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

张广驰 何梓楠 崔 苗*
(广东工业大学信息工程学院 广州 510006)

摘 要：近年来，部署搭载有移动边缘计算(MEC)服务器的无人机(UAVs)为地面用户提供计算资源已成为一种新兴的技术。针对无人机辅助多用户移动边缘计算系统，该文构建了以最小化用户平均能耗为目标的模型，联合优化无人机的飞行轨迹和用户计算策略的调度。通过深度强化学习(DRL)求解能耗优化问题，提出基于柔性参与者-评论者(SAC)的优化算法。该算法应用最大熵的思想来探索最优策略并使用高效迭代更新获得最优策略，通过保留所有高回报值的策略，增强算法的探索能力，提高训练过程的收敛速度。仿真结果表明与已有算法相比，所提算法能有效降低用户的平均能耗，并具有很好的稳定性和收敛性。

关键词：无人机通信；深度强化学习；移动边缘计算；轨迹规划；调度策略
中图分类号：TN925 **文献标识码**：A **文章编号**：1009-5896(2023)05-1635-09
DOI: 10.11999/JEIT220352

Energy Consumption Optimization of Unmanned Aerial Vehicle Assisted Mobile Edge Computing Systems Based on Deep Reinforcement Learning

ZHANG Guangchi HE Zinan CUI Miao
(*School of Information Engineering*, *Guangdong University of Technology*, *Guangzhou* 510006, *China*)

Abstract: In recent years, the deployment of Unmanned Aerial Vehicles (UAVs) equipped with Mobile Edge Computing (MEC) servers to provide computing services for ground users has become an emerging method. Considering an UAV-assisted MEC system with multi-users, a scheme is investigated to minimize the average energy consumption for all users to complete their computation tasks via optimizing the trajectory of UAV and computation strategies of the users during the UAV's whole flight duration. A Deep Reinforcement Learning (DRL)-based Soft Actor-Critic (SAC) algorithm is proposed to tackle the energy consumption optimization problem. With the iteration of the network training procedure, the best action is obtained according to the maximum entropy rule, which does not neglect any action with high reward value and thus can enhance the exploration and convergence performance of the proposed algorithm. Simulation results reveal that the proposed SAC algorithm can effectively decrease the average energy consumption of all users and achieves better stability and convergence performance, as compared to some existing baseline algorithms.

Key words: Unmanned Aerial Vehicles (UAV) communication; Deep Reinforcement Learning (DRL); Mobile Edge Computing (MEC); Trajectory optimization; Scheduling policy

收稿日期：2022-03-31；改回日期：2022-07-05；网络出版：2022-07-06
*通信作者： 崔苗 cuimiao@gdut.edu.cn
基金项目：广东省科技计划(2021A0505030015, 2020A050515010)，广东特支计划(2019TQ05X409)，智慧城市物联网国家重点实验室(澳门大学)开放课题(SKL-IoTSC(UM)-2021-2023/ORPF/A04/2022)
Foundation Items: The Science and Technology Plan Project of Guangdong Province (2021A0505030015, 2020A050515010), The Special Support Plan for High-Level Talents of Guangdong Province (2019TQ05X409), The Open Research Project Programme of the State Key Laboratory of Internet of Things for Smart City (University of Macau) (SKL-IoTSC(UM)-2021-2023/ORPF/A04/2022)

1 引言

物联网技术的飞速发展，推动计算密集型智能设备的普及，使人们的生活更加便捷。虽然现阶段的智能电子设备配备了强大的处理器，但是由于规模小和资源有限，如电池容量，在确保满足移动应用程序的计算能力和低时延要求的情况下，需要大量的能量开销，无法提供令人满意的服务质量。移动边缘计算[1](Mobile Edge Computing, MEC)在近些年被提出克服这一缺点，将用户设备的计算任务转移到网络边缘进行计算。移动边缘计算是一种新兴技术，在移动网络边缘的小型云计算平台上部署MEC服务器来支持任务密集型的应用程序，并已被证明可以极大提高用户设备执行计算任务的能力。文献[2]将移动边缘计算系统应用于能量采集设备，提出一种动态计算的卸载策略，能够有效地降低任务时延和解决任务失败的问题。

无人机搭载MEC服务器在工业界和学术界也被广泛讨论[3]，利用无人机的覆盖能力和机动性，实现更低时延的要求和提供更加灵活的计算服务。对于MEC和无人机的结合使用，已经有学者对此做了相关的研究。文献[4]提出了一种量化的动态规划算法来解决MEC的资源分配问题。文献[5]将无人机轨迹离散化为无人机位置序列，将连续空间转化成离散的有限空间，使得问题可处理化。文献[6]通过离散变量近似无人机轨迹，通过传统的凸优化方法进行优化。在上述工作中，通常将无人机飞行轨迹和用户调度联合优化问题建模为非凸优化问题，然后将问题分解为多个优化子问题，并采用凸优化方法逐一求解。然而，凸优化方法在求解更复杂环境下(如更高维的约束条件、更大量相互耦合的优化变量等)的无人机轨迹和调度优化问题时，面临复杂度过高的难题。

由于控制决策和资源优化的优势，深度强化学习(Deep Reinforcement Learning, DRL)被广泛认为可应用于复杂的动态环境，是解决策略控制问题的理想工具[7]。文献[8]基于双深度Q网络(Double Deep Q Network, DDQN)算法优化无人机的飞行轨迹，以最大化设备卸载数据量，同时最小化无人机的能量消耗。然而，基于DDQN的算法不适用于具有高维动作空间的环境，且难以应用于优化连续的变量。文献[9]将深度确定性策略梯度(Deep Deterministic Policy Gradient, DDPG)算法应用于多无人机集群。作者考虑一个中央控制器负责训练观测数据，并将训练后的数据广播到多无人机集群网络。然而，DDPG算法只考虑当前状态下的最优动作，在训练过程中通常手动添加噪声来实现智能体探索动作空间，这种做法在复杂环境下难以在开发和探索中寻求平衡，往往导致算法陷入次优解[10]。

为了克服上述困难，本文提出将柔性参与者-评论者(Soft Actor-Critic, SAC)算法[11]应用于无人机辅助移动边缘计算的系统，无人机在指定区域内从起始位置飞往终点位置，在飞行过程中为覆盖范围内的多个地面用户提供计算服务，即接收地面用户的卸载数据，利用搭载的MEC服务器进行计算并将结果返还给所服务的用户。本文拟解决的问题：(1)如何选择适当的设备关联，即用户设备选择是否卸载计算任务或者本地处理计算任务，在有限的通信资源的条件下，减少用户设备的长期能量消耗。(2)考虑用户设备计算任务不同，如何实时控制无人机的飞行轨迹，为适当的设备提供计算服务，尤其是考虑到无人机需要达到特定终点的环境。本文提出基于SAC算法的用户平均能耗最小化方案，联合优化无人机的飞行轨迹和用户计算策略调度以最小化用户平均能耗。由于DDPG算法是解决连续优化变量问题的主流深度强化学习算法[12]，已被用于解决不同无线通信系统的优化问题，本文将DDPG算法以及另外两种算法作为基准方案，与本文所提算法进行对比，结果显示所提SAC算法能有效降低地面用户的平均能耗，并具有更佳的稳定性和收敛性。

2 系统模型

2.1 场景描述

本文考虑一个具有多用户的无人机辅助移动边缘计算系统，如图1所示。该系统由N个地面用户与1架配备MEC服务器的无人机组成。其中，N个地面用户随机分布在目标区域内，第i ($i \in \{1,2,\cdots,N\}$)个用户的3维坐标为$w_i=[x_i,y_i,0]$。无人机以固定安全高度H飞行在目标区域上空，其3维坐标描述为$u(t)=[X(t),Y(t),H]$。将无人机的飞行时间划分为T个时隙，每个时隙长度为τ。令$d_{\mathrm{U}}(t)$和$\theta_{\mathrm{U}}(t)$分别为无人机在第t时隙的飞行距离和水平方向角度[13]，且满足$0 \le d_{\mathrm{U}}(t) \le d_{\max}$和$0 \le \theta_{\mathrm{U}}(t) \le 2\pi$，其中$d_{\max}$为无人机单位时隙内最大飞行距离。因此

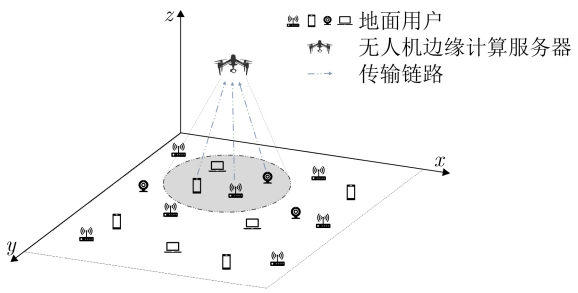

图 1 无人机辅助的多用户移动边缘计算系统

无人机在第t时隙的水平坐标可表示为$X(t)=X(0)+\sum_{l=1}^{t}\mathrm{d_U}(l)\cos(\theta_{\mathrm{U}}(l))$,$Y(t)=Y(0)+\sum_{l=1}^{t}\mathrm{d_U}(l)\sin(\theta_{\mathrm{U}}(l))$，其中$u(0)=[X(0),Y(0),H]$为无人机的起始位置坐标。限定无人机只能在规定的区域飞行，即$0\le X(t)\le X_{\max}$和$0\le Y(t)\le Y_{\max}$，其中$X_{\max}$和$Y_{\max}$为该区域的长度和宽度。

在本文中，地面用户在每个时隙产生计算任务为$I_i(t)=\{D_i(t),F_i(t)\},i\in\{1,2,\cdots,N\}$，其中，$D_i(t)$表示计算任务的数据量大小，$F_i(t)$表示完成该任务所需的CPU周期数。每个用户可以选择将计算任务以卸载方式交由无人机计算或者是本地计算，定义$\alpha_i(t)\in\{0,1\}$为第i个用户设备在时隙t的计算策略，$\alpha_i(t)=1$表示卸载计算，$\alpha_i(t)=0$表示本地计算，用户根据其计算任务的调度策略来选择执行方式。无人机在时隙内接收并计算来自用户卸载的任务，再将计算结果返回给所服务的用户。

2.2 任务决策

(1)卸载计算。在时隙t，地面用户选择将计算任务以卸载方式传输给无人机，即$\alpha_i(t)=1$。无人机为地面用户提供计算服务必须满足覆盖范围的约束，即$\alpha_i(t)d_i(t)\le R_{\max}$，$d_i(t)=\sqrt{(X(t)-x_i)^2+(Y(t)-y_i)^2}$为该用户与无人机的水平距离，$R_{\max}$为无人机的最大覆盖半径，它由无人机的飞行高度$H$和接收天线的最大接收角度$\theta$决定，即$R_{\max}=H\cdot\tan\theta$。在本文中，地面用户均配备单根天线，无人机采用频分多址(Frequency Division Multiple Access, FDMA)协议可以避免设备之间的干扰。设无人机同时接收K个用户的计算任务数据，则将总带宽W平均分为K个子信道，每个通信信道的带宽为$B=W/K$。受限于MEC服务器的计算资源，无人机在每个时隙最多可接收$K_{\max}$个地面用户的卸载计算任务数据，即$\sum_{i=0}^{N}\alpha_i(t)\le K_{\max}$。本文假设在该场景下无人机和用户之间没有遮挡物，且考虑到无人机飞行高度较高，我们将无人机和地面用户之间的信道建模为自由空间信道[14,15]。因此，在时隙t内第i个地面用户向无人机的数据卸载速率为

$$r_i(t)=B\log_2\left(1+\frac{\rho P^{\mathrm{Tr}}}{H^2+d_i^2(t)}\right)\tag{1}$$

其中，P^{Tr}是用户数据卸载的传输功率；$\rho=g_0G_0/n^2$，g_0为参考距离(1 m)的信道功率增益，$G_0=2.284\,6$，n^2为噪声功率[16,17]。在数据卸载过程中，地面用户将计算任务传输给无人机的时间开销为

$$T_i^{\mathrm{Tr}}(t)=\frac{D_i(t)}{r_i(t)}\tag{2}$$

无人机MEC服务器处理计算任务所耗费的时间为

$$T_i^{\mathrm{U}}(t)=\frac{F_i(t)}{f^{\mathrm{U}}(t)}\tag{3}$$

其中，$f^{\mathrm{U}}(t)$是无人机MEC服务器所提供的CPU周期数。用户卸载计算过程所耗费的时间为$T_i^{\mathrm{O}}(t)=T_i^{\mathrm{Tr}}(t)+T_i^{\mathrm{U}}(t)$。用户的发射功率为$P^{\mathrm{Tr}}$，则用户设备卸载计算的能量消耗为

$$E_i^{\mathrm{Tr}}(t)=P^{\mathrm{Tr}}\cdot T_i^{\mathrm{Tr}}(t)\tag{4}$$

需要额外说明的是由于计算结果的数据量非常小，可以忽略无人机向用户发送计算结果所耗费的时间和能耗[18]。

(2)本地计算。在时隙t，地面用户选择将计算任务进行本地处理，即$\alpha_i(t)=0$。本地计算过程所耗费的时间为

$$T_i^{\mathrm{L}}(t)=\frac{F_i(t)}{f_i^{\mathrm{L}}(t)}\tag{5}$$

将用户设备功率消耗记为$k_i\left(f_i^{\mathrm{L}}(t)\right)^v$，$\mathrm{k}_i$是常数且仅取决于设备的处理器芯片架构，通常设置为$1.5\times10^{-28}$，$v$通常设置为3，$f_i^{\mathrm{L}}(t)$为用户设备处理器的CPU周期数[19]。因此，地面用户设备在本地计算过程中的能量消耗为

$$E_i^{\mathrm{L}}(t)=\mathrm{k}_i\left(f_i^{\mathrm{L}}(t)\right)^v\cdot T_i^{\mathrm{L}}(t)\tag{6}$$

综上所述，地面用户在每个时隙所消耗的能量为

$$E_i(t)=\begin{cases}E_i^{\mathrm{Tr}}(t),\text{卸载计算}\\E_i^{\mathrm{L}}(t),\text{本地计算}\end{cases}\tag{7}$$

所有用户的平均总能耗为

$$E_{\mathrm{average}}=\frac{1}{T}\sum_{t}^{T}\sum_{i}^{N}\left(\alpha_i(t)\cdot E_i^{\mathrm{Tr}}(t)+(1-\alpha_i(t))\cdot E_i^{\mathrm{L}}(t)\right),\alpha_i(t)\in\{1,0\}\tag{8}$$

为了满足服务质量要求，在任意时刻完成计算任务的时间不能超过最大时延$T_{\max}$，即$\alpha_i(t)\cdot T_i^{\mathrm{O}}(t)+(1-\alpha_i(t))\cdot T_i^{\mathrm{L}}(t)\le T_{\max}$。

3 问题构建

为了最小化地面用户的平均能耗，本文研究联合优化无人机轨迹和用户的计算策略。考虑到可以通过无人机的飞行距离$d_{\mathrm{U}}(t)$和水平方向角度$\theta_{\mathrm{U}}(t)$计算得到无人机在时隙t的坐标，令$U\triangleq\{d_{\mathrm{U}}(t),\theta_{\mathrm{U}}(t),\forall t\in T\}$表示无人机的飞行轨迹，$A\triangleq\{\alpha_i(t),\forall i\in N,\forall t\in T\}$表示地面用户的计算策略，优化问题可以表述为

$$\min_{U,A}E_{\mathrm{average}}\tag{9}$$

$$\text{s.t.}\ \ \alpha_i(t)=\{0,1\},\forall i\in N,\forall t\in T \tag{10}$$

$$\sum_{i=0}^{N}\alpha_i(t)\le K_{\max},\forall t\in T \tag{11}$$

$$0\le X(t)\le X_{\max},\forall t\in T \tag{12}$$

$$0\le Y(t)\le Y_{\max},\forall t\in T \tag{13}$$

$$0\le d_{\mathrm{U}}(t)\le d_{\max},\forall t\in T \tag{14}$$

$$0\le \theta_{\mathrm{U}}(t)\le 2\pi,\forall t\in T \tag{15}$$

$$\alpha_i(t)\cdot d_i(t)\le R_{\max},\forall i\in N,\forall t\in T \tag{16}$$

$$\alpha_i(t)\cdot T_i^{\mathrm{O}}(t)+(1-\alpha_i(t))T_i^{\mathrm{L}}(t)\le T_{\max},\forall i\in N,\ \forall t\in T \tag{17}$$

其中，式(10)为用户计算策略的约束，式(11)为无人机在每个时隙最大覆盖用户数量的约束，式(12)、式(13)为无人机水平坐标的约束，式(14)、式(15)为无人机飞行距离和水平方向角度的约束，式(16)为无人机覆盖范围约束，式(17)为用户完成计算任务的时延约束。

问题式(9)—式(17)是一个非凸优化问题，采用传统的优化算法难以求解。即使简化为多个子问题近似求解也需要复杂的数学分析和公式推导，且并不能保证求得最优解。深度强化学习作为机器学习的重要分支，可以在不使用复杂的数值优化算法的情况下解决决策优化问题，具有强大的数据处理能力。

4 基于SAC的用户平均能耗最小化算法

SAC算法是一种新兴的深度强化学习算法[10]，是基于最大熵的随机策略算法，具有很强的探索性和鲁棒性，可以应用于复杂的动态环境，从动作空间的所有可能策略中探索最优策略。本文提出基于SAC算法设计无人机的飞行轨迹和地面用户的计算决策调度，求解上述优化问题。

4.1 深度强化学习算法概述

强化学习是由智能体和环境构成的机器学习技术，智能体与环境进行交互，通过获得累计回报来计算用以评判智能体动作好坏的Q值，调整其动作使其累积奖励的期望最大化。智能体从所在环境观察到某种特征表达，即状态$s(t)$，并在此基础上选择一个动作，即动作$a(t)$，同时获得一个数值化的奖励，即回报$r(t)$。

4.2 SAC算法框架及神经网络

本文所提的SAC算法的框架如图2所示，它主要由智能体、环境、经验缓冲区、1个表演者(Actor)网络、2个批评者(Critic)网络及其2个Critic网络的目标网络6个模块组成。SAC算法设置了经验缓冲区来解决数据关联性问题，用于深度神经网络参数的训练。经验缓冲区作为存放以往的经验样本$(s(t),a(t),r(t),s(t+1))$，用以网络训练时批次采样。

Actor网络，由该网络根据智能体的状态信息输出动作。具体而言，将状态$s(t)$输入Actor网络，输出均值μ和方差σ，策略$\pi_\varphi(\cdot|s(t))$是以均值为μ和方差为σ的正态分布，智能体的动作从该正态分布采样得到，即$a(t)\sim\pi_\varphi(\cdot|s(t))$。其中，$\phi$表示Actor网络的参数。SAC算法还包括两个参数化的Critic网络，由该网络根据智能体的状态信息和动作输出Q值，作为评判Actor网络表现的好坏。从经验缓存区中采样小批次经验样本$(s(t),a(t),r(t),s(t+1))$，将$s(t)$, $a(t)$作为Critic网络的输入，分别

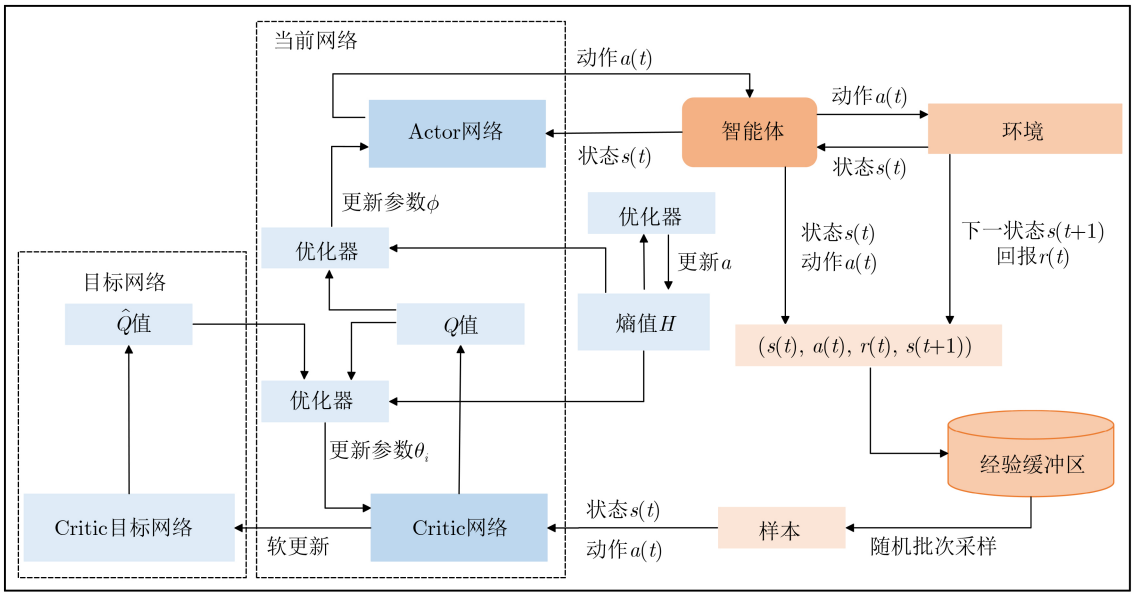

图 2 SAC算法框架图

输出$Q_{\theta_1}(s(t),a(t))$和$Q_{\theta_2}(s(t),a(t))$。其中，θ_i表示 Critic网络的参数。此外，SAC算法还构造了结构相同而参数不同的Critic目标网络，输出目标$\hat{Q}$值$Q_{\hat{\theta}_i}(s(t),a(t)),i\in\{0,1\}$，$\hat{\theta}_i$表示Critic目标网络的参数。目标$\hat{Q}$值的作用是构造目标值$\hat{y}$，以训练神经网络，提高训练的稳定性和收敛性。构造两个Critic网络和两个Critic目标网络，可以有效防止对Q值和$\hat{Q}$值过高估计，我们在计算目标函数时会选取较小的Q值和$\hat{Q}$值，减轻策略改进过程中的积极偏差，加快训练速度。

Actor网络、Critic网络及目标网络的神经网络如图3所示，Actor网络包括1个输入层、两个隐藏层和两个输出层。输入层的神经元个数与状态信息维度相同。两个输出层的神经元个数均与动作维数相同。输入层和隐藏层使用ReLU函数作为激活函数，输出层使用tanh函数作为激活函数，将均值μ和方差σ限制在[–1,1]范围内。Critic网络及目标网络包括1个输入层、两个隐藏层、两个层归一化层和1个输出层。输入层的神经元个数为状态信息和动作的维数之和，输出层仅由1个神经元构成。输入层和隐藏层同样使用ReLU函数作为激活函数。

4.3 SAC算法的深度神经网络训练

SAC算法的最主要特征是熵正则化(entropy regularization)。熵是策略随机性的一种衡量，提高熵值可以带来更多的策略探索。通过权衡回报和熵值来训练策略π，可以提高智能体的学习速度，同时避免策略π收敛至局部最优解。该算法的目标是最大化回报与熵值之和的期望值，最优策略可以表示为

$$\pi*=\underset{\pi}{\operatorname{argmax}}\mathrm{E}\left[\sum_{t=0}^{\infty}\gamma^t r(s(t),a(t))+\alpha H(\pi_\varphi(\cdot|s(t+1)))\right] \tag{18}$$

其中，γ为折扣因子，熵H的计算方式为$H(\pi_\varphi(\cdot|s(t+1)))=E_{s(t+1)}[-\lg\pi_\varphi(\cdot|s(t+1))]$，$\alpha$为熵正则化系数，控制熵值相对于累计回报的重要程度，其目的是随机化该策略，使每个动作输出的可能性都尽可能分散。

根据动作价值函数的贝尔曼方程[7]可得前后时刻Q值的关系

$$Q(s(t),a(t))=r(t)+\gamma E_{(s(t+1),a(t+1))}\cdot[Q(s(t+1),a(t+1))-\lg\pi_\varphi(\cdot|s(t+1))] \tag{19}$$

定义目标值为$\hat{y}(r(t),s(t+1))$，即

$$\hat{y}(r(t),s(t+1))=r(t)+\gamma\left[\min_{i=1,2}Q_{\hat{\theta}_i}(s(t+1),\tilde{a}(t+1))-\lg\pi_\varphi(\cdot|s(t+1))\right] \tag{20}$$

此外，用于计算目标值的动作$\tilde{a}(t+1)$是将$s(t+1)$输入Actor网络，根据当前策略获得的，并非从经验缓冲区中采样。

从经验缓冲区中随机采样小批次经验样本进行训练。Critic网络的损失函数$L_{\mathrm{C}_i}(\theta)$为

$$L_{\mathrm{C}_i}(\theta)=\mathrm{E}\left[Q_{\theta_i}(s(t),a(t))-(\hat{y}(r(t),s(t+1)))^2\right] \tag{21}$$

其中，$i\in\{0,1\}$。Actor网络的损失函数$L_{\mathrm{A}}(\phi)$为

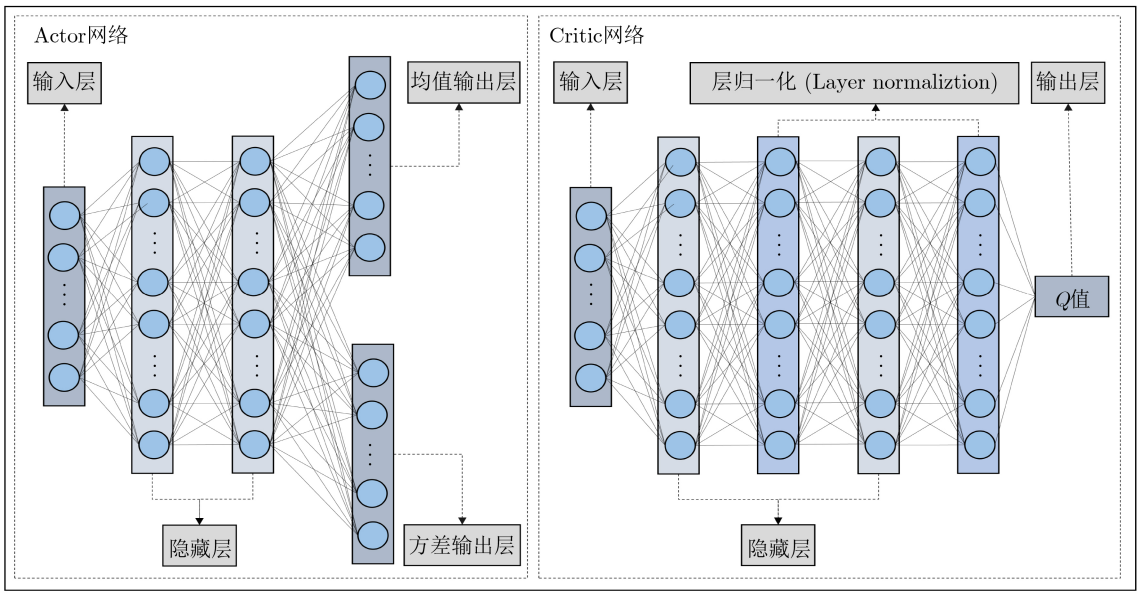

图 3 Actor网络与Critic网络的神经网络图

$$L_{\mathrm{A}}(\phi)=\max_{\phi}\mathrm{E}\left[\min_{i=1,2}Q_{\theta_i}(s(t),\tilde{a}(t))-\alpha\lg\pi_{\phi}(\tilde{a}(t)|s(t))\right] \tag{22}$$

根据损失函数$L_{\mathrm{A}}(\phi)$和$L_{\mathrm{C}_i}(\theta)$，使用梯度下降法反向更新Actor网络和Critic网络参数ϕ,θ_i。Actor网络和Critic网络的参数进行网络优化器更新，而Critic目标网络的参数则进行“软”更新，即通过缓慢跟踪网络参数以更新目标网络的参数，$\hat{\theta}_i\leftarrow\tau\theta_i+(1-\tau)\cdot\hat{\theta}_i$，其中，$\tau$为软更新系数且通常设置为0.002。

在智能体学习的初始阶段，动作空间尚未被充分探索，适当地提高α有利于探索更多策略，随着动作空间的扩大，α也应降低。在算法迭代中，更新网络参数的同时也需要更新熵值H，本文是通过优化熵正则化系数α来更新熵值，损失函数$L(\alpha)$

$$L(\alpha)=-\lg\alpha\left(\lg\pi_{\varphi}(\cdot|s(t+1))+\overline{H}\right) \tag{23}$$

其中，$\bar{H}$为目标熵值，通常设置为–2。本文同样使用梯度下降法更新熵正则化系数α。本文使用Adam作为Actor网络参数ϕ、Critic网络参数θ_i和熵正则化系数α的优化器。

4.4 基于SAC的用户平均能耗最小化算法的算法流程

为了实现用户平均能耗的最小化，基于本文所考虑的模型对强化学习的基本要素进行设置，即依次对状态空间、动作空间和回报进行设置，如下所示：

(1)状态空间$s(t)$：在计算用户能耗的过程中，卸载速率与无人机的位置以及地面用户的位置相关，卸载时间与用户的任务量大小相关。此外，无人机需要满足抵达终点的约束。因此，状态空间的表达式为$s(t)=\{X(t),Y(t),H,D_{\mathrm{U}}(t)\}$。

(2)动作空间$a(t)$：为了实现无人机轨迹规划，Actor网络输出无人机飞行距离$d_{\mathrm{U}}(t)$和水平方向角度$\theta_{\mathrm{U}}(t)$，根据位移公式可以计算无人机下一时刻的坐标。此外，Actor网络还输出N个用户设备的计算策略$\alpha_i(t)\in\{0,1\}$，$i\in\{1,2,\cdots,N\}$。动作空间可表示为$a(t)=\{d_{\mathrm{U}}(t),\theta_{\mathrm{U}}(t),\alpha_i(t)\}$，$i\in\{1,2,\cdots,N\}$。

(3)回报设计$r(t)$：将优化目标作为回报的一部分，即$R_{\mathrm{energy}}=-E_{\mathrm{average}}$。为了满足无人机到达指定终点位置的约束，设置回报$R_{\mathrm{destination}}$使无人机自主学习飞往终点位置的轨迹。此外，当约束式(12)、式(13)不满足时，即无人机飞出规定区域时设置惩罚P_{out}。当约束式(14)不满足时，即无人机水平飞行位移超过最大位移$d_{\max}$时设置惩罚P_{displace}。当约束式(11)不满足时，即无人机的服务用户数量超过最大服务数量$K_{\max}$时设置惩罚P_{α}。因此回报表达式为

$$r(t)=R_{\mathrm{energy}}+R_{\mathrm{destination}}+P_{\mathrm{out}}+P_{\mathrm{displace}}+P_{\alpha} \tag{24}$$

基于以上设置，本文所提出的基于深度强化学习的SAC的最小化用户平均能量损耗算法流程，算法如算法1所示，具体解释如下。步骤(1)，首先对4.2节提到的SAC算法的网络框架进行初始化。步骤(2)—步骤(11)为训练过程。步骤(3)，重新初始化无人机坐标和智能体状态。步骤(5)，智能体从环境中观测到状态$s(t)$，包括无人机的坐标以及与终点的距离，将$s(t)$输入Actor网络，输出动作$a(t)$，即无人机的飞行方向和距离，以及用户的计算策略。步骤(6)—步骤(8)，无人机根据相应的动作更新其坐

算法1 基于SAC的最小化用户设备平均能量损耗算法的算法流程

(1)初始化经验缓冲区，Actor网络，Critic网络及目标网络，初始化无人机起始位置坐标及终点位置坐标，随机生成用户坐标以及计算任务；

(2)循环训练幕数 Episode = 1,2,…,M：

(3) 重新初始化无人机起始坐标以及初始状态$s(0)$；

(4) 循环时间步数 Time = 1,2,…,T：

(5) 由状态$s(t)$根据策略π_{ϕ}选择动作$a(t)$；

(6) 无人机在状态$s(t)$下执行动作$a(t)$，进入下一状态$s(t+1)$且更新无人机坐标$[X(t),Y(t),H]$，并根据式(24)得到回报$r(t)$以及根据式(8)计算所有用户的能耗$\sum_{i=1}^{N}E_i(t)$；

(7) 将$[s(t),a(t),r(t),s(t+1)]$存储在经验缓冲区；

(8) 更新状态$s(t)=s(t+1)$；

(9) 从经验缓冲区中随机采样批次经验样本，根据式(21)、式(22)和式(23)分别计算损失函数$L_{\mathrm{C}_i}(\theta_i)$,$L_{\mathrm{A}}(\phi)$和$L(\alpha)$，并更新Critic网络参数$\theta_i$、Actor网络参数$\phi$和熵正则化系数$\alpha$；更新Critic目标函数参数，$\hat{\theta}_i\leftarrow\tau\theta_i+(1-\tau)\hat{\theta}_i$；

(10) 直到Episode = M；

(11) 直到Time = T；

(12) 输出无人机飞行轨迹以及用户平均能量损耗。

标位置，根据式(24)和式(8)分别计算用户的能耗和回报$r(t)$，以及更新下一时刻的状态$s(t+1)$，并将$(s(t),a(t),r(t),s(t+1))$储存在经验缓冲区。将$s(t)$更新为$s(t+1)$，重复上述步骤直到训练结束。步骤(9)对应4.3节的深度神经网络训练，从经验缓冲区中随机采样小批量经验样本，根据式(21)、式(22)和式(23)进行网络参数的更新。

5　仿真结果

本文使用计算机仿真验证所提算法的性能，仿真的软件环境为Python 3.6和Pytorch框架，仿真的硬件平台为具有AMD Ryzen 7-5800H 3.20 GHz处理器、NVIDIA RTX 3050Ti显卡、16 GB内存的个人电脑。设定无人机在400 m×400 m范围的规定区域内飞行，且飞行高度固定为75 m，无人机的起始位置和终点位置分别为[200,10,75] m和[200,350,75] m。在飞行区域内随机分布着40个需要执行计算任务的用户，用户的计算任务$I_i=\{D_i,F_i\}$随机生成，其中$D_i\in[10,100]$ kB和$F_i\in[1\times10^8,1\times10^9]$ Hz。训练阶段的算法迭代次数为2 000次，每次迭代无人机飞行的最大时隙数为100。本文其他参数设置如表1所示。

下面提供了仿真结果来验证本文提出的联合优化无人机轨迹和用户调度方案的性能。在结果中，将所提方案称为“联合优化方案”。为了进行比较，本文还考虑了以下3个对比方案。

在对比方案1中，取所有用户的坐标中心点$[X_{\rm C},Y_{\rm C}]$，无人机从起始位置出发，经过中心点再抵达终点位置，无人机在飞行过程以恒定速度v'匀速飞行。在每个时隙，无人机随机接收覆盖范围内的$K(K\le K_{\max})$个用户的卸载任务，利用MEC服务器为其提供计算服务，将该方案简称为“随机调度方案”。

在对比方案2中，无人机采用与“随机调度方案”相同的飞行轨迹，并利用上节提出的SAC算法进行用户调度。无人机根据用户调度的优化策略为覆盖范围内的$K(K\le K_{\max})$个用户提供计算服务。该方案简称为“基于SAC的优化用户调度方案”。

对比方案3中，采用基于DDPG算法[12]的联合优化无人机轨迹和用户调度的对比方案。该方案与本文提出的“基于SAC的联合优化方案”作比较，分析DDPG算法和SAC算法应用于无人机辅助移动边缘计算系统的性能表现。在该方案中，状态空间、动作空间和回报的定义与“基于SAC的联合优化方案”基本一致。该方案简称为“基于DDPG的联合优化方案”。

图4为不同方案下的无人机2维飞行轨迹对比图，“随机调度方案”和“基于SAC的优化用户调度方案”的无人机飞行轨迹相同，无人机沿着一条相对直接的路径匀速飞到用户中心点，然后到达终点位置。“基于DDPG的联合优化方案”的无人机经过用户稀疏的区域上空，服务远处的地面用户接着飞往终点位置。这也说明了DDPG算法的探索性较差，在满足抵达终点位置的前提下难以探索到更优的轨迹，陷入次优解。相比之下，“基于SAC的联合优化方案”的无人机飞行轨迹偏离对比方案轨迹，向用户密集的区域靠近。无人机从起始位置出发，以较快速度飞往用户密集区域上空，接着从多个地面用户之间低速飞行，最后沿着一条弧形路径快速抵达终点位置，在弧形路径上无人机为较远处的地面用户提高计算服务。与固定飞行轨迹的对比方案比较，在所提算法中无人机在满足抵达终点位置的约束前提下，偏离了最短的飞行距离，能够自适应地调整其飞行轨迹以覆盖计算任务特别重的用户为其提供计算服务。与“基于DDPG的联合优化方案”对比，本文提出“基于SAC的联合优化方

表 1　实验仿真参数

参数	符号表示	设定值
无人机最大覆盖用户数量	$K_{\max}$	3
最大时延(s)	$T_{\max}$	1
无人机最大飞行距离(m)	$d_{\max}$	10
用户发射功率(W)	$P^{\rm Tr}$	0.1
无人机总带宽(MHz)	W	6
无人机接收天线的最大接收角度	θ	$\pi/4$
参考距离(1 m)的信道功率增益	g_0	1.42×10^{-4}
噪声功率(dBm)	n^2	–90
无人机提供的CPU周期数(Hz)	$f^{\rm U}$	5×10^9

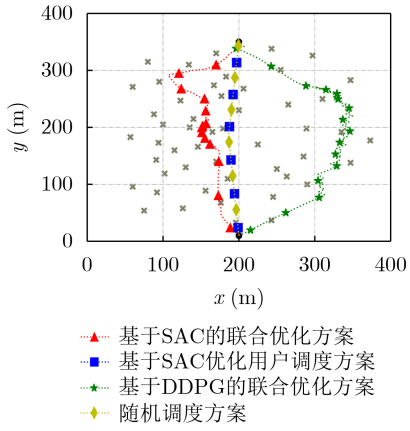

图 4 不同方案下的无人机2维飞行轨迹对比图

案”的最优飞行轨迹更加平滑，且无人机在每个时隙与被服务的多个用户保持合适的距离，合理分配卸载速率以降低平均卸载时延。

图5表示不同方案的用户平均能耗的对比图。“随机调度方案”的用户平均能耗保持不变且始终高于其他3个方案的能耗。在“基于SAC的优化用户调度方案”中，智能体在200幕之前处在探索阶段，用户的平均能耗较高。在200～300幕，随着网络参数更新，用户快速学习到最优的卸载方案，用户平均能耗快速降低。在300幕之后达到收敛，且明显低于“随机调度方案”的能耗。仿真结果显示用户的策略调度对于降低用户能耗尤为重要。对于“基于DDPG的联合优化方案”和“基于SAC的联合优化方案”的能耗曲线，我们仅保存智能体充分适应环境之后，无人机满足抵达终点位置约束的用户平均能耗数据。两个方案的用户平均能耗达到收敛时，“基于DDPG的联合优化方案”用户平均能耗仅比“基于SAC的优化用户调度方案”低0.05 J，但比“基于SAC的联合优化方案”高接近0.5 J。本文提出“基于SAC的联合优化方案”的用户能量消耗性能明显低于其他3种对比方案，无人机和用户分别学习到最优的飞行轨迹和卸载方案。

图6显示了在不同无人机最大覆盖用户数量$K_{\max}$下，不同方案实现的用户平均能耗对比，其中地面用户个数N固定为50，$K_{\max}$分别取3,4,5,6和7。本文提出的目标方案和对比方案的用户平均能耗随着无人机最大覆盖用户数量增加而降低，且“基于SAC的联合优化方案”的用户平均能耗最低，“随机调度方案”的用户平均能耗最高。在用户数量固定的情况下，当无人机最大覆盖用户数量$K_{\max}$为3时，“基于SAC的联合优化方案”与其他方案的用户平均能耗差距较小，但是随着$K_{\max}$的增加，优势差距逐渐扩大。“基于DDPG的联合优化方案”在无人机最大覆盖用户数量$K_{\max}$时等于3,4和5时，用户平均能耗表现与“基于SAC的优化用户调度方案”接近。由于无人机可自适应调整其飞行轨迹，当无人机最大覆盖用户数量$K_{\max}$增加到6和7时，“基于DDPG的联合优化方案”性能优势得以体现，然而相比于“基于SAC的联合优化方案”仍然较差。以上结果进一步表明，在无人机最大覆盖用户数量$K_{\max}$较大时，轨迹优化更能有效地降低用户能耗。

图7显示了在不同地面用户个数N下，不同方案实现用户平均能耗的对比，其中无人机最大覆盖用户数量$K_{\max}$固定为3，N分别取35,40,45和50。显而易见，在固定无人机最大覆盖用户数量下，所有方案的用户平均能耗随着用户数量的增加而增加。然而在不同用户数量的场景下，“基于SAC的联合优化方案”的用户平均能耗始终低于其他3个对比方案，“基于DDPG的联合优化方案”次之，“随机调度方案”性能最差。以上结果再一次表明了联合优化无人机轨迹和用户调度在最小化用户平均能耗方面的重要性和必要性，且SAC算法的性能表现比DDPG算法更优。

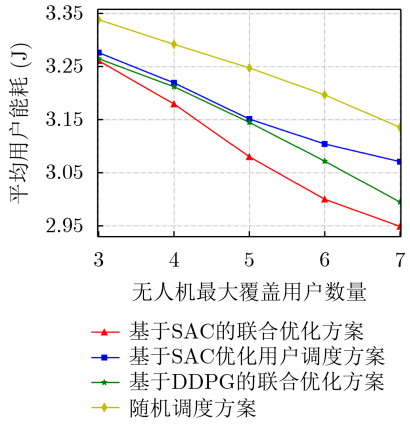

图 6 不同无人机最大覆盖用户数量条件下的用户平均能耗对比图

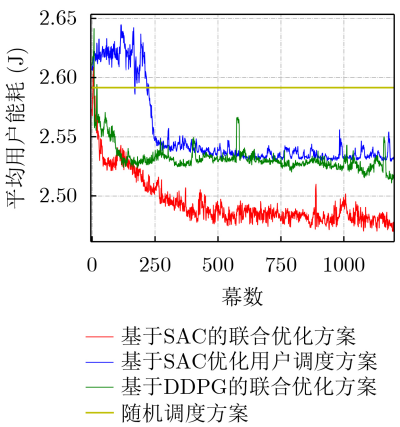

图 5 不同算法下的用户平均能耗对比图

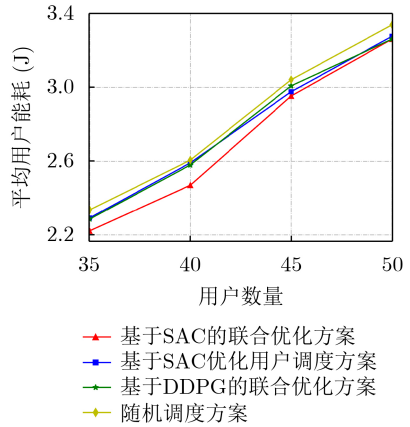

图 7 不同用户数量条件下的用户平均能耗对比图

6　结论

本文考虑了无人机辅助移动边缘计算系统，无人机作为搭载MEC服务器的平台。本文的目标是通过联合优化无人机的飞行轨迹和地面用户计算策略调度来最小化用户的平均能耗。为了解决连续动作空间的问题，本文提出了一种基于深度强化学习的SAC算法。仿真结果表明，所提基于SAC算法的联合优化算法可以有效地降低用户执行计算任务的能耗，且SAC在该模型上的能耗表现比DDPG算法更加突出。

参 考 文 献

[1] MAO Yuyi, YOU Changsheng, ZHANG Jun, *et al*. A survey on mobile edge computing: The communication perspective[J]. *IEEE Communications Surveys & Tutorials*, 2017, 19(4): 2322–2358. doi: 10.1109/COMST.2017.2745201.

[2] MAO Yuyi, ZHANG Jun, and LETAIEF K B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices[J]. *IEEE Journal on Selected Areas in Communications*, 2016, 34(12): 3590–3605. doi: 10.1109/JSAC.2016.2611964.

[3] LIU Tianyu, CUI Miao, ZHANG Guangchi, *et al*. 3D trajectory and transmit power optimization for UAV-enabled multi-link relaying systems[J]. *IEEE Transactions on Green Communications and Networking*, 2021, 5(1): 392–405. doi: 10.1109/TGCN.2020.3048135.

[4] LYU Xinchen, TIAN Hui, NI Wei, *et al*. Energy-efficient admission of delay-sensitive tasks for mobile edge computing[J]. *IEEE Transactions on Communications*, 2018, 66(6): 2603–2616. doi: 10.1109/TCOMM.2018.2799937.

[5] WU Qingqing and ZHANG Rui. Common throughput maximization in UAV-enabled OFDMA systems with delay consideration[J]. *IEEE Transactions on Communications*, 2018, 66(12): 6614–6627. doi: 10.1109/TCOMM.2018.2865922.

[6] LI Zhiyang, CHEN Ming, PAN Cunhua, *et al*. Joint trajectory and communication design for secure UAV networks[J]. *IEEE Communications Letters*, 2019, 23(4): 636–639. doi: 10.1109/LCOMM.2019.2898404.

[7] LI Yuxi. Deep reinforcement learning: An overview[EB/OL]. https://arxiv.org/abs/1701.07274, 2021.

[8] PENG Yingsheng, LIU Yong, and ZHANG Han. Deep reinforcement learning based path planning for UAV-assisted edge computing networks[C]. 2021 IEEE Wireless Communications and Networking Conference, Nanjing, China, 2021: 1–6. doi: 10.1109/WCNC49053.2021.9417292.

[9] SEID A M, BOATENG G O, ANOKYE S, *et al*. Collaborative computation offloading and resource allocation in multi-UAV-assisted IoT networks: A deep reinforcement learning approach[J]. *IEEE Internet of Things Journal*, 2021, 8(15): 12203–12218. doi: 10.1109/JIOT.2021.3063188.

[10] FUJIMOTO S and GU S S. A minimalist approach to offline reinforcement learning[C]. The 34th Annual Conference on Neural Information Processing Systems, Vancouver, Canada, 2021.

[11] HAARNOJA T, ZHOU A, HARTIKAINEN K, *et al*. Soft actor-critic algorithms and applications[EB/OL]. https://arxiv.org/abs/1812.05905, 2021.

[12] LILLICRAP T P, HUNT J J, PRITZEL A, *et al*. Continuous control with deep reinforcement learning[C]. The 4th International Conference on Learning Representations, San Juan, Puerto Rico, 2021.

[13] ZHANG Guangchi, YAN Haiqiang, ZENG Yong, *et al*. Trajectory optimization and power allocation for multi-hop UAV relaying communications[J]. *IEEE Access*, 2018, 6: 48566–48576. doi: 10.1109/ACCESS.2018.2868117.

[14] YU Zhe, GONG Yanmin, GONG Shimin, *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. doi: 10.1109/JIOT.2020.2965898.

[15] HUANG Yingqian, CUI Miao, ZHANG Guangchi, *et al*. Bandwidth, power and trajectory optimization for UAV base station networks with backhaul and user QoS constraints[J]. *IEEE Access*, 2020, 8: 67625–67634. doi: 10.1109/ACCESS.2020.2986075.

[16] YANG Zhaohui, PAN Cunhua, WANG Kezhi, *et al*. Energy efficient resource allocation in UAV-enabled mobile edge computing networks[J]. *IEEE Transactions on Wireless Communications*, 2019, 18(9): 4576–4589. doi: 10.1109/TWC.2019.2927313.

[17] ZHANG Guangchi, WU Qingqing, CUI Miao, *et al*. Securing UAV communications via joint trajectory and power control[J]. *IEEE Transactions on Wireless Communications*, 2019, 18(2): 1376–1389. doi: 10.1109/TWC.2019.2892461.

[18] WANG Xinhou, WANG Kezhi, WU Song, *et al*. Dynamic resource scheduling in mobile edge cloud with cloud radio access network[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2018, 29(11): 2429–2445. doi: 10.1109/TPDS.2018.2832124.

[19] JIANG Feibo, WANG Kezhi, DONG Li, *et al*. Deep-learning-based joint resource scheduling algorithms for hybrid MEC networks[J]. *IEEE Internet of Things Journal*, 2020, 7(7): 6252–6265. doi: 10.1109/JIOT.2019.2954503.

张广驰：男，教授，研究方向为新一代无线通信技术.

何梓楠：男，硕士生，研究方向为无人机通信、强化学习.

崔　苗：女，讲师，研究方向为新一代无线通信技术.

责任编辑：余　蓉