



基于 DDQN 的边缘算力融合网络资源管理

董玉池, 闫亚旗, 冉沛, 王东, 张阔, 张文龙
(中国铁塔股份有限公司, 北京 100080)

摘要: 边缘算力融合网络将算力下沉至近用户侧, 通过分布式边缘算力节点相互协同以在本地完成计算任务, 显著降低云端负担和传输时延。然而, 随着用户接入密度提高和场景复杂化, 如何动态优化网络资源以协同应对多样化服务需求和大规模数据处理任务成为重大挑战。因此, 提出了一种基于双深度 Q 网络 (double deep Q network, DDQN) 边缘算力融合网络资源管理方法, 结合虚拟网络嵌入 (virtual network embedding, VNE) 技术, 建立了以长期资源收益成本比最大化为目标的多约束优化模型。通过 DDQN 架构的在线学习能力, 利用环境交互反馈实现动态优化决策。仿真实验表明, 该方法在虚拟网络请求 (virtual network request, VNR) 接受成功率、长期嵌入收益和长期嵌入收益成本比 3 个指标上, 较现有方法分别提升了 13.3%、25.7% 和 8.5%。

关键词: 算力融合网络; 算力下沉; 双深度 Q 网络; 资源管理

中图分类号: TP393

文献标志码: A

doi: 10.11959/j.issn.1000-0801.2025137

A DDQN-based resource management method for edge computing fusion network

DONG Yuchi, YAN Yaqi, RAN Pei, WANG Dong, ZHANG Kuo, ZHANG Wenlong
China Tower Co., Ltd., Beijing 100080, China

Abstract: The edge computing fusion network sinks the computing resources to the user side and completes the computing tasks locally through the coordination of distributed edge computing nodes, which significantly reduces the cloud burden and transmission delay. However, with the increase of user access density and the complexity of scenarios, how to dynamically optimize network resources to cope with diversified service demands and large-scale data processing tasks has become a major challenge. Therefore, a resource management method for edge computing networks based on double deep Q network (DDQN) was proposed. Integrating the virtual network embedding (VNE) technology, the proposed method formulated a multi-constraint optimization model to maximize the long-term embedding revenue-to-cost ratio. By leveraging the online learning capabilities of the DDQN framework, it enabled dynamic decision-making through interaction and feedback with the environment. Simulation results demonstrate that the proposed method achieves average improvements of 13.3%, 25.7% and 8.5% of virtual network request (VNR) ac-



ceptance rate, long-term embedding revenue, and long-term revenue-to-cost ratio, respectively, compared with the existing methods.

Key words: edge computing fusion network, computing shifting, DDQN, resource management

0 引言

随着 B5G 和 6G 等新一代网络架构和通信技术的迅速演进,边缘计算已成为提升网络服务质量和降低通信时延的核心技术之一^[1-3]。边缘算力融合网络如图 1 所示,边缘计算通过在网络边缘侧部署计算资源,将数据处理能力从集中式云端转移至分布式边缘节点,显著地缩短了数据传输路径,并支持本地计算任务的高效协作。这种模式不仅能够有效地减轻核心网络的负担,还能够满足低时延和高带宽的通信需求^[4]。特别是在诸如自动驾驶、智能制造、远程医疗等场景中,边缘计算已展现出巨大的应用潜力和技术优势^[5]。然而,随着用户数量的增长和应用场景的复杂化,单一边缘节点的计算资源和网络资源难以满足多样化服务需求和复杂数据处理任务^[6]。因此,实现边缘算力的高效协同和网络资源的动态优化管理,已成为当前网络技术研究的一个重要方向。

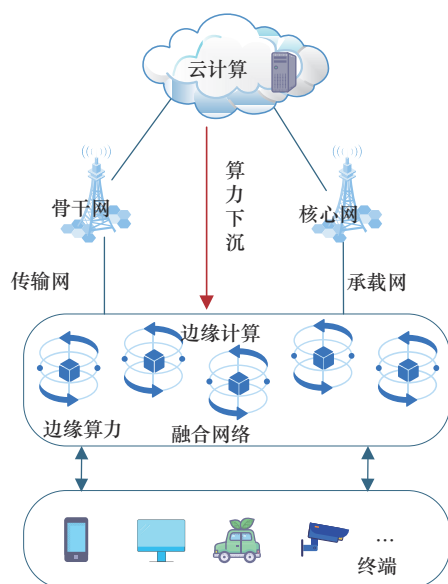


图1 边缘算力融合网络

在此背景下,虚拟网络嵌入(virtual network embedding, VNE)技术因其能够灵活调度和管理网络资源,被视为解决资源分配与优化问题的核心手段之一^[7]。通过将虚拟网络请求(virtual network request, VNR)映射到物理网络基础设施,VNE技术能够根据不同应用场景的需求,为多样化服务提供定制化支持^[8]。与传统静态资源分配方法相比,VNE技术在资源动态优化和服务定制化方面具有显著优势,特别是在支持复杂拓扑结构和多约束条件方面展现了强大的适应能力^[9]。在VNE方法研究初期,多依赖于启发式算法和静态规划。例如,经典算法NodeRank^[10],其以物理节点的资源权重作为排名依据,并优先嵌入资源更丰富的物理节点。Melo等^[11]提出了Node-link formulation VNE(VNE-NLF)算法,以最小化资源消耗和负载均衡作为优化目标,基于整数线性规划(integer linear programming, ILP)解决在线VNE问题。然而,这些传统方法大多基于静态已知的网络环境假设,在应对网络动态变化和复杂多约束条件方面表现出明显的不足。特别是其依赖于经验规则或预定义策略,易陷入局部最优解,且计算复杂度随着网络规模的增加而显著提高,限制了其实际应用场景的适用性。

近年来,强化学习(reinforcement learning, RL)技术凭借其在高维状态空间和动态决策问题中的优势,逐渐成为VNE领域的重要研究方向^[12-13]。RL方法通过环境交互学习,能够动态建模网络资源的演化规律,并为资源调度提供数据驱动优化策略。例如,Yao等^[14]一种提出了基于RL的连续决策VNE(continuous-decision VNE, CDVNE)算法,其将同一请求的节点嵌入视作经典Seq2Seq模型制定的时间序列问题,通过递

归神经网络进行建模,并通过策略梯度算法更新网络参数。Yan等^[15]提出了一种结合RL和图卷积神经网络的自动VNE算法,基于图卷积神经网络自动提取不规则图拓扑结构的空问特征,使用并行策略梯度训练方法确保采样训练的效率和鲁棒性。此外,深度强化学习(deep reinforcement learning, DRL)技术进一步结合了深度学习的表征能力和RL的决策能力,能够通过深度神经网络近似复杂的状态—动作值函数,在高维和连续状态空间中实现更高效的优化决策^[16-17]。例如,Chen等^[18]提出了一种结合谱图理论与DRL的VNE算法(GRVNE),通过智能体与环境的多轮交互优化决策过程,显著地提高了复杂网络场景中的资源调度效率和嵌入性能。Zhang等^[19]提出了一种面向云雾边多层算力网络的多目标感知的动态资源调度算法,考虑了算力、存储、带宽、时延等差异化服务质量(quality of service, QoS)需求,建立了多目标感知的嵌入约束。并且,建立了基于DRL的调度网络,同时在与环境的交互中优化策略。此外,Lim等^[20]提出一种基于分层协作多智能体强化学习(MARL)算法,将VNE问题划分为多个子问题,利用分层多强化学习使得不同的智能体专注于提高各自性能。并且,还将图卷积网络应用于提取特征过程中,以识别网络中节点与链路的空间关系。这些方法表明,RL和DRL的引入能够在动态环境中有效地提高算法的适应性和性能表现,为资源管理提供了新的思路和技术途径。

基于上述启发,本文聚焦于边缘算力融合网络中的资源管理问题,提出了一种基于DRL的边缘算力融合网络资源管理方法。本文的贡献总结如下。

(1) 提出了一种基于双深度Q网络(double deep Q network, DDQN)的VNE(DDQN-VNE)算法,用于边缘算力融合网络资源的动态管理,以智能化的决策过程优化边缘算力和网络资源的

分配。通过引入双Q网络的更新机制,解决了传统强化学习算法易受过高估计偏差影响的问题。

(2) 构建了长期资源收益成本比的多约束优化模型,综合考虑了算力节点的异构特性、资源分配的动态性以及用户需求的多样性,并通过DDQN算法的在线学习能力,赋予系统对网络环境变化的高适应性。这种方法不仅显著提高了系统的资源利用效率,还通过优化通信路径有效降低了网络时延,同时减少了资源调度过程中的开销。

(3) 仿真实验表明,该算法在VNR接受成功率、长期嵌入收益和长期嵌入收益成本比3个指标上,较现有方法分别提升了13.3%、25.7%和8.5%。

1 问题与优化目标建模

1.1 问题定义

为了满足对实时性和低延迟的应用需求,算力资源从集中式云端向网络边缘下沉形成边缘算力融合网络,在接近用户终端的地方部署计算资源,使得数据处理和分析可以在本地进行,从而减少了数据传输的时延。通过将计算任务分布到多个边缘节点,网络能够更有效地协同利用现有的算力资源,减轻了云端的负担,同时也提高了整体网络的响应速度和处理能力^[21]。因此,基于网络虚拟化技术,将边缘算力融合网络建模为一个带权无向图 $\mathcal{G}^e = \{EN, EL, EC, EB, ED\}$,其中,图节点EN表示部署在网络边缘、靠近用户或数据源的计算和存储单元,即边缘算力节点;图链路EL表示边缘间链路,负责连接不同的边缘节点,支持它们之间的数据交换和协作,以实现资源共享、负载均衡和任务调度等功能,从而增强边缘计算网络的整体性能、提高效率;权重EC表示边缘节点所拥有的算力资源,权重EB和ED分别表示边缘间链路所拥有的带宽资源以及传输时延。同样,用户计算任务请求在VNE技术中被



称作 VNR，它被建模为一个虚拟网络，同样是一个带权无向图 $\mathcal{G}^v = \{VN, VL, VC, VB, VD\}$ ，其中，VN 和 VL 分别表示该虚拟网络的虚拟节点和虚拟链路，VC、VB 和 VD 分别表示该虚拟网络所请求的计算资源、带宽资源以及时延要求。为方便理解，问题建模如图 2 所示，展示了边缘算力融合网络以及 VNR 的样例。此外，本文使用小写字母来表示具体的元素，例如， en_i 表示第 i 个边缘算力节点， el_{ij} 表示 i 第个边缘算力节点和第 j 个边缘算力节点之间的边缘间链路。

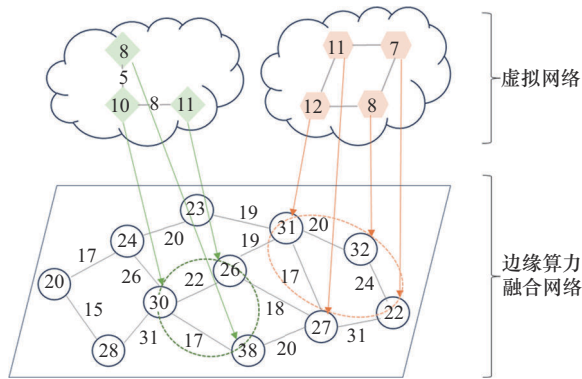


图2 问题建模

因此，通过 VNE 技术对边缘算力网络进行资源分配的过程，可以认为是将代表算力任务请求的虚拟网络映射到边缘算力网络的过程，在这个过程中需要满足一系列的相关约束，表示为：

for $i = 1, \dots, |VNR|$:

for $t_s \rightarrow t_e$: Equation Chapter 1 Section (Next) (1 a)

$$\mathcal{G}_v \rightarrow \mathcal{G}_e,$$

$$\text{s.t. } \zeta_{en_j}^{vn_i} = \begin{cases} 1, & vn_i \text{ 成功映射至 } en_j \\ 0, & \text{其他} \end{cases} \quad (1b)$$

$$\zeta_{el_j}^{vl_i} = \begin{cases} 1, & vl_i \text{ 成功映射至 } el_j \\ 0, & \text{其他} \end{cases} \quad (1c)$$

$$A_EC(en_j) = EC(en_j) - \sum_{k=1}^{|VNR|} \sum_{i=1}^{|EC|} \zeta_{en_j}^{vn_i} VC(vn_i) \quad (1d)$$

$$A_EC(en_j) \geq 0, \forall en_j \quad (1e)$$

$$A_EB(el_j) = EB(el_j) - \sum_{k=1}^{|VNR|} \sum_{i=1}^{|EL|} \zeta_{el_j}^{vl_i} VB(vl_i) \quad (1f)$$

$$A_EB(el_j) \geq 0, \forall el_j \quad (1g)$$

$$\text{when } \zeta_{en_j}^{vn_i} = 1, A_EC(en_j) \geq VC(vn_i) \quad (1h)$$

$$\text{when } \zeta_{el_j}^{vl_i} = 1, A_EB(el_j) \geq VB(vl_i) \quad (1i)$$

$$\sum_{j=1}^{|EN|} \zeta_{en_j}^{vn_i} = 1, \sum_{j=1}^{|EL|} \zeta_{el_j}^{vl_i} \geq 1, \forall vn_i \& vl_i$$

Equation Chapter (Next) Section 1 (2a)

其中， t_s 和 t_e 分别表示相应 VNR 的生命周期的开始时间和结束时间，在 VNR 生命周期结束时其所占用的边缘算力网络的资源将会被释放； $\zeta_{en_j}^{vn_i}$ 表示虚拟节点 vn_i 是否成功映射到边缘算力节点 en_j ，如果成功则为 1，否则为 0；类似地， $\zeta_{el_j}^{vl_i}$ 表示虚拟链路 vl_i 是否成功映射到边缘间链路 el_j ，如果成功则为 1，否则为 0；约束 (1d) 表示边缘算力节点的可用算力资源，其值等于该节点初始拥有的算力资源与已经分配给虚拟网络的算力资源的差值；约束 (1e) 表示边缘算力节点的可用算力资源不能为负值；约束 (1f) 表示边缘间链路的可用带宽资源，其值等于该链路初始拥有的带宽资源与已经分配给虚拟网络的带宽资源的差值；约束 (1g) 表示边缘间链路的可用带宽资源不能为负值；约束 (1h) 表示如果某个虚拟节点 vn_i 想要映射到边缘算力节点 en_j ，则其所请求的算力资源应该小于该边缘算力节点的可用算力资源；约束 (1i) 表示如果某个虚拟链路 vl_i 想要映射到边缘间链路 el_j ，则其所请求的带宽资源应该小于该边缘间链路的可用带宽资源，并且该边缘间链路的传输时延应该小于 vl_i 所要求的时延限制；约束 (1j) 表示任意虚拟节点只能映射到 1 个边缘算力节点上，任意虚拟链路可以映射到多条边缘间链路上，这主要由于 2 个边缘算力节点可能需要横跨多条边缘间链路互通。

1.2 优化目标

在 VNE 技术中, 通常从资源成本和嵌入收益角度衡量策略效果。具体来说, 旨在最小化物理资源的消耗, 如计算资源和链路带宽, 通过合理选择物理节点和路径来降低嵌入成本。同时, 在满足资源约束的条件下, 算法会优先选择能够带来更高收益的映射策略, 从而提升整体网络效能, 最大化收益。嵌入成本表示其映射虚拟网络到边缘算力网络时, 所损耗的算力网络的物理资源总量, 表示为:

$$\text{Cost}(\mathcal{G}_i^v) = \sum_{j=1}^{|\text{VNR}_i|} \text{VC}(\text{vn}_j) + \sum_{k=1}^{|\text{VL}_i|} \text{VB}(\text{vl}_k) \text{hops}(\text{vl}_k) \quad (2)$$

其中, $\text{hops}()$ 表示虚拟链路所占用的边缘间链路路径的跳数。此外, 嵌入收益表示虚拟网络成功映射到边缘算力网络所获得的收益, 表示为该虚拟网络所请求的资源, 如式 (3)。

$$\text{Revenue}(\mathcal{G}_i^v) = \sum_{j=1}^{|\text{VNR}_i|} \text{VC}(\text{vn}_j) + \sum_{k=1}^{|\text{VL}_i|} \text{VB}(\text{vl}_k) \quad (3)$$

进而, 在此基础上, 可以推导出 VNE 常用的 VNR 接受成功率 (Acc)、长期嵌入收益 (R) 和长期嵌入收益成本比 (R/C) 3 个常用评价指标的表达方式, 分别如下。

$$\text{Acc} = \lim_{T \rightarrow \infty} \frac{\sum_{t=0}^T \sum_{i=1}^{|\text{VNR}|} \left(\prod_{j=1}^{\text{VNR}_i} \zeta_{\text{en}_j}^{\text{vn}_i} \prod_{k=1}^{\text{VL}_i} \zeta_{\text{el}_j}^{\text{vl}_i} \right)}{\sum_{t=0}^T |\text{VNR}|} \quad (4)$$

$$R = \lim_{T \rightarrow \infty} \frac{\sum_{t=0}^T \sum_{i=1}^{|\text{VNR}|} \text{Revenue}(\mathcal{G}_i^v)}{T} \quad (5)$$

$$R/C = \lim_{T \rightarrow \infty} \frac{\sum_{t=0}^T \sum_{i=1}^{|\text{VNR}|} \text{Revenue}(\mathcal{G}_i^v)}{\sum_{t=0}^T \sum_{i=1}^{|\text{VNR}|} \text{Cost}(\mathcal{G}_i^v)} \quad (6)$$

2 基于 DDQN 的边缘算力融合网络资源管理算法

深度强化学习因其在高维状态空间和连续决策问题中的优势, 是解决该问题的一个有效方

法。其中, DDQN^[22]通过有效减少 Q 值过度估计问题, 在动态环境中的资源管理和优化问题上展现出强大的适应性和鲁棒性。因此, 本文将 DDQN 应用于边缘算力融合网络的资源管理, 以智能化的决策过程优化边缘算力和网络资源的资源配置。

2.1 算法模型

本文所部署的边缘算力融合网络资源管理算法模型如图 3 所示。DDQN 架构是一种将深度神经网络与 double Q-learning 相结合的先进方法, 其中, 智能体内部集成了 2 个深度神经网络结构, 即 Q 网络和目标 Q 网络。两者共享相同的网络结构, 智能体中 Q 网络和目标 Q 网络的结构见表 1。

(1) Q 网络。负责估计当前状态下不同动作的 Q 值, 进而选择最优动作。为了在探索和利用之间取得平衡, 使用 ε -贪婪算法来选择动作。

$$a_t = \begin{cases} \text{argmax}_{a_t} Q(s_t, a_t; \theta), & 1 - \varepsilon \text{ 的概率} \\ \text{随机选择动作}, & \varepsilon \text{ 的概率} \end{cases} \quad (7)$$

其中, $Q(s_t, a_t; \theta)$ 表示动作 a_t 在状态 s_t 下的估计 Q 值, θ 表示 Q 网络的参数。

采取动作 a_t 后, 将获得奖励 r_t 和下一时刻的状态 s_{t+1} , 这些元素将组成四元组的经验项 (s_t, a_t, r_t, s_{t+1}) , 存放于经验回放池中, 表示为:

$$R = \{(s_t, a_t, r_t, s_{t+1})_1, (s_t, a_t, r_t, s_{t+1})_2, \dots, (s_t, a_t, r_t, s_{t+1})_n\} \quad (8)$$

(2) 目标 Q 网络。通过经验回放机制从经验回放池中抽样批量经验项, 并估计下一状态的目标 Q 值, 该目标 Q 值将用于训练和更新 Q 网络。对于经验项 (s_t, a_t, r_t, s_{t+1}) , 目标 Q 值计算为:

$$Q'(s_t, a_t; \theta') = r_t + \gamma Q'(s_{t+1}, \text{argmax}_{a_t} Q(s_{t+1}, a_t; \theta); \theta') \quad (9)$$

其中, θ' 表示目标 Q 网络的参数, γ 表示折扣因子。

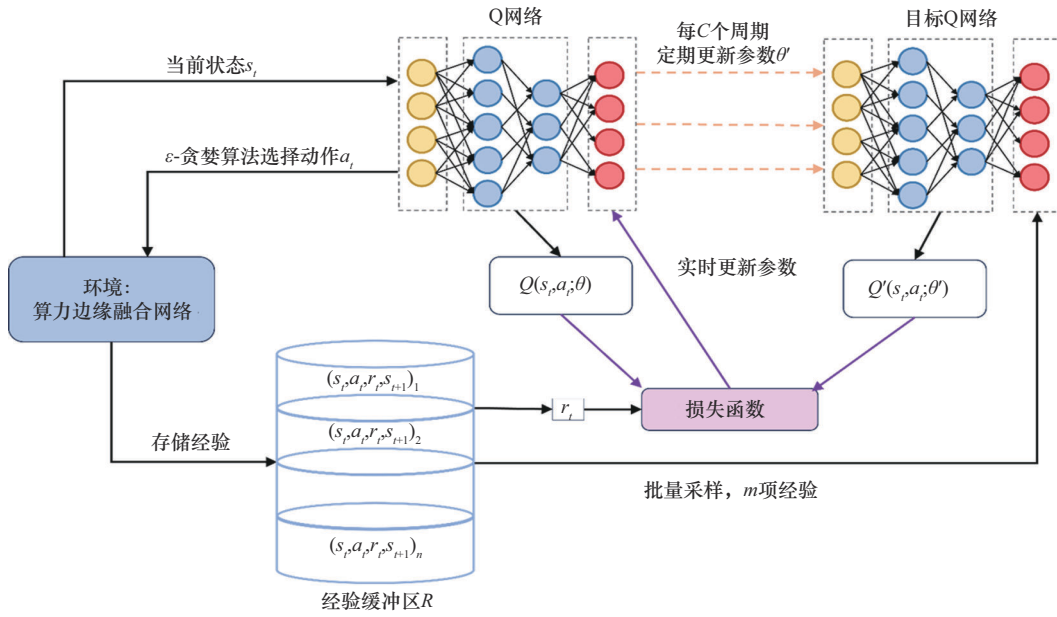


图3 本文所部署的边缘算力融合网络资源管理算法模型

表1 智能体中Q网络和目标Q网络的结构

网络层	输入维度	输出维度	激活函数
输入层	$1 \times (\text{EN} \times 3)$	256	ReLU
隐藏层	256	256	ReLU
隐藏层	256	256	ReLU
输出层	256	$ \text{EN} $	-

2.2 DDQN 基本元素配置

(1) 状态。表示智能体某一时刻所处的环境信息，用来描述当前的环境情况，作为智能体的输入。基于状态选择影响动作决策的原则，本文提取如下属性作为状态信息。

- 边缘算力节点可用算力资源 ($A_EC(en_j)$)。如约束 (1d) 所示表示该边缘算力节点的剩余的可分配的算力资源，其值暗示了该边缘算力节点承载虚拟节点的可能性。
- 边缘算力节点周围的带宽资源。其值等于与该边缘算力节点相连的所有边缘间链路

的可用带宽资源 (约束 (1f)) 之和。同样地，其值暗示了该边缘算力节点承载虚拟节点的可能性，且其值越大越容易被用于分配资源。其计算方式为：

$$\text{SUM_EB}(en_j) = \sum_{\forall el_{jk} \in EL} A_EB(el_{jk}) \quad (10)$$

- 边缘算力节点周围的传输时延。其值等于与该边缘算力节点相连的所有边缘间链路的传输时延之和。同样地，其值暗示了该边缘算力节点承载虚拟节点的可能性，且其值越小越容易被用于分配资源。其计算方式为：

$$\text{SUM_ED}(en_j) = \sum_{\forall el_{jk} \in EL} ED(el_{jk}) \quad (11)$$

为了加速和稳定模型的收敛过程，还需要对以上属性进行归一化处理，表示为：

$$\bar{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (12)$$

结合上述归一化属性向量可以得到状态矩阵，其将用作智能体的输入。

$$s_t = \begin{bmatrix} \overline{A_EC}(en_1) & \overline{\text{SUM_EB}}(en_1) & \overline{\text{SUM_ED}}(en_1) \\ \vdots & \ddots & \vdots \\ \overline{A_EC}(en_{|\text{EN}|}) & \overline{\text{SUM_EB}}(en_{|\text{EN}|}) & \overline{\text{SUM_ED}}(en_{|\text{EN}|}) \end{bmatrix} \quad (13)$$

(2) 动作。表示智能体根据当前状态所选择的决策，旨在与环境进行交互并改变环境状态。需要说明的是，VNE 是节点映射和链路映射的两阶段算法，而本文中智能体仅用于生成节点映射的决策，链路映射基于宽度优先搜索算法执行。因此，动作表示为：

$$a_t = \{a_{t,j} | a_{t,j} = 0 \text{ 或者 } 1, j \in [1, |EN|] \sum_{j=1}^{|EN|} a_{t,j} = |VN|\} \quad (14)$$

其中， $a_{t,j}$ 说明边缘算力节点 en_j 被选择用于承载虚拟节点。

(3) 奖励。表示智能体执行某一动作后环境反馈的信号数值，用于衡量该动作的好坏，引导智能体学习最优策略。本文采用式 (6) 长期嵌入收益成本比 (R/C) 作为奖励值。

$$r_t = R/C \quad (15)$$

2.3 优化过程

Q 网络参数是实时更新的，目标 Q 网络的参数是每隔 C 步周期性更新的，并且通过从 Q 网络复制参数 ($\theta' \leftarrow \theta$) 来保持同步。此外，Q 网络的损失函数采用均方误差 (mean squared error, MSE)，并且以目标 Q 网络值作为估计值，以 Q 网络值作为真实值，计算方式为：

$$L = \frac{\sum_{i=1}^m (Q'(s_t, a_t; \theta') - Q(s_t, a_t; \theta))^2}{m} \quad (16)$$

其中， m 表示采样的批量大小。

此外，Q 网络参数的更新是基于梯度下降策略，表示为：

$$\theta \leftarrow \theta - \mu \nabla_{\theta} L \quad (17)$$

3 仿真实验验证

3.1 环境仿真

与文献[14-15]、文献[19]和文献[23]采用类似的设置，本文对边缘算力融合网络和 VNR 进行了仿真，模型相关参数及仿真参数配置见表 2。此外，生成了一个连续的时间序列，并且所有 VNR 是按

照泊松过程到达的，完成映射的 VNR 在其生命周期结束后会释放其所占用的边缘算力网络资源。

表 2 模型相关参数及仿真参数配置

	参数	数值	
模型参数	学习率	10^{-3}	
	探索率	0.9→0	
	折扣因子	0.99	
	经验回放池	10 000	
	批量大小	64	
	目标网络更新间隔	200	
	优化器	Adam	
	算力任务请求参数	VNR 数量	2 000
		虚拟节点数量	2~10
		虚拟节点之间的虚拟链路的生成概率	50%
虚拟结点所请求的算力资源		1~20 TFLOPS	
虚拟链路所请求的带宽资源		1~20 Mbit/s	
虚拟链路所要求的传输时延		1~50 ms	
边缘算力融合网络参数	边缘算力节点	100	
	边缘间链路	600	
	边缘算力节点的算力资源	50~100 TFLOPS	
	边缘间链路的带宽资源	50~100 Mbit/s	
	边缘间链路的传输时延	1~50 ms	

3.2 实验结果与对比分析

首先，需要验证所提出的模型在训练阶段的奖励收敛曲线。通过观察该曲线，可以评估模型的学习进度，判断其是否逐渐趋向最优策略，同时也可以检测训练过程的稳定性。因此，本文分别记录了学习率为 0.010、0.005 和 0.001 的收敛曲线，VNR 接受成功率训练阶段收敛曲线、长期嵌入收益训练阶段收敛曲线和长期嵌入收益成本比训练收敛曲线分别如图 4、图 5 和图 6 所示。可以发现，当学习率为 0.010 时，虽然模型可以更快收敛，但是由于步幅更大，性能表现有待提升，且奖励曲线波动较大；当学习率为 0.001 时，虽然模型也可以收敛，但是由于步幅较小，因此其收敛速度较慢；当学习率为 0.005 时，模型收敛速度和性能表现均更优，模型可以在 190~200 次迭代后快速收敛。

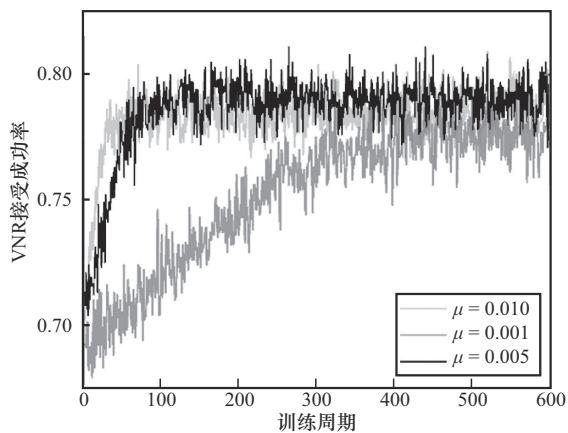


图4 VNR接受成功率训练阶段收敛曲线

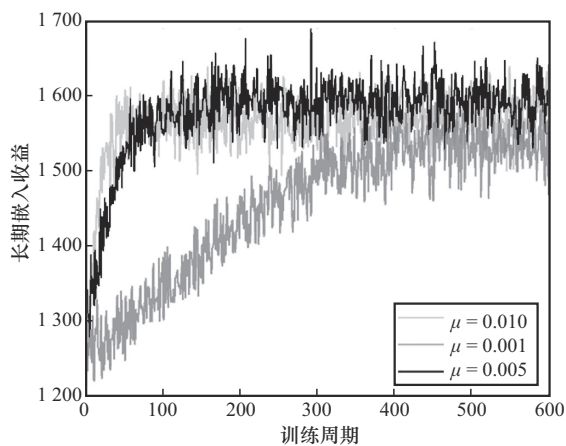


图5 长期嵌入收益训练阶段收敛曲线

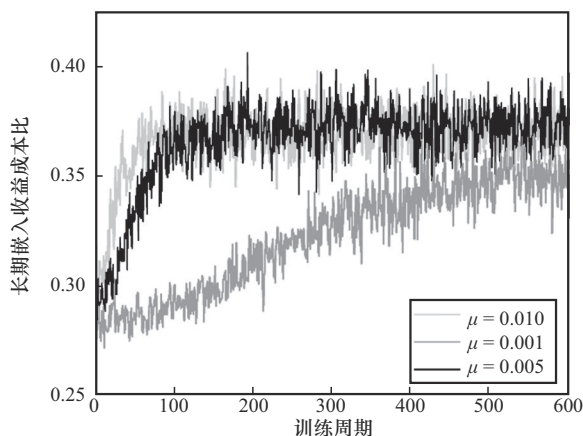


图6 长期嵌入收益成本比训练阶段收敛曲线

此外，还在VNR接受成功率、长期嵌入收益和长期嵌入收益成本比3个指标上与已有的相关算法进行了对比，包括NodeRank^[9]、CDVNE^[14]、GRVNE^[17]。以2 000 s为时间间隔，记录不同时

间片的各项指标值，VNR接受成功率对比、长期嵌入收益对比和长期嵌入收益成本比对比分别如图7、图8和图9所示。可以发现，随着时间的推移，边缘算力融合网络资源不断被分配，这导致后续可成功嵌入的VNR不断减少，因此，VNR接受成功率与长期嵌入收益不断减少。而在后期阶段，成功映射的VNR在生命周期结束时所占用的边缘算力网络资源将被释放，与新到达VNR的映射达成平衡，因此VNR接受成功率与长期嵌入收益达到一个稳定状态。而长期嵌入收益成本比指标同时与嵌入收益和嵌入成本有关，因此呈现出波动状态。此外，NodeRank算法依赖于物理节点的资源权重排名，进行嵌入前期表现良好，但是，随着VNR不断被嵌入，物理资源碎片化现象显著，导致后续VNR嵌入成功率下降，因此后期表现较差。CDVNE算法将同一个VNR的节点嵌入视作一个连续过程，进一步降低了路径映射可能导致的资源损耗，显著地提高了各项指标值。GRVNE算法还通过图神经网络结构提取物理网络的空间拓扑结构，纳入了对全局映射的考虑，进一步提高了各指标值。而本文算法通过DDQN架构分离动作选择和动作价值评估2个Q网络，降低了过高估计可能带来的偏差，并且通过经验回访池稳定学习过程，提高策略的泛化能力，因此表现得更加出色。

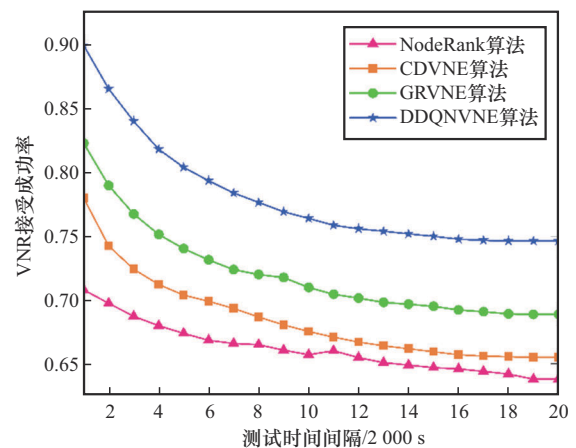


图7 VNR接受成功率对比

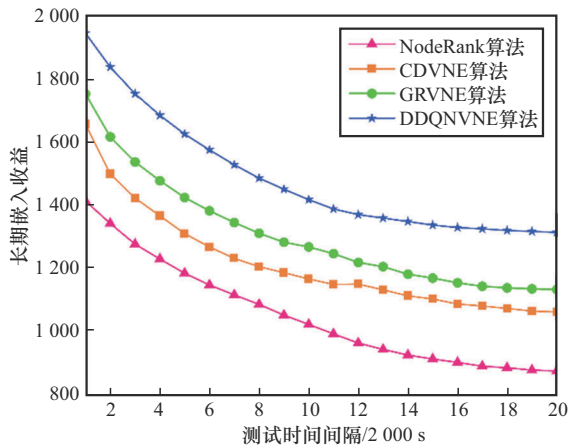


图8 长期嵌入收益对比

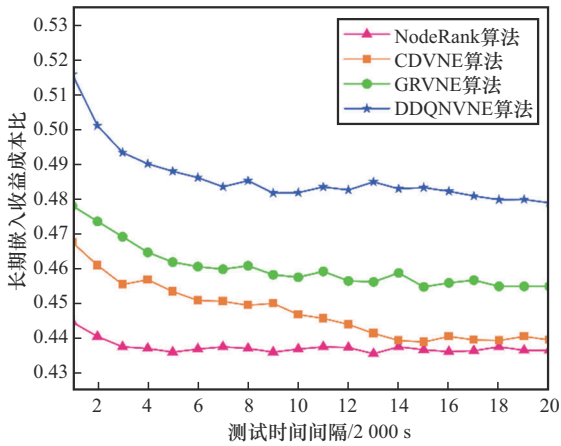


图9 长期嵌入收益成本比对比

4 结束语

本文提出了一种基于DDQN的边缘算力融合网络资源管理方法，旨在应对边缘算力网络中日益增长的用户需求和复杂多样的应用场景。通过构建长期资源收益成本比的多约束优化目标，并结合虚拟网络嵌入技术，实现了对算力资源的高效分配和动态调整。仿真实验结果表明，该方法在提升VNR接受成功率、长期嵌入收益和长期嵌入收益成本比方面，均表现出显著优势。未来工作将进一步优化模型的鲁棒性，探索在实际网络部署中的适应性。

参考文献:

[1] 刘云毅, 张建敏, 冯晓丽, 等. 5G MEC系统安全能力部署方

案[J]. 电信科学, 2022, 38(11): 143-152.

LIU Y Y, ZHANG J M, FENG X L, et al. 5G MEC system security capability deployment scheme[J]. Telecommunications Science, 2022, 38(11): 143-152.

[2] WANG S, SHENG H, ZHANG Y, et al. Blockchain-empowered distributed multicamera multitarget tracking in edge computing[J]. IEEE Transactions on Industrial Informatics, 2024, 20(1): 369-379.

[3] 雷波, 刘增义, 王旭亮, 等. 基于云、网、边融合的边缘计算新方案: 算力网络[J]. 电信科学, 2019, 35(9): 44-51.

LEI B, LIU Z Y, WANG X L, et al. Computing network: a new multi-access edge computing[J]. Telecommunications Science, 2019, 35(9): 44-51.

[4] 谢人超, 廉晓飞, 贾庆民, 等. 移动边缘计算卸载技术综述[J]. 通信学报, 2018, 39(11): 138-155.

XIE R C, LIAN X F, JIA Q M, et al. Survey on computation offloading in mobile edge computing[J]. Journal on Communications, 2018, 39(11): 138-155.

[5] XIAO A L, WU S, OU Y K, et al. QoE-fairness-aware bandwidth allocation design for MEC-assisted ABR video transmission[J]. IEEE Transactions on Network and Service Management, 2024, PP(99): 1.

[6] 雷波, 赵倩颖, 赵慧玲. 边缘计算与算力网络综述[J]. 中兴通讯技术, 2021, 27(3): 3-6.

LEI B, ZHAO Q Y, ZHAO H L. Overview of edge computing and computing power network[J]. ZTE Technology Journal, 2021, 27(3): 3-6.

[7] 冯川, 张旭, 马天纯, 等. 节点链路同时映射的多类型服务资源编排算法[J]. 通信学报, 2024, 45(4): 54-64.

FENG C, ZHANG X, MA T C, et al. Node-link simultaneous mapping based multi-type service resource orchestration algorithm[J]. Journal on Communications, 2024, 45(4): 54-64.

[8] ZHANG P Y, CHEN N, KUMAR N, et al. Energy allocation for vehicle-to-grid settings: a low-cost proposal combining DRL and VNE[J]. IEEE Transactions on Sustainable Computing, 2024, 9(1): 75-87.

[9] WU S, CHEN N, WEN G H, et al. Virtual network embedding for task offloading in IIoT: a DRL-assisted federated learning scheme[J]. IEEE Transactions on Industrial Informatics, 2024, 20(4): 6814-6824.

[10] CHENG X, SU S, ZHANG Z B, et al. Virtual network embedding through topology-aware node ranking[J]. ACM SIGCOMM Computer Communication Review, 2011, 41(2): 38-47.

[11] MELO M, SARGENTO S, KILLAT U, et al. Optimal virtual network embedding: node-link formulation[J]. IEEE Transactions on Network and Service Management, 2013, 10(4): 356-368.



- [12] WU S, CHEN N, XIAO A L, et al. AI-empowered virtual network embedding: a comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2024(99): 1.
- [13] 丁世飞, 杜威, 张健, 等. 多智能体深度强化学习研究进展[J]. 计算机学报, 2024, 47(7): 1547-1567.
DING S F, DU W, ZHANG J, et al. Research progress of multi-agent deep reinforcement learning[J]. Chinese Journal of Computers, 2024, 47(7): 1547-1567.
- [14] YAO H P, MA S H, WANG J J, et al. A continuous-decision virtual network embedding scheme relying on reinforcement learning[J]. IEEE Transactions on Network and Service Management, 2020, 17(2): 864-875.
- [15] YAN Z X, GE J G, WU Y L, et al. Automatic virtual network embedding: a deep reinforcement learning approach with graph convolutional networks[J]. IEEE Journal on Selected Areas in Communications, 2020, 38(6): 1040-1057.
- [16] ZHANG P Y, CHEN N, LI S B, et al. Multi-domain virtual network embedding algorithm based on horizontal federated learning[J]. IEEE Transactions on Information Forensics and Security, 2023, 18: 3363-3375.
- [17] ZHANG P Y, CHEN N, SHEN S G, et al. AI-enabled space-air-ground integrated networks: management and optimization[J]. IEEE Network, 2024, 38(2): 186-192.
- [18] CHEN N, ZHANG P Y, KUMAR N, et al. Spectral graph theory-based virtual network embedding for vehicular fog computing: a deep reinforcement learning architecture[J]. Knowledge-Based Systems, 2022, 257: 109931.
- [19] ZHANG P Y, CHEN N, XU G J, et al. Multi-target-aware dynamic resource scheduling for cloud-fog-edge multi-tier computing network[J]. IEEE Transactions on Intelligent Transportation Systems, 2024, 25(5): 3885-3897.
- [20] LIM H K, ULLAH I, KIM J B, et al. Virtual network embedding based on hierarchical cooperative multiagent reinforcement learning[J]. IEEE Internet of Things Journal, 2024, 11(5): 8552-8568.
- [21] 余竞航, 赵一辰, 宋浒. 基于强化学习的边缘计算智能电网资源调度算法[J]. 电信科学, 2024, 40(1): 115-122.
YU J H, ZHAO Y C, SONG H. Edge computing smart grid resource scheduling algorithm based on reinforcement learning[J]. Telecommunications Science, 2024, 40(1): 115-122.
- [22] VAN HASSELT H, GUEZ A, SILVER D. Deep reinforcement learning with double Q-learning[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2016, 30(1): 1-7.
- [23] XIAO A L, CHEN N, WU S, et al. DNFS-VNE: deep neuro fuzzy system driven virtual network embedding[J]. IEEE Internet of Things Journal, 2024(99): 1.

[作者简介]



董玉池 (1988-), 男, 现就职于中国铁塔股份有限公司通信技术研究院, 主要研究方向为边缘计算、算力网络和在网计算等。



闫亚旗 (1988-), 男, 中国铁塔股份有限公司通信技术研究院高级工程师, 主要研究方向为物联网、边缘计算、算力网络相关技术及产品创新。



冉沛 (1981-), 男, 中国铁塔股份有限公司通信技术研究院高级工程师, 主要研究方向为云及边缘计算、AI工程、行业数字化应用等。



王东 (1994-), 男, 现就职于中国铁塔股份有限公司通信技术研究院, 主要研究方向为云及边缘计算。



张阔 (1988-), 男, 中国铁塔股份有限公司通信技术研究院高级工程师, 主要研究方向为边缘计算、算力网络。



张文龙 (1998-), 男, 现就职于中国铁塔股份有限公司通信技术研究院, 主要研究方向为边缘计算、算力网络等。