基于LLM-DDPG协同决控实现闭环自动驾驶*

郭彤颖　樊烁　郑岩

（沈阳建筑大学电气与控制工程学院，沈阳 110168）

【摘要】针对自动驾驶领域中规则驱动方法在长尾场景下泛化能力有限，数据驱动模型存在数据过拟合、决策过程缺乏可解释性等问题，提出一种基于大语言模型（LLM）和深度确定性策略梯度（DDPG）驱动的自动驾驶框架。通过整合常识与环境数据，生成可解释性决策；利用DDPG算法实现底层控制；在存储模块中，设计数值+语义特征相似性检索机制，为当前决策任务实时匹配相似历史案例辅助动态决策。试验结果表明：相较于Rule-Driving和DQN-Driving，提出的方法在异构测试场景A中的成功率分别提升约33%和14%；在异构测试场景B中的成功率分别提升约70%和42%，表现出更强的跨场景泛化能力和环境适应能力。

主题词：自动驾驶　大语言模型　思维链　深度确定性策略梯度算法　案例推理

中图分类号：U461.91　　**文献标志码：**A　　**DOI:** 10.19620/j.cnki.1000-3703.20250272

Closed-Loop Autonomous Driving Based on LLM-DDPG Collaborative Decision and Control

Guo Tongying, Fan Shuo, Zheng Yan

（School of Electrical and Control Engineering, Shenyang Jianzhu University, Shenyang 110168）

【**Abstract**】To address the limited generalization ability of rule-based methods in long-tail scenarios, as well as the overfitting issue and poor interpretability of data-driven models in autonomous driving, this paper proposes an autonomous driving framework, based on a Large Language Model (LLM) and Deep Deterministic Policy Gradient (DDPG). The framework integrates commonsense knowledge with environmental information to generate interpretable driving decisions, while the DDPG algorithm is employed to achieve low-level continuous control. In the memory module, a hybrid retrieval mechanism that combines numerical-feature similarity and semantic-feature similarity is designed to match similar historical cases in real time, thereby supporting dynamic decision-making. Experimental results show that compared with Rule-Driving and DQN-Driving, the proposed method achieves relative improvements of approximately 33% and 14% in success rate, respectively, in heterogeneous test Scenario A. In heterogeneous test Scenario B, the relative improvements further increase to approximately 70% and 42%, demonstrating stronger cross-scenario generalization capability and environmental adaptability.

Key words: Autonomous driving, Large Language Model (LLM), Chain of thought, Deep Deterministic Policy Gradient (DDPG) algorithm, Case-based reasoning

【引用格式】郭彤颖，樊烁，郑岩. 基于LLM-DDPG协同决控实现闭环自动驾驶[J]. 汽车技术，2026(4): 10-16.

GUO T Y, FAN S, ZHENG Y. Closed-Loop Autonomous Driving Based on LLM-DDPG Collaborative Decision and Control[J]. Automobile Technology, 2026(4): 10-16.

1　前言

面向自动驾驶系统自主决策与协同控制的深层需求，增强复杂环境下系统运行的可靠性与决策合理性，对于提升自动驾驶安全性、稳定性和可解释性具有重要意义。

现有研究主要从规则驱动与数据驱动2个方向展开。早期的自动驾驶系统采用基于规则的方法，针对结构化场景设计确定性决策机制。Jaswanth等[1]提出基于有限状态机的自动化高速公路驾驶调节器，通过离散状态对驾驶行为建模，但当系统超出场景规则范围时，模型的鲁棒性明显下降。Brewitt等[2]利用决策树实现自动驾驶的目标识别，但仅限于固定场景且环境需完全可观测。金立生等[3]通过分析强化学习在单车自动驾驶决策中的应用，认为基于数据驱动[4]的自动驾驶模型存

*基金项目：教育部高等学校科学研究发展中心（2023BD008）。

在过拟合、数据偏差、决策过程无法解释等问题。规则驱动方法具有较强的逻辑明确性，但其泛化能力有限，难以有效应对复杂多变的长尾场景；而数据驱动方法虽然具备较强的环境感知与决策能力，但普遍存在过拟合风险，且决策过程缺乏可解释性。

随着大语言模型（Large Language Model，LLM）在认知智能领域的突破，为自动驾驶技术提供全新视角[5]。受Smallville[6]、Human[7]等最新进展的启发，基于LLM构建的广义知识驱动框架对于自动驾驶技术至关重要。凭借LLM强大的自然语言处理能力[8]、跨任务迁移学习特性[9]与小样本快速适配等优势，可弥补现有自动驾驶系统在逻辑推理、可解释性及场景适应性等方面的缺陷，但LLM仅提供最终决策，无法进行连续的动作控制。

因此，本文提出一种融合LLM与深度确定性策略梯度[10]（Deep Deterministic Policy Gradient，DDPG）算法的自动驾驶方法，围绕驾驶场景建模、推理决策、连续控制、反馈修正与经验存储等关键环节进行系统化设计，实现多模块协同工作。通过引入DDPG算法实现自动驾驶连续动作控制，同时，在存储模块中融合基于案例推理的相似性检索方法，通过数值特征与语义特征的联合匹配，为当前场景提供历史决策参考，提升系统在复杂交通环境下的决策准确性。

2 LD-Driving框架

基于LLM与DDPG融合方法构建自动驾驶框架LD-Driving，其整体结构如图1所示。该框架利用结构化推理模板，将驾驶场景转为文本描述输入至LLM。同时，基于案例推理检索与当前场景相似的案例，通过存储模块将结果传递至LLM，并进行信息融合。随后，LLM根据思维链提示逐步分析推理场景，生成最优决策。该决策通过DDPG算法映射为具体的底层控制指令，从而实现精确的车辆控制。

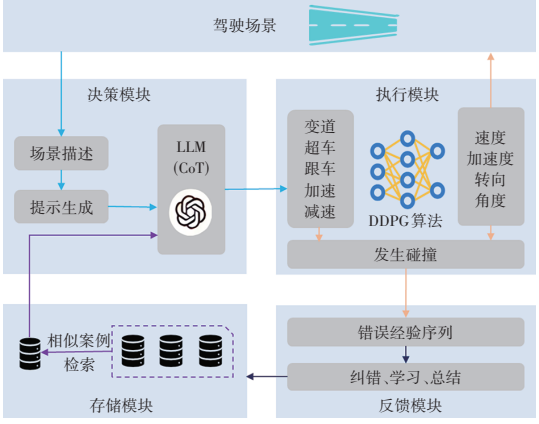


图1　LD-Driving框架结构

基于思维链[11]（Chain-of-Thought，CoT）提示设计结构化推理模板（见图2），构建显式的多步推理路径，使基于LLM的决策过程具备可分解性与可追溯性。系统能够对推理链中各中间状态进行校验，从而定位决策偏差，并抽取相应的错误经验序列。将该序列输入至反馈模块，由错误修正代理执行误差归因与策略修正，实现对原始决策的迭代优化。最终，优化后的决策及其对应场景被存储至案例库，为后续相似驾驶任务提供可复用的经验知识，提升系统整体的学习效率与决策可靠性。

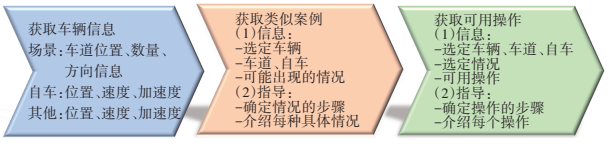


图2　基于CoT提示的结构化推理模板

2.1 决策模块

从Highway-env仿真环境[12]中提取自车和周车的状态信息，并进行特征描述。假设$i=0$为自车编码，$i=(1,\cdots,N)$为与当前决策相关的周车编码，对于任意车辆，其状态向量为：

$$S_i^t=(L_j,x_i^t,y_i^t,v_i^{x_t},v_i^{y_t},d_{0i})^{\mathrm{T}},i\in[0,N],j\in[1,10] \quad (1)$$

式中：L_j为车辆在j车道，x_i^t、y_i^t、v_i^x、v_i^y分别为t时刻车辆i的横向、纵向位置信息和速度，d_{0i}为自车与车辆i的相对距离。

决策模块采用GPT-3.5-Turbo作为LLM。首先，将当前驾驶场景中的结构化状态信息（自车与周车的位置、速度及车道关系等）通过CoT预定义语义模板进行编码，转换为统一格式的自然语言场景描述；然后，存储模块检索与当前场景相似的历史案例，将其作为上下文信息输入至LLM，用于决策参考；最后，将历史决策过程组织为多轮对话形式，引导模型在当前场景下生成更加合理的决策结果，同时保留完整的推理与决策过程，以增强决策的可解释性与可追溯性。

构建输入序列后，模型先通过预训练语言模型BERT（Bidirectional Encoder Representations from Transformers）的WordPiece分词算法[13]对长语句进行子词切分，将文本转化为词元序列。嵌入层利用预训练模型的嵌入矩阵，将每个词元映射为高维向量表示，并通过位置编码显式注入序列的位置信息，构建融合语义与位置特征的自然语言指令表征空间。

基于Transformer多层自注意力机制[14]捕捉全局语义信息，系统通过自车的状态生成查询矩阵$\boldsymbol{Q}=[\boldsymbol{q}_0],\boldsymbol{Q}\in\mathbf{R}^d$，其中，$\boldsymbol{q}_0$为查询向量，$d$为特征向量维度。同时，从周车中筛关键目标，构建动态特征集

合，形成键矩阵 $\boldsymbol{K}=(\boldsymbol{k}_1,\cdots,\boldsymbol{k}_N),\boldsymbol{K}\in\mathbf{R}^{N\times d}$ 和值矩阵 $\boldsymbol{V}=(\boldsymbol{v}_1,\cdots,\boldsymbol{v}_N),\boldsymbol{V}\in\mathbf{R}^{N\times d}$，其中，$N$ 为周车的数量，$\boldsymbol{k}_i$、$\boldsymbol{v}_i$ 为对应周车的隐空间表征。

通过缩放点积注意力机制，计算 $\boldsymbol{q}_0$ 和 $\boldsymbol{k}_i$ 的相似性：

$$\alpha_i=\frac{\boldsymbol{q}_0\boldsymbol{k}_i^{\mathrm{T}}}{\sqrt{d}} \tag{2}$$

式中：$\sqrt{d}$ 为缩放因子。

对关联强度序列 $\alpha_{1:N}$ 进行归一化处理，转为概率分布，即 $\boldsymbol{\omega}=\mathrm{softmax}(\alpha_1,\cdots,\alpha_N)$，基于权重 $\boldsymbol{\omega}$ 对值矩阵中各列向量 $\boldsymbol{v}_i$ 进行加权求和，为后续决策提供语义表示：

$$\boldsymbol{O}_{\mathrm{output}}=\sum_{i=1}^{N}\boldsymbol{\omega}_i\boldsymbol{v}_i \tag{3}$$

2.2 控制模块

由于LLM的决策过程缺乏对车辆运动学系统的约束，导致生成的高层指令可能违反运动学可行性。因此，LD-Driving框架采用分层控制架构，通过语义-物理动作转换接口（Semantic-to-Physical Interface，SPI），将决策模块生成自然语言指令转为连续语义向量，作为DDPG的附加输入，利用动作解码层将语义向量映射到符合物理约束的连续控制信号。

假设高层语义动作空间为 A_{high}，底层物理动作空间为 $\boldsymbol{A}_{\mathrm{low}}$。基于场景上下文将离散语义转为约束参数，采用参数化模型生成平滑且符合运动学约束的轨迹，得到系数 a_0~a_5 并生成无冲击轨迹。最终，对目标状态和约束进行编码，嵌入DDPG的奖励函数与状态空间：

$$\begin{cases}A_{\mathrm{high}}=\{I_{\mathrm{idle}},a_{\mathrm{acc}},a_{\mathrm{dec}},T_{\mathrm{left}},T_{\mathrm{right}}\}\\ \boldsymbol{A}_{\mathrm{low}}=(\delta,a)^{\mathrm{T}},\delta\in[-1,1],a\in[-3,3]\end{cases} \tag{4}$$

式中：I_{idle} 为保持当前状态，a_{acc} 为加速，a_{dec} 为减速，T_{left} 为左转，T_{right} 为右转，δ 为转向角度，a 为加速度。

采用五次多项式进行横向轨迹规划，满足自动驾驶变道舒适性约束。横向轨迹表达式为 $d(t)=a_5t^5+a_4t^4+a_3t^3+a_2t^2+a_1t+a_0$，横向运动的边界约束条件为：

$$\begin{cases}d(0)=d_{\mathrm{current}},\dot{d}(0)=0,\ddot{d}(0)=0\\ d(T)=d_{\mathrm{target}},\dot{d}(T)=0,\ddot{d}(T)=0\end{cases} \tag{5}$$

式中：T 为轨迹规划总时长，d_{current} 为当前横向位置，d_{target} 为到达目标位置。

采用匀加速直线运动生成纵向轨迹，并施加安全与舒适性约束，纵向速度为：

$$v(t)=v_0+at \tag{6}$$

式中：v_0 为纵向初始速度，a 为纵向加速度。

纵向运动约束条件为 $|a|\leqslant 3\ \mathrm{m/s^2}$，$j_t=\left|\frac{\mathrm{d}a}{\mathrm{d}t}\right|\leqslant 2.5\ \mathrm{m/s^3}$。其中，$j_t$ 为加速度变化率。

为实现自动驾驶中的连续动作控制，本文引入基于Actor-Critic框架[15]构建的DDPG算法。通过策略网络与价值网络的协同优化，实现模型对连续动作的高效学习。假设 $\boldsymbol{s}_t\in\boldsymbol{S}$ 表示时刻 t 的状态向量，其中，$\boldsymbol{S}$ 为状态空间。Critic网络 $Q(\boldsymbol{s},\boldsymbol{a}|\boldsymbol{\theta}^Q)$ 用于估计状态-动作值函数，Actor网络 $\mu(\boldsymbol{s}|\boldsymbol{\theta}^\mu)$ 用于将当前状态映射为最优动作。其中，Critic价值网络通过最小化时序差分误差进行训练，更新 Q 网络的权重 $\boldsymbol{\theta}^Q$，使 Q 网络能够准确估计不同状态下采取不同动作的价值，减少预测值与真实值间的差距。故价值网络的损失函数为：

$$L(\boldsymbol{\theta}^Q)=E_{(\boldsymbol{s},\boldsymbol{a},r,\boldsymbol{s}')_D}\left[\left(y_{\mathrm{target}}-Q(\boldsymbol{s},\boldsymbol{a}|\boldsymbol{\theta}^Q)\right)^2\right] \tag{7}$$

式中：$E_{(\boldsymbol{s},\boldsymbol{a},r,\boldsymbol{s}')_D}$ 为在经验回放缓冲区 D 的样本分布上求期望，y_{target} 为时序差分目标值。

利用目标网络的延迟更新机制构建时序差分（Temporal Difference，TD）目标值 y_{target}，由即时奖励 r 与折扣后的未来期望回报共同组成。其中，r 基于下一时刻状态 $\boldsymbol{s}'$ 进行估计：首先，由目标策略网络 μ' 生成对应的确定性动作 $\boldsymbol{a}'=\mu'(\boldsymbol{s}')$；随后，将该动作输入目标价值网络 Q'，得到未来状态-动作对的价值估计。最终，得到TD目标值为：

$$y_{\mathrm{target}}=r+\gamma Q'(\boldsymbol{s}',\mu'(\boldsymbol{s}'|\boldsymbol{\theta}^{\mu'})|\boldsymbol{\theta}^{Q'}),\gamma\in(0,1) \tag{8}$$

式中：γ 为折扣因子，$\boldsymbol{\theta}^{\mu'}$、$\boldsymbol{\theta}^{Q'}$ 分别为目标策略网络和目标价值网络参数。

为了保证训练过程的稳定性，目标网络参数采用软更新策略进行迭代更新，更新形式为：

$$\boldsymbol{\theta}'\leftarrow\tau\boldsymbol{\theta}+(1-\tau)\boldsymbol{\theta}' \tag{9}$$

式中：$\boldsymbol{\theta}$、$\boldsymbol{\theta}'$ 分别为当前网络与目标网络的参数，$\tau=0.001$ 为更新系数。

Actor网络通过最大化策略的期望回报函数，对策略参数进行梯度上升更新，从而学习从状态到连续动作的映射关系。其策略梯度形式为：

$$\nabla_{\boldsymbol{\theta}^\mu}J\approx E\left[\nabla_a Q(\boldsymbol{s},\boldsymbol{a}|\boldsymbol{\theta}^Q)|_{\boldsymbol{a}=\mu(\boldsymbol{s})}\nabla_{\boldsymbol{\theta}^\mu}\mu(\boldsymbol{s}|\boldsymbol{\theta}^\mu)\right] \tag{10}$$

式中：$\nabla_{\boldsymbol{\theta}^\mu}J$ 为策略梯度，$\nabla_a Q()$ 为价值网络对动作的梯度，$\nabla_{\boldsymbol{\theta}^\mu}\mu()$ 为策略网络的参数梯度。

通过上述梯度更新机制，Actor网络能够在Critic网络的价值评估指导下，对策略参数进行持续优化，学习在长期回报意义下更优的状态-动作映射关系。在自动驾驶任务中，该过程使车辆在加速、转向等连续控制过程中，生成更加平滑且符合安全约束的控制信号，从而提升整体驾驶性能与系统稳定性。

2.3 反馈模块

在复杂驾驶环境中，自动驾驶系统的决策模块因场

景理解不准确或推理链缺陷，会导致错误决策。为此，可引入反馈模块，通过错误检测和校正，生成实现系统的持续优化，增强决策的安全性。

当车辆发生碰撞时，系统会检测与异常情况相对应的环境状态、决策动作，利用思维链定位故障点，为后续的反思推理提供完整的信息。随后，LLM成为纠错代理，分析错误的内部逻辑缺陷或外部环境因素，结合领域知识、错误因素和历史经验，进一步生成改进策略。同时，系统将修正后的经验保存至向量数据库，进行持久化存储，形成结构化经验知识库。

2.4 存储模块

基于预训练的LLM缺乏驾驶经验，其推理能力受限于先验知识覆盖范围，无法实现有效推理。因此，构建存储模块，使系统能够通过持续的驾驶训练积累经验，提高驾驶技能，并将所学经验运用到未来的交互中，协助LLM进一步决策。为了确保系统能够快速检索这些历史案例，提高决策效率并降低错误率，在存储模块中引入基于案例推理（Case-Based Reasoning，CBR）功能。

存储模块采用键值记忆网络，将决策数据组织为索引案例，每个案例捕获场景特征作为键$\boldsymbol{k}_i$，决策元组作为值$\boldsymbol{v}_i$。键空间通过编码函数$\boldsymbol{k}_i = f_{\text{enc}}(\boldsymbol{s}_t, \boldsymbol{a}_t)$，将当前状态$\boldsymbol{s}_t$和动作$\boldsymbol{a}_t$映射为高维特征向量，值空间则存储对应的状态-动作奖励三元组$(\boldsymbol{s}_t, \boldsymbol{a}_t, r_t)$和与其关联的LLM生成驾驶策略片段$\boldsymbol{\pi}^i_{\text{LLM}}$，并以支持高效查询的嵌入向量形式存储至Chroma向量数据库。

当面临驾驶任务时，从当前环境中提取关键特征，采用相似性度量方法对比数据与存储案例，将检索结果作为提示输入至LLM，辅助改进实时决策。本文设计2类相似性度量方法，分别为特征相似性和语义相似性。

在数值特征相似性方面，对案例中的数值型特征（车速、车道位置、前后车距等）进行归一化处理，消除不同驾驶场景特征间的尺度差异。特征x的归一化方式为：

$$x^{\text{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{11}$$

式中：$x_{\max}$、$x_{\min}$分别为该数值型特征在案例库中的最大值和最小值。

利用加权欧几里得距离，计算案例A与案例B在n个归一化特征维度上的相似性$d(A,B)$，相似性值越小，表明案例相似性越高：

$$d(A,B) = \sqrt{\sum_{i=1}^{n} \boldsymbol{w}_i \left(A_i^{\text{norm}} - B_i^{\text{norm}}\right)^2} \tag{12}$$

式中：$\boldsymbol{w}_i$为第i个特征的权重。

将$d(A,B)$转换为数值特征相似性：

$$Sim_{\text{num}}(A,B) = \frac{1}{1 + d(A,B)} \tag{13}$$

在语义特征相似性方面，对于非数值化特征，利用预训练的嵌入模型对输入文本进行编码，将其映射为固定维度的语义向量表示。通过向量间的余弦相似度$Sim_{\text{sem}}(A,B)$衡量文本间的语义相似性，其值越接近1，表示语义相似性越高：

$$Sim_{\text{sem}}(A,B) = \frac{\boldsymbol{A} \cdot \boldsymbol{B}}{\|\boldsymbol{A}\| \|\boldsymbol{B}\|} \tag{14}$$

式中：$\boldsymbol{A}$、$\boldsymbol{B}$为文本描述的向量表示，$\|\boldsymbol{A}\|$、$\|\boldsymbol{B}\|$为向量模长。

单一维度的相似性度量难以全面反映两个驾驶场景间的真实相关程度。为此，本文引入模态融合系数，对数值特征相似性与语义相似性进行加权融合，构建综合相似性度量函数：

$$Sim(A,B) = \beta Sim_{\text{num}}(A,B) + (1 - \beta) Sim_{\text{sem}}(A,B) \tag{15}$$

式中：$\beta = 0.5$为模态融合系数。

在实际应用中，存储模块通过相似性度量方法比较当前场景描述与历史案例，选择相似度最高的k个案例，即$\{A_1, A_2, \cdots, A_k\}$。根据相似度为$Sim(A, A_i)$特征相似性与语义相似性的加权结果，生成当前场景的参考决策为：

$$S_{\text{current}} = \frac{\sum_{i=1}^{k} Sim(A, A_i) \cdot S_i}{\sum_{i=1}^{k} Sim(A, A_i)} \tag{16}$$

式中：S_i为第i个历史案例对应的决策指令，A为当前驾驶场景的数值特征与语义特征，A_i为第i个历史案例的数值特征与语义特征。

3 试验结果与分析

3.1 试验参数与指标

本文在Highway-env模拟器中进行仿真试验，根据试验需求设置道路类型、车流密度、车辆速度、车道线数量等参数，并将车辆信息输入至场景描述。试验中使用高速公路场景，所有车辆的速度范围为15~32 m/s，自车的初始速度为15 m/s。对于闭环驾驶任务，仿真周期设置为30 s，决策频率为1 Hz。若自车正常行驶，且30 s内未发生碰撞，则认为任务完成。

为评估本文方法的性能，采用成功率A_{SR}和成功步骤数A_{SS}作为评估指标。成功率的定义为：

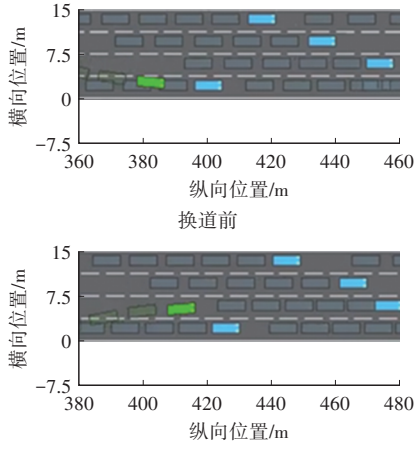
$$A_{\text{SR}} = \frac{A_{\text{SS}}}{A_{\text{SS,T}}} \times 100\% \tag{17}$$

式中：$A_{SS,T}$为仿真步骤总数。

若在一个仿真周期内，车辆在30次决策中无碰撞，则$A_{SS}=30$。在N次试验中，分别计算A_{SS}的最小值$A_{SS,min}$、最大值$A_{SS,max}$和平均值$A_{SS,mean}$。其中，$A_{SS,mean}=\frac{1}{N}\sum_{i=1}^{N}A_{SS,i}$。

3.2 决策流程试验

基于LLM的思维链推理机制，验证本文系统其在自动驾驶决策场景中的可解释性，结果如图3所示。在驾驶场景中，决策模块通过思维链提示技术实现了分步透明化推理并提供最终决策，避免了“黑盒”问题，从而增强了用户对自动驾驶系统的信任感和接受度。



（a）驾驶场景与车辆状态可视化

驾驶场景描述：
您正在一条4车道的道路上行驶，当前行驶在最右侧车道上。您的当前位置为(400.30 m, 11.38 m)，速度为25.15 m/s，加速度为−0.27 m/s²，车道位置为400.30 m。您的周围还有其他车辆行驶，基本信息如下：
• 136号车与您在同一车道上行驶，并且位于您的前方。其位置为(410.52 m,12.00 m)，速度为23.13 m/s，加速度为−1.55 m/s²，车道位置为410.52 m。
• 192号车行驶在您左侧的车道上，并且位于您的前方。其位置是(440.85 m,8.00 m)，速度是21.83 m/s，加速度是−2.50 m/s²，车道位置是440.85 m。

代理答案(推理解释过程)：
我有3个动作可供选择。现在，我想知道哪个动作是可行的。首先检查我是否可以加速、保持，还是左转。
• **我想知道我是否可以加速**，所以我需要观察当前车道上前面的车，也就是136号车。我和136号车之间的距离是410.52−400.30=10.22 m，速度差是23.13−25.15=−2.02 m/s。136号车在我前面行驶了10.22 m，我的速度已经比当前车道上前车快2.02 m/s，所以我不想加速。
• **然后，让我们检查我是否可以保持**。我需要观察当前车道上前面的车，也就是136号车。通过以上计算，我知道我与136号车的距离为10.22 m，速度差为−2.02 m/s。10.22 m的距离小于我期望的30 m跟车距离，而我的速度高于当前车道上的前车。如果我选择怠速，我与136号车的距离会进一步缩小，所以我不应该怠速。
• **最后，我可以考虑换道**。左侧车道上的192号车在我前方440.85−400.30=40.55 m。这个距离大于我期望的30 m跟车距离，而192号车的速度为21.83 m/s，低于我的速度。但是，由于距离足够远，我可以在换道后调整速度。因此，向左换道是可行的。太好了，我现在可以做决定了。决策耗时2.4 s。
最终答案：左转

（b）思维链推理与决策生成过程

图3 基于LLM思维链的自动驾驶决策可解释性

3.3 对比试验

本文通过对照试验定量评估DDPG算法在底层连续控制中的性能，在4车道动态交通场景下，对比分析LLM系统与LLM+DDPG分层架构的性能差异，结果如图4所示。试验采用固定随机种子初始化存储模块，构建容量$|M|=30$的存储库，few-shot示例数设置为5，并通过30次独立重复试验消除初始条件偏差。其中，试验组由LLM生成语义目标后，经DDPG策略网络优化为连续控制量；而对照组直接将LLM输出的离散指令映射至预设基础动作空间。

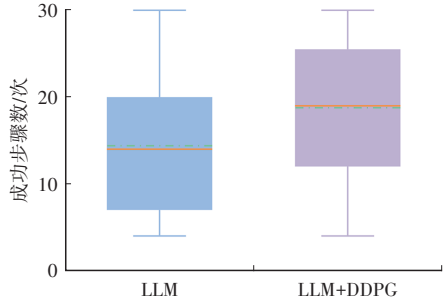


图4 不同方案成功步骤数对比结果

由图4可知，LLM+DDPG分层架构的成功步骤数指标显著优于LLM系统，其成功步骤中位数较LLM提升约33%，且数据分布区间更趋收敛。相较于直接映射预设基础动作空间的方式，LLM+DDPG在动作空间连续性和环境交互精度方面更具优势。

为了评估迭代反馈模块的有效性，通过启用反馈模块(YF)和禁用反馈模块(NF)2种方式进行对比试验，结果如表1所示。初始化存储模块，构建容量$|M|=30$的存储库，few-shot示例数设置为5，并通过随机种子进行30次独立重复试验消除初始条件偏差。$A_{SS,Q1}$、$A_{SS,Q3}$和$A_{SS,median}$分别为成功步骤数的下四分位数、上四分位数和中位数。

表1 有/无反馈模块成功步骤数对比结果 次

模块	$A_{SS,min}$	$A_{SS,Q1}$	$A_{SS,median}$	$A_{SS,Q3}$	$A_{SS,max}$	$A_{SS,mean}$
NF	2.0	5.0	9.0	13.0	22.0	11.5
YF	4.0	9.0	15.0	23.0	30.0	19.0

由表1可知，引入反馈模块后，$A_{SS,median}$和$A_{SS,mean}$分别提高8次和7.5次，说明反馈机制通过修正决策偏差，能够有效增强系统的任务执行能力。

3.4 存储模块

存储模块可帮助自动驾驶系统学习历史经验，避免发生重复错误。为了验证该模块的有效性，设置车道数量为4车道，初始化存储模块，并构建容量$|M|=30$的存储库；设置few-shot示例数为$k=\{0,1,3,5\}$，其中，$k=0$为零样本基线，表示不启用存储模块；$k>0$指在推理过程中，将k个最相似的历史经验作为提示，通过提示工程将k个驾驶场景描述-动作对注入LLM的上下文窗口，协助LLM进行决策。不同few-shot示例数对决策性能的影响结果如表2所示。

表2　不同few-shot示例数的成功步骤数对比结果　次

k	$A_{SS,min}$	$A_{SS,Q1}$	$A_{SS,median}$	$A_{SS,Q3}$	$A_{SS,max}$	$A_{SS,mean}$
0	2.0	3.0	5.0	7.0	8.0	5.03
1	3.0	6.0	9.0	12.0	19.0	8.72
3	4.0	9.0	14.0	18.5	30.0	13.84
5	4.0	15.0	19.0	23.0	30.0	18.38

由表2可知，当$k=0$时，系统的$A_{SS,median}$、$A_{SS,mean}$、$A_{SS,Q1}$、$A_{SS,Q3}$均保持最低水平，说明零样本场景下LLM难以应对复杂高速场景，决策稳定性与性能上限存在明显短板。当$k=1$增至$k=5$，$A_{SS,median}$、$A_{SS,mean}$、$A_{SS,Q1}$、$A_{SS,Q3}$均持续显著提升，$A_{SS,median}$、$A_{SS,mean}$的增幅分别达111.1%、110.8%，且在$k\geqslant 3$时$A_{SS,max}$达到场景上限，$A_{SS,Q1}$的提升表明系统决策鲁棒性显著增强。

如图5所示，随着few-shot示例数的增加，系统成功步骤数显著提升，表明存储模块可有效增强决策能力。当存储模块关闭，自动驾驶系统的决策能力较弱，成功步骤数普遍偏低，表明系统缺乏足够的先验知识来有效应对复杂的高速公路情况。综上，存储模块的引入，有效提升了自动驾驶系统在复杂高速场景下的决策能力与可靠性。

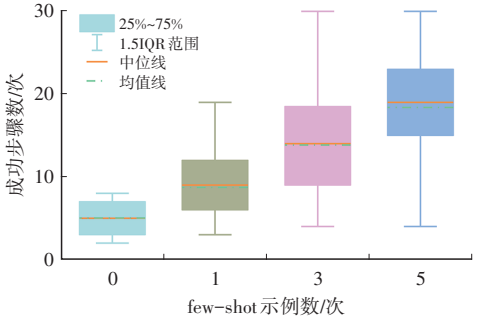


图5　不同few-shot示例数下成功步骤数箱线图

3.5　检索与修正

为验证本文方法在驾驶任务中的案例检索与决策修正功能，其可视化结果如图6所示。当系统获取到当前驾驶场景信息时，存储模块启动，检索出与当前场景高度相似的历史案例。依据历史案例及其对应的成功驾驶决策，为当前驾驶决策提供参考依据。在决策执行过程中，反馈模块持续监测系统状态和行为。当检测到决策错误，该模块触发自我反馈机制，识别错误决策并生成具体的修正意见。确认修正决策的有效性后，系统决策和与其对应的情境信息会作为一个新的案例项，被动态添加回案例存储模块，说明该机制显著增强了系统的适应性和持续改进能力。

3.6　跨场景泛化能力试验

数据驱动的自动驾驶决策模型依赖大规模训练数据实现行为泛化，本文基于LLM构建小样本决策代理。为了系统地验证LD-Driving框架的泛化能力，本文设计跨场景泛化能力试验，选取基于规则的驾驶模型(Rule-Driving)以及基于深度Q网络驾驶模型[16](DQN-Driving)进行对比分析。所有模型训练周期：LD-Driving迭代100次，Rule-Driving和DQN-Driving通过网格搜索确定最优训练次数。在相同条件下完成训练后，构建2个异构测试场景进行样本迁移。试验测试场景为：

a. 基准场景：4车道、车流密度为2辆/(100 m·车道)。

b. 测试场景A：3车道、车流密度为1.5辆/(100 m·车道)；

c. 测试场景B：5车道、车流密度为2.5辆/(100 m·车道)。

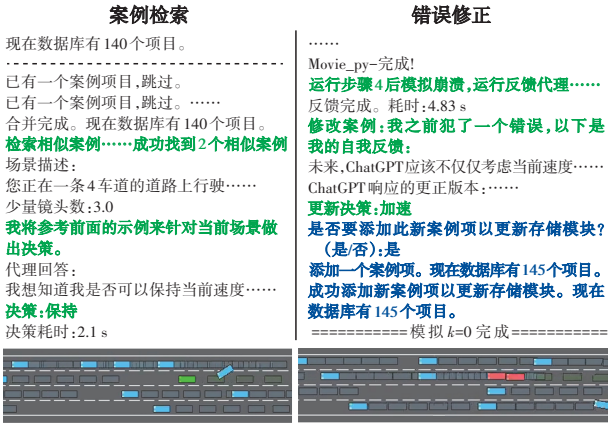


图6　案例检索及错误修正示例

试验采用统一的评估指标体系和随机种子策略，以保证试验的可重复性与可靠性，不同模型的泛化性评估结果如图7所示。相较于Rule-Driving和DQN-Driving，LD-Driving在异构测试场景A中的成功率分别提升约33%和14%；在异构测试场景B中的成功率分别提升约70%和42%。LD-Driving在较少训练迭代次数下即可获得稳定的性能表现，表明该方法在新场景中仍具备较强的决策能力，从而体现出良好的跨场景泛化能力与环境适应能力。

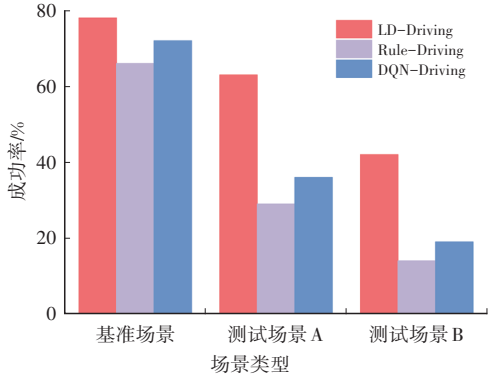


图7　异构测试场景下模型性能对比结果

4 结束语

本文提出一种闭环自动驾驶框架LD-Driving，有效弥补了传统规则驱动和数据驱动方法的缺陷。该框架融合LLM的认知推理能力与DDPG算法的连续控制能力，构建包含案例推理的存储模块和错误修正的反馈模块。在复杂交通环境中具备较强的决策稳定性与跨场景泛化能力。同时，为构建具备可解释性与自适应能力的自动驾驶系统提供了一种可行思路。未来，将结合多模态感知信息与真实交通数据对模型进行进一步优化，以提升其在实际部署中的可靠性与安全性。

参 考 文 献

[1] JASWANTH M, NARAYANA N K L, RAHUL S, et al. Autonomous Car Controller Using Behaviour Planning Based on Finite State Machine[C]// 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI). Tirunelveli, India: IEEE, 2022: 296-302.

[2] BREWITT C, GYEVNAR B, GARCIN S, et al. GRIT: Fast, Interpretable, and Verifiable Goal Recognition with Learned Decision Trees for Autonomous Driving[C]// 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Prague, Czech Republic: IEEE, 2021: 1023-1030.

[3] 金立生, 韩广德, 谢宪毅, 等. 基于强化学习的自动驾驶决策研究综述[J]. 汽车工程, 2023, 45(4): 527-540.
JIN L S, HAN G D, XIE X Y, et al. Review of Autonomous Driving Decision-Making Research Based on Reinforcement Learning[J]. Automotive Engineering, 2023, 45(4): 527-540.

[4] ZHAO R, LI Y, FAN Y Z, et al. A Survey on Recent Advancements in Autonomous Driving Using Deep Reinforcement Learning: Applications, Challenges, and Solutions[J]. IEEE Transactions on Intelligent Transportation Systems, 2024, 31(5): 256-263.

[5] 秦小林, 古徐, 李弟诚, 等. 大语言模型综述与展望[J]. 计算机应用, 2025, 45(3): 685-696.
QIN X L, GU X, LI D C, et al. Survey and Prospect of Large Language Models[J]. Journal of Computer Applications, 2025, 45(3): 685-696.

[6] PARK J S, O'BRIEN J, CAI C J, et al. Generative Agents: Interactive Simulacra of Human Behavior[C]// Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology. New York, United States: ACM, 2023: 1-22.

[7] FU D, LI X, WEN L, et al. Drive Like a Human: Rethinking Autonomous Driving with Large Language Models[C]// 2024 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW). IEEE, 2024: 910-919.

[8] 车万翔, 窦志成, 冯岩松, 等. 大模型时代的自然语言处理：挑战、机遇与发展[J]. 中国科学: 信息科学, 2023, 53(9): 1645-1687.
CHE W X, DOU Z C, FENG Y S, et al. Towards a Comprehensive Understanding of The Impact of Large Language Models on Natural Language Processing: Challenges, Opportunities and Future Directions[J]. Scientia Sinica (Informationis), 2023, 53(9): 1645-1687.

[9] TAN Y, ZHANG E M, LI Y, et al. Transferability-Guided Cross-Domain Cross-Task Transfer Learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2025, 36(2): 2423-2436.

[10] 高兴媛, 和铁行. 基于深度确定性策略梯度算法的交通协同智能控制技术分析[J]. 国外电子测量技术, 2025, 44(1): 54-61.
GAO X Y, HE T X. Technology Analysis of Traffic Cooperative Intelligent Control Based on Depth Deterministic Strategy Gradient Algorithm[J]. Foreign Electronic Measurement Technology, 2025, 44(1): 54-61.

[11] WEI J, WANG X Z, SCHUURMANS D, et al. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models[C]// Proceedings of the 36th International Conference on Neural Information Processing Systems. Red Hook, NY, United States: NIPS, 2022, 35: 24824-24837.

[12] GOPINATH D, DECASTRO J, ROSMAN G, et al. HMIway-Env: A Framework for Simulating Behaviors and Preferences to Support Human-AI Teaming in Driving[C]// 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). New Orleans, LA, USA: IEEE, 2022: 4342-4350.

[13] FENG Z Y, TANG D Y, FENG X C, et al. Pretraining without Wordpieces: Learning over a Vocabulary of Millions of Words[J]. International Journal of Machine Learning and Cybernetics, 2024, 15(9): 3989-3998.

[14] 杨海飞, 唐勇, 刘喆, 等. 基于注意力机制的交织区轨迹预测及应用[J]. 汽车技术, 2025(12): 1-9.
Yang H F, Tang Y, Liu Z, et al. Trajectory Prediction and Application of Interweaving Area Based on Attention Mechanism[J]. Automobile Technology, 2025(12): 1-9.

[15] HAN M H, ZHANG L X, WANG J, et al. Actor-Critic Reinforcement Learning for Control with Stability Guarantee[J]. IEEE Robotics and Automation Letters, 2020, 5(4): 6217-6224.

[16] BELLOTTI F, LAZZARONI L, CAPELLO A, et al. Explaining a Deep Reinforcement Learning (DRL)-Based Automated Driving Agent in Highway Simulations[J]. IEEE Access, 2023, 11: 28522-28550.

（责任编辑 瑞 秋）
修改稿收到日期为2025年6月13日。