基于多智能体深度强化学习的智能网联汽车服务迁移优化方法

芮兰兰，邓淑予，陈子轩，高志鹏，邱雪松，郭少勇
（北京邮电大学网络与交换技术全国重点实验室，北京 100876）

摘　要：为应对智能网联汽车在高动态车联网环境中服务迁移所面临的多用户资源竞争与边缘节点可用性动态变化等挑战，提出了一种基于多智能体组相对策略优化（MAGRPO）的服务迁移方法，将服务迁移问题形式化为带资源约束的长期多用户联合优化问题，并设计了一种不需要显式Critic网络的MAGRPO算法。基于组内折扣回报的相对排序构建策略更新信号，有效缓解由强约束惩罚（如节点过载或故障）引起的训练不稳定问题，并降低训练开销。仿真结果表明，所提方法在服务总时延、迁移能耗及迁移成功率等关键指标上均优于现有基线方法，尤其在边缘节点资源受限且可用性动态变化的场景下，展现出更强的鲁棒性与可扩展性。

关键词：移动边缘计算；智能网联汽车；服务迁移；多智能体深度强化学习；组相对策略优化
中图分类号：TP393
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2026005

Service migration optimization method for intelligent connected vehicles based on multi-agent deep reinforcement learning

Rui Lanlan, Deng Shuyu, Chen Zixuan, Gao Zhipeng, Qiu Xuesong, Guo Shaoyong
State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract: To address the challenges of multi-user resource competition and dynamic changes in edge node availability faced by intelligent connected vehicles during service migration in a highly dynamic Internet of vehicles environment, a service migration method based on multi-agent group relative policy optimization (MAGRPO) was proposed. The service migration problem was formalized as a long-term multi-user joint optimization problem with resource constraints, and a MAGRPO algorithm that did not require an explicit critic network was designed. A policy update signal was constructed based on the relative ranking of discounted returns within the group, thereby effectively mitigating training instability caused by severe penalties (e.g., node overload or failure) and reducing training cost. Simulation results show that the proposed method outperforms existing baseline methods in key metrics such as total service delay, migration energy consumption, and migration success rate. It exhibits stronger robustness and scalability in scenarios where edge node resources are limited and their availability changes dynamically.

Keywords: mobile edge computing, intelligent connected vehicles, service migration, multi-agent deep reinforcement learning, group relative policy optimization

收稿日期：2025–11–12；**修回日期：**2025–12–31
通信作者：邓淑予，shuyudeng@bupt.edu.cn
基金项目：国家自然科学基金资助项目（No.62471051）；河北省创新能力提升计划基金资助项目（No.V1755673688106）
Foundation Items: The National Natural Science Foundation of China (No.62471051), Hebei Provincial Innovation Capacity Enhancement Program Project (No.V1755673688106)

0 引言

随着车联网（Internet of vehicles, IoV）技术的发展，智能网联汽车（intelligent connected vehicle, ICV）日益承载自动驾驶决策、增强现实导航、沉浸式车载娱乐等高实时性服务，对数据传输速率和端到端时延提出了严苛要求。然而，车载嵌入式系统受限于本地资源瓶颈，难以满足上述服务的性能需求[1]。远程云计算可提供充足的计算与存储资源，但其长距离传输时延难以满足自动驾驶等安全攸关场景的严格时延要求[2]。为克服上述局限，移动边缘计算（mobile edge computing, MEC）被引入车联网架构。MEC通过在路侧单元（road side unit, RSU）等网络边缘部署轻边缘服务器（edge server, ES）实现算力下沉，从而缩短数据传输路径，降低服务响应时延，并缓解核心网络的传输压力。

然而，在高动态车联网环境中，高速移动的车辆常驶出当前服务节点覆盖范围，引发服务链路切换或回程路径延长，进而造成时延抖动，严重威胁安全攸关型服务的服务质量（quality of service, QoS），甚至导致交通事故[3]。为维持服务连续性并保障QoS，已有研究采用服务迁移机制[4-5]，在用户移动过程中将服务实例及其运行状态动态迁移至邻近的可用边缘节点。然而，在高动态车联网环境中，实现高效可靠的服务迁移决策面临多重挑战。

现有研究在服务迁移机制方面仍存在若干局限。首先，多数研究聚焦于单用户场景，忽视了高密度交通环境下的多车资源竞争，易导致目标节点过载，进而引发迁移失败或服务性能下降。其次，主流方案通常假设边缘节点始终可用，缺乏对节点故障等异常状况的鲁棒性考量。此外，在决策算法层面，传统深度Q学习（deep Q-learning, DQL）方法在大规模网络中面临动作空间爆炸问题；而以多智能体近端策略优化（multi-agent proximal policy optimization, MAPPO）为代表的多智能体演员–评论家（multi-agent actor-critic, MAAC）[6]算法虽可避免该问题，但其依赖的全局Critic网络在面临强惩罚（如节点过载或高故障风险引起的显著负奖励）时，价值估计易产生偏差。此外，全局Critic网络的输入维度随智能体数量线性增长，制约了该算法在大规模MEC场景中的可扩展性。针对上述挑战，本文提出了一种面向动态车联网环境的服务迁移目标决策方法，旨在协同优化服务总时延与迁移能耗，同时保障高迁移成功率，以满足智能网联汽车对车载服务的QoS要求。本文的主要贡献如下。

1) 构建了一个融合车辆移动性、多用户资源竞争及边缘节点动态可用性的综合系统模型，将服务迁移问题建模为资源约束下的长期多用户联合优化问题，目标是在满足边缘计算资源限制的前提下，最小化服务总时延与迁移能耗的加权和。

2) 提出了一种基于多智能体组相对策略优化（multi-agent group relative policy optimization, MAGRPO）的服务迁移方法。针对传统多智能体算法中集中式Critic网络在高维空间及极端负奖励（如由节点过载或故障引发的惩罚）下训练不稳定的问题，MAGRPO摒弃显式Critic网络，利用组内折扣回报的相对排序构建策略更新信号，从而实现自适应且鲁棒的目标迁移选择。此外，由于不需要训练Critic网络，该方法有效降低了显存占用。

3) 基于真实罗马出租车轨迹数据集构建了动态车联网仿真环境。实验结果表明，本文方法在高动态和高竞争场景下展现出更强的训练稳定性与适应性，在服务总时延、迁移能耗及迁移成功率等关键指标上均优于现有基线方法。

1 相关工作

1.1 多用户资源竞争下的服务迁移研究

早期研究多聚焦于单用户场景下的服务迁移决策优化。例如，Taleb等[7]提出了Follow-Me Cloud框架，以平衡迁移成本和用户体验。Wang等[8]设计了服务迁移框架Mig-RL，以最小化总服务成本为目标。然而，这些方法通常假设边缘节点资源无限，忽略了多用户并发迁移所引发的资源竞争问题。在高密度车联网环境中，若多辆车辆同时迁移至同一边缘节点，极易导致计算资源过载，进而引发迁移失败或QoS劣化。近期已有部分研究关注多用户服务迁移场景。例如，Kang等[9]提出了一种基于多智能体深度强化学习的任务迁移方法，综合考虑多车辆并发迁移决策与边缘节点的有限计算资源约束。

1.2 服务迁移动态环境建模

现有研究通常依赖历史轨迹或交通流模型预测车辆移动性以辅助主动迁移决策[10]。然而，在多

用户场景下，移动性预测可能存在偏差[11]。若完全依赖预测结果执行迁移，可能引入不必要的开销。此外，仅有少量研究考虑了边缘节点因故障或突发高负载而临时不可用的情况对迁移决策的影响。例如，Tuli 等[12]基于边缘计算环境故障预测模型制定服务迁移决策，以提升 QoS。Ma 等[13]提出了一种结合移动性预测和冗余副本的容错服务迁移方法。

1.3　服务迁移决策算法发展与挑战

在决策算法设计方面，Lyapunov 优化、粒子群算法及遗传算法等传统方法[14-16]虽被广泛应用，但计算复杂度较高，在高度动态的车辆边缘网络环境中，上述方法易陷入局部最优，难以应对高维状态空间与实时决策需求。强化学习（reinforcement learning, RL）因其在处理序列决策与不确定性环境方面的优势，已被用于求解服务迁移问题。例如，Peng 等[17]采用 DQL 方法学习服务迁移策略。然而，在大规模 MEC 场景中，候选节点数量庞大，易导致动作空间维度爆炸，且易受 Q 值过估计影响，难以收敛至最优策略。为克服 DQL 在高维动作空间中的局限，以近端策略优化（proximal policy optimization, PPO）[18]为代表的 Actor-Critic 架构被广泛采用[19]。然而，在实际部署中，单智能体集中式决策方法依赖的全局状态难以实时获取，且决策复杂度随车辆数量增长而显著上升，导致可扩展性受限。部分研究采用多智能体强化学习（multi-agent reinforcement learning, MARL）求解服务迁移问题，其中每个车辆用户或 ES 作为独立智能体，仅基于局部观测进行自主决策。例如，Cui 等[20]将多用户服务迁移问题建模为去中心化部分可观测马尔可夫决策过程（decentralized partially observable Markov decision process, Dec-POMDP），并提出一种深度确定性策略梯度算法求解。尽管多智能体方法能够通过集中式训练与分布式执行（centralized training and distributed execution, CTDE）框架高效学习与优化策略，但其集中式 Critic 网络在服务迁移等强约束场景中易受极端负奖励干扰，导致价值估计偏差与训练不稳定。此外，Critic 网络的输入维度随智能体数量线性增长，制约了该类算法的可扩展性。近期，大语言模型领域的组相对策略优化（group relative policy optimization, GRPO）算法[21]通过摒弃 Critic 网络，并基于组内折扣回报的相对排序构建策略更新信号。该算法不依赖绝对价值估计，对奖励尺度变化及极端惩罚具有天然鲁棒性，同时避免了 Critic 网络的训练开销。受此启发，本文将 GRPO 算法拓展至多用户服务迁移场景，提出了 MAGRPO 算法，该算法可有效应对高维动作空间、极端奖励波动及多用户资源竞争等挑战，提升训练稳定性与可扩展性。

2　系统模型和问题构建

2.1　系统模型

如图 1 所示，考虑一个异构多用户 MEC 系统，该系统由一个 MEC 控制器、M 个 ES 和 N 个 ICV 用户组成。其中，车辆用户集合 $\mathcal{U}=\{1,2,\cdots,N\}$ 在由 ES 集合 $\mathcal{M}=\{1,2,\cdots,M\}$ 覆盖的地理区域中移动，每个 ES 与一个基站（base station, BS）共址部署，ES 之间通过回程链路相互连接；BS 接收来自车辆的服务请求，并将其转发至对应的 ES 进行处理。

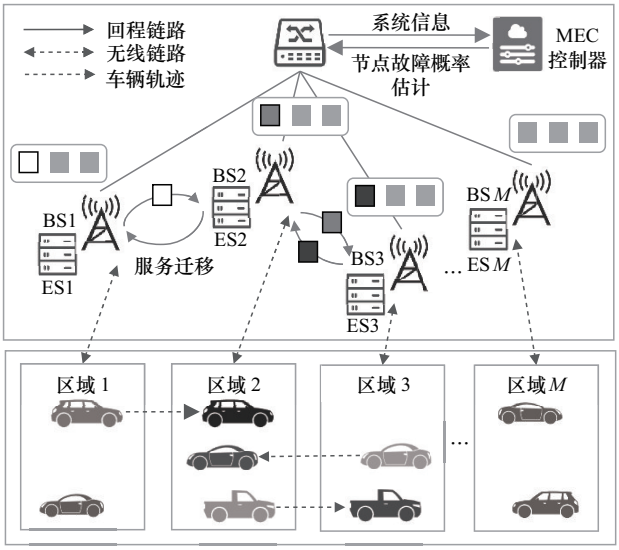


图 1　智能网联车辆服务迁移场景

考虑离散时隙模型，其中车辆用户的位置在每个时隙 t 开始时更新，并通过最近的 BS 与对应的 ES 建立通信，其中 $\mathcal{T}=\{1,2,\cdots,T\}$，每个时隙的持续时间为 τ。在时隙 $t\in\mathcal{T}$，与用户 $u\in\mathcal{U}$ 直连的边缘节点称为接入节点，记为 $m^a_{u,t}$。正在为用户 u 提供服务的节点称为服务节点，记为 $m^s_{u,t}$。

由于车载计算资源有限，移动车辆用户 u 将其运行过程中产生的计算任务卸载至边缘节点，由服务所在节点处理该任务并返回结果。在此过程中，受限于单一边缘服务器的有限覆盖范围及车辆的高

移动性，用户 u 可能驶出当前服务节点的通信覆盖范围，尽管计算任务可通过边缘服务器之间的回程链路传输，但仍可能引入额外通信时延，甚至导致服务中断，难以保障服务连续性。因此，为了保障车辆用户移动过程中智能车载服务的QoS，有必要实施服务迁移，以降低服务时延并控制迁移过程中因数据传输引入的额外开销。

2.1.1 时延模型

服务总时延由以下三部分组成。

1) 计算时延 $T_{u,t}^{\text{comp}}$

对于用户 u，设其在时隙 t 将计算任务卸载到边缘服务器 m，任务数据量为 $\text{data}_{u,t}^{o}$，所需处理密度为 κ，则计算该任务所需的CPU周期数为 $c_{u,t}=\text{data}_{u,t}^{o}\kappa$。假设边缘服务器以恒定计算能力 f^m 处理任务，其中为每个任务分配的计算资源与其所需的CPU周期数成比例，令 w_t^m 表示时隙 t 边缘服务器 m 的计算负载，则用户 u 的计算任务在时隙 t 获得的计算资源为 $f_{u,t}^m=f^m\dfrac{c_{u,t}}{w_t^m+c_{u,t}}$。计算时延定义为

$$T_{u,t}^{\text{comp}}=\frac{c_{u,t}}{f_{u,t}^m}=\frac{w_t^m+c_{u,t}}{f^m} \tag{1}$$

2) 通信时延 $T_{u,t}^{\text{comm}}$

通信时延由接入时延和回程时延组成。接入时延 $T_{u,t}^{\text{acc}}$ 由车辆用户到接入节点的任务卸载过程产生，可定义为

$$T_{u,t}^{\text{acc}}=\frac{\text{data}_{u,t}^{o}}{\rho_t} \tag{2}$$

其中，ρ_t 为用户向接入节点卸载任务的无线上行传输速率。回程时延 $T_{u,t}^{\text{bh}}$ 由接入节点和服务节点之间的回程链路传输过程产生，如果服务节点即为用户当前接入节点，则不存在回程时延。记用户 u 在时隙 t 接入的边缘服务器为 $m_{c(1\leqslant c\leqslant M)}\in\mathcal{M}$，当前时隙服务所在的边缘服务器为 $m_{j(1\leqslant j\leqslant M)}$。用 $h_{c,j}$ 表示 m_c 和 m_j 之间的跳数距离，若 $c=j$，则 $h_{c,j}=0$。否则，用户 u 产生的计算任务从 m_c 传输到 m_j 进行处理，产生回程时延。回程时延定义为

$$T_{u,t}^{\text{bh}}=\begin{cases}0, & h_{c,j}=0\\ \dfrac{\text{data}_{u,t}^{o}}{\eta^{\text{bh}}}+\sigma^{\text{bh}}h_{c,j}, & h_{c,j}\neq 0\end{cases} \tag{3}$$

其中，η^{bh} 为回程链路传输速率，$\sigma^{\text{bh}}>0$ 为回程链路平均每跳引入的时延开销。

3) 迁移时延 $T_{u,t}^{\text{mig}}$

记用户 u 在时隙 $t-1$ 的服务实例所在的边缘服务器为 $m_{i(1\leqslant i\leqslant M)}\in\mathcal{M}$，时隙 t 的服务迁移决策为 $m_{j(1\leqslant j\leqslant M)}$。类似式(4)，用 $d_{i,j}$ 表示服务实例从上一时隙服务所在的源节点迁移到当前时隙服务决策的目标节点所需经过的跳数距离。迁移时延定义为

$$T_{u,t}^{\text{mig}}=\begin{cases}0, & d_{i,j}=0\\ \dfrac{\text{data}_{u,t}^{s}}{\eta^{\text{bh}}}+\sigma^{\text{mig}}d_{i,j}, & d_{i,j}\neq 0\end{cases} \tag{4}$$

其中，$\text{data}_{u,t}^{s}$ 表示用户 u 在时隙 t 迁移的服务数据量，$\sigma^{\text{mig}}>0$ 表示迁移链路平均每跳引入的时延开销。

2.1.2 能耗模型

在服务迁移过程中，数据传输、相关资源配置及服务初始化等操作均会产生能耗。其中，数据传输能耗取决于迁移数据量、网络拓扑和传输距离；目标边缘服务器在启动与配置新服务实例时亦需消耗能量。此外，服务迁移过程中还可能引入状态同步等额外开销。鉴于上述因素种类繁多且难以精确量化，本文参考文献[22]进行简化，将迁移能耗归纳为迁移动作能耗（与是否执行迁移相关）和迁移过程能耗（与迁移耗时成正比）。其中，执行单次迁移动作能耗为 $E_{u,t}^{\text{act}}$，迁移过程能耗定义为

$$E_{u,t}^{\text{pro}}=\begin{cases}0, & d_{i,j}=0\\ p_{u,t}\left(\dfrac{\text{data}_{u,t}^{s}}{\eta^{\text{bh}}}+\sigma^{\text{mig}}d_{i,j}\right), & d_{i,j}\neq 0\end{cases} \tag{5}$$

其中，$p_{u,t}$ 是迁移能耗系数，表示时隙 t 用户 u 的计算任务在边缘服务器之间执行服务迁移时单位时间产生的能耗，其值与传输功率、链路质量及边缘服务器处理能耗等因素相关。若当前时隙未发生服务迁移，则迁移能耗为0。

2.1.3 边缘节点可用性模型

在实际车联网环境中，边缘服务器可能因硬件故障、软件崩溃或突发高负载而临时不可用。为提升服务迁移策略的鲁棒性，本文显式建模边缘节点的动态可用性。

设边缘节点 m 在时隙 t 的真实可用性状态为二元随机变量 $A_{m,t}\in\{0,1\}$，其中 $A_{m,t}=1$ 表示节点正常可用，$A_{m,t}=0$ 表示节点发生故障不可用。假设各时隙节点的可用性相互独立，且发生故障的真实概率为固定值 $p_m^{\text{fail}}=\mathbb{P}(A_{m,t}=0)$，该值由节点长期运

行特性（如硬件可靠性、部署环境等）决定。

为支持车辆进行前瞻性服务迁移决策，MEC控制器持续收集各边缘节点的历史可用性记录，并在每个时隙末计算截至当前时隙的经验故障频率作为对p_m^{fail}的统计估计，该估计值通过控制信道定期广播给所有车辆。

$$\hat{p}_{m,t}^{\text{fail}} = \frac{1}{t}\sum_{\tau=1}^{t}\mathbb{I}\{A_{m,\tau}=0\} \tag{6}$$

由于信息广播存在时延，当车辆u在时隙t进行决策时，仅能获取上一时隙末发布的故障概率统计$\hat{p}_{m,t-1}^{\text{fail}}$。车辆虽无法获知当前真实可用性$A_{m,t}$，但可利用该滞后的故障概率统计进行风险感知决策，从而在部分可观测条件下提升迁移可靠性。

在服务迁移执行阶段，若车辆u选择的目标节点s恰好处于故障状态（即$A_{m,t}=0$），则迁移失败，服务中断，需退回至原服务节点或触发紧急重调度，造成显著性能损失。为引导策略网络主动规避高风险节点，本文在奖励函数中引入强惩罚机制，当车辆选择故障节点时，施加一个较大的负奖励，具体形式在第3.1节详细叙述。

2.2　问题形式化

基于上述系统模型，本文将动态车联网环境下的服务迁移决策问题形式化为一个带约束的多用户联合优化问题。其核心目标是在满足边缘节点资源约束与迁移可行性的前提下，协同最小化服务总时延与迁移能耗，同时保证较高的迁移成功率。本文采用混合建模策略，将边缘节点计算资源上限和迁移决策域有效性作为显式硬约束，迁移开销、节点故障风险等潜在限制因素则通过在奖励函数中引入相应的惩罚项进行软约束建模，引导策略在训练过程中自主权衡性能与风险。

令$\boldsymbol{a}_t=\{a_{u,t}\}_{u\in\mathcal{U}}$表示时隙$t$的联合迁移决策，其中$a_{u,t}=m\in\mathcal{M}$表示为用户$u$在时隙$t$选择的服务迁移目标。用户$u$在时隙$t$的服务总时延为

$$T_{u,t}^{\text{tot}} = T_{u,t}^{\text{comp}} + T_{u,t}^{\text{comm}} + T_{u,t}^{\text{mig}} \tag{7}$$

其中，各项时延由第2.1.1节定义。用户u在时隙t的迁移能耗为

$$E_{u,t}^{\text{tot}} = E_{u,t}^{\text{act}} + E_{u,t}^{\text{pro}} \tag{8}$$

优化目标为在时间范围$\mathcal{T}$内，最小化所有用户的综合成本之和，即所有用户的服务总时延与迁移能耗的加权和。

$$\begin{aligned}
\text{P:}\quad & \min \sum_{t=1}^{T}\sum_{u=1}^{N}\left(\omega_1 T_{u,t}^{\text{tot}} + \omega_2 \beta E_{u,t}^{\text{tot}}\right)\\
\text{s.t.}\quad & \text{C1:}\ \sum_{u:a_{u,t}=m} c_{u,t} \le f^m\tau,\ \forall m\in\mathcal{M}, t\in\mathcal{T}\\
& \text{C2:}\ a_{u,t}\in\mathcal{M},\ \forall u\in\mathcal{U}, t\in\mathcal{T}
\end{aligned} \tag{9}$$

其中，$\omega_1,\omega_2\ge 0$分别表示迁移时延与能耗的权重系数，二者满足$\omega_1+\omega_2=1$，β表示数值归一化系数。在实际的智能网联车辆服务迁移问题中，该优化问题需满足以下约束条件。约束C1表示边缘服务器资源总量，任意边缘服务器m在时隙t的总计算负载不得超过其最大计算能力。约束C2表示迁移决策，每个车辆用户u在时隙t的迁移决策$a_{u,t}=m$必须是一个有效的边缘节点索引。

综上，问题P是一个带硬约束的长期组合优化问题，其核心挑战在于系统状态（如车辆位置、节点可用性和负载）高度动态变化，且联合动作空间维度随用户数呈指数增长。本文提出的MAGRPO算法旨在通过组相对策略优化机制有效应对高维动作空间与极端奖励引发的训练不稳定问题，实现自适应的最优迁移决策。

3　基于MAGRPO的服务迁移算法

3.1　Dec-POMDP建模

服务迁移本质上是一个序贯决策问题，系统下一时刻的状态仅由当前时刻状态和所采取的迁移动作决定，可自然建模为MDP。但在实际车联网场景中，全局状态难以实时获取，若仅由单一控制器集中决策所有车辆计算任务的迁移目标，不仅通信开销巨大，而且难以满足低时延响应需求。更贴合实际的建模方式是将每辆车视为一个智能体，仅基于局部观测进行分布式决策。因此，本文将在动态车联网环境中的服务迁移问题建模为一个Dec-POMDP，记为$<\mathcal{U},\mathcal{S},\{\mathcal{O}_u\}_{u\in\mathcal{U}},\{\mathcal{A}_u\}_{u\in\mathcal{U}},\mathcal{P},\mathcal{R},\gamma>$，其中各项分别代表智能体集合、全局状态空间、局部观测空间、动作空间、状态转移函数、联合奖励函数和折扣因子。问题P转化为每个智能体在每个时隙寻找一个迁移动作，以最大化决策时间窗口内的累积奖励。

1) 智能体集合$\mathcal{U}$：每个智能网联车辆$u\in\mathcal{U}$作为一个智能体，自主决策其服务迁移目标。

2) 全局状态空间$\mathcal{S}$：状态$\boldsymbol{s}_t\in\mathcal{S}$包含所有车辆的接入节点和服务节点信息、所有边缘节点的计算负载和故障概率。

3) 局部观测空间$\{\mathcal{O}_u\}$：车辆智能体u无法观测全局状态空间$\mathcal{S}$，仅能获取局部状态信息。智能体u在时隙t的观测为

$$\boldsymbol{o}_{u,t}=\left(m_{u,t}^{a},m_{u,t}^{s},\{\hat{p}_{m,t-1}^{\text{fail}}\}_{m\in\mathcal{M}},\{\hat{w}_{t-1}^{m}\}_{m\in\mathcal{M}}\right) \tag{10}$$

其中，$m_{u,t}^{a}$和$m_{u,t}^{s}$分别为当前时隙的接入节点和服务节点信息，$\hat{p}_{m,t-1}^{\text{fail}}$和$\hat{w}_{t-1}^{m}$分别为通过MEC控制器广播的上一时隙节点故障概率估计与负载信息。

4) 动作空间$\{\mathcal{A}_u\}$：每个车辆智能体u的动作$a_{u,t}\in\mathcal{A}_u=\mathcal{M}$表示选择的服务迁移目标节点。若$a_{u,t}=m_{u,t}^{s}$，则表示不迁移。所有智能体的动作共同构成一个联合动作$\boldsymbol{a}_t=\{a_{u,t}\}_{u\in\mathcal{U}}$。

5) 状态转移函数$\mathcal{P}$：$\mathcal{P}(\cdot\,|\,\boldsymbol{s}_t,\boldsymbol{a}_t)$表示给定当前全局状态和联合动作得到的下一时隙状态的概率分布，由车辆移动性、边缘节点资源变化、节点故障情况等动态因素共同决定。

6) 联合奖励函数$\mathcal{R}$：在时隙t，环境返回一个所有智能体共享的联合奖励r_t，该奖励定义为所有车辆服务迁移综合成本的负值之和，并包含资源超载与节点故障惩罚，以软化原问题的硬约束条件C1并提升迁移可靠性。

具体而言，令$m_u=a_{u,t}$表示车辆u在时隙t选择的迁移目标节点，则车辆u在时隙t的联合奖励为

$$r_{u,t}=-\left(\omega_1 T_{u,t}^{\text{tot}}+\omega_2\beta E_{u,t}^{\text{tot}}\right)-\lambda_{\text{over}}\max\left(0,\frac{L_{m_u,t}-f^{m_u}}{f^{m_u}}\right)-\lambda_{\text{fail}}\mathbb{I}\{A_{m_u,t}=0\} \tag{11}$$

其中，$\lambda_{\text{over}},\lambda_{\text{fail}}>0$分别为资源超载与节点故障的惩罚系数，当惩罚系数足够大时，可有效抑制约束违反行为，从而近似满足原始硬约束条件；$L_{m_u,t}=\sum_{u'\in\mathcal{U}}c_{u',t}\mathbb{I}\{a_{u',t}=m_u\}$表示该节点在时隙$t$的负载。时隙$t$所有智能体的联合奖励为

$$r_t=\sum_{u\in\mathcal{U}}r_{u,t} \tag{12}$$

7) 折扣因子$\gamma\in(0,1]$：用于平衡短期与长期收益。

通过上述设计，问题P被近似转化为一个无约束Dec-POMDP，其目标是最大化期望累积奖励。

$$\max_{\{\pi_u\}}\mathbb{E}_{\pi}\left[\sum_{t=1}^{T}r_t\right] \tag{13}$$

3.2 基于MAGRPO的服务迁移算法

为高效求解第3.1节构建的Dec-POMDP模型，本文提出了一种面向MEC场景的MAGRPO算法。该算法通过引入组相对优势计算，在多智能体环境下稳定高效地实现各智能体策略的协同优化。

GRPO是由DeepSeek提出的一种高效强化学习算法，旨在克服PPO在复杂数学推理任务中因需同步训练大规模Critic网络而导致的高内存开销与训练不稳定问题。与传统Actor-Critic方法不同，GRPO摒弃显式Critic网络，通过在相同初始状态下多次执行策略采样，并基于组内折扣回报的相对排序构建策略更新信号。该方法不需要训练Critic网络，对高方差或极端奖励具有天然鲁棒性，同时有效降低了模型复杂度与训练开销。受此启发，本文将GRPO算法拓展至MEC服务迁移环境，以应对高维动作空间、多用户资源竞争及节点可用性动态变化等挑战。

MAGRPO算法框架如图2所示。在每轮训练中，对于相同的环境初始状态，每个智能体基于当前策略$\pi_{\theta_u}(\cdot\,|\,\boldsymbol{h}_{u,t}^{(g)})$采样$G$次，形成$G$条联合轨迹，每条轨迹由$T$个时隙构成。所有轨迹共享相同的初始状态$s_0$。其中$\boldsymbol{h}_{u,t}^{(g)}=(\boldsymbol{o}_{u,1}^{(g)},a_{u,1}^{(g)},\cdots,\boldsymbol{o}_{u,t-1}^{(g)},a_{u,t-1}^{(g)},\boldsymbol{o}_{u,t}^{(g)})$表示第$g$条轨迹中智能体$u$在时隙$t$的局部观测–动作历史。$a_{u,t}^{(g)}\in\mathcal{A}_u$为智能体$u$在第$g$次采样的第$t$个时隙生成的服务迁移动作。

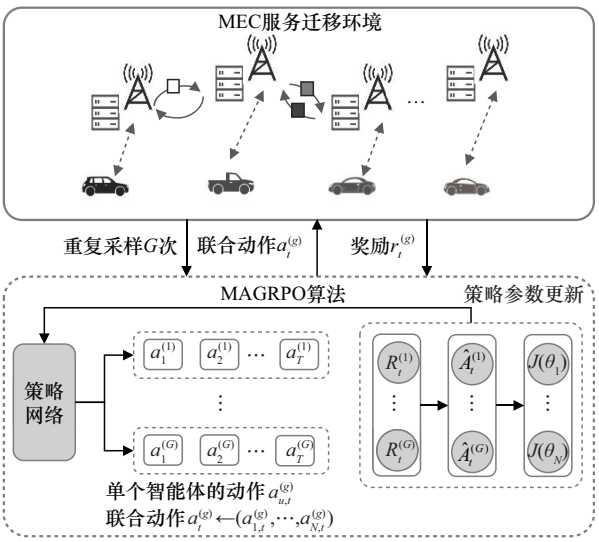


图2 MAGRPO算法框架

第g条轨迹在时隙t的联合动作记为$\boldsymbol{a}_t^{(g)}=$

$\left(a_{1,t}^{(g)},a_{2,t}^{(g)},\cdots,a_{N,t}^{(g)}\right)$。该联合动作被提交至MEC服务迁移环境执行，该环境根据上一时隙的状态$\boldsymbol{s}_{t-1}^{(g)}$和联合动作$\boldsymbol{a}_t^{(g)}$按状态转移函数$\mathcal{P}$演化，并返回一个所有智能体共享的联合奖励$r_t^{(g)}$。在MEC服务迁移环境中，系统在每个时隙均可根据观测到的时延、能耗等指标计算即时奖励，从而获得密集的时序反馈信号。为充分利用这一特性，本文对第g条轨迹的每个时隙t计算从该时隙开始的折扣累积回报，如式(14)所示。

$$R_t^{(g)}=\sum_{\tau=t}^{T}\gamma^{\tau-t}r_\tau^{(g)} \tag{14}$$

其中，$\gamma\in(0,1]$为折扣因子。

MAGRPO算法采用上述回报作为策略评估的基础信号。具体而言，基于G条轨迹的折扣累积回报集合为$\{R_t^{(1)},R_t^{(2)},\cdots,R_t^{(G)}\}$，计算其均值与标准差，分别表示为

$$\mu_t=\frac{1}{G}\sum_{g=1}^{G}R_t^{(g)} \tag{15}$$

$$\sigma_t=\sqrt{\frac{1}{G}\sum_{g=1}^{G}\left(R_t^{(g)}-\mu_t\right)^2} \tag{16}$$

进而计算标准化的组相对优势，如式(17)所示。

$$\hat{A}_t^{(g)}=\frac{R_t^{(g)}-\mu_t}{\sigma_t+\epsilon} \tag{17}$$

其中，$\epsilon>0$为数值稳定常数。该优势函数通过标准化保留轨迹间的相对优劣，降低策略梯度估计的方差，并为多智能体提供精细的时序信用分配信号。第g条轨迹的组相对优势$\hat{A}_t{}^{(g)}$基于从时隙t开始的折扣累积回报计算，并被该轨迹中所有智能体共享，从而在无显式Critic网络的前提下实现隐式的协同信用分配，准确评估每个智能体在每个时隙的决策对系统长期性能的贡献。

最终，每个智能体u的策略网络参数θ_u通过逐时隙策略更新进行优化。具体而言，对于每个时隙t，优化目标函数如式(18)所示。

$$J_t(\theta_u)=\frac{1}{G}\sum_{g=1}^{G}\min\left(\rho_{u,t}^{(g)}\hat{A}_t^{(g)},L_{\text{clip}}^{(g,t)}\right) \tag{18}$$

其中，$\rho_{u,t}^{(g)}$为重要性采样比率，用于衡量新旧策略下动作概率的差异，定义为

$$\rho_{u,t}^{(g)}=\frac{\pi_{\theta u}(a_{u,t}^{(g)}\left|h_{u,t}^{(g)}\right.)}{\pi_{\theta u^{\text{old}}}(a_{u,t}^{(g)}\left|h_{u,t}^{(g)}\right.)} \tag{19}$$

$L_{\text{clip}}^{(g,t)}$为裁剪项，定义为

$$L_{\text{clip}}^{(g,t)}=\text{clip}\left(\rho_{u,t}^{(g)},1-\varepsilon,1+\varepsilon\right)\hat{A}_t^{(g)} \tag{20}$$

其中，$\text{clip}(x,a,b)=\min(\max(x,a),b)$为裁剪函数，$\varepsilon$为裁剪范围超参数。该目标函数在鼓励策略向高优势方向更新的同时，通过裁剪机制限制策略变化幅度，保障训练稳定性。在实现过程中，对每个时隙t计算组相对目标函数$J_t(\theta_u)$，并在一轮训练结束时对其在T个时隙上取平均值，得到最终优化目标函数$J(\theta_u)=\frac{1}{T}\sum_{t=1}^{T}J_t(\theta_u)$。MAGRPO算法的完整流程如图3和算法1所示。

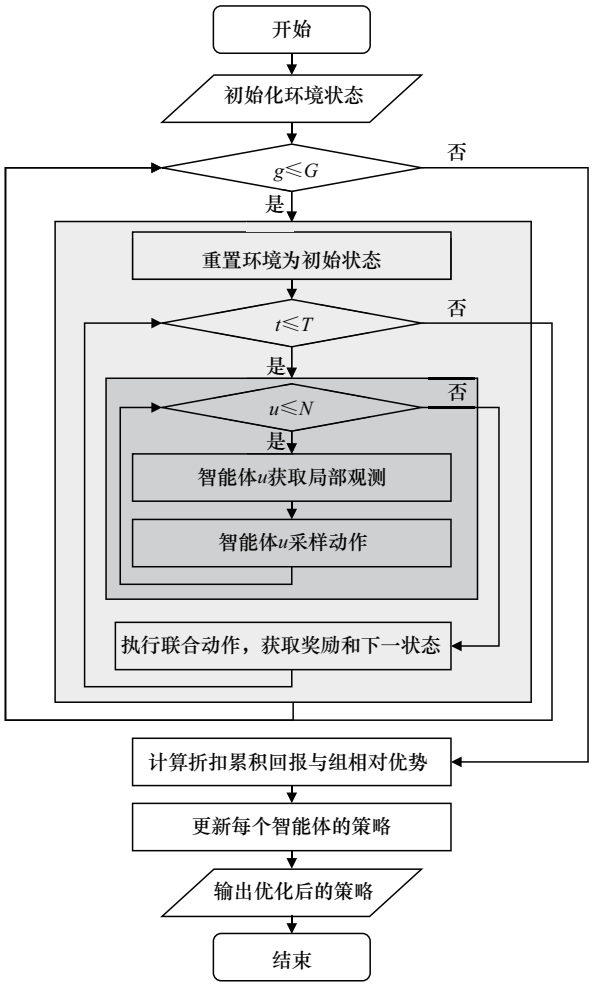


图3　MAGRPO算法流程

算法1　MAGRPO

输入　智能体集合$\mathcal{U}$，初始策略参数$\{\theta_u\}_{u\in\mathcal{U}}$，采样数$G$，最大时隙数$T$，折扣因子$\gamma$，裁剪范围超参数$\varepsilon$，学习率$\alpha$，训练轮数$N_{\text{ep}}$

输出　优化后的策略参数$\{\theta_u\}_{u\in\mathcal{U}}$

1)for 训练轮次 $e = 1$ to N_{ep} do
2) 从环境采样共享初始状态 $\boldsymbol{s}_0$
3) 保存旧策略 $\theta_u^{\text{old}} \leftarrow \theta_u, \forall u \in \mathcal{U}$
4) for 轨迹 $g = 1$ to G do
5) 设置当前状态 $\boldsymbol{s}^{(g)} \leftarrow \boldsymbol{s}_0$
6) for 时隙 $t = 1$ to T do
7) for 智能体 $u \in \mathcal{U}$ do
8) 获取当前观测 $\boldsymbol{o}_{u,t}^{(g)}$
9) //构建决策历史 $\boldsymbol{h}_{u,t}^{(g)}$
10) if $t = 1$ then
11) $\boldsymbol{h}_{u,t}^{(g)} \leftarrow (\boldsymbol{o}_{u,t}^{(g)})$
12) else
13) $\boldsymbol{h}_{u,t}^{(g)} \leftarrow (\boldsymbol{h}_{u,t-1}^{(g)}, a_{u,t-1}^{(g)}, \boldsymbol{o}_{u,t}^{(g)})$
14) end if
15) 采样当前动作 $a_{u,t}^{(g)} \sim \pi_{\theta_u}(\cdot \mid \boldsymbol{h}_{u,t}^{(g)})$
16) end for
17) 执行联合动作 $\boldsymbol{a}_t^{(g)} \leftarrow (a_{1,t}^{(g)}, \cdots, a_{N,t}^{(g)})$
18) 从环境获取奖励 $r_t^{(g)}$ 和下一状态 $\boldsymbol{s}'$
19) 更新当前状态 $\boldsymbol{s}^{(g)} \leftarrow \boldsymbol{s}'$
20) end for
21) end for
22) //计算折扣累积回报与组相对优势
23) for 时隙 $t = T$ down to 1 do
24) for 轨迹 $g = 1$ to G do
25) $R_t^{(g)} \leftarrow \sum_{\tau=t}^{T} \gamma^{\tau-t} r_\tau^{(g)}$
26) end for
27) $\mu_t \leftarrow \frac{1}{G}\sum_{g=1}^{G} R_t^{(g)}, \sigma_t \leftarrow \sqrt{\frac{1}{G}\sum_{g=1}^{G}(R_t^{(g)} - \mu_t)^2}$
28) for 轨迹 $g = 1$ to G do
29) $\hat{A}_t^{(g)} \leftarrow \frac{R_t^{(g)} - \mu_t}{\sigma_t + \epsilon}$
30) end for
31) end for
32) //策略更新
33) for 智能体 $u \in \mathcal{U}$ do
34) $\nabla J(\theta_u) \leftarrow 0$
35) for 时隙 $t = 1$ to T do
36) $\nabla J_t(\theta_u) \leftarrow 0$
37) for 轨迹 $g = 1$ to G do
38) 计算 $\rho_{u,t}^{(g)}$ 和 $L_{\text{clip}}^{(g,t)}$
39) $\nabla J_t(\theta_u) \leftarrow \nabla J_t(\theta_u) + \nabla_{\theta_u} L_{\text{clip}}^{(g,t)}$
40) end for
41) $\nabla J_t(\theta_u) \leftarrow \frac{1}{G}\nabla J_t(\theta_u)$
42) $\nabla J(\theta_u) \leftarrow \nabla J(\theta_u) + \nabla J_t(\theta_u)$
43) end for
44) $\nabla J(\theta_u) \leftarrow \frac{1}{T}\nabla J(\theta_u)$
45) $\theta_u \leftarrow \theta_u + \alpha \nabla J(\theta_u)$
46) end for
47)end for
48)return $\{\theta_u\}_{u \in \mathcal{U}}$

4 仿真分析

4.1 实验设置

4.1.1 仿真环境

本文采用意大利罗马的真实出租车 GPS 轨迹数据集[23]模拟车辆移动性。该数据集包含 320 辆出租车的驾驶数据，时间跨度从 2014 年 2 月 1 日至 2014 年 3 月 2 日（共 30 天），每条记录都包含唯一的出租车 ID、时间戳和轨迹坐标。

如图 4 所示，本文将经纬度在 [41.856° N, 12.442°E]至[41.928°N, 12.538E]之间的市中心区域作为实验场景。为便于分析，选取其中车辆轨迹较为密集的 4 km×4 km 的地理区域，在其中均匀部署 16 个边缘服务器，每个边缘服务器与一个基站共址，覆盖以它为中心的 1 km×1 km 的网格。假设每个边缘服务器都可以与其相邻的节点进行通信，边缘服务器之间通过带宽为 500 Mbit/s 的回程链路互联，每跳固定时延为 0.02 s。每个边缘服务器的计算能力为 128 GHz（等效于 4 台 16 核服务器，每核 2 GHz）。从上述区域内随机选取了高峰时段的 450 条轨迹数据，并对原始轨迹数据进行标准化处理，仅保留长度大于或等于 100 个时隙的车辆轨迹，并截取前 100 个连续时隙（每个时隙长度为 3 min），用于模拟智能网联车辆的移动轨迹。

参考文献[24]，本文假设服务大小在[0.5, 100] MB 内服从均匀分布。在每个时隙，每辆汽车的车载服务生成一个计算任务，并在该时隙内完成计算。假设计算任务大小在[0.5,20] MB 内服从均匀分布。考虑到不同任务的计算复杂度差异较大，假设每个任务所需的处理密度在[100, 4 000] cycle/bit 内均匀分布。为模拟现实场景中的不确定性，本文假设各

边缘节点的故障概率在(0, 0.05]内服从均匀分布。仿真环境参数设置如表1所示。

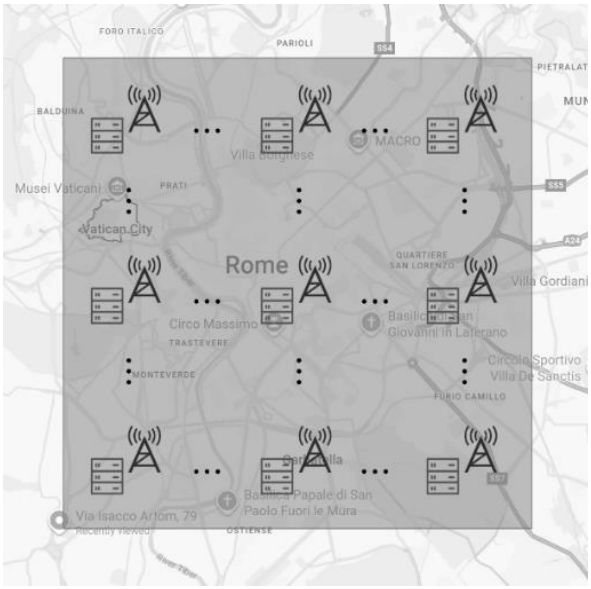


图4　仿真实验边缘节点分布

表1　　仿真环境参数

参数	取值
边缘节点数量M/个	16
车辆用户数量N/个	{40, 60, 80}
边缘节点最大计算能力f^m/GHz	128
上行链路传输速率ρ_t/(Mbit·s^{-1})	{60, 48, 36, 24, 12}
回程链路传输速率η^{bh}/(Mbit·s^{-1})	500
回程时延传播系数σ^{bh}/(s·hop^{-1})	0.02
迁移时延传播系数σ^{mig}/(s·hop^{-1})	U[1.0,3.0]
服务数据大小$\mathrm{data}^s_{u,t}$/MB	U[0.5,100]
任务数据大小$\mathrm{data}^o_{u,t}$/MB	U[0.5,20]
任务所需处理密度κ/(cycle·bit^{-1})	U[100,4 000]
单位时间迁移能耗$p_{u,t}$/W	U[20,40]
边缘节点故障概率p_m^{fail}	U(0,0.05]
时延权重系数ω_1	0.5
能耗归一化系数β	0.15

所有仿真实验在配备NVIDIA RTX 3090 GPU和Intel Xeon Gold 6242R CPU的服务器上完成，使用PyTorch实现并训练MAGRPO的策略网络。该网络是一个多层感知机，包含4个隐藏层，隐藏单元数依次为512、512、256和256，隐藏层之间采用ReLU激活函数。优化器为Adam，学习率α=1×10^{-2}，折扣因子γ = 0.99，裁剪参数ε = 0.2。

4.1.2　评估指标与基准方法

本文实验采用以下性能指标评估所提方法的有效性。

1) 服务总时延：用户请求从发起至服务响应完成所经历的端到端时延，包括计算时延、通信时延和迁移引入的额外时延。

2) 迁移能耗：服务迁移过程中因状态传输、上下文同步及目标节点资源预热等操作所消耗的能量，用于衡量迁移对系统能效的影响。

3) 服务过载率：位于过载边缘节点的服务实例数与总服务实例数之比，用于衡量算法在多用户资源竞争场景下的表现。

4) 迁移成功率：成功完成迁移的次数与总迁移尝试次数之比，用于衡量算法在边缘节点可用性动态变化环境下的鲁棒性。

本文选取以下基线方法与本文方法进行对比。

1) 总是迁移（always migrate, AM）[14]：在每个时隙，将服务迁移到当前用户接入的边缘节点。

2) 从不迁移（never migrate, NM）[24]：服务在整个时间窗口内始终驻留在初始部署的边缘节点，不受用户移动或负载变化的影响。

3) 贪心算法（Greedy）[25]：在每个时隙，基于当前系统状态计算所有候选边缘节点的综合成本，并选择成本最低者执行迁移。

4) 独立Q学习（independent Q-learning, IQL）[26]：每个用户作为独立智能体，利用深度Q网络近似其Q函数，并选择使Q值最大的迁移动作。

5) MAPPO[27]：将服务迁移建模为完全协作的多智能体决策问题，在训练阶段利用全局状态训练共享的集中式Critic网络，执行阶段各智能体仅基于本地观测独立决策。

6) 多智能体深度确定性策略梯度（multi-agent deep deterministic policy gradient, MADDPG）[28]：将服务迁移问题建模为多智能体马尔可夫决策过程，采用确定性策略梯度并通过CTDE框架优化各智能体策略。各智能体基于本地观测输出连续动作，并将其映射为离散迁移动作。

为确保实验公平性，所有RL方法采用相同的神经网络结构，并使用相同的学习率、批量大小与

训练步数进行优化。

4.2　实验结果分析

4.2.1　收敛性分析

1) 学习率α。MAGRPO算法在不同学习率下的收敛情况如图5所示。实验结果表明，学习率对策略优化的稳定性与收敛速度具有一定影响。过大的学习率（如1×10^{-1}）会导致策略更新幅度过大，容易陷入次优解；过小的学习率（如5×10^{-4}）则会导致较低的收敛速度。本文将MAGRPO算法中Actor的学习率设置为1×10^{-2}，以在收敛速度与稳定性之间取得良好平衡。

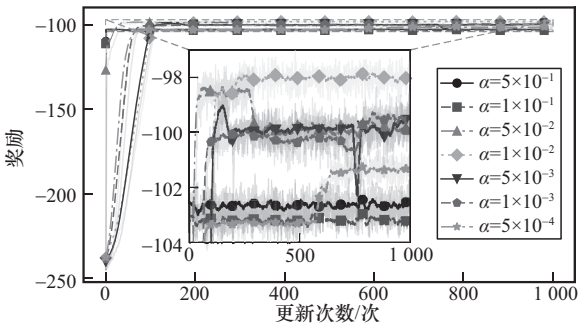


图5　不同学习率下MAGRPO算法的收敛情况

2) 组大小G。G控制在相同初始状态下并行采样的轨迹数量，直接影响组内回报相对排序的统计可靠性。在多智能体场景下，G的取值需与智能体数量N和环境复杂度协同调整。一方面，随着N增加，联合动作空间呈指数增长，需适当增大G，以保证组内采样覆盖策略分布的多样性。另一方面，本文服务迁移场景涉及多用户并发迁移、节点资源竞争与故障风险等，环境复杂度较高，因此需设置较大的G以抑制极端负奖励对策略的干扰。

值得注意的是，MAGRPO算法因不需要Critic网络，减少了超参数数量，有效降低了调参复杂度。同时，其核心超参数组大小G在合理范围内表现出良好的训练稳定性，这有助于提升MAGRPO算法在不同任务环境下的泛化能力。如图6所示，本文在$G\in\{4,8,16,32,64,128\}$内进行多次实验以确定最优组大小。结果表明，当$G<32$时，优势估计方差较大，模型性能未达到最佳；当$G>32$时，性能增益趋于饱和；当$G=32$时，算法在测试集上的平均累积奖励最高。因此，本文在后续实验中将组大小G固定为32。

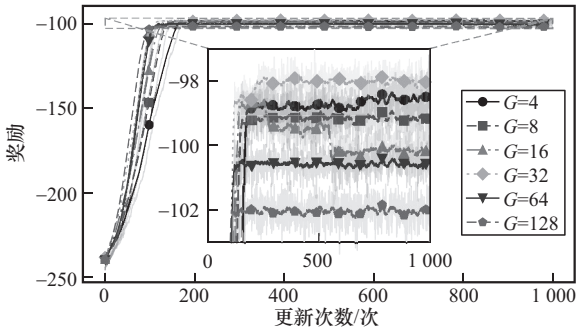


图6　不同组大小下MAGRPO算法的收敛情况

3) 智能体数量N。如图7所示，MAGRPO算法在不同智能体数量（$N=40,60,80$）下均展现出良好的收敛性能。所有实验配置在约400次策略更新内迅速收敛至高性能的稳定区间，且在后续训练过程中保持平稳，未出现明显的性能振荡、退化或发散现象。这表明MAGRPO算法的策略优化机制能够有效应对多智能体系统中因智能体数量变化带来的复杂性增长，在扩展至更大规模多智能体场景时，依然维持了高效的梯度估计与稳定的策略更新。

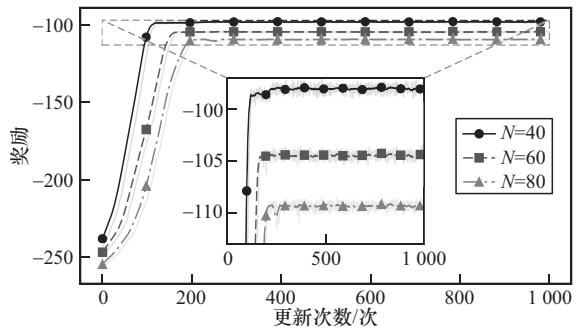


图7　不同智能体数量下MAGRPO算法的收敛情况

在不同智能体数量设置下，本文算法与第4.1.2节中各个基线方法在训练过程中奖励的收敛性如图8~图10所示。NM和Greedy均为基于规则的启发式策略，其奖励在训练过程中保持不变。相比之下，IQL、MAPPO、MADDPG和MAGRPO作为深度强化学习方法，均随更新次数增加逐步收敛并显著优于基线方法。其中，MAGRPO表现最优，得益于其组相对优势计算有效降低了策略更新方差，并增强了多智能体协同能力。以$N=40$为例，MAGRPO收敛后的平均累积奖励较次优方法MADDPG提升约14.8%，验证了本文方法的有效性。

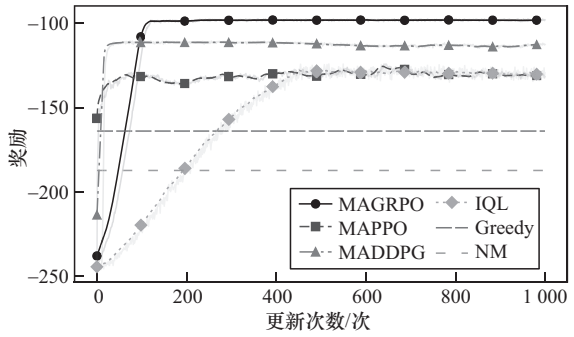


图8　不同算法的训练奖励曲线（$N = 40$）

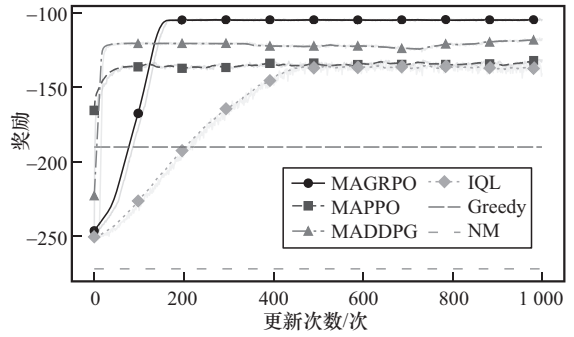


图9　不同算法的训练奖励曲线（$N = 60$）

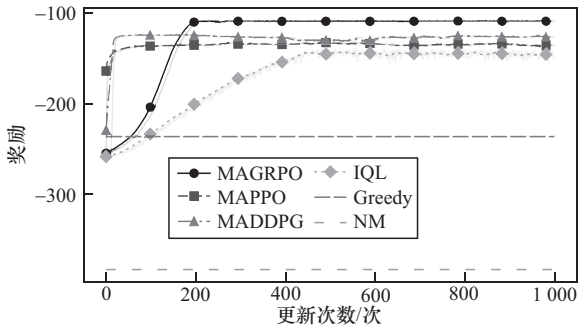


图10　不同算法的训练奖励曲线（$N = 80$）

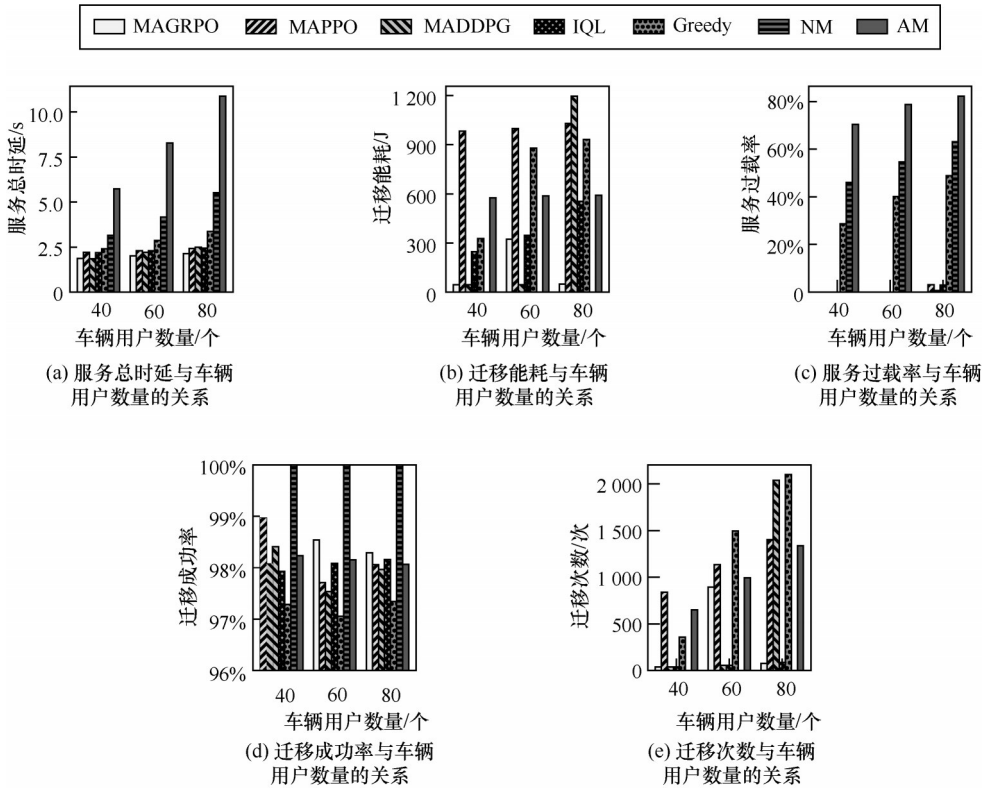
4.2.2　服务迁移指标对比

1) 服务总时延。服务总时延包括计算时延、通信时延和迁移引入的额外时延。如图11(a)所示，采用NM时，服务实例固定部署在初始节点，虽然不产生迁移时延，但车辆在移动过程中远离服务实例后，需经过多跳回程链路传输任务，导致通信时延增加，服务总时延较高。AM始终跟随车辆移动迁移服务实例，虽然有助于降低通信时延，但它忽视了多用户间的资源竞争，盲目跟随车辆移动性迁移容易导致边缘节点过载，从而增加计算时延，导致服务总时延最高。Greedy仅基于当前时隙的综合成本做出迁移决策，未考虑未来状态，导致其在多时隙场景中容易获得次优解。相比之下，MAGRPO通过组内轨迹的相对优势计算，有效协调多智能体迁移决策，在避免计算资源过载的同时兼顾通信效率，从而有效降低服务总时延。以$N = 80$为例，MAGRPO相比IQL、MAPPO和MADDPG在服务总时延上分别降低约12.2%、11.7%和14.1%。

2) 迁移能耗。迁移能耗反映迁移操作对系统能效的影响。如图11(b)所示，AM在车辆每次移出当前接入节点覆盖范围时都会触发迁移，盲目跟随车辆移动性而忽视迁移能耗，导致其产生较高的迁移能耗。Greedy由于缺乏长期规划能力，频繁触发低收益或冗余迁移，产生了较高的迁移能耗。MAGRPO通过组相对优势计算，能够准确识别高价值迁移时机，有效抑制无效迁移行为，迁移能耗较低。以$N = 80$为例，MAGRPO相比MAPPO、MADDPG和IQL在迁移能耗上分别降低约94.2%、95.9%和91.3%。

3) 服务过载率。如图11(c)所示，启发式方法的服务过载率显著高于强化学习方法。其中，AM因始终将服务迁移至用户当前接入的边缘节点，未考虑节点实时负载状态，易引发边缘节点过载。相比之下，MAPPO与MADDPG通过集中式Critic网络缓解了该问题。MAGRPO通过引入组内轨迹相对优势估计机制，在策略学习过程中有效利用包含过载惩罚的奖励信号，隐式捕捉多智能体迁移行为对系统负载的影响，从而引导智能体主动避开潜在过载节点。以$N = 80$为例，MAGRPO在实验中未观察到服务过载，相比之下，MAPPO和MADDPG的服务过载率分别为3.1%和0.7%，说明MAGRPO能够有效提升系统在高负载场景下的可靠性，保障QoS。

4) 迁移成功率：如图11(d)所示，以$N = 40$为例，IQL因采用独立学习机制而缺乏显式的多智能体协作，难以有效感知全局状态与节点风险，导致其迁移成功率（97.93%）低于其他协同策略。尽管MAPPO与MADDPG采用集中式Critic网络进行多智能体协调，但在存在极端负奖励（如节点突发故障）的场景下，其价值估计易受偏差影响，导致策略对高风险节点的规避能力不足，迁移成功率分别为98.1%与98.4%。MAGRPO通过标准化的组相对优势机制，天然抑制极端回报对策略梯度的扰动。同时，该机制鼓励智能体选择在群体中表现稳



(a) 服务总时延与车辆用户数量的关系　(b) 迁移能耗与车辆用户数量的关系　(c) 服务过载率与车辆用户数量的关系

(d) 迁移成功率与车辆用户数量的关系　(e) 迁移次数与车辆用户数量的关系

图 11　车辆用户数对各项指标的影响

健的迁移目标，从而提升迁移决策的可靠性，迁移成功率达到了 98.9%，相比 IQL、MAPPO 和 MADDPG，分别提升了约 1.0%、0.9% 和 0.6%。

5) 迁移次数：如图 11(b)和图 11(e)所示，各方法的迁移次数与其迁移能耗呈正相关。MAPPO 虽能通过集中式 Critic 网络协调多智能体决策，但其策略更新受全局价值估计影响，在节点故障或负载突变时易频繁迁移。相比之下，MAGRPO 通过组内相对优势机制精准识别高价值迁移时机，在保障低时延的同时有效抑制冗余迁移，从而降低迁移能耗。

4.2.3　参数敏感性分析

为验证 MAGRPO 在不同系统配置下的泛化能力与鲁棒性，本文进一步开展参数敏感性实验，考察以下两类关键系统参数变化对算法性能的影响：车辆用户数量和任务处理密度。

1) 车辆用户数量 N：如图 11(a)所示，各方法的服务总时延均随 N 增大而增加，源于用户对有限边缘资源的竞争加剧，引发节点过载与计算时延增加。在 $N \in \{40,60,80\}$ 时，MAGRPO 始终实现接近最优的时延性能。如图 11(b)所示，随着车辆用户数量增加，MAGRPO 的迁移能耗并无显著上升，这是因为 MAGRPO 倾向于仅在迁移收益显著高于综合成本时才触发迁移，从而避免车辆高密度场景下多用户并发迁移引发目标节点过载，进而导致计算时延激增的情况。由于迁移能耗与次数（如图 11(e)所示）呈正相关，该策略自然导致迁移能耗较低。如图 11(c)所示，AM、NM 和 Greedy 的服务过载率均随车辆用户数量 N 增加而呈现较为明显的上升趋势。AM 因始终迁移至最近节点，在高密度场景下易造成局部过载。NM 因服务位置固定，无法适应用户空间分布的动态变化，当测试轨迹在局部区域高度聚集时，初始部署难以覆盖突发高密度需求，导致部分节点瞬时过载。Greedy 缺乏多用户协同机制，易引发多用户同时迁入同一节点。在强化学习方法中，IQL 因忽略智能体间协作，在高负载下迁移决策易产生冲突，迁移成功率相对较低。MAPPO 通过全局 Critic 网络增强智能体间协作，在中等负载下性能表现稳健，但在高负载（$N = 80$）时仍存在 3.1% 的服务过载率。相比之下，MAGRPO 通过组相对优势机制隐式建模智能体间交互，在不需要显式通信的条件下实现迁移策略的协同优化，不仅维持了与 MADDPG 相当的低时延，还降低了迁移能耗和服务过载率，展现出更优的综合性能与可扩展性。

2) 任务处理密度κ：如图12(a)所示，在车辆用户数量$N=40$时，随着任务处理密度上限从500 cycle/bit增至4 000 cycle/bit，边缘服务器计算负载显著上升，导致计算时延增加，所有方法的服务总时延呈上升趋势。MAGRPO在不同κ下始终维持最低时延，优于Greedy与MAPPO，并与MADDPG和IQL相当。如图12(c)所示，NM和AM分别维持46%和70%以上的服务过载率，Greedy的服务过载率也在$\kappa=4\,000$ cycle/bit时上升至33.7%。而所有MARL方法均实现零过载，表明其具备有效负载均衡能力。

4.2.4　消融实验

为验证奖励函数中惩罚项的有效性，本文设计两项消融实验：1) MAGRPO w/o Overload Penalty：设$\lambda_{\text{over}}=0$，保留故障惩罚；2) MAGRPO w/o Failure Penalty：设$\lambda_{\text{fail}}=0$，保留过载惩罚。其余设置与完整MAGRPO一致，实验在$N=40$下进行。如表2所示，移除过载惩罚后，智能体忽略服务器负载状态，导致多个车辆同时迁移至同一边缘节点，服务过载率显著上升。移除故障惩罚后，智能体无法有效规避高风险节点，导致迁移成功率下降。实验结果表明，过载惩罚项有助于引导策略遵守边缘服务器的计算资源约束，而故障惩罚项则有效抑制了向高风险节点的迁移行为，二者共同支撑了高QoS的服务迁移决策。

表2　消融实验结果

方法	服务过载率	迁移成功率
MAGRPO	0	98.9%
MAGRPO w/o Overload Penalty	5.8%	98.6%
MAGRPO w/o Failure Penalty	0	97.3%

4.2.5　系统开销分析

为评估MAGRPO算法的训练开销，本文进一步对比了在不同智能体数量下所提方法与基线方法的显存占用与平均每轮训练时间。其中，MAGRPO的组大小G选取在测试集上平均累积奖励最高的4个取值，即$G\in\{4,8,16,32\}$。

如图13(a)所示，在所有测试场景中，MAGRPO的显存占用均低于MAPPO和MADDPG，这是因为MAGRPO基于组内折扣回报的相对排序构建策略更新信号，避免了训练全局Critic网络所带来的内存开销。相比之下，MAPPO和MADDPG采用集中式Critic网络，需要聚合所有智能体的观测信息，其显存需求随智能体数量增长迅速。以MADDPG为例，在$N=80$时其显存需求高达8 050 MB，约为MAGRPO（$G=32$）的3.5倍。IQL虽显存占用最低，但因其独立学习机制，缺乏多智能体协同能力，在多用户资源竞争场景下性能受限。

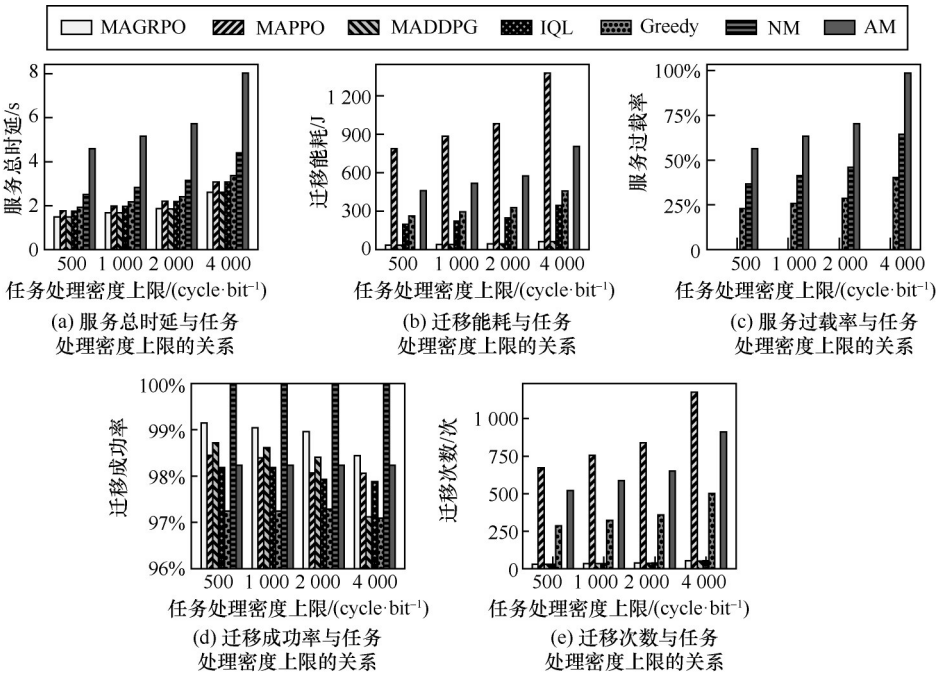


图12　任务处理密度上限对各项指标的影响

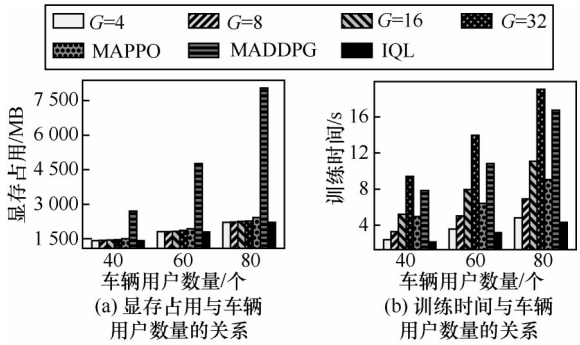


图 13 不同算法的训练开销

如图 13(b)所示，MAGRPO 的训练时间随 G 和 N 的增大而增长。其根本原因在于该算法需在相同初始状态下执行 G 次策略采样以获得更稳定的组相对优势估计，其环境交互次数是单次采样方法的 G 倍，且每条轨迹的计算开销随 N 线性增加。尽管如此，由于避免了 Critic 网络的计算开销，MAGRPO 在较小的组大小（如 $G = 4$ 或 $G = 8$）下，不同 N 值的平均每轮训练时间均低于 MAPPO 和 MADDPG。当 $G = 16$ 时，其耗时略高于 MAPPO，但仍优于 MADDPG。相比之下，MAPPO 和 MADDPG 因需在每轮训练中聚合所有智能体的观测以更新集中式 Critic 网络，导致计算开销较高，训练时间更长。

5 结束语

针对动态车联网环境下的服务迁移问题，本文提出了一种基于 MAGRPO 的智能决策方法，构建了包含车辆移动性、资源竞争与节点可用性的系统模型，并将迁移决策建模为带约束的多用户联合优化问题。受 GRPO 算法在高方差奖励场景中鲁棒性优势的启发，本文将其扩展至多用户服务迁移场景，提出了 MAGRPO 算法。该算法摒弃传统多智能体 Actor-Critic 架构中的显式 Critic 网络，通过在相同初始环境状态下多次执行策略采样，生成多条完整迁移轨迹，并基于组内折扣回报的相对排序构建策略更新信号，从而有效缓解因节点过载或故障等强惩罚约束导致的训练不稳定问题，降低训练开销。仿真实验表明，本文方法在服务总时延、迁移能耗和迁移成功率等指标上的表现均优于现有基线方法，尤其在边缘节点资源受限且节点可用性动态变化的场景下仍能保持较优性能。未来工作将探索服务迁移决策与任务卸载、服务缓存等机制的联合优化。

参考文献：

[1] 王海艳, 张霖, 骆健. 移动边缘计算场景下针对资源竞争的服务迁移优化方法[J]. 通信学报, 2024, 45(8): 37-50.
Wang H Y, Zhang L, Luo J. Service migration optimization method for resource competition in mobile edge computing scenarios[J]. Journal on Communications, 2024, 45(8): 37-50.

[2] 郭辉, 芮兰兰, 高志鹏. 车辆边缘网络中基于多参数 MDP 模型的动态服务迁移策略[J]. 通信学报, 2020, 41(1): 1-14.
Guo H, Rui L L, Gao Z P. Dynamic service migration strategy based on MDP model with multiple parameter in vehicular edge network[J]. Journal on Communications, 2020, 41(1): 1-14.

[3] Edwan T A, Tahat A, Yanikomeroglu H, et al. An analysis of a stochastic ON-OFF queueing mobility model for software-defined vehicle networks[J]. IEEE Transactions on Mobile Computing, 2022, 21(5): 1552-1565.

[4] Aissioui A, Ksentini A, Gueroui A M, et al. On enabling 5G automotive systems using follow me edge-cloud concept[J]. IEEE Transactions on Vehicular Technology, 2018, 67(6): 5302-5316.

[5] Lu W, Meng X Y, Guo G F. Fast service migration method based on virtual machine technology for MEC[J]. IEEE Internet of Things Journal, 2019, 6(3): 4344-4354.

[6] Lowe R, Wu Y, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[J]. arXiv Preprint, arXiv: 1706.02275, 2017.

[7] Taleb T, Ksentini A, Frangoudis P A. Follow-me cloud: when cloud services follow mobile users[J]. IEEE Transactions on Cloud Computing, 2019, 7(2): 369-382.

[8] Wang Y, Cao S, Ren H S, et al. Towards cost-effective service migration in mobile edge: a Q-learning approach[J]. Journal of Parallel and Distributed Computing, 2020, 146: 175-188.

[9] Kang J W, Chen J L, Xu M R, et al. UAV-assisted dynamic avatar task migration for vehicular metaverse services: a multi-agent deep reinforcement learning approach[J]. IEEE/CAA Journal of Automatica Sinica, 2024, 11(2): 430-445.

[10] Labriji I, Meneghello F, Cecchinato D, et al. Mobility aware and dynamic migration of MEC services for the Internet of vehicles[J]. IEEE Transactions on Network and Service Management, 2021, 18(1): 570-584.

[11] Zhou X B, Ge S X, Qiu T, et al. Energy-efficient service migration for multi-user heterogeneous dense cellular networks[J]. IEEE Transactions on Mobile Computing, 2023, 22(2): 890-905.

[12] Tuli S, Casale G, Jennings N R. PreGAN: preemptive migration prediction network for proactive fault-tolerant edge computing[C]//Proceedings of the IEEE INFOCOM 2022 - IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2022: 670-679.

[13] Ma Y, Dai M X, Shao S Y, et al. A performance and reliability-guaranteed predictive approach to service migration path selection in mobile computing[J]. IEEE Internet of Things Journal, 2023, 10(20): 17977-17987.

[14] Ouyang T, Zhi Z, Xu C. Follow me at the edge: mobility-aware dynamic service placement for mobile edge computing[J]. IEEE Journal on Selected Areas in Communications, 2018, 36(10): 2333-2345.

[15] Velrajan S, Ceronmani Sharmila V. QoS-aware service migration in multi-access edge compute using closed-loop adaptive particle swarm optimization algorithm[J]. Journal of Network and Systems Management, 2022, 31(1): 17.

[16] Maia A M, Ghamri-Doudane Y, Vieira D, et al. An improved multi-objective genetic algorithm with heuristic initialization for service placement and load distribution in edge computing[J]. Computer Networks, 2021, 194: 108146.

[17] Peng Y, Liu L, Zhou Y Q, et al. Deep reinforcement learning-based dynamic service migration in vehicular networks[C]//Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM). Piscataway: IEEE Press, 2020: 1-6.

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

[19] Ren H S, Wang Y, Xu C Z, et al. SMig-RL: an evolutionary migration framework for cloud services based on deep reinforcement learning[J]. ACM Transactions on Internet Technology, 2020, 20(4): 1-18.

[20] Cui Y Y, Zhang D G, Zhang J, et al. Distributed task migration optimization in MEC by deep reinforcement learning strategy[C]//Proceedings of the 2021 IEEE 46th Conference on Local Computer Networks (LCN). Piscataway: IEEE Press, 2021: 411-414.

[21] Shao Z H, Wang P Y, Zhu Q H, et al. DeepSeekMath: pushing the limits of mathematical reasoning in open language models[J]. arXiv Preprint, arXiv: 2402.03300, 2024.

[22] Yuan Y, Yang B, Su W, et al. Service migration optimization for system overhead minimization in VECNs via deep reinforcement learning[J]. IEEE Internet of Things Journal, 2025, 12(4): 3905-3920.

[23] Bracciale L, Bonola M, Loreti P, et al. Crawdad dataset roma/taxi[J]. CRAWDAD Wireless Network Data Archive, 2014. DOI: 10.15783/C7QC7M

[24] Wang J, Hu J, Min G Y, et al. Online service migration in mobile edge with incomplete system information: a deep recurrent actor-critic learning approach[J]. IEEE Transactions on Mobile Computing, 2023, 22(11): 6663-6675.

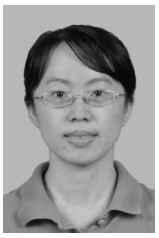
[25] Chen S Y, Rui L L, Gao Z P, et al. Service migration with edge collaboration: multi-agent deep reinforcement learning approach combined with user preference adaptation[J]. Future Generation Computer Systems, 2025, 165: 107612.

[26] Yao Z X, Xia S C, Li Y, et al. Cooperative task offloading and service caching for digital twin edge networks: a graph attention multi-agent reinforcement learning approach[J]. IEEE Journal on Selected Areas in Communications, 2023, 41(11): 3401-3413.

[27] Liu H R, Jiang N, Guo F X, et al. Mobility-aware dynamic service migration in communication and computing integrated VNETs[C]//Proceedings of the 2023 IEEE Globecom Workshops (GC Wkshps). Piscataway: IEEE Press, 2024: 2061-2066.

[28] Du J B, Kong Z W, Sun A J, et al. MADDPG-based joint service placement and task offloading in MEC empowered air-ground integrated networks[J]. IEEE Internet of Things Journal, 2024, 11(6): 10600-10615.

[作者简介]



芮兰兰（1979– ），女，安徽潜山人，博士，北京邮电大学副教授、硕士生导师，主要研究方向为网络智能管控、边缘计算、区块链、数据可信共享。

邓淑予（2000– ），女，云南楚雄人，北京邮电大学硕士生，主要研究方向为边缘计算。

陈子轩（1999– ），男，广东湛江人，北京邮电大学博士生，主要研究方向为网络智能管控。

高志鹏（1980– ），男，山东滨州人，博士，北京邮电大学教授、硕士生导师，主要研究方向为区块链关键算法与应用、边缘计算与边缘智能、数据应用与管理、隐私计算。

邱雪松（1973– ），男，江西上饶人，博士，北京邮电大学教授、硕士生导师，主要研究方向为网络智能管理、工业互联网与区块链、数据资产可信共享与交易。

郭少勇（1985– ），男，河北隆尧人，博士，北京邮电大学教授、硕士生导师，主要研究方向为工业互联网网络管控、数据可信共享与隐私计算、边缘智能应用。