分类号：TP301.6　　学校代码：10363
密　级：公开　　学　　号：2220110168

安徽工程大学
Anhui Polytechnic University

硕士学位论文

题目　基于深度强化学习的无人驾驶车辆行为决策研究

论文作者　李帅
校内导师　时培成
校外导师　倪绍勇
专业学位类别　机械
研究方向（领域）　智能网联汽车

论文提交日期：2025 年 5 月 20 日

分类号：TP301.6　　学校代码：10363

密　级：公开　　学　　号：2220110168

基于深度强化学习的无人驾驶车辆行为决策研究

Research on Autonomous Vehicle Behavior Decision-Making Based on Deep Reinforcement Learning

学生姓名：　李帅

校内导师：　时培成

校外导师：　倪绍勇

专　　业：　机械

研究方向：　智能网联汽车

论文答辩日期：2024 年 5 月 10 日

二、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： 

日期：　　2025 年 5 月 9 日

三、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：  　　指导教师签名： 

日期：2025 年 5 月 9 日　　日期： 2025 年 5 月 9 日

基于深度强化学习的无人驾驶车辆行为决策研究

摘要

随着无人驾驶技术的快速发展，如何在复杂动态环境中实现高效、稳定的行为决策已成为关键挑战之一。传统深度强化学习（Deep Reinforcement Learning, DRL）算法在无人驾驶场景下仍然面临诸多问题，例如 Q 值估计存在高估误差、样本利用率较低，以及难以充分捕捉时序依赖信息等，这些问题严重影响了无人驾驶系统的决策质量和训练效率。为了解决这些挑战，本文提出了两种创新性强化学习算法：基于双优先级经验回放的决斗双深度 Q 网络（DPD3QN）和融合门控循环单元（GRU）的多时间序列决策模型（D3QN-GRU），旨在优化 Q 值估计、提高学习效率，并增强对复杂交通环境的适应能力。

（1）针对强化学习算法中 Q 值估计精度不足、环境不确定性导致的误差累积以及关键经验样本利用效率低下等问题，提出了基于双深度决斗 Q 网络（DPD3QN）的改进算法。DPD3QN 结合了决斗 Q 网络（Dueling Q Network）与双深度 Q 网络（Double Deep Q Network, DDQN）结构，分别对状态值（state value）和动作优势值（advantage value）进行独立建模，从而提升 Q 值估计的精度，减少因环境不确定性带来的误差。此外，该算法引入双优先级经验回放机制

（Dual-Priority Experience Replay, DP-ER），综合 TD 误差和分段抽样策略进行样本选择，使关键经验样本的利用效率最大化，从而加速收敛过程并优化训练效果。

（2）针对动态交通场景中多维时序特征建模困难、长期依赖关系捕捉不足以及关键决策特征权重失衡等问题，提出了融合门控循环单元与注意力机制的 D3QN-GRU 算法。D3QN-GRU 在传统 D3QN 结构的基础上，进一步融入门控循环单元（Gated Recurrent Unit, GRU）和注意力机制（Attention Mechanism），提升了对多维时序数据的建模能力。GRU 结构能够有效捕捉长期时序依赖关系，使得模型在动态交通环境下具备更强的预测能力，而注意力机制则用于强化关键特征提取，提高决策的稳定性和适应性，从而增强智能体在复杂驾驶场景中的长期规划能力。

关键词：无人驾驶，深度强化学习，决斗双深度 Q 网络，双优先级经验回放，多时间序列决策

RESEARCH ON AUTONOMOUS VEHICLE BEHAVIOR DECISION-MAKING BASED ON DEEP REINFORCEMENT LEARNING

ABSTRACT

As autonomous driving technology continues to progress rapidly, ensuring efficient and stable decision-making in highly complex and dynamic environments has emerged as a significant challenge. Traditional Deep Reinforcement Learning (DRL) algorithms still face several issues in autonomous driving scenarios, such as overestimation of Q-values, low sample utilization, and insufficient handling of temporal dependencies. These limitations significantly affect decision-making quality and training efficiency in autonomous driving systems. To address these challenges, this paper proposes two innovative reinforcement learning algorithms: the Dueling Double Deep Q-Network with Dual-Priority Experience Replay (DPD3QN) and the Multi-Time Series Decision Model incorporating Gated Recurrent Units (D3QN-GRU). These models aim to optimize Q-value estimation, improve learning efficiency, and enhance adaptability in complex traffic environments.

(1) Addressing the issues of insufficient Q-value estimation accuracy, error accumulation caused by environmental uncertainty, and low

utilization efficiency of critical experience samples in reinforcement learning algorithms, an improved algorithm based on the Dueling Prioritized Double Deep Q-Network (DPD3QN) was proposed. DPD3QN integrates Dueling Q-Networks with Double Deep Q-Networks (DDQN) to separately model state values and action advantage values, improving the accuracy of Q-value estimation and reducing errors caused by environmental uncertainty. Additionally, the algorithm introduces a Dual-Priority Experience Replay (DP-ER) mechanism, which combines TD-error-based prioritization with segmented sampling strategy to enhance the utilization of critical experience samples. This accelerates the learning process and optimizes training efficiency.

(2) To resolve the challenges of modeling multi-dimensional temporal features, capturing long-term dependencies, and balancing key decision feature weights in dynamic traffic scenarios, the D3QN-GRU algorithm integrating Gated Recurrent Units (GRU) and attention mechanisms was developed. The D3QN-GRU model enhances the conventional D3QN framework by integrating Gated Recurrent Units (GRU) along with an Attention Mechanism. enhancing its ability to model multi-dimensional temporal data. The GRU structure effectively captures long-term temporal dependencies, improving the model's predictive capabilities in dynamic traffic environments. Meanwhile, the attention mechanism strengthens key feature extraction, enhancing the stability and adaptability of decision-making, thereby improving the agent’s long-term planning ability in complex driving scenarios.

KEY WORDS： autonomous driving, deep reinforcement learning, dueling double deep Q-network, double prioritized experience replay, multi-time-series decision making

目录

第 1 章 绪论

1.1 研究工作的背景及意义

中国工业和信息化部（工信部）与交通运输部（交通部）在 2024 年发布了多项关于智能网联汽车的通知[1]，主要包括以下内容：

2024 年 1 月 15 日，我国多个中央部委联合印发智能交通系统创新实施方案。根据该文件要求，住房和城乡建设部会同交通运输部、公安部、自然资源部与工业和信息化部共同推进新型智能交通体系建设，重点围绕车联网协同发展机制开展技术验证。此次示范工程将构建“端-路-网”三维融合体系，通过车载终端、道路感知设备与云端控制平台的系统集成，在特定区域开展智慧出行系统实证研究。

2024 年 6 月，经多部委协同推进，我国智能驾驶领域迎来制度创新。住房和城乡建设部联合工业和信息化部、交通运输部及公安部最新出台《关于推进智能网联汽车道路测试与示范应用的通知》，该政策聚焦于建立新型车辆管理机制，旨在指导相关企业及用户强化技术研发与安全管理能力。

2024 年 8 月 1 日，工信部发布了《关于进一步加强智能网联汽车准入、召回及软件在线升级管理的通知（征求意见稿）》。该通知旨在进一步加强搭载组合驾驶辅助系统的智能网联汽车准入、召回和车辆软件在线升级管理。

工信部和交通部确实发布了多项关于智能网联汽车的通知，涵盖了试点工作、城市名单公布、准入管理等多个方面，旨在推动智能网联汽车技术的发展和产业化应用。

无人驾驶技术是指通过自动化技术使得车辆在没有人工干预的情况下自主行驶。随着人工智能和自动控制技术的迅猛发展，无人驾驶已经不再是科幻小说中的幻想，而是逐步成为现实。无人驾驶技术涉及的领域非常广泛，包括感知、决策、控制等多个层面，其中决策系统作为整个自动驾驶系统的核心，直接决定了车辆如何在复杂的道路环境中做出实时响应。决策系统的研发不仅是自动驾驶技术的基础，同时也是技术突破的关键[2]。

从 20 世纪 80 年代起，随着计算机科学和机器人学的快速发展，自动驾驶技术开始逐步崭露头角。早期的研究主要集中在环境感知和路径规划领域，研究人员通过研究导航算法和简单的控制系统，探索如何实现车辆的自主行驶。到了 2004 年，随着 DARPA 挑战赛的启动，自动驾驶的研究进入了一个新的阶段。该比赛不仅提供了一个展示自动驾驶技术的平台，也推动了无人驾驶技术的飞速发展。在此之后，Google[3]推出的自动驾驶项目为技术的普及和应用奠定了基础，并催生了大量相关企业和研究机构的参与。自动驾驶的分级标准由国际机械工程师协会（SAE International）[4]于 2014 年首次发布，并在 2016 年进行了进一步的修订。该标准定义了从 L0 到 L5 的六个自动驾驶级别，并且这些级别已被广泛接受和使用。如图 1-1 所示。L0-L2（ADAS）[5]仅需处理行为数量有限且环境工况确定的驾驶环境，车辆主要由人类驾驶员负责驾驶；L3-L5 处理的行为数量和环境工况范围很大，驾驶行为主要由自动驾驶系统负责。

图 1-1 国际汽车工程师学会（SAE）自动驾驶的分级标准[4]

然而，在无人驾驶技术的整个研发过程中，最具挑战性的部分往往是决策系统。无人驾驶决策系统作为车辆智能化的核心模块，承担着将感知数据转化为驾驶行为的关键任务。这一系统需要实时处理大量来自多源异构传感器的数据，包括雷达、摄像头、激光雷达（LiDAR）、超声波传感器以及高精度地图等。这些传感器数据在时间、空间和精度上存在差异，决策系统必须通过高效的数据融合算法将其整合为同步的环境表示。

在复杂的动态交通环境中，决策系统不仅要准确识别道路上的车辆、行人、交通标志和信号灯，还需要结合当前的道路状况，如路面湿滑、施工区域，交通规则，如限速、车道划分以及周围交通参与者的潜在驾驶意图，如变道、超车、停车等进行综合判断。此外，决策系统还需考虑车辆自身的状态，如速度、加速度、转向角度以及乘客的舒适度和安全性[6]，从而决定合理的驾驶策略。

如图 1-2 所示，无人驾驶决策系统通常采用分层架构，包括环境感知层、行为决策层和控制规划层。

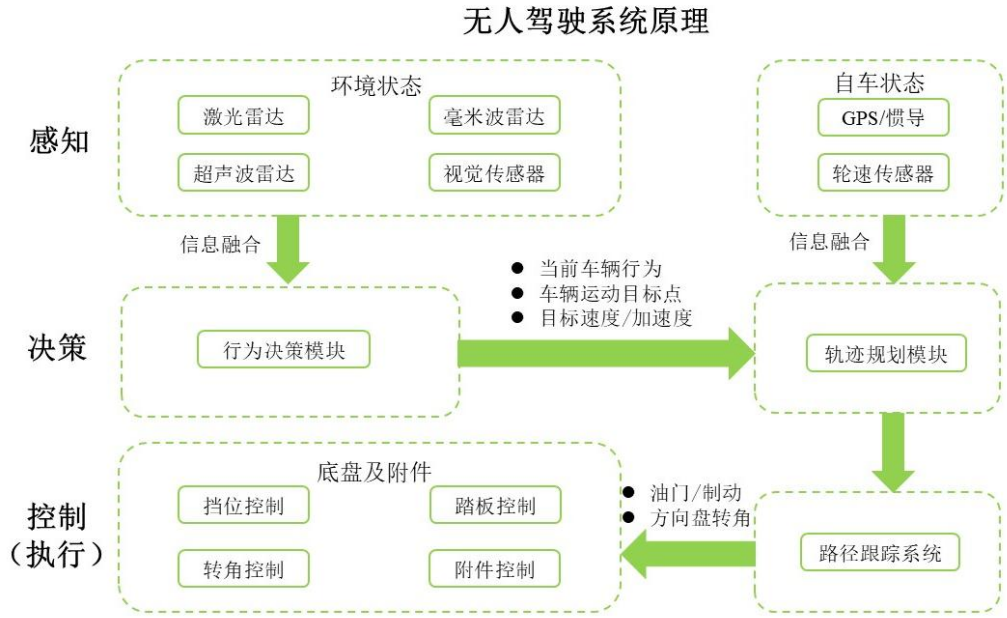

图 1-2 无人驾驶原理图

无人驾驶决策系统的核心任务之一是确保驾驶安全[7]。随着决策系统的逐步完善，无人驾驶将在大大减少交通事故、保障道路安全方面发挥至关重要的作用。同时，传统的交通系统中，驾驶员的决策往往受到个人经验、反应速度以及交通规则的限制，导致许多交通资源的浪费。自动驾驶车辆之间的协作也能通过车与车之间的通信系统实现更高效的交通流动，进而改善城市交通状况。无人驾驶技术不仅仅是单一车辆的技术革新，它代表了智能交通系统的一个重要方向。自动驾驶技术的发展不仅能带来更高效的交通，还能够推动与之相关的智能基础设施建设，如智能红绿灯、智慧停车场等。

1.2 国内外研究现状

无人驾驶汽车的决策研究是自动驾驶系统中的一个关键领域，涉及如何基于

环境感知信息、交通规则、驾驶目标等多种因素，做出实时、合理的驾驶决策。随着深度学习、强化学习、优化算法等技术的快速发展，自动驾驶决策系统取得了显著进展。

1.1.2 无人驾驶行为决策研究现状

无人驾驶决策系统的目标是根据环境感知信息和目标任务，制定合适的驾驶策略。当前的研究方法主要包括以下几种：

（1）基于规则的决策方法：此类方法在简单的驾驶任务中较为有效，但在复杂、动态的交通环境中面临挑战，难以应对未知情况。在上世纪 80 年代至 2004 年，无人驾驶技术的基础开始形成。早期的自动驾驶决策系统多依赖预设的规则和模型，例如基于模型的控制（MPC）[8]方法。这一时期的研究重点集中在导航与路径规划算法上，美国国防高级研究计划局（DARPA）的挑战赛 DARPA Urban Challenge[9]成为了这一阶段的重要推动力。从 2004 年到 2012 年，分别为无人驾驶技术的不同发展阶段提供了展示的平台。卡内基梅隆大学团队为参加 2007 年 DARPA 举办的挑战赛而开发了自动驾驶系统 Boss[10]，并于 2008 年发表了此智驾系统的技术总结。在此阶段，学术界和工业界逐步积累了关于自动导航和环境感知的基础知识。2004 年到 2014 年，自动驾驶技术迎来了重要的技术突破。Google 在 2010 年发布的自动驾驶车和引入的关键决策系统开始引起行业的广泛关注。Marti[11]等人提出了基于 LiDAR 的动态物体检测与跟踪方法，适用于自动驾驶车辆。这一时期，基于激光雷达（LiDAR）、相机和雷达等的环境感知能力显著提升，而路径规划和决策算法也逐步实现了更加可靠的自动化驾驶，为后来深度学习和机器学习算法在各行各业发展也打下基础。

（2）基于学习的决策方法：随着人工智能和深度学习的发展，越来越多的自动驾驶系统开始采用基于数据驱动的决策方法。通过训练模型，系统能够自主学习如何根据不同的环境信息做出决策。基于学习的决策算法可细分为模仿学习和深度强化学习这两种研究方法。模仿学习应用大约始于 Google 自动驾驶项目（Waymo）[12]，在这一时期开始大量使用模仿学习（Imitation Learning）来训练其自动驾驶系统。在 2015 年，Google 自动驾驶项目（Waymo）开始收集大量的驾驶数据，通过让人工驾驶员进行驾驶，系统能够学习如何处理各种复杂的道路情境（如交通信号、道路标线、障碍物等）。Mayank 提出了 ChauffeurNet[13]，一

种基于模仿学习的驾驶策略模型。该模型通过模仿人类驾驶行为并结合合成数据来提升鲁棒性。自 2014 年起，深度学习技术特别是卷积神经网络（CNN）[14]成为了自动驾驶决策系统的重要推动力。通过引入复杂的神经网络结构，车辆能够更准确地感知复杂的交通环境，并做出更加精确的驾驶决策。英伟达，Mobileye 开始在汽车应用卷积网络。从 2018 年开始，Transformer[15]等先进的人工智能技术逐渐被应用于自动驾驶决策系统中，2019 年特斯拉提出 BEV[16]，因算力消耗问题导致并未落地。同时期 Xu[17]等人提出了新颖的 LSTM-FCN 架构，此架构可以从大规模不同时间段的车辆动作数据中提取重要动作，并利用当前图像信息和过去动作信息，预测未来的动作行为。Zhao[18]等人提出了一种基于改进双深度 Q 网络（DDQN）算法的驾驶决策模型，该模型采用十个离散化动作来输出三车道车辆的驾驶操作，将连续的动作空间映射为有限的动作集合。这一阶段，自动驾驶系统的决策能力得到了大幅提升，尤其是在多目标决策、环境理解和实时反应方面。

展望未来，自动驾驶技术的决策系统将继续走向更高效、更智能的方向。随着车载计算平台（OCC）[19]和端到端的深度学习模型[20]的逐步实现，自动驾驶车辆将能够更好地适应复杂多变的交通环境，并具备更高的安全性与舒适性。最后，无人驾驶决策技术的进步，得益于各项核心技术的创新与突破。未来，随着硬件计算力的提升和 AI 算法的持续优化，自动驾驶决策系统将变得更加精准和可靠，推动智能交通系统的全面普及。

1.1.2 强化学习研究现状

强化学习（Reinforcement Learning, RL）作为机器学习领域的重要组成部分[21]，在过去几年中取得了显著的研究进展。它模拟了人类或动物在环境中通过试错方式学习最优行为策略的过程。强化学习的核心问题是如何在与环境交互的过程中，通过奖励信号引导智能体选择一系列行动，使得某个目标或收益最大化。

在强化学习的早期，研究者主要集中在值迭代（Value Iteration）和策略迭代（Policy Iteration）方法的研究上。值迭代通过计算状态价值函数来评估每个状态的好坏，而策略迭代则通过不断改进策略来逼近最优策略。这些方法在较简单的离散环境中表现良好，但在大规模、高维度的实际应用中，计算量和存储空间的需求使其面临挑战。

随着深度学习技术的兴起，深度强化学习成为了强化学习的重要发展方向。深度 Q 网络（DQN）作为深度强化学习的开创性成果，在 2013 年通过在 Atari[22]游戏中的应用，成功将深度学习与强化学习结合，解决了传统强化学习算法在高维感知问题中的困难。DQN 通过使用卷积神经网络（CNN）从图像中提取特征，极大地提高了强化学习算法在复杂环境中的性能。随后，许多深度强化学习算法相继问世，如 Double DQN[23]、Prioritized Experience Replay[24]、Dueling DQN[25]等，这些算法都在提高训练稳定性和效率方面做出了创新和贡献。

策略梯度方法直接对策略进行优化，而不是通过值函数来评估策略的优劣。常见的策略梯度算法包括 REINFORCE、Actor-Critic 等。REINFORCE[26]这是最基本的策略梯度方法，依赖蒙特卡洛采样，但可能方差较大；然后是 Actor-Critic[27]，结合了值函数和策略函数，使用 Critic 来减少方差，提高稳定性。这些方法通过梯度下降更新策略，从而实现策略的逐步优化。近端策略优化（Proximal Policy Optimization, PPO）[28]是近年来广泛应用的强化学习算法，通过限制每次更新的幅度来避免训练过程中的不稳定性。PPO 相比于传统的策略梯度方法，在保持较高样本效率的同时，训练过程更加稳定，广泛应用于机器人控制、自动驾驶等任务中。

强化学习从理论发展到研究应用的过程如图 1-3 所示。

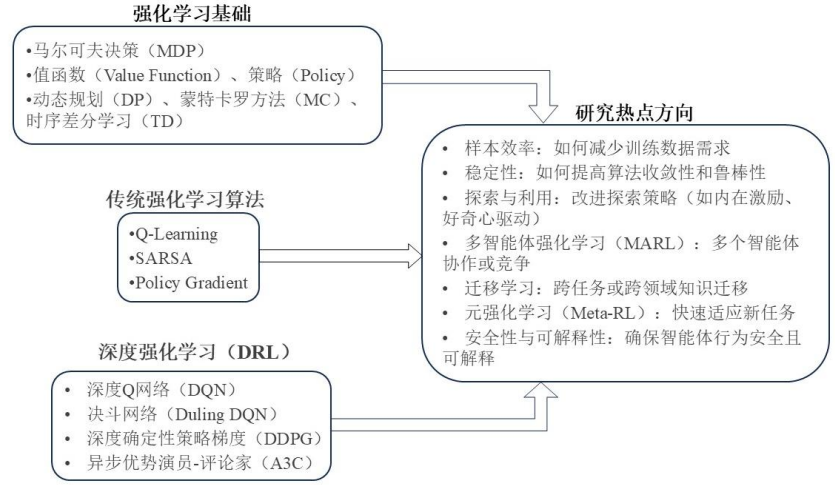

图 1-3 强化学习发展过程

当前，强化学习的研究热点主要集中在以下几个方面：首先，提升样本效率以减少训练数据需求；其次，增强算法的稳定性和鲁棒性，确保其在实际应用中的可靠性；此外，探索与利用的平衡问题也备受关注，研究者通过引入内在激励

和好奇心驱动等机制改进探索策略。多智能体强化学习（MARL）[29]则致力于解决多个智能体在协作或竞争环境中的决策问题。与此同时，迁移学习和元强化学习（Meta-RL）[30]为智能体快速适应新任务提供了新的思路。安全性与可解释性也成为研究重点，旨在确保智能体行为的安全性和透明性。

强化学习的应用领域广泛，涵盖游戏（如 AlphaGo[31]、星际争霸）、机器人控制、自动驾驶、金融交易、医疗决策和资源管理等[32]。然而，尽管取得了诸多成果，强化学习仍面临诸多挑战：理论瓶颈尚未完全突破，算法的收敛性和泛化能力仍需进一步提升；训练过程对计算资源的需求较高，限制了其在实际场景中的部署；此外，智能体行为的伦理与安全问题也亟待解决。未来，强化学习的研究将继续朝着提高算法效率、增强泛化能力、降低计算成本以及解决伦理问题的方向发展，为人工智能的广泛应用提供更强大的支持。

1.3 论文研究内容

本文的研究聚焦于基于深度强化学习（Deep Reinforcement Learning, DRL）[33]方法的无人驾驶车辆行为决策系统，提出了两种创新性的算法。算法的核心思想在于通过结合决斗网络和双深度 Q 网络的优点，并重构原有网络的输出层。分别结合上优秀的优先级经验回放和改进的多时序网络，从而提升车辆决策系统的智能化程度和安全性。其主要内容包括：

（1）本文通过引入决斗网络的架构，在主网络和目标网络中明确区分状态值和动作优势值。这一改进提高了动作价值估计的准确性，避免了传统网络中状态动作估计不准确所引发的局部最优问题。决斗网络的结构使得系统能够更有效地评估每个状态下选择不同动作的相对优势，进而做出更精确的决策。为了进一步提升算法的学习效率和收敛速度，本文引入了双优先级经验回放机制。通过优先级抽样的方式，模型能够更快速地识别并有效利用关键经验，避免了传统经验回放机制中对低优先级样本的过度依赖。该机制结合了基于 TD 误差的优先级和分段抽样的优先级，不仅能够提升样本的利用率，还能够加速训练过程。通过这种方式，DPD3QN 在处理稀疏奖励和较为复杂的环境时，能够表现出更好的学习性能。

（2）在引入决斗网络的基础上，为了提高模型在处理多维时间序列数据时的能力，本文引入了改进的 GRU 网络。GRU 相较于传统的长短期记忆网络

（LSTM），其结构更为精简，计算效率更为高效，更适合处理动态环境中的复杂时间依赖关系。GRU 网络的引入使得模型在面对车辆动态行为（如速度、加速度和相对位置等）时，能够更精准地捕捉和分析这些多维数据，从而做出更为精确的决策。使模型能够在动态交通仿真环境中更好地处理时间序列数据，进一步增强决策的鲁棒性和有效性。

（3）将本文提出的改进算法部署在 OpenAI Gym 模拟平台上进行全方位的训练和测试。结果表明，与传统的其他多种深度强化学习算法进行比较，本文提出改进算法在多种复杂的场景下，均具有很好的表现；同时，在复杂的高速公路上的测试环境中，本文提出的改进算法相较于其他算法的成功率更高，尤其在复杂程度更高的测试场景中，有更好的决策精度和更快的收敛速度。

1.4 论文研究结构

本文的组织结构共分为五个主要章节，逐步展开对基于深度强化学习的无人驾驶行为决策系统的研究。

绪论

本章首先介绍了自动驾驶技术的发展背景及其在智能交通系统中的重要性，强调了无人驾驶技术对于未来交通的潜在影响。随后，详细回顾了国内外在自动驾驶决策算法、强化学习等领域的研究现状，特别是在行为决策和自动驾驶安全性方面的挑战。最后，明确提出本文的研究目标和主要贡献，介绍了所提出的改进算法。

第 2 章 相关工作与理论基础

本章详细介绍了深度学习和强化学习的基本理论。首先，阐述了人工神经网络和深度神经网络的基本概念，并介绍了深度强化学习（DRL）中的重要算法，如深度 Q 网络（DQN）及其衍生算法。其次，探讨了强化学习中的核心概念，尤其是马尔可夫决策过程（MDP），优先经验回放和多时间序列网络。最后，分析了深度强化学习在自动驾驶中的应用，重点讨论了如何利用深度学习提升决策精度。

第 3 章 基于双优先级经验回放的决斗双深度 Q 行为决策网络

本章提出了 DPD3QN 算法，并详细介绍了其核心架构和创新点。首先，结合了决斗网络与双深度 Q 网络，优化了 Q 值的估计方法，提高了决策系统的精

度和稳定性。其次，介绍了双优先级经验回放机制，进一步提高了模型训练的效率。最后，通过实验验证了 DPD3QN 在高速公路环境下的效果，展示了其相较于传统 DQN 和 DDQN 算法的优势。

第 4 章 基于决斗双深度 Q 网络的多时间序列车辆行为决策

本章提出了 D3QN-GRU 组合模型，介绍了引入注意力机制之后的门控递归单元，并将其与 D3QN 网络相结合，并对其进行了详细的实验分析。通过对多种复杂场景的实验验证，证明了该模型在实际应用中的高效性和可靠性。此章不仅补充了基于时间序列的数据处理方法，还为进一步改进自动驾驶决策系统提供了理论支持。

第 5 章 总结与展望

论文结构如图 1-4 所示。

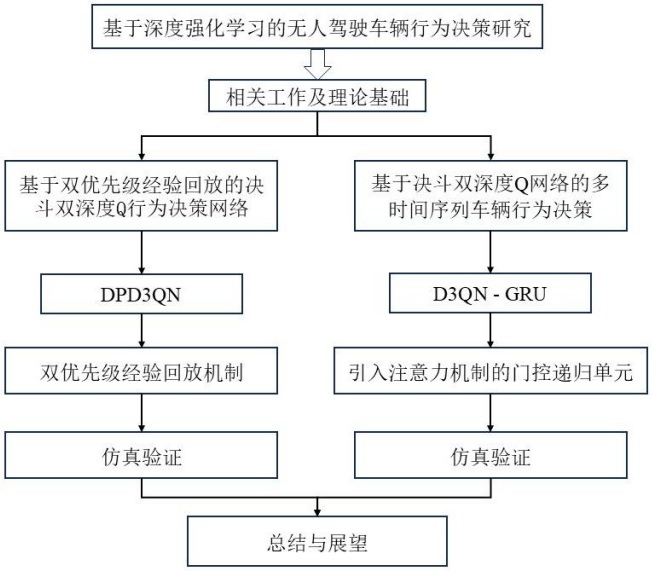

图 1-4 论文结构图

本章总结了全文的主要研究成果，回顾了改进算法的设计思路及其在自动驾驶中的应用效果。同时，讨论了当前研究中的局限性，并展望了未来的研究方向，特别是在算法优化、实时性提高以及与其他智能交通系统的协同方面的潜力。

第 2 章 基于无人驾驶行为决策的理论基础及相关工作

通过融合深度学习的强大特征提取能力与强化学习在复杂决策任务中的高效性，深度强化学习为解决复杂状态空间下的学习问题开辟了一条全新的技术路径。这种融合不仅充分发挥了深度学习在识别和提取复杂特征方面的技术优势，还借助神经网络的强大函数逼近能力，为传统方法难以建模或解决的复杂问题提供了创新的解决方案。本章将系统阐述深度强化学习中神经网络与强化学习相结合的核心理念，并重点对经典深度 Q 网络算法进行深入剖析。此外，本章还将详细介绍所构建的仿真实验平台，为理论研究与算法验证提供坚实的实验基础，从而为后续研究工作奠定重要支撑。

2.1 深度学习理论基础

深度学习（Deep Learning）[34]技术的核心思想是通过建立由多个层次构成的神经网络，这些层次能够执行从简单到复杂的信息转换，类似于人脑的工作机制[35]。人工神经网络（Artificial Neural Network，ANN）[36] 是一种受生物神经网络启发而构建的理论模型，旨在模拟和处理信息。作为深度学习的支柱，该模型在机器学习领域得到了广泛的应用，涵盖图像识别、路径规划、目标检测等多个任务。2006 年，Hinton[37]与他的同事们在著名的《Science》期刊上首次发表了神经网络论文，标志着一个通过学习高度复杂的神经网络像人脑一样进行信息处理的新时代开始了。从宏观角度分析，人工神经网络构成一个由大量神经元节点通过广泛而精密的连接网络组成的自适应非线性动态系统。神经网络模型如图 2-1 所示。

单个神经元通过模拟神经细胞的建模，展示了其内部结构和信息处理的方式。这种建模方法为神经网络的深入理解和应用提供了重要的基础。

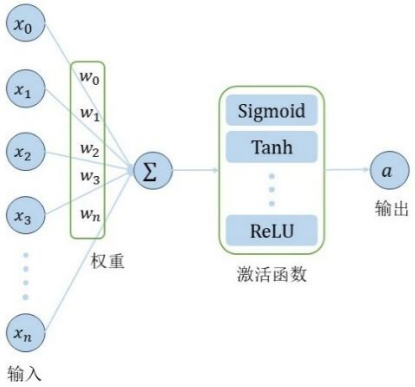

图 2-1 神经元示意图

在图 2-1 所示神经网络架构中，输入向量组（x_1 ,x_2…x_n）的设计仿生模拟了生物神经元突触连接机制，其对应的突触权重参数（ω_1,ω_2…ω_n）表征着前馈信号与处理单元之间的联结强度。在模型迭代优化过程中，这些权重参数通过梯度反传算法持续动态调整，直至系统收敛至误差最小化状态。每个计算单元配置有阈值偏置量b和信号门控函数f。前者用于调节神经元激活基线，后者实现非线性特征映射功能。最后通过输入x和其对应的权重ω进行加权之和，经过偏置项形成预激活值，后经激活函数激活，得到输出a。目前比较常见的激活函数有 Sigmoid 函数[38]、双曲正切函数 Tanh 函数[39]和 ReLU 函数[40]。

如图 2-2 四层神经网络所示，分别由输入层隐藏层和输出层组成。

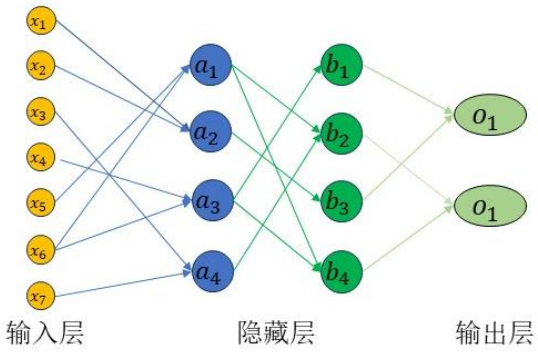

图 2-2 四层神经网络结构图

在整个神经网络结构中，为了适应更为复杂的信息处理任务，单一神经元模型的工作方式经过了扩展和优化，形成了多层神经网络。这种多层结构能够更有效地处理抽象层次的特征提取和复杂模式识别，通过层层堆叠的方式实现对输入数据的逐级抽象表示[41]。如图 2-2 四层神经网络所示，分别由输入层、隐藏层和

输出层组成。这种深度结构的引入使得神经网络能够更灵活地学习并捕捉数据中的抽象关系，提高了网络的表达能力和性能。因此，多层神经网络模型成为深度学习领域的核心组成部分，为各种复杂任务的解决提供了强大的工具。

2.2 强化学习理论基础

强化学习（Reinforcement Learning, RL）是机器学习的三大种类之一，与监督学习和无监督学习共同构成了这一领域的重要组成部分。强化学习问题可被定义为离散时间的随机控制过程，如图 2-3 所示。在此过程中，智能体与环境的互动展开过程如下：智能体从初始状态$s \in S$开始感知状态的表示。对于每个时间步$t \in \mathbb{N}$，智能体选择在$a \in A$处执行一个动作，随后获得该动作的相关奖励$r \in R$，并观察在环境中的新状态s。这个过程持续进行直到智能体找到最优奖励策略为止，这一过程最初由 Bellman 提出[42]。

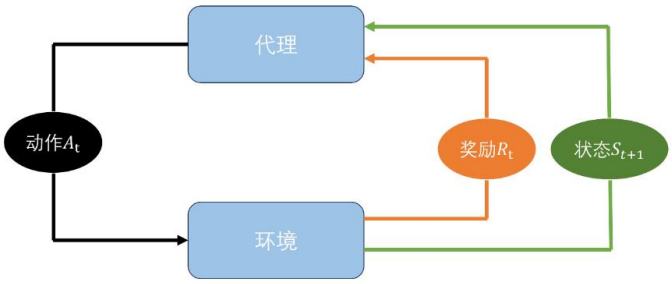

图 2-3 强化学习循环结构

2.2.1 马尔可夫决策过程

马尔可夫性指的是系统下一个时刻状态s_{t+1}仅取决于当前状态s_t和动作a_t，而不受过去的状态和动作影响。在强化学习的语境下，智能体与环境的不断 互动过程可被视为具有马尔可夫性质的决策过程（MDP）[43]。在随机过程理论框架下，MDP 可建模为具有记忆无特性的序贯决策系统，其形式化架构由五维参数组(S, A, P, R, γ)构成。具体而言：状态空间S构成系统观测的完备离散基集，决策行为空间A表征智能体的可选操作域，状态转移概率矩阵P: $S \times A \times S \rightarrow [0,1]$量化环境动态演化规律，即时收益函数$R$: $S \times A \rightarrow R$映射行为价值评估，而时序贴现系数$\gamma \in [0,1)$则用于平衡当前收益与远期回报的效用权重。

马尔可夫动态系统状态转移机制以有向图形式可视化呈现，如图 2-4。其演进机制可形式化描述为：决策主体基于当前环境观测状态$s_t \in S$，依据既定策略从行为空间A中选取执行动作$a_t \in A$，触发环境状态从转移概率$(s_{t+1}|s_t, a_t)$分布跃迁

至下一状态s_{t+1}。

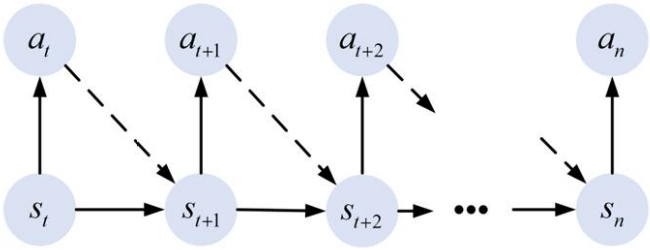

图 2-4 马尔科夫决策过程

该系统其中节点表征系统状态，决策主体通过持续交互生成状态动作轨迹(s_t, a_t, s_{t+1}, a_{t+1},... s_n, a_n)直至抵达终止条件约束的吸收态，完成完整的序贯决策周期。

若 MDP 是完全可观察的，那么环境中的状态 s 将与智能体获得的观察完全一致。以现实世界中移动的智能体为例，它仅能通过传感器获取信息，仅拥有代表其状态的信息子集。因此，在这种情况下，可以将 MDP 描述为部分可观察的。在本文的研究范围内考虑完全可观察的情形，此设定有助于问题的简化，使得能够更深入地研究 MDP 的特性以及相应的决策制定问题。

2.2.2 Q-learning

Q 学习算法(Q-learning) [44]如表 2-1 所示，

表 2-1

Q-learning
初始化 Q-table，$\alpha \in$（0，1]，$\varepsilon > 0$
For *Episode* = 1 to *M* do:
获取当前环境状态 *s*
For *t* = 1 to *T* do:
根据 ε-*greedy* 策略，选择动作 a
执行动作 a，获得奖励 r，新的状态 s'
令 $Q(s,a) \leftarrow Q(\mathrm{s},a) + \alpha\left[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right]$
令 $S \rightarrow S'$
End for
End for

在基于值迭代的强化学习范式下，Q-learning 通过贝尔曼最优方程构建状态动作价值矩阵，该矩阵通过异步动态规划策略实现对环境状态空间S与决策空间A的全映射。

该算法的优越性体现在其免模型特性与离线策略（Off-Policy）[45]更新机制的耦合，通过时序差分误差的持续修正，最终使得价值估计函数收敛至最优策略对应的不动点。这种基于价值评估的策略优化方法，有效克服了传统动态规划中的维度灾难问题。相比于传统的在线策略（on-policy）[46]算法只存在一种策略做动作的选取，也用该策略去做优化的方法，鲁棒性更强。

强化学习的目标是在策略优化的过程中使R_t值达到最大化，并基于累计折扣回报定义状态价值函数 V 和动作价值函数 Q，用于估算在特定状态或执行特定动作时的长期回报[47]。分别定义为式(2-1)和式(2-2)，其中π是控制动作的策略，为了便于更新，状态价值函数通常表示为式(2-3)的递归形式，并且最优动作的控制策略π是由状态价值函数决定，其定义为式(2-4)：

$$V^{\pi}(s) = E_{\pi}[R \mid s, \pi] \tag{2-1}$$

$$Q^{\pi}(s,a) = E_{\pi}[R \mid s, a, \pi] \tag{2-2}$$

$$Q^{\pi}(s,a) = E_{\pi}[r + \gamma \max_{a'} Q^{\pi}(s', a')] \tag{2-3}$$

$$\pi(s) = \arg\max_{a} Q(s,a) \tag{2-4}$$

在时序差分学习框架下，Q-learning 是一种基于值迭代更新的算法，即逼近最优动作价值函数Q^*。该算法依据公式(2-5)对当前动作价值函数的估计值进行迭代调整和优化。

$$Q^*(s,a) \leftarrow Q(s,a) + \alpha\left(r + \gamma \max_{a'} Q(s',a') - Q(s,a)\right) \tag{2-5}$$

式中$Q(s,a)$是 agent 在状态s下选择动作a的期望奖励。r是在状态s下选择动作a的即时奖励。α为学习率，$\max_{a'} Q(s',a')$表示在状态s'中选择其他动作a'的期望奖励。通过迭代式(2-5)，$Q(s,a)$将最终收敛到$Q^*(s,a)$，即最优动作值函数[48]。这种通过迭代获得最优动作值函数的方法称为 Q-learning。

2.3 深度强化学习理论基础

深度强化学习（Deep Reinforcement Learning, DRL）[33]是结合深度学习（Deep Learning）[34]与强化学习（Reinforcement Learning, RL）[21]的前沿技术，是在通过智能体（Agent）与环境的交互学习，能够最大化最终累积奖励的最优策略。在众多强化学习算法中，Q 学习算法（Q-learning）由于其简单有效，被广泛应用于解决各种顺序决策问题。然而当环境状态空间较大时，Q-learning 会存在内存消耗大、查存时间长、维度爆炸等系列问题。针对这些问题，常将神经网络进行函数逼近来近似求取动作价值函数。为拟合合适的损失函数，将这些神经网络参数通过梯度下降法进行训练。这种方法可以大大降低需要存储大量状态的需求，从而提高算法效率、改善可伸缩性。因此，在强化学习中引入卷积和递归神经网络等深度学习结构已成为一种趋势[49]。

2.3.1 经典深度 Q 网络

深度 Q 网络（DQN）[22]针对 Q-learning 存在的问题进行了优化，具体是将神经网络和 Q-learning 相结合，使用参数化的模型来代替 Q 值表。Q 学习算法记录每个动作在每个状态下的 Q 值，并反复更新，它适用于相对小规模的问题。但是状态空间可能太大甚至连续，无法保存在一个表中练习。在这种情况下，可以使用值函数进行近似。这个值函数可以是线性函数，也可以是非线性函数。$Q(s,a;\theta)$表示用参数θ近似估计的动作价值函数。

深度 Q 学习以$r_t+\gamma \arg\max_{a'} Q(s',a';\theta')$为目标 Q 值，为了计算 DQN 中的 Q 表和实际 Q 表之间的差异，引入损失函数$L(\theta)$，定义损失函数网络输出 Q 值与目标 Q 值函数之间的关系如式(2-6)所示。其中y_t表示为动作的真实价值，定义为式(2-7)：

$$L(\theta)=E\Big[\sum_{t=1}^{N}\big(y_t-Q(s,a;\theta)\big)^2\Big] \tag{2-6}$$

$$y_t=r_t+\gamma \arg\max_{a'} Q(s',a';\theta') \tag{2-7}$$

式中s'，a'表示状态s采取动作a后的下一个状态和动作。$Q(s',a';\theta')$表示 Q 网络估计的动作价值。式(2-6)，(2-7)的θ和θ'分别代表着 DQN 中的主网络和目标网

络的两个参数，前者用于估计当前行为动作，后者用于产生目标值。y_t具体由即时性奖励r_t和γ倍下一状态最佳动作价值组成。

这一过程是通过精心设计的结构和策略来实现值函数的有效逼近和训练过程的稳定性。通过这一算法，DQN 在强化学习领域取得了显著的成功，为解决实际问题提供了强大的工具和方法。

2.3.2 双深度 Q 网络

神经网络与强化学习技术的结合带来了高估的问题，过高的估计会导致估值函数与实际价值函数之间的误差。神经网络的函数评价模型的输出包含一个估计误差，因此函数评价模型的估价值并不能真实地反映实际值，两者之间总是存在误差。假设同一状态下，不同状态动作值之间的差异很小。在这种情况下，模型可能会选择当前状态的次最优动作，导致模型最终无法学习到最优策略。

所以假设在 t 时刻，选择状态 s 下动作 a 的估计值为 $Q^{estimation}(s,a)$，目标值为 $Q^{target}(s,a)$。可得式(2-8)：

$$Q^{estimation}(s,a)=Q^{target}(s,a)+Y_s^a \tag{2-8}$$

其中函数评价模型带来的评价误差用 Y_s^a 表示。设定 Y_s^a 均匀分布在[−e, e]上，且 e 为上界。

在深度强化学习的价值函数逼近框架下，双深度 Q 网络（DDQN）[23]提出了一种策略解耦机制以克服传统 DQN 的价值过估计偏差。其创新架构通过构建双重参数化网络实现操作决策与价值评估的功能分离：在线策略网络（参数 θ）负责生成候选动作空间内的最优策略，而后延迟更新目标网络（参数 θ'），基于贝尔曼最优方程计算状态动作价值估计值，如式(2-9)。然后根据评估值 $Q(s,a,\theta)$ 计算损失函数 $L(\theta)$，最后再反向误差传递更新主网络的参数 θ，如式(2-10)所示。

$$Y^{DDQN}=r_s^a+\gamma * Q\left(s',\arg max\, Q(s',a',\theta),\theta'\right) \tag{2-9}$$

$$L(\theta)=E\left[\left(Y^{DDQN}-Q(s,a,\theta)\right)^2\right] \tag{2-10}$$

式中 γ 为折扣因子，r_s^a 为奖励值，根据估计值与目标值的均方误差计算损失函数 $L(\theta)$。

2.3.2 决斗网络

决斗网络（Dueling Network）[25]是一种改进的深度强化学习架构，通过分离状态价值函数和动作优势函数来提高 Q 值估计的准确性和稳定性。它是对传统深度 Q 网络（DQN）的扩展，特别适用于动作选择对最终结果影响较小或动作空间较大的环境。

决斗网络通过将 Q 值函数分解为两个部分来解决上述问题，通过这种分解，Q 值函数可以表示为式(2-11)：

$$Q(s,a)=V(s)+A(s,a) \tag{2-11}$$

其中 $V(s)$ 为状态价值函数反映了状态本身的价值， $A(s,a)$ 为动作优势函数，反映了选择特定动作的相对优势。决斗网络的架构与 DQN 类似，首先通过卷积神经网络或全连接网络提取状态 s 的特征。

为了确保 $V(s)$ 和 $A(s,a)$ 的独立性，通常会对优势函数进行中心化处理，最后将 $V(s)$ 和 $A(s,a)$ 结合，生成最终的值 $Q(s,a)$，如下式（2-12）：

$$Q(s,a)=V(s)+\left(A(s,a)-\frac{1}{|A|}\sum_{a'}A(s,a')\right) \tag{2-12}$$

其中 $|A|$ 是动作空间的大小。这种中心化操作可以避免 Q 值函数对 $V(s)$ 和 $A(s,a)$ 的冗余依赖。

2.3.3 经验回放机制

在强化学习领域，神经网络的训练数据通常具有高度相关性，这意味着连续的样本可能在状态空间中相似，从而导致神经网络在训练过程中出现不稳定性。为了应对这一问题，DeepMind[24]团队提出了一种被称为经验回放的关键技术，这一技巧显著提升了强化学习的性能。在强化学习的样本效率优化框架下，经验回放技术通过构建样本存储器实现交互数据的时空解耦。

该存储系统采用循环队列数据结构持续收录智能体与环境交互产生的状态，形成具有时间衰减特性的经验分布池。如图 2-5 架构所示。

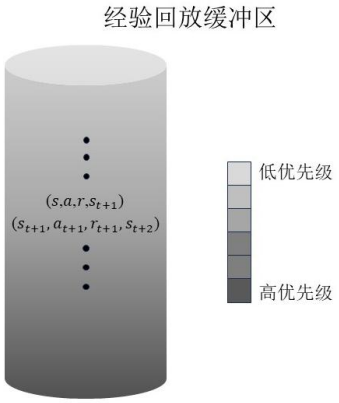

图 2-5 存储单元示意图

具体而言，智能体的行动轨迹被划分为四元组(s,a,r,s_{t+1})并存储在缓存中。缓存的容量需要事先确定，它仅保留最近的缓存数据；当缓存达到容量上限时，将删除最旧的数据。缓存大小是一个需要调整的超参数，对训练结果会产生影响。在神经网络训练过程中，智能体收集经验时，连续的四元组(s,a,r,s_{t+1})和$(s_{t+1},a_{t+1},r_{t+1},s_{t+2})$之间存在显著的相关性。若按照顺序使用这些密切关联的四元组进行神经网络的训练，通常会导致训练效果不佳。相反，经验回放通过从缓存中随机抽取一个四元组进行网络参数的更新。通过随机批采样机制从存储器中抽取非连续交互片段，有效打破数据时序相关性。这一机制不仅有助于提升训练效果，而且使得神经网络能够更好地适应各种复杂的环境和任务。此外，经验回放具有不同的优先级，将优先回放优先级高，即为重要性高的经验。这一特性极大地提高了样本的利用率，使得在达到相同训练效果的情况下，所需的样本数量大为减少。因此，经验回放不仅仅是为了解决序列相关性问题，更是通过更少的样本数量达到相同的效果。

2.3.3 分段抽样方法

优先级经验回放（Prioritized Experience Replay, PER）是一种对传统经验回放机制的有效改进。它通过为不同的训练样本分配不同的优先级，从而更加高效地引导学习过程，尤其适用于处理稀疏奖励或关键状态较少的强化学习任务。在此框架下，分段抽样方法[24]（Segment-Based Sampling）作为一种典型实现形式，被广泛用于提升样本选择的公平性与训练过程的稳定性。

该方法的核心思想是：智能体将整个经验池划分为若干个等长的优先级区间（segments），然后在每个区间内按照局部优先级概率分布随机采样一个样本。

这种方式不同于传统的全局采样机制，它有效缓解了高优先级样本因权重过大而被重复采样的现象，避免了样本利用的单一化。同时，该策略确保了低优先级样本仍有被选中的机会，从而提升了经验池的多样性与信息覆盖面。

从数学上看，设定经验池中有 N 个样本，划分为 K 个子区段，则每一段约含有 N/K 个样本。对于每个段内的样本，依据其相对优先级进行一次局部采样，最终从 K 个区段中获得一个大小为 K 的样本批次。

这种分段采样策略的优势体现在平衡探索与利用、提升样本效率、提高训练稳定性这几个方面：它在充分利用高优先级经验的同时，保证了低优先级样本的曝光机会，有助于提高策略的鲁棒性；通过局部采样降低了重复样本的风险，提高了经验数据的独立性，从而改善了神经网络的训练效果；适当分散采样权重可缓解训练中的梯度震荡，促进策略的稳定收敛。总的来说，分段抽样作为优先级经验回放的优化方法，在确保经验利用效率的同时，兼顾样本的广泛覆盖与策略泛化能力，是提升强化学习性能的重要机制之一。

2.4 门控循环单元理论基础

在时序建模的神经网络演进中，门控循环单元（Gated Recurrent Unit, GRU）是一种循环神经网络的变体，具备动态记忆调控机制。由 Kyunghyun Cho[50]团队于 2014 年率先构建。该架构针对传统循环神经网络（Recurrent Neural Network，RNN）[51]存在的梯度弥散与梯度爆炸现象，创新性地引入双门调控机制，更新门（Update Gate）与重置门（Reset Gate），通过参数化路径选择实现状态的自适应更新。相较于长短期记忆网络（Long Short-Term Memory, LSTM）[52]的三门复杂结构，GRU 采用结构精简变体，将遗忘门与输入门耦合为单一更新门，在保持长程依赖建模能力的同时显著降低计算复杂度。

2.4.1 LSTM 的门控机制

LSTM 包含 3 个门和一个单独的记忆单元，分别是遗忘门、输入门和输出门。图 2-6 表示了 LSTM 网络的基础网络结构。LSTM 网络结构很复杂，使用了更多的参数和结构，分别维护一个记忆单元和三个门控。

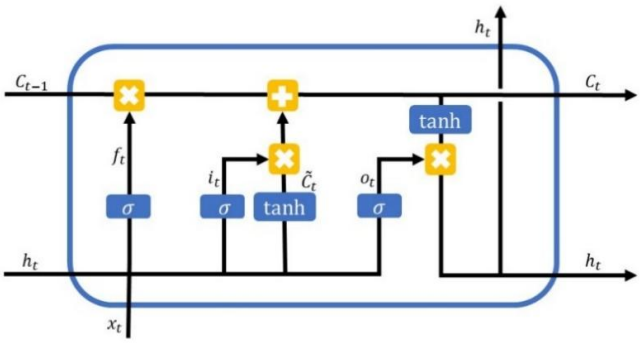

图 2-6 LSTM 网络结构图

如图 2-6，模块初始处 x_t为当前时间步的输入数据，h_t为当前时间步输出的隐藏状态，c_t为当前时间步的细胞状态，c_{t-1}为上一时间步的细胞状态，h_{t-1}为上一时间步输出的隐藏状态。首先，遗忘门f_t决定丢弃多少过去的信息，它会根据当前输入和前一时刻的隐藏状态来调整细胞状态；之后，输入门i_t会评估当前输入的重要性，并更新细胞状态；最后，输出门o_t决定当前时间步的隐藏状态。它会结合新的细胞状态，通过一个非线性变换来计算最终的输出结果。

整个计算流程的核心在于对信息的筛选和更新[53]，使 LSTM 具备长期记忆能力。因此，LSTM 在时间序列预测等任务中有广泛应用。

2.4.2 GRU 的门控机制

GRU 相比于 LSTM，减少了输入门和遗忘门，将它们合并成了一个更新门。这种简化使得 GRU 在训练时计算开销较小，尤其是在数据量较少或计算资源有限的情况下，表现尤为明显[54]。GRU 主要通过更新门（Update Gate）和重置门（Reset Gate）协同调控来实现跨时间步的状态动力学建模[50]。以下是它们的功能和计算方式：

重置门决定了前一时刻的隐藏状态在当前时刻的候选隐藏状态计算中的影响程度。通过调整重置门的值，GRU 控制前一时刻的隐状态对当前时刻隐状态的贡献。公式(2-13)为：

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \tag{2-13}$$

其中W_r是重置门的权重矩阵，h_{t-1}是前一时刻的隐藏状态，x_t是当前时刻的输入，b_r为重置门的偏置项。激活函数$\sigma \in [0, 1]$，通常为 sigmoid 函数。值为 0 表示隐藏状态完全屏蔽，实现历史信息的动态遗忘；而值为 1 表示完整传递至候选状态计算，维持时序连续性。

更新门决定了当前时刻新信息和前一时刻信息的融合程度，即控制当前输入信息对当前隐藏状态的影响。公式(2-14)为：

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \tag{2-14}$$

其中W_z是更新门的权重矩阵，h_{t-1}是前一时刻的隐藏状态，x_t是当前时刻的输入，b_r是更新门的偏置项。该门通过调整z_t的大小，控制保留多少前一时刻的隐藏状态的信息，接收多少当前时刻的输入信息。

2.4.2 加入注意力机制的时序网络

尽管 GRU 和 LSTM 在处理时序依赖任务中表现出较强的记忆能力，但在面对复杂的长序列或信息冗余场景时，依然存在“信息稀释”或“关注分散”的问题。为进一步提升模型的判别能力和对关键时间片段的关注度，研究者引入了注意力机制，使模型能够根据当前任务动态地聚焦于输入序列中最相关的信息区域，从而增强输出的判别性与表示能力。

注意力机制最初在神经机器翻译中提出，随后广泛应用于自然语言处理、图像识别和智能决策等多个领域。其核心思想是：对于每个时刻的输出，模型不再仅依赖于隐藏状态的固定传递，而是引入一组可学习的权重，对历史时间步的隐藏状态进行加权求和，以提取对当前任务最有贡献的信息。注意力机制最初应用于神经机器翻译领域，随后被广泛拓展至自然语言处理、图像识别、视频理解以及智能决策等任务中。其核心思想是：对于某一时间步 t 的输出，模型不再仅依赖于单一隐藏状态h_t，而是考虑对整个历史隐藏状态序列$[h_1, h_2, ..., h_t]$的加权组合。该加权方式由一个可学习的注意力分数函数（score function）确定，通常采用如下公式(2-15)：

$$e_t^i = \text{score}(h_t, h_i) \tag{2-15}$$

其中，h_t 是当前时间步的隐藏状态，h_i 是历史时间步的隐藏状态，e_t^i 是其注意力分数。

常用的打分函数包括：在结合 GRU 网络时，注意力模块通常作为一个上层组件，通过计算每个历史隐藏状态对当前决策的贡献度，动态调整其影响力。这种融合使得模型在无人驾驶等复杂决策环境中，能够更精准地捕捉到交通流中关键的行为信号（如前车突然减速、相邻车辆加速变道等），有效增强对关键时

序特征的提取能力。

例如，有研究在交通行为预测中将 Attention-GRU[67]应用于车辆轨迹建模，显著提升了对交互行为的感知精度；也有工作在行为决策网络中引入注意力机制，以实现动态聚焦重要观测维度，从而提升 Q 值估计的准确性与策略的稳定性。

2.5 本章小结

本章系统阐述了强化学习与深度强化学习的基本原理及核心实现步骤。首先介绍了深度学习在特征提取和表示学习中的优势，以及强化学习的基本框架及其在智能决策中的重要性。随后讲解了深度强化学习（DRL）的基本思想，即融合深度学习的表征能力与强化学习的决策能力。章节重点分析了策略学习、值函数估计、探索与利用的平衡等关键技术，并对各类 DRL 算法进行了比较，探讨其适用性与优劣。此外，重点介绍了本研究采用的优先级经验回放（PER）和门控循环单元（GRU）。PER 通过优先采样机制提升了训练效率和收敛速度；GRU 凭借其高效的门控机制，在处理时序数据上相较 LSTM 具有一定优势，提升了模型在强化学习中的表现。以上内容为后续仿真实验和算法设计提供了坚实的理论支持和科学依据。

第 2 章 基于双优先级经验回放的决斗双深度 Q 决策网络

无人驾驶车辆行为决策控制在自动驾驶的发展中起着至关重要的作用，然而，现有基于深度强化学习的自动驾驶行为决策算法，存在安全性低和稀疏奖励等问题。为了解决这些问题，本章提出一种基于双优先级经验回放的决斗双深度 Q 网络（DPD3QN）。首先，将决斗网络与双深度 Q 网络相结合，并对原有网络输出层进行重构，提高动作价值的估计准确性；然后，引入双优先级经验回放，促使模型更快识别和有效利用关键经验；最后，在 OpenAI gym 仿真平台进行详尽训练和测试，并进行了条件更为苛刻的适应性测试，相比传统决策算法表现出了更安全、高效的无人驾驶决策能力。

3.1 DPD3QN

在强化学习中，需要估计每个状态的值，但对于很多状态，不需要估计每个动作的值。决斗双深度 Q 网络（D3QN）是在 DDQN 的基础上，将 wang[55]提出的决斗网络架构添加到主网络和目标网络中。明确区分了状态值和动作优势值，并将状态值与状态动作分离，以更准确地评估 Q 值。解决了决斗 Q 网络使用同一网络可能导致局部最优的问题。

状态动作值函数 $Q^{\pi}(s,a)$ 表示策略π在状态 s 下选择动作 a 时的期望奖励，状态值 $V^{\pi}(s)$ 表示策略π在状态 s 下产生的期望奖励，则两者之差表示在状态 s 下选择动作 a 的优势 $A^{\pi}(s,a)$[56]，定义为式(3-1):

$$A^{\pi}(s,a)=Q^{\pi}(s,a)-V^{\pi}(s) \tag{3-1}$$

在决斗网络中有两个数据流，一个流输出状态值 $V(s;\theta,\beta)$，另一流输出动作优势值 $A(s,a;\theta,\alpha)$，采用决斗网络结构的双深度 Q 网络输出为式(3-2):

$$Q(s,a;\theta,\alpha,\beta)=V(s;\theta,\beta)+A(s,a;\theta,\alpha) \tag{3-2}$$

其中 θ 表示在输入层上执行特征处理的网络参数；α 和 β 是决斗网络架构中动作优势函数 A 和状态值函数 V 的参数。

由于网络直接输出 Q 值，导致状态值 V 和动作 A 无法直接获取。为体现这种可识别性，对动作 A 进行了特殊处理，调整后的 Q 值如公式(3-3)所示:

$$Q(s,a;\theta,\alpha,\beta)=V(s;\theta,\beta)+\left(A(s,a;\theta,\alpha)-avgA\left(s,a';\theta,\alpha\right)\right) \tag{3-3}$$

其中 $avgA\left(s,a';\theta,\alpha\right)$ 代表所有动作优势函数的平均值，α 和 β 是决斗网络架构中动作优势函数 A 和状态值函数 V 的参数。目标值函数定义式为(3-4)，用于更新网络参数的损失函数定义式为(3-5)：

$$y^{D3QN}=E_{(s,a,r,s')}[r+\gamma\cdot Q_T(s',\arg\max Q_M(s',a';\theta,\omega^V\omega^A);\theta',\alpha,\beta)] \tag{3-4}$$

$$L\left(\theta,\alpha,\beta\right)=E_{(s,a,r,s')}[y^{D3QN}-Q_M(s',a';\theta,\alpha,\beta)^2] \tag{3-5}$$

其中 θ 和 θ' 是主网络和目标网络的参数，Q_T 为目标网络动作价值函数，Q_M 为主网络动作价值函数，$\arg\max Q_M(s',a';\theta,\alpha,\beta)$ 表示选择在状态 s' 下使主网络的 Q 值最大的动作 a'。

本章为解决以上问题，提出了一种双优先级经验回放机制的决斗双深度 Q 网络算法，如图 3-1 所示，并在高速公路场景下进行训练和测试。

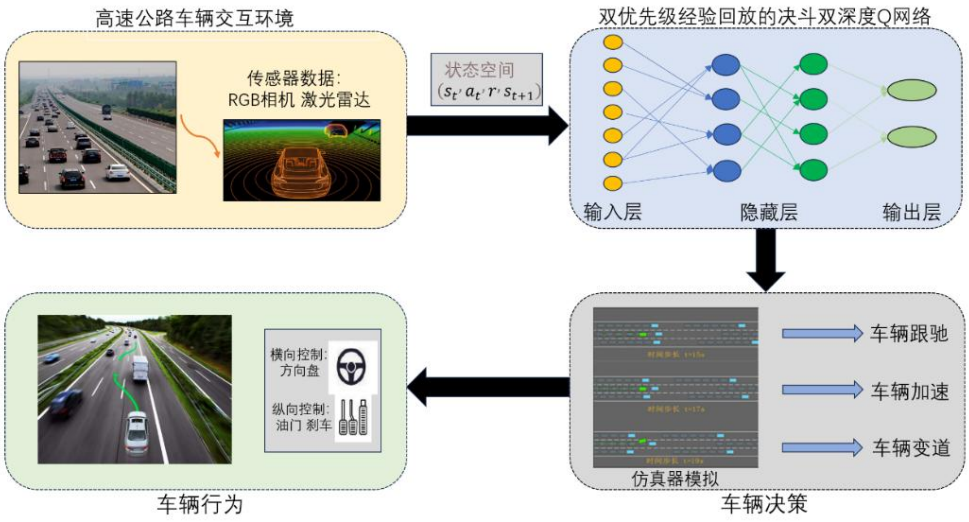

图 3-1 自动驾驶汽车高速公路行为决策方法

图 3-1 基于双优先级经验回放的决斗双深度 Q 网络（DPD3QN）无人驾驶决策流程图。该图展示了自动驾驶车辆在高速公路交互环境中的行为决策机制。

系统首先通过摄像头和激光雷达等传感器感知环境，构建状态空间信息并输入至 DPD3QN 决策网络。决策网络由输入层、隐藏层和输出层组成，融合了决

斗结构和双 Q 学习方法，结合双优先级经验回放机制，有效提升了值函数估计精度与训练样本利用率。输出的最优动作（如跟车、加速、变道）被输入至车辆控制模块，执行相应的横向和纵向控制操作，完成智能驾驶闭环决策过程。该结构显著提升了系统在复杂交通环境中的稳定性与决策效率。本章建立了一个四车道高速公路仿真环境，并采用分层结构对周围车辆进行运动控制。自动驾驶汽车装备的传感器实时获取当前场景下的环境信息，并将环境状态传入经验回放池，产出动作状态方程后传入决斗双深度 Q 网络。输入层神经网络通过隐藏层逼近状态方程，输出层将相对价值映射到当前场景下的每个决策动作。

3.2 网络结构优化

DPD3QN 是融合了双优先级经验回放的决斗双深度 Q 网络，其结构如图 3 所示。其核心思路是通过决斗网络架构提高状态和动作的估计准确性，并通过双优先级经验回放机制增强样本的利用率，加速训练过程，特别适用于自动驾驶等的顺序决策问题。图 3-2 展示了一种基于双优先级经验回放的决斗双深度 Q 网络的强化学习架构，以提高智能体的学习效率和决策精度。

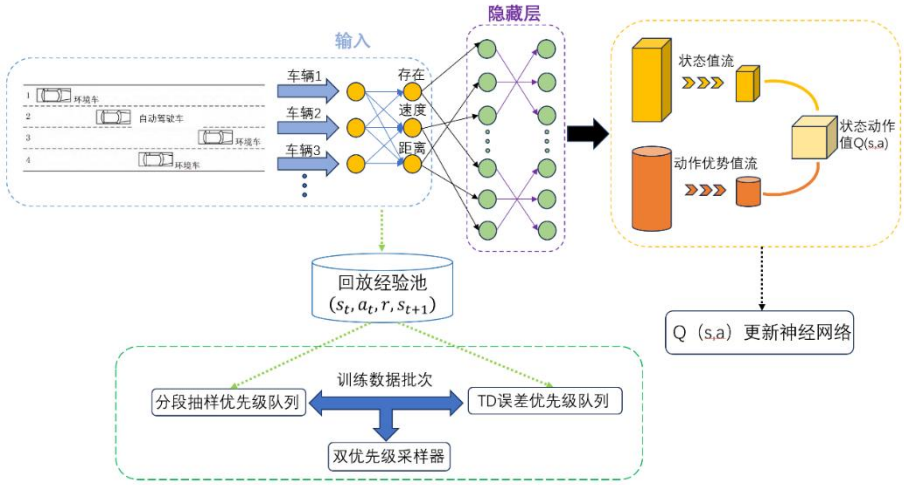

图 3-2 DPD3QN 网络结构图

整个流程可以分为以下几个模块：

（1）输入模块：在输入层，使用了多辆车的状态信息作为输入，包括车辆的存在、速度和距离等特征。每一辆车的特征被向量化，构成状态表示 s_t，并输入到神经网络的隐藏层中。

（2）回放经验池：智能体在与环境交互过程中生成的经验元组(s_t, a_t, r, s_{t+1})

被存储在经验回放池中。经验回放池的使用能够有效打破数据样本的时间相关性，并且提高样本的使用效率，防止过拟合。这一模块通过存储过去的经验数据，保证了训练过程的稳定性。

（3）优先采样机制：为了进一步提高学习效率，回放经验池采用了双优先级采样机制。首先采用 TD[57]误差来衡量样本的重要性，并通过优先级队列对经验样本进行排序，优先选择误差较大的样本进行训练。同时，采用分段采样机制，从不同优先级队列中抽取样本，确保了训练样本的多样性。这种方法能够加速模型收敛，并提升样本利用率。

（4）神经网络结构：如图 3 所示，网络的隐藏层负责提取输入状态的高维特征。隐藏层之后，网络分为状态值流和动作优势流。状态值流 V 估计当前状态s_t的价值，而动作优势流 A 则估计在该状态下各个动作a_t的相对优势。这样的设计基于决斗网络结构，能够使模型更好地区分不同状态下的动作选择，从而提升决策质量。最后两条流合并，输出 Q 值$Q(s,a)$，用于智能体选择最优动作。

(5)Q 网络更新：通过计算估计 Q 值与目标 Q 值之间的差距来更新网络参数。具体而言，使用贝尔曼方程计算 TD[57]误差，指导网络权重的调整。这一过程通过多次迭代，使得模型逐渐收敛，智能体能够在动态环境中学习到最优策略。

3.3 双优先级经验回放

3.3.1 基于 TD-Error 的优先级

经验回放最早是由 Schual[25]提出，并将其应用 DQN，该方法根据时间差误差(temporal difference, TD) [57]来确定每个样本的重要性，优先级越高的样本越有可能被挑选出来进行训练。这个优先级可以基于 TD 误差来计算。TD 误差表示预测值与实际奖励值之间的差异，其计算式(3-6)如下：

$$\delta = | r + \gamma \max_{a'} Q(s', a'; \theta^{-}) - Q(s, a; \theta) | \tag{3-6}$$

式中，r 为即时奖励，γ 为折扣因子，$Q(s', a'; \theta^{-})$ 是目标网络输出，$Q(s, a; \theta)$ 是当前网络的输出。将双网络产生的两个 TD 误差与即时奖励之和作为划分重要性的依据。经验的优先级值 $p(i)$ 可以定义为式(3-7)：

$$p(i) = (\delta_i + \text{ò})^{a} \tag{3-7}$$

式中，ò是一个很小的常数，防止数据的优先级为零，超参数a决定了 TD 误差δ_i对样本选择的影响程度，控制了采样时对误差的敏感性，优先级高的经验将有更大的机会被采样。TD 误差越大，优先级越高。

在早期的训练过程中，TD 误差可能不准确。考虑到上述因素，因此需要在训练过程的早期对训练数据进行采样时降低优先级权重。同时，为了确保甚至可以选择优先级极低的样本，相比传统优先级概略分布，在优先级中加入了一个偏差b[25]。一个经验样本被取走的概略分布改进为式(3-8)：

$$P(i)=\frac{p_i^{\alpha}}{\sum_{i=1}^{k} p_i^{\alpha}}+b \tag{3-8}$$

式中$P(i)$为第i个经验样本被取走的概率，α为决定$P(i)$优先级的常数。这里使用超参数α来防止选择始终具有高优先级的经验。否则，贪婪的优先级会导致过拟合，因为相似的经验总是会被选择。α值接近 0 意味着代理关注的是均匀抽样，而接近 1 的值表明它关注的是基于优先级的抽样。

由于经验回放系统更偏向于重复利用 TD 误差较大的经验，这将不可避免地影响对不同状态的采样频率，从而可能导致神经网络训练过程中出现震荡，甚至发散的情况。[58]。为了解决这一问题，在计算权重更新时引入了重要性采样权重。具体如公式(3-9)所示。

$$w(i)=\left(\frac{1}{N}\cdot\frac{1}{P(i)}\right)^{\beta} \tag{3-9}$$

式中，β是控制采样重要性偏差的参数，N是经验池的总量。在车辆与环境的训练交互之中，β逐渐从初始值增加到 1。根据重要性采样权重调整损失函数，更新梯度，以减少由优先级采样引入的偏差。

3.3.2 基于分段抽样的优先级

在基于 TD 误差的优先级基础上，进一步引入了分段抽样方法（Stochastic Prioritization），该方法的灵感来源于 DDPG 深度强化学习算法中的小批量梯度下降（mini-batch Gradient Descent）[59]。通过将数据划分为多个小批次，分段抽样能够在复杂环境中实现高效学习与稳定性。此方法有效提升了经验池中数据的利

用率，使不同优先级的样本能够得到有效抽取，从而加快模型的收敛速度，避免高维连续动作空间中的过拟合或数据利用不足问题。

分段抽样优先级的具体步骤如下：首先将特征划分为两个主要段，位置段（Position Segment），包括前车距离或相对车道位置；速度段（Speed Segment），涵盖车辆速度范围。整个经验回放缓冲区的数据根据这些特征被分成不同的段。定义每个样本 i 所属的段为 $s(i)$，并为每个段设定权重。

对于位置段，可以根据前车距离的稀缺性和重要性设定优先级。在某些特定的距离范围内（如距离较近或较远），样本数量较少，则应给予更高的权重，以确保这些关键距离的场景得到充分关注。可以通过测量位置段内样本的标准差来捕捉这种多样性。具体的权重设置可以表示为式(3-10)：

$$w_{\text{position}}(s(i)) = g(n(s(i)), \sigma_{\text{pos}}(s(i))) \tag{3-10}$$

式中 $w_{\text{position}}(s(i))$ 是位置段 $s(i)$ 的权重，$n(s(i)$ 是段 $s(i)$ 中样本的数量，$\sigma_{\text{pos}}(s(i))$ 是段 $s(i)$ 中样本的前车距离标准差。较高的标准差表示段内场景变化较大，因此需要更多关注。

同时，速度段的权重可以表示为式(3-11)：

$$w_{\text{speed}}(s(i)) = h(n(s(i)), \sigma_{\text{speed}}(s(i))) \tag{3-11}$$

式中 $w_{\text{speed}}(s(i))$ 是速度段 $s(i)$ 的权重，$\sigma_{\text{speed}}(s(i))$ 是段 $s(i)$ 中样本的速度标准差。此外，高速公路上的紧急情况（如急刹车或避让）相较于普通驾驶场景显得尤为重要，因此人为给予这些段更高的权重。可以设定普通驾驶的权重为 $w_{\text{normal}} = 1$，紧急刹车的权重为 $w_{\text{emergency}} = 3$，变道操作的权重为 $w_{\text{lane change}} = 3$，而近距离行驶的权重为 $w_{\text{close}} = 2$。通过这样的优先级设定，确保训练过程中对重要场景的充分关注，从而提升模型的整体性能与适应性。

3.3.3 双优先级计算

基于 TD 误差优先级和分段抽样优先级计算出综合的优先级，即双优先级。对于每个样本 i，其综合优先级 P_i 可以通过结合 TD 误差优先级和分段权重来计算，综合优先级可以表示为式(3-12)：

$$P(i)=\alpha\cdot P_{\mathrm{TD}}(i)+\beta\cdot(w_{\mathrm{position}}(s(i))+w_{\mathrm{speed}}(s(i))) \tag{3-12}$$

式中，$P_{\mathrm{TD}}(i)$是样本i的 TD 误差优先级，$w_{\mathrm{position}}(s(i)$ 和 $w_{\mathrm{speed}}(s(i))$分别是样本所属的位置信息和速度信息段的权重，α和β是控制各自权重影响的超参数。基于双优先级$P(i)$，接下来进行加权抽样，以确保模型能够更加关注重要的样本。具体而言，样本i的抽样概率可以通过下式(3-13)确定：

$$P(s(i)=k)=\frac{P_i}{\sum_j P_j} \tag{3-13}$$

式中，$P(s(i)=k)$表示在第i次选择中，选择的是第k个对象的概率，而分母 $\sum_j P_j$则是所有样本综合优先级的总和这样，样本i被抽样的概率与其综合优先级成正比。

需要说明的是，α+β并不强制约束为 1，这一设定为算法设计提供了更大的灵活性，使得在训练的不同阶段可以根据模型的学习需求动态地调整 TD 误差与分段采样权重之间的相对影响。例如，在训练初期，模型尚未掌握有效的策略，TD 误差可能波动较大，此时可以适当增加 α 的权重，以鼓励模型更快地修正错误估计；而在训练后期，当策略逐渐稳定时，可以增大 β 的权重，引导模型聚焦于稀有但关键的场景，如高速变道、紧急刹车等，从而提升策略的泛化能力和实际适应性。双优先级经验回放的结构图如图 3-3 所示。

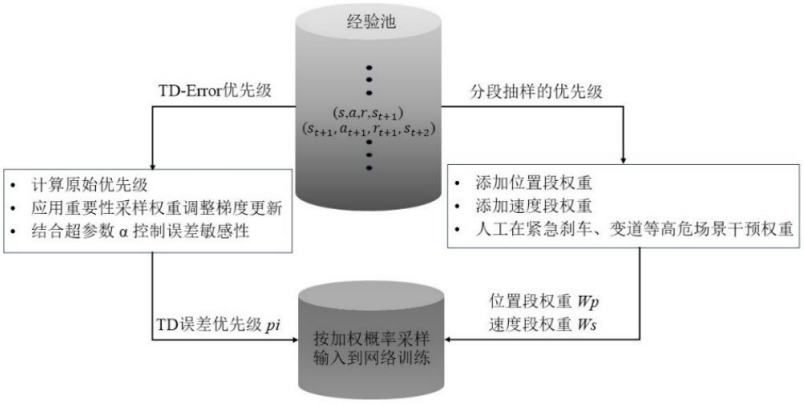

图 3-3 双优先级经验回放结构图

这种结合了 TD 误差和分段权重的优先级抽样方法-即双优先级抽样，能够在无人驾驶高速公路场景下更有效地利用经验回放数据，尤其是在处理稀有但重

要的场景时，如高速变道、紧急刹车。可以帮助 DPD3QN 网络更好地学习这些关键决策。并保证训练过程的稳定性和收敛速度。

为了更直观地展示双优先级的计算过程，以下给出一个简要的计算示例，如下表 3-1 所示。该示例清晰地展示了如何将 TD 误差与语义段权重相结合，计算出综合优先级，并进一步归一化为采样概率。

表 3-1 双优先级计算示例

示例：双优先级计算
考虑一个样本，其属性如下：
TD 误差: p_i^{TD} = 0.75
位置段权重: w_i^{pos} = 0.20
速度段权重: w_i^{vel} = 0.15
给定的超参数: α = 0.6, β = 0.4
则该样本的综合优先级为：
P_i = 0.6 × 0.75 + 0.4 × (0.20 + 0.15) = 0.45 + 0.14 = 0.59
当前采样批次中所有样本的优先级之和为： $\sum_j P_j$ = 5.9,
则该样本的归一化采样概率为：
P_i^{sample} = 5.9/0.59 = 0.10

3.4 网络参数更新

在融合了决斗网络和 DDQN 网络结构的基础上，本方法使用双优先级经验回放更新经验池，对神经网络的参数进行更新，如图 3-3 所示。

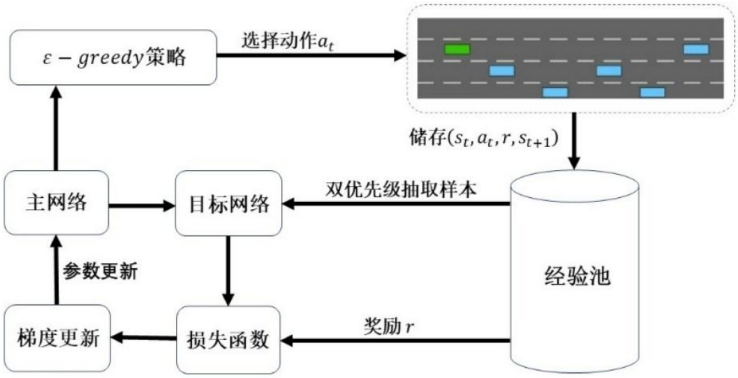

图 3-4 DPD3QN 算法更新图

在无人驾驶过程中，首先系统通过ϵ-greedy 策略从主网络中选择当前时刻动作 a_t。这种策略在探索选择随机动作和利用选择最优动作之间进行权衡，通过一个小概率的随机选择动作探索新的策略，之后选择主网络评估出的最优动作。选定动作后，系统在模拟的驾驶环境中执行动作 a_t，并与环境进行交互，产生下一个状态 s_{t+1} 和奖励值 r，这些信息（当前状态 s_t、动作 a_t、奖励 r 和下一个状态 s_{t+1}）会被存储到经验池中。经验回放池会记录多个历史数据，用于训练时从中进行经验回放。无人驾驶车辆系统从经验池中通过双优先级抽取样本，并通过目标网络计算这些样本的 TD 误差，以此计算损失函数。TD 误差反映了预测与实际奖励的差距，是训练的核心优化目标。通过反向传播算法，使用梯度下降方法对主网络的参数进行更新，以最小化损失函数，使得主网络的预测能力不断提升。为了保持稳定性，目标网络的参数并不会实时更新，而是周期性地从主网络同步其参数。这种结构有助于防止目标值波动过大，从而提高训练的稳定性。系统不断重复上述过程，逐步更新网络参数，改善策略，从而在无人驾驶场景下实现更加智能的决策。

ϵ-greedy 策略用于在强化学习的训练过程中平衡探索和利用，确保智能体在学习过程中既能探索新的状态和动作，又能利用已有的知识选择最优动作。具体应用步骤如下：对于每一个时间步，以概率ϵ选择一个随机动作，以概率 1−ϵ选择当前 Q 网络预测的最优动作。在一定的训练步骤后，对策略进行评估，根据评估结果调整探索策略或算法参数。探索的过程通常是逐渐减少ϵ的值，使得智能体在训练初期更多地进行探索，而在训练后期更多地进行利用。其算法如公式(3-14)所示：

$$\epsilon_t = \max(\epsilon_{\min}, \epsilon_{decay} \cdot \epsilon_{t-1}) \tag{3-14}$$

式中 ϵ_t 是在时间步 t 的探索率，$\epsilon_{\min}$ 是探索率的最小值，ϵ_{decay} 是探索率的衰减因子，ϵ_{t-1} 是在时间步 *t-1* 的探索率。该策略确保探索率在每个时间步以固定比例逐步递减，但不会低于预设的最小值，从而在训练初期鼓励探索，在训练后期逐渐趋于稳定、侧重策略利用。

3.5 实验及其仿真环境

本节主要研究的车道高速公路驾驶环境以及用于管理驾驶场景中环境车辆横向运动和纵向运动的分层控制模型。具体而言，上层控制模型由智能驾驶员模型（IDM）[60]和最小化车道变换组成的整体制动模型（MOBIL）[61]组成，主要用于管理车辆的纵向和横向运动。而下层控制模型则采用比例和微分控制器进行运动控制。此外，本节还详细介绍了实验所涉及的具体参数。

3.5.1 实验环境配置

本章实验所使用的实验软件环境配置参数如表 3-2 所示。

表 3-2 环境配置参数

环境配置	版本序列
操作系统	Windows11 64 位
显卡	GeForce RTX 3070 Ti (8G)
运行内存	16G
Python	3.7.16
torch	2.2.1
gym	0.26.2

3.5.2 高速公路环境

无人驾驶中的行为决策主要是指通过序列化驾驶行为实现特定的驾驶目标。在本实验的高速公路驾驶场景中，上述的驾驶行为主要包括，加速、减速、换道（左转和右转）和巡航驾驶。具体的研究场景如图 3-5 所示，包含无人驾驶车和环境车。

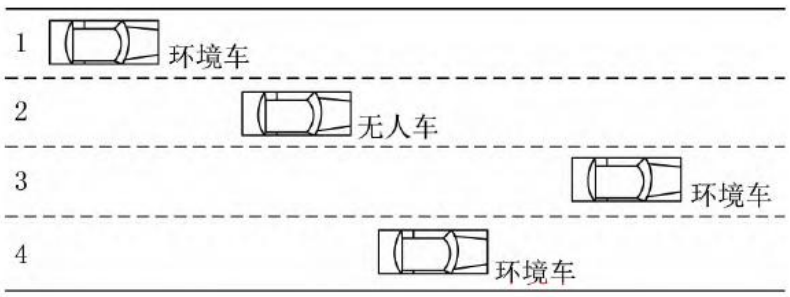

图 3-5 高速公路四车道行车决策环境

为了增加驾驶场景的真实性和不确定性，实验对环境车辆的初始速度和初始

位置进行了随机设置。这种做法有助于更真实地模拟实际驾驶情况，并增强自动驾驶汽车在复杂环境中的适应能力。此外，为了更好地模仿人类的驾驶习惯，实验设定无人驾驶车辆在没有超车需求时尽量靠右行驶。这些设置旨在提高驾驶场景的逼真度。

在驾驶任务开始前，在四车道高速公路上随机生成 50 辆环境车，并将它们排列在自动驾驶车辆的前方。在驾驶场景中，从无人驾驶车辆开始行驶到停止的整个过程被定义为一个回合。当无人驾驶汽车遇到以下情况之一时，回合结束并自动开始下一回合：（1）车辆顺利行驶到达预设的时间上限；（2）自动驾驶车辆与其他环境车辆发生碰撞。

关于车辆参数的具体设置，参见表 3-3。

表 3-3　车辆参数设置

参数	数值
车道数	4
车辆长度/m	5
车辆宽度/m	2
环境车辆初始速度/（m/s）	20-25
自动驾驶车辆初始速度/（m/s）	25
模拟仿真频率/hz	10
回合时间上限/s	30

高速公路驾驶场景中的所有车辆均设置为 5m 长、2m 宽。为了增加不确定性，环境车辆的初始速度随机分配在 20-25m/s 之间，而自动驾驶车辆的初始速度为 25m/s。仿真频率为 10Hz，每个回合的持续时间上限为 30s。

3.5.3 车辆行为控制

为实现复杂交通场景下的车辆交互模拟，本章研究采用分层控制架构设计环境车辆行为。环境车辆由智能驾驶模型（Intelligent Driver Model，IDM）[60]控制和整体制动模型（Minimizing Overall Braking Induced by Lane Changes，MOBIL）[61]来实现。同时使用深度强化学习方法来控制自动驾驶车辆的行为。

IDM 是一种常见的微观模型，用于控制车辆的纵向行为。该模型通过监测车辆之间的距离和速度差异，来实现车辆的跟随和避免碰撞。通常情况下，IDM

模型中的纵向加速度可以通过公式(3-15)和(3-16)来描述。

$$\dot{v} = a_{\max}\left[1-\left(\frac{v_c}{v_t}\right)^{\delta}-\left(\frac{d^*}{d_c}\right)^2\right] \tag{3-15}$$

$$d^* = d_{\min} + Tv_c + \frac{v_c\Delta v_c}{2\sqrt{a_{\max}b_{comfort}}} \tag{3-16}$$

式中v_c和$\dot{v}$是环境车辆的速度和加速度，$a_{\max}$是环境车辆最大加速度，d_c是环境车辆到前方车辆的距离，v_t为环境车辆的目标速度，δ被称为速度指数参数。$d_{\min}$是预定义的与环境车之间的最小跟驰距离，T是安全目标的预期时间间隔，Δv_c是控制车辆与前方车辆的相对速度，$a_{\min}$为环境车辆最大减速度。

本章中设置 IDM 参数如下：最大加速度$a_{\max}$为 6m/s^2,速度指数参数δ为 5，所需时间间隔T为 1.5s，舒适减速度阈值$b_{comfort}$为-5m/s2，最小跟驰距离$d_{\min}$为 10m。该参数配置可平衡通行效率与安全裕度，适应高速公路场景需求。

与此不同的是，MOBIL 模型控制车辆的横向车道变更决策。该模型考虑了车辆之间的安全性和激励标准，确保在进行车道变更时，新的跟随车辆不会由于突然的减速而造成碰撞风险。具体而言，MOBIL 模型利用公式(3-17)来计算新跟随者的加速度$\tilde{a}_n$，其中b_{safe}是在变道过程中施加给跟随者的最大制动。加速度阈值a_{th}参见式(3-18)，对自车及其追随车辆施加激励，以确保变道过程安全顺利，避免与其他车辆的碰撞。

$$\tilde{a}_n \geq -b_{safe} \tag{3-17}$$

$$\underbrace{\tilde{a}_e - a_e}_{egovehicle} + p\cdot\left(\underbrace{a_n - a_n}_{newfollower} + \underbrace{a_0 - a_0}_{oldfollower}\right) \geq a_{th} \tag{3-18}$$

通过遵循式(3-17)，(3-18)可以确保在变道后控制车辆、原有跟随者和新的跟随者三者的速度都有提高。实验中，MOIBL 模型使用的参数设置如下：礼貌因子p设置为 0.0001，安全减速度限制b_{safe}为 2.5m/s^2，加速度增益a_{th}为 0.25m/s^2。该机制确保换道后自车、原跟随车与新跟随车的整体加速度收益正向增长，同时避

免引发连锁制动反应。

3.5.4 车辆运动控制

在车辆运动决策过程完成后，底层控制层主要通过比例控制器和比例微分控制器来调节车辆的纵向和横向运动[62]。纵向运动由比例控制器管理加速度，其数学定义如式(3-19)。

$$\dot{v} = K_p \left(v_r - v \right) \tag{3-19}$$

其中 $\dot{v}$ 表示车辆的当前纵向加速度，v 是车辆的当前速度，v_r 是参考速度，由 IDM 动态生成。K_p 是控制器的比例增益。

在横向运动方面，比例微分控制器负责控制车辆的横向位置，其表达式定义如式(3-20)和(3-21)。

$$v_{lat,r} = -K_{p,lat} \Delta_{lat} \tag{3-20}$$

$$\Delta \psi_r = \arcsin\left(\frac{v_{lat,r}}{v} \right) \tag{3-21}$$

其中 $v_{lat,r}$ 是横向速度命令，$K_{p,lat}$ 是位置增益，Δ_{lat} 是车辆相对于车道中心线的横向位置。控制架构在仿真环境中实现 10Hz 的执行频率，满足实时性要求。

3.5.5 实验参数设置

本章设定无人驾驶汽车能够感知距离其位置前后各 200 米范围内最近的 6 辆环境车辆，并准确获取这些车辆相对于自动驾驶车辆的距离和速度信息，并将其作为环境状态的重要组成部分。策略更新频率设置 1Hz[12]。实验中的道路被划分为 4 条车道，周围环境包含了 50 辆其他车辆。其环境状态的定义式如下：

自动驾驶车辆感知半径 200 米，优先追踪最近 6 辆环境车辆（前 3 后 3）的相对位置(xi ,yi)与速度 vi ，构成状态向量，并通过式(3-22)构建归一化状态向量，以消除量纲差异对网络训练的影响：

$$s_t = \bigcup_{i=1}^{6} \left[\frac{x_i}{200}, \frac{y_i}{3.5}, \frac{v_i}{30} \right] \tag{3-22}$$

其中分母项分别对应感知半径最大值（200 m）、车道标准宽度（3.5 m）及高速公路限速阈值（30 m/s），该设计可增强状态空间表征的物理可解释性。

基于 Dueling Double DQN 框架，采用解耦的价值流（Value Stream）与优势流（Advantage Stream）双分支结构。价值网络与优势网络均采用 3 层全连接架构，隐层维度为 64，激活函数选用 ReLU（Rectified Linear Unit）以缓解梯度消失问题。输出层采用线性激活函数，分别生成状态价值估计 $V(s)$与动作优势值 $A(s,a)$，并通过聚合操作，计算最终 Q 值，如公式(3-23)。

$$Q(s,a)=V(s)+(A(s,a)-\frac{1}{|\mathrm{A}|}\sum_{a'}A(s,a')) \tag{3-23}$$

设置经验回放池容量设置为N_{buffer} =5×104，采用双优先级采样机制，其温度参数配置为α=0.6（TD 误差优先级权重）与β=0.4（片段权重系数），以平衡探索与利用的效能。同时通过ϵ-greedy 策略动态调节探索率，初始值 ϵ_{init} =1.0（完全随机探索），随训练进程线性衰减至ϵ_{final} =0.05（5000 回合内完成迭代），兼顾早期探索效率与后期策略收敛性。

奖励函数r_t的设计融合多目标优化思想，由最优控制目标组成，该目标是避免在行驶过程中发生碰撞，尽可能快速地行驶在右侧车道。为实现这一控制目标，回合奖励函数设置如下式：

$$r_t=1-c_{collision}+0.3\cdot\frac{v}{30}+0.2\cdot(2-|l_{\text{lane}}|) \tag{3-24}$$

式中r_v为速度奖励，$r_v\in[24,30]$，L为所在车道位置。对拟确定的行为决策策略，其中碰撞标志$collision\in[0,1]$（0 代表未发生碰撞，1 代表碰撞发生），车道偏离惩罚系数$|l_{\text{lane}}|\in[0,2]$。在 OpenAI Gym[63]开源平台的环境中进行评估和测试。

3.6 仿真结果分析

3.6.1 训练分析

本部分主要从反应算法性能的不同指标对三种算法 DQN[22]、DDQN[23]、DPD3QN 进行优越性的比较，三种算法默认参数相同。

图 3-6 展示了三种算法的回合奖励情况，根据式(3-24)奖励函数的定义，高奖励表示无人驾驶汽车在高速公路上以一种更有效的方法行驶。所提出的 DPD3QN 算法在训练过程中的学习速度显著快于传统的 DQN 和改进后的 DDQN 方法。

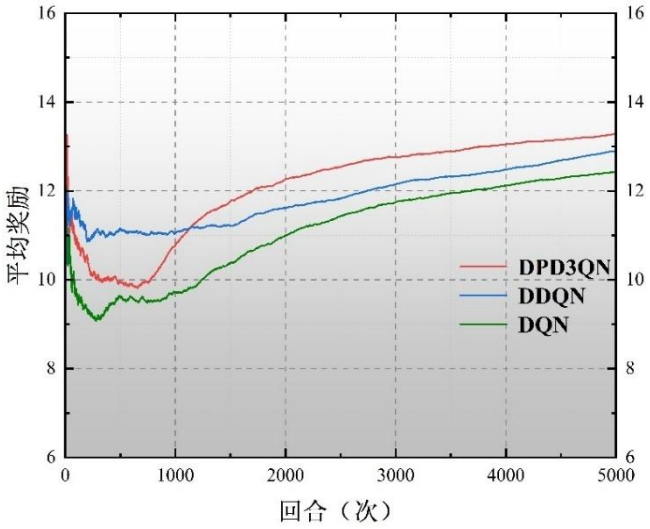

图 3-6 DPD3QN、DDQN、DQN 平均奖励

具体表现为，在大约第 1200 个训练回合后，DPD3QN 的平均奖励值首次超过其余两种对比算法，并在此之后持续保持领先优势，直至整个训练阶段结束。这表明 DPD3QN 能够更早地收敛到较优的策略，并在训练过程中展现出更强的策略稳定性与收敛效率。

在训练总回合数达到 5000 时，三种算法的平均奖励值分别为：DQN 为 12.48，DDQN 为 12.92，而 DPD3QN 达到 13.35，展现出相较基准模型明显的性能提升。该性能优势主要得益于 DPD3QN 中引入的决斗网络结构（Dueling Network Architecture），该结构将状态值函数 $V(s)$ 与动作优势函数 $A(s,a)$ 分开建模，使得在某些状态下即使不同动作的影响微弱，网络也能更准确地评估状态的重要性。这一机制有效提升了动作价值估计的精度，从而帮助自动驾驶智能体更快地识别和执行高质量的动作策略。

图 3-7 展示了三种决策方法在训练过程中每个回合的行驶距离。行驶距离主要受碰撞条件和车辆速度影响，较长的行驶距离意味着无人驾驶车辆能在长时间内高速行驶而不发生碰撞。由于高速公路环境中其他交通参与者（即周围车辆）的初始速度和空间位置存在一定的随机性，这种不确定性在强化学习训练初期对智能体（无人驾驶车辆）的行驶策略产生较大干扰，从而导致车辆在前期训练过程中其行驶距离波动较大。在这一阶段，由于探索机制的主导作用，三种算法在行为表现上未拉开明显差距，均处于策略初步形成和环境适应阶段。

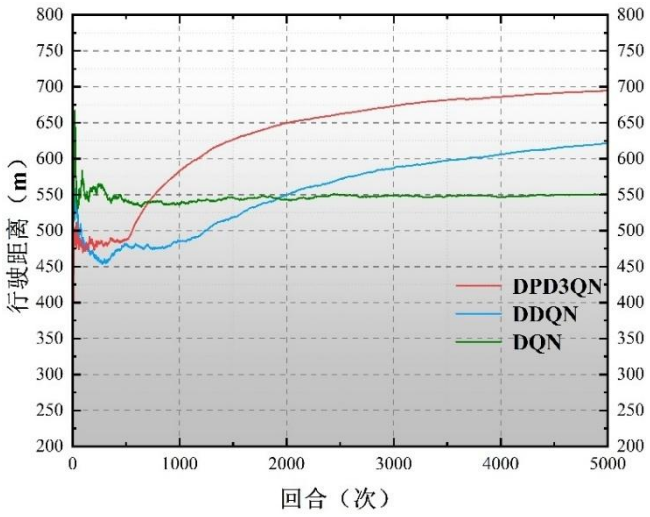

图 3-7 训练过程中的行驶距离

然而，随着训练回合数的增加，尤其是在大约第 800 回合之后，所提出的 DPD3QN 算法开始在行驶距离指标上逐渐超越传统 DQN 和改进的 DDQN 方法，并且这一领先趋势一直持续至训练结束。根据图表数据显示，在训练达到第 5000 回合时，各方法在平均行驶距离方面的表现分别为：DQN 为 551 米，DDQN 为 622 米，而 DPD3QN 则显著提升至 694 米，领先幅度明显。

该性能提升主要归因于 DPD3QN 所引入的双优先级经验回放机制，该机制在样本采样阶段同时考虑了 TD 误差与分段优先策略，从而使模型能够更加高效地识别并利用关键经验样本。通过加权抽取更具代表性的经验片段，DPD3QN 有效增强了对有价值行为路径的学习能力，加速了策略的优化进程。同时，在面对高速公路场景中动态变化的交通流状态时，该方法能够更快速地适应周围环境变化，并做出更为合理和安全的驾驶决策，从而在有限的学习时间内实现更长的行驶距离。

图 3-8 展示了三种算法在回合持续时间方面的表现。较长的回合时间意味着自动驾驶车辆在高速公路上能够更安全地行驶。实验结果显示，在训练进行到第 5000 回合时，三种强化学习算法在平均每回合步数（即智能体在无碰撞情况下持续决策与行驶的时间长度）上的表现呈现出明显差异。

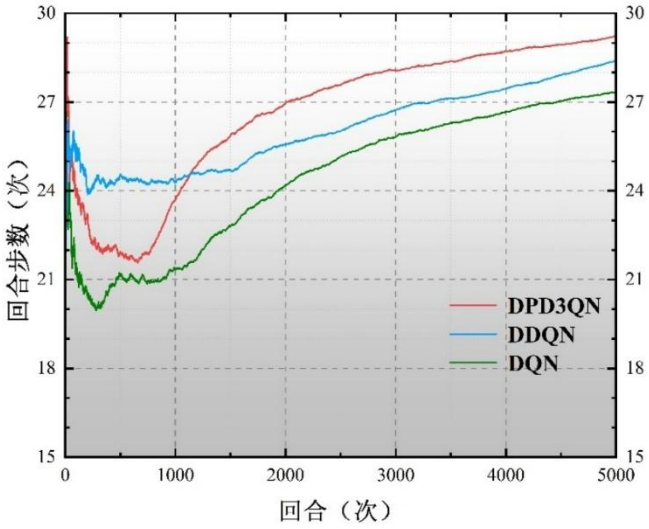

图 3-8 算法训练过程中的回合步数

具体而言，传统 DQN 的平均回合步数为 27.33，改进的 DDQN 为 28.38，而所提出的 DPD3QN 模型则达到了 29.23，处于显著领先位置。进一步统计各算法在整个训练过程中回合步数超过 26 步的次数，结果显示：DQN 为 1811 次，DDQN 为 2519 次，而 DPD3QN 则高达 3438 次。这一指标是衡量模型安全性和驾驶持续性的关键参考，因为较高的回合步数通常意味着智能体在当前场景中有效规避了碰撞、刹车过急或违规变道等不良行为，从而能够维持更长时间的稳定行驶。相比之下，传统 DQN 和 DDQN 在此方面仍存在一定局限，容易在复杂情境中做出潜在危险决策。

3.6.2 适应性测试

A、测试场景I

在对高速公路环境进行学习和训练后，本章构建了两个更加严峻的驾驶场景，测试无人驾驶汽车的自适应能力。测试场景I，测试回合均设置为 1000，车道数均设置为 3，具体的三车道研究场景如图 3-9 所示，包含无人驾驶车和环境车。

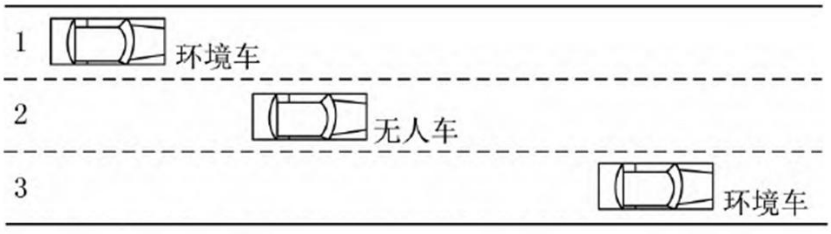

图 3-9 高速公路三车道行车决策环境

图 3-10 展示了三种模型在测试场景中成功率的变化轨迹。周围汽车数量和

其他默认参数均与训练过程相同。同时，训练阶段的神经网络参数已被保存，可以直接在新的环境条件下使用。

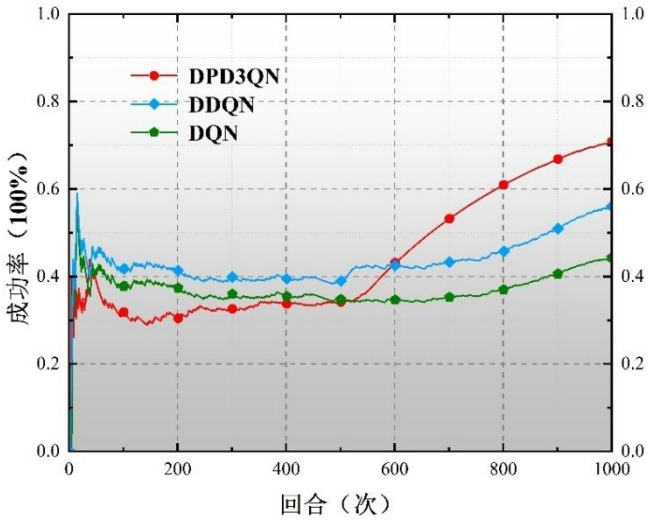

图 3-10 测试场景I成功率

随着测试回合数的增加，控制动作的选择使无人驾驶车辆能够持续达到规定的时间上限，因此成功率呈上升趋势。在测试回合结束时，DPD3QN 模型的成功率达到了 72.6%。这表明，DPD3QN 能够有效量化当前控制动作的潜在价值，并对目标网络参数进行更新，从而提升了控制性能。

测试场景I无人驾驶车辆的成功率、平均奖励以及平均速度的具体测试结果见表 3-4：

表 3-4 测试结果I

算法模型	成功率（%）	平均奖励	平均速度（m/s）
DPD3QN	70.6	13.78	27.78
DDQN	58.8	12.56	26.54
DQN	44.8	11.19	26.31

在诸如图 3-11 的两种驾驶场景时，无人驾驶车辆表现出不同的结果。图 3-11（a）显示了一种现实生活常见的驾驶场景，三辆环境车辆分别占据着三个车道，导致后面的无人驾驶车辆无法进行更高速度的行驶，无人驾驶车辆不得不做出等待，在跟随环境车辆两秒以后做出加速换道决策，并成功完成了车道变换。图 3-11（b）则是另外一种换道结果，因无人驾驶车辆在跟随前车时未能及时判断右车道环境车辆速度，导致不能及时进行变道，与其他车辆发生了碰撞。

（a）　　（b）

图 3-11 自动驾驶车辆换道场景

B、测试场景II

在测试场景I的基础上，进一步构建更为复杂的驾驶场景，把感知距离无人驾驶车辆位置前后各 200 米范围内的 6 辆环境车辆，添加到 10 辆。同时，将策略更新频率设置为 0.5Hz，使模型对环境变化的适应速度较慢。其他参数均和场景I保持相同。

图 3-12 展示了三种决策算法在测试过程中的碰撞情况，其中碰撞的定义为两个值：碰撞=0 表示无人驾驶车辆与其他车辆未发生碰撞；碰撞=1 表示无人驾驶车辆与其他车辆发生碰撞。

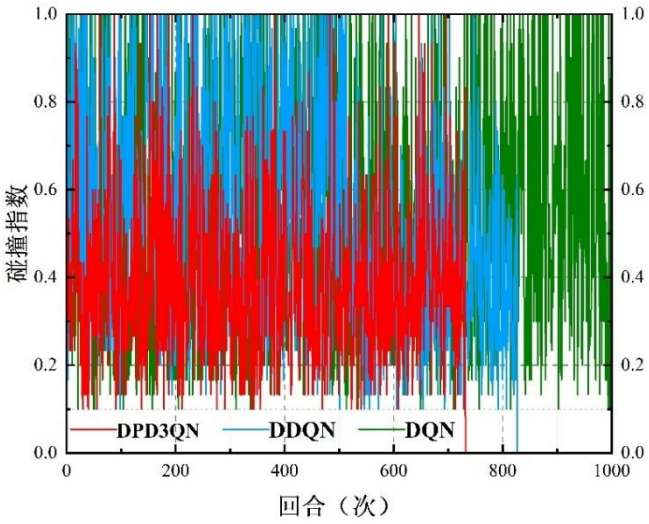

图 3-12 测试场景II中碰撞情况

结果显示，DPD3QN、DDQN 分别在经过 733、和 827 回合后成功避免了碰撞事件的发生。这表明，DPD3QN 方法在处理复杂场景时表现尤为出色。这种优势归因于双优先级经验回放的引入，这种方法帮助无人驾驶车辆更快地识别和利用关键的经验，从而加速学习过程并做出更优的选择。此外，决斗网络结构的

存在使得该算法在更新动作价值时，同时考虑了状态价值，从而更准确地获得每个动作对应的 Q 值。场景II无人驾驶车辆的具体测试结果见表 3-5：

表 3-5 测试结果II

算法模型	成功率（%）	平均奖励	平均速度（m/s）
DPD3QN	65.7	12.65	26.95
DDQN	56.9	11.01	26.11
DQN	43.5	9.79	25.89

这一测试结果源于测试场景复杂度的提高。随着场景的复杂性增加，无人驾驶车辆在决策过程中的成功率下降，平均速度减缓，同时碰撞发生的频率上升，导致每一回合获得的奖励减少，从而使得平均奖励出现下降趋势。

从表 3-5 中可以看出，在场景II的测试结果中，DPD3QN 方法在成功率方面分别比 DDQN 和 DQN 高出 8.8%和 22.2%。在平均奖励和平均速度方面，DPD3QN 方法的表现也明显优于其他两种方法。具体而言，在测试回合结束时，DPD3QN、DDQN 和 DQN 三种算法的平均奖励分别为 12.65、11.01 和 9.75，平均速度分别为 26.95 m/s、26.11 m/s 和 25.89 m/s。这进一步证明了本章所提方法在学习效率、决策质量和稳定性方面优于对比方法。

3.7 本章小结

本章围绕高速公路场景下的无人驾驶行为决策，提出了一种基于双优先级经验回放的决斗双深度 Q 网络（DPD3QN），并通过与 DQN、DDQN 的对比验证了其性能优势。

构建了融合决斗网络与双 DQN 的高效决策框架，优化输出结构以提升状态理解与动作价值估计精度；引入双优先级经验回放机制，提升了关键经验的利用效率，加快了模型收敛；在多环境实验中，DPD3QN 在学习效率、决策质量与稳定性方面表现优越，场景I和II中的决策成功率分别较 DDQN 提升 11.8%、8.8%，较 DQN 提升 25.8%、22.2%，在平均奖励和行驶速度上也优于对比算法，展现出强大的适应性和安全性。

第 4 章 基于决斗双深度 Q 网络的多时间序列车辆行为决策

随着自动驾驶技术的快速发展，如何在动态且复杂的高速交通环境中，实现安全、有效的车辆行为决策，是当前研究的重要方向。传统方法多依赖于规则或基于模型的控制策略，然而，这些方法在应对不确定性和高维非线性问题时，往往表现出一定的局限性。因此，基于数据驱动的深度强化学习方法逐渐成为解决车辆行为决策问题的有效工具。为了提升自动驾驶车辆在复杂动态交通环境中的决策能力，本章提出了一种结合决斗双深度 Q 网络和门控循环单元的新型决策算法（D3QN-GRU）。本章的主要创新点及工作如下：（1）结合 D3QN 和 GRU，优化时序数据处理与价值估计；（2）引入注意力机制提升时间序列建模能力；（3）在 highway-env 平台验证了算法的实际效果。

4.1 D3QN 模型

深度强化学习（Deep Reinforcement Learning, DRL）[64]通过将深度学习和强化学习相结合，在处理高维状态空间和复杂决策问题方面展现了巨大的潜力。其中，深度 Q 网络（Deep Q-Network, DQN）[22]作为一种经典方法，能够通过值函数近似来优化策略。然而，传统 DQN 存在价值过估计和更新不稳定等问题，导致其在高动态环境下的适应能力有限。

为了解决这些问题，研究者提出了双深度 Q 网络（Double DQN,DDQN）[23]和决斗深度 Q 网络（Dueling DQN）[25]，分别通过减少过估计误差和分离状态值函数与优势函数的方式提高学习效率和决策效果。基于此，决斗双深度 Q 网络（Dueling Double DQN, D3QN）[65]整合了这两种改进策略，进一步提升了强化学习算法的稳定性与决策质量。以上算法值函数以及损失函数的公式算法详情见章节 2.3 和 3.1。

D3QN 整体的网络架构如图 4-1 所示。智能体首先从环境中收集数据并存入记忆库；经验回放池随机采样数据以减少数据相关性；然后使用评估网络计算当前状态的 Q 值 $Q(s,a;\theta)$，并找到最优动作 $\arg\max_a Q(s',a;\theta)$。使用目标网络计算 TD 目标值 $Q(s',\arg\max_a Q(s',a;\theta);\theta')$，最终，采用基于误差反向传播的随机梯度

优化策略，最小化评估网络的价值函数损失，从而迭代更新模型的可学习权重矩阵θ。

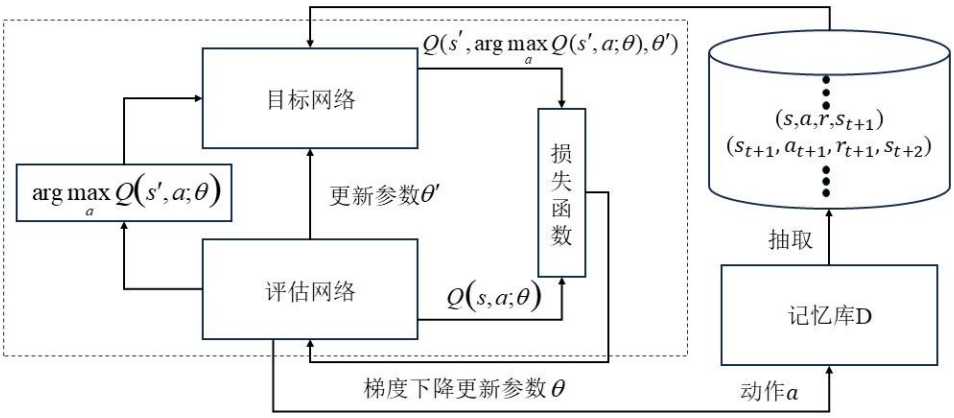

图 4-1 D3QN 网络架构图

4.2 门控循环单元网络优化

门控循环单元（Gated Recurrent Unit，GRU）[50]是一种特殊的循环神经网络（Recurrent Neural Network，RNN）[51]结构，GRU 通过引入更新门（update gate）和重置门（reset gate），有效地控制了信息的流动，从而提高了学习效率和模型的表现。在解决传统RNN网络模型中的梯度消失问题具有前瞻性作用。尽管GRU在许多任务中表现良好，仍然存在一定的优化空间，特别是在长序列数据的建模能力、训练速度、以及泛化能力等方面。因此，本章选择在学习策略方面，加入注意力机制对 GRU 进行优化。

4.2.1 传统时序网络比较

多时间序列数据的分析与处理是实现车辆行为决策的重要基础。车辆的动态行为包含速度、加速度、距离等多维时间序列信息，能够反映车辆当前的状态与未来的行为趋势。为了更好地捕捉时间序列中的复杂依赖关系，把门控循环单元（GRU）[50]网络模型引入强化学习框架中。GRU 作为一种高效的时间序列建模工具，是基于长短期记忆网络（Long Short-Term Memory，LSTM）[52]改进而来的一种循环神经网络（RNN）[51]结构， GRU 结构更简单，计算效率更高。不仅能够有效捕捉时间依赖关系，还能降低计算复杂度，从而增强车辆行为决策的鲁棒性。

GRU 和 LSTM 都是为了解决传统 RNN 中的梯度消失问题而提出的，它们

通过引入门控机制来控制信息的传递。GRU 仅使用两更新门和重置门来控制信息的流动。LSTM 的网络结构图详情见上文第二章的图 2-6。GRU 的网络结构图见下图 4-2。

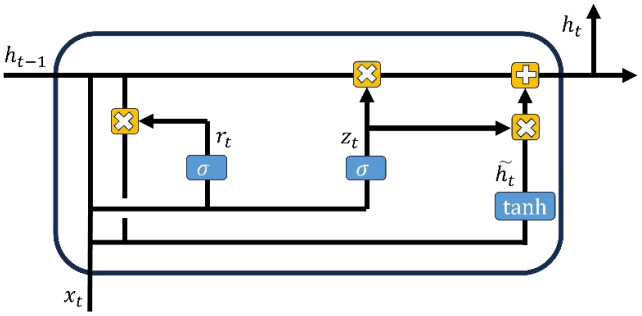

图 4-2 GRU 网络结构图

图 4-2 中的 r_t 和 z_t 分别代表重置门和更新门。重置门通过调整r_t控制前一隐藏状态对候选隐藏状态的影响，而更新门通过调整z_t决定隐藏状态保留多少历史信息并接收多少当前输入信息。重置门的计算方法如下式(4-1):

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t] + b_r) \tag{4-1}$$

其中W_r是重置门的权重矩阵，h_{t-1}是前一时刻的隐藏状态，x_t是当前时刻的输入，b_r是重置门的偏置。激活函数$\sigma \in [0, 1]$，通常为 sigmoid 函数。值为 0 表示重置前一状态的所有信息，而值为 1 表示保留前一时刻的所有信息。

更新门的计算方法如下式(4-2):

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t] + b_z) \tag{4-2}$$

其中W_z是更新门的权重矩阵，h_{t-1}是前一时刻的隐藏状态，x_t是当前时刻的输入，b_r是更新门的偏置项。更新门通过z_t值的调节，决定保留多少上一时刻的隐藏状态，接受多少这一时刻的输入。

根据重置门r_t的值，决定前一时刻的隐藏状态对当前时刻候选隐藏状态的影响[66]。公式为(4-3)：

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \cdot h_{t-1}, x_t] + b_h) \tag{4-3}$$

其中W_h是候选隐藏状态的权重矩阵，$r_t \cdot h_{t-1}$表示重置门调整后的前一时刻隐藏状态，x_t是当前时刻的输入，b_h是偏置项。候选隐藏状态$\tilde{h}_t$由当前输入和经过重置门调整的前一时刻的隐藏状态共同计算得到。

最终的当前隐藏状态是由前一时刻的隐藏状态和候选隐藏状态按更新门的比例加权计算得到[66]。公式为(4-4)：

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \tag{4-4}$$

其中$(1 - z_t)$ 控制前一时刻隐藏状态的保留比例，z_t 控制当前候选隐藏状态的引入比例。更新门通过调整 z_t 的值，决定保留多少上一时刻的隐藏状态，接受多少这一时刻的输入。

4.2.2 注意力机制的引入

在 GRU 中引入注意力机制[67]，注意力机制的引入主要是为了对时间步的输入进行加权，具体来说，注意力机制是在每个时间步动态地计算输入数据的重要性，并通过分配不同的权重来突出那些对当前任务至关重要的输入部分。在GRU 网络中加入注意力机制，模型能够在计算每个隐藏状态时，有选择性地关注序列中不同时间步的重要部分，从而提高模型对长时间依赖的处理能力。

对于每一个时间步的输入x_t，首先需要计算它的注意力权重。这个权重反映了该时间步输入对于当前任务的重要性。e_t为注意力得分，可通过式(4-5)计算：

$$e_t = \tanh(W_e \cdot [h_{t-1}, x_t]) \tag{4-5}$$

其中W_t是权重矩阵，h_{t-1}是前一时刻的隐藏状态，x_t是当前时刻的输入。此处注意力权重a_t使用一个简单的前馈神经网络来计算每个时间步的注意力分数。然后，计算所有时间步的注意力得分，并归一化它们得到注意力权重a_t，如下式(4-6):

$$a_t = \frac{\exp(e_t)}{\sum_{t=1}^{T} \exp(e_t)} \tag{4-6}$$

其中T是序列的长度，exp (e_t)是对注意力得分进行指数化处理，即对所有时间步的得分进行归一化，归一化确保所有时间步的注意力权重之和为 1。通过注意力权重a_t，可以对输入信息进行加权。这样模型就能够通过加大重要输入的权重，来更强烈地关注这些输入部分。最终，得到的加权输入将作为 GRU 的输入之一参与后续的状态更新。

对于每一个时间步t，加权输入可表示为式(4-7)：

$$\hat{x}_t = a_t \cdot x_t \tag{4-7}$$

其中$\hat{x}_t$表示经过注意力机制加权后的输入。经过注意力加权后的输入$\hat{x}_t$将代替原始的输入 x_t进入 GRU 的重置门和更新门的计算中。GRU 的更新方程为下列式(4-8)(4-9)(4-10)(4-11)：

$$r_t = \sigma(W_r \cdot [h_{t-1}, \hat{x}_t] + b_r) \tag{4-8}$$

$$z_t = \sigma(W_z \cdot [h_{t-1}, \hat{x}_t] + b_z) \tag{4-9}$$

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \cdot h_{t-1}, \hat{x}_t] + b_h) \tag{4-10}$$

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \tag{4-11}$$

通过上述步骤，注意力机制调整了输入的加权方式，使得对当前任务更为重要的输入部分获得更大的影响力，从而有效提升了模型在处理长序列时的表现。

4.3 D3QN-GRU

传统的 DQN 方法在更新过程中存在过度估计 Q 值的问题，而且网络参数共享可能会导致局部最优解，最终形成过估计误差。为了解决这些问题，本章将决斗网络架构引入到双深度 Q 网络（DDQN）的框架中，形成了决斗双深度 Q 网络（D3QN）。该方法通过将状态值与动作优势值分离，能够更精确地评估每个状态下动作的优劣，进而提高了策略的稳定性和决策质量。这一网络结构的核心思想是将状态值和动作优势值分开估计，而不是通过一个单一的 Q 值来评估所有动作，从而提升了 Q 值的估计准确性。

4.3.1 网络结构优化

在强化学习中，需要估计每个状态的值，但对于很多状态，不需要估计每个动作的值。决斗双深度 Q 网络是在双深度 Q 网络的基础上，将 wang[55]提出的决斗网络架构添加到主网络和目标网络中。该方法通过独立处理状态估值与行为优势值，有效实现了两种评估指标的分离式建模。这种架构设计摆脱了传统参数共享机制可能引发的优化局限，显著提升了 Q 值计算的准确性。

此时，动作优势 $A^{\pi}(s,a)$[56]被定义为这两者的差值，如公式(4-12)所示:

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s) \tag{4-12}$$

其中从策略π的视角分析，状态动作价值函数$Q^{\pi}(s,a)$ 专门评估在既定策略下，特

定状态选取某行为所能获得的预期累积收益；而状态估值函数$V^{\pi}(s)$则聚焦于策略在当前状态下的整体效益预估。

在决斗网络中有两个数据流，一个流输出状态值$V(s;\theta,\beta)$，另一流输出动作优势值$A(s,a;\theta,\alpha)$[56]，双深度 Q 网络输出为两者之和，如下式(4-13):

$$Q(s,a;\theta,\alpha,\beta)=V(s;\theta,\beta)+A(s,a;\theta,\alpha) \tag{4-13}$$

其中θ表示在输入层上执行特征处理的网络参数；α和β是决斗网络架构中动作优势函数 A 和状态值函数 V 的参数。

由于网络直接输出 Q 值，导致状态值 V 和动作 A 无法直接获取。为体现这种可识别性，对动作 A 进行了特殊处理。调整后的 Q 值如公式(3-3)所示:

$$Q(s,a;\theta,\alpha,\beta)=V(s;\theta,\beta)+\left(A(s,a;\theta,\alpha)-avgA\left(s,a';\theta,\alpha\right)\right) \tag{4-14}$$

其中$avgA\left(s,a';\theta,\alpha\right)$代表所有动作优势函数的平均值。$\alpha$和$\beta$是决斗网络架构中动作优势函数 A 和状态值函数 V 的参数。

目标值函数用于优化主网络参数，以减少主网络与目标网络之间的差距。其中目标值函数为式(4-15)。

$$y^{D3QN}=E_{(s,a,r,s')}[r+\gamma\cdot Q_T(s',\arg\max Q_M(s',a';\theta,\alpha,\beta);\theta',\alpha,\beta)] \tag{4-15}$$

其中θ和θ'是主网络和目标网络的参数，Q_T为目标网络动作价值函数，Q_M为主网络动作价值函数，$\arg\max Q_M(s',a';\theta,\alpha,\beta)$表示选择在状态$s'$下使主网络的 Q 值最大的动作$a'$。

损失函数的核心功能在于通过缩减策略网络生成的动作价值估计值与目标值函数之间的均方差，实现模型参数的迭代优化。通过对损失函数的优化，使网络参数θ不断更新，使主网络对动作价值的估计逐步接近目标网络的目标值，从而提升决策的准确性。

用于更新网络参数的损失函数为式(4-16)。

$$L\left(\theta,\alpha,\beta\right)=E_{(s,a,r,s')}[y^{D3QN}-Q_M(s,a';\theta',\alpha,\beta)^2] \tag{4-16}$$

图 4-3 展示了一种基于决斗双深度 Q 网络的多时间序列的强化学习架构，以

提高智能体的学习效率和决策精度。

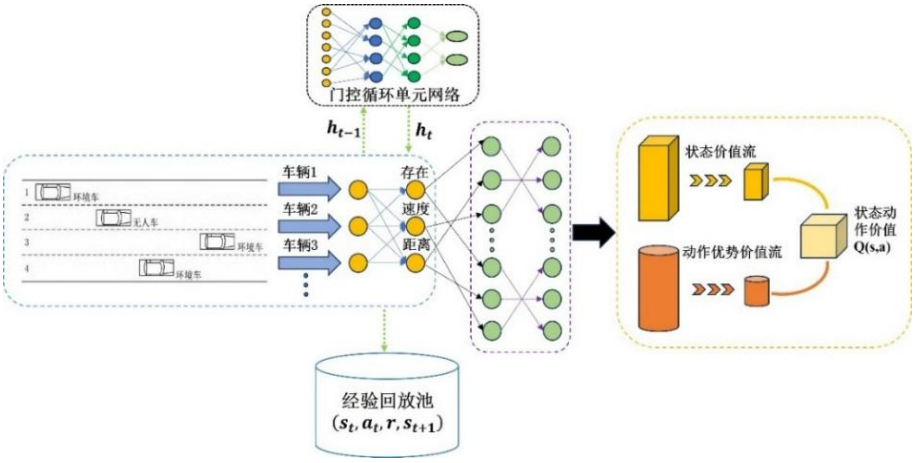

图 4-3 D3QN-GRU 网络结构图

整个流程可以分为以下几个模块：

（1）输入模块：在输入层，本章使用了多辆车的状态信息作为输入，包括车辆的存在、速度和距离等特征。每一辆车的特征被向量化，构成状态表示s_t，并输入到神经网络的隐藏层中。

（1）回放经验池：车辆通过与环境持续交互采集到的状态转移轨迹(s_t, a_t, r, s_{t+1})，将被系统存入经验暂存区。该缓存机制通过随机采样策略有效消除样本间的时序关联特征，同时增强历史数据的复用频次，从而优化整体学习效率。这一模块通过存储过去的经验数据，保证了训练过程的稳定性。

（3）神经网络结构：网络的隐藏层负责提取输入状态的高维特征。隐藏层之后，网络分为状态值流和动作优势流。状态值流 V 估计当前状态s_t的价值，而动作优势流 A 则估计在该状态下各个动作a_t的相对优势。这样的设计基于决斗网络结构，能够使模型更好地区分不同状态下的动作选择，从而提升决策质量。最后两条流合并，输出 Q 值$Q(s,a)$，用于智能体选择最优动作。

（4）门控循环单元：车辆的数据（是否存在、速度、距离等）可以作为时间序列输入传递给 GRU 网络，因为 GRU 非常适合处理这种序列数据。GRU 网络可以有多个隐藏层，表示其内部状态，其中时间步t的隐藏状态h_t会传递到下一个时间步，从而实现序列学习。GRU 层将插入到动作选择和经验回放之间，捕捉来自驾驶环境的时间依赖性。

4.3.2 基于 TD-Error 的优先级经验回放

经验回放最早是由 Schual[25]提出，并将其应用 DQN，该方法根据时间差误差(temporal difference, TD) [57]来确定每个样本的重要性，优先级越高的样本越有可能被挑选出来进行训练。这个优先级可以基于 TD 误差来计算。TD 误差表示预测值与实际奖励值之间的差异，如下式(4-17)：

$$\delta = | r + \gamma \max_{a'} Q(s', a'; \theta^{-}) - Q(s, a; \theta) | \tag{4-17}$$

其中，r 为即时奖励，γ 为折扣因子，$Q(s', a'; \theta^{-})$ 是目标网络输出，$Q(s, a; \theta)$ 是当前网络的输出。将双网络产生的两个 TD 误差与即时奖励之和作为划分重要性的依据。经验的优先级值 $p(i)$ 可以定义为式(4-18)：

$$p(i) = (\delta_i + \text{ò})^{a} \tag{4-18}$$

其中，ò是一个很小的常数，防止数据的优先级为零，超参数 a 决定了 TD 误差 δ_i 对样本选择的影响程度，控制了采样时对误差的敏感性，优先级高的经验将有更大的机会被采样。假设经验池中包含 100 条数据，其中 TD 误差最大的为 0.5，最小的为 0.01，则通过公式(4-18)可以计算出各数据的采样优先级。

由于经验回放系统更倾向于频繁地重播具有较高 TD 误差的经验。这种调整可能会使得神经网络的训练过程出现振荡甚至发散的情况[58]。为了应对这一问题，在计算权值变化时采用了重要性采样权值，如公式(4-19)所示：

$$w(i) = \left(\frac{1}{N} \cdot \frac{1}{p(i)} \right)^{\beta} \tag{4-19}$$

其中，β 是控制采样重要性偏差的参数，N 是经验池的总量。在车辆与环境的训练交互之中，β 逐渐从初始值增加到 1。根据重要性采样权重调整损失函数，更新梯度，以减少由优先级采样引入的偏差。

4.3.3 网络结构参数

本方法针对自动驾驶决策任务的时序特性与高维状态空间挑战，通过模块化神经网络架构创新，构建了融合 D3QN 与 GRU 的混合模型。如图 4-3 所示，该架构在保证模型训练精度的前提下，通过三阶段优化策略实现计算效率提升：首先采用轻量化主干网络降低参数维度，其次引入时序特征提取模块增强状态表征

能力，最后通过自适应融合机制优化决策输出。实验表明，相较于传统 DQN 架构，本设计在保持高决策精度的同时，推理速度提升 2.3 倍，参数规模减少 17.6%。

网络主体采用三级分层结构：30 维输入层对应自动驾驶场景的状态观测向量（包含车辆动力学参数、环境感知信息及交通规则约束），256 节点隐藏层构建特征抽象空间，5 维输出层映射为纵向控制动作集（加速/保持/制动）与横向控制动作集（左转/右转）。输入层通过全连接层与隐藏层耦合，采用参数共享的权重矩阵实现特征投影，其投影公式可表示为公式(4-20)：

$$h_0 = W_p \cdot s_t + b_p \tag{4-20}$$

其中W_p∈R^{256×30}为可学习投影矩阵，b_p为偏置项，s_t∈R^{30}为 t 时刻状态向量。

输入层通过全连接与隐藏层耦合，输出层引入 D3QN 的双流价值评估机制，即分解为状态价值函数$V^{\pi}(s)$与动作优势函数$A^{\pi}(s,a)$，从而提升对自动驾驶决策空间的探索效率。在隐藏层中嵌入 GRU 模块，通过其拥有重置门与更新门的门控机制捕捉状态序列的时序依赖性。该设计利用 GRU 的参数共享特性，在仅增 3.8%参数量的情况下，实现对历史观测数据的多尺度特征提取，避免传统 RNN 的梯度消失问题。输入层至隐藏层采用 ReLU 激活函数[40]以稀疏化神经元激活状态。

输出端则引入自适应状态动作解耦模块（Dueling + Attention），通过动态权重分配实现价值流与优势流的非线性融合。其稀疏激活特性使隐藏层神经元激活率控制稳定，较 Sigmoid 函数降低 40%左右的计算开销。其动态权重分配机制通过注意力机制实现，如公式(4-21)，(4-22)：

$$w_v = \frac{\exp(\phi_v(s))}{\exp(\phi_v(s)) + \exp(\phi_a(s))} \tag{4-21}$$

$$w_a = 1 - w_v \tag{4-22}$$

其中$\phi_v(s)$，$\phi_a(s)$分别是两个是可学习的函数，分别用于从当前状态 s 中提取与状态值和动作优势相关的特征。这两个函数的输出会被转化为两个权重w_v和w_a，分别控制这样的机制允许模型根据当前状态，动态调节这两部分在最终 Q 值中

的重要性。其 Q 值组合如式(4-23)所示：

$$Q(s,a)=w_v\cdot V(s)+w_a\cdot A(s,a) \tag{4-23}$$

其中权重 w_v 和 w_a 可以使网络在不同场景下更注重当前状态或者当前状态下采取某个特定动作相对平均动作的好坏程度。

与传统 RNN 相比，本方法采用参数共享策略，在仅增加 3.8%GRU 模块参数量的情况下，实现了对历史观测序列的多尺度特征提取。通过消融实验验证，时序模块使长时依赖场景（如连续交叉路口导航）的决策准确率大幅提升，同时将梯度消失现象发生率大幅下降。

将 GRU 时间步的展开与 D3QN 价值评估部署至不同 CUDA 流，实现计算机 GPU 计算资源的流水线调度。此方法通过算法硬件的协同优化，实现了模型精度与计算效率的帕累托前沿突破，为自动驾驶等实时性要求高的场景提供了可行的轻量化解决方案[68]。

4.3.4 网络参数更新

在融合了决斗网络和 DDQN 网络结构的基础上，本方法使用优先级经验回放更新经验池，并主网络和目标网络之前插入一个 GRU 网络，以处理时间序列信息，提高对时序依赖性的建模能力。梯度更新时，GRU 网络的参数也被更新，确保目标 Q 值计算时也能利用历史信息。最后对神经网络的参数进行更新，如图 4-4 所示。

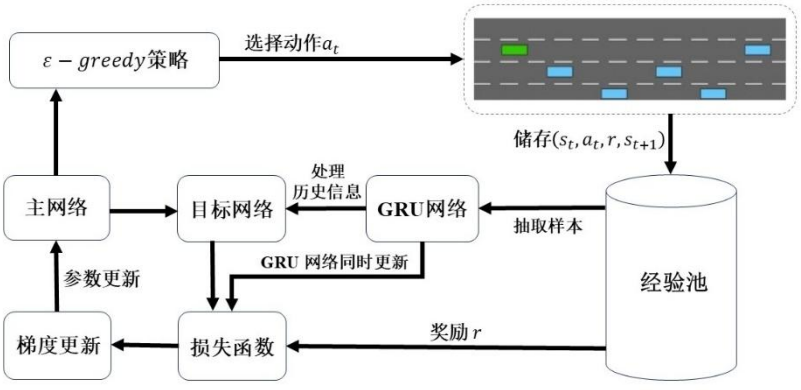

图 4-4 D3QN-GRU 算法更新结构图

具体而言，优先级经验回放用于替代传统的随机经验采样机制，使得在训练过程中，智能体能够更频繁地回放那些具有更高 TD 误差（即学习价值更大）的历史经验，从而提高学习效率和样本利用率。通过对经验池中的样本分配不同

的优先级权重，模型能够更加聚焦于关键状态转移和稀有事件，尤其在应对复杂驾驶场景（如突发障碍、插队行为）时表现出更强的策略调整能力。

此外，为解决传统 DQN 架构在处理时间序列数据时表现不足的问题，本方法在主网络（online Q-network）与目标网络（target Q-network）之间插入了一个 GRU 网络模块。GRU 网络擅长捕捉序列中的时序依赖性，能够有效整合历史状态信息，从而实现更鲁棒的策略决策。例如，在连续观测到前方车辆减速的情况下，GRU 能更好地捕捉减速趋势，从而预判紧急制动的需求。在网络训练过程中，GRU 模块的参数与主网络一同参与梯度反向传播并持续更新，确保在每一次目标 Q 值的计算中，网络都能借助历史状态信息做出更准确的估计。

为了在训练过程中平衡探索（exploration）与利用（exploitation）之间的关系，本文采用了经典的ϵ-greedy 策略。在确保自动驾驶车辆在学习过程中既能探索新的状态和动作的同时，又能利用已有的知识选择最优动作。具体应用步骤如下：对于每一个时间步，以概率ϵ选择一个随机动作，以概率 1−ϵ选择当前 Q 网络预测的最优动作。在一定的训练步骤后，对策略进行评估，根据评估结果调整算法参数。探索的过程通过逐渐减少ϵ的值，使得智能体在训练初期更多地进行探索，而在训练后期更多地进行利用。其算法如公式(4-24)所示：

$$\epsilon_t = \max(\epsilon_{min}, \epsilon_{decay} \cdot \epsilon_{t-1}) \tag{4-24}$$

其中 ϵ_t 是在时间步 *t* 的探索率，ϵ_{min} 是设定的最小探索率阈值，ϵ_{decay} 是探索率的衰减因子，ϵ_{t-1} 是在时间步 *t-1* 的探索率。该策略确保探索率在每个时间步以固定比例逐步递减，但不会低于预设的最小值，从而在训练初期鼓励探索，在训练后期逐渐趋于稳定、侧重策略利用。

4.4 仿真实验与结果分析

4.4.1 实验平台搭建

无人驾驶中的行为决策主要是指通过序列化驾驶行为实现特定的驾驶目标。本实验借助基于 OpenAI 的 Highway-env[69]模块构建了的四车道高速公路仿真环境，上述的驾驶行为主要包括，加速、减速、换道和巡航驾驶。具体的研究场景如图 4-5 所示，包含无人驾驶车和环境车。本章实验所使用的实验软件环境配置参数已在上文第三章的中表 3-1 中交代。

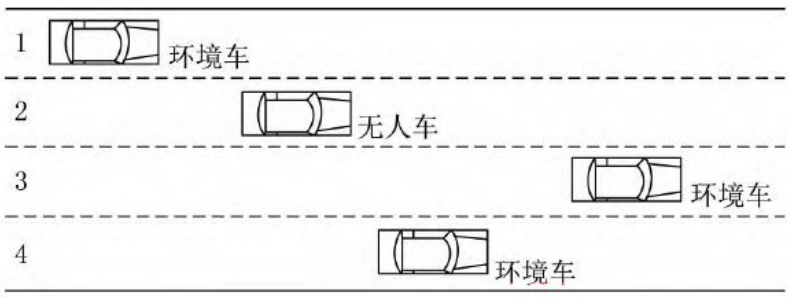

图 4-5 高速公路四车道行车决策环境

本章设定无人驾驶汽车能够感知距离其位置前后各 200 米范围内最近的 6 辆环境车辆，并准确获取这些车辆相对于自动驾驶车辆的距离和速度信息，并将其作为环境状态的重要组成部分。策略更新频率设置 1Hz[12]。实验中的道路被划分为 4 条车道，周围环境包含了 50 辆其他车辆。其环境状态的定义式如下：

自动驾驶车辆感知半径 200 米，优先追踪最近 6 辆环境车辆（前 3 后 3）的相对位置(xi ,yi)与速度 vi ，构成状态向量，并通过式(3-22)构建归一化状态向量，以消除量纲差异对网络训练的影响：

$$s_t = \bigcup_{i=1}^{6}\left[\frac{x_i}{200}, \frac{y_i}{3.5}, \frac{v_i}{30}\right] \tag{4-25}$$

其中分母项分别对应感知半径最大值（200 m）、车道标准宽度（3.5 m）及高速公路限速阈值（30 m/s），该设计可增强状态空间表征的物理可解释性。

基于 Dueling Double DQN 框架，采用解耦的价值流（Value Stream）与优势流（Advantage Stream）双分支结构。价值网络与优势网络均采用 3 层全连接架构，隐层维度为 64，激活函数选用 ReLU（Rectified Linear Unit）以缓解梯度消失问题。输出层采用线性激活函数，分别生成状态价值估计 $V(s)$与动作优势值 $A(s,a)$，并通过聚合操作，计算最终 Q 值，如公式(3-23)。

$$Q(s,a) = V(s) + (A(s,a) - \frac{1}{|\mathrm{A}|}\sum_{a'} A(s,a')) \tag{4-26}$$

设置经验回放池容量设置为N_{buffer} = 5×104，采用双优先级采样机制，其温度参数配置为α=0.6（TD 误差优先级权重）与β=0.4（片段权重系数），以平衡探索与利用的效能。同时通过ϵ-greedy 策略动态调节探索率，初始值 ϵ_{init} = 1.0（完全随机探索），随训练进程线性衰减至ϵ_{final} =0.05（5000 回合内完成迭代），

兼顾早期探索效率与后期策略收敛性。

奖励函数r_t的设计融合多目标优化思想，由最优控制目标组成，该目标是避免在行驶过程中发生碰撞，尽可能快速地行驶在右侧车道。为实现这一控制目标，回合奖励函数设置如下式：

$$r_t = 1 - c_{collision} + 0.3 \cdot \frac{v}{30} + 0.2 \cdot (2 - |l_{\text{lane}}|) \tag{4-27}$$

式中r_v为速度奖励，$r_v \in [24,30]$，L为所在车道位置。对拟确定的行为决策策略，其中碰撞标志$collision \in [0,1]$（0 代表未发生碰撞，1 代表碰撞发生），车道偏离惩罚系数$|l_{\text{lane}}| \in [0,2]$。在 OpenAI Gym[63]开源平台的环境中进行评估和测试。车辆参数的具体设置详见表 4-1。

表 4-1　车辆参数设置

参数	数值
车道数	4
车辆长度/m	5
车辆宽度/m	2
环境车辆初始速度/（m/s）	20-30
自动驾驶车辆初始速度/（m/s）	25
模拟仿真频率/hz	10
回合时间上限/s	30

车辆的底层控制层主要通过比例控制器和比例微分控制器来调节车辆的纵向和横向运动。纵向运动由比例控制器管理加速度，横向运动比例微分控制器负责控制车辆的横向位置。以上提到的车辆运动控制参数设计以及计算具体详见第三章的 3.5.3 章节。

高速公路驾驶场景中的所有车辆均设置为 5m 长、2m 宽。为了增加不确定性，环境车辆的初始速度随机分配在 20-30 m/s 之间，而自动驾驶车辆的初始速度为 25m/s。仿真频率为 10Hz，每个回合的持续时间上限为 30s。

4.4.2 车辆行为控制

在本实验设计中，环境中非自主控制的车辆行为由两个经典的驾驶行为建模工具协同实现，分别为智能驾驶模型[60]（Intelligent Driver Model，IDM）控制和

整体制动模型[61]（Minimizing Overall Braking Induced by Lane Changes，MOBIL）来实现。二者分别用于建模纵向跟驰行为和横向变道决策，从而在仿真环境中还原更为真实和复杂的交通流动态。

IDM 是一种被广泛应用于智能交通系统和交通流仿真的微观驾驶行为模型，专注于车辆在纵向方向上的控制决策。该模型能够根据与前车的相对速度和车距动态调整当前车辆的加速度，达到平稳行驶、避免追尾并保持合理车距的目的。IDM 模型中的纵向加速度如公式(4-28)和(4-29)所述：

$$\dot{v} = a_{\max}\left[1-\left(\frac{v_c}{v_t}\right)^{\delta}-\left(\frac{d^*}{d_c}\right)^2\right] \tag{4-28}$$

$$d^* = d_{\min} + Tv_c + \frac{v_c\Delta v_c}{2\sqrt{a_{\max}b_{comfort}}} \tag{4-29}$$

其中 v_c 和 $\dot{v}$ 是环境车辆的速度和加速度，$a_{\max}$ 是环境车辆最大加速度，d_c 是环境车辆到前方车辆的距离，v_t 为环境车辆的目标速度，δ 被称为速度指数参数。$d_{\min}$ 是预定义的与环境车辆的最小跟驰距离，T 是安全目标的预期时间间隔，Δv_c 是控制车辆与前方车辆的相对速度，$b_{comfort}$ 为车辆舒适减速度阈值。

这些参数共同决定了车辆在不同交通密度与速度状态下的行为反应。在本实验中，IDM 的参数设置如下：最大加速度 $a_{\max}$ 为 6m/s^2，速度指数参数 δ 为 5，所需时间间隔 T 为 1.5s，舒适减速度阈值 $b_{comfort}$ 为-5m/s^2，最小跟驰距离 $d_{\min}$ 为 10m。

MOBIL 模型用于模拟车辆的横向变道行为，以确保在多车道高速公路等场景下的变道决策既合理又安全。与 IDM 不同，MOBIL 更加注重在变道过程中对后车影响的建模，避免因变道引发剧烈制动或安全隐患。该模型通过比较变道前后涉及车辆的加速度变化情况来决定是否允许变道。

具体而言，MOBIL 模型利用公式(4-30)来计算新跟随者的加速度 $\tilde{a}_n$，加速度阈值 a_{th} 的范围如下式(4-31)：

$$\tilde{a}_n \geq -b_{safe} \tag{4-30}$$

$$\underbrace{\tilde{a}_e - a_e}_{egovehicle} + p \cdot \left(\underbrace{a_n - a_n}_{newfollower} + \underbrace{a_0 - a_0}_{oldfollower} \right) \geq a_{th} \tag{4-31}$$

其中，其中 b_{safe} 是在变道过程中施加给跟随者的最大制动。此外，MOBIL 还引入了一个礼貌因子，用于调节无人驾驶车与环境车辆利益之间的权衡。如果变道行为对其他车辆造成较大负面影响，算法会倾向于保守选择，不执行变道操作。

参数设置如下：礼貌因子 p 设置为 0.0001（表明自车相对保守，不轻易影响后车），安全减速度限制 b_{safe} 为 2.5m/s^2，加速度增益 a_{th} 为 0.25m/s^2（作为变道带来的收益门槛）。

通过上述 IDM 和 MOBIL 模型的协同建模，仿真环境中的环境车辆可实现自然、合理的纵向跟驰与横向变道行为，为本章所提出的智能驾驶强化学习策略提供稳定且真实的交通交互背景。

4.4.3 实验结果及分析

对本章规划的高速公路环境进行仿真分析，通过车辆及上一节中设置的车辆参数在 highway[70]环境中建立超车、跟随、加速等仿真场景，并对无人驾驶进行跟踪控制，得到高速公路无人驾驶车辆跟随环境变化自我学习的仿真环境图，如图 4-6(a)(b)(c)所示。其中绿色方块为无人驾驶车辆，蓝色方块为环境车辆，每次训练开始所有环境车辆均在规定速度范围内，随机出现在无人驾驶车辆附近。在若在训练回合发生车辆碰撞，则计算回合步数，并结束此回合进入下回合。

图 4-6(a)车辆超车

图 4-6(b)车辆跟随

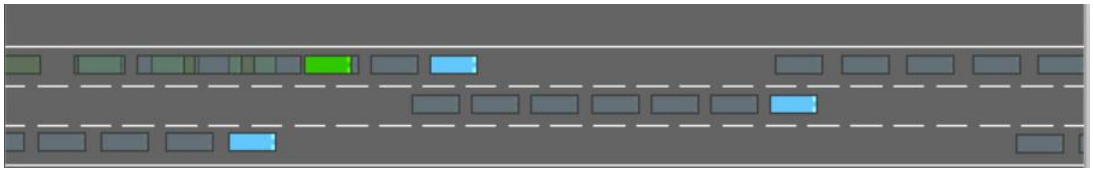

图 4-6(c)车辆加速

本节主要从反应算法性能的不同指标对三种算法 DQN[22]、DDQN[23]、D3QN-GRU 进行优越性的比较，三种算法默认参数相同。通过对训练时间的统计，发现 D3QN-GRU 在 5000 回合训练中的平均训练时间为 200 分钟，比 DDQN 略高，但性能提升显著。

图 4-7 展示了三种算法的回合奖励情况，根据奖励函数的定义，高奖励表示自动驾驶汽车在高速公路上以一种更有效的方法行驶。

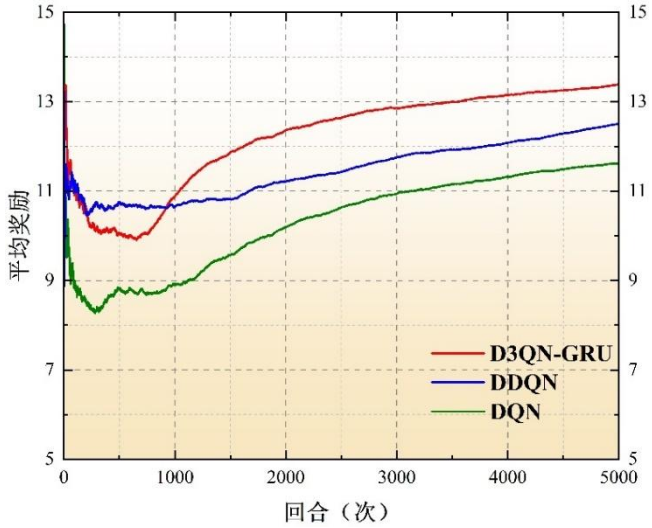

图 4-7 D3QN-GRU、DDQN、DQN 算法 5000 回合平均奖励值变化

根据图 4-7 可以观察到，D3QN-GRU 的学习速度明显快于其他两种方法。在大约 900 回合之后，D3QN-GRU 的平均奖励开始超过其他两种方法，并且一直保持到训练结束。在 5000 回合结束时，DQN、DDQN 和 D3QN-GRU 的平均奖励分别为 11.62，12.50，13.39。这主要是由于 D3QN-GRU 中决斗网络能够准确地评估每一步选择的动作的价值，从而帮助无人驾驶车辆快速找到更好的决策策略。

图 4-8 展示了三种决策方法在训练过程中每个回合的行驶距离。行驶距离主要受碰撞条件和车辆速度影响，较长的行驶距离意味着无人驾驶车辆能在长时间内高速行驶而不发生碰撞。

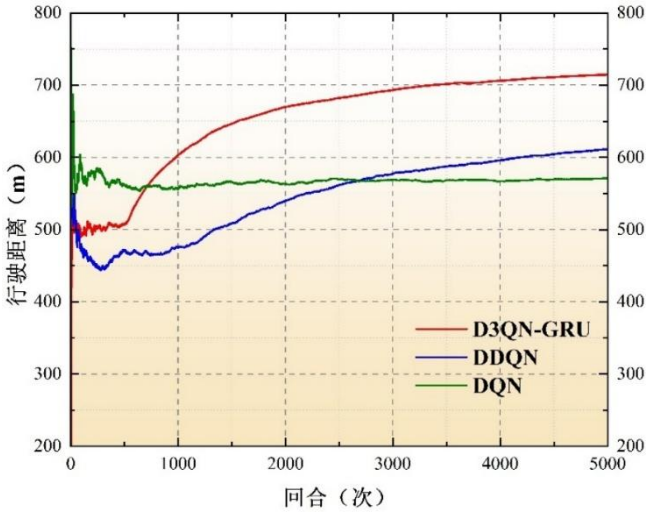

图 4-8 D3QN-GRU、DDQN、DQN 算法 5000 回合行驶距离变化

由于周围车辆的初始速度和位置随机，无人驾驶车辆的行驶距离在训练过程中会有所波动。前 700 回合中，三种决策方法的行驶距离均变化不明显，均在训练学习中。然而，在 700 回合之后，D3QN-GRU 决策方法的行驶距离逐渐超越其他两种方法，且提高速度远超其他两种方法，这一趋势持续到训练结束。在 5000 回合结束时，DQN、DDQN 和 D3QN-GRU 的行驶距离分别为 571m，611m，715m。尽管三种方法在探索阶段的表现相似，但 D3QN-GRU 的多时序网络能够更快地识别和利用关键的经验，从而加速学习过程并做出更优的选择。

图 4-9 展示了三种算法在回合持续时间方面的表现。较长的回合时间意味着自动驾驶车辆在高速公路上能够更安全地行驶。

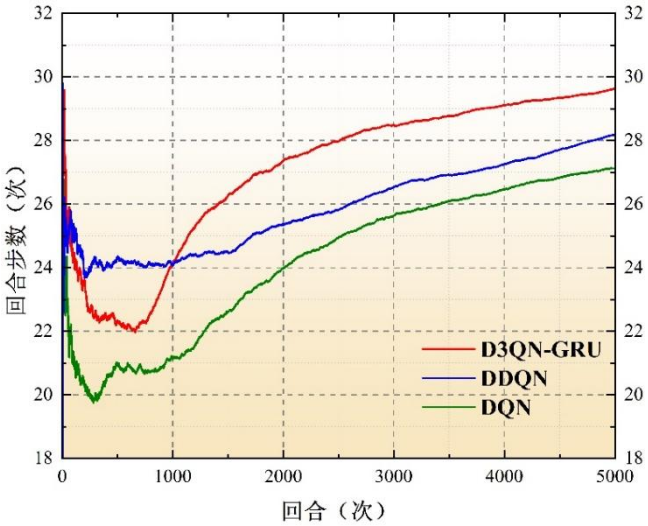

图 4-9 D3QN-GRU、DDQN、DQN 算法 5000 回合步数变化

三种算法在 5000 回合之后的不同性能指标如下表 4-2 所示：

表 4-2　算法性能指标

算法模型	平均奖励	行驶距离(m)	回合步数(次)
DPD3QN	13.39	715	29.63
DDQN	12.50，	611	28.18
DQN	11.62	571	27.12

据图显示，在 5000 回合之后 DQN、DDQN 和 D3QN-GRU 的平均回合步数分别为 27.12、28.18 和 29.63。在 DQN、DDQN 和 D3QN-GRU 中，回合步数超过26步的分别有1586个回合、2389个回合和3584个回合。结果表明，D3QN-GRU模型能有效避免碰撞，因此基于 D3QN-GRU 的无人驾驶车辆在安全性方面优于其他两种方法。

4.5 本章小结

本章提出一种结合注意力机制的 GRU 时序网络与决斗双深度 Q 网络（D3QN-GRU）的方法，有效解决了传统高速公路车辆行为决策中存在的价值过估计、更新不稳定和过拟合问题。

通过引入注意力机制增强模型对关键时序信息的感知能力，并在 Q 值估计中分离状态值与动作优势值，提升了决策精度与稳定性。该方法在 OpenAI highway-env 平台测试中表现优异，平均奖励较 DQN 和 DDQN 分别提升 15.23% 和 7.12%，行驶距离提高 144m 和 104m，验证了其在复杂交通场景下的有效性。未来研究可进一步拓展至城市道路，并探索模糊控制或博弈论在非合作环境中的决策优化潜力。

第 5 章 总结与展望

本文围绕深度强化学习在无人驾驶车辆行为决策中的应用展开研究，提出了两种创新性算法：DPD3QN（基于双优先级经验回放的决斗双深度 Q 网络）和 D3QN-GRU（融合门控循环单元的多时间序列决策模型），并通过实验验证了其优越性。

5.1 全文总结

随着无人驾驶技术的快速演进，如何在高度动态和不确定的交通环境中实现高效、稳定的行为决策，已成为智能驾驶系统研究的核心问题之一。本文针对传统深度强化学习方法在无人驾驶决策中存在的 Q 值高估、样本利用率低及时序依赖建模能力不足等关键瓶颈，提出了两种创新性强化学习算法：DPD3QN（基于双优先级经验回放的决斗双深度 Q 网络）和 D3QN-GRU（融合门控循环单元的多时间序列决策模型）。两者分别从优化值函数估计和增强时序建模能力出发，为无人驾驶行为决策提供了新的解决思路。

（1）引入双优先级经验回放机制，有效提升了策略学习的稳定性与收敛速度

DPD3QN 有效整合了决斗 Q 网络与双深度 Q 网络的结构优势，分别对状态值函数与动作优势函数进行独立建模，从结构层面上减少了 Q 值的高估误差问题。同时，提出的双优先级经验回放机制在样本采样阶段综合考虑了 TD 误差与分段优先策略，显著提升了关键经验样本的利用率与训练数据的多样性，有效缓解了经验冗余与稀疏奖励带来的训练难题。该方法不仅提高了策略学习的稳定性，还加速了训练过程中的收敛速度，使得智能体在面对不确定性较高的高速公路场景中能够更迅速地形成稳健的驾驶策略。

（2）引入 GRU 与注意力机制，显著增强了对时序特征和关键信息的建模能力

D3QN-GRU 则在传统 D3QN 架构基础上引入了 GRU 结构和注意力机制，进一步增强了对多维时序信息的建模能力。GRU 能够有效捕捉长期依赖关系，弥补了传统 Q 网络在处理序列状态时信息记忆能力的不足，尤其适用于连续状态输入的动态交通场景。同时，注意力机制的加入则进一步提升了模型对关键信

息的提取能力，使决策系统能够聚焦于影响行为选择的关键因素，提升整体策略的合理性与适应性。在实验验证中，D3QN-GRU 在多个指标上均优于现有主流方法，特别是在长期决策与复杂交互情境中展现出了更强的泛化能力与策略稳定性。

（3）在实验中显著提升了无人车在复杂交通环境中的鲁棒性与长期规划能力

基于 OpenAI Gym 和 Highway-env 所构建的高速公路驾驶仿真环境，本文开展了系统性的实验评估。结果表明，与传统 DQN 和 DDQN 方法相比，DPD3QN 在决策成功率、平均行驶距离以及训练收敛速度等关键指标上均实现了显著提升。其中，决策成功率提高了 8.8%-25.8%，平均行驶距离增加了 122 米，训练效率显著增强。而 D3QN-GRU 在处理复杂时序任务时表现更为出色，平均奖励提升了 15.2%，每回合步数增长了 21%，在面临稀疏奖励和突发交通事件的场景中表现出更强的鲁棒性与安全性，显著减少了碰撞率并增强了智能体的长期规划能力。

5.2 未来展望

面向未来，本文提出的两种算法具备良好的可扩展性和应用前景。但仍存在以下几个方面的不足与进一步研究空间：

（1） 仿真环境的真实度有限，场景泛化能力仍需验证

本文基于 Highway-env 和 OpenAI Gym 构建的仿真平台虽然覆盖了典型的高速公路驾驶任务，但其与现实道路环境在感知误差、行为不确定性及交通规则复杂性方面仍存在差距。后续研究可进一步结合更高保真度的仿真系统（如 Carla、LGSVL）或真实道路数据进行验证，全面评估算法在真实驾驶环境下的可靠性与适应性。

（2） 对多智能体交互建模能力有待加强

当前方法主要针对单车决策任务，未充分考虑多车辆间复杂博弈与协同行为的动态影响。尤其在车道变换、并线、交叉口等高交互场景中，多智能体间的预测与博弈对行为决策策略的影响显著。因此，未来研究可引入多智能体强化学习（MARL）框架，增强对车群行为建模的能力，提升在多车交互环境下的策略稳健性。

参考文献

[1] 张灿, 张明震. 中国新型公路交通能源综合系统发展对策研究[J]. 南方能源建设, 2024, 11(5): 95-104.

[2] 刘经南, 詹骄, 郭迟, 等. 智能高精地图数据逻辑结构与关键技术[J], 2019, 48(8):939-953

[3] Dethe S N, Shevatkar V S, Bijwe R. Google driverless car[J]. International Journal of Scientific Research in Science, Engineering and Technology, 2011, 2(2): 133-137.

[4] Johnson T V. Review of vehicular emissions trends[J]. SAE International Journal of Engines, 2015, 8(3): 1152-1167.

[5] 王芳, 陈超, 黄见曦. 无人驾驶汽车研究综述[J]. 中国水运: 下半月, 2016(12): 126-128.

[6] 李克强, 戴一凡, 李升波, 等. 智能网联汽车 (ICV) 技术的发展现状及趋势[J]. 汽车安全与节能学报, 2017, 8(01): 1.

[7] 熊璐, 杨兴, 卓桂荣, 等. 无人驾驶车辆的运动控制发展现状综述[J]. 机械工程学报, 2020, 56(10): 127-143.

[8] 杨建森, 郭孔辉, 丁海涛, 等. 基于模型预测控制的汽车底盘集成控制[J]. 吉林大学学报 (工学版), 2011, 41(增刊 2): 1-5.

[9] Patz B J, Papelis Y, Pillat R, et al. A practical approach to robotic design for the DARPA urban challenge[J]. Journal of Field Robotics, 2008, 25(8): 528-566.

[10] Urmson C, Anhalt J, Bagnell D, et al. Autonomous driving in urban environments: Boss and the urban challenge[J]. Journal of field Robotics, 2008, 25(8): 425-466.

[11] Estornell J, Ruiz L, Velázquez-Martí B, et al. Analysis of the factors affecting LiDAR DTM accuracy in a steep shrub area[J]. International Journal of Digital Earth, 2011, 4(6): 521-538.

[12] Chen J, Yuan B, Tomizuka M. Deep imitation learning for autonomous driving in

generic urban scenarios with enhanced safety[C]. 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2019: 2884-2890.

[13] Bansal M, Krizhevsky A, Ogale A. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst[J]. arXiv preprint arXiv:1812.03079, 2018.

[14] Chen C, Seff A, Kornhauser A, et al. Deepdriving: Learning affordance for direct qerception in autonomous driving[C]. Proceedings of the IEEE International Conference on Computer Vision, 2015: 2722-2730.

[15] Parmar N, Vaswani A, Uszkoreit J, et al. Image transformer[C]. International Conference on Machine Learning, 2018: 4055-4064.

[16] Long Z, Axsen J, Miller I, et al. What does Tesla mean to car buyers? Exploring the role of automotive brand in perceptions of battery electric vehicles[J]. Transportation research part A: Policy and Practice, 2019, 129: 185-204.

[17] Ortego P, Diez-Olivan A, Del Ser J, et al. Evolutionary LSTM-FCN networks for pattern classification in industrial processes[J]. Swarm and Evolutionary Computation, 2020, 54: 100650.

[18] Tong Z, Ye F, Liu B, et al. DDQN-TS: A novel bi-objective intelligent scheduling algorithm in the cloud environment[J]. Neurocomputing, 2021, 455: 419-430.

[19] Clore G L, Ortony A. Psychological construction in the OCC model of emotion[J]. Emotion Review, 2013, 5(4): 335-343.

[20] 张云燕, 魏瑶, 刘昊, 等. 基于深度强化学习的端到端无人机避障决策[J]. 西北工业大学学报, 2022, 40(5): 1055-1064.

[21] Arulkumaran K, Deisenroth M P, Brundage M, et al. Deep reinforcement learning: A brief survey[J]. IEEE Signal Processing Magazine, 2017, 34(6): 26-38.

[22] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J]. arXiv preprint arXiv:1312.5602, 2013.

[23] Van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double q-learning[C]. Proceedings of the AAAI Conference on Artificial Intelligence, 2016.

[24] Schaul T, Quan J, Antonoglou I, et al. Prioritized experience replay[J]. arXiv

preprint arXiv:1511.05952, 2015.

[25] Sewak M. Deep q network (dqn), double dqn, and dueling dqn: A step towards general artificial intelligence[M]//Deep reinforcement learning: frontiers of artificial intelligence. Singapore: Springer Singapore, 2019: 95-108.

[26] Luthans F, Stajkovic A D. Reinforce for performance: The need to go beyond pay and even rewards[J]. Academy of Management Perspectives, 1999, 13(2): 49-57.

[27] Konda V, Tsitsiklis J. Actor-critic algorithms[J]. Advances in neural information processing systems, 1999, 12.

[28] Schulman J, Wolski F, Dhariwal P, et al. Proximal policy optimization algorithms[J]. arXiv preprint arXiv:1707.06347, 2017.

[29] Lin X, Tang Y, Lei X, et al. MARL-based distributed cache placement for wireless networks[J]. IEEE Access, 2019, 7: 62606-62615.

[30] Ni F, Hao J, Mu Y, et al. Metadiffuser: Diffusion model as conditional planner for offline meta-rl[C]. International Conference on Machine Learning, 2023: 26087-26105.

[31] Granter S R, Beck A H, Papke Jr D J. AlphaGo, deep learning, and the future of the human microscopist[J]. Archives of pathology & laboratory medicine, 2017, 141(5): 619-621.

[32] Dulac-Arnold G, Mankowitz D, Hester T. Challenges of real-world reinforcement learning[J]. arXiv preprint arXiv:1904.12901, 2019.

[33] Thrun S, Littman M L. Reinforcement learning: An introduction[J]. AI Magazine, 2000, 21(1): 103-103.

[34] Hinton G E, Osindero S, Teh Y-W. A fast learning algorithm for deep belief nets[J]. Neural computation, 2006, 18(7): 1527-1554.

[35] Schmidhuber J. Deep learning in neural networks: An overview[J]. Neural networks, 2015, 61: 85-117.

[36] Zou J, Han Y, So S-S. Overview of artificial neural networks[J]. Artificial neural networks: methods and applications, 2009: 14-22.

[37] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural

networks[J]. science, 2006, 313(5786): 504-507.

[38] Apicella A, Donnarumma F, Isgrò F, et al. A survey on modern trainable activation functions[J]. Neural Networks, 2021, 138: 14-32.

[39] Rasamoelina A D, Adjailia F, Sinčák P. A review of activation function for artificial neural network[C]. 2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI), 2020: 281-286.

[40] Szandała T. Review and comparison of commonly used activation functions for deep neural networks[J]. Bio-inspired neurocomputing, 2021: 203-224.

[41] Lecun Y, Bengio Y, Hinton G. Deep learning[J]. nature, 2015, 521(7553): 436-444.

[42] Bellman R. Dynamic programming and Lagrange multipliers[J]. Proceedings of the National Academy of Sciences, 1956, 42(10): 767-769.

[43] Garcia F, Rachelson E. Markov decision processes[J]. Markov Decision Processes in Artificial Intelligence, 2013: 1-38.

[44] Ahmadabadi M N, Asadpour M. Expertness based cooperative Q-learning[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2002, 32(1): 66-76.

[45] Degris T, White M, Sutton R S. Off-policy actor-critic[J]. arXiv preprint arXiv:1205.4839, 2012.

[46] Singh S, Jaakkola T, Littman M L, et al. Convergence results for single-step on-policy reinforcement-learning algorithms[J]. Machine learning, 2000, 38: 287-308.

[47] Watkins C J C H. Learning from delayed rewards[J], 1989.

[48] Watkins C J, Dayan P. Q-learning[J]. Machine learning, 1992, 8: 279-292.

[49] 刘全, 翟建伟, 章宗长, 等. 深度强化学习综述[J]. 计算机学报, 2018, 41(1): 1-27.

[50] Chung J, Gulcehre C, Cho K, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. arXiv preprint arXiv:1412.3555, 2014.

[51] Elman J L. Finding structure in time[J]. Cognitive science, 1990, 14(2): 179-211.

[52] Yu Y, Si X, Hu C, et al. A review of recurrent neural networks: LSTM cells and network architectures[J]. Neural computation, 2019, 31(7): 1235-1270.

[53] 李亚超, 熊德意, 张民. 神经机器翻译综述[J]. 计算机学报, 2018, 41(12): 2734-2755.

[54] Zheng H, Cao Y, Sun D, et al. Research on time series prediction of multi-process based on deep learning[J]. Scientific Reports, 2024, 14(1): 3739.

[55] Wang Z, Schaul T, Hessel M, et al. Dueling network architectures for deep reinforcement learning[C]. International Conference on Machine Learning, 2016: 1995-2003.

[56] Baird L C, Klopf A H. Reinforcement learning with high-dimensional continuous actions[J]. Wright Laboratory, Wright-Patterson Air Force Base, Tech. Rep. WL-TR-93-1147, 1993, 15.

[57] Tesauro G. Temporal difference learning and TD-Gammon[J]. Communications of the ACM, 1995, 38(3): 58-68.

[58] Fujimoto S, Hoof H, Meger D. Addressing function approximation error in actor-critic methods[C]. International Conference on Machine Learning, 2018: 1587-1596.

[59] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[J]. arXiv preprint arXiv:1509.02971, 2015.

[60] Treiber M, Hennecke A, Helbing D. Congested traffic states in empirical observations and microscopic simulations[J]. Physical review E, 2000, 62(2): 1805.

[61] Krauß S, Wagner P, Gawron C. Metastable states in a microscopic model of traffic flow[J]. Physical Review E, 1997, 55(5): 5597.

[62] 郭景华, 李克强, 罗禹贡. 智能车辆运动控制研究综述[J]. 汽车安全与节能学报, 2016, 7(2): 151-159.

[63] Brockman G, Cheung V, Pettersson L, et al. Openai gym[J]. arXiv preprint arXiv:1606.01540, 2016.

[64] Ahmed M, Abobakr A, Lim C P, et al. Policy-based reinforcement learning for

training autonomous driving agents in urban areas with affordance learning[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 23(8): 12562-12571.

[65] Huang Y, Wei G, Wang Y. Vd d3qn: the variant of double deep q-learning network with dueling architecture[C]. 2018 37th Chinese Control Conference (CCC), 2018: 9130-9135.

[66] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.

[67] Wang L, Guan Z, Liu J, et al. Research on the Driving Behavior and Decision-Making of Autonomous Vehicles (AVs) in Mixed Traffic Flow by Integrating Bilayer-GRU-Att and GWO-XGBoost Models[J]. World Electric Vehicle Journal, 2024, 15(8): 333.

[68] 李升波, 关阳, 侯廉, 等. 深度神经网络的关键技术及其在自动驾驶领域的应用[J]. 汽车安全与节能学报, 2019, 10(2): 119-145.

[69] Leurent E, Mercat J. Social attention for autonomous decision-making in dense traffic[J]. arXiv preprint arXiv:1911.12250, 2019.

[70] 刘志强, 韩静文, 倪捷. 智能网联环境下的多车协同换道策略研究[J]. 汽车工程, 2020, 42(3): 299-306.

攻读学位期间发表的学术论文成果

一、 学术论文

[1] 李帅，时培成，潘艺鑫，许宇翔，樊金生. 基于决斗双深度 Q 网络的多时间序列车辆行为决策[J]. 安徽工程大学学报，2025（三类，一作，已录用）

[2] Li Shuai, Shi P, Yang A. et al. DPD3QN: A Dueling Double Deep Q Network with Dual-Priority Experience Replay for Autonomous Driving Behavior Decision-Making [J]. Algorithms, 2025 (EI，中科院四区，一作，已录用)

二、 专利

[1] 时培成，李帅，等.一种用于智能驾驶汽车的信息采集装置[P].中国：CN 114715046A，2022-07-08.(发明实审)

三、 参与项目

[1] 智能网联汽车操作系统研究及产业化，芜湖市“赤铸之光”重大科技项目，项目编号：2023zd01

[2] 基于场景重构和协同控制的智驾域控制系统研究与产业化应用，长三角科技创新共同体联合攻关，项目编号:2023CSJGG1600

[3] 特定场景无人车通用化全线控底盘开发与产业化应用，安徽省省重点研究与开发计划项目，科技成果登记编号：2023F021Y013020

致谢

一段光阴，几番风雨，几多欢愉。回首研究生生涯，初入学府，懵懂憧憬；今至学成，踌躇满志。此番求索，不独学识之增益，亦乃心性之砥砺，思维之升华。谨以此文，致谢诸师长、同门、亲友，感念其提携扶持，厚德深恩，铭记于心。

首谢吾师，时培成教授，于学问之道，谆谆教诲，指点迷津；论文撰作，悉心指正；实验研讨，精心指引。师恩浩荡，启吾思维，授吾以渔，使余于学术之途愈行愈远，渐成独立之能。

次谢同门，诸位师兄、师姐，张建国师兄敦促教诲，毛飞、张庆、李屹师兄及江彤师姐谆谆指导，慷慨解惑。更感同窗潘艺鑫、许宇翔、周梦如、吴文超、董心龙诸君，相携共进，切磋琢磨，既增学识，亦厚情谊。复谢吾师弟，或于技艺指点，或于学理共论，启余思考，助余成长，愿其不懈钻研，学业精进，鹏程万里。

家人厚恩，铭刻于心。求学之路，崎岖坎坷，赖父母庇佑，方得安心治学。父母关爱，予吾无尽温暖；吾爱相伴，令吾心安神怡。此恩此情，终生不忘。

余自本科至硕士，皆就读于安徽工程大学。承母校教化，受师长熏陶，得学术资源之丰富，享良师益友之切磋，使吾才学有所成，眼界益拓，衷心感激。更谢诸位曾助吾者，或授业解惑，或扶持相助，皆铭感五内。

前路茫茫，然余心志弥坚。昔年求索，知坚持之可贵；今日学成，悟砥砺之真谛。展望未来，必当以热忱赴之，以勤勉践之，以学识报之。愿不负师长厚望，不违初心，不负韶华！

谨此致谢！

尚德教学　唯实惟新