S→ 計算机科学 COMPUTER SCIENCE

多无人机辅助MEC环境中基于Wardrop路由博弈的计算卸载

汪昕隆, 林兵, 陈星

引用本文

汪昕隆, 林兵, 陈星. 多无人机辅助MEC环境中基于Wardrop路由博弈的计算卸载[J]. 计算机科学, 2024, 51(3): 309-316.

WANG Xinlong, LIN Bing, CHEN Xing. Computation Offloading with Wardrop Routing Game in Multi-UAV-aided MEC Environment [J]. Computer Science, 2024, 51(3): 309-316.

相似文章推荐(请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

基于改进NSGA-III的D2D协同MEC多目标优化研究

Multi-objective Optimization of D2D Collaborative MEC Based on Improved NSGA-III 计算机科学, 2024, 51(3): 280-288. https://doi.org/10.11896/jsjkx.221100250

面向高速行驶车辆的在线任务卸载决策算法

Online Task Offloading Decision Algorithm for High-speed Vehicles 计算机科学, 2024, 51(2): 286-292. https://doi.org/10.11896/jsjkx.221200069

一种基于SCD文件的合并单元高速数据压缩方法

High Speed Data Compression Method of Merge Unit Based on SCD File 计算机科学, 2023, 50(12): 123-129. https://doi.org/10.11896/jsjkx.230700230

基于博弈论的多边缘服务器负载均衡策略

Multi-edge Server Load Balancing Strategy Based on Game Theory 计算机科学, 2023, 50(11A): 221200150-8. https://doi.org/10.11896/jsjkx.221200150

用户公平保障的边缘服务缓存与任务卸载算法

Fairness-aware Service Caching and Task Offloading with Cooperative Mobile Edge Computing 计算机科学, 2023, 50(11A): 230200095-8. https://doi.org/10.11896/jsjkx.230200095



多无人机辅助 MEC 环境中基于 Wardrop 路由博弈的计算卸载

汪昕降¹ 林 兵^{1,2,3} 陈 星^{3,4}

- 1 福建师范大学物理与能源学院 福州 350117
- 2 北京大学信息科学技术学院 北京 100871
- 3 福建省网络计算与智能信息处理重点实验室(福州大学) 福州 350116
- 4 福州大学计算机与大数据学院/软件学院 福州 350108

(qsz20211425@student.fjnu.edu.cn)

摘 要 无人机(Unmanned Aerial Vehicles, UAVs)与多接入边缘计算(Multi-access Edge Computing, MEC)技术的结合突破 了传统地面通信的局限性,已成为解决 MEC 中任务卸载问题的重要手段。由于单无人机可提供的计算资源和能量有限,为了 应对日益扩大的网络规模,考虑了多无人机辅助 MEC 环境中的任务卸载问题。基于问题定义,任务卸载过程可以视为一个在 平行链路上进行的、具有玩家特定延迟函数的 Wardrop 路由博弈,目的是得到均衡状态和最优状态下的卸载策略,并量化分析 两者间的差距。由于均衡解难以计算,因此构造了一个新的势函数,将均衡问题转换成最小化势函数问题。同时使用 Frank-Wolfe 算法最终获得均衡和最优卸载策略。算法在每次迭代中将目标函数线性化,通过求解线性规划得到可行方向,进而沿此 方向在可行域内作一维搜索。仿真实验表明,相比其他基准测试方法,基于平行链路 Wardrop 路由博弈的均衡卸载策略能够 有效降低模型总成本,且与最优卸载策略下总成本的比值约为 1。

关键词:多接入边缘计算;任务卸载;无人机;Wardrop 路由博弈;Frank-Wolfe 算法 中图分类号 TP393

Computation Offloading with Wardrop Routing Game in Multi-UAV-aided MEC Environment

WANG Xinlong¹, LIN Bing^{1,2,3} and CHEN Xing^{3,4}

1 College of Physics and Energy, Fujian Normal University, Fuzhou 350117, China

2 School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China

3 Fujian Key Laboratory of Network Computing and Intelligent Information Processing(Fuzhou University), Fuzhou 350116, China

4 College of Computer and Data Science/College of Software, Fuzhou University, Fuzhou 350108, China

Abstract The combination of Unmanned aerial vehicles(UAVs) and multi-access edge computing(MEC) technology breaks the limitations of traditional terrestrial communications, which has become a significant approach to solve the tasks offloading problem in MEC. Due to the limited computing resources and energy that a single UAV can provide, the tasks offloading problem in a multi-UAV-assisted MEC environment is considered to cope with the growing network scale. Based on the problem definition, to obtain the offloading strategies in the equilibrium and optimal states and analyze the gap between them quantitatively, the tasks offloading process can be viewed as a Wardrop routing game on parallel links with player-specific latency functions. Since the equilibrium solution is difficult to compute, a new potential function is introduced to convert the equilibrium and the optimal offloading strategies finally. At each iteration of this algorithm, the objective function is linearized, and the feasible direction is thus obtained by solving the linear programming, along which a one-dimensional search is performed in the feasible domain. Simulation experiments verify that the equilibrium offloading strategy based on the Wardrop routing game on parallel links can effectively reduce the model's total cost compared with other benchmark methods, and the ratio between the total costs caused by the equilibrium and optimal offloading strategies is about 1.

到稿日期:2022-11-26 返修日期:2023-03-29

基金项目:国家自然科学基金(62072108);福建省高校产学合作项目(2022H6024);福建省社科规划项目(FJ2020C046)

This work was supported by the National Natural Science Foundation of China(62072108), University-Industry Cooperation of Fujian Province (2022H6024) and Social Science Planning Project of Fujian Province(FJ2020C046).

Keywords Multi-access edge computing, Tasks offloading, Unmanned aerial vehicles, Wardrop routing game, Frank-Wolfe algorithm

1 引言

随着物联网(IoT)技术的爆炸性发展,以及第六代移动通 信技术(6G)时代即将到来,各种计算密集型和延迟敏感型应 用相继涌现,例如增强现实(AR)、虚拟现实(VR)、智慧城市 和自动驾驶等。然而,由于能耗和计算资源的限制,目前的物 联网设备无法支持这些应用^[1-2]。2016年,欧洲电信标准协 会将移动边缘计算(MEC)中的"M"进行扩展,提出了多接入 边缘计算这一概念。多接入边缘计算立足于用户设备(User Equipment,UE),UEs 除了可以在本地上处理任务,也可以将 任务卸载到附近的边缘云(Edge Clouds,ECs),大大减少了时 延和能耗。尽管 MEC 是帮助资源受限的物联网设备处理这 些应用程序的一个极具吸引力的范式,但 MEC 不适合一些 通信基础设施有限的地区(如偏远地区、山区等)和一些紧急 情况(如抢险救灾、军事演习等)。可以说,通信障碍严重影响 了通信服务的灵活性和效率。

UAVs 为上述问题提供了一种可行的解决方法。具体来 说,UAVs 部署更快、配置更灵活,机动性高且成本低。最重 要的是,受益于高海拔,UAVs 在大多数场景下均可以建立空 地视距(Line of Sight,LoS)无线通信链路,与传统的地面非 视距(Non-line of Sight,NLoS)通信相比,性能有显著的改进。 因此在通信基础设施有限的地区和紧急情况下,UAVs 可以 作为移动通信基站,为地面的 UEs 提供通信服务,从而构筑 地空一体化通信网络^[3]。

目前,一些研究已涉及无人机辅助 MEC 系统中的计算 卸载。这些研究大多关注单无人机辅助边缘云的 MEC 架 构,通过综合考虑系统的成本、时延和能耗等,优化无人机的 飞行轨迹和任务卸载比例。但是随着网络规模的扩大,即用 户设备数量增多、分布更广,单无人机可提供的计算资源和能 量相当有限,因此本文主要研究多无人机和多边缘云协同工 作的 MEC 架构。

从经济学角度来看,UEs 在通信网络中通常是自私且理性的,它们总是尽可能地选择成本最小的链路来传输数据。 这种 UEs 既不服从某个中央控制器也不相互合作的环境可以被看成一个非合作博弈。特别地,由于任务是可分割的,因此可以利用 Wardrop 路由博弈来模拟 UEs 之间的任务数据分配。据我们所知,目前还没有很好的研究将 Wardrop 路由 博弈应用于多无人机辅助 MEC 环境中的任务卸载问题。

综上所述,本文考虑多无人机辅助 MEC 环境中的任务 卸载问题,将 UEs 之间的任务数据分配模拟为一个 Wardrop 路由博弈。我们利用势函数法分析该博弈问题,并用 Frank-Wolfe 算法得到均衡和最优卸载策略。本文的主要贡献 包括:

(1)将多无人机辅助 MEC 环境中的任务卸载过程视为 一个 Wardrop 路由博弈。

(2)构建一个新的势函数,证明该势函数取得最小值时,

博弈处于 Wardrop 均衡状态。

(3)通过大量仿真实验评估所提多无人机辅助 MEC 模型的性能,实验结果表明该模型的均衡卸载策略大大降低了 总成本,缓解了 ECs 的压力,且几乎等同于最优卸载策略的 总成本。

本文第2章回顾了相关工作;第3章描述了系统模型;第 4章在基于 Wardrop 路由博弈的模型上,将问题进行公式化 表述;第5章基于势函数法对博弈进行了分析,并用 Frank-Wolfe 算法求解卸载策略;第6章进行了仿真实验评估与分 析;最后总结全文并展望未来。

2 相关工作

本章主要从两方面展开论述,一方面总结现有工作如何 利用无人机辅助边缘计算,另一方面总结现有工作如何利用 博弈工具解决计算卸载问题。

2.1 无人机辅助边缘计算

无人机辅助通信由于具有优越的灵活性和自主性,已被 认为是一种新兴技术^[4]。Zhan 等^[5]考虑了一个用于物联网 计算卸载的无人机 MEC 系统,该系统具有有限的(或没有) 共同的云/边缘基础设施。该研究联合设计了计算卸载和资 源分配,以及在物联网设备的任务和能源预算约束下,使无人 机的能耗和完成时间最小化的无人机轨迹。Zhang 等^[6]利用 无人机作为计算节点和中继节点来改善无人机辅助 MEC 网 络中的平均用户时延,并以最小化所有用户的平均时延为目 标来制定 UAV-MEC 问题。Zhang 等[7] 提出了一种移动无 人机辅助的流量卸载和保密传输方案,解决了传统蜂窝网络 中边缘用户上行通信链路质量较差且容易遭到窃听的问题。 Sun 等^[8]提出了一种新的无人机辅助边缘计算框架,该框架 联合优化了固定翼无人机的轨迹、CPU 频率以及卸载计划, 以使无人机的能耗降至最低。Yu 等^[9]构建了一种无人机辅 助边缘计算模型,目的是最小化用户终端计算和卸载能耗,以 获得基于无人机托管计算资源的卸载方案。Wang 等^[10]研究 了在复杂的三维环境中无人机辅助的物联网系统,该研究优 化了无人机的轨迹,以有效地从多个物联网地面节点收集 数据。

上述研究的任务卸载策略均使用单无人机来设计增强型 边缘计算系统,研究重点大多是优化飞行轨迹和任务卸载比 例。然而,随着物联网设备数量的增加和分布区域的扩 大^[11],无人机的能耗会急剧增加,因此无人机提供的 QoS 也 难以维持在设备数量少时的水平^[12]。考虑到网络规模在持 续扩大,本文主要研究多无人机和多边缘云协同工作的 MEC 架构,并且卸载策略将同时考虑模型的时延和能耗,以提升模 型的性能。

2.2 博弈论

博弈论作为一种强大的工具,可以对每个成员之间的相 互作用进行建模,分析均衡并以分布式方式解决优化 问题^[13]。Zhou等^[14]提出了一个无人机辅助 MEC 网络,并 将边缘服务提供商和移动用户之间的互动看作 Stackelberg 博弈,最后利用一种基于梯度的动态迭代搜索算法(GDISA) 来获得近似的最优解。Long 等^[15]为解决 MEC 网络中计算 能力不足的移动终端在处理低时延、高可靠应用时产生的高 时延问题,提出了一种基于博弈论的计算卸载与资源分配的 联合优化策略。Apostolopoulos 等^[16]提出了一个新的数据卸 载决策框架,将用户如何获得最佳的任务卸载策略表述为每 个用户满意度的最大化问题,且该问题可被视为用户之间的 非合作博弈。其中用户满意度是从风险意识用户行为的角 度,通过前景理论的效用函数来刻画的。

可以看出,上述研究主要针对均衡的求解,很少量化分析 均衡解与最优解间的差距。1952 年 Wardrop^[17]研究了可以 任意分割交通流的自私路由模型,即 Wardrop 模型,并提出 了交通流的均衡原理。由于任务卸载时可以任意分割,因此 将 Wardrop 模型用于通信网络是合适的。Orda 等^[18]的研究 表明,当所有玩家控制相同的流量,并且有相同的源点和终点 时,均衡是唯一的。Gairing 等^[19]证明了博弈如果在并行链 路网络上进行,且玩家特定的延迟函数是无偏移的线性函数, 则均衡存在。Wardrop 均衡可以通过 Frank-Wolfe 方法^[18]来 计算,最终收敛到一个解。

3 系统模型

如图 1 所示,本文考虑一个多无人机辅助 MEC 模型,包括 n^{UE} 个 UEs、 n^{UAV} 架部署在空中的 UAVs,以及 n^{EC} 个部署 在地面的 ECs。每个 $UE_i \in \{1, 2, \dots, n^{UE}\}$ 需要执行时延敏感 和计算密集型的任务 $H_i = (B_i, \phi_i)$,其中 B_i 表示任务的数据 量大小, ϕ_i 表示任务的计算密集程度(处理每 bit 数据量所需 的 CPU 周期数)。由于 UE 的计算能力有限,因此 UE 选择 将一部分任务卸载到提供计算服务的 UAVs 或 ECs 上执行。 将任务的卸载部分记为 $h_i = (b_i, \phi_i)$,其中 b_i 就是任务需要卸 载的数据量大小。





由于无人机的尺寸、重量以及功率等因素限制,因此提供的计算资源和传输资源有限。相比之下,由地面 MEC 服务器组成的 EC 能够提供更多的计算资源和传输资源。下文也将详细介绍任务卸载的 4 个过程:(1)UE 到 UAV 的地对空(Ground-to-Air,G2A)传输;(2)在 UAV 上进行计算;(3)UE 到 EC 的地对地(Ground-to-Ground,G2G)传输;(4)在 EC 上进行计算。

为了便于阅读,表1列出了本节中主要涉及的符号及相 应的定义。

表1 关键符号及定义

Table 1 Key	symbols	and	definition
-------------	---------	-----	------------

<i>抗</i> 巳	
付 丂	
$n^{\mathrm{UE}}, n^{\mathrm{UAV}}, n^{\mathrm{EC}}$	UE,EC,UAV 的数量
H_i , h_i	UE _i 的全部任务、卸载部分任务
B_i , b_i	任务 H_i 和 h_i 的数据量大小
ϕ_i	任务的计算密集程度
g_0	参考距离 d _{ref} =1m 的信道功率增益
$d_{i,j}$, $d_{i,k}$	UE_i 到 UAV_j 和 EC_k 的距离
α_1 , α_2	UE 到 UAV 和 EC 的路径损耗指数
$g_{i,j}, g_{i,k}$	UE_i 到 UAV_j 和 EC_k 的信道增益
${W}_j$, ${W}_k$	UAV_j 和 EC _k 的信道带宽
P_i^s	UE_i 的发送功率
σ_j^2 , σ_k^2	UAV_j 和 EC_k 处的高斯白噪声功率
$R_{i,j}$, $R_{i,k}$	UE_i 到 UAV_j 和 EC_k 的传输速率
$T^{ m G2A}_{i,j}$, $T^{ m G2G}_{i,k}$	UE_i 到 UAV_j 和 EC_k 的传输时延
P^r_j , P^r_k	UAV_j 和 EC_k 的接收功率
$E_{i,j}^{\mathrm{G2A}}$, $E_{i,k}^{\mathrm{G2G}}$	UE_i 到 UAV_j 和 EC_k 的传输能耗
$f_{i,j}$, $f_{i,k}$	UAV _j 和 EC _k 分给 UE _i 的计算能力
$T_{i,j}^{\mathrm{UAV}}$, $T_{i,k}^{\mathrm{EC}}$	UAV_j 和 EC_k 的计算时延
κ_j	UAV_{j} 的有效开关电容系数
$E_{i,j}^{\rm UAV}$	UAV_{j} 的计算能耗
$T_{i,j}, T_{i,k}$	UE_i 到 UAV_j 和 EC_k 的总时延
$E_{i,i}, E_{i,k}$	UE_i 到 UAV;和 EC _k 的总能耗

3.1 UEs 到 UAVs 的 G2A 传输模型

记 UE_i 的坐标为 $p_i = (x_i, y_i, 0), UAV_j$ 的坐标为 $p_i = (x_i, y_i, 0),$ 其中 z_j 表示 UAV_j 的飞行高度,则 UE_i 和 UAV_j 之间的 距离为:

 $d_{i,j} = \| \mathbf{p}_j - \mathbf{p}_i \|_2 = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + z_j^2} \quad (1)$

与文献[14]类似,本文忽略 UEs 之间的干扰,并且假设 无人机的高机动性导致的多普勒频移可以在 UE 处得到完美 补偿。此外,由于无人机的飞行高度较高,UE 和 UAV 之间 的链路主要是 LoS 链路,因此它们之间的信道增益服从自由 空间路径损耗(Free Space Path Loss)模型,可由式(2)给出;

$$g_{i,j} = \frac{g_0}{(d_{i,j})^{s_1}} \tag{2}$$

其中,g₀ 表示参考距离为 1m 时的信道功率增益,α₁ 表示 UE 到 UAV 的路径损耗指数。

因此,UE;到UAV;的数据上行传输速率为:

$$R_{i,j} = W_j \log_2\left(1 + \frac{P_i^s g_{i,j}}{\sigma_j^2}\right) \tag{3}$$

其中, W_i 表示 UAV_i 的信道带宽, P_i^i 表示 UE_i 的发送功率, σ_i^j 表示位于 UAV_i 的加性高斯白噪声功率。

考虑 UE_i将其所有需要卸载的任务全部传输到 UAV_i 上,则 UE_i到 UAV_i的传输时延为:

$$T_{i,j}^{G2A} = \frac{b_i}{R_{i,j}} \tag{4}$$

对应地, UE_i 到 UAV_i 的传输能耗为:

$$E_{i,j}^{G2A} = T_{i,j}^{G2A} (P_i^s + P_j^r) = \frac{b_i}{R_{i,j}} (P_i^s + P_j^r)$$
(5)

3.2 UAV 计算模型

UEs 将数据传输至 UAV; 之后, 就在 UAV; 上进行计算,

由此产生计算延迟和计算能耗。其中计算时延由式(6)给出:

$$T_{i,j}^{\text{UAV}} = \frac{\partial_i \phi_i}{f_{i,j}} \tag{6}$$

其中 f_{i,i}表示 UAV_i分给 UE_i的计算能力,即单位时间的 CPU 周期数。

 $E_{i,j}^{\text{UAV}} = \kappa_i (f_{i,j})^3 T_{i,j}^{\text{UAV}} = \kappa_i (f_{i,j})^2 b_i \varphi_i$ $\tag{7}$

其中, κ_i 表示 UAV_i 的有效开关电容系数,该系数取决于其处理器的芯片结构。

因此,UE;选择将任务卸载至UAV;所需的总时延和总能 耗分别为:

 $T_{i,j} = T_{i,j}^{G2A} + T_{i,j}^{UAV}$ (8)

$$E_{i,i} = E_{i,i}^{\text{G2A}} + E_{i,i}^{\text{UAV}} \tag{9}$$

3.3 UEs 到 ECs 的 G2G 传输模型

同样定义 EC_k 的坐标为 $p_k = (x_k, y_k, 0), 则 UE_i 和 EC_k 之 间的距离为:$

$$d_{i,k} = \| \mathbf{p}_k - \mathbf{p}_i \|_2 = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}$$
(10)
与 G2A 传输模型类似, UE: 和 EC, 之间的信道 增益为.

$$g_{i,k} = \frac{g_0}{(d_{i,k})^{a_2}} \tag{11}$$

其中,α2 表示 UE 到 EC 的路径损耗指数。

UEⁱ到ECⁱ的数据上行传输速率为:

$$R_{i,k} = W_k \log_2 \left(1 + \frac{P_{ig\,i,k}^*}{\sigma_k^2} \right) \tag{12}$$

其中, W_k 表示 EC_k 的信道带宽, σ_k^2 表示位于 EC_k 的加性高斯 白噪声功率。

当 UE_i将其所有需要卸载的任务全部传输到 EC_k上时, UE_i到 EC_k的传输时延为:

$$T_{i,k}^{G2G} = \frac{b_i}{R_{i,k}} \tag{13}$$

对应地,UE;到EC,的传输能耗为:

$$E_{i,k}^{G2G} = T_{i,k}^{G2G} (P_i^s + P_k^r) = \frac{b_i}{R_{i,k}} (P_i^s + P_k^r)$$
(14)

3.4 EC 计算模型

每个 EC 收到来自 UEs 的数据之后,在其拥有的 MEC 服务器上进行计算,产生了计算延迟。与 UAV 相比,EC 可 以获得充足的能量资源,因此我们不考虑 EC 完成任务所需 的计算能耗。EC_k的计算时延为:

$$T_{i,k}^{\text{EC}} = \frac{b_i \phi_i}{f_k} \tag{15}$$

其中, $f_{i,k}$ 表示 EC_k 分配给 UE_i 的计算能力。

因此,UE;选择将任务卸载至EC, 所需的总时延和总能 耗分别为:

$$T_{i,k} = T_{i,k}^{G2G} + T_{i,k}^{EC}$$
(16)

$$E_{i,k} = E_{i,k}^{G2G} \tag{17}$$

4 问题定义

为研究交通规划问题, Wardrop于1952年提出用户均衡 原理与系统最优原理。这是两个不同而又相互联系的重要概 念,前者适用于用户独立自由选择路径的交通环境;后者对应 着统一调度下的交通系统。Wardrop原理同样也适用于通信 网络,假设 UEs 都是自私且理性的,那么它们都会选择对自 己来说延迟(时延、能耗等)最小的路径,经过一段时间的选 择,整个网络就会达到 Wardrop 均衡。

Wardrop 路由博弈所在网络如图 2 所示,网络中有源节 点和汇节点,分别表示任务卸载开始和任务卸载结束。两个 节点之间有 n^{UAV} 条边表示 G2A 链路,以及 n^{EC} 条边表示 G2G 链路,将边集记为 $\varepsilon = \{e_1^{G2A}, e_2^{G2A}, \dots, e_n^{G2A}\} \cup \{e_1^{G2G}, e_2^{G2G}, \dots, e_{e_{EC}}^{G2G}\}$ 。



图 2 并行链路网络 Fig. 2 Parallel links network

每个 UE; 在该网络上路由大小为 b; 的数据。将网络中 所有可选路径表示为一个集合 P。假设每个 UE 可选择的路 径都是相同的,因此每个 UE; 的可选择路径集 P; = P。由于 每个 UE 都有相同的可选路径集,且每个路径只包含一条边, 我们将该博弈称作在并行链路上的博弈。

定义 UE *i* 在路径 $p \in \mathscr{P}_i$ 上分配的数据量为 x_p^i ,它满足: (1) $x_p^i \ge 0$;(2) $b_i = \sum_{p \in \mathscr{P}_i} x_p^i$ 。因此 UE_{*i*} 的一个策略就是分配在 各个可选路径上的数据量,记为 $\mathbf{x}^i = (x_p^i)_{p \in \mathscr{P}_i}$ 。定义博弈的 一个策略组合为 $\mathbf{x} = (\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^{n^{UE}}),$ 定义策略空间(即所有 可行的策略组合组成的集合)为 \mathscr{D}_o 此外,定义 UE *i* 在边 *e* 上分配的数据量为 $x_e^i = \sum_{e \in p, p \in \mathscr{P}_i} x_p^i$,则边 *e* 上的总流量为 $\bar{x}_e =$ $\sum_{i=1}^{n^{UE}} x_e^i = \sum_{i=1}^{n^{UE}} \sum_{e \in p, p \in \mathscr{P}_i} x_p^i$ 。由于并行链路网络中每条路径只包含 一条边,因此 $e \in p, p \in \mathscr{P}_i$ 时,有 $x_e^i = x_p^i$ 。

本文中延迟函数是玩家特定的(Player-specific),即同一条边上,不同玩家的延迟函数可能不同。我们将上述网络中每条边上的延迟函数 $L_e^i: \mathbb{R}_+ \rightarrow \mathbb{R}_+$ 定义为:

$$L_{\epsilon}^{i}(\bar{x}_{\epsilon}) = \begin{cases} \frac{a_{\epsilon}^{i}}{c_{\epsilon} - \bar{x}_{\epsilon}}, & \bar{x}_{\epsilon} < c_{\epsilon} \\ \\ \infty, & \bar{x}_{\epsilon} \ge c_{\epsilon} \end{cases}$$
(18)

其中

$$a_{e}^{i} = \begin{cases} \omega_{1} T_{i,j} + \omega_{2} E_{i,j}, & e \ \mathcal{H} U E_{i} - U A V_{j} \ \mathfrak{H} \mathfrak{B} \\ \omega_{1} T_{i,k} + \omega_{2} E_{i,k}, & e \ \mathcal{H} U E_{i} - E C_{k} \ \mathfrak{H} \mathfrak{B} \end{cases}$$
(19)

这种形式的延迟函数被广泛用于计算机通信网络中链路 行为的建模^[20-21]。式(18)中 c_e 表示通信链路的容量,当链路 上的总数据量超过链路容量时,则延迟函数的取值为无穷大。 容易证明,该定义保证了延迟函数的不减性、连续性。式(19) 中 ω_1 和 ω_2 分别为能耗和延迟的权重,它们指示了两者之间 的重要性。 ω_1 越大表示考虑的是时延敏感的案例; ω_2 越大表 示考虑的场景更偏向于节能。

 UE_i 在边 e 上所产生的成本定义为 $C_e^i(\mathbf{x}) = x_e^i L_e^i \overline{x}_e^i$, 在所有边上产生的总成本定义为 $C^i(\mathbf{x}) = \sum_{e \in \mathbf{x}} C_e^i(\mathbf{x}) =$ $\sum_{e \in E} x_e^i L_e^i(\bar{x}_e) \circ 定义模型的总成本为所有 UEs 的总成本之和,$ $记为 SC(x) = \sum_{i=1}^{n^{\text{UE}}} C^i(x) = \sum_{i=1}^{n^{\text{UE}}} \sum_{e \in F} x_e^i L_e^i \bar{x}_e) \circ$

定义 1(Wardrop 均衡解) 在 Wardrop 路由博弈中,如 果对任意 UE_i ,并且对任意 $p,q \in \mathcal{P}_i$,其中 $x_i^i > 0$,均有

 $\sum_{e \in p} L_e^i \bar{x}_e) \leqslant \sum_{e \in q} L_e^i \bar{x}_e)$

则策略组合 $x \in Wardrop$ 均衡解。

容易得到,由于每个 UE 都不断自私地选择当前延迟函数值最小的路径进行数据量的分配,因此系统处于 Wardrop 均衡状态时,对于每个 UE,其所有分配数据量不为 0 的路径上都具有相等的延迟。

定义2(系统最优解) 在 Wardrop 路由博弈中,如果策 略组合 x* 是

min $SC(\mathbf{x}^{i}, \mathbf{x}^{-i})$ s. t. $\sum_{p \in i} x_{p}^{i} = b_{i}, \forall i \in \{1, 2, \dots, n^{\text{UE}}\}$ $x_{p}^{i} \ge 0, p \in i, \forall i \in \{1, 2, \dots, n^{\text{UE}}\}$

的最优解,则策略组合 x* 是系统最优解。

定义 3(Price of Anarchy, PoA) 在路由博弈中, 若x和 x^* 分别是博弈的均衡解和最优解,则 POA 定义为:

$$PoA = \frac{SC(\mathbf{x})}{SC(\mathbf{x}^*)}$$

在一些文章中, PoA 也被称为调和率。

因此,博弈的均衡问题可以表示为:

$$P1: \sum_{e \in p} L_{e}^{i} \overline{x}_{e}) \leqslant \sum_{e \in q} L_{e}^{i} \overline{x}_{e}), \forall p, q \in \mathcal{P}_{i}, x_{p}^{i} > 0$$

s. t. C1:
$$\sum_{p \in \mathcal{P}_{i}} x_{p}^{i} = b_{i}, \forall i \in \{1, 2, \cdots, n^{\text{UE}}\}$$
 (20)

C2: $x_p^i \geq 0$, $p \in \mathcal{P}_i$, $\forall i \in \{1, 2, \cdots, n^{\text{UE}}\}$

博弈的最优化问题可以表示为:

P2:min $SC(\mathbf{x}^i, \mathbf{x}^{-i})$

s. t. C1:
$$\sum_{p \in \mathscr{P}_i} x_p^i = b_i$$
, $\forall i \in \{1, 2, \cdots, n^{\mathrm{UE}}\}$ (21)

C2: $x_p^i \geq 0, p \in \mathcal{P}_i, \forall i \in \{1, 2, \cdots, n^{\text{UE}}\}$

其中,式(20)和式(21)中的约束条件 C1 保证 UE;分配的数据 量之和等于任务需要卸载的数据量,不等式约束条件 C2 确 保 UE;分配的数据量是非负的。

5 博弈分析及计算

5.1 均衡问题的分析

本节将针对具有玩家特定的延迟函数、且在平行链路上 进行的路由博弈,计算 Wardrop 均衡解。我们采用势函数法 将均衡问题转换为最优化问题。这种方法的思想是用一个实 值函数定义博弈的结果,而博弈处于均衡状态正好对应势函 数取得最优化的结果。

对于延迟函数为所示的路由博弈,以及策略组合 x,我们 定义以下函数:

$$\Psi(\mathbf{x}) = \sum_{i=1}^{n^{out}} \sum_{e \in \varepsilon} x_e^i \ln a_e^i + \sum_{e \in \varepsilon} (c_e - \bar{x}_e) \ln(c_e - \bar{x}_e)$$
(22)

定理1(Wardrop 均衡的势函数) 策略组合 *x* 是 Wardrop 均衡解,当且仅当它是相应势函数 Ψ 的全局最小值。

证明:利用 Karush-Kuhn-Tucker 定理,当函数(20)取得 最小值时,等价于对任意 UE_i,存在一个拉格朗日乘子 λ_i, 使得对任意 $p \in \mathcal{P}_i$,

```
若 x_p^i > 0, 则
```

$$\ln a_{e}^{i} - \ln(c_{e} - \bar{x}_{e}) - 1 = \lambda_{i} \Rightarrow \frac{a_{e}^{i}}{c_{e} - \bar{x}_{e}} = e^{\lambda_{i} + i}$$

若 $x_{p}^{i} = 0, 则$
$$\ln a_{e}^{i} - \ln(c_{e} - \bar{x}_{e}) - 1 \ge \lambda_{i} \Rightarrow \frac{a_{e}^{i}}{c_{e} - \bar{x}_{e}} \ge e^{\lambda_{i} + i}$$

可以看出,当 UE_i 在路径p上分配的数据量 x_p^i 不为零时,链路的延迟函数值均为 e^{λ_i+1} ,符合 wardrop 均衡的定义。因此,Wardrop 博弈均衡问题可以转化成下述最小化问题:

P1:
$$\min_{x \in \mathscr{X}} \Psi(\mathbf{x}^{i}, \mathbf{x}^{-i})$$

s. t. C1: $\sum_{p \in \mathscr{P}_{i}} x_{p}^{i} = b_{i}$, $\forall i \in \{1, 2, \cdots, n^{\text{UE}}\}$ (23)
C2: $x_{p}^{i} \ge 0, p \in \mathscr{P}_{i}, \forall i \in \{1, 2, \cdots, n^{\text{UE}}\}$

5.2 均衡解和最优解的计算

Frank 和 Wolfe 于 1956 年提出了一种适合求解的形如 式(24)的非线性规划问题的算法。该算法的基本思想是,首 先给定初始可行解 $\mathbf{x}^{(0)}$,在算法每次迭代中,将目标函数在可 行解 $\mathbf{x}^{(k)}$ 处一阶泰勒展开,使问题转化为线性规划问题,并求 得最优解 $\mathbf{y}^{(k)}$;然后构造可行下降方向 $\mathbf{y}^{(k)} - \mathbf{x}^{(k)}$,在可行域范 围内进行一维搜索找到下一个可行解 $\mathbf{x}^{(k+1)}$,直到满足迭代 停止条件,即找到问题的最优解。因此,本节利用 Frank-Wolfe 算法求解问题,具体细节如算法 1 所示。

min
$$f(\mathbf{x})$$

s. t. $A\mathbf{x} = \mathbf{b}$ (24)
 $\mathbf{x} \ge 0$

算法1 Frank-Wolfe 算法

/*Frank-Wolfe 算法计算 Wardrop 博弈均衡解*/

输入: $(n^{UE}, n^{UAV}, n^{EC}, \epsilon)$

输出:均衡解 x

```
1. 初始化:迭代次数 k←0
```

- 2. 给定初始可行解 x⁽⁰⁾
- 3. while TRUE do
- 4. 求解公式的线性规划问题,得到最优解 y^(k)
- 5. if $|\nabla \Psi(\mathbf{x}^{(k)})^{\mathrm{T}}(\mathbf{y}^{(k)}-\mathbf{x}^{(k)})| \leq \epsilon$ then
- 6. break
- 7. else
- 8. 采用二分法等方法求解公式,得到最优解 α^(k)
- 9. end if

10. 更新可行解 $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)} (\mathbf{y}^{(k)} - \mathbf{x}^{(k)})$

11. 更新迭代次数 k=k+1

```
12. end while
```

13. return ${\bf x}$

s.

下面给出算法的具体步骤。

第1步 首先初始化迭代次数 k=0,并给定满足 C1 和 C2 的初始可行解 $\mathbf{x}^{(0)}$,以及允许误差 ϵ 。

第2步 求解线性规划问题: min $\nabla \Psi(\mathbf{x}^{(k)})^{\mathrm{T}} \mathbf{y}$

t.
$$Ax = b$$
 (25)
 $x \ge 0$

其中, $A = \begin{pmatrix} 1 \\ & \ddots \\ & 1 \end{pmatrix}$,1 表示全为1的 $n^{UAV} + n^{EC}$ 维向量, $b = (b^1, b^2, \cdots, b^{n^{UE}})$ 。

第3步 若 $|\nabla \Psi (\mathbf{x}^{(k)})^{T} (\mathbf{y}^{(k)} - \mathbf{x}^{(k)})| \leq \epsilon$,则停止计算, 得到点 $\mathbf{x}^{(k)}$,否则进行步骤(4)。

第4步 从 $x^{(k)}$ 出发,沿 $y^{(k)} - x^{(k)}$ 方向作一维搜索:

 $\min_{\alpha \in \mathcal{A}} \Psi(\mathbf{x}^{(k)} + \alpha(\mathbf{y}^{(k)} - \mathbf{x}^{(k)}))$ (26)

得到最佳步长 a^(k)。

第5步 令 $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)} (\mathbf{y}^{(k)} - \mathbf{x}^{(k)})$,然后置 k = k+1,返回步骤 2。

上述 Frank-Wolfe 算法仍可以求解最优解问题。具体细节如算法 2 所示。

算法2 Frank-Wolfe 算法

/ * Frank-Wolfe 算法计算 Wardrop 博弈最优解 * /

输入: $(n^{UE}, n^{UAV}, n^{EC}, \epsilon)$

输出:最优解 x

1. 初始化:迭代次数 k←0

2. 给定初始可行解 x⁽⁰⁾

3. while TRUE do

 求解线性规划问题 min ∇SC(x^(k))^Tx s. t. Ax=b

x≥0

得到最优解 y^(k)

```
5. if |\nabla SC(\mathbf{x}^{(k)})^T(\mathbf{y}^{(k)}-\mathbf{x}^{(k)})| \leq_{\epsilon} then
```

```
6. break
```

7. else

8. 采用二分法等方法求解

 $\min_{k \in \mathcal{SC}} SC(\mathbf{x}^{(k)} + \alpha(\mathbf{y}^{(k)} - \mathbf{x}^{(k)}))$

得到最优解 α^(k)

```
9. end if
```

```
10. 更新可行解 \mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)} (\mathbf{y}^{(k)} - \mathbf{x}^{(k)})
```

```
11. 更新迭代次数 k=k+1
```

12. end while

13. return \mathbf{x}

最后讨论 Frank-Wolfe 算法的时间复杂度。可以看出, 算法每次迭代都需要求解线性规划问题以及利用一维搜索方 法寻找最优步长。这里我们假设求解线性规划问题所需的时 间复杂度为 $\mathcal{O}(\Delta)$,一维搜索算法所需的时间复杂度为 $\mathcal{O}(\Theta)$, 确定收敛到最优解所需的迭代次数记为 It,则该算法总体时 间复杂度为 $\mathcal{O}(It \cdot (\Delta + \Theta))$ 。一般线性规划问题可以在多项 式时间内解决,因此时间复杂度为 $\mathcal{O}([n^{UE} \cdot (n^{UAV} + n^{EC})]^{k})$ 。 本文采用二分法进行一维搜索,假设将[0,1]区间切分成 γ 等 份,即精度为 $1/\gamma$,则所需的时间复杂度为 $\mathcal{O}(\log_2(\gamma))$ 。综上, 该算法总体时间复杂度可表示为 $\mathcal{O}(It \cdot ([n^{UE} \cdot (n^{UAV} + n^{EC})]^{k})$

6 仿真结果及讨论

本章进行了仿真实验,以评估我们所研究的模型及算法

的性能。实验所考虑的多无人机辅助 MEC 模型,由 2 个固定位置的 ECs,2 个 UAVs 以及 30 个 UEs 构成。假设模型所在的平面空间面积为 400 × 400 m², ECs 的坐标分别为(0, 200)和(400,200), UAVs 的坐标分别为(150,250,50)和(250,150,100), UEs 随机分布在两个热点地区,如图 3 所示。



图 3 UEs, UAVs 和 ECs 的二维位置图 Fig. 3 2D-locations of UEs, ECs and UAVs

每个 UE 卸载任务的数据量大小 b_i 在[10,50]MB 之间随机生成,任务的计算密集程度 $\phi_i = 10^3$ cycles/bit。其他主要的仿真参数总结如表 2 所列。

表 2 系统参数设置

Table 2	Table 2 System parameters	
参数	值	
b_i	[10,50] MB	
ϕ_i	1000 cycles/bit	
g_0	$-50 \mathrm{dB}$	
α_1 , α_2	2,2	
${m W}_j$, ${m W}_k$	10 MHz,1 MHz	
P_i^s	0.1W	
σ_{j}^{2} , σ_{k}^{2}	-100 dBm	
P_j^r , P_k^r	0.1W,0.1W	
$f_{i,j}$, $f_{i,k}$	0.2G cycles/s,1G cycles/s	
κ_{j}	10-28	

此外,我们还将所提算法与以下4种基准方法进行性能 比较。

(1)随机(Random,Rd):UEs 的任务数据量分配策略是随机生成的。

(2) 无 UAV-MEC 服务器(No UAV-MEC server, NU): UEs 将所有任务均卸载到 ECs 上, UAVs 不提供卸载服务。

(3)均衡(Equilibrium, Eq):均衡状态下任务的数据量分配。

(4)最优(Optimal, Opt):最优状态下任务的数据量分配。

6.1 仿真结果及收敛性能

图 4(a)和图 4(b)分别给出了模型分别处于均衡状态和 最优状态下的任务拆分比率。根据图 3 所示,前 10 个 UEs 所在的热点区域更靠近 EC1 和 UAV1,后 20 个 UEs 所在的 热点区域距离 EC2 和 UAV2 更近。而 UEs 更愿意选择卸载 至离自己较近的服务器,以减少卸载所需的时延和能耗。因 此从图 4(a)、图 4(b)可以看出,模型无论是处于均衡状态还 是最优状态,前 10 个 UEs 几乎全将数据卸载至 EC1 或 UAV1,后 20 个 UEs 中大部分将数据卸载至 EC2 或 UAV2。 但是将数据全部传输至某台服务器容易造成拥塞,因此后 20 个 UEs 中也有少部分将数据传输至 UAV1 上。



图 4 UEs 均衡和最优的任务拆分比率

Fig. 4 Equilibrium and optimal tasks splitting ratios of UEs

图 5 给出了 Frank-Wolfe 算法在求解均衡解和最优解时 模型总成本的迭代过程。首先,可以看出 Frank-Wolfe 算法 求解上述问题时收敛速度快,大约迭代5次以后,模型的总成 本就趋于稳定并接近最终解。其次,均衡解和最优解最后都 几乎收敛于同一数值。实际中,本实验计算出来的均衡解和 最优解下的总成本分别约为 70.73 和 70.65,由此算得该模 型下博弈的 PoA 约为1。这是因为,如果博弈在平行链路网 络上进行,那么均衡状态与最优状态下的总成本几乎相等。





6.2 参数影响

本小节将评估不同参数对模型总成本的影响。

首先研究 UAV; 分配给 UE; 不同计算能力下所产生的 总成本,实验将 fi,i的范围设置为[0.1,0.3]。如图 6 所示,随 着计算能力提高,均衡解与最优解下产生的总成本均降低。 这是由于计算能力提高,意味着数据在 UAV 搭载的服务器 上的执行时间会减少,但是产生的能耗会增加。由于实验是 时延敏感案例,因此对应的G2A链路上产生的时延和能耗加 权和总体上是减少的。因此对于 UEs, 它们也更愿意将数据 传输至 UAVs。

其次,研究 G2A 通信链路容量和模型总成本之间的关 系,实验将 c。的范围设置为「1500,2500],并假设不同 UAV

都有相同的通信链路容量。如图 7 所示,随着 G2A 通信链路 容量的增加,均衡解与最优解下的总成本均降低。由第4节 延迟函数的定义可知,通信链路上的总数据量越接近通信链 路容量,产生的延迟越接近于无穷大。因此通信链路容量增 大,可承受的数据量增多,延迟函数值就会降低。故 UEs 同 样将更多比例的数据传输至 UAVs。



图 6 UAV; 分配给 UE; 不同计算能力下的模型总成本

Model's total cost with different computation capacities Fig 6

that UAV_i assigns to UE_i



图 7 不同 G2A 链路容量下的模型总成本

Fig. 7 Model's total cost with different capacities of G2A links

最后,研究不同数量的 UEs 所产生的模型总成本。与之 前的实验不同的是,UEs的位置将在整个平面空间内随机生 成,而不是在热点区域附近随机生成。

图 8 给出了在不同基准方法下,不同数量 UEs 所产生的 模型总成本。可以看出,随着 UEs 数量的增加,所产生的总 成本也逐渐增加。其次,与6.1节实验的结论类似,由于博弈 在平行链路网络上进行,因此均衡状态和最优状态下产生的 总成本几乎一致。此外,如果 UAVs 不提供卸载服务,UEs 将任务全部卸载至 ECs,即使数据量在 ECs 之间是最优分 配,所产生的总成本也高于 Eq 场景和 Opt 场景。当 UEs 的 数量达到100时,所产生的总成本甚至高于 Rd 场景。这说 明本文提出的多无人机辅助 MEC 模型确实能够有效减少总 成本,从而提升通信网络的性能。



图 8 不同 UEs 数量下的模型总成本 Fig. 8 Model's total cost with different number of UEs 结束语

考虑了模型的时延和能耗,目的是优化任务的卸载策略。我 们将 UEs 的任务卸载过程视为 Wardrop 路由博弈,以此量化 分析博弈的均衡解和最优解。特别地,本文构建了一个新的 势函数,将均衡问题转化为最小化势函数问题,大大简化了均 衡的计算。我们比较了不同场景下的基准测试方法,实验结 果表明,Frank-Wolfe 算法能迅速收敛到近似解。此外,所提 模型性能明显优于传统无 UAV 的 MEC 模型。最后我们发 现,系统处于均衡状态与最优状态下所产生的模型总成本相 差不大,这表明平行链路网络下,UEs 的自私行为对模型的 总成本造成的影响很小。

尽管本文利用 Wardrop 路由博弈来模拟 UEs 的数据分 配过程,得到的均衡解下的总成本与最优解下的总成本相当, 但是面对 UEs 非对称(Asymmetric)场景(即每个 UE 可卸载 的服务器不是相同的)或者延迟函数更复杂甚至非凸的场景, 本文对均衡的分析都将不再适用。未来的工作中,我们将进 一步研究 UEs 非对称和延迟函数非凸的路由博弈。

参考文献

- [1] JIANG K, SUN C, ZHOU H, et al. Intelligence-Empowered Mobile Edge Computing: Framework, Issues, Implementation, and Outlook[J]. IEEE Network, 2021, 35(5):74-82.
- [2] MA Z,NUERMAIMAITI N,ZHANG H,et al. Deployment Model and Performance Analysis of Clustered D2D Caching Networks Under Cluster-Centric Caching Strategy[J]. IEEE Transactions on Communications, 2020, 68(8): 4933-4945.
- [3] NGUYEN V,KHANH T T,VAN NAM P,et al. Towards Flying Mobile Edge Computing [C] // 2020 34th International Conference on Information Networking(ICOIN). 2020;723-725.
- [4] YANG Z, CHEN M Z, LIU X, et al. AI-Driven UAV-NOMA-MEC in Next Generation Wireless Networks[J]. IEEE Wireless Communications, 2021, 28(5):66-73.
- [5] ZHAN C, HU H, SUI X F, et al. Completion Time and Energy Optimization in the UAV-Enabled Mobile-Edge Computing System[J]. IEEE Internet of Things Journal, 2020, 7 (8): 7808-7822.
- [6] ZHANG L,ANSARI N. Latency-Aware IoT Service Provisioning in UAV-Aided Mobile-Edge Computing Networks [J]. IEEE Internet of Things Journal,2020,7(10):10573-10580.
- [7] ZHANG S W, LIU J J, XU T X. An Unmanned Aerial Vehicle-Assisted Cellular Data Traffic Offloading and Secure Transmission Scheme [J]. Journal of Xi' an Jiaotong University, 2021, 55(2):121-128.
- [8] SUN C,NI W,WANG X. Joint Computation Offloading and Trajectory Planning for UAV-Assisted Edge Computing [J].
 IEEE Transactions on Wireless Communications, 2021, 20(8): 5343-5358.
- [9] YU X Y, ZHU Y, QIU L X, et al. Energy efficient offloading strategy for UAV aided edge computing systems[J]. Journal of Systems Engineering and Electronics, 2022, 44(3):1022-1029.
- [10] WANG Y,GAO Z,ZHANG J,et al. Trajectory Design for UAV-Based Internet of Things Data Collection: A Deep Reinforcement Learning Approach [J]. IEEE Internet of Things

Journal, 2022, 9(5): 3899-3912.

- [11] WU P,XIAO F,HUANG H,et al. Load Balance and Trajectory Design in Multi-UAV Aided Large-Scale Wireless Rechargeable Networks[J]. IEEE Transactions on Vehicular Technology, 2020,69(11):13756-13767.
- [12] 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 Trans Cybern, 2020, 50(9): 3984-3997.
- [13] CHEN R,CUI L,WANG M,et al. Joint Computation Offloading, Channel Access and Scheduling Optimization in UAV Swarms: A Game-Theoretic Learning Approach[J]. IEEE Open Journal of the Computer Society, 2021, 2:308-320.
- [14] ZHOU H, WANG Z, MIN G, et al. UAV-Aided Computation Offloading in Mobile-Edge Computing Networks: A Stackelberg Game Approach [J]. IEEE Internet of Things Journal, 2022, 10(8):6622-6633.
- [15] LONG L,LIU Z C,SHI J L,et al. Joint optimization strategy of computation offloading and resource allocation in mobile edge computing[J]. High Technology Letters, 2020, 30(8):765-773.
- [16] APOSTOLOPOULOS P A, FRAGKOS G, TSIROPOULOU E E, et al. Data offloading in UAV-assisted multi-access edge computing systems under resource uncertainty[J]. IEEE Transactions on Mobile Computing, 2021, 22(1):175-190.
- [17] WARDRO P. Theoretical Aspects of Road Traffic Research[J]. Proc. of the Institution of Civil Engineers(Part II):325-362.
- [18] ORDA A, ROM R, SHIMKIN N. Competitive routing in multiuser communication networks[J]. IEEE/ACM Transactions on Networking, 1993, 1(5): 510-521.
- [19] GAIRING M, MONIEN B, TIEMANN K. Routing(un-) splittable flow in games with player-specific linear latency functions
 [J]. Automata, Languages and Programming, Pt 1, 2006, 4051: 501-512.
- [20] BERTSEKAS D, GALLAGER R. Data networks [M]. Athena Scientific, 2021.
- [21] KLEINROCK L. Queueing systems [M]. Wiley Interscience, 1975.



WANG Xinlong, born in 1997, postgraduate. His main research interests include cloud computing technology and computation offloading.



LIN Bing, born in 1986, Ph.D, associate professor, postgraduate supervisor, is a member of CCF(No. 83773M). His main research interests include cloud computing technology and computational intelligence.