基于深度强化学习的无人机辅助移动边缘计算优化

李旭东 [1]
LI Xudong

摘 要 针对在复杂障碍物环境中的无人机辅助移动边缘计算优化问题，文章将柔性 Actor-Critic（SAC）算法与分层式经验回放机制有机结合，提出 SAC-HER 算法对无人机的飞行轨迹优化。该算法通过分层筛选关键交互经验，强化对高价值场景的学习，提高学习过程对特定经验的学习效率，以提升算法收敛速度与决策精度。实验结果表明，所提算法能够有效优化无人机轨迹，降低用户能耗，同时增强在动态障碍物场景下的适应性。

关键词 深度强化学习；无人机优化；移动边缘计算

doi：10.3969/j.issn.1672-9528.2025.09.019

0 引言

随着 5G 技术的快速发展，终端产生的计算密集型任务量呈现指数级增长趋势，终端用户对计算能力的需求也愈发提高。移动边缘计算（mobile edge computing, MEC）技术将用户的计算密集型任务卸载到边缘节点处理，能够有效缓解用户在计算资源上的压力。传统静态部署的边缘计算节点在应对终端用户高动态特性时面临显著挑战：一方面终端设备的空间移动性导致服务连续性难以保障；另一方面区域性业务流量产生的潮汐效应造成边缘计算资源池利用率不足，这些技术瓶颈制约了 MEC 系统在车联网、工业互联网等高动态场景中的规模化部署。

无人机凭借其灵活部署、超高机动性等特点，搭载 MEC 服务器后可作为空中基站提供空中边缘计算服务 [1-2]，能够在密集区域进行快速部署，提高通信网络覆盖范围和无线通信链路质量，是对传统地面固定基站的有效补充。文献 [3] 为缓解现有 MEC 技术在面对用户爆炸增长及网络设施分布稀疏等情况，使用无人机来辅助 MEC 系统，并使用凸优化方法来优化无人机的位置、通信和资源分配。文献 [4] 在无人机辅助车辆任务的环境中，使用基于近端策略优化（PPO）的 DRL 算法优化无人机任务卸载和功率。文献 [5] 针对无人机辅助多用户移动边缘计算系统中用户平均能耗优化问题，提出使用柔性参与者－评论者（SAC）算法，联合优化无人机轨迹和用户计算策略。然而，目前针对无人机辅助移动边缘计算的研究多在理想环境下进行，较少考虑到环境中存在的障碍物干扰，因此本文采用深度强化学习算法对存在障碍物干扰环境下的无人机辅助移动边缘计算问题进行研究。

针对上述问题，本文基于柔性参与者－评论者（soft actor-critic, SAC）算法做出改进，结合分层式强化经验回放机制，提出 SAC-HER 算法应用于复杂环境下的无人机辅助移动边缘计算系统。

1 基于改进 SAC 的无人机优化算法

强化学习算法通过利用智能体与环境交互得到的反馈进行学习。本文以无人机作为智能体，在柔性参与者－评论者（soft actor-critic, SAC）算法的基础上，与分层式经验回放机制进行有机结合，通过额外构造的具有更高探索度的 Actor 网络来针对性收集经验样本，使得算法具有更高的探索性与收敛性。

1.1 SAC 算法原理

SAC 算法使用熵正则化（entropy regularization）的方法自动平衡算法的探索与开发策略，是十分高效的深度强化学习算法。算法通过在强化学习的目标中添加一项熵的正则项，进而通过优化熵值起到调控算法的效果。该算法定义最优策略为：

$$\pi^{*}=\arg\max_{\pi}\mathrm{E}\left[\sum_{t}^{\infty}\gamma^{t}r(s(t),a(t))+\alpha H(\pi_{\varphi}(\cdot\mid s(t+1)))\right] \tag{1}$$

式中：γ 为奖励 r 的折扣因子；α 为熵的正则化系数；熵 $H(\pi(\cdot|s))$ 表示策略 π 在状态 s 下的随机程度，其计算方式为：

$$H(\pi_{\varphi}(\cdot\mid s(t+1)))=\mathrm{E}_{s(t+1)}[-\lg\pi_{\varphi}(\cdot\mid s(t+1))] \tag{2}$$

1. 三峡大学计算机与信息学院 湖北宜昌 443002

动作价值函数 $Q(s(t), a(t))$ 由贝尔曼方程可表示为：

$$Q(s(t),a(t)) = r(t) + \gamma \mathrm{E}_{(s(t+1),a(t+1))} \cdot [Q(s(t+1),a(t+1)) - \lg \pi_{\varphi}(\cdot \mid s(t+1))] \tag{3}$$

TD 目标 $\hat{y}_t$ 可定义为：

$$\hat{y}_t = r(t) + \gamma[\min_{i=1,2} Q_{\hat{\theta}_i}(s(t+1), \tilde{a}(t+1) - \lg \pi_{\varphi}(\cdot \mid s(t+1))] \tag{4}$$

式中：$\tilde{a}(t+1)$ 为 Actor 根据状态 $s(t+1)$ 生成，并非实际动作也无需智能体执行。

Critic 网络的损失函数 $L_{C_i}(\theta)$ 为：

$$L_{C_i}(\theta) = \mathrm{E}[(Q_{\theta_i}(s(t),a(t)) - \hat{y}_t)^2] \tag{5}$$

Actor 网络的损失函数 $L_A(\theta)$ 为：

$$L_A(\theta) = \max_{\phi} \mathrm{E}[\min_{i=1,2} Q_{\hat{\theta}_i}(s(t),\tilde{a}(t)) - \alpha \lg \pi_{\phi}(\tilde{a}(t) \mid s(t))] \tag{6}$$

据上述损失函数使用梯度下降更新网络参数 ϕ 和 θ_i，对应的目标网络则使用软更新方法进行更新，即 $\hat{\theta}_i \leftarrow \tau\theta_i + (1-\tau)\hat{\theta}_i$，$\tau$ 为软更新系数本文取常用值 0.002。

通过优化熵的正则化系数 α 达到逐步降低熵 H 的效果，损失函数 $L(\alpha)$ 定义为：

$$L(\alpha) = -\lg \alpha(\lg \pi_{\varphi}(\cdot \mid s(t+1)) + \overline{H}) \tag{7}$$

式中：$\overline{H}$ 为目标熵值，设置动作维数的 -2 倍。

1.2 SAC-HER 算法框架

本文所提算法如图 1 所示，算法采用 Actor-Critic 网络框架，结合分层式经验回放机制，以提升算法对特定样本经验的学习效率。在该算法中，Actor 网络根据智能体对环境的观测输出动作。Critic 网络则用于评价 Actor 网络表现的好坏。SAC 算法构造了两个参数不同的 Critic 网络辅助训练 Actor 网络，以缓解高估问题。其输入为状态与动作，输出为 Q 值。Actor 网络根据 Critic 网络输出的 Q 值做梯度上升以更新网络参数。Critic 网络则采用时序差分（temporal difference, TD）方法更新参数。同时采用目标网络，以提高训练时的稳定性。算法采用主经验缓冲区与辅助经验缓冲区两个经验样本池。其中主经验缓冲区容量更大，其存储的经验样本更丰富，经过随机采样的经验样本用于保证算法稳定；而辅助经验缓冲区的容量较小，存储其中的经验样本能够高效地被采样，以提升算法对其中经验的学习效率。辅助经验缓冲区中的样本由辅助 Actor 网络产生，该网络与 Actor 网络参数一致，但在算法运行过程中仅从某一特定状态 s_t 而非从 s_0 开始运行，以达到快速收集关键经验的作用。

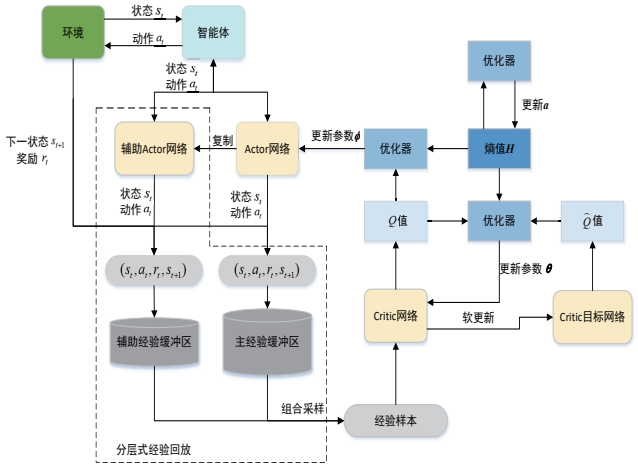

图 1 SAC-HER 算法示意图

算法所采用的神经网络中，Actor 网络由 1 个输入层、1 个 LSTM 层、1 个隐藏层以及 2 个输出层所组成。输入层神经元数量与状态 s_t 长度相同，输出层神经元数量与动作维数相同。Critic 网络由 1 个输入层、2 个隐藏层以及 1 个输出层所组成。输入层神经元数量为状态长度与动作维数之和，输出层神经元数量为 1。采用 ReLU 函数作为神经网络激活函数。

2 仿真验证与分析

2.1 实验场景与参数设置

本文所考虑的场景由 1 架搭载 MEC 服务器的无人机、3 个地面用户所组成的用户群，以及多个禁止飞跃的建筑群所组成。其中，无人机具有固定起始地点，以安全高度 H 在任务区域内飞行，并为地面上的用户提供计算服务。而地面用户满足潮汐分布，即在某些连续时刻内、小范围区域中集中出现计算资源不足。因此用户群由 N 个均匀分布在以 a 为边长的矩形区域内的用户组成，每个用户群中的用户在连续的时间内持续产生计算需求。

以最小时间间隔 δ 将一轮任务划分为 T 个时隙。则无人机在时隙 t 的坐标可表示为：

$$u(t) = [x_t, y_t, h] \tag{8}$$

用户在时隙 t 产生的计算任务可表示为：

$$I_i(t) = \{D_i(t), F_i(t)\} \tag{9}$$

式中：$i \in \{1, 2, \cdots, N\}$；$D_i(t)$ 和 $F_i(t)$ 分别表示第 i 个用户在 t 时刻的计算任务量以及完成该任务所需的 CPU 周期数，其中 $D_i \in [10, 100]$ kB，$F_i \in [1\times10^8, 1\times10^9]$ Hz。

每个用户群仅活跃 30 个时隙，即在时隙 $t = 30$ 时，第 1 个用户群停止产生任务，第 2 个用户群中的用户向无人发送辅助计算请求。

无人机在接收到用户的请求后，获得用户的坐标 w_i=[x_i, y_i, 0]，并飞向用户群的中心点，为附近用户提供计算服务。

当无人机为一定数量的用户群完成计算服务后，无人机将返回起始点，此时视为无人机完成一轮任务。最大训练幕数 $M=3\,000$ 次，每幕循环无人机最大飞行步数 $T=120$ 步。

为验证所提算法性能，本章使用计算机仿真方法对所提算法进行模拟实验，所用软件环境为 Python 3.7 同时采用 PyTorch 深度学习框架，硬件平台为具有 Intel Core i9-12900H 2.50 GHz 处理器、NVIDIA RTX 3060 显卡的个人电脑。

2.2 实验结果与分析

本文分别设置 SAC 算法、与 TD3 算法作为所提 SAC-HER 算法方案的对比方案。

图 2 展示了各方案下的无人机辅助用户通信的飞行轨迹，其中本文所提 SAC-HER 算法下无人机轨迹平滑程度优于其余两种对比算法。

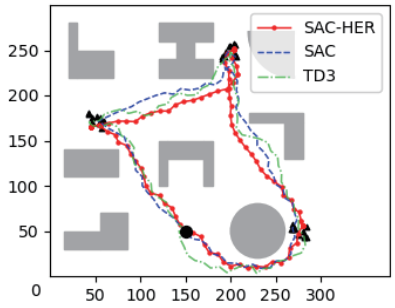

图 2 飞行轨迹示意图

表 1 为 100 次重复实验后得出的平均用户能耗与无人机的平均飞行轨迹长度。

表 1 实验结果

优化方案	用户能耗 /J	飞行轨迹长度 /m
base	12.43	—
SAC-HER	10.17	765
SAC	10.82	769
TD3	10.65	822

表 1 数据表明，采用强化学习算法优化无人机辅助移动边缘计算问题能够有效降低用户平均能耗，对比不使用无人机辅助的 base 方案，采用本文 SAC-HER 算法方案的用户能耗下降了 1.55 J，可为用户节约至少 12% 的能耗。

图 3 为各算法方案在该三维环境中的平滑处理后奖励收敛曲线图。本文所提 SAC-HER 算法在总计 3 000 幕的训练过程中，在 400 左右即达到收敛，且其收敛值高于其余两种对比方案；SAC 算法与 TD3 算法均在 700 幕左右达到收敛，且其收敛值均低于所提算法。对比本文所提 SAC-HER 算法与基础 SAC 算法，所提算法收敛速度更快，而 SAC 算法在收敛过程中具有明显阶梯性。以上结果表明所提算法能够有效提升无人机对环境探索与开发能力，进而提高算法收敛性。

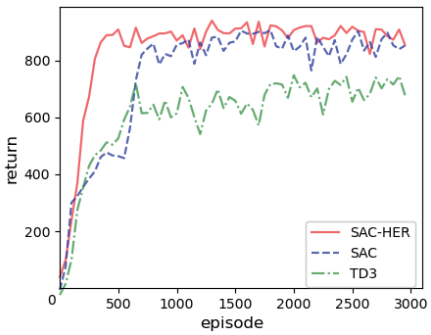

图 3 收敛曲线图

此外，在 3 种算法均能够有效控制无人机躲避环境中的障碍物，表明基于强化学习的无人机优化算法能够有效解决无人机辅助移动边缘计算时的避障问题。而本文所提 SAC-HER 算法无人机飞行轨迹长度最短，证明所提算法能够有效控制无人机选择更为优秀的路线躲避障碍物。

3 结论

为解决无人机在障碍环境下的辅助移动通信问题，本文提出了一种基于深度强化学习的无人机导航及资源优化策略。该方法通过引入分层式经验回放机制，以提高无人机在训练过程中获得经验的利用效率。实验表明所提算法具有优秀的收敛性能。该方法能够在解决无人机在辅助障碍环境下的导航问题的同时，对用户平均能耗进行显著优化。

参考文献：

[1] ABRAR M, ALMOHAIMEED Z M, AJMAL U, et al. Resource management in UAV enabled MEC networks [J]. Computers materials & continua, 2023, 74(3): 4847-4860.

[2] CHEN P P, LUO X S, GUO D K, et al. Secure task offloading for MEC-Aided-UAV system [J]. IEEE transactions on intelligent vehicles, 2023, 8(5): 3444-3457.

[3] YU Z, GONG Y M, GONG S M, et al. Joint task offloading and resource allocation in UAV-Enabled mobile edge computing [J]. IEEE internet of things journal, 2020, 7(4): 3147-3159.

[4] 谭国平，易文雄，周思源，等．无人机辅助 MEC 车辆任务卸载与功率控制近端策略优化算法 [J]. 电子与信息学报，2024, 46(6): 2361-2371.

[5] 张广驰，何梓楠，崔苗．基于深度强化学习的无人机辅助移动边缘计算系统能耗优化 [J]. 电子与信息学报，2023, 45(5): 1635-1643.

【作者简介】

李旭东（1999—），男，安徽淮北人，硕士研究生，研究方向：无人机优化。

（收稿日期：2025-05-15 修回日期：2025-09-16）