

分类号: TP391

密 级: 公开

学校代码: 10363

学 号: 2220910114



安徽工程大学

Anhui Polytechnic University

硕士学位论文

题 目 面向复杂场景的自动驾驶
 决策方法研究

论 文 作 者 孙梦婷
校 内 导 师 李钧 副教授
校 外 导 师 朱刚 副总经理
专业学位类别 计算机技术
研究方向(领域) 智能系统与应用

论文提交日期: 2025 年 5 月 9 日

分类号：TP391

密 级：公开

单位代码：10363

学 号：2220910114

题 目 面向复杂场景的自动驾驶
决策方法研究

英文并列
题 目 **Research on decision-making method for
Autonomous Driving in complex scenes**

论 文 作 者 ： 孙梦婷

校 内 导 师 ： 李钧 副教授

校 外 导 师 ： 朱刚 副总经理

专 业 学 位 类 别 ： 计算机技术

研究方向（领域）： 智能系统与应用

论文答辩日期：2025 年 5 月 10 日

安徽工程大学学位论文原创性声明

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

学位论文作者签名：孙梦婷

日期：2015 年 5 月 9 日

安徽工程大学学位论文版权使用授权书

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

保密 ☐，在_____年解密后适用本版权书。

本学位论文属于

不保密 ☒。

学位论文作者签名：孙梦婷

日期：2025年5月9日

指导教师签名：李娟

日期：2025年5月9日

面向复杂场景的自动驾驶决策方法研究

摘 要

自动驾驶决策方法一直备受研究者们的关注，因其在提升交通效率、减少交通事故以及推动智能交通系统发展方面具有重要价值。随着人工智能、深度学习、强化学习等技术的快速发展，自动驾驶决策方法也在不断优化。现有的自动驾驶决策方法大多数聚焦于单一目标优化，如安全性或效率性，或简单场景下的决策问题。然而，在现实世界中，这些现有的方法不足以解决所有问题，自动驾驶决策方法仍然面临着诸多挑战。

决策方法在自动驾驶领域面临的挑战主要体现在以下三个方面：一是复杂交通场景问题。复杂交通场景对不确定性建模的要求更高。二是周围车辆运动轨迹影响车辆决策问题。周围车辆的未来轨迹通常具有多种可能性，这些不确定性很大程度上会影响车辆的决策过程。三是多目标优化问题。这个问题在换道决策方法中尤为重要，自动驾驶决策方法需要在安全性、效率性和舒适性等多个目标之间实现平衡。

本文以面向复杂场景的自动驾驶决策方法为研究对象，针对影响复杂环境下自动驾驶车辆决策的关键因素开展了研究，论文的主要工作内容如下：

(1) 针对周围车辆运动轨迹对自动驾驶车辆决策的影响问题，提出了一种基于 **Informer** 的多模态轨迹预测模型框架 (**MTPI**)。MTPI 通过结合自注意提取和 **ProbSparse Self-attention** 等操作，旨在捕捉周

围车辆运动轨迹中的长时间依赖关系，同时降低计算复杂度和内存占用。将 MTPI 方法与 CV (Constant Velocity)、LSTM+map(prior) 1-G,n-C、DiversityGAN 这三种轨迹预测方法进行了比较，实验结果表明，提出的方法 MTPI 在轨迹预测的精度和降低复杂度方面取得了良好的性能。

(2) 针对复杂交通场景下车辆换道决策问题，提出了基于强化学习的多目标车辆换道决策优化方法，命名为 MOMA2C。该方法以 A2C 强化学习算法为基础框架，综合考虑安全性、舒适性和效率性等多重优化目标，实现对自我车辆的换道决策。在 MOMA2C 框架中，Actor 网络通过动作预测和概率采样生成策略，Critic 网络评估状态价值指导 Actor 更新。两者交互优化，形成闭环，实现换道决策在安全性、舒适性和效率性上的多重优化。将 MOMA2C 与基于规则的 DQN 方法在相同实验环境下进行了对比实验，实验结果表明，MOMA2C 在换道决策问题方面取得了较好的效果。

关键词：自动驾驶，复杂场景，轨迹预测，换道决策，Informer 模型

RESRARCH ON DECISION-MAKING METHOD FOR AUTONOMOUS DRIVING IN COMPLEX SCENES

ABSTRACT

Autonomous driving decision-making methods have garnered significant attention from researchers due to their critical value in enhancing traffic efficiency, reducing traffic accidents, and advancing the development of intelligent transportation systems. With the rapid development of technologies such as artificial intelligence, deep learning, and reinforcement learning, the decision-making methods for autonomous driving are being continuously optimized. Most existing decision-making methods for autonomous driving focus on single-objective optimization, such as safety or efficiency, or address decision-making problems in relatively simple scenarios. However, in the real world, these existing methods are not enough to solve all problems, and autonomous decision-making methods still face many challenges.

The challenges of decision-making methods in the field of autonomous driving are mainly reflected in the following three aspects: First, the problem of complex traffic scenes. Complex traffic scenarios have higher requirements for uncertainty modeling. Second, the surrounding vehicle trajectory affects the vehicle decision. The future

trajectories of surrounding vehicles often exhibit multiple possibilities, and these uncertainties can significantly influence the decision-making process of the vehicle. The third is the multi-objective optimization problem. The multi-objective optimization problem is particularly critical in lane-changing decision methods, as it necessitates a careful balance among various objectives, including safety, efficiency, and comfort, which are essential for autonomous driving.

In this paper, the decision-making method for autonomous driving in complex scenes is taken as the research object, and the key factors affecting the decision-making of autonomous driving vehicles in complex environments are studied. These key factors are primarily reflected in two aspects: firstly, the impact of surrounding vehicle trajectories on the decision-making of autonomous driving vehicles, and secondly, the challenge of lane-changing decisions. Centered around these two issues, this paper conducts in-depth research. The main contributions of this work are outlined as follows:

(1) A Multi-modal trajectory prediction model framework (MTPI) based on Informer is proposed to solve the impact of surrounding vehicle trajectories on autonomous vehicle decision-making. By integrating self-attention extraction with probspare self-attention operations, MTPI achieves a dual advantage: it effectively captures long-term dependencies within the trajectories of surrounding vehicles while simultaneously

reducing computational complexity and memory consumption. Compared with three trajectory prediction methods such as CV (Constant Velocity), LSTM+map(prior) 1-G,n-C and DiversityGAN, the experimental results show that the proposed method MTPI has achieved good performance in terms of trajectory prediction accuracy and complexity reduction.

(2) Aiming at the problem of vehicle lane changing decision in complex traffic scenarios, this paper proposed a multi-objective vehicle lane changing decision optimization method based on reinforcement learning, named MOMA2C. The A2C reinforcement learning algorithm was used as the basic framework, and multiple optimization objectives such as safety, comfort and efficiency were comprehensively considered to realize the lane changing decision for the ego vehicle. In the MOMA2C framework, the Actor network generates policies through action prediction and probability sampling, and the Critic network evaluates the state value to guide the Actor update. The two optimization interactions form a closed loop, and realize the multiple optimization of lane-changing decision in safety, comfort and efficiency. MOMA2C was compared with rule-based DQN method in the same experimental environment. The experimental results show that MOMA2C achieves better results in lane changing decision problem.

KEY WORDS: autonomous driving, complex scenes, trajectory prediction, lane changing decision, Informer model

目录

摘 要	I
ABSTRACT.....	III
第 1 章 绪论	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	4
1.2.1 自动驾驶轨迹预测方法研究现状.....	4
1.2.2 自动驾驶换道决策方法研究现状.....	6
1.3 主要研究内容.....	9
1.4 论文组织结构.....	11
第 2 章 相关理论基础	12
2.1 自动驾驶技术概述.....	12
2.1.1 车辆轨迹预测.....	13
2.1.2 车辆换道行为.....	14
2.2 时间序列预测模型.....	15
2.2.1 循环神经网络.....	15
2.2.2 长短期记忆网络.....	17
2.2.3 Transformer 模型.....	18
2.3 强化学习基础理论.....	21
2.3.1 马尔可夫决策过程.....	22
2.3.2 深度强化学习.....	24
2.4 数据集及仿真模拟平台.....	25
2.4.1 数据集.....	25
2.4.2 仿真模拟平台.....	27
2.5 本章小结.....	30
第 3 章 基于 Informer 模型的多模态轨迹预测方法研究	31
3.1 多模态轨迹预测问题概述.....	31
3.2 研究目标.....	33

3.3 Informer 模型	33
3.4 基于 Informer 的多模态轨迹预测模型框架	36
3.4.1 问题建模.....	36
3.4.2 模型框架.....	37
3.4.3 特征处理模块.....	38
3.4.4 MTPI-encoder.....	39
3.4.5 MTPI-decoder.....	41
3.5 实验与分析.....	42
3.5.1 实验设置.....	42
3.5.2 评价指标.....	42
3.5.3 实验结果与分析.....	43
3.6 本章小结.....	47
第 4 章 基于强化学习的多目标车辆换道决策优化方法研究	48
4.1 车辆换道问题概述.....	48
4.2 研究目标.....	49
4.3 基于强化学习的多目标车辆换道决策优化方法.....	49
4.3.1 问题定义.....	49
4.3.2 车辆换道环境建模.....	50
4.3.3 换道决策模型设计.....	50
4.3.4 换道决策问题要素设计.....	53
4.4 实验与分析.....	57
4.4.1 SUMO 仿真实验环境	57
4.4.2 评估指标.....	58
4.4.3 实验结果分析.....	58
4.5 本章小结.....	63
第 5 章 总结与展望	64
5.1 工作总结.....	64
5.2 工作展望.....	64
参考文献	66

攻读学位期间发表的学术论文目录	74
攻读学位期间比赛获奖目录	74
攻读学位期间参与科研项目	74
致谢	75

图录

图 1-1 全国交通事故发生情况图	1
图 1-2 中国自动驾驶产业政策发展规划图	2
图 1-3 国内外企业自动驾驶车辆	3
图 1-4 论文研究思路框架图	10
图 2-1 自动驾驶分级标准示意图	12
图 2-2 轨迹预测算法输入表示	13
图 2-3 轨迹预测基本流程图	14
图 2-4 强制换道与自由换道对比图	15
图 2-5 RNN 神经网络结构图	16
图 2-6 LSTM 神经网络结构图	17
图 2-7 Transformer 模型结构图	19
图 2-8 Transformer 模型输入示例图	20
图 2-9 Transformer 模型编码器示例图	20
图 2-10 Transformer 模型解码器示例图	21
图 2-11 强化学习示意图	22
图 2-12 马尔可夫链示例图	23
图 2-13 Actor-Critic 框架图	25
图 2-14 SUMO 路网绘制方法	27
图 2-15 CARLA 城市环境示例图	28
图 2-16 MetaDrive 驾驶场景示例图	29
图 3-1 交通场景示例图	32
图 3-2 Informer 模型结构图	34
图 3-3 交互细节图	36
图 3-4 基于 Informer 的多模态轨迹预测模型框架	38
图 3-5 潜变量自回归模型示意图	39
图 3-6 特征细节图	39
图 3-7 MTPI-encoder 框架图	41

图 3-8 MTPI-decoder 框架图.....	42
图 3-9 十字路口直行轨迹预测可视化结果图	45
图 3-10 十字路口左转轨迹预测可视化结果图	45
图 3-11 十字路口右转轨迹预测可视化结果图	46
图 3-12 高速公路匝道直行轨迹预测可视化结果图	46
图 4-1 城市道路换道场景示例图	49
图 4-2 多车道环形路场景建模图	50
图 4-3 基于 A2C 的换道决策模型结构图.....	51
图 4-4 MOMA2C 网络框架图.....	52
图 4-5 SUMO 环形路交通建模图.....	58
图 4-6 累积奖励变化趋势图	59
图 4-7 A2C 与 DQN 速度效率对比图	60
图 4-8 车辆速度变化图	61
图 4-9 rlagent 与 vehicle_2 的行驶速度对比图.....	62
图 4-10 rlagent 与 vehicle_2 的航向角对比图.....	62
图 4-11 rlagent 与 vehicle_2 加速度变化图.....	63

表录

表 2-1 主流自动驾驶数据集对比表26

表 3-1 Informer 模型优化表35

表 3-2 不同模型的预测性能对比44

表 4-1 基于 A2C 的换道决策模型算法流程.....53

表 4-2 不同方法在换道决策中的结果对比表59

第 1 章 绪论

1.1 研究背景及意义

在我国科技迅猛发展的推动下，人民生活水平显著提升，幸福感和获得感不断增强。随之而来的是城市化进程的加快，城区道路网络覆盖范围不断扩大。汽车作为现代化生活的重要标志，已走进千家万户，在城市道路覆盖范围不断扩大的同时，私家车的保有量也在持续增长。根据公安部最新数据显示^[1]，截至 2024 年，我国机动车保有量再创新高，总量突破 4.53 亿辆，其中汽车占比显著，达到 3.53 亿辆。与此同时，机动车驾驶人规模持续扩大，总数攀升至 5.42 亿人，其中汽车驾驶人以 5.06 亿人占据主导地位。2024 年全年，全国新增机动车注册登记数量达 3583 万辆，新增驾驶人数量也达到 2226 万人，充分展现了我国汽车产业的蓬勃发展和居民出行需求的持续增长。然而，随着车辆数量的快速增长，城市交通在早晚高峰时段面临严重的拥堵问题，人类驾驶员在长时间等待和频繁启停中容易产生疲劳和焦躁情绪，从而增加驾驶风险^[2]。同时，随着城市道路复杂度的不断提升，交通标志、信号灯、行人过街设施等元素的密集分布，使得驾驶员需要在短时间内处理大量信息，极易出现注意力分散和决策失误，进而导致交通事故的发生^[3]。

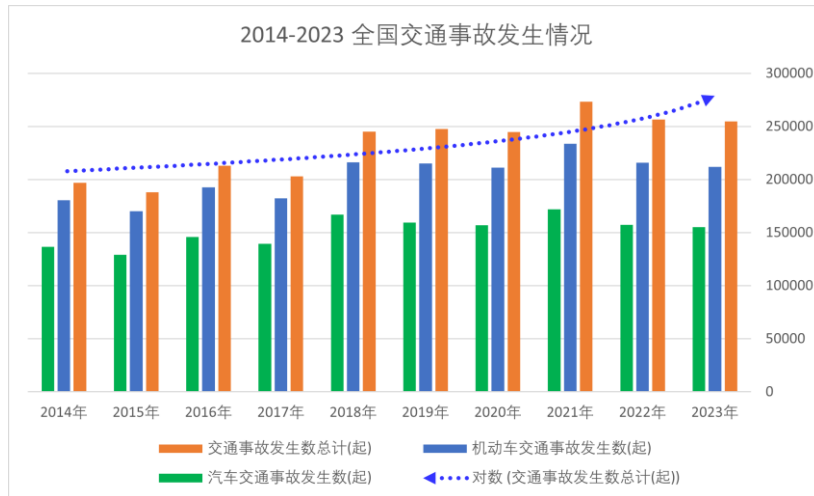


图 1-1 全国交通事故发生情况图

Figure 1-1 Traffic accidents nationwide

（注：数据来源为中华人民共和国国家统计局）

如图 1-1 所示, 自 2014 年以来, 我国交通事故发生总数总体呈现波动上升趋势, 从 2014 年的 196,812 起增长至 2023 年的 254,738 起, 增幅达 29.4%。其中, 机动车交通事故发生数从 2014 年的 180,321 起上升至 2023 年的 211,974 起, 增长 17.5%; 而汽车交通事故发生数则从 2014 年的 136,386 起增至 2023 年的 155,158 起, 增长 13.8%。交通事故的频发不仅显著降低了道路通行效率, 还对城市交通安全构成了严重威胁。因此, 自动驾驶技术的发展对缓解城市交通拥堵、降低交通事故发生率具有重要现实意义。

自动驾驶技术是一种基于多模态感知、智能决策与精准控制的综合性技术体系, 其核心是通过感知模块实时采集车辆周围环境信息, 为车辆所处的动态场景建模^[4]。20 世纪 80 年代初期, 人们就已经开始了对自动驾驶技术的研究^[5]。例如, 美国 DARPA 通过资金支持和举办竞赛推动自动驾驶技术研发、日本鼓励车企开展国际合作^[6]; 中国 2015 年发布《中国制造 2025》^[7], 将自动驾驶列为重点发展领域。此外, 我国高度重视自动驾驶技术的发展, 近年来持续出台多项支持政策^[8], 如图 1-2 所示。

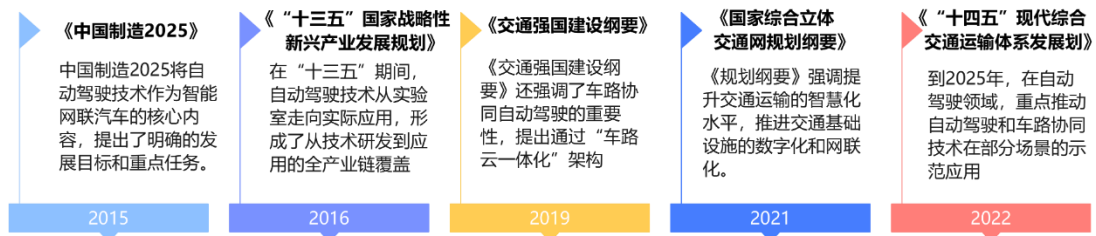


图 1-2 中国自动驾驶产业政策发展规划图

Figure 1-2 China's autonomous driving Industry Policy development Plan

此外, 很多国内外企业也在进行自动驾驶技术的研究。例如: Tesla^[9]自研的 Dojo 超级计算机专注于自动驾驶模型的训练, 所有 Tesla 车型均标配 Autopilot 系统; Waymo^[10]开发了完整的自动驾驶技术栈, Waymo Via 专注于自动驾驶货运; 百度 Apollo^[11]推出了无人驾驶巴士服务和自动驾驶出租车 (Robotaxi); 蔚来汽车推出了 NAD 的订阅服务, 蔚来 ET7、ET5 等车型已搭载 NAD 系统, 支持高级辅助驾驶功能。以下是国内外企业研究的一些自动驾驶车辆, 如图 1-3 所示:



图 1-3 国内外企业自动驾驶车辆

Figure 1-3 Domestic and foreign enterprises self-driving vehicles

自动驾驶对复杂场景的理解是确保系统安全性和可靠性的核心挑战之一。复杂场景通常涉及高动态性、不确定性、多目标交互等特征，需要自动驾驶系统具备强大的感知、预测、决策和规划能力。这三个特性相互耦合，共同构成了复杂场景的本质特征。其中，道路拓扑结构复杂性和移动实体行为模式是关键影响因素，具体体现在交通密度和动态障碍物上。

交通密度可通过车辆数与道路长度的比例衡量，分为三个等级：低密度（ <15 辆/km），车辆间距大，决策时间充裕；中密度（ $15-30$ 辆/km），需频繁调整车速和跟车距离；高密度（ >30 辆/km），车辆间距小，对实时决策和精准控制要求极高。动态障碍物分为三类：行人（行为随机性强，可能突然横穿道路）；非机动车（运动轨迹和速度复杂，尤其在城市混合交通中）；其他机动车（运动规律相对可预测，但仍存在突发行为，如变道、急刹）。

人工智能技术的飞速发展在促进自动驾驶技术发展的同时，也带来了一些挑战。当前，自动驾驶技术的研究框架主要围绕三个方面：环境感知层、决策规划层和控制执行层^[12]。这三个层次相互协同，共同构成自动驾驶系统的完整技术链条。本文的研究重点聚焦于自动驾驶技术中的决策规划层。在现实世界中，想要良好实现自动驾驶技术还需要很多技术的突破，尤其是周围交通参与者的轨迹预测和自我车辆的决策^[13]。其中，周围交通参与者与复杂的交通环境会影响自动驾驶车辆决策模型的判断；其次，决策过程需同时优化安全性、效率和舒适度等多

重目标,而这些目标在某些场景下可能存在冲突,如何实现平衡是一大难题;最后,系统需有效应对环境动态变化及其他交通参与者行为的不确定性,这对模型的鲁棒性和适应性提出了更高要求。解决这些挑战是推动自动驾驶技术实用化的关键。因此,开展面向复杂场景下的自动驾驶决策技术研究具有重要意义。

1.2 国内外研究现状

本文所研究的自动驾驶决策方法属于自动驾驶技术中的决策规划层。自动驾驶车辆在进行决策时,需要考虑复杂的交通环境以及周围交通参与者的轨迹动向,然后结合自我车辆的状态去生成决策策略。多模态轨迹预测可以帮助自我车辆生成更好的驾驶决策策略。本研究从自动驾驶决策方法出发,对多模态轨迹预测研究方向和换道决策研究方向的国内外研究现状进行了分析与总结。

1.2.1 自动驾驶轨迹预测方法研究现状

多模态轨迹预测(Multimodal trajectory prediction, MTP^[14])方法可分为基于传统统计学模型的 MTP 方法和基于深度学习的 MTP 方法。基于传统统计学模型的 MTP 方法主要依赖于概率统计和时间序列分析技术,如卡尔曼滤波、自回归模型、移动平均模型等。基于深度学习的 MTP 方法利用神经网络模型,如 Transformer 模型等对多模态数据进行建模和预测。

(1) 基于传统统计学模型的 MTP 方法

基于传统统计学模型的 MTP 方法主要通过数学模型和传统机器学习方法对车辆的运动状态进行建模,以预测其未来可能的轨迹。现有的主要方法包括:

Helgo Dyckmanns 等人^[15]提出了一种用于城市交叉口目标跟踪的扩展滤波方法。该方法通过通过将先验背景知识纳入滤波过程,可以更早地识别机动,并且可以避免错误的机动检测。

Thomas Streubel 等人^[16]提出了一种基于隐马尔可夫模型(HMMs)的预测框架。该模型引入了三种不同的 HMMs,每一种都有一种可能的方向,使用 Baum-Welch 算法选择序列来学习 HMM 参数。

Mattias Brännström 等人^[17]提出了一种基于 Linear Bicycle Model 的威胁评估算法,用于估计驾驶员如何通过转向、制动或加速避免与任意物体碰撞。算法中,车辆运动由 Linear Bicycle Model 描述,车辆外形用矩形表示,目标物体外形用

可变多边形表示。通过建模、恒定加速度（CA）和近似处理潜在避撞操作，推导出解析表达式以估计可行的避撞操作集合，并评估是否需要驾驶员协助。

Dietmar Kasper 等人^[18]提出了一种基于面向对象贝叶斯网络（OOBNs）的新方法，用于识别高速公路场景中的典型驾驶操作。此外还引入了与车道相关的坐标系统和针对所有车辆的个体占用网格，有效分类交通场景中的车辆-车道和车辆-车辆关系，增强对复杂交通的理解，为 MTP 任务奠定了良好的基础。

总体来说，基于传统统计学的车辆多模态轨迹预测模型有着坚实的理论基础，易于理解和实现。但是这种方法往往高度依赖于训练数据的质量和数量，对数据分布的假设可能不总是成立。此外，在需要快速响应的实时系统中，传统统计方法可能无法满足实时性要求。

（2）基于深度学习的 MTP 方法

在 MTP 方法研究中，需要研究有效建模时间与空间维度之间的交互关系，因为交通参与者的运动轨迹不仅依赖于当前空间状态，还受到时间维度上历史行为与未来意图的影响。深度学习方法拥有强大的非线性建模能力，能够同时捕捉时间序列中的动态变化和空间结构中的复杂关系，从而在 MTP 任务中表现出显著优势。现有的主要方法有：

Henggang Cui 等人^[19]提出了一种新方法，该方法通过将参与者周围环境编码为栅格图像，并利用深度卷积网络自动提取相关特征，以处理交通行为的不确定性，最终用于预测自动驾驶车辆周围交通参与者的多种可能轨迹及其概率；Lei Li 等人^[20]提出了本文提出了一种针对非结构化道路的轨迹预测方法。首先，通过建模场景上下文和编码目标车辆的历史位置，捕捉环境与车辆运动的关联性；其次，采用基于 CNN 的轨迹预测模型，结合目标车辆运动的不确定性，生成多条可能的未来轨迹及其概率分布。

Nachiket Deo 等人^[21]提出了一种基于 LSTM 的模型，用于预测高速公路上周围车辆的运动。该模型为车辆的操作分配置信值，并输出基于这些操作的未来运动的多模态分布；Long Xin 等人^[22]提出了一种使用双长短期记忆（LSTM）网络对周围车辆进行长距离轨迹预测的算法，该算法能够有效提高强交互驾驶环境中的预测精度。与需要轨迹匹配和手动特征选择的传统方法相比，该方法可以通过序列学习从自然驾驶数据中自动学习驾驶员行为的高级时空特征。

Francesco Giuliani 等人^[23]提出了一种基于 Transformer 网络的轨迹预测新方法，与传统的 LSTM 模型相比，Transformer 采用注意力机制而非顺序处理。该方法使用了原始的 Transformer 网络（TF）和先进的双向 Transformer（BERT）来预测场景中个体的轨迹。Yicheng Liu 等人^[24]提出了一个新颖的多模态轨迹预测的 Transformer 框架，称为 mmTransformer。整个模型可以看作是堆叠的编码器解码器结构，其中过去的轨迹、道路信息和社会互动通过几个编码器和解码器模块分层地聚集在一起。

Xinshuo Weng 等人^[25]提出了一种并行化的模块化自动驾驶架构，名为 PARA-Drive。通过系统地探索模块化自动驾驶架构的设计空间，包括模块的必要性、位置和信息流动等关键方面，提出了 PARA-Drive 架构，并展示了其在感知、预测和规划性能方面的优越性；Kaituo Feng 等人^[26]提出了一种新的框架，命名为 Plankton。首个针对端到端运动规划器压缩的知识蒸馏框架，旨在解决深度神经网络模型过大导致的计算资源瓶颈问题。

综上所述，基于传统统计学模型的 MTP 方法：适用于简单场景，计算效率高，但对复杂环境的适应性有限；基于深度学习的 MTP 方法：适用于复杂场景，能够处理非线性和多模态数据，但计算资源需求大，往往需要增加模型的深度来提高预测效果。

1.2.2 自动驾驶换道决策方法研究现状

自动驾驶换道（Automatic driving lane change, LC^[27]）决策方法可分为基于规则的 LC 方法和基于学习的 LC 方法。基于规则的方法依赖于预先定义的一系列明确的规则，这些规则通常包括交通法规、车辆操作规范以及预先设计的逻辑条件。基于学习的方法利用机器学习或深度学习等技术，通过大量的数据训练来优化 LC 策略。

（1）基于规则的 LC 方法

基于规则的 LC 方法是一类通过预定义逻辑和条件判断来实现换道决策的方法，这类方法通常依赖于专家知识或交通规则。

Gipps 模型^[28]是 Gipps, Peter G.在 1986 年提出的换道决策模型，这也是换道模型最早被提出。该模型主要基于有障碍（如信号灯、障碍物等）的情况，将换道行为分为三个阶段：产生意图、探测条件和动作实施。Gipps 模型是基于驾驶

员以最高速度行驶的原则，以确保在突然刹车时不会发生碰撞。Leonhard Lücken^[29]认为如果跟随车辆的制动强度明显高于领导车辆，即使跟随车辆对领导车辆的最大减速度的估计，该模型也可能产生不安全行为。所以他在 Gipps 模型上做出了改进，具体来说，原始 Gipps 模型的安全原则仅要求两车在完全停止时的假设车头间距大于零。改进后的模型将安全原则扩展到假设制动过程的所有时间点，以避免轨迹相交导致的碰撞。

有限状态机（Finite State Machine, FSM）是一种常用的基于规则的换道决策方法，通过定义一系列状态和状态转移条件来实现决策。Yufeng Hu 等人^[30]基于 FSM 提出了一种新的决策模型，该模型通过设定控制点来控制变道路径，采用横向和纵向控制方法进行路径跟踪；Sang-Hyeon Bae 等人^[31]提出了一种由感知、规划和控制三部分组成的车辆系统。其中动作规划部分是基于运动有限状态机（M-FSM）和控制有限状态机（C-FSM）实现的。C-FSM 表示控制动作：停止，紧急停止，加速，减速等；M-FSM 表示运动动作：保持车道，变道，转弯，右转，左转等。利用 M-FSM 和 C-FSM 进行规划，推导出适合驾驶情况的动作。

基于规则的启发式方法（Rule-Based Heuristic Methods）通过预定义的启发式规则来判断是否执行换道。常见规则有：安全距离规则：确保换道过程中与周围车辆保持安全距离；速度优势规则：如果目标车道的平均车速显著高于当前车道，则触发换道；交通流规则：根据交通流密度和车道占用情况选择最佳换道时机。Alexander Kanaris 等人^[32]提出了一种计算变道最小安全间距（MSSLC）的算法，用于确定车辆在合并或变道时应保持的间距，以避免紧急制动时发生碰撞；Harry Chia-Hung Hsu 等人^[33]提出了两种 PLC 机动，分别是 leader and predecessor PLC。其中 predecessor PLC 控制着两种情况。predecessor-I PLC 使用最小安全距离（MSS），适用于车辆间距较小的场景，依赖通信的及时性来确保安全。predecessor-II PLC 适用于车辆间距较大的场景，通过更大的安全间距来降低通信延迟带来的风险。

基于代价函数的方法（Cost Function-Based Methods）通过定义代价函数来评估换道的收益和风险，选择代价最小的换道策略。代价函数设计主要从以下三方面考虑：安全性代价：与周围车辆的碰撞风险；效率代价：换道后预计的行驶时间或油耗；舒适性代价：换道过程中的加速度和转向角变化。Muhua Guan 等人

[34]提出了一种基于模糊推理的分类驾驶员换道决策模型，并通过驾驶实验确定了隶属函数。为了适应不同驾驶员的驾驶习惯和偏好，模型被分为三种类型：激进型、中性型和保守型；Wen Hu 等人[35]提出了一种概率决策和轨迹规划框架。该模型主要从以下三个方面进行改进：驾驶决策过程分解：将驾驶决策分为意图生成和可行性评估两个阶段，分别利用效用理论和风险评估来实现；攻击性指数（AI）：提出并量化了攻击性指数，用于推断换道操作的不对称风险水平，以反映重卡对周围车辆的较大风险；轨迹规划约束：在规划阶段，将横向和侧倾动力学稳定性作为约束条件，排除可能导致车辆不稳定的候选轨迹。

（2）基于学习的 LC 方法

基于学习的自动驾驶换道决策方法是一类通过数据驱动和利用机器学习或深度学习算法从大量驾驶数据中学习换道策略的方法。

图神经网络通过建模交通参与者之间的交互关系，学习换道决策。此种方法的特点是：图结构：将交通参与者（车辆、行人）表示为图中的节点，将其交互关系表示为边；信息传递：通过图神经网络聚合节点信息，学习全局交通场景的表示。Zihao Wen[36]等人提出了一种环境表示，即时间占用流图（TOFG）。具体来说，基于占用流的表示将地图信息和车辆轨迹统一为同质数据格式，并实现一致的预测。车辆间的时间依赖关系有助于及时捕捉占用流的变化，从而进一步提高模型的性能。为了证明 TOFG 能够简化模型结构，TOFG 与基于简单图注意力（GAT）的神经网络相结合，提出了 TOFG-GAT，可以用于轨迹预测和运动规划。

强化学习通过智能体与环境的交互学习最优策略，适用于换道决策的序列决策问题。此种方法主要通过设计状态空间、动作空间以及奖励函数来完成换道决策。主要方法包括：

Tianyu Shi[37]等人提出了一种基于分层强化学习的变道决策和控制体系结构。该方法将决策和控制过程分为两个相互关联的过程：何时进行变道机动和如何进行变道机动。具体而言，首先在考虑安全的基础上应用深度 q 网络决定何时进行机动；然后，设计了一个带有二次逼近器的深度 Qlearning 框架，用于决定如何在纵向上完成机动。

Fei Ye 等人[38]提出了一种基于安全近端策略优化(PPO)的深度强化学习方法，该方法将策略与安全干预模块相结合。PPO 算法与经典算法 TRPO 相比，可以减

少计算非线性共轭梯度的负担，此外，PPO 在采样和学习策略方面比 TRPO 更有效。

Branka Mirchevska 等人^[39]提出了 AQL-MAP 算法，这是一种基于模型的动作估计取代学习到的建议分布的方法，该方法在整个连续动作空间上平摊最大化的代价，通过提出一组有限的可达动作来最大化。尽管提议的动作集是有限的，但它涵盖了智能体在给定状态下可以执行的所有安全和可行的动作。

Cenxin Yu 等人^[40]采用深度确定性策略梯度方法（DDPG）建立了一种可以同时控制车辆横向和纵向运动的变道模型。这个变道模型的具体过程如下：DDPG 模型根据 SV（变道车）的当前状态给出动作，并通过安全修正模型修正动作值以避免碰撞，SV（变道车）在完成安全动作后到达下一个状态，直到换道结束。

基于规则的方法逻辑清晰、易于理解和实现，实时性好，可解释性强。然而，它灵活性差，难以适应复杂和动态的交通环境，且难以处理不确定性和优化规则。相比之下，基于学习的方法通过数据驱动的方式自动优化决策策略，能够适应复杂的交通场景，处理复杂任务能力强，但需要大量数据和计算资源，可解释性差，训练过程不稳定。然而学习类方法的可解释性和稳定性仍面临挑战，如复杂场景下的实时性难以满足自动驾驶的需求以及如何有效建模和处理交通环境中的不确定性。未来的研究需要在这些方面取得突破，以推动自动驾驶技术的进一步发展。

1.3 主要研究内容

近年来，自动驾驶车辆及其相关技术正逐步从实验室走向实际应用，日益融入人们的日常生活。随着自动驾驶技术的快速发展和商业化落地，其普及程度不断提高，但与此同时，公众对其安全性、舒适性以及应对复杂场景能力的关注也日益增加。作为自动驾驶系统的核心组成部分，决策方法直接关系到车辆在动态交通环境中的行驶安全与效率，尤其在面对复杂多变的城市道路、突发交通事件以及多车交互场景时，如何实现智能、可靠且人性化的决策成为待解决的关键问题。因此，本文聚焦于面向复杂场景的自动驾驶决策方法研究，旨在通过创新算法设计和多目标优化，提升自动驾驶系统在复杂环境下的决策能力。

通过分析自动驾驶轨迹预测方法和自动驾驶换道决策方法的国内外研究现状，针对其面临的挑战，从周围车辆影响换道决策和车辆的的不同行驶意图的角度出发，提出相应的解决方案。论文研究思路框架图如图 1-4 所示。本文的主要研究内容包括：

（1）基于 **Informer** 模型的多模态轨迹预测方法研究。周围车辆的行驶轨迹是影响自动驾驶进行决策的一个主要因素，所以本文针对现有的车辆的轨迹预测问题展开了研究，通过分析多模态因素，其中包括周围车辆的位置，状态，历史轨迹等多模态信息，设计并实现了一种基于 **Informer** 模型的轨迹预测模型 **MTPI**，以提高轨迹预测是准确性，并且计算效率方面表现出良好性能。

（2）基于强化学习的多目标车辆换道决策优化方法研究。针对自动驾驶自由换道决策问题，现有的自动驾驶换道决策算法往往只注重换道本身而忽视了安全、舒适以及效率等因素，从而导致换道激进的问题。基于此，提出了一种基于强化学习的多目标车辆换道决策优化方法研究。

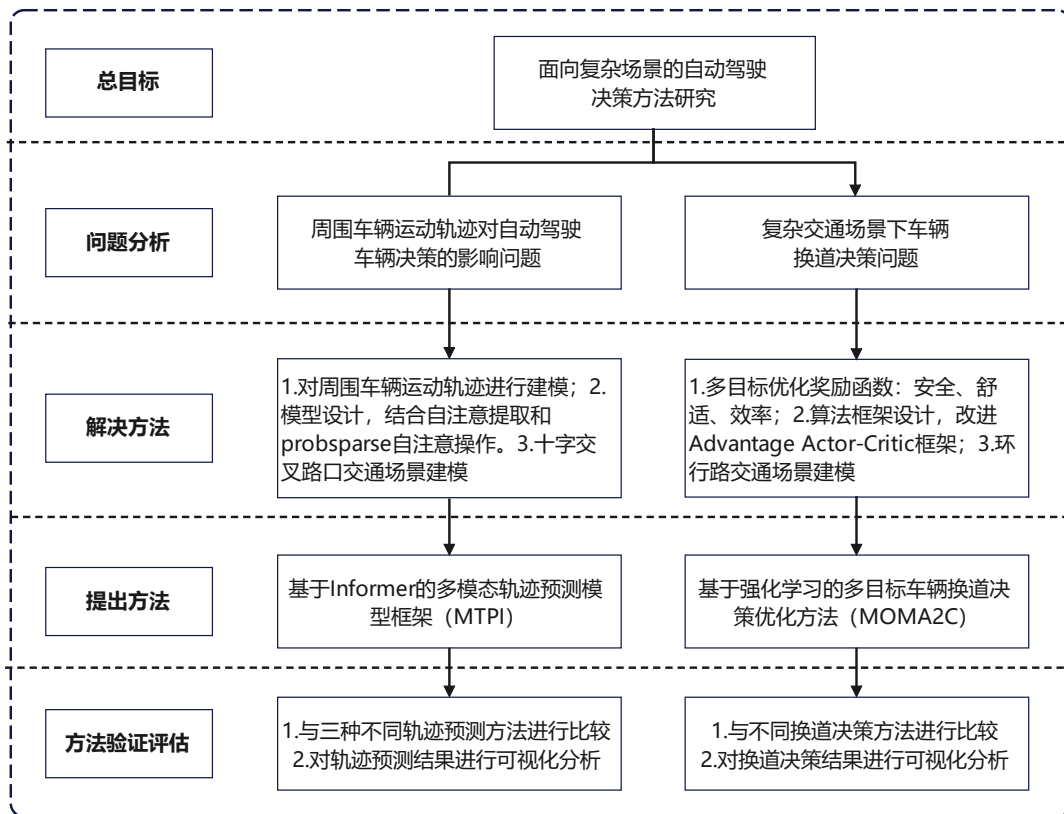


图 1-4 论文研究思路框架图

Figure 1-4 The frame diagram of the research idea of this paper

1.4 论文组织结构

本文针对面向复杂的自动驾驶决策问题，从两个方面进行了研究。一方面是影响自动驾驶进行决策的周围车辆的轨迹运行动向的问题，也就是对目标车辆的轨迹预测；另一方面是自我车辆在行驶过程中，需要考虑其他车辆的动态，然后进行决策的问题。下面是本文的结构安排：

第一章为绪论。首先介绍了自动驾驶的研究背景，介绍了自动驾驶技术的组成结构以及我国对自动驾驶技术发展的鼓励与支持。然后对自动驾驶轨迹预测方法和自动驾驶换道决策方法进行了国内外方法研究现状的分析与总结，最后阐明了本文的研究内容以及结构安排。

第二章为相关理论基础。首先介绍了自动驾驶技术的分级标准以及研究重点。接着介绍了自动驾驶任务中常用的深度学习知识，包括 Transformer 模型和 Informer 模型以及强化学习基础理论知识，并和监督学习做了对比。然后介绍了现在的主流自动驾驶数据集，包括 NGSIM 数据集、HighD 数据集、Argoverse 数据集、INTERACTION 数据集和 Argoverse 2 数据集。最后介绍了仿真模拟平台，其中包括交通仿真分类以及 SUMO 微观仿真模拟平台。

第三章为基于 Informer 模型的多模态轨迹预测方法研究。本章首先进行了多模态轨迹预测问题概述，指出了现有方法的不足。然后对 Informer 模型进行介绍。接着阐述了本章所设计的基于 Informer 的多模态轨迹预测模型框架。然后对多模态轨迹预测问题进行建模，详细描述了模型框架以及每个模块的作用。最后对实验结果进行了分析，并且对实验结果进行可视化分析。

第四章为基于强化学习的多目标车辆换道决策优化方法研究。首先描述了自动驾驶换道决策的问题定义，阐述了自由换道与强制换道的定义与区别。然后明确本章要研究的是自由换道问题，并针对此问题提出了基于强化学习的多目标车辆换道决策优化方法。该方法以 A2C 强化学习算法为基础框架，通过优化安全性、舒适性和效率性等多重优化目标，实现对自我车辆的换道决策。最后对提出方法进行了对比实验，并分析实验结果。

第五章为总结与展望，总结了本文的研究内容以及未来的研究方向。

第 2 章 相关理论基础

2.1 自动驾驶技术概述

自动驾驶技术是人工智能、传感器技术、高精度地图和车辆控制系统等多学科交叉的产物，旨在实现车辆在无需人类干预的情况下自主完成感知、决策、规划和控制等驾驶任务^[41]。其核心目标是通过技术手段提升交通安全性、效率和舒适性，同时减少人为操作失误和交通拥堵问题。早期的自动驾驶技术主要以辅助驾驶为核心，旨在通过智能化功能减少人们的操作负担，提升行车安全性和效率。随着人工智能、传感器技术和计算能力的飞速发展，自动驾驶技术取得了显著突破，逐步从辅助驾驶向高度自动化乃至完全自动化迈进^[42]。如图 2-1 所示，目前，L2/L3 级技术主要由驾驶员和系统共同操作，已实现商业化应用，L4 级技术则是完全依赖于系统执行，这只在特定场景中逐步落地，而 L5 级技术仍处于探索阶段^[43]。

等级	名称	操作执行	动态驾驶任务	驾驶情景	
L0	无自动化	驾驶员	驾驶员	无	SAE J3016
L1	辅助驾驶	驾驶员和系统	驾驶员	部分	
L2	部分自动化	系统	驾驶员	部分	
L3	有条件自动化	--	驾驶员	部分	
L4	高度自动化	--	--	部分	
L5	完全自动化	--	--	全部	
0级	应急辅助	驾驶员	驾驶员	有限制	中国分级 标准
1级	部分辅助驾驶	驾驶员和系统	驾驶员	有限制	
2级	组合辅助驾驶	驾驶员和系统	驾驶员	有限制	
3级	有条件自动驾驶	驾驶员和系统	动态驾驶任务接管用户	有限制	
4级	高度自动驾驶	系统	系统	有限制	
5级	完全自动驾驶	--	--	无限制	

图 2-1 自动驾驶分级标准示意图

Figure 2-1 Illustration of the classification standard for autonomous driving

现在自动驾驶技术研究的重点是解决复杂场景下的感知、决策和控制问题。本文的研究重点属于决策规划层。自动驾驶决策模块是自动驾驶系统的核心组成

部分之一，决策模块的性能直接影响到自动驾驶车辆的安全性、舒适性和通行效率。决策模块的主要功能是根据感知模块提供的环境信息和对其他交通参与者的行为预测，结合自车的状态，生成具体的驾驶策略。本文针对面向复杂环境的自动驾驶决策方法进行研究：一是周围车辆行驶轨迹的预测，为自动驾驶决策提供了重要的环境动态信息；二是多个车辆在同一路段影响自动驾驶车辆进行决策的协同问题。

2.1.1 车辆轨迹预测

周围车辆的行驶轨迹是影响自动驾驶进行决策的一个主要因素，所以对周围车辆的行驶轨迹预测是有必要的。准确预测周围车辆的行驶轨迹不仅能够帮助自动驾驶系统更好地理解交通环境的动态变化，还能为决策模块提供关键输入，从而生成安全、高效的驾驶策略^[44]。在复杂的交通场景中，周围车辆的行为往往具有多模态性和不确定性，如加速、减速、变道等。通过轨迹预测，自动驾驶车辆可以得知环境动态信息，然后进行更好的决策。

轨迹预测的目标是生成交通参与者未来一段时间内的运动轨迹。轨迹预测得到的结果是未来一段时间内的一种或者多种轨迹的可能性^[45]，这些预测结果能够为自动驾驶车辆提供关键的环境动态信息，可以辅助自动驾驶车辆进行更好的决策。在轨迹预测算法中，常见的输入表示方法主要包括基于车辆运动状态和基于地图^[46]的两种方式，如图 2-2 所示。基于车辆运动状态的表示通常是以自我车辆和周围车辆的各种状态信息，如位置、速度、加速度等信息作为输入；基于地图的则是通常将地图矢量化表示，构建车辆与车道之间的复杂依赖关系，从而更好地捕捉交通场景的全局上下文信息。

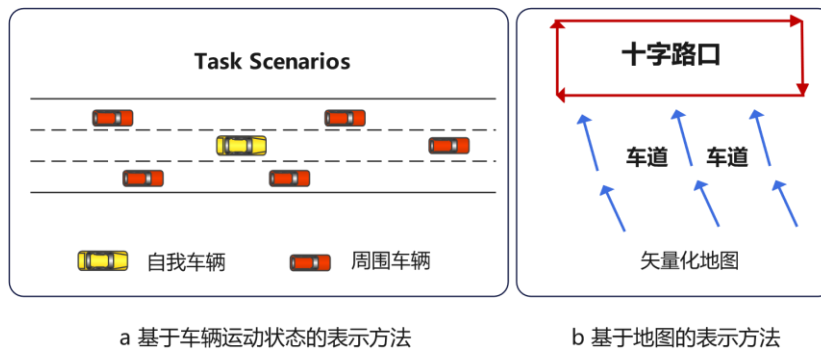


图 2-2 轨迹预测算法输入表示

Figure 2-2 Trajectory prediction algorithm input representation

自动驾驶轨迹预测的基本流程是一个从数据输入到结果输出的系统性过程。如图 2-3 所示，轨迹预测的基本流程包含四个大致步骤。首先是数据输入，包括基于车辆运动状态数据和基于地图数据的两种方式；接下来是特征提取阶段，主要目的是将原始数据转化为模型可用的特征表示；然后是模型预测阶段，这一阶段是轨迹预测的核心阶段，通过算法生成交通参与者的未来轨迹；最后是预测结果输出。

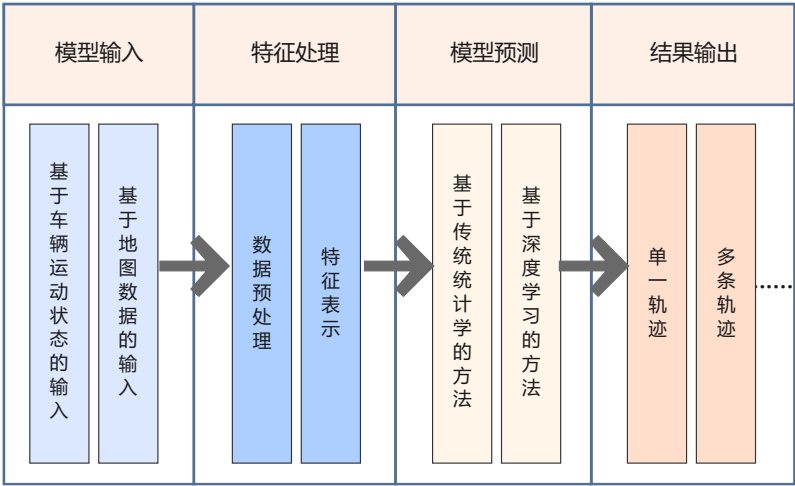


图 2-3 轨迹预测基本流程图

Figure 2-3 Basic flowchart of trajectory prediction

2.1.2 车辆换道行为

自动驾驶车辆的换道行为是指车辆在行驶过程中，由于周围交通参与者的行为变化，例如前方车辆减速或目标车道车辆加速，或者出于自身驾驶需求，例如提高通行效率或选择更优路径以更快到达目的地，而触发的一种驾驶策略^[47]。在自动驾驶系统中，换道行为的实现依赖于感知模块、预测模块、决策模块和控制模块的协同工作，其核心目标是在复杂交通场景中生成安全、高效且符合交通规则换道策略。

换道行为分为强制换道行为和自主换道行为。强制换道是指车辆由于外部条件或自身需求必须进行的换道行为，例如高速公路出口匝道前的换道。自由换道是指车辆出于优化行驶效率或驾驶舒适性等目的，主动选择进行的换道行为，例如多车道道路中为优化路径而进行的换道。强制换道与自由换道的对比如图 2-4 所示：

特征	强制换道	自由换道
触发条件	道路约束、紧急情况等	驾驶舒适性、通行效率
决策逻辑	必要性优先	优化目标优先
执行方式	快速执行	平稳执行
应用场景	高速公路匝道、紧急避让等	超车、选择畅通车道

图 2-4 强制换道与自由换道对比图

Figure 2-4 Comparison of forced lane change and free lane change

车辆换道行为通常可以分为四个主要阶段：换道需求识别、换道可行性评估、换道执行和换道完成^[48]。每个阶段都涉及不同的决策逻辑和控制策略。换道需求识别的主要任务是判断是否需要换道，通常是基于当前交通状况和驾驶意图，判断是否满足换道条件。如果满足条件，则进入下一阶段；否则，继续在当前车道行驶。换道可行性评估是换道行为的关键阶段，主要任务是判断换道是否安全可行。例如：如果目标车道后方车辆距离较远、速度较慢，且道路条件允许，则判断为可行；如果目标车道后方车辆距离较近、速度较快，或道路条件不允许，则判断为不可行。换道执行是换道行为的核心阶段，主要任务是将车辆从当前车道平稳、安全地转移到目标车道。换道完成是换道行为的最后阶段，主要任务是确保车辆完全进入目标车道并稳定行驶。

2.2 时间序列预测模型

时间序列预测模型是一种用于分析和预测时间序列数据的数学模型或算法，它能够根据历史数据捕捉到的时间依赖性，对未来数据进行预测。在自动驾驶场景下，这些模型可以对车辆的行驶轨迹、交通流量、行人行为等进行预测，从而帮助自动驾驶系统提前做出合理的决策，提高行驶的安全性和效率。下面将对常用的时间序列预测模型进行详细介绍。

2.2.1 循环神经网络

循环神经网络（Recurrent Neural Network，RNN^[49]）是一种用于处理序列数据的神经网络，其核心特点是具有循环结构，能够对序列中的每个元素进行处理，

并将前一个元素的处理结果传递给下一个元素。RNN 的结构如图 2-5 所示，其中， x 表示输入层数据； O 表示 RNN 在当前时间步的预测结果； V 表示输出层的权重矩阵； U 表示输入层到隐藏层的权重矩阵。

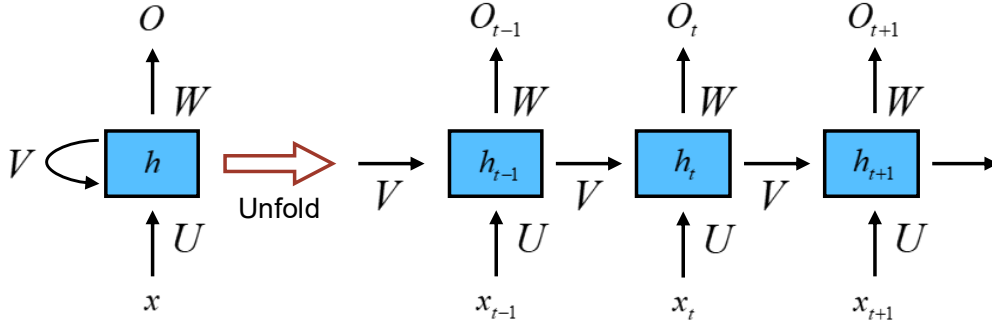


图 2-5 RNN 神经网络结构图

Figure 2-5 Neural network architecture diagram of RNN

(1) 输入层：RNN 的输入层负责接收序列数据中的每个元素。输入层的神经元数量通常与向量的维度一致，假设向量的维度是 300，那么输入层就有 300 个神经元来接收这些数据。

(2) 隐藏层：隐藏层是 RNN 的核心部分，它不仅接收来自输入层的当前元素，还接收来自前一个时间步的隐藏状态。隐藏层通过循环结构实现了信息的传递和记忆，在开始处理序列之前，隐藏层的状态通常被初始化为零向量。隐藏层的隐藏状态更新公式为：

$$h_t = \sigma(W \cdot [h_{t-1}, x_t] + b) \quad (2.1)$$

其中， W 是权重矩阵， b 是偏置项， σ 是激活函数。

(3) 输出层：输出层根据隐藏层的输出生成最终的预测结果。输出层的计算公式一般为：

$$O_t = V \cdot h_t + c \quad (2.2)$$

其中， V 是输出层的权重矩阵， c 是输出层的偏置项。

综上所述，RNN 的核心优势是能够捕捉序列数据中元素之间的时间依赖性。通过隐藏状态在时间步之间的传递，RNN 可以利用之前的信息来理解当前的输入，从而更准确地进行预测和分析。但是，在训练深层网络时，RNN 容易出现梯度消失问题，导致网络难以学习到长期依赖关系。

2.2.2 长短期记忆网络

长短期记忆网络（Long Short-Term Memory, LSTM^[50]）是一种特殊的循环神经网络，专门用于解决长期依赖问题，使用 LSTM 可以有效的传递和表达长时间序列中的信息并且不会导致长时间前的有用信息被忽略。LSTM 的结构如图 2-6 所示，图中的每一个黄色矩形代表一个神经网络层，上面的字符代表神经网络所使用的激活函数。LSTM 的核心是其单元状态和三个门的协同工作。具体来说：输入门（Input Gate）：决定哪些新信息应该被存储到单元状态中；遗忘门（Forget Gate）：决定哪些信息应该从单元状态中丢弃；输出门（Output Gate）：决定哪些信息应该从单元状态中输出。

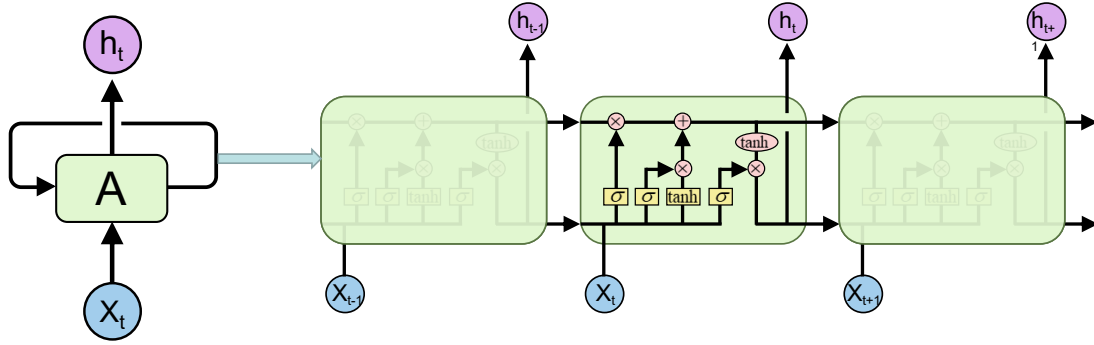


图 2-6 LSTM 神经网络结构图

Figure 2-6 Neural network architecture of an LSTM

单元状态能够在时间步之间传递信息，使得 LSTM 能够记住长时间之前的重要信息，并在需要时使用这些信息。这种信息传递机制是 LSTM 能够处理长期依赖关系的关键。LSTM 通过三个门来调节单元状态中的信息流动。这些门可以决定哪些信息应该被丢弃、哪些新信息应该被存储以及哪些信息应该被输出。下面将具体介绍 LSTM 单元状态的更新流程。

(1) 遗忘门：决定丢弃哪些信息。遗忘门通过一个 sigmoid 函数来输出介于 0 到 1 之间的值，表示对细胞状态中每个元素的遗忘程度。公式为：

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2.3)$$

其中遗忘门的输出是 f_t ； W_f 是权重矩阵； b_f 是偏置项， σ 是 sigmoid 函数。

(2) 输入门：决定存储哪些新信息。输入门也由一个 sigmoid 函数和一个 tanh 函数组成。sigmoid 函数决定哪些值需要更新，tanh 函数创建新的候选值来

更新状态。公式为：

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2.4)$$

$$C_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (2.5)$$

其中输入门的输入是 f_i ； W_i 是权重矩阵； b_i 是偏置项， σ 是 sigmoid 函数； C_t 是候选单元状态。

(3) 更新单元状态：将旧状态与遗忘门的输出按元素相乘，再将输入门的输出与候选单元状态按元素相乘，最后将两者相加，得到新的细胞状态。公式为：

$$C_t = f_t \odot C_{t-1} + i_t \odot C_t \quad (2.6)$$

(4) 输出门：决定输出哪些信息。输出门通过一个 sigmoid 函数和一个 tanh 函数来生成最终的输出。公式为：

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (2.7)$$

$$h_t = o_t \odot \tanh(C_t) \quad (2.8)$$

其中， o_t 是输出门的输出， h_t 是当前时间步的隐藏状态。

综上所述，LSTM 通过其门控机制能够有效地捕捉序列中的长期依赖关系，使得 LSTM 在处理长序列数据时表现出色。但是 LSTM 的结构相对复杂，训练和推理过程需要较多的计算资源和时间。其循环结构和门控机制导致计算量较大，特别是在处理长序列时，训练速度较慢，难以并行化。

2.2.3 Transformer 模型

Transformer 模型^[51]最早由 Vaswani 等人于 2017 年在论文《Attention is All You Need》中提出，该模型的核心思想是通过自注意力机制捕捉输入序列中元素之间的全局依赖关系。在自动驾驶领域，Transformer 被广泛应用于轨迹预测等任务，为复杂交通场景下的智能决策提供了强有力的技术支持。Transformer 模型结构如图 2-7 所示。该模型主要由四部分组成，分别是：输入嵌入、位置编码、编码器和解码器。接下来分别介绍每一个组成部分。

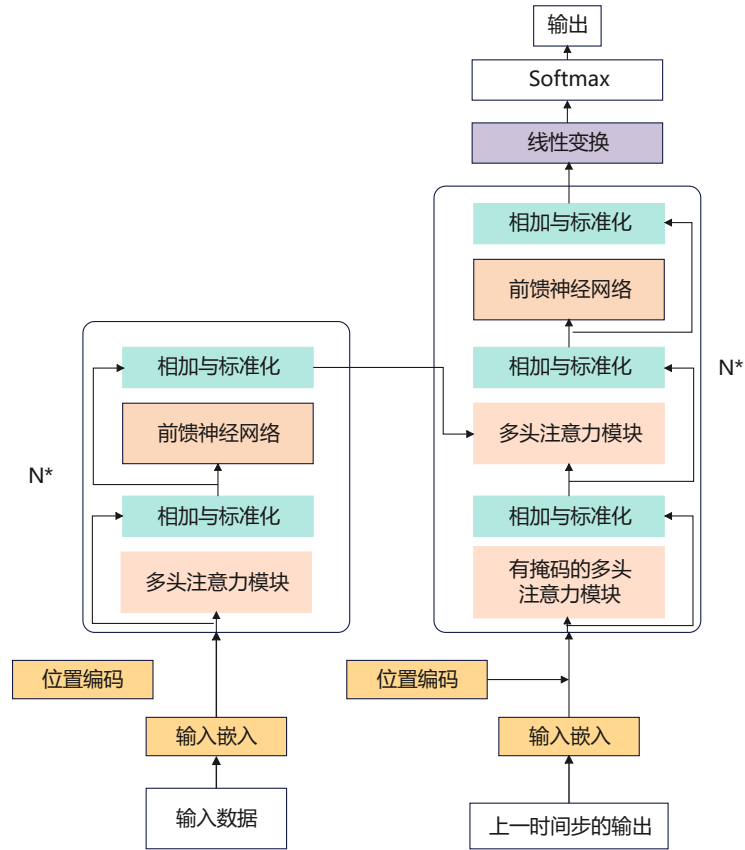


图 2-7 Transformer 模型结构图

Figure 2-7 Transformer Model Architecture

(1) 输入嵌入

Transformer 模型的输入由两部分组成：输入序列的嵌入表示（Input Embedding）和位置编码（Positional Encoding）。如图 2-8 所示，假设输入序列包含三个向量（例如，三个时间步的轨迹点或三个单词），其输入过程如下：这种输入表示方式使 Transformer 能够同时捕捉输入序列的语义信息和位置信息，从而有效建模序列中的全局依赖关系。在自动驾驶任务中，输入序列可以是车辆的历史轨迹点，嵌入表示将每个轨迹点映射为一个向量。这种输入表示方法可以用于处理车辆轨迹、传感器数据等序列化信息，为轨迹预测任务提供支持。

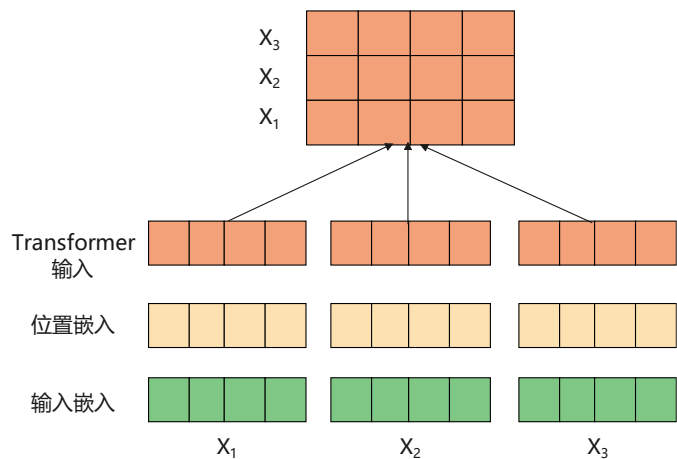


图 2-8 Transformer 模型输入示例图

Figure 2-8 Example diagram of Transformer model input

(2) 编码器

Transformer 模型的编码器模块负责将输入序列转换为高维特征表示，捕捉序列中元素之间的全局依赖关系。编码器模块由 N 个相同的编码器层堆叠而成，每个编码器层的结构完全相同。如图 2-9 所示，将输入嵌入矩阵传入编码器模块，经过 N 个编码器模块后得到输出编码信息矩阵 C 。自注意力机制使 Transformer 模型能够同时考虑输入序列中的所有位置，允许模型根据输入序列中的不同部分来赋予不同的注意权重，是编码器的核心模块，用于捕捉序列中元素之间的全局依赖关系。

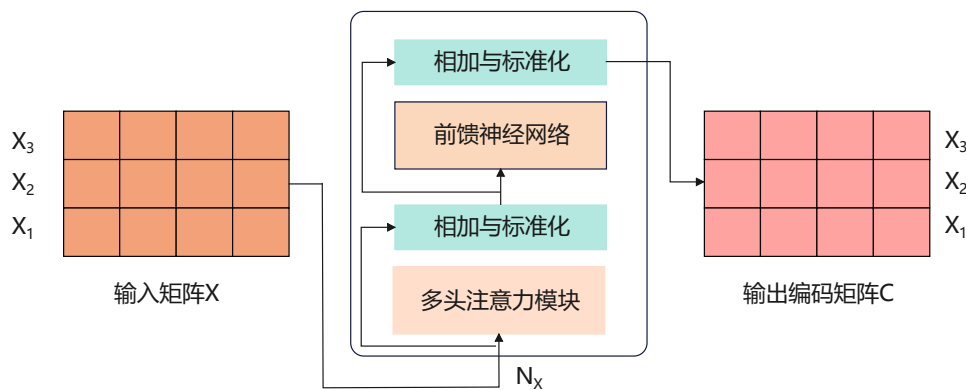


图 2-9 Transformer 模型编码器示例图

Figure 2-9 Example diagram of a Transformer model encoder

(3) 解码器

Transformer 模型的解码器模块同样由 N 个相同的解码器层堆叠而成，每个解码器层的结构完全相同。掩码多头自注意力是解码器中特有的模块，其核心在

于引入掩码机制以确保在生成输出序列时只能使用当前位置之前的信息。如图 2-10 所示,解码器模块在预测第 i 个输出时,需要将第 $i+1$ 之后的单词掩盖住。

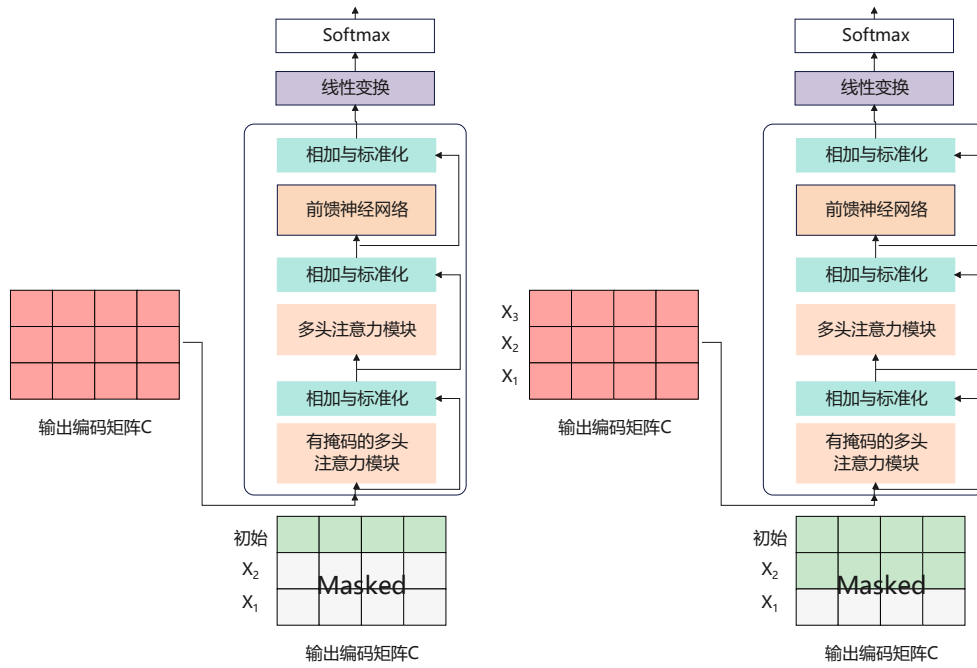


图 2-10 Transformer 模型解码器示例图

Figure 2-10 Example diagram of a Transformer model decoder

综上所述,Transformer 模型通过其独特的自注意力机制和架构设计,在并行计算、长距离依赖建模和全局上下文捕捉等方面显著优于传统神经网络。然而,Transformer 依赖全局信息进行建模,因此对输入数据的细微变化较为敏感,可能影响模型的稳定性和输出结果。同时它也存在计算成本高、数据需求量大等缺点。

2.3 强化学习基础理论

强化学习 (RL) 讨论的问题是智能体 (agent) 如何在复杂的、不确定的环境 (environment) 中最大化的去获得奖励^[52]。它是一门关于决策的学科,在强化学习中,智能体与环境是一直交互的,智能体在获得环境中某个状态时,他会利用该状态输出一个动作 (action),这个动作也被称为决策 (decision)。然后智能体会执行这个动作,环境会根据智能体采取的动作,给出相应的奖励。这里要注意的是,奖励并不是瞬时奖励,也就是说,奖励不是及时响应的。与此同时,奖励也不全是正向奖励,也有负奖励,所以强化学习的目标就是让奖励尽可能的大^[53]。强化学习过程示意图如图 2-11 所示:

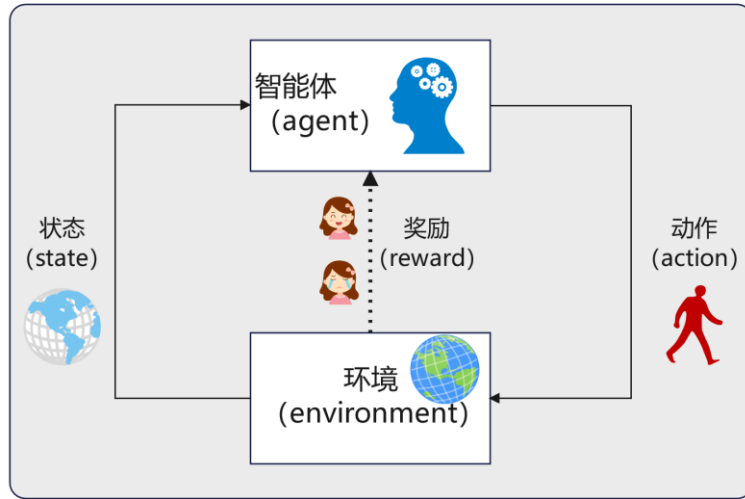


图 2-11 强化学习示意图

Figure 2-11 Illustration of Reinforcement learning

强化学习与监督学习不同，监督学习的样本都需要满足独立同分布的条件^[54]，而强化学习处理的数据大多数都是序列数据；其次监督学习需要告诉学习器正确的标签，而强化学习是不需要这个标签的，它需要自己去发现哪些动作可以带来最多的奖励。这其实是一个不断试错的过程，也是强化学习中的核心问题：探索和利用。探索^[55]是指智能体去尝试新的动作，利用^[56]指智能体采取已知的可以获得最多奖励的动作，然后重复执行这个动作，这也是监督学习中没有的情况。

2.3.1 马尔可夫决策过程

在强化学习问题中，马尔可夫决策过程（Markov Decision Process, MDP）是一个核心的数学框架，用于形式化描述智能体（Agent）与环境（Environment）之间的交互过程。MDP 提供了一种结构化的方式来表示决策问题，使得智能体能够在不确定的环境中通过学习和优化策略来实现长期目标。以下是对 MDP 在强化学习中的详细描述。

MDP 是一组具有马尔可夫性质的随机变量序列 $(s_1, \dots, s_t)$ ，其中下一个时刻的状态 s_{t+1} 只取决于当前状态 s_t 。马尔可夫过程满足以下条件：

$$p(s_{t+1} | s_t) = p(s_{t+1} | h_t) \quad (2.9)$$

其中， h_t 表示历史状态，公式（2.9）所表达的意思是，从当前状态 s_t 转移到状态 s_{t+1} ，就等于他之前所有状态转移到 s_{t+1} 。离散的马尔可夫过程也称马尔可夫链（Markov Chain），马尔可夫链示例如图 2-12 所示：图中一共有四个状态，这

4 个状态在 s_1, s_2, s_3, s_4 之间相互转移。例如从 s_4 开始，他有 0.3 的概率转移到 s_2 ，有 0.2 的概率转移到 s_3 ，有 0.5 的概率留在当前状态。

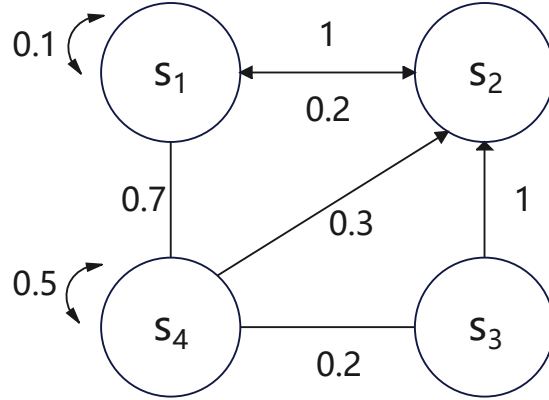


图 2-12 马尔可夫链示例图

Figure 2-12 Example diagram of a Markov chain

在马尔可夫过程的基础上加入奖励函数和折扣因子，就可以得到马尔可夫奖励过程（Markov reward process）。在马尔可夫奖励过程中，一个状态的期望回报（即从这个状态出发的未来累积奖励的期望）被称为这个状态的价值（value）。所有状态的价值就组成了价值函数（value function），价值函数的输入为某个状态，输出为这个状态的价值。马尔可夫过程和马尔可夫奖励过程都是自发改变的随机过程，在马尔可夫奖励过程（MRP）的基础上加入智能体来执行动作，就得到了马尔可夫决策过程（MDP）。

MDP 可以由一个五元组表示： $(S, A, \{P_{(s,a)}\}, \gamma, R)$ 。其中， S 是状态的集合； A 是动作的集合； $P_{(s,a)}$ 是状态转移概率； γ 是对未来奖励的折扣因子； R 是奖励函数。

（1）状态价值函数：在 MDP 中基于策略 π 的状态价值函数定义为从状态出发遵循策略 π 能获得的期望回报，它的数学表达式为：

$$V^\pi(s) = \mathbb{E}_\pi[G_t | S_t = s] \quad (2.10)$$

（2）动作价值函数：在 MDP 中，由于动作的存在，额外定义一个动作价值函数（action-value function），表示 MDP 遵循策略 π 时，对当前状态执 s 动作 a 得到的期望回报。

$$Q^\pi(s, a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a] \quad (2.11)$$

2.3.2 深度强化学习

深度强化学习的核心思想是将深度神经网络引入强化学习框架中,用于近似值函数、策略或环境模型。具体来说:(1)用深度神经网络近似状态值函数 $V^\pi(s)$ 或动作值函数 $Q^\pi(s,a)$;(2)策略近似:用深度神经网络直接参数化策略;(3)模型近似:用深度神经网络学习环境的动态模型。基于智能体在学习过程中优化目标的不同,深度强化学习方法可以分为基于值函数的方法和基于策略的方法[57]。

(1) 基于值函数的方法:这类方法的核心思想是通过估计状态值函数 V 或动作值函数 Q 来推导最优策略。基于值函数的方法包括多种经典算法,如深度 Q 网络 (Deep Q -Network, DQN)、Double DQN 等[58]。

(2) 基于策略的方法:这类方法的核心思想是直接优化策略,使得策略能够最大化累计奖励。这种方法的代表算法有蒙特卡洛策略梯度方法[59]、近端策略优化算法[60]等。

在基于策略的深度强化学习算法中,值得注意的是 Actor-Critic 算法。这种方法既学习了价值函数,又学习了策略函数。但是 Actor-Critic 算法本质上是基于策略的算法,因为它的算法目标是优化一个带参数的策略,只是会额外学习价值函数,从而帮助策略函数更好地学习。

Actor-Critic 分为两个部分,其中 Actor 要做的是与环境交互,并在 Critic 价值函数的指导下用策略梯度学习一个更好的策略。Critic 要做的是通过 Actor 与环境交互收集的数据学习一个价值函数,这个价值函数会用于判断在当前状态什么动作是好的,什么动作不是好的,进而帮助 Actor 进行策略更新。Actor-Critic 算法交互流程如图 2-13 所示:环境首先提供当前状态 S_t ,并同时输入给 Actor (策略网络)和 Critic (价值网络)。Actor 根据当前策略选择动作并执行,环境随后更新状态并返回奖励。Critic 通过价值函数评估当前状态的价值,并结合环境反馈的奖励计算动作的“优势”(即动作的好坏程度)。Actor 根据 Critic 的评估结果优化策略,同时 Critic 也同步更新价值函数。通过这种循环协作机制,Actor 的策略逐渐优化,最终趋于最优。

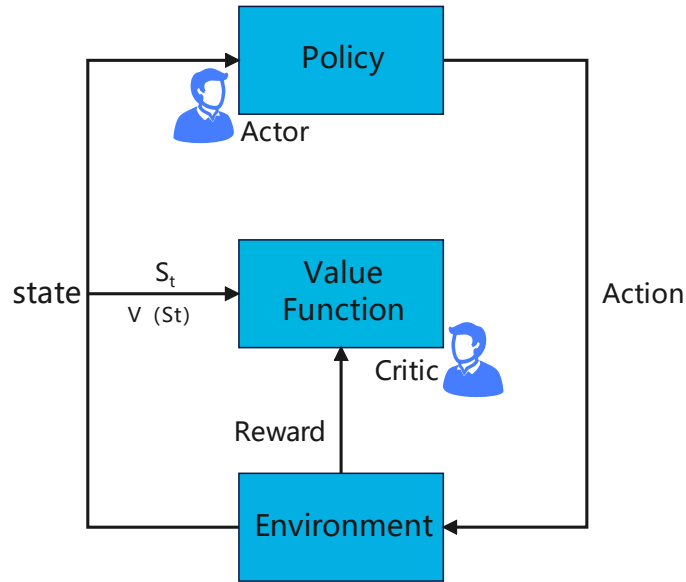


图 2-13 Actor-Critic 框架图

Figure 2-13 The Actor-Critic framework

Actor-Critic 算法作为深度强化学习领域中的核心方法之一，在处理连续动作空间和复杂任务方面展现出了显著的优势。在实际应用中，该算法在机器人控制、自动驾驶、游戏 AI 等多个领域均展现了强大的潜力。例如，在机器人控制中，**Actor-Critic** 算法能够使机器人通过与环境的交互学习到最优的动作策略，从而实现精准的运动控制；在自动驾驶领域，它可以帮助车辆做出更安全、高效的驾驶决策；在游戏 AI 方面，该算法能够使虚拟角色具备更智能的行为模式，提升游戏体验。通过这些方向的研究和实践，**Actor-Critic** 算法将能够更好地应对复杂的现实问题，推动人工智能技术在更多领域的应用和发展。

2.4 数据集及仿真模拟平台

2.4.1 数据集

现有的自动驾驶技术在不断发展与进步，研究人员在研究过程中，遇到了环境复杂、数据质量以及安全等多方面的问题与挑战，自动驾驶数据集正是为了解决这些问题^[61]。自动驾驶数据集提供了大量多样化的数据，覆盖了各种场景和条件，比如城市道路、高速公路和乡村道路等多种驾驶场景以及晴天、阴天和雨天等不同天气条件。现有的主流自动驾驶数据集主要有 NGSIM 数据集^[62]、HighD 数据集^[63]、Argoverse 数据集^[64]、INTERACTION 数据集^[65]和 Argoverse 2 数据

集^[66]。

(1) NGSIM 数据集在 2006 年由美国联邦公路管理局发布，它包含美国多个高速公路和城市道路的车辆轨迹数据，对于现在的自动驾驶任务来说，该数据集的数据比较旧，没有传感器数据，场景复杂度也较低；

(2) HighD 数据集在 2018 年由德国亚琛工业大学发布，它包含包含 60 小时的高速公路驾驶数据，覆盖 110,000 辆车的轨迹，该数据集专注于高速公路场景，缺乏城市道路数据；

(3) Argoverse 数据集在 2019 年由 Argo AI 发布，它分为两个不同的数据集：3D Tracking Dataset 和 Motion Forecasting Dataset。3D Tracking Dataset 包含 113 个场景，每个场景 15-30 秒，提供 2 个 LiDAR、7 个摄像头的多模态数据，以及 3D 边界框标注。Motion Forecasting Dataset 包含 30,000+ 个车辆轨迹片段，提供 5 秒历史轨迹和 3 秒未来轨迹；

(4) INTERACTION 数据集是 2019 年发布的，包含多种复杂场景的车辆和行人轨迹数据，该数据集在行人轨迹预测任务中也取得了良好的效果；

(5) Argoverse 2 数据集在 2021 年由 Argo AI 发布，它分为三个不同的数据集：3D Tracking Dataset、Motion Forecasting Dataset 和 Lidar Dataset。该数据集提供了更多多模态数据。

表 2-1 主流自动驾驶数据集对比表展示了现在主流自动驾驶数据集的特点。

表 2-1 主流自动驾驶数据集对比表

Table 2-1 Comparison table of mainstream autonomous driving datasets

数据集	NGSIM	HighD	Argoverse	INTERACTION	Argoverse 2
时间	2016	2018	2019	2020	2021
地图数据	无	无	高清地图	无	高清地图
场景	高速公路 城市道路	高速公路	城市道路	交叉口、环行路	城市道路
主要任务	交通流建模	交通流建模 轨迹预测	目标检测 轨迹预测	交互行为建模 轨迹预测	目标检测 轨迹预测

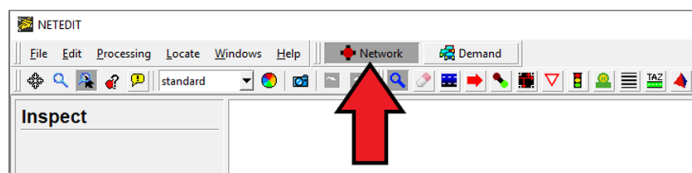
2.4.2 仿真模拟平台

研究人员在进行自动驾驶技术研究时,需要通过大量的实验来验证自己的想法的可行性。但是在真实道路场景下进行自动驾驶实验存在安全风险,尤其是进行高速公路换道等实验。仿真平台可以在虚拟环境中模仿不同的驾驶场景,不需要实际车辆和道路资源,帮助研究人员进行算法测试和优化。在实际应用中,可根据具体研究目标和项目需求选择合适的仿真平台。以下是部分主流的自动驾驶模拟仿真平台:

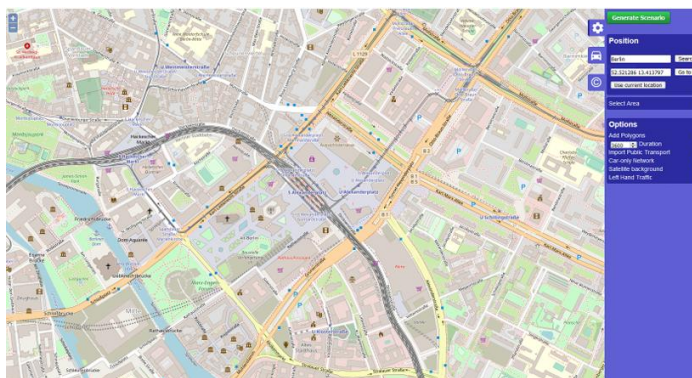
(1) SUMO

SUMO^[67]是一个开源的,微观的,多模态交通仿真。它允许模拟由单个车辆组成的给定交通需求如何在给定道路网络中移动。该模拟允许解决大量交通管理主题。它纯粹是微观的:每辆车都经过明确建模,拥有自己的路线,并通过网络进行单独移动。默认情况下,模拟是确定性的,但是引入随机性有多种选择。

在 SUMO 中进行模拟仿真时,必须包括三种文件:道路网络、行程文件、SUMO 配置文件。首先,网络文件可以通过 `netedit` 手动绘制道路网络,也可以使用 `openstreetmap` 地图转换为 SUMO 可用的网络文件,如图 2-14 所示:



(a) netedit手动绘制



(b) openstreetmap地图转换

图 2-14 SUMO 路网绘制方法

Figure 2-14 Road network rendering method in SUMO

生成道路网络后可以使用 `sumo-gui` 对其进行查看,但是此时,道路网络中

不会有车辆在行使，这时候就需要定义车辆行驶路线，也就是行程。行程是由起点，目的地终点和出发时间定义的从一个地方到另一个地方的车辆运动。在创建行程文件时，必须确保路径由连续的路段组成，即车辆行驶的路径在网络拓扑中是连通的。例如，可以通过 SUMO 提供的 `duarouter` 工具或手动编写 `.rou.xml` 文件来定义行程。行程文件中通常包含车辆的类型、出发时间、行驶路径等信息。通过将行程文件与路网文件结合，可以在仿真中生成动态的交通流，从而为自动驾驶决策方法的研究提供真实的交通场景。

(2) CARLA

CARLA^[68] 一个用于自动驾驶研究的开源模拟器，它提供的环境包括城市布局、多种车辆模型、建筑物、行人、路标等。CARLA 提供的城市环境示例如图 2-15 所示。图 2-15 (a) 中三处红框分别对应着右侧的三个子图，图 2-15 (b) 是典型的匝道口场景、图 2-15 (c) 是典型的十字交叉路口场景、d 是典型的环形路场景。除此之外，环境还可以提供可用于训练驾驶策略的信号，如 GPS 坐标、速度、加速度以及碰撞和其他违规行为的详细数据。

CARLA 采用服务器-客户端架构，其中服务器端负责渲染和物理计算，而客户端则通过 Python 或 C++ 接口来控制仿真逻辑，并支持多客户端并行测试。其核心工具包括 Scenario Runner，它支持 OpenSCENARIO 标准场景的解析与执行，以及 Traffic Manager，负责管理 NPC 车辆的路径规划和行为决策。这些功能共同为自动驾驶算法的开发、测试与验证提供了一个高度灵活且可扩展的仿真环境。

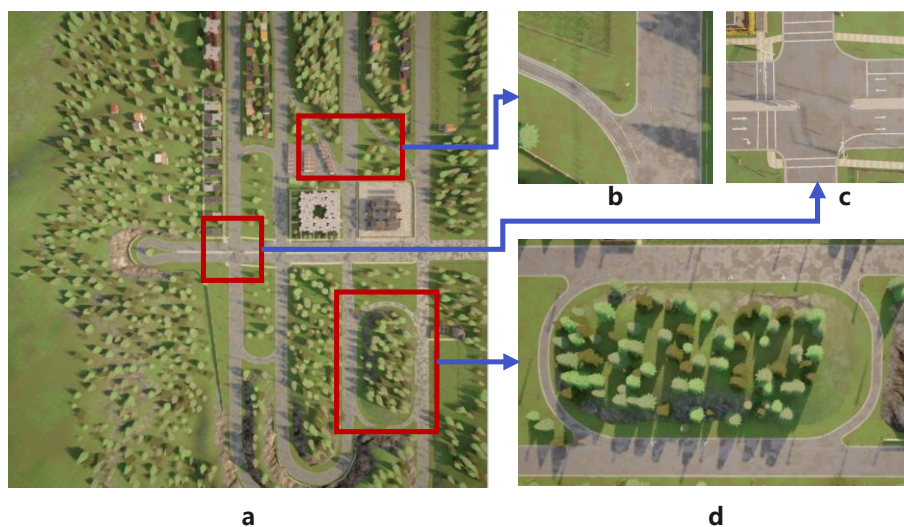


图 2-15 CARLA 城市环境示例图

Figure 2-15 Example diagram of an urban environment by CARLA

(3) MetaDrive

MetaDrive 是一个由 Quanyi Li 等人在论文《MetaDrive: Composing Diverse Driving Scenarios for Generalizable Reinforcement Learning》中提出的驾驶仿真平台^[69]。该平台旨在通过构建多样化的驾驶场景，支持强化学习算法的泛化能力研究，为自动驾驶领域的算法开发和测试提供了一个高度灵活且可扩展的环境。MetaDrive 提供的多样化驾驶场景示例如图 2-16 所示：

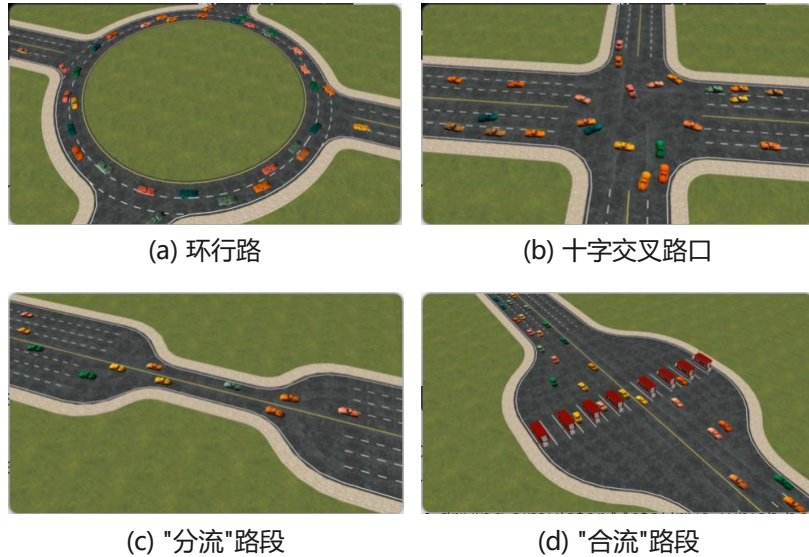


图 2-16 MetaDrive 驾驶场景示例图

Figure 2-16 Example figure of MetaDrive driving scenario

MetaDrive 的核心架构基于强化学习的“策略-环境-价值”协同框架：策略网络（Actor）利用深度神经网络处理车辆状态等多模态数据，生成转向角和油门以及刹车力度等连续控制指令；环境模拟器构建高保真的虚拟场景，支持自定义道路拓扑（如十字路口、环形匝道）、动态交通流及突发事件；价值评估网络通过多维奖励函数量化策略优劣，并计算优势函数以指导策略梯度更新。这些优点使其在自动驾驶研究和开发领域具有广泛的应用前景，能够有效支持从学术研究到工业应用的多种需求。

(4) SMARTS

自动驾驶仿真平台 SMARTS（Smart Vision-based Training Simulator）是一个由 Ming Zhou 等人在论文《SMARTS: Scalable Multi-Agent Reinforcement Learning Training School for Autonomous Driving》中提出的驾驶仿真平台^[70]，旨在为自动

驾驶算法开发、测试和验证提供高效且逼真的模拟环境。

SMARTS 的核心架构围绕多智能体交互与模拟环境构建展开。它能够生成包括城市道路、高速公路以及特殊交通区域在内的多样化场景，支持自动驾驶车辆、人类驾驶车辆、行人以及交通基础设施等多种智能体的动态交互。在车辆动力学与传感器仿真方面，SMARTS 内置了高精度车辆动力学模型，涵盖非线性动力学特性、电动汽车性能测试及轨迹跟踪控制，适用于紧急避撞和换道场景模拟。同时，它支持激光雷达、摄像头、毫米波雷达等传感器建模，并生成与实际硬件高度一致的仿真数据，为感知算法验证提供基础。SMARTS 还提供了实时驾驶仿真模式，内置碰撞检测模块，支持 6 自由度交互设备（如方向盘、力反馈装置）的接入，增强用户体验的真实性。其动态相机编辑功能和多通道集群渲染技术，允许用户自由调整视角并实现大规模场景的同步渲染。

这些仿真平台在自动驾驶与智能交通领域被广泛使用，但是不同的仿真平台在功能定位和应用场景上存在显著差异。SUMO 作为开源交通流仿真工具，专注于微观交通行为建模和大规模路网分析，但由于其车辆动力学模型简化、传感器模拟缺失和图形渲染粗糙，难以满足高保真自动驾驶算法的测试需求。MetaDrive 专注于强化学习算法训练，为研究者提供了快速迭代策略的轻量化环境，但其物理精度和视觉真实感不足，限制了其在复杂现实场景中的应用潜力。

Carla 凭借高度逼真的 3D 环境和多模态传感器模拟，成为自动驾驶感知与端到端训练的首选平台。相比之下，SMARTS 专攻多智能体协同决策，在复杂交互优化中占据优势，但是其视觉渲染功能成熟度不及 Carla。它们共同构成从算法原型验证到系统级落地的完整技术链。

2.5 本章小结

本章首先介绍了自动驾驶技术概述，其中包括车辆轨迹预测和车辆换道行为，描述了车辆轨迹预测的基本处理流程以及车辆的不同换道行为。接着介绍了时间序列预测模型以及强化学习基础。在时间序列预测模型中分别介绍了 RNN，LSTM 以及 Transformer 模型。在强化学习基础中介绍了强化学习与监督学习的区别，以及 MDP 和深度强化学习。最后介绍了自动驾驶任务中常用的数据集以及仿真模拟平台。

第3章 基于 Informer 模型的多模态轨迹预测方法研究

车辆行驶轨迹的准确预测是保障自动驾驶系统安全性和可靠性的关键技术之一^[71]。在复杂的城市交通环境中，自动驾驶车辆面临着多源异构数据的处理挑战：一方面，静态环境信息（如车道线、交通标志等）与动态交通参与者信息（如车辆位置、速度等）共同构成了高维度的输入空间；另一方面，环境信息的部分可观测性以及交通参与者的不确定性等因素进一步增加了轨迹预测的难度。传统的单模态预测方法难以充分表征复杂的交通场景特征，而基于多模态的预测框架能够有效整合静态环境特征和动态时序信息，通过建立更完备的特征表示模型，显著提升轨迹预测的精度和鲁棒性。

因此，本章提出了基于 Informer 模型的多模态轨迹预测方法，旨在通过融合多源异构数据（包括动态交通参与者信息、静态环境特征以及时序历史轨迹数据），构建高效的特征表示模型，以解决复杂交通场景下部分可观测性、数据异构性以及不确定性管理等关键问题，从而实现对目标车辆轨迹的精准预测。本章首先对多模态轨迹预测问题进行了系统性描述，随后详细介绍了 Informer 模型的基本原理及其在时序预测任务中的优势。在此基础上，提出了基于 Informer 的多模态轨迹预测模型框架，该框架主要包括三个核心模块：特征处理模块、基于 Informer 的编码器模块和基于 Informer 的解码器模块。为验证所提方法的有效性，在 Argoverse 数据集上进行了实验验证，并通过 minADE（最小平均位移误差）和 minFDE（最小最终位移误差）等标准评价指标对模型性能进行了定量分析。实验结果表明，所提出的方法在轨迹预测精度方面取得了显著提升。

3.1 多模态轨迹预测问题概述

多模态轨迹预测是轨迹预测问题的一个重要研究方向，其核心在于通过融合多源异构数据^[72]（如视觉、雷达、激光雷达、高精度地图等）来提升预测精度和鲁棒性，以应对复杂交通场景中部分可观测性、数据异构性以及不确定性等挑战。在自动驾驶任务中，现实世界中的交通场景具有高度的复杂性和动态性^[73]。每辆车辆在遵守交通规则的前提下，通常存在多个可行的轨迹选择。例如，在十字路口场景中，当一辆汽车即将右转时，其可能选择直接右转，也可能选择先直行再

右转,具体决策往往受到多种因素的影响,包括车辆自身状态(如速度、加速度)、周围交通参与者的行为(如其他车辆、行人)以及环境约束(如车道线、交通信号灯)等。因此,准确建模目标车辆与其他移动实体以及交通场景之间的交互关系,是提升轨迹预测精度的关键。如图 3-1 所示,在双向六车道的复杂交通场景中,存在众多交通参与者。每个参与者的意图具有不确定性和动态性,其行为会随时间而变化。同时,周围交通参与者的动态变化和复杂交互作用进一步增加了轨迹预测的难度。为了应对这些挑战,需要综合考虑时空特征和多模态信息,以实现更可靠、更准确的预测结果。

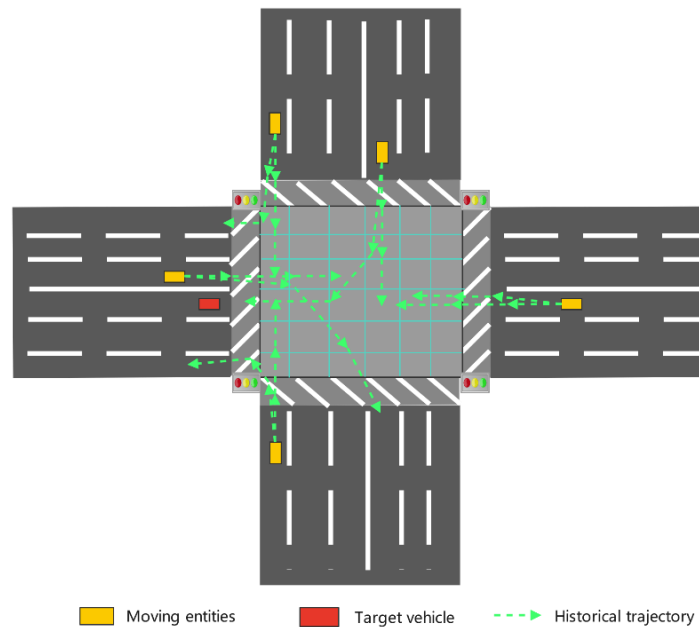


图 3-1 交通场景示例图

Figure 3-1 Illustration of traffic scene

现有的多模态轨迹预测算法主要基于序列建模和注意力机制的方法,这些方法通过捕捉时序依赖关系和空间交互特征,能够有效处理轨迹预测任务中的复杂动态场景。具体而言,基于序列建模的方法擅长处理时间维度上的长程依赖问题,而基于注意力机制的方法则能够动态聚焦于关键的空间和时间信息,从而提升预测精度^[74]。然而,尽管这些方法在多模态轨迹预测任务中取得了显著进展,但仍面临部分可观测性和数据异构性等方面的挑战。部分可观测性意味着模型在预测时可能无法获取所有相关的信息,导致预测结果的不确定性增加。数据异构性则指不同数据源生成的数据在特征上的差异,这增加了数据融合和处理的复杂性。综上所述,尽管现有的多模态轨迹预测算法在处理复杂动态场景方面取得了显著

进展，但仍需在部分可观测性以及数据异构性等方面进行深入研究和优化，以实现更可靠、更高效的预测结果。

3.2 研究目标

虽然多模态轨迹预测问题已经得到了广泛研究，但大多数现有的多模态轨迹预测方法往往仅侧重于提升预测准确性，而忽略了实际交通场景中多种影响因素的综合建模。例如，复杂交通场景中其他交通参与者的行为意图，如车与车之间的动态交互、行人与车辆之间的交互等等。这些因素对轨迹预测的结果都具有重要影响。因此，如何在保证预测精度的同时，全面考虑交通场景中的多种影响因素，仍然是当前研究中的一个关键挑战。因此，本章旨在探索如何综合多维影响因素对目标车辆轨迹进行更准确的预测。

综上，本章的目标是通过构建能够同时建模时间维度、空间维度以及多模态数据交互的预测框架，以应对复杂交通场景中的部分可观测性、数据异构性以及不确定性等挑战，从而提升轨迹预测的精度和鲁棒性。

3.3 Informer 模型

Informer 模型^[75]由 Haoyi Zhou 等人于 2021 年在论文《Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting》中提出，该项工作获得了 AAAI 2021 的 Best Paper。该模型基于 Transformer 架构进行了优化和改进，是一种专为长序列时间序列预测问题设计的深度学习模型。Informer 模型结构图如图 3-2 所示。在自动驾驶轨迹预测任务中，Informer 模型的输入通常是车辆的历史轨迹数据，包括位置、速度、加速度等信息。这些数据被编码为时间序列，并通过嵌入层映射为高维向量表示。该模型通过引入 ProbSparse 自注意力机制和蒸馏操作，能够有效降低高维输入数据的计算复杂度和存储开销，同时保留关键信息，显著提升了长序列时间预测任务的效率和性能。这一特性使其特别适合处理多模态轨迹预测任务中的高维时空数据，为解决复杂交通场景中的部分可观测性和数据异构性问题提供了新的技术途径。

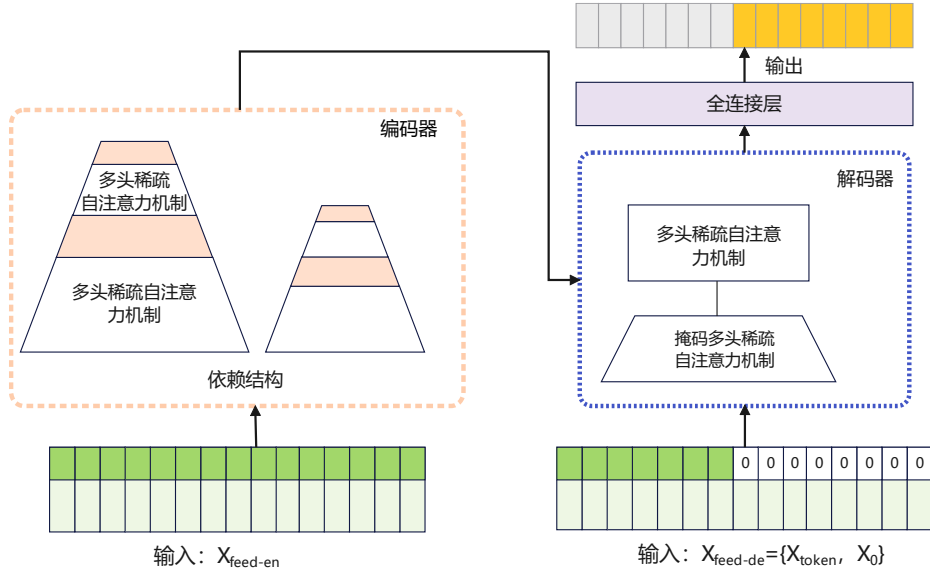


图 3-2 Informer 模型结构图

Figure 3-2 Informer model structure

在编码器部分，Informer 模型通过多层稀疏自注意力机制和前馈神经网络逐步提取输入序列的特征表示。每一层编码器后接一个蒸馏操作，通过卷积和最大池化减少序列长度，降低内存占用，同时保留重要信息。这种设计使得 Informer 模型能够更高效地处理长序列数据。图中深绿色的部分是输入嵌入，是原始输入的向量；浅绿色的部分包含位置编码来表示输入序列的相对位置，除此之外，还加入了时序相关编码。每个解码器模块有 3 个 Attention Block，每个 Attention Block 内有 n 个头权重矩阵。Encoder 中重复的部分可以由下面的公式 3.1 概括：

$$X_{j+1}^t = \max \text{pool}(ELU(\text{conv1d}([X_j^t]_{ab}))) \quad (3.1)$$

$[X_j^t]_{ab}$ 表示 Attention Block，包括多头 ProbSparse 自相关，conv1d 是 1D-CNN，使用 ELU 激活函数在时间维度上执行一维卷积过滤器，外面套了 max-pooling 层 (stride=2)，这样每次会下采样一半来节省了内存占用率。

表 3-1 分析了 Informer 模型相较于 Transformer 模型的改进，下文将详细介绍这些改进。

(1) Attention 计算的不同

在长序列任务中，并非所有位置的注意力权重都具有同等重要性。Transformer 模型在计算注意力权重时，需要对序列中每个位置进行全局计算，导致时间复杂度较高。Informer 通过引入稀疏注意力机制和概率注意力机制显著降

低了计算复杂度。稀疏注意力机制通过构建稀疏查询矩阵，选择性地计算部分查询向量与键向量的点积，而不是整个注意力矩阵，从而将计算复杂度从 $O(L^2)$ 降低到 $O(L\log L)$ ，同时减少了内存占用。概率注意力机制进一步通过概率采样和自适应权重分配，只计算关键信息的点积，进一步降低计算复杂度至 $O(L\log L)$ 或更低，同时忽略冗余信息，确保模型在处理长序列时的高效性和实用性。

Informer 通过稀疏注意力机制和概率注意力机制，显著降低了计算复杂度和内存占用，使其更适合处理长序列任务。这种优化不仅提高了模型的效率，还增强了其在实际应用中的可扩展性。

(2) 编码器结构的不同

Transformer 模型的编码器由多层相同的网络结构堆叠而成，这种设计导致了较高的内存占用和计算开销。Informer 模型针对这一问题，提出了蒸馏堆叠编码器结构。具体来说，Informer 的编码器在每一层中通过蒸馏操作筛选出最重要的特征，并将这些特征传递到下一层，最终实现高效的特征提取。

(3) 解码器结构的不同

Transformer 模型的解码器采用自回归式预测方式，即 step by step 生成预测结果，具体来说，Transformer 模型每次预测一个时间步，并将当前预测结果作为下一时间步的输入，这种方式存在预测速度慢、误差累积等问题。Informer 模型针对这一问题，提出了生成式预测解码器结构。该结构通过一次性生成所有时间步的预测结果，避免了自回归预测中的逐步计算过程，从而显著提升了预测效率。

表 3-1 Informer 模型优化表

Table 3-1 Optimization table for the Informer model

Transformer 模型（不足）	Informer 模型（优化）
self-attention 平方级的计算复杂度: $O(L^2)$	提出 ProbSparse Self-attention
	筛选出重要的 query，降低计算的复杂度: $O(L\log L)$ 或更低
堆叠多层网络，内存占用遇到瓶颈	提出 Self-attention 蒸馏
	减少维度和网络参数
step-by-step 解码预测，速度较慢	提出生成式预测解码器结构
	一步得到所有预测结果

Informer 模型的上述优化使其在长序列预测任务中取得了显著成效。多模态轨迹预测任务的核心在于利用目标车辆的历史轨迹以及周围交通参与者的历史轨迹信息进行未来轨迹预测，因此可以将其视为一种典型的长序列预测问题。本章基于 Informer 模型构建多模态轨迹预测框架，通过其高效的注意力机制和长序列建模能力，能够更好地捕捉目标车辆与周围环境之间的多维交互关系。实验结果表明，所提出的模型在预测精度和计算效率方面均表现出优越性能。

3.4 基于 Informer 的多模态轨迹预测模型框架

3.4.1 问题建模

车辆与其他移动实体（如周围车辆、行人等）以及交通场景之间的动态交互是影响自动驾驶车辆安全性和决策可靠性的关键因素。因此，准确预测目标车辆及其周围实体的未来轨迹对于改进车辆的决策策略以及实现安全自动驾驶至关重要。图 3-3 展示了同一方向车道上部分车辆间交互的详细场景：以三车道交通场景为例，目标车辆所在车道及其相邻车道上分布着多辆行驶车辆，同时人行横道上也存在移动实体，如行人等。

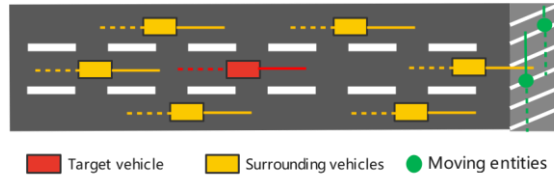


图 3-3 交互细节图

Figure 3-3 Interaction detail diagram

将多模态轨迹预测任务定义为：综合考虑交通场景与移动实体（本文将其统称为“智能体”，即 agent）之间的交互关系，预测目标车辆未来的多种可能轨迹。每个智能体的轨迹包含空间和时间两个维度的信息，因为其在空间中的位置会随时间动态变化。具体而言，智能体的轨迹可以表示为一系列按时间顺序排列的状态序列，其中每个状态包含位置、速度、加速度等关键运动特征。对于智能体的轨迹，将其定义为 T ：

$$T = \{T^0, T^1, T^2, \dots, T^A\} \quad (3.2)$$

其中， A 为智能体的个数。对于每个智能体 i ，其历史轨迹可以表示为：

$$T^i = \{(x_i^{t_h}, y_i^{t_h}), \dots, (x_i^{t_h-1}, y_i^{t_h-1}), (x_i^t, y_i^t)\} \quad (3.3)$$

T^i 代表智能体在不同时刻的纵向位置信息和水平位置信息。为了更好地表示智能体的历史轨迹，将这些位置信息矢量化，并将每条轨迹表示为一个向量序列。对于智能体的每条历史轨迹，将其视为一条有向折线。具体来说，属于每条有向折线 DP_m 的向量 DV_i 定义如下：

$$DV_i = [p_i^s, p_i^e, f_L, \theta_i, m] \quad (3.4)$$

$$L = \{k_{\text{vehicle}}, k_{\text{lane}}, k_{\text{crossroad}}, k_{\text{light}}, k_{\text{other}}\} \quad (3.5)$$

p_i^s 和 p_i^e 分别表示初始位置和终端位置； θ_i 表示有向折线的角度， m 表示有向折线的索引； f_L 表示向量的特征，其中 L 表示不同的特征标签，如车道特征、交通灯状态等。向量 DV_i 的集合用 H 表示。

所有特征向量在时刻 t 的集合嵌入如下：

$$X_{\text{input}} = \{h_t \mid h \in H, t \in T_p\} \quad (3.6)$$

未来生成的预测轨迹定义如下：

$$X_{\text{output}} = \{O_t \mid t \in T_{\text{pre}}\} \quad (3.7)$$

其中，历史轨迹的时间范围为：

$$T_p = \{t_h, \dots, t\} \quad (3.8)$$

其中，预测轨迹时间范围为：

$$T_{\text{pr}} = \{t, \dots, t_s\} \quad (3.9)$$

由于不同模态的特征向量在长度上存在差异，这可能导致在后续处理过程中出现不一致性。为了消除这种差异并确保所有特征向量在后续分析和处理中具有统一的尺度，采用了归一化方法。具体设计如下：

$$h_t^{\text{norm}} = \frac{h_t - \min(h_t)}{\max(h_t) - \min(h_t)} \quad (3.10)$$

3.4.2 模型框架

本章提出了一种多模态轨迹预测模型，该模型通过综合考虑智能体与周围环

境之间以及不同智能体之间的交互关系来实现轨迹预测，并将该模型框架命名为 MTPI（Multimodal Trajectory Prediction with Informer）。

MTPI 的整体结构如图 3-4 所示，主要由四个核心模块组成：特征处理模块、编码器模块、解码器模块和全连接层模块。其中，特征处理模块负责提取和融合多源异构数据（如目标车辆的历史轨迹、周围交通参与者的状态以及环境信息等），并通过归一化操作消除数据尺度差异；编码器模块基于 Informer 模型设计，利用 ProbSparse 自注意力机制捕捉长序列数据中的时空依赖关系；解码器模块采用生成式预测方式，一次性输出未来多个时间步的轨迹预测结果；全连接层模块则用于将解码器输出的高维特征映射为最终的预测轨迹。通过上述模块的协同工作，MTPI 能够有效建模复杂交通场景中的多维交互关系，从而提升轨迹预测的精度和鲁棒性。

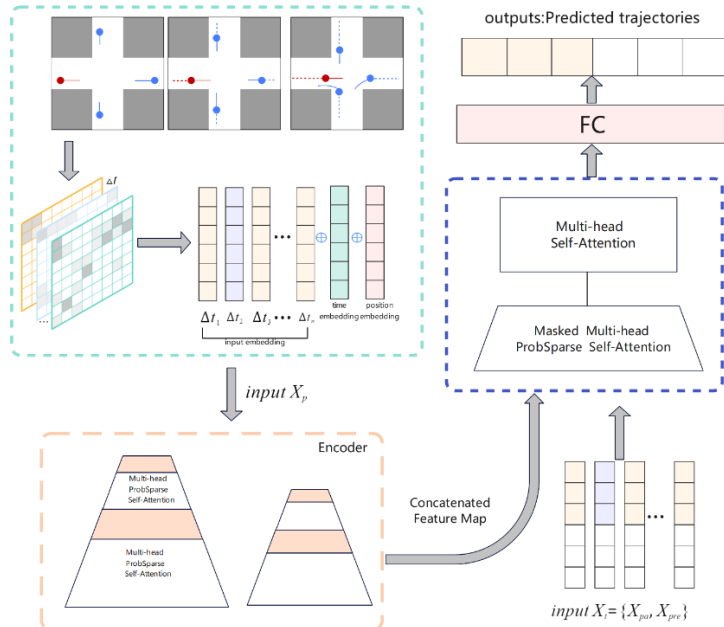


图 3-4 基于 Informer 的多模态轨迹预测模型框架

Figure 3-4 Framework of multimodal trajectory prediction model based on Informer

3.4.3 特征处理模块

为了提取和选择最相关的信息，对输入数据进行了特征处理，旨在降低数据维度并消除冗余信息。特征处理的具体流程如图 3-6 所示。首先，基于交通场景提取目标车辆（agent）和场景的特征信息。由于不同模态的输入特征向量具有不同的尺度，通过归一化操作将其统一到相同的数值范围。随后，将融合后的特征

信息转换为包含时间信息的特征矩阵，并通过可视化展示不同特征之间的关联度。此外，在特征处理过程中，还引入了潜在变量特征 h_t^i ，以进一步捕捉数据中的隐含关系。

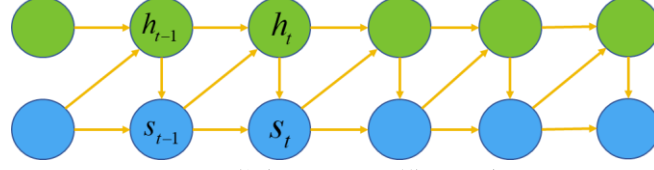


图 3-5 潜变量自回归模型示意图

Figure 3-5 Schematic representation of the latent variable autoregressive model

潜变量自回归模型示意如图 3-5 所示，它的主要作用是建模过去时间步的历史信息，从而捕捉序列数据中的长期依赖关系。具体来说， h_t 与上一时刻的 h_{t-1} 和 s_{t-1} 相关，将上一时刻的潜变量和观测变量作为输入，生成当前时刻的潜变量。它的具体设计如下：

$$h_t^i = RNN(S_{t-1}^i, h_{t-1}^i) \quad (3.11)$$

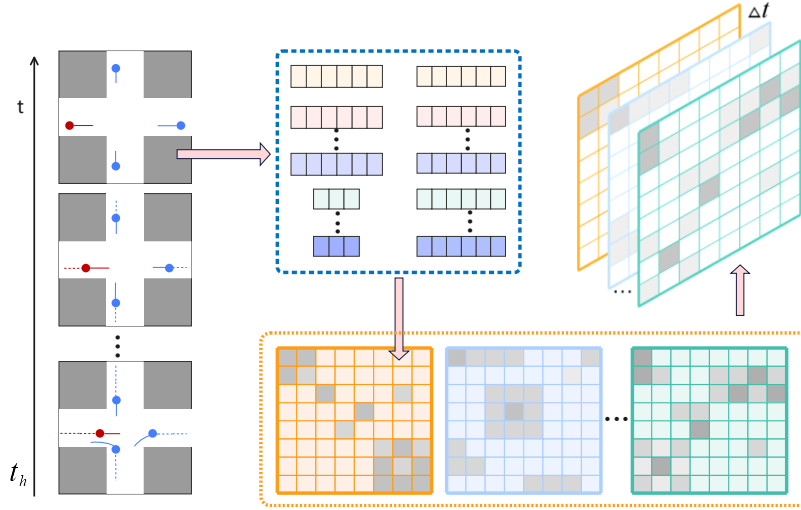


图 3-6 特征细节图

Figure 3-6 Feature detail map

3.4.4 MTPI-encoder

本章提出的编码器模块（后文称 MTPI-encoder 模块）通过对输入序列进行编码来捕捉时空依赖关系和重要特征，从而提高了轨迹预测的精度。MTPI-encoder 模块通过使用自注意机制和其他操作，捕获输入序列中的时间顺序和不

同特征之间的相互关系。此外，它还为后续的解码器模块提供了有价值的信息。

具体来说，MTPI-encoder 模块的输入由三部分组成：

(1) 轨迹数据

第一部分是目标车辆与其他运动实体、交通场景交互产生的轨迹数据。这些数据首先由特征处理模块进行预处理，然后将特征处理后的信息编码到 MTPI-encoder 模块中。轨迹数据定义为：

$$v_i^t = \text{embedding}(h_t) \quad (3.12)$$

(2) 位置编码

采用位置编码对过去和现在位置进行编码，目的是将空间信息嵌入到输入序列中，使模型能够更好地理解和捕捉空间特征和空间关系。位置编码记为 PE。PE 定义为：

$$PE_{(POS,2i)} = \sin(pos / 10000^{2i/d_{model}}) \quad (3.13)$$

$$PE_{(POS,2i+1)} = \cos(pos / 10000^{2i/d_{model}}) \quad (3.14)$$

(3) 时间编码

时间编码将时间信息转换为可学习的向量表示，使模型在处理输入序列时考虑时间因素。时间编码用 TE 表示。TE 定义为：

$$TE_{(tp,2i)} = \sin(tp / 10000^{2i/d_{model}}) \quad (3.15)$$

$$TE_{(tp,2i+1)} = \cos(tp / 10000^{2i/d_{model}}) \quad (3.16)$$

最后，MTPI-encoder 模块输入为：

$$X_{en_input} = v_i^t + PE_{l \times (t-1)+i} + SE_{l \times (t-1)+i} \quad (3.17)$$

$$X_{en_input}^t = \{X_{en_input(1)}^t, X_{en_input(2)}^t, \dots, X_{en_input(l)}^t\} \quad (3.18)$$

经过预处理的序列数据被输入到解码器模块中，MTPI-encoder 框架如图 3-7 所示。解码器模块的输入是经过 Embedding 层映射后的序列信息。然后通过多头 ProbSparse 自注意力机制对序列信息进行处理。具体来说，解码器主要由三个 attention block 构成，每个 attention block 均基于 Informer 模型的解码器模块设计。首先，Attention Block 1 对输入的时间编码、位置编码以及轨迹数据进行多模态融合，并通过 ProbSparse 自注意力机制筛选关键信息，同时利用多头注意力机制捕捉多维度的时序依赖关系。然后，Attention Block 2 通过最大池化操作对序列进行降采样，增强特征表达能力，有效捕捉长尾依赖关系。最后，Attention

Block 3 通过进一步提取特征，将不同的时空特征融合，最终输出适配于轨迹生成模块的特征图。

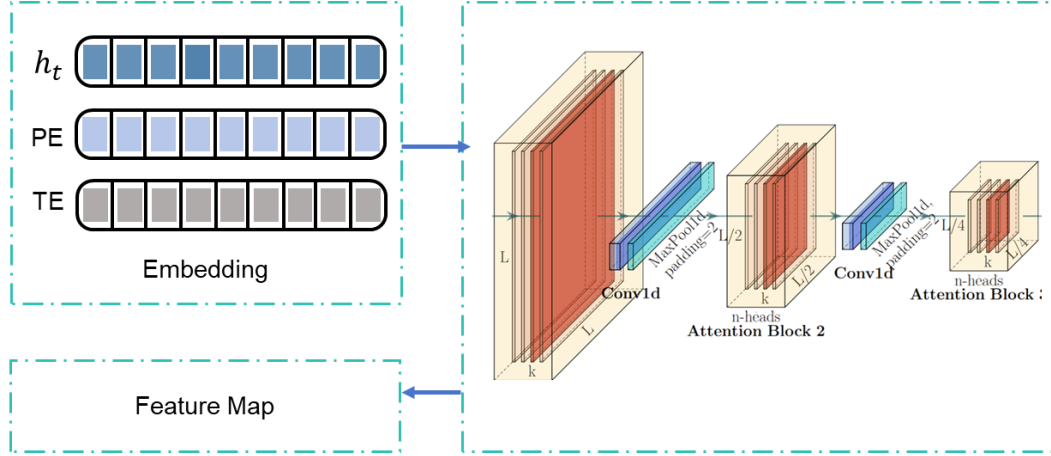


图 3-7 MTPI-encoder 框架图

Figure 3-7 The frame diagram of the MTPI encoder module

3.4.5 MTPI-decoder

本章提出的解码器模块（后文称 MTPI-decoder）通过提取的上下文信息和多模态特征，生成目标车辆在未来时间内的运动轨迹。MTPI-decoder 模块的主要目标是预测目标车辆的未来轨迹，MTPI-decoder 框架如图 3-8 所示。

MTPI-decoder 模块的输入由两部分组成：（1）MTPI-encoder 模块的输出，经过编码器模块处理后的高维特征表示，通常包含序列的全局时空信息、多模态特征融合结果以及关键时间步的注意力筛选信息；（2）原始序列信息和待预测信息，具体表示为：

$$X_{de_input}^t = \text{link}(X_{en_token}^t, X_{en_0}^t) \quad (3.19)$$

$X_{en_token}^t$ 表示初始数据标记，即在目标车辆附近采集的初始数据。 $X_{en_0}^t$ 表示目标序列的未来观测值，初始化为零。

首先，对 MTPI-decoder 模块的输入进行特征对齐与初始化，然后利用 ProbSparse 自注意力机制筛选关键时间步，将稀疏化后的特征输入到多头注意力机制中，捕捉多维度的时序依赖关系，并通过掩码机制防止未来信息泄露。最后 MTPI-decoder 模块的输出通过一个全连接层进行处理，得到最终结果。

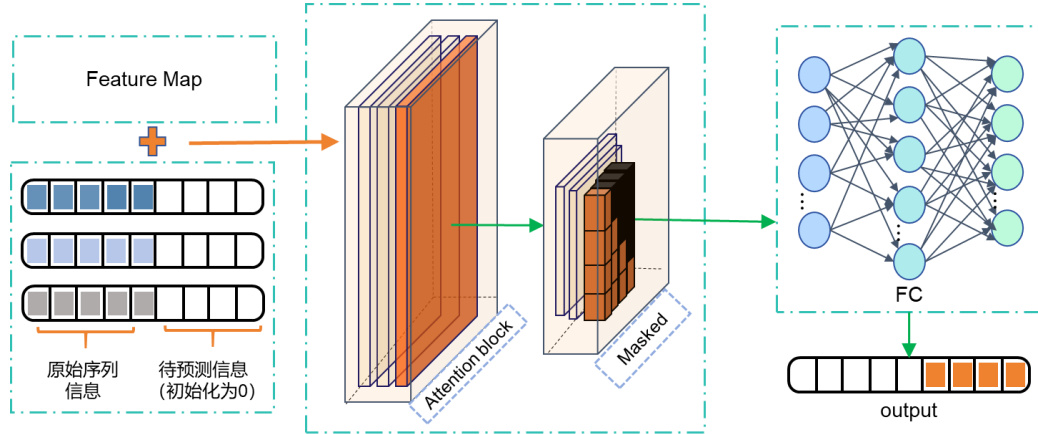


图 3-8 MTPI-decoder 框架图

Figure 3-8 The frame diagram of MTPI-decoder

3.5 实验与分析

3.5.1 实验设置

实验环境：PC 机配置为 NVIDIA Corporation、64G 内存、64 位 Ubuntu 操作系统作为实验平台，python 3.7, Pytorch1.1.0, torchvision0.3.0, cuda9.0 等开发工具。

Argoverse 数据集：Argoverse 数据集由 Argo AI 于 2019 年发布，是自动驾驶领域的高质量数据资源之一。在本文实验中，主要使用 Argoverse 的运动预测子集（Argoverse Motion Forecasting dataset），该子集专注于车辆运动预测任务，提供了丰富的车辆历史轨迹和未来轨迹数据。Argoverse 运动预测数据集包含训练集、验证集和测试集，每个集合由不同数量的用例组成。具体来说，该数据集包含 33.3 万条 5 秒长的序列，分为 21.1 万条训练序列、4.1 万条验证序列和 8 万条测试序列。轨迹以 10Hz 的频率采样，其中(0, 2]秒用于观测，(2, 5]秒用于轨迹预测。每个序列中均有一个目标智能体。通过利用该子集，能够进行精确且可靠的车辆运动预测实验，为模型性能评估提供了坚实的基础。

3.5.2 评价指标

为了评估提出模型的性能，采用了三个常用的指标：**minADE**（最小平均位移误差）、**minFDE**（最小最终位移误差）和 **MR**（缺失率）。

minADE（最小平均位移误差）：预测轨迹与地面真实轨迹之间的平均距离误差。

$$\min ADE = \frac{1}{T_{pred}} \min \sum_t^{T_{pred}} \|y_t - \hat{y}_t^k\|_2 \dots k \in [1, k] \pi \quad (3.20)$$

其中, T_{pred} 是预测时间范围, y_t 是 真实轨迹在时间步 t 的位置, $\hat{y}_t^k$ 是

表示第 k 条预测轨迹在时间步 t 的位置。minADE 越小, 表示模型在整体预测轨迹上的精度越高。

minFDE (最小最终位移误差): 预测轨迹与地面真实轨迹之间的最终距离误差。

$$\min ADE = \min \sum_t^T \|y_{T_{pred}} - y_{T_{pred}}^k\|_2 \dots k \in [1, k] \quad (3.21)$$

其中, $y_{T_{pred}}$ 和 $y_{T_{pred}}^k$ 分别表示真实轨迹和第 k 条预测轨迹在最终时间步 T_{pred} 的位置。minFDE 越小, 说明模型对轨迹终点的预测越准确。

MR (缺失率): 在指定误差范围内未能准确预测的轨迹的比例。MR 越低, 说明模型的预测结果越接近真实轨迹。

$$MR = \frac{1}{N} \sum_i^N \Pi(\min FDE_i > 2) \quad (3.22)$$

3.5.3 实验结果与分析

首先, 在相同的实验条件下, 评估了 MTPI 的预测精度, 并将其与 Argoverse 运动预测数据集上的现有方法进行了比较。以下是正在考虑比较的一些现有方法:

(1) CV (Constant Velocity)^[64]: 该方法基于目标车辆的当前位置和速度, 通过线性外推预测其未来轨迹。其假设车辆在未来时间段内保持恒定速度运动, 是一种简单但广泛使用的基线方法。

(2) LSTM+map(prior) 1-G,n-C^[64]: 该模型利用 LSTM 网络捕捉目标车辆历史轨迹中的时序依赖关系, 并结合高精度地图先验信息(如车道线、交通规则等)来增强轨迹预测的准确性。该模型的不同之处在于能够有效融合动态轨迹数据与静态环境信息。

(3) DiversityGAN^[76]: 该模型通过潜在语义采样实现多样性感知的车辆运动预测, 能够生成多条可能的未来轨迹。其核心思想是通过生成对抗网络(GAN)捕捉交通场景中的不确定性, 从而提升预测结果的多样性和鲁棒性。

通过对比上述方法，我们能够全面评估 MTPI 模型在多模态轨迹预测任务中的性能优势，并验证其在复杂交通场景中的适用性。

表 3-2 展示了不同模型在 Argoverse 运动预测数据集上的预测性能。MTPI 的 minADE 为 2.56，均低于其他模型，这表明 MTPI 在预测过程中能够更准确地捕捉车辆的运动轨迹。MTPI 的 minFDE 为 5.51，均低于其他模型，这说明 MTPI 在预测的最终位置上也具有更高的准确性。MTPI 的 MR 为 0.80，虽然略高于 LSTM+map 的 0.75，但仍然低于 CV 和 DiversityGAN，这表明 MTPI 在捕捉车辆行为变化方面表现较好。综上所述，MTPI 在 Argoverse 数据集上的表现优于其他模型，特别是在 minADE 和 minFDE 指标上，这表明 MTPI 能够更准确地预测车辆的运动轨迹，为自动驾驶系统提供了更可靠的预测结果。

表 3-2 不同模型的预测性能对比

Table 3-2 Comparison of prediction performance of different models

Model	Metrics ↓		
	<i>minADE</i>	<i>minFDE</i>	<i>MR</i>
CV(Constant Velocity)	3.53	7.89	0.83
LSTM+map(prior) 1-G,n-C	2.92	6.45	<u>0.75</u>
DiversityGAN	2.91	6.54	0.82
MTINformer(ours)	<u>2.56</u>	<u>5.51</u>	0.80

此外，为了更好地展示模型的预测效果，从 Argoverse 运动预测数据集中选择了不同的场景进行了可视化。这些场景涵盖了车辆在不同驾驶场景以及车辆在十字路口的不同行驶目标，包括直行、左转和右转。

此外，为了更好地展示模型的预测效果，从 Argoverse 运动预测数据集中选择了具有代表性的场景进行可视化。这些场景包括十字交叉路口场景和高速公路匝道场景，还特别关注了车辆在十字路口的三种主要行驶目标：直行、左转和右转。接下来进一步介绍可视化结果，其中红色轨迹代表真实轨迹，蓝色线圈轨迹代表预测轨迹。

如图 3-9 所示，展示了十字路口场景车辆直行的轨迹预测效果。可以看出，预测轨迹与真实轨迹在整体趋势上较为一致，尤其是在直行区域的行驶方向和位置上。并且预测轨迹在直行区域表现出一定的稳定性，没有出现剧烈的波动或偏

离，说明模型在处理直行场景时具有一定的鲁棒性。

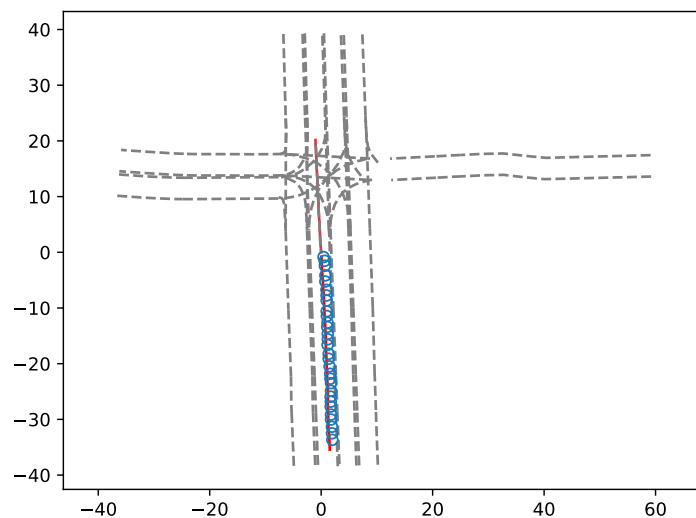


图 3-9 十字路口直行轨迹预测可视化结果图

Figure 3-9 Visualization result of straight trajectory prediction at the intersection

如图 3-10 所示，展示了十字路口场景车辆左转的轨迹预测效果。可以看出，在车辆左转时，真实轨迹和预测轨迹在整体趋势上较为一致。这表明模型能够较好地捕捉车辆在十字路口左转时的运动模式，预测出合理的轨迹。

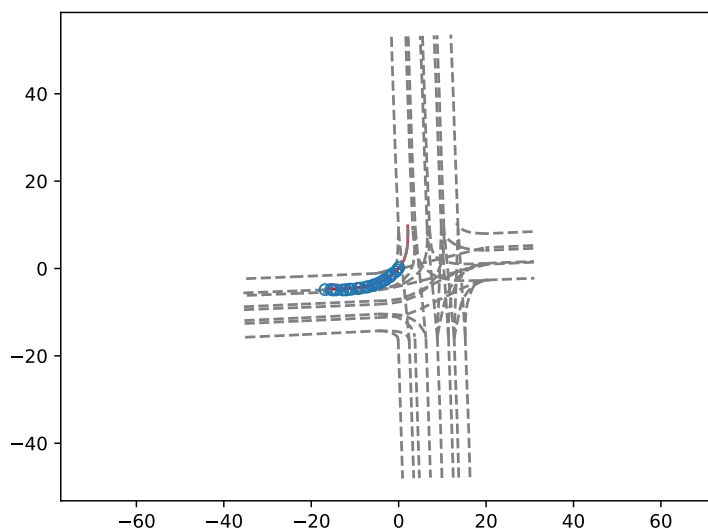


图 3-10 十字路口左转轨迹预测可视化结果图

Figure 3-10 Visualization result of left-turn trajectory prediction at the intersection

如图 3-11 所示，展示了十字路口场景车辆右转的轨迹预测效果。可以看出，在轨迹的起点和终点，预测轨迹与真实轨迹的匹配度较高。这表明模型在初始条件和最终目标的预测上表现较好。在车辆右转时，也就是轨迹的中间段，预测轨

迹与真实轨迹存在一定的偏差，但是仍能保持较高的预测精度。这表明模型能够较好地捕捉车辆在十字路口右转时的运动模式，预测出合理的轨迹。

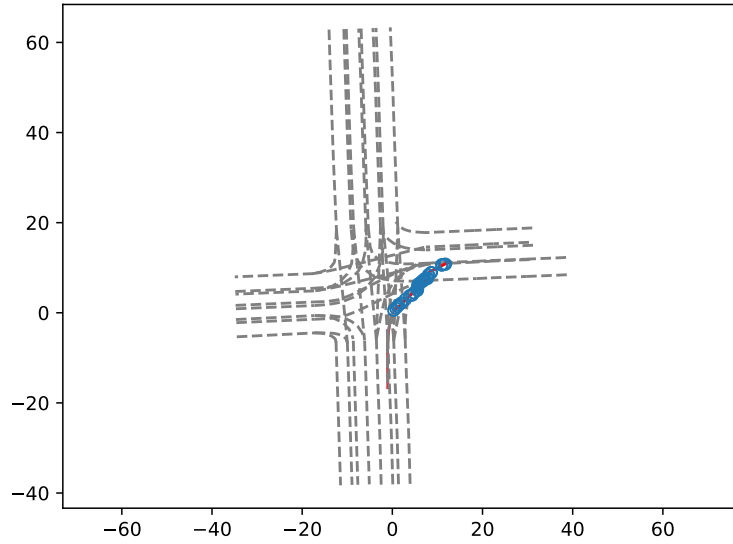


图 3-11 十字路口右转轨迹预测可视化结果图

Figure 3-11 Visualization result of right-turn trajectory prediction at the intersection

如图 3-12 所示，展示了高速公路匝道场景的轨迹预测效果。可以看出，在刚开始的直行路段，预测轨迹和真是轨迹的匹配度较高，但是在轨迹预测后半段，接近匝道处时，预测轨迹出现了微小偏差，这是因为在匝道区域，车辆的运动意图具有更高的不确定性，导致预测模型在长时序依赖下的累积误差被放大。

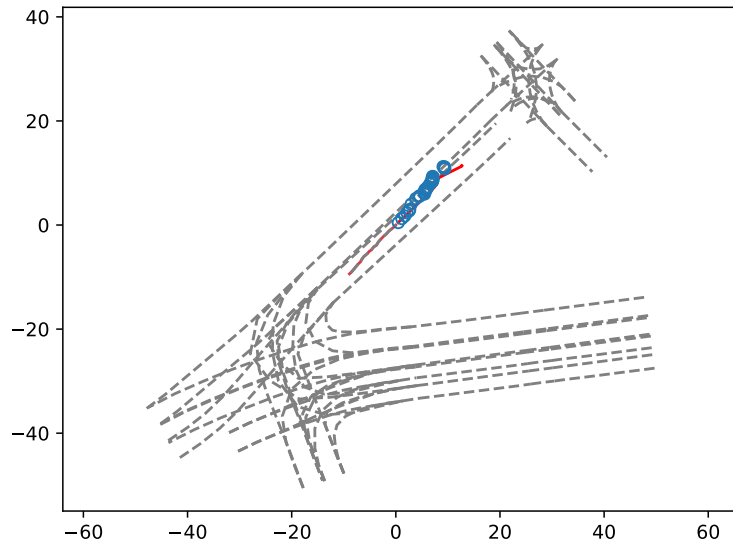


图 3-12 高速公路匝道直行轨迹预测可视化结果图

Figure 3-12 Visualization result of freeway ramp straight trajectory prediction

3.6 本章小结

本章首先介绍了多模态轨迹预测问题是什么,对于这个问题进行研究的意义。随后又介绍了 Informer 模型并描述了 Informer 模型相较于 transformer 模型的改进,这也是为什么选择 informer 模型的原因。随后本章对多模态轨迹预测问题进行了问题公式化,提出了基于 informer 模型的多模态轨迹预测模型,命名为 MTPI。MTPI 结合了自注意提取和 probsparse 自注意操作,旨在降低计算复杂度和提高内存使用效率。本章将 MTPI 与 CV (Constant Velocity)、LSTM+map(prior) 1-G,n-C、DiversityGAN 这三种轨迹预测方法进行了比较,实验结果表明,MTPI 在轨迹预测精度方面取得了较好的效果。

第 4 章 基于强化学习的多目标车辆换道决策优化方法研究

自动驾驶技术的进步推动了自动驾驶车辆换道策略的发展。自动驾驶车辆的换道策略研究对于提升道路利用率、缓解交通拥堵及保障行车安全具有重要意义^[77]。在实际道路行驶过程中，自动驾驶车辆可能因主动或被动因素需要进行换道操作，例如主动驶出高速公路岔路口或被动应对周围车辆的加塞等行为。因此，研究并优化自动驾驶车辆的换道策略具有重要意义。

针对现有自动驾驶换道策略在安全性、舒适性及效率性等方面存在的不足，本章提出了一种基于强化学习的多目标车辆换道决策优化方法。该方法以 A2C^[78]（Advantage Actor-Critic）强化学习算法为基础框架，通过构建多维状态空间，实现对自车及周围车辆状态与动作的精确表征，并综合考虑安全性、舒适性和效率性等多重优化目标。为验证方法的有效性，本研究构建了包含四个转弯路口、两段直行道路，两段弯道的环形道路复杂场景进行测试，该场景可模拟城市道路中的典型换道需求。基于 SUMO 仿真平台的实验结果表明，所提出的方法在安全性、舒适性和效率性等评价指标上取得了良好效果。

4.1 车辆换道问题概述

在城市道路环境中，道路拓扑结构主要呈现为双向两车道或三车道布局，其中包含十字路口、环形交叉口等复杂交通场景。这些多样化的道路结构为车辆换道行为提供了必要的空间条件，同时也带来了相应的决策挑战。在车辆行驶过程中，根据决策自主性的差异，车辆换道行为可分为自由换道（Free Lane Change）和强制换道（Mandatory Lane Change）两种基本类型^[79]。自由换道行为是指车辆根据当前交通状况和自身目标，自主决定是否换道。这种换道行为通常没有急迫性，时间约束较为宽松。强制换道行为是指车辆在某种特定情形下必须执行换道。这种换道行为有着急迫性和时间约束。例如高速公路驶出匝道，交通信号或道路标志的指示的强制车道变更。

在车辆换道决策过程中，无论是自由换道还是强制换道行为，其执行过程都

受到自车状态与周围车辆运动状态的共同影响。如图 4-1 所示，黄色小车表示自我车辆（Ego Vehicle），红色小车表示周围车辆（Surrounding Vehicles）。换道行为本质上是车辆在道路横向空间上的运动过程，所以自我车辆在进行换道决策时，不仅要考虑当前车道周围车辆的状态，也要考虑周围车道周围车辆的状态。

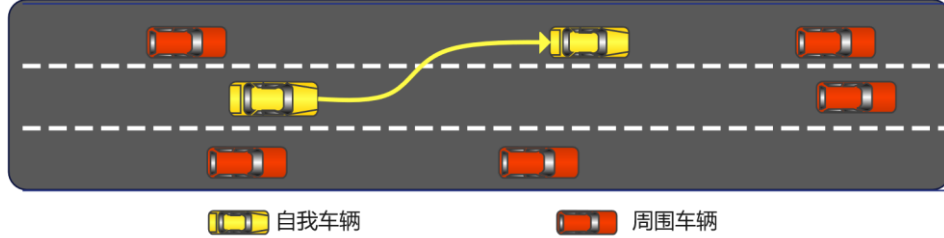


图 4-1 城市道路换道场景示例图

Figure 4-1 Example of an urban road lane-changing scenario

4.2 研究目标

本章针对自由换道问题，主要研究目标是设计并实现基于强化学习的多目标车辆换道决策优化方法，重点解决现有方法在安全性、效率和舒适性等方面的不足。研究目标是通过改进 Advantage Actor-Critic (A2C) 算法，结合多目标优化和概率建模方法，使自动驾驶车辆能够在复杂交通环境中实现安全、高效和舒适的换道决策。

本章首先设计了综合性的多目标奖励函数，将安全性、效率和舒适性三个关键特性结合；其次，扩展了状态空间表示，通过引入自车和周围车辆的多维度动态信息，提高系统对复杂交通环境的感知和理解能力；最后，利用优势函数降低策略更新的方差，提高算法的稳定性和探索能力。

4.3 基于强化学习的多目标车辆换道决策优化方法

4.3.1 问题定义

自动驾驶换道决策过程可以看作是一个马尔可夫决策过程 (Markov Decision Process, MDP^[80])，其中自我车辆作为智能体，在每个决策时间步 t 根据当前交通状态 S 执行动作 a 。执行动作之后，环境转移到新状态 S_{t+1} ，并产生奖励 r_t 。这一过程持续进行，直到车辆完成换道目标或到达终止状态。自我车辆也就是智能体的目标是在换道决策过程中，尽可能地以更安全更舒适更高效的去完成换道行

为。为实现这一目标，本研究采用基于 A2C 的方法，通过设计合理的状态空间、动作空间和奖励函数，使智能体能够在与环境的交互中自主学习最优换道策略。

4.3.2 车辆换道环境建模

在真实世界中，城市道路场景通常比简单的直线道路复杂得多，包括各种复杂的交叉路口，如十字立交桥、高速公路的进出口等^[81]。换道决策行为作为日常驾驶中最频繁发生的驾驶行为之一，其执行过程受到多种动态因素的影响，例如周围车辆的加减速行为、交通信号灯的指示、道路标志的约束以及其他交通参与者的行为不确定性。因此，如何合理的设计车辆换道决策策略，是有待解决优化的问题。

为了模拟真实城市道路中的复杂换道场景并验证自动驾驶车辆的决策能力，本研究设计了一个具有代表性的环形路实验场景。该场景包含两段直行路，两个弯道以及四个转弯路口，如图 4-2 所示。直行道模拟城市道路中的常规行驶路段，车辆可能因超车或避让触发换道行为；弯道模拟转弯区域，考验车辆在曲率变化条件下的换道控制能力；四个转弯路口模拟车辆在转弯路口的换道需求，例如左转或右转前的车道变更。

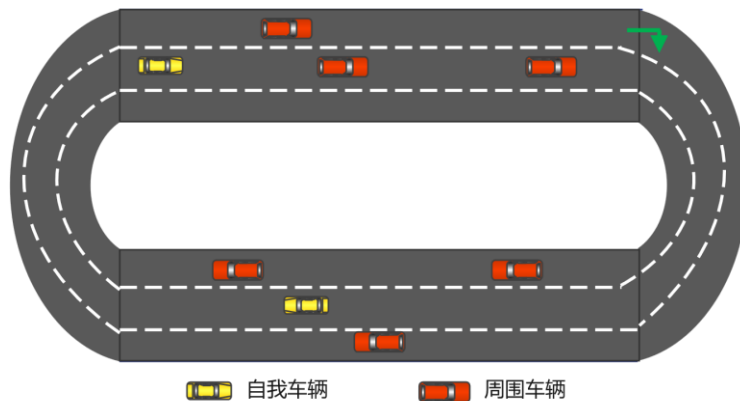


图 4-2 多车道环形路场景建模图

Figure 4-2 Multi-lane circular road scene modeling diagram

4.3.3 换道决策模型设计

本章提出了一种基于强化学习的多目标车辆换道决策优化方法。该方法以 A2C (Advantage Actor-Critic) 强化学习算法为基础框架，通过综合考虑安全性、舒适性和效率性等多重优化目标，实现对自我车辆的换道决策，并将该方法命名为 MOMA2C (Optimization method of multi-objective vehicle lane changing decision

based on A2C)。MOMA2C 的整体结构如图 4-3 所示,图的左侧是车辆换道场景,它的主要功能是向右侧的算法模块提供实时的环境观测信息,并反馈算法执行动作后的奖励值。具体而言,环境观测信息包括自车状态、周围车辆状态以及道路结构等多维度数据。同时,环境根据算法执行的动作计算奖励。图的右侧为算法模块,其核心功能是基于左侧环境提供的观测信息,输出最优换道动作。算法模块采用 A2C 框架,其中 Actor 网络负责生成动作策略,Critic 网络则评估状态价值以指导策略优化。算法输出再次反馈至左侧环境,形成闭环交互,直至换道行为结束。

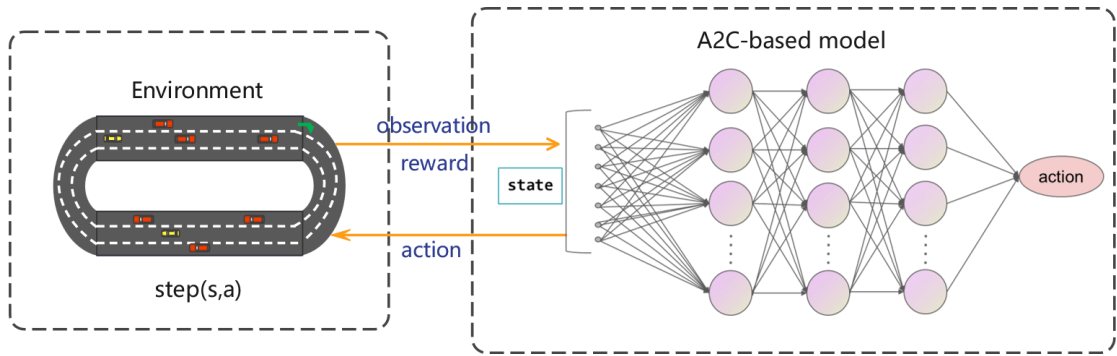


图 4-3 基于 A2C 的换道决策模型结构图

Figure 4-3 Structure diagram of lane changing decision model based on A2C

(1) Actor

Actor 是策略优化的核心组件,通过对当前状态 s 进行决策,执行优质动作 s 以获得更高累积期望奖励。具体而言,Actor 本质上是一个策略函数的近似器,它将环境的观测状态映射到动作概率分布。Actor 由两个模块组成,动作预测模块和概率采样模块,两者共同支持 Actor 对状态 s 进行可靠决策。

动作预测模块 N_{θ} 用于帮助智能体预测当前状态下所有候选动作的可能概率,如 (4.2) 式所示。 N_{θ} 通过对输入的状态进行特征提取和处理,捕捉状态与候选动作之间的关系,支持智能体进行灵活决策。

$$\log its = N_{\theta}(s_t) \quad (4.1)$$

概率采样模块 $P_{\theta}(\log its, flag)$ 用于根据 $flag$ 选择不同的采样模式,以适应不同训练时期对策略的偏好要求。具体而言,智能体在探索和利用困境中可以通过不同的选择偏好,对动作进行选择。在训练初始阶段,智能体通过随机策略按照概率分布随机选择动作以接触更多未知动作。在训练后期,智能体面对已知动

作，根据贪婪策略选择 Critic 评价最高的动作以获得最优决策。两者灵活变动，支持智能体在复杂环境中根据当前状态进行动态决策。

$$\pi_{\theta} = P_{\theta}(\log its, flag) = \begin{cases} \text{random sample}(\log its), & flag = 0 \\ \max(\log its), & flag = 1 \end{cases} \quad (4.2)$$

(2) Critic

Critic 通过对当前状态 S 进行价值评估，指导 Actor 进行更新。

$$V_{\alpha}(s_i) = \max_{a_i} \sum_{s_{i+1}, r_i} p(s_{i+1}, r_i | s_i, a_i) (r_i + \gamma V_{\alpha}(s_{i+1})) \quad (4.3)$$

为了衡量在给定状态下采取某个特定动作相对于平均情况的“优势”或“价值”，考虑使用 Critic 进行优势函数的评估。优势函数的具体设计如公式 (4.4) 所示：

$$A_{\alpha}(s_i, a_i) \approx r_i + \gamma V_{\alpha}(s_{i+1}) - V_{\alpha}(s_i) \quad (4.4)$$

其中， r_i 表示在状态 s_i 下执行动作 a_i 获得的即时奖励，反映当前动作的直接收益； $\gamma V_{\alpha}(s_{i+1})$ 表示下一状态 s_{i+1} 的折扣价值； $V_{\alpha}(s_i)$ 表示当前状态 s_i 的状态价值。MOMA2C 网络框架中 Actor 和 Critic 的交互如图 4-4 所示：

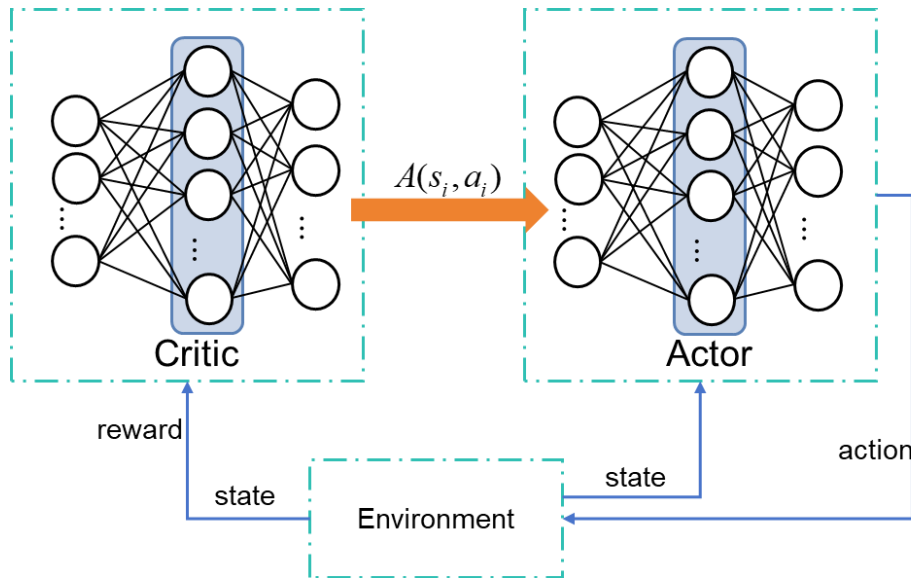


图 4-4 MOMA2C 网络框架图

Figure 4-4 MOMA2C network framework diagram

综上，基于 A2C 的换道决策模型的算法流程如表 4-1 所示：

表 4-1 基于 A2C 的换道决策模型算法流程

Table 4-1 Algorithm flow of lane changing decision model based on A2C

算法 1 基于 A2C 的换道决策模型算法	
初始化：动作预测模块 N_θ ，概率采样模块 $P_\theta(logits, flag)$ ，critic 网络 V_α ， $flag_0=0$;	
1:	for 每次迭代 do:
2:	采样状态 s_i
3:	# Actor 决策
4:	$flag_i \leftarrow flag_{i-1}$
5:	根据(4.1)式,动作预测模块 N_θ 预测当前状态 s_i 下所有候选动作的可能概率 logits
6:	根据(4.2)式, 概率采样模块 $P_\theta(logits, flag_i)$ 获得当前 $flag_i$ 下所选动作 a_i
7:	# 环境交互
8:	环境收到 Actor 的决策结果 a_i 后, 给予反馈 r_i, s_{i+1}
9:	# Critic 评价
10:	根据(4.3)式计算 $V_\alpha(s_i)$ 和 $V_\alpha(s_{i+1})$
11:	根据(4.4)式计算优势函数 $A_\alpha(s_i, a_i)$
12:	# 参数更新
13:	根据(4.17)式计算得到 l_{value} 更新 Critic
14:	根据(4.16)式和(4.18)式计算得到 l_{policy} 和 $l_{entropy}$
15:	根据(4.15)式计算得到 L_{tot} 更新 Actor
16:	end for

4.3.4 换道决策问题要素设计

本节详细介绍了自动驾驶换道决策问题中状态空间、动作空间以及奖励函数的设计。具体来说：

(1) 状态空间设计

自由换道决策是综合考虑自车状态以及周围车辆状态来决定是否需要进行换道决策的过程。首先，需要识别决策过程中涉及的车辆及其状态，例如当前自我车辆所在车道的前方车辆行驶速度比限速慢、前方车辆的横向偏移和横向速度会等都会影响换道过程。因此，自我车辆当前车道的周围车辆的行驶状态是重要的换道决策因素；左右相邻车道是潜在的换道目标，因此需要相邻车道上前后车的行驶状态信息。

状态空间 S 包括自我车辆的状态信息、周围车辆的状态信息以及道路结构

信息。假设自车处于中间车道，其最直观视野中影响换道决策的关键车辆包括自我车辆前车、自我车辆后车，自我车辆相邻车道前车以及自我车辆相邻车道后车。具体来说，自我车辆的状态主要由自车的位置、速度、加速度、航向角以及自车所在车道信息组成；周围车辆信息主要由纵向距离（与自我车辆的纵向间距）、横向距离（与自我车辆的横向间距）、速度差、加速度差、航向角以及所在车道信息组成。

状态空间 S 的具体表示如下：

$$S = \{S_t^{ego}, S_t^{sur}, S_t^{lane}\} \quad (4.5)$$

其中， S_t^{ego} 是自我车辆在时刻 t 的状态信息； S_t^{sur} 是周围车辆在时刻 t 的状态信息； S_t^{lane} 是不同车道上面车辆信息以及车道占用率信息。 S_t^{ego} 、 S_t^{sur} 、 S_t^{lane} 的具体设计如下：

$$S_t^{ego} = \{(x, y), v, v_a, \varphi, lane_{id}\} \quad (4.6)$$

其中， (x, y) 表示车的位置坐标， v 表示车辆速度， v_a 表示车辆加速度， φ 表示车辆航向角， $lane_{id}$ 表示车辆所在车道 id 。

$$S_t^{sur} = \{\Delta x, \Delta y, (x, y), \Delta v, \Delta v_a, \varphi, lane_{id}\} \quad (4.7)$$

其中， $\Delta x, \Delta y$ 表示周围车辆与自我车辆的横向和纵向距离差， $\Delta v, \Delta v_a$ 表示周围车辆和自我车辆的速度和加速度差值。

$$S_t^{lane} = (id, occupy) \quad (4.8)$$

其中， id 表示道路车道索引， $occupy$ 表示车道的占用率，计算公式如下：

$$occupy = U_{\max} \cdot e^{-k \cdot \frac{n}{l}} \quad (4.9)$$

其中， $U_{\max}$ 为车道最大利用率； k 为衰减系数，反映车辆密度对利用率的负面影响； n 是通过车辆数量； l 是车道长度。

综上所述，状态空间 S 通过整合自我车辆状态信息、周围车辆状态信息以及道路结构信息，形成了一个全面的交通场景描述，使得模型能够在复杂的交通环境中做出合理、安全且高效的换道决策。

(2) 动作空间设计

动作决策空间是指自动驾驶车辆在给定状态下可以采取的所有可能动作的集合。基于实际驾驶场景，车辆换道决策动作空间通常包括以下三种基本动作：向左换道、向右换道、不换道，因此，动作空间的设计如下：

$$A = \{0, 1, 2\} \quad (4.10)$$

其中，0 代表保持当前车道；1 代表向左换道；2 代表向右换道。

(3) 多目标奖励函数设计

本节通过设计多目标优化奖励函数来优化换道决策方法，具体从安全性、舒适性和效率性三个维度构建综合奖励函数。

安全奖励：驾驶安全性主要体现在自我车辆与周围车辆的安全距离、速度差值以及碰撞这三个方面。

$$R_{safe} = \begin{cases} -100, & \text{if collision} \\ -\partial \cdot \exp(\frac{d_{safe} - d}{\beta}) - \gamma \cdot \frac{\Delta v}{d + \epsilon} & \text{if } d < d_{safe} \\ 0 & \text{otherwise} \end{cases} \quad (4.11)$$

其中，如果车辆发生碰撞行为，直接给与-100 的奖励值；，如果车辆之间的距离小于安全距离 d_{safe} ，这里设置 $d_{safe}=2m$ ，则执行函数，其他其情况的奖励值为 0。

效率奖励：驾驶效率主要体现在自我车辆与周围车辆的车速变化率，具体设计如下：

$$R_{eff} = 1 - \frac{|v_{ego} - v_{sur}|}{v_{max}} \quad (4.12)$$

舒适奖励：驾驶舒适度包括纵向舒适度以及横向舒适度，具体设计如下：

$$R_{comf} = \varphi_1 \cdot (1 - \frac{d_a / d_t}{j_{max}}) + \varphi_2 \cdot (1 - \frac{a - a_{safe}}{a_{max} - a_{safe}}) \quad (4.13)$$

其中， φ_1 和 φ_2 表示权重系数； j_{max} 表示加速度的导数也就是加速度变化率；公式（4.10）中，前一项为纵向舒适性设计，后一项为横向舒适性设计。

总奖励：总奖励的具体设计如下，其中 $\lambda_1, \lambda_2, \lambda_3$ 是权重系数。具体来说，在

公式 4.11 中，车辆碰撞的奖励值被设定为-100，这凸显了安全性在自动驾驶换道决策中的绝对重要性，因此赋予其最高权重。舒适性权重的设定需满足在确保安全的基础上，有效限制激进的换道行为，从而优化乘客体验。效率性权重的设定需满足在确保安全性和舒适性的前提下，适度优化换道过程中的通行效率。通过随机试验方法发现，舒适性权重为 0.2，在确保安全的基础上，可有效限制激进的换道行为；当效率权重超过 0.15 时，其对通行时间的改善率已趋于饱和。最终，在本章中，安全性奖励的权重设置为 0.7；舒适性奖励的权重设置为 0.2；效率性奖励的权重设置为 0.1。

$$R_{tot} = \lambda_1 R_{safe} + \lambda_2 R_{eff} + \lambda_3 R_{comf} \quad (4.14)$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 1 \quad (4.15)$$

(4) 损失函数设计

损失函数由策略损失函数、价值损失函数以及熵正则项三部分组成。策略损失函数用于更新 Actor 网络，其目标是通过策略梯度方法最大化累积奖励，即通过选择更好的动作来提高策略性能。价值损失用于更新 Critic 网络，其目标是 minimized 估计值函数与实际回报之间的均方误差。熵正则项用于鼓励策略的探索性，增加策略的随机性，避免过早收敛到局部最优解。损失函数的具体设计如下：

$$L_{tot} = l_{policy} + l_{value} + l_{entropy} \quad (4.16)$$

其中 l_{policy} 代表策略损失函数、 l_{value} 代表价值损失函数、 $l_{entropy}$ 代表熵正则项。

策略损失函数的核心思想是通过优势函数加权动作概率的对数梯度，使高优势动作的概率提升，低优势动作的概率降低。 l_{policy} 的具体设计如公式 (4.17) 所示：

$$l_{policy} = -\frac{1}{N} \sum_{i=1}^N [\log \pi(a_i | s_i) \cdot A(s_i, a_i)] \quad (4.17)$$

其中，N 表示一个训练批次中的样本数量，即每次参数更新时使用的经验样本数量。它的作用是将梯度归一化到单样本贡献，避免批量大小不同导致更新幅度不稳定。 $\log \pi(a_i | s_i)$ 表示策略网络在状态 s_i 下选择动作 a_i 的概率的对数值； $A(s_i, a_i)$ 是优势函数，计算公式如 (4.4) 所示。它的作用是衡量在状态 s_i 下选择

动作 a_i 的好坏程度。

价值损失函数通过最小化预测值与目标值的均方误差，优化 Critic 网络的状态价值评估能力，为 Actor 网络提供可靠的优势估计。 l_{value} 的具体设计如公式 (4.18) 所示：

$$l_{value} = \frac{1}{N} \sum_{i=1}^N (V_{tar}(s_i) - V(s_i))^2 \quad (4.18)$$

$V_{tar}(s_i)$ 是目标状态价值， $V(s_i)$ 是 Critic 网络对当前状态 s_i 的价值预测值，平方误差的作用是衡量预测值与目标值的差距，放大较大误差的影响，促使 Critic 网络优先修正显著错误。

熵正则项的目的是鼓励智能体进行更多探索，防止策略过早收敛。的具体设计如公式 (4.19) 所示：

$$l_{entropy} = -\beta \cdot E[H(\pi(\cdot | s))] \quad (4.19)$$

当策略 π 对所有动作的选择概率越均匀时，熵值越大，表示策略的探索性越强；当策略 π 集中在少数动作（如某个动作概率接近 1）时，熵值趋近于 0，表示策略确定性高但探索性弱。

4.4 实验与分析

4.4.1 SUMO 仿真实验环境

本章采用 SUMO 仿真平台搭建了一个环形路场景来进行模拟实验。环形路的建模如图 4-5 所示，其中：子图是环形路的整体布局，包括直路和弯路的连接方式；子图 b 是直路车道的结构；子图 c 是弯路车道的结构。

在该路网结构中，道路采用单向三车道设计，每条车道宽度为 3.5 米。其中，两段直行道路的长度均为 305 米，限速为 130 公里/小时；弯道部分的长度为 55 米，限速为 90 公里/小时。环境中的车辆主要分为两种，颜色为红色的自我车辆以及颜色为红色的周围车辆，如图 4-5 的子图 c 所示。

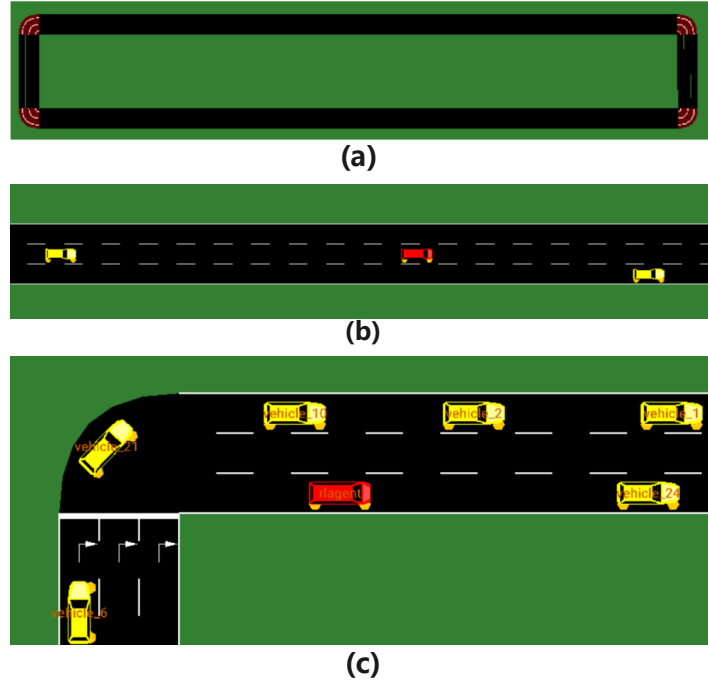


图 4-5 SUMO 环形路交通建模图

Figure 4-5 Sumo-based traffic modeling diagram for a roundabout

4.4.2 评估指标

为评估自动驾驶车辆换道决策的综合性能，本研究选取碰撞率（Collision rate）、换道成功率（Success rate）累积奖励（ R_{tot} ）三个指标^[37]作为关键评价维度。这些指标能够全面反映换道决策的安全性、有效性和优化效果。

1. **Collision rate**：碰撞率是衡量自动驾驶车辆安全性的重要指标。较低的碰撞率表明车辆在换道时能够更好地感知周围环境，避免与他车或障碍物发生碰撞，从而保障乘客和道路安全。Collision rate 指标越小越好；

2. **Success rate**：换道成功率反映了自动驾驶车辆在换道任务中的有效性和可靠性。较高的换道成功率意味着车辆能够更准确地判断换道时机，更顺利地完成任务。Success rate 指标越高越好；

3. R_{tot} ：总奖励是一个综合性能指标，能够全面反映自动驾驶车辆在换道过程中的表现， R_{tot} 指标越高越好。

4.4.3 实验结果分析

为验证本章提出的多目标换道决策优化方法 MOMA2C 的有效性，将其与基于规则的 DQN 方法^[37]在相同实验环境下进行了对比实验。不同模型性能对比如

表 4-2 所示。

表 4-2 不同方法在换道决策中的结果对比表

Table 4-2 Comparison table of results of different methods in lane changing decision

Model	Metrics		
	<i>Collision rate</i> ↓	<i>Success rate</i> ↑	R_{tot} ↑
DQN_based policy	0.01092	0.8752	31.1633
MOMA2C (ours)	0.01031	0.8800	37.6548

MOMA2C 相较于 DQN 方法在多个指标上均有提升。首先，Collision rate 从 0.01092 降低至 0.01031，相对提升 5.59%。其次，Success rate 从 0.8752 提升至 0.8800，相对提升 0.55%。最后， R_{tot} 从 31.1633 提升至 37.6548，相对提升 20.83%，反映了 MOMA2C 在多目标协同优化上的优势。

图 4-6 展示了强化学习过程中的累积奖励变化情况。在初始阶段，累积奖励从接近零快速攀升至约 40 的峰值，反映策略探索初期的学习过程。在中期阶段，数值在 20-40 区间频繁起伏，体现强化学习中的策略调优与探索-利用平衡。在后期阶段，波动幅度较小，数值趋于 30-40，表明策略逐步收敛至较优解。

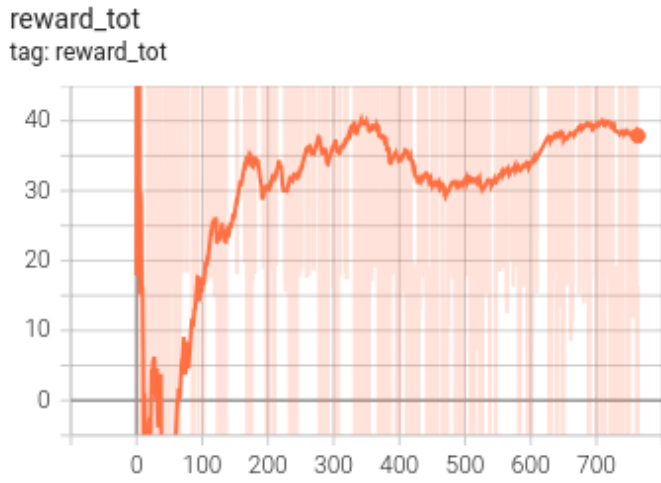


图 4-6 累积奖励变化趋势图

Figure 4-6 Trend graph of cumulative reward

MOMA2C 与 DQN 速度效率对比图如图 4-7 所示。图的横轴是训练轮数，这里我们按照规律取了 5, 500, 1000, 1500, 2000；纵轴是 rlagent 当前速度与目标速度的百分比，也就是速度效率比。在训练初期，MOMA2C 算法展现出显著优

势，速度百分比达到 0.29，表明其 Actor-Critic 框架在早期探索阶段能够迅速生成有效策略。相比之下，DQN 算法初始探索效率较低，速度百分比为 0.27，反映出其依赖经验回放的特性需要更多数据积累。在训练中期，DQN 算法反超 MOMA2C，500 轮时速度百分比升至 0.28，表明其离线学习机制在数据量积累后开始生效，Q 值估计更为精准。而 MOMA2C 算法在 500 轮时降至 0.24，这是因为 Critic 网络早期价值评估不稳定，导致策略梯度更新幅度过大；1000 轮时两者持平（0.26），显示 MOMA2C 算法进入策略调整平衡期。在训练后期，MOMA2C 算法的稳定性凸显，1500 轮回升至 0.28，2000 轮保持 0.25，说明其在线策略更新能够持续优化，避免陷入局部最优。

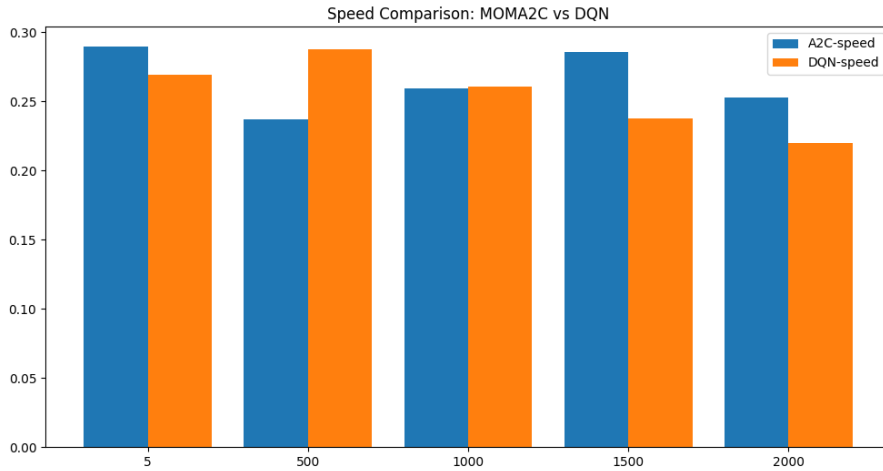


图 4-7 A2C 与 DQN 速度效率比对比图

Figure 4-7 Comparison plot of speed efficiency ratio between A2C and DQN

训练所用行驶场景为环形路，车辆速度随路径实况变化而变化，如图 4-8 所示。在实验中，除 `rlagent` 外，其他车辆的速度变化相似且均呈现周期性波动。此外，由于 `rlagent` 所设计的多目标奖励函数需平衡车辆速度、效率、安全性和舒适度等诸多性能，故在 0m-600m 内呈现明显区别于其他车辆的速度变化。

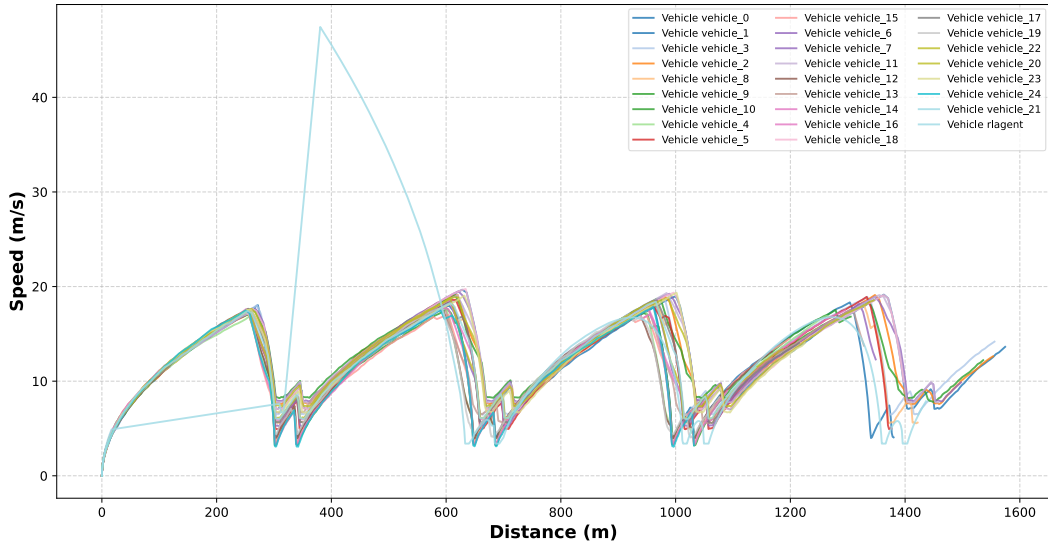


图 4-8 车辆速度变化图

Figure 4-8 Graph of vehicle speed change

具体而言，以 `rlagent` 与 `vehicle_2` 对比为例，如图 4-9 所示。`vehicle_2` 面对平直道路时，会保持加速度不断，稳定加速。当速度过高时，为避免负奖励，`vehicle_2` 进行稳定减速；面对转弯道路时，由于弯道阻塞需要对过往车辆进行避让，`vehicle_2` 按照实际车况进行加速再减速。车辆再行驶过程中根据直道-弯道-直道-弯道-直道-弯道的变化对速度进行调节，但整个过程缺乏动态调整。面对突发状况，无法进行紧急避险等操作。而 `rlagent` 算法则呈现出明显不同。`rlagent` 在第三个直道前（0m-700m），为学习准确奖励函数对多目标进行权衡，`rlagent` 通过探索新动作和利用已知动作对速度进行调整，故产生明显波动。当进入第三个直道时，`rlagent` 已充分学习到良好策略应对直弯道切换需求。面对路况变化时，`rlagent` 可以通过对加速度进行缓慢调整进而提供平稳的速度变化，为乘客提供更加舒适的乘车体验。

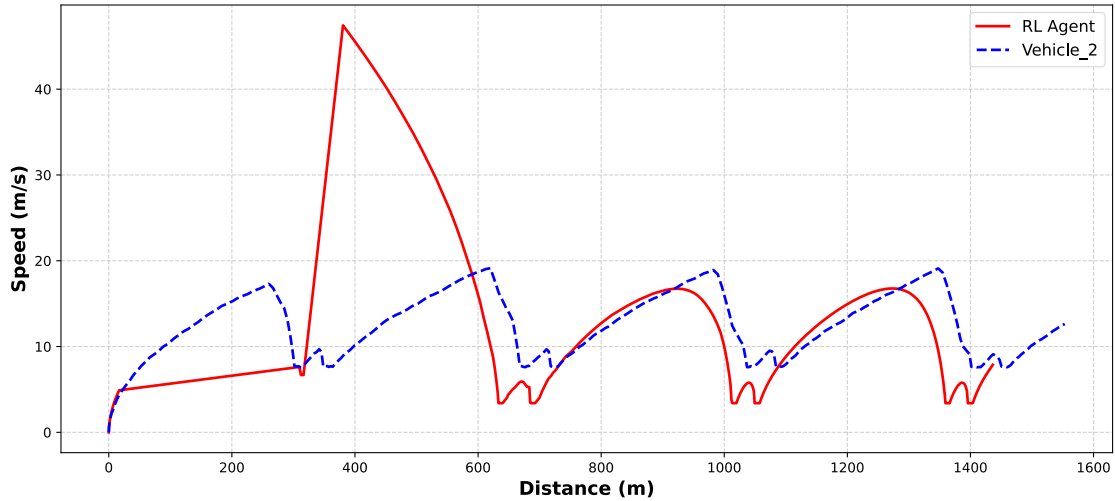


图 4-9 rlagent 与 vehicle_2 的行驶速度对比图

Figure 4-9 Comparison of driving speed between rlagent and vehicle_2

在训练过程中, rlagent 与 vehicle_2 的航向角变化如图 4-10 所示, 可以看到, rlagent 相较于 vehicle_2 转向调整更早且幅度更大。在 0-500m 距离段: rlagent 在初期即主动调整方向, 体现探索性策略; vehicle_2 采用保守跟随策略, 仅在环境压力下被动转向。在 500-1600m 距离段: rlagent 相较于 vehicle_2, 更早转弯, 这更加说明了 rlagent 在面对路况变化时, 能更早的做出合理决策。

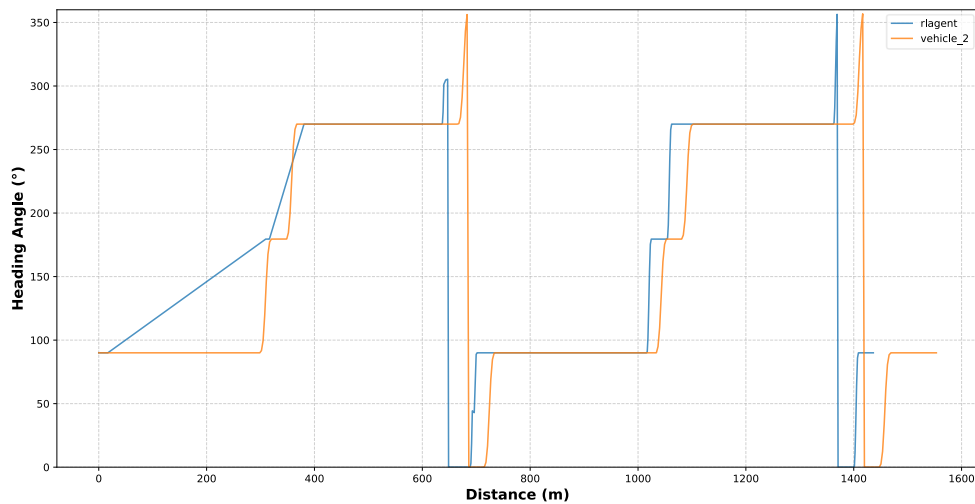


图 4-10 rlagent 与 vehicle_2 的航向角对比图

Figure 4-10 Comparison of heading angles between rlagent and vehicle_2

在训练过程中, rlagent 在行驶过程中加速的变化如图 4-11 所示。在 0-200m 距离段: rlagent 的加速度频繁波动, 这是因为 Actor 网络通过策略尝试不同加速度动作, 收集环境反馈的奖励; Critic 网络根据奖励指导 Actor 网络调整

动作概率分布。在 200-600m 距离段：rlagent 的加速度急剧下降，这是因为 Critic 网络评估到弯道风险，触发减速。在 600-1600m 距离段：rlagent 的加速度呈现规律性变化，说明了策略收敛且与场景适应。而 vehicle_2 的加速度在全程呈现均匀波动 (± 0.3)，说明 vehicle_2 未能适应环境变化，未学到有效的应对策略。

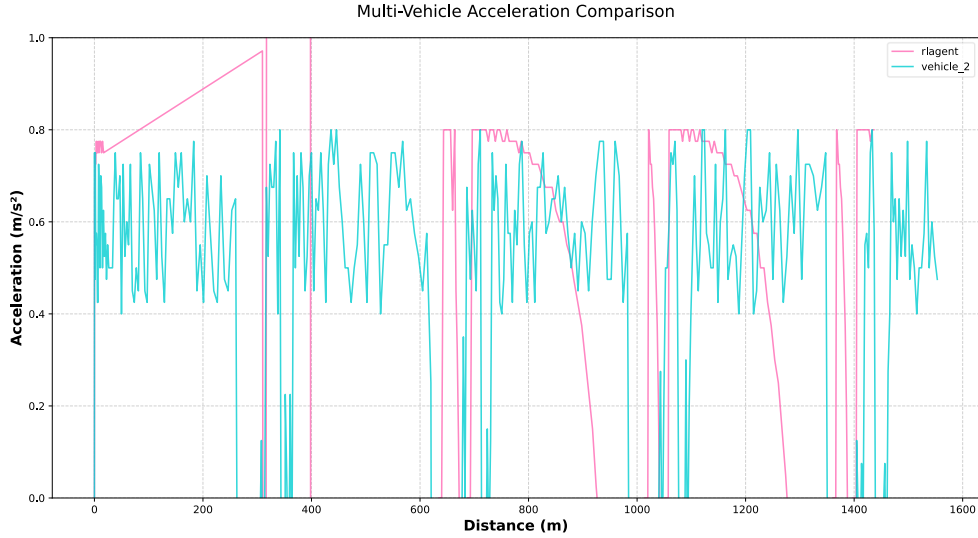


图 4-11 rlagent 与 vehicle_2 加速度变化图

Figure 4-11 Plot of acceleration changes between rlagent and vehicle_2

4.5 本章小结

本章首先介绍了自动驾驶换道决策问题是什么，阐述了自由换道与强制换道的定义与区别。然后本章对复杂场景下自动驾驶车辆换道决策问题进行了方法设计，提出了基于强化学习的多目标车辆换道决策优化方法，命名为 MOMA2C。MOMA2C 以 A2C (Advantage Actor-Critic) 强化学习算法为基础框架，通过优化安全性、舒适性和效率性等多重优化目标，实现对自我车辆的换道决策。本章将 MOMA2C 与基于规则的 DQN Agent 在相同实验环境下进行了对比实验，实验结果表明，MOMA2C 在换道决策问题方面取得了较好的效果。

第 5 章 总结与展望

5.1 工作总结

自动驾驶汽车的技术正在不断从 L2 级向 L3 级、L4 级发展。随着现代科学技术的进步，各大研究机构和企业也在持续升级自动驾驶技术。然而，这一发展过程仍面临诸多困难和挑战。本文针对面向复杂环境的自动驾驶决策问题进行了研究。首先，对自动驾驶技术进行了简要介绍，并分析了影响自动驾驶决策的两大关键因素：一是周围车辆的运动轨迹影响自我车辆一段时间的决策；另一个是换道决策问题。针对这两个问题，提出了相应的解决方案。主要工作总结如下：

(1) 针对第一个问题，即周围车辆运动轨迹对自动驾驶车辆决策的影响，设计并实现了基于 Informer 模型的多模态轨迹预测方法 MTPI。该方法通过结合自注意力机制 (Self-attention) 和 ProbSparse 自注意力操作，显著降低了计算复杂度并提高了内存使用效率，从而能够高效地处理多车辆的轨迹预测任务。本文将 MTPI 与 CV (Constant Velocity)、LSTM+map(prior) 1-G,n-C、DiversityGAN 这三种轨迹预测方法进行了比较，实验结果表明，MTPI 在轨迹预测精度方面取得了较好的效果。

(2) 针对第二个问题，即换道决策问题，设计并实现了基于强化学习的多目标车辆换道决策优化方法 MOMA2C。MOMA2C 以 A2C (Advantage Actor-Critic) 强化学习算法为基础框架，通过优化安全性、舒适性和效率性等多重优化目标，实现对自我车辆的换道决策。具体而言，MOMA2C 设计了多目标奖励函数，分别从避免碰撞、保持安全距离、平滑加速度变化以及优化行驶速度等方面进行优化，从而在复杂交通场景中实现安全、舒适且高效的换道行为。将 MOMA2C 与基于规则的 DQN Agent 在相同实验环境下进行了对比实验，实验结果表明，MOMA2C 在换道决策问题方面取得了较好的效果。

5.2 工作展望

本文面向复杂环境的自动驾驶方法展开研究，针对周围车辆运动轨迹对自动驾驶车辆决策的影响问题，提出了基于 Informer 模型的多模态轨迹预测方法

MTPI；针对换道决策问题，提出了基于强化学习的多目标车辆换道决策优化方法 MOMA2C。通过对比实验，本文所提出的 MTPI 和 MOMA2C 方法在复杂环境下的自动驾驶决策中均实现了良好的性能。为了进一步提升面向复杂环境的自动驾驶决策方法研究，未来的工作主要聚焦于以下几点：

（1）针对周围车辆运动轨迹对自动驾驶车辆决策的影响问题，提出的方法 MTPI 在轨迹预测的精度和降低复杂度方面取得了良好的性能。然而，目前的实验仅基于单个数据集进行验证，未来计划通过引入更多多样化的数据集来进一步丰富实验，以验证 MTPI 方法的可行性和泛化能力。此外，在多模态轨迹预测任务中，模型的可解释性对于实际应用至关重要。为了直观展示 MTPI 的预测能力，本文采用了可视化技术，在四种不同场景下呈现了轨迹预测结果。通过对比预测轨迹与真实轨迹，可以清晰地评估模型在多模态轨迹生成中的准确性及其与实际情况的一致性。在接下来的工作中，将重点关注通过改进模型的决策规则来增强模型的可解释性。最后，真实世界场景的复杂性远高于当前实验所用的测试场景，未来研究将着重针对这一方面进行优化和改进。

（2）针对换道决策问题，提出的方法 MOMA2C 在换道决策任务中取得了良好的性能。实验结果表明，MOMA2C 在安全性、舒适性和效率性等多个目标上优于传统的基于规则的 DQN 方法。然而，尽管 MOMA2C 方法在换道决策任务中取得了显著成效，但仍存在进一步优化的空间。在复杂场景适应性方面，虽然 MOMA2C 在现有实验场景中表现良好，但其在更复杂的道路结构中的性能仍需进一步验证。通过构建更多多样化的道路场景，可以更全面地评估算法的适应性和鲁棒性。在多智能体协同决策方面，当前的 MOMA2C 主要针对单车决策进行优化，未来可以引入更复杂的多智能体交互机制以及扩展动作设计维度，例如基于博弈论或协同强化学习的策略，以实现车辆之间的协同换道和避让，从而优化整体交通流的效率并减少竞争性行为导致的交通拥堵。

参考文献

- [1] 公安部发布 2024 年度全国机动车、驾驶人数据[J]. 道路交通管理, 2025(01): 9.
- [2] 饶永明, 张延孔, 谢文军, 等. 交通事故时空模式可视分析方法[J]. 计算机科学, 2019, 46(04): 14-21.
- [3] 王祥, 金正烽. 我国道路交通事故发展趋势研究[J]. 交通科技与经济, 2019, 21(03): 29-32+37.
- [4] Yurtsever E, Lambert J, Carballo A, et al. A Survey of Autonomous Driving: Common Practices and Emerging Technologies[J]. IEEE Access, 2020, 8: 58443-58469.
- [5] 朱冰, 贾士政, 赵健, 等. 自动驾驶车辆决策与规划研究综述[J]. 中国公路学报, 2024, 37(01): 215-240.
- [6] 魏强, 陆平, 侯雪. 基于专利数据的全球自动驾驶感知技术创新态势[J]. 智能网联汽车, 2020(01): 74-77.
- [7] 王婧. 中国智能汽车政策现状及完善研究[D]. 上海: 上海交通大学, 2018.
- [8] 孙超, 黄愉文, 张凯, 等. 智能网联汽车产业政策趋势分析及发展思考[J]. 城市交通, 2022, 20(01): 52-58.
- [9] Talpes E, Sarma D D, Venkataramanan G, et al. Compute Solution for Tesla's Full Self-Driving Computer[J]. IEEE Micro, 2020, 40(2): 25-35.
- [10] Ettinger S, Cheng S, Caine B, et al. Large Scale Interactive Motion Forecasting for Autonomous Driving : The Waymo Open Motion Dataset[C]. 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021: 9690-9699.
- [11] Carvalho M V L D, Simoni R, Yoshioka L R, et al. A Performance Evaluation of Open Source Autonomous Driving Frameworks: Case studies of Apollo and Autoware[J]. IEEE Access, 2025: 1-1.
- [12] Chen L, Li Y, Huang C, et al. Milestones in Autonomous Driving and Intelligent Vehicles: Survey of Surveys[J]. IEEE Transactions on Intelligent Vehicles, 2023, 8(2): 1046-1056.

-
- [13] Chougule A, Chamola V, Sam A, et al. A Comprehensive Review on Limitations of Autonomous Driving and Its Impact on Accidents and Collisions[J]. IEEE Open Journal of Vehicular Technology, 2024, 5: 142-161.
- [14] Huang R, Xue H, Pagnucco M, et al. Multimodal Trajectory Prediction: A Survey[J]. ArXiv, 2023, abs/2302.10463.
- [15] Dyckmanns H, Matthaei R, Maurer M, et al. Object tracking in urban intersections based on active use of a priori knowledge: Active interacting multi model filter[C]. 2011 IEEE Intelligent Vehicles Symposium (IV), 2011: 625-630.
- [16] Streubel T, Hoffmann K H. Prediction of driver intended path at intersections[C]. 2014 IEEE Intelligent Vehicles Symposium Proceedings, 2014: 134-139.
- [17] Brännström M, Coelingh E, Sjöberg J. Model-Based Threat Assessment for Avoiding Arbitrary Vehicle Collisions[J]. IEEE Transactions on Intelligent Transportation Systems, 2010, 11(3): 658-669.
- [18] Kasper D, Weidl G, Dang T, et al. Object-Oriented Bayesian Networks for Detection of Lane Change Maneuvers[J]. IEEE Intelligent Transportation Systems Magazine, 2012, 4(3): 19-31.
- [19] Cui H, Radosavljevic V, Chou F C, et al. Multimodal Trajectory Predictions for Autonomous Driving using Deep Convolutional Networks[C]. 2019 International Conference on Robotics and Automation (ICRA), 2019: 2090-2096.
- [20] Li L, Chen Z, Wang J, et al. Multimodal Trajectory Prediction for Autonomous Driving on Unstructured Roads using Deep Convolutional Network[J]. ArXiv, 2024, abs/2409.18399.
- [21] Deo N, Trivedi M M. Multi-Modal Trajectory Prediction of Surrounding Vehicles with Maneuver based LSTMs[C]. 2018 IEEE Intelligent Vehicles Symposium (IV), 2018: 1179-1184.
- [22] Xin L, Wang P, Chan C Y, et al. Intention-aware Long Horizon Trajectory Prediction of Surrounding Vehicles using Dual LSTM Networks[C]. 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018: 1441-1446.

-
- [23] Giuliari F, Hasan I, Cristani M, et al. Transformer Networks for Trajectory Forecasting[C]. 2020 25th International Conference on Pattern Recognition (ICPR), 2021: 10335-10342.
- [24] Liu Y, Zhang J, Fang L, et al. Multimodal Motion Prediction with Stacked Transformers[C]. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021: 7573-7582.
- [25] Weng X, Ivanovic B, Wang Y, et al. PARA-Drive: Parallelized Architecture for Real-Time Autonomous Driving[C]. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024: 15449-15458.
- [26] Feng K, Li C, Ren D, et al. On the Road to Portability: Compressing End-to-End Motion Planner for Autonomous Driving[C]. 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024: 15099-15108.
- [27] Lv K, Pei X, Chen C, et al. A Safe and Efficient Lane Change Decision-Making Strategy of Autonomous Driving Based on Deep Reinforcement Learning[J]. Mathematics, 2022.
- [28] Gipps P G. A MODEL FOR THE STRUCTURE OF LANE-CHANGING DECISIONS[J]. Transportation Research Part B-methodological, 1986, 20: 403-414.
- [29] Lucken L. Resolving Collisions for the Gipps Car-Following Model[J]. arXiv: Adaptation and Self-Organizing Systems, 2019.
- [30] Hu Y, Yan L, Zhan J, et al. Decision-making System based on Finite State Machine for Low-speed Autonomous Vehicles in the Park[C]. 2022 IEEE International Conference on Real-time Computing and Robotics (RCAR), 2022: 721-726.
- [31] Bae S H, Joo S H, Pyo J W, et al. Finite State Machine based Vehicle System for Autonomous Driving in Urban Environments[C]. 2020 20th International Conference on Control, Automation and Systems (ICCAS), 2020: 1181-1186.
- [32] Kanaris A, Kosmatopoulos E B, Loannou P A. Strategies and spacing requirements for lane changing and merging in automated highway systems[J]. IEEE Transactions on Vehicular Technology, 2001, 50(6): 1568-1581.

-
- [33] Hsu H C H, Liu A. Kinematic Design for Platoon-Lane-Change Maneuvers[J]. IEEE Transactions on Intelligent Transportation Systems, 2008, 9(1): 185-190.
 - [34] Guan M, Wang Z, Yang B, et al. A Classified Driver's Lane-Change Decision-Making Model Based on Fuzzy Inference for Highly Automated Driving[C]. 2021 IEEE 2nd International Conference on Human-Machine Systems (ICHMS), 2021: 1-4.
 - [35] Hu W, Deng Z, Cao D, et al. Probabilistic Lane-Change Decision-Making and Planning for Autonomous Heavy Vehicles[J]. IEEE/CAA Journal of Automatica Sinica, 2022, 9(12): 2161-2173.
 - [36] Wen Z, Zhang Y, Chen X, et al. TOFG: A Unified and Fine-Grained Environment Representation in Autonomous Driving[J]. 2023 IEEE International Conference on Robotics and Automation (ICRA), 2023: 1565-1571.
 - [37] Shi T, Wang P, Cheng X, et al. Driving Decision and Control for Automated Lane Change Behavior based on Deep Reinforcement Learning[C]. 2019 IEEE Intelligent Transportation Systems Conference (ITSC), 2019: 2895-2900.
 - [38] Ye F, Cheng X, Wang P, et al. Automated Lane Change Strategy using Proximal Policy Optimization-based Deep Reinforcement Learning[C]. 2020 IEEE Intelligent Vehicles Symposium (IV), 2020: 1746-1752.
 - [39] Mirchevska B, Hügle M, Kalweit G, et al. Amortized Q-learning with Model-based Action Proposals for Autonomous Driving on Highways[C]. 2021 IEEE International Conference on Robotics and Automation (ICRA), 2021: 1028-1035.
 - [40] Yang D, Zheng S, Wen C, et al. A dynamic lane-changing trajectory planning model for automated vehicles[J]. Transportation Research Part C: Emerging Technologies, 2018, 95: 228-247.
 - [41] Deshmukh V M, R B, Krishna G B, et al. An Overview of Deep Learning Techniques for Autonomous Driving Vehicles[C]. 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), 2022: 979-983.
 - [42] Atakishiyev S, Salameh M, Yao H, et al. Explainable Artificial Intelligence for Autonomous Driving: A Comprehensive Overview and Field Guide for Future

- Research Directions[J]. IEEE Access, 2024, 12: 101603-101625.
- [43] Zhan H, Wan D, Huang Z. On the Responsible Subjects of Self-Driving Cars under the SAE System: An Improvement Scheme[C]. 2020 IEEE International Symposium on Circuits and Systems (ISCAS), 2020: 1-5.
- [44] Jiang H, Niu Y, Hu C, et al. Motion Query-based Multimodal Vehicle Trajectory Prediction for Autonomous Driving[C]. 2023 7th CAA International Conference on Vehicular Control and Intelligence (CVCI), 2023: 1-6.
- [45] Tang Y, He H, Wang Y, et al. Using a Diffusion Model for Pedestrian Trajectory Prediction in Semi-Open Autonomous Driving Environments[J]. IEEE Sensors Journal, 2024, 24(10): 17208-17218.
- [46] Gao J, Sun C, Zhao H, et al. VectorNet: Encoding HD Maps and Agent Dynamics From Vectorized Representation[C]. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020: 11522-11530.
- [47] Guo L, Liu X. Lane-Changing Decisions Making for Autonomous Vehicles via Behavior Cloning and Decision Tree[C]. 2023 China Automation Congress (CAC), 2023: 8648-8652.
- [48] Wang X, Hu J, Wei C, et al. A Novel Lane-Change Decision-Making With Long-Time Trajectory Prediction for Autonomous Vehicle[J]. IEEE Access, 2023, 11: 137437-137449.
- [49] Kaur M, Mohta A. A Review of Deep Learning with Recurrent Neural Network[C]. 2019 International Conference on Smart Systems and Inventive Technology (ICSSIT), 2019: 460-465.
- [50] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8): 1735-1780.
- [51] Vaswani A, Shazeer N, Parmar N, et al. Attention is All you Need[C], 2017: 5998-6008.
- [52] Jain G, Kumar A, Bhat S A. Recent Developments of Game Theory and Reinforcement Learning Approaches: A Systematic Review[J]. IEEE Access, 2024, 12: 9999-10011.

-
- [53] Ahmadi C, Chen J L. Survey on Reinforcement Learning Techniques for Enhancing Security and Efficiency in Zero Trust Networks[C]. 2024 10th International Conference on Applied System Innovation (ICASI), 2024: 427-429.
- [54] Liu H, Qu W, Jia J, et al. Pre-trained Encoders in Self-Supervised Learning Improve Secure and Privacy-preserving Supervised Learning[C]. 2024 IEEE Security and Privacy Workshops (SPW), 2024: 144-156.
- [55] Chu S. Leverage of generative adversarial model for boosting exploration in deep reinforcement learning[C]. 2024 6th International Conference on Internet of Things, Automation and Artificial Intelligence (IoTAAI), 2024: 315-318.
- [56] Kaloev M, Krastev G. Comprehensive Review of Benefits from the Use of Sparse Updates Techniques in Reinforcement Learning: Experimental Simulations in Complex Action Space Environments[C]. 2023 4th International Conference on Communications, Information, Electronic and Energy Systems (CIEES), 2023: 1-7.
- [57] Khanzhahi N, Masoumi B, Karasfi B. Deep Reinforcement Learning Issues and Approaches for The Multi-Agent Centric Problems[C]. 2018 9th Conference on Artificial Intelligence and Robotics and 2nd Asia-Pacific International Symposium, 2018: 87-95.
- [58] Jiang W, Bao C, Xu G, et al. Research on Autonomous Obstacle Avoidance and Target Tracking of UAV Based on Improved Dueling DQN Algorithm[C]. 2021 China Automation Congress (CAC), 2021: 5110-5115.
- [59] Kővári B, Pelenczei B, Bécsi T. Monte Carlo Tree Search to Compare Reward Functions for Reinforcement Learning[C]. 2022 IEEE 16th International Symposium on Applied Computational Intelligence and Informatics (SACI), 2022: 000123-000128.
- [60] Siboo S, Bhattacharyya A, Raj R N, et al. An Empirical Study of DDPG and PPO-Based Reinforcement Learning Algorithms for Autonomous Driving[J]. IEEE Access, 2023, 11: 125094-125108.
- [61] Ji P, Li R, Xue Y, et al. Perspective, Survey and Trends: Public Driving Datasets

- and Toolsets for Autonomous Driving Virtual Test[C]. 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), 2021: 264-269.
- [62] Coifman B, Li L. A critical evaluation of the Next Generation Simulation (NGSIM) vehicle trajectory dataset[J]. Transportation Research Part B: Methodological, 2017, 105: 362-377.
- [63] Krajewski R, Bock J, Kloecker L, et al. The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems[C]. 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018: 2118-2125.
- [64] Chang M F, Lambert J, Sangkloy P, et al. Argoverse: 3D Tracking and Forecasting With Rich Maps[C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019: 8740-8749.
- [65] Mo X, Huang Z, Xing Y, et al. Multi-Agent Trajectory Prediction With Heterogeneous Edge-Enhanced Graph Attention Network[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(7): 9554-9567.
- [66] Wilson B, Qi W, Agarwal T, et al. Argoverse 2: Next Generation Datasets for Self-Driving Perception and Forecasting[J]. CoRR, 2023, abs/2301.00493.
- [67] Krajzewicz D, Hertkorn G, Rössel C, et al. SUMO (Simulation of Urban MObility) - an open-source traffic simulation[C], 2002.
- [68] Dosovitskiy A, Ros G, Codevilla F, et al. CARLA: An Open Urban Driving Simulator[C]. Conference on Robot Learning, 2017.
- [69] Li Q, Peng Z, Feng L, et al. MetaDrive: Composing Diverse Driving Scenarios for Generalizable Reinforcement Learning[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(3): 3461-3475.
- [70] Zhou M, Luo J, Villela J, et al. SMARTS: Scalable Multi-Agent Reinforcement Learning Training School for Autonomous Driving[J]. ArXiv, 2020, abs/2010.09776.
- [71] Li D, Zhang Q, Xia Z, et al. Planning-Inspired Hierarchical Trajectory Prediction via Lateral-Longitudinal Decomposition for Autonomous Driving[J]. IEEE

- Transactions on Intelligent Vehicles, 2024, 9(1): 692-703.
- [72] 王庆荣, 郝福乐, 朱昌锋, et al. 基于多特征融合的车辆轨迹预测研究[J]. 计算机工程, 2025: 1-14.
- [73] 任佳佳, 柳寅奎, 胡学敏, et al. 面向复杂交通场景的自动驾驶运动规划模型[J]. 计算机工程与应用, 2024, 60(15): 91-100.
- [74] Chen J, Chen G, Li Z, et al. Multimodal Vehicle Trajectory Prediction Based on Graph Convolutional Networks[C]. 2022 International Conference on Advanced Robotics and Mechatronics (ICARM), 2022: 605-610.
- [75] Zhou H, Zhang S, Peng J, et al. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting[C], 2021: 11106-11115.
- [76] Huang X, McGill S G, Decastro J A, et al. DiversityGAN: Diversity-Aware Vehicle Motion Prediction via Latent Semantic Sampling[J]. IEEE Robotics and Automation Letters, 2020, 5(4): 5089-5096.
- [77] He X, Yang H, Hu Z, et al. Robust Lane Change Decision Making for Autonomous Vehicles: An Observation Adversarial Reinforcement Learning Approach[J]. IEEE Transactions on Intelligent Vehicles, 2023, 8(1): 184-193.
- [78] Mukhopadhyay R, Bandyopadhyay S, Sutradhar A, et al. Performance Analysis of Deep Q Networks and Advantage Actor Critic Algorithms in Designing Reinforcement Learning-based Self-tuning PID Controllers[C]. 2019 IEEE Bombay Section Signature Conference (IBSSC), 2019: 1-6.
- [79] Zhang S, Zhuang W, Li B, et al. Integration of Planning and Deep Reinforcement Learning in Speed and Lane Change Decision-Making for Highway Autonomous Driving[J]. IEEE Transactions on Transportation Electrification, 2025, 11(1): 521-535.
- [80] Li S, Liu C, Chen W H. An Integrated MPC Decision-Making Method Based on MDP for Autonomous Driving in Urban Traffic[C]. 2024 6th International Conference on Industrial Artificial Intelligence (IAI), 2024: 1-6.
- [81] 闫慧敏. 网联自动驾驶环境下城市路网车道动态利用研究[D]. 北京: 清华大学, 2023.

攻读学位期间发表的学术论文目录

[1] mengting Sun, jun Li.et al. MTPI: Multimodal Trajectory Prediction for Autonomous Driving via Informer [C]. 2024 IEEE 24th International Conference on Software Quality, Reliability, and Security Companion (QRS-C), 2024: 569-576. (第一作者, CCF-C, 已发表)

攻读学位期间比赛获奖目录

- [1] 第十九届全国大学生智能车汽车竞赛 (安徽省二等奖)
- [2] 安徽工程大学计算机能力挑战赛程序设计赛 (校三等奖)
- [3] 第三届全国大学生数据分析科普竞赛系列活动之理论赛 (二等奖)

攻读学位期间参与科研项目

- [1] “基于 5G 网络的智能交互展示系统及应用研究” 项目号: 2023cyd04

致谢

时光荏苒，转眼间我的研究生生涯即将画上句号。回首这段求学之路，充满了挑战与收获，也让我深刻体会到学术研究的艰辛与乐趣。在这篇论文完成之际，我怀着无比感激的心情，向所有在我求学路上给予帮助和支持的老师、同学、家人和朋友致以最诚挚的谢意。

首先，我尤为的感激我的导师，李钧老师。从论文选题到研究方法的确定，从实验设计到最终成稿，导师始终以严谨的治学态度和渊博的学识为我指明方向。每当我遇到困难时，导师总是耐心地为我答疑解惑，帮助我突破瓶颈。导师不仅在学术上给予我悉心指导，更以其高尚的品格和敬业精神深深影响了我，让我明白了做学问与做人同样重要的道理。导师的言传身教将是我一生受用的宝贵财富。

感谢 **Prise** 实验室的全体老师和同学们。实验室浓厚的学术氛围和团结协作的精神让我受益匪浅。每周六组会的论文分享和进度汇报，更加督促我学习和帮助我进步。感谢小组中王老师、卢老师、唐老师以及其他老师在组会上给予的学术指导以及其他帮助。特别感谢刘江师兄在我刚进入实验室时给予的耐心指导，帮助我快速熟悉研究方向和相关技术；感谢杨小涵同学在实验过程中与我并肩作战，共同探讨问题、分享经验；感谢陈宝柱师弟、徐恺师弟、杜浩杰师弟和李国俊师弟在数据收集和实验验证中提供的帮助。感谢你们共同营造了一种和谐、友好的学习氛围，同时感谢你们的知识分享，真的让我收获很多。

感谢我的家人。父母多年来无私的付出和支持是我不断前行的动力。他们始终尊重我的选择，鼓励我追求自己的梦想。每当我遇到挫折时，家人总是第一时间给予我温暖的关怀和坚定的信心。他们的爱与包容让我在学术道路上走得更加坚定和从容。

感谢我的朋友们。在研究生期间，我有幸结识了一群志同道合的伙伴。我们一起讨论学术问题，分享生活中的点滴，互相鼓励、共同进步。感谢学校和学院提供的优质资源和平台。学校图书馆丰富的文献资源为我的研究提供了坚实的理论基础；学院组织的学术讲座和研讨会让我开阔了视野，了解了领域内的最新动态。同时，感谢所有授课老师的辛勤付出，他们的课程让我打下了扎实的专业基础。研究生阶段的学习不仅让我掌握了专业知识，更让我学会了如何独立思考、

如何面对挑战、如何与他人合作。这段经历将成为我人生中一段难忘的回忆，也为我未来的职业发展指明了方向。在未来的日子里，我将继续秉持严谨求实的学术态度，努力在自己的领域内做出更多有价值的贡献。

最后，我要感谢所有参与论文评审和答辩的专家和老师。感谢你们在百忙之中抽出时间审阅我的论文，并提出宝贵的意见和建议。你们的专业指导将帮助我进一步完善研究工作，为未来的学术道路奠定更加坚实的基础。