

基于无人机辅助移动边缘计算的任务卸载

刘 佳, 贾小林

青岛大学数学与统计学院, 山东 青岛

收稿日期: 2025年1月21日; 录用日期: 2025年2月22日; 发布日期: 2025年3月11日

摘 要

由于无人机能够灵活部署, 因此可以帮助提高覆盖范围和通信质量。本文考虑了一种无人机辅助的移动边缘计算系统, 其中配备有计算资源的无人机可以向附近的用户设备提供卸载服务。用户将部分计算任务卸载到无人机, 而其余任务在用户本地执行。我们的目标是通过联合优化用户任务调度、任务卸载比率、传输功率、无人机飞行角度和飞行速度到达最小化系统成本的目的。并且考虑到该优化问题是非凸的, 我们提出了一种基于深度确定性策略梯度的强化学习计算卸载算法。通过该算法, 我们可以在不可控的动态环境中获得最优的计算卸载策略。并且通过仿真结果表明, 该算法优于其他强化学习算法。

关键词

计算卸载, 无人机, 移动边缘计算, 深度强化学习

Task Offloading Based on UAV Assisted Mobile Edge Computing

Jia Liu, Xiaolin Jia

College of Mathematics and Statistics, Qingdao University, Qingdao Shandong

Received: Jan. 21st, 2025; accepted: Feb. 22nd, 2025; published: Mar. 11th, 2025

Abstract

Due to the flexible deployment of drones, they can help improve coverage and communication quality. This paper considers a UAV assisted mobile edge computing system, in which the UAV equipped with computing resources can provide unloading services to nearby user devices. Users offload some computing tasks to the drone, while the remaining tasks are executed locally by the user. Our goal is to minimize system costs by jointly optimizing user task scheduling, task offloading ratio, transmission power, drone flight angle, and flight speed. And considering that the optimization problem is non-convex, we propose a reinforcement learning computation offloading algorithm

based on Soft Actor Critic. Through this algorithm, we can obtain the optimal computation offloading strategy in uncontrollable dynamic environments. And the simulation results show that this algorithm is superior to other reinforcement learning algorithms.

Keywords

Computing Offloading, UAV, Mobile Edge Computing, Deep Reinforcement Learning

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着科学技术的发展以及智能手机、平板电脑及各种移动设备的普及, 移动计算已成为日常生活中不可或缺的一部分, 并且应用到很多领域如医疗健康、物联网以及金融风险监控等。这些应用往往需要大量的计算资源并且会产生非常高的能耗, 然而, 对于用户的本地设备来说, 该设备的计算能力、存储空间和电池寿命等都是有限的, 因此只能进行一些简单的计算任务。云计算的出现可以弥补某些方面的不足, 但是, 在面向众多用户并产生大量数据的应用场景中, 将计算任务从移动设备迁移到云端的过程会消耗巨量的网络带宽。这很可能突破网络的安全承载上限, 引发网络拥堵, 进而导致通信延迟至无法容忍的程度[1]。因此, 对于 5G 时代那些对延迟极为敏感、复杂度与可靠性要求又极高的计算任务而言, 传统的云计算卸载方式已不再适用[2]。为了普及并妥善处理这类计算任务, 我们必须探索并采用更为先进的计算模式。

移动边缘计算(Mobile Edge Computing, MEC)克服了传统云计算的不足, MEC 系统通过在网络的边缘部署计算节点, 为用户就近提供了丰富的计算能力, 有效解决了用户与远端云中心之间因距离过远而产生的问题, 同时也缓解了因数据回传至云端而造成的网络容量拥堵状况[3]。用户的无线设备可以将计算任务卸载到具有通信资源的节点上, 克服计算限制[4]。但是, 在某些情况下, MEC 也会受到限制。例如, 当无线设备与通信基站相距很远并且有障碍物阻挡时, 通信链路受到阻碍, 那么通信会受到阻碍, 影响任务卸载过程。无人机(Unmanned Aerial Vehicle, UAV)具有出色的灵活性和机动性, 因此可以用于 MEC 中, 为远程用户提供通信服务[5]。UAV 辅助 MEC 提供了到 UE 的更好的无线连接(视线), 并且在实现中具有更大的灵活性[6]。因此, UAV 辅助 MEC 系统的研究具有非常重要的实际意义。

近年来, 对于 UAV 辅助 MEC 系统用于任务卸载的研究有非常多, Cheng 等[7]在任务和能量约束条件下, 通过联合优化任务卸载、资源分配以及 UAV 轨迹最小化 UAV 能耗和完成时间。Liang 等[8]为了保证通信服务, 考虑优化 UAV 部署和资源分配达到降低平均用户延迟的目的。Li 等[9]联合优化 UAV 的路径轨迹和任务分配, 实现系统的能耗最小化。Cao 等人[10]研究了三节点系统(UE、中继节点和接入点)中 MEC 的联合计算和通信问题, 目标是最小化 UE 和中继节点的整体能耗。

然而, 大多数研究 UAV 辅助边缘计算时往往只考虑 MEC 系统能耗最小化, 但是 UAV 本身电量有限这一条件往往都忽略掉了[11], 这可能会出现任务未处理完而 UAV 由于电量耗尽无法飞行, 导致最终的任务卸载无法完成。在无人机辅助 MEC 系统中, 网络连接、带宽和延迟的变化都可能影响任务的卸载效率和调度性能。无人机的飞行路径不确定, 可能导致通信链路的不稳定, 给任务卸载和调度带来额外的复杂性。无人机的计算能力有限, 同时飞行时的能源消耗也使得其无法长时间提供高效的计算服务。因此, 在设计卸载和调度策略时, 如何平衡计算负载、能源消耗和任务完成时间成为一个主要挑战。无

人机辅助 MEC 系统需要同时优化多个目标, 如减少延迟、提高吞吐量、优化能效等, 如何设计合适的调度策略来解决这些目标之间的权衡, 是一个重要的问题。

针对以上问题, 本文研究 UAV 辅助 MEC 环境下考虑 UAV 自身电量的同时, 优化 MEC 系统总成本最小化问题也就是时延和能耗权重和最小化问题, 主要贡献如下:

(1) 提出了基于 UAV 辅助 MEC 系统模型。考虑到 UAV 的飞行能耗和电池容量等条件, 建立了用户调度、任务卸载比、传输功率、无人机飞行角度和飞行速度的联合优化问题, 从而实现终端用户总成本最小化即总能耗和时延的权重之和最小化的目的。

(2) 将优化问题转化为一个无转移概率的马尔可夫决策过程(Markov decision process, MDP), 并根据问题设计场景和需求, 同时定义了 MDP 的状态空间、动作空间和奖励函数, 并且使用深度强化学习(Deep reinforcement learning, DRL)中的深度确定性策略梯度算法(Deep Deterministic Policy Gradient, DDPG)求解问题模型, 并且我们的算法达到了更好的效果。

2. 系统模型

考虑一个 UAV 辅助的 MEC 系统, 有 U 个 UAV 和 M 个 UE 组成, 其中 $U = \{1, \dots, k, \dots, u\}$, $M = \{1, \dots, i, \dots, m\}$, 每个 UE 可以处理少量的计算任务。由于 UE 本地计算能力有限, 所以, UE 会选择将计算任务在本地处理或者是将部分计算任务卸载到 UAV 上处理。考虑采取正交频分多址将 UE 的任务向 UAV 传输, 将持续时间 T 分成 I 个相同的时隙, 在每个时隙下 UAV 仅处理一个 UE 任务。假设 UAV 在时间 T 内的飞行高度始终是 H , 建立 3D 笛卡尔坐标系。假设 UAV k 在时隙 i 下的位置定义为 $L_k(i) = (x_k(i), y_k(i), H)$, 由于固定了 UAV 的飞行高度 H , 考虑 UAV 的 2D 坐标, 定义为 $L'_k(i) = (x_k(i), y_k(i))$, 在下一时隙下 UAV 的位置为: $L_k(i+1) = (x_k(i+1), y_k(i+1), H)$, 并且有:

$$x_k(i+1) = x_k(i) + v_k(i)t_{fy} \cos \theta_k(i) \quad (1)$$

$$y_k(i+1) = y_k(i) + v_k(i)t_{fy} \sin \theta_k(i) \quad (2)$$

其中, $v_k(i)$ 是 UAV k 在时隙 i 下的飞行速度, $0 \leq v_k(i) \leq v_{\max}$ 且 $\theta_k(i) \in [0, 2\pi)$ 是 UAV k 在时隙 i 下水平方向移动距离的角度, t_{fy} 是飞行时间。

此外, 还要考虑到 UAV 的飞行范围, 保证其只在服务范围内移动, 定义 X_{\max} , Y_{\max} 为 UAV 服务范围的总长度和总宽度。因此, 在任意时间间隔下都有:

$$0 \leq x_k(i) \leq X_{\max}, \forall k \in U \quad (3)$$

$$0 \leq y_k(i) \leq Y_{\max}, \forall k \in U \quad (4)$$

同时, 考虑任意两个无人机之间的移动, 为了避免无人机发生碰撞, 设置一个最小距离 L_{\min} , 因此, 在任意时间间隔下有以下碰撞约束条件:

$$\|L_k(i) - L_j(i)\| \geq L_{\min}, \forall k, j \in U \quad (5)$$

2.1. 通讯模型

UAV k 在时隙 i 下的位置定义为: $L_k(i) = (x_k(i), y_k(i), H)$, 同时 UE m 在时隙 i 下的坐标可以定义为 $L_m(i) = (x_m(i), y_m(i), 0)$ 。那么在时隙 i 下 UE m 到 UAV k 的信道链路的信道增益表示为:

$$h_{km}(i) = \frac{h_0}{\|L_k(i) - L_m(i)\|^2} \quad (6)$$

其中 $\|L_k(i) - L_m(i)\|$ 表示 UE m 和 UAV k 的欧几里得距离, 且 h_0 表示在参考距离 $d_0 = 1$ 米处传播的信道增益。

在 UE m 向 UAV k 卸载过程中, 在时隙 i 下 UE m 的数据传输速率为:

$$R_{km}(i) = B \log_2 \left(1 + \frac{h_{km}(i) P_m(i)}{\sigma^2} \right) \quad (7)$$

其中, B 表示通信带宽, $0 \leq P_m(i) \leq P_{\max}$ 表示在时隙 i 下 UE m 向 UAV k 进行计算卸载时在上传链路中的发射功率, P_{\max} 表示 UE m 的最大发射功率, σ^2 表示噪声功率。

定义 $D_m(i)$ 为 UE m 的计算任务大小, 在我们的系统中, 用户的卸载策略采用的是部分卸载策略。定义 $\varphi_m(i) \in [0, 1]$ 为 UE m 的在时隙 i 下的卸载比例。其中, 卸载比例表示从本地设备卸载到边缘处理器的任务比例, 卸载比为 0 时意味着用户不将自己的计算任务卸载到边缘处理器上而是用本地的处理器进行, 卸载比为 1 时意味着用户将自己的计算任务全部卸载到边缘处理器上进行处理。

$$\varphi_m(i) = \begin{cases} 0, & \text{不卸载} \\ (0, 1), & \text{部分卸载} \\ 1, & \text{完全卸载} \end{cases} \quad (8)$$

在 MEC 系统中, 由服务器提供的计算结果通常非常小可以忽略不记。因此, 不考虑下行链路的发送延迟。考虑以下几个方面的延迟和能耗: UE m 与 UAV k 通信的传输时间延迟、计算时间延迟和能量消耗。

UE m 与 UAV k 通信的传输延迟和传输能耗为:

$$t_{UAVk,m}^{tra}(i) = \frac{\varphi_m(i) D_m(i) s}{R_{km}(i)} \quad (9)$$

$$E_{UAVk,m}^{tra}(i) = P_m(i) t_{UAVk,m}^{tra}(i) \quad (10)$$

其中 s 表示处理每个单位字节所需的 CPU 周期。

2.2. 计算模型

UE m 在时隙 i 下的本地计算时间延迟为:

$$t_{local,m}(i) = \frac{(1 - \varphi_m(i)) D_m(i) s}{C_{UE}} \quad (11)$$

其中 C_{UE} 表示 UE 的计算能力, 我们假设每个用户 UE 的计算能力相同。

UE m 与 UAV k 通信计算时间延迟为:

$$t_{UAVk,m}^{com}(i) = \frac{\varphi_m(i) D_m(i) s}{C_{UAVk}} \quad (12)$$

其中 C_{UAVk} 是 UAV k 的 CPU 计算能力。

当 MEC 服务器在处理任务时, 它的计算功率可以表示为

$$P_{UAVk,m}(i) = \kappa C_{UAVk}^3 \quad (13)$$

因此计算能耗为:

$$E_{UAVk,m}(i) = P_{UAVk,m}(i) t_{UAVk,m}(i) = \kappa C_{UAVk}^2 \varphi_m(i) D_m(i) s \quad (14)$$

UAV 在时隙 i 下的飞行过程中的飞行能耗[12]可表示为:

$$E_{fly}(i) = 0.5M_{UAV}t_{fly} \|v_k(i)\|^2 \quad (15)$$

其中 M_{UAV} 为无人机的质量。

2.3. 问题公式

所有计算任务处理完成的时间为整个系统的最终时间, 在时隙 i 下整个系统的时间延迟可以表示为:

$$T(i) = \sum_{m=1}^M \alpha_m(i) \max\{t_{local,m}(i), t_{UAVk,m}^{tra}(i) + t_{UAVk,m}^{com}(i)\} \quad (16)$$

其中 $\alpha_m(i) = \{0,1\}$, 表示在任一时隙下都有任务需要处理。

因此, 系统的总时延表示为:

$$T = \sum_{i=1}^I T(i) \quad (17)$$

在时隙 i 下处理完计算任务的能量消耗可以表示为:

$$E(i) = \sum_{m=1}^M \alpha_m(i) (E_{UAVk,m}^{tra}(i) + E_{fly} + E_{UAVk,m}^{com}(i)) \quad (18)$$

$$\sum_{i=1}^I E(i) \leq E_C \quad (19)$$

其中 E_C 为 UAV 的电池总电量。

因此, 系统的总能耗表示为:

$$E = \sum_{i=1}^I E(i) \quad (20)$$

定义时间延迟和能耗的权重和表示为系统成本, 因此, 优化问题可以表示为:

$$\min_{(v_k(i), \theta_k(i), \varphi_m(i), P_m(i), \alpha_m(i))} \sum_{k=1}^U \omega_1 T + \omega_2 E \quad (21)$$

$$s.t. \ 0 \leq \varphi_m(i) \leq 1, \forall i \in I, m \in M \quad (22)$$

$$\sum_{i=1}^I \sum_{m=1}^M \alpha_m(i) D_m(i) = D \quad (23)$$

$$0 \leq v_k(i) \leq v_{\max}, \forall k \in U \quad (24)$$

$$0 \leq \theta_k(i) < 2\pi, \forall k \in U \quad (25)$$

$$\omega_1 + \omega_2 = 1 \quad (26)$$

$$\sum_{i=1}^I E(i) \leq E_C \quad (27)$$

$$0 \leq x_k(i) \leq X_{\max}, \forall k \in U \quad (28)$$

$$0 \leq y_k(i) \leq Y_{\max}, \forall k \in U \quad (29)$$

其中约束(22)表示计算任务的卸载率的值范围; 约束(23)指定了在整个时间段内要完成的所有计算任务; 约束条件(24)和(25)给出 UAV 的水平移动速度和角度的范围; 约束(27)确保无人机在所有时隙飞行能耗和计算能耗不超过最大电池容量; 约束条件(28)和(29)表明 UAV 只能在给定区域内移动。

3. 基于深度强化学习的计算卸载

由于我们要解决的优化问题是非凸优化问题, 传统的数学方法解决较为困难, 使用深度强化学习算法可以很好的解决。将问题描述为马尔可夫决策过程, 并此提出了基于 DDPG 的计算卸载算法。

DDPG 算法是强化学习领域中的一种基于策略梯度的算法, 旨在解决连续动作空间中的强化学习问题。它结合了深度学习和确定性策略梯度方法, 通过使用深度神经网络来近似策略和价值函数, 从而有效地处理高维和复杂的状态 - 动作空间, 同时 DDPG 是基于确定性策略梯度(DPG)的扩展, 传统的策略梯度方法, 通常适用于离散动作空间, 而在处理连续动作空间时, DDPG 提供了更高效的解决方案。

状态空间: 在无人机辅助 MEC 系统框架中, 状态空间是由用户、无人机和环境一起决定的, 因此在时隙 i 下的状态空间可以描述为:

$$s_i = \{L_u(i), L_m(i), Q_m(i), E_{battery}(i)\} \quad (30)$$

其中, $L_u(i)$ 表示无人机在时隙 i 下的位置信息, $L_m(i)$ 表示用户在时隙 i 下的位置信息, $Q_m(i)$ 表示用户的任务信息包括任务大小和传输功率, $E_{battery}(i)$ 表示无人机的剩余电量。

动作空间: 智能体通过与环境不断交互, 当智能体获得系统状态时, 根据当前环境会做出相应的动作。相应的动作有无人机的飞行速度和角度, 用户的卸载比率。因此, 在时隙 i 的动作空间可以描述为:

$$a_i = \{\theta_k(i), v_k(i), \varphi_m(i)\} \quad (31)$$

其中, $\theta_k(i)$ 表示无人机在时隙 i 下的飞行角度, $v_k(i)$ 表示无人机的飞行速度, $\varphi_m(i)$ 表示用户的卸载比率。

奖励函数: 智能体在执行动作后会得到相应的奖励, 奖励函数与我们的优化目标函数有关。我们的目标是最小化优化问题(19), 因此, 将奖励函数定义为:

$$r_i = \{s_i, a_i\} = -C(i) \quad (32)$$

其中, 在时隙 i 下, $C(i)$ 表示为

$$C(i) = T(i) + E(i). \quad (33)$$

利用 DDPG 算法, 我们能够持续地对策略网络和价值网络进行训练与更新, 旨在探寻最优的用户任务卸载动作。下面说明基于 DDPG 的卸载决策算法的实现过程。

输入: 初始化神经网络参数权重和经验缓冲区

输出: 神经网络参数

- 1) 对于每个回合数
- 2) 初始化: 无人机位置, 用户位置, 用户任务, 无人机电量
- 3) 从环境中获取状态 s_i , 并根据当前状态做出当前动作 a_i
- 4) 计算下一状态 s_{i+1}
- 5) 根据优化问题公式(20)计算奖励 r_i
- 6) 存储经验 (s_i, a_i, s_{i+1}, r_i)
- 7) 从经验缓冲区随机采样小批次经验样本
- 8) 更新神经网络参数
- 9) end for

4. 仿真结果与分析

我们通过数值模拟说明了所提出的基于 DDPG 的无人机辅助 MEC 系统计算卸载算法。首先介绍了仿真参数的设置。然后, 基于 DDPG 的算法的性能进行了验证, 在不同的场景下, 并与其他基线方案进行了比较。

4.1. 参数设置

本文仿真环境基于 Python3.9 和 Pytorch2.0.0 编写, 以评估所提出的算法性能。在 UAV 辅助的 MEC 系统中, 我们考虑的一般场景是 $L \times W$ 为 $1000 \times 1000 \text{ m}^2$ 的正方形区域, 含有 2 个 UAV 辅助的边缘计算节点, 并且假设 UAV 以固定高度 H 为 100 m 飞行, 根据[13]的定义, 每个 UAV 的总质量 M_{UAV} 为 9.65 kg, 无人机的最大飞行速度 v_{\max} 为 20 m/s, 定义初始情况下无人机随机分散在区域内。用户数目为 10, 并且随机分散在区域内。整个系统被分为 40 时隙。在参考距离为 1 米时, 信道功率增益 h_0 设置为 50 dB。传输带宽 B 设置为 1 MHz。假设用户的噪声功率 σ^2 为 -100 dB。

我们假设用户的最大发射功率为 1 W, UAV 的电池容量为 500 KJ。并且 UE 和 UAV 的计算能力和分别为 2 GHz 和 5 GHz。对于所提算法的实现, Actor 和 Critic 网络的架构描述如下。Actor 和 Critic 网络是一个四层全连接神经网络, 包含两个隐藏层, 神经元数量分别为 256、256。目标 Actor 网络的架构与 Actor 网络相同。目标 Critic 网络的架构与 Critic 网络相同。利用 ReLU 作为激活函数, 利用 Adam 优化器更新网络权值, 分布式参与者的数量设置为 $J=10$ 。具体参数如下(表 1)。

Table 1. Simulation parameter settings

表 1. 仿真参数设置

参数	数值	参数	数值
L, W	1000 m	h_0	50 dB
H	100 m	B	1 MHz
I	40	σ^2	-100 db
M_{UAV}	9.65 kg	P_{\max}	1 W
v_{\max}	20 m/s	E_C	500 KJ
C_{UE}	2 GHz	C_{UAV}	5 GHz

为了进行性能比较, 我们将 DDPG 算法与三种深度强化学习算法进行了比较, 分别是近端策略优化算法(Proximal policy optimization, PPO)、软演说-评论家算法(Soft Actor-Critic, SAC)以及双延迟确定性策略梯度算法(Twin delayed deep deterministic policy gradient, TD3), 下面对其进行简要介绍:

(1) PPO 算法: PPO 算法是一种基于策略梯度的强化学习算法, 它通过限制新旧梯度之间的差异来确保策略更新。

(2) SAC 算法: SAC 算法是一种结合了最大熵强化学习和基于策略梯度方法的深度强化学习算法, 旨在通过最大化策略的熵来鼓励智能体在探索和利用之间找到平衡。

(3) TD3 算法: TD3 算法是引入了双重网络和延迟更新等技术来优化策略网络, 解决连续控制问题中的方差和偏差问题。

4.2. 数值分析

我们对算法的参数进行讨论, 图 1 是关于算法的学习率, 可以看出选择合适的学习率会使得算法收敛并能够获得更大的奖励。可以看出在学习率为 0.1 时, 算法后面虽然稳定但是智能体不断学习却获得的奖励变小; 当取 0.01 时, 随着训练回合数的增加, 算法虽然会收敛, 但是智能体获得的奖励值很低; 而当学习率为 0.001 和 0.0001 时, 可以看出算法收敛并且能够获得更大的奖励。并且当学习率为 0.001 时更稳定, 因此设置 Actor 网络和 Critic 网络的学习率为 0.001。

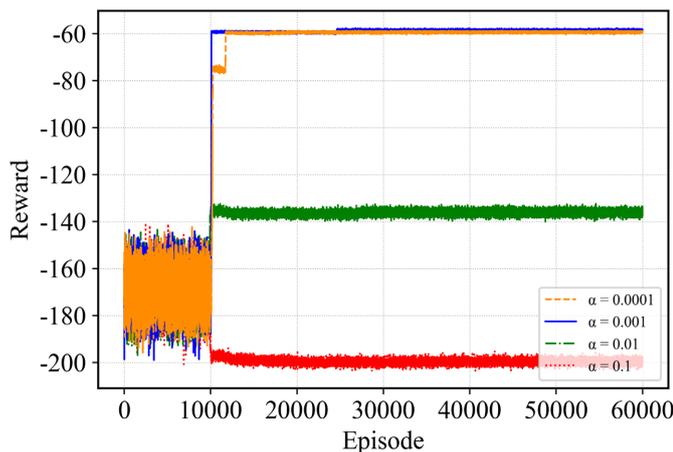


Figure 1. Performance of DDPG algorithm at different learning rates
图 1. 不同学习率下 DDPG 算法性能

图 2 展示了四种不同强化学习算法在奖励收敛性上的对比情况。随着训练步数的不断增加, 这四种算法的奖励均呈现出上升的趋势, 并最终都能够达到一个相对稳定的奖励值。这一现象说明, 强化学习智能体在与环境的交互过程中, 能够逐步学习到降低终端用户总成本的有效策略。我们发现 DDPG 虽然前期有较大的波动, 但是它最先达到收敛, 它的收敛速度是最快的, 并且 DDPG 算法的奖励值是更高的。而 SAC 算法和 TD3 算法收敛的速度要比 DDPG 算法慢同时奖励值也没有 DDPG 算法高, 而 PPO 算法的收敛速度是最慢的, 并且奖励是最低的。因此在效果上, DDPG 算法更好。

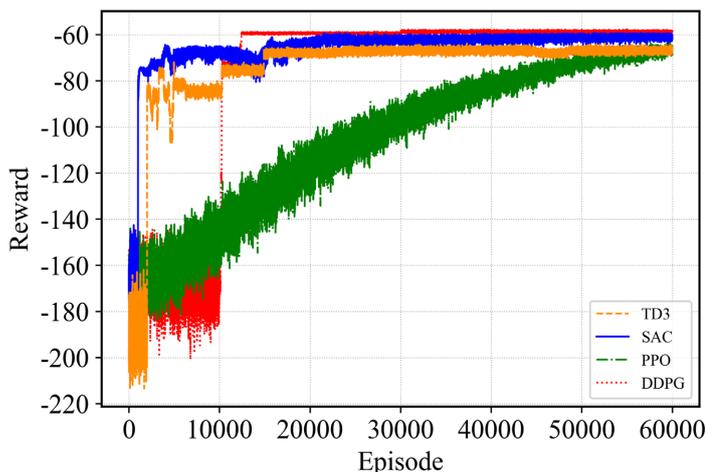


Figure 2. Convergence of different algorithms
图 2. 不同算法的收敛性

图 3 和图 4 展示了不同用户下各个算法的系统成本和时延情况, 图 3 展示了不同用户下各个算法的系统成本, 可以看出, 当用户数为 1 时, 不同算法下的系统成本相差不大, 但是随着用户的增加, 系统成本在不断增加, 并且 DDPG 算法的系统成本是最小的, 可以证明我们算法是有效性。图 4 展示了不同用户下各个算法的时延, 可以看出, 当用户数为 1 时, 不同算法下的时延相差不大, 但是随着用户的增加, 系统时延也是在不断增加, 并且 DDPG 算法的系统时延是最小的, 可以证明我们算法是有效性。

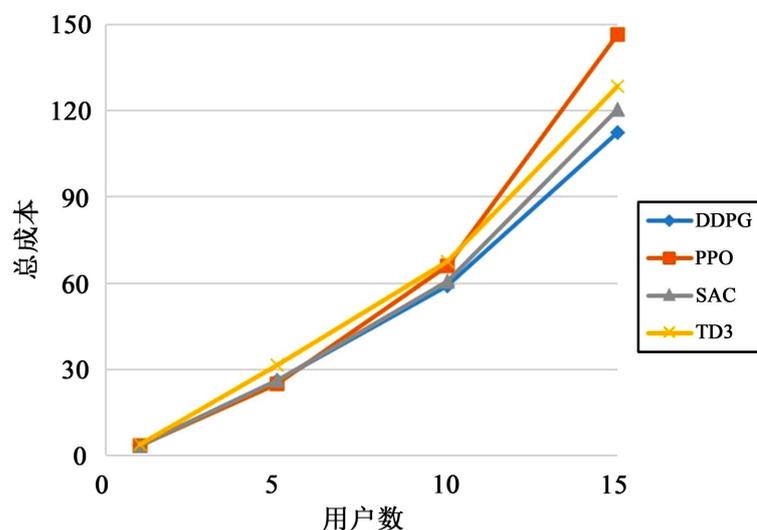


Figure 3. Cost of different users

图 3. 不同用户的成本

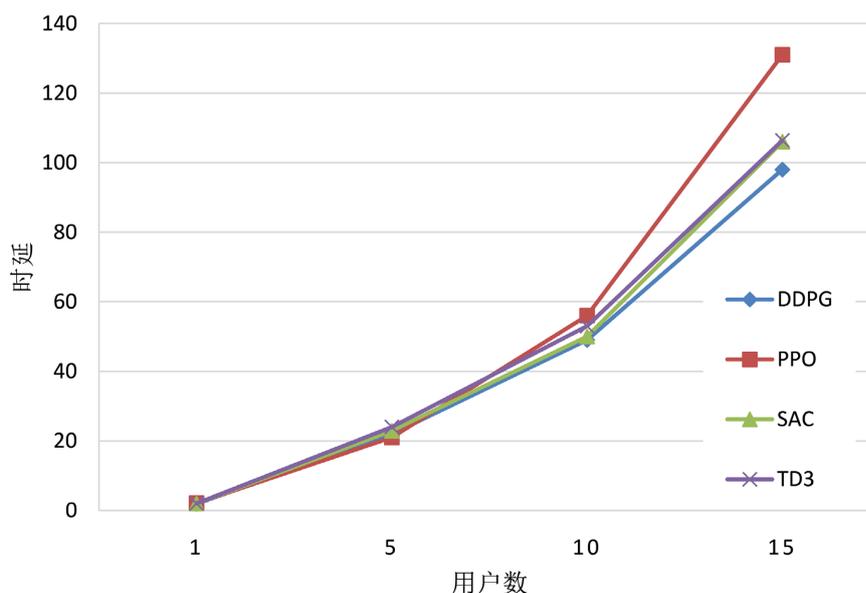


Figure 4. Time delay of different algorithm

图 4. 不同算法的时延

图 5 考虑 DDPG 算法下不同 UAV 的系统成本, 随着 UAV 的增加也就是边缘处理器的增加, 可以看出在计算任务不变的情况下, 我们的成本得到减小。

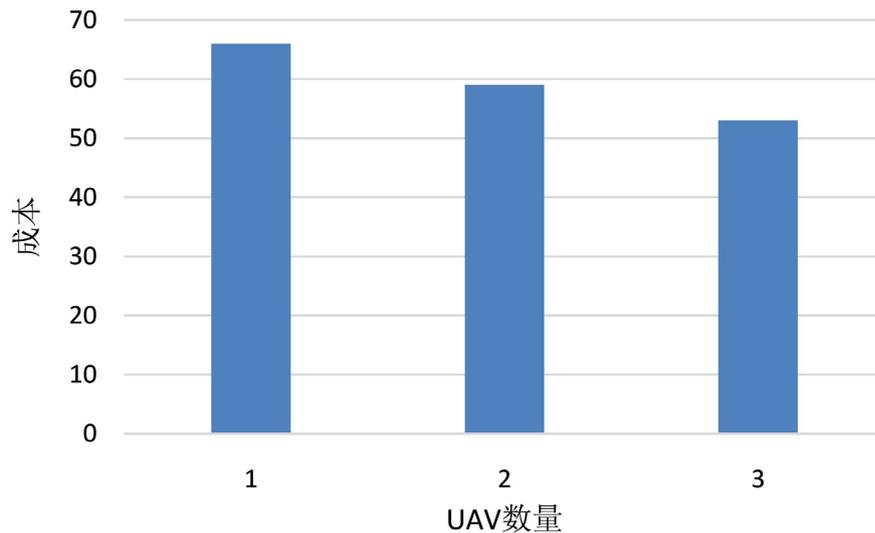


Figure 5. Cost of different UAV
图 5. 不同 UAV 数量的成本

5. 结束语

本文研究了无人机辅助移动边缘计算的任务卸载问题。首先, 我们考虑了联合用户任务调度、任务卸载比、传输功率、无人机飞行角度和飞行速度, 建立 MEC 系统的总能耗和时延权重和最小化问题; 其次, 将该问题转化成一个 MDP 过程并且使用 DDPG 算法求解。仿真结果表明所提算法能有效降低 MEC 系统的总成本。未来工作将考虑更复杂的情况, 多无人机和基站下的任务卸载等, 并且进一步提高系统性能。

参考文献

- [1] Agarwal, S., Philipose, M. and Bahl, P. (2014) Vision. *Proceedings of the 5th international workshop on Mobile Cloud Computing & Services-MCS'14*, Bretton Woods, 16 June 2014, 1-5. <https://doi.org/10.1145/2609908.2609946>
- [2] Tran, T.X., Hajisami, A., Pandey, P. and Pompili, D. (2017) Collaborative Mobile Edge Computing in 5G Networks: New Paradigms, Scenarios, and Challenges. *IEEE Communications Magazine*, **55**, 54-61. <https://doi.org/10.1109/mcom.2017.1600863>
- [3] Cheng, K., Fang, X. and Wang, X. (2023) Energy Efficient Edge Computing and Data Compression Collaboration Scheme for UAV-Assisted Network. *IEEE Transactions on Vehicular Technology*, **72**, 16395-16408. <https://doi.org/10.1109/tvt.2023.3289962>
- [4] Alnoman, A., Sharma, S.K., Ejaz, W. and Anpalagan, A. (2019) Emerging Edge Computing Technologies for Distributed IoT Systems. *IEEE Network*, **33**, 140-147. <https://doi.org/10.1109/mnet.2019.1800543>
- [5] Diao, X., Yang, W., Yang, L. and Cai, Y. (2021) UAV-Relaying-Assisted Multi-Access Edge Computing with Multi-Antenna Base Station: Offloading and Scheduling Optimization. *IEEE Transactions on Vehicular Technology*, **70**, 9495-9509. <https://doi.org/10.1109/tvt.2021.3101298>
- [6] Zhang, J., Zhou, L., Tang, Q., Ngai, E.C., Hu, X., Zhao, H., *et al.* (2019) Stochastic Computation Offloading and Trajectory Scheduling for UAV-Assisted Mobile Edge Computing. *IEEE Internet of Things Journal*, **6**, 3688-3699. <https://doi.org/10.1109/jiot.2018.2890133>
- [7] Zhan, C., Hu, H., Sui, X., Liu, Z. and Niyato, D. (2020) Completion Time and Energy Optimization in the UAV-Enabled Mobile-Edge Computing System. *IEEE Internet of Things Journal*, **7**, 7808-7822. <https://doi.org/10.1109/jiot.2020.2993260>
- [8] Zhang, L. and Ansari, N. (2020) Latency-Aware IoT Service Provisioning in UAV-Aided Mobile-Edge Computing Networks. *IEEE Internet of Things Journal*, **7**, 10573-10580. <https://doi.org/10.1109/jiot.2020.3005117>
- [9] Li, Y., Fang, Y. and Qiu, L. (2021) Joint Computation Offloading and Communication Design for Secure UAV-Enabled

-
- MEC Systems. 2021 *IEEE Wireless Communications and Networking Conference (WCNC)*, Nanjing, 29 March-1 April 2021, 1-6.
- [10] Cao, X., Wang, F., Xu, J., Zhang, R. and Cui, S. (2019) Joint Computation and Communication Cooperation for Energy-Efficient Mobile Edge Computing. *IEEE Internet of Things Journal*, **6**, 4188-4200.
<https://doi.org/10.1109/jiot.2018.2875246>
- [11] Hu, X., Wong, K., Yang, K. and Zheng, Z. (2019) UAV-Assisted Relaying and Edge Computing: Scheduling and Trajectory Optimization. *IEEE Transactions on Wireless Communications*, **18**, 4738-4752.
<https://doi.org/10.1109/twc.2019.2928539>
- [12] Hu, Q., Cai, Y., Yu, G., Qin, Z., Zhao, M. and Li, G.Y. (2019) Joint Offloading and Trajectory Design for UAV-Enabled Mobile Edge Computing Systems. *IEEE Internet of Things Journal*, **6**, 1879-1892.
<https://doi.org/10.1109/jiot.2018.2878876>
- [13] Jeong, S., Simeone, O. and Kang, J. (2018) Mobile Edge Computing via a UAV-Mounted Cloudlet: Optimization of Bit Allocation and Path Planning. *IEEE Transactions on Vehicular Technology*, **67**, 2049-2063.
<https://doi.org/10.1109/tvt.2017.2706308>