



计算资源受限的移动边缘计算服务器收益优化策略

黄冬艳¹, 付中卫², 王波^{2*}

(1. 深圳大学 电子与信息工程学院, 广东 深圳 518060;

2. 认知无线电与信息处理部共建教育部重点实验室(桂林电子科技大学), 广西 桂林 541004)

(* 通信作者电子邮箱 gbluwind@126.com)

摘要: 移动边缘计算(MEC)服务器通过向用户提供计算资源获得收益。对MEC服务器而言,如何在计算资源受限的情况下提高自身收益至关重要,为此提出一种通过优化计算任务执行次序提高MEC服务器收益的策略。首先,将MEC服务器收益最大化问题建模为以任务执行次序为优化变量的优化问题;然后提出了一种基于分支定界法的算法求解任务执行次序。仿真结果表明,采用所提算法获得的MEC服务器平均收益分别比大任务优先(LTF)算法、低延迟任务优先(LLTF)算法和先到先服务(FCFS)算法提高了11%、14%和21%。在保证卸载用户服务质量(QoS)同时,所提策略可以显著提高服务器的收益。

关键词: 移动边缘计算;收益最大化;计算资源受限;计算资源分配;分支定界法

中图分类号: TP391.9 **文献标志码:** A

Revenue maximization strategy for mobile-edge computing server with limited computing resources

HUANG Dongyan¹, FU Zhongwei², WANG Bo^{2*}

(1. College of Electronics and Information Engineering, Shenzhen University, Shenzhen Guangdong 518060, China;

2. Ministry Education Key Laboratory of Cognitive Radio and Information Processing (Guilin University of Electronic Technology), Guilin Guangxi 541004, China)

Abstract: Mobile Edge Computing (MEC) servers receive revenue by leasing computing resources to users. Improving revenue with limited computing resources is critical for MEC servers. Therefore, a strategy of improving MEC server revenue by the optimization of computing task execution order was proposed. Firstly, the revenue maximization problem of servers was modeled to an optimization problem with task execution order as optimization parameter. Then, an algorithm based on the branch and bound approach was proposed to find the optimal task execution order. Simulation results show that the average revenue of MEC server of the proposed algorithm is 11%, 14% and 21% higher than those of Large Task First (LTF) algorithm, Low-Latency Task First (LLTF) algorithm, and First Come First Served (FCFS) algorithm respectively. The proposed strategy can significantly improve the servers' revenue while guaranteeing offloading users' Quality of Service (QoS).

Key words: Mobile Edge Computing (MEC); revenue maximization; limited computing resource; computing resource allocation; branch and bound approach

0 引言

随着物联网和5G移动通信技术的发展,在智能手机、传感器和可穿戴计算设备等移动设备上运行计算密集型和延迟关键型应用已经成为趋势^[1-2]。但由于受到自身计算能力和能量的限制,移动设备通常不具备运行这类应用的能力。

移动边缘计算(Mobile Edge Computing, MEC)^[3]通过将计算任务从移动设备卸载到具有相对丰富计算资源的边缘服务器上执行,实现了在移动设备上运行计算密集型和延迟关键型应用的愿景。与移动云计算不同,MEC服务器通常部署在网络边缘(例如,基站和无线局域网接入点),因此可以避免移动用户和远程云中心之间的长距离数据传输,从而显著降低

延迟和移动用户的能量消耗。因此,MEC是5G网络的关键技术,获得了业界的广泛关注^[4]。

通过优化任务卸载决策、资源分配或任务执行次序实现MEC的吞吐量,端到端延迟或能量效率等性能的提升是MEC研究中的热点。

考虑到频谱资源受限,为了提高MEC系统的吞吐量,文献[5-6]分别提出了相应的接入控制策略、频谱资源和计算资源联合优化算法。另一方面,为了降低移动设备的延迟和能耗,有文献提出通过多用户博弈^[7]、联合优化子载波和功率的分配^[8]等方式实现移动设备延迟最小化,以及结合数据压缩与频谱资源分配以降低移动用户的能耗^[9]。

收稿日期:2019-08-05;修回日期:2019-10-13;录用日期:2019-10-14。

基金项目:广西科技基地和人才专项(桂科AD19110042);广西无线宽带通信与信号处理重点实验室主任基金资助项目(GXKL06160111)。

作者简介:黄冬艳(1984—),女,广西南宁人,博士,主要研究方向:移动边缘计算、区块链;付中卫(1993—),男,河南商丘人,硕士研究生,主要研究方向:移动边缘计算、资源优化;王波(1977—),男,陕西西安人,讲师,博士,主要研究方向:5G移动通信。



但在文献[5-9]的分析中均假设 MEC 服务器具有无限的计算能力。事实上,受到部署场地和成本的制约,MEC 服务器的计算能力相比大型云计算中心是有限的。这导致任务在服务器的处理时间以及任务在 MEC 服务器任务缓存区内的排队延迟不可忽略,特别是在负载密集的网络中。据研究表明,在 5G 场景下,任务在 MEC 服务器的处理时间远大于其上传时间^[10]。以图像识别为例,对于 569 KB 大小的图像,在 4G 网络下的传输时间为 1.24 s,在 5G 网络下的传输时间为 0.001 s,而在 MEC 服务器处理时间为 1.12 s^[10]。可见,在 5G 网络中,MEC 处理时间比上传时间高了 3 个数量级。因此,MEC 面临的挑战从频谱资源和计算资源受限转变为计算资源受限,需要在计算资源受限的情况下,进一步研究 MEC 的性能优化问题^[11-14]。

文献[11]提出了一种延迟最小化的计算任务卸载方案,由移动用户根据 MEC 服务器任务缓存区的状态、本地处理单元的执行状态和传输单元的状态做出卸载决策。文献[12-13]则研究了基于定价的分布式计算任务卸载决策,将 MEC 服务器和移动用户之间的交互建模为 Stackelberg 博弈模型。在该博弈模型中,MEC 服务器依据收益最大化设定服务费。给定服务费后,每个用户依据延迟最小化^[12]或是能耗最小化^[13]自主做出卸载决定。此外,文献[14]通过优化任务执行次序,减小移动用户和 MEC 服务器的加权能耗。

在重业务负载的场景下,由于计算能力有限,为保证卸载任务的 QoS(例如,在线游戏、增强现实(Augmented Reality, AR)、虚拟现实(Virtual Reality, VR)等延迟敏感型应用需要保证延迟需求),MEC 服务器只能进行接入控制,为部分用户提供计算服务。另一方面,为了收回设备部署和维护成本并实现盈利,MEC 服务器非常关注如何利用有限的资源最大化自身的收益。因此,为了确保卸载任务的 QoS,同时最大化自身的收益,服务器必须合理地确定允许哪些任务卸载并确定卸载任务的执行次序。

本文关注于计算资源受限的 MEC 服务器收益优化问题。与文献[12-13]不同,本文研究了存在不同 QoS 需求的多用户 MEC 系统,提出通过优化任务执行次序提高 MEC 服务器的收益。主要贡献如下:1)将 MEC 服务器收益最大化问题建模为以任务执行次序为优化变量的优化问题;2)提出了一种基于分支定界法的排序算法,以获得收益优化的任务执行次序。

1 系统模型

考虑一个由基站和 K 个移动用户组成的多用户 MEC 系统。该系统的每个用户都有一个计算密集型任务,并请求将任务卸载到 MEC 服务器。每个任务都具有严格的截止期限。系统模型如图 1 所示。

此外,本文采用如下假设:

- 1)信道状态信息是已知的;
- 2)信道状态在任务卸载期间保持不变;
- 3)一旦决定将任务卸载到 MEC 服务器,移动用户将不会停止卸载,直到卸载完成。

任务卸载过程如图 2 所示。首先,移动用户 $k(k \in \{1, 2, \dots, K\})$ 向 MEC 服务器发送卸载请求消息。卸载请求消息包括客户端中间件收集的任务信息。收到一批卸载

请求后,MEC 服务器做出卸载决定并将该决定反馈给用户。如果 MEC 服务器同意卸载,那么用户将任务上传并向 MEC 服务器支付相应的费用;否则,用户不需要支付费用。

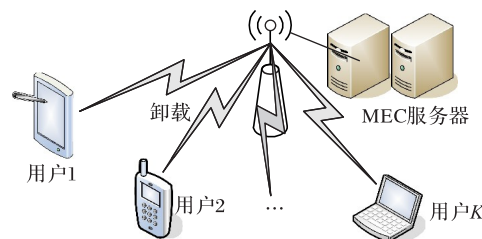


图 1 多用户 MEC 系统

Fig. 1 Multi-user MEC system

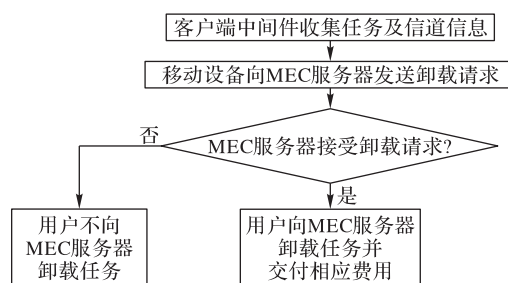


图 2 MEC 系统的任务卸载流程

Fig. 2 Task offloading process in MEC system

2 问题建模及优化算法

考虑一个有多个计算密集型任务同时请求卸载的重业务负载场景。首先,将 MEC 服务器收益最大化问题建模成以任务执行次序为优化变量的最优化问题;然后,提出了一种基于分支定界法的排序算法来寻找该问题的最优解。

2.1 问题建模

定义 MEC 服务器中一个执行次序为 $s = (s(1), s(2), \dots, s(k))$ 。其中 s 是 $1, 2, \dots, K$ 的一种排列, $s(k) \in \{1, 2, \dots, K\}$ 。举例说明,当 $K = 3, s = (2, 1, 3)$,则 $s(1) = 2$,表示第 2 号任务排在次序 s 的第 1 个位置。

若将任务 $s(k)$ 卸载到 MEC 服务器执行,则完成该任务的所需时间包括任务上传时间,在 MEC 服务器的队列等待时间,服务器处理时间和计算结果下载时间。由于计算结果的大小通常远小于计算任务本身,因此可以合理地认为计算结果的下载时间远小于任务上传时间。为简化分析,在接下的分析中主要关注总时间的 3 个主要部分:上传时间、队列等待时间和执行时间。

令 $l_{s(k)}$ (单位:bit) 表示任务 $s(k)$ 的大小, $G_{s(k)}$ (单位:cycle/bit) 表示计算强度, $d_{s(k)}$ (单位:s) 表示任务的截止期限,MEC 服务器的 CPU 时钟频率为 f_c (单位:Hz)。

1) 任务上传时间 $t_{u,s(k)}$ 为:

$$t_{u,s(k)} = l_{s(k)} / r_{s(k)} \quad (1)$$

其中 $r_{s(k)}$ 是传输速率。根据香农定理,可知:

$$r_{s(k)} = B_{s(k)} \log \left(1 + \frac{p_{s(k)} h_{s(k)}}{B_{s(k)} N_0} \right)$$

其中: $B_{s(k)}$ 为分配给移动用户 $s(k)$ 的传输带宽, N_0 为噪声功率谱密度, $h_{s(k)}$ 是介于移动用户 $s(k)$ 和基站之间的信道增益, $p_{s(k)}$ 为传输功率。



2) 任务在 MEC 服务器的处理时间 $t_{c,s(k)}$ 为:

$$t_{c,s(k)} = (l_{s(k)} G_{s(k)}) / f_c \quad (2)$$

3) 队列等待时间 $t_{w,s(k)}$: 记 $Q_{s(k)}$ 为任务 $s(k)$ 的完成时间。

若任务 $s(k)$ 上传至服务器的时刻 $\sum_{m=1}^k t_{u,s(m)}$ 大于前一个任务 $s(k-1)$ 的完成时间 $Q_{s(k-1)}$, 那么服务器执行该任务的起始时刻为 $\sum_{m=1}^k t_{u,s(m)}$; 否则, 该任务在队列等待的时间为零, 起始执行时刻为 $Q_{s(k-1)}$, 即该任务的起始执行时刻为上一个任务的完成时刻。可得队列等待时间为:

$$t_{w,s(k)} = \begin{cases} \left(\sum_{m=1}^k t_{u,s(m)} \right) - Q_{s(k-1)}, & \sum_{m=1}^k t_{u,s(m)} \geq Q_{s(k-1)} \\ 0, & \sum_{m=1}^k t_{u,s(m)} < Q_{s(k-1)} \end{cases} \quad (3)$$

因此, 任务 $s(k)$ 的完成时间可以表示为:

$$Q_{s(k)} = \max \left(Q_{s(k-1)}, \sum_{m=1}^k t_{u,s(m)} \right) + t_{c,s(k)} \quad (4)$$

当任务上传时间远小于任务执行时间^[10], 任务 $s(k)$ 的完成时间可以进一步简化为:

$$Q_{s(k)} = Q_{s(k-1)} + t_{c,s(k)} \quad (5)$$

MEC 服务器按照以下原则决定是否接受用户的卸载请求: 如果任务 $s(k)$ 可以在给定的截止期限之前完成, 即 $Q_{s(k)} \leq d_{s(k)}$, MEC 服务器将接受任务; 否则, MEC 服务器将拒绝任务 $s(k)$ 的卸载请求。MEC 服务器主要处理计算密集型 and 延迟关键型应用的计算任务, 此类应用包括在线游戏、AR、VR、图像识别等, 对延迟非常敏感。这样的原则确保了卸载任务可以在截止期限之前完成, 确保了卸载用户的 QoS。

若 MEC 服务器接受任务 $s(k)$ 将获得 $\mu l_{s(k)} G_{s(k)}$ 的收益, 其中 μ 是每单位 CPU 周期的收益; 若 MEC 服务器拒绝任务 $s(k)$ 的卸载请求则不收取该任务费用。

根据上述决策原则, 可得 MEC 服务器的收益为:

$$U(s) = \mu \sum_{k=1}^K l_{s(k)} G_{s(k)} \sigma(d_{s(k)} - Q_{s(k)}) \quad (6)$$

其中 $\sigma(\cdot)$ 函数被定义如下:

$$\sigma(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (7)$$

因此, MEC 服务器收益的最优化可以表示为如下形式的优化问题:

$$P1: \max_s U(s) \quad (8)$$

2.2 基于分支界定法的任务执行排序算法

根据 MEC 服务器的卸载决策原则, 无法在给定的截止期限之前完成的任务实际上不会被卸载至服务器。同时, MEC 服务器也不能获得用户为该任务所支付的费用。为了实现收益的最大化, MEC 服务器应尽可能减少因拒绝任务造成的损失。每个可能的任务执行次序都具有唯一的目标函数值 (MEC 服务器的收益)。因此, 问题 P1 是 NPC 问题。

注意到任务的上传与处理可以部分并行进行, 即在处理一个任务的同时可上传另一个任务, 且任务的上传时间远小于 MEC 服务器的处理时间^[10], 因此, 采用依次上传任务的方

案, 可以在处理第一个上传任务的同时上传后续任务, 提高处理效率。计算出所有任务上传完毕时刻未开始处理的任务集合, 对未开始处理的任务进行排序以确定收益最优的任务执行次序。

对上传完毕的 n 个任务进行排序, 采用穷举法和动态规划方法分别需搜索 $n!$ 次和 2^n 次可以求解出最优解。显然当问题规模较大时, 这两种方法的复杂度很高。采用分支定界法, 通过考虑任务执行次序的优先关系, 可以减少分支, 提高搜索速度, 在最坏情况下需要搜索 2^{n-1} 次可求解出最优解。因此, 提出了一种基于分支定界法^[15]的算法来求解问题 P1, 旨在以较低的计算复杂度寻找 MEC 服务器收益优化的任务执行次序。

所提算法流程如下:

1) 将给定任务集合 A 中的任务按截止期限由小到大进行排序, 得到初始任务执行次序 $s_0 = \{s_0(1), s_0(2), \dots, s_0(K)\}$ 。

2) 假设按照次序 s_0 依次上传所有任务 (即, 在同一时刻仅有一个任务上传)。记最后一个任务的上传完成时刻为 t_l , 有

$$t_l = \sum_{k=1}^K t_{u,s_0(k)}$$

3) 假设在 t_l 时刻前, MEC 服务器按照次序 s_0 依次处理上传完毕的任务。记任务子集 A_1 为在 t_l 时刻之前已被处理的任务 (包括 t_l 时刻正在处理的任务) 的集合。子集 A_1 的执行次序为 $s_1 = (s_0(1), s_0(2), \dots, s_0(|A_1|))$, 其中 $|A_1|$ 为集合 A_1 的势, 等于 A_1 中元素的个数。

4) 记任务子集 A_2 为在 t_l 时刻之前未被处理的任务集合, 有 $A_2 = A - A_1$ 。运行分支定界法对子集 A_2 中的任务进行排序, 得到关于该子集的最优执行次序 s_2^* 。

5) 得到最终的任务执行次序 $s = \{s_1, s_2^*\}$ 。

举例说明: 给定任务集合 $A = \{1, 2, 3, 4, 5\}$ 。初始任务执行次序 $s_0 = \{5, 3, 2, 1, 4\}$, 表示第 5 号任务首先执行, 依此类推。根据次序 $s_0 = \{5, 3, 2, 1, 4\}$ 计算时刻 t_l , 得到任务子集 $A_1 = \{5\}$ 与相应的排序 $s_1 = \{5\}$ 。同时得到任务子集 $A_2 = \{1, 2, 3, 4\}$ 。运行分支定界法对 A_2 中的任务进行排序, 得到该子集的最优任务排序 $s_2^* = \{2, 4, 3\}$ 。合并 s_1 和 s_2^* , 得到任务集合 A 的任务执行次序 $s = \{5, 2, 4, 3\}$ 。根据排序结果 s , 任务 1 的卸载请求被服务器拒绝, 集合 A 中其余任务按照该次序上传, 同时 MEC 服务器根据该次序处理计算任务。至此, MEC 服务器做出卸载决策并确定任务执行次序, 算法结束。

2.3 算法复杂度分析

所提算法的复杂度主要在于步骤 4) 中运行分支定界法对子集 A_2 中的任务进行排序。根据分支定界法^[15]的计算复杂度分析, 给定 n 个任务, 当任务之间不存在任何优先关系时, 所有分支均不能减去。在这种最坏情况下, 在第 i 层中必须进行分支的节点个数为 $C_{n-1}^i (i = 0, 1, \dots, n-1)$, 所以分支总数是 $\sum_{i=0}^{n-1} C_{n-1}^i = 2^{n-1}$ 。因此, 当 $K > |A_1|$ 时, 子集 A_2 中的任务个数为 $K - |A_1|$, 所提算法在最坏情况下需要搜



索 $2^{K-|A_i|-1}$ 次。

3 仿真结果

本章通过仿真验证所提算法的性能。仿真设定参照 5G 环境设定^[10,12]。仿真设定每个任务的大小均匀分布在 100~500 Kbit, 计算强度均匀分布在 1 000~2 000 cycle/bit, 截止期限均匀分布在 30~150 ms。此外, 可用带宽 $B = 20$ MHz, 信道增益在 $[-50, -30]$ dBm 范围内均匀分布, 用户的传输功率设置为 200 mW, 噪声功率谱密度为 -174 dBm/Hz, MEC 服务器单位收益为每 CPU 周期 1×10^{-8} 元。

首先比较了本文算法 (Proposed algorithm)、低延迟任务优先 (Low-Latency Task First, LLTF) 算法、大任务优先 (Large Task First, LTF) 算法和先到先服务 (First Come First Served, FCFS) 算法的平均收益。具体而言, LLTF 算法和 LTF 算法分别优先考虑具有更高延迟要求和更高计算资源要求的任务, FCFS 算法则优先考虑上传时间最小的任务。在相同的仿真参数下, 独立运行所提算法与对比算法各 10 000 次并记录每种算法的平均收益。然后, 将所提算法的平均收益与对比算法的平均收益进行比较。

如图 3 所示: 1) 所提算法的平均收益高于其他算法的平均收益。随着移动用户数量的增加, 所提算法优势变得更加明显。给定 MEC 服务器的 CPU 频率 $f_c = 10$ GHz, 当移动用户数 $K = 24$ 时, 所提算法的平均收益分别比 LTF、LLTF 和 FCFS 高 11%、14% 和 21%。2) 随着移动用户数 K 的增加, 每种算法的平均收益均趋于稳定。这是因为在工作负载重的网络中, f_c 成为收益增加的瓶颈。

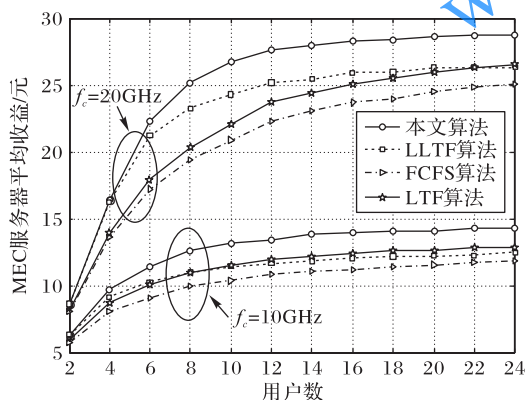


图 3 不同算法的 MEC 服务器平均收益

Fig. 3 Average revenue of MEC server in different algorithms

本文还比较了不同算法的任务完成率, 即 MEC 服务器能够接受的任务数占任务总数的百分比。任务完成率体现了 MEC 服务器容纳用户的能力。任务完成率越高意味着可以满足更多用户的需求。如图 4 所示, 当用户数较少时, 所提算法的平均任务完成率高于 LTF 和 FCFS, 并且该算法的平均任务完成率接近 LLTF。随着用户数量的增加, 当 $f_c = 20$ GHz, $K = 24$ 时, 所提算法的平均完成率低于 LLTF 约 4%, 比 FCFS 低约 3%, 高于 LTF。采用所提算法可以完成近 32% 的任务, 但采用 LTF 仅完成 18% 的任务。

从图 3 和图 4 中, 还可观察到: 1) 随着用户数量的增加, LLTF 和 FCFS 具有较高的完成率但是这两种算法的平均收益均低于所提算法。这意味着收益并不完全等价于已完成任务的数量。2) 所提算法的平均收益高于其他对比算法; 同时, 该算法的任务平均完成率略低于 LLTF 与 FCFS。因此, 所提算法在优化收益同时也很好地兼顾容纳用户的能力。

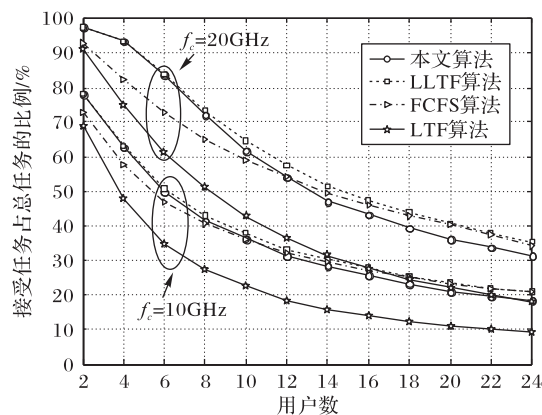


图 4 不同算法的平均任务完成率

Fig. 4 Average task completion rate of different algorithms

4 结语

本文研究了计算资源受限的 MEC 服务器收益优化问题。以最大化 MEC 服务器收益为优化目标, 提出了一种基于分支定界法的算法, 以获得最优的接入策略和任务执行次序。仿真结果表明, 在重负载网络中, 该算法能够有效提高 MEC 服务器的平均收益。本文仅讨论了每单位 CPU 周期的价格固定的情况, 未来拟在价格可更改的场景下进一步研究 MEC 服务器收益优化问题。

参考文献 (References)

- [1] CHIANG M, ZHANG T. Fog and IoT: an overview of research opportunities [J]. IEEE Internet of Things Journal, 2016, 3 (6): 854-864.
- [2] EVANS D. The Internet of things: how the next evolution of the Internet is changing everything [EB/OL]. [2019-03-26]. https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf.
- [3] European Telecommunications Standards Institute, Mobile-edge computing-introductory technical white paper [EB/OL]. [2019-03-23]. <https://portal.etsi.org/portals/0/tbpages/mec/docs/mobile-edge-computing-introductory-technical-white-paper-v1>.
- [4] TALEB T, SAMDANIS K, MADA B, et al. On multi-access edge computing: a survey of the emerging 5G network edge architecture and orchestration [J]. IEEE Communications Surveys and Tutorials, 2017, 19(3):1657-1681.
- [5] HOANG D T, NIYATO D, WANG P. Optimal admission control policy for mobile cloud computing hotspot with cloudlet [C]// Proceedings of the 2012 IEEE Wireless Communications and Networking Conference: Services, Applications, and Business. Piscataway: IEEE, 2012: 3145-3149.
- [6] YU Y, ZHANG J, LETAIEF K B. Joint subcarrier and CPU time allocation for mobile edge computing [C]// Proceedings of the 2016



- IEEE Global Communications Conference. Piscataway: IEEE, 2016:1-6.
- [7] CHEN X, JIAO L, LI W, et al. Efficient multi-user computation offloading for mobile-edge cloud computing[J]. IEEE/ACM Transactions on Networking, 2016, 24(5):2795-2808.
- [8] LI M, YANG S, ZHANG Z, et al. Joint subcarrier and power allocation for OFDMA based mobile edge computing system[C]// Proceedings of the IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications. Piscataway: IEEE, 2017:1-6.
- [9] XU D, LI Q, ZHU H. Energy-saving computation offloading by joint data compression and resource allocation for mobile-edge computing[J]. IEEE Communications Letters, 2019, 23(4): 704-707.
- [10] CAO J, YANG L, CAO J N. Revisiting computation partitioning in future 5G-based edge computing environments[J]. IEEE Internet of Things Journal, 2019, 6(2): 2427-2438.
- [11] LIU J, MAO Y, ZHANG J, et al. Delay-optimal computation task scheduling for mobile-edge computing systems[C]// Proceedings of the 2016 IEEE International Symposium on Information Theory. Piscataway: IEEE, 2016:1451-1455.
- [12] LIU M, LIU Y. Price-based distributed offloading for mobile-edge computing with computation capacity constraints[J]. IEEE Wireless Communications Letters, 2017, 7(3):420-423.
- [13] KIM S H, PARK S, CHEN M, et al. An optimal pricing scheme for the energy efficient mobile edge computation offloading with OFDMA[J]. IEEE Communications Letters, 2018, 22(9): 1992-1925.
- [14] GUO J, SONG Z, CUI Y, et al. Energy-efficient resource allocation for multi-user mobile edge computing[C]// Proceedings of the 2017 IEEE Global Communications Conference. Piscataway: IEEE, 2017: 1-7.
- [15] 黄婉珍,唐国春. 分支定界法求解最小带权误工工件数排序[J]. 应用数学学报, 1992, 15(2): 194-199. (HUANG W Z, TANG G C. A branch and bound approach to minimizing the weighted number of tardy jobs[J]. Acta Mathematicae Applicatae Sinica, 1992, 15(2): 194-199.)

This work is partially supported by the Scientific Base and Talent Special Project of Guangxi (GUIKEAD19110042), the Director Fund of Guangxi Key Laboratory of Wireless Broadband Communication and Signal Processing (GXKL06160111).

HUANG Dongyan, born in 1984, Ph. D. Her research interests include mobile edge computing, blockchain.

FU Zhongwei, born in 1993, M. S. candidate. His research interests include mobile edge computing, resource optimization.

WANG Bo, born in 1977, Ph. D., lecturer. His research interests include 5G mobile communication.