

面向空地协同移动边缘计算的服务布置策略

屈毓铨^{1),2),3)} 秦 蓁⁴⁾ 马靖豪⁵⁾ 戴海鹏⁵⁾ 董 超^{1),2)}
王 海⁴⁾ 吴 帆³⁾ 陈贵海³⁾

¹⁾(南京航空航天大学电磁频谱空间认知动态系统工信部重点实验室 南京 211106)

²⁾(南京航空航天大学电子信息工程学院 南京 211106)

³⁾(上海交通大学计算机科学与工程系 上海 200240)

⁴⁾(中国人民解放军陆军工程大学通信工程学院 南京 210001)

⁵⁾(南京大学计算机软件新技术国家重点实验室 南京 210023)

摘 要 无人机辅助的移动边缘计算被认为是在下一代移动通信网络中能高效灵活处理时延敏感的计算密集型任务的潜力技术之一. 本文研究了基于无人机的空地协同移动边缘计算的服务布置问题, 具体而言, 如何在满足任务时延需求和其他资源约束的情况下, 通过联合优化无人机和地面基站的服务布置、无人机航迹、任务卸载和计算资源分配, 以最小化所有用户的总能耗. 由于问题的非凸和各种变量的复杂耦合, 该问题属于一个非凸混合整数非线性规划问题, 较难求解. 本文针对多无人机和多地面基站协同提供计算服务的场景, 提出了一种基于交替优化的服务布置算法. 该算法通过迭代求解三个不同的子问题来获得具有收敛性保证的次优解决方案. 首先, 采用分支定界法求解联合服务布置和任务卸载的子问题. 其次, 采用连续凸逼近方法求解无人机航迹优化的子问题. 然后, 利用计算资源分配子问题的特性得到该子问题的闭式最优解. 最后, 对上述过程重复迭代, 得到问题的一个次优解. 仿真结果表明, 相比随机布置策略、贪心布置策略、本地计算策略, 所提布置策略能够大大减少用户总能耗.

关键词 移动边缘计算; 空地协同; 无人机; 服务布置; 交替优化

中图分类号 TN925 DOI号 10.11897/SP.J.1016.2022.00781

Service Provisioning for Air-Ground Collaborative Mobile Edge Computing

QU Yu-Ben^{1),2),3)} QIN Zhen⁴⁾ MA Jing-Hao⁵⁾ DAI Hai-Peng⁵⁾ DONG Chao^{1),2)}
WANG Hai⁴⁾ WU Fan³⁾ CHEN Gui-Hai³⁾

¹⁾(The Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space, Ministry of Industry and Information Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106)

²⁾(College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106)

³⁾(Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240)

⁴⁾(College of Communications Engineering, Army Engineering University of PLA, Nanjing 210001)

⁵⁾(State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023)

Abstract Mobile edge computing (MEC) emerges as a promising technique to directly provide

收稿日期: 2021-02-28; 在线发布日期: 2021-09-18. 本课题得到国家重点研发计划(2018YFB1800801)、国家自然科学基金(61931011、62072303、61872178、62171465)、国家博士后创新人才支持计划(BX20190202)、中央高校基本科研业务费(14380059)资助. 屈毓铨, 博士, 副研究员, 主要研究领域为边缘计算、空地一体化网络. E-mail: quyuben@nuaa.edu.cn. 秦 蓁(通信作者), 博士研究生, 主要研究领域为无人机通信和边缘计算. E-mail: qzqzla912@163.com. 马靖豪, 硕士研究生, 主要研究领域为边缘计算、视频分析. 戴海鹏, 博士, 副教授, 主要研究领域为物联网、移动计算和数据挖掘. 董 超, 博士, 教授, 主要研究领域为无人机组网、边缘计算和抗干扰网络协议. 王 海, 博士, 教授, 主要研究领域为无线网络、Ad Hoc网络和车载通信技术. 吴 帆, 博士, 教授, 国家杰出青年科学基金入选者, 主要研究领域为无线网络与移动计算、博弈论算法及其应用和隐私保护. 陈贵海, 博士, 教授, 国家杰出青年科学基金入选者, 主要研究领域为传感器网络、高性能计算机体系结构和组学.

cloud computing services to the network edge, whose benefits mainly include low response latency and high computation power for resource-constrained end users. Nevertheless, the traditional infrastructure-based MEC might incur high deployment cost or cannot work in rural regions without terrestrial infrastructure and hot-spot areas with time-varying heavy traffic loads. To compensate of the aforementioned shortcomings, unmanned aerial vehicle (UAV)-assisted MEC has been generally recognized as a potential technology to efficiently and flexibly cope with latency-sensitive and computation-intensive tasks in next generation communication network. Most existing works about MEC consider how to provide computation service either only by terrestrial MEC servers deployed at local wireless access points (APs) as well as cellular base stations (BSs), or solely by aerial MEC servers at UAVs, while how to provide such computation services by both terrestrial and aerial MEC servers has been seldom studied so far. In contrast, we study the problem of service provisioning for air-ground collaborative MEC in this work. To be specific, we aim to minimize the total energy consumption of all terrestrial users with task latency and other resource constraints, by jointly optimizing service placement at all UAVs and BSs, computation task offloading, UAV movement trajectory, and computation resource allocation. Due to the non-convexity of the problem and complex coupling of mixed integral variables, this problem is naturally a non-convex mixed integer nonlinear programming (MINLP) problem that is extremely difficult to solve. In this paper, we focus on the scenario of multiple UAVs and multiple terrestrial BSs to collaboratively provide computing services to terrestrial networks, and propose a suboptimal algorithm with convergence guarantee to the aforementioned problem based on alternating optimization techniques. Specifically, we decompose the original problem into three different subproblems: the service placement and task offloading subproblem, the UAV movement trajectory subproblem, and the computation resource allocation subproblem, which are solved in an iterative manner. Firstly, we utilize the Branch and Bound (BnB) method to solve the first subproblem of joint service placement and task offloading, which can be formulated as a multi-dimensional multiple-choice knapsack problem (MMKP). Secondly, we solve the second UAV trajectory subproblem by existing successive convex approximation (SCA) techniques. Then, using the characteristics of the computation resource allocation subproblem, the closed-form optimal computation resource allocation of the last subproblem can be efficiently obtained. Finally, the above process is iterated repeatedly to obtain a suboptimal solution of the problem. The simulation results show that, the proposed placement strategy can greatly reduce the total energy consumption of all terrestrial users, compared with the random placement strategy, greedy placement strategy, and local computing strategy. Specifically, for the convergence, the proposed strategy can converge within about only 10 iterations. We also evaluate the performance of the proposed strategy under different UAVs' storage capacities, different users' computation workloads, different initial feasible solutions, and mobility of users. In addition, the results show that, with more aerial MEC servers or more terrestrial MEC servers, the performance of the proposed strategy is also better.

Keywords mobile edge computing; air-ground collaboration; unmanned aerial vehicle; service provisioning; alternating optimization

1 引 言

随着移动通信和物联网技术的普及与发展,各种移动应用对强算力和低延迟的需求与终端用户相

对有限的能量供应和计算能力之间存在着巨大的矛盾. 移动边缘计算(mobile edge computing, MEC)^[1]已被广泛认为是一种能部署在网络边缘提供云计算服务的潜力技术,它可以突破终端用户的能力限制,以满足上述移动应用的需求. 移动边缘计算的核心

思想是在蜂窝基站或者本地无线接入点部署小型边缘计算服务器,然后将用户的计算密集型任务卸载到边缘计算服务器上执行.近年来,移动边缘计算受到了学术界和工业界的极大关注.例如,多个国际标准组织和行业协会(如欧洲电信标准化协会和5G汽车联盟)已经明确了许多应用实例,包括应用感知的性能优化和自动驾驶等^[1-2].

然而,上述部署在蜂窝无线网络上的移动边缘计算是依赖于地面已有固定基础通信设施,其部署的成本可能较高,且在具有时变负载的热点地区和没有足够地面基础设施的边远地区中无法工作^[3-4].为了弥补基于地面固定基础设施的移动边缘计算的不足,最近提出了无人机(unmanned aerial vehicle, UAV)辅助的移动边缘计算,其通过在无人机上搭载边缘计算服务器来进一步提高地面用户的计算性能^[3-5].具体而言,无人机因其高机动性、易于部署、视距链路连接以及不断增加的机载负荷和长距离飞行能力等优点,可以灵活地为分散的地面移动用户提供计算服务.自从文献[6]提出第一个无人机辅助的移动边缘计算结构以来,人们对无人机辅助的移动边缘计算进行了深入研究.

对于无人机辅助的移动边缘计算研究,大致可以分为两类:一类仅包括无人机边缘计算服务器^[7-11],另一类同时包括无人机和地面基站的边缘计算服务器^[12-14].对于前一类,文献[7]介绍了三种无人机辅助的移动边缘计算体系结构,它们通过将无人机集成到移动边缘计算网络中来提高计算性能和减少延迟.文献[8]提出了一种新型的空地一体化移动边缘网络,该网络可以灵活地部署和调度无人机,并辅助边缘网络进行计算、缓存和通信.文献[9]研究了一个无人机辅助的移动云计算系统,该系统赋予无人机计算能力,为本地计算能力有限的移动用户提供计算卸载机会.该系统旨在满足移动应用的服务质量要求的同时最小化无人机移动总能耗.文献[10]研究了一个无人机辅助无线供电的移动边缘计算系统的资源分配问题.其中,无人机发射能量信号给多个移动用户进行充电,并为他们提供计算服务.文献[11]通过联合优化多无人机移动边缘计算网络中的用户关联、功率控制、计算资源分配和位置规划解决了功率和最小化问题.对于后一类,文献[12]提出了一个空地一体化移动边缘计算框架,其中地面基站、车辆和无人机作为边缘计算服务器提供有效的计算服务.文献[13]研究了一种异构的移动边缘计算架构,旨在利用基于人工智能的方

法解决异构移动边缘计算架构在动态环境中的关键挑战.文献[14]研究了一种无人机辅助的移动边缘计算体系结构,其中无人机可以作为计算服务器来处理用户的计算任务,或者作为中继进一步将它们的任务卸载到本地无线接入点.

虽然理论上可以利用网络功能虚拟化(network functions virtualization, NFV)技术将移动边缘计算服务部署在任意边缘计算服务器上,但移动边缘计算中需要解决的一个基本问题是服务布置,即在何处预先存储/布置哪种计算服务,以满足用户的各种服务需求^[15-23].现有的大多数流行移动应用都是数据密集型的,并且总是需要预先在边缘计算服务器上缓存大量相关数据,例如与服务相关的数据库、代码和经过训练的机器学习模型等.因此,存储容量相对有限的边缘计算服务器往往只能存储部分服务^[24-25].由于无人机资源相对匮乏,如何为无人机辅助的移动边缘计算尤其是无人机和地面基站协同的移动边缘计算系统合理地布置服务显得十分重要,并且更具挑战性.前面关于无人机辅助的移动边缘计算工作几乎没有考虑服务布置的问题^[7-14],而现有关于服务布置的研究是仅针对于地面移动边缘计算系统^[15-23].具体而言,根据优化目标的不同,针对地面移动边缘计算系统的服务布置的研究可分为以下三类.1)最大化成功服务的用户数量:文献[15]、[16]和[17]研究如何通过联合优化地面边缘计算服务器中的服务布置和用户请求调度,以实现服务的用户数量的最大化.2)最小化服务器和用户的总成本:文献[18]、[19]和[20]针对地面移动边缘计算系统中虚拟现实等特定应用,研究如何最小化服务放置和用户请求调度的总成本.3)最小化用户经历的总时延:文献[21]、[22]和[23]研究如何通过优化地面移动边缘计算服务器中的动态服务布置和任务卸载,以最小化移动用户的总时延.由于无人机航迹规划的必要性,这些研究^[15-23]不能应用于无人机辅助的移动边缘计算网络.综上所述,无人机辅助的移动边缘计算的服务布置问题值得深入研究.

本文针对多个无人机和多个地面基站协同提供边缘计算服务的场景,首次研究了空地协同移动边缘计算的服务布置问题.具体而言,本文研究了在 N 架无人机和 B 个地面基站上部署 S 种计算服务时,如何确定一种服务布置策略,以最小化 U 个用户的总能耗.该问题需要联合优化服务布置、无人机航迹、任务卸载和计算资源分配四种变量.由于服务布置和任务卸载变量属于整型变量、无人机航

迹和计算资源分配变量属于连续型变量,且问题在目标函数和部分约束的非凸性,该问题属于一个非凸混合整数非线性规划(mixed integer nonlinear programming, MINLP)问题,较难求解.

为求解该问题,本文提出了一种基于交替优化的算法,该算法通过迭代求解三个不同的子问题来获得具有收敛性保证的次优解决方案.首先,采用分支定界(Branch and Bound, BnB)法求解联合服务布置和任务卸载的子问题.其次,采用连续凸逼近(successive convex approximation, SCA)方法求解无人机航迹优化的子问题.然后,利用计算资源分配子问题的特性,得到该子问题的闭式最优解.最后,对上述过程重复迭代,得到问题的一个次优解.仿真结果表明,相比随机布置策略、贪心布置策略、本地计算策略,所提基于交替优化的服务布置策略能够大大减少用户总能耗,且两架无人机和单个基站协同的情况下能比单架无人机和单基站协同的情况下进一步减少约20%的用户总能耗.

本文在第2节将介绍系统模型和所研究的问题的数学描述;在第3节介绍所提出的基于交替优化的服务布置算法;在第4节通过仿真验证所提算法的有效性;在第5节进行了讨论;最后,第6节是全文的总结.

2 系统模型及问题描述

2.1 系统模型

如图1所示,本文考虑一个空地协同移动边缘计算网络,该网络由 N 架无人机、 B 个地面基站和 U 个地面用户组成,且假设所有设备均配有一个天线.无人机、基站和用户集合分别可以由 $\mathcal{N}=\{1, 2, \dots, N\}$, $\mathcal{B}=\{1, 2, \dots, B\}$ 和 $\mathcal{U}=\{1, 2, \dots, U\}$ 表示.根据现有研究,如文献[14],假设无人机和用户均具有由嵌入式电池供电的机载通信模块和计算处理器,而基站配备了更强大的边缘计算服务器.这意味着无人机和基站可以协同为用户提供计算服务,以提高计算性能、降低用户能耗.假定在一个固定的服务周期内, N 架无人机可以从初始位置飞行到各自指定的最终位置.此外,所有用户都是移动的,且具有已知的运动轨迹.根据文献[26]和[27],整个服务周期可以被划分为具有相同长度 τ^{max} 的 T 个时隙,其中时隙集合由 $\mathcal{T}=\{1, 2, \dots, T\}$ 表示.假设每个用户 $u \in \mathcal{U}$ 在每个时隙 $t \in \mathcal{T}$ 中都产生一个计算密集型任务.表1列举了本文使用的主要符号.

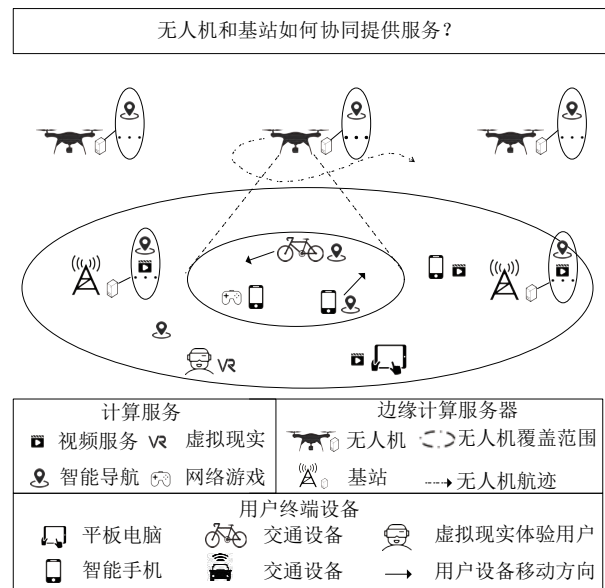


图1 系统模型示意图

表1 变量说明

变量符号	变量含义
$\mathcal{N}=\{1, 2, \dots, N\}$	无人机集合
$\mathcal{B}=\{1, 2, \dots, B\}$	基站集合
$\mathcal{U}=\{1, 2, \dots, U\}$	用户集合
$\mathcal{T}=\{1, 2, \dots, T\}$	时隙集合
$\mathcal{S}=\{1, 2, \dots, S\}$	服务集合
τ^{max}	单位时隙长度
l_s	服务所需的存储容量
$\hat{L}_n^{max}, \bar{L}_b^{max}$	无人机和基站的存储容量
$\hat{x}_n^s, \bar{x}_b^s \in \{0, 1\}$	服务布置变量
H_n	无人机的飞行高度
$G_n^t=(X_n^t, Y_n^t)$	无人机的二维坐标
$\hat{G}_u^t=(\hat{X}_u^t, \hat{Y}_u^t)$	用户的坐标
$\hat{G}_b^t=(\hat{X}_b^t, \hat{Y}_b^t)$	基站的坐标
R_n^{max}	无人机最大水平覆盖半径
v_n	无人机恒定飞行速度
$\hat{z}_{mu}^t, \hat{z}_{bu}^t \in \{0, 1\}$	任务卸载变量
s_u^t	任务所需的服务类型
a_u^t	任务输入数据大小
w_u^t	任务所需的CPU周期数
C_n^{max}	无人机最大计算能力
\bar{C}_u^{max}	用户最大计算能力
\bar{C}_b^{max}	基站最大计算能力
R_{nu}^t, R_{bu}^t	用户与服务器的水平距离
r_{nu}^t, r_{bu}^t	用户与服务器的传输速率
B_n, B_b	无人机和基站分配的带宽
p_u^{tx}	用户的传输功率
$\tilde{c}_u^t, \tilde{c}_{mu}^t, \tilde{c}_{bu}^t$	计算资源分配变量
$L_u^{loc}(t), E_u^{loc}(t)$	本地计算时延和能耗
$L_{nu}^{com}(t), L_{bu}^{com}(t); E_{nu}^{com}(t), E_{bu}^{com}(t)$	卸载通信时延和能耗
$L_{nu}^{cop}(t), L_{bu}^{cop}(t)$	卸载计算时延
$L_{nu}^{off}(t), L_{bu}^{off}(t)$	卸载的总时延
$L_u(t)$	用户在时隙 $t \in \mathcal{T}$ 中计算任务的总时延
$E_u(t)$	用户在时隙 $t \in \mathcal{T}$ 中计算任务的总能耗

2.1.1 协同服务布置和无人机运动模型

为了给不同的用户提供个性化的计算服务,无人机和基站中的边缘计算服务器都应该预先存储相应的数据,包括与服务相关的数据库和经过训练的机器学习模型等.假设此空地协同移动边缘计算网络向用户提供的服务类型不超过 S 种,且服务集合由 $\mathcal{S}:=\{1,2,\dots,S\}$ 表示.进一步地,假设 $l_s>0$, $\hat{L}_n^{\max}>0$ 和 $\tilde{L}_b^{\max}>0$ 分别表示服务 $s\in\mathcal{S}$ 所需的存储容量、无人机 $n\in\mathcal{N}$ 的存储容量和基站 $b\in\mathcal{B}$ 的存储容量.为了对服务布置决策变量进行建模,假设 $\hat{x}_n^s\in\{0,1\}$ 和 $\tilde{x}_b^s\in\{0,1\}$ 分别指示服务类型 $s\in\mathcal{S}$ 是否部署在无人机和基站上.例如,当服务 $s\in\mathcal{S}$ 部署在无人机 $n\in\mathcal{N}$ 上时, \hat{x}_n^s 为1,否则为0.由于无人机和基站的存储空间有限,可以得到相应的存储约束分别为 $\sum_{s=1}^S l_s \hat{x}_n^s \leq \hat{L}_n^{\max}$ 和 $\sum_{s=1}^S l_s \tilde{x}_b^s \leq \tilde{L}_b^{\max}$.

与地面移动边缘计算网络提供计算服务的工作不同^[15-23],空地协同移动边缘计算网络在设计中需要考虑无人机的航迹.与现有研究^[11-26]保持一致,本文采用三维笛卡尔坐标系.假设 (X_n^t, Y_n^t, H_n) 为无人机 $n\in\mathcal{N}$ 在时隙 $t\in\mathcal{T}$ 时的坐标,且在执行任务期间,无人机 $n\in\mathcal{N}$ 以固定高度 H_n 飞行,起始坐标为 (X_n^0, Y_n^0, H_n) .无人机 $n\in\mathcal{N}$ 的最大水平覆盖半径用 R_n^{\max} 表示,这是由天线的发射角和无人机的飞行高度决定的.除此之外,假设 $d_n^{\max} = v_n \tau^{\max}$ 表示无人机在每个时隙中的最大飞行距离,其中 v_n 表示无人机 $n\in\mathcal{N}$ 的恒定飞行速度^[28].

2.1.2 时延和能耗模型

根据现有的研究^[11-26],本文采用二进制计算卸载策略.具体来说,二进制变量 $\hat{z}_{mu}^t\in\{0,1\}$ 和 $\tilde{z}_{bu}^t\in\{0,1\}$ 分别表示用户 $u\in\mathcal{U}$ 在时隙 $t\in\mathcal{T}$ 时的计算任务被卸载到无人机 $n\in\mathcal{N}$ ($\hat{z}_{mu}^t=1$ 和 $\tilde{z}_{bu}^t=0$)、地面基站 $b\in\mathcal{B}$ ($\hat{z}_{mu}^t=0$ 和 $\tilde{z}_{bu}^t=1$)或在本地执行($\hat{z}_{mu}^t=0$ 和 $\tilde{z}_{bu}^t=0$).为了便于研究,本文采用三元组 (s_u^t, a_u^t, w_u^t) 表示时隙 $t\in\mathcal{T}$ 时用户 $u\in\mathcal{U}$ 的第 t 个计算任务,其中 $s_u^t\in\mathcal{S}$ 表示任务所需的服务类型,即用户请求服务.变量 a_u^t 表示卸载时需要传输的输入数据大小.变量 w_u^t 表示任务所需的CPU周期数,即用户计算负载.假设每个时隙产生的计算任务都必须当前时隙内完成,且在不同时隙时用户所需的服务类型可能是不同的.例如,用户移动到不同地方时可能需要不同的服务.如图1所示,用户请求的服务可能是视频服务、智能导航、网络游戏或虚拟现实等.类似于文献^[11]和^[26],假设可以通过程序探

测器有效获取 a_u^t 和 w_u^t 的值.令 $\tilde{C}_u^{\max}>0$ 表示用户 $u\in\mathcal{U}$ 所拥有的最大计算能力.如果用户 $u\in\mathcal{U}$ 在时隙 $t\in\mathcal{T}$ 中的计算任务在本地执行,其时延为

$$L_u^{\text{loc}}(t) = \frac{w_u^t}{\tilde{c}_u^t}, \forall u \in \mathcal{U}, t \in \mathcal{T} \quad (1)$$

其中 $\tilde{c}_u^t \leq \tilde{C}_u^{\max}$ 是用户 $u\in\mathcal{U}$ 在时隙 $t\in\mathcal{T}$ 时的计算能力.用户本地计算消耗的能量为

$$E_u^{\text{loc}}(t) = \kappa_u (\tilde{c}_u^t)^{\nu_u} \frac{w_u^t}{\tilde{c}_u^t} = \kappa_u (\tilde{c}_u^t)^{\nu_u-1} w_u^t, \forall u \in \mathcal{U}, t \in \mathcal{T} \quad (2)$$

其中 $\kappa_u \geq 0$ 和 ν_u 分别表示有效开关电容和值为3的常数^[26].

如果用户将计算任务卸载到无人机或基站,则总时延主要包括三个部分:卸载计算任务所需的通信时延,在无人机或基站的边缘计算服务器中执行计算任务的时延,以及返回计算结果的时间.其中,第三部分的时延通常被省略^[26].接下来,以用户 $u\in\mathcal{U}$ 将任务卸载到无人机 $n\in\mathcal{N}$ 为例,介绍如何计算总时延.首先,需要计算用户 $u\in\mathcal{U}$ 和无人机 $n\in\mathcal{N}$ 之间的信道容量.根据欧几里德公式,用户 $u\in\mathcal{U}$ 与无人机 $n\in\mathcal{N}$ 之间的水平距离为

$$R_{nu}^t = \sqrt{(X_n^t - \tilde{X}_u^t)^2 + (Y_n^t - \tilde{Y}_u^t)^2} \quad (3)$$

其中 $\tilde{G}_u^t := (\tilde{X}_u^t, \tilde{Y}_u^t)$ 是用户 $u\in\mathcal{U}$ 在时隙 $t\in\mathcal{T}$ 时的二维坐标.在时隙 $t\in\mathcal{T}$ 中,如果用户 $u\in\mathcal{U}$ 将任务卸载到无人机 $n\in\mathcal{N}$ 上,则该用户必须位于无人机的覆盖范围内,即, $R_{nu}^t \leq R_n^{\max}$.此时,上行链路传输速率为

$$r_{nu}^t = B_n \log_2 \left(1 + \frac{\delta p_u^{\text{tr}}}{(R_{nu}^t)^2 + H_n^2} \right) \quad (4)$$

其中, B_n 是分配给无人机 $n\in\mathcal{N}$ 的频谱带宽, $\delta = G_0 g_0 / \sigma^2$ ($G_0 \approx 2.2846^{\text{[11-26]}}$, g_0 是单位信道功率增益, σ^2 是噪声功率), p_u^{tr} 是用户 $u\in\mathcal{U}$ 的传输功率.因此,用户 $u\in\mathcal{U}$ 将任务上传到无人机 $n\in\mathcal{N}$ 的通信时延可以表示为

$$L_{nu}^{\text{com}}(t) = \frac{a_u^t}{r_{nu}^t} = \frac{a_u^t}{B_n \log_2 \left(1 + \frac{\delta p_u^{\text{tr}}}{(R_{nu}^t)^2 + H_n^2} \right)} \quad (5)$$

此时,用户 $u\in\mathcal{U}$ 消耗的通信能量为

$$\begin{aligned} E_{nu}^{\text{com}}(t) &= p_u^{\text{tr}} L_{nu}^{\text{com}}(t) \\ &= \frac{a_u^t p_u^{\text{tr}}}{B_n \log_2 \left(1 + \frac{\delta p_u^{\text{tr}}}{(R_{nu}^t)^2 + H_n^2} \right)} \end{aligned} \quad (6)$$

假设无人机 $n\in\mathcal{N}$ 可以为多个任务分配计算资

源并同时执行它们,其中 $C_n^{max} > 0$ 为无人机 $n \in \mathcal{N}$ 的最大计算能力. 在时隙 $t \in \mathcal{T}$ 中,如果用户 $u \in \mathcal{U}$ 将任务卸载到无人机 $n \in \mathcal{N}$,则计算时延可以表示为

$$L_{nu}^{cop}(t) = \frac{\omega_u^t}{c_{nu}^t}, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T} \quad (7)$$

其中 c_{nu}^t 是无人机 $n \in \mathcal{N}$ 在时隙 $t \in \mathcal{T}$ 时分配给用户 $u \in \mathcal{U}$ 的 CPU 周期数. 需要注意的是,用户 $u \in \mathcal{U}$ 所需的服务应该预先存储在该无人机中,即, $\hat{x}_n^s = 1$. 因此,如果用户 $u \in \mathcal{U}$ 将在时隙 $t \in \mathcal{T}$ 中的任务卸载到无人机 $n \in \mathcal{N}$ 上,则总时延可以表示为

$$\begin{aligned} L_{nu}^{off}(t) &= L_{nu}^{com}(t) + L_{nu}^{cop}(t) \\ &= \frac{a_u^t}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{nu}^t)^2 + H_n^2} \right)} + \frac{\omega_u^t}{c_{nu}^t} \end{aligned} \quad (8)$$

如果用户 $u \in \mathcal{U}$ 将任务卸载到基站 $b \in \mathcal{B}$ 的上,可以采取类似的方法计算总时延. 具体而言,在时隙 $t \in \mathcal{T}$ 时,用户 $u \in \mathcal{U}$ 与基站 $b \in \mathcal{B}$ 的上行链路传输速率可以表示为

$$r_{bu}^t = B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right), \forall u \in \mathcal{U}, b \in \mathcal{B}, t \in \mathcal{T} \quad (9)$$

其中 $R_{bu}^t = \sqrt{(\hat{X}_b - \hat{X}_u^t)^2 + (\hat{Y}_b - \hat{Y}_u^t)^2}$ 是用户 $u \in \mathcal{U}$ 与基站 $b \in \mathcal{B}$ 的水平距离, $\hat{G}_b := (\hat{X}_b, \hat{Y}_b)$ 是基站 $b \in \mathcal{B}$ 的坐标. 用户 $u \in \mathcal{U}$ 将任务上传到基站 $b \in \mathcal{B}$ 的通信时延可以表示为

$$L_{bu}^{com}(t) = \frac{a_u^t}{r_{bu}^t} = \frac{a_u^t}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right)} \quad (10)$$

其中 B_b 是分配给基站 $b \in \mathcal{B}$ 的频谱带宽. 此时,用户 $u \in \mathcal{U}$ 消耗的通信能量为

$$E_{bu}^{com}(t) = p_u^{tx} L_{bu}^{com}(t) = \frac{a_u^t p_u^{tx}}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right)} \quad (11)$$

本文假设基站 $b \in \mathcal{B}$ 可以为多个任务分配计算资源并同时执行它们,其中 $\bar{C}_b^{max} > 0$ 为基站 $b \in \mathcal{B}$ 的最大计算能力. 对于计算部分,基站 $b \in \mathcal{B}$ 处理用户 $u \in \mathcal{U}$ 的第 t 个任务的计算时延为

$$L_{bu}^{cop}(t) = \frac{\omega_u^t}{\bar{c}_{bu}^t}, \forall u \in \mathcal{U}, b \in \mathcal{B}, t \in \mathcal{T} \quad (12)$$

其中 \bar{c}_{bu}^t 是基站 $b \in \mathcal{B}$ 在时隙 t 中分配给用户 $u \in \mathcal{U}$ 的 CPU 周期数. 当然,用户 $u \in \mathcal{U}$ 所需的服务也应该预先存储在基站 $b \in \mathcal{B}$ 中,即, $\hat{x}_b^s = 1$. 因此,如果用户 $u \in \mathcal{U}$ 在时隙 $t \in \mathcal{T}$ 时将任务卸载到基站 $b \in \mathcal{B}$ 上,则总时延可以表示为

$$\begin{aligned} L_{bu}^{off}(t) &= L_{bu}^{com}(t) + L_{bu}^{cop}(t) \\ &= \frac{a_u^t}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right)} + \frac{\omega_u^t}{\bar{c}_{bu}^t} \end{aligned} \quad (13)$$

总而言之,如果合并所有的任务卸载决策,则用户 $u \in \mathcal{U}$ 在时隙 $t \in \mathcal{T}$ 中产生的计算任务的总时延和能耗分别可以表示为

$$\begin{aligned} L_u(t) &= \sum_{n=1}^N \tilde{z}_{nu}^t \left[\frac{a_u^t}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{nu}^t)^2 + H_n^2} \right)} + \frac{\omega_u^t}{c_{nu}^t} \right] + \\ &\quad \sum_{b=1}^B \tilde{z}_{bu}^t \left[\frac{a_u^t}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right)} + \frac{\omega_u^t}{\bar{c}_{bu}^t} \right] + \\ &\quad \left(1 - \sum_{n=1}^N \tilde{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t \right) \frac{\omega_u^t}{\bar{c}_u^t} \end{aligned} \quad (14)$$

$$\begin{aligned} E_u(t) &= \sum_{n=1}^N \tilde{z}_{nu}^t \frac{a_u^t p_u^{tx}}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{nu}^t)^2 + H_n^2} \right)} + \\ &\quad \sum_{b=1}^B \tilde{z}_{bu}^t \frac{a_u^t p_u^{tx}}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right)} + \\ &\quad \left(1 - \sum_{n=1}^N \tilde{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t \right) \kappa_u (\bar{c}_u^t)^{\nu_u - 1} \omega_u^t \end{aligned} \quad (15)$$

其中,如果计算任务卸载到无人机,相应的任务卸载决策变量 $\sum_{n=1}^N \tilde{z}_{nu}^t$ 为 1; 如果计算任务卸载到基站,则

相应的任务卸载决策变量 $\sum_{b=1}^B \tilde{z}_{bu}^t$ 为 1; 如果计算任务在本地执行,则相应的任务卸载决策变量 $1 - \sum_{n=1}^N \tilde{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t$ 为 1. 由此可见,三者对应的任务卸载决策变量之和为 1. 如果计算任务卸载到无人机和基站上,则时延由通信时延和计算时延两部分组成,用户能耗为通信能耗. 如果计算任务在本地执行,则时延仅有计算时延,用户能耗为计算能耗.

2.2 问题描述

本文通过联合优化服务布置 $\mathbf{X} := \{\hat{x}_n^s, \hat{x}_b^s\}_{n \in \mathcal{N}, b \in \mathcal{B}, s \in \mathcal{S}}$, 多架无人机的航迹 $\mathbf{G} := \{G_n^t\}_{n \in \mathcal{N}, t \in \mathcal{T}} = \{(X_n^t, Y_n^t)\}_{n \in \mathcal{N}, t \in \mathcal{T}}$, 任务卸载 $\mathbf{Z} := \{\tilde{z}_{nu}^t, \tilde{z}_{bu}^t\}_{u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}, t \in \mathcal{T}}$ 和计算资源分配 $\mathbf{C} := \{c_{nu}^t, \bar{c}_{nu}^t, \bar{c}_{bu}^t\}_{u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}, t \in \mathcal{T}}$, 以最小化所有用户在此期间的总能耗. 因此,面向空

地协同移动边缘计算的服务布置问题可以形式化表示为

$$\begin{aligned}
 (\text{P1}) \quad & \text{Min}_{\text{X,G,Z,C}} \sum_{u=1}^U \sum_{t=1}^T E_u(t) \\
 \text{s.t.} \quad & \\
 1. \quad & \sum_{s=1}^S l_s \hat{x}_n^s \leq \hat{L}_n^{\max}, \forall n \in \mathcal{N} \\
 2. \quad & \sum_{s=1}^S l_s \tilde{x}_b^s \leq \tilde{L}_b^{\max}, \forall b \in \mathcal{B} \\
 3. \quad & \sum_{n=1}^N \hat{z}_{nu}^t + \sum_{b=1}^B \tilde{z}_{bu}^t \leq 1, \forall u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}, t \in \mathcal{T} \\
 4. \quad & \hat{z}_{nu}^t \leq \hat{x}_n^s, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T} \\
 5. \quad & \tilde{z}_{bu}^t \leq \tilde{x}_b^s, \forall u \in \mathcal{U}, b \in \mathcal{B}, t \in \mathcal{T} \\
 6. \quad & \hat{z}_{nu}^t \|G_n^t - \tilde{G}_u^t\| \leq R_n^{\max}, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T} \\
 7. \quad & \|G_n^{t+1} - G_n^t\| \leq d_n^{\max}, \forall n \in \mathcal{N}, t \in \mathcal{T} \\
 8. \quad & \sum_{u=1}^U \hat{z}_{nu}^t c_{nu}^t \leq C_n^{\max}, \forall n \in \mathcal{N}, t \in \mathcal{T} \\
 9. \quad & \sum_{u=1}^U \tilde{z}_{bu}^t \bar{c}_{bu}^t \leq \bar{C}_b^{\max}, \forall b \in \mathcal{B}, t \in \mathcal{T} \\
 10. \quad & \sum_{u=1}^U \hat{z}_{nu}^t \leq \tilde{N}_n^{\max}, \forall n \in \mathcal{N}, t \in \mathcal{T} \\
 11. \quad & \sum_{u=1}^U \tilde{z}_{bu}^t \leq \tilde{N}_b^{\max}, \forall b \in \mathcal{B}, t \in \mathcal{T} \\
 12. \quad & L_u(t) \leq \tau^{\max}, \forall u \in \mathcal{U}, t \in \mathcal{T} \\
 13. \quad & \hat{x}_n^s, \tilde{x}_b^s \in \{0, 1\}, X_n^t, Y_n^t \in \mathbf{R}, \hat{z}_{nu}^t, \tilde{z}_{bu}^t \in \{0, 1\} \\
 14. \quad & \tilde{c}_{bu}^t \in [0, \bar{C}_b^{\max}], c_{nu}^t \in [0, C_n^{\max}], \bar{c}_{bu}^t \in [0, \bar{C}_b^{\max}]
 \end{aligned} \tag{16}$$

其中,约束1和约束2意味着无人机 n 和基站 b 在存储计算服务时分别不能超过它们各自的存储容量 \hat{L}_n^{\max} 和 \tilde{L}_b^{\max} [15].约束3确保任何计算任务只能卸载到某无人机或某基站上或本地处理.约束4和约束5意味着某个用户要想将其计算任务卸载到某无人机或基站中执行,则用户所需的服务数据必须预先存储在相应的卸载计算服务器中.具体而言,如果用户 $u \in \mathcal{U}$ 在时隙 $t \in \mathcal{T}$ 时的计算任务被卸载到无人机 $n \in \mathcal{N}(\hat{z}_{nu}^t = 1)$.此时用户 $u \in \mathcal{U}$ 所需的服务应该预先存储在该无人机中,即, $\hat{x}_n^s = 1$.如果用 $u \in \mathcal{U}$ 在时隙 $t \in \mathcal{T}$ 时的计算任务被卸载到地面基站 $b \in \mathcal{B}(\tilde{z}_{bu}^t = 1)$.此时用户 $u \in \mathcal{U}$ 所需的服务应该预先存储在该基站中,即, $\tilde{x}_b^s = 1$ [15].约束6意味着如果某个用户想要将其计算任务卸载到某个无人机上,则其前提是此用户必须在这个无人机的通信覆盖范围内[29].约束7限制了每个时隙中最大飞行距离[28].约束8和约束9表示无人机和基站都不能违反其计算容量约束[15].由于频谱有限,约束10和约

束11分别限制了在任何时隙中与无人机或基站相关联的最大用户数目.约束12表示每个时隙中任何用户的任务必须在时隙长度 τ^{\max} 内完成.约束13和约束14提供了所有优化变量的基本可行范围.约束14表示用户、无人机和基站的计算资源分配变量都不能违反其计算容量约束.

3 基于交替优化的服务布置策略

本文提出了一种基于交替优化的算法,其通过迭代求解三个不同的子问题来获得问题的次优解[14,30,31].

3.1 服务布置和任务卸载联合优化子问题

在问题P1中,当给定任意的无人机航迹 G 和计算资源分配 C 时,联合优化服务布置 X 和任务卸载 Z 的子问题P2可以表示为

$$\begin{aligned}
 (\text{P2}) \quad & \text{Min}_{\text{X,Z}} \sum_{u=1}^U \sum_{t=1}^T E_u(t) \\
 \text{s.t.} \quad & \\
 1. \quad & \sum_{s=1}^S l_s \hat{x}_n^s \leq \hat{L}_n^{\max}, \forall n \in \mathcal{N} \\
 2. \quad & \sum_{s=1}^S l_s \tilde{x}_b^s \leq \tilde{L}_b^{\max}, \forall b \in \mathcal{B} \\
 3. \quad & \sum_{n=1}^N \hat{z}_{nu}^t + \sum_{b=1}^B \tilde{z}_{bu}^t \leq 1, \forall u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}, t \in \mathcal{T} \\
 4. \quad & \hat{z}_{nu}^t \leq \hat{x}_n^s, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T} \\
 5. \quad & \tilde{z}_{bu}^t \leq \tilde{x}_b^s, \forall u \in \mathcal{U}, b \in \mathcal{B}, t \in \mathcal{T} \\
 6. \quad & \hat{z}_{nu}^t \|G_n^t - \tilde{G}_u^t\| \leq R_n^{\max}, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T} \\
 8. \quad & \sum_{u=1}^U \hat{z}_{nu}^t c_{nu}^t \leq C_n^{\max}, \forall n \in \mathcal{N}, t \in \mathcal{T} \\
 9. \quad & \sum_{u=1}^U \tilde{z}_{bu}^t \bar{c}_{bu}^t \leq \bar{C}_b^{\max}, \forall b \in \mathcal{B}, t \in \mathcal{T} \\
 10. \quad & \sum_{u=1}^U \hat{z}_{nu}^t \leq \tilde{N}_n^{\max}, \forall n \in \mathcal{N}, t \in \mathcal{T} \\
 11. \quad & \sum_{u=1}^U \tilde{z}_{bu}^t \leq \tilde{N}_b^{\max}, \forall b \in \mathcal{B}, t \in \mathcal{T} \\
 12. \quad & L_u(t) \leq \tau^{\max}, \forall u \in \mathcal{U}, t \in \mathcal{T} \\
 13. \quad & \hat{x}_n^s, \tilde{x}_b^s \in \{0, 1\}, X_n^t, Y_n^t \in \mathbf{R}, \hat{z}_{nu}^t, \tilde{z}_{bu}^t \in \{0, 1\}
 \end{aligned} \tag{17}$$

其中,目标函数和所有约束条件都是线性的.实际上,上述问题是一个经典的多维多选择背包问题(multi-dimensional multiple-choice knapsack problem, MMKP).尽管MMKP问题通常是NP-hard的,并且问题P2有较多约束条件,但问题规模相对较小,可以利用标准优化求解器(如MOSEK)通过分支定界法有效地获得最优解[32].

3.2 无人机航迹优化子问题

在问题P1中,对于任何给定的服务布置 \mathbf{X} 、任务卸载 \mathbf{Z} 和计算资源分配 \mathbf{C} ,无人机航迹优化的子问题P3可以由下式给出

$$(P3) \text{ Min}_{\mathbf{G}} \sum_{u=1}^U \sum_{t=1}^T \sum_{n=1}^N \tilde{z}_{nu}^t \frac{a_u^t p_u^{tx}}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|G_n^t - \tilde{G}_u^t\|^2 + H_n^2} \right)}$$

s.t.

$$6. \tilde{z}_{nu}^t \|G_n^t - \tilde{G}_u^t\| \leq R_n^{max}, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T}$$

$$7. \|G_n^{t+1} - G_n^t\| \leq d_n^{max}, \forall n \in \mathcal{N}, t \in \mathcal{T}$$

$$15. \sum_{n=1}^N \tilde{z}_{nu}^t \left[\frac{a_u^t}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{nu}^t)^2 + H_n^2} \right)} + \frac{\omega_u^t}{c_{nu}^t} \right] + \sum_{b=1}^B \tilde{z}_{bu}^t \left[\frac{a_u^t}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right)} + \frac{\omega_u^t}{\tilde{c}_{bu}^t} \right] + \left(1 - \sum_{n=1}^N \tilde{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t \right) \frac{\omega_u^t}{\tilde{c}_u^t} \leq \tau^{max}$$

$$\forall u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}, t \in \mathcal{T}$$

(18)

其中,目标函数和约束15相对于优化变量 \mathbf{G} 都是非凸的.因此,问题P3是一个连续的、非凸的优化问题.由于目标函数和约束15中的非凸部分完全相同,本文引入一个松弛变量 $\beta: = \{\beta_{mu}^t\}_{n \in \mathcal{N}, u \in \mathcal{U}, t \in \mathcal{T}}$ 以降低问题的复杂度,其中 $\beta_{mu}^t \geq [B_n \log_2(1 + \delta p_u^{tx} (\|G_n^t - \tilde{G}_u^t\|^2 + H_n^2)^{-1})]^{-1}$.此时可以将问题P3改写为问题P4.首先,问题P3中只涉及无人机相关的约束,约束15中关于地面基站的相关变量可以设置为0.此时,约束15可以改写为问题P4中的约束16.其次,由于引入了松弛变量,问题P4需要加入关于松弛变量的约束17.同时,目标函数变成仅关于约束变量的函数.此外,为了简化问题的表达,令 \mathcal{U}_t^{UAV} , \mathcal{U}_t^{BS} 和 \mathcal{U}_t^t 分别表示在时隙 $t \in \mathcal{T}$ 时将其任务卸载到无人机、基站处理,以及本地计算的用户集合,即:

$$\mathcal{U}_t^{UAV}: = \{u \in \mathcal{U}, n \in \mathcal{N}: \tilde{z}_{nu}^t == 1\}$$

$$\mathcal{U}_t^{BS}: = \{u \in \mathcal{U}, b \in \mathcal{B}: \tilde{z}_{bu}^t == 1\}$$

$$\mathcal{U}_t^t: = \{u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}: \tilde{z}_{nu}^t == 0, \tilde{z}_{bu}^t == 0\}$$

因此,问题P3可以改写为

$$(P4) \text{ Min}_{\mathbf{G}, \beta} \sum_{t=1}^T \sum_{u \in \mathcal{U}_t^{UAV}} a_u^t p_u^{tx} \beta_{mu}^t$$

s.t.

$$6. \tilde{z}_{nu}^t \|G_n^t - \tilde{G}_u^t\| \leq R_n^{max}, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T}$$

$$7. \|G_n^{t+1} - G_n^t\| \leq d_n^{max}, \forall n \in \mathcal{N}, t \in \mathcal{T}$$

$$16. B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|G_n^t - \tilde{G}_u^t\|^2 + H_n^2} \right) \geq \frac{a_u^t}{\tau^{max} - \frac{\omega_u^t}{c_{nu}^t}}$$

$$\forall u \in \mathcal{U}_t^{UAV}, n = \{n | \tilde{z}_{nu}^t == 1\}$$

$$17. B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|G_n^t - \tilde{G}_u^t\|^2 + H_n^2} \right) \geq \frac{1}{\beta_{mu}^t}$$

(19)

在问题P4中,目标函数在 β 中是凸的,约束16的左侧在 $\|G_n^t - \tilde{G}_u^t\|$ 中是凸的,约束17的左侧和右侧分别在 $\|G_n^t - \tilde{G}_u^t\|$ 和 β 中都是凸的.这意味着只需要处理约束16和17的不等式左侧即可.本文采用SCA技术来寻找问题P4的局部最优解^[26].具体而言,在第 j 次SCA迭代中对于 $\mathbf{G}(j): = \{G_n^t(j)\}$ 中给定的任何局部点 $G_n^t(j)$,约束16和17的不等式左侧可以用下面的下界来近似

$$B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|G_n^t - \tilde{G}_u^t\|^2 + H_n^2} \right) \geq \phi_{mu}^t(j) (\|G_n^t - \tilde{G}_u^t\|^2 - \|G_n^t(j) - \tilde{G}_u^t\|^2) + \varphi_{mu}^t(j)$$

(20)

其中,

$$\phi_{mu}^t(j) = \frac{-B_n \delta p_u^{tx} \log_2 e}{(\|G_n^t(j) - \tilde{G}_u^t\|^2 + H_n^2)} \times \frac{1}{(\|G_n^t(j) - \tilde{G}_u^t\|^2 + H_n^2 + \delta p_u^{tx})}$$

(21)

$$\varphi_{mu}^t(j) = B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|G_n^t(j) - \tilde{G}_u^t\|^2 + H_n^2} \right)$$

因此,第 j 次迭代中的问题P4可以表示为

$$(P4^j) \text{ Min}_{\mathbf{G}, \beta} \sum_{t=1}^T \sum_{u \in \mathcal{U}_t^{UAV}} a_u^t p_u^{tx} \beta_{mu}^t$$

s.t.

$$6. \tilde{z}_{nu}^t \|G_n^t - \tilde{G}_u^t\| \leq R_n^{max}, \forall u \in \mathcal{U}, n \in \mathcal{N}, t \in \mathcal{T}$$

$$7. \|G_n^{t+1} - G_n^t\| \leq d_n^{max}, \forall n \in \mathcal{N}, t \in \mathcal{T}$$

$$\phi_{mu}^t(j) (\|G_n^t - \tilde{G}_u^t\|^2 - \|G_n^t(j) - \tilde{G}_u^t\|^2) + \varphi_{mu}^t(j)$$

$$\geq \frac{a_u^t}{\tau^{max} - \frac{\omega_u^t}{c_{nu}^t}}, \forall u \in \mathcal{U}_t^{UAV}, n \in \mathcal{N}, t \in \mathcal{T}$$

$$\phi_{mu}^t(j) (\|G_n^t - \tilde{G}_u^t\|^2 - \|G_n^t(j) - \tilde{G}_u^t\|^2) + \varphi_{mu}^t(j) \geq \frac{1}{\beta_{mu}^t}$$

(22)

问题P4^j是一个凸二次约束的二次规划问题,其最优解可由标准的凸求解器(如CVX)来获得^[33].

3.3 计算资源分配子问题

在给定的服务布置 \mathbf{X} 、任务卸载 \mathbf{Z} 和无人机航迹 \mathbf{G} 的情况下,可以得到计算资源分配的最优解,如下面引理所示.

引理 1. 在问题 P1 中,对于任何给定的服务布置 \mathbf{X} 、任务卸载 \mathbf{Z} 和无人机航迹 \mathbf{G} ,若问题可解,则最优计算资源分配 $\{\tilde{c}_u^{*t}, c_{nu}^{*t}, \bar{c}_{bu}^{*t}\}$ 的闭式解为

$$\tilde{c}_u^{*t} = \begin{cases} \frac{w_u^t}{\tau^{max}}, & u \in \mathcal{U}_t^L \\ 0, & u \in \mathcal{U}_t^{UAV} \cup \mathcal{U}_t^{BS} \end{cases} \quad (23)$$

$$c_{nu}^{*t} = \begin{cases} \frac{w_u^t}{\tau^{max} - \frac{a_u^t}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|G_n^t - \tilde{G}_u^t\|^2 + H_n^2} \right)}}, & u \in \mathcal{U}_t^{UAV} \\ 0, & \text{otherwise} \end{cases} \quad (24)$$

$$\bar{c}_{bu}^{*t} = \begin{cases} \frac{w_u^t}{\tau^{max} - \frac{a_u^t}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|\hat{G}_b - \tilde{G}_u^t\|^2} \right)}}, & u \in \mathcal{U}_t^{BS} \\ 0, & \text{otherwise} \end{cases} \quad (25)$$

其中 $u \in \mathcal{U}_t^L, \mathcal{U}_t^{UAV}$ 和 \mathcal{U}_t^{BS} 分别表示在时隙 $t \in \mathcal{T}$ 时将其任务在本地处理、卸载到无人机和基站的用户集合.

证明. 给定服务布置 \mathbf{X} 、任务卸载 \mathbf{Z} 和无人机航迹 \mathbf{G} ,计算资源分配子问题 P5 可以简化为

$$(P5) \text{ Min}_C \sum_{u=1}^U \sum_{t=1}^T \left(1 - \sum_{n=1}^N \hat{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t \right) \kappa_u(\tilde{c}_u^t)^{\nu_u-1} w_u^t$$

s.t.

$$8. \sum_{u=1}^U \hat{z}_{nu}^t c_{nu}^t \leq C_n^{max}, \forall n \in \mathcal{N}, t \in \mathcal{T}$$

$$9. \sum_{u=1}^U \tilde{z}_{bu}^t \bar{c}_{bu}^t \leq \bar{C}_b^{max}, \forall b \in \mathcal{B}, t \in \mathcal{T}$$

$$15. \sum_{n=1}^N \hat{z}_{nu}^t \left[\frac{a_u^t}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{nu}^t)^2 + H_n^2} \right)} + \frac{w_u^t}{c_{nu}^t} \right] + \sum_{b=1}^B \tilde{z}_{bu}^t \left[\frac{a_u^t}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{(R_{bu}^t)^2} \right)} + \frac{w_u^t}{\bar{c}_{bu}^t} \right] +$$

$$\left(1 - \sum_{n=1}^N \hat{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t \right) \frac{w_u^t}{\tilde{c}_u^t} \leq \tau^{max} \quad \forall u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}, t \in \mathcal{T} \quad (26)$$

由于目标函数在 $(\tilde{c}_u^t) (\nu_u = 3)$ 的加权和是线性的且约束在 c_{nu}^t, \bar{c}_{bu}^t 和 \tilde{c}_u^t 中是凸的,因此问题 P5 是一个凸问题. 需要注意的是,目标函数和所有约束条件都是关于时隙 $t \in \mathcal{T}$ 独立的. 这意味着我们可以逐个时隙确定计算资源分配的解. 因此,上述优化问题可以通过求解 T 个独立子问题等价地来解决. 具体而言,独立的求解任意时隙 $t \in \mathcal{T}$ 中的子问题 P5' 可以得到 $\tilde{c}_u^t, c_{nu}^t, \bar{c}_{bu}^t$. 综合 T 个独立的子问题的解,我们可以得到计算资源分配子问题 P5 的解 $\mathbf{C} := \{\tilde{c}_u^t, c_{nu}^t, \bar{c}_{bu}^t\}_{u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B}, t \in \mathcal{T}}$. 其中,在时隙 t 的子问题 P5' 可以表示为

$$(P5') \text{ Min}_{\tilde{c}_u^t, c_{nu}^t, \bar{c}_{bu}^t} \sum_{n=1}^N \left(1 - \sum_{n=1}^N \hat{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t \right) \kappa_u(\tilde{c}_u^t)^{\nu_u-1} w_u^t$$

s.t.

$$8. \sum_{u=1}^U \hat{z}_{nu}^t c_{nu}^t \leq C_n^{max}, \forall n \in \mathcal{N}, t \in \mathcal{T}$$

$$9. \sum_{u=1}^U \tilde{z}_{bu}^t \bar{c}_{bu}^t \leq \bar{C}_b^{max}, \forall b \in \mathcal{B}, t \in \mathcal{T}$$

$$15. \sum_{n=1}^N \hat{z}_{nu}^t \left[\frac{a_u^t}{B_n \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|G_n^t - \tilde{G}_u^t\|^2 + H_n^2} \right)} + \frac{w_u^t}{c_{nu}^t} \right] +$$

$$\sum_{b=1}^B \tilde{z}_{bu}^t \left[\frac{a_u^t}{B_b \log_2 \left(1 + \frac{\delta p_u^{tx}}{\|\hat{G}_b - \tilde{G}_u^t\|^2} \right)} + \frac{w_u^t}{\bar{c}_{bu}^t} \right] +$$

$$\left(1 - \sum_{n=1}^N \hat{z}_{nu}^t - \sum_{b=1}^B \tilde{z}_{bu}^t \right) \frac{w_u^t}{\tilde{c}_u^t} \leq \tau^{max}, \forall u \in \mathcal{U}, n \in \mathcal{N}, b \in \mathcal{B} \quad (27)$$

如果将集合 $\mathcal{U}_t^L, \mathcal{U}_t^{UAV}$ 和 \mathcal{U}_t^{BS} 表示的任务卸载决策带入到 P5' 中,可以得到该问题的最优解. 此时,便可获得引理 1 中得到的最优解. 证毕.

3.4 算法设计和分析

算法 1 详细介绍了基于交替优化的服务布置策略. 其主要思想是分别通过 BnB 方法、SCA 技术和引理 1,迭代优化服务布置和任务卸载(第 3 行)、无人机航迹(第 4 行)和计算资源分配(第 5 行).

算法 1. 基于交替优化的服务布置算法.

输入:可行的初始解 $(\mathbf{X}^0, \mathbf{Z}^0, \mathbf{G}^0, \mathbf{C}^0)$, $\epsilon > 0$, $k = 1$,

$K^{max} = 100$.

输出: 服务布置 X 、任务卸载 Z 、无人机航迹 G 和计算资源分配 C .

1. 计算 $W^0 = Obj(X^0, Z^0, G^0, C^0)$, 其中 $Obj(X, Z, G, C)$ 是问题 P1 的目标函数;
2. WHILE $k < K^{max}$ DO
3. 给定 (G^{k-1}, C^{k-1}) , 通过 BnB 法获得 X^k 和 Z^k ;
4. 给定 (X^k, Z^k, C^{k-1}) , 通过 SCA 技术获得 G^k ;
5. 给定 (X^k, Z^k, G^k) , 通过引理 1 获得 C^k ;
6. IF $W^k - W^{k-1} / W^{k-1} \leq \epsilon$ THEN
7. BREAK.
8. ELSE
9. $k \leftarrow k + 1$.
10. 返回 (X^k, Z^k, G^k, C^k) .

定理 1. 算法 1 能够收敛, 复杂度为 $O[K^{max}(2^{(N+B)S+(N+B)UT} + I^{max}N^3T^3)]$.

证明. 定理 1 只需要证明, 当 (X, Z, G, C) 在算法 1 中更新时, 目标函数 $Obj(X, Z, G, C)$ 的值保持不增即可. 通过算法 1, 可以得到

$$\begin{aligned}
 V^{k-1} &= Obj(X^{k-1}, Z^{k-1}, G^{k-1}, C^{k-1}) \\
 &\geq Obj(X^k, Z^k, G^{k-1}, C^{k-1}) \\
 &\geq Obj(X^k, Z^k, G^{k-1}, C^*(X^k, Z^k, G^{k-1})) \\
 &\geq Obj(X^k, Z^k, G^k, C^*(X^k, Z^k, G^{k-1})) \quad (28) \\
 &\geq Obj(X^k, Z^k, G^k, C^*(X^k, Z^k, G^k)) \\
 &= Obj(X^k, Z^k, G^k, C^k) = V^k
 \end{aligned}$$

其中, $C^*(X, Z, G)$ 是通过引理 1 获得的计算资源分配的最优解. 第一个不等式成立的原因是 (X^k, Z^k) 的最优性. 第二个和第四个不等式成立的原因是 $C^*(\cdot, \cdot, \cdot)$ 的最优性. 第三个不等式成立的原因是 G^k 的次优性. 因此, 目标函数的值在每次迭代后始终不增加, 从而在多次迭代后收敛. 关于复杂度, 由于利用 BnB 法来解决 MMKP 问题, 算法 1 的复杂度是指数级的. 然而, 由于问题规模不大, 其运行时间是可以接受的. 具体而言, 算法 1 通过迭代求解三个不同的子问题来获得问题的最优解. 首先, 采用分支定界法求解联合服务布置和任务卸载的子问题, 该子算法的复杂度为 $O(2^{(N+B)S+(N+B)UT})$. 其次, 采用 SCA 方法求解无人机航迹优化的子问题. 该子算法的复杂度与 SCA 的迭代次数 I^{max} 和航迹优化变量的维度有关, 即 $O(I^{max}N^3T^3)$. 因此, 算法 1 的总体复杂度为 $O[K^{max}(2^{(N+B)S+(N+B)UT} + I^{max}N^3T^3)]$, 其中 K^{max} 为该算法的最大迭代次数.

证毕.

4 仿真结果与分析

为了验证所提出的基于交替优化的服务布置策略的有效性, 本文进行了大量的仿真, 并且实现了以下三种比较算法: 1) 随机布置策略: 该算法将服务随机布置在无人机或基站上, 直到达到其存储容量; 2) 贪心布置策略: 该算法使用贪心策略每次将所需存储容量最小的服务布置在无人机或基站上, 使得存储在无人机或基站上的服务数量尽可能最大化; 3) 本地计算策略: 所有任务都是在本地执行的, 其性能可以看作是用户总能耗的上界.

4.1 仿真设置

本文考虑了一个空地协同移动边缘计算网络, 该网络由 2 架无人机、1 个静态地面基站和 15 个地面移动用户组成, 其中基站和无人机都配备了边缘计算服务器. 假设基站部署在 $200\text{ m} \times 200\text{ m}$ 的正方形区域中, 其坐标为 $(100, 50)$, 且用户随机分布在正方形区域中^[26,30,34]. 假设整个任务周期共有 $T = 100$ 个时隙, 并且每个时隙的长度为 $\tau^{max} = 1\text{ s}$. 每个用户在每个时隙中都会产生一个计算密集型任务, 如智能导航、人脸识别和增强现实等^[11,15]. 本文考虑多种类型的计算任务, 并且给不同的计算任务设置了不同的仿真参数值, 其中输入数据大小为 $a'_u \in [100, 1000]\text{ KB}$ ^[30], 所需的 CPU 周期数(即计算负载)为 $w'_u \in [10^8, 10^9]$ ^[11]. 例如, 在安全和监控应用中, 人脸识别和检测是一项关键的计算任务, 其输入数据大小可以设置为 420 KB ^[35]. 计算负载与输入数据大小的比值为计算一个输入比特所需的 CPU 周期数^[30,36]. 假设用户请求的服务 $S = 30$ 是根据常用的 Zipf 分布生成的, 其偏度参数值为 0.5 ^[15]. 每种服务所需的存储容量 l_s 归一化后从 $[0.5, 1]$ 随机取值, 无人机和基站存储容量的范围分别为 $[0.2S, 0.4S]$ 和 $[0.4S, 0.6S]$ ^[34]. 对于通信参数设置, 本文假设用户的最大传输功率为 $p_u^a = 0.1\text{ w}$, 每个信道的可用频谱带宽为 $B = 1\text{ MHz}$, 单位信道功率增益为 $g_o = 1.42 \times 10^{-4}$, 噪声功率为 $\sigma^2 = -90\text{ dbm/Hz}$ ^[26]. 对于计算能耗参数设置, 参数 κ_u 和 ν_u 分别设置为 10^{-27} 和 3 ^[26]. 与无人机相关的参数设置如下: 最大水平覆盖范围为 $R_n^{max} = 100\text{ m}$, 每个时隙中无人机的最大移动距离为 $d_n^{max} = 30\text{ m}$, 与无人机相关联的最大用户数为 $\tilde{N}_n^{max} \in [3, 4]$, 无人机的最大计算能力为 $C_n^{max} \in [5, 10]\text{ GHz}$ ^[5]. 与基站相关联的最大

用户数为 $\tilde{N}_b^{max} \in [6, 8]$, 基站的最大计算能力为 $\bar{C}_b^{max} \in [10, 15]$ GHz.

4.2 仿真结果分析

4.2.1 算法收敛性验证

图2验证了所提出的基于交替优化的服务布置策略的收敛性. 如图2所示, 该策略的收敛速度较快, 采用1架无人机1个基站时, 仅需4次迭代算法就收敛; 采用2架无人机1个基站时, 也只需6次迭代. 相比一架无人机的情况, 采用2架无人机后明显减少了所有用户的总能耗: 从图中可以看到, 总的用户能耗从147.19 J下降到44.94 J. 除此之外, 随着无人机和基站数目的增多, 算法仍然能够收敛. 无人机和基站数量的增加能够降低总的用户能耗, 直到运算能力足够, 不再需要增加更多无人机.

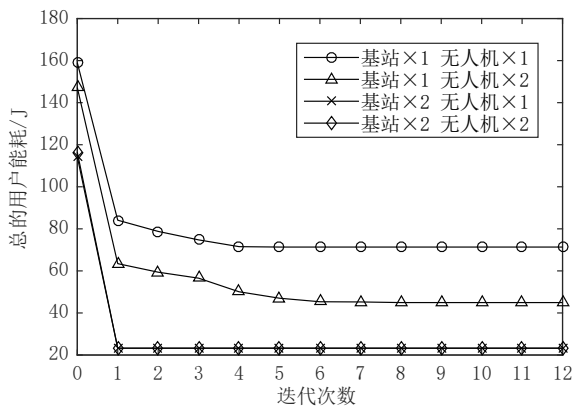


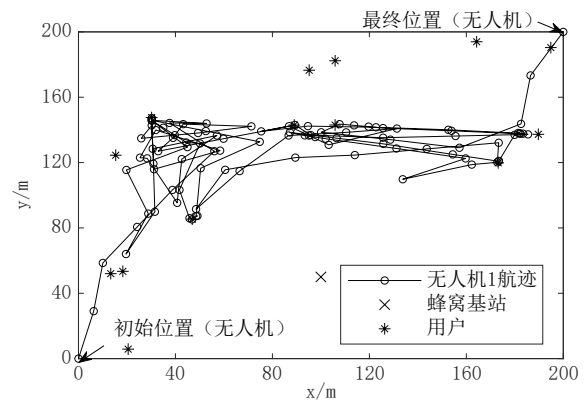
图2 收敛结果分析

4.2.2 无人机航迹

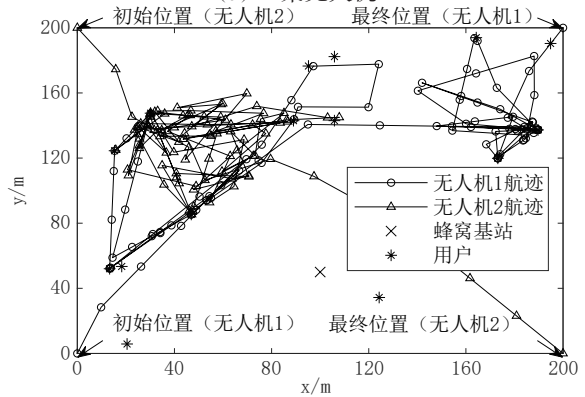
图3展示了基于交替优化的服务布置策略优化的无人机航迹. 其中, 无人机1的初始和最终的水平位置分别为(0, 0)和(200, 200), 无人机2的初始和最终的水平位置分别为(0, 200)和(200, 0). 由于无人机能耗有限, 如果不进行航迹优化, 无人机为了节省能量资源, 可能会沿最短路径飞行. 在本文的仿真场景中, 两架无人机可能会沿对角线飞行, 从而失去不少满足地面用户计算需求的机会. 相比之下, 本文提出的策略优化了无人机的航迹, 使其接近用户, 以便更好地提供计算服务. 图4和图5展示了随机布置策略和贪心布置策略优化的无人机航迹. 从图中可以看出, 不同方法得到的无人机航迹有所不同. 这是因为, 不同方法的服务布置策略不同, 那么相应得到的无人机航迹也有所不同.

4.2.3 任务执行位置分析

图6从任务数量和任务计算量两个角度分析了



(a) 一架无人机



(b) 两架无人机

图3 基于交替优化的服务布置策略下的无人机航迹

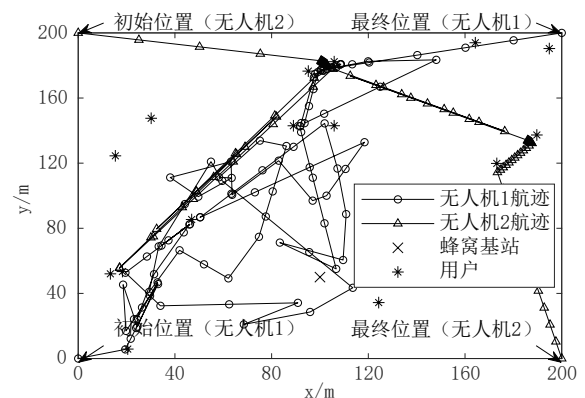


图4 随机布置策略下的无人机航迹

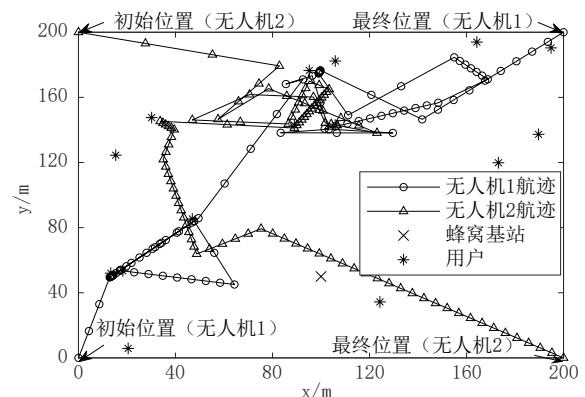


图5 贪心布置策略下的无人机航迹

计算任务的执行位置. 具体而言,其展示了图3、图4和图5中三种算法多少任务是用户在本地直接完成的,多少任务是通过无人机来完成的,以及多少任务是通过基站来完成. 从完成任务数量的角度,随机布置算法里三者处理比例分别是45.4%,12.3%,42.2%;贪心布置算法里三者处理比例分别是46.3%,3.5%,50.3%;本文提出的算法里三者处理比例分别是29.5%,20.3%,50.2%. 从完成任务计算量的角度,随机布置算法里三者处理比例分别是35.5%,14.8%,49.6%;贪心布置算法里三者处理比例分别是36.0%,4.3%,59.8%;本文提出的算法里三者处理比例分别是14.1%,24.7%,61.2%. 从图中可以看出,相比随机布置策略和贪心布置策略,本文提出的算法充分利用无人机的优势,将部分计算任务卸载到无人机上,减少了总的用户能耗.

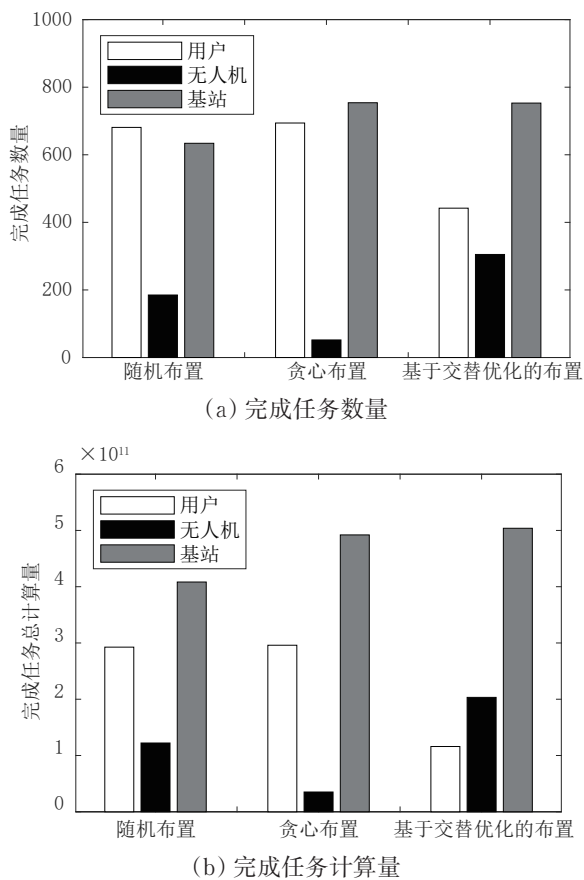


图6 计算任务的执行位置对比

4.2.4 无人机存储容量的影响

图7展示了总的用户能耗与无人机的存储容量之间的关系. 特别的,本文的存储空间都是归一化之后的,实际的服务器存储容量应该是要乘以多少M或者G的. 为方便起见,类似于文献[34],本文将

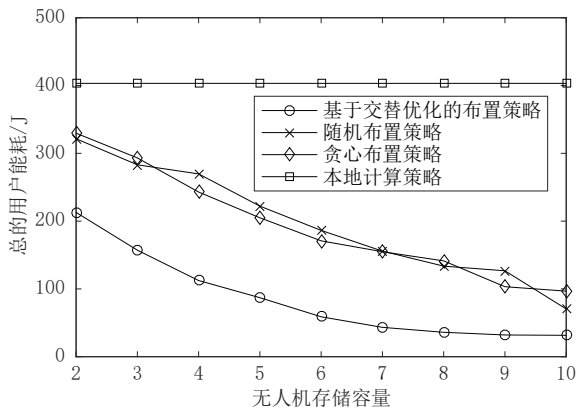
其归一化,以展示不同算法随着存储容量变化下的性能变化趋势. 如图7(a)和图7(b)所示,除本地计算策略以外,其他三种策略的总的用户能耗都降低了. 这是由于增加无人机的存储容量,使得其可以存储更多用户所需的服务,减少了用户在本地执行的能量消耗. 对于一架无人机的移动边缘计算网络,当无人机的存储容量从2增加到10时,基于交替优化的服务布置策略与本地计算策略、随机布置策略和贪心布置策略相比,总的用户能耗减少了78.74%,60.10%和60.16%. 对于两架无人机的移动边缘计算网络,当无人机的存储容量从2增加到10时,基于交替优化的服务布置策略与本地计算策略、随机布置策略和贪心布置策略相比,总的用户能耗减少了83.36%,61.13%和68.50%. 由图7(c)可知,两架无人机的移动边缘计算网络相比一架无人机来说,在不同的无人机存储容量方案下总的用户能耗平均减少了20.50%. 然而,当无人机存储容量增大到10时,一台无人机与两台无人机对应的用户总能耗逐渐接近一致. 这是由于,当无人机存储容量增大时,单个无人机可以存储更多的计算服务,计算能力增强. 因此,在地面用户数目和计算任务不变的情况下,随着无人机存储容量的增加,单个无人机可以为更多的地面用户提供服务,计算性能提升,原有的计算任务可能只需要一架无人机就可以完成,从而一架无人机与两架无人机的性能接近.

4.2.5 用户计算负载的影响

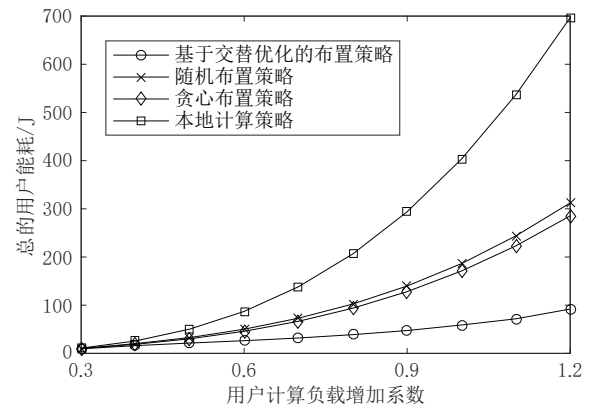
图8显示了总的用户能耗与用户计算负载之间的关系. 如图8(a)所示,对于一架无人机的移动边缘计算网络,当每个用户的计算负载从仿真设置的0.3倍变化到1.2倍时,基于交替优化的服务布置策略的总的用户能耗平均比本地计算策略、随机布置策略和贪心布置策略减少了65.78%,48.01%和44.17%. 如图8(b)所示,对于两架无人机的移动边缘计算网络,基于交替优化的服务布置策略的总的用户能耗平均比本地计算策略、随机布置策略和贪心布置策略减少了68.78%,47.50%和50.68%. 由图8(c)可知,两架无人机的移动边缘计算网络相比一架无人机来说,在不同的用户计算负载方案下总的用户能耗平均减少了19.12%.

4.2.6 用户移动性的影响

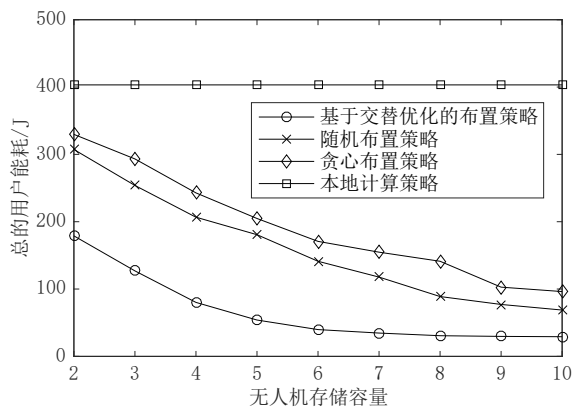
图9展示了用户移动轨迹和无人机航迹. 为了验证用户移动时所提出算法的性能和无人机航迹的变化,本文假设用户采用随机游走的策略. 从图中可以看出,优化的无人机航迹与大部分用



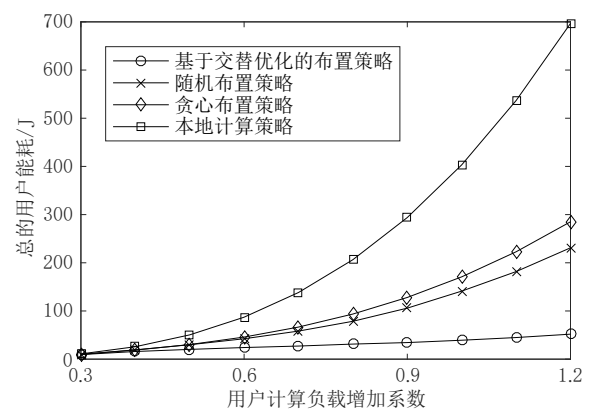
(a) 一架无人机



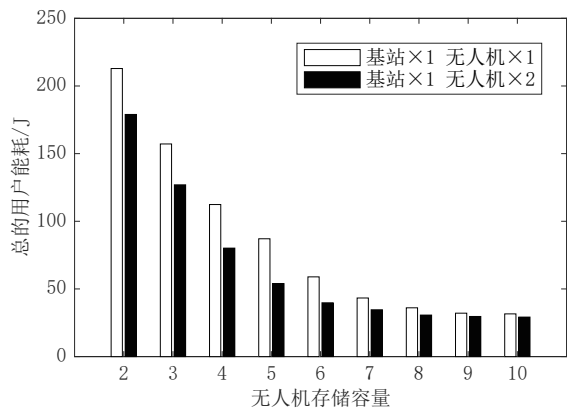
(a) 一架无人机



(b) 两架无人机

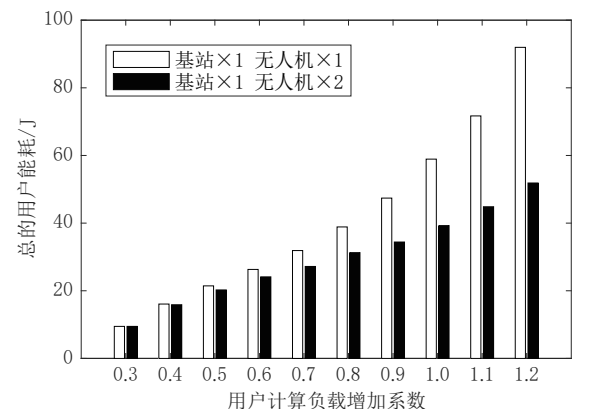


(b) 两架无人机



(c) 不同无人机数目对比

图7 不同无人机存储容量下总的用户能耗变化



(c) 不同无人机数目对比

图8 不同用户计算负载下总的用户能耗变化

用户的移动轨迹密切相关. 这主要是因为本文提出的算法会根据用户的移动性信息优化无人机的航迹, 使其更接近用户, 以便更好地提供计算服务. 由此可见, 本文所提出的方法是能够解决用户移动时的问题.

4.2.7 可行初始解的影响

图10验证了不同的初始解对于算法性能的影响. 本文采用了三个不同的初始解进行验证. 从图中可以看出来, 本文提出的算法在6次迭代后都能

够收敛至一个较为接近的值. 由此可以得到, 不同的初始解对于算法的求解精度和收敛速度的影响较小. 同时, 可以看出本文提出的算法性能较好, 不会因为初始解的不同, 从而陷入局部最优.

5 讨论

5.1 用户能耗和无人机能耗的权衡

无人机辅助的移动边缘计算系统不同于地面移

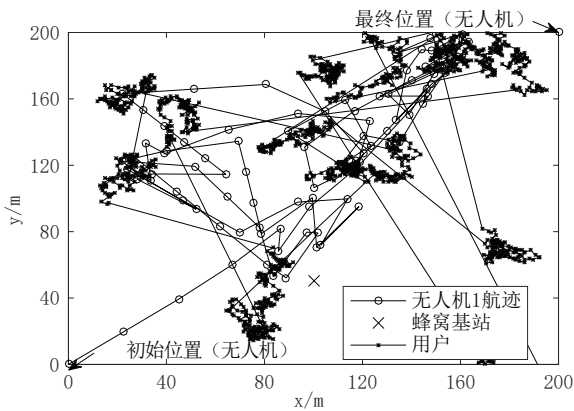


图9 用户移动轨迹和无人机航迹示意图

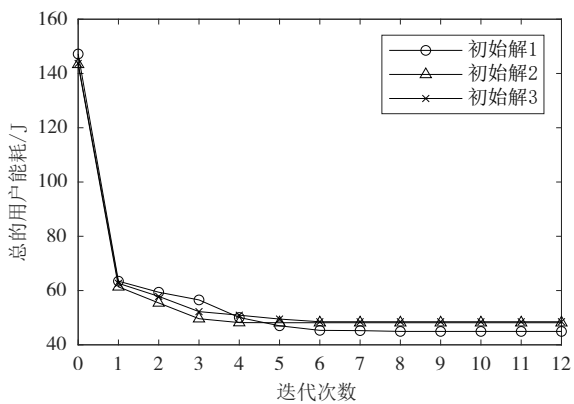


图10 不同初始解下总的用户能耗变化

动边缘计算系统,其能量资源有限.本文所提出的方法可以降低用户的能耗,而且无人机越多用户能耗降低越多,但是引入无人机本身也将会产生大量能耗.事实上,增加无人机的数量并不是一个有效的方法,因为它会增加计算成本,从而带来经济压力.因此,在提高计算性能的同时,需要权衡无人机能量消耗和用户能量消耗.在后续的研究中,可以采用加权的方式在目标函数中同时考虑用户和无人机的能量消耗.通过调整加权因子,可以针对不同的应用场景,灵活的调节对于用户和无人机的关注度.

5.2 用户移动性未知的情况

事实上,无论用户采取何种游走策略,只要用户的移动轨迹已知,本文提出的方法都适用.如果用户的移动轨迹不是已知的但可以通过现有的预测方法以较高的概率预测,算法依然可以求解.例如,用户的移动性可以采用无监督学习工具预测(如核密度估计方法^[37]).在未来工作中,如何在用户移动轨迹无法预测或者预测不准确时提升算法的性能值得我们深入研究.

5.3 可行初始解构建

关于算法1的可行初始解:首先,将无人机航迹

的初始解 G^0 设置为沿正方形仿真区域的对角线飞行的航迹.然后采用随机布置的策略获得一个服务布置初始解 X^0 .具体而言,将服务随机布置在无人机或基站上,直到达到其存储容量,由此可以得到一个可行的服务布置初始解.在上述无人机航迹的初始解 G^0 和服务布置初始解 X^0 确定后,问题P1中的约束1、2和7自然满足了.此时,剩余约束条件都是关于时隙 t 独立的.这意味着我们可以逐个时隙确定任务卸载初始解 Z^0 和计算资源配的初始解 C^0 .对于任意的时隙 t ,从尚未选择的用户集合中随机选择一个用户,并根据下面的规则为该用户进行任务卸载决策和计算资源分配.具体而言,先为该用户选择一个无人机或基站,前提是该用户所需的服务已经存储在这个无人机或基站中,即满足约束4和5.如果选择的是无人机,用户还应该在无人机的覆盖范围内,即满足约束6.用户所需的计算资源由无人机或基站提供(约束8和9),其同时也不能违反无人机和基站服务的最大用户数目(约束10和11),并且能够满足时延约束(约束12).用户所需的最小计算资源由时延约束12所确定.如果上述条件都能满足,则将相应的任务卸载决策值设置为1,并记录计算资源的分配情况;否则,计算任务将在本地执行.需要注意的是,如果存在多个满足上述条件的无人机和基站,随机选择其中的一个即可,得到相应的初始解.综上所述,可得到问题P1的一个可行初始解 (X^0, Z^0, G^0, C^0) .

6 总结与展望

本文研究了空地协同移动边缘计算的服务布置问题,其可以建模为一个在各种网络资源和任务延迟约束下的用户总能耗最小化的问题.该问题涉及到服务布置、无人机航迹、任务卸载和计算资源分配的联合优化,属于一个非凸混合整数非线性规划问题.基于交替优化技术,本文提出了一种具有收敛性保证的次优算法.在后续的研究中,可以进一步考虑地面上存在其他潜在可用的边缘计算服务器的应用场景.

参 考 文 献

- [1] Hu Y C, Patel M, Sabella D, et al. Mobile edge computing—a key technology towards 5G. ETSI White Paper, 2015, 11(11): 1-16

- [2] Sabella D, Moustafa H, Kuure P, et al. Toward fully connected vehicles: edge computing for advanced automotive communications. 5G Automotive Association White Paper, 2017
- [3] Liu Y, Xiong K, Ni Q, et al. UAV-assisted wireless powered cooperative mobile edge computing: joint offloading, CPU control, and trajectory optimization. *IEEE Internet of Things Journal*, 2020, 7(4): 2777-2790
- [4] Wang J R, Liu K Y, Pan J P. Online UAV-mounted edge server dispatching for mobile-to-mobile edge computing. *IEEE Internet of Things Journal*, 2020, 7(2): 1375-1386
- [5] Zhang X C, Zhang J, Xiong J, et al. Energy-efficient multi-UAV-enabled multiaccess edge computing incorporating NOMA. *IEEE Internet of Things Journal*, 2020, 7(6): 5613-5627
- [6] Motlagh N H, Bagaa M, Taleb T. UAV-based IoT platform: a crowd surveillance use case. *IEEE Communications Magazine*, 2017, 55(2): 128-134
- [7] Zhou F, Hu R Q, Li Z, et al. Mobile edge computing in unmanned aerial vehicle networks. *IEEE Wireless Communications*, 2020, 27(1): 140-146
- [8] Cheng N, Xu W, Shi W, et al. Air-ground integrated mobile edge networks: architecture, challenges, and opportunities. *IEEE Communications Magazine*, 2018, 56(8): 26-32
- [9] Jeong S, Simeone O, Kang J. Mobile edge computing via a UAV-mounted cloudlet: optimization of bit allocation and path planning. *IEEE Transactions on Vehicular Technology*, 2018, 67(3): 2049-2063
- [10] Zhou F H, Wu Y P, Hu R Q, et al. Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing networks. *IEEE Journal on Selected Areas in Communications*, 2018, 36(9): 1927-1941
- [11] Yang Z H, Pan C H, Wang K, et al. Energy efficient resource allocation in UAV-enabled mobile edge computing networks. *IEEE Transactions on Wireless Communications*, 2019, 18(9): 4576-4589
- [12] Zhou Z, Feng J, Tan L, et al. An air-ground integration approach for mobile edge computing in IoT. *IEEE Communications Magazine*, 2018, 56(8): 40-47
- [13] Jiang F, Wang K, Dong L, et al. AI driven heterogeneous MEC system with UAV assistance for dynamic environment: challenges and solutions. *IEEE Network*, 2020, 35(1): 400-408
- [14] Hu X Y, Wong K, Yang K, et al. UAV-assisted relaying and edge computing: scheduling and trajectory optimization. *IEEE Transactions on Wireless Communications*, 2019, 18(10): 4738-4752
- [15] Farhadi V, Mehmeti F, He T, et al. Service placement and request scheduling for data-intensive applications in edge clouds// *Proceedings of the IEEE Conference on Computer Communications*. Paris, France, 2019: 1279-1287
- [16] Poularakis K, Llorca J, Tulino A M, et al. Joint service placement and request routing in multi-cell mobile edge computing networks// *Proceedings of the IEEE Conference on Computer Communications*. Paris, France, 2019: 10-18
- [17] Huang M, Liang W, Shen X, et al. Reliability-aware virtualized network function services provisioning in mobile edge computing. *IEEE Transactions on Mobile Computing*, 2020, 19(11): 2699-2713
- [18] Wang L, Jiao L, He T, et al. Service entity placement for social virtual reality applications in edge computing// *Proceedings of IEEE Conference on Computer Communications*. Honolulu, USA, 2018: 468-476
- [19] Bahreini T, Grosu D. Efficient algorithms for multi-component application placement in mobile edge computing. *IEEE Transactions on Cloud Computing*, 2020, Early Access
- [20] Zhang Y, Jiao L, Yan J, et al. Dynamic service placement for virtual reality group gaming on mobile edge cloudlets. *IEEE Journal on Selected Areas in Communications*, 2019, 37(8): 1881-1897
- [21] Xu J, Chen L X, Zhou P. Joint service caching and task offloading for mobile edge computing in dense networks// *Proceedings of the IEEE Conference on Computer Communications*. Honolulu, USA, 2018: 207-215
- [22] Ouyang T, Zhou Z, Chen X. Follow me at the edge: mobility-aware dynamic service placement for mobile edge computing. *IEEE Journal on Selected Areas in Communications*, 2018, 36(10): 2333-2345
- [23] Gouareb R, Friderikos V, Aghvami A. Virtual network functions routing and placement for edge cloud latency minimization. *IEEE Journal on Selected Areas in Communications*, 2018, 36(10): 2346-2357
- [24] Wang L, Jiao L, He T, et al. Cost aware cloudlet placement for big data processing at the edge// *Proceedings of the IEEE International Conference on Communications*. Paris, France, 2017: 1-6
- [25] Han B, Gopalakrishnan V, Kathirvel G, et al. On the resiliency of virtual network functions. *IEEE Communications Magazine*, 2017, 55(7): 152-157
- [26] Wang L, Wang K Z, Pan C H, et al. Deep reinforcement learning based dynamic trajectory control for UAV-assisted mobile edge computing. *IEEE Transactions on Mobile Computing*, 2021, Early Access
- [27] Zhang J, Zhou L, Zhou F H, et al. Computation-efficient offloading and trajectory scheduling for multi-UAV assisted mobile edge computing. *IEEE Transactions on Mobile Computing*, 2020, 69(2): 2114-2125
- [28] Wu Q, Zeng Y, Zhang R. Joint trajectory and communication design for multi-UAV enabled wireless networks. *IEEE Transactions on Wireless Communications*, 2018, 17(3): 2109-2121
- [29] Al-Hourani A, Kandeepan S, Lardner S. Optimal LAP altitude for maximum coverage. *IEEE Wireless Communications Letters*, 2014, 3(6): 569-572
- [30] Hu J, Jiang M, Zhang Q, et al. Joint optimization of UAV position, time slot allocation, and computation task partition in multiuser aerial mobile-edge computing systems. *IEEE Transactions on Vehicular Technology*, 2019, 68(7): 7231-7235
- [31] Bi S, Zhang Y J. Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading. *IEEE Transactions on Wireless Communications*, 2018, 17(6): 4177-4190

- [32] MOSEK software, <http://www.mosek.com>.
- [33] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. <http://cvxr.com/cvx>, March 2014
- [34] Qu Y, Dai H P, Wang H C, et al. Service provisioning for UAV-enabled mobile edge computing. *IEEE Journal on Selected Areas in Communications*, 2021, 39(11): 3287-3305
- [35] Tran T X and Pompili D. Joint task offloading and resource allocation for multi-server mobile-edge computing networks. *IEEE Transactions on Vehicular Technology*, 2019, 68(1): 856-868
- [36] Wang F, Xu J, Wang X, et al. Joint offloading and computing optimization in wireless powered mobile-edge computing systems. *IEEE Transactions on Wireless Communications*, 2018, 17(3): 1784-1797
- [37] Li M, Cheng N, Gao J, et al. Energy-efficient UAV-assisted mobile edge computing: resource allocation and trajectory optimization. *IEEE Transactions on Vehicular Technology*, 2020, 69(3): 3424-3438



QU Yu-Ben, Ph. D., associate research fellow. His research interests include mobile edge computing and air-ground integrated networks.

QIN Zhen, Ph. D. candidate. Her research interests include UAV communications and edge computing.

MA Jing-Hao, M. S. candidate. His research interests include edge computing and video analytics.

DAI Hai-Peng, Ph. D., associate professor. His research interests are mainly in the areas of Internet of Things, mobile computing, and data mining.

DONG Chao, Ph. D., professor. His current research interests include D2D communications, UAVs swarm networking, and anti-jamming network protocol.

WANG Hai, Ph. D., professor. His research interests include wireless networking, Ad Hoc networking, and vehicular communications technologies.

WU Fan, Ph. D., professor. His research interests include wireless networking and mobile computing, algorithmic game theory and its applications, and privacy preservation.

CHEN Gui-Hai, Ph. D., professor. He has a wide range of research interests with focus on sensor network, high-performance computer architecture, and combinatorics.

Background

The content of this paper belongs to the optimization problem in the domain of mobile edge computing (MEC). MEC has been widely recognized as a promising technique to offer cloud computing service to the network edge directly, which can break through resource-constrained users' limitations while satisfying the computing requirements. The core of MEC is to deploy small edge computing servers at local wireless access points (APs) or cellular base stations (BSs) and then execute end users' computation-intensive tasks at those edge servers after offloading these tasks to them. Because of the dependence of fixed ground infrastructure, the ground infrastructure-based MEC system has its natural limitations, which cannot work in many existing critical applications, such as emergency relief, unmanned area monitoring, and disaster response. Unmanned aerial vehicle (UAV)-assisted MEC has recently received its increasing popularity, which is considered to be a promising technology to address the aforementioned challenges caused by ground MEC systems.

UAV-assisted MEC architecture has been extensively studied in existing works, including UAV-assisted MEC solely

by UAVs, and collaboratively by UAVs and terrestrial APs/BSs. While the MEC services can be deployed on any edge computation server employing the techniques of network functions virtualization (NFV), a critical and essential problem in MEC is service provisioning, that is, how to determine where to place/pre-store which MEC service, to cater to the various service demands. Since most existing popular mobile applications are generally data-intensive and need a nonnegligible amount of related data cached at MEC server including library/database, trained machine learning model, and code, each computation server with relatively limited storage capacity can store only partial services. Similarly, how to reasonably provision such service for UAV-assisted MEC matters, and is more challenging owing to the scarcity of resources at UAVs. The existing works about UAV-assisted MEC seldom consider the problem of MEC service provisioning, while existing studies about service provisioning are about generic MEC systems, which cannot be applied to UAV-assisted MEC due to the necessity of UAV trajectory optimization. To sum up, service provisioning for UAV-assisted MEC needs to be studied.

In this work, we focus on the scenario of MEC by multiple

UAVs and multiple terrestrial BSs collaboratively, and study the problem of service provisioning for air-ground collaborative MEC. Specifically, we study how to determine a service placement strategy to minimize the total energy consumption of users, when placing services at multiple UAV MEC servers and multiple static terrestrial MEC server to collaboratively provide the required computing services. This problem is quite challenging to deal with, since it is a joint optimization of service placement, UAV movement trajectory, task offloading, and computation resource allocation, which is naturally a non-convex mixed integer nonlinear programming (MINLP)

problem, because of coupled integral service placement and task offloading variables, and continuous UAV movement trajectory and computation resource allocation variables. To address this problem, leveraging alternating optimization techniques, we propose a suboptimal solution with convergence guarantee as follows. Specifically, we iteratively solve the integral joint service placement and task offloading subproblem by Branch and Bound (BnB), and UAV movement trajectory subproblem by successive convex approximation (SCA), while the optimal solution of the computation resource allocation subproblem can be efficiently obtained in its closed form.