计算机工程
Computer Engineering
ISSN 1000-3428,CN 31-1289/TP

《计算机工程》网络首发论文

题目：空地算力网络中的异构资源协同优化
作者：李斌，山慧敏
网络首发日期：2024-08-22
引用格式：李斌，山慧敏．空地算力网络中的异构资源协同优化[J/OL]．计算机工程.
https://link.cnki.net/urlid/31.1289.TP.20240821.1510.011

网络首发时间：2024-08-22 10:35:27
网络首发地址：https://link.cnki.net/urlid/31.1289.TP.20240821.1510.011

空地算力网络中的异构资源协同优化

李斌*，山慧敏

（南京信息工程大学计算机学院，江苏 南京 210044）

摘 要：针对算力网络中终端用户计算能力不足及边缘节点算力分配不均的问题，本文提出了一种以激励机制为基础的无人机协同终端直连（Device-to-Device, D2D）边缘计算方案。首先，在满足计算资源、发射功率、计算资源单价等限制条件下，通过联合优化任务卸载比例、计算资源限制、无人机飞行轨迹，以及无人机及用户的发射功率和计算资源出售单价，构建一个系统收益最大化问题。其次，利用近端策略优化(Proximal Policy Optimization, PPO)方法确定用户卸载和购买策略，通过在多个时间步骤上迭代优化策略，最大化累积奖励，并引入剪切项以限制策略更新的幅度，以确保算法的稳定性。仿真结果显示，基于 PPO 的算法具有更好的收敛性，并能够有效提升系统总收益。

关键词：空地算力网络；激励机制；D2D 通信；计算卸载；近端策略优化

Collaborative Optimization of Heterogeneous Resources in Aerial-Ground Computing Power Networks

LI Bin*, SHAN Huimin

(School of Computer Science, Nanjing University of Information Science and Technology, Nanjing 210044, Jiangsu, China)

【Abstract】 In order to address the challenge of insufficient computing capacity of end users and unbalanced computing power distribution of edge nodes in computing power networks, this paper proposes an Unmanned Aerial Vehicle (UAV)-assisted Device-to-Device (D2D) edge computing solution based on incentive mechanisms. Firstly, under constraints involving computing resources, transmission power, and unit pricing of computing resources, a unified optimization problem is formulated. This problem aims to maximize system revenue by jointly optimizing task offloading ratio, computing resource constraints, UAV trajectory, as well as the transmission power and unit pricing of computing resources for both UAVs and users. Then, the Proximal Policy Optimization (PPO) method is employed to establish user offloading and purchasing strategies. In addition, an iterative approach is implemented at each time step to solve the optimization problem and obtain the optimal solution. Simulation results demonstrate that the PPO-based algorithm exhibits superior convergence and improve the overall system revenue in comparison to the baseline cases.

【Key words】 aerial-ground computing power network; incentive mechanism; D2D Communication; computation offloading; proximal policy optimization

1 引言

移动边缘计算（Mobile Edge Computing, MEC）技术能够促进物联网实时信息的评估，并为物联网设备提供更多的计算资源。通过计算卸载，物联网设备可以将任务发送到边缘服务器进行处理，从而降低物联网设备的能耗[1-3]。这一新型计算范式不仅有助于提升网络性能，还为新型应用和服务的部署创造了更加可靠和高效的基础设施。由于城市复杂的环境，地面边缘服务器和用户之间的无线链路往往被建筑物挡住，导致信道条件不稳定，服务质量差。为了克服这一问题，终端直连(Device-to-Device, D2D)被引入 MEC 网络中[4-5]，利用邻近空闲设备的计算能力辅助任务卸载，有效缓解了 MEC 服务器的压力，提高系统服务性能[6-8]。然而，当物联网设备部署在偏远地区或灾难地区时，由于远距离传输，与传统基站（Base Stations, BSs）建立通信链路效率低下。

无人机（Unmanned Aerial Vehicle, UAV）具有机动性强、覆盖范围广、灵活性高的优势，已经被广泛部署在各种场景中，特别是为边缘用户提供无线通信服务[9]。作为空中移动的边缘节点，UAV 可根据需求进行迅速部署，从而扩展立体覆盖范围并为地面用户提供灵活的计算服务[10]。文献[11]通过优化多个无人机的航路规划和计算资源分配，最小化系统的总能耗。文献[12]和文献[13]探讨了无人机中继协作 MEC 任务卸载，研究内容分别集中在最小化能耗和最小化时延两个方面。近期，UAV 协同 D2D 通信系统中的资源分配问题引起了广泛关注并取得了一些有价值的研究成果。譬如，文献[14]

基金项目：国家自然科学基金(62101277)

通信作者 E-mail：bin.li@nuist.edu.cn

针对无人机资源有限的问题，提出了 UAV 卸载和 D2D 卸载机制共存的策略，通过联合优化卸载决策和功率分配以最小化 UAV 计算资源。文献[15]通过 UAV 和 D2D 之间的协作提高任务卸载效率并降低计算时延，用以解决 UAV 资源有限的问题。文献[16]研究了 D2D 辅助通信下的 UAV 网络鲁棒资源分配问题，通过联合优化用户关联、功率分配、UAV 飞行高度和传输时间，提出了一种鲁棒的资源分配算法。文献[17]研究了 UAV 辅助的 D2D 能量采集网络，通过优化能量采集时间和功率分配最大化 UAV 网络的能效性。

然而，这些工作均是建立在空闲设备和 UAV 无私共享计算资源的假设基础上，由于 D2D 设备和 UAV 的能量和计算资源有限，不考虑计算资源提供者的利益是不现实和不公平的。因此，为了鼓励闲置设备加入协同卸载，有必要对资源提供者的利润激励机制进行建模。当邻近设备提供其资源(即功率和计算能力)时，它们也将相应地收到付款。文献[18]通过联合考虑具有计算需求的用户和能够提供计算服务的空闲用户之间的利益竞争关系，提出了一种基于 D2D 通信的计算卸载与资源分配方法。该算法通过优化需求用户的任务卸载决策、计算资源租赁单价决策以及空闲用户的计算资源分配决策，以最大化需求用户和空闲用户的效用。文献[19]提出了一种计算资源共享拍卖算法，激励空闲用户参与任务卸载。文献[20]研究了 MEC 网络的 D2D 协作任务卸载策略，通过 Stackelberg 博弈实现价格资源均衡提出的解决方案在任务分配中通过图匹配取得了较高计算利润，同时充分利用了空闲设备的计算资源。上述工作研究主要集中在用户间的激励机制，在 D2D 协助 UAV 边缘计算的激励机制方面研究相对匮乏，特别是忽略了 UAV 与用户的协同合作。因此，本文考虑将 UAV 和空闲用户同时作为计算资源提供者，联合考虑任务繁忙用户、空闲用户和 UAV 之间的利益关系，以系统总收益的最大化为目标。本文的主要工作如下：

（1）本文引入了 UAV 协同 D2D 技术，综合考虑了计算资源和计算成本等因素，建立了效用收益函数，通过联合优化价格策略、任务卸载策略和无人机飞行高度，实现了对系统总收益的最大化。

（2）为了解决这一非凸问题，本文采用深度强化学习(Deep Reinforcement Learning, DRL)，将卸载过程建模为马尔可夫决策(Markov Decision Process, MDP)过程，并运用近端策略优化(Proximal Policy Optimization, PPO）算法进行求解。与其他方案仿真比较，本文所提方案实现了在卸载过程中收益的最大化。

2 系统模型及问题描述

2.1 系统模型

UAV 协同 D2D 边缘计算网络如图 1 所示，该网络由 UAV 和多个用户组成。系统中的用户分为两类：计算任务繁忙用户和本地没有任务需要处理的空闲用户。假设任务繁忙用户数为 I，用集合 $I=\{1,2,...i,...,I\}$ 表示，空闲用户数为 J，用集合 $J=\{1,2,...,j,...,J\}$ 表示。在该网络中，UAV 和空闲用户充当资源提供者，为所有用户提供计算服务。计算任务繁忙用户是资源卸载的主体，每个繁忙用户都能够选择将不同的计算任务卸载到 UAV 或空闲用户上。为了实现服务的均衡和高效利用，本文规定 UAV 的飞行周期为 $T=N\partial$，整个通信周期 T 等步长的划分为 N 个时隙。在笛卡尔三维坐标系中，时隙 $n\in\{1,2,...,N\}$ 时，任务繁忙用户 i 的坐标表示为 $\boldsymbol{L}_i(n)=[x_i(n),y_i(n),0]^{\mathrm{T}}$，空闲用户 j 的坐标为 $\boldsymbol{L}_j(n)=[x_j(n),y_j(n),0]^{\mathrm{T}}$。UAV 在某时刻的三维位置由 $\boldsymbol{u}(n)=[x(n),y(n),H(n)]^{\mathrm{T}}$ 给出。用户在时隙中能够自由选择任务卸载的目标。

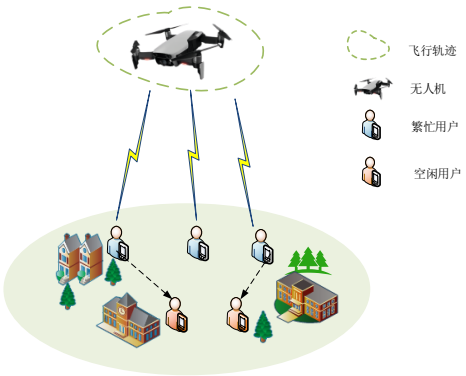

图 1 系统模型

Fig. 1 System model

UAV 的飞行速度 $\boldsymbol{v}(n)$ 和加速度 $\boldsymbol{a}(n)$ 在不同时隙之间的变换应满足以下约束

$$\boldsymbol{u}(n+1)=\boldsymbol{u}(n)+\boldsymbol{v}(n)\partial+\frac{1}{2}\boldsymbol{a}(n)\partial^2 \tag{1}$$

$$\|\boldsymbol{a}(n)\|=\frac{\|\boldsymbol{v}(n+1)-\boldsymbol{v}(n)\|}{\partial} \tag{2}$$

其中，$\boldsymbol{v}(n)\in[0,v_{\max}]$ 表示飞行速度，$\boldsymbol{a}(n)$ 表示加速度。

2.2 通信模型

为了有效避免不同卸载用户之间的严重干扰，本文引入正交频分多址技术。假设网络中所有传输链路均为视距链路，在时隙 n 内任务繁忙用户 i 和空闲用户 j 之间的信道增益为 $h_{i,j}(n)=\beta_0 d_{i,j}^{-2}(n)$，任务繁忙用户 i 和 UAV 之间的信道增益为 $h_{i,u}(n)=\beta_0 d_{i,u}^{-2}(n)$。其中 $d_{i,j}(n)$ 为任务繁忙用户和空闲用户之间的传输距离，$d_{i,u}(n)$ 为任务繁忙用户与 UAV 之间的传输距离，β_0 表示单位距离 $d=1\,m$ 的路径损耗因子。根据香农公式 D2D 链路从任务繁忙用户 i 到空闲用户 j 的信道传输速率为：

$$r_{i,j}(n)=B\log_2\left(1+\frac{p_d(n)h_{i,j}(n)}{\sigma_0^2}\right) \tag{3}$$

其中，$p_d(n)$ 为用户 i 卸载到空闲用户时的发射功率，σ_0 为高斯白噪声功率，B 为网络总带宽。

相应的任务繁忙用户 i 到UAV的信道传输速率可表示为

$$r_{i,u}(n)=B\log_2\left(1+\frac{p_{\text{uav}}(n)h_{i,u}(n)}{\sigma_0^2}\right) \tag{4}$$

其中，$p_{\text{uav}}(n)$ 为用户 i 卸载到 UAV 时的发射功率。

2.3 计算模型

如图 1 所示，本文考虑了三种计算模式，包括本地执行、D2D 执行和 UAV 执行。对于系统中的各个用户，计算任务的执行与传输是可以同时进行的，即在周期 T 内的任务可以同时传输到空闲用户和 UAV 上以完成执行。在每个时隙 n 内，用户 i 随机产生的任务大小记为 $D_i(n)$，用户处理单位比特任务所需的 CPU 周期记为 $C(n)$。对每个时隙的用户采用部分卸载策略。定义 $\{\varepsilon_1(n),\varepsilon_2(n),\varepsilon_3(n)\}\in(0,1)$ 为任务卸载集，其中 $\varepsilon_1(n)$ 为用户 i 在时隙 n 卸载到 UAV 上的计算任务比例，$\varepsilon_2(n)$ 表示卸载到空闲用户的任务比例，$\varepsilon_3(n)$ 表示本地处理的任务比例。鉴于执行结果的数据量相对较小，本文在此忽略了与结果返回相关的能耗和传输时间。

(1) 用户在本地执行时，生成的任务由本地 CPU 执行，在时隙 n 内用户 i 本地所拥有的计算资源为 $f_i(n)$。用户 i 的本地执行时延可表示为：$t_{\text{local},i}=\varepsilon_3(n)D_i(n)C(n)/f_i(n)$。用户 i 的计算能耗为 $E_{\text{local},i}(n)=\kappa f_i^2(n)\varepsilon_3(n)D_i(n)C(n)$。

(2) 由于 UAV 的计算能力通常强大，远远超过用户，因此可以在一个时隙时间内完全完成从用户 i 卸载到 UAV 的计算任务。需要注意的是，由于 UAV 通常使用电池供电，因此在任务执行过程中 UAV 的飞行能耗也需要考虑。UAV 在第 n 个时隙的飞行能耗计算为 $E_{\text{fly}}(n)=p_{\text{fly}}(n)\partial$，其中 $p_{\text{fly}}(n)$ 表示 UAV 飞行时的推进功率，建模表示为：

$$p_{\text{fly}}(n)=\frac{1}{2}d_0\rho g_0 A_0\|\boldsymbol{v}(n)\|^3+P_a\left(1+\frac{3\|\boldsymbol{v}(n)\|^2}{U_{\text{tip}}^2}\right)+P_b\left(\sqrt{1+\frac{\|\boldsymbol{v}(n)\|^4}{4v_f^2}}-\frac{\|\boldsymbol{v}(n)\|^2}{2v_f^2}\right)^{\frac{1}{2}} \tag{5}$$

其中，P_a 为 UAV 叶片功率，P_b 为悬停时的诱导功率，v_f 为旋翼平均速度，ρ 为空气密度。U_{tip} 为叶尖速度，d_0 为机身阻力比，A_0 为旋翼面积，g_0 为旋翼固体度。

UAV 服务器的处理时延可以分为两部分。一部分是传输时延，可以表示为 $t_{\text{off},i}(n)=\varepsilon_1(n)D_i(n)/r_{i,u}(n)$，另一部分是由服务器计算产生的时延，可以表示为 $t_{\text{uav},i}(n)=\varepsilon_1(n)D_i(n)C(n)/f_{\text{uav}}(n)$，其中 $f_{\text{uav}}(n)$ 表示 UAV 在时隙 n 时处理用户卸载任务所需的计算资源。相应地，将计算任务卸载到时隙 n 中的服务器所消耗的能量也可以分为两部分，包括传输的能量，另一部分用于计算。传输过程中产生的能耗为 $E_{\text{off},i}(n)=p_{\text{uav}}(n)t_{\text{off},i}(n)$，在 UAV 上进行计算时，计算能耗为 $E_{\text{uav},i}(n)=\kappa f_{\text{uav}}^2(n)\varepsilon_1(n)D_i(n)C(n)$。

(3) 对于 D2D 执行模式，任务繁忙用户 i 生成的任务首先通过 D2D 链路卸载到空闲用户 j。然后由具有空闲计算能力的用户 j 自行执行任务。最后，在每个时隙的末尾通过 D2D 链路将执行结果返回给任务繁忙用户 i。在时隙 n 中卸载到空闲用户 j 的任务量为 $\varepsilon_2(n)D_i(n)$，同理卸载到空闲用户上的处理也可以分成两部分，传输时延可表示为 $t_{\text{off},ij}(n)=\varepsilon_2(n)D_i(n)/r_{i,j}(n)$，空闲用户 j 计算产生的时延为 $t_j(n)=\varepsilon_2(n)D_i(n)C(n)/f_j(n)$。需要注意的是，由于任务处理结果一般数据量较小，因此本文不考虑设备接收任务处理结果需要消耗的能量。传输过程中产生的能耗为 $E_{\text{off},ij}(n)=p_d(n)t_{\text{off},ij}(n)$，计算能耗为 $E_j(n)=\kappa f_j^2(n)\varepsilon_2(n)D_i(n)C(n)$。

2.4 资源定价模型

在 UAV 协同 D2D 计算卸载的单 UAV 系统中，UAV 和空闲用户通过向任务繁忙用户出售计算资源来获取效用收益，但是他们需要消耗自身能量来处理卸载任务。因此，本文定义 UAV 的效用收益函数如下：

$$\begin{aligned}U_{i,\mathrm{uav}} &= f_{\mathrm{uav}}(n)p_r-\beta E_{\mathrm{uav},i}(n)-\beta E_{\mathrm{fly}}(n)\\ &= f_{\mathrm{uav}}(n)p_r-\beta\kappa f_{\mathrm{uav}}^2(n)\varepsilon_1(n)D_i(n)C(n)\\ &\quad -\beta p_{\mathrm{fly}}(n)\partial\end{aligned}\tag{6}$$

其中，p_r 表示 UAV 向用户提供的计算资源的价格，β 为不便因子，根据文献[20]，$\beta=1/1-\varepsilon_1(n)$，效用收益函数也随着不便因子 β 而降低，因为 UAV 电池有限或者计算资源较少而不愿意共享计算资源。定义空闲用户的效用收益为除去能耗开销后出售计算资源所获得的收益，收益函数如下：

$$\begin{aligned}U_{i,j} &= \sum_{j=1}^{J}\left(f_j(n)p_j-\beta_j E_j(n)\right)\\ &= \sum_{j=1}^{J}\left(f_j(n)p_j-\beta_j\kappa f_j^2(n)\varepsilon_2(n)D_i(n)C(n)\right)\end{aligned}\tag{7}$$

其中，p_j 表示空闲用户向任务繁忙用户提供的计算资源的价格，β_j 为空闲用户的能耗不便因子 $\beta_j=1/1-\varepsilon_2(n)$。任务繁忙用户可以使用从 UAV 和空闲用户购买的计算资源来获得效用，但它应该为计算资源付费。除此之外，用户本身也具有处理任务的计算资源。定义任务繁忙用户的效用收益函数如下：

$$\begin{aligned}U_i &= \sum_{i=1}^{I}\left(u_l f_i(n)-\beta_i\left(E_{\mathrm{local},i}(n)-E_{\mathrm{off},i}(n)-E_{\mathrm{off,ij}}(n)\right)\right)\\ &+\sum_{j=1}^{J}\left(u_{\mathrm{ji}}f_j(n)-f_j(n)p_j\right)+u_{\mathrm{ui}}f_{\mathrm{uav}}(n)-p_r f_{\mathrm{uav}}(n)\end{aligned}\tag{8}$$

其中，$u_{\mathrm{ui}}=f_{\mathrm{uav}}^{\max}/f_i^{\max}$，$u_{\mathrm{ji}}=f_j^{\max}/f_i^{\max}$，$u_l=f_i^{\max}/[(f_i^{\max}+f_{\mathrm{uav}}^{\max}+f_j^{\max})/3]$ 分别为 UAV，本地以及空闲用户处理的激励因子，β_i 为任务繁忙用户处理本地任务的能耗不便因子 $\beta_i=1/1-\varepsilon_3(n)$。任务繁忙的用户可以从资源丰富的 UAV 和空闲用户中获得更多的利润。

3 优化问题描述

本文通过联合优化 $\varepsilon=\{\varepsilon_1(n),\varepsilon_2(n),\varepsilon_3(n),\forall n\in N\}$，UAV 计算资源 $f_{\mathrm{uav}}=\{f_{\mathrm{uav}}(n),\forall n\in N\}$，空闲用户计算资源 $f_j=\{f_j(n),\forall n\in N,j\in J\}$，任务繁忙用户计算资源 $f_i=\{f_i(n),\forall n\in N,i\in I\}$，UAV 飞行轨迹 $\boldsymbol{u}=\{\boldsymbol{u}(n),\forall n\in N\}$，UAV 计算资源单价 $p_r=\{p_r(n),\forall n\in N\}$ 以及空闲用户计算资源单价 $p_j=\{p_j(n),\forall n\in N,j\in J\}$，以最大化整个周期 T 内 UAV，空闲用户以及任务繁忙用户的加权总收益。具体来说，优化问题可以表述为：

$$\left.\begin{aligned}&\max_{p_r,p_j,\varepsilon_1,\varepsilon_2,\varepsilon_3,\omega_1,\omega_2,\omega_3}\ \sum_{n=1}^{N}\left(\omega_1U_{i,\mathrm{uav}}+\omega_2U_{i,j}+\omega_3U_i\right)\\ &s.t.\ \mathrm{C1}: x(n)\in[0,X],y(n)\in[0,Y]\\ &\mathrm{C2}: H_{\min}(n)\le H(n)\le H_{\max}(n),\\ &\mathrm{C3}: \|\boldsymbol{u}(n)-\boldsymbol{u}(n-1)\|_2\le v_{\max}\tau\\ &\mathrm{C4}: \boldsymbol{u}(1)=\boldsymbol{u}(n)\\ &\mathrm{C5}: \varepsilon_1(n)\in(0,1),\varepsilon_2(n)\in(0,1),\varepsilon_3(n)\in(0,1)\\ &\mathrm{C6}: \varepsilon_1(n)+\varepsilon_2(n)+\varepsilon_3(n)=1\\ &\mathrm{C7}: \omega_1+\omega_2+\omega_3=1\\ &\mathrm{C8}: 0\le p_{\mathrm{uav}}(n)\le p_{\mathrm{uav}}^{\max},\forall n\in N\\ &\mathrm{C9}: 0\le p_d(n)\le p_d^{\max},\forall n\in N,j\in J\\ &\mathrm{C10}: 0\le f_i(n)\le f_i^{\max},\forall n\in N,i\in I\\ &\mathrm{C11}: 0\le f_j(n)\le f_j^{\max},\forall n\in N,j\in J\\ &\mathrm{C12}: 0\le f_{\mathrm{uav}}(n)\le f_{\mathrm{uav}}^{\max},n\in N\\ &\mathrm{C13}: p_r^{\min}\le p_r\le p_r^{\max}\\ &\mathrm{C14}: p_j^{\min}\le p_j\le p_j^{\max},\forall j\in J\end{aligned}\right\}\tag{9}$$

其中，$\omega_1,\omega_2,\omega_3$ 表示 UAV，空闲用户以及任务繁忙用户三部分收益之间的权重因子，反应这三部分收益对总收益的影响程度。约束条件 C1 和 C2 为 UAV 在给定区域移动，约束条件 C3~C4 为 UAV 的飞行轨迹，约束条件 C5 和 C6 为任务繁忙用户的卸载比例，约束条件 C7 表示 UAV，空闲用户以及任务繁忙用户收益的权重约束，三者之和为 1，约束条件 C8~C9 限制了用户的发射功率，约束条件 C10~C12 为空闲用户，任务繁忙用户以及 UAV 处理任务所能提供的计算资源不能超过设备限制，C13 和 C14 为 UAV 和空闲用户资源单价的限制。上述函数是一个非凸问题，解决起来复杂程度较高。

4 优化问题求解

由于问题(10)是一个包含连续变量和离散变量的混合整数非线性规划问题，因此利用传统的凸优化方法难以求解。因此，本文提出了一种基于 PPO 的基于 UAV-D2D 协作边缘计算网络训练框架来解决收益最大化问题。传统的优化算法如穷举搜索、分支定价法和线性规划等，在处理优化问题时可能受到维度灾难的影响，导致计算成本高昂、搜索空间巨大等问题。而传统的智能算法如遗传算法、粒子群算法和退火模拟算法等，尽管适用于解决复杂的优化问题，但存在陷入局部最优的问题。因此为了能得到优化问题的最优解，本文提出了近端策略优化算法。PPO 算法通过限制策略更新的幅度并在此基础上进行多次迭代优化，以平衡探索和利用，

稳健地提升长期奖励的期望值，从而逼近最优解。首先系统性地阐述了强化学习框架中的马尔可夫决策过程的基本组成要素，其次介绍了 PPO 算法的详尽流程。

4.1 PPO 算法

PPO 算法采用动作-评价（Actor-Critic, AC）结构，其中涉及三个参数网络，新、旧动作网络的参数分别对应 θ^* 和 θ'，评价网络参数为 ζ。新的 Actor 网络用于基于当前时间步的智能体状态生成下一个动作策略。Critic 网络则用于评估当前状态的价值。由于在更新过程中新 Actor 网络会进行多次迭代，旧 Actor 网络的引入旨在确保更新前采样的数据同样适用于更新策略，从而保障网络的收敛性。

在训练的过程中，智能体通过与环境的交互不断获得经验信息。随着每次与环境的互动，智能体从经验缓存中提取最新收集的一批经验，并将这些信息用于更新其策略。每次进行参数更新时，Actor 网络和 Critic 网络分别以策略损失函数和状态-价值损失函数为目标进行优化。具体来说，Actor 网络的损失函数定义如下

$$I^a(\theta^*)=\mathbb{E}\left\{\min\left[\frac{\pi_{\theta^*}(a_n\mid s_n)}{\pi_{\theta'}(a_n\mid s_n)}\hat{B}(s_n),\ \mathrm{clip}\left(\frac{\pi_{\theta^*}(a_n\mid s_n)}{\pi_{\theta'}(a_n\mid s_n)},1-\ell,1+\ell\right)\hat{B}(s_n)\right]\right\} \tag{10}$$

其中 $\mathbb{E}[\cdot]$ 表示期望值，π_{θ^*} 和 $\pi_{\theta'}$ 分别表示新、旧策略函数，ℓ 为截断参数，$\hat{B}(s_n)$ 为优势函数，用来评估动作 a_n 的性能。

为确保策略更新的鲁棒性，本文采用了广义优势估计（General Advantage Estimation, GAE）。GAE 因子 η 的取值范围被限制在 0 到 1 之间，其计算方法可表示为

$$\hat{B}(s_n)=\sum_{i=0}^{\infty}(\gamma\eta)^l\left(r(n)+\gamma V(s_{n+1})-V(s_n)\right) \tag{11}$$

其中 $V^{\zeta}(s_n)=\mathbb{E}_{s_n,a_n}\left[\sum_{i=0}^{\infty}\gamma^l R(a_{n+l}\mid s_{n+l})\right]$ 为状态价值函数。Critic 网络 ζ 的目标函数可表示为

$$I^c(\zeta)=\left[V^{\zeta}(s_{n+1})-V^{\zeta}(s_n)\right]^2 \tag{12}$$

4.2 基于 PPO 的收益最大化算法

在本文模型中存在多个用户和一个 UAV，且 UAV 不需要具备先验环境信息，而是通过环境状态的实时观测来获取因果信息。这意味着状态转移是未知的，因此本文可以将其描述为一个无模型、无转移概率的马尔可夫决策过程。在 MDP 中，智能体与动态环境交互，不断调整其策略以最大化累积奖励。因此，在每个时间步内，状态、行动和奖励定义如下:

(1) 状态空间：$s_n=\{L_i(n),D_i(n),\boldsymbol{v}(n),C(n)\}$。其中，$L_i(n)$ 表示需要向 UAV 购买资源的终端的位置，$D_i(n)$ 表示用户 i 需要处理的任务数据量，$\boldsymbol{v}(n)$ 表示单 UAV 的速度，$C(n)$ 表示用户处理单位比特任务所需的 CPU 周期。

(2) 动作空间：智能体需要根据当前状态，选择一组动作，以完成任务调度和资源分配。这组动作可以包括任务卸载比例、UAV 和空闲用户资源单价，空闲用户、任务繁忙用户和 UAV 收益权重比，无人机、空闲用户和任务繁忙用户计算资源以及 UAV 位置信息。因此，动作空间可以表示为 $a(n)=\{[\varepsilon_1(n),\varepsilon_2(n),\varepsilon_3(n)],p_r,p_j,[\omega_1,\omega_2,\omega_3],f_{uav}(n),f_j(n),f_i(n),\boldsymbol{u}(n)\}$。其中 $[\varepsilon_1(n),\varepsilon_2(n),\varepsilon_3(n)]$ 为时隙 n 时卸载到无人机，空闲用户和本地处理的卸载比例，p_r 为 UAV 向任务繁忙用户出售计算资源的单价，p_j 为空闲用户向任务繁忙用户可出售计算资源的单价，$[\omega_1,\omega_2,\omega_3]$ 为无人机、空闲用户和任务繁忙用户收益的权重，$f_{uav}(n)$ 为无人机在时隙 n 的计算资源，$f_j(n)$ 为空闲用户 j 在时隙 n 的计算资源，$f_i(n)$ 为任务繁忙用户 i 在时隙 n 的计算资源，$\boldsymbol{u}(n)$ 为时隙 n 时的 UAV 位置坐标。

(3) 奖励函数：$r(n)=\omega_1U_{i,\mathrm{uav}}+\omega_2U_{i,j}+\omega_3U_i$。为时隙 n 无人机、空闲用户和任务繁忙用户的收益权重之和。为了长期实现本文的优化目标，并考虑约束条件的满足程度，本文设计了与系统收益相同的奖励函数。约束条件 C6 为用户的卸载比例约束，C7 为收益的权重约束，为了找到一个最优的卸载比例和收益权重，对卸载比例和权重进行定义，在处理卸载比例和权重的时候，考虑到其和可能会超过 1 的情况，为了确保数据的准确性和合理性，采取单位化的方法。对卸载比例和相应权重进行标准化处理，使其在一个单位范围内，即 0 到 1 之间。

基于上述定义说明，本文所提基于 PPO 的系统收益最大化算法具体流程如图 2 所示，具体流程详见表 1。

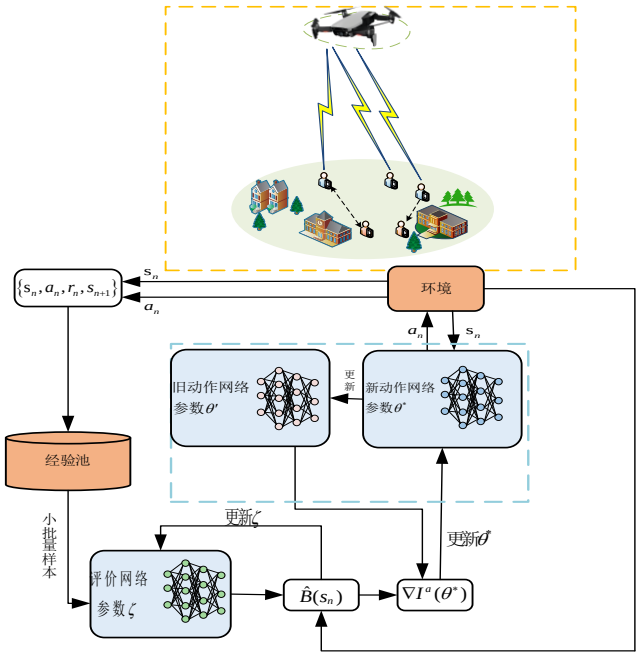

图 2 基于 PPO 的 DRL 训练框架

Fig. 2 PPO-based DRL training framework

在表 1 所描述的算法中，系统初始状态的观测值经过处理，被传递至一个深度神经网络结构。对于该神经网络的第 k 层，其计算复杂度可表示为 $O\left(N_{k-1}N_k+N_kN_{k+1}\right)$，其中 N_k 为第 k 层隐藏层神经元数量。值得特别强调的是，相对于隐藏层而言，输入层和输出层的乘法计算次数相对较少，因此在整体复杂性的计算中，可以忽略它们的影响。总体而言，K 层深度神经网络的总复杂性可以表述为 $O\left(\sum_{k=2}^{K-1}N_{k-1}N_k+N_kN_{k+1}\right)$。因此，该算法的总复杂度可以表示为 $O\left(M\left(ep\left(\sum_{k=2}^{K-1}N_{k-1}N_k+N_kN_{k+1}\right)\right)\right)$。

表 1 基于 PPO 的收益最大化算法

Table 1 Revenue maximization algorithm based on PPO

输入：学习率 χ，GAE 因子 λ，截断参数 ℓ，评价网络参数 ζ，最大训练集 M，每一个训练集长度 ep

步骤 1 初始化评价网络参数 ζ，动作网络参数 θ^*

步骤 2 for $m=1:M$

初始化：$[x_i(n),y_i(n),0]$，$[x_j(n),y_j(n),0]$，$D_i(n)$，UAV 高度 $H(n)$

for $n=1:ep$

智能体从环境中获取状态 s_n

根据当前状态 s_n 选择动作 a_n

执行动作 a_n，接收奖励 r_n，并转移到下一个状态 s_{n+1}；

将数据 (s_n,a_n,r_n,s_{n+1}) 存储在经验回放缓冲区中；

end for

for $n=1:K$

计算优势值 $\hat{B}(s_n)$

end for

对评价网络参数 ζ 进行更新

对动作网络参数 θ^* 进行更新

更新 $\theta'\leftarrow\theta^*$

清空缓冲区

end for

步骤 3 输出训练后的动作网络和评价网络

5 仿真结果与分析

本节将使用 PyTorch 框架进行仿真，并评估所提出方案的性能。以(0,0,0)建立三维坐标系，25 个用户设备和 1 个 UAV，任务的数据量 $D_i(n)\in[0.1,0.6]$ Mbits，根据有无任务量产生 25 个用户设备分为任务繁忙用户和空闲用户，UAV 的飞行时隙数 N 为 20，飞行高度为 $[0,200]$ m，时间 T 为 10 s，系统宽度 B 为 100 kHz，噪声功率 σ_0^2 为-120 dBm，芯片结构对 CPU 处理的影响因子为 10^{-27}，最大发射功率 $p_i^{\max}$ 和 $p_d^{\max}$ 为 24 dBm，任务繁忙用户的最大计算能力 $f_i^{\max}$ 为 2 GHz，空闲用户的最大计算能力 $f_j^{\max}$ 为 3 GHz，UAV 的最大计算能力 $f_{\text{uav}}^{\max}$ 为 5 GHz。PPO 训练参数如表 2 所示。

表 2 PPO 训练参数

Table 2 PPO training parameter

参数	值
最大回合数	6k
折扣因子 γ	0.6
隐藏层大小	[64,128]
截断参数 ℓ	0.1
学习率 χ	0.005
GAE 因子 η	0.9

为验证所提 PPO 算法的性能，本文将其与以下三种基准算法进行对比：

（1）深度确定性策略梯度（Deep Deterministic Policy Gradient, DDPG）：DDPG 是由单个 Actor 网

络和单个 Critic 网络构成的，其中 Actor 网络负责输出动作，而 Critic 网络用于估计动作值函数。引入经验回放和目标网络的概念，以提高算法的稳定性和收敛性。

（2）优势动作评论（Advantage Actor Critic, A2C）：该方法采用同策略框架，引入优势函数替代原始回报评价，以提高稳定性和学习效率。

（3）贪婪算法（Greedy Algorithm）：该方法基于现有知识，在第 n 个时隙贪婪地选择无人机轨迹以及计算和通信资源分配，使得能耗开销减小。

特别地，在处理连续动作空间方面，PPO 直接优化连续动作策略，避免了 DDPG 和 A2C 在处理连续空间时可能遇到的复杂性和稳定性挑战。尤其是 PPO 通过“截断重要性采样比率”限制策略更新步长，减少训练不稳定性。

图 3 显示了 PPO 算法与其他算法在收敛性上的比较。随着迭代次数的增加，三种方案的奖励值逐渐趋于稳定。PPO 算法在不到 1k 步时就趋于收敛，A2C 算法在约 1k 步时趋于收敛，而 DDPG 算法在大约 2k 步左右才开始趋于收敛。PPO 算法不仅收敛速度较快，而且总体上获得了更高的奖励值。这进一步说明了 PPO 算法在解决本文问题时具有更强的收敛性，更为适用。

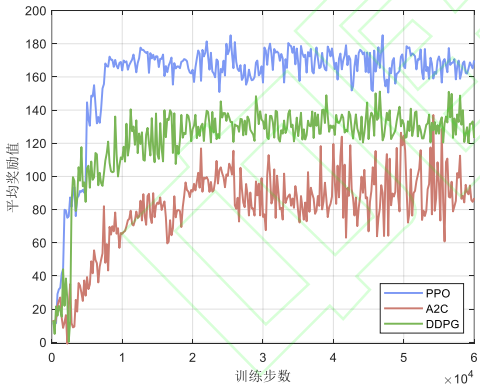

图 3 不同算法的奖励收敛性对比图

Fig. 3 Comparison of convergence of rewards for different algorithms

图 4 显示了任务繁忙用户收益与用户数量之间的关系。在总用户数量为 25 的情况下，可以观察到随着用户数量的增加，任务繁忙用户的收益也呈现逐渐增加的趋势。随着总用户数量的增加，用户的任务卸载需求也随之增加。此时，空闲用户和 UAV 将提供更多的计算资源，以满足处理任务繁忙用户任务的需求。这种增加的计算资源将带来激励回报的提升，随着资源增加，任务繁忙用户能够获得的收益也相应增加。

图 5 展示了总收益与用户数量之间的关系。随着用户数量的逐步增加，总收益呈现稳定的增长趋势。随着总用户数量的提升，任务需求也相应增加，为空闲用户和 UAV 提供了更多销售资源的机会。随着销售资源的不断增加，空闲用户和 UAV 的收益也随之增加。根据图 4 的数据，可以看出随着用户数量的增加，任务繁忙用户的收益也在持续上升，这进一步推动了总收益的增长。

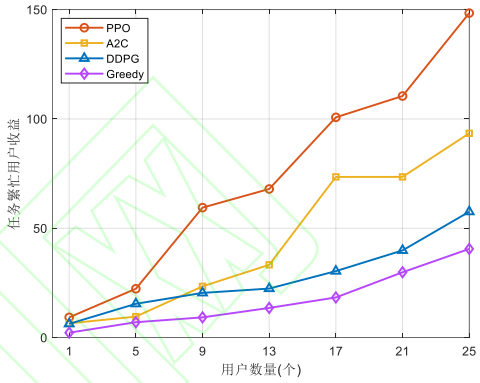

图 4 任务繁忙用户收益与用户数量的关系

Fig. 4 Relationship between revenue of busy users and number of users

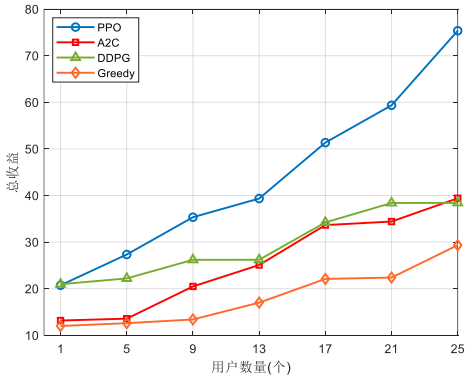

图 5 总收益与用户数量的关系

Fig. 5 Relationship between total revenue and number of users

图 6 展示了 UAV 的三维飞行轨迹图。在此场景中，UAV 的最大通信高度限定为 200 米，总用户数量为 25。从图 6 的轨迹图中可以观察到 PPO 算法相较于 A2C 算法能够覆盖更多的用户，从而更有效地为用户提供服务。在用户密集区域，PPO 算法的无人机轨迹呈现出更弯曲的形状，且停留时间更长；相反，在用户稀疏的区域，轨迹更为平滑。相比之下，A2C 算法仅在用户密集区域提供服务，并且其飞行轨迹不稳定，难以兼顾用户密度较低的区

域。

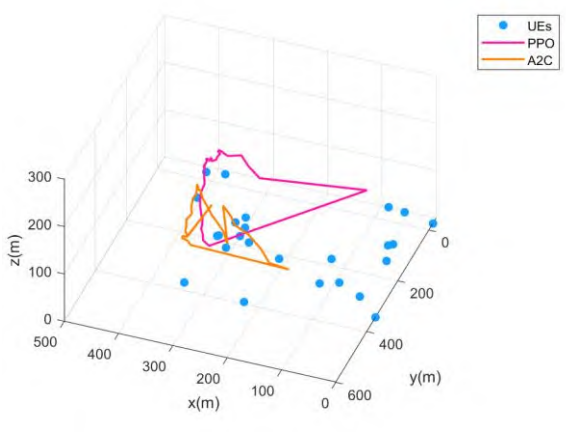

图 6 多用户 UAV 三维飞行轨迹图

Fig. 6 3D flight trajectory of UAV with multiple users

6 结束语

本文提出了一种以激励机制为基础的 UAV 协同 D2D 算力网络框架，其目的是通过联合优化任务卸载比例、计算资源限制、UAV 飞行轨迹，以及发射功率和计算资源出售单价，最大化 UAV、空闲用户和任务繁忙用户的总收益。为有效求解该问题，引入了基于 PPO 的深度强化学习算法，激励空闲用户和 UAV 参与任务卸载并分享计算资源。仿真结果验证了算法具有良好的收敛性，有效提高系统总收益。未来工作将进一步研究在多 UAV 情况下，如何通过优化卸载比例和资源单价实现系统收益最大化。

参考文献

[1] GAO M, SHEN R, SHI L, et al. Task partitioning and offloading in DNN-task enabled mobile edge computing networks[J]. IEEE Transactions on Mobile Computing, 2023, 22(4): 2435-2445.

[2] CHEN Yang, PI Dechang, DAI Chenglong, et al. Energy minimization for multi-UAVs cooperative ground access points assisted mobile edge computing[J]. ACTA ELECTONICA SINICA, 2023, 51(4): 984-992.(in Chinese)陈阳,皮德常,代成龙等.多无人机协同陆地设施辅助移动边缘计算的系统能耗最小化方法[J]. 电子学报, 2023, 51(04):984-992.

[3] LI Yiquan, YANG Chenxi, DENG Miaoxin, et al. A dynamic resource optimization scheme for MEC task offloading based on policy gradient[C]//Proceedings of 2022 IEEE 6th Information Technology and Mechatronics Engineering Conference, Chongqing, China: IEEE Press, 2022: 342-345.

[4] FANG Tao, YUAN Feng, AO Liang, et al. Joint task offloading, D2D pairing, and resource allocation in device-enhanced MEC: A potential game approach[J]. IEEE Internet of Things Journal, 2022, 9(5): 3226-3237.

[5] LU Weidang, DING Yu, GAO Yuan, et al. Secure NOMA-based UAV-MEC network towards a flying eavesdropper[J]. IEEE Transactions on Communications, 2022, 70(5): 3364-3376.

[6] LIU Zhaoyuan, FAN Jingyi, GENG Suiyan, et al. Joint optimization of task offloading and computing resource Allocation in MEC-D2D Network[C]//Proceedings of 2022 IEEE 5th International Conference on Computer and Communication Engineering Technology. Beijing, China: IEEE Press, 2022: 256-260.

[7] MOGHADDASI K , RAJABI S. Double deep Q-learning networks for energy-efficient IoT task offloading in D2D MEC environments[C]//Proceedings of 2023 7th International Conference on Internet of Things and Applications. Isfahan, Iran: IEEE Press, 2023: 1-6.

[8] DAI Xingxia, XIAO Zhu, JIANG Hongbo, et al. Task co-offloading for D2D-assisted mobile edge computing in industrial internet of things[J]. IEEE Transactions on Industrial Informatics, 2023, 19(1): 480-490.

[9] GUO Hongzhi, LIU Jiajia. UAV-enhanced intelligent offloading for internet of things at the edge[J]. IEEE Transactions on Industrial Informatics, 2020, 16(4): 2737-2746.

[10] CHEN Xianfu, CHEN Tao, ZHAO Zhifeng, et al. Resource awareness in unmanned aerial vehicle-assisted mobile-edge computing systems[C]//Proceedings of 2020 IEEE 91st Vehicular Technology Conference. Antwerp, Belgium: IEEE Press, 2020: 1-6.

[11] LI Yiyang, FANG Yuan, QIU Ling. Joint computation offloading and communication design for secure UAV-enabled MEC systems[C]//Proceedings of 2021 IEEE Wireless Communications and Networking Conference. Nanjing, China: IEEE Press, 2021: 1-6.

[12] HU Xiaoyan, WONG K K, YANG Kun, et al.

UAV-assisted relaying and edge computing: Scheduling and trajectory optimization[J]. IEEE Transactions on Wireless Communications, 2019, 18(10): 4738-4752.

[13] ZHANG Liang, ANSARI N. Latency-aware IoT service provisioning in UAV-aided mobile-edge computing Networks[J]. IEEE Internet of Things Journal, 2020, 7(10): 10573-10580.

[14] CHEN Jiafa, ZHAO Yisheng, XU Zhimeng, et al. Resource allocation strategy for D2D-assisted edge computing system with hybrid energy harvesting[J]. IEEE Access, 2020, 8: 192643-192658.

[15] SONG Qinglin, QU Long. UAV-D2D assisted latency minimization and load balancing in mobile edge computing with deep reinforcement learning[C]// Proceedings of International Conference on Green, Pervasive, and Cloud Computing. Singapore: Springer, 2023: 108-122.

[16] XU Yongjun, LIU Zijian, HUANG Chongwen, et al. Robust resource allocation algorithm for energy-harvesting-based D2D communication underlaying UAV-assisted networks[J]. IEEE Internet of Things Journal, 2021, 8(23): 17161-17171.

[17] NGUYEN M N, NGUYEN L D, DUONG T Q, et al. Real-time optimal resource allocation for embedded UAV communication systems[J]. IEEE Wireless Communications Letters, 2019, 8(1): 225-228.

[18] HAN Yuelin, ZHU Qi. Multi user D2D computation offloading and resource allocation algorithm[J/OL]. Journal of Signal Processing, 2024:1-19.(in Chinese)韩跃林,朱琦.多用户 D2D 计算卸载与资源分配算法[J/OL]. 信号处理, 2024:1-19.

[19] PU Xumin, LEI Tiantian, WEN Wanli, et al. Incentive mechanism and resource allocation for collaborative task offloading in energy-efficient mobile edge computing[J]. IEEE Transactions on Vehicular Technology, 2023, 72(10): 13775-13780.

[20] SUN Weijie, ZHANG Haixia, WANG Leiyu, et al. Profit maximization task offloading mechanism with D2D collaboration in MEC networks[C] // Proceedings of 2019 11th International Conference on Wireless Communications and Signal Processing, Xi'an, China: IEEE Press, 2019: 1-6.