计算机科学
COMPUTER SCIENCE

空中智能反射面辅助边缘计算中基于 PPO 的任务卸载方案

彭云聪，秦小林，张力戈，顾勇翔

引用本文

谢万城，李斌，代玥玥. 空中智能反射面辅助边缘计算中基于 PPO 的任务卸载方案[J]. 计算机科学, 2022, 49(6): 3-11.

XIE Wan-cheng, LI Bin, DAI Yue-yue. PPO Based Task Offloading Scheme in Aerial Reconfigurable Intelligent Surface-assisted Edge Computing[J]. Computer Science, 2022, 49(6): 3-11.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220100249

空中智能反射面辅助边缘计算中基于 PPO 的任务卸载方案

谢万城[1]　李　斌[1,2]　代玥玥[3]
1 南京信息工程大学计算机与软件学院　南京 210044
2 南京邮电大学宽带无线通信与传感网技术教育部重点实验室　南京 210003
3 华中科技大学 6G 研究中心与网络空间安全学院　武汉 430074
(zuoyeyiwancheng@gmail.com)

摘　要　针对 6G 时代“智慧物联网”边缘计算系统中障碍物阻挡对任务卸载性能的影响,提出了一种无人机搭载智能反射面(Reconfigurable Intelligent Surfaces,RIS)辅助的计算任务部分卸载方案。首先,在满足用户传输功率、无人机高度、任务卸载比例限制的条件下,通过联合优化时隙分配、任务卸载比例、无人机高度、RIS 相移和用户传输功率,建立用户总能耗最小化问题;其次,将该非凸优化问题分解为 4 个子问题,使用深度强化学习中的近端策略优化(Proximal Policy Optimization,PPO)方法确定时隙分配策略;最后,将每个训练时间步作为一次求解,基于交替迭代方法和连续凸逼近方法得到问题的优化解。仿真结果表明,基于 PPO 的算法训练速度较快其用户总能耗比采用全部卸载方案的能耗减少了约 23%,比采用无人机高度固定方案的能耗减少了约 5.3%。

关键词:移动边缘计算;智能反射面;无人机;深度强化学习;任务卸载
中图法分类号　TN929.5

PPO Based Task Offloading Scheme in Aerial Reconfigurable Intelligent Surface-assisted Edge Computing

XIE Wan-cheng[1],LI Bin[1,2] and DAI Yue-yue[3]
1 School of Computer and Software,Nanjing University of Information Science and Technology,Nanjing 210044,China
2 Key Lab of Broadband Wireless Communication and Sensor Network Technology (Nanjing University of Posts and Telecommunications),Ministry of Education,Nanjing 210003,China
3 Research Center of 6G Mobile Communications and School of Cyber Science and Engineering,Huazhong University of Science and Technology,Wuhan 430074,China

Abstract　In order to compensate the performance loss caused by obstacle blocking in mobile edge computing (MEC) system in 6G-enabled “intelligent Internet of Things”,this paper proposes a partial task offloading scheme supported by aerial reconfigurable intelligent surface (RIS). Firstly,we investigate the joint design of the RIS phase shift vector,the proportion of offloading task,time slot allocation,the transmit power of users and the position of UAV,formulating a non-convex problem for minimization of the total energy consumption of users. Then,the original non-convex problem is decomposed into four subproblems,and the proximal policy optimization (PPO) method in deep reinforcement learning (DRL) is utilized to provide time slot allocation. The alternative optimization (AO) is leveraged to decouple the original problem into four subproblems,including the RIS phase shift design,the convex optimization of transmit power and offloading task amount,and the UAV altitude optimization. Simulation results show that the proposed PPO model can be trained quickly,the total energy consumption of users can be reduced by about 23% and 5.3%,compared with the fully-offload strategy and fixed-UAV-height strategy,respectively.

Keywords　Mobile edge computing,Reconfigurable intelligent surface,Unmanned aerial vehicle,Deep reinforcement learning,Task offloading

到稿日期:2022-01-26　返修日期:2022-03-10
基金项目:国家自然科学基金(62101277);江苏省自然科学基金(BK20200822);江苏省高校自然科学基金面上项目(20KJB510036);南京邮电大学宽带无线通信与传感网技术教育部重点实验室开放研究基金资助课题(JZNY202103)
This work was supported by the National Natural Science Foundation of China(62101277),National Natural Science Foundation of Jiangsu Province(BK20200822),Natural Science Foundation of Jiangsu Higher Education Institutions of China(20KJB510036),Open Research Fund of Key Lab of Broadband Wireless Communication and Sensor Network Technology(Nanjing University of Posts and Telecommunications) and Ministry of Education (JZNY202103).
通信作者:李斌(bin.li@nuist.edu.cn)

1 引言

随着人工智能的不断发展，物联网和人工智能相互融合，6G 时代“智慧物联网”的蓝图正逐渐展开。通过分析、处理海量数据，智慧物联网技术可以在各种场景下对物理世界进行监测和控制，如自动驾驶、智能交通、工业自动化、应急管理等[1]。这种新的范式促成了物与人之间的一种全新的互动模式，将极大地重塑人们的日常生活。未来，6G 时代的无线设备类型将更加异质化，应用形态更加丰富化，网络数据更加巨量化。面对日益增长的用户体验需求，若将所有的原始数据都卸载到云端，势必会造成严重的网络流量拥塞和过高的服务延迟。移动边缘计算(Mobile Edge Computing，MEC)范式提出将传统云计算的功能扩展到网络边缘，实现业务的就近接入，可以有效缓解核心网的计算压力，保证低时延互联互通[2]。

尽管传统的 MEC 系统架构能在多数场景下提供服务，但地面固定基础设施易受自然灾害的影响甚至被摧毁，导致用户很难直接与邻近的 MEC 服务器通信。为有效提升计算服务质量，将无人机(Unmanned Aerial Vehicles，UAVs)部署为空中移动中继或计算节点是地面 MEC 系统的一种候补方案[3-5]。无人机具有高移动性和可控性，能够灵活地按需部署以缓解突发状况下的计算压力。因此，近年来 UAV 辅助 MEC 系统受到了广泛关注。文献[3]研究了一种无人机中继辅助的 MEC 网络，该网络联合优化通信带宽、无人机传输功率、计算资源、任务划分和无人机的位置部署，以最小化任务的完成时间。文献[4]将无人机同时作为中继和计算节点，考虑了 3 种不同的卸载策略，并联合优化无人机轨迹、功率分配与时隙调度，以最小化总能耗。文献[5]通过联合优化卸载和调度策略，尽量减少传输能耗与悬停能耗的加权和。文献[6]考虑了多无人机场景下轨迹和用户关联以及时隙长度，以最大化所有用户的最小速率。

处于蜂窝边缘或者被障碍物遮挡的用户卸载速率较低，导致计算卸载时延较长或能量消耗较大。低卸载速率限制了 MEC 网络的整体计算速度，使得 MEC 任务卸载的优势未得到充分发挥。因此，通过增强计算卸载链路的信道增益来提升 MEC 的性能势在必行。智能反射面(RIS)作为一种能够主动控制无线传播环境的革新技术，可以通过软件编程调整入射信道的幅值和相位增强信号传播链路质量，显著提高频谱空间效率[7]。一方面，RIS 通过重新配置无线传播环境可以提供优质的卸载链路，从而提高 MEC 系统的实时服务性能；另一方面，联合优化 RIS 和 MEC 系统的通信和计算资源可以实现 RIS 的收益。RIS 协同 MEC 是一种双赢组合，如何统筹 RIS 和 MEC 之间的相互协同问题至关重要。文献[8]使用非正交多址接入协议，通过联合优化 RIS 无源波束成形、传输数据大小、传输速率、功率控制、传输时间和解码顺序，使所有用户的总能耗最小化。基于深度强化学习的方法，文献[9]使用双深度 Q 网络(Double Deep Q Network，DDQN)考虑无线传输的时隙分配，以提升 MEC 网络的计算能力。文献[10]考虑了优化传输功率与 RIS 相移，以最大化 MEC 场景中的机器学习性能。

因 RIS 的无源组件和简单的硬件结构，其可以安装到无人机上构成空中 RIS，或放置于建筑物表面，接收来自基站/用户的发射信号，并反射至无线接入点[11]。结合无人机的高移动性和 RIS 的优质链路质量，在不增加无人机侧复杂性的情况下可进一步提高通信性能。关于空中 RIS 的相关研究也取得了一定的成果，例如，文献[12-13]建立了一种无人机与 RIS 相结合的网络框架，该框架将 RIS 置于建筑物表面以提升信道质量，联合优化无人机轨迹、RIS 相移和任务卸载策略以最小化能耗或最大化能效；针对空中 RIS 作为中继节点的场景，文献[14-15]使用凸优化方法建立了能效最大化与能耗最小化模型；文献[16]应用深度强化学习中的近端策略优化(Proximal Policy Optimization，PPO)框架优化无人机的高度、通信进度和 RIS 相移矩阵，最小化信息年龄。以上研究工作仅考虑了空中 RIS 辅助的通信资源分配问题，未考虑任务计算性能。为了更好地满足远端计算服务需求，空中 RIS 协同下的 MEC 任务卸载成为了一个值得研究的问题。

如何合理地进行用户和边缘服务器的协作以及实现空中 RIS 与 MEC 之间的协同是一个新的挑战。基于上述研究现状，本文考虑了一种空中 RIS 辅助 MEC 的计算卸载方案，主要工作如下：

(1)针对用户电量受限且处于偏远地带的场景，本文将空中 RIS 引入 MEC 系统中，考虑 RIS 相移、传输功率、无人机高度、任务卸载量与时隙分配约束，构建了非凸、组合优化的用户能耗最小化优化问题。

(2)提出了一种基于 PPO 的深度强化学习(Deep Reinforcement Learning，DRL)方法，该方法可用于优化时隙分配，并使用交替迭代方法求解其他优化变量。首先，使用解析解计算最优 RIS 相移；其次，将任务卸载比例与传输功率优化合并为一个凸问题；然后，对于无人机高度优化这个非凸问题，使用连续凸逼近(Successive Convex Approximation，SCA)方法将非凸约束条件转化为凸约束条件；最后将以上交替迭代求解部分嵌入强化学习训练环境，提出一种最小化用户总能耗的资源分配算法。

2 系统模型

2.1 通信模型

空中 RIS 辅助 MEC 的系统模型如图 1 所示，该系统由 K 个单天线用户、一架搭载空中 RIS 的无人机和一个配备 MEC 计算服务器的单天线基站组成。其中，无人机搭载的 RIS 是由 M 个反射单元组成的均匀线性阵列。

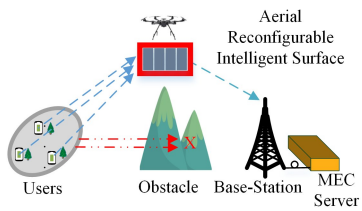

图 1 系统模型图

Fig. 1 System model

为便于表达和分析，定义用户、RIS 反射单元的集合分别为 $\forall k \in K \triangleq \{1,2,\cdots,K\}$ 和 $\forall m \in \mathcal{M} \triangleq \{1,2,\cdots,M\}$。假设用

户与基站之间有障碍物遮挡，并且相隔较远，无法直接通信，那么用户发送信号就需要经过空中 RIS 反射给基站。本系统使用时分多址接入来避免任务卸载时不同地面设备之间的干扰[17]，将周期 T 等分为 N 个时隙，定义时隙集合为 $\forall n\in\mathcal{N}\triangleq\{1,2,\cdots,N\}$。用户在分配到的时隙内进行任务卸载，同时在周期 T 内完成剩余任务的计算。令时隙 n 内，表示用户 k 任务卸载的二元决策变量为 $\alpha_k[n]$，若 $\alpha_k[n]=1$，则用户 k 在时隙 n 内进行任务卸载，否则 $\alpha_k[n]=0$。一个时隙只能分配给一个用户，故应满足：

$$\alpha_k[n]\in\{0,1\},\ \forall k\in\mathcal{K} \tag{1}$$

$$\sum_{k=1}^{K}\alpha_k[n]\leqslant 1,\ \forall n\in\mathcal{N} \tag{2}$$

基站将对无人机的高度以及 RIS 反射单元的相移进行实时控制，以优化信道质量，决策用户的传输功率、时隙内用户调度以及任务卸载比例。具体而言，无人机通过收发机接收来自基站的控制信号，并与基站交换用户状态信息，根据基站的控制信号调整各个时隙中 RIS 的反射单元相移以及自身高度，向各用户发送对应的时隙分配策略以及传输功率任务卸载比例。

假设所有用户在同一平面内，用户 k 的坐标为 $(x_k,y_k,0)$，基站高度为 H_B，则基站的坐标为 (x_B,y_B,H_B)。为简化问题，本文考虑无人机在每个时隙内进行高度的调整。设在时隙 n 内，无人机的高度为 $H_U[n]$，则无人机坐标为 $(x_U,y_U,H_U[n])$，$H_U[n]$ 需满足的约束条件如下：

$$H_{\min}\leqslant H[n]\leqslant H_{\max},\ \forall n\in\mathcal{N} \tag{3}$$

$$|H_U[n+1]-H_U[n]|\leqslant D_{\max},\ \forall n\in\mathcal{N} \tag{4}$$

$$H[1]=H_s \tag{5}$$

其中，H_s 表示无人机的初始高度，$H_{\min}$ 和 $H_{\max}$ 分别表示无人机的最小、最大飞行高度，$D_{\max}=v_{\max}\delta_t$，表示无人机每次移动的最大距离，$v_{\max}$ 表示无人机的最大飞行速度。

定义时隙 n 内，用户 k 与无人机的距离为 $d_{k,U}[n]$，无人机与基站之间的距离为 $d_{U,B}[n]$，则：

$$d_{k,U}[n]=\sqrt{(x_k-x_U)^2+(y_k-y_U)^2+(H_U[n])^2} \tag{6}$$

$$d_{U,B}[n]=\sqrt{(x_B-x_U)^2+(y_B-y_U)^2+(H_U[n]-H_B[n])^2} \tag{7}$$

因此用户 k-空中 RIS 链路、空中 RIS-基站链路之间的信道增益 $\boldsymbol{h}_{k,U}$ 和 $\boldsymbol{h}_{U,B}$ 可以表示为：

$$\boldsymbol{h}_{k,U}[n]=\hat{\boldsymbol{h}}_{k,U}[n]\Delta_{k,U}[n] \tag{8}$$

$$\boldsymbol{h}_{U,B}[n]=\hat{\boldsymbol{h}}_{U,B}[n]\Delta_{U,B}[n] \tag{9}$$

其中，$\Delta_{k,U}[n]$ 和 $\Delta_{U,B}[n]$ 分别表示第 n 个时隙内用户 k-空中 RIS 链路、空中 RIS-基站链路的路径损耗，即：

$$\Delta_{k,U}[n]=\sqrt{\gamma_0 d_{k,U}^{-2}[n]} \tag{10}$$

$$\Delta_{U,B}[n]=\sqrt{\gamma_0 d_{U,B}^{-2}[n]} \tag{11}$$

其中，γ_0 是参考距离为 1 m 时的信道功率增益。由于两条链路均为视距路径，可以用莱斯(Rician)信道建模，故 $\hat{\boldsymbol{h}}_{k,U}[n]\in\mathbb{C}^{M\times 1}$ 和 $\hat{\boldsymbol{h}}_{U,B}[n]\in\mathbb{C}^{M\times 1}$ 可表示为：

$$\hat{\boldsymbol{h}}_{k,U}[n]=\sqrt{\frac{K_1}{K_1+1}}\bar{\boldsymbol{h}}_{k,U}[n] \tag{12}$$

$$\hat{\boldsymbol{h}}_{U,B}[n]=\sqrt{\frac{K_2}{K_2+1}}\bar{\boldsymbol{h}}_{U,B}[n] \tag{13}$$

其中，K_1 和 K_2 是莱斯因子，$\bar{\boldsymbol{h}}_{k,U}[n]$ 和 $\bar{\boldsymbol{h}}_{U,B}[n]$ 可表示为：

$$\bar{\boldsymbol{h}}_{k,U}[n]=[1,\mathrm{e}^{-j\frac{2\pi}{\lambda}d\hat{\varphi}_{k,2}[n]},\cdots,\mathrm{e}^{-j\frac{2\pi}{\lambda}d(M-1)\hat{\varphi}_{k,M}[n]}]^{\mathrm{T}} \tag{14}$$

$$\bar{\boldsymbol{h}}_{U,B}[n]=[1,\mathrm{e}^{-j\frac{2\pi}{\lambda}d\hat{\omega}_2[n]},\cdots,\mathrm{e}^{-j\frac{2\pi}{\lambda}d(M-1)\hat{\omega}_M[n]}]^{\mathrm{T}} \tag{15}$$

其中，d 为反射元件间距，$\hat{\varphi}_{k,m}[n]=\dfrac{x_U-x_k}{d_{k,U}[n]}$ 和 $\hat{\omega}_m[n]=\dfrac{x_B-x_U}{d_{U,B}[n]}$ 分别表示时隙 n 内用户 k-空中 RIS 链路和空中 RIS-基站链路的信号到达角的余弦。为方便讨论，本文假设信道状态信息是已知的，令 $\varphi_{k,m}[n]=-\dfrac{2\pi}{\lambda}d(m-1)\hat{\varphi}_{k,m}[n]$，$\omega_m[n]=-\dfrac{2\pi}{\lambda}d(m-1)\hat{\omega}_m[n]$。

由式(6)－式(15)可知，在时隙 n 内，基站处接收信号的信噪比为：

$$\gamma_k[n]=\frac{p_k[n]|\boldsymbol{h}_{U,B}^H[n]\boldsymbol{\Phi}[n]\boldsymbol{h}_{k,U}[n]|^2}{\sigma^2} \tag{16}$$

其中，σ^2 为信道中的噪声功率，$p_k[n]$ 表示用户 k 在时隙 n 内的传输功率。用户-空中 RIS-基站链路的总信道增益可表示为：

$$\begin{aligned}&\boldsymbol{h}_{k,U}^H[n]\boldsymbol{\Phi}[n]\boldsymbol{h}_{U,B}[n]\\&=\frac{\gamma_0\sum_{m=1}^{M}|[\boldsymbol{h}_{U,B}]_m||[\boldsymbol{h}_{k,U}]_m|\mathrm{e}^{\mathrm{j}(\phi_m[n]+\varphi_{k,m}[n]+\omega_m[n])}}{d_{k,U}[n]d_{U,B}[n]}\end{aligned} \tag{17}$$

其中，$[\boldsymbol{h}]_m$ 表示 $\boldsymbol{h}$ 的第 m 个元素，$\boldsymbol{\Phi}[n]\triangleq\mathrm{diag}([\mathrm{e}^{\varphi_1[n]},\cdots,\mathrm{e}^{\varphi_M[n]}]^{\mathrm{T}})\in\mathbb{C}^{M\times M}$ 表示第 n 个时隙内空中 RIS 的对角相移矩阵。由式(17)可知，基站处的接收信号信噪比最大时，有：

$$\phi_m[n]=-(\varphi_{k,m}[n]+\omega_m[n]) \tag{18}$$

其中，$\alpha_k[n]=1$。根据香农公式，任务的卸载速率为：

$$R_k[n]=B\log_2\left(1+\frac{p_k[n]|\boldsymbol{h}_{U,B}^H[n]\boldsymbol{\Phi}[n]h_{k,U}[n]|^2}{\sigma^2}\right) \tag{19}$$

2.2 计算模型

由于每个用户的计算能力和电池容量有限，用户可将任务卸载到 MEC 服务器上处理。在一个周期 T 内，用户 k 只产生一个计算密集型任务，需要计算的任务量为 L_k，计算单位比特任务需要的 CPU 周期数为 c_k，并假设所有用户都采用部分卸载模式[18]，这意味着用户 k 的任务可分为两部分，即 $\rho_k L_k$ 的任务在本地处理，$(1-\rho_k)L_k$ 的任务卸载到 MEC 服务器上处理。用户在整个周期 T 内完成本地任务的计算，在分得的时隙内，将卸载的任务传输给 MEC 服务器。通常 MEC 服务器有强大的计算能力，并且计算结果的数据量往往较小，因此本文忽略了 MEC 服务器的计算时延以及结果回传给用户的时延。

2.3 能耗模型

2.3.1 本地计算能耗

本地任务需要在周期 T 内执行完成，假设用户的 CPU 采用动态电压和频率调节技术[19]，因而本地计算能耗可表示为[20]：

$$E_k^{\mathrm{loc}}=\frac{\kappa c_k^3 L_k^3(1-\rho_k)^3}{T^2} \tag{20}$$

其中，κ 为处理器电容系数，该参数与芯片架构有关。

2.3.2 卸载传输能耗

根据分配给用户的时隙长度、用户传输功率以及用户的卸载决策可以得到用户的传输能耗，表示为：

$$E_k^{\text{off}}=\sum_{n=1}^{N}\alpha_k[n]p_k[n]\delta_t \tag{21}$$

其中，$\delta_t=T/N$ 表示一个时隙的长度。

由于用户能耗预算有限，因此用户 k 在分到的时隙内传输能耗总和不能超过最大传输能耗预算 $E_{\text{tr},k}^{\max}$，即：

$$0\leqslant\sum_{n=1}^{N}\alpha_k[n]p_k[n]\delta_t\leqslant E_{\text{tr},k}^{\max} \tag{22}$$

3 问题描述

本文旨在通过联合优化时隙分配策略、任务卸载比例、无人机高度、RIS 相移和用户传输功率，以最小化用户总能耗，因此优化问题可表述为：

$$\begin{aligned}
&\min_{\boldsymbol{A},\boldsymbol{H},\boldsymbol{\Psi},\boldsymbol{\rho},\boldsymbol{p}}\ \sum_{k=1}^{K}(E_k^{\text{loc}}+E_k^{\text{off}})\\
&\text{s.t. C1}:0\leqslant\rho_k\leqslant 1,\ \forall k\in\mathcal{K}\\
&\quad\text{C2}:\alpha_k[n]\in\{0,1\},\ \forall k\in\mathcal{K},\ \forall n\in\mathcal{N}\\
&\quad\text{C3}:\sum_{k=1}^{K}\alpha_k[n]\leqslant 1,\ \forall n\in\mathcal{N}\\
&\quad\text{C4}:\phi_m[n]\in[0,2\pi),\ \forall n\in\mathcal{N},\ \forall m\in\mathcal{M}\\
&\quad\text{C5}:H_{\min}\leqslant H[n]\leqslant H_{\max},\ \forall n\in\mathcal{N}\\
&\quad\text{C6}:|H[n+1]-H[n]|\leqslant D_{\max},\ \forall n\in\mathcal{N}\\
&\quad\text{C7}:H[1]=H_s\\
&\quad\text{C8}:\sum_{n=1}^{N}(\alpha_k[n]R_k[n])\delta_t\geqslant(1-\rho_k)L_k,\ \forall k\in\mathcal{K}\\
&\quad\text{C9}:0\leqslant\sum_{n=1}^{N}\alpha_k[n]p_k[n]\delta_t\leqslant E_{\text{tr},k}^{\max}\\
&\quad\text{C10}:0\leqslant p_k[n]\leqslant p_{\max},\ \forall k\in\mathcal{K},\ \forall n\in\mathcal{N}
\end{aligned}\tag{23}$$

其中，$\boldsymbol{A}=\{\alpha_k[n],\ \forall k\in\mathcal{K},\ \forall n\in\mathcal{N}\}$，$\boldsymbol{H}=\{H[n],\ \forall n\in\mathcal{N}\}$，$\boldsymbol{\rho}=\{\rho_k,\ \forall k\in\mathcal{K}\}$，$\boldsymbol{\Psi}=\{\phi_m[n],\ \forall m\in\mathcal{M},\ \forall n\in\mathcal{N}\}$，$\boldsymbol{p}=\{p_k[n],\forall k\in\mathcal{K},\ \forall n\in\mathcal{N}\}$。我们注意到，当用户的任务全部处于本地计算时，用户总能耗来自本地计算；当用户的任务全部卸载至边缘服务器时，用户总能耗来自卸载传输能耗。卸载比例优化对两种能耗进行了权衡，以最小化总能耗。约束 C1 用于限制本地计算比例；约束 C2 和 C3 保证每个时隙只能分给一个用户；约束 C4 表示 RIS 相移的调整范围；约束 C5 限制了无人机的飞行高度；约束 C6 限制了每个时隙内无人机移动的最大距离；约束 C7 表示无人机的初始高度；约束 C8 表示在周期 T 内，用户 k 需传输所有将要卸载的任务；约束 C9 和 C10 表示周期 T 内的传输能耗约束与传输功率约束。

4 优化问题求解

由于问题(23)非凸，因此难以直接求解，可分解为 4 个子问题：首先，使用 PPO 算法输出时隙分配决策 $\boldsymbol{A}$；然后，在 DRL 训练环境中，使用交替迭代的方法，在给定 $\boldsymbol{A},\boldsymbol{H},\boldsymbol{\rho},\boldsymbol{p}$ 下优化 $\boldsymbol{\Psi}$，在给定 $\boldsymbol{A},\boldsymbol{H},\boldsymbol{\Psi}$ 下优化 $\boldsymbol{\rho}$ 和 $\boldsymbol{p}$；最后，在给定 $\boldsymbol{A},\boldsymbol{\Psi},\boldsymbol{\rho},\boldsymbol{p}$ 下优化 $\boldsymbol{H}$。为方便讨论，本节首先阐述了 DRL 训练环境中的交替迭代方法，然后提出了基于 PPO 的时隙分配算法与 DRL 训练框架。

4.1 RIS 相移优化

根据式(18)、式(19)可知，RIS 最优相移 $\boldsymbol{\Psi}$ 为：

$$\phi_m[n]=-(\varphi_{k,m}[n]+\omega_m[n]),\ \forall n\in\mathcal{N},\ \forall k\in\mathcal{K},\alpha_k[n]=1 \tag{24}$$

4.2 传输功率与卸载比例优化

给定时隙分配策略 $\boldsymbol{A}$、无人机高度 $\boldsymbol{H}$、RIS 相移 $\boldsymbol{\Psi}$，问题(23)可以描述为：

$$\begin{aligned}
&\min_{\boldsymbol{\rho},\boldsymbol{p}}\ \sum_{k=1}^{K}(E_k^{\text{loc}}+E_k^{\text{off}})\\
&\text{s.t. C1}:0\leqslant\rho_k\leqslant 1,\ \forall k\in\mathcal{K}\\
&\quad\text{C8}:\sum_{n=1}^{N}(\alpha_k[n]R_k[n])\delta_t\geqslant(1-\rho_k)L_k,\ \forall k\in\mathcal{K}\\
&\quad\text{C9}:0\leqslant\sum_{n=1}^{N}\alpha_k[n]p_k[n]\delta_t\leqslant E_{\text{tr},k}^{\max}\\
&\quad\text{C10}:0\leqslant p_k[n]\leqslant p_{\max},\ \forall k\in\mathcal{K},\ \forall n\in\mathcal{N}
\end{aligned}\tag{25}$$

上述问题是关于 $\boldsymbol{\rho}$ 和 $\boldsymbol{p}$ 的凸优化问题，可以借助凸优化求解软件(如 CVX)进行求解。

4.3 无人机高度优化

在给定 $\boldsymbol{A},\boldsymbol{\Psi},\boldsymbol{\rho},\boldsymbol{p}$ 时，问题(23)可以描述为：

$$\begin{aligned}
&\min_{\boldsymbol{H}}\ \sum_{k=1}^{K}(E_k^{\text{loc}}+E_k^{\text{off}})\\
&\text{s.t. C5}:H_{min}\leqslant H[n]\leqslant H_{\max},\ \forall n\in\mathcal{N}\\
&\quad\text{C6}:|H[n+1]-H[n]|\leqslant D_{\max},\ \forall n\in\mathcal{N}\\
&\quad\text{C7}:H[1]=H_s\\
&\quad\text{C8}:\sum_{n=1}^{N}(\alpha_k[n]R_k[n])\delta_t\geqslant(1-\rho_k)L_k,\ \forall k\in\mathcal{K}
\end{aligned}\tag{26}$$

上述问题中，约束条件 C8 关于 $\boldsymbol{H}$ 是非凸的，所以(26)是非凸问题，因此，需要对 C8 进行处理。C8 可表示为：

$$\sum_{n=1}^{N}\left(B\log_2\left(1+\frac{g_{k,0}}{(d_{k,\text{U}}[n]d_{\text{U,B}}[n])^2}\right)\right)\delta_t\geqslant(1-\rho_k)L_k \tag{27}$$

其中，

$$\begin{aligned}
g_{k,0}&=\frac{\alpha_k[n]p_k[n](\gamma_0\sum_{m=1}^{M}|[\boldsymbol{h}_{U,B}[n]]_m||[\boldsymbol{h}_{k,U}[n]]_m|)^2}{\sigma^2}\\
(d_{k,U}[n]d_{U,B}[n])^2&=H_U^2[n](H_U[n]-H_B)^2+\\
&\quad H_U^2[n]U_{U,B}+U_{k,U}(H_U[n]-H_B)^2+\\
&\quad U_{U,B}U_{k,U}
\end{aligned}\tag{28}$$

其中，

$$U_{U,B}=(x_U-x_B)^2+(y_U-y_B)^2 \tag{29}$$

$$U_{k,U}=(x_U-x_k)^2+(y_U-y_k)^2 \tag{30}$$

记：

$$\begin{aligned}
F_k[n]&=H_U^2[n](H_U[n]-H_B)^2+H_U^2[n]U_{U,B}+\\
&\quad U_{k,U}(H_U[n]-H_B)^2
\end{aligned}\tag{31}$$

令 $b_0=U_{U,B}U_{k,U}$，则式(27)可改写为如下形式：

$$\sum_{n=1}^{N}\left(B\log_2\left(1+\frac{a_{k,0}}{F_k[n]+b_{k,0}}\right)\right)\delta_t\geqslant(1-\rho_k)L_k \tag{32}$$

我们注意到，式(32)左边关于 F_k 是凸函数，因此可以使用 SCA 方法来处理。对于第 l 次迭代，利用一阶泰勒展开替换左边，表达式如下：

$$R_k[n]\geqslant R_k^{lb}[n]=C_k^{(l)}[n](F_k[n]-F_k^{(l)}[n])+D_k^{(l)}[n] \tag{33}$$

其中，$C_k^{(l)}[n]=-\dfrac{a_{k,0}}{(F_k^{(l)}[n]+b_{k,0})(F_k^{(l)}[n]+a_{k,0}+b_{k,0})}$，$D_k^{(l)}[n]=B\log_2\left(1+\dfrac{a_{k,0}}{F_k^{(l)}[n]+b_{k,0}}\right)$。因为 $F_k[n]$ 关于 $H[n]$

是凸函数，所以 $R_k^{lb}[n]$关于 $H[n]$是凹函数，最终式(32)可转换为：

$$\sum_{n=1}^{N}(\alpha_k[n]R_k^{lb}[n])\delta_t \geqslant (1-\rho_k)L_k \tag{34}$$

上述约束即为凸集。

通过 SCA 方法，将式(26)中的约束 C8 转换为凸约束，重构后的优化问题可表示为：

$$\begin{aligned}&\min_{\boldsymbol{H}} \sum_{k=1}^{K}(E_k^{\text{loc}}+E_k^{\text{off}})\\ &\text{s.t. C5}: H_{\min}\leqslant H[n]\leqslant H_{\max},\forall n\in\mathcal{N}\\ &\quad\text{C6}: |H[n+1]-H[n]|\leqslant D_{\max},\forall n\in\mathcal{N}\\ &\quad\text{C7}: H[1]=H_s\\ &\quad\text{C11}: \sum_{n=1}^{N}(\alpha_k[n]R_k^{lb}[n])\delta_t\geqslant(1-\rho_k)L_k,\forall k\in\mathcal{K}\end{aligned} \tag{35}$$

因此，问题(35)已经具有凸结构，可以使用标准凸优化方法处理。

基于前文的分析，RIS 相移、传输功率、卸载比例与无人机高度可通过交替迭代方法求解，具体如算法 1 所示。

算法 1 求解 RIS 相移、传输功率、卸载比例与无人机高度

输入：最大迭代次数 M_t，时隙分配策略 $\mathbf{A}$

输出：本地计算比例 $\boldsymbol{\rho}[t+1]$，传输功率 $\mathbf{p}[t+1]$，无人机高度 $\mathbf{H}[t+1]$，RIS 相移 $\boldsymbol{\Psi}[t+1]$

1. 初始化 $\boldsymbol{\rho}^0=\boldsymbol{\rho}[t]$，$\mathbf{p}^0=\mathbf{p}[t]$，$\mathbf{H}^0=\mathbf{H}[t]$，$\boldsymbol{\Psi}^0=\boldsymbol{\Psi}[t]$

2. for i←0 to M_t do

3. 根据式(24)计算 $\boldsymbol{\Psi}^{i+1}$

4. 给定 $\mathbf{A}$，$\boldsymbol{\Psi}^{i+1}$，$\mathbf{H}^i$，使用 CVX 求解问题(25)，得到 $\boldsymbol{\rho}^{i+1}$，$\mathbf{p}^{i+1}$

5. 给定 $\mathbf{A}$，$\boldsymbol{\Psi}^{i+1}$，$\boldsymbol{\rho}^{i+1}$，$\mathbf{p}^{i+1}$，使用 SCA 方法求解问题(35)得到 $\mathbf{H}^{i+1}$

6. end for

7. 更新 $\boldsymbol{\rho}[t+1]=\boldsymbol{\rho}^{M_t}$，$\mathbf{p}[t+1]=\mathbf{p}^{M_t}$，$\mathbf{H}[t+1]=\mathbf{H}^{M_t}$，$\boldsymbol{\Psi}[t+1]=\boldsymbol{\Psi}^{M_t}$

其中，$\boldsymbol{\rho}[t]$，$\mathbf{p}[t]$，$\mathbf{H}[t]$，$\boldsymbol{\Psi}[t]$分别表示训练时间步 t 的各优化变量。

4.4 时隙分配优化

在给定 $\boldsymbol{H},\boldsymbol{\Psi},\boldsymbol{\rho},\boldsymbol{p}$ 时，问题(23)可以描述为：

$$\begin{aligned}&\min_{\boldsymbol{A}} \sum_{k=1}^{K}(E_k^{\text{loc}}+E_k^{\text{off}})\\ &\text{s.t. C2}:\alpha_k[n]\in\{0,1\},\forall k\in\mathcal{K},\forall n\in\mathcal{N}\\ &\quad\text{C3}:\sum_{k=1}^{K}\alpha_k[n]\leqslant 1,\forall n\in\mathcal{N}\end{aligned} \tag{36}$$

由于该问题是一个整数优化问题，寻找时隙分配策略 $\boldsymbol{A}$ 的最优解比较困难。本节提出了基于 PPO 算法的时隙分配优化方案，首先介绍了强化学习中 MDP 的基本要素，然后介绍了基于 PPO 算法的时隙分配算法。

4.4.1 MDP 基本要素定义

MDP 的核心是对智能体与环境的交互过程进行建模。在本文场景中，基站不需要任何关于环境的先验信息，只能从环境状态中获取因果信息，因此本文模型中转移概率未知，可建模为无模型、无转移概率的马尔可夫决策过程(Markov Decision Process，MDP)[21]。在 MDP 中，智能体与动态环境不断交互，以优化自身策略，例如，在某个时间步 t，环境处于状态 $s[t]$，智能体执行动作 $a[t]$，环境以某种可能性转移到任何可行的后继状态 $s[t+1]$中，智能体接收奖励 $r[t]$，随后 t 增加 1。智能体通过观察 $s[t+1]$与 $r[t+1]$来调整自身策略，从而使积累奖励最大化。下面分别定义 3 个关键要素，即状态空间、动作空间和奖励函数。

(1)状态空间定义：

$$S[t]=\{s[t]\,|\,s[t]=\{\boldsymbol{L}[t],\boldsymbol{d}_{\text{user}}[t],d_{U,B}[t],\boldsymbol{\rho}[t]\}\} \tag{37}$$

其中，$\boldsymbol{L}[t]=[L_1[t],\cdots,L_K[t]]^{\mathrm{T}}$，代表训练时间步 t 的用户任务总量；$\boldsymbol{d}_{\text{user}}[t]=[d_{1,U}[t],\cdots,d_{K,U}[t]]^{\mathrm{T}}$ 和 $d_{U,B}[t]$分别表示用户与无人机、无人机与基站之间的距离；$\boldsymbol{\rho}[t]=[\rho_1[t],\cdots,\rho_K[t]]^{\mathrm{T}}$，表示算法 1 初始化时的用户本地计算比例。

(2)动作空间定义：

$$A[t]=\begin{cases}a[t]\,|\,a[t]=\{\boldsymbol{b}[t]\}\\ [\boldsymbol{b}[t]]_k=\arg\max\limits_{n}\{\alpha_k[n]=1\}\\ \sum\limits_{n=1}^{N-1}(\alpha_k[n]-\alpha_k[n+1])^2\leqslant 2\\ \arg\min\limits_{n}(\alpha_k[n]-\alpha_k[n+1])\leqslant\\ \quad\arg\max\limits_{n}(\alpha_k[n]-\alpha_k[n+1])\\ \sum\limits_{k=1}^{K}\alpha_k[n]=1,\alpha_k[n]\in\{0,1\},\forall k\in\mathcal{K},\forall n\in\mathcal{N}\end{cases} \tag{38}$$

其中，$[\boldsymbol{b}[t]]_k$ 表示向量 $\boldsymbol{b}[t]$的第 k 个元素。式(38)是压缩后的动作空间，其原始动作空间为：

$$A_0[t]=\{a_0[t]\,|\,a_0[t]=\{\boldsymbol{A}[t]\},\forall k\in\mathcal{K},\forall n\in\mathcal{N}\} \tag{39}$$

其中，$\boldsymbol{A}[t]$表示训练时间步 t 内的时隙分配策略 $\boldsymbol{A}$。求解 $\boldsymbol{A}$ 的原始问题为组合优化，动作空间离散且有限，将 $\boldsymbol{A}$ 的所有可能值进行编码是一种表示状态的方案。然而，动作数量随用户数 K 与时隙数 N 呈指数爆炸，即对于同一个系统，需要考虑 K^N 种动作，这会使得训练难度与模型复杂度迅速增加，几乎无法找到有效的 DRL 方法应对如此超大规模的动作空间，因此这种动作表示方案具有较大的局限性。为使问题的求解难度在可接受范围内，考虑压缩动作空间，以获得满意的次优解。本文在时隙分配策略 $\boldsymbol{A}$ 中加入约束，使每个用户被分配到一段连续的时隙，并按照用户与无人机之间的距离排列用户编号，将动作数量减小为 C_{N+K-1}^{K-1}，即将 N 个时隙分为 K 段的方案数。

压缩后的动作空间对应的时隙分配方式如图 2 所示，时间步 t 的动作向量 $\boldsymbol{b}[t]$的第 k 个元素 $[\boldsymbol{b}[t]]_k$ 表示用户 k 分到的连续时隙段中最后一个时隙的编号。

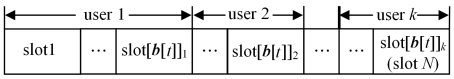

图 2 压缩动作空间后的时隙分配方式

Fig. 2 Time slot allocation scheme of compressed action space

因此问题(36)可以重构为：

$$\begin{aligned}&\min_{\boldsymbol{A}} \sum_{k=1}^{K}(E_k^{\text{loc}}+E_k^{\text{off}})\\ &\text{s.t. C2}:\alpha_k[n]\in\{0,1\},\forall k\in\mathcal{K},\forall n\in\mathcal{N}\\ &\quad\text{C12}:\sum_{n=1}^{N-1}(\alpha_k[n]-\alpha_k[n+1])^2\leqslant 2,\forall k\in\mathcal{K},\forall n\in\mathcal{N}\\ &\quad\text{C13}:\arg\min_{n}(\alpha_k[n]-\alpha_k[n+1])\leqslant\arg\max_{n}(\alpha_k[n]-\alpha_k[n+1]),\forall k\in\mathcal{K},\forall n\in\mathcal{N}\end{aligned}$$

$$C14: \sum_{k=1}^{K}\alpha_k[n]=1, \alpha_k[n]\in\{0,1\}, \forall n\in\mathcal{N} \tag{40}$$

其中，约束 C12 和 C13 表示每个用户分到的时隙是连续的，约束 C14 保证每个时隙只有一个用户接入。我们注意到，所求问题为离散动作空间，若使用基于值函数(Value-Based)的 DRL 方法，如深度 Q 网络(Deep Q Network，DQN)与 DDQN，将所有可能的动作的 Q 值列举为输出层，则会导致输出层大小及模型复杂度随 $N!$ 与 $K!$ 快速增大。相比之下，本文所采用的 PPO 算法是基于策略(Policy-Based)的 DRL 方法，该方法可以输出动作向量每一维度的概率分布，输出层大小仅随 K 线性变化，模型复杂度及训练开销显著低于基于值函数的方法，对 PPO 算法输出的动作向量 $\hat{\boldsymbol{b}}[t]$ 取整后，即可得到 $\boldsymbol{b}[t]$ 与动作 $a[t]$。

(3)奖励函数定义：

$$r[t]=\begin{cases}-(V_1(\mathrm{obj}[t]-\bar{E}[t]))^{\frac{1}{2}}, & \text{If } \mathrm{obj}[t]>\bar{E}[t]\\ 10\min\{e^{V_2(\bar{E}[t]-\mathrm{obj}[t])}-1,1\}, & \text{Otherwise}\end{cases} \tag{41}$$

其中，

$$\mathrm{obj}[t]=\sum_{k=1}^{K}(E_k^{\mathrm{loc}}[t]+E_k^{\mathrm{off}}[t]) \tag{42}$$

在第 t 时间步，智能体给出时隙分配决策 $a[t]$ 后，使用算法 1 求解得到目标函数。其中，$E_k^{\mathrm{loc}}[t]$ 和 $E_k^{\mathrm{off}}[t]$ 表示时间步 t 时用户 k 的计算与卸载能耗；$\bar{E}[t]$ 表示均分时隙策略得到的能耗；V_1 和 V_2 为针对不同场景设置的常数，将在仿真部分说明。本文使用奖励截断(Reward-Clipping)方法来限制奖励最大值[22]，并通过减弱惩罚、与基准值做比较等方法来鼓励智能体探索，加快训练速度。

4.4.2 基于 PPO 的时隙分配算法

PPO 算法考虑了新动作策略和旧动作策略，设置了一个新的目标函数，将动作值稳定在近端范围内，使新的动作策略可以参照旧动作策略进行更新，具有动态决策的优势，可以快速决定模型的优化方向。算法的基本框架如图 3 所示。

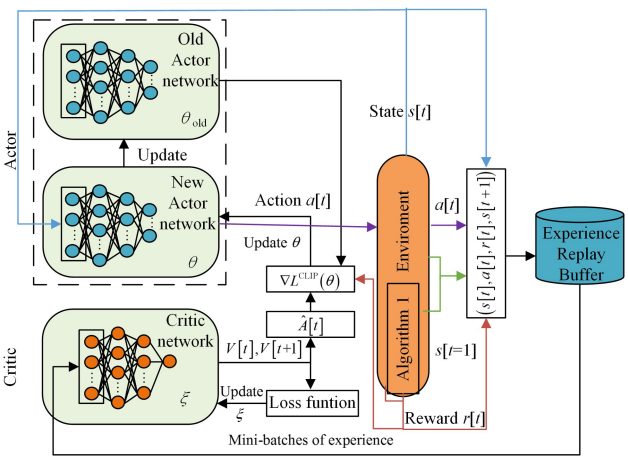

图 3 基于 PPO 的 DRL 训练框架

Fig. 3 DRL training framework based on PPO

图 3 中的环境为系统模型中各个对象构成的任务空间，其中算法 1 嵌入在环境中，提供下一状态 $s[t+1]$ 和奖励 $r[t]$，PPO 方法与行动者-评论家(Actor-Critic，AC)方法同样具有动作网络和评价网络，其中动作网络又分为新动作网络和旧动作网络。新、旧动作网络的参数分别为 θ 和 θ_{old}，评价网络的参数为 ξ。动作网络根据状态输出动作，与环境交互；评价网络根据状态计算状态价值，估计动作的优劣。PPO 的训练过程如下。

本次回合开始时，随机初始化用户位置 $(x_k, y_k, 0)$、任务数 L_k、无人机高度 H_S，随机初始化各网络的参数 $\theta, \theta_{\mathrm{old}}, \xi$，以及回放记忆单元。每次迭代中，PPO 算法将环境状态 $s[t]$ 输入新动作网络中，输出相应动作 $a[t]$ 后，输入到环境中得到奖励 $r[t]$ 与下一状态 $s[t+1]$，向回放记忆单元中存入一个完整的经验 $(s[t], a[t], r[t], s[t+1])$，而后将 $s[t+1]$ 输入新动作网络中，直到存储 Γ 个经验。其次根据式(43)计算每个时间步的优势函数：

$$\hat{A}[t]=\sum_{i=0}^{\Gamma-t-1}(\gamma\lambda)^i\hat{\delta}[t+i] \tag{43}$$

式(43)使用广义优势估计(General Advantage Estimation，GAE)的形式，γ 为折扣因子，λ 是 GAE 的一个系数，$0\leqslant\lambda\leqslant1$，$\hat{\delta}[t]=r[t]+\gamma\cdot v[t+1]-v[t]$，其中 $v[t]$ 为状态价值，由评价网络根据 $s[t]$ 计算得到。随后计算动作网络 θ 的目标函数：

$$L^{\mathrm{CLIP}}(s[t],a[t];\theta)=\mathbb{E}[\min(\frac{\pi_\theta(a[t]|s[t])}{\pi_{\theta_{\mathrm{old}}}(a[t]|s[t])}\hat{A}[t], \mathrm{clip}(\frac{\pi_\theta(a[t]|s[t])}{\pi_{\theta_{\mathrm{old}}}(a[t]|s[t])},1+\varepsilon,1-\varepsilon))\hat{A}[t]] \tag{44}$$

为降低训练难度，本文引入具有剪切概率比的目标函数式(44)，其中 ε 是用于控制截断范围的参数，其值较小，决定了新旧策略之间的差异；然后使用多个时期的数据对 θ 和 ξ 进行小批量更新；最后，将策略 $\pi_{\theta_{\mathrm{old}}}$ 更新为 π_θ，清除经验数据，开始新一轮迭代。

总体而言，在训练过程中，智能体通过与环境交互得以不断学习，优化自身策略，逐渐使奖励值与目标函数最大化。上述算法的具体总结如算法 2 所示。

算法 2 基于 PPO 的时隙分配智能体训练算法

输入：最大回合数 E，最大时间步 Γ，学习率 lr，限制参数 ε，GAE 参数 λ，评价网络参数 ξ

输出：动作网络参数 θ

1. 初始化动作网络参数 θ，评价网络参数 ω
2. for episode←1 to E do
3. 初始化：(x_k, y_k)，Lk，c_k，无人机初始高度 H_S，基站高度 H_B
4. for t←1 to Γ do
5. 从环境中获取状态 s[t]
6. 使用 π_θ 根据状态选择动作 a[t]
7. 根据 a[t]，使用算法 1 求解 **ρ**[t+1]，**p**[t+1]，**H**[t+1]，**Ψ**[t+1]
8. 计算下一状态 s[t+1]
9. 根据式(42)计算 obj[t]
10. 根据(41)计算奖励 r[t]
11. 存储经验(s[t]，a[t]，r[t]，s[t+1])
12. end for

13. for t←1 to Γ do

14. 用式(43)计算 $\hat{A}[t]$

15. end for

16. 更新动作网络 θ，更新评价网络 ξ

17. 更新 $\theta_{old}\leftarrow\theta$

18. 清理经验数据

19. end for

5 计算复杂度分析

对于算法 1，交替迭代进行 M_t 次，每次迭代中，问题(25)含有$(N+1)K$ 个变量，使用 CVX 求解；问题(35)含有 N 个变量，使用 SCA 方法求解。因此算法 1 最坏情况的复杂度为 $O(M_t(((N+1)K+N)^{3.5}+N^3\log_2(1/\varepsilon)))$，其中 ε 是 SCA 方法的精度。

算法 2 中 DRL 框架的复杂度通常以一次迭代中的乘法计算次数来衡量。因此框架的复杂度为 $O(\sum_{q=1}^{Q}n_q n_{q-1})$，其中 n_q 是第 q 隐藏层的神经元数量[23]，由于输入与输出层的乘法运算次数远少于隐藏层，故忽略了其对复杂度的贡献。本文设置的各个隐藏层大小相近，因此，DRL 框架的复杂度可以表示为 $O((q-1)n_q^2)\sim O(n_q^2)$。结合算法 1 进行分析，可以得到算法 2 的总复杂度为：

$$O(\sum_{q=1}^{Q}n_q n_{q-1}+M_t(((N+1)K+N)^{3.5}+N^3\log_2(1/\varepsilon)))$$

6 仿真结果与分析

为了验证本文提出的基于 PPO 与凸优化方法的空中 RIS 辅助 MEC 任务卸载方案对用户总能耗的影响，采用 PyTorch 框架搭建仿真环境并分析所提方案的性能。本文考虑用户随机分布在一个半径为 200 m 的地面圆形区域内，并以用户区域中心为原点，空中 RIS 的水平坐标为(400，0)m，配备 MEC 服务器的基站在(500，0，100)m 处，任务数据量 $L_k\in[10^4,8\times10^5]$bit，单位比特平均计算次数 $c_k\in[300,500]$cycles/bit。周期 $T=2$s，时隙个数 $N=20$，无人机最小飞行高度 $H_{\min}=50$ m，最大飞行高度 $H_{\max}=500$ m，假设所有用户的通信能耗预算均为 $E_{tr,k}^{\max}=0.02$ J，最大传输功率 $p_{\max}=0.1$ W，噪声功率 $\sigma^2=-110$ dBm，$K_1=K_2=10$ dB，信道功率增益 $\gamma_0=-20$ dB。表 1 列出了 PPO 训练参数。

表 1 PPO 训练参数

Table 1 Parameters of PPO-training

Parameter	Value
Number of Hidden Layers Q	3
GAE-λ	0.95
Discount Rate γ	0.8
Data Reuse Times n_r	8
Optimizer	Adam
Clip Fraction，ε	0.2
Active Functions	Softmax and Tanh
Step per Episode Γ	20

本文首先通过图 4 验证了 PPO 算法的收敛性，可以看出，强化学习智能体可以显著提升所述的奖励值，其中设定用户数 $K=10$，$V_1=1000$，$V_2=200$，可根据式(41)使用的奖励截断调整。使用 PyTorch 收集了训练 15000 步的结果，每个结果为一个回合内的奖励值之和。随着训练次数增加，智能体的时隙分配策略逐渐变好，性能的震荡有明显的下降趋势，最终能获得较稳定的奖励值，说明其已经训练成了可以实时决策的智能体。为验证学习率对收敛性的影响，本文还比较了不同学习率下的奖励值收敛曲线。可以看出，学习率为 2×10^{-5} 时曲线在 5000 步左右收敛，学习率为 2×10^{-4} 时曲线在 2000 步左右收敛，学习率为 5×10^{-5} 时介于两者之间，这三者收敛得到的奖励值相差不大且比较稳定，说明学习率对本文 PPO 算法的收敛速度具有一定的影响，但对性能的影响相对较小。

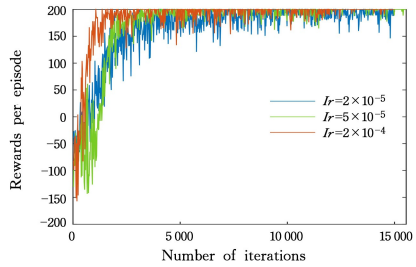

图 4 PPO 算法的收敛图

Fig. 4 Convergence of PPO algorithm

图 5 和图 6 分别给出了 RIS 反射元数目与用户数目对系统能耗的影响。在用户数目固定的情况下，随着 RIS 反射元数目增加，信道增益逐渐增加，信道质量明显提升，系统能耗逐渐下降，并且随着反射元数目的增加，增加同等数量反射元时能耗的下降量逐渐减小。固定 RIS 反射元数目，当用户数目增加，在周期 T 不变的情况下，用户分得的用于任务卸载的时隙减小，因而可卸载任务量减少，本地计算任务量增加，导致系统能耗不断增加。

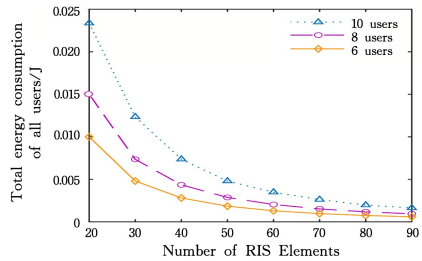

图 5 RIS 反射单元数量变化与用户总能耗间的关系

Fig. 5 Total energy consumption of all users versus the number of RIS reflection elements

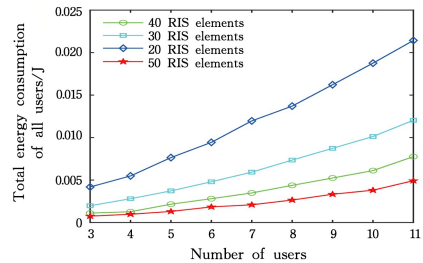

图 6 用户数量变化与用户总能耗间的关系

Fig. 6 Total energy consumption of all users versus the number of users

图 7 给出了用户的本地计算比例和 RIS 反射单元数量与用户数量的关系。可以观察到，当用户数目增加时，在固定 T 值的情况下，由于分得的时隙减小，可卸载的任务量减少，本地计算比例逐渐升高，并且随着 RIS 数量增加，链路增益逐渐增加，信道质量变好，最大可卸载任务量增加，因此优化得到的本地计算比例不断减小。

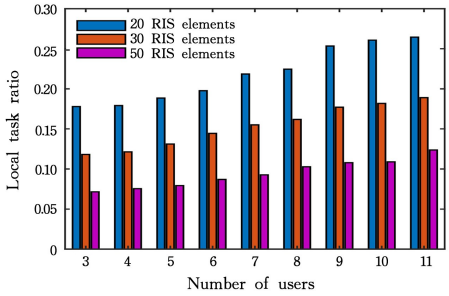

图 7 用户数量变化与用户本地计算比例的关系

Fig. 7 Local task ratio versus the number of users

图 8 给出了 5 种算法或卸载策略的性能比较。当用户数量 $K=10$ 时，任务全部本地计算时的系统能耗在 0.2 J 左右，明显大于所提算法。与任务全部卸载相比，所提算法的能耗减少了约 23%。当无人机在固定高度提供服务时，相比优化高度时的能耗增加了约 5.3%，可见高度优化对目标函数具有一定的积极作用。使用优势行动者-评论家算法(Advantage Actor Critic，A2C)时，能耗增加了 12.8%，相比 A2C 这种改进的行动家-评论者算法，PPO 有明显的提升，可见 PPO 的性能较好，是一种更加适合本文模型的优秀强化学习算法。

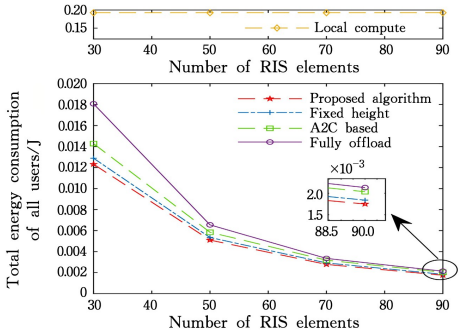

图 8 不同算法或卸载策略的用户总能耗对比

Fig. 8 Total energy consumption under different schemes

结束语 本文研究了空中 RIS 辅助边缘计算中的部分任务卸载方案，应用空中 RIS 来增强地面用户卸载链路质量。为了最小化用户的总能耗，本文提出了一种基于 PPO 与连续凸逼近的任务卸载算法，用于联合优化 RIS 相移、传输功率、无人机高度、任务卸载策略以及时隙分配策略。仿真结果表明，训练得到的智能体能以较低的复杂度生成时隙分配决策，智能体训练环境中的凸优化方法能结合智能体的动作快速得到满意解。本文考虑优化的是固定用户，但在实际场景中，用户也可能具有移动性，未来的工作将在本文的基础上考虑可移动用户的任务卸载策略。

参 考 文 献

[1] WU D P, ZHANG P N, WANG R Y. Smart Internet of things Aided by "Terminal-Edge-Cloud" Cooperation[J]. Chinese Journal on Internet of Things, 2018, 2(3): 21-28.

[2] LI Z J, ZHANG X L. Resource Allocation and Offloading Decision of Edge Computing for Reducing Core Network Congestion [J]. Computer Science, 2021, 48(3): 281-288.

[3] XU Y, ZHANG T, YANG D, et al. UAV-Assisted Relaying and MEC Networks: Resource Allocation and 3D Deployment[C]// 2021 IEEE International Conference on Communications Workshops (ICC Workshops). 2021: 1-6.

[4] ZHANG T, XU Y, LOO J, et al. Joint Computation and Communication Design for UAV-Assisted Mobile Edge Computing in IoT[J]. IEEE Transactions on Industrial Informatics, 2020, 16(8): 5505-5516.

[5] DIAO X B, YANG W D, YANG L X, et al. UAV-Relaying-Assisted Multi-Access Edge Computing With Multi-Antenna Base Station: Offloading and Scheduling Optimization [J]. IEEE Transactions on Vehicular Technology, 2021, 70(9): 9495-9509.

[6] WANG J, NA Z, LIU X. Collaborative Design of Multi-UAV Trajectory and Resource Scheduling for 6G-Enabled Internet of Things[J]. IEEE Internet of Things Journal, 2021, 8(20): 15096-15106.

[7] TIAN H, NI W L, WANG W, et al. Data-Importance-Aware Resource Allocation in IRS-Aided Edge Intelligent System [J]. Journal of Beijing University of Posts and Telecommunications, 2020, 43(6): 51-58.

[8] LI Z Y, CHEN M, YANG Z H, et al. Energy Efficient Reconfigurable Intelligent Surface Enabled Mobile Edge Computing Networks With NOMA[J]. IEEE Transactions on Cognitive Communications and Networking, 2021, 7(2): 427-440.

[9] LI A C, LIU Y, LI M, et al. Joint Scheduling Design in Wireless Powered MEC IoT Networks Aided by Reconfigurable Intelligent Surface[C]// 2021 IEEE/CIC International Conference on Communications in China (ICCC Workshops). Xiamen, China: IEEE, 2021: 159-164.

[10] HUANG S F, WANG S, WANG R, et al. Reconfigurable Intelligent Surface Assisted Mobile Edge Computing With Heterogeneous Learning Tasks [J]. IEEE Transactions on Cognitive Communications and Networking, 2021, 7(2): 369-382.

[11] YANG Z H, HUANG C W, SHI J F, et al. Optimal Control for Full-Duplex Communications with Reconfigurable Intelligent Surface[C]// ICC 2021 - IEEE International Conference on Communications. Montreal, QC, Canada: IEEE, 2021: 1-6.

[12] LIU X, LIU Y W, CHEN Y. Machine Learning Empowered Trajectory and Passive Beamforming Design in UAV-RIS Wireless Networks[J]. IEEE Journal on Selected Areas in Communications, 2021, 39(7): 2042-2055.

[13] MEI H B, YANG K, SHEN J, et al. Joint Trajectory-Task-Cache Optimization With Phase-Shift Design of RIS-Assisted UAV for MEC[J]. IEEE Wireless Communications Letters, 2021, 10(7): 1586-1590.

[14] LONG H, CHEN M, YANG Z H, et al. Joint Trajectory and Passive Beamforming Design for Secure UAV Networks with

RIS[C]// 2020 IEEE Globecom Workshops. Taipei, Taiwan: IEEE, 2020: 1-6.

[15] MURSIA P, DEVOTI F, SCIANCALEPORE V, et al. RISe of Flight: RIS-Empowered UAV Communications for Robust and Reliable Air-to-Ground Networks[J]. IEEE Open Journal of the Communications Society, 2021, 2: 1616-1629.

[16] SAMIR M, ELHATTAB M, ASSI C, et al. Optimizing Age of Information Through Aerial Reconfigurable Intelligent Surfaces: A Deep Reinforcement Learning Approach[J]. IEEE Transactions on Vehicular Technology, 2021, 70(4): 3978-3983.

[17] ZHAN C, HU H, SUI X F, et al. Completion Time and Energy Optimization in the UAV-Enabled Mobile-Edge Computing System[J]. IEEE Internet of Things Journal, 2020, 7(8): 7808-7822.

[18] LI A, DAI L B, YU L S, et al. Resource Allocation for Unmanned Aerial Vehicle-assisted Mobile Edge Computing to Minimize Weighted Energy Consumption[J]. Journal of Electronics & Information Technology, 2021: 1-8.

[19] WANG F, XU J, CUI S. Optimal Energy Allocation and Task Offloading Policy for Wireless Powered Mobile Edge Computing Systems[J]. IEEE Transactions on Wireless Communications, 2020, 19(4): 2443-2459.

[20] WANG F, XU J, WANG X, et al. Joint Offloading and Computing Optimization in Wireless Powered Mobile-Edge Computing Systems[J]. IEEE Transactions on Wireless Communications, 2018, 17(3): 1784-1797.

[21] LIANG J B, ZHANG H H, JIANG C, et al. Research Progress of Task Offloading Based on Deep Reinforcement Learning in Mobile Edge Computing[J]. Computer Science, 2021, 48(7): 316-323.

[22] ENGSTROM L, ILYAS A, SANTURKAR S, et al. Implementation Matters in Deep Policy Gradients: A Case Study on PPO and TRPO[C]// 2020 International Conference on Learning Representations. 2019: 1-14.

[23] LIU C H, DAI Z, ZHAO Y, et al. Distributed and Energy-Efficient Mobile Crowdsensing with Charging Stations by Deep Reinforcement Learning[J]. IEEE Transactions on Mobile Computing, 2021, 20(1): 130-146.

[24] BOYD S, VANDENBERGHE L. Convex Optimization [M]. Cambridge: Cambridge University Press, 2004.

XIE Wan-cheng, born in 2001, postgraduate, is a student member of China Computer Federation. His main research interests include IoT and edge intelligence.

LI Bin, born in 1987, Ph. D, associate professor, is a member of China Computer Federation. His main research interests include IoT and edge intelligence.

(责任编辑:喻藜)