

多指标意图驱动的无人机计算卸载与轨迹规划自适应优化策略

林鹏¹, 黄新梁², 宁兆龙³, 刘艳¹, 郭磊⁴, 张治中²

(1. 南京信息工程大学复杂系统与智能保障技术教育部重点实验室, 江苏 南京 210044;

2. 南京信息工程大学电子与信息工程学院, 江苏 南京 210044; 3. 重庆邮电大学通信与信息工程学院, 重庆 400065;

4. 东北大学计算机科学与工程学院, 辽宁 沈阳 110189)

摘要: 无人机辅助边缘计算系统为提升移动网络服务性能、缓解算力不足提供了有效方案。针对低空复杂业务场景, 提出一种多指标意图驱动的无人机计算卸载与轨迹规划策略, 综合考虑服务时延、能耗及负载均衡的动态变化对系统决策的影响, 构建以动态多指标意图为目标的联合优化问题。将目标参数动态不确定、解空间维度高、难以直接求解的原问题分解为计算卸载与资源分配、轨迹规划两个子问题, 设计元强化学习与连续凸逼近融合算法, 实现子问题的交替优化。仿真实验评估了所提算法在动态多目标优化中的收敛性, 并与现有方案对优化目标进行对比评估。实验表明, 该算法可快速适配多指标动态变化, 在均衡无人机负载的同时, 显著降低系统能耗与服务时延。

关键词: 计算卸载; 负载均衡; 轨迹优化; 资源分配; 元强化学习

中图分类号: TN92

文献标志码: A

DOI: 10.11959/j.issn.1000-436x.2026056

Multi-indicator intention-driven adaptive optimization strategy for UAV computation offloading and trajectory planning

Lin Peng¹, Huang Xinliang², Ning Zhaolong³, Liu Yan¹, Guo Lei⁴, Zhang Zhizhong²

1. Key Laboratory of Intelligent Support Technology for Complex Environments, Ministry of Education, Nanjing University of Information Science and Technology, Nanjing 210044, China

2. School of Electronic and Information Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China

3. School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

4. School of Computer Science and Engineering, Northeastern University, Shenyang 110189, China

Abstract: Unmanned aerial vehicle (UAV)-assisted mobile edge computing (MEC) systems have proven to provide an effective solution to improve the performance of mobile network services and alleviate the shortage of computing power. Targeting complex low-altitude service scenarios, a multi-indicator intent-driven strategy was proposed for UAV computation offloading and trajectory planning. The impact of dynamic changes in service delay, energy consumption and load balancing on system decisions was comprehensively considered, and a joint optimization problem with dynamic multi-indicator intents as the objective was constructed. The original problem, which was characterized by dynamic uncertainty of target parameters, high dimensionality of the solution space and difficulty in direct solution, was decomposed into two subproblems, namely computation offloading and resource allocation, and trajectory planning. A hybrid algorithm integrating meta-reinforcement learning and successive convex approximation was designed to realize the alternating optimization of the two subproblems. Simulation experiments were carried out to evaluate the convergence of the proposed algorithm in dynamic multi-objective optimization, and the optimization objectives were compared with those of existing schemes. Experimental results show that the proposed algorithm can quickly adapt to the dynamic changes of multiple indicators, and significantly reduce system energy consumption and service delay while balancing the UAV load.

Keywords: computation offloading, load balancing, trajectory optimization, resource allocation, meta-reinforcement learning

收稿日期: 2026-01-05; 修回日期: 2026-02-20

通信作者: 宁兆龙, z.ning@ieec.org

基金项目: 国家自然科学基金资助项目(No.62201271, No.62303232)

Foundation Items: The National Natural Science Foundation of China (No.62201271, No.62303232)

0 引言

新型移动互联网应用算力的高需求和人工智能 (artificial intelligence, AI) 技术的深度应用, 对移动网络的运营和管理提出了新的挑战。多接入边缘计算 (multi-access edge computing, MEC) 技术, 通过将用户终端 (user terminal, UT) 的计算任务卸载至网络边缘集中处理, 为移动网络的算力瓶颈问题提供了解决方案^[1]。无人机 (unmanned aerial vehicle, UAV) 凭借灵活部署、覆盖广的优势, 可动态充当空中移动 MEC 服务器, 通过直连或非直连链路为 UT 提供算力支持与数据传输服务, 能够有效缓解地理受限区域及高密度区域的算力瓶颈, 提升移动网络对高负载场景的算力支撑能力^[2-3]。

UAV 辅助的 MEC 系统中, 服务时延和传输效率是需要关注的首要问题。对于时延敏感的移动计算任务, 如何减少 MEC 至 UT 的服务时延至关重要。文献[4]提出了一种基于速率分割多址接入的多 MEC 节点计算卸载和资源分配策略, 实现了对无线传输和计算资源分配的联合优化。文献[5]针对射频 (RF) 供电的 MEC 网络, 提出了一种基于智能反射面的 MEC 方案, 通过联合优化反射面的相移矩阵、计算资源分配以及多用户检测系数实现时延最小化。文献[6]研究了面向感知时延的服务部署和任务分配问题。基于问题的 NP-难属性, 提出了一种基于混合整数线性规划的局部搜索算法, 降低了 MEC 系统服务总成本和用户时延。文献[7]研究了 UAV 位置部署、地面 UT 接入控制和任务卸载联合优化问题, 针对联合优化导致的非凸属性, 将该问题分解为 3 个子问题, 并设计了交替优化算法有效降低了系统响应时延。文献[8]综合考虑了时延与负载公平性问题, 使用队列模型对系统建模, 设计了基于单一策略奖励机制的最大熵强化学习算法对系统时延进行优化。上述研究以优化服务时延为核心目标, 从无线传输、资源分配、节点部署等多个维度展开探索, 为保证 UAV 辅助 MEC 系统的时延性能提供了多种技术路径。但在低空复杂场景下未充分考虑服务时延、能耗效率的动态协同需求, 缺乏对时延与能耗动态需求的自适应响应机制。

能耗约束与多目标优化需求的不平衡问题, 严重制约了 UAV 辅助 MEC 系统的服务稳定性与场景

适应性。文献[9]针对 MEC 辅助 5G 网络面临的计算资源受限与能效不平衡问题, 提出了一种融合对抗式深度 Q 网络与线性规划的算法, 通过对吞吐量和负载公平性等指标进行优化, 解决了 MEC 系统在计算密集型场景的能效和资源分配问题。文献[10]提出了一种基于强化学习的随机博弈模型, 通过动态聚类 and 调度方法解决了 UAV 集群竞争和协作过程中的长期能效优化问题。文献[11]探索了 UAV 能效与轨迹规划的多目标优化问题, 其利用改进型多目标强化学习寻求帕累托最优。该方案在面临不同决策任务阶段需更改模型权重实现任务策略切换。文献[12]研究了 UAV 在不同通信模式下的资源分配和能量效率问题, 设计了针对非正交多址接入 (NOMA)、频分多址接入 (FDMA) 和时分多址接入 (TDMA) 3 种模式下的能量最小化的交替优化算法, 优化了任务卸载比例、卸载时间和 UAV 位置。文献[13]研究了 UAV 系统在轻负载状态下的路径规划和任务卸载问题, 在能源效率、任务时间、任务优先级等多重约束条件下, 有效降低了 MEC 系统总成本与服务时间。针对空地协同的 UAV 辅助 MEC, 文献[14]研究了计算密集型应用算力覆盖不足的问题, 通过构建 UAV 与地面基站协作框架, 联合优化用户关联、资源分配及 UAV 轨迹, 显著提升了系统计算效率与服务性能。文献[15]将 UAV 空地协同边缘计算的核心优化问题分解为任务调度、UAV 轨迹优化、传输与计算资源分配 3 个子问题, 并使用连续凸逼近 (successive convex approximation, SCA) 算法解决空地协同的能耗与任务效率问题。上述研究围绕 MEC 系统的能耗问题, 从算法设计、场景适配、多目标优化等维度开展了 UAV 轨迹优化与任务卸载策略研究, 为系统性能提升提供了多样化解决方案。然而, 在面临复杂业务场景时, 对系统能耗、时延与负载多个指标的变化缺乏敏感性, 仍需进一步探索。

尽管 UAV 辅助的 MEC 系统资源配置策略已得到广泛研究, 但在低空复杂业务场景下仍面临一些亟待解决的问题。现有方案一方面聚焦于对单一性能目标的极致优化, 忽略了 UAV 部署与资源配置策略对服务时延、能耗效率及负载均衡等多维度指标的综合影响。另一方面, 现有研究中面向多目标联合优化的资源配置方案大都忽略了动态业务场景中多个指标意图的实时变化。为此,

本文从多个性能指标的动态协同优化入手,通过联合设计 UAV 轨迹规划、计算卸载与资源分配策略,实现对任务卸载服务时延、系统能耗及负载均衡指数的动态联合优化,从而提升低空场景下 UAV 辅助 MEC 系统的整体服务效能。本文的主要贡献概括如下。

1)提出了一种面向时延、能耗与负载均衡的多指标意图驱动 UAV 轨迹规划与资源分配策略。考虑到实际环境中场景切换导致的多个指标意图的动态变化,本文通过设置动态关联参数来模拟指标意图的切换。进而将 UAV 轨迹规划、计算卸载与资源分配过程建模为多指标动态优化问题。

2)由于优化问题中目标参数具有不确定性,造成解空间维度扩大,直接求解困难。本文将原问题分解为 UAV 计算卸载与资源分配和 UAV 轨迹规划两个子问题,并分别设计了元强化学习算法和连续凸逼近方法对两个子问题进行交替决策优化。

3)在仿真实验中,本文将所提方案与现有方案在算法收敛速度、多目标适应性,以及 UAV 多个指标方面进行比较。结果表明,所提方案在以上方面均具有显著优势。

1 系统模型

多指标意图驱动的 UAV 辅助边缘计算场景如图 1 所示, N 个 UAV 在 K 个基站 (base station, BS) 的覆盖下为 M 个 UT 提供边缘计算服务。场景中 UT 持续产生计算任务,定义 UT m 产生的计算任务为 $\mathbf{W}_m = (D_m, C_m, T_m^{\max}, \rho_{m,t})$, 其中 D_m 是任务数据大小, C_m 是计算任务所需的 CPU 周期数, T_m^{\max} 是任务 \mathbf{W}_m 时延要求, $\rho_{m,t} \in [0,1]$ 为 UT m 在 t 时隙根据自身负载状态标记的任务卸载比例。UT 持续产生计算任务并将部分计算任务卸载至 BS 或 UAV 中进行处理。为便于描述与系统,本文通过时隙 $t \in T = \{1,2,\dots,+\infty\}$ 对系统运行状态进行划分,相邻时隙的时间间隔定义为 $\Delta\tau$ 。系统中 UAV 始终保持在高度 H 飞行以维持其服务状态,并避免轨迹重叠。当 UAV n 辅助任务计算过程中,任务 \mathbf{W}_m 首先经历从 UT m 到 UAV n 的地对空 (ground to air, G2A) 传输。随后, UAV n 作出对任务 \mathbf{W}_m 的处理决策,并将剩余部分任务通过空对地 (air to ground, A2G) 链路从 UAV n 传输到 BS k 。最终, BS k 执行剩余任务计算,并将结果返回 UT m 。

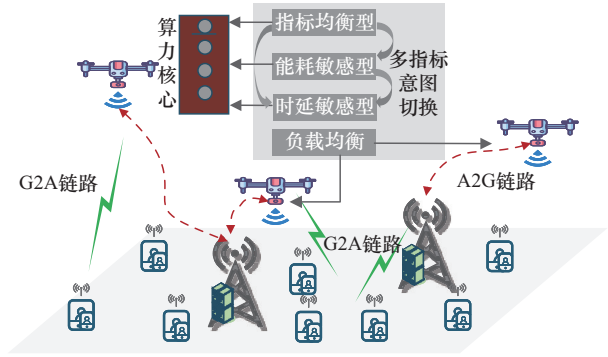


图1 多指标意图驱动的 UAV 辅助边缘计算场景

1.1 UAV 移动模型

UAV n 的空间位置坐标表示为 $\omega_n(t) = [x_n(t), y_n(t), H]^T$, 其中 $x_n(t)$ 和 $y_n(t)$ 分别表示 UAV n 在时隙 t 的平面轴坐标, H 为 UAV n 的固定高度。UAV n 在 t 时隙水平飞行距离为 $l_n(t)$, 角度方向为 $\theta_n(t) \in [0, 2\pi]$, 则有

$$\begin{aligned} x_n(t+1) &= x_n(t) + l_n(t) \cos \theta_n(t) \\ y_n(t+1) &= y_n(t) + l_n(t) \sin \theta_n(t) \end{aligned} \quad (1)$$

UAV n 在有限的服务区域内移动,并基于距离、链路状态与 UT 进行实时关联。定义 UAV n 的天线最大功率波束宽度为 $2\varphi_n$, 则 t 时隙 UAV n 的最大覆盖半径 $C_{\max}^n(t)$ 可表示为

$$C_{\max}^n(t) = H \times \tan(\varphi_n) \quad (2)$$

为了确保 UAV 的服务覆盖范围不重叠,必须满足以下覆盖约束条件。

$$\|\omega_n(t) - \omega_j(t)\| \geq C_{\max}^n(t) + C_{\max}^j(t), \forall n, \forall j, n \neq j \quad (3)$$

1.2 通信模型

定义 UT m 的平面坐标为 $\omega_{m,0}(t) = [x_m(t), y_m(t), 0]^T$, 基于 UAV n 的位置坐标 $\omega_n(t)$, 可得 UAV n 和 UT m 之间的距离为

$$d_{m,n}(t) = \|\omega_n(t) - \omega_m(t)\| \quad (4)$$

UT 与 UAV 之间的通信采用正交频分多址方法,因此,可忽略 UT 之间的通信干扰。UAV n 与 UT m 之间的传输路径损耗可表示为

$$h_{m,n}(t) = K_o d_{m,n}^{-2} [P_{m,n}^{\text{LoS}} \mu_{\text{LoS}} + P_{m,n}^{\text{NLoS}} \mu_{\text{NLoS}}] \quad (5)$$

其中, K_o 为路径损耗常数, μ_{LoS} 为直视链路衰落变量, μ_{NLoS} 为非直视链路的衰落变量, $P_{m,n}^{\text{LoS}}$ 为直视链路概率, $P_{m,n}^{\text{NLoS}}$ 为非直视链路概率。

任务卸载过程中, UT 只能将计算任务卸载至

与其相关联的 UAV 上。定义 $z_{m,n}(t)$ 为关联变量, $z_{m,n}(t) = 1$ 表示 t 时隙 UT m 与 UAV n 处于关联状态, 否则, $z_{m,n}(t) = 0$ 。定义 B_n 为 UAV n 的上行带宽, 其被平均分配给每个 UT。由此可得, UT m 和 UAV n 之间的 G2A 传输速率为

$$R_{m,n}(t) = \frac{B_n}{\sum_{m=1}^M z_{m,n}(t)} \text{lb} \left[1 + \frac{h_{m,n}(t) P_m}{\sigma_{\text{UAV}}^2} \right] \quad (6)$$

其中, P_m 是 UE m 的传输功率, σ_{UAV}^2 为 UAV 上的加性白高斯噪声功率。

类似地, UAV n 和 BS k 间的 A2G 传输信道损耗为

$$h_{n,k}(t) = K_o d_{n,k}^{-2} [P_{n,k}^{\text{LoS}} \mu_{\text{LoS}} + P_{n,k}^{\text{NLoS}} \mu_{\text{NLoS}}] \quad (7)$$

其中, $d_{n,k}(t) = \|\omega_n(t) - \omega_k\|$ 表示 UAV n 和 BS k 之间的距离。同理, UAV n 与 BS k 之间的 A2G 传输速率为

$$R_{n,k}(t) = B_{n,k} \text{lb} \left[1 + \frac{h_{n,k}(t) P_n(t)}{\sigma_{\text{BS}}^2} \right] \quad (8)$$

其中, $B_{n,k}$ 是预分配给 BS k 的带宽, $P_n(t)$ 为 UAV n 在时间 t 的传输功率, σ_{BS}^2 为噪声功率。

1.3 服务时延模型

UT m 产生的任务 W_m 的一部分被卸载到所关联的 UAV n , 随后 UAV n 确定在本地进行处理的任务比例或进一步卸载到地面 BS, 取决于 UAV 自身负载状态。定义任务 W_m 由 UT m 传输到 UAV n 的 G2A 传输时延为 $T_{m,n}^{\text{G2A}}(t)$, 可表示为

$$T_{m,n}^{\text{G2A}}(t) = \frac{\rho_{m,t} D_m}{R_{m,n}(t)} \quad (9)$$

定义 $\gamma_{m,k}^n(t) \in [0,1]$ 为 t 时隙 UAV n 将任务 W_m 的数据卸载到 BS k 的决策, 特别的, 当 $\sum_{k=1}^K \gamma_{m,k}^n(t) = 0$ 时表示任务数据完全由 UT m 进行处理。定义任务 W_m 由 UAV n 传输到 BS k 的 A2G 传输时延为 $T_{m,n,k}^{\text{A2G}}(t)$, 即

$$T_{m,n,k}^{\text{A2G}}(t) = \frac{\gamma_{m,k}^n(t) \rho_{m,t} D_m}{R_{n,k}(t)} \quad (10)$$

相应地, 定义任务数据分别由 UAV n 和 BS k 进行处理的时延为 $T_{m,n}^{\text{UAV}}(t)$ 和 $T_{m,n}^{\text{BS}}(t)$, 可表示为

$$T_{m,n}^{\text{UAV}}(t) = \frac{\left(1 - \sum_{k=1}^K \gamma_{m,k}^n(t) \right) \rho_{m,t} D_m C_m}{f_{m,n}(t)} \quad (11)$$

$$T_{m,n,k}^{\text{BS}}(t) = \frac{\gamma_{m,k}^n(t) \rho_{m,t} D_m C_m}{f_{m,k}(t)}$$

其中, $f_{m,n}(t)$ 为 UAV n 分配给任务 W_m 的计算资源, 单位为 Hz, $f_{m,k}(t)$ 为 BS k 分配给任务 W_m 的计算资源。

综上可得, 任务 W_m 经 UAV n 服务的时延可表示为

$$T_{m,n}(t) = z_{m,n}(t) \left[T_{m,n}^{\text{G2A}}(t) + \max_k \left\{ T_{m,n}^{\text{UAV}}(t), T_{m,n,k}^{\text{A2G}}(t) + T_{m,n,k}^{\text{BS}}(t) \right\} \right] \quad (12)$$

为了评估 UAV 的服务效率, 定义 t 时隙 UAV n 的时延效率函数为

$$T_n^{\text{uty}}(t) = \frac{\sum_{m=1}^M T_{m,n}(t)}{\sum_{m=1}^M z_{m,n}(t) \left(T_{m,n}^{\text{G2A}}(t) + \frac{\rho_{m,t} D_m C_m}{f_{m,n}(t)} \right)} \quad (13)$$

其中, 分母项为 t 时隙 UAV n 关联的所有任务均在本地执行的总时延。

1.4 能耗模型

UAV 任务处理过程中的能耗包括任务计算能耗、传输能耗以及飞行推进能耗。任务计算能耗与 UAV 的 CPU 运行功率及架构有关^[16]。定义 UAV n 的 CPU 运行功率为 $P_n(t) = \kappa_n (f_n)^{\varepsilon_n}$, 其中 κ_n 和 ε_n 为与 CPU 架构相关的功率参数, f_n 为 CPU 频率。考虑到 UAV 的任务处理时延 $T_{m,n}^{\text{UAV}}(t)$, UAV n 在 t 时隙的预估任务计算能耗可表示为

$$E_{m,n}^{\text{UAV}}(t) = \kappa_n \left(\sum_{\forall m \in \mathbb{M}} z_{m,n}(t) f_{m,n}(t) \right)^{\varepsilon_n} T_{m,n}^{\text{UAV}}(t) \quad (14)$$

定义 UAV n 的 G2A 传输功率为 P_n^{up} , 则 t 时隙的传输能耗可表示为

$$E_{m,n}^{\text{G2A}}(t) = P_n^{\text{up}} T_{m,n}^{\text{G2A}}(t) \quad (15)$$

相应地, UAV n 到 BS k 的 A2G 传输能耗为

$$E_{m,n,k}^{\text{A2G}}(t) = P_n^{\text{dw}} T_{m,n,k}^{\text{A2G}}(t) \quad (16)$$

其中, P_n^{dw} 是 UAV n 的发射功率。

此外, UAV n 需保持一定的推进功率维持其悬停和移动状态。本文忽略了 UAV 加速和减速所造成的能量消耗。对于移动速度为 V 的旋翼无人机,

在一个时隙 $\Delta\tau$ 的推进能耗可表示为 $E_{\text{fly}}^n(t) = P(V)\Delta\tau$, 其中, $P(V)$ 为 UAV n 在移动速度为 V 下的稳态飞行功率。

综上所述, UAV n 在 t 时隙为 UT m 提供计算服务的总能耗为

$$E_n(t) = \sum_{m=1}^M z_{m,n}(t) [E_{m,n}^{\text{G2A}}(t) + E_{m,n}^{\text{UAV}}(t) + E_{m,n,k}^{\text{A2G}}(t)] + E_{\text{fly}}^n(t) \quad (17)$$

为了评估 UAV 的负载状态, 定义 UAV n 的能量效率函数为

$$E_n^{\text{ue}}(t) = \frac{E_n(t)}{P_n^{\text{up}}\Delta\tau + \kappa_n(f_n^{\text{max}})^{\epsilon_n}\Delta\tau + E_{\text{fly}}^n(t)} \quad (18)$$

其中, $E_n^{\text{ue}}(t) \in [0,1]$ 为效率函数, f_n^{max} 为 UAV n 的 CPU 最高频率, 分母项为 UAV n 在一个时隙内全负荷运行的能耗。

1.5 服务均衡模型

UT 的分布不均匀及其动态特性导致算力需求分布存在差异, 而此时 UAV 之间因其负载状态导致局部算力供应不足将引发局部功耗过载, 进而导致服务性能下降。UAV 的负载状态可通过其在特定时隙的任务处理量来衡量。定义 $\zeta_n(t)$ 为 UAV n 在 t 时隙的实时任务处理量, 结合 t 时隙 UAV n 与 UT 关联决策 $z_{m,n}(t)$ 以及任务卸载决策 $\gamma_{m,k}^n(t)$, 任务处理量 $\zeta_n(t)$ 可表示为

$$\zeta_n(t) = \sum_{m=1}^M z_{m,n}(t) \rho_{m,t} \left(1 - \sum_{k=1}^K \gamma_{m,k}^n(t) \right) D_m, \forall n, \forall t \in T \quad (19)$$

基于任一 UAV n 的实时任务处理量 $\zeta_n(t)$, 构建 UAV 集群的负载均衡评估函数为

$$J^{\text{UAV}}(t) = \frac{\left(\sum_{n=1}^N \zeta_n(t) \right)^2}{N \sum_{n=1}^N \zeta_n^2(t)} \quad (20)$$

其中, $J^{\text{UAV}}(t) \in [\frac{1}{N}, 1]$ 且当 $J^{\text{UAV}}(t)=1$ 时, $\zeta_n(t) = \zeta_j(t), \forall j \neq n$, 此时 UAV 集群实现负载均衡^[17]。

1.6 问题描述

基于 UAV n 的能量效率函数 $E_n^{\text{ue}}(t)$ 和时延效率函数 $T_n^{\text{ue}}(t)$, 构建多指标意图函数 $U_n(t)$ 。

$$U_n(t) = \alpha E_n^{\text{ue}}(t) + \beta T_n^{\text{ue}}(t) \quad (21)$$

其中, α 和 β 为动态变化的指标意图驱动参数, 其值变化反映了系统在不同时段, 由于业务场景的改变, 对 UAV 能耗和时延的优化权重。为了构建不同场景下的优化问题, 设计了多种任务场景。初始时隙, 系统随机产生当前的任务场景 τ_i , 在该任务持续时间 Δt 结束时隙, 以概率 P_i 产生下一阶段任务, 如此循环, 以此产生任务场景序列。表 1 展示了 5 种任务场景的权重参数、场景持续时间, 以及场景触发概率 P_i 分布情况。

τ_i	场景类型	权重 α	权重 β	Δt	触发概率
τ_1	能耗优先	1.0	0	3	15%
τ_2	能耗敏感	0.7	0.3	5	20%
τ_3	均衡	0.5	0.5	8	30%
τ_4	时延敏感	0.3	0.7	10	20%
τ_5	时延优先	0	1.0	15	15%

同时, 考虑到 UAV 集群的负载均衡指数, 优化问题可构建为

$$\begin{aligned} P_1: & \min_{\gamma_{m,k}^n, z_{m,n}, \omega_n} \sum_{n=1}^N \frac{1}{J^{\text{UAV}}(t)} U_n(t) \\ \text{s.t.} & C_1: T_{m,n}(t) \leq T_m^{\text{max}}, \forall n, m \\ & C_2: \sum_{k=1}^K \gamma_{m,k}^n(t) = 1, \forall n, m \\ & C_3: \sum_{n=1}^N z_{m,n}(t) = 1, \forall m \\ & C_4: \|\omega_n(t) - \omega_j(t)\| \geq C_{\text{max}}^n(t) + C_{\text{max}}^j(t), \forall n, \forall j, n \neq j \\ & C_5: \sum_{k=1}^K \gamma_{m,k}^n(t) \leq z_{m,n}(t), \forall n, m \end{aligned} \quad (22)$$

其中, 条件 C_1 为任务 W_m 的时延容忍条件, 条件 C_2 是 UAV n 的任务卸载比例约束, 条件 C_3 表明 t 时隙 UT m 仅与一个 UAV 关联, 条件 C_4 为 UAV 的运动约束, 条件 C_5 为任务卸载的用户关联前置条件。

2 问题转化与求解

对于问题 P_1 , 在确定性指标意图参数 α 和 β 的条件下可被转换为基于动作-状态的马尔可夫决策 (Markov decision process, MDP) 问题。但由于上述优化目标动态变化, 且变量中包含 UAV 的轨迹

决策, 直接决策极易造成维数灾难并引发重复训练。为此, 本文首先将问题 P_1 分解为任务卸载与轨迹优化两个子问题, 确定可行解空间, 随后基于两个子问题的耦合关联关系建立交替优化过程, 最终实现子问题的收敛。

2.1 基于元强化学习的任务卸载与资源分配

在给定 UAV 位置和 UT 位置分布的情况下, $P(V)$ 、 $x_n(t)$ 和 $y_n(t)$ 为常量, UAV 位置以及 UAV 与 UT 之间的信道状态可认为是恒定的。此时, 问题 P_1 转化为任务卸载与资源分配优化问题。

$$\begin{aligned} P_2: \min_{\gamma_{m,k}^n, \tau_{m,n}} & \sum_{n=1}^N \frac{1}{J^{\text{UAV}}(t)} \cdot U_n(t) \\ \text{s.t.} & C_1 \sim C_3, C_5 \end{aligned} \quad (23)$$

对于问题 P_2 , 参数 α 和 β 的变化体现了实际环境中时延和能耗指标的动态不确定性。为此, 本文提出一种融合元学习的多智能体强化学习 (meta-multiagent reinforcement learning, Meta-MARL) 算法, 将动态联合优化问题转换为多任务强化学习过程, 并且能够使其适应动态变化的指标意图, 从而有效提高决策效率。

将问题 P_2 转换为具有明确状态、动作和奖励值的 MDP 问题。系统状态包含 UAV n 的位置信息、任务数据量, 以及用户关联状态。将用户关联状态 $z_{m,n}(t-1)$ 作为系统状态 $s_n(t)$ 的元素, 可避免神经网络无法建立历史关联决策与当前决策的“连续性倾向”问题, 从而帮助策略网络在训练过程中提升关联分布的稳定性和模型训练效率。因此, UAV n 在 t 时隙的状态被定义为 $s_n(t) = \{\rho_{m,t}, z_{m,n}(t-1), \omega_n(t), \forall m\}$ 。系统动作包含待执行的任务卸载与用户关联决策, 则 t 时隙 UAV n 的动作可定义为 $\mathbf{a}_n(t) = \{z_{m,n}(t), \gamma_{m,k}^n(t), \forall m, \forall k\}$ 。系统奖励以实现优化问题 P_2 为目标, 对于 UAV n , t 时隙的奖励函数可定义为 $r_n = -\frac{U_n(t)}{J^{\text{UAV}}(t)}$ 。

2.2 Meta-MARL 算法

问题 P_2 的目标参数动态变化导致决策空间的不确定性, 在训练过程中呈现出多任务属性。Meta-MARL 算法的核心目标是培养一种能够在极少量交互学习后, 迅速适配新任务的决策模型, 以适应优化指标意图的动态变化。其核心创新在于将元学习框架融入强化学习算法中, 使智能体能够通过多样化的任务集上训练, 系统性地归纳并使用跨任务

的通用知识。当面临决策目标切换或未知新任务时, 该智能体可高效复用所积累的知识, 实现决策模型的快速迁移与适应。Meta-MARL 算法在架构上包含两个模块: 基础学习器与元学习器。

1) 基础学习器

基础学习器采用多智能体深度确定性策略梯度 (multi-agent deep deterministic policy gradient, MADDPG) 算法^[18], 其任务是根据当前观测数据做出动作决策, 训练并学习如何优化其动作以最大化系统奖励。在问题 P_2 中, UAV 被视为智能体, 具备策略网络 $\mu(\mathbf{s}|\theta^\mu)$ 和价值网络 $Q(\mathbf{s}, \mathbf{a}|\theta^Q)$ 。前者输出 UAV n 在给定状态 s_n 下的动作 \mathbf{a}_n , 后者输出 UAV n 对于状态-动作 (s_n, \mathbf{a}_n) 的预期奖励 r_n 。对于 UAV n , 任一参数 $\{\alpha, \beta\}$ 组合下的决策过程可被视为一个确定的基础学习任务。定义 $\tau = (\tau_1, \tau_2, \dots, \tau_p, \dots, \tau_H)$ 为基础学习器的任务集合, 其包含了不同参数 $\{\alpha, \beta\}$ 组合的决策训练任务。对于突发的任一任务 $\tau_i \sim \tau$, 以全局参数 $\tilde{\theta}^\mu$ 和 $\tilde{\theta}^Q$ 初始化该任务的网络 $\mu_i(\mathbf{s}|\theta^\mu)$ 和 $Q_i(\mathbf{s}, \mathbf{a}|\theta^Q)$ 以及相应的目标网络参数 $\theta^{\mu'}$ 和 $\theta^{Q'}$ 。随后, UAV n 基于 $\mu_i(\mathbf{s}|\theta^\mu)$ 和 $Q_i(\mathbf{s}, \mathbf{a}|\theta^Q)$ 在状态 s_n 下执行动作 \mathbf{a}_n , 获取奖励 r_n 以及新状态 s'_n 。将 $(s_n, \mathbf{a}_n, r_n, s'_n)$ 存储在轨迹 J_i 中构建任务支撑集, 并在 J_i 中采样样本集合 h_i , 计算损失函数。

$$\begin{aligned} L_i(\theta) = \mathbb{E}_{(s_n, \mathbf{a}_n, r_n, s'_n) \sim h_i} & \left[\left(r_n + \right. \right. \\ & \left. \left. \rho Q_i(s'_n, \mu_i(s'_n|\theta^\mu)|\theta^Q) - Q_i(s_n, \mathbf{a}_n|\theta^Q) \right)^2 \right] \end{aligned} \quad (24)$$

其中, ρ 为折扣因子。随后, 基于损失函数的梯度更新相应的网络参数。

$$\theta'_i \leftarrow \theta - \text{LR} \cdot \nabla_{\theta} L_i(\theta) \quad (25)$$

其中, 更新 $\mu_i(\mathbf{s}|\theta^\mu)$ 网络参数时, LR 为 $\mu_i(\mathbf{s}|\theta^\mu)$ 网络的学习率。更新 $Q_i(\mathbf{s}, \mathbf{a}|\theta^Q)$ 网络参数时, LR 为 $Q_i(\mathbf{s}, \mathbf{a}|\theta^Q)$ 网络的学习率。使用适应后的参数 θ'_i 在 τ_i 中重新进行采样, 并再次计算任务损失 $L_i(\theta'_i)$ 。

2) 元学习器

元学习器旨在对基础学习器的不同任务学习过程进行泛化。基础学习器每完成一次更新后, 元学习器向任务集 τ 抽取不同任务最新的损失函数, 结合当前任务 τ_i 的任务损失 $L_i(\theta'_i)$, 计算总体元损失。

$$L_{\text{meta}}(\theta) = \sum_{\tau_l} L_l[\theta_l^i] \quad (26)$$

执行梯度下降,更新全局参数 $\tilde{\theta}^\mu$ 和 $\tilde{\theta}^Q$ 。

$$\tilde{\theta} \leftarrow \tilde{\theta} - \text{LR}_{\text{meta}} \cdot \nabla_{\theta} L_{\text{meta}}(\theta) \quad (27)$$

其中, LR_{meta} 为元学习率。每次元更新后,元学习率将全局参数 $\tilde{\theta}^\mu$ 和 $\tilde{\theta}^Q$ 更新到整个系统,保持模型对突发任务切换的适应性。通过多次迭代,UAV n 在面临新的决策任务时具备较好的模型参数(元知识),使其在新任务上能通过少量更新快速适应,提高任务卸载与资源分配决策效率。

2.3 基于SCA的无人机轨迹优化

在给定问题 P_1 输出决策 $z_{m,n}(t)$ 和 $\gamma_{m,k}^n(t)$ 的情况下,UT与UAV的关联状态、负载均衡指数是确定的,此时问题 P_1 被转化为UAV的轨迹优化问题。在该条件下,问题 P_1 可简化为

$$\begin{aligned} P_3: \min_{\omega_n(t)} & \sum_{n=1}^N \alpha E_n^{\text{uty}}(t) + \\ & \sum_{n=1}^N \beta \frac{\sum_{m=1}^M z_{m,n}(t) \times [T_{m,n}^{\text{G2A}}(t) + X_m]}{\sum_{m=1}^M z_{m,n}(t) \left(T_{m,n}^{\text{G2A}}(t) + \frac{\rho_{m,t} D_m C_m}{f_{m,n}(t)} \right)} \\ \text{s.t. } & C_4 \\ & X_m \geq T_{m,n}^{\text{UAV}}(t), \forall m, n \\ & X_m \geq T_{m,n,k}^{\text{A2G}}(t) + T_{m,n,k}^{\text{BS}}(t), \forall m, n, k \end{aligned} \quad (28)$$

其中, X_m 为UT m 定义的辅助变量,用于将式(12)的服务时延线性化。注意到上行链路和下行链路传输速率表达式(6)和式(8)相对于UAV位置是非凸的,问题 P_3 仍然难以求解。为此,本文针对问题 P_3 中的非凸目标函数和非凸约束构建凸近似,并通过SCA算法迭代求解得到次优解。

2.4 SCA算法

SCA算法主要思想是通过将原函数进行凸逼近并迭代优化,最终通过合适的凸逼近获得原问题的次优解。子问题 P_3 的凸逼近问题一般形式可表示为

$$\begin{aligned} P_x: \min_x & \tilde{U}(\mathbf{x}; \mathbf{x}^i) \\ \text{s.t. } & \tilde{g}_l(\mathbf{x}; \mathbf{x}^i) \leq 0 \quad \forall l = 1, \dots, m \\ & \mathbf{x} \in K \end{aligned} \quad (29)$$

其中, $\tilde{U}(\mathbf{x}; \mathbf{x}^i)$ 为子问题 P_3 的目标函数在第 i 次迭代的近似值, $\tilde{g}_l(\mathbf{x}; \mathbf{x}^i)$ 是子问题 P_3 的约束条件在第 i 次迭代的近似值, \mathbf{x}^i 为子问题 P_3 在第 i 次迭代的解向量, K 为自变量 \mathbf{x} 的取值范围。为保证SCA算

法的正确性和收敛性,要求在每次迭代中,子问题 P_3 中的原始函数都近似于它们的上界,同时保留相同的一阶行为。因此,构造上述凸近似问题需选取合适的近似函数 $\tilde{U}(\mathbf{x}; \mathbf{x}^i)$ 和 $\tilde{g}_l(\mathbf{x}; \mathbf{x}^i)$ 。文献[19]给出了构造这类非凸项近似函数的规则方法。在满足近似函数基本规则后,函数 $\tilde{g}_l(\mathbf{x}; \mathbf{x}^i)$ 可构造为具有差凸结构和乘积结构的凸近似,函数 $\tilde{U}(\mathbf{x}; \mathbf{x}^i)$ 同样可构造为具有乘积结构的凸近似。针对原问题中与UAV位置 $\omega_n(t)$ 相关的非凸函数 $\frac{1}{R_{n,k}(t)}$ 和

$\frac{1}{R_{m,n}(t)}$,由于这两个非凸项无法直接通过现有方法构造凸近似,故引入非负辅助变量 ϕ_n 和 μ_m 分别替换 $\frac{1}{R_{n,k}(t)}$ 和 $\frac{1}{R_{m,n}(t)}$,将原本依赖UAV位置的复杂非凸结构转化为“凸函数×非负凸函数”的乘积结构,从而适配文献[19]中Sample 2(乘积结构非凸项的凸上界构造方法)和Sample 3(乘积结构目标函数的凸近似方法)的应用条件。基于此,在第 i 次迭代中给定当前可行解 $\phi_n(t)^i$ 、 $\mu_m(t)^i$ 和 $\gamma_{m,k}^n(t)^i$,可直接应用上述方法构造出目标函数和约束中非凸项 $E_{m,n}^{\text{G2A}}$ 、 $E_{m,n,k}^{\text{A2G}}$ 、 $T_{m,n}^{\text{G2A}}$ 和 $T_{m,n,k}^{\text{A2G}}$ 的凸上近似 $\tilde{E}_{m,n}^{\text{G2A}}$ 、 $\tilde{E}_{m,n,k}^{\text{A2G}}$ 、 $\tilde{T}_{m,n}^{\text{G2A}}$ 、 $\tilde{T}_{m,n,k}^{\text{A2G}}$ 。

综上,将优化问题的决策变量集表示为 $\psi(t) = (X_m, \phi_n, \mu_m, \gamma_{m,k}^n, \omega_n)$ 。在第 i 次迭代中,给定由SCA算法产生的可行解 $\psi(t)^i$,可获得重新表述后的问题 P_3 的凸上近似形式。该问题可直接通过凸优化工具箱(CVX)快速求解。

问题 P_3 的凸上近似形式为

$$\begin{aligned} P_4: \min_{\omega_n(t)} & \sum_{n=1}^N \left\{ \alpha \tilde{E}_n^{\text{uty}}(\psi(t); \psi(t)^i) + \beta \tilde{T}_n^{\text{uty}}(\psi(t); \psi(t)^i) \right\} \\ \text{s.t. } & X_m \geq T_{m,n}^{\text{UAV}}(\psi(t); \psi(t)^i), \forall m, n \\ & X_m \geq \tilde{T}_{m,n,k}^{\text{A2G}}(\psi(t); \psi(t)^i) + T_{m,n,k}^{\text{BS}}(\psi(t); \psi(t)^i), \forall m, n \\ & 0 \leq \phi_n \leq R_{n,k,\text{LB}}(\psi(t); \psi(t)^i), \quad \forall n \\ & 0 \leq \mu_m \leq R_{m,n,\text{LB}}(\psi(t); \psi(t)^i), \quad \forall m \\ & C_4 \end{aligned} \quad (30)$$

其中, $\psi(t); \psi(t)^i$ 表示以第 i 次迭代参考点 $\psi(t)^i$ 为锚点对优化变量 $\psi(t)$ 构建近似函数的参数分隔。

2.5 基于Meta-MARL-SCA的交替决策优化

问题 P_1 经过分解后可分别通过Meta-MARL算法和SCA算法进行交替决策求解。首先,对问题 P_2

利用 Meta-MARL 算法作为上层策略输出当前任务下的决策 $z_{m,n}(t)$ 和 $\gamma_{m,k}^n(t)$, 继而固定该决策。采用 SCA 算法进行轨迹决策时将 $z_{mn}(t)$ 、 $\gamma_{mk}^n(t)$ 作为约束条件, 其输出的 UAV 轨迹 $\omega_n(t)$ 反向作为 Meta-MARL 算法的输入, 进行强化学习训练。每次迭代时, 两个算法互相耦合作用, 随着子问题 P_2 和 P_3 逐渐收敛, 满足终止准则 $0 \leq \frac{U(t) - U(t-1)}{U(t-1)} < \varepsilon$, 最终实现对原问题 P_1 的优化。具体算法如算法 1 所示。

算法 1 基于 Meta-MARL-SCA 的交替决策算法

输入 UAV 数量 N , BS 数量 K , UT 数量 M , 任务 $W_m, I_{\max}, LR_\mu, LR_Q, LR_{\text{meta}}$

输出 $z_{mn}(t), \gamma_{mk}^n(t), \omega_n(t), \varepsilon$

初始化 意图参数 $\{\alpha, \beta\}$, $\theta^\mu, \theta^\varrho, \tilde{\theta}^\mu, \tilde{\theta}^\varrho$, 任务集合 $\tau, \omega_n(0), \eta, \xi, \delta$

1) for $e=1$ to e_{\max} do

2) 获取 t 时隙的决策任务 $\tau_t \sim \tau$, 确定指标意图参数 $\{\alpha, \beta\}_t$, 以及各 UAV 位置 $\omega_n(t)$

3) if $\{\alpha, \beta\}_t \neq \{\alpha, \beta\}_{t-1}$ 且 $\tau_t \notin \tau$ then

4) 对目标任务 τ_t 初始化网络参数:
 $\mu_i(s|\theta^\mu) \leftarrow \tilde{\theta}^\mu, Q_i(s,a|\theta^\varrho) \leftarrow \tilde{\theta}^\varrho$

5) end if

6) for $t=1$ to T_{\max} do

7) for $n=1$ to N do

8) UAV n 执行动作 $\mathbf{a}_n(t) = \mu_i(s_n|\theta^\mu)$, 并获取奖励值 $r_n = Q_i(s_n, \mathbf{a}_n|\theta^\varrho)$ 和状态 $s_n(t+1)$

9) 将元组 $(s_n, \mathbf{a}_n, r_n, s_n')$ 存储在轨迹 J_t 中构建任务支撑集

10) 在支撑集 J_t 中采样任务样本集 h_t , 并基于式(24)计算任务损失函数 $L_t(\theta)$

11) 更新策略网络 $\mu_i(s|\theta^\mu)$ 和价值网络 $Q_i(s,a|\theta^\varrho)$ 的参数

12)
$$\begin{aligned} \theta^\mu &\leftarrow \theta^\mu - LR_\mu \cdot \nabla_{\theta^\mu} L_t(\theta^\mu), \\ \theta^\varrho &\leftarrow \theta^\varrho - LR_Q \cdot \nabla_{\theta^\varrho} L_t(\theta^\varrho) \end{aligned}$$

13) end for

14) end for

15) 基于更新后的网络采样新的样本集 h'_t , 并基于式(24)计算新的任务损失 $L_t(\theta'_t)$

16) 计算元损失函数: $L_{\text{meta}}(\theta) = \sum_{\tau_t} L_t[\theta'_t]$

17) 执行梯度下降, 更新全局参数 $\tilde{\theta}^\mu$ 和 $\tilde{\theta}^\varrho$

18)
$$\begin{aligned} \tilde{\theta}^\mu &\leftarrow \tilde{\theta}^\mu - LR_{\text{meta}} \cdot \nabla_{\theta^\mu} L_{\text{meta}}(\theta^\mu), \\ \tilde{\theta}^\varrho &\leftarrow \tilde{\theta}^\varrho - LR_{\text{meta}} \cdot \nabla_{\theta^\varrho} L_{\text{meta}}(\theta^\varrho) \end{aligned}$$

19) 通过 $\mathbf{a}_n(t)$ 获取 $z_{m,n}(t)$ 和 $\gamma_{m,k}^n(t)$, 计算式(28)中与 X_m 约束相关的时延以及 $R_{n,k}(t)$ 和 $R_{m,n}(t)$

20) 初始化 $X_m(t), \phi_n(t)$ 和 $\mu_m(t)$, $\text{Converged} \leftarrow \text{FALSE}$, $\eta(i) = \eta$, 构造变量集 $\psi(t)^i, i = 0$

21) for $i=1$ to I_{\max} do

22) 基于当前可行解 $\psi(t)^{i-1}$, 构造凸近似目标函数和凸约束

23) 使用 CVX 求解子问题 P_4 得到本次迭代最优解 $\tilde{\psi}(\psi(t)^{i-1})$

24) 更新可行解 $\psi(t)^i = \psi(t)^{i-1} + \eta(i-1)(\tilde{\psi}(\psi(t)^{i-1}) - \psi(t)^{i-1})$, $\eta(i) = \eta(i-1)(1 - v\eta(i))$

25) if $\|\tilde{\psi}(\psi(t)^{i-1}) - \psi(t)^{i-1}\| \leq \xi$ then

26) $\text{Converged} \leftarrow \text{true}$

27) break

28) end if

29) end for

30) 基于 SCA 最优解 $\psi(t)^i$ 更新 UAV 位置 $\omega_n(t)$

31) end for

2.6 算法复杂度分析

算法的复杂度由 Meta-MARL 模块与 SCA 模块共同决定。Meta-MARL 算法中, UAV 的状态维度为 d_s , 动作维度为 d_a , 隐藏层神经元数为 N_h , 任务集规模为 T , 基础学习器迭代次数为 K , 则 UAV 策略网络前向传播复杂度为 $O(d_s \times N_h + N_h \times d_a)$, 反向传播复杂度为 $O((d_s \times N_h + N_h \times d_a) \times N_h)$ 。综合考虑元学习器中的任务集合维度 T , 可得 Meta-MARL 算法复杂度为 $O(TK(d_s + d_a)N_h^2)$ 。属于多项式复杂度。SCA 算法中, 决策变量维度为 d_x , 设定最大迭代次数为 I_{\max} , 每次迭代中凸优化问题的求解复杂度为 $O(d_x^3)$ (基于内点法), 故可得 SCA 算法复杂度为 $O(I_{\max}d_x^3)$ 。综上, 考虑交替迭代最大次数为 T_{total} , 则总复杂度为 $O(T_{\text{total}}(TKd_sN_h^2 + I_{\max}d_x^3))$, 属于多项式复杂度。

3 仿真分析

实验基于 Intel (R) Core (TM) i5-13500HX CPU 硬件平台构建实验环境, 深度学习架构采用 tensorflow 2.30.0, 凸近似优化问题通过 CVX 2.2 工具箱求解。仿真实验目标区域设置为 600 m×600 m, 场景中 UAV 数量为 4, BS 数量为 2, UT 数量范围设置为 [10, 40] 随机分布在目标区域内。任务数据量 D_m 范围设置为 [1, 5] MHz, CPU 周期数在 [100, 200] cycle/bit 范围内随机选择。预分配给 BS k 的带宽 B_k 设置为 0.5 MHz, UAV n 传输功率 P_n^{up} 为 0.1 W, 发射功率为 5 W。无人机最大计算能力配置为 3 GHz, 基站计算资源在 [6, 9] GHz 内。上行带宽统一为 10 MHz。Meta-MARL 算法的网络配置为两个包含 512 个神经元的隐藏层网络, 使用学习率为 10^{-3} 的 ADAM 优化器对演员网络和评论员网络进行训练, 实验对比了不同学习率对训练结果的影响。元训练阶段的任务分布参照表 1 设置。

图 2 为确定性任务场景中不同算法与理论最优解的对比, 其中, 理论最优曲线通过遍历系统的决策空间获取全局最优解计算得出, 作为评估算法的收敛状态的依据。图 3 和图 4 展示了算法在任务切换下的收敛状态, 其中决策任务在迭代 300 次、700 次时发生切换。表 2 定量分析了不同学习率下, 任务切换时本文算法与 MADDPG 算法的系统函数波动幅度以及恢复速度。在执行时间方面, 算法执行 300 次回合训练耗时约 5 min, 每回合完成 200 步迭代训练, 在线决策单步优化耗时为毫秒级。元学习器同步进行离线小批量训练, 单步离线过程为毫秒级。从图 3 和图 4 可以看出, 相比传统的深度 Q 网络 (deep Q-network, DQN) 算法, MADDPG 算法和本文算法均能实现优化问题的收敛, 这体现了 MADDPG 算法对分布式决策优化问题的适应性。结合表 2 中目标切换点的奖励值波动幅度、后续算法恢复速度的量化结果可知, 当发生任务切换时, 不同算法系统函数波动幅度相对一致, 本文算法仅需迭代数十次回合便能实现自适应收敛, MADDPG 算法则需要较长时间来适应新的目标分布, 表明 MADDPG 算法对任务切换意图的识别效率低。本文算法在任务发生切换时, 能够快速达到收敛状态, 表明其通过元学习策略实现了对不同决策任务的快速泛化, 适应了指标意图的动态变化。

值得注意的是, 尽管上述结果验证了本文算法在适应多任务意图、跨任务泛化方面的优势, 但在系统初始阶段仍需构建一定的任务集样本来支撑元知识库的构建。特别是在任务切换频繁场景下, 元学习器面对稀疏任务集样本, 无法实现快速收敛, 造成额外计算开销。针对该问题, 在未来工作中可通过引入图网络、神经网络剪枝等方法对算法进行轻量化设计。

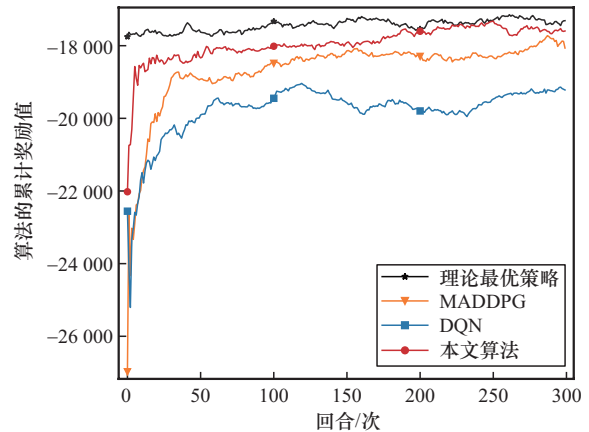


图 2 确定性任务场景中不同算法与理论最优解的对比

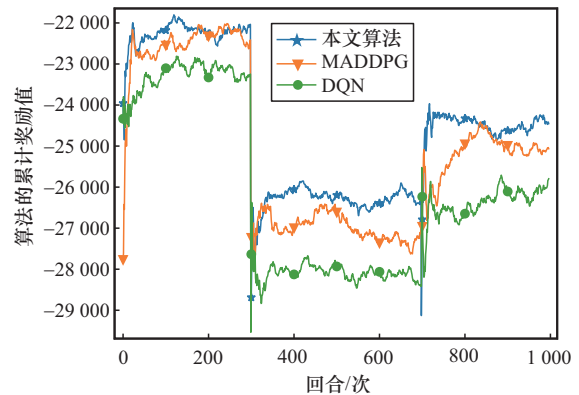


图 3 学习率为 8×10^{-4} 时任务切换收敛性能

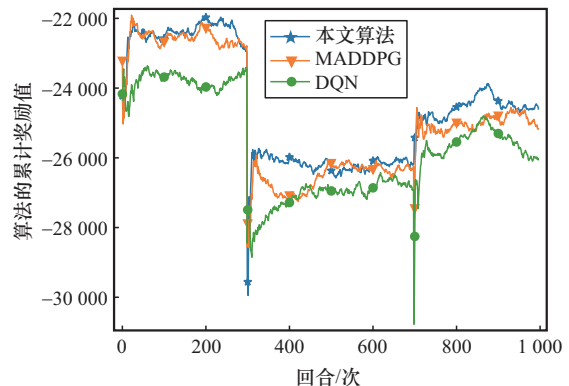


图 4 学习率为 5×10^{-3} 时任务切换收敛性能

表2 任务切换时隙算法波动与恢复状态

学习率	切换点	本文算法波动	MADDPG算法波动	本文算法恢复	MADDPG算法恢复
8×10^{-4}	300	4 909.35	4 609.45	31 回合	150 回合
8×10^{-4}	700	1 157.56	1 162.79	26 回合	150~200 回合
5×10^{-3}	300	4 496.04	4 332.80	50 回合	未收敛
5×10^{-3}	700	1 755.65	1 292.15	34 回合	170 回合

在 UAV 服务性能方面，实验通过评估 UAV 在 t 时隙的服务能耗、服务时延以及负载均衡指数，来评判本文方案的有效性，并选取以下 3 种方案进行对比：1) 卸载固定 (task offloading fixed, TOF) 策略，为评估本文方案所构造的基准策略，该策略基于 MADDPG 算法在不考虑指标意图和负载状态下实现确定性比例的任务卸载与轨迹规划，用于验证动态多指标优化的必要性；2) 多智能体轨迹控制 (multi-agent based trajectory control, MAT) 策略^[20]，该策略基于 MADDPG 算法，基于用户个数建立负载公平指数，同时对负载状态和能耗的静态指标进行联合优化，用于评估本文方案动态多指标以及负载均衡方案的有效性；3) 轨迹联合优化 (joint optimization of trajectory, JOT) 策略^[21]，该策略基于协作式多智能体双延迟深度确定性策略梯度 (DDPG) 算法，通过优化无人机轨迹、任务卸载和通信过程，在不考虑 UAV 负载状态条件下，最小化能耗与时延的加权总成本，用于验证本文方案中负载均衡特性和动态指标意图的适应性。上述方案聚焦于 UAV 任务卸载与轨迹优化的核心问题，与本文场景高度一致。所述算法覆盖了不同算法框架，能够有效验证本文方案对多指标意图变化的适应性。

图 5~图 7 分别给出了 UT 数量从 10 递增至 40 时，UAV 每时隙的服务能耗、服务时延和负载均衡指数的变化趋势。从图 5~图 7 可以看出，UAV 服务能耗和服务时延均随着 UT 数量递增呈上升趋势。UT 的数量增加直接导致计算任务规模扩大，进而使 UAV 的服务能耗和服务时延上升。4 种方案中，本文方案始终保持最低的时延和能耗性能。同时，在负载均衡指数方面，本文方案始终使各 UAV 的负载维持在较高的均衡状态，进一步表明了本文方案通过控制负载均衡实现时延和能耗指标联合优化的有效性。

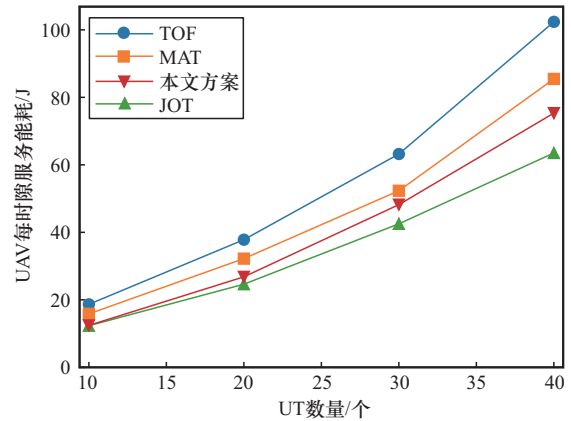


图5 UAV 服务能耗随 UT 数量变化趋势

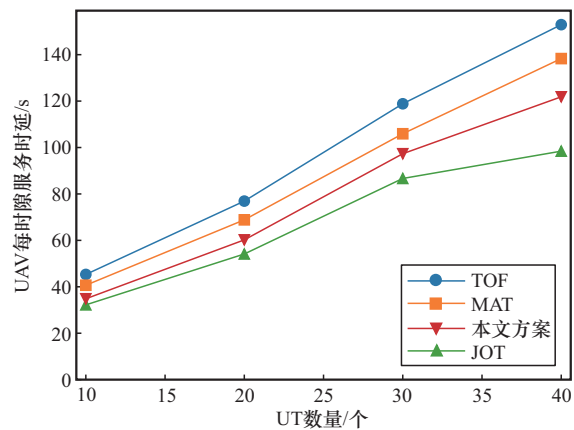


图6 系统服务时延随 UT 数量变化趋势

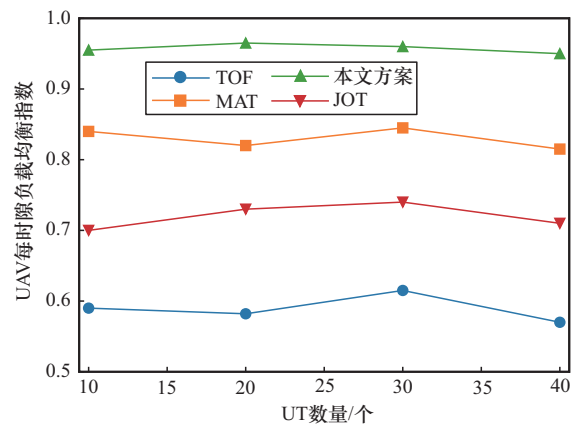


图7 UAV 负载均衡指数随 UT 数量变化趋势

图 8~图 10 展示了系统每个时隙的服务能耗、服务时延和负载均衡指数随不同任务场景的变化趋势。在不同权重配置的任务场景中 (场景 1: $\alpha = 0.9, \beta = 0.1$, 场景 2: $\alpha = 0.7, \beta = 0.3$, 场景 3: $\alpha = 0.5, \beta = 0.5$, 场景 4: $\alpha = 0.1, \beta = 0.9$), 采用负载均衡机制的本文方案和 MAT 方案负载均衡指数均优于无负载均衡的 TOF 方案和 JOT 方案。随着场景的改变,

负载指数呈对应趋势:当场景从场景3切换至场景1和场景4时,本文方案始终保持较高水平。MAT方案UAV负载仅考虑服务用户设备的数量,未引入量化的公平性指数,其UAV均衡指数存在波动且低于本文方案。这一结果验证了在服务能耗、服务时延、负载平衡三大优化目标上,既能达到传统联合优化算法性能,又能基于指标意图状态实现快速适应。

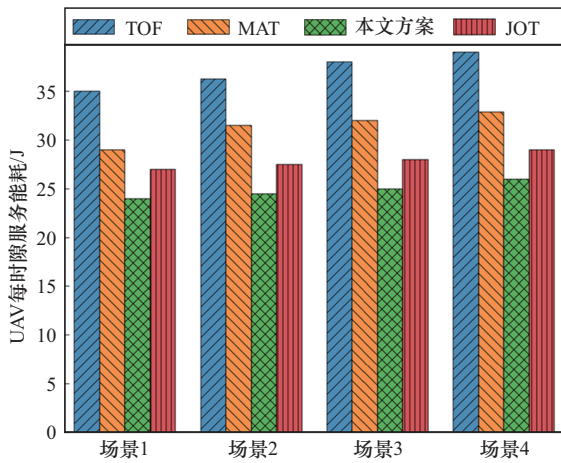


图8 不同场景下的UAV服务能耗

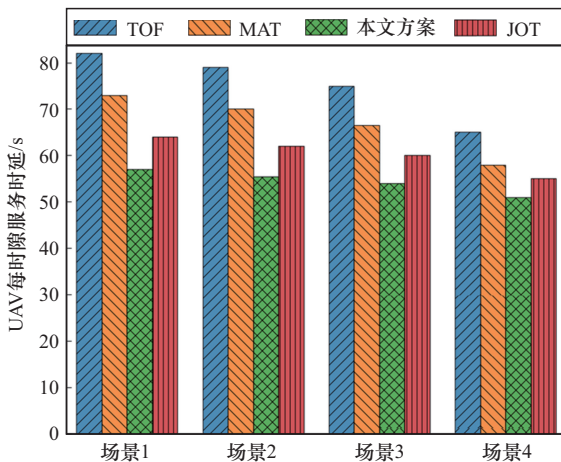


图9 不同场景下的系统UAV服务时延

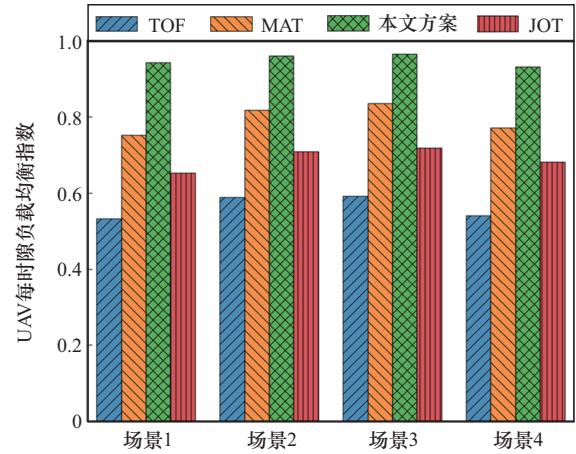


图10 不同场景下的UAV负载均衡指数

图11和图12展示了不同UT分布下的UAV移动轨迹,表3列出了均匀分布场景中UAV轨迹各时间点的负载均衡指数与相关优化目标的量化结果。结合图11和图12与表3可以看出,UAV1与UAV2的轨迹均围绕UE集群与基站位置动态调整,不同时段的数据精准反映了UAV覆盖范围,以及服务能耗、服务时延与负载均衡指数3个目标间的协同与权衡。UAV1轨迹更贴近右侧UE集群,服务能耗与服务时延随覆盖任务量先升后降,负载均衡指数稳定在0.76~0.86,体现其在保障覆盖的同时,通过路径规划实现负载均衡条件下的服务能耗与服务时延协同优化。UAV2轨迹向左侧UE集群延伸,服务能耗与服务时延呈先升后稳再回落趋势,负载均衡指数在0.69~0.88波动,虽然出现局部负载均衡指标降低,但整体仍通过轨迹调整适配覆盖需求。二者轨迹分布既实现了对左右UE集群的有效覆盖,又通过动态路径优化,在服务能耗、服务时延与服务负载均衡指数之间实现有效平衡。

表3 UAV多指标性能的时间分布

序号	时间	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
UAV 1	服务能耗	0	32.15	38.76	45.23	51.89	58.42	62.17	53.81	41.59
	服务时延	0	68.42	75.19	82.37	89.64	95.28	98.73	87.52	72.65
	负载均衡指数	0.50	0.78	0.82	0.79	0.85	0.81	0.79	0.76	0.86
UAV 2	服务能耗	0	28.63	43.17	50.48	57.36	60.25	54.71	48.29	53.89
	服务时延	0	55.78	71.83	80.26	87.91	92.17	85.34	77.68	86.42
	负载均衡指数	0.50	0.75	0.82	0.88	0.84	0.83	0.81	0.78	0.69

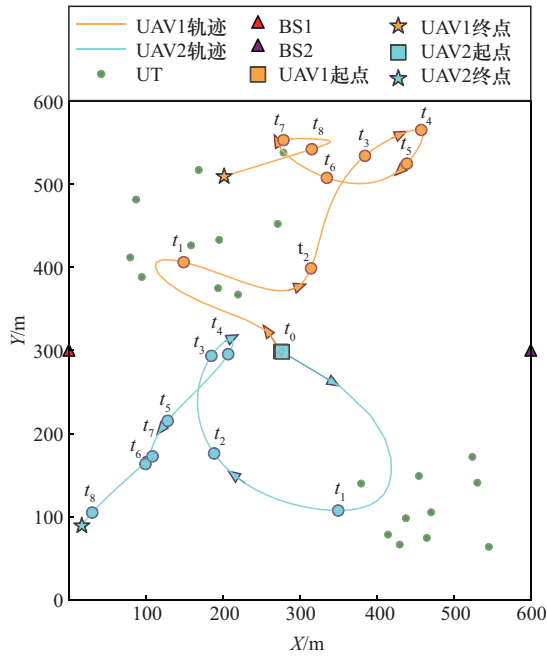


图 11 非均匀 UT 分布下的 UAV 轨迹

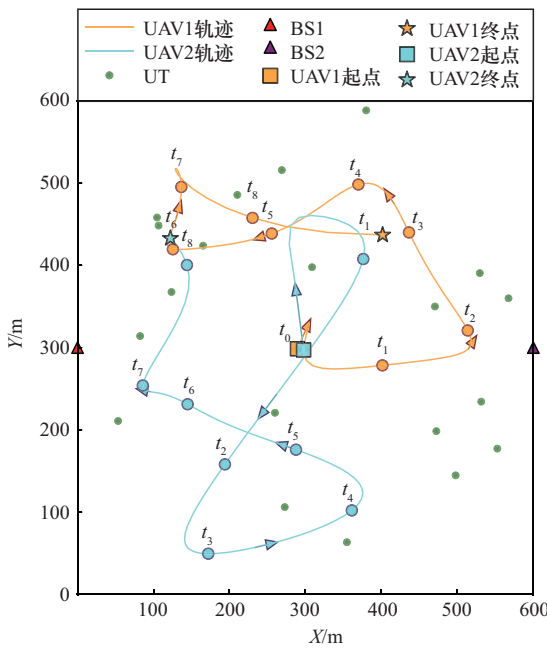


图 12 均匀 UT 分布下的 UAV 轨迹

4 结束语

本文研究了 UAV 辅助的边缘计算任务卸载与轨迹优化问题，提出了一种多指标意图驱动 UAV 计算卸载与轨迹联合优化策略，该策略考虑了 UAV 的服务时延、服务能耗和负载均衡状态多个性能指标，建立了以动态多指标意图为目标的联合优化问题，以优化系统控制决策。在解决该问题

的过程中，将原问题分解为 UAV 计算卸载与资源分配和 UAV 轨迹规划两个子问题。考虑到目标参数的动态不确定性，本文设计了元强化学习和连续凸逼近融合算法实现对两个子问题的交替优化。通过仿真实验，验证了本文算法的收敛性以及系统多个指标意图变化的快速适应能力。在多个指标性能方面，所提方案能够在平衡 UAV 负载状态下显著降低服务能耗和服务时延。

附录 1 算法收敛性证明

对于 Meta-MARL 算法，其基础学习器采用 MADDPG 算法。在平稳状态环境下，文献[22]已证实该类算法的收敛特性。在非平稳状态环境下，由于子问题 P_2 属性可知，Meta-MARL 算法训练过程中每回合经历 200 次迭代时 SCA 算法输出一组轨迹 $\omega_n(t)$ 。因此，环境状态变化速率满足： $\frac{|\omega_n(t+1) - \omega_n(t)|}{\Delta t} \ll \frac{|\theta^\mu(t+1) - \theta^\mu(t)|}{\Delta t}$ 且 $\frac{|\omega_n(t+1) - \omega_n(t)|}{\Delta t} \ll \frac{|\theta^Q(t+1) - \theta^Q(t)|}{\Delta t}$ ，即 SCA 的输出 $\omega_n(t)$ 对于 Meta-MARL 为慢变动态。定义环境状态扰动 $\Delta s_t = s(t+1) - s(t)$ （由 $\omega_n(t)$ 变化导致），则累积奖励的扰动为 $\Delta J_t = J(a(t+1), s(t+1)) - J(a(t), s(t))$ 。根据子问题 P_2 属性可知， Δs_t 有界，故 ΔJ_t 有界。结合平稳状态 MADDPG 算法的策略收敛性，策略更新的增量可抵消环境扰动 Δs_t ，即 $\lim_{t \rightarrow \infty} |J(a(t+1), s(t+1)) - J(a(t), s(t))| = 0$ 。文献[23]已证实元学习算法的收敛性。因此，Meta-MARL 算法在慢变动态非平稳环境中，可收敛至局部最优。对于 SCA 算法，在给定 Meta-MARL 算法输出 $\{z_{mn}(t), \gamma_{mk}^n(t)\}$ 条件下，利用 CVX 求解凸近似子问题 P_1 为平稳环境，文献[24]提出的内凸逼近算法 (NOVA) 理论框架已对其收敛性进行了证明。由此可得，Meta-MARL 算法和 SCA 算法在任一轮交替迭代中满足。

$$\begin{aligned}
 J(t+1) &= J(z_{mn}(t+1), \gamma_{mk}^n(t+1), \omega_n(t+1)) \leq \\
 &J(z_{mn}(t+1), \gamma_{mk}^n(t+1), \omega_n(t)) \leq \\
 &J(z_{mn}(t), \gamma_{mk}^n(t), \omega_n(t))
 \end{aligned} \tag{31}$$

即，原问题 P_1 目标函数序列 $\{J(t)\}$ 在交替迭代过程中单调非增。同时考虑物理资源约束，目标函数序列 $\{J(t)\}$ 有下界。在满足迭代过程无循环振荡条件下，根据波尔查诺-魏尔斯特拉斯定理，存在子序列收敛至极限点 $\{z_{m,n}^*, \gamma_{m,k}^{n,*}, \omega_n^*\}$ ，该极限点为原问题 P_1 的局部平稳点。结合终止准则： $0 \leq \frac{|J(t) - J(t-1)|}{J(t-1)} < \varepsilon$ ，当全局目标函数的变化

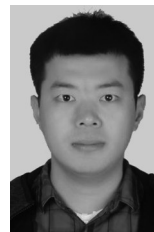
量小于 ε 时，考虑算法的在线决策效率，可认为迭代已接近极限值，继续迭代对全局目标的优化增益可忽略。

综上，Meta-MARL-SCA 算法在交替迭代过程中使原问题 P_1 的全局目标函数单调非增且有下界，交替过程无振荡、无循环，可收敛至原问题的局部平稳点。同时，结合实验（图 2）观察到本文算法与理论最优解相比，使累积奖励值随迭代回合快速下降并趋于稳定，呈现出良好的稳定收敛趋势。

参考文献:

- [1] Yuan L, He Q, Tan S Y, et al. CoopEdge: enabling decentralized, secure and cooperative multi-access edge computing based on blockchain[J]. IEEE Transactions on Parallel and Distributed Systems, 2023, 34(3): 894-908.
- [2] Wang L, Wang K Z, Pan C H, et al. Deep reinforcement learning based dynamic trajectory control for UAV-assisted mobile edge computing[J]. IEEE Transactions on Mobile Computing, 2022, 21(10): 3536-3550.
- [3] Liang X, Liu J, Xie L F. Meta-reinforcement learning for UAV-assisted mobile edge computing of virtual reality services[C]//Proceedings of the 2023 International Conference on Wireless Communications and Signal Processing (WCSP). Piscataway: IEEE Press, 2023: 165-170.
- [4] Diamanti M, Pelekis C, Tsiropoulou E E, et al. Delay minimization for rate-splitting multiple access-based multi-server MEC offloading[J]. IEEE/ACM Transactions on Networking, 2024, 32(2): 1035-1047.
- [5] Li Y D, Wang F, Zhang X, et al. IRS-based MEC for delay-constrained QoS over RF-powered 6G mobile wireless networks[J]. IEEE Transactions on Vehicular Technology, 2023, 72(7): 8722-8737.
- [6] Chen Y, Zhang S, Jin Y B, et al. LOCUS: user-perceived delay-aware service placement and user allocation in MEC environment[J]. IEEE Transactions on Parallel and Distributed Systems, 2022, 33(7): 1581-1592.
- [7] Zhang J S, Luo H B, Chen X, et al. Minimizing response delay in UAV-assisted mobile edge computing by joint UAV deployment and computation offloading[J]. IEEE Transactions on Cloud Computing, 2024, 12(4): 1372-1386.
- [8] Tang Q, Li B, Yang H H, et al. Delay and load fairness optimization with queuing model in multi-AAV assisted MEC: a deep reinforcement learning approach[J]. IEEE Transactions on Network and Service Management, 2025, 22(2): 1247-1258.
- [9] Lin N, Tang H L, Zhao L, et al. A PDDQNLP algorithm for energy efficient computation offloading in UAV-assisted MEC[J]. IEEE Transactions on Wireless Communications, 2023, 22(12): 8876-8890.
- [10] Li J, Yi C Y, Chen J Y, et al. A reinforcement learning-based stochastic game for energy-efficient UAV swarm-assisted MEC with dynamic clustering and scheduling[J]. IEEE Transactions on Green Communications and Networking, 2025, 9(1): 255-270.
- [11] Song F H, Deng M S, Xing H L, et al. Energy-efficient trajectory optimization with wireless charging in UAV-assisted MEC based on multi-objective reinforcement learning[J]. IEEE Transactions on Mobile Computing, 2024, 23(12): 10867-10884.
- [12] Wu Q J, Cui M, Zhang G C, et al. Energy-efficient UAV-enabled MEC systems: NOMA, FDMA, or TDMA offloading?[J]. IEEE Transactions on Vehicular Technology, 2025, PP(99): 1-16.
- [13] Wang Y Q, Zhu J, Huang H P, et al. Bi-objective ant colony optimization for trajectory planning and task offloading in UAV-assisted MEC systems[J]. IEEE Transactions on Mobile Computing, 2024, 23(12): 12360-12377.
- [14] Hu H, Chen Z, Zhou F H, et al. Joint resource and trajectory optimization for heterogeneous-UAVs enabled aerial-ground cooperative computing networks[J]. IEEE Transactions on Vehicular Technology, 2023, 72(7): 8812-8826.
- [15] Kuang Z F, Pan Y H, Yang F, et al. Joint task offloading scheduling and resource allocation in air-ground cooperation UAV-enabled mobile edge computing[J]. IEEE Transactions on Vehicular Technology, 2024, 73(4): 5796-5807.
- [16] Liu Q, Shi L, Sun L L, et al. Path planning for UAV-mounted mobile edge computing with deep reinforcement learning[J]. IEEE Transactions on Vehicular Technology, 2020, 69(5): 5723-5728.
- [17] Wang Z Q, Rong H G, Jiang H B, et al. A load-balanced and energy-efficient navigation scheme for UAV-mounted mobile edge computing[J]. IEEE Transactions on Network Science and Engineering, 2022, 9(5): 3659-3674.
- [18] Du P F, Xiao T Y, Cao H T, et al. MADDPG-based resource allocation for integrated communication, sensing and computing in UAV-assisted IoV networks[J]. IEEE Transactions on Vehicular Technology, 2025. DOI:10.1109/TVT.2025.3633518.
- [19] 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.
- [20] Wang L, Wang K Z, Pan C H, et al. Multi-agent deep reinforcement learning-based trajectory planning for multi-UAV assisted mobile edge computing[J]. IEEE Transactions on Cognitive Communications and Networking, 2021, 7(1): 73-84.
- [21] Zhao N, Ye Z Y, Pei Y Y, et al. Multi-agent deep reinforcement learning for task offloading in UAV-assisted mobile edge computing[J]. IEEE Transactions on Wireless Communications, 2022, 21(9): 6949-6960.
- [22] Zhang K Q, Yang Z R, Liu H, et al. Fully decentralized multi-agent reinforcement learning with networked agents[PP]. V2.arXiv (2018-02-27)[2026-01-05]. arXiv: arXiv. 1802.08757.
- [23] Fallah A, Mokhtari A, Ozdaglar A. On the convergence theory of gradient-based model-agnostic meta-learning algorithms[PP]. V4. arXiv (2020-05-16)[2026-01-05]. arXiv: arXiv. 1908.10400.
- [24] Scutari G, Facchinei F, Lampariello L. Parallel and distributed methods for constrained nonconvex optimization: part I: theory[J]. IEEE Transactions on Signal Processing, 2017, 65(8): 1929-1944.

[作者简介]



林鹏 (1991-), 男, 山东临沂人, 博士, 南京信息工程大学副教授, 主要研究方向为无线内生智能网络、算力网络和无线资源管理等。



黄新梁 (2000-), 男, 江苏南京人, 南京信息工程大学硕士生, 主要研究方向为强化学习、边缘计算。



宁兆龙 (1986-), 男, 辽宁沈阳人, 博士, 重庆邮电大学教授, 主要研究方向为边缘智能、应急通信网络和网络资源优化。



郭磊 (1980-), 男, 四川眉山人, 博士, 东北大学教授、博士生导师, 主要研究方向为无线通信网络和光通信网络。



刘艳 (1991-), 女, 山东临沂人, 博士, 南京信息工程大学副教授, 主要研究方向为信息物理系统、分布式控制与优化和智能决策。



张治中 (1972-), 男, 湖北恩施人, 博士, 南京信息工程大学教授、博士生导师, 主要研究方向为通信网测试和宽带信息网络。