

基于深度强化学习的能源互联网智能巡检任务分配机制

徐思雅¹, 邢逸斐¹, 郭少勇¹, 杨超², 邱雪松¹, 孟洛明¹

(1. 北京邮电大学网络技术与交换重点实验室, 北京 100876; 2. 国网辽宁省电力有限公司信息通信分公司, 辽宁 沈阳 110004)

摘要: 在能源互联网中引入无人机进行电力线路巡查, 并借助移动边缘计算技术实现巡检任务的接入和处理, 可降低服务成本, 提高工作效率。但是, 由于无人机数据传输需求和地理位置的动态变化, 易造成边缘服务器负载不均衡, 致使巡检业务处理时延和网络能耗较高。为解决以上问题, 提出基于深度强化学习的能源互联网智能巡检任务分配机制。首先, 综合考虑无人机和边缘节点的运动轨迹、业务差异化的服务需求、边缘节点有限的服务能力等, 建立面向时延、能耗等多目标联合优化的双层边缘网络任务卸载模型。进而, 基于 Lyapunov 优化理论和双时间尺度机制, 采用近端策略优化的深度强化学习算法, 对固定边缘汇聚层和移动边缘接入层边缘节点间的连接关系和卸载策略进行求解。仿真结果表明, 所提机制能够在保证系统稳定的情况下降低服务时延和系统能耗。

关键词: 巡检无人机; 任务卸载; 近端策略优化; 李雅普诺夫优化; 人工智能

中图分类号: TN92

文献标识码: A

DOI: 10.11959/j.issn.1000-436x.2021071

Deep reinforcement learning based task allocation mechanism for intelligent inspection in energy Internet

XU Siya¹, XING Yifei¹, GUO Shaoyong¹, YANG Chao², QIU Xuesong¹, MENG Luoming¹

1. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

2. Information and Communication Branch, State Grid Liaoning Electric Power Co., Ltd., Shenyang 110004, China

Abstract: In order to reduce the cost and improve efficiency of power line inspection, UAV (unmanned aerial vehicle), which use mobile edge computing technology to access and process service data, are used to inspect power lines in the energy internet. However, due to the dynamic changes of UAV data transmission demand and geographical location, the edge server load will be unbalanced, which causes higher service processing delay and network energy consumption. Thus, an intelligent inspection task allocation mechanism for energy internet based on deep reinforcement learning was proposed. First, a two-layer edge network task offloading model was established to archive joint optimization of multi-objectives, such as delay and energy consumption. It was designed by comprehensively considering the route of UAV and edge nodes, different demands of services and limited service capabilities of edge nodes. Furthermore, based on Lyapunov optimization theory and dual-time-scaled mechanism, proximal policy optimization algorithm based deep reinforcement learning was used to solve the connection relationship and offloading strategy of edge servers between fixed edge sink layer and mobile edge access layer. The simulation results show that, the proposed mechanism can reduce the service request delay and system energy consumption while ensuring the stability of system.

Keywords: patrol UAV, task offloading, proximal policy optimization, Lyapunov optimization, artificial intelligence

收稿日期: 2020-12-07; 修回日期: 2021-03-10

通信作者: 郭少勇, syguo@bupt.edu.cn

基金项目: 国家重点研发计划基金资助项目 (No.2019YFB2102302); 国家自然科学基金资助项目 (No.61702048); 工业互联网创新发展工程基金资助项目 (基于泛在电力物联网的工业互联网测试床)

Foundation Items: The National Key Research and Development Program of China (No.2019YFB2102302), The National Natural Science Foundation of China (No.61702048), Innovative Development Project of Industrial Internet (Construction of Industrial Internet Test Bed Based on Ubiquitous Power Internet of Things)

1 引言

能源互联网是综合运用先进的电力电子技术、通信技术、信息技术和智能管理技术支撑能源节点互联,实现能量双向流动的能量对等交换与共享网络。我国倡导构建全球能源互联网,推动以清洁和绿色的方式满足全球电力需求。但是,随着能源互联网规模的逐渐扩大,海量的物联网设备对电能输送容量与传输距离的要求越来越高,这使我国架空输电线路工程的规模也随之提升。因此,输电线路日常巡检成为电力公司各级运检部门的重要工作之一^[1]。但是,部分输电线路需要穿过地形复杂和自然环境恶劣的地区^[2],这给运维人员的日常巡检工作带来了极大的挑战。近年来,电力运检部门借助无人机(UAV, unmanned aerial vehicle)技术来解决传统人力巡检效率低下的问题。UAV技术是指以无线电遥控技术为基础、以适配性极强的程序为控制手段的一种无人驾驶航空设备,可提供实时数据采集和无线数据传输等功能。这种新型巡检方式安全高效,可以灵活地在人力难以到达的地理区域进行巡检作业,降低操作成本并保障运维人员的人身安全^[3]。

然而,随着无人机巡检方式的推广,新的问题也随之产生。由于无人机的计算能力有限,不能独自处理巡检过程中产生的海量图像数据,需要将任务卸载到其他服务器上处理。在无人机数据卸载方面,移动边缘计算(MEC, mobile edge computing)技术发挥着重要作用。MEC按照我国输电线路的实际地理位置部署边缘服务器,可充分发挥边缘计算在网络中灵活分布的优势,并可按照一定的卸载机制进行无人机数据卸载,以降低任务时延和系统能耗^[4],延长整个巡检系统续航时间并提高网络资源利用率。因此,研究无人机巡检任务的卸载机制具有重要的理论和应用价值。

近年来,针对无人机任务卸载的研究已有较多成果,主要解决无人机任务处理中的节能问题。例如,文献[5]针对MEC场景提出了一种面向能耗的任务调度算法,联合优化无人机的路径规划和移动设备的信道资源,可降低所有移动设备的能耗,但未考虑任务生成的动态特性和任务传输的排队时延。文献[6]提出一种基于块坐标下降和连续凸逼近(SCA, successive convex approximation)相结合的迭代算法,结合无人机轨迹、发射功率和时延要求对

通信资源进行调度,提高用户通信带宽,但忽略了无人机正常飞行的能耗。文献[7]在三级雾计算网络中以任务所经历的能量消耗和时延加权和为目标函数,并提出3种决策算法求解该联合优化问题,分析并论证了无人机位置、移动设备和雾节点的处理频率、传输功率的分配方案,但其应用场景中雾节点数量较少,实用价值较低。文献[8]在任务约束、信息因果关系约束、带宽分配约束和无人机轨迹约束下,联合优化无人机和用户设备的能量消耗,但其应用场景中无人机数量较少,存在算法普适性较低的问题。文献[9]在满足传感器节点传输速率需求的条件下,提出了一种基于块坐标下降和SCA相结合的迭代算法,解决了无人机调度方案、功率分配策略和飞行轨迹等的联合优化问题,降低了无人机的功耗,但缺乏对传感器节点能耗的分析。文献[10]通过3次计算优化了无人机辅助网络资源分配方案,减少包括通信相关能量在内的总能耗,但忽略了终端设备和边缘服务器的能耗问题。综上分析可知,现有方法可以有效实现无人机能耗和时延的联合优化,但大多数仅考虑了无人机的能耗,忽略了边缘服务器的长期能量约束以及不同电力巡检任务多样化的服务要求,实用性较差,不能直接应用于电力巡检场景。

为解决以上问题,本文对实际巡检任务需求进行分析,提出了一种基于深度强化学习的能源互联网智能巡检任务分配机制,建立了基于双层边缘网络的云边端网络架构,并结合Lyapunov优化理论和强化学习技术设计巡检任务智能分配算法,在满足不同巡检场景业务需求的条件下,减少了系统能耗和业务时延。

本文主要的贡献如下。

1) 为了延长电力巡检移动设备的使用寿命,针对输电线路巡检任务的实际场景,建立了基于双层边缘网络的云边端网络架构。双层边缘网络由移动边缘接入层和固定边缘汇聚层组成。其中,固定边缘汇聚层分别通过光纤和无线网络与云中心和无人机相连;无人机通过无线网络与移动边缘接入层或固定边缘汇聚层相连;移动边缘接入层具有移动性,可以更加灵活地处理终端层的任务,更契合复杂多样的地理环境,满足电力巡检实际场景要求。

2) 提出了一种双时间尺度机制来解决双层边缘网络模型之间的关联问题,该机制在较长时间尺度内解决双层边缘网络之间的关联问题,在较短时

间尺度内执行动态任务卸载策略，可以降低算法复杂度并减少系统处理时延。

3) 根据巡检业务对通信和计算资源的需求，建立了任务卸载的能耗和时延模型。基于 Lyapunov 优化理论，将长期能量限制下的联合卸载优化问题转化为每个时隙的 Lyapunov 漂移加惩罚项在线优化问题。本文设计了固定边缘汇聚层和移动边缘接入层的能量亏损队列来监测巡检系统移动设备的能量消耗状态，并提出了一种基于近端策略优化 (PPO, proximal policy optimization) 算法的巡检任务分配机制来求解卸载策略。

2 系统模型

2.1 网络架构

针对无人机输电线路巡检任务的实际应用场景，本文建立了基于双层边缘网络的云边端网络架构，它由终端层、移动边缘接入层、固定边缘汇聚层和云平台层组成，如图 1 所示。

1) 终端层。终端层由无人机构成。无人机按照

既定航线接近电力终端设备后悬停飞行或在一定范围内巡航^[1]。本文假设无人机不在本地进行数据处理工作，只负责采集不同类型的数据（图像、视频数据等），并通过无线通信的方式将任务数据卸载到移动边缘接入层、固定边缘汇聚层或云平台层进行处理。

2) 移动边缘接入层。移动边缘接入层由计算能力有限的移动通信车组成。移动通信车承载着底层 MEC 服务器 (Sub-MECS, subordinate MEC server)，担任无人机控制站的角色^[12]。Sub-MECS 可以接收并处理终端层发送的任务，以实现负载均衡和资源共享。

3) 固定边缘汇聚层。固定边缘汇聚层由计算能力较强的 MEC 服务器 (M-MECS, main MEC server) 及其所属的基站组成，可以接收并处理终端层 UAV 的任务，也可以将部分对时延敏感度不高的任务卸载到云平台层进行计算处理。所有的 M-MECS 都通过光纤链路连接到云平台层。

4) 云平台层。云平台层由具有强大计算能力的

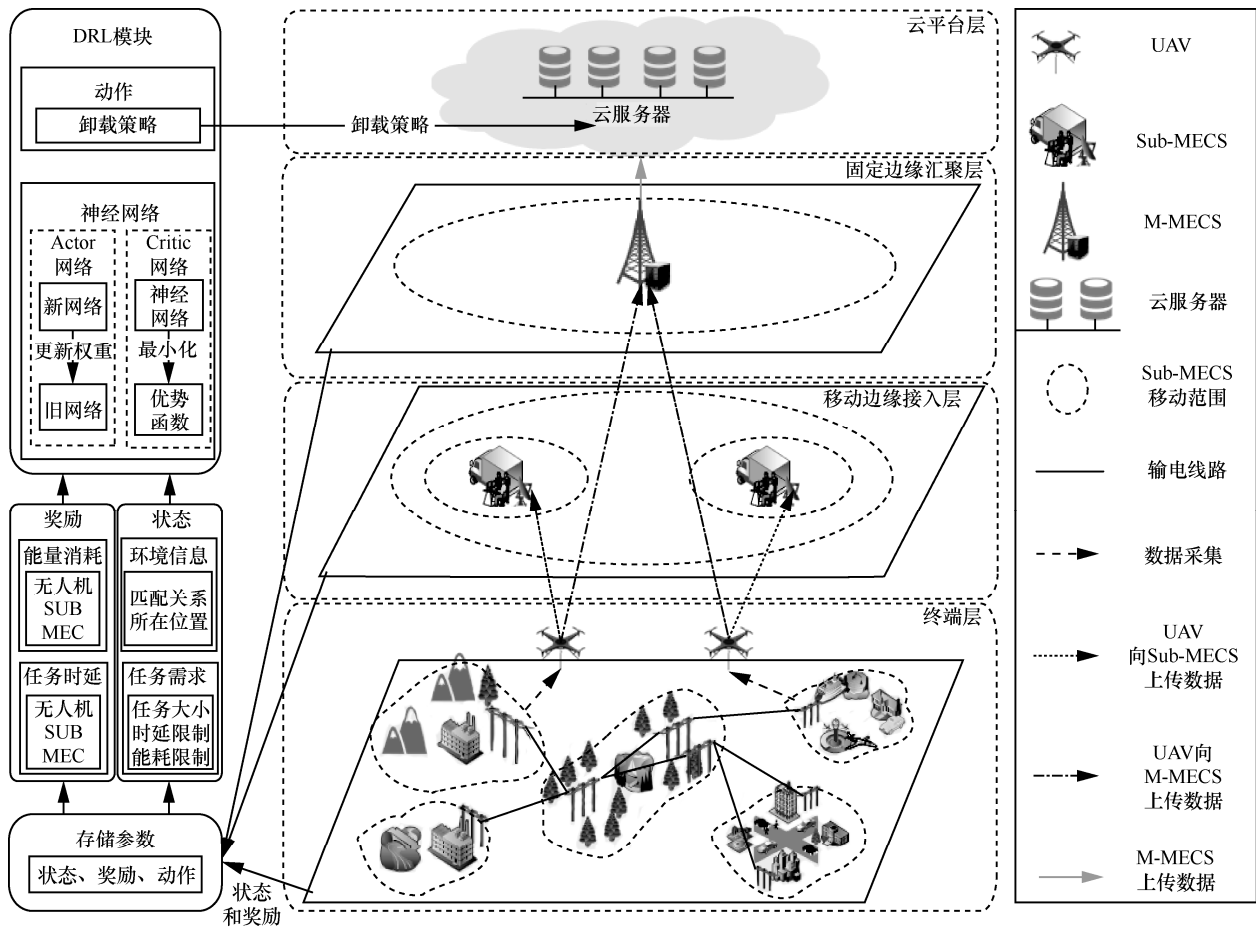


图 1 基于双层边缘网络的云边端网络架构

云服务器组成，通过固定边缘汇聚层接收终端层的任务包头数据，收集任务所需的计算资源和时延要求，并对任务卸载策略进行学习和决策，存储不同个体的决策模型，也可以统一计算处理固定边缘汇聚层发送的任务。

2.2 巡检任务

1) 巡检任务性能指标

输电线路巡检是指对电力线路及附属设备的运行状态以及电力线路走廊周边环境进行及时准确的巡视检查，排查发现隐患点并进行有效的消除，确保电网安全平稳运行^[13]。无人机电力巡检的任务指标定义如下。

时延。任务生成到获得处理结果花费的时间，包括传输时延、传播时延、排队时延和处理时延。

能耗。设备执行任务所需的能量消耗，包括无人机、Sub-MECS、M-MECS 和云平台。

2) 巡检任务服务需求

在输电线路巡检过程中，有不同的电力巡检场景，具体介绍如下。

电塔巡检。电塔巡检过程中需要无人机对杆塔、周边环境和绝缘子等金属部件进行图像采集，进而分析电塔的运行情况。由于电塔位置分散，尤其是恶劣自然环境下，电塔间的距离进一步扩大。这种情况下，为了减少单次巡检任务中无人机对同一区域的通勤频率，应延长无人机的生命周期。因此，需着重考虑电塔巡检任务的能耗问题。

导线巡检。导线巡检过程中需要无人机对导线磨损、导线划伤、导线断股以及沿途树木和建筑物等情况进行图像采集并分析。导线故障可能会造成大面积区域断电，但实际处理导线故障需要人工实地进行更换导线，无人机传输导线异常的时延与之相比可以忽略不计。因此，相比于时延，导线巡检任务对能耗的要求更高。

变电站巡检。实际巡检过程中，巡检人员会采

用目测法、耳听法、鼻嗅法和手触法等方法对变电站设备是否处于异常进行判断。在这种情况下，巡检人员会实地完成变电站巡检任务，并尽快处理变电站故障，防止发生断电和爆炸事故。无人机则主要完成辅助巡检任务，对部分变电站设备和周围低空范围的植被和违章建筑等进行隐患巡检。因此，变电站巡检任务具有较高的时延要求。

2.3 双时间尺度模型

假设 UAV 和 Sub-MECS 在网络中是随机分布且具有移动性的，而固定边缘汇聚层的 M-MECS 位置不变。由于 UAV 和 Sub-MECS 的地理位置不断变化，本文采用就近原则解决 UAV 和 Sub-MECS 与上级边缘节点的连接问题。

根据 UAV 和 Sub-MECS 在电力场景中的位置变化频率，本文提出了一种双时间尺度机制来解决 UAV 和 Sub-MECS 的位置变换和任务卸载转移问题，如图 2 所示。

系统在时域上分为若干时隙，短时隙的集合可表示为 $t = \{t_1, t_2, \dots, t_n\}$ ，每个短时隙的长度为 l 。为了解决 UAV 和 Sub-MECS 的位置变换问题，将连续的 ξ 个时隙认定为一个大气隙，用 $T = \{T_1, T_2, \dots, T_m\}$ 表示，其中大气隙与小时隙的转换关系为 $T_i = \{t_{(i-1)\xi+1}, t_{(i-1)\xi+2}, \dots, t_{i\xi}\}$ 。为了简化模型，假设第 t 个短时隙的数据在第 $t-1$ 个时隙的末尾生成。因此，本文认为第 t 个时隙的数据可以在当前时隙一开始就进行处理。需要注意的是，各种任务的执行速度是不同的，即有些任务可能在当前时隙内没有完成，将在下一个时隙继续处理。参数符号定义如表 1 所示。

在每个短时间尺度时隙的开始，每个 UAV 可以移动，并选择与一个新的 Sub-MECS 相关联。每个长时间尺度时隙的开始，每个 Sub-MECS 可以移动，并可以与一个新的 M-MECS 关联。然后，UAV 的任务根据卸载策略卸载到 Sub-MECS、M-MECS 或云平台层上进行任务处理。

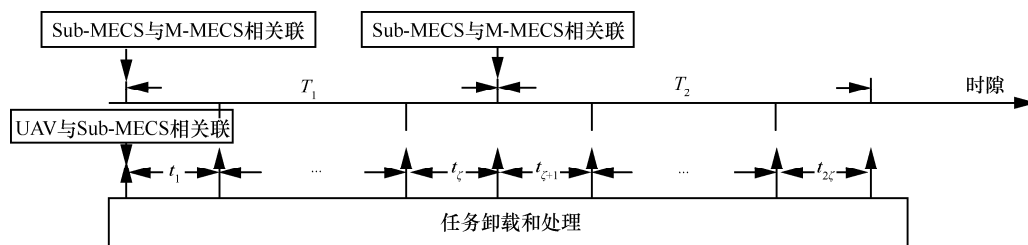


图 2 双时间尺度模型

表1 系统参数符号

参数	含义
$t = \{t_1, t_2, \dots, t_n\}$	短时隙集合
$T = \{T_1, T_2, \dots, T_m\}$	长时隙集合
$U = \{u_1, \dots, u_i, \dots, u_x\}$	UAV 的集合
$S = \{s_1, \dots, s_j, \dots, s_y\}$	Sub-MECS 的集合
$M = \{m_1, \dots, m_k, \dots, m_z\}$	M-MECS 的集合
$R_i^u(t)$	u_i 在时隙 t 产生的任务请求
$x^h(t)$	时隙 t 的计算卸载策略
$\lambda^{\text{sub}}(t) = \{\lambda_1^{\text{sub}}(t), \dots, \lambda_y^{\text{sub}}(t)\}$	Sub-MECS 在时隙 t 的任务到达率
$\lambda^{\text{main}}(t) = \{\lambda_1^{\text{main}}(t), \dots, \lambda_z^{\text{main}}(t)\}$	M-MECS 在时隙 t 的任务到达率
$P_i^{u,c}$	每时隙中 UAV 的运行功耗
$E_i^{us}(t)$ 和 $E_i^{um}(t)$	UAV 在时隙 t 向 Sub-MECS 和 M-MECS 的传输能耗
$E_j^{s,c}(t)$	Sub-MECS 在时隙 t 处理能耗
$P_k^{m,c}(t)$	M-MECS 在时隙 t 的运行功耗
$D_i^{us}(t)$ 和 $D_i^{um}(t)$	UAV 在时隙 t 向 Sub-MECS 和 M-MECS 的传输时延
$D_j^{s,c}(t)$	Sub-MECS 在时隙 t 处理时延
$D_k^{m,c}(t)$	M-MECS 在时隙 t 处理时延
$D_k^{\text{cloud}}(t)$	M-MECS 在时隙云传输时延

2.4 卸载模型

本文所考虑的网络架构使用 $U = \{u_1, \dots, u_i, \dots, u_x\}$ 、 $S = \{s_1, \dots, s_j, \dots, s_y\}$ 和 $M = \{m_1, \dots, m_k, \dots, m_z\}$ 来表示 UAV、Sub-MECS 和 M-MECS 的集合。假设 $R_i^u(t)$ 表示 u_i 在时隙 t 根据泊松过程生成的任务，使用 $(L_i^u(t), \tau_i^u(t))$ 来描述任务 $R_i^u(t)$ ，其中 $L_i^u(t)$ 表示 $R_i^u(t)$ 的输入数据大小（用字节表示）， $\tau_i^u(t)$ 表示 $R_i^u(t)$ 的时延要求，即任务必须在 $\tau_i^u(t)$ 时间内完成。为了简化模型，本文使用 $R(L, \tau)$ 作为不可分割的单位输入任务，且 u_i 所有的生成任务大小都是 $R(L, \tau)$ 的整数倍，即 $L_i^u = C_i^u L$ 。假设 $R_j^s(t)$ 为 s_j 在时隙 t 被卸载的任务， $R_k^m(t)$ 为 m_k 在时隙 t 被卸载的任务。然后，用 $x^h(t) \in \{0, 1\}$ 表示无人机在时隙 t 的卸载策略，其中， $h \in \{\text{sub}, \text{mec}, \text{cloud}\}$ 。 $x_{i,j,k}^{\text{sub}}(t) = 1$ 表示 $R_i^u(t)$ 从 u_i 卸载到 s_j ， $x_{i,j,k}^{\text{mec}}(t) = 1$ 表示 $R_i^u(t)$ 从 u_i 卸载到 m_k ， $x_{i,j,k}^{\text{cloud}}(t) = 1$ 表示 $R_i^u(t)$ 从 u_i 通过固定边缘汇聚节点 m_k 卸载到云端。因此，任务 $R_i^u(t)$ 的卸载策略应满足

$$x_{i,j,k}^{\text{sub}}(t) + x_{i,j,k}^{\text{mec}}(t) + x_{i,j,k}^{\text{cloud}}(t) = 1 \quad (1)$$

2.5 能耗模型

1) 无人机能耗模型

在网络中，无人机只承担数据的采集和打包转发功能，不进行数据的计算和处理。因此，无人机的能耗包含运行能耗和无线传输能耗 2 个部分。为了简化模型，在运行能耗方面，将每个时隙中无人机的运行功耗设置为定值 $P_i^{u,c}$ ；在无线传输能耗方面，本文主要关注无人机任务数据 $R_i^u(t)$ 的上行链路流量，忽略下行链路流量和任务描述数据 $(L_i^u(t), \tau_i^u(t))$ 。

本文假设 u_i 在正交信道上进行数据传输。

当 $x_{i,j,k}^{\text{sub}}(t) = 1$ 时，任务 $R_i^u(t)$ 卸载到移动边缘接入层，则 u_i 和 s_j 之间的上行链路传输速率 r_i^{us} 可由香农容量给出

$$r_i^{us} = W \ln \left(1 + \frac{h_i P_i^{us}}{\sigma^2} \right) \quad (2)$$

其中， P_i^{us} 是 u_i 向移动边缘汇聚层卸载的无线传输功率， W 是信道带宽， h_i 是信道增益， σ^2 是噪声功率^[14]。则 u_i 在时隙 t 的无线传输的能量消耗为

$$E_i^{us}(t) = \frac{P_i^{us} R_i^u(t)}{r_i^{us}} \quad (3)$$

当 $x_{i,j,k}^{\text{mec}}(t) = 1$ 或 $x_{i,j,k}^{\text{cloud}}(t) = 1$ 时，任务 $R_i^u(t)$ 卸载到固定边缘汇聚层，假设 u_i 向固定边缘汇聚层卸载的传输功率为 P_i^{um} ，则 u_i 和 m_k 之间的上行链路传输速率 r_i^{um} 可由香农容量给出

$$r_i^{um} = W \ln \left(1 + \frac{h_i P_i^{um}}{\sigma^2} \right) \quad (4)$$

则 u_i 在时隙 t 的无线传输的能量消耗为

$$E_i^{um}(t) = \frac{P_i^{um} R_i^u(t)}{r_i^{um}} \quad (5)$$

此外，根据 Friis 传输方程组，自由空间功率损耗与发射端和接收端之间距离的平方成正比^[15-16]，即

$$\frac{P_r}{P_s} = \frac{K}{S^2} \quad (6)$$

其中， K 为不同环境下的影响因子， P_r 和 P_s 分别为接收功率和发射功率， S 为接收端和发射端之间的距离。因此，当传输距离变大时，UAV 的无线发射功率也随之增大。

2) 无人机控制站 (Sub-MECS) 能耗模型

Sub-MECS 的能量消耗主要由任务计算处理能耗组成, 计算处理能耗可由计算任务所需的 CPU 周期衡量。设 $f_j(t)$ 表示 s_j 调度的 CPU 周期频率, 允许的最大 CPU 周期频率定义为 f_{\max} , 即 $0 < f_j(t) < f_{\max}$ 。所以在时隙 t 中, s_j 的任务处理能耗为^[17]

$$E_j^{s,c}(t) = \kappa(f_j(t))^3 \min\left(l, \frac{R_j^s(t)}{f_j(t)}\right) \quad (7)$$

其中, κ 为与芯片结构相关的有效开关电容^[18]。

因此, 如果 s_j 的当前能量预算不足以满足 s_j 的任务处理需求, 则需要将当前任务顺延到下一个时隙进行计算, 由此产生的排队时延将在后续时延模型中详细阐述。

3) M-MECS 能耗模型

M-MECS 的处理能耗是边缘计算网络的主要消耗。为了简化模型, 在 M-MECS 稳定工作的条件下, 本文仅考虑了 M-MECS 的计算能耗, 其中, M-MECS 的功耗包括两部分: 服务器空闲功耗和与 CPU 工作负载相关的动态功耗^[19]。功率模型可以表示为

$$P_k^{m,c}(t) = \alpha P_{\max} + (1 - \alpha) P_{\max} \theta(t) \quad (8)$$

其中, P_{\max} 表示 M-MECS 被充分利用时的峰值功率消耗; α 表示 M-MECS 空闲时的功耗比例, 平均值在 50%~70%; $\theta(t)$ 表示 CPU 利用率, 定义为处理的总计算任务与 CPU 最大服务速率的比值, 即

$$\theta(t) = \frac{R_k^m(t)}{\beta_k} \quad (9)$$

其中, $R_k^m(t)$ 表示 m_k 在时隙 t 中卸载到 M-MECS 处理的全部任务, 且 $R_k^m(t) = \sum_{i=1}^x x_{i,j,k}^{\text{mec}}(t) R_i^u(t)$, β_k 是 m_k 的最大服务率。

2.6 时延模型

1) UAV 时延模型

UAV 产生的任务 $R_i^u(t)$ 在 UAV 进行任务卸载的过程中, 其时延主要包括无线传输时延和传播时延。

当 $x_{i,j,k}^{\text{sub}}(t) = 1$ 时, u_i 通过上行链路信道向 s_j 发送计算任务, P_i^{us} 表示其传输功率, r_i^{us} 表示其传输速率, 可由香农定理计算得出。因此, UAV 向

Sub-MECS 的传输时延为

$$D_i^{\text{us}}(t) = \frac{R_i^u(t)}{r_i^{\text{us}}} + \frac{\text{Len}_i^{\text{us}}(t)}{v_i^u} \quad (10)$$

其中, $R_i^u(t)$ 表示 u_i 向 s_j 传输的任务大小, $\text{Len}_i^{\text{us}}(t)$ 表示 u_i 与 s_j 间的实际距离, v_i^u 表示电磁波的传播速度, 一般设为 3×10^8 m/s。

当 $x_{i,j,k}^{\text{mec}}(t) = 1$ 或 $x_{i,j,k}^{\text{cloud}}(t) = 1$ 时, u_i 通过上行链路信道向 m_k 发送计算任务, P_i^{um} 表示其传输功率, r_i^{um} 表示其传输速率, 则 UAV 向 M-MECS 的传输时延为

$$D_i^{\text{um}}(t) = \frac{R_i^u(t)}{r_i^{\text{um}}} + \frac{\text{Len}_i^{\text{um}}(t)}{v_i^u} \quad (11)$$

2) Sub-MECS 时延模型

假设 Sub-MECS 的任务到达率为 $\lambda^{\text{sub}}(t) = \{\lambda_1^{\text{sub}}(t), \dots, \lambda_y^{\text{sub}}(t)\}$, 其中, $\lambda_j^{\text{sub}}(t) = \sum_{i=1}^x x_{i,j,k}^{\text{sub}}(t) R_i^u(t)$ 。因此, s_j 的本地处理任务为

$$R_j^s(t) = \lambda_j^{\text{sub}}(t) = \sum_{i=1}^x x_{i,j,k}^{\text{sub}}(t) R_i^u(t)$$

s_j 的本地处理时延与本地卸载的任务量 $R_j^s(t)$ 和当前 CPU 频率 $f_j(t)$ 有关, 则 s_j 处理时延为

$$D_j^{s,c}(t) = \frac{R_j^s(t)}{f_j(t)} + D_j^{s,q}(t) \quad (12)$$

由于 s_j 的能量预算可能不满足当前时隙的能量消耗, 因此将占用下一时隙的能量, 并产生排队时延 $D_j^{s,q}(t)$ 。

3) M-MECS 时延模型

假设 M-MECS 的任务到达率为 $\lambda^{\text{main}}(t) = \{\lambda_1^{\text{main}}(t), \lambda_2^{\text{main}}(t), \dots, \lambda_z^{\text{main}}(t)\}$, 其中, $\lambda_k^{\text{main}}(t) = \sum_{i=1}^x (x_{i,j,k}^{\text{mec}}(t) + x_{i,j,k}^{\text{cloud}}(t)) R_i^u(t)$ 。 m_k 的本地处理任务为

$$R_k^m(t) = \sum_{i=1}^x x_{i,j,k}^{\text{mec}}(t) R_i^u(t)$$

网络中的无人机可能产生不同类型的数据, 因此计算任务所需的 CPU 周期数也有所不同。假设单个任务所需 CPU 周期数服从指数分布, 在处理速率不变的情况下, 任务的服务时间服从指数分布。进一步假设计算任务的生成速率服从泊松分

布,则每个 M-MECS 的计算时延可以建立为 M/M/1 排队模型^[20-22], m_k 的预期计算时延 $D_k^{m,c}(t)$ 为

$$D_k^{m,c}(t) = \frac{1}{\beta_k(t) - R_k^m(t)} \quad (13)$$

本文假设云服务器的计算时延和能量消耗可以忽略不计。将任务卸载到云的主要时延成本是回程时延,这同样是一个 M/M/1 系统^[14]。那么,任务从 m_k 向云平台的传输时延为

$$D_k^{\text{cloud}}(t) = \psi_k(t) \frac{\tau}{1 - \tau\psi(t)}, \psi(t) < \frac{1}{\tau} \quad (14)$$

$$\text{其中, } \psi_k(t) = \sum_{i=1}^x x_{i,j,k}^{\text{cloud}}(t) R_i^u(t), \psi(t) = \sum_{k=1}^z \psi_k(t),$$

τ 表示在没有拥塞的情况下发送和接收一个单位任务负载的平均传输时间。

3 算法分析

3.1 问题描述与转换

本文针对所提出的基于双层边缘网络的云端网络架构,设计了一种基于深度强化学习的能源互联网智能巡检任务分配机制,以实现无人机在电力场景巡检过程中通信和计算资源的联合优化分配。该机制通过将计算任务卸载到不同的移动边缘接入层和固定边缘汇聚层的边缘服务器来提高网络资源的利用率,满足用户不同的 QoS 要求。Sub-MECS 可以处理计算简单且时延敏感型的任务,而在 M-MECS 上可以处理计算密集型 and 时延敏感型的任务。此外,当时延要求不严格时,可以将任务卸载到云平台。对于巡检任务的工作强度和um时间跨度来说,智能设备的能量储备和生命周期有限。因此,为了延长设备的生命周期,本文假设每一个设备都有极限功率,设备有预先确定的长期能耗限制,以满足巡检任务的实际应用情况。

为了简化模型, u_i 任务的处理结果在上层服务器处理完自身所有任务后统一回传,不考虑单一任务完成后立即回传。根据第 2 节讨论的系统模型, u_i 的任务 R_i^u 时延为

$$\begin{aligned} D_i^u(t) = & x_{i,j,k}^{\text{sub}}(t)(D_i^{\text{us}}(t) + D_j^{s,c}(t)) + \\ & x_{i,j,k}^{\text{mec}}(t)(D_i^{\text{um}}(t) + D_k^{m,c}(t)) + \\ & x_{i,j,k}^{\text{cloud}}(t)(D_i^{\text{um}}(t) + D_k^{\text{cloud}}(t)) \end{aligned} \quad (15)$$

无人机 u_i 的能耗为

$$\begin{aligned} E_i^u(t) = & P_i^{u,c} l + x_{i,j,k}^{\text{sub}}(t) E_i^{\text{us}}(t) + (x_{i,j,k}^{\text{mec}}(t) + x_{i,j,k}^{\text{cloud}}(t)) E_i^{\text{um}}(t) = \\ & P_i^{u,c} l + x_{i,j,k}^{\text{sub}}(t) \frac{P_i^{\text{us}} R_i^u(t)}{r_i^{\text{us}}} + (x_{i,j,k}^{\text{mec}}(t) + x_{i,j,k}^{\text{cloud}}(t)) \frac{P_i^{\text{um}} R_i^u(t)}{r_i^{\text{um}}} \end{aligned} \quad (16)$$

移动边缘接入节点 s_j 的能耗为

$$E_j^s(t) = E_j^{s,c}(t) = \kappa (f_j(t))^3 \min \left(l, \frac{R_j^s(t)}{f_j(t)} \right) \quad (17)$$

固定边缘汇聚节点 m_k 的能耗为

$$\begin{aligned} E_k^m(t) = & l P_k^{m,c}(t) = l (\alpha P_{\text{max}} + (1 - \alpha) P_{\text{max}} \theta(t)) = \\ & l \left(\alpha P_{\text{max}} + (1 - \alpha) P_{\text{max}} \frac{R_k^m(t)}{w_{\text{max}}} \right) = \\ & l \left(\alpha P_{\text{max}} + (1 - \alpha) P_{\text{max}} \frac{\sum_{i=1}^x \sum_{j=1}^y x_{i,j,k}^{\text{mec}}(t) R_i^u(t)}{w_{\text{max}}} \right) \end{aligned} \quad (18)$$

其中, w_{max} 为 m_k 最大服务率。

因此,所有任务的总时延为

$$\begin{aligned} D_{\text{all}}(t) = & \sum_{i=1}^x D_i^u(t) = \sum_{i=1}^x x_{i,j,k}^{\text{sub}}(t) (D_i^{\text{us}}(t) + D_j^{s,c}(t)) + \\ & \sum_{i=1}^x x_{i,j,k}^{\text{mec}}(t) (D_i^{\text{um}}(t) + D_k^{m,c}(t)) + \\ & \sum_{i=1}^x x_{i,j,k}^{\text{cloud}}(t) (D_i^{\text{um}}(t) + D_k^{\text{cloud}}(t)) \end{aligned} \quad (19)$$

该优化问题的目标为在给定单个 UAV、Sub-MECS 和 M-MECS 的能量限制的情况下,最小化系统总时延成本。无人机的能耗分为固定运行能耗和无线传输能耗,其中,固定运行能耗不受卸载策略影响,而无线传输能耗受卸载策略影响较小。为了简化模型,本文假设无人机的能耗满足能耗约束。综上分析,通信和计算资源的联合优化问题可表述为

$$\text{P1 } \min_{x^s} \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} E\{D_{\text{all}}(t)\}$$

$$\text{s.t. (a) } D_i^u(t) \leq \tau_i^u(t), \forall t \in \mathbf{t}, \forall i \in U$$

$$\text{(b) } C_i^u(t) \sim P(\lambda), \forall t \in \mathbf{t}, \forall i \in U$$

$$\text{(c) } E_i^u(t) \leq E_{\text{max}}^u, \forall t \in \mathbf{t}, \forall i \in U$$

$$\text{(d) } E_j^s(t) \leq E_{\text{max}}^s, \forall t \in \mathbf{t}, \forall j \in S$$

$$\text{(e) } E_k^m(t) \leq E_{\text{max}}^m, \forall t \in \mathbf{t}, \forall k \in M$$

$$\begin{aligned}
 & \text{(f)} \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{T=0}^{T-1} E\{E_j^s(t)\} \leq \overline{E_j^s}, \quad \forall t \in \mathbf{t}, \forall j \in S \\
 & \text{(g)} \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{T=0}^{T-1} E\{E_k^m(t)\} \leq \overline{E_k^m}, \quad \forall t \in \mathbf{t}, \forall k \in M \\
 & \text{(h)} x^h(t) \in x^{\text{able}}(t), \quad \forall t \in \mathbf{t}
 \end{aligned} \tag{20}$$

其中，约束(a)表示每个时隙中每个任务的时延要求；约束(b)表示无人机任务的生成速率服从泊松分布；约束(c)、约束(d)和约束(e)表示每个时隙的 UAV、Sub-MECS 和 M-MECS 的能量消耗不能超过上限；约束(f)和约束(g)表示长期能源消耗必须小于能源供应，用于限制 Sub-MECS 和 M-MECS 长时间的总能耗；约束(h)表示任务卸载策略的可用性。

问题 P1 是一个长期优化问题，需要在所有时间段内拥有完整的离线信息，才能得到最优解。然而，准确预测未来的工作量是难以实现的。此外，不同时间段的计算卸载决策将与长期能耗交互。因此，本文提出了一种基于当前信息就能做出卸载决策的在线学习方法。

3.2 能量亏损队列

为了解决 P1 所述问题，本文利用 Lyapunov 优化理论为每个 Sub-MECS 和 M-MECS 构建一个虚拟的能量亏损队列，指导每个 Sub-MECS 和 M-MECS 的任务卸载决策遵循长期能源约束^[23]。

1) Sub-MECS 能量亏损队列

假设 Sub-MECS 的能量亏损队列集合为 $\mathbf{Q}(t) = \{Q_j(t)\}, j \in S$ 。每个 s_j 的能量亏损队列为

$$Q_j(t+1) = \max\{Q_j(t) + E_j^s(t) - \overline{E_j^s}, 0\} \tag{21}$$

其中， $E_j^s(t)$ 是 s_j 在时隙 t 内的能量消耗， $\overline{E_j^s}$ 是 s_j 的平均能量预算，当 $t=0$ 时，队列为 0，即 $Q_j(0)=0$ 。 $Q_j(t)$ 表示时隙 t 预算供应的平均能量与消耗能量的差值，当 $Q_j(t) > 0$ 时，代表目前的能量不能支持当前时隙的任务完成，会占用下一个时隙的能量，并需在下一时隙等待该时隙任务完成，由此产生的排队时延可以表示为

$$D_j^{s,q} = \frac{Q_j(t)}{\kappa(f_j)^3} \tag{22}$$

为了满足式(20)中的能量预算约束，必须保持能量亏损队列 $\mathbf{Q}(t)$ 稳定，即 $\lim_{T \rightarrow \infty} \frac{E\{\mathbf{Q}(t)\}}{T} = 0$ 。

2) M-MECS 能量亏损队列

假设 M-MECS 的能量亏损队列集合为 $\mathbf{Z}(t) = \{Z_k(t)\}, k \in M$ 。每个 m_k 的能量亏损队列为

$$Z_k(t+1) = \max\{Z_k(t) + E_k^m(t) - \overline{E_k^m}, 0\} \tag{23}$$

其中， $E_k^m(t)$ 是 m_k 在时隙 t 的能量消耗， $\overline{E_k^m}$ 是 m_k 的平均能量预算，当 $t=0$ 时，队列为 0，即 $Z_k(0)=0$ 。 $Z_k(t)$ 表示时隙 t 预算供应的能量与消耗能量的差值。为了满足式(20)中的能量预算约束，必须保持能量亏损队列 $\mathbf{Z}(t)$ 稳定，即 $\lim_{T \rightarrow \infty} \frac{E\{\mathbf{Z}(t)\}}{T} = 0$ 。

3.3 算法设计

1) Lyapunov 优化

基于 Lyapunov 优化理论，本文可以将随机优化问题转化为每个时隙内的已知问题，而不需要任何未来信息。对 $\mathbf{Q}(t)$ 和 $\mathbf{Z}(t)$ 引入 $\boldsymbol{\Theta}(t) = [\mathbf{Q}(t), \mathbf{Z}(t)]$ ，则其 Lyapunov 函数为

$$L(\boldsymbol{\Theta}(t)) \triangleq \frac{1}{2} \sum_{j=1}^y Q_j(t)^2 + \frac{1}{2} \sum_{k=1}^z Z_k(t)^2 \tag{24}$$

Lyapunov 漂移函数为

$$\Delta(\boldsymbol{\Theta}(t)) \leq \sum_{j=1}^y Q_j(t) E_j^s(t) + \sum_{k=1}^z Z_k(t) E_k^m(t) + B_1 + B_2 \tag{25}$$

其中， B_1 和 B_2 表示边界值， $B_1 = \frac{1}{2} \sum_{j=1}^y (\overline{E_{\max}^s} - \overline{E_j^s})^2 + Q_j(t) \overline{E_j^s}$ ， $B_2 = \frac{1}{2} \sum_{k=1}^z (\overline{E_{\max}^m} - \overline{E_k^m})^2 + Z_k(t) \overline{E_k^m}$ 。

根据 Lyapunov 优化理论，使用漂移加惩罚函数来平衡能耗和时延对优化结果的影响，如式(26)所示。

$$\begin{aligned}
 \Delta(\boldsymbol{\Theta}(t)) + V \sum_{i=1}^x E\{D_i^u | \boldsymbol{\Theta}(t)\} & \leq B_1 + \\
 B_2 + \sum_{i=1}^x E\{D_i^u | \boldsymbol{\Theta}(t)\} V + (1-V) & \left\{ \begin{aligned} & \sum_{j=1}^y Q_j(t) E_j^s(t) \\ & + \sum_{k=1}^z Z_k(t) E_k^m(t) | \boldsymbol{\Theta}(t) \end{aligned} \right\}
 \end{aligned} \tag{26}$$

其中， B_1 和 B_2 为常数。因此，可以将原来长期最小化问题 P1 转化为问题 P2，使每个时隙中的漂移加惩罚函数的上界最小，即

$$P2 \min_{x^h} V \sum_{i=1}^x D_i^u(t) + (1-V) \left\{ \begin{array}{l} \sum_{j=1}^y Q_j(t) E_j^s(t) \\ + \sum_{k=1}^z Z_k(t) E_k^m(t) \end{array} \right\} \quad (27)$$

s. t. (a),(b),(c),(d),(e),(f),(g),(h) in P1

本文重点关注卸载问题，因此，将 P2 的目标函数分为 2 个部分：与卸载策略相关的 AP₁ 和与卸载策略无关的 AP₂。则 P2 可以进一步表示为

$$\begin{aligned} & \min_{x^h, y} V \sum_{i=1}^x D_i^u(t) + \\ & (1-V) \left\{ \sum_{j=1}^y Q_j(t) E_j^s(t) + \sum_{k=1}^z Z_k(t) E_k^m(t) \right\} = \\ & AP_1 + AP_2 \end{aligned} \quad (28)$$

其中，AP₁ 部分直接决定任务计算卸载策略。AP₁ 计算式为

$$\begin{aligned} AP_1 = & V \sum_{i=1}^x \left\{ x_{i,j,k}^{\text{sub}}(t) (D_i^{\text{us}}(t) + D_j^{\text{s.c}}(t)) + x_{i,j,k}^{\text{mec}}(t) (D_i^{\text{um}}(t) + \right. \\ & \left. D_k^{\text{m.c}}(t)) + x_{i,j,k}^{\text{cloud}}(t) (D_i^{\text{um}}(t) + D_k^{\text{cloud}}(t)) \right\} + \\ & (1-V) \left\{ \sum_{j=1}^y \left(Q_j(t) \kappa(f_j(t))^3 \min \left(l, \frac{\sum_{i=1}^x (x_{i,j,k}^{\text{sub}}(t) R_i^u(t))}{f_j(t)} \right) \right) \right\} + \\ & (1-V) \left\{ \sum_{k=1}^z \left(Z_k(t) l (1-\alpha) P_{\max} \frac{\sum_{i=1}^x (x_{i,j,k}^{\text{mec}}(t) R_i^u(t))}{W_{\max}} \right) \right\} \end{aligned} \quad (29)$$

而 AP₂ 部分间接给卸载决策带来长期影响。AP₂ 计算式为

$$AP_2 = (1-V) \sum_{k=1}^z (Z_k(t) l \alpha P_{\max}) \quad (30)$$

本文主要关注与卸载相关的 AP₁ 部分，因此，P2 的卸载问题可转化为

$$\begin{aligned} P3 \min_{x^h} & V \sum_{i=1}^x \left\{ x_{i,j,k}^{\text{sub}}(t) (D_i^{\text{us}}(t) + D_j^{\text{s.c}}(t)) + x_{i,j,k}^{\text{mec}}(t) (D_i^{\text{um}}(t) + \right. \\ & \left. D_k^{\text{m.c}}(t)) + x_{i,j,k}^{\text{cloud}}(t) (D_i^{\text{um}}(t) + D_k^{\text{cloud}}(t)) \right\} + \\ & (1-V) \left\{ \sum_{j=1}^y \left(Q_j(t) \kappa(f_j(t))^3 \min \left(l, \frac{\sum_{i=1}^x (x_{i,j,k}^{\text{sub}}(t) R_i^u(t))}{f_j(t)} \right) \right) \right\} + \end{aligned}$$

$$\begin{aligned} & (1-V) \left\{ \sum_{k=1}^z \left(Z_k(t) l (1-\alpha) P_{\max} \frac{\sum_{i=1}^x (x_{i,j,k}^{\text{mec}}(t) R_i^u(t))}{W_{\max}} \right) \right\} \\ & \text{s. t. (a),(b),(c),(d),(e),(f),(g),(h) in P1} \end{aligned} \quad (31)$$

与文献[24]相似，当 $x_{i,j}^{\text{sub}} = 1$ 时，P3 可以表示为

$$\begin{aligned} & \text{Lyp}_i^{\text{sub}} = V(D_i^{\text{us}}(t) + D_j^{\text{s.c}}(t)) + \\ & (1-V) \left\{ \sum_{j=1}^y \left(Q_j(t) \kappa(f_j(t))^3 \min \left(l, \frac{R_i^u(t)}{f_j(t)} \right) \right) \right\} \end{aligned} \quad (32)$$

当 $x_{i,j,k}^{\text{mec}} = 1$ 时，P3 可以表示为

$$\begin{aligned} & \text{Lyp}_i^{\text{mec}} = V(D_i^{\text{um}}(t) + D_k^{\text{m.c}}(t)) + \\ & (1-V) \left\{ \sum_{k=1}^z \left(Z_k(t) l (1-\alpha) P_{\max} \frac{R_i^u(t)}{W_{\max}} \right) \right\} \end{aligned} \quad (33)$$

当 $x_{i,j,k}^{\text{cloud}} = 1$ 时，P3 可以表示为

$$\text{Lyp}_i^{\text{cloud}} = V(D_i^{\text{um}}(t) + D_k^{\text{cloud}}(t)) \quad (34)$$

进而，问题 P3 可以转换为求解 $\text{Lyp}_i^{\text{all}}$ 的最小值，即

$$\begin{aligned} P4 \min & \sum_{i=1}^x \left\{ x_{i,j,k}^{\text{sub}} \text{Lyp}_i^{\text{sub}} + x_{i,j,k}^{\text{mec}} \text{Lyp}_i^{\text{mec}} + x_{i,j,k}^{\text{cloud}} \text{Lyp}_i^{\text{cloud}} \right\} \\ & \text{s. t. (a),(b),(c),(d),(e),(f),(g),(h) in P1} \end{aligned} \quad (35)$$

由分析可知，P4 是一个时延和能耗联合优化问题，与背包问题类似，其可行集和目标函数是非凸的，因此 P4 是一个 NP 问题，且当 Sub-MECS 和 M-MECS 个数增加时，P4 问题规模会迅速增大。由于使用传统算法求解背包问题存在时间复杂度分析困难的问题，因此本文针对上述系统模型，采用人工智能算法进行求解，提出了一种近端任务卸载策略优化 (PTOPO, proximal task offload policy optimization) 机制，可以规范高效地求解卸载策略。

2) PTOPO 机制

本文基于经典的 PPO 强化学习算法^[25-26]，设计了 PTOPO 机制，求解最优的卸载策略。如图 3 所示，整个电力巡检应用场景包括 3 个部分：环境、个体和动作。个体与环境进行交互，从一个状态出发，根据自己的策略分布选择动作，并获得奖励。环境由被巡检的电力场景物理设备组成，为个体提供环境状态信息。个体可以根据状态做出不同的动作，施加在环境上，并计算出相应的奖赏，反馈给个体，然后进行卸

载操作。

S 表示有限的状态空间,在本文中用来表示每个时隙无人机的任务数量大小和时延要求。

A 表示有限的工作空间,在本文中被定义为无人机任务的卸载向量 $X_i = \{x_{i,j}^{sub}, x_{i,j,k}^{mec}, x_{i,j,k}^{cloud}\}$ 。假设之前的动作策略为 $\{0,0,1\}$,表示任务卸载到云端,但随着个体与环境的交互,状态发生了改变,则任务有可能卸载到 Sub-MECS 或 M-MECS 上,即动作策略变为 $\{1,0,0\}$ 或 $\{0,1,0\}$ 。

R 表示在当前时隙的状态和动作对 (S, A) 出现后,根据 Lyp_i^{all} 定义获得的值的变化,即

$$R(s_t, a_t) = Lyp_i^{all,before} - Lyp_i^{all,after} \quad (36)$$

随着迭代的累积,系统可以收敛到最佳状态,在该状态下,所有 Lyp_i^{all} 值都不会改变并保持在最小值。在通过应用不同的状态-动作对来最大化累积奖励的过程中,使 PTOPO 的卸载策略接近最优。

PPO 算法是基于 Actor-Critic 结构的强化学习算法,其基本思想是设定策略函数和行为价值函数近似化策略梯度,策略函数可以基于价值函数进行策略评估和优化,优化的策略函数也能使价值函数更加准确地反映状态的价值,相互影响求得最优解^[27]。

PPO 算法的目标函数为

$$\max E_t \left[\frac{\pi(a_t | s_t)}{\pi_{old}(a_t | s_t)} \hat{A}_{\pi_{old}}(s_t, a_t) \right] \quad (37)$$

其中, $\hat{A}_{\pi_{old}}(s_t, a_t) = \sum_{t'>t} \gamma^{t'-t} r_{t'} - V_{\pi_{old}}(s_t)$ 为对优势函数的估计, $\pi(a_t | s_t)$ 为基于新策略状态 s_t 采取动作 a_t 的概率, $\pi_{old}(a_t | s_t)$ 为基于旧策略状态 s_t 采取动作 a_t 的概率。PPO 算法采用重要性采样定理来约束新策略的概率分布与旧策略的概率分布情况相似度较高,该约束条件可等价于 $\frac{\pi(a_t | s_t)}{\pi_{old}(a_t | s_t)}$ 的值接近 1。因此, PPO 算法新的目标函数为^[28]

$$L_t = E_t \left[\min \left(\frac{\pi(a_t | s_t)}{\pi_{old}(a_t | s_t)} \hat{A}_{\pi_{old}}(s_t, a_t), \text{clip} \left(\frac{\pi(a_t | s_t)}{\pi_{old}(a_t | s_t)}, 1 - \epsilon, 1 + \epsilon \right) \right) \right] \quad (38)$$

式(38)中目标函数为算法规定了上下界约束,使 $\frac{\pi(a_t | s_t)}{\pi_{old}(a_t | s_t)}$ 被约束在 $[1 - \epsilon, 1 + \epsilon]$, 防止算法策略更新太快。

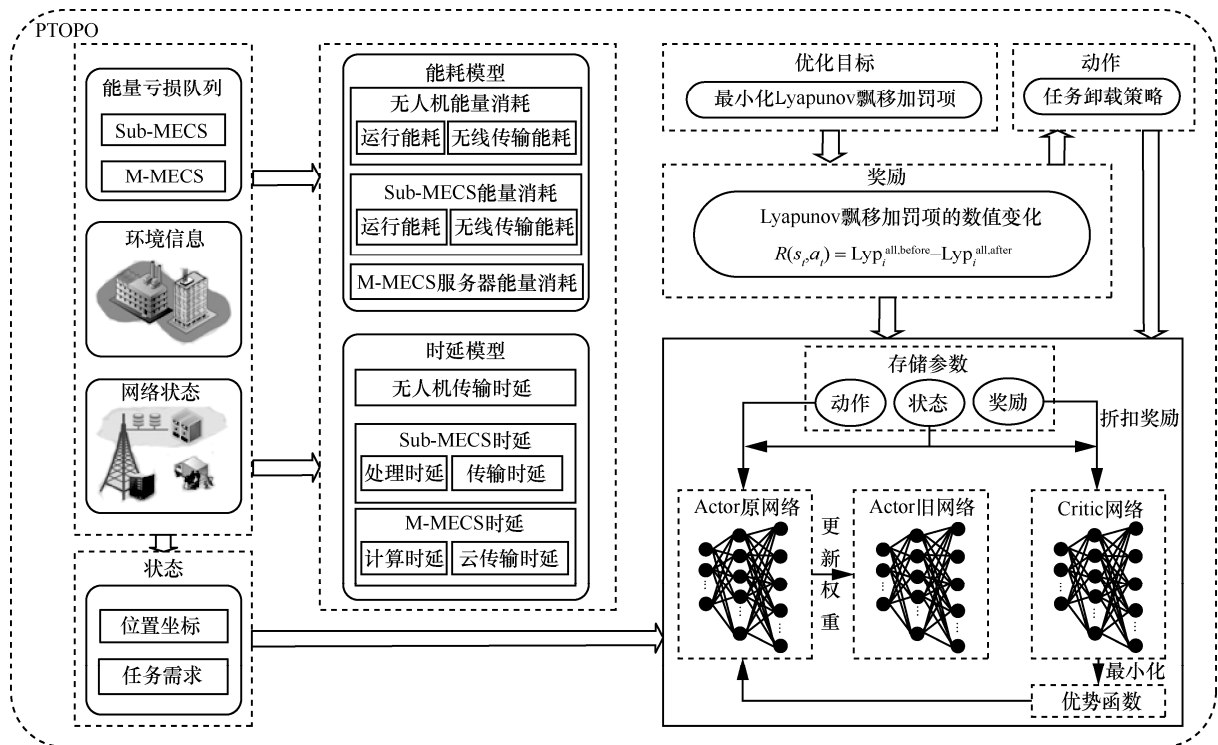


图3 PTOPO 机制工作原理

PTOPO 机制流程如算法 1 所示。

算法 1 近端任务卸载策略优化机制

输入 $O(\mathbf{0})=0$, $Z(\mathbf{0})=0$, 权重系数 V 和

M-MECS 功耗比例系数 α

输出 卸载策略 X_i

① 初始化 UAV、Sub-MECS 和 M-MECS, 随机初始化系统状态和个体策略网络 π 和 π_{old} ;

② for big_episode=1, N do

③ 更新每个 Sub-MECS 位置和每个 Sub-MECS 链接的 M-MECS;

④ for small_episode=1, ξ do

⑤ 根据目前的状态选择动作, 执行动作;

⑥ 通过式(36)计算奖励, 环境生成 UAV 的下一个状态;

⑦ 将当前状态、动作和奖励存入记忆池中, 并更新每个 UAV 的位置和每个 UAV 链接的 Sub-MECS;

⑧ end for

⑨ 根据 $\hat{A}_{\pi_{old}}(s_t, a_t) = \sum_{t'>t} \gamma^{t'-t} r_{t'} - V_{\pi_{old}}(s_t)$ 计算

优势函数, 并反向传播更新 Critic 网络;

⑩ for $i=1, K$ do

⑪ 根据式(37)更新 Actor 网络

⑫ end for

⑬ $\pi_{old} \leftarrow \pi$

⑭ end for

⑮ return X_i

3) PTOPO 复杂度分析

在每个决策周期中, 学习阶段在每个步骤通过执行 PTOPO 得到最优卸载策略。这个过程重复进行, 并更新状态和动作。在学习阶段结束时, 个体从环境中获得最大回报, 并由此确定了全局最优策略。由分析可知, 算法的复杂性取决于 PTOPO 的迭代次数, 因此, PTOPO 的算法复杂度为 $O(n^2)$ 。

4 仿真分析

4.1 系统参数设置

本文在 Python3.7.6 和 TensorFlow1.14.0 环境下对 PTOPO 机制进行了仿真实验。假设巡检场景为 $100 \text{ km} \times 100 \text{ km}$ 的正方形区域, 随机分布具有数据采集和发送功能的 UAV, 数量为 50 个。根据实际应用场景, 每个 Sub-MECS 可对应 3~4 个 UAV,

且同一时间同一区域只有一个 UAV 执行任务。UAV 任务的生成速率服从泊松分布, 到达率为每秒 0~10 个单位任务不等, 其中单位任务的数据大小为 0.2 MB。任务的数据大小是单位任务的倍数。因此, 对于典型的 100 MB 以太网, 单位任务的传输时延期望为 $\tau=200 \text{ ms}$ 。信道带宽为 20 MHz。信道增益分布为平均值 $g_0(1/100)^4$, 其中 $g_0 = -30 \text{ dB}$ 是 1 m 的路径损耗常数。假设噪声功率为 $\sigma^2=10^{-10} \text{ W/Hz}$ 。

为了验证本文提出的 PTOPO 卸载机制的性能, 将其与基于随机算法 (RA, random algorithm)、模拟退火算法 (SAA, simulated annealing algorithm)、Q-learning 算法和 A3C 算法的卸载机制进行对比, 下面简要介绍各对比算法的运行机制。

1) RA: 在满足约束条件的情况下, 随机进行任务卸载。

2) SAA^[29]: 基于 Monte-Carlo 迭代求解策略的一种随机寻优算法。该算法基于固体退火原理, 将内能模拟成优化目标函数值, 设置初始温度和初始解, 经过多次迭代和降温, 直到满足终止条件, 算法结束。

3) Q-learning^[30]: 强化学习中一种基于价值的算法。通过将 State 与 Action 构建一张 Q-table 来存储 Q 值, 然后根据 Q 值来选取能够获得最大收益的动作。

4) A3C^[31]: 利用 Actor 网络计算动作策略, 并利用 Critic 网络计算策略优劣, 采用多线程计算方法, 将 Actor-Critic 网络放置在多个线程里分别和环境进行交互; 线程间数据共享, 并在经过一段时间学习后, 根据先前的学习经验指导后续的学习和交互。

4.2 仿真结果分析

本文将提出的 PTOPO 卸载机制与对比算法进行比较, 考察的参数包括时延和能耗等性能指标。本文的仿真性能为 50 次仿真实验的平均结果。

图 4 和图 5 展示了实际应用中主要关注的 2 个指标: 系统时延和系统能耗。其中, RA 和 SAA 因其平均时延和平均能耗不满足任务要求, 不适用于无人机巡检系统的实际应用场景。同时, 与 Q-learning 算法和 A3C 算法相比, 本文提出的 PTOPO 机制具有较低的系统能耗和系统时延, 这是由于它不仅优化了云边端多个层级之间及同层节点间的计算卸载, 而且遵循了每个节点的长期能量约束。

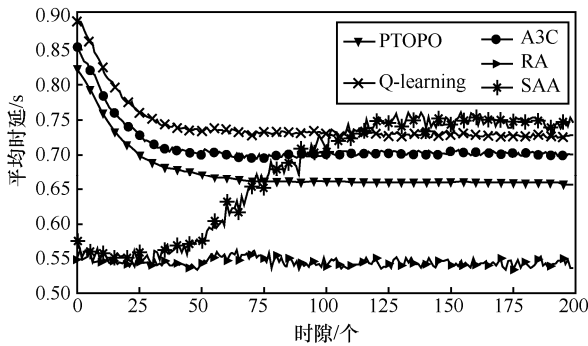


图 4 系统时延

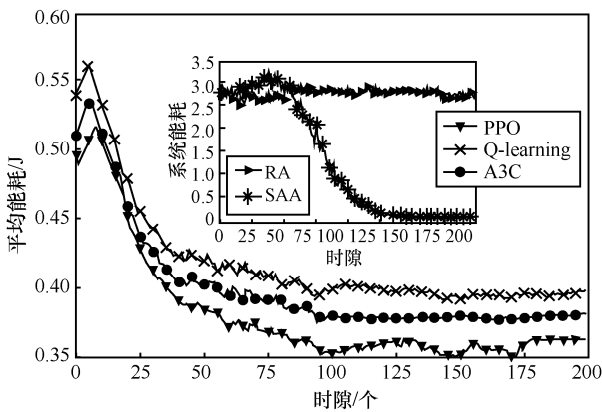


图 5 系统能耗

对于不同的巡检任务需求，Sub-MECS 和 UAV 的链接比例也会不同。图 6 展示了 Sub-MECS 和 UAV 链接比例对任务平均时延的影响。从图 6 可以看出，随着 Sub-MECS 连接的 UAV 数量增多，整个巡检系统的业务平均时延将逐渐增加。这是由于当 Sub-MECS、M-MECS 和云服务器的计算能力保持不变时，不断增加的任务工作量会造成任务溢出，进而导致时延增加。本文提出的 PTOPO 机制可在一定链接比例内保证任务在时延要求内处理完成。其中，变电站巡检对任务时延要求较高，当 Sub-MECS 和 UAV 的链接比例越接近 1:1 时，系统时延越低，符合变电站巡检要求。电塔和导线巡检任务对时延要求较低，链接比例可适当调整至远离 1:1，不但可以满足其时延要求，还可以扩大巡检范围，节约经济成本。

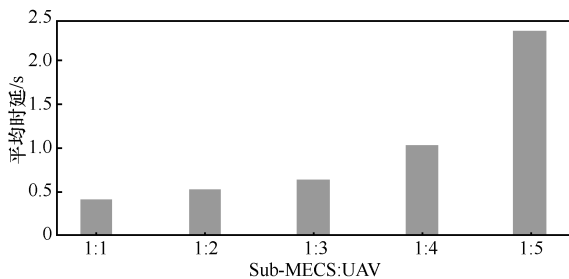


图 6 Sub-MECS 和 UAV 链接比例对任务平均时延的影响

如图 7 和图 8 所示，对于场景中不同数量的 UAV，其能耗收敛率和时延收敛率会随着 UAV 数量的增多而提高并趋于稳定。这是因为 UAV 数量的增多意味着场景中 UAV 分布越加均匀，上级 Sub-MECS 和 M-MECS 所连接的 UAV 个数趋于稳定，任务过载现象明显减少。因此，合适的 UAV 数量和链接比例对提高巡检任务的运行效率具有重要作用。

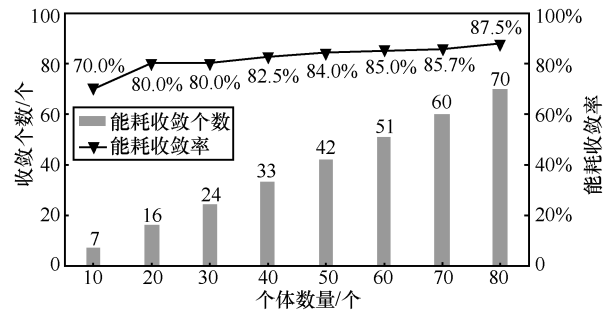


图 7 能耗收敛率与 UAV 数量的影响关系

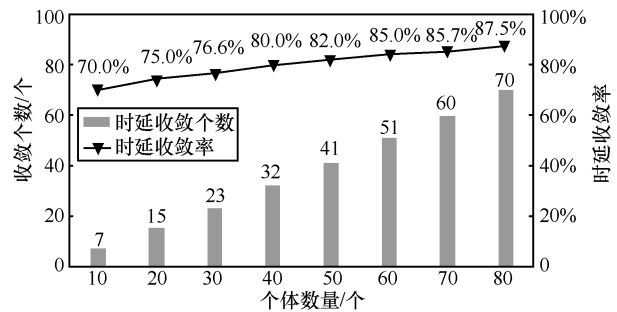


图 8 时延收敛率与 UAV 数量的影响关系

图 9 展示了巡检场景的能量和时延成本随参数 V 变化的性能。算法通过调整引入的参数 V 实现了系统时延成本和长期能耗之间的平衡，结果与文献[32]的理论分析一致。随着 V 的增大，算法对时延的限制要求逐渐提高，对能耗的限制要求逐渐降低，更适用于对时延要求较高的变电站巡检业务等巡检场景。当 V 趋近 1 时，算法实现了最小的时延成本，但是能耗较大。当减小 V 的取值时，算法对能耗的

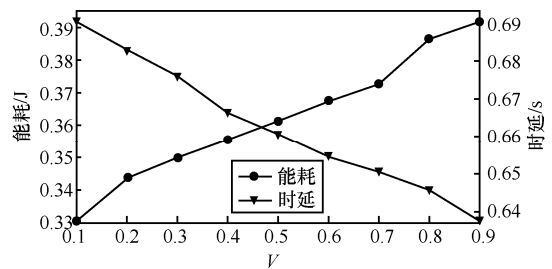


图 9 不同 V 对时延和能耗的影响

限制要求逐渐提高, 对时延的限制要求逐渐降低, 更适用于对能耗要求较高的电塔巡检和导线巡检等巡检场景。所以, 针对不同的电力巡检场景, 可通过找到合适的 V 值以获得相对较低的综合成本。

5 结束语

在能源互联网中, 利用无人机能够高效完成大范围电路巡检, 并减少运维风险。需基于终端变化的行动轨迹和有限的边缘资源, 研究合理的任务卸载方法, 在满足多种巡检任务需求的条件下, 优化系统消耗和服务时延。因此, 本文利用边缘计算和人工智能技术建立了基于双层边缘网络的云边端网络架构, 并结合 PPO 强化学习算法和 Lyapunov 优化理论建立了可联合优化通信和计算资源的任务卸载机制。仿真结果表明, 与传统的 RA 和 SAA 相比, 本文提出的 PTOPO 机制在保持相对较低时延的同时, 更适用于对能耗要求高的输电线路巡检场景; 与基于 Q-learning 算法的混合控制机制和基于 A3C 算法的异步优势卸载机制相比, 本文提出的 PTOPO 机制在节省能量消耗、减少处理时间和保证服务质量方面均具有优势, 其长期平均能耗分别提升了 10% 和 6%。在未来的工作中, 将引入流量预测模型和任务迁移机制, 设计离线和在线相结合的计算任务卸载方法, 进一步提高服务效率。

参考文献:

- [1] 李君海, 张苗苗, 熊道洋. 基于实时信息传输技术的无人机巡检管控平台[J]. 测绘与空间地理信息, 2020, 43(6): 165-167,171.
LI J H, ZHANG M M, XIONG D Y. An UAV power inspection management platform based on real-time message transmission technology[J]. Geomatics & Spatial Information Technology, 2020, 43(6): 165-167,171.
- [2] 陈兰波. 电力线路无人机巡检方案研究[J]. 科技与创新, 2020(11): 36-38,41.
CHEN L B. Research on UAV patrol scheme of power line[J]. Science and Technology & Innovation, 2020(11): 36-38,41.
- [3] 严波, 林世忠, 张振威, 等. 无人机电力巡检技术应用分析[J]. 自动化应用, 2019(12): 155-156.
YAN B, LIN S Z, ZHANG Z W, et al. Application analysis of UAV power inspection technology[J]. Automation Application, 2019(12): 155-156.
- [4] HAN D S, LI S J, CHEN Z X. Hybrid energy ratio allocation algorithm in a multi-base-station collaboration system[J]. IEEE Access, 2019, 7: 147001-147009.
- [5] JEONG S, SIMEONE O, KANG J. Mobile edge computing via a UAV-mounted cloudlet: optimization of bit allocation and path planning[J]. IEEE Transactions on Vehicular Technology, 2018, 67(3): 2049-2063.
- [6] WU Q Q, ZENG Y, ZHANG R. Joint trajectory and communication design for multi-UAV enabled wireless networks[J]. IEEE Transactions on Wireless Communications, 2018, 17(3): 2109-2121.
- [7] WEI X L, TANG C G, FAN J H, et al. Joint optimization of energy consumption and delay in cloud-to-thing continuum[J]. IEEE Internet of Things Journal, 2019, 6(2): 2325-2337.
- [8] HU X Y, WONG K K, YANG K, et al. UAV-assisted relaying and edge computing: scheduling and trajectory optimization[J]. IEEE Transactions on Wireless Communications, 2019, 18(10): 4738-4752.
- [9] HUA M, WANG Y, ZHANG Z M, et al. Power-efficient communication in UAV-aided wireless sensor networks[J]. IEEE Communications Letters, 2018, 22(6): 1264-1267.
- [10] ZHANG T K, 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.
- [11] ATEYA A A, MUTHANNA A, KIRICHEK R, et al. Energy-and latency-aware hybrid offloading algorithm for UAVs[J]. IEEE Access, 2019, 7: 37587-37600.
- [12] 王立科. 无人机技术在电力巡检信息化管理中的应用研究[J]. 机电信息, 2019(27): 88-89.
WANG L K. Research on application of UAV technology in information management of power patrol inspection[J]. Mechanical and Electrical Information, 2019(27): 88-89.
- [13] 马青岷. 无人机电力巡检及三维模型重建技术研究[D]. 济南: 山东大学, 2017.
MA Q M. Research of powerline inspection based on UAV and 3D model technology[D]. Jinan: Shandong University, 2017.
- [14] SHUO W, XING Z, ZHI Y, et al. Cooperative edge computing with sleep control under nonuniform traffic in mobile edge networks[J]. IEEE Internet of Things Journal, 2019, 6(3): 4295-4306.
- [15] SHAW J A. Radiometry and the Friis transmission equation[J]. American Journal of Physics, 2013, 81(1): 33-37.
- [16] DOWNEY C. Understanding wireless range calculations[J]. Electronic Design, 2013.
- [17] MAO Y Y, ZHANG J, SONG S H, et al. Power-delay tradeoff in multi-user mobile-edge computing systems[C]//2016 IEEE Global Communications Conference. Piscataway: IEEE Press, 2016: 1-6.
- [18] BURD T D, BRODERSEN R W. Processor design for portable systems[J]. Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology, 1996, 13(2/3): 203-221.
- [19] FAN X B, WEBER W D, BARROSO L A. Power provisioning for a warehouse-sized computer[J]. ACM SIGARCH Computer Architecture News, 2007, 35(2): 13-23.
- [20] CHEN L X, ZHOU S, XU J. Computation peer offloading for energy-constrained mobile edge computing in small-cell networks[J]. IEEE/ACM Transactions on Networking, 2018, 26(4): 1619-1632.
- [21] YANG L, CAO J N, TANG S J, et al. A framework for partitioning and execution of data stream applications in mobile cloud computing[C]//2012 IEEE Fifth International Conference on Cloud Computing. Piscataway: IEEE Press, 2012: 794-802.
- [22] CHEN L X, XU J, ZHOU S. Computation peer offloading in mobile edge computing with energy budgets[C]//GLOBECOM 2017 - 2017 IEEE Global Communications Conference. Piscataway: IEEE Press, 2017: 1-6.
- [23] NEELY J M. Stochastic network optimization with application to

communication and queueing systems[J]. Synthesis Lectures on Communication Networks, 2010, 3(1): 211.

- [24] XU S Y, LIU Q C, GONG B, et al. RJCC: reinforcement-learning-based joint communicational-and-computational resource allocation mechanism for smart city IoT[J]. IEEE Internet of Things Journal, 2020, 7(9): 8059-8076.
- [25] HEES N, DHARUVA T B, SRIRAM S, et al. Emergence of locomotion behaviours in rich environments[J]. arXiv Preprint, arXiv: 1707.02286, 2017.
- [26] SCHULMAN J, WOLSKI F, DHARIWAL P, et al. Proximal policy optimization algorithms[J]. arXiv Preprint, arXiv: 1707.06347, 2017.
- [27] 王志夫. 基于深度强化学习的双足机器人步行运动控制[D]. 济南: 山东大学, 2020.
WANG Z F. Deep reinforcement learning based walking control of biped robot[D]. Jinan: Shandong University, 2020.
- [28] 王鸿涛. 基于强化学习的机械臂自主学习控制[D]. 哈尔滨: 哈尔滨工业大学, 2019.
WANG H T. Self learning control of mechanical arm based on reinforcement learning[D]. Harbin: Harbin Institute of Technology, 2019.
- [29] 胡彦娟. 移动边缘计算中任务卸载及资源分配算法研究[D]. 重庆: 重庆邮电大学, 2020.
HU Y J. Research on task offloading and resource allocation algorithm in mobile edge computing[D]. Chongqing: Chongqing University of Posts and Telecommunications, 2020.
- [30] GAO Z B, WEN B, HUANG L F, et al. Q-learning-based power control for LTE enterprise femtocell networks[J]. IEEE Systems Journal, 2017, 11(4): 2699-2707.
- [31] LI R, ZHANG C Y, PATRAS P, et al. Learning driven mobility control of airborne base stations in emergency networks[J]. ACM SIGMETRICS Performance Evaluation Review, 2019, 46(3): 163-166.
- [32] KWAK J, KIM Y, LEE J, et al. DREAM: dynamic resource and task allocation for energy minimization in mobile cloud systems[J]. IEEE Journal on Selected Areas in Communications, 2015, 33(12): 2510-2523.

[作者简介]



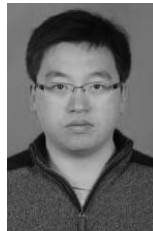
徐思雅 (1988-), 女, 北京人, 博士, 北京邮电大学讲师, 主要研究方向为信息网络管理、SDN/NFV、移动边缘计算和人工智能等。



邢逸斐 (1997-), 男, 河北保定人, 北京邮电大学硕士生, 主要研究方向为智能电网网络管理、移动边缘计算等。



郭少勇 (1985-), 男, 河北邢台人, 博士, 北京邮电大学副教授, 主要研究方向为区块链、物联网等。



杨超 (1988-), 男, 山东平度人, 博士, 国网辽宁省电力有限公司工程师, 主要研究方向为电力物联网、人工智能工程化应用等。



邱雪松 (1973-), 男, 江西上饶人, 博士, 北京邮电大学教授、博士生导师, 主要研究方向为网络与业务管理、物联网与区块链。



孟洛明 (1955-), 男, 河南洛阳人, 博士, 北京邮电大学教授、博士生导师, 主要研究方向为通信网、网络管理、通信软件等。