

分类号: \_\_\_\_\_

单位代码: 10300

密 级: \_\_\_\_\_

学 号: 20191220033

# 南京信息工程大学

## 硕 士 学 位 论 文



论文题目: 基于无人机的移动边缘计算任务卸载策略研究

申请人姓名: 余世明

指 导 教 师: 苏健

学 科 名 称: 计算机科学与技术

研 究 方 向: 移动边缘计算

培 养 学 院: 计算机学院、软件学院、  
网络空间安全学院

提 交 时 间: 2022 年 6 月 10 日

二〇二二 年 六 月

## 独创性声明

本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。本论文尽我所知，除了文中特别加以标注和致谢的内容外，不包含其他人或其他机构已经发表或撰写过的研究成果，也不包含为获得南京信息工程大学或其他教育机构的学位或证书而使用过的材料。与我一同工作同志对本研究所做的贡献均已在论文中作了声明并表示谢意。

本人学位论文及涉及相关资料若有不实，愿意承担一切相关的法律责任。

学位论文作者签名：余世朋

签字日期：2022.6.10

## 关于论文使用授权的说明

本人授权南京信息工程大学可以保留并向国家有关部门或机构送交论文的复印件和电子文档；允许论文被查阅和借阅；可以将学位论文的全部或部分内容编入有关数据库进行检索；可以采用影印、缩印或扫描等复制手段保存、汇编本学位论文。本文电子文档的内容和纸质论文的内容相一致。论文的公布（包括刊登）授权南京信息工程大学研究生院办理。

非国家秘密类涉密学位论文在解密后适用本授权书。

☒公开    ☐保密（\_\_\_\_年 \_\_\_\_月）（保密的学位论文在解密后应遵守此协议）

学位论文作者签名：余世朋

签字日期：2022.6.10

指导教师签名：苏健

签字日期：2022.6.10

# 目 录

摘 要.....	I
Abstract.....	II
第一章 绪论.....	1
1.1 研究背景和意义.....	1
1.2 国内外研究现状与问题分析.....	2
1.2.1 移动边缘计算网络下任务卸载策略研究.....	3
1.2.2 无人机辅助移动边缘计算网络下任务卸载策略研究.....	4
1.2.3 问题分析.....	6
1.3 主要研究思路与内容.....	6
1.4 论文组织结构.....	8
第二章 相关技术概述.....	10
2.1 移动边缘计算技术.....	10
2.2 凸优化相关理论.....	11
2.2.1 凸优化概述.....	11
2.2.2 交替方向乘子法概述.....	12
2.3 强化学习相关理论.....	13
2.4 本章小结.....	15
第三章 基于单无人机辅助移动边缘计算的任务卸载策略研究 .....	16
3.1 问题描述与分析.....	16
3.2 系统模型.....	17
3.2.1 网络模型.....	17
3.2.2 通信模型.....	18
3.2.3 计算模型.....	18
3.3 优化问题描述.....	20
3.4 优化问题求解.....	21

3.4.1 优化问题重构 .....	21
3.4.2 基于 ADMM 算法的任务卸载策略 .....	22
3.4.3 复杂度分析 .....	26
3.5 仿真实验与结果分析 .....	27
3.5.1 仿真描述与参数配置 .....	27
3.5.2 仿真结果分析 .....	27
3.6 本章小结 .....	31
<b>第四章 基于多无人机辅助移动边缘计算的任务卸载策略研究 .....</b>	<b>33</b>
4.1 问题描述与分析 .....	33
4.2 系统模型 .....	34
4.2.1 网络模型 .....	34
4.2.2 通信模型 .....	35
4.2.3 计算模型 .....	35
4.2.4 飞行模型 .....	37
4.3 优化问题描述 .....	38
4.4 优化问题求解 .....	38
4.4.1 基于 A3C 算法的任务卸载策略 .....	39
4.4.2 生成任务卸载决策 .....	41
4.5 仿真实验与结果分析 .....	41
4.5.1 仿真描述与参数配置 .....	41
4.5.2 仿真结果分析 .....	42
4.6 本章小结 .....	46
<b>第五章 总结与展望 .....</b>	<b>47</b>
5.1 工作总结 .....	47
5.2 研究展望 .....	47
<b>参考文献 .....</b>	<b>49</b>
<b>致谢 .....</b>	<b>55</b>
<b>作者简介 .....</b>	<b>56</b>

## 摘 要

随着移动通信技术的迭代更新，物联网应用逐渐多样化、个性化，人们的日常生活与物联网设备的联系越来越紧密。但是，物联网设备受物理体积影响，计算资源和存储的能量有限，使物联网的发展迎来了挑战。移动边缘计算的适时出现，让物联网用户将计算任务卸载至网络边缘计算节点运行，成为降低时延和减少能耗的有效方案。

然而，位置固定、部署成本高昂的地面基站限制了移动边缘计算的发展。与此同时，机动性强、成本低的无人机与移动边缘计算的联合具有重要的研究意义和应用价值。本文针对现有移动边缘计算中存在的过于依赖地面基站、无人机相关约束、地面设备移动性等挑战，展开以下两方面的研究：

(1) 针对传统移动边缘计算过于依赖地面基站问题，本文提出了一种基于单无人机辅助移动边缘计算的任务卸载策略。首先，在单无人机辅助移动边缘计算网络中，考虑到计算时延、计算资源和无人机能量均有限的条件下，以最小化地面用户设备总能耗为目标，构建任务卸载决策与无人机位置联合优化问题。同时，本文提出了一种基于交替方向乘子法（Alternating Direction Method of Multiplies, ADMM）算法的任务卸载策略。数值结果表明，所提出的基于 ADMM 算法的任务卸载策略能够有效降低地面用户设备总能耗。

(2) 由于第一部分工作中单无人机辅助移动边缘计算的应用场景有限，因此本文进一步研究了基于多无人机辅助移动边缘计算的任务卸载策略。本文将第一部分网络模型中的地面基站舍弃，将单无人机拓展为多无人机，同时引入地面设备的移动特性。在时延和无人机相关约束的限制下，将问题构建为任务卸载决策与无人机位置联合优化问题，实现地面移动设备总能耗最小化的目标。同时，本文提出了一种基于异步优势行动者-评论家（Asynchronous Advantage Actor-Critic, A3C）算法的任务卸载策略。数值结果表明，本文提出的任务卸载策略比传统算法更具优势，能够有效减少地面移动设备总能耗。

**关键词：**移动边缘计算，无人机，任务卸载，联合优化

## Abstract

With the iterative update of mobile communication technology, Internet of Things (IoT) applications are gradually diversified and personalized, and people's daily life is more and more closely connected with IoT devices. However, IoT devices are affected by physical size, and the computing resources and stored energy are limited, which brings challenges to the development of IoT. The timely emergence of mobile edge computing allows IoT users to offload computing tasks to the network edge computing nodes to run as an effective solution to reduce latency and reduce energy consumption.

However, the ground base stations (BSs) with fixed locations and high deployment costs limit the development of mobile edge computing. Meanwhile, the combination of unmanned aerial vehicle (UAV) with strong mobility and low cost and mobile edge computing has important research significance and application value. Aiming at the challenges existing in the existing mobile edge computing, such as over-reliance on ground BSs, UAV-related constraints, and ground device mobility, this paper carries out the following two aspects of research:

(1) To address the problem that traditional mobile edge computing is too dependent on ground BSs, this paper proposes a task offloading strategy based on single UAV-assisted mobile edge computing. First, in the single UAV-assisted mobile edge computing network, a joint task offloading decision and UAV location optimization problem is constructed with the objective of minimizing the total energy consumption of ground user devices, considering the limited computational delay, computational resources and UAV energy. Meanwhile, this paper proposes a task offloading strategy based on the alternating direction method of multipliers (ADMM) algorithm. Numerical results show that the proposed task offloading strategy based on ADMM algorithm can effectively reduce the total energy consumption of ground user devices.

(2) Since the application scenario of single UAV-assisted mobile edge computing in the first part of the work is limited, this paper further investigates the task offloading strategy based on multi-UAV-assisted mobile edge computing. In this paper, the ground BS in the first part of the network model is discarded and the single UAV is expanded into multiple UAVs, while the mobile characteristics of ground devices are introduced. Under the constraints of time delay

and UAV-related constraints, the problem is constructed as a joint optimization problem of task offloading decision and UAV position to achieve the goal of minimizing the total energy consumption of ground mobile devices. Meanwhile, this paper proposes a task offloading strategy based on the asynchronous advantage actor-critic (A3C) algorithm. Numerical results show that the task offloading strategy proposed in this paper has more advantages than the traditional algorithm, and can effectively reduce the total energy consumption of ground mobile devices.

**Key words: mobile edge computing, unmanned aerial vehicles, task offloading, joint optimization**

## 第一章 绪论

### 1.1 研究背景和意义

移动通信技术每一次代际跃迁，每一次技术迭代，都极大地促进了社会产业升级和经济发展<sup>[1][2]</sup>。从 1G（the First Generation, 1G）到 2G（the Second Generation, 2G），我们从模拟通信时代进入数字通信时代，移动通信进入人们的日常生活。从 2G 到 3G（the Third Generation, 3G）、4G（the Fourth Generation, 4G），我们从语音时代进入数据时代，移动互联网应用蓬勃发展。随着移动互联网的迅猛发展，新型服务、新型业务不断涌现，用户移动数据流量呈爆炸式增长，5G（the Fifth Generation, 5G）技术应运而生，物联网时代来临<sup>[3][4]</sup>。

物联网即“万物相连的互联网”，是现代无线通信领域迅速发展起来的一种新模式，通过特殊的寻址方式，人、机、物相互交流，相互合作，以实现共同目标<sup>[5]</sup>。随着万物互联的进程不断推进，车联网、智能家居、可穿戴设备等物联网设备的数量不断增加<sup>[6]</sup>，网络数据传输能力无法匹配数据流量的增长速度<sup>[7]</sup>。同时，增强现实、虚拟现实、移动医疗等新型物联网服务逐步渗透到我们的日常生活，这些时延敏感的新型应用向物联网技术的发展提出了挑战。某些物联网设备（如监控摄像机、能量传感器、智能手表）受其物理体积的影响，通常存储能量有限、计算能力低下，所以计算能力有限和带宽压力也是物联网技术发展亟待考虑的问题<sup>[8][9][10]</sup>。

为了促进物联网的发展，专家学者们提出了云计算的概念<sup>[11][12][13]</sup>。利用云计算集中式处理数据的运行模式，云服务行业获得了较高的经济收益<sup>[14]</sup>。但是在物联网蓬勃发展的背景下，用户对智能终端设备的期许也越来越高，小巧轻便的智能设备无法提供处理爆炸式增长的数据的能力，云计算暴露出在实时性反馈、能耗、数据安全隐私等方面隐含的不足，无法提供更优质的服务。为了改进云计算的不足之处，面向处理网络边缘物联网设备产生的数据的移动边缘计算模式应运而生<sup>[15][16]</sup>。

移动边缘计算是一种就近为无线接入网络覆盖范围内的物联网设备提供云端计算能力和服务<sup>[17][18]</sup>，实现低时延、高带宽、高性能优质服务环境的新型计算模式，其致力于降低时延和计算能耗。移动边缘计算模式的出现极大地缓解了网络带宽和云计算中心的压力，优化了计算、存储服务的响应能力，降低了用户数据隐私泄露的隐患<sup>[19][20][21]</sup>。随着国内外专家学者们在移动边缘计算领域的研究，移动边缘计算模式迎来了新的挑战。



传统的移动边缘计算模式大多在网络边缘（如地面基站）部署服务器，这些往往意味着高昂的成本，其次这些服务器很难移动实现二次复用。在一些有着社会经济价值的偏远地区，地貌复杂，人烟稀少，部署服务器的成本高于收益，移动边缘计算的应用场景（如地质勘测、荒野救援、视频拍摄等应用场景）因此受限<sup>[22][23]</sup>。

无人机全称无人驾驶飞机，其工作原理是利用无线电遥控装置和程序控制设备操控载具，或者由计算机自动或半自动地控制<sup>[24][25][26]</sup>。基于无人驾驶的特点，无人机往往更加适合一些危险、机械式的作业。从应用领域角度分析，无人机可分为军用和民用。在军用领域，无人机可用于军情侦察或靶机。在民用领域，无人机可用于航拍、快递运输、灾难救援<sup>[27]</sup>。以 2021 年 7 月河南洪灾为例，翼龙无人机为救援工作提供了网络保障，证明了无人机在民用领域的价值。民用领域的应用需求促使各国大力发展无人机技术，积极扩展无人机应用领域。随着无人机技术的迭代更新，其制造成本不断降低，并且机动性强，使移动边缘计算技术迎来新的研究方向<sup>[28][29][30]</sup>。图 1-1 描述了移动边缘计算结合无人机的发展模式。在网络覆盖率低或受灾地区，无人机辅助移动边缘计算可以为地面用户设备提供网络保障和计算服务。

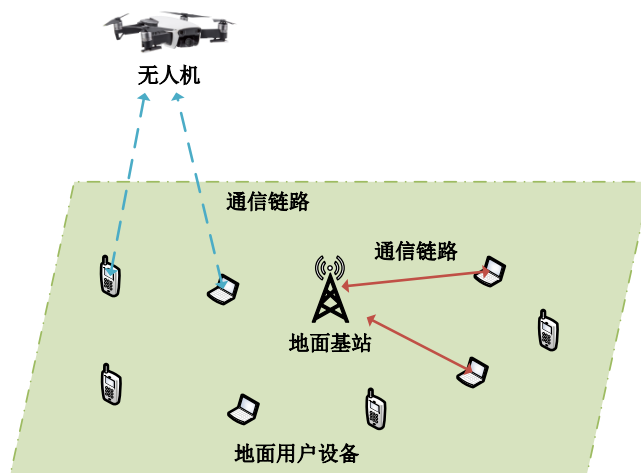


图 1-1 无人机与移动边缘计算联合的发展模式

在万物互联的时代，移动边缘计算技术展现了重要的应用价值<sup>[31][32][33]</sup>。本文主要针对基于无人机的移动边缘计算任务卸载策略问题，再综合考虑问题的影响因素，进而优化地面用户设备的总能耗。

## 1.2 国内外研究现状与问题分析

相较于传统云计算的数据处理模式，移动边缘计算技术将用户数据卸载至接近数据源的网络边缘设备上计算，从而缓解云中心的计算压力，进而降低时延和减少能耗。

然而，在移动边缘计算网络中，地面用户设备的任务卸载策略决定其任务计算的对象，进而间接影响其能量消耗的程度和计算响应时长，最终决定其是否能够持续性工作和提供优质服务。下面对移动边缘计算的研究现状展开分析。

### 1.2.1 移动边缘计算网络下任务卸载策略研究

在移动边缘计算中，由于边缘节点的计算能力、资源、带宽和能量有限，任务卸载是一个关键问题。一种高效的任务卸载策略能够有效降低计算延迟和移动终端能耗，在业内得到了广泛的研究。表 1-1 展示了移动边缘计算任务卸载策略研究的相关文献。

表 1-1 移动边缘计算任务卸载策略研究

主要研究内容	文献	贡献点	不足
优化能耗	[34]	人工鱼群算法，优化能耗	未考虑时延约束
	[35]	两步求解算法，性能均衡	完全依赖地面基站
优化时延	[36]	三种不同的压缩卸载模型	未考虑资源动态分配
	[37]	两层算法，优化时延	完全依赖地面基站
	[38]	车辆辅助移动边缘计算	未考虑能量约束
联合优化能耗和时延	[39]	资源分配，联合优化	未考虑性能均衡

现如今，移动边缘计算的卸载策略研究已经有着丰富的参考文献。在以能耗为优化目标的研究中，Yang 等人<sup>[34]</sup>为了实现能量高效利用，从任务计算和通信两个方面构建计算卸载的能量消耗的模型。然后，Yang 等人提出了一个以最小化系统所有实体的总能耗为目标，同时考虑计算资源和计算时延约束的计算卸载优化问题。在此基础上，又提出了一种基于人工鱼群算法的能量优化方案来实现全局收敛。Chen 等人<sup>[35]</sup>综合考虑了面向移动边缘计算的小蜂窝网络的频谱分配和计算卸载选择，以达到性能均衡。在此过程中，Chen 等人将频谱分配因子、计算卸载决策和终端的传输功率作为一个以最小化能耗为目标的优化问题，并约束每个终端的时延。接着，Chen 等人提出一个两步求解算法来求解这个问题。最后，仿真结果表明了该方案在能耗、延迟、计算卸载和收敛性能方面的有效性。

与此同时，在以时延为优化目标的研究中，Ren 等人<sup>[36]</sup>研究了联合优化通信和计算资源分配的多用户时分多址、移动边缘计算系统的时延最小化问题。研究了三种不同的计算模型，即本地压缩卸载、边缘云压缩卸载和部分压缩卸载模型。首先，导出本地压缩卸载和边缘云压缩卸载模型的最优资源分配和最小系统延迟的表达式。然后，对于部

分压缩卸载模型, Ren 等人构建了一个分段优化问题, 并证明了最优的数据分割策略具有分段结构。在此基础上, 提出了一种最优的联合通信和计算资源分配算法。最后通过数值实验验证了所提算法的有效性, 结果表明所提出的部分压缩卸载模型能够显著降低端到端的时延。Qian 等人<sup>[37]</sup>通过非正交多址访问方式, 研究了蜂窝辅助移动边缘计算问题。Qian 等人首先描述蜂窝用户和边缘计算用户的传输功率, 然后对边缘计算用户的计算工作量、持续时间进行联合优化, 以最小化所有边缘计算用户的计算需求的总时延为目标。最后, Qian 等人提出了一个有效的分层算法来解决该问题。Liu 等人<sup>[38]</sup>探索了一种车辆边缘计算网络结构, 在这种网络结构中, 车辆可以作为移动边缘服务器, 为附近的终端提供计算服务。然后, 在考虑计算任务时延的情况下, Liu 等人提出了终端车辆辅助卸载方案。在此基础上, 提出了一个使车辆边缘计算网络长期效率最大化的优化问题。最后, Liu 等人提出了基于 Q-learning 算法的计算卸载和资源分配的优化策略。

此外, 在针对联合优化能耗和时延的研究中, Kiran 等人<sup>[39]</sup>研究了无线移动边缘计算中的任务卸载和资源分配问题, 以最小化时延和用户设备的能耗为目标。然而, 在这样的动态任务卸载系统中, 为获得最优卸载策略, Kiran 等人提出了一种基于软件定义边缘云的强化学习优化框架, 以解决无线移动边缘计算的任务卸载和资源分配问题。

综上所述, 在传统移动边缘计算的研究中地面基站是不可或缺的角色, 在地面基站覆盖率较低或地面基站被自然灾害摧毁的应用场景中, 传统的移动边缘计算将失去其应有的效用。因此, 将成本较低、机动性强的无人机引入移动边缘计算网络中具有非常广阔的研究前景。

### 1.2.2 无人机辅助移动边缘计算网络下任务卸载策略研究

目前, 针对无人机辅助移动边缘计算的研究可以分为两大类: 单无人机辅助移动边缘计算、多无人机辅助移动边缘计算。这些研究大多致力于最小化能耗或时延, 从而延长设备工作时间或提升服务质量。表 1-2 展示了无人机辅助移动边缘计算任务卸载策略研究的相关文献。

针对单无人机辅助移动边缘计算的研究, Guo 等人<sup>[40]</sup>通过做出卸载决策、在上行和下行链路上分配传输比特以及设计无人机的轨迹, 研究了无人机增强边缘的能量优化问题。综合考虑物联网移动设备有限的电池容量和无人机的能量预算, Guo 等人提出了一种基于块坐标下降和连续凸逼近技术的交替优化算法, 实现能量的高效利用。Liu 等人<sup>[41]</sup>研究了一种基于无人机的移动边缘计算网络, 该网络由无人机执行从移动终端用户卸

载的计算任务,每个移动终端用户的运动遵循高斯-马尔可夫随机模型。为保证每个移动终端用户的服务质量,能量有限的无人机根据移动终端用户的位置动态规划飞行轨迹。为了使系统奖励最大化并满足服务质量约束,Liu 等人提出了一种基于服务质量的行为选择策略的 DDQN (Double Deep Q-Network, DDQN) 网络。Guo 等人<sup>[42]</sup>研究了单无人机辅助移动边缘计算系统。在该系统中,Guo 等人引入了非正交多址来提高频谱效率,将无人机充当用户和基站之间的中继。在考虑无人机能耗和用户服务质量的前提下,进行无人机轨迹和计算卸载联合优化。为了便于处理分析,首先将原始问题划分为两个子问题:无人机轨迹和功率分配联合优化问题和用户调度问题。然后,通过连续凸逼近算法来获取第一个问题的解。最后根据得到的结果,优化用户调度指标。

表 1-2 无人机辅助移动边缘计算任务卸载策略研究

主要研究内容	文献	贡献点	不足
单无人机	[40]	交替优化算法,优化能耗	应用场景受限
	[41]	用户移动性	应用场景受限
	[42]	联合优化、用户调度	未考虑无人机相关约束
多无人机	[43]	双层优化方法	未考虑无人机能量约束
	[44]	多目标联合优化	未考虑无人机防碰撞约束

针对多无人机辅助移动边缘计算的研究,Wang 等人<sup>[43]</sup>建立了一种基于多无人机的移动边缘计算网络,该网络将多架无人机部署为航空边缘云,供大规模移动用户使用。在该网络中,Wang 等人通过考虑无人机的数量和位置来优化无人机的部署。在此基础上,Wang 等人提出了一种以最小化系统能耗为目标,联合优化无人机部署和任务调度的双层优化方法。最后,仿真实验证明所提出的双层优化方法和所建立的多无人机辅助的移动边缘计算网络的有效性。Shang 等人<sup>[44]</sup>研究了空-地综合无线网络中的移动边缘计算。在卸载变量、终端时延、计算资源、无人机功耗等约束条件下,Shang 等人通过联合优化用户关联、上行功率分配、信道分配、计算资源分配和无人机的三维布局,实现地面实体总能耗最小化的目标。

综上所述,在无人机辅助移动边缘计算的研究中,无人机的能耗、信号覆盖、防碰撞以及地面用户设备的移动性是研究优化能耗或时延等问题无法避免的影响因素,也是无人机辅助移动边缘计算拓展其应用领域需要考虑的问题。

### 1.2.3 问题分析

在国内外关于移动边缘计算网络的任务卸载策略研究工作中，主要针对以最小化时延和能耗为目标的优化。传统移动边缘计算模式下，地面基站担当重要的边缘节点，在地面基站覆盖率较低或地面基站被天灾摧毁的应用场景中，移动边缘计算的优势将不复存在。因此，学者专家们将成本较低、机动性强的无人机引入移动边缘计算中极大地拓展了移动边缘计算的应用场景。在无人机辅助移动边缘计算中，如表 1-2 所示，无人机的能耗、信号覆盖、防碰撞以及地面用户设备的移动性是研究新型移动边缘计算问题无法回避的问题影响因素。基于上述分析，移动边缘计算研究主要包含以下两个问题。

#### （1）传统移动边缘计算网络模型过于依赖地面基站的问题

针对移动边缘计算中最小化能耗和时延的研究中，主要目的是延长物联网设备的工作时间和提升服务质量。尽管现有一些通过引入无人机解决传统移动边缘计算过于依赖地面基站的研究，但这些研究仍存在忽视无人机能量有限的问题。因此，在传统移动边缘计算模式中引入一架无人机，如何设计一个基于单无人机辅助移动边缘计算的任务卸载策略，在满足计算资源约束、时延约束和无人机能量约束的同时，实现地面用户设备总能耗最小化是一项可研究的工作内容。

#### （2）单无人机辅助移动边缘计算应用场景受限和用户移动性的问题

针对单无人机辅助移动边缘计算应用场景不足的问题，现有一些多无人机辅助移动边缘计算的研究，但这些研究缺乏考虑无人机信号覆盖和无人机能量的问题，以及引入多架无人机时，无人机之间的防碰撞问题。此外，在移动边缘计算的相关研究中，大多研究默认地面用户设备在一定时间段内时静止不动的，忽视了地面用户设备具备移动性的可能性。因此，在多无人机辅助移动边缘计算中考虑地面用户设备移动性，如何设计一个基于多无人机辅助移动边缘计算的任务卸载策略，在满足时延约束、无人机能量约束、无人机信号覆盖约束以及无人机之间防碰撞约束的同时，实现地面移动设备总能耗最小化也是一项可研究的工作内容。

## 1.3 主要研究思路与内容

综上所述，在无人机辅助移动边缘计算网络中，如何在满足时延约束、无人机能量约束、无人机信号覆盖约束以及无人机之间防碰撞约束的前提下，设计一个有效的任务卸载策略，能够最小化所有地面用户设备总能耗，实现地面用户设备持续性工作是一个

巨大的挑战。为解决上述移动边缘计算的问题，本文对基于无人机的移动边缘计算任务卸载策略展开研究。首先，梳理移动边缘计算的研究背景，分析国内外研究现状。然后，针对现有移动边缘计算中存在的过于依赖地面基站、无人机相关约束、地面设备移动性等不足展开研究。图 1-2 展示了本文主要研究思路和内容。

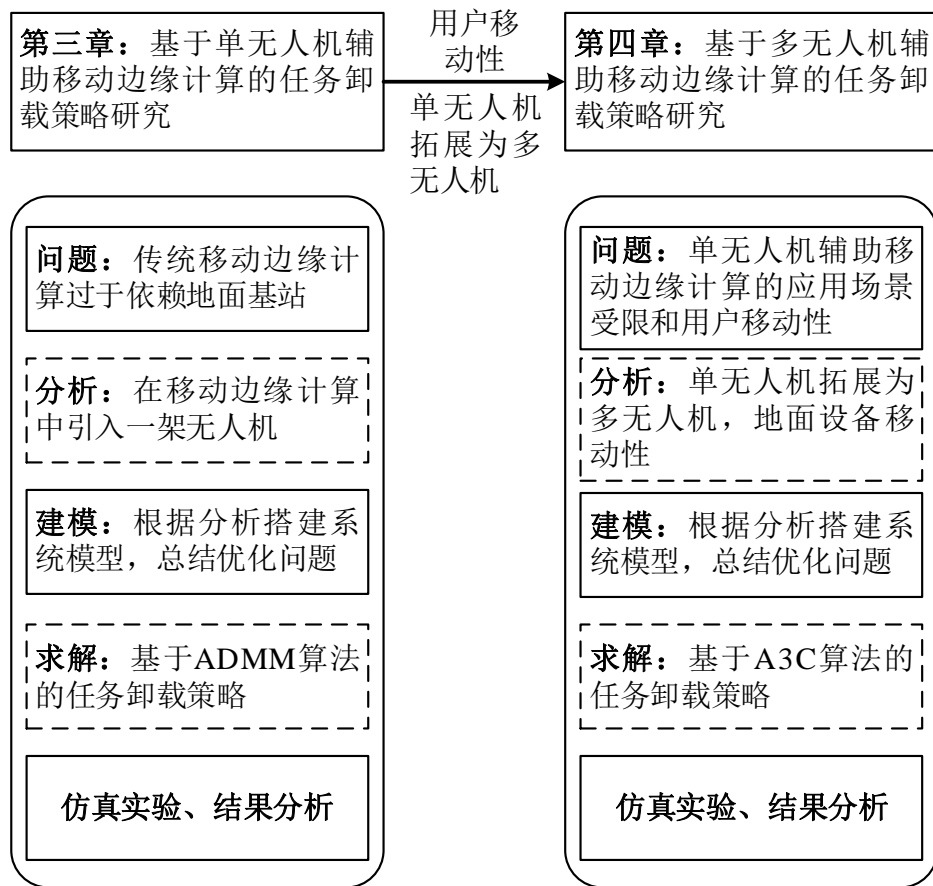


图 1-2 基于无人机的移动边缘计算任务卸载策略研究思路

### (1) 基于单无人机辅助移动边缘计算的任务卸载策略研究

为解决传统移动边缘计算过于依赖地面基站的问题，实现地面用户设备持续性运行，本文提出了基于单无人机辅助移动边缘计算的任务卸载策略，在进行计算任务卸载后的地面用户设备能够降低能耗，延长工作时间。具体而言，首先，分析单无人机辅助移动边缘计算网络的特点，总结计算资源、时延、无人机能量等问题影响因素，构建一个在满足计算资源约束、时延约束和无人机能量约束的同时，以最小化地面用户设备总能耗为目标的联合优化问题模型。然后，设计一个基于交替方向乘子法（Alternating Direction Method of Multiplies, ADMM）算法的任务卸载策略，以获得最佳卸载变量序列，实现最小化地面用户设备总能耗的目标。最后，进行充分的仿真实验，验证该卸载策略的可行性和有效性。

## (2) 基于多无人机辅助移动边缘计算的任务卸载策略研究

基于单无人机辅助移动边缘计算的有效性,本文进一步将单无人机拓展为多无人机,并舍弃位置固定的地面基站作为边缘节点。此外,考虑到现有无人机辅助移动边缘计算研究中地面用户设备多为位置固定,本文设定在一定时间段内地面用户设备具备移动性。为延展单无人机辅助移动边缘计算的应用场景,同时实现地面移动设备持续性运行,本文提出了基于多无人机辅助移动边缘计算的任务卸载策略,在进行计算任务卸载后的地面移动设备能够降低能耗,延长工作时间。具体而言,首先,分析多无人机辅助移动边缘计算网络的特点,总结时延、无人机能量、无人机信号覆盖以及无人机之间防碰撞等问题影响因素,建立在满足时延约束、无人机能量约束、无人机信号覆盖约束以及无人机之间防碰撞约束的同时,以最小化地面移动设备总能耗为目标的联合优化问题模型。然后,设计一个基于异步优势行动者-评论家(Asynchronous Advantage Actor-Critic, A3C)算法的任务卸载策略,以获得最佳卸载变量序列,实现最小化地面移动设备总能耗的目标。最后,进行充分的仿真实验,验证该卸载策略的可行性和优越性。

## 1.4 论文组织结构

论文主要完成无人机辅助移动边缘计算网络中任务卸载策略的研究,通过以下章节进行论述:

第一章为绪论。首先梳理移动边缘计算的发展过程和无人机与移动边缘计算相结合的发展模式,阐述移动边缘计算的研究背景和意义。然后,分析移动边缘计算的国内外研究现状以及当前工作的不足之处。进一步,介绍本文的研究思路与内容。最后,给出本文的组织结构安排。

第二章为相关技术概述。首先,简要介绍移动边缘计算的架构和优点等理论知识。然后,介绍任务卸载策略所涉及的凸优化和强化学习的相关理论知识。

第三章为基于单无人机辅助移动边缘计算的任务卸载策略研究。首先,针对传统移动边缘计算过于依赖地面基站的问题展开描述,并分析在移动边缘计算中引入一架无人机的可能性。然后,在单无人机辅助移动边缘计算网络中,构建一个在满足计算资源约束、时延约束和无人机能量约束的同时,以最小化所有地面用户设备总能耗为目标的联合优化问题。接着,对系统模型下的优化问题进行描述和求解,设计一个基于 ADMM 算法的任务卸载策略。最后,数值仿真结果表明该任务卸载策略的有效性。

第四章为基于多无人机辅助移动边缘计算的任务卸载策略研究。首先,针对第三章中单无人机辅助移动边缘计算应用场景不足的问题,本章将单无人机拓展为多无人机。此外,考虑到现有无人机辅助移动边缘计算研究中地面用户设备多为位置固定,本文设定在一定时间段内地面用户设备具有移动性。然后,在多无人机辅助移动边缘计算网络中,构建一个在满足时延约束、无人机能量约束、无人机信号覆盖约束以及无人机间防碰撞约束的同时,以最小化所有地面移动设备总能耗为目标的联合优化问题。接着,对系统模型下的优化问题进行描述和求解,设计一个基于 A3C 算法的任务卸载策略。最后,通过仿真实验验证该任务卸载策略的优越性。

第五章为总结与展望。首先,对本文的研究工作进行总结。然后,展望未来研究工作的方向。



## 第二章 相关技术概述

### 2.1 移动边缘计算技术

随着进入万物互联时代，物联网设备的数量急剧增加，设备产生的数据量也呈爆炸式增长，网络带宽的压力与日俱增<sup>[45]</sup>。同时，物联网新型应用（如智能家居、车联网、在线直播等）需要更短的反应时间，对实时性要求越来越高。传统云计算模式无法高效处理数据，进而移动边缘计算模式应运而生。移动边缘计算的核心理念是将设备产生的数据迁移至数据源附近的计算设备上运行<sup>[46][47]</sup>。这种边缘式大数据处理方式引起了学界和工业界的兴趣并作出广泛的研究。

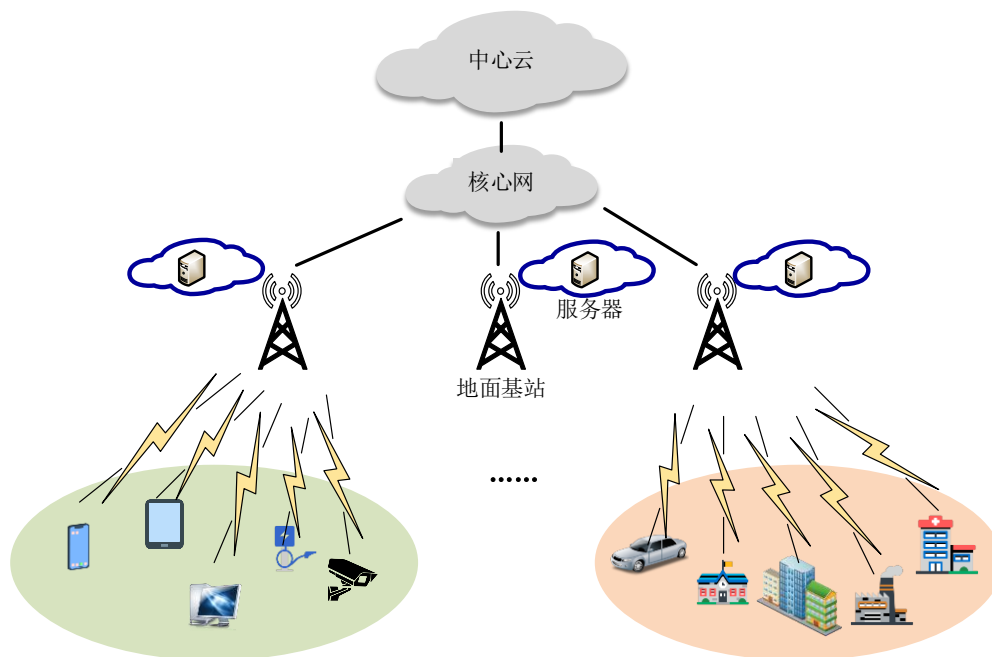


图 2-1 移动边缘计算网络结构示例

图 2-1 展示了移动边缘计算的网路结构，其主要分为三个层次，最底层包括地面物联网设备及其应用场景，中间层主要是配备有服务器的地面基站，最上层是远离地面设备的云端服务器。这种将地面基站作为边缘计算节点的发展模式，对云计算模式作出了有效的补充。

作为物联网时代的关键技术之一，移动边缘计算的诸多优点如下：

(1) 缓解网络带宽压力：在网络边缘处理数据源产生的大量临时数据，不再需要上传至云端服务器，缓解了网络带宽压力。

(2) 降低时延：在数据源附近处理数据，不需要等待远离物联网设备的云端服务器的请求响应，极大地降低了计算任务的响应时间。

(3) 减少能耗：高速增长的数据流量使得云中心的能耗巨大，移动边缘计算模式的发展理念有效地降低了云中心的计算负载，进而减少云中心的能耗。

(4) 保护用户隐私：移动边缘计算模式不需要物联网新型应用的用户上传个人隐私数据至云端处理，而是存储在网络边缘物联网设备上，降低了用户隐私泄露的风险，进而为用户数据安全和隐私提供了保障。

综上所述，移动边缘计算模式的发展理念可以有效地缓解网络带宽压力、降低时延、减少能耗以及保护用户数据隐私和安全。基于移动边缘计算的优势，移动边缘计算已经在公共安全、车联网、智能家居、虚拟现实、工业物联网等应用场景中展现出巨大的潜力和更加广阔的应用前景。

## 2.2 凸优化相关理论

在运算能力持续增强和最优化理论蓬勃发展的时代背景下，常规的凸优化方法已经近似于简单的线性规划方法一样简单可行。因此，若是最优化问题可以转化成凸优化问题，那么最优化问题可以以凸优化的方式解决<sup>[48][49]</sup>。凸优化作为数学最优化研究子领域方法，它主要研究定义于凸集中的凸函数最小化问题。在某种意义上，凸优化相较于传统的数学最优化问题简单，即在凸优化问题中局部最优解默认为是全局最优解。

### 2.2.1 凸优化概述

优化问题定义：

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq 0, \quad i = 1, 2, \dots, m \\ & b_i(x) = 0, \quad i = 1, 2, \dots, k \end{aligned} \quad (2-1)$$

问题 (2-1) 表示在所有满足约束条件下的  $x$  中，最终的解值  $x^*$  是使得目标函数取得最小值的变量。其中， $x \in \mathbf{R}^n$  是需要优化的变量；函数  $f_0: \mathbf{R}^n \rightarrow \mathbf{R}$  是目标函数；函数  $f_i: \mathbf{R}^n \rightarrow \mathbf{R}, i = 1, 2, \dots, m$  是  $m$  个不等式约束函数；函数  $b_i(x) = 0, i = 1, 2, \dots, k$  是  $k$  个等式约束函数。此外，若  $m = k = 0$ ，则称问题 (2-1) 一类无约束问题。

在标准优化问题研究领域内，凸优化是目标函数和约束函数均是凸函数的一类优化问题。若一个现实案例可以按照凸优化的方式描述，那么基本上意味着相应问题可以彻

底解决，这是非凸的优化问题所不具有的性质。在凸优化知识体系中，包含以下几个基本概念：

凸集：目标函数和约束函数的定义域。

凸函数：优化相关函数的凸性限制。

凸优化：中心内容的标准描述。

凸优化问题的求解方法：梯度下降法、牛顿法、内点法等。

对偶问题：可以将一般优化问题转化为凸优化问题。

综上所述，凸优化问题可定义为：

$$\begin{aligned} \min \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) \leq 0, \quad i = 1, 2, \dots, m \\ & a_i^T x = b_i, \quad i = 1, 2, \dots, k \end{aligned} \quad (2-2)$$

其中，约束函数 $f_i(x), i = 1, 2, \dots, m$ 是凸函数。相较于问题（2-1），若目标函数和不等式约束均为凸函数，且等式约束是仿射函数，那么此类优化问题可称为凸优化问题。

### 2.2.2 交替方向乘子法概述

交替方向乘子法是一种兼备两种算法优点的高效收敛性算法<sup>[50][51]</sup>，即对偶上升法的问题可分解性与乘子法的优异收敛性。该算法以形式（2-3）解决问题。

$$\begin{aligned} \min \quad & f(x) + g(y) \\ \text{s.t.} \quad & Ax + By = c \end{aligned} \quad (2-3)$$

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & Ax = b \end{aligned} \quad (2-4)$$

其中，变量 $x \in \mathbf{R}^n, y \in \mathbf{R}^m, A \in \mathbf{R}^{k \times n}, B \in \mathbf{R}^{k \times m}, c \in \mathbf{R}^k$ 。假设 $f$ 和 $g$ 都是凸函数，不同于一般线性等式约束问题（2-4），问题（2-3）中的变量 $x$ 可分为 $x$ 和 $y$ ，目标函数在这个变量 $x$ 分裂过程中也是可分离的。问题（2-3）的最优值可表示为：

$$v^* = \inf \{f(x) + g(y) | Ax + By = c\} \quad (2-5)$$

以乘子法的表达形式为参照，增广拉格朗日量可表示为：

$$\begin{aligned} L_\rho(x, y, z) = & f(x) + g(y) + z^T(Ax + By - c) \\ & + \frac{\rho}{2} \|Ax + By - c\|_2^2 \end{aligned} \quad (2-6)$$

那么，ADMM 算法的迭代方式可参照以下形式：

$$x^{t+1} := \operatorname{argmin} L_\rho(x, y^t, z^t) \quad (2-7)$$

$$y^{t+1} := \operatorname{argmin} L_{\rho}(x^{t+1}, y, z^t) \quad (2-8)$$

$$z^{t+1} := z^t + \rho(Ax^{t+1} + By^{t+1} - c) \quad (2-9)$$

其中  $\rho > 0$ 。该算法和对偶上升法以及乘子法类似，包括最小化变量  $x$  和  $y$  的步骤，对偶变量更新的步骤。与乘子法类似，对偶变量更新使用的是步长等于增广拉格朗日参数  $\rho$ 。问题 (2-3) 的乘子法有如下形式：

$$(x^{t+1}, y^{t+1}) := \operatorname{argmin} L_{\rho}(x, y, z^t) \quad (2-10)$$

$$z^{t+1} := z^t + \rho(Ax^{t+1} + By^{t+1} - c) \quad (2-11)$$

其中增广拉格朗日量是关于两个主变量  $x$  和  $y$  共同最小化的。另一方面，在 ADMM 算法中， $x$  和  $y$  以交替或顺序的方式更新，这就解释了交替方向这一特性。ADMM 算法可以被看作是乘子方法的另一种表达形式，在  $x$  和  $y$  上使用一个高斯-赛德尔 (Gauss-Seidel) 迭代方法，而不是常规的联合最小化方法。当  $f$  或  $g$  可分离时，将求解  $x$  和  $y$  的最小值分为两个步骤正是分解的原理。

ADMM 中的算法状态由  $y^t$  和  $z^t$  组成。换言之， $(y^{t+1}, z^{t+1})$  是  $(y^t, z^t)$  的一个函数。变量  $x^t$  不是状态的一部分，它是前一状态  $(y^{t-1}, z^{t-1})$  计算的中间结果。若是在问题 (2-3) 中重新标记  $x$  和  $y$ 、 $f$  和  $g$ 、 $A$  或  $B$ ，将会得到 ADMM 的变体，其顺序为  $x$  更新步骤 (2-7) 和  $y$  更新步骤 (2-8) 的反转。 $x$  和  $y$  的作用几乎是对称的，但并不完全对称，因为对偶变量更新是在  $y$  更新之后，但在  $x$  更新之前完成的。

综上所述，ADMM 算法是一种可以把一个大问题分解成多个可分布式同时求解的小问题的算法，ADMM 算法的框架可以解决大部分实际中的大规模问题。

## 2.3 强化学习相关理论

强化学习的理念起源于动物心理学，其模仿人类和动物在自然界中学习的试错机制，是一种通过与环境不断交互，学习状态到行为的映射关系，以期望获得最大累计回报的学习方法<sup>[52][53]</sup>。强化学习包括智能体和环境两大对象。图 2-2 阐述了强化学习的基本原理。在一个离散的时间序列  $t = \{0, 1, 2, \dots, T\}$  中，智能体以完成某项任务为目标。在每一个时间  $t$ ，智能体从环境中获取一个状态  $s_t$ ，并通过策略机制生成的动作  $a_t$  与环境进行交互，环境会反馈产生的新状态  $s_{t+1}$ ，同时给出立即回报  $r_{t+1}$ 。如此智能体与环境不断循环交互，从而产生更多数据（状态和立即回报），然后利用新数据进一步优化自身行为。

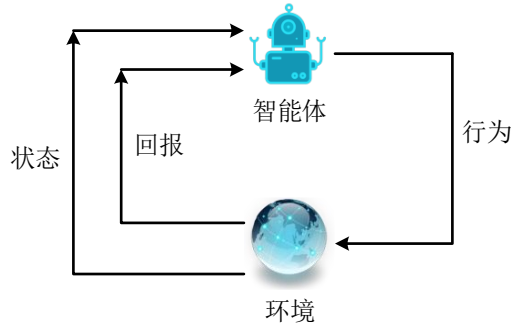


图 2-2 强化学习原理

智能体是指存在于环境中，能够与环境进行交互，并且能够自主采取行动来完成某项任务的强化学习系统<sup>[54][55]</sup>。它主要由策略、值函数和模型三个元素中的一个或多个构成。

(1) 策略：策略是生成智能体行为的机制，是状态到行为的映射，用公式 (2-12) 表示，它描述了智能体在不同状态下的各种行为的可能性。

$$\pi(a|s) = P(A_t = a | S_t = s) \quad (2-12)$$

(2) 值函数：值函数表示智能体在某一确定状态下的表现，或者在某一确定状态下采取某一行为的好坏程度。而好坏程度用未来的期望回报表示，回报和采取的策略有关，所有值函数的估计都是基于某一策略进行的。公式 (2-13) 表示从  $t$  时刻开始往后所有有衰减的回报的总和。

$$G_t = R_{t+1} + \gamma R_{t+2} + \dots = \sum_{n=0}^{\infty} \gamma^n R_{t+n+1} \quad (2-13)$$

其中  $\gamma$  表示衰减系数，用于描述未来回报值在当前时刻的价值比例。

(3) 模型：在强化学习任务中，模型  $M$  是智能体对外部环境的建模。智能体以模型来模拟与环境交互的机制，即给定一个状态和行为，环境模型能够反馈立即回报和预测下一状态。环境模型至少需要解决两个关键问题：状态转换概率  $P_{ss'}^a$ 、预测可能获得的立即回报  $R_s^a$ 。  $P_{ss'}^a$  表示环境的动态特征，用以预测产生下一个状态的概率，其使用公式 (2-14) 表示。  $R_s^a$  表示在状态  $s$  时采取的行动  $a$  的立即回报，用公式 (2-15) 表示。

$$p_{ss'}^a = P(S_{t+1} = s' | S_t = s, A_t = a) \quad (2-14)$$

$$R_s^a = E[R_{t+1} | S_t = s, A_t = a] \quad (2-15)$$

智能体为获取更高回报，在进行决策时需要综合考虑立即回报和未来回报。因此，求解强化学习问题一般分为两步：(1) 构建数学模型——马尔可夫决策模型；(2) 求解马尔可夫决策模型最优解。根据智能体在解决强化学习问题时是否建立环境动力学模型，将强化学习分为两大类：有模型方法、无模型方法，如图 2-3 所示。

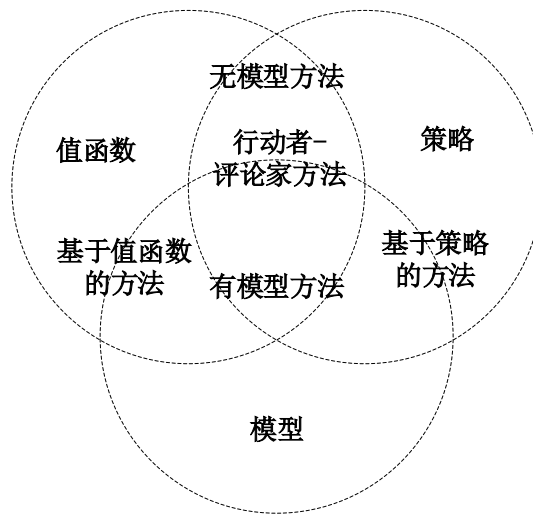


图 2-3 强化学习分类

综上所述，强化学习是解决机器认知的关键技术之一，特别是在人工智能领域的发展日益成熟，可以应对传统算法难以处理的问题。

## 2.4 本章小结

本章首先阐述了移动边缘计算的发展背景和优势。接着介绍了无人机辅助移动边缘计算网络下任务卸载策略所涉及的相关理论知识，主要包括凸优化相关理论和深度强化学习相关理论。

## 第三章 基于单无人机辅助移动边缘计算 的任务卸载策略研究

### 3.1 问题描述与分析

随着智能终端的普及和现代物联网无线应用的出现<sup>[56]</sup>,各种新的网络服务和应用层出不穷,如增强/虚拟现实、联网汽车、远程医疗、智能家居等低延迟需求服务和移动数据流量的需求急剧增加<sup>[57]</sup>。然而,除了需要高效可靠的通信之外,这些新的应用程序还需要强大的计算能力。在进行实时信息处理时,大量的数据计算和资源消耗对计算资源和能量有限的移动终端设备构成了巨大的挑战。云计算技术的出现为解决这些问题提供了一种新思路,但这种方法需要将大量数据卸载到距离用户设备很远的云端服务器上,终端和服务端之间的距离导致了很长的传输延迟,这不足以满足时延敏感的应用。为了解决这个问题,移动边缘计算的概念被提出,通过利用网络边缘的计算资源,提供更高效、更灵活的计算服务<sup>[58]</sup>。移动边缘计算中的许多重要问题得到了广泛的研究,包括计算任务卸载<sup>[59]</sup>、能耗优化<sup>[60]</sup>、延迟优化问题<sup>[61]</sup>,以及资源分配与卸载决策的联合优化<sup>[62]</sup>。

尽管超密集的边缘网络可以提供指数级增长的边缘计算资源,但它们可能无法覆盖郊区和农村地区,这可能需要广泛部署物联网设备来执行某些计算密集型应用程序。由于地面接入网覆盖不完全,传统的移动边缘计算模式无法用于此类场景。近年来,生产成本低、机动性强的无人机在军事领域(如监视敌方活动)、民用和商用领域(如快递运输、视频拍摄、危险地区紧急救援行动)受到了广泛的关注。因此,研究能够提高远程物联网应用计算能力的无人机辅助移动边缘计算网络具有重要的研究价值和应用价值。

基于上述分析,如何设计一种基于单无人机辅助移动边缘计算的任务卸载策略,为网络覆盖率低的地区的物联网用户提供强大的计算服务是一项可研究的内容。在单无人机辅助移动边缘计算网络下,本章将重点放在联合卸载决策和无人机位置上,以最小化地面用户设备总能耗。由此产生的优化问题是一个离散非线性非凸优化问题,是一个难以直接处理的问题<sup>[63]</sup>。为了解决不同变量之间的耦合问题,通过将耦合的二元变量松弛为连续变量并引入辅助变量,将考虑的问题转化为线性优化问题。由此,提出了一种基于 ADMM 算法的低复杂度的任务卸载策略,以分布式方式获得优化问题的近似最优解。

## 3.2 系统模型

### 3.2.1 网络模型

考虑如图 3-1 所示的单无人机辅助移动边缘网络，其中地面基站和无人机各有一个服务器，为 $K$ 个地面用户设备提供计算卸载，并且所有设备都配有单独的天线。在单无人机辅助移动边缘计算网络中，地面用户设备生成的计算任务可以在本地计算，或者在无人机配备的服务器上计算，或者在地面基站配备的服务器上计算。本章节用 $\mathcal{K} = \{1, 2, 3, \dots, K\}$ 表示 $K$ 个地面用户设备组成的集合。在紧急救援或密集通信等场景下，无人机服务器离地面用户侧更近，降低了地面基站服务器处理计算任务的压力。无人机部署小型分布式微云服务器，大小、重量和功率均有限。地面基站部署的是边缘云服务器，拥有更强的计算能力。由于单无人机辅助移动边缘计算网络综合考虑了集中式和分布式计算模型的优点，地面用户设备可以通过无线传输链路将计算任务卸载给地面基站的边缘云或无人机的微云。不同于部署严格的传统移动边缘计算网络，单无人机辅助移动边缘计算网络部署高效灵活。在人口密集地区，可临时部署单无人机辅助移动边缘计算网络，为地面用户设备提供高效服务，例如视频制作者拍摄短视频。在网络覆盖范围有限的偏远地区，单无人机辅助移动边缘计算网络展现较高的机动性，如地形测量和采矿。

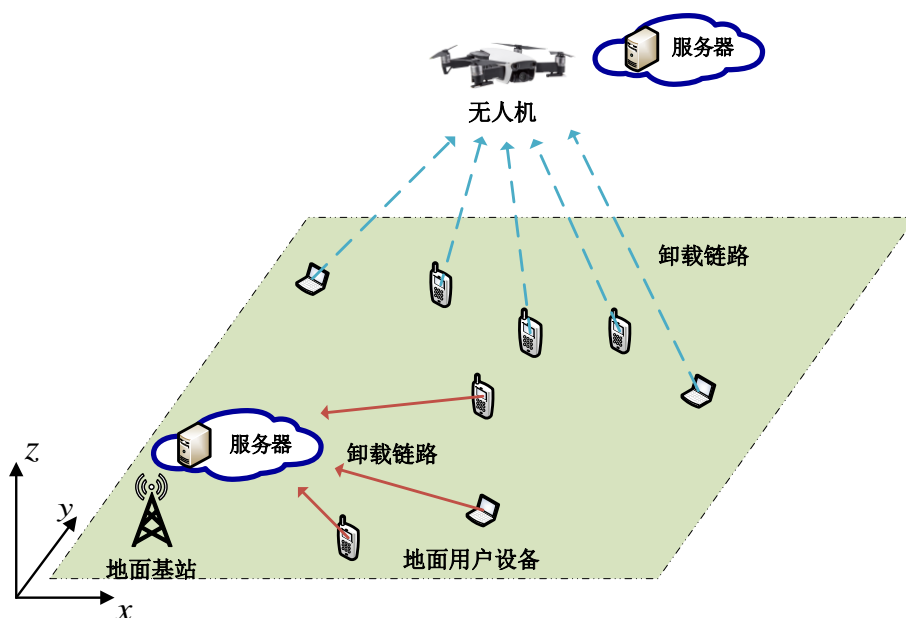


图 3-1 单无人机辅助移动边缘计算网络模型

为便于阐述，每个实体的位置使用三维笛卡尔坐标系来表示，其坐标单位为米。在不失一般性的条件下，本章假设所有用户设备和地面基站的位置都是固定的，地面基站



的位置用  $\mathbf{w}_{BS} = (0,0,0)$  表示，地面设备  $k$  的位置用  $\mathbf{w}_k = (x_k, y_k, 0)$  表示。假设无人机飞行在固定高度  $H$ ，无需频繁升降， $\mathbf{q}_U = (x_u, y_u, H)$  表示无人机的 3D 位置，其中  $x_u$  和  $y_u$  分别表示无人机平面坐标位置。

### 3.2.2 通信模型

无人机和地面基站均兼备信息传输和计算服务<sup>[31]</sup>两种功能，而所有地面用户设备都具有信息传输/接收功能。对于设备与设备之间的通信链路，地面用户设备与无人机是以视距传输方式作为通信链路，地面用户设备与地面基站是以非视距传输方式作为通信链路。根据 3GPP 技术报告，地面用户设备  $k$  与无人机之间的通信链路表示如下：

$$h_k^{UAV} = \frac{\beta_0}{(x_u - x_k)^2 + (y_u - y_k)^2 + H^2} \quad (3-1)$$

其中  $\beta_0$  表示参考距离为 1 米处的信道功率增益。设  $h_k^{BS}$  表示地面用户设备  $k$  与地面基站之间的通信链路，通过公式 (3-2) 计算得出。

$$h_k^{BS} = (d_k^{BS})^{-\frac{\alpha}{2}} g_k^{BS} \quad (3-2)$$

其中  $d_k^{BS}$  表示地面用户设备  $k$  与地面基站之间的距离， $g_k^{BS}$  是归一化的瑞利衰落信道状态， $\alpha$  为路径损耗系数。

当地面用户设备通过无线信道与无人机通信，地面用户设备  $k$  将计算任务卸载给无人机微云时，上行数据传输速率  $R_k^{UAV}$  可由公式 (3-3) 计算得出。

$$R_k^{UAV} = B \log_2 \left( 1 + \frac{h_k^{UAV} p_k}{\sum_{j \neq k} h_j^{UAV} p_j + \sigma^2} \right) \quad (3-3)$$

其中， $p_k$  表示地面用户设备  $k$  的发射功率， $\sum_{j \neq k} h_j^{UAV} p_j$  是地面其他终端的干扰信号， $B$  为可用信道带宽， $\sigma^2$  表示噪声功率。

当地面用户设备  $k$  将计算任务卸载至地面基站的边缘云时，上行数据传输速率  $R_k^{BS}$  可由公式 (3-4) 计算得出。

$$R_k^{BS} = B \log_2 \left( 1 + \frac{h_k^{BS} p_k}{\sum_{j \neq k} h_j^{BS} p_j + \sigma^2} \right) \quad (3-4)$$

### 3.2.3 计算模型

假设每个地面用户设备都有一个计算密集型任务，无人机微云与地面基站边缘云合作为地面终端提供计算服务。设本地计算的变量为  $a_{k,l}$ ，卸载至无人机服务器的变量为  $a_{k,u}$ ，卸载至地面基站服务器的变量为  $a_{k,BS}$ ，且三个变量满足  $a_{k,l} + a_{k,u} + a_{k,BS} = 1$ 。在

计算任务卸载阶段，地面用户设备 $k$ 的计算任务可表述为一个三元组 $W_k = \langle L_k, C_k, T_k \rangle$ ，其中 $L_k$ 表示计算卸载任务的数据大小， $C_k$ 表示计算每比特数据所需的 CPU 周期数， $T_k$ 表示可容忍的时延。

**1) 本地计算：**假设 $f_k^{\text{loc}}$ 表示地面用户设备 $k$ 的 CPU 频率。当地面用户设备 $k$ 决定在本地执行计算任务，计算任务的执行时间可由公式（3-5）计算得出。

$$t_k^{\text{loc}} = \frac{L_k C_k}{f_k^{\text{loc}}} \quad (3-5)$$

本地计算所耗能量 $E_k^{\text{loc}}$ 可由公式（3-6）表示。

$$E_k^{\text{loc}} = p_k^{\text{loc}} t_k^{\text{loc}} = p_k^{\text{loc}} \frac{L_k C_k}{f_k^{\text{loc}}} \quad (3-6)$$

其中， $p_k^{\text{loc}}$ 表示地面用户设备 $k$ 计算自身任务时的 CPU 功耗。

**2) 计算任务卸载至无人机微云：**无人机服务器是地面用户设备 $k$ 卸载计算任务的首选位置。无人机微云搭载的是一个小型服务器，其处理频率 $f^{\text{UAV}}$ 大于地面用户设备的计算频率，即 $f_k^{\text{loc}} \leq f^{\text{UAV}}$ 。如果只有一个地面用户设备向无人机微云卸载计算任务，无人机微云会使用整个频率进行计算。否则，多个地面用户设备共享无人机计算频率。记分配给地面用户设备 $k$ 的计算频率为 $f_k^{\text{UAV}}$ ，分配给多个地面用户设备总的计算频率不能超过无人机服务器的最大可用处理频率，即

$$\sum_{k=1}^K a_{k,u} f_k^{\text{UAV}} \leq f^{\text{UAV}} \quad (3-7)$$

地面用户设备向无人机服务器卸载任务分为三个步骤：（1）将计算任务卸载至无人机微云；（2）无人机服务器执行计算；（3）将计算结果返回到地面用户设备。通常，计算结果远小于输入数据的大小，因此计算结果返回过程可忽略不计。基于通信模型，地面用户设备 $k$ 将计算任务卸载至无人机服务器所需的传输时间和消耗的能量可分别由公式（3-8）和（3-9）表示。

$$t_{k,u}^{\text{tr}} = \frac{L_k}{R_k^{\text{UAV}}} \quad (3-8)$$

$$E_{k,u}^{\text{off}} = p_k t_{k,u}^{\text{tr}} = p_k \frac{L_k}{R_k^{\text{UAV}}} \quad (3-9)$$

无人机服务器接收到地面用户卸载的任务数据后，计算进程即可开始执行。那么，无人机服务器执行计算任务所消耗的时间和能量可分别由公式（3-10）和（3-11）表示。

$$t_{k,u}^{\text{cal}} = \frac{L_k C_k}{f_k^{\text{UAV}}} \quad (3-10)$$

$$E_{k,u}^{\text{cal}} = p_k^{\text{UAV}} t_{k,u}^{\text{cal}} = p_k^{\text{UAV}} \frac{L_k C_k}{f_k^{\text{UAV}}} \quad (3-11)$$

其中,  $p_k^{\text{UAV}}$  表示无人机执行地面用户设备  $k$  的计算任务时的 CPU 功耗。

**3) 计算任务卸载至地面基站边缘云:** 地面基站服务器是地面用户设备计算任务卸载的第二选择。假设地面基站服务器比无人机服务器拥有更多的计算资源, 分配给地面用户设备的计算频率表示为  $f_k^{\text{BS}}$ 。分配给多个地面用户设备总的计算频率不能超过地面基站边缘云的最大可用处理频率  $f^{\text{BS}}$  [32], 即

$$\sum_{k=1}^K a_{k,\text{BS}} f_k^{\text{BS}} \leq f^{\text{BS}} \quad (3-12)$$

基于通信模型, 地面用户设备将计算任务卸载给地面基站的服务器时所消耗的传输时间和能量可分别由公式 (3-13) 和 (3-14) 表示。

$$t_{k,\text{BS}}^{\text{tr}} = \frac{L_k}{R_k^{\text{BS}}} \quad (3-13)$$

$$E_{k,\text{BS}}^{\text{off}} = p_k t_{k,\text{BS}}^{\text{tr}} = p_k \frac{L_k}{R_k^{\text{BS}}} \quad (3-14)$$

地面基站接收到地面终端卸载的任务数据后, 基站服务器开始执行计算进程。那么, 地面基站服务器执行计算任务所消耗的时间和能量可分别由公式 (3-15) 和 (3-16) 表示。

$$t_{k,\text{BS}}^{\text{cal}} = \frac{L_k C_k}{f_k^{\text{BS}}} \quad (3-15)$$

$$E_{k,\text{BS}}^{\text{cal}} = p_k^{\text{BS}} t_{k,\text{BS}}^{\text{cal}} = p_k^{\text{BS}} \frac{L_k C_k}{f_k^{\text{BS}}} \quad (3-16)$$

其中,  $p_k^{\text{BS}}$  表示地面基站执行地面用户设备  $k$  计算任务时的 CPU 功耗。

根据公式 (3-8) 至公式 (3-15), 地面用户设备  $k$  卸载计算任务至无人机服务器或地面基站服务器的总时延分别由公式 (3-17) 和 (3-18) 表示。

$$t_k^{\text{UAV}} = t_{k,u}^{\text{tr}} + t_{k,u}^{\text{cal}} \quad (3-17)$$

$$t_k^{\text{BS}} = t_{k,\text{BS}}^{\text{tr}} + t_{k,\text{BS}}^{\text{cal}} \quad (3-18)$$

### 3.3 优化问题描述

本文假设地面用户设备  $k$  的决策向量为  $\mathbf{a}_k = [a_{k,l}, a_{k,u}, a_{k,\text{BS}}]$ 。本文的目标是在保证资源约束和延迟约束的前提下, 通过联合优化地面用户设备卸载决策变量和无人机飞行

位置,使所有地面用户设备的能源消耗最小化。基于此目标,本章可将该优化问题定义为一个多目标优化问题,问题建模和约束条件可以表示为:

$$\begin{aligned}
& \min_{a_k, q_U} \sum_{k=1}^K (a_{k,l} E_k^{\text{loc}} + a_{k,u} E_{k,u}^{\text{off}} + a_{k,BS} E_{k,BS}^{\text{off}}) \\
& \text{s.t. C1: } \sum_{k=1}^K a_{k,u} f_k^{\text{UAV}} \leq f^{\text{UAV}} \\
& \quad \text{C2: } \sum_{k=1}^K a_{k,BS} f_k^{\text{BS}} \leq f^{\text{BS}} \\
& \quad \text{C3: } \sum_{k=1}^K a_{k,u} E_{k,u}^{\text{cal}} \leq P_{\max}^{\text{UAV}} \\
& \quad \text{C4: } a_{k,l} t_k^{\text{loc}} + a_{k,u} t_k^{\text{UAV}} + a_{k,BS} t_k^{\text{BS}} \leq T_k \\
& \quad \text{C5: } \{a_{k,l}, a_{k,u}, a_{k,BS}\} \in \{0,1\} \\
& \quad \text{C6: } a_{k,l} + a_{k,u} + a_{k,BS} = 1
\end{aligned} \tag{3-19}$$

其中,约束 C1 和约束 C2 分别限制分配给多个地面用户设备总的计算频率不能超过无人机微云和地面基站边缘云的最大处理频率。约束 C3 限制无人机的能耗。约束 C4 表示计算任务允许的最大延迟。约束 C5 表明二进制计算卸载变量的取值范围。约束 C6 表明任何一个地面用户设备产生的计算任务,最多只能卸载至一个边缘计算节点。

由于上述问题(3-19)是一个混合非线性问题,在一般情况下难以求解最优解。为了有效地解决这类问题,需要设计相应的算法。下一小节将证明非线性组合优化问题可以通过二元变量松弛法转化为线性规划问题,然后提出一个分布式算法来求解它。

### 3.4 优化问题求解

#### 3.4.1 优化问题重构

为了解决目标问题的非凸性,本小节将二进制变量 $a_{k,l}$ 、 $a_{k,u}$ 和 $a_{k,BS}$ 松弛为连续的决策变量,即 $0 \leq a_{k,l} \leq 1$ ,  $0 \leq a_{k,u} \leq 1$ 和 $0 \leq a_{k,BS} \leq 1$ 。松弛变量可以解释为相关计算任务的一部分,可以分别在本地计算,或卸载至无人机微云计算,或卸载至地面基站边缘云计算。为了简化问题(3-19)的数学表达式,地面用户设备 $k$ 的计算任务卸载的总能耗可由公式(3-20)定义。

$$Q(a_k, q_U) = a_{k,l} E_k^{\text{loc}} + a_{k,u} E_{k,u}^{\text{off}} + a_{k,BS} E_{k,BS}^{\text{off}} \tag{3-20}$$

松弛二进制变量后，问题（3-19）可重新定义为问题（3-21）：

$$\begin{aligned}
 & \min_{\mathbf{a}_k, \mathbf{q}_U} \sum_{k=1}^K Q(\mathbf{a}_k, \mathbf{q}_U) \\
 \text{s.t. C1: } & \sum_{k=1}^K a_{k,u} f_k^{\text{UAV}} \leq f^{\text{UAV}} \\
 \text{C2: } & \sum_{k=1}^K a_{k,\text{BS}} f_k^{\text{BS}} \leq f^{\text{BS}} \\
 \text{C3: } & \sum_{k=1}^K a_{k,u} E_{k,u}^{\text{cal}} \leq P_{\max}^{\text{UAV}} \\
 \text{C4: } & a_{k,l} t_k^{\text{loc}} + a_{k,u} t_k^{\text{UAV}} + a_{k,\text{BS}} t_k^{\text{BS}} \leq T_k \\
 \text{C5: } & 0 \leq a_{k,l}, a_{k,u}, a_{k,\text{BS}} \leq 1 \\
 \text{C6: } & a_{k,l} + a_{k,u} + a_{k,\text{BS}} = 1
 \end{aligned} \tag{3-21}$$

从问题（3-21）可以看出目标函数是关于二进制变量 $a_{k,l}$ 、 $a_{k,u}$ 和 $a_{k,\text{BS}}$ 的一个线性问题，很容易分析出问题（3-21）是一个凸函数。同时，目标问题（3-21）中的约束 C1 至约束 C6 均是线性约束。综上，问题（3-21）是一类具有多约束、线性和优化变量的凸优化问题，具有可行解。

对于一类线性规划的凸优化问题，有多种凸优化方法，如内点法和梯度下降法<sup>[34]</sup>。随着物联网设备的增加和网络数据的爆炸式增长，传统的集中式处理方法对解决这些问题提出了极大的挑战。而分布式处理方法可以很好地解决这类问题，并得到近似最优解。

### 3.4.2 基于 ADMM 算法的任务卸载策略

分布式 ADMM 算法是一种简单但功能强大的算法，非常适用于分布式凸优化<sup>[53]</sup>。对于优化问题（3-21），可以看出变量 $\mathbf{a}_{k,u}$ 和 $\mathbf{q}_U$ 的线性组合使问题复杂化。因此，通过定义 $r_{k,u} = \frac{a_{k,u}}{R_k^{\text{UAV}}}$ 来简化问题。为了使问题（3-21）易处理，本小节引入一些辅助变量。具体地，复制全局变量 $a_{k,l}$ 、 $a_{k,u}$ 和 $a_{k,\text{BS}}$ 而产生的局部变量，分别对应于 $\hat{a}_{k,l}$ 、 $\hat{a}_{k,u}$ 和 $\hat{a}_{k,\text{BS}}$ ，从而得到公式（3-22）。

$$\hat{\mathbf{g}}_k = \begin{Bmatrix} \hat{a}_{k,l} \\ \hat{a}_{k,u} \\ \hat{a}_{k,\text{BS}} \\ \hat{r}_{k,u} \end{Bmatrix} = \begin{cases} \hat{a}_{k,l} = a_{k,l}, & k \in \mathcal{K} \\ \hat{a}_{k,u} = a_{k,u}, & k \in \mathcal{K} \\ \hat{a}_{k,\text{BS}} = a_{k,\text{BS}}, & k \in \mathcal{K} \\ \hat{r}_{k,u} = r_{k,u}, & k \in \mathcal{K} \end{cases} \tag{3-22}$$

根据 ADMM 算法的原理, ADMM 算法可以对问题 (3-21) 进行有效地分解。此外, 问题 (3-21) 中的约束认为各地面用设备的局部变量应该相等。通过使用增广拉格朗日乘子法, 问题 (3-21) 可以重新表示为问题 (3-23)。

$$\begin{aligned}
Q_{\rho}(\mathbf{a}_k, \mathbf{q}_U, \boldsymbol{\pi}_k, \hat{\mathbf{g}}_k) = & \sum_{k=1}^K Q(\hat{\mathbf{a}}_k, \hat{\mathbf{q}}_U) + \sum_{k=1}^K \delta_k(\hat{a}_{k,l} - a_{k,l}) \\
& + \sum_{k=1}^K \phi_k(\hat{a}_{k,u} - a_{k,u}) + \sum_{k=1}^K \eta_k(\hat{a}_{k,BS} - a_{k,BS}) \\
& + \sum_{k=1}^K \psi_k(\hat{r}_{k,u} - r_{k,u}) + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,l} - a_{k,l}\|_2^2 \\
& + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,u} - a_{k,u}\|_2^2 + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,BS} - a_{k,BS}\|_2^2 \\
& + \frac{\rho}{2} \sum_{k=1}^K \|\hat{r}_{k,u} - r_{k,u}\|_2^2
\end{aligned} \tag{3-23}$$

其中,  $\boldsymbol{\pi}_k = \{\delta_k, \phi_k, \eta_k, \psi_k\}$  是问题 (3-21) 的拉格朗日乘子,  $\rho$  表示正惩罚参数。同时,  $\rho$  值的变化影响迭代法的效率。为方便优化目标变量, 将等式约束的线性项和二次项组合在一起, 增广拉格朗日函数 (3-23) 可改写为公式 (3-24)。

$$\begin{aligned}
Q_{\rho}(\mathbf{a}_k, \mathbf{q}_U, \boldsymbol{\pi}_k, \hat{\mathbf{g}}_k) = & Q_{\rho}(\mathbf{a}_k, \mathbf{q}_U, \boldsymbol{\xi}_k, \hat{\mathbf{g}}_k) \\
= & \sum_{k=1}^K Q(\hat{\mathbf{a}}_k, \hat{\mathbf{q}}_U) + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,l} - a_{k,l} + \lambda_k\|_2^2 \\
& + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,u} - a_{k,u} + \theta_k\|_2^2 + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,BS} - a_{k,BS} + \mu_k\|_2^2 \\
& + \frac{\rho}{2} \sum_{k=1}^K \|\hat{r}_{k,u} - r_{k,u} + \varsigma_k\|_2^2
\end{aligned} \tag{3-24}$$

其中,  $\boldsymbol{\xi}_k = \{\lambda_k, \theta_k, \mu_k, \varsigma_k\}$ ,  $\lambda_k = \frac{\delta_k}{\rho}$ ,  $\theta_k = \frac{\phi_k}{\rho}$ ,  $\mu_k = \frac{\eta_k}{\rho}$ ,  $\varsigma_k = \frac{\psi_k}{\rho}$ 。

根据优化问题的定义, ADMM 算法需要不断迭代问题 (3-24) 中  $\mathbf{a}_k$ 、 $\hat{\mathbf{g}}_k$ 、 $r_{k,u}$  和  $\boldsymbol{\xi}_k$  的值。 $\mathbf{a}_k$ 、 $\hat{\mathbf{g}}_k$ 、 $r_{k,u}$  和  $\boldsymbol{\xi}_k$  的值在第  $i$  次迭代中获取并用来在第  $(i+1)$  次迭代中更新旧值, 图 3-2 展示了基于 ADMM 算法的任务卸载策略关键步骤, 具体的算法迭代方法描述如下:

**1) 局部变量更新:** 首先, 初始化  $\mathbf{a}_k$  和  $\boldsymbol{\xi}_k$ 。然后, 根据函数  $Q_{\rho}(\mathbf{a}_k, \mathbf{q}_U, \boldsymbol{\xi}_k, \hat{\mathbf{g}}_k)$  最小化的情况下, 求解局部变量  $\hat{\mathbf{g}}_k$  的值, 即

$$\hat{\mathbf{g}}_k^{(i+1)} = \underset{\{\hat{a}_{k,l}, \hat{a}_{k,u}, \hat{a}_{k,BS}, \hat{r}_{k,u}\}}{\operatorname{argmin}} Q_\rho(\mathbf{a}_k^{(i)}, \mathbf{q}_U^{(i)}, \boldsymbol{\xi}_k^{(i)}, \hat{\mathbf{g}}_k) \quad (3-25)$$

根据公式 (3-25)，算法需要得到问题 (3-26) 的最优解。

$$\begin{aligned} \min_{\{\hat{a}_{k,l}, \hat{a}_{k,u}, \hat{a}_{k,BS}, \hat{r}_{k,u}\}} & \sum_{k=1}^K Q(\hat{\mathbf{a}}_k, \hat{\mathbf{q}}_U) + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,l} - a_{k,l}^{(i)} + \lambda_k^{(i)}\|_2^2 \\ & + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,u} - a_{k,u}^{(i)} + \theta_k^{(i)}\|_2^2 + \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,BS} - a_{k,BS}^{(i)} + \mu_k^{(i)}\|_2^2 \\ & + \frac{\rho}{2} \sum_{k=1}^K \|\hat{r}_{k,u} - r_{k,u}^{(i)} + \varsigma_k^{(i)}\|_2^2 \\ \text{s.t. C1: } & \sum_{k=1}^K \hat{a}_{k,u} f_k^{\text{UAV}} \leq f^{\text{UAV}} \\ \text{C2: } & \sum_{k=1}^K \hat{a}_{k,BS} f_k^{\text{BS}} \leq f^{\text{BS}} \\ \text{C3: } & \sum_{k=1}^K \hat{a}_{k,u} E_{k,u}^{\text{cal}} \leq P_{\max}^{\text{UAV}} \\ \text{C4: } & \hat{a}_{k,l} t_k^{\text{loc}} + \hat{a}_{k,u} t_k^{\text{UAV}} + \hat{a}_{k,BS} t_k^{\text{BS}} \leq T_k \\ \text{C5: } & 0 \leq \hat{a}_{k,l}, \hat{a}_{k,u}, \hat{a}_{k,BS} \leq 1 \\ \text{C6: } & \hat{a}_{k,l} + \hat{a}_{k,u} + \hat{a}_{k,BS} = 1 \end{aligned} \quad (3-26)$$

问题 (3-26) 是一个凸优化问题，MATLAB 中的 CVX 可以推导出问题<sup>[34]</sup>的一个最优解。

**2) 全局变量更新：**在这一步中，根据上一步中得到的局部变量  $\hat{\mathbf{g}}_k$  的值来更新全局变量  $\mathbf{a}_k$  和  $r_{k,u}$ 。详细更新如下：

$$a_{k,l}^{(i+1)} = \underset{\{a_{k,l}\}}{\operatorname{argmin}} \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,l}^{(i+1)} - a_{k,l} + \lambda_k^{(i)}\|_2^2 \quad (3-27)$$

$$a_{k,u}^{(i+1)} = \underset{\{a_{k,u}\}}{\operatorname{argmin}} \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,u}^{(i+1)} - a_{k,u} + \theta_k^{(i)}\|_2^2 \quad (3-28)$$

$$a_{k,BS}^{(i+1)} = \underset{\{a_{k,BS}\}}{\operatorname{argmin}} \frac{\rho}{2} \sum_{k=1}^K \|\hat{a}_{k,BS}^{(i+1)} - a_{k,BS} + \mu_k^{(i)}\|_2^2 \quad (3-29)$$

$$r_{k,u}^{(i+1)} = \underset{\{r_{k,u}\}}{\operatorname{argmin}} \frac{\rho}{2} \sum_{k=1}^K \|\hat{r}_{k,u}^{(i+1)} - r_{k,u} + \varsigma_k^{(i)}\|_2^2 \quad (3-30)$$

通过对公式 (3-27) 至公式 (3-30) 求导，结果如下：

$$\begin{cases} \rho(\hat{a}_{k,l}^{(i+1)} - a_{k,l} + \lambda_k^{(i)}) = 0, & k \in \mathcal{K} \\ \rho(\hat{a}_{k,u}^{(i+1)} - a_{k,u} + \theta_k^{(i)}) = 0, & k \in \mathcal{K} \\ \rho(\hat{a}_{k,BS}^{(i+1)} - a_{k,BS} + \mu_k^{(i)}) = 0, & k \in \mathcal{K} \\ \rho(\hat{r}_{k,u}^{(i+1)} - r_{k,u} + \varsigma_k^{(i)}) = 0, & k \in \mathcal{K} \end{cases} \quad (3-31)$$

从而，得出第  $(i+1)$  次迭代中全局变量  $\mathbf{a}_k$  和  $r_{k,u}$  的值，如公式 (3-32) 所示：

$$\begin{cases} a_{k,l}^{(i+1)} = \hat{a}_{k,l}^{(i+1)} + \lambda_k^{(i)}, & k \in \mathcal{K} \\ a_{k,u}^{(i+1)} = \hat{a}_{k,u}^{(i+1)} + \theta_k^{(i)}, & k \in \mathcal{K} \\ a_{k,BS}^{(i+1)} = \hat{a}_{k,BS}^{(i+1)} + \mu_k^{(i)}, & k \in \mathcal{K} \\ r_{k,u}^{(i+1)} = \hat{r}_{k,u}^{(i+1)} + \varsigma_k^{(i)}, & k \in \mathcal{K} \end{cases} \quad (3-32)$$

3) 拉格朗日乘子更新：这一步骤参照以上更新的变量来更新对偶变量  $\xi_k$ ：

$$\begin{cases} \lambda_k^{(i+1)} = \lambda_k^{(i)} + \hat{a}_{k,l}^{(i+1)} - a_{k,l}^{(i+1)}, & k \in \mathcal{K} \\ \theta_k^{(i+1)} = \theta_k^{(i)} + \hat{a}_{k,u}^{(i+1)} - a_{k,u}^{(i+1)}, & k \in \mathcal{K} \\ \mu_k^{(i+1)} = \mu_k^{(i)} + \hat{a}_{k,BS}^{(i+1)} - a_{k,BS}^{(i+1)}, & k \in \mathcal{K} \\ \varsigma_k^{(i+1)} = \varsigma_k^{(i)} + \hat{r}_{k,u}^{(i+1)} - r_{k,u}^{(i+1)}, & k \in \mathcal{K} \end{cases} \quad (3-33)$$

4) 二进制变量恢复：为了将非线性问题转化为线性问题，上一小节对变量  $\mathbf{a}_k$  进行松弛处理。现在，这一步骤需要将松弛变量恢复为二进制变量，方法如下：

$$a_{k,l}^* = \begin{cases} 1, & \hat{a}_{k,l} = \max\{\hat{a}_{k,l}, \hat{a}_{k,u}, \hat{a}_{k,BS}\}, & k \in \mathcal{K} \\ 0, & \text{other}, & k \in \mathcal{K} \end{cases} \quad (3-34)$$

其中， $a_{k,l}^*$  表示还原为二进制变量的局部本地卸载变量。相同地，对于变量  $a_{k,u}$  和  $a_{k,BS}$  的恢复也是如此。

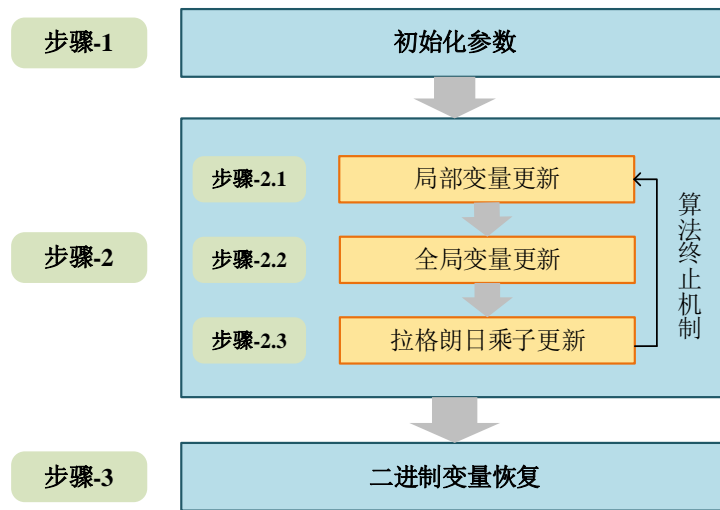


图 3-2 基于 ADMM 算法的任务卸载策略关键步骤



如算法 3-1 所示，其给出了在单无人机辅助移动边缘计算网络中基于 ADMM 算法的最优任务卸载策略的具体步骤。

算法 3-1 单无人机辅助移动边缘计算网络中基于 ADMM 算法的任务卸载策略

---

**输入:**  $K, \sigma^2, \alpha, T_k$

**输出:**  $a_{k,l}, a_{k,u}, a_{k,BS}, \mathbf{q}_U$

- 1: 初始化:  $\mathbf{g}_k^{(i)} = \{a_{k,l}^{(i)}, a_{k,u}^{(i)}, a_{k,BS}^{(i)}, r_{k,u}^{(i)}\}$ ,  $\xi_k$  的值,  $i = 0$
- 2: **repeat**
- 3:     **for**  $k = 1$  to  $K$  **do**
- 4:         通过问题 (3-26) 更新局部变量  $\hat{\mathbf{g}}_k^{(i+1)}$
- 5:     **end for**
- 6:     通过问题 (3-27) 至问题 (3-30) 更新全局变量  $\mathbf{g}_k^{(i+1)}$
- 7:     通过公式 (3-31) 更新拉格朗日乘子  $\xi_k^{(i+1)}$
- 8:      $i = i + 1$
- 9: **until** 变量  $\mathbf{g}_k$  收敛
- 10: **return** 全局变量  $\mathbf{g}_k^{(i+1)}$  到原问题 (3-21) 中的值
- 11: 二进制变量  $\mathbf{a}_k$  恢复
- 12: 输出恢复的二进制变量  $a_{k,l}^*$ ,  $a_{k,u}^*$  和  $a_{k,BS}^*$

---

### 3.4.3 复杂度分析

在基于 ADMM 算法的任务卸载策略中，局部变量更新的计算复杂度为  $O(vK)$ ，其中  $v$  表示问题 (3-26) 中计算所需的基本步骤。全局变量更新这一步骤包含  $4K$  个优化变量和  $(2K + 3)$  个约束，其计算复杂度可表示为  $O(4K(2K + 3))$ 。然后，拉格朗日乘子更新的计算复杂度可表示为  $O(\varepsilon K)$ ，其中  $\varepsilon$  表示问题 (3-33) 中所需的基本步骤数。因此，在单无人机辅助移动边缘计算网络中基于 ADMM 算法的任务卸载策略在每次迭代中的总的计算复杂度为  $O(vK + 4K(2K + 3) + \varepsilon K)$ 。综上，基于 ADMM 算法的任务卸载策略总的计算复杂度是  $I \cdot O(vK + 4K(2K + 3) + \varepsilon K) = I \cdot O(K^2)$ ，其中  $I$  标记为算法收敛所需的迭代次数。

### 3.5 仿真实验与结果分析

#### 3.5.1 仿真描述与参数配置

在本节中，首先，描述了仿真实验的运行环境。然后，展示了仿真实验的参数配置。本仿真实验基于 Matlab R2018b，运行的操作系统为 Windows10，计算机中央处理器平台为 Intel i5-9500，内存为 8GB，机械硬盘为 1TB。假设仿真实验场景由一个地面基站、一架无人机和  $K$  个地面用户设备组成，仿真实验区域为  $500\text{m} \times 500\text{m}$  的方形区域，地面基站位于原点位置<sup>[64]</sup>。表 3-1 展示了本仿真实验的相关参数。

表 3-1 参数设置

参数	数值	参数	数值
$\beta_0$	-80 dB	$\alpha$	8
$\sigma^2$	$10^{-16}$ W	$p_k$	50 mW
$p_k^{\text{loc}}$	100 mW	$p_k^{\text{UAV}}, p_k^{\text{BS}}$	200 mW
$f_k^{\text{loc}}$	0.5G cycles/s	$B$	40 MHz
$f_k^{\text{UAV}}$	5G cycles/s	$L_k$	[5, 10] MB
$f_k^{\text{BS}}$	30G cycles/s	$C_k$	[150, 200] cycles/bit

#### 3.5.2 仿真结果分析

假设地面用户设备总数量  $K = 10$ 。如图 3-3 所示，在几次迭代中，地面用户设备的总能耗迅速下降。为了与有效算法进行比较，以集中式卸载策略为基准，采用内点法求解问题 (3-19)，从而得到地面用户设备最优卸载决策。从图 3-3 可以看出，ADMM 算法在前 5 次迭代中迅速下降，之后趋于稳定。经过 10 次迭代，ADMM 算法逐步拟合集中式卸载策略。此外，为了找到全局最优解，ADMM 算法引入了辅助变量来避免局部最优。

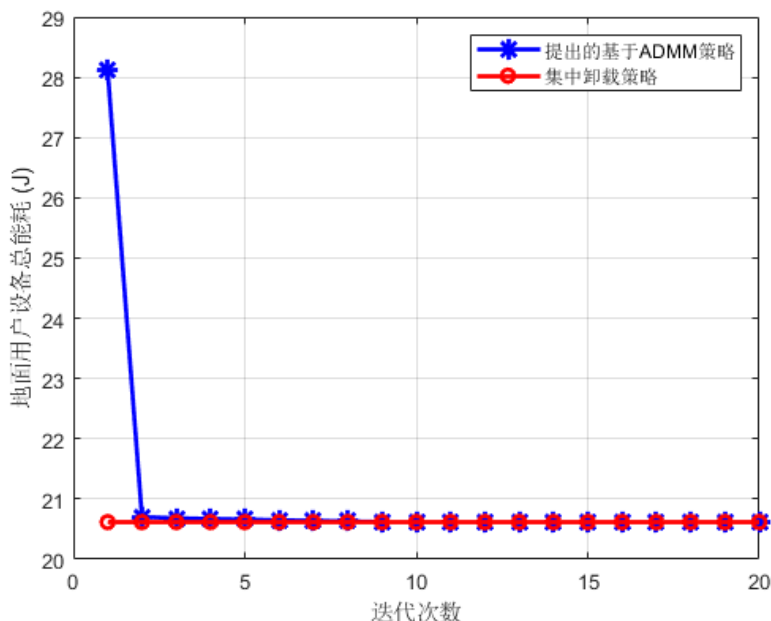


图 3-3 所提出的基于 ADMM 卸载策略收敛图

图 3-4 展示了随着地面用户设备数量增加，地面用户设备总能耗的变化情况。假设计算任务规模  $L_k = 7\text{MB}$ ，计算每比特所需 CPU 周期数  $C_k = 160\text{cycles/bit}$ 。从图 3-4 可以看出，对于不同的卸载策略，在相同的计算任务需求下，地面用户设备的总能耗也随着地面用户设备的数量的增加而线性增长。本章提出的基于 ADMM 算法的任务卸载策略与本地计算相比，总能耗减少了 73.29%；与基于随机算法的任务卸载策略相比<sup>[65]</sup>，总能耗减少了 59.31%；与基于贪心算法的任务卸载策略相比<sup>[66]</sup>，总能耗减少了 11.48%。同时，基于 ADMM 算法的任务卸载策略的性能与集中式任务卸载策略非常相近。

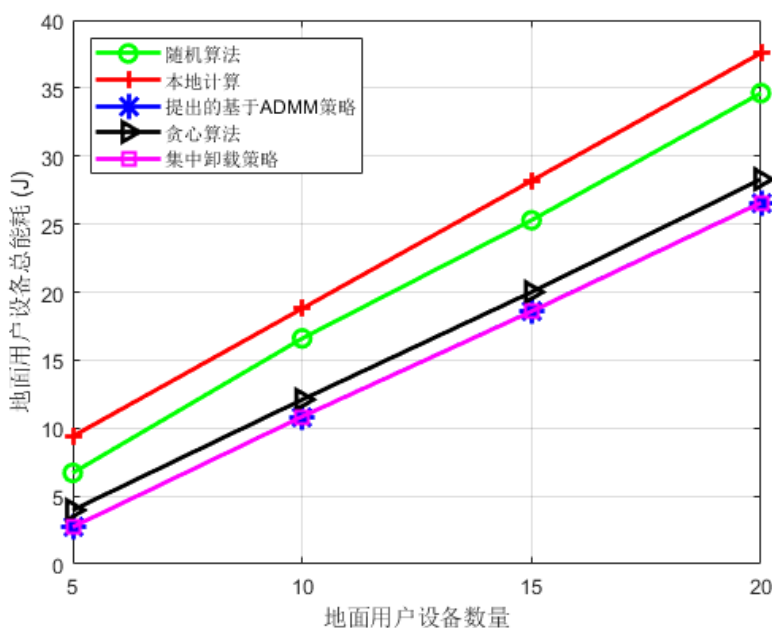


图 3-4 地面用户设备总能耗随地面用户设备数量变化图

图 3-5 展示了随着计算任务规模增加,不同卸载策略下地面用户设备总能耗的变化情况。假设地面用户设备总数量 $K = 10$ ,计算每比特所需 CPU 周期数 $C_k = 160\text{cycles/bit}$ 。如图 3-5 所示,随着计算任务规模的增加,四种卸载策略的能耗差距逐渐增大。原因是随着计算任务规模的增加和本地计算能力的限制,计算任务更有可能被卸载至无人机或地面基站。分析图中随机算法、贪心算法和 ADMM 算法的线性趋势,可以清楚地看到,数据量越大,本章提出的基于 ADMM 算法的任务卸载策略优势就越明显。此外,基于 ADMM 的任务卸载策略的曲线与集中式任务卸载策略曲线基本一致,这表明所提出的基于 ADMM 算法的任务卸载策略能够实现集中式算法的最优解。

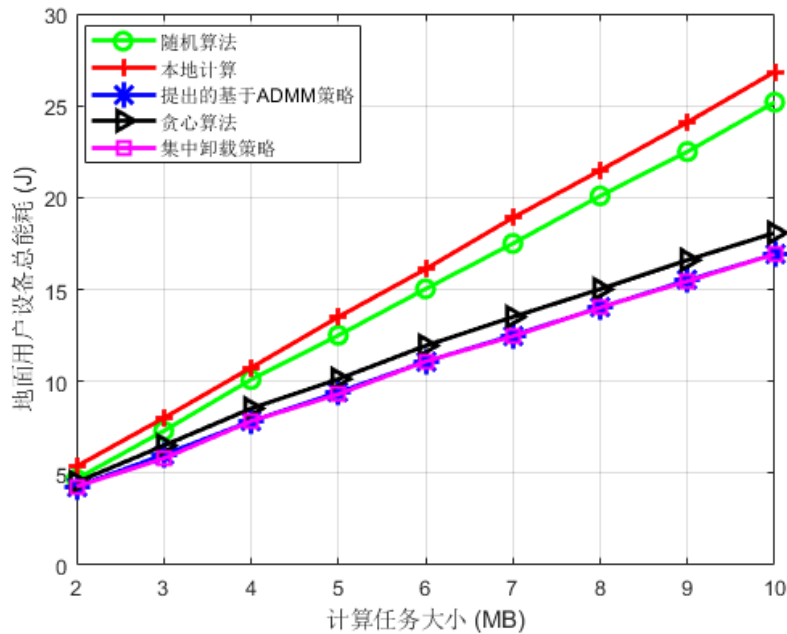


图 3-5 地面用户设备总能耗随计算任务规模大小变化图

图 3-6 描述了对应于不同计算每比特所需 CPU 周期数下的地面用户设备总能耗变化情况。假设地面用户设备总数量 $K = 10$ ,计算任务大小设置 $L_k = 8\text{MB}$ 。如图 3-6 所示,贪心算法和 ADMM 算法的线性趋势越来越近。原因是一旦地面用户设备计算需求增加,计算任务很可能被转移到具有强大计算资源的无人机或地面基站上。当地面用户设备与无人机或与地面基站之间的数据传输速率相差不大时,由于计算时延的限制,计算任务会被转移到计算能力最强的地面基站服务器上,导致基于两种算法的任务卸载策略趋于一致。此外,可以很容易地看到,本章提出的基于 ADMM 算法的任务卸载策略与集中式策略之间的差距是相当小的。

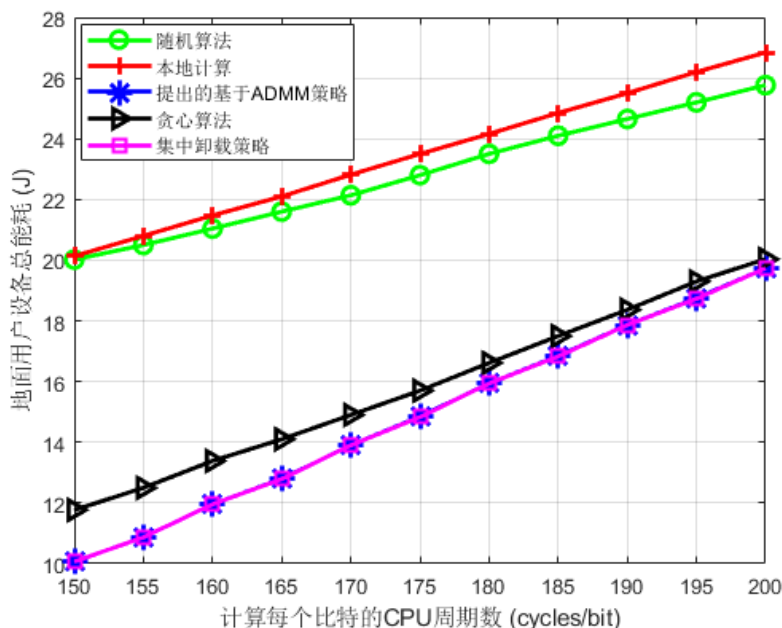


图 3-6 地面用户设备总能耗随计算每比特所耗不同 CPU 周期数变化图

为了进一步研究本章提出的基于 ADMM 算法的任务卸载策略，并考虑单无人机辅助移动边缘计算网络的实际应用场景，图 3-7 和图 3-8 阐述了不同卸载策略在随机分配计算任务规模和计算每比特所需的 CPU 周期数的条件下的性能表现。图 3-7 展示不同地面用户设备数量下不同卸载策略的性能对比。随着地面用户设备数量增长，本章提出的基于 ADMM 算法的任务卸载策略优势越加明显。图 3-8 描述了四种卸载策略的性能表现。其中，柱状图表示不同卸载策略下三种卸载方式的各自占比，折线表示不同卸载策略下地面用户设备总能耗。假设地面用户设备总数量  $K = 10$ 。从图 3-8 可以看出，与其他三种卸载策略相比，本章提出的基于 ADMM 算法的任务卸载策略更频繁地向无人机卸载。无人机的灵活性和强大的计算能力提供了低延迟，大大降低了地面用户设备的能耗。图 3-8 展示的结果为本章在单无人机辅助移动边缘计算网络中提出的基于 ADMM 算法的任务卸载策略的高效性提供了有力的论据。

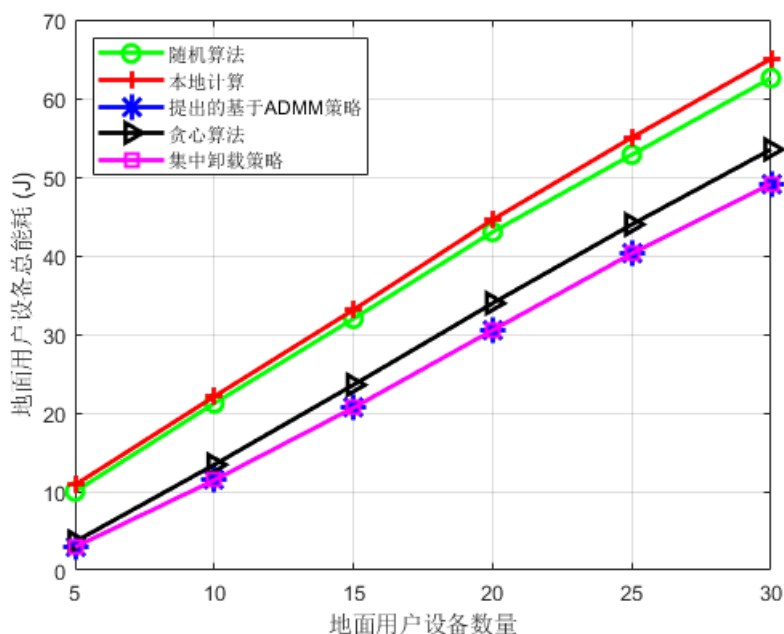


图 3-7 不同地面用户设备数量下不同卸载策略性能表现图

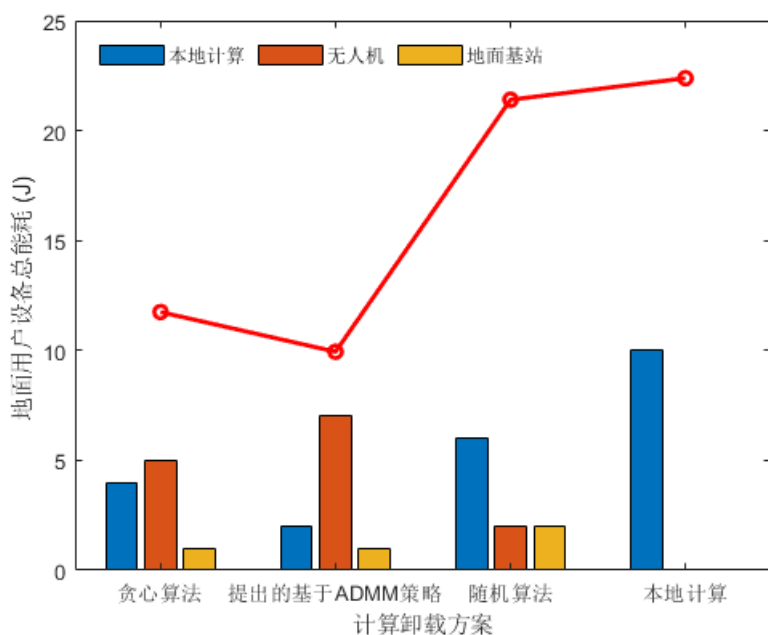


图 3-8 卸载策略性能分析图

### 3.6 本章小结

本章节研究了单无人机辅助移动边缘计算网络中的任务卸载问题，给出了一个以最小化地面用户设备总能耗为目标的约束优化问题。原问题是一类多约束的非线性问题，具有耦合的二元变量。为了解决这一问题，本章首先松弛二进制变量，然后引入辅助变量，将问题转化为线性问题。然后，提出一个基于 ADMM 算法的任务卸载策略解决优

化问题。最后，数值结果表明，与基准算法相比，提出的基于 ADMM 算法的任务卸载策略能有效降低地面用户设备总能耗，论证了本章提出的任务卸载策略的合理性和有效性。

## 第四章 基于多无人机辅助移动边缘计算 的任务卸载策略研究

### 4.1 问题描述与分析

近年来,随着无线蜂窝网络和物联网技术的发展,移动设备呈现爆炸式发展<sup>[56]</sup>。与此同时,智能移动设备的急剧增加导致了具有计算密集型特性的新应用的快速发展,如虚拟现实、智能汽车、导航和多媒体转换<sup>[57]</sup>。然而,移动设备通常体积小,电池容量有限,提供令人满意的计算服务和实现低服务延迟是一个挑战。因此,移动边缘计算被认为是缓解这些问题的关键技术。在移动边缘计算的帮助下,移动设备可以选择将其计算任务转移到附近具有强大计算能力的边缘服务器上,从而实现更低的能耗和更低的延迟需求<sup>[60][61]</sup>。

移动边缘计算服务器的位置通常是固定的,不能根据移动用户的需求灵活改变,限制了移动边缘计算的扩展。由于自然资源的开发和破坏,自然灾害的频繁发生可能会破坏地面上的基本通信设施。这使得救援通信工作变得困难,传统的移动边缘计算模型的缺点越来越明显。与传统基础通信设施相比,无人机具有高度灵活性、低成本性和通信可靠性,可作为移动边缘计算服务器,从而提高传统移动边缘计算模式的应用可扩展性。

因此,无人机技术与移动边缘计算模型的联合开发,即利用无人机增强移动边缘计算能力,是移动边缘计算发展的一个很有前景的方向。第一部分工作中研究的是基于单无人机辅助移动边缘计算的任务卸载策略,其中单无人机辅助移动边缘计算网络可以应用于赛事直播、视频拍摄等小范围的场景,具有一定的局限性。因此,本章将第三章的单无人机辅助移动边缘计算网络拓展为多无人机辅助移动边缘计算网络,延展无人机辅助移动边缘计算的应用场景。

基于上述分析,在基于单无人机辅助移动边缘计算研究的基础上,本章提出了一种基于多无人机辅助移动边缘计算的任务卸载策略。目前,联合优化无人机位置和计算卸载决策的工作相对较少,且多数文献默认地面设备位置固定。因此,考虑地面设备的移动性是本章研究工作的动力之一。为此,本章以最小化所有地面移动设备的总能耗为目标,研究了多无人机辅助移动边缘计算网络中的任务卸载决策和无人机飞行位置的联合优化问题。该优化问题是一个混合整数非凸问题,本章提出了基于 A3C 算法的任务卸



载策略, 该算法异步训练在不同环境下计算任务的最优卸载决策。然后将训练参数统一上传到全局网络中进行更新并不断训练, 最终获得最优的网络参数。

## 4.2 系统模型

### 4.2.1 网络模型

本章考虑一个多无人机辅助移动边缘计算网络模型, 其中包括 $M$ 个部署服务器的无人机, 记集合 $\mathcal{M} = \{1, 2, 3, \dots, M\}$ , 和 $K$ 个地面移动设备, 记集合 $\mathcal{K} = \{1, 2, 3, \dots, K\}$ 。网络模型如图 4-1 所示。假设在一定的时间段内, 能量有限的无人机为 $K$ 个地面移动设备提供计算卸载服务。在不失一般性的条件下, 在该时间段内,  $K$ 个地面移动设备与 $M$ 个无人机一对一服务, 并且所有的任务必须保证在规定的时段 $L$ 内完成。为了模拟地面用户设备和无人机的移动性, 本章将周期 $L$ 内非执行任务的计算时间划分为 $T$ 帧, 每帧的时间一致, 记为 $t \in \{0, 1, 2, \dots, T\}$ 。本章假设无人机飞行在固定高度 $H$ , 不需要频繁起落, 通过时分多址协议与地面移动设备在每帧内保持通信。为了便于说明, 本章使用三维笛卡尔坐标系来模拟每个节点的位置, 坐标单位为米。记无人机 $m$ 的 3D 位置坐标为 $\mathbf{U}_m^u(t) = (x_m^u(t), y_m^u(t), H)$ , 同时相邻无人机需要满足约束 $d_{\min}^{\text{uu}} \leq d_{m_1 m_2}^{\text{uu}}(t)$ , 其中 $d_{m_1 m_2}^{\text{uu}}(t) = \sqrt{(x_{m_1}^u(t) - x_{m_2}^u(t))^2 + (y_{m_1}^u(t) - y_{m_2}^u(t))^2}$ ,  $\forall m_1, m_2 \in \mathcal{M}, m_1 \neq m_2$  表示两架相邻无人机之间的距离, 而  $d_{\min}^{\text{uu}}$  表示两架相邻无人机之间允许的最小距离<sup>[43]</sup>。

假设在多无人机辅助移动边缘计算网络中, 所有的地面移动设备在 $t = 0$ 时的位置随机分布, 并且在 $\Delta_{t, t+1}$ 内不改变其位置。以高斯-马尔可夫随机模型为核心<sup>[41]</sup>, 地面移动设备 $k$ 在第 $t$ 帧时的移动速度和移动方向可分别表示为公式 (4-1) 和公式 (4-2)。

$$v_k(t) = \tau_1 v_k(t-1) + (1 - \tau_1) \bar{v}_k + \sqrt{1 - \tau_1^2} \Omega_k \quad (4-1)$$

$$\alpha_k(t) = \tau_2 \alpha_k(t-1) + (1 - \tau_2) \bar{\alpha}_k + \sqrt{1 - \tau_2^2} \Psi_k \quad (4-2)$$

其中,  $0 \leq \tau_1, \tau_2 \leq 1$  表示调整前一帧状态的参数,  $\bar{v}_k$  和  $\bar{\alpha}_k$  分别描述地面移动设备 $k$ 的平均移动速度和移动方向,  $\Omega_k$  和  $\Psi_k$  遵循两个不同均值-方差的不相关随机高斯分布来模拟地面移动设备 $k$ 的随机移动性。根据公式 (4-1) 和公式 (4-2), 地面移动设备 $k$ 在第 $t$ 帧时的 3D-X 轴和 3D-Y 轴可分别由公式 (4-3) 和公式 (4-4) 表示。

$$x_k(t) = x_k(t-1) + v_k(t-1) \Delta_{t-1, t} \cos(\alpha_k(t-1)) \quad (4-3)$$

$$y_k(t) = y_k(t-1) + v_k(t-1) \Delta_{t-1, t} \sin(\alpha_k(t-1)) \quad (4-4)$$

综上,地面移动设备 $k$ 在第 $t$ 帧时的 3D 位置坐标为 $\mathbf{G}_k(t) = (x_k(t), y_k(t), 0)$ 。网络模型局部的三维可视化模型见图 4-1 右侧。

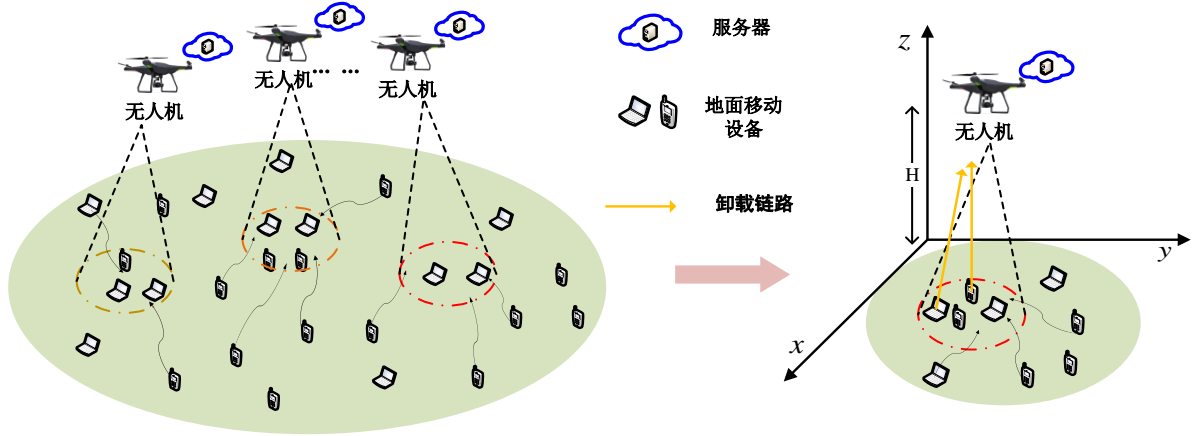


图 4-1 多无人机辅助移动边缘计算网络模型

#### 4.2.2 通信模型

在本章中,地面移动设备与无人机之间的通信链路采用视距传输方式。为简单起见,本章将通信链路视为一个清晰的视距通信链路,然后忽略衰落的影响,如阴影和小尺度衰落。所以,地面移动设备 $k$ 与无人机 $m$ 之间的通信链路可由公式(4-5)表示。

$$h_{k,m}^{\text{ul}}(t) = \frac{\beta_0}{(x_m^{\text{u}}(t) - x_k(t))^2 + (y_m^{\text{u}}(t) - y_k(t))^2 + H^2} \quad (4-5)$$

其中,  $\beta_0$ 是参考距离为 1 米时的信道功率增益。

为了提高信噪比,地面移动设备 $k$ 在一定时帧内将计算任务卸载至无人机 $m$ 的上行数据传输速率计算方式为公式(4-6)。

$$R_{k,m}^{\text{ul}}(t) = B \log_2 \left( 1 + \frac{h_{k,m}^{\text{ul}}(t)p_k}{\sigma^2} \right) \quad (4-6)$$

其中,  $B$ 表示可用信道带宽,  $p_k$ 是地面移动设备 $k$ 的传输功率,  $\sigma^2$ 标记为高斯噪声功率。

#### 4.2.3 计算模型

假设所有分布在多无人机辅助移动边缘计算网络中的地面移动设备都会产生一个计算密集型、时延敏感型的计算任务 $W_k = \{L_k, C_k, t_k^{\text{max}}\}$ , 其中 $L_k$ 表示用于计算任务的数据规模,  $C_k$ 表示计算每比特所需的 CPU 周期数,  $t_k^{\text{max}}$ 表示最大可容忍的任务是时延。这

些无人机相互合作，为地面移动设备提供计算服务。本章中， $a_{k,m} \in \{0,1\}$ 记为地面移动设备 $k$ 的任务卸载决策变量。其中 $a_{k,m} = 0$ 表示地面移动设备 $k$ 选择在本地执行计算任务， $a_{k,m} = 1$ 表示地面移动设备 $k$ 的计算任务卸载至无人机 $m$ 进行计算。

**1) 本地计算：**当地面移动设备 $k$ 决定在本地执行计算时，其本地计算执行时间可以由公式（4-7）表示。

$$t_k^{\text{loc}} = \frac{L_k C_k}{f_k^{\text{loc}}} \quad (4-7)$$

其中， $f_k^{\text{loc}}$ 是地面移动设备 $k$ 的本地计算能力。相应地，本地计算所消耗的能量可由公式（4-8）计算表达。

$$E_k^{\text{loc}} = \rho_k^{\text{loc}} L_k C_k (f_k^{\text{loc}})^2 \quad (4-8)$$

其中， $\rho_k^{\text{loc}}$ 标记为地面移动设备 $k$ 的芯片相关系数。

**2) 计算任务卸载至无人机：**当地面移动设备 $k$ 移动到无人机 $m$ 的信号覆盖区域，即满足约束  $d_{k,m}^{\text{gu}}(t) \leq R$ ，其中  $d_{k,m}^{\text{gu}}(t) = \sqrt{(x_k(t) - x_m^{\text{u}}(t))^2 + (y_k(t) - y_m^{\text{u}}(t))^2}$ ,  $\forall k \in \mathcal{K}, \forall m \in \mathcal{M}$ 表示地面移动设备 $k$ 与无人机 $m$ 之间的水平距离， $R = H \tan \vartheta$ 表示无人机的信号覆盖半径， $\vartheta$ 是无人机天线仰角。此时，无人机 $m$ 成为地面移动设备 $k$ 的一个计算任务卸载选择对象。当地面移动设备 $k$ 在多个无人机的信号覆盖范围内，地面移动设备 $k$ 随机选择一架无人机作为计算任务卸载对象。将计算任务从地面移动设备迁移到无人机主要分为三个步骤：（1）地面移动设备将计算任务卸载至选定的无人机；（2）被选定的无人机接收计算任务并进行计算；（3）被选定的无人机将结果返回给相应的地面移动设备。此外，由于返回的数据量很小，因此步骤三可以忽略不计。因此，地面移动设备 $k$ 卸载计算任务至无人机 $m$ 的数据传输时间、发送数据消耗的能量和无人机接收数据消耗的能量分别由公式（4-9）、公式（4-10）和公式（4-11）表示。

$$t_{k,m}^{\text{tr}}(t) = \frac{L_k}{R_{k,m}^{\text{ul}}(t)} \quad (4-9)$$

$$E_{k,m}^{\text{tr}}(t) = p_k \frac{L_k}{R_{k,m}^{\text{ul}}(t)} \quad (4-10)$$

$$E_{k,m}^{\text{re}}(t) = p_m^{\text{u}} \frac{L_k}{R_{k,m}^{\text{ul}}(t)} \quad (4-11)$$

其中， $p_m^{\text{u}}$ 表示无人机 $m$ 的接收功率。

#### 4.2.4 飞行模型

**1) 边缘计算的能耗:** 为了使能量有限的无人机能够持续工作, 需要对其能量进行约束使用<sup>[40]</sup>。本章将无人机的能耗分为三个主要部分: (1) 接收数据能耗和计算能耗(统称为边缘计算能耗); (2) 无人机飞行能耗; (3) 无人机盘旋能耗。假设 $f_m^u$ 和 $\rho_m^u$ 分别表示无人机 $m$ 的计算能力和芯片相关系数。相应的, 无人机 $m$ 接收的任务计算所需时间及能量消耗可分别由公式(4-12)和公式(4-13)计算表示。

$$t_{k,m}^{\text{cal}} = \frac{L_k C_k}{f_m^u} \quad (4-12)$$

$$E_{k,m}^{\text{cal}} = \rho_m^u L_k C_k (f_m^u)^2 \quad (4-13)$$

根据公式(4-12)和公式(4-13), 边缘计算的能量消耗可由公式(4-14)推导得出。

$$\begin{aligned} E_{k,m}^{\text{edg}}(t) &= E_{k,m}^{\text{re}}(t) + E_{k,m}^{\text{cal}} \\ &= p_m^u \frac{L_k}{R_{k,m}^{\text{ul}}(t)} + \rho_m^u L_k C_k (f_m^u)^2 \end{aligned} \quad (4-14)$$

**2) 无人机飞行能耗:** 假设无人机飞行在固定高度 $H$ , 即本章中无人机的重力势能没有变化。因此, 无人机飞行能耗只需要考虑无人机的动能变化。无人机 $m$ 在第 $t$ 帧时的飞行速度和能耗如公式(4-15)和公式(4-16)所示。

$$v_m^u(t) = \frac{U_m^u(t) - U_m^u(t-1)}{\Delta} \quad (4-15)$$

$$E_m^f(t) = \frac{1}{2} w \Delta \|v_m^u(t)\|^2 \quad (4-16)$$

其中,  $w$ 是无人机的有效重量,  $\Delta$ 记为每帧的持续时间。

**3) 无人机盘旋能耗:** 在无人机信号覆盖范围内, 收到来自地面移动设备的任务卸载请求时, 无人机将从飞行状态切换到悬停状态, 并在整个边缘计算周期内保持悬停状态。在本章中, 计算任务卸载主要分为两个阶段: 任务卸载和计算任务执行计算, 其表达公式如(4-17)所示。

$$t_{k,m}^{\text{edg}}(t) = t_{k,m}^{\text{tr}}(t) + t_{k,m}^{\text{cal}} \quad (4-17)$$

为了简化问题分析, 无人机 $m$ 的悬停能耗 $E_m^{\text{st}}$ 默认为常量。

综上所述, 在无人机 $m$ 总能量 $E_m^u$ 受限的前提下, 无人机运行需要满足能量约束公式(4-18)。

$$\sum_{k \in \mathcal{K}} a_{k,m} (E_{k,m}^{\text{edg}}(t) + E_m^{\text{st}} t_{k,m}^{\text{edg}}(t)) + \sum_{t=1}^T E_m^f(t) \leq E_m^u \quad (4-18)$$

### 4.3 优化问题描述

根据上述多无人机辅助移动边缘计算网络模型，本小节确立了目标优化问题。在满足防碰撞约束、信号覆盖约束、能量约束和时延约束的前提下，通过联合优化卸载决策变量和无人机位置，实现多无人机辅助移动边缘计算网络系统中所有地面移动设备的总能耗最小化的目标。记卸载决策变量  $\mathbf{a} = \{a_{k,m}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}\}$ ，则目标优化问题表示如下：

$$\begin{aligned}
 & \min_{\mathbf{a}, \{(x_m^u, y_m^u)\}} \sum_{k=1}^K \sum_{m=1}^M \left( (1 - a_{k,m}) E_k^{\text{loc}} + a_{k,m} E_{k,m}^{\text{tr}} \right) \\
 \text{s.t. C1: } & \sum_{k \in \mathcal{K}} a_{k,m} \left( E_{k,m}^{\text{edg}}(t) + E_m^{\text{st}} t_{k,m}^{\text{edg}}(t) \right) + \sum_{t=1}^T E_m^{\text{f}}(t) \leq E_m^{\text{u}} \\
 \text{C2: } & a_{k,m} d_{k,m}^{\text{gu}}(t) \leq R, \forall k \in \mathcal{K}, \forall m \in \mathcal{M} \\
 \text{C3: } & d_{\min}^{\text{uu}} \leq d_{m_1 m_2}^{\text{uu}}(t), \forall m_1, m_2 \in \mathcal{M}, m_1 \neq m_2 \\
 \text{C4: } & (1 - a_{k,m}) t_k^{\text{loc}} + a_{k,m} (t_{k,m}^{\text{tr}}(t) + t_{k,m}^{\text{cal}}) \leq t_k^{\text{max}} \\
 \text{C5: } & a_{k,m} \in \{0, 1\}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M} \\
 \text{C6: } & \sum_{m=1}^M a_{k,m} = 1, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}
 \end{aligned} \tag{4-19}$$

其中，约束 C1 表示无人机能量的使用限制；约束 C2 表示无人机信号的覆盖范围；约束 C3 保证相邻无人机之间的安全距离以防止碰撞；约束 C4 表示地面移动设备计算任务所允许的最大延迟。

如问题（4-19）所示，问题（4-19）是一个混合整数非线性问题，一般情况下很难得到最优解。因此，本章需要设计合适的算法来有效地解决这类问题<sup>[63]</sup>。在接下来的小节中，本章提出了一种基于 A3C 算法的计算卸载策略来获得次优解。

### 4.4 优化问题求解

本章使用深度强化学习算法<sup>[54]</sup>来探索未知环境，其中地面移动设备通过不同的任务卸载决策，无人机通过尝试不同的移动，从反馈中学习。全局网络不断优化任务卸载决策和无人机位置移动，直到得到一个次优解。

#### 4.4.1 基于 A3C 算法的任务卸载策略

在多无人机辅助移动边缘计算网络模型中，在有无人机信号覆盖的区域的移动设备，其计算任务可以在每帧内选择本地计算或卸载至无人机计算。在考虑防碰撞约束、能量约束和延迟约束的情况下，本章目标是最大化所有地面移动设备的总能耗。本小节将地面移动设备卸载任务建模为马尔可夫决策过程（Markov Decision Process, MDP）。

一个 MDP 过程主要包含五个部分，记为  $MDP = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$ ，其中  $\mathcal{S}$  为环境状态的集合， $\mathcal{A}$  描述了卸载决策行为的集合， $\mathcal{P}$  表示状态转移概率， $\mathcal{R}$  表示奖励函数， $\gamma$  为衰减系数。基于多无人机辅助移动边缘计算网络的地面移动设备卸载决策详细的马尔可夫决策过程如下。

多无人机辅助移动边缘计算网络的状态空间表示如下：

$$\mathcal{S} = \{s_t | s_t = \{U_m^u(t), G_k(t), E_m^u\}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}\} \quad (4-20)$$

多无人机辅助移动边缘计算网络中的动作空间包括本地计算和卸载至无人机两种动作，表示如下：

$$\mathcal{A} = \{a_{k,m}(t) | a_{k,m}(t) \in \{0,1\}\}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M} \quad (4-21)$$

多无人机辅助移动边缘计算网络中地面移动设备状态转移和动作决策只与地面移动设备位置，无人机位置以及无人机的能量状态有关，所以状态转移概率可以表示为：

$$\mathcal{P}_{ss'} = \mathcal{P}(s_{t+1} = s' | s_t = s)。 \quad (4-22)$$

为了最大化所有地面移动设备的总能耗，本小节考虑设计一个奖励函数，当地面移动设备在当前帧状态下所采取的行动满足问题（4-19）中约束 C1 至约束 C6 时，奖励函数将给予负奖励。简单地说，奖励函数可以计算为：

$$r(s_t, a_{k,m}(t)) = -((1 - a_{k,m})E_k^{\text{loc}} + a_{k,m}E_{k,m}^{\text{tr}}) \quad (4-23)$$

相反，如果地面移动设备违反约束，将予以惩罚。例如，地面移动设备的本地计算违反了延迟约束，将对本地计算能耗进行如下处理：

$$r(s_t, a_{t,m}) = -\frac{t_k^{\text{loc}}}{t_k^{\text{max}}} E_k^{\text{loc}} \quad (4-24)$$

对于优化问题（4-19），可以看出二进制决策变量的值序列直接影响优化问题的次优解。本小节将环境状态传递给本地网络，得到一个任务卸载决策序列，然后采用奖励函数累积奖励值（Reward）。多个线程以这种方式异步进行，将训练参数上传给全局网络来协调。最终得到了最优网络参数和次优奖励值。如图 4-2 所示，展示了基于 A3C 算

法的任务卸载策略训练框架。如算法 4-1 所示，给出了多无人机辅助移动边缘计算网络中基于 A3C 的任务卸载策略的最优网络参数的具体步骤。

---

算法 4-1 基于 A3C 算法的任务卸载策略——任意单线程执行过程

---

**输入：** 全局共享参数 $\theta$ （actor 网络参数）， $\omega$ （critic 网络参数）；  
 本线程网络参数 $\theta'$ （actor 网络参数）， $\omega'$ （critic 网络参数）；  
 奖励值衰减因子 $\gamma$ ；全局共享计数器 $N$ ；全局最大计数器 $N_{\max}$ ；

**输出：** 全局共享参数 $\theta, \omega$ ；回报值 $\mathcal{R}(s_n, a_n)$

- 1: 初始化本地计数器  $n = 1$
- 2: **repeat**
- 3:     重置 actor 和 critic 的梯度:  $d\theta \leftarrow 0, d\omega \leftarrow 0$
- 4:     使用全局参数同步更新本线程网络参数:  $\theta' = \theta, \omega' = \omega$
- 5:     令  $n_{\text{start}} = n$ ，初始化状态  $s_n$
- 6:     **repeat**
- 7:         基于策略  $\pi(a_n|s_n; \theta')$ ，采样获取动作  $a_n$
- 8:         执行动作  $a_n$ ，获得回报  $r_n$ ，进入新状态  $s_{n+1}$
- 9:          $N \leftarrow N + 1, n \leftarrow n + 1$
- 10:     **until**  $s_n$  为终止状态，或  $n - n_{\text{start}} == n_{\max}$
- 11:     计算最后一个时间序列位置状态  $s_n$  的  $Q(s, n)$ :

$$Q(s, n) = \begin{cases} 0, & s_n \text{ 是终止状态} \\ V(s_n, \omega'), & \text{其他} \end{cases}$$

- 12:     **for**  $i \in (n - 1, n - 2, \dots, n_{\text{start}})$  **do**
- 13:          $Q(s, i) = r_i + \gamma Q(s, i + 1)$
- 14:         计算 Actor 的本地累计梯度:
- 15:         计算 critic 的本地累计梯度:

$$d\theta \leftarrow d\theta + \nabla_{\theta'} \log \pi(a_n|s_n; \theta')(Q(s, i) - V(s_i, \omega'))$$

$$d\omega \rightarrow d\omega + \frac{\partial (Q(s, i) - V(s_i, \omega'))^2}{\partial \omega'}$$

- 17:     **end for**
  - 18:     使用本地累计梯度  $d\theta$  和  $d\omega$  异步更新全局参数  $\theta$  和  $\omega$
  - 19:     **until**  $N > N_{\max}$
-

#### 4.4.2 生成任务卸载决策

本小节特别介绍了任意一个线程的环境状态序列和动作序列的交互过程。地面移动设备在第 $t$ 帧时生成计算任务后，将地面移动设备的位置、无人机的位置和无人机的能量状态作为一组状态。然后，将状态序列输入到线程的本地网络模型中，网络训练线程生成一个动作序列，动作序列的元素对应于每个地面移动设备的任务卸载决策。

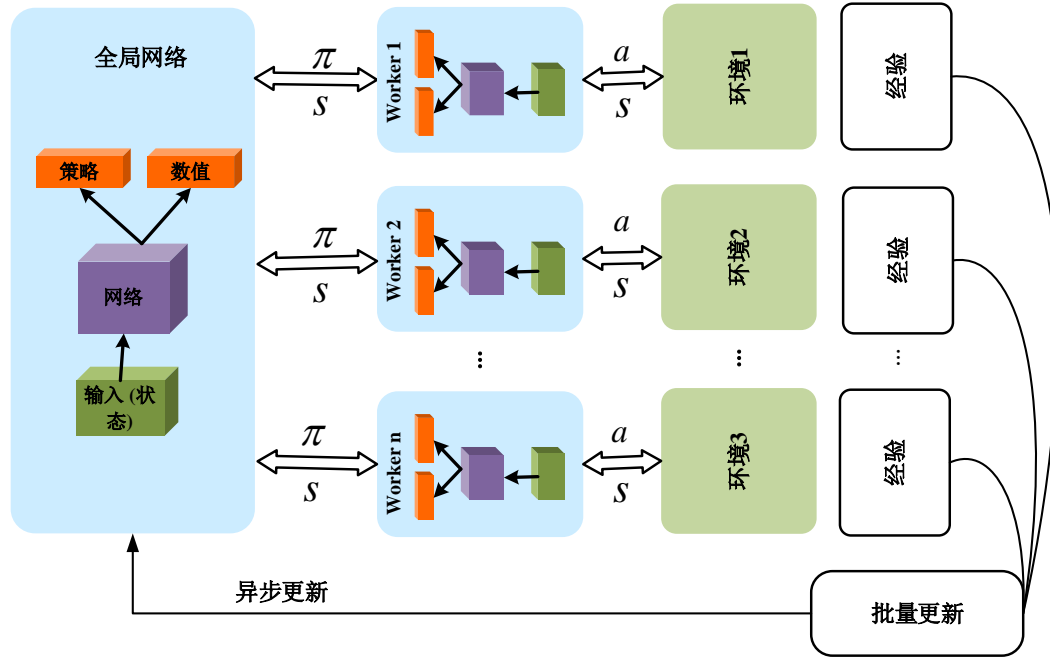


图 4-2 基于 A3C 算法的任务卸载策略训练框架

### 4.5 仿真实验与结果分析

#### 4.5.1 仿真描述与参数配置

在本节中，将给出仿真结果来评估本章提出的基于 A3C 算法的任务卸载策略的性能。本仿真实验将与以下三种常用的基线算法进行比较。三种算法的基本思想如下：

(1) 贪心算法：当地面移动设备处于无人机的信号覆盖区域时，地面移动设备根据本地计算时延和传输时延的大小选择本地计算还是卸载至无人机计算<sup>[67]</sup>。

(2) 随机算法：无人机信号覆盖范围内的地面移动设备可以随机选择计算任务执行的对象，即本地计算或卸载至无人机计算<sup>[68]</sup>。

(3) DQN 算法：神经网络接受环境状态（即地面移动设备位置、无人机位置和无人机能量状态）来计算值函数，然后使用 $\epsilon$ -贪心策略输出任务卸载决策<sup>[69]</sup>。

本仿真实验基于 Visual Studio Code，运行的操作系统为 Windows10，计算机中央处



理器平台为 Intel i5-9500，内存为 8GB，机械硬盘为 1TB。假设仿真实验场景由  $M$  架无人机和  $K$  个地面移动设备组成，仿真实验区域为  $300m \times 300m$  的方形区域<sup>[68]</sup>。表 4-1 展示了本仿真实验的相关参数。

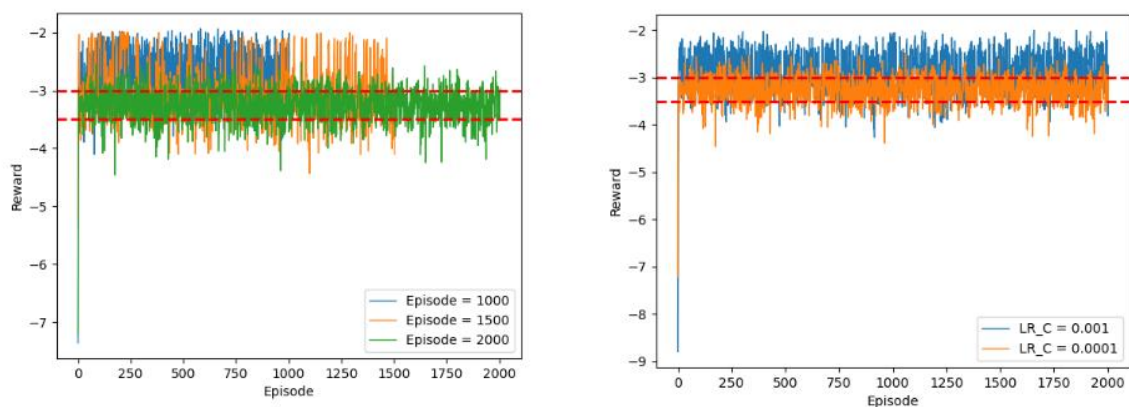
表 4-1 参数设置

参数	数值	参数	数值
$\beta_0$	-50 dB	$\vartheta$	$\pi/4$
$\sigma^2$	$10^{-16}$ W	$p_k$	50 mW
$\rho_k^{\text{loc}}, \rho_m^{\text{u}}$	$10^{-28}$	$p_m^{\text{u}}$	50 mW
$f_k^{\text{loc}}$	0.5G cycles/s	$B$	40 MHz
$f_m^{\text{u}}$	5G cycles/s	$L_k$	[5, 10] MB
$d_{\min}^{\text{uu}}$	4 m	$C_k$	[150, 200] cycles/bit

#### 4.5.2 仿真结果分析

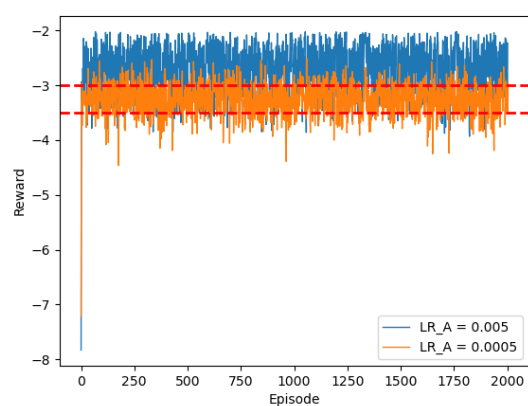
假设地面移动设备数量  $K = 20$ ，无人机数量  $M = 3$ 。如图 4-3 所示，经过几次迭代地面移动设备总能耗迅速下降。A3C 算法的异步特性使得奖励值在一个区间内振荡，需要尽可能减少振荡区间。当地面移动设备规模较大时，奖励值在 0.5 范围内波动是可以接受的。从图 4-3 (a) 可以看出，随着迭代次数的增加，振荡间隔逐渐减小。此时，可以将其视为逐渐收敛的奖励价值。从图 4-3 (b) 可以看出，奖励值的振荡间隔迅速减小，说明批评家网络学习率的降低可以降低奖励值的振荡间隔，加速奖励值的收敛。无独有偶，降低了行动者网络的学习率，得到的图 4-3 (c) 中奖励值也可以快速收敛。另外，由于奖励值在一定范围内振荡的特点，本章使用振荡范围上下限的平均值作为奖励值的最终结果。

假设无人机数量  $M = 3$ ，计算任务规模  $L_k = 8\text{MB}$ ，计算每比特所需 CPU 周期数  $C_k = 160\text{cycles/bit}$ 。如图 4-4 所示，展示了不同地面移动设备数量下最小化地面移动设备总能耗的变化情况。对于不同的卸载策略，地面移动设备总能耗也随着地面移动设备数量的增加而线性增长。当无人机覆盖率较低且地面移动设备数量较少时，很难满足所有地面移动设备都在无人机信号覆盖范围内。在图 4-4 中，当地面移动设备数量  $K = 5$  时，四种卸载策略下的地面移动设备总能耗相差不大。但可以看出，在相同的计算任务需求下，地面移动设备数量越大，地面移动设备总能耗就越大，本章提出的基于 A3C 算法的任务卸载策略更具有优势。



(a) 不同 episode 数对比

(b) 不同 critic 网络学习率对比



(c) 不同 actor 网络学习率对比

图 4-3 所提出的基于 A3C 算法的任务卸载策略收敛图

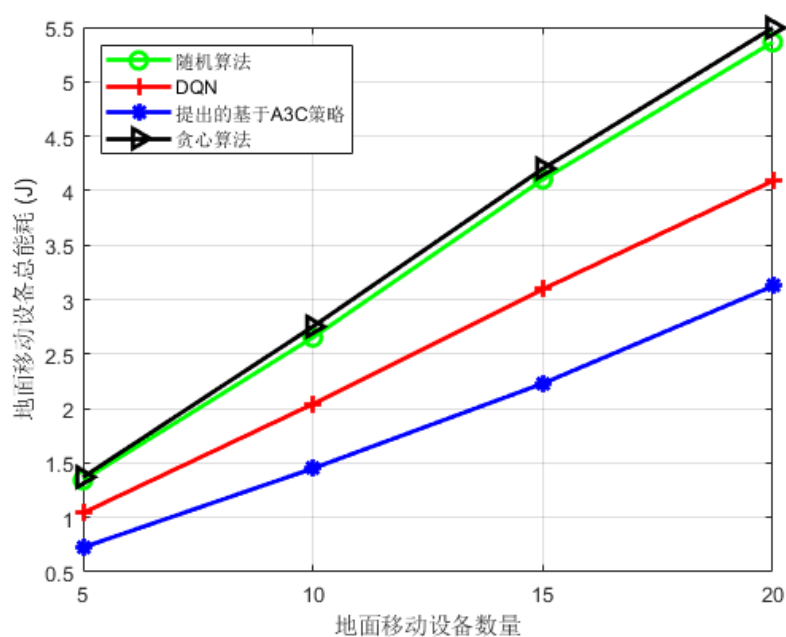


图 4-4 地面移动设备总能耗随地面移动设备数量变化图

假设无人机数量 $M = 3$ ，地面移动设备数量 $K = 20$ ，计算每比特所需 CPU 周期数 $C_k = 160\text{cycles/bit}$ 。如图 4-5 所示，展示了不同计算任务规模下地面移动设备总能耗的变化情况。从图 4-5 可以看出，随着计算任务规模的增加，四种卸载策略的能耗差距逐渐增大。其原因是随着数据规模的增大，由于计算延迟的限制，随机策略和贪婪策略逐渐失去作用。分析图中随机算法、贪心算法、DQN 算法和 A3C 算法的线性趋势，可以清楚地看出，计算任务规模越大，本章提出的任务卸载策略的优势就越明显。

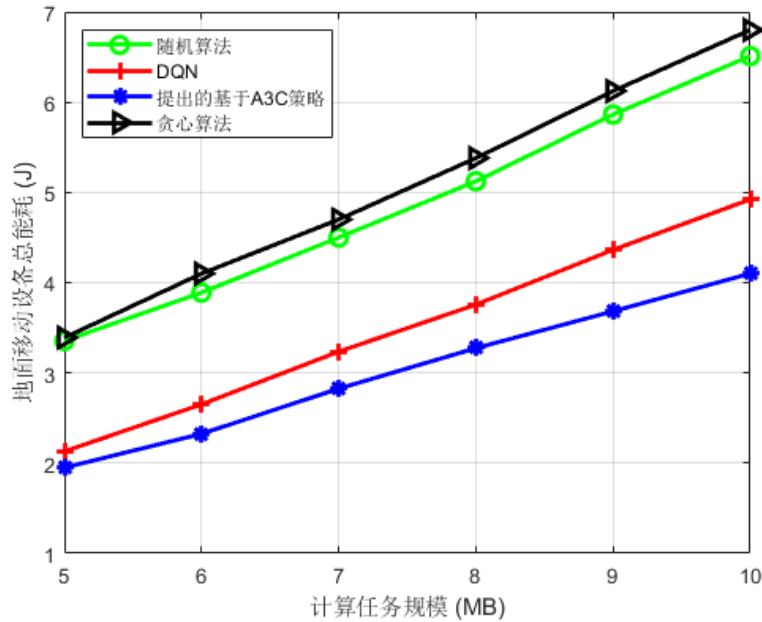


图 4-5 地面移动设备总能耗随计算任务规模大小变化图

假设无人机数量 $M = 3$ ，地面移动设备数量 $K = 20$ ，计算任务规模 $L_k = 8\text{MB}$ 。如图 4-6 所示，展示了计算每比特所需不同 CPU 周期数下地面移动设备总能耗的变化情况。在图 4-6 中，基于深度强化学习算法的卸载策略相较于基于随机算法和贪心算法的卸载策略有着明显的优势。原因是计算每比特所所需的 CPU 周期数越大，基于传统算法的卸载策略很难满足时延敏感型任务的计算要求。

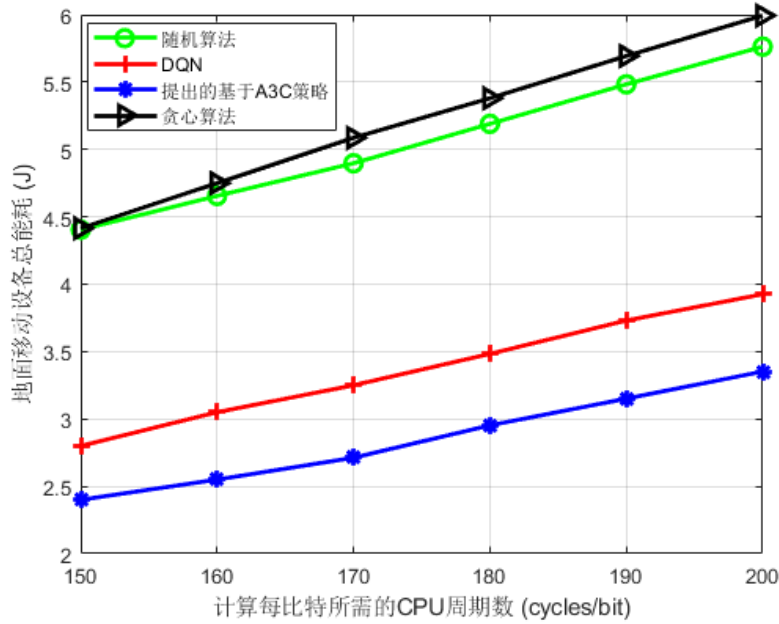


图 4-6 地面移动设备总能耗随计算每比特所需 CPU 周期数变化图

为进一步研究本文提出的基于 A3C 算法的任务卸载策略,并结合多无人机辅助移动边缘计算网络的实际应用场景。假设无人机数量 $M = 3$ 。如图 4-7 所示,展示了随机分配计算任务规模大小和计算每比特所需的 CPU 周期数的条件下,不同地面移动设备数量下不同卸载策略的性能对比。随着地面移动设备数量增长,本章提出的基于 A3C 算法的任务卸载策略优势越加显著。原因是地面移动设备数量的增长,增加了算法的计算复杂度,传统算法难以提供高效的卸载策略,本章提出的基于 A3C 算法的任务卸载策略能够有效降低计算复杂度,减少计算任务响应时间,降低地面移动设备总能耗。

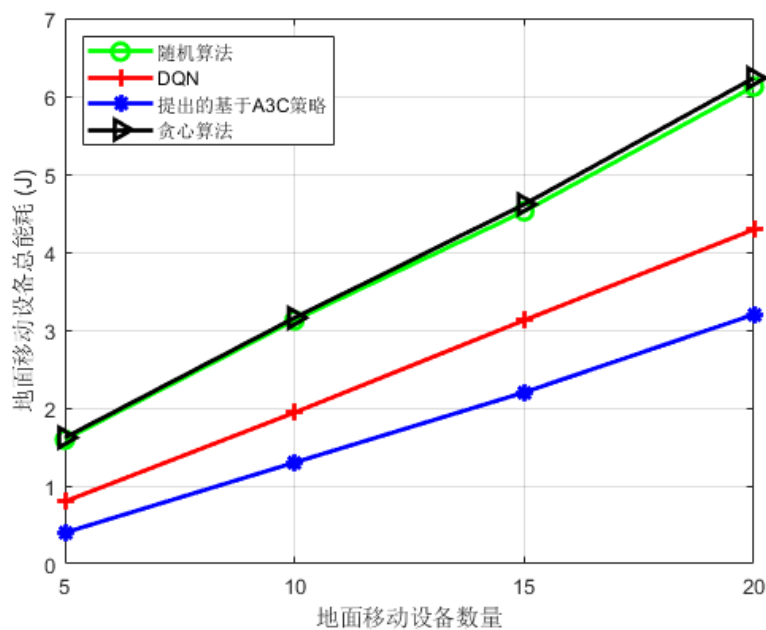


图 4-7 不同地面移动设备数量下不同卸载策略性能表现图

## 4.6 本章小结

本章研究了多无人机辅助移动边缘计算网络中的任务卸载问题，并构建了一个联合优化计算卸载决策和无人机位置的约束优化问题，其目标是 minimized 所有地面移动设备的总能耗。为解决此类约束优化问题，本章提出了一个具有异步特性的无模型深度强化学习方案，即基于 A3C 算法的任务卸载策略，能够有效地生成卸载决策。大量数值结果表明，本章所提出的基于 A3C 算法的任务卸载策略可以加快算法的收敛速度，有效降低地面移动设备的总能耗。

## 第五章 总结与展望

### 5.1 工作总结

本文首先梳理了移动边缘计算的研究背景，分析国内外研究现状。然后，介绍移动边缘计算、凸优化、强化学习等理论知识。最后，针对现有移动边缘计算中存在的过于依赖地面基站、无人机相关约束、地面设备移动性等挑战，展开如何在满足时延约束、无人机相关约束的前提下，设计一个有效的任务卸载策略，能够最小化所有地面用户设备总能耗的研究。本文主要针对上述问题，展开对无人机辅助移动边缘计算网络下任务卸载策略的研究，具体研究内容如下：

(1) 针对传统移动边缘计算过于依赖地面基站问题，本文提出了一种基于单无人机辅助移动边缘计算的任务卸载策略。首先，在单无人机辅助移动边缘计算网络中，考虑到计算时延、计算资源和无人机能量均有限的条件下，以最小化地面用户设备总能耗为目标，构建任务卸载决策与无人机位置联合优化问题。其次，考虑到所构建问题中卸载变量耦合，引入辅助变量将原问题转化为凸问题，并提出一种基于 ADMM 算法的任务卸载策略来求解优化问题。数值结果表明，所提出的基于 ADMM 算法的任务卸载策略能够有效降低地面用户设备总能耗。

(2) 由于第一部分研究的单无人机辅助移动边缘计算的应用场景有限，因此本文进一步研究了基于多无人机辅助移动边缘计算的任务卸载策略。本文将第一部分网络模型中的地面基站舍弃，同时引入地面设备的移动特性，在时延和无人机相关约束的限制下，将问题构建为任务卸载决策与无人机位置联合优化问题，实现地面移动设备总能耗最小化的目标。然后，本文提出一种基于 A3C 算法的任务卸载策略生成最优卸载决策。数值结果表明，本文提出的任务卸载策略比传统算法更具优势。

### 5.2 研究展望

虽然本文研究的任务卸载策略在无人机辅助移动边缘计算网络中对地面设备总能耗和无人机位置进行了联合优化研究且表现优越，但在研究过程中仍存在不足，可以在以下几方面开展下一步的研究工作：

(1) 本文主要研究在时延约束的条件下，地面设备总能耗的优化问题，但在物联网

应用日益繁杂的今天，需要考虑的是物联网设备和应用性能均衡的问题。在下一步的研究工作中，可以考虑时延和能耗的联合优化，实现性能更加均衡的系统模型。

（2）本文对无人机辅助移动边缘计算网络下任务卸载策略的研究，主要考虑的是二进制任务卸载方式，即地面设备的计算任务只有全部留和全部迁移这两种计算方式。但是随着数据流量的爆炸式增长，计算任务规模日益庞大，二进制卸载策略增加了网络带宽的压力以及带来较高的传输时延。因此，在未来的研究工作中，可以考虑动态部分卸载方式。

（3）本文研究内容主要基于无人机在固定高度上运行工作，在此基础上联合优化地面设备任务卸载变量和无人机飞行位置来优化地面设备总能耗，但是考虑到一些特殊的应用场景，无人机的灵活性优势将失去效用。因此，在后续的研究工作中，可以针对无人机的飞行轨迹进行联合优化。

## 参考文献

- [1] Kaloxylos A. A survey and an analysis of network slicing in 5G networks[J]. IEEE Communications Standards Magazine, 2018, 2(1): 60-65.
- [2] Li S, Da Xu L, Zhao S. 5G Internet of Things: A survey[J]. Journal of Industrial Information Integration, 2018, 10: 1-9.
- [3] Shafi M, Molisch A F, Smith P J, et al. 5G: A tutorial overview of standards, trials, challenges, deployment, and practice[J]. IEEE Journal on Selected Areas in Communications, 2017, 35(6): 1201-1221.
- [4] 陈海明, 崔莉, 谢开斌. 物联网体系结构与实现方法的比较研究[J]. 计算机学报, 2013, 36(1): 168-188.
- [5] Balaji S, Nathani K, Santhakumar R. IoT technology, applications and challenges: A contemporary survey[J]. Wireless Personal Communications, 2019, 108(1): 363-388.
- [6] Chen M, Yang J, Zhu X, et al. Smart home 2.0: Innovative smart home system powered by botanical IoT and emotion detection[J]. Mobile Networks and Applications, 2017, 22(6): 1159-1169.
- [7] Cheng N, Lyu F, Chen J, et al. Big data driven vehicular networks[J]. IEEE Network, 2018, 32(6): 160-167.
- [8] Qiu T, Chi J, Zhou X, et al. Edge computing in industrial internet of things: Architecture, advances and challenges[J]. IEEE Communications Surveys & Tutorials, 2020, 22(4): 2462-2488.
- [9] Mao Y, Zhang J, Letaief K B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices[J]. IEEE Journal on Selected Areas in Communications, 2016, 34(12): 3590-3605.
- [10] Lyu F, Zhu H, Zhou H, et al. SS-MAC: A novel time slot-sharing MAC for safety messages broadcasting in VANETs[J]. IEEE Transactions on Vehicular Technology, 2017, 67(4): 3586-3597.
- [11] 周悦芝, 张迪. 近端云计算: 后云计算时代的机遇与挑战[J]. 计算机学报, 2019, 42(04): 677-700.



- [12]李滕. 2022年云计算的发展趋势[J]. 计算机与网络, 2021, 47(24): 41.
- [13]Sadiku M N O, Musa S M, Momoh O D. Cloud computing: Opportunities and challenges[J]. IEEE Potentials, 2014, 33(1): 34-36.
- [14]Shu W, Cai K, Xiong N N. Research on strong agile response task scheduling optimization enhancement with optimal resource usage in green cloud computing[J]. Future Generation Computer Systems, 2021, 124: 12-20.
- [15]Barbarossa S, Sardellitti S, Di Lorenzo P. Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks[J]. IEEE Signal Processing Magazine, 2014, 31(6): 45-55.
- [16]施巍松, 张星洲, 王一帆, 等. 边缘计算: 现状与展望[J]. 计算机研究与发展, 2019, 56(1): 69.
- [17]Hu Y C, Patel M, Sabella D, et al. Mobile edge computing—A key technology towards 5G[J]. ETSI White Paper, 2015, 11(11): 1-16.
- [18]PremSankar G, Di Francesco M, Taleb T. Edge computing for the Internet of Things: A case study[J]. IEEE Internet of Things Journal, 2018, 5(2): 1275-1284.
- [19]Tong Z, Deng X, Ye F, et al. Adaptive computation offloading and resource allocation strategy in a mobile edge computing environment[J]. Information Sciences, 2020, 537: 116-131.
- [20]Kao Y H, Krishnamachari B, Ra M R, et al. Hermes: Latency optimal task assignment for resource-constrained mobile computing[J]. IEEE Transactions on Mobile Computing, 2017, 16(11): 3056-3069.
- [21]You C, Huang K, Chae H, et al. Energy-efficient resource allocation for mobile-edge computation offloading[J]. IEEE Transactions on Wireless Communications, 2016, 16(3): 1397-1411.
- [22]Yang X, Chen P, Gao S, et al. CSI-based low-duty-cycle wireless multimedia sensor network for security monitoring[J]. Electronics Letters, 2018, 54(5): 323-324.
- [23]Li B, Fei Z, Zhang Y. UAV communications for 5G and beyond: Recent advances and future trends[J]. IEEE Internet of Things Journal, 2018, 6(2): 2241-2263.

- [24] Hayat S, Yanmaz E, Muzaffar R. Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint[J]. IEEE Communications Surveys & Tutorials, 2016, 18(4): 2624-2661.
- [25] Du Y, Wang K, Yang K, et al. Energy-efficient resource allocation in UAV based MEC system for IoT devices[C]//2018 IEEE Global Communications Conference (GLOBECOM). IEEE, 2018: 1-6.
- [26] Liu T, Cui M, Zhang G, et al. 3D trajectory and transmit power optimization for UAV-enabled multi-link relaying systems[J]. IEEE Transactions on Green Communications and Networking, 2020, 5(1): 392-405.
- [27] Shakhathreh H, Sawalmeh A H, Al-Fuqaha A, et al. Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges[J]. IEEE Access, 2019, 7: 48572-48634.
- [28] 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, 2017, 67(3): 2049-2063.
- [29] Zhao N, Pang X, Li Z, et al. Joint trajectory and precoding optimization for UAV-assisted NOMA networks[J]. IEEE Transactions on Communications, 2019, 67(5): 3723-3735.
- [30] Cao X, Xu J, Zhang R. Mobile edge computing for cellular-connected UAV: Computation offloading and trajectory optimization[C]//2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC). IEEE, 2018: 1-5.
- [31] Dinh T Q, Tang J, La Q D, et al. Offloading in mobile edge computing: Task allocation and computational frequency scaling[J]. IEEE Transactions on Communications, 2017, 65(8): 3571-3584.
- [32] Dinh T Q, La Q D, Quek T Q S, et al. Learning for computation offloading in mobile edge computing[J]. IEEE Transactions on Communications, 2018, 66(12): 6353-6367.
- [33] Yu S, Wang X, Langar R. Computation offloading for mobile edge computing: A deep learning approach[C]//2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC). IEEE, 2017: 1-6.
- [34] Yang L, Zhang H, Li M, et al. Mobile edge computing empowered energy efficient task offloading in 5G[J]. IEEE Transactions on Vehicular Technology, 2018, 67(7): 6398-6409.

- [35]Chen L, Li X, Ji H, et al. Computation offloading balance in small cell networks with mobile edge computing[J]. Wireless Networks, 2019, 25(7): 4133-4145.
- [36]Ren J, Yu G, Cai Y, et al. Latency optimization for resource allocation in mobile-edge computation offloading[J]. IEEE Transactions on Wireless Communications, 2018, 17(8): 5506-5519.
- [37]Qian L, Wu Y, Ouyang J, et al. Latency optimization for cellular assisted mobile edge computing via non-orthogonal multiple access[J]. IEEE Transactions on Vehicular Technology, 2020, 69(5): 5494-5507.
- [38]Liu Y, Yu H, Xie S, et al. Deep reinforcement learning for offloading and resource allocation in vehicle edge computing and networks[J]. IEEE Transactions on Vehicular Technology, 2019, 68(11): 11158-11168.
- [39]Kiran N, Pan C, Wang S, et al. Joint resource allocation and computation offloading in mobile edge computing for SDN based wireless networks[J]. Journal of Communications and Networks, 2019, 22(1): 1-11.
- [40]Guo H, Liu J. UAV-enhanced intelligent offloading for Internet of Things at the edge[J]. IEEE Transactions on Industrial Informatics, 2019, 16(4): 2737-2746.
- [41]Liu Q, Shi L, Sun L, et al. Path planning for UAV-mounted mobile edge computing with deep reinforcement learning[J]. IEEE Transactions on Vehicular Technology, 2020, 69(5): 5723-5728.
- [42]Guo F, Zhang H, Ji H, et al. Joint trajectory and computation offloading optimization for UAV-assisted MEC with NOMA[C]//2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, 2019: 1-6.
- [43]Wang Y, Ru Z Y, Wang K, et al. Joint deployment and task scheduling optimization for large-scale mobile users in multi-UAV-enabled mobile edge computing[J]. IEEE Transactions on Cybernetics, 2019, 50(9): 3984-3997.
- [44]Shang B, Liu L. Mobile-edge computing in the sky: Energy optimization for air-ground integrated networks[J]. IEEE Internet of Things Journal, 2020, 7(8): 7443-7456.
- [45]Papagiannakis G, Singh G, Magnenat - Thalmann N. A survey of mobile and wireless technologies for augmented reality systems[J]. Computer Animation and Virtual Worlds, 2008, 19(1): 3-22.

- [46]Mach P, Becvar Z. Mobile edge computing: A survey on architecture and computation offloading[J]. IEEE Communications Surveys & Tutorials, 2017, 19(3): 1628-1656.
- [47]Jin X, Zhang J, Sun X, et al. Computation offloading and resource allocation for MEC in C-RAN: A deep reinforcement learning approach[C]//2019 IEEE 19th International Conference on Communication Technology (ICCT). IEEE, 2019: 902-907.
- [48]Chi C Y, Li W C, Lin C H. Convex optimization for signal processing and communications: from fundamentals to applications[M]. CRC Press, 2017.
- [49]Boyd S, Boyd S P, Vandenberghe L. Convex optimization[M]. Cambridge University Press, 2004.
- [50]Boyd S, Parikh N, Chu E, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers[J]. Foundations and Trends® in Machine learning, 2011, 3(1): 1-122.
- [51]Bai X, Scheinberg K. Alternating direction methods for non convex optimization with applications to second-order least-squares and risk parity portfolio selection[J]. Optimization-Online, 2015.
- [52]Mousavi S S, Schukat M, Howley E. Deep reinforcement learning: An overview[C]// Proceedings of SAI Intelligent Systems Conference. Springer, Cham, 2016: 426-440.
- [53]Luong N C, Hoang D T, Gong S, et al. Applications of deep reinforcement learning in communications and networking: A survey[J]. IEEE Communications Surveys & Tutorials, 2019, 21(4): 3133-3174.
- [54]Mao Q, Hu F, Hao Q. Deep learning for intelligent wireless networks: A comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2018, 20(4): 2595-2621.
- [55]赵星宇, 丁世飞. 深度强化学习研究综述[J]. 计算机科学, 2018, 45(07): 1-6.
- [56]Lorenzo B, Garcia-Rois J, Li X, et al. A robust dynamic edge network architecture for the internet of things[J]. IEEE Network, 2018, 32(1): 8-15.
- [57]Zhao L, Zhao W, Hawbani A, et al. Novel online sequential learning-based adaptive routing for edge software-defined vehicular networks[J]. IEEE Transactions on Wireless Communications, 2020, 20(5): 2991-3004.
- [58]Dogra A, Jha R K, Jain S. A survey on beyond 5G network with the advent of 6G: Architecture and emerging technologies[J]. IEEE Access, 2020, 9: 67512-67547.

- [59] Qiao G, Leng S, Zhang K, et al. Collaborative task offloading in vehicular edge multi-access networks[J]. IEEE Communications Magazine, 2018, 56(8): 48-54.
- [60] Li Z, Zhu Q. An offloading strategy for multi-user energy consumption optimization in multi-MEC scene[J]. KSII Transactions on Internet and Information Systems (TIIS), 2020, 14(10): 4025-4041.
- [61] Ren J, Yu G, Cai Y, et al. Latency optimization for resource allocation in mobile-edge computation offloading[J]. IEEE Transactions on Wireless Communications, 2018, 17(8): 5506-5519.
- [62] Cheny H, Zhao D, Cheny Q, et al. Joint computation offloading and radio resource allocations in wireless cellular networks[C]//2018 10th International Conference on Wireless Communications and Signal Processing (WCSP). IEEE, 2018: 1-6.
- [63] Pardalos P M, Vavasis S A. Quadratic programming with one negative eigenvalue is NP-hard[J]. Journal of Global Optimization, 1991, 1(1): 15-22.
- [64] Kuutti S, Fallah S, Katsaros K, et al. A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications[J]. IEEE Internet of Things Journal, 2018, 5(2): 829-846.
- [65] Zhan W, Luo C, Wang J, et al. Deep-reinforcement-learning-based offloading scheduling for vehicular edge computing[J]. IEEE Internet of Things Journal, 2020, 7(6): 5449-5465.
- [66] Chen X, Zhang H, Wu C, et al. Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning[J]. IEEE Internet of Things Journal, 2018, 6(3): 4005-4018.
- [67] Yang B, Cao X, Bassey J, et al. Computation offloading in multi-access edge computing: A multi-task learning approach[J]. IEEE Transactions on Mobile Computing, 2020, 20(9): 2745-2762.
- [68] Cheng N, Lyu F, Quan W, et al. Space/aerial-assisted computing offloading for IoT applications: A learning-based approach[J]. IEEE Journal on Selected Areas in Communications, 2019, 37(5): 1117-1129.
- [69] Chen Z, Wang X. Decentralized computation offloading for multi-user mobile edge computing: A deep reinforcement learning approach[J]. EURASIP Journal on Wireless Communications and Networking, 2020, 2020(1): 1-21.

## 致谢

落笔于此，意味着我三年的研究生生涯即将结束，也为在母校的人生最美好的七年画上了圆满的句号。三年时间太长，经历了太多的事情，我结识了许多志同道合的朋友，有幸得到老师们的谆谆教诲，同时也受到家人们的热切关怀。三年时间太短，转瞬间就要和同学、老师们分别，我们就又要踏上新的征程。

在此，我首先要特别感谢我的两位研究生指导老师，苏健老师和李斌老师。在三年的硕士学习和生活中，初入课题组的我对如何更有效率地检索和阅读论文知之甚少，苏老师对我遇到的疑惑给予了悉心的指导，提出许多建设性的意见。生活中，苏老师对我的个人成长提出了宝贵的意见。在我的科研道路上，李老师给我提供许多珍贵的经验。在我撰写小论文时，李老师为我的论文进行了详细的批注，教导我如何写出规范严谨的论文。正是在苏老师和李老师的帮助和教导下，我才可以顺利完成学业要求，不断进步。能够遇到这样负责的老师，是我的荣幸。谨此向苏老师和李老师致以真挚的感谢和崇高的敬意！

同时，我也要感谢辅导员和任课老师们。辅导员为我们处理日常事务，让我们可以心无旁骛地学习。任课老师们为我们拓展知识面，让我们了解更多研究领域的知识。我还要感谢我的同学、室友和师兄弟们，是他们让我的研究生生活过的丰富多彩且充实。

另外，我要感谢我的父母，是他们无条件的支持，让我安心完成学业。他们无私的爱是我在人生道路上砥砺前行的无限动力。

最后，由衷感谢各位专家和学者百忙之中来审阅和指导本篇论文。

虽然暂别求学的旅程，但是求知的道路永无止境。雄关漫道真如铁，而今迈步从头越，与君共勉。

## 作者简介

### 基本情况:

余世明, 男, 1996 年 7 月生, 江苏扬州人。2015 年 9 月至 2019 年 6 月, 就读于南京信息工程大学滨江学院软件工程专业, 获得学士学位。2019 年 9 月至今, 就读于南京信息工程大学计算机学院、软件学院、网络空间安全学院计算机科学与技术专业, 攻读硕士学位, 导师为苏健, 主要研究方向为移动边缘计算。研究生在读期间, 学习课程 15 门, 总学分 26 分, 包括学位课程学分 16 分。

### 发表学术论文:

- [1] Su J, **Yu S**, Li B, et al. Distributed and collective intelligence for computation offloading in aerial edge networks[J]. IEEE Transactions on Intelligent Transportation Systems. (学生二作, Online)
- [2] Li B, **Yu S**, Su J, et al. Computation offloading in multi-UAV-enhanced mobile edge networks: A deep reinforcement learning approach[J]. Wireless Communications and Mobile Computing. (学生二作, Online)

### 申请的软件著作权:

- [1] 汽车4S店管理系统V1.0, 登记号: 2020SR0666083.