{1 + d_o(x, y)} \right) \left(\frac{d_v(x, y)}{d_o(x, y) + d_v(x, y)} \right) \frac{(d_o - d_o^{max})^2}{(d_o^{max})^2}, & d_o < d_o^{max} \\ 0, & else \end{cases} \quad (4-2)$$

$$H(\mathbf{q}, \mathbf{p}, \dot{\mathbf{p}}) = \sum_{i=1}^n \frac{e^{-(\mathbf{q}-\mathbf{p}_i)^T \Omega (\mathbf{q}-\mathbf{p}_i)}}{1 + e^{-\alpha \dot{\mathbf{p}}_i^T (\mathbf{q}-\mathbf{p}_i)}}, \Omega = \text{diag} \left\{ \frac{1}{\sigma_x^2}, \frac{1}{\sigma_y^2} \right\}$$

以图 4-7 所示场景为例，场景中的障碍物包含两类：静态不规则障碍物，边缘用绿色点集表示，主要包括三个无法通过的地形和五辆阻塞的汽车，阻塞汽车用黑色矩形表示；动态规则障碍物，包括两辆运动的汽车，用红色矩形表示。

动态规则障碍物通过风险势场建模的结果如图 4-7 所示。通过风险势场对动态障碍物在规划空间中产生的风险进行量化，充分反映了障碍物的尺寸和速度特征。

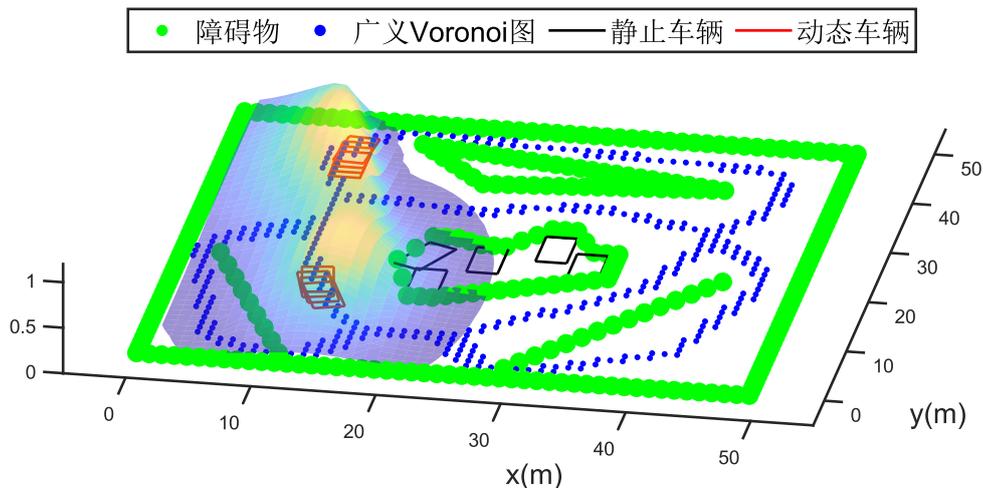


图4-7 动态规则障碍物势场

对于静态不规则障碍物，使用 Voronoi 势场进行描述。Voronoi 势场如图 4-8 所示，Voronoi 势场在运动规划时为自动驾驶汽车提供了沿着 Voronoi 分割线行驶的引导，从而将复杂地形穿越过程的碰撞风险降到最低。

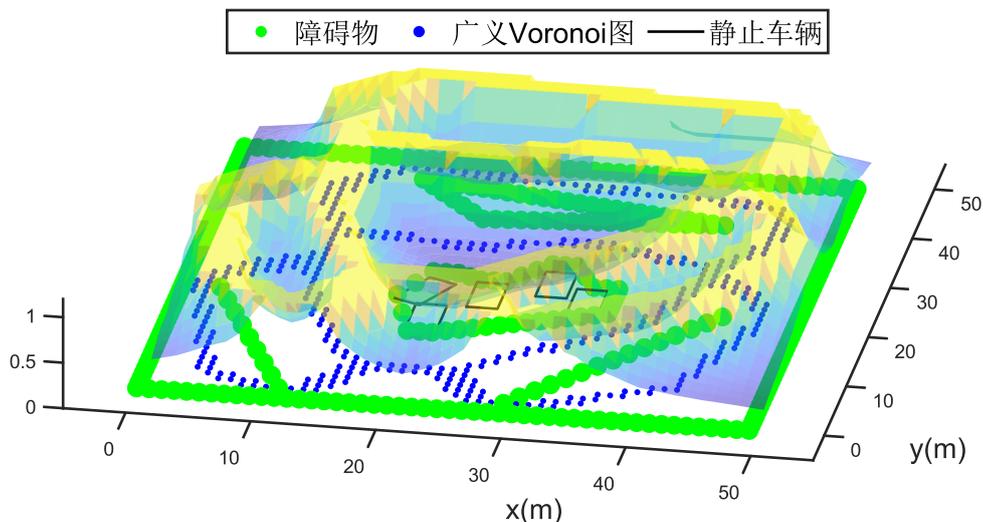


图4-8 静态不规则障碍物势场

图 4-9 为上面所述两个势场的叠加结果，即改进 Voronoi 势场。由于 Voronoi 势场是引力场，风险势场是斥力场，因此在势场叠加时斥力场取负号。从图中可以看出，两个动态障碍物的风险势场导致了对应区域的局部势场的降低。对于运动规划算法来说，对应区域的可通行性将下降，相应的规划路径将被惩罚。

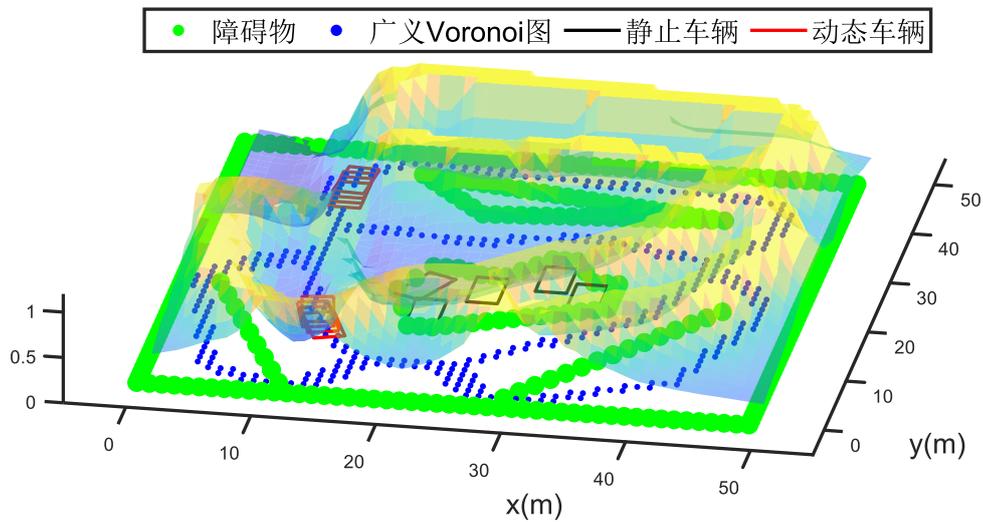


图4-9 改进 Voronoi 势场

4.3 启发式最优规划器

启发式搜索能够通过启发式函数来估计每个可能解的优劣程度，从而能够快速找到可行解。这在运动规划中非常重要，因为运动规划需要快速响应行驶环境的变化，例如突然出现的障碍物或者其他车辆的行驶。适当的启发式函数能够减少搜索空间，从而能够提高运动规划算法的效率。

A*搜索算法是一种改进的启发式搜索算法，将启发式函数与目标节点的位置结合起来，保证找到最优解。由于其最优性、高效性等优点，A*搜索算法在解决自动驾驶运动规划问题时经常被采纳，如^{[51][52]}。在自动驾驶中应用A*算法时，需要做出一些改进以使搜索路径更好的满足车辆约束。

本文通过上文建立动力学模型，定义了搜索的动作基元和动力学搜索代价，设计了一种考虑动力学约束的启发式最优规划器，保证了运动规划的轨迹具有最优性。

4.3.1 搜索动作

在定义搜索算法之前，首先要确定搜索的动作基元。搜索动作决定了搜索算法在空间中扩展状态的方式。对于A*搜索、深度优先搜索、广度优先搜索、代价一致搜索等搜索算法，一种定义其搜索动作的常用方法是将搜索动作定义

为在栅格地图中扩展到相邻的状态。对于移动机器人来说，这些方法比较适合用于能够全向运动的机器人，比如四旋翼无人机。但是对于自由空间中的自动驾驶汽车运动规划来说并不能使用上述搜索动作定义，因为车辆的运动常受到动力学约束和运动学约束。近年来的研究中，自动驾驶汽车运动规划的搜索动作通常定义为运动学搜索动作。本文在定义除前轮转向的运动学搜索动作的同时，提出了前后轮转向的运动学搜索动作，以使算法可扩展到更多场景和车型。

运动学搜索动作采用两轮的自行车模型约束车辆运动轨迹的曲率半径，如下如所示。

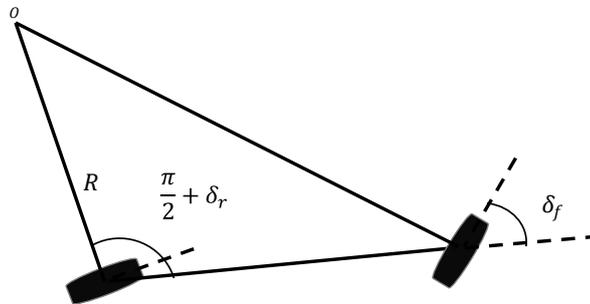


图4-10 车辆运动学模型

对于前后轮转向的汽车来说，车辆后轴中心运动的曲率半径满足：

$$R = \frac{\cos(\delta_f) L}{\sin(\delta_f - \delta_r)} \quad (4-3)$$

其中 R 为汽车后轴中心运动的曲率半径， δ_f 和 δ_r 分别为前后轮转角，可以通过对 δ_f 和 δ_r 进行采样获得搜索动作，如下图所示。

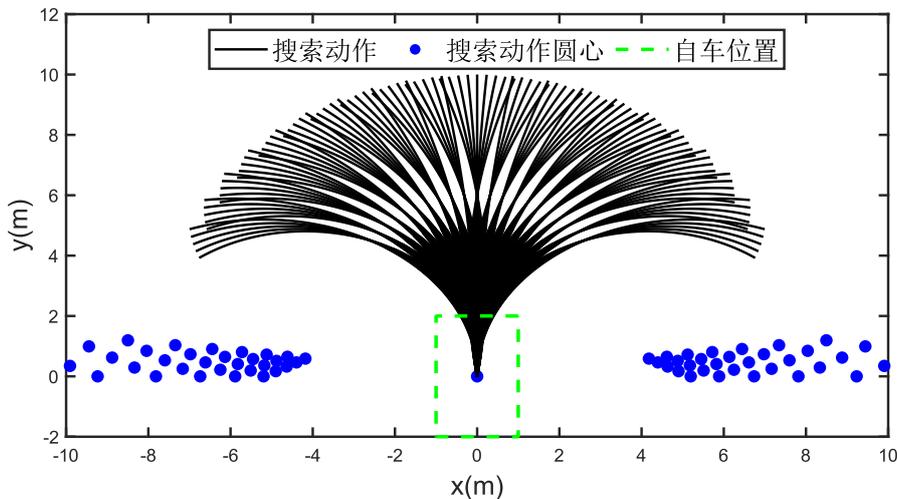


图4-11 前后轮转向搜索动作

在本文第二章建立动力学模型的过程中，为了简化设计将车辆视为常见的前轮转向汽车，令 $\delta_r = 0$ 。为了统一模型形式在此处也进行相应的假设，前轮转向的车辆后轴中心运动的曲率半径表达式可退化为：

$$R = \frac{L}{\tan(\delta_f)} \quad (4-4)$$

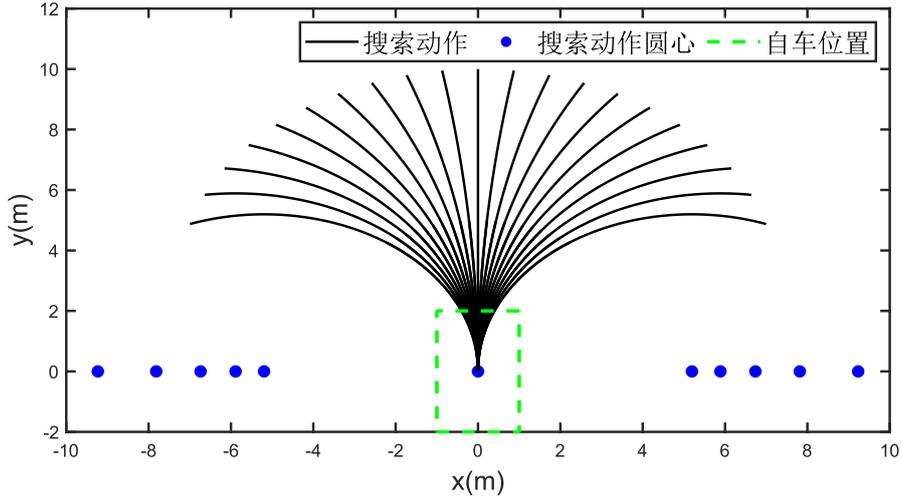


图4-12 前轮转向搜索动作

图 4-11 和图 4-22 中为车速为巡航速度为 10ms、时间步长为 1s 的搜索动作。在规划器中可以采用固定的时间步长，与规划周期一致为 100ms，而搜索动作的速度不同场景的巡航速度进行采样，可以得到不同速度的搜索动作，从而能够实现搜索过程中的速度规划。

4.3.2 启发式搜索

启发式搜索的算法流程如下：

首先定义两个节点集 OPEN 集和 CLOSE 集，OPEN 集代表未被探索的节点，而 CLOSE 集代表已经被探索节点。最开始 OPEN 集保存了所有采样搜索动作到达的状态，CLOSE 集为空。S 代表初始节点，G 代表目标节点。在算法初始化阶段将 S 作为将要被探索的节点，加入到 OPEN 表中。

从初始节点 S 开始，遍历所有采样搜索动作到达的状态，将它们加入到 OPEN 集中，则 S 为这些节点的父节点。从 OPEN 中删除 S，并将 S 加入到 CLOSE 集中。计算 OPEN 集中结点的代价，并且选择代价最小的节点 N，将其

加入到 CLOSE 集中，并且从 OPEN 集中删除。接下来检查 N 所有搜索动作到达的状态 X：

(1) 如果搜索动作导致碰撞，或者状态在 CLOSE 中，跳过本次循环；

(2) 如果 X 不在 OPEN 集中，则进行添加，计算其代价以及将 N 设置为其父节点；

(3) 如果 X 已经在 OPEN 中，则从 S 到 X 计算新的路径，如果代价值更低则修改父节点为 N，重新计算代价；

(4) 继续从 OPEN 集中寻找代价值最小的状态，从 OPEN 中删除，加入到 CLOSE 中，并检查其所有搜索动作到达的状态，如此循环。

结束条件：当 G 出现在 OPEN 集中时，搜索成功；当 OPEN 集为空时，搜索失败。

搜索产生的轨迹的横摆角可能不够平滑，会导致后续控制环节中出现方向盘抖动，因此对输出轨迹的横摆角进行了样条插值处理。

4.3.3 代价函数设计

与上一章设计的 Frenet 最优规划器类似，本文设计的启发式规划器同样需要定义代价函数，以确保轨迹的最优性。自由空间规划器代价函数由四部分组成，如下所示：

$$C_f = C_h + C_g + C_{vor} + C_{mil} \quad (4-5)$$

其中 C_r 为自由空间规划器代价函数， C_h 为启发式代价函数， C_g 为累计代价函数， C_{obs} 为避障代价函数， C_{mil} 为动力学代价函数， C_{dir} 为方向代价

4.3.3.1 启发式代价

启发式代价用于估算当前状态到目标状态的距离，减少搜索空间，提高规划的效率。启发式代价如下定义：

$$C_h(N) = w_h \max(C_{h,1}(N), C_{h,2}(N)) \quad (4-6)$$

其中 $C_h(N)$ 为当前状态 N 到目标状态 G 的距离估计，这个轨迹由两个估计距离取最大值确定；其中 w_g 为累积代价的权重系数为； $C_{h,1}(N)$ 为考虑环境障碍物、不考虑车辆约束的距离估计，通常以 A* 搜索获取； $C_{h,2}(N)$ 为考虑车辆约束、

不考虑环境障碍物的距离估计，通常以 Reeds-Shepp 曲线^[53]的长度获取。

4.3.3.2 累积代价

累积代价 $C_g(N)$ 为初始节点到当前状态的实际距离，为了确保搜索路径的最优性。

$$C_g(N) = w_g \sum_{i=1}^N len(N_{i-1}, N) \quad (4-7)$$

其中 w_g 为累积代价的权重系数。

4.3.3.3 障碍物代价

障碍物代价为当前状态与障碍物的碰撞风险估计，在本章的 4.2 节中已经通过改进 Voronoi 势场对碰撞风险进行了量化。由于 Voronoi 势场为引力场且满足取值范围为 $[0,1]$ ，因此障碍物代价为

$$C_{vor}(N) = w_{vor}(1 - F(N)) \quad (4-8)$$

4.3.3.4 动力学代价

与本文 3.4 中相同，动力学代价 C_{mil} 根据安全距离如下定义，用于评价车辆在通过搜索动作进行状态扩展时车辆的动力学安全性：

$$C_{mil}(N) = -w_{mil}D_{mil}(N) \quad (4-9)$$

4.3.3.5 代价函数权重

在启发式最优规划器中，代价函数权重同样需要在实验过程中人工整定。模型的代价函数权重最终整定如下：

表4-1 启发式最优规划器代价权重

权重	权重值	权重	权重值
w_h	1.0	w_g	1.0
w_{mil}	2.0	w_{vor}	2.0

4.4 自由空间规划器仿真验证

4.4.1 仿真场景选择

对于本章的模型，需要通过实验验证以下内容：

改进 Voronoi 势场能首先生成广义 Voronoi 图，根据不规则障碍物的形状对自由空间进行正确的分割，每个区域包含且仅包含一个完整的障碍物。Voronoi 势场和风险势场的叠加能正确反映不同障碍物类型带来的风险，并形成合理的最低碰撞风险引导。

考虑动力学约束的启发式最优规划器能生成符合车辆约束的搜索基元，并且通过启发式代价函数、累计代价函数、避障代价函数、动力学代价函数对搜索动作进行评价。

在上文中已经完成了算法各个子模块的验证，下面将针对两个典型场景，验证本文设计的考虑动力学约束的启发式最优规划器能够在规划中及时发现潜在的动力学风险，并且实现对风险的抑制。

4.4.2 越野场景验证

越野场景是一种比较常见的自由空间场景。在该场景中，障碍物包括岩石、不可通过地形(深坑)等，可以通过静止不规则障碍物来描述。设置越野场景对规划器进行验证，给定障碍物边缘，在初始状态车辆的速度为 15m/s，搜索动作的速度采样范围为 10-15m/s，速度采样间隔为 1m/s。在 200*200m 的地图范围内的无障碍区域采样车辆的初始位置和目标位置。使用上文所设计的规划器，规划能够使车辆在 25s 内到达目标位置。

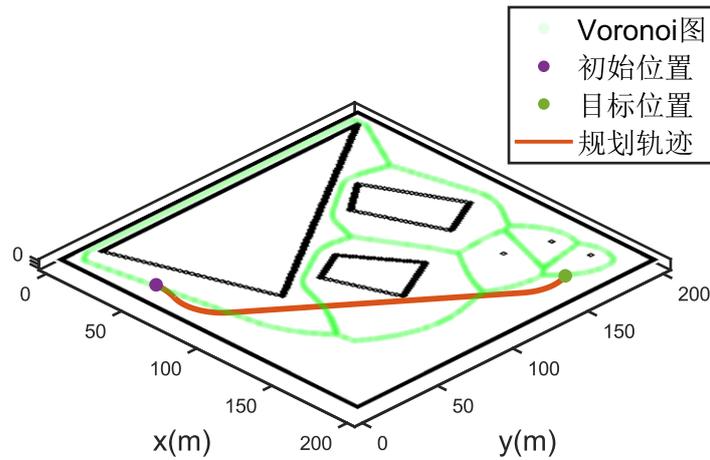


图4-13 越野场景轨迹

下图为轨迹的横摆角曲线，可以看出经过降采样插值处理后的横摆角曲线更加平滑，能够避免轨迹跟踪中的方向盘震动。

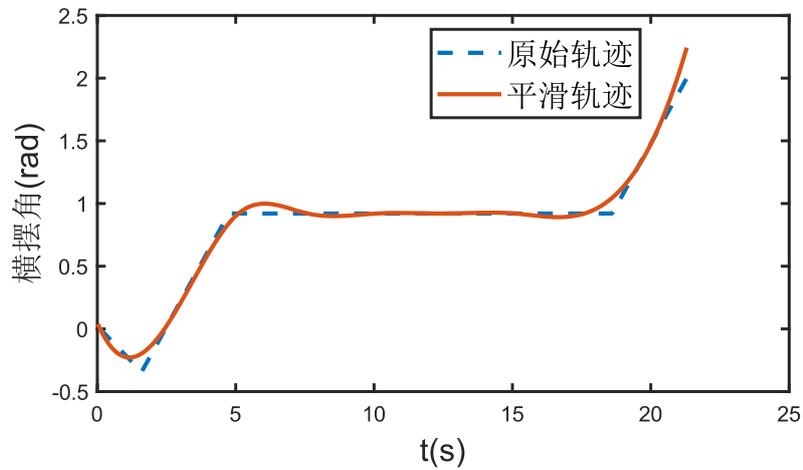


图4-14 越野场景轨迹横摆角

从图 4-15 中可以看出，在越野场景中，自由空间规划器输出的规划轨迹状态没有超出 Milliken 动力学中的安全边界的状态。

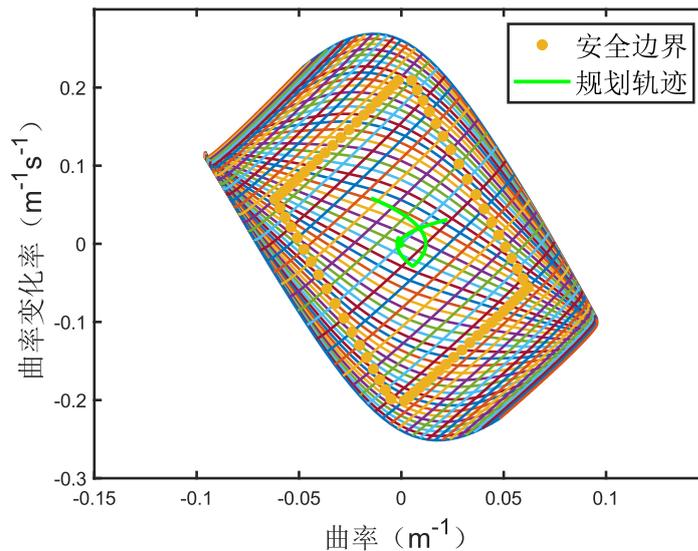


图4-15 越野场景轨迹动力学安全性

4.4.3 阻塞场景

阻塞场景是另一种典型的自动驾驶自由空间场景。在该场景中，障碍物包括建筑物、路障、栏杆、施工区域、多个相连阻塞交通参与者（机动车、非机动车、行人等）等，可以通过静止不规则障碍物来描述。同时也包含行人、自行车、汽车、电动车、摩托车等障碍物，可以通过动态规则障碍物来描述。

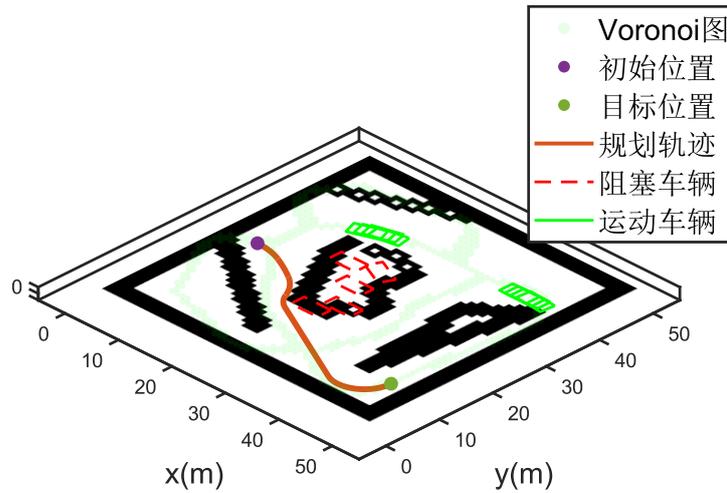


图4-16 阻塞场景轨迹

设置阻塞场景对规划器进行验证，场景中的障碍物包含两类：静态不规则障碍物，主要包括三个无法通过的地形和五辆阻塞的汽车；动态规则障碍物，包括两辆运动的汽车如图 4-16 所示。给定障碍物边缘，在初始状态车辆的速度为 5m/s，搜索动作的速度采样范围为 0-5m/s，速度采样间隔为 1m/s。阻塞场景一般出现在大型路口、停车场，因此在地图范围为 50*50m，面积小于越野场景。在无障碍区域采样车辆的初始位置和目标位置，使用上文所设计的规划器，规划能够使车辆在 20s 内到达目标位置。

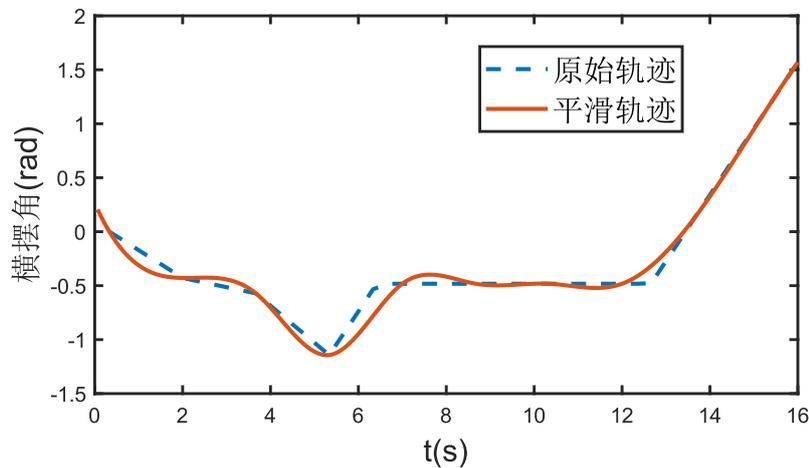


图4-17 阻塞场景轨迹横摆角

阻塞场景中如自由空间规划器输出的规划轨迹同样能保证横摆角的平滑，由于初始状态和搜索状态的纵向速度均比较低，安全边界约束基本等效于搜索动作的曲率约束。

4.5 本章小结

本章设计了考虑动力学约束的自由空间运动规划。针对障碍物不规则的自由空间，本文设计了改进 Voronoi 势场，以描述不规则的地形和障碍物。改进 Voronoi 势场通过广义 Voronoi 图对静态不规则障碍物进行分割，并通过风险势场描述障碍物风险，从而量化了自由空间中多种障碍物的风险。为了降低运动规划对参考线的依赖，本文设计了基于启发式搜索的运动规划算法。本文定义了与动力学相关的启发式信息，从而在选择搜索动作时考虑动力学安全边界约束。通过越野场景和阻塞场景的仿真验证了规划算法的有效性。在不同的自由空间场景实验场景下，在进行合理的参数整定后本文设计的规划方法都产生了到达目标状态的轨迹，避免了与各种障碍物的碰撞，同时满足了动力学安全边界约束。

第 5 章 系统架构设计与硬件在环验证

5.1 引言

在本章节中，综合上文针对道路空间和自由空间设计的考虑动力学约束的规划方法，进行了自动驾驶运动规划系统的总体设计与硬件在环验证。运动规划的功能是为当前场景分配对应的规划方法，并产生最优规划轨迹。本文将运动规划系统通过 C++代码实现，并且在车规级计算平台上进行了部署，并优化了代码效率，保证了 100ms 的规划周期。为了提高运动规划系统在当前车辆自动驾驶域中的部署效率，本文设计了系统的进程间通信接口和 CAN 总线接口。最后，本文通过 CAN 总线实现车规级计算平台与场景仿真平台通信，通过硬件在环实验验证了算法的有效性。

5.2 系统架构设计

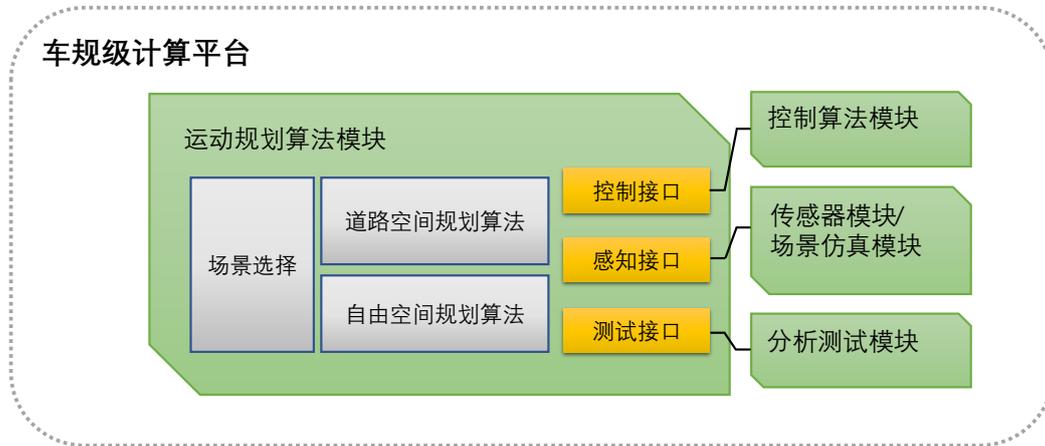


图5-1 系统架构

5.2.1 系统的硬件平台

本算法的硬件平台采用了 Nvidia 公司的 Jetson Orin 平台。Jetson Orin 平台是 Nvidia 在 2022 年发布的车规级高性能计算平台，专门服务于自动驾驶边缘计算任务，能够多级自动驾驶需求，能够支持开发自变道辅助、防碰撞、车道

偏离预警、适应巡航控制、自动紧急制动等自动驾驶功能，进而实现在高速、城区、泊车、换电等场景的自动驾驶。

本研究与多个其他自动驾驶模块的开发在同一个平台上同步进行，这些模块包括车辆动力学分析、CAN 总线通信、轨迹跟踪、轮毂电机的转矩分配等。因此在软件部署时进行了充分的封装和继承，并且根据与其他模块的兼容性要求进行了接口设计。

5.2.2 系统软件架构

在实际部署自动驾驶算法时，运动规划算法的开发效率和计算效率必须要得到保证。为此本文设计了分布式计算的软件架构，如下图所示：

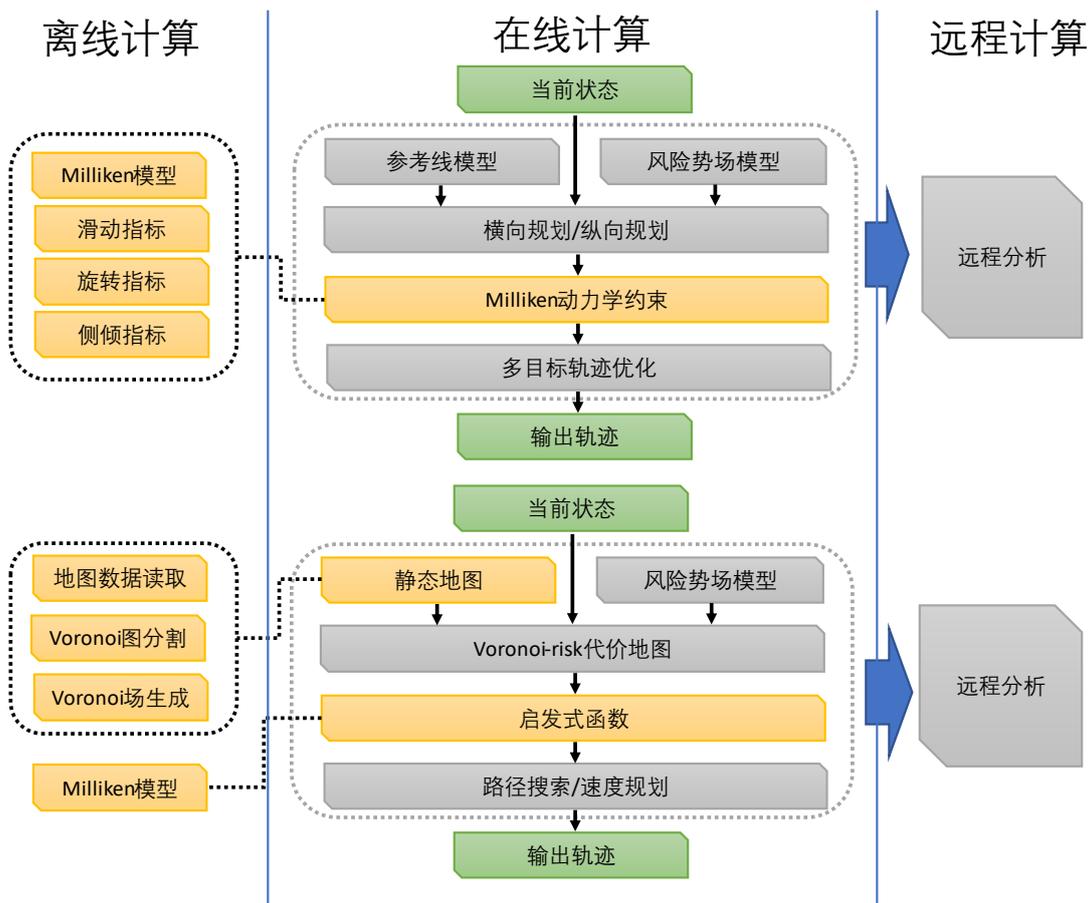


图5-2 系统软件架构

5.2.2.2 离线计算

离线计算主要完成 Milliken 动力学的离线计算和静态障碍物的广义 Voronoi

势场计算。包括系统动力学模型的 Milliken 力矩法计算，以及 Milliken 动力学模型中根据各项指标的安全边界的获取。这部分算法存在着多种计算任务的高次数迭代，这些计算任务包括二维空间最近点匹配、三角形外接圆圆形半径计算、线段与不规则障碍物的相交性判断等。由于车辆的动力学参数和静态障碍物的状态变化频率都远低于规划周期，因此可以通过离线计算来分担在线计算的计算负担。

5.2.2.3 在线计算

系统的在线运算任务主要是根据规划场景选择规划方法，并且调用对应规划器进行规划，规划的结果将通过配置文件中选择的输出接口进行输出。

5.2.2.4 远程计算

远程计算主要服务于代码的嵌入式开发阶段。系统的代码与文档通过 Git 工具进行管理，并通过私有代码仓库托管。这部分软件只集成在算法的 Debug 版本中。在代码的嵌入式 Debug 版本中，代码将提供简略信息可视化脚本与远程接口；在代码的远程 Debug 版本中，代码将提供详细信息可视化脚本。代码的 Git 代码树如图 5-3 所示。

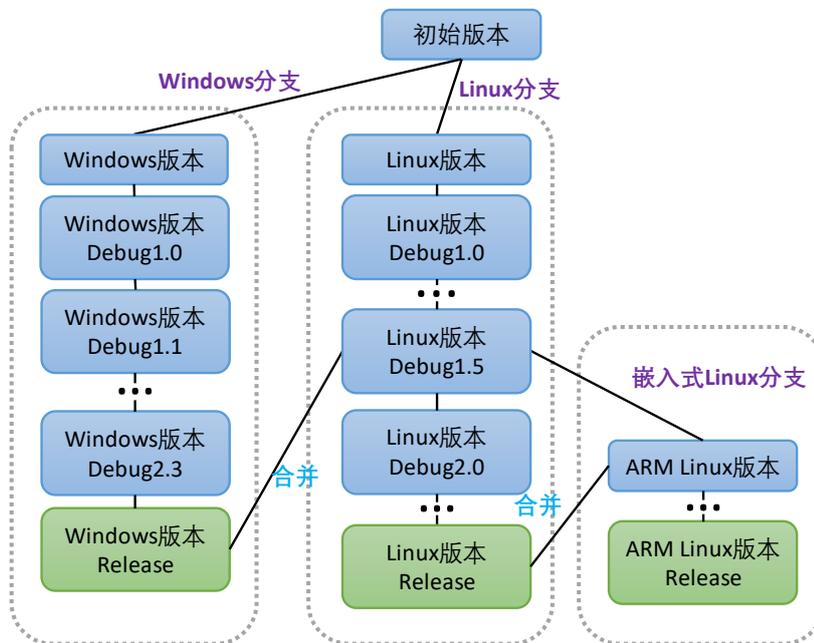


图5-3 Git 代码树

在所有 Windows、Linux、嵌入 Linux 的分支结点中都包含算法的主进程和

编译需要连接的所有库。在嵌入式 Linux 分支代码的 Release 版本中，代码集成了进程间通信、CAN 总线通信、SSH 通信功能，作为与其他算法模块的接口。除此之外，还定义了调试接口，把模型调试参数、可视化参数通过通信接口进行输出，从而发送到远程平台完成调试工作，从而能使算法支持 OTA (Over-the-Air Technology)。在 Release 版本中，将只根据需求保留算法和一种输入输出接口，删除其他通信和调试接口。

在 Linux 分支中，算法实现了在 Linux 平台上的开发，除算法本身外，还移植了 Linux 平台调试工具，包括了四种 Linux 环境下的可视化工具 Bokeh、GNU-Plot、matplotlib-cpp、Octave，从而支持了 Python、C++、Web 环境下进行调试。算法还根据处理器内核版本移植了火焰图分析工具，用于分析算法的每个进程的内存占用。Linux 分支包含了算法和算法的开发、调试环境，在 Release 版本中，算法将保留完整的代码可执行文件与对应开发工具。

在 Windows 分支中，部署了 Windows11 平台下的算法开发环境，从而实现算法和算法开发环境的跨平台可移植性。在 Windows 环境下，为算法集成了算法调试脚本、SSH 远程连接配置、VeDYNA 动力学分析脚本、CAN 通信测试接口和场景仿真工具。

5.2.3 系统输入输出接口

在嵌入式 Linux 分支中，代码集成了进程间通信、CAN 总线通信、SSH 通信接口，用于兼容自动驾驶平台架构和其他自动驾驶模块。

5.2.3.1 进程间通信接口

本文根据 Linux 中的的进程间通信 (IPC, Inter-process communication) 工具，基于两种机制开发了进程间通信接口：管道和共享内存。

共享内存存在进程之间共享同一块物理内存区域，进程可以直接读写该内存区域中的数据，是一种比较快进程间通信方式。本文在 Linux 中以与下游控制算法进程的进程间通信为例，使用 `shmat()` 系统调用创建共享内存区域。在共享内存中写入规划结果，并使用 `shmat()` 系统调用将共享内存区域附加到进程的地址空间中，从而可以直接读写共享内存中的数据。本文以与控制算法的联合仿

真为例，通过共享内存为控制算法提供了轨迹输出，如图 5-4 所示。管道是另一种半双工的进程间通信方式，数据只能单向流动。在 Linux 中，可以分别使用 pipe()和 fork()创建管道和子进程，父进程和子进程就可以通过管道进行通信。本文提供了管道通信和共享内存通信接口，以提高系统的兼容性，默认使用共享内存通信机制。

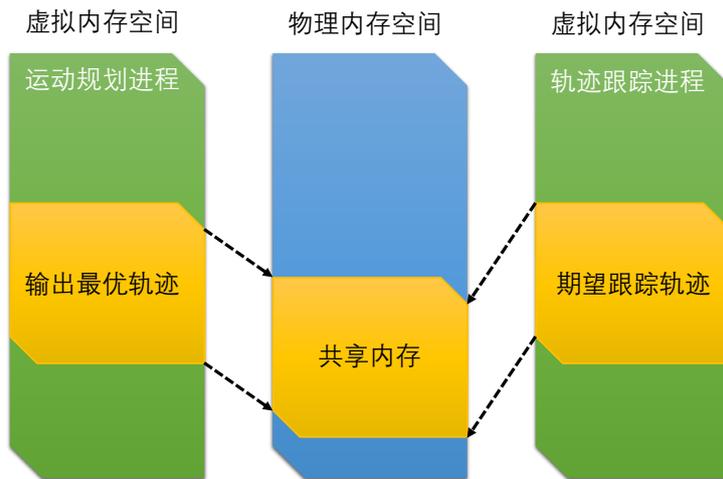


图5-4 规划-控制共享内存通信

5.2.3.2 CAN 总线通信接口

本研究采用的 Nvidia Orin 车规级计算平台提供了 CAN 总线的硬件支持，包含两路 Nvidia CAN 控制器，因此本文基于 Linux 系统的 CAN 总线通信库，开发了系统的 CAN 总线收发接口。

5.2.3.3 SSH 通信接口

SSH 通信接口用于在运动规划系统集成安装后，通过 Windows 或者 Linux 系统的开发机和计算机网络接入嵌入式系统进行调试。SSH 使用 Internet 协议 (IP) 来进行网络通信。作为 SSH 客户端，而搭载运动规划系统的嵌入式平台配置为 SSH 服务器，其 IP 地址用于在网络中唯一标识它们，SSH 客户端使用 SSH 服务器的 IP 地址来建立连接。

在算法开发过程中，搭建了如图 5-5 所示的计算机网络：

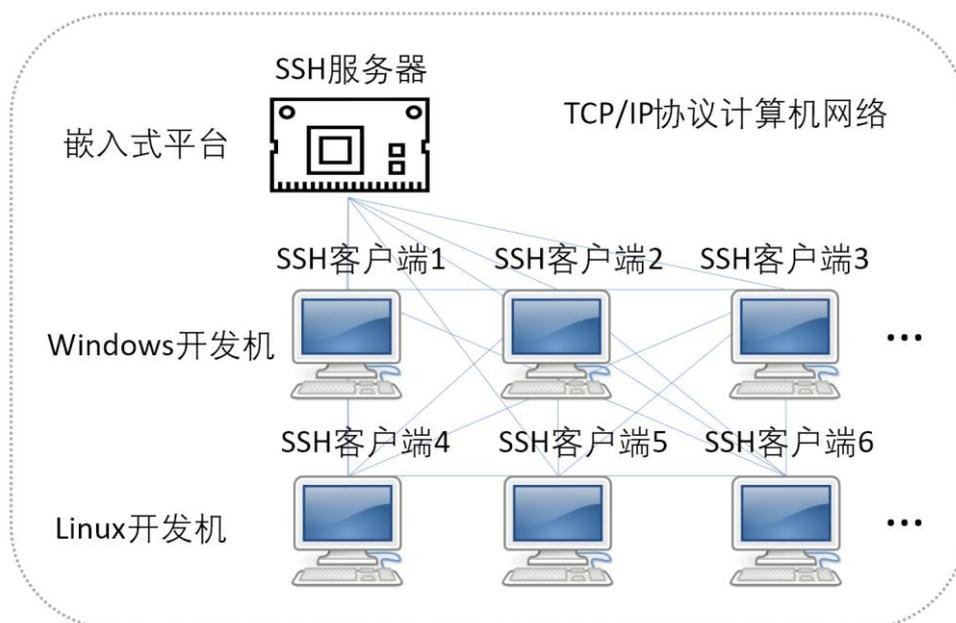


图5-5 SSH 通信网络

本文通过 OpenSSH 在嵌入式 Linux 平台中进行服务器配置，然后通过 Windows 开发及和 Linux 开发机自带的 SSH 客户端程序进行 SSH 通信，通过远程登陆修改嵌入式平台代码，并在开发机上建立调试分析环境，实现远程计算。

5.3 算法的嵌入式实现

本系统在上述软件架构的基础上，对运动规划系统中的算法进行了 C++ 开发。C++ 属于一种面向对象的编程语言，并且支持泛型编程和函数式编程等编程范式。本文在充分掌握 C++ 语言开发流程的基础上，实现了算法在车规级平台上的部署。在开发时遵循标准代码风格的 C++ 规范，以提高代码的规范性和可读性，提高开发效率。

由于 C++ 是面向对象的编程语言，因此本文将从各个类的设计来介绍代码的嵌入式实现。

5.3.1 基本数学类

基本数学类的代码结构如图 5-6 所示。

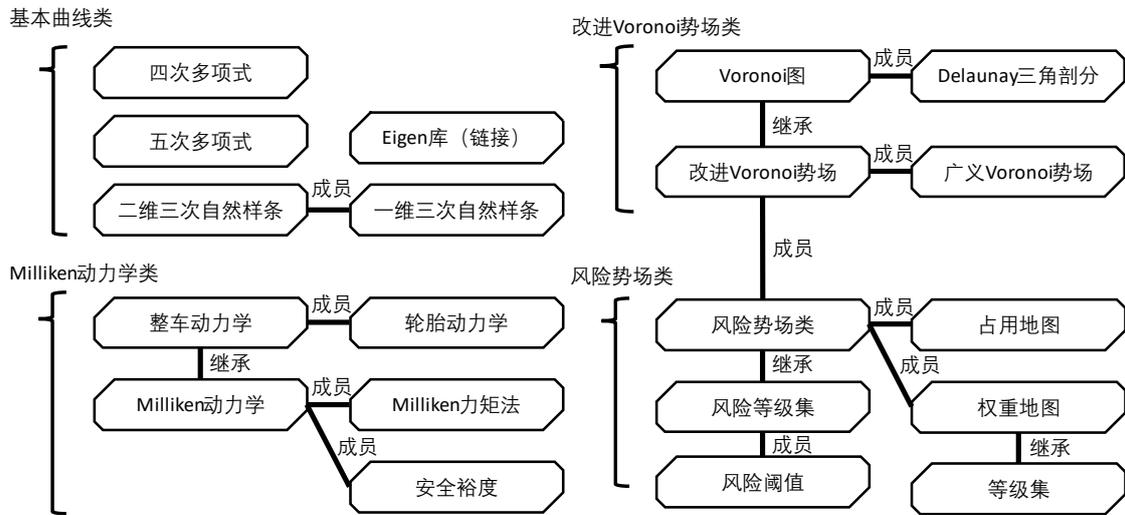


图5-6 基本数学类代码结构

5.3.1.2 基本曲线类

本文建立的 C++基本曲线工具包括四次多项式类、五次多项式类、一维三次自然样条类、二维三次自然样条类。

多项式类的成员变量包括多项式的系数，在构造对象时初始化。可以通过输入多项式的系数构造，也可以通过输入起始和终止条件进行参数计算。成员函数包括计算系数的重载构造函数、计算函数值和计算各阶导数。

一维自然三次样条的成员变量包括样条曲线的控制点和样条曲线的各段参数。在构造时通过控制点计算样条曲线的各段参数。成员函数包括计算参数中间变量矩阵和向量，以及计算函数值和计算各阶导数。为了方便矩阵和向量运算，在基本曲线类代码中链接了 Eigen 第三方库。

5.3.1.3 Milliken 动力学类

Milliken 动力学类是由整车动力学基类继承的派生类。在基类中，成员变量包括车辆的各项动力学参数和状态变量，成员函数包括轮胎模型和整车动力学的微分方程求解。在派生类中，成员函数加入了模型的 Milliken 力矩法转换、安全指标计算、安全边界搜索、安全距离计算。

5.3.1.4 风险势场类

包括风险势场类基类和风险等级集派生类。风险势场类包含占用地图和权重地图，以及风险势场值计算函数。风险等级集派生类包括等级集阈值计算，风险势场值计算函数重载为等级集状态查询函数。

5.3.1.5 改进 Voronoi 势场类

包括广义 Voronoi 图基类和改进 Voronoi 场派生类。基类的成员变量包括 Voronoi 分割线，成员函数包括 Delauny 三角剖分、Voronoi 分割线计算。派生类成员变量增加了风险势场类对象，并且加入了改进 Voronoi 势场计算函数。

5.3.2 规划器类

本文将第 3 章和第 4 章设计的规划方法在 C++ 中设计为规划器类，设计了规划器基类，并且设计了道路空间规划器和自由空间规划器。基类将作为抽象类定义，禁止了基类的构造。

抽象类中包含了预处理函数和规划函数，在派生类中进行了重载。重载的预处理函数和规划函数如图 5-4 所示。图中预处理函数用黄色表示，在规划器构造时执行，作为算法的离线计算部分；图中规划函数用灰色表示，在每个规划周期执行，作为算法的在线计算部分。

5.3.3 开发工具类

5.3.3.1 SSH 控制类

通过 Linux system 库在 C++ 进程中开启 shell 进程条用系统命令，进行 OpenSSH 远程链接；成员变量包括了主机地址 ip、用户名、密码，成员变量包括了远程连接和文件系统操作

5.3.3.2 CAN 控制类

通过将 Linux can-utils 的 C 语言库适配为 C++ 版本提供了 CAN 总线的收发函数。成员变量包括数据发送缓存、波特率、CAN 通道号，成员函数包括 CAN 初始化、CAN 发送、CAN 接收。

5.3.3.3 可视化脚本类

建立了 GnuPlotter 类和 MatplotlibCPP 类，分别定义了各种格式数据的可视化函数，并通过泛型编程使可视化工具对不同变量类型的应用率。

5.4 硬件在环实验验证

5.4.1 方案设计

在本课题中通过开发机和嵌入式平台的联合仿真实现模型的硬件在环实验验证。在开发机中，系统提供场景仿真，并通过 CAN 总线设备与嵌入式平台通信设备，将驾驶场景信息发送到嵌入式平台。规划算法在嵌入式平台上以 100ms 为周期运行，并返回规划结果到开发机。硬件在环流程图如图 5-7 所示：

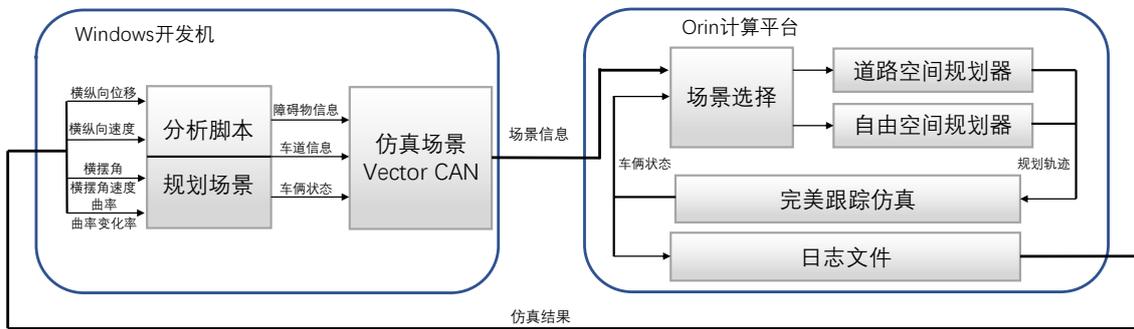


图5-7 硬件在环方案设计

硬件在环实验需要的设备如下：一台电脑作为 Windows 开发机，在运行 CAN 通信模块 Vehicle Network Toolbox，通过 Vehicle Network Toolbox Support Package for Vector CAN Devices 驱动 Vector CAN 总线设备，将规划场景通过 CAN 总线发送到嵌入式平台；一台 Nvidia Jetson Orin 车规级高性能计算单元，作为运动规划算法的硬件平台。实验所需的硬件设备如下表 5-1 所示：

表5-1 实验设备

实验设备	配件/软件	作用
Windows 个人电脑	算法测试脚本	提供仿真场景与分析工具
	Vector CAN	提供 CAN 通信硬件和驱动
Orin 计算单元	无线网络模块	远程通信
	运动规划系统	计算轨迹

5.4.2 实验验证及结果分析

实验场景一：弯道场景。本文将 3.5.4 章节中的车道数据和道路空间规划器参数导入实验系统，嵌入式平台中的运动规划系统根据场景类型，调用道路空

间规划器进行规划。在进行实验时，本文除了部署第三章设计的道路空间规划器外，还将规划器中的动力学约束移除，作为对照试验使用的规划器。

有无动力学约束的道路空间规划算法规划轨迹如下图所示。

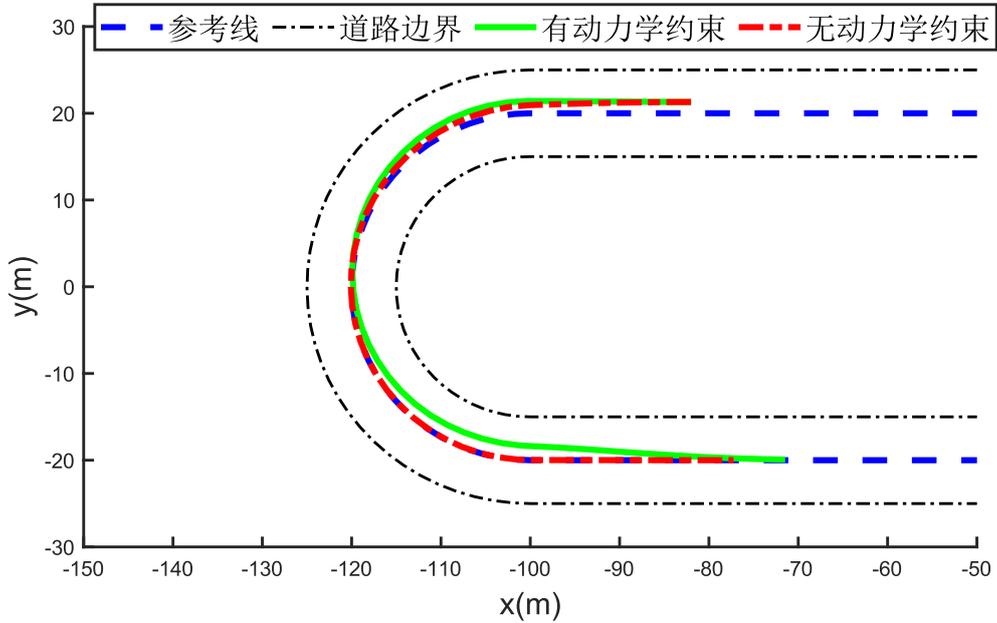


图5-8 弯道场景规划轨迹对比

无动力学约束的规划器选择在入弯时直接快速贴近车道中心线，并且保持沿车道中心线过弯。但由于在进出弯道时中心线曲率均存在从 0 到常量的阶跃变化，曲率的变化率超出了车辆的动力学限制，因此存在动力学风险。加入动力学约束时，规划器在贴近车道中心线的前提下采取了更合理的入弯轨迹。从图 5-9 可以看出无动力学约束的轨迹存在超出动力学安全边界的状态，而加入动力学约束后将规划轨迹状态约束在了安全边界内。

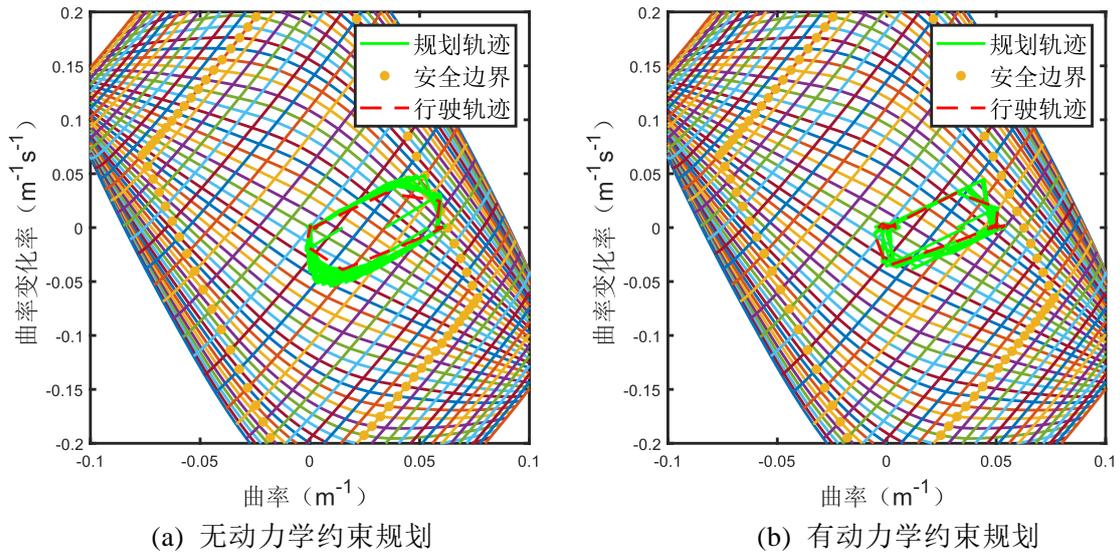


图5-9 弯道场景规划轨迹状态对比

实验场景二：越野场景。本文将 4.4.2 章节中的不规则障碍物边界数据和自由空间规划器参数导入实验系统，嵌入式平台中的运动规划系统根据场景类型，调用自由空间规划器进行规划。在进行实验时，同样设置了与场景一类似的对照试验。

有无动力学约束的自由空间规划算法规划轨迹如图 5-10 所示。

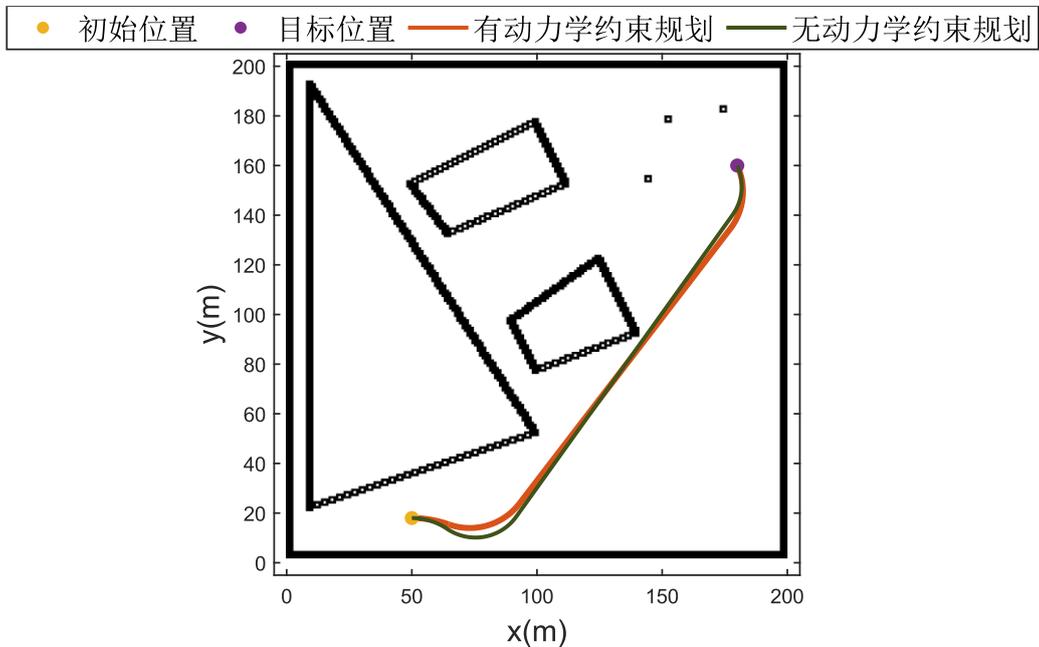


图5-10 越野场景规划轨迹对比

无动力学约束的规划器会更强调降低路径的碰撞风险，因此也倾向于采取

紧急的转向动作远离障碍物。而有动力学约束的规划器会采取更加平缓的路径，其平缓性可以从图 5-11 中的横摆角比较中更明显地体现。

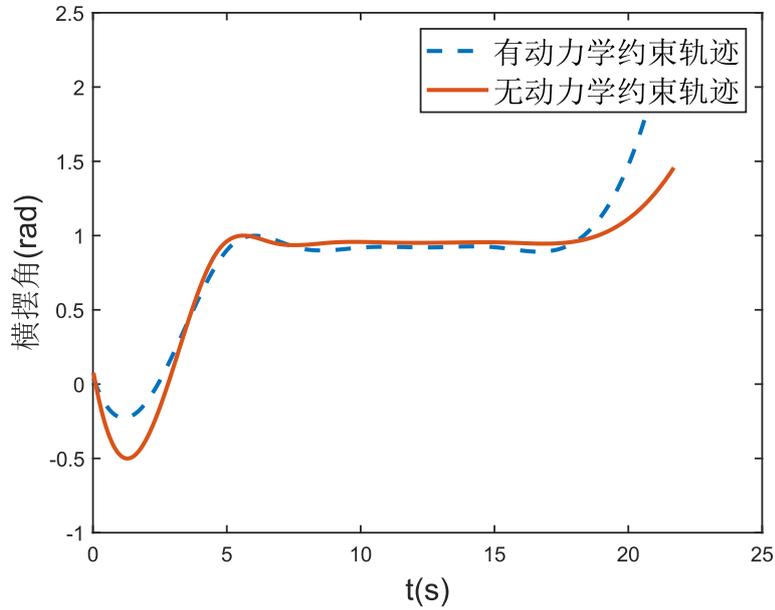


图5-11 越野场景规划横摆角对比

图 5-12 可以看出加入动力学约束后使规划轨迹状态更加远离安全边界，降低了轨迹的动力学风险。

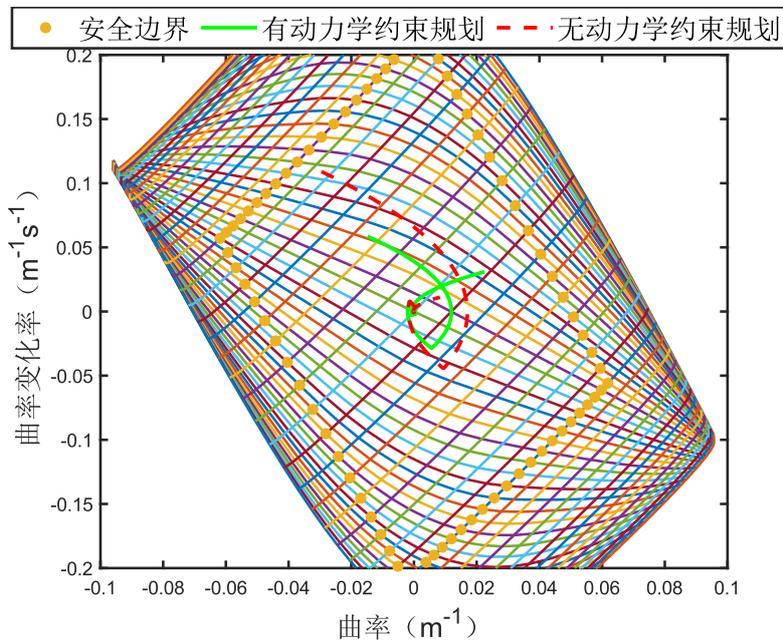


图5-12 越野场景规划轨迹状态对比

5.5 本章小结

在第五章，本文设计并实现了算法的硬件在环验证。首先设计了运动规划系统软硬件架构，将运动规划系统通过 C++ 代码实现，并且在车规级计算平台上进行了部署。为了提高开发效率与算法效率，本文设计了系统的进程间通信接口和 CAN 总线接口。本文通过 CAN 总线实现车规级计算平台与场景仿真平台通信，通过硬件在环实验验证了算法的有效性。在实验过程中，本文部署了不含动力学约束的规划方法，与本文设计的规划方法进行对比。本文设计的规划方法在加入动力学约束后能使规划轨迹状态更加远离安全边界，降低规划轨迹的动力学风险。

结 论

为推广运动规划算法的应用场景，提高规划轨迹的动力学安全性，本文提出了考虑动力学约束的自动驾驶汽车运动规划算法。

车辆动力学模型研究方面，论文首先建立并分析了 14 自由度车辆动力学模型，通过 Milliken 力矩法将模型转换为 Milliken 动力学模型。同时，定义滑动指数、旋转指数、侧倾指数，并通过深度优先搜索获得动力学安全边界，并定义安全距离。针对有道路空间和自由空间，本文分别设计了规划方法，并进行了仿真验证。本文还进行了系统架构设计，包含硬件平台、软件架构和输入输出接口，给出了算法的开发和实现方案。设计了硬件在环实验流程开发流程，并通过 Jetson Orin 平台的仿真和硬件在环仿真，验证了算法的有效性。

本论文的主要创造性工作归纳如下：

1. 建立了适用于汽车运动规划的 Milliken 动力学模型，量化了规划轨迹中的动力学风险，实现了动力学模型与规划算法的结合方法，并通过分布式计算架构保证了规划算法的计算效率。

2. 建立了一种适用于道路空间巡航模式下的自动驾驶运动规划方法，实现了在避障、过弯等场景中保证规划轨迹的动力学安全性。

3. 建立了一种适用于自由空间的自动驾驶运动规划方法，能通过改进 Voronoi 势场补充了自由空间中缺失的车道引导信息，并通过启发式搜索获得动力学安全的轨迹。

4. 对系统的硬件平台、软件架构、输入输出接口都进行了设计，提供了完整的系统开发、维护和测试流程，并完成了系统可行性验证。

今后还应在以下几个方面继续深入研究：

1. 在各种细分场景下，可以引入自适应的参数调节机制，从而使同一个规划器在过弯、避障等细分场景下的性能实现进一步优化；

2. 加强与感知、控制等上下游模块的联合调试和兼容性提升，使得规划系统在接收不同形式的感知输入，以及适配不同的跟踪算法时，都能具有比较好的性能。

参考文献

- [1] Chen, C., Seff, A., Kornhauser, A., et al. "Deepdriving: Learning Affordance for Direct Perception in Autonomous Driving." Proceedings of the IEEE international conference on computer vision, 2015.
- [2] Buehler, M., Iagnemma, K., and Singh, S. The Darpa Urban Challenge: Autonomous Vehicles in City Traffic. Vol. 56: springer, 2009.
- [3] Ferguson, D., Howard, T. M., and Likhachev, M. "Motion Planning in Urban Environments: Part I." Paper presented at the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, 22-26 Sept. 2008.
- [4] Howard, T. M., and Kelly, A. "Optimal Rough Terrain Trajectory Generation for Wheeled Mobile Robots." The International Journal of Robotics Research 26, no. 2 (2007): 141-66.
- [5] Montemerlo, M., Becker, J., Bhat, S., et al. "Junior: The Stanford Entry in the Urban Challenge." Journal of field Robotics 25, no. 9 (2008): 569-97.
- [6] Werling, M., Ziegler, J., Kammel, S., et al. "Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenet Frame." Paper presented at the 2010 IEEE International Conference on Robotics and Automation, 2010.
- [7] Schlechtriemen, J., Wabersich, K. P., and Kuhnert, K.-D. "Wiggling through Complex Traffic: Planning Trajectories Constrained by Predictions." Paper presented at the 2016 IEEE Intelligent Vehicles Symposium (IV), 2016.
- [8] Wang, Z., Deng, W., Zhang, S., et al. "Vehicle Automatic Lane Changing Based on Model Predictive Control." SAE International Journal of Passenger Cars-Electronic and Electrical Systems 9, no. 1 (2016): 231-37.
- [9] Fan, H., Zhu, F., Liu, C., et al. "Baidu Apollo Em Motion Planner." arXiv preprint arXiv:1807.08048 (2018).
- [10] Fan, H., Xia, Z., Liu, C., et al. "An Auto-Tuning Framework for Autonomous Vehicles." arXiv preprint arXiv:1808.04913 (2018).
- [11] 李柏. 复杂约束下自动驾驶车辆运动规划的计算最优控制方法研究 [D]; 浙江大学, 2018.
- [12] Zhang, Y., Sun, H., Zhou, J., et al. "Optimal Vehicle Path Planning Using Quadratic Optimization for Baidu Apollo Open Platform." Paper presented at the 2020 IEEE Intelligent Vehicles Symposium (IV), 2020.
- [13] Guo, H., Shen, C., Zhang, H., et al. "Simultaneous Trajectory Planning and Tracking Using an MPC Method for Cyber-Physical Systems: A Case Study of

- Obstacle Avoidance for an Intelligent Vehicle." *IEEE Transactions on Industrial Informatics* 14, no. 9 (2018): 4273-83.
- [14] Huang, Y., Wang, H., Khajepour, A., et al. "A Novel Local Motion Planning Framework for Autonomous Vehicles Based on Resistance Network and Model Predictive Control." *IEEE Transactions on Vehicular Technology* 69, no. 1 (2019): 55-66.
- [15] Zhou, J., Zheng, H., Wang, J., et al. "Multiobjective Optimization of Lane-Changing Strategy for Intelligent Vehicles in Complex Driving Environments." *IEEE Transactions on Vehicular Technology* 69, no. 2 (2020): 1291-308. <https://doi.org/10.1109/TVT.2019.2956504>.
- [16] Yang, S., Zheng, H., Wang, J., et al. "A Personalized Human-Like Lane-Changing Trajectory Planning Method for Automated Driving System." *IEEE Transactions on Vehicular Technology* 70, no. 7 (2021): 6399-414.
- [17] Liu, Y., Zhou, B., Wang, X., et al. "Dynamic Lane-Changing Trajectory Planning for Autonomous Vehicles Based on Discrete Global Trajectory." *IEEE Transactions on Intelligent Transportation Systems* 23, no. 7 (2021): 8513-27.
- [18] Lim, W., Lee, S., Sunwoo, M., et al. "Hybrid Trajectory Planning for Autonomous Driving in on-Road Dynamic Scenarios." *IEEE Transactions on Intelligent Transportation Systems* 22, no. 1 (2019): 341-55.
- [19] 张鹏, 龚显旭, 孙港, 等. 基于CarSim的平行泊车仿真分析 [J]. *汽车实用技术*, 2022, 47(7): 12-5.
- [20] 李荣粲, 庄伟超, 殷国栋, 等. 自动驾驶赛车路径与车速协同规划方法 [J]. *机械工程学报*, 2022, 58(10): 200-8.
- [21] 殷国栋, 王凡勋, 任彦君, 等. 基于稳定域的四轮独立驱动智能电动汽车路径规划方法, CN113008240A [P/OL].
- [22] 龚建伟, 姜岩, 徐威. 无人驾驶车辆模型预测控制[M].北京理工大学出版社.2014.
- [23] Buehler, M., Iagnemma, K., and Singh, S. *The 2005 Darpa Grand Challenge: The Great Robot Race*. Vol. 36: Springer, 2007.
- [24] Urmson, C., Anhalt, J., Clark, M., et al. "High Speed Navigation of Unrehearsed Terrain: Red Team Technology for Grand Challenge 2004." Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-04-37 1 (2004).
- [25] Gutierrez, A., Galatali, T., Gonzalez, J. P., et al. "Preplanning for High Performance Autonomous Traverse of Desert Terrain Exploiting a Priori Knowledge to Optimize Speeds and to Detail Paths." CMU-RI-TR-05-54 (2005).

- [26] Thrun, S., Montemerlo, M., Dahlkamp, H., et al. "Stanley: The Robot That Won the Darpa Grand Challenge." *Journal of field Robotics* 23, no. 9 (2006): 661-92.
- [27] Trepagnier, P. G., Nagel, J., Kinney, P. M., et al. "Kat-5: Robust Systems for Autonomous Vehicle Navigation in Challenging and Unknown Terrain." *Journal of Field Robotics* 23, no. 8 (2006): 509-26.
- [28] Urmson, C., Ragusa, C., Ray, D., et al. "A Robust Approach to High-Speed Navigation for Unrehearsed Desert Terrain." *Journal of Field Robotics* 23, no. 8 (2006): 467-508.
- [29] Persson, Sven Mikael, and Inna Sharf. "Sampling-Based a* Algorithm for Robot Path-Planning." *The International Journal of Robotics Research* 33, no. 13 (2014): 1683-708.
- [30] 张浩. 基于A*算法的自动泊车全局路径规划算法研究 [D]; 吉林大学, 2022.
- [31] Cai, L., Guan, H., Zhou, Z. Y., et al. "Parking Planning under Limited Parking Corridor Space." *IEEE Transactions on Intelligent Transportation Systems* 24, no. 2 (2023): 1962-81.
- [32] 冯欣阳. "基于港口环境的自动泊车系统路径规划与运动控制." 硕士, 河北工业大学, 2022.
- [33] Kuffner, James J, and Steven M LaValle. "RRT-Connect: An Efficient Approach to Single-Query Path Planning." Paper presented at the Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065), 2000.
- [34] Karaman, Sertac, Matthew R Walter, Alejandro Perez, Emilio Frazzoli, and Seth Teller. "Anytime Motion Planning Using the RRT." Paper presented at the 2011 IEEE international conference on robotics and automation, 2011.
- [35] Gammell, Jonathan D, Siddhartha S Srinivasa, and Timothy D Barfoot. "Informed Rrt: Optimal Sampling-Based Path Planning Focused Via Direct Sampling of an Admissible Ellipsoidal Heuristic." Paper presented at the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014.
- [36] Perez, A., R. Platt, G. Konidaris, L. Kaelbling, and T. Lozano-Perez. "LQR-RRT*: Optimal Sampling-Based Motion Planning with Automatically Derived Extension Heuristics." Paper presented at the 2012 IEEE International Conference on Robotics and Automation, 14-18 May 2012 2012.
- [37] 王圣懿, 黄劲松. 基于先验概率分布改进的RRT车辆运动规划算法 [J]. *导航定位学报*, 2022, 10(2): 85-92.
- [38] 杜卓洋. 无人驾驶车辆轨迹规划算法研究 [D]; 浙江大学, 2019.
- [39] 郑凯林, 韩宝玲, 王新达. 基于改进TEB算法的阿克曼机器人运动规划系

- 统 [J]. 科学技术与工程, 2020, 20(10): 3997-4003.
- [40] 余冬冬. 移动机器人避障与轨迹规划 [D]; 浙江大学, 2017.
- [41] Li, Z., Xiong, L., Leng, B., et al. Path Planning Method for Perpendicular Parking Based on Vehicle Kinematics Model Using Mpc Optimization. SAE Technical Paper (2022).
- [42] Sorniotti, A., and Crocombe, A. Chassis Torsional Stiffness: Analysis of the Influence on Vehicle Dynamics. SAE Technical Paper (2010).
- [43] Milliken, W. F., Milliken, D. L., and Metz, L. D. Race Car Vehicle Dynamics. Vol. 400: SAE international Warrendale, 1995.
- [44] Currier, P. N. "A Method for Modeling and Prediction of Ground Vehicle Dynamics and Stability in Autonomous Systems." Virginia Tech, 2011.
- [45] Komandi, G. "Establishment of Soil-Mechanical Parameters Which Determine Traction on Deforming Soil." Journal of Terramechanics 27, no. 2 (1990): 115-24.
- [46] Will, A. B., and Zak, S. H. "Modelling and Control of an Automated Vehicle." Vehicle System Dynamics 27, no. 3 (1997): 131-55.
- [47] Hsu, Y.-H. J. Estimation and Control of Lateral Tire Forces Using Steering Torque. Stanford University, 2009.
- [48] Pacejka, H. Tire and Vehicle Dynamics. Elsevier, 2005.
- [49] Wenzel, T. A. "State and Parameter Estimation for Vehicle Dynamic Control." Coventry University, 2005.
- [50] Pierson, A., Schwarting, W., Karaman, S., et al. "Navigating Congested Environments with Risk Level Sets." Paper presented at the 2018 IEEE International Conference on Robotics and Automation (ICRA), 21-25 May 2018 2018.
- [51] Dolgov, D., Thrun, S., Montemerlo, M., et al. "Path Planning for Autonomous Vehicles in Unknown Semi-Structured Environments." The international journal of robotics research 29, no. 5 (2010): 485-501.
- [52] Shang, E., Dai, B., Nie, Y., et al. "A Guide-Line and Key-Point Based a-Star Path Planning Algorithm for Autonomous Land Vehicles." Paper presented at the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), 20-23 Sept. 2020 2020.
- [53] Reeds, J., and Shepp, L. "Optimal Paths for a Car That Goes Both Forwards and Backwards." Pacific journal of mathematics 145, no. 2 (1990): 367-93.

攻读硕士学位期间取得创新性成果

（一）发表的学术论文

- [1] You, Y., Zhao, L., Liu, H., et al. "A Hybrid Trajectory Planning Strategy for Intelligent Vehicles with Collision Avoidance." Paper presented at the 2022 41st Chinese Control Conference (CCC), 25-27 July 2022.

哈尔滨工业大学学位论文原创性声明和使用权限

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《考虑动力学约束的自动驾驶汽车运动规划研究》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：  日期： 2023 年 5 月 26 日

学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

(1)学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；(2)学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；(3)研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：  日期： 2023 年 5 月 26 日

导师签名：  日期： 2023 年 5 月 26 日

致 谢

衷心感谢我的导师赵林辉老师。您是我在硕士研究期间的引路人和学术指导者，您的专业知识、经验和耐心对我产生了深远的影响。您鼓励我追求自己的研究兴趣，并在我遇到困难时给予了我耐心的指导和解答。您的严谨治学态度、精湛的研究技巧以及对学术事业的热爱，都使我受益匪浅。我也要感谢您为我提供的宝贵建议和指导，这些将对我今后的人生发展产生长远的影响。

感谢评阅专家和答辩委员会的各位成员。感谢您在百忙之中抽出时间来审阅我的论文，并提出宝贵的意见和建议。

向我的家人和朋友们致以最衷心的感谢。感谢你们对我学业上的理解、支持和鼓励。你们的陪伴和鼓励是我坚持下去的最大动力，是我在学术和人生道路上的坚实后盾。感谢我的父母对我的养育之恩，我的母亲总是会无微不至地照顾我，我的父亲总是像我的朋友一样陪伴我。我人生中的一大遗憾，是父亲在我即将完成学业前离开了人世，但我在心中向你保证我不仅不会消沉，而且活出两倍的幸福，带着你的那份。感谢生活中陪伴我的女朋友，既给我的生活带来了许多快乐，又愿意与我分担悲伤，帮我化解许多消极的情绪，给了我许多理解、鼓励与支持。

感谢我的同门师兄姐妹们。在这段学术旅程中，我们一起探讨问题、互相学习、共同成长。你们的友谊和团结使我的研究生生活更加丰富多彩，我将永远珍视与你们的共同经历。感谢秦鹏亮师兄如同兄长一般的照顾，感谢孙宝良、李锐、李怀宇、曹庆、邓峙崇、李尚鸿师兄在学习上指导我，陪我健身、打羽毛球，李锐师兄还送了我一把吉他，是我人生中数得上号的贵重礼物。感谢同门的任文昊、张旭杭、刘子铭、陶政、王建昊、殷特尔、陶佳宇、丰瑞雪、黄思思，有机会能与你们并肩前行是我的荣幸，感谢你们给了我许多帮助和快乐，也让我从优秀的大家身上学到了很多。

感谢实习工作期间的同事们，特别是我的前辈霍钊，你既给予了我充分的肯定和信任，也帮助我应对了很多困难的问题。让我有了更多自信，更收获成长。未来我们仍将在一个行业内并肩前行，希望我们来日方长。

最后，想对二十年求学生涯中的自己说，谢谢你们每一次面对困难都没有放弃，从今往后我仍然不会惧怕生活中的任何困难。感谢祖国每一寸养育我的土地，祝愿祖国繁荣昌盛。

2021年5月
于哈尔滨工业大学