

DOI: 10.13382/j.jemi.B2407696

D2D 辅助移动边缘计算下的任务卸载和资源分配研究*

占慧芳^{1,2} 李正权^{1,2} 武贵路^{3,4} 聂高峰²

(1. 江南大学物联网工程学院 无锡 214122; 2. 北京邮电大学网络与交换技术全国重点实验室 北京 100876;
3. 海南大学信息与通信工程学院 海口 570000; 4. 海南大学海南省 Internet 信息检索重点实验室 海口 570000)

摘要:针对移动边缘计算 (MEC) 环境下终端设备任务卸载时资源分配效率低的问题, 提出一种基于终端直传通信 (D2D) 技术辅助 MEC 系统的多任务部分卸载方案。该方案基于块坐标下降法 (BCD) 实现任务卸载和资源分配策略的联合优化, 借助动态定价策略激励服务型设备 (SSDs) 共享剩余可用计算资源, 以最小化系统执行成本。首先利用重构-线性化技术 (RLT) 和凸优化理论优化计算资源分配和卸载比例划分, 决定任务分配至本地计算、D2D 卸载和边缘卸载时数据量; 其次根据优化后卸载策略选择最优 SSDs 执行 D2D 卸载任务。仿真结果表明, 与传统部分卸载方案、中继辅助卸载方案和协同计算卸载方案相比, 所提卸载方案在不同设备数目下系统执行总成本分别减少 27.62%、25.58% 和 19.98%, 在不同最大容忍时延条件下系统执行总成本分别平均下降约 43.35%、38.19% 和 36.79% 及在不同任务数据大小下系统执行总成本分别平均下降约 36.47%、30.60% 和 29.15%。进一步实验表明, 与贪婪卸载方案相比, 所提卸载方案在不同设备数目、最大容忍时延和任务数据量下分别平均优化 7.59%、0.39% 和 3.10% 系统执行成本, 有效提高系统资源利用率并降低执行成本。

关键词: 移动边缘计算; 任务卸载; D2D 通信; 资源分配; 块坐标下降法

中图分类号: TN92 **文献标识码:** A **国家标准学科分类代码:** 520

Joint optimization of task offloading and resource allocation in D2D-assisted mobile edge computing

Zhan Huifang^{1,2} Li Zhengquan^{1,2} Wu Guilu^{3,4} Nie Gaofeng²

(1. School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China; 2. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China 3. School of Information and Communication Engineering, Hainan University, Haikou 570000, China 4. Hainan Provincial Key Laboratory for Internet Information Retrieval, Hainan University, Haikou 570000, China)

Abstract: To address the problem of inefficient resource allocation during task offloading in mobile edge computing (MEC) environments for terminal devices, a multi-task partial offloading scheme is proposed that leverages device-to-device (D2D) communication technology to assist the MEC system. The scheme is based on the block coordinate descent (BCD) method to optimize the task offloading and resource allocation strategies jointly. Additionally, a dynamic pricing strategy is adopted to incentivize the service-oriented smart devices (SSDs) to share the remaining available computational resources, aiming to minimize the system-wide execution cost. Firstly, the reconfiguration linearization technique (RLT) and convex optimization theory are utilized to optimize the allocation of computational resources and the offloading ratios, determining the tasks to be allocated to local computing, D2D offloading and edge offloading. Secondly, the appropriate SSDs are selected to perform D2D offloading tasks based on the optimal offloading policy. Simulation results show that, compared with the traditional partial offloading scheme, relay-assisted offloading scheme, and cooperative computing offloading scheme, the proposed offloading scheme reduces the total cost of system execution by 27.62%, 25.58% and 19.98%, respectively, under different numbers of devices, and reduces the total cost of system execution by about 43.35%, 38.19% and

收稿日期: 2024-07-17 Received Date: 2024-07-17

* 基金项目: 国家自然科学基金 (62361023)、海南省自然科学基金 (623MS022)、海南省教育厅项目资助 (Hnky2024ZD-3)、海南大学高层次人才科研启动项目 (KYQD(ZR)-22063)、网络与交换技术全国重点实验室 (北京邮电大学) 开放课题项目 (SKLNST-2023-1-13) 资助

36.79%, respectively, under different conditions of the maximum tolerable delay. The average reduction in total system execution cost under different task data sizes is about 36.47%, 30.60%, and 29.15%, respectively. Further experiments indicate that, compared with the greedy offloading scheme, the proposed offloading scheme optimizes the system execution cost by an average of 7.59%, 0.39% and 3.10% for different numbers of devices, maximum tolerable delays and task data sizes, respectively. Therefore, this scheme effectively enhances the resource utilization while reducing the execution cost.

Keywords: mobile edge computing (MEC); task offloading; device-to-device (D2D) communications; resource allocation; block coordinate descent

0 引言

随着物联网和无线通信技术快速发展,无人驾驶、增强/虚拟现实和高清直播视频流等技术得到广泛应用,进而促进智能城市、智能家居、智能交通等应用领域发展^[1-3]。然而,网络终端设备受限于计算能力和电池寿命,难以应对新业务带来的海量数据及其引发的能耗和时延问题^[4-5]。移动边缘计算(mobile edge computing, MEC)技术依赖边缘计算节点(如基站、接入点)处理卸载的任务,为计算密集型任务和时延敏感型任务提供一种新的解决思路。MEC服务器保障应用程序在任务设备附近执行,极大降低端到端时延,有效缓解核心网络拥塞问题^[6]。然而随着资源需求的增加和用户密度的提高,边缘节点可能面临计算和通信资源瓶颈,导致 MEC 技术在实际应用中面临新的挑战。

为应对上述挑战,MEC系统和终端直传通信(device-to-device, D2D)技术的结合引出 D2D-MEC 系统的概念。边缘计算节点通过其强大计算和通信资源,负责卸载任务处理,同时辅助 D2D 设备动态整合临近设备的可用资源,从而提升系统的任务处理能力。当终端设备受限于计算能力或能耗无法完成任务时, D2D-MEC 系统能借助移动终端富余的计算和通信能力,将任务卸载至边缘节点或附近 D2D 设备进行处理^[7]。已有研究表明, D2D-MEC 集成系统的任务卸载方案可以显著提升系统性能。文献[8]利用 MEC 和 D2D 通信的临近性,在资源受限条件下最大化蜂窝网络支持的设备数量,验证 D2D-MEC 系统能显著提升系统计算能力。文献[9]综述基于中继辅助计算方案和计算辅助中继选择方法的研究,指明 MEC 结合 D2D 技术在提高资源利用率、能量效率和通信容量方面具有显著优势。

目前,移动边缘计算卸载(mobile edge computation offloading, MECO)研究工作主要关注时延^[10]和能耗^[11]及其综合权衡的优化。文献[10]针对车辆完全卸载场景提出一种基于多智能体深度强化学习(deep reinforcement learning, DRL)计算卸载方案,有效降低任务处理总延迟。针对系统能耗优化问题,文献[11]提出多用户任务非正交多址(none-orthogonal multiple

access, NOMA)协作中继卸载树算法方案,解决多用户多任务 MEC 系统中频谱和能量双受限问题。除了时延和能耗,优化目标取决于具体应用场景。为解决高安全通信速率与低功耗之间的矛盾,文献[12]提出一种最大化安全通信效率的卸载策略,并通过快坐标下降与连续凸逼近(successive convex approximation, SCA)相结合的全局优化算法进行优化。文献[13]则基于离散粒子群优化算法,在时延和能耗之间进行进一步权衡。尽管上述研究在处理延迟或能耗等方面取得一定进展,但是忽视系统的经济支出和运营费用,不适合大规模应用场景,且未充分考虑 D2D 通信场景下的性能优化^[14]。

现有 D2D-MEC 系统下的任务卸载方案研究大多集中于完全卸载。文献[15]基于 D2D 协同边缘网络系统模型,设计一种基于深度强化学习的双深度 Q 网络(double DQN, DDQN)算法。文献[16]考虑 D2D 通信技术,从两个角度提出基于较优响应的分布式卸载算法,最小化全网用户累计时延。文献[17]构建基于无线携能通信(simultaneous wireless information and power transfer, SWIPT)的多用户 D2D 通信辅助 MEC 系统模型,提出一种解决能耗最小化问题的任务卸载策略。然而,上述研究通常假设理想的网络条件,依赖 D2D 设备的高质量通信,因此在网络条件恶劣或任务量过大的情况下,任务完成率和系统性能受到影响^[18]。

为此,针对 D2D-MEC 系统,提出一种多用户协作的部分卸载方案,旨在实现系统成本最小化。方案结合 MEC 和 D2D 通信的优势,允许任务设备根据实际网络条件和资源状况动态选择部分任务进行卸载。与现有卸载方案不同,部分卸载策略引入动态资源定价机制,通过灵活分配任务至边缘节点和 D2D 节点,将任务处理过程中消耗的计算和通信资源以价格形式售卖给用户,引导用户卸载流量,从而提高系统资源分配利用效率。在任务设备端,研究系统成本最优下任务卸载策略与资源分配方案,考虑时延、能耗等限制条件。在此基础上,提出一种基于块坐标下降法(block coordinate descent, BCD)的任务卸载和资源分配算法,进一步整合邻近服务型设备可用剩余资源。最后,通过实验仿真验证所提算法的有效性。

1 系统模型

一个 MEC 辅助的 D2D 通信系统,由一个基站 (base station, BS) 和若干个智能设备 (smart devices, SDs) 组成,如图 1 所示。边缘服务器部署于 BS。

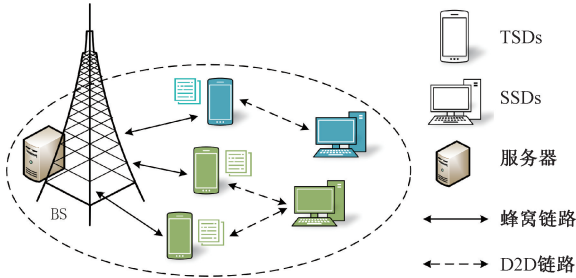


图 1 D2D 辅助的 MEC 系统模型
Fig. 1 D2D-assisted MEC system model

SDs 数据存储和处理能力有限,因此系统中空闲设备采用 D2D 配对方式辅助用户完成卸载。根据本地计算资源能否按时完成任务,将系统中所有 SDs 划分为任务型 SDs (task-oriented smart devices, TSDs) 和服务型 SDs (service-oriented smart devices, SSDs) 两类,分别表示为 $I = \{1, 2, \dots, I\}$ 和 $J = \{1, 2, \dots, J\}$ 。SSDs 可以与 TSDs 之间可以建立独立 D2D 链路,辅助 $TSDsi \in I$ 在最大可容忍时间阈值内完成任务。部分任务无法在 TSDs 完成本地计算,则可以通过蜂窝链路卸载至边缘服务器,或通过 D2D 链路卸载至 SSDs 完成。

假设 TSDs 产生一个计算密集型、时延紧张型任务用数据元组 (C_i, L_i, T_i) , $i \in I$, 其中 C_i 表示处理单位 bit 任务所需 CPU 周期转数, L_i 表示当前任务数据总量, T_i 表示执行任务最大可容忍时延限制。此外,为激励 SSDs 服务更多用户, TSDs 采用动态收费机制为卸载过程占用的计算资源缴纳费用,缓解多用户下有限资源拥堵问题。核心变量符号及其含义如表 1 所示。

受限于链路质量和计算资源, TSDs 选择相应选择策略与 SSDs 建立关联^[19]。假设 $TSDsi \in I$ 与给定距离限制范围 R 内的 $SSDs sj \in J$ 建立 D2D 通路,其中 R 由设备本身特性决定。利用决策矩阵 $X = \{x_{i,j} | i \in I, j \in J\}$ 描述 D2D 链路关联关系,其中 $x_{i,j} \in \{0, 1\}$ 是二进制选择变量, $x_{i,j} = 1$ 表示 $TSDsi \in I$ 与 $SSDs sj \in J$ 成功建立 D2D 通信链路,而 $x_{i,j} = 0$ 表示两者之间不存在 D2D 通信链路。

移动终端部分卸载模型计算和卸载流程,如图 2 所示, TSDs 上任务划分为 3 个部分,即部分任务本地计算、部分任务卸载至 SSDs 计算,部分任务卸载至边缘服务器计算。 L_i^l, L_i^d 和 L_i^e 分别表征分配给本地计算、D2D 卸载和

边缘卸载 3 个阶段时任务计算比特数目。显然地,任务不同划分比例影响 TSDs 电池能量和资源付费情况。TSDs $i \in I$ 产生 L_i 任务,且 $L_i^l + L_i^d + L_i^e = L_i$, 其中 L_i^l 任务执行本地计算, L_i^d 任务通过 D2D 链路在 SSDs 上执行,而 L_i^e 任务卸载至 BS 处理。由于卸载处理后传输至用户数据量远小于卸载处理前数据量,因此暂不考虑任务卸载回传阶段时延情况。对于 TSDs $i \in I$ 来说,本地计算、D2D 卸载和边缘卸载 3 个阶段时延采用 $T_i^k, k = \{l, d, e\}$ 表示。特别地, D2D 卸载和边缘卸载两个阶段时延还包括传输和计算两种状态,标记为 $T_i^{k,s}, s = \{t, c\}$ 。

表 1 变量符号及含义

Table 1 Symbols and meanings of variables			
符号	含义	符号	含义
C_i	处理任务所需计算资源	m^e	BS 提供单位 bit 计算服务的单价
L_i	TSDs 待执行任务总数据大小	M_i	TSDs 执行成本
L_i^l	分配至本地计算任务数据大小	f_i	TSDs 本地计算频率
L_i^d	分配至 D2D 卸载任务数据大小	f_j^l	SSDs 本地计算频率
L_i^e	分配至边缘卸载任务数据大小	$f_{i,j}^d$	SSDs 分配至 TSDs 卸载任务计算频率
T_i	最大可容忍时间阈值	f_i^e	BS 分配至 TSDs 卸载任务计算频率
$x_{i,j}$	TSDs 与 SSDs 选择关系	p_i^d	D2D 卸载阶段 TSDs 发射功率
κ_i	有效开关电容系数	p_i^e	边缘卸载阶段 TSDs 发射功率
m^l	TSDs 提供单位 bit 计算服务的单价	$R_{i,j}^d$	D2D 通信传输速率
m_j^d	SSDs 提供单位 bit 计算服务的单价	$R_{i,j}^e$	蜂窝移动通信传输速率

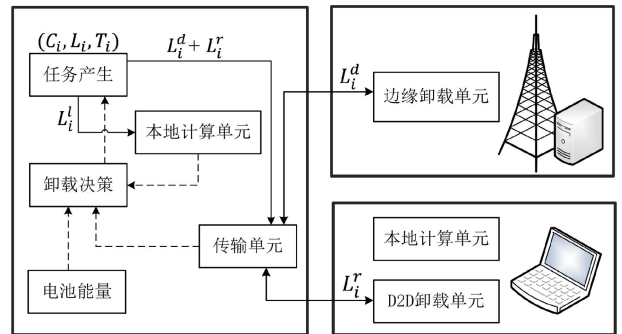


图 2 计算和卸载流程

Fig. 2 Computing and offloading procedures

1.1 通信模型

传输时延根据通信路径不同由 D2D 卸载和边缘卸载两个阶段确定,通信过程采用正交频分多址 (OFDMA)

技术,分配一个子信道用于 TSDs $i \in I$ 蜂窝链路传输,同时分配另一个子信道用于 D2D 链路传输。在时间块 T_i 内,假设信道分配、发射功率等参数恒定,每个子信道的传输带宽和噪声功率分别标记为 B 和 N 。

1) D2D 卸载通信:借助 SSDs 计算资源,将所卸载数据包从 TSDs $i \in I$ 直接传输到 SSDs $j \in J$ 上执行。在此过程中,将 TSDs 发射功率和 (i,j) 配对下 D2D 链路信道功率增益定义为 p_i^d 和 $h_{i,j}^d$ 。根据香农定理,计算 D2D 卸载阶段数据传输速率 $R_{i,j}^d$ ^[20],即:

$$R_{i,j}^d = B \log_2 \left(1 + \frac{h_{i,j}^d \cdot p_i^d}{N} \right) \quad (1)$$

D2D 卸载阶段花费时间和能量损耗主要受 D2D 链路传输 L_i^d -bit 的 TSDs 任务影响。该过程时延和能耗表达式分别为:

$$T_i^{d,t} = \frac{L_i^d}{\sum_{j=1}^J R_{i,j}^d x_{i,j}^d} \quad (2)$$

$$E_i^{d,comm} = p_i^d T_i^{d,t} \quad (3)$$

2) 边缘卸载通信:SSDs $j \in J$ 本地计算无法及时完成任务将通过蜂窝链路卸载至 BS 完成。类似 D2D 卸载阶段定义, p_i^e 和 h_i^e 分别表示为 TSDs $i \in I$ 传输数据至 BS 时发射功率和蜂窝链路信道增益。因此,边缘卸载中数据传输速率 $R_{i,0}^e$ 计算为:

$$R_{i,0}^e = B \log_2 \left(1 + \frac{h_i^e \cdot p_i^e}{N} \right) \quad (4)$$

执行边缘卸载处理数据包仅有 L_i^e -bit,因此计算 TSDs $i \in I$ 传输时延和能量损耗数学表达式分别为:

$$T_i^{e,t} = \frac{L_i^e}{R_{i,0}^e} \quad (5)$$

$$E_i^{e,comm} = p_i^e T_i^{e,t} \quad (6)$$

1.2 计算模型

由于 BS 和 SSDs 具有一定计算和存储资源,为 TSDs $i \in I$ 提供有限的计算、缓存和存储服务,并进行合理规划分资源来实现最大化利用率。假设 TSDs 和 SSDs 全部采用单工通信模式,成功接收数据后执行数据处理。依据计算模式不同可将计算模型分为本地计算模型和卸载模型两个部分。

1) 本地计算

在能耗和资源受限条件下,TSDs 将充分调用自身计算资源,减少卸载数据所占计算和通信资源比例。本地计算阶段所消耗时间为:

$$T_i^{l,c} = \frac{C_i L_i^l}{f_i} \quad (7)$$

其中, f_i 为 TSDs $i \in I$ 的 CPU 频率。为最大限度地调用本地资源,假设 $T_i^{l,c} = T_i$,则本地计算任务数据大小

为 $L_i^l = f_i T_i / C_i$ 。本地计算阶段能量消耗为:

$$E_i^{comp} = \kappa_i f_i^2 C_i \quad (8)$$

其中, κ_i 表示有效开关电容系数。

2) D2D 卸载计算

SSDs $j \in J$ 接收到卸载的 L_i^d -bit 数据包后,利用自身剩余计算资源处理数据包,并将计算结果返回至 TSDs。D2D 卸载过程中,SSDs 执行任务所花费时间 $T_i^{d,c}$ 表示为:

$$T_i^{d,c} = \frac{C_i L_i^d}{\sum_{j=1}^J x_{i,j} f_{i,j}^d} \quad (9)$$

其中, $f_{i,j}^d$ 为 SSDs $j \in J$ 所调用计算频率。值得注意的是,SSDs 可能产生待执行任务,因而需要先执行本地任务,再协助 TSDs 完成卸载。因此 D2D 卸载的前提条件是 SSDs 完成本地任务并且完成卸载数据包接收。采用 T_j^l 表示 SSDs 本地计算时间,则 D2D 卸载阶段总时延为:

$$T_i^d = \max(T_i^{d,t}, \sum_{j=1}^J x_{i,j} T_j^l) + T_i^{d,c} \quad (10)$$

其中, $T_j^l = C_j L_j / f_j^l$ 。

3) 边缘卸载计算

TSDs 向 BS 成功传输 L_i^e -bit 数据包后,边缘服务器为 SSDs $j \in J$ 合理分配计算资源,并协助完成任务卸载。此过程时间消耗为:

$$T_i^{e,c} = \frac{C_i L_i^e}{f_i^e} \quad (11)$$

其中, f_i^e 为 BS 分配给 SSDs $j \in J$ 计算频率。至此,边缘卸载总时延为:

$$T_i^e = T_i^{e,t} + T_i^{e,c} \quad (12)$$

1.3 动态定价

在任务计算和卸载阶段,系统需要依据动态定价机制,规范 TSDs $i \in I$ 卸载行为及 SSDs $j \in J$ 有效激励。针对本地计算任务 L_i^l 、D2D 卸载任务 L_i^d 和边缘卸载任务 L_i^e ,定义 TSDs $i \in I$ 、SSDs $j \in J$ 和 BS 提供单位 bit 计算服务的单价分别为 m^l 、 m_j^d 和 m^e ^[21]。为在规定时间内完成任务,D2D-MEC 系统内 TSDs $i \in I$ 执行成本 M_i 表示为:

$$M_i = L_i^l \cdot m^l + \sum_{j=1}^J x_{i,j} L_i^d \cdot m_j^d + L_i^e \cdot m^e \quad (13)$$

假设 TSDs $i \in I$ 初始能量状态标记为 E_i ,经过本地计算、D2D 卸载和边缘卸载 3 个阶段处理,则 TSDs $i \in I$ 剩余电池能量状态 E_i^r 计算为:

$$E_i^r = E_i - E_i^{d,comm} - E_i^{e,comm} - E_i^{comp} \quad (14)$$

1.4 执行成本建模

D2D-MEC 系统主要目标是在时延、能耗和算力限制下优化系统总执行成本。联合优化任务卸载方案、计算资源分配和 D2D 关联策略,将执行成本问题建模为:

$$\begin{aligned}
 P1: \min & \sum_{i=1}^I M_i \\
 & L_i^d, L_i^e \\
 & f_{i,j}^d, f_{i,j}^e, x_{i,j} \\
 \text{s. t.} & \begin{cases}
 C1: L_i^d \geq 0, L_i^e \geq 0 \\
 C2: L_i^l + L_i^d + L_i^e = L_i \\
 C3: f_j^l + \sum_{i=1}^I f_{i,j}^d x_{i,j} \leq F_j^{\max} \\
 C4: \sum_{i=1}^I f_i^e \leq F_0^{\max} \\
 C5: T_i^k \leq T_i, \forall k = \{d, e\} \\
 C6: E_i^r \geq 0 \\
 C7: x_{i,j} \in \{0, 1\} \\
 C8: \sum_{j=1}^J x_{i,j} = 1
 \end{cases} \quad (15)
 \end{aligned}$$

其中,约束 C1 和 C2 表明任务划分策略所满足限制条件,C3、C4 和 C5 限定任务在指定时间和 CPU 频率范围内执行,C6 表明 TSDs 所需要维持的电池能量状态,C7 和 C8 表示 D2D 关联策略。显然,由于离散优化变量 $x_{i,j}$ 存在以及目标函数和限制条件 C7、C9 具有非凸性,可判定目标函数 P1 是一个非凸问题,难以直接采用传统凸优化技术求解。

2 卸载方案

目标函数 P1 是非凸函数,求解 P1 主要挑战在于离散变量和多个连续变量之间耦合问题,这种耦合导致直接求解的复杂和计算密集。考虑原问题的可分解性,尝试采用交替优化子问题方法求解问题近似解^[22]。

首先将原问题根据优化变量类型解耦为两个子问题,即资源分配优化子问题和服务设备选择子问题。资源分配优化子问题涉及如何在给定资源约束下进行资源的最优分配,以最小化执行成本和提高资源利用率;服务设备选择子问题集中于如何在不同的服务设备之间进行选择,以减少网络拥塞并保持服务质量。成功解耦后,采用交替优化法求解,即一次仅更新一个或几个变量块,并结合凸优化技术逐一优化子问题。通过不断迭代交替优化过程,逐步逼近全局最优解。最后,基于块坐标下降法进行全局优化,避免陷入局部最优解的同时加速收敛,最终求解原非凸函数问题^[23]。在保证计算效率的前提下显著减少执行成本和网络拥塞,最终实现对移动边缘计算环境中资源的最优管理。

2.1 资源分配优化

用 $X = \{x_{i,j} \mid i \in I, j \in \mathcal{J}\}$ 表示所有 SSDs 选择策略,给定初始策略 X^0 为基于距离的选择策略, $X^l (l=1, 2, \dots,$

$l)$ 表示第 l 次迭代时选择策略。在给定选择策略为 X^l 时,将 SSDs $j \in J$ 资源供应给 TSDs $i \in I$,从而形成设备与设备之间配对关系 (i, j) ,此时存在 D2D 关联矩阵 $x_{i,j} = 1$ 。结合式(10)、(12)和(14),P1 可以重写为:

$$\begin{aligned}
 P2: \min & \sum_{i=1}^I (L_i^l \cdot m^l + \sum_{j=1}^J x_{i,j} L_i^d \cdot m_j^d + L_i^e \cdot m^e) \\
 & L_i^d, L_i^e, f_{i,j}^d, f_{i,j}^e \\
 \text{s. t.} & \begin{cases}
 C1: L_i^d \geq 0, L_i^e \geq 0 \\
 C2: L_i^l + L_i^d + L_i^e = L_i \\
 C3: f_j^l + \sum_{i=1}^I f_{i,j}^d x_{i,j} \leq F_j^{\max} \\
 C4: \sum_{i=1}^I f_i^e \leq F_0^{\max} \\
 C5: \frac{L_i^d}{R_{i,j}^d} + \frac{C_i L_i^d}{f_{i,j}^d} \leq T_i \\
 C6: \frac{C_j L_j}{f_j^l} + \frac{C_i L_i^d}{f_{i,j}^d} \leq T_i \\
 C7: \frac{L_i^e}{R_{i,0}^e} + \frac{C_i L_i^e}{f_i^e} \leq T_i \\
 C8: p_i^d \frac{L_i^d}{R_{i,j}^d} + p_i^e \frac{L_i^e}{R_{i,0}^e} \leq E_i - E_i^{\text{comp}}
 \end{cases} \quad (16)
 \end{aligned}$$

由于 C5 和 C7 中存在变量耦合关系 $L_i^d f_{i,j}^d$ 和 $L_i^e f_i^e$,因此问题 P2 是非凸问题,借助重构-线性化技术(reformulation linearization technique, RLT)转换乘积项^[24]。引入辅助变量 $\mu_{i,j}$ 和 ν_i ,令 $\mu_{i,j} = L_i^d f_{i,j}^d, \nu_i = L_i^e f_i^e$,则 P2 重写为:

$$\begin{aligned}
 P2.1: \min & \sum_{i=1}^I (L_i^l \cdot m^l + \sum_{j=1}^J x_{i,j} L_i^d \cdot m_j^d + L_i^e \cdot m^e) \\
 & L_i^d, L_i^e, f_{i,j}^d, f_{i,j}^e \\
 \text{s. t.} & \begin{cases}
 C1: L_i^d \geq 0, L_i^e \geq 0 \\
 C2: L_i^l + L_i^d + L_i^e = L_i \\
 C3: f_j^l + \sum_{i=1}^I f_{i,j}^d x_{i,j} \leq F_j^{\max} \\
 C4: \sum_{i=1}^I f_i^e \leq F_0^{\max} \\
 C5: \mu_{i,j} + C_i L_i^d R_{i,j}^d - T_i f_{i,j}^d R_{i,j}^d \leq 0 \\
 C6: (C_j L_j - f_j^l T_i) f_{i,j}^d + C_i L_i^d f_j^l \leq 0 \\
 C7: \nu_i + C_i L_i^e R_{i,0}^e - T_i f_i^e R_{i,0}^e \leq 0 \\
 C8: p_i^d \frac{L_i^d}{R_{i,j}^d} + p_i^e \frac{L_i^e}{R_{i,0}^e} \leq E_i - E_i^{\text{comp}}
 \end{cases} \quad (17)
 \end{aligned}$$

其中, $0 \leq L_i^d \leq L_i, 0 \leq L_i^e \leq L_i, 0 \leq f_{i,j}^d \leq F_j^{\max} - f_j^l, 0 \leq f_i^e \leq F_0^{\max}$,则 $\mu_{i,j}$ 的 RLT 约束因子乘积限制可以描述为:

$$\begin{cases} \{[L_i^d - 0][f_{i,j}^d - 0] \geq 0\}_{LS} \\ \{[L_i^d - 0][F_j^{\max} - f_j^l - f_{i,j}^d] \geq 0\}_{LS} \\ \{[L_i - L_i^d][f_{i,j}^d - 0] \geq 0\}_{LS} \\ \{[L_i - L_i^d][F_j^{\max} - f_j^l - f_{i,j}^d] \geq 0\}_{LS} \end{cases} \quad (18)$$

其中, $\{\cdot\}_{LS}$ 表示线性化运算, 代入 $\mu_{i,j} = L_i^d f_{i,j}^d$ 化简可得:

$$\begin{cases} \mu_{i,j} \geq 0 \\ L_i^d (F_j^{\max} - f_j^l) - \mu_{i,j} \geq 0 \\ L_i f_{i,j}^d - \mu_{i,j} \geq 0 \\ (L_i - L_i^d) (F_j^{\max} - f_j^l) - L_i f_{i,j}^d + \mu_{i,j} \geq 0 \end{cases} \quad (19)$$

同理,

$$\begin{cases} \{[L_i^e - 0][f_i^e - 0] \geq 0\}_{LS} \\ \{[L_i^e - 0][F_0^{\max} - f_i^e] \geq 0\}_{LS} \\ \{[L_i - L_i^e][f_i^e - 0] \geq 0\}_{LS} \\ \{[L_i - L_i^e][F_0^{\max} - f_i^e] \geq 0\}_{LS} \end{cases} \quad (20)$$

代入 $\nu_i = L_i^e f_i^e$ 后可得:

$$\begin{cases} \nu_i \geq 0 \\ L_i^e F_0^{\max} - \nu_i \geq 0 \\ L_i f_i^e - \nu_i \geq 0 \\ L_i F_0^{\max} - L_i f_i^e - L_i^e F_0^{\max} + \nu_i \geq 0 \end{cases} \quad (21)$$

综上所述, 问题 P2.1 经过 RLT 消除二次项后可以表示为:

$$\begin{aligned} P2.2: \min & \sum_{i=1}^I (L_i^l \cdot m^l + \sum_{j=1}^J x_{i,j} L_i^d \cdot m_j^d + L_i^e \cdot m^e) \\ \text{s. t.} & \begin{cases} C1 - C8; \text{式(17)} \\ C9 - C12; \text{式(19)} \\ C13 - C16; \text{式(21)} \end{cases} \end{aligned} \quad (22)$$

分析可知变换后目标函数 P2.2 为凸优化问题, 可借助凸优化技术求解, 如 CVX 工具箱, 从而得到当前迭代次数下局部最优解。

2.2 服务型设备选择

执行资源分配后, 以局部最优解 $\mathbf{L}^* = \{L_i^{l,*}, L_i^{d,*}, L_i^{e,*}\}$ 和 $\mathbf{F}^* = \{f_{i,j}^{d,*}, f_i^{e,*}\}$ 为基础, 将原始优化问题 P1 转化为 SSDs 选择子问题:

$$\begin{aligned} P3: \min & \sum_{i=1}^I (L_i^l \cdot m^l + \sum_{j=1}^J x_{i,j} L_i^d \cdot m_j^d + L_i^e \cdot m^e) \\ \text{s. t.} & \begin{cases} C1: f_j^l + \sum_{i=1}^I f_{i,j}^d x_{i,j} \leq F_j^{\max} \\ C2: \frac{L_i^d}{\sum_{j=1}^J R_{i,j}^d x_{i,j}} + \frac{C_i L_i^d}{\sum_{j=1}^J x_{i,j} f_{i,j}^d} \leq T_i \\ C3: \sum_{j=1}^J x_{i,j} \frac{C_j L_j}{f_j^l} + \frac{C_i L_i^d}{\sum_{j=1}^J x_{i,j} f_{i,j}^d} \leq T_i \\ C4: p_i^d \frac{L_i^d}{\sum_{j=1}^J R_{i,j}^d x_{i,j}} + p_i^e \frac{L_i^e}{R_{i,0}^e} \leq E_i - E_i^{\text{comp}} \\ C5: x_{i,j} \in \{0, 1\} \\ C6: \sum_{j=1}^J x_{i,j} = 1 \end{cases} \end{aligned} \quad (23)$$

由于 P3 中决策变量 $x_{i,j}$ 为离散变量, 目标函数 P3 是非凸函数, 需借助松弛变量法来降低算法复杂度。针对离散变量 $x_{i,j}$, 放松限制条件为 $0 \leq x_{i,j} \leq 1$ 。则 P3 简化后为:

$$\begin{aligned} P3.1: \min & \sum_{i=1}^I (L_i^l \cdot m^l + \sum_{j=1}^J x_{i,j} L_i^d \cdot m_j^d + L_i^e \cdot m^e) \\ \text{s. t.} & \begin{cases} C1: f_j^l + \sum_{i=1}^I f_{i,j}^d x_{i,j} \leq F_j^{\max} \\ C2: \frac{L_i^d}{\sum_{j=1}^J R_{i,j}^d x_{i,j}} + \frac{C_i L_i^d}{\sum_{j=1}^J x_{i,j} f_{i,j}^d} \leq T_i \\ C3: \sum_{j=1}^J x_{i,j} \frac{C_j L_j}{f_j^l} + \frac{C_i L_i^d}{\sum_{j=1}^J x_{i,j} f_{i,j}^d} \leq T_i \\ C4: p_i^d \frac{L_i^d}{\sum_{j=1}^J R_{i,j}^d x_{i,j}} + p_i^e \frac{L_i^e}{R_{i,0}^e} \leq E_i - E_i^{\text{comp}} \\ C5: 0 \leq x_{i,j} \leq 1 \\ C6: \sum_{j=1}^J x_{i,j} = 1 \end{cases} \end{aligned} \quad (24)$$

经过变换后, 问题 P3.1 是凸的, 可通过 CVX 求解器获得可行解, 并对优化变量 $x_{i,j}$ 进行恢复^[25]。系统执行成本优化问题的具体迭代步骤如表 2 所示。

表 2 任务卸载和资源分配算法

Table 2 The algorithm of task offloading and resource allocation

算法 1 任务卸载和资源分配算法

输入: 任务 (C_i, L_i, T_i) , 初始选择决策 X^0 , 收敛精度 ξ 。输出: 卸载方案 L^* , 计算频率 F^* , 选择策略 X^* 。初始化: $m \leftarrow 0$

Repeat

固定 X^m , 求解问题 P2.2, 得到 L^{m+1} 和 F^{m+1} ;固定 L^{m+1} 和 F^{m+1} , 求解问题 P3.1, 得到 X^{m+1} ;将 L^{m+1} , F^{m+1} 和 X^{m+1} 代入目标函数, 得到 M^{m+1} ; $m \leftarrow m + 1$

end

Until $\|M^m - M^{m-1}\| < \xi$

3 仿真结果

采用 MATLAB 软件对所提卸载方案进行仿真以评估性能。在 D2D 辅助 MEC 系统中, BS 位置固定于中心位置处, 覆盖范围 300 m, 每个 D2D 链路最大范围 R 设置为 50 m。系统内随机分布 10 个 TSDs, 假设每个设备都存在一个计算密集型任务, 数据大小和 CPU 循环周期满足 $L_i = 500$ kB 和 $C_i = 1$ MHz 的均匀分布, 所有任务需要在时延 $T_i \sim U(3, 4)$ s 内完成。上行链路中子信道带宽为 $B = 2$ MHz, 噪声功率为 $N = -100$ dBm。TSDs 传输功率设置为 $p_i^d = 2.3$ dBm 和 $p_i^e = 2.6$ dBm, 有效开关电容为 $\kappa_i \sim U(1, 5) \times 10^{-27}$ 。TSDs 计算能力 f_i 满足 $f_i \sim U(0.9, 1) \times 10^8$ cycles/s, SSDs 和 MEC 服务器可提供最大计算资源分别为 $F_j^{\max} \sim U(1, 2)$ GHz 和 $F_0^{\max} = 10$ GHz。除此之外, TSDs 剩余电池容量取值为 $E_i = 50$ J。

3.1 卸载方案性能比较

所提算法是系统成本最优下 SSDs 选择和部分卸载方案, 下面讨论该算法与已有卸载策略性能比较分析:

1) 传统部分卸载方案。采用传统任务卸载方式, 将用户计算任务部分留在用户端执行本地计算, 部分卸载至 BS 执行边缘计算, 相应参数设置为: $J = 0, L_i^d = 0, m_j^d = 0$ 。

2) 协同计算卸载方案^[26]。将每个分割的计算任务任意划分为 $J + 1$ 份, 以便分别在用户自身和代理 AP 处并行执行, 相应参数设置为: $L_i^e = 0, m^e = 0$ 。

3) 中继辅助卸载方案^[27]。通过选择合适 AP 作为中继, 用户可以将计算任务卸载至服务器, 但是任务无法划分, 必须作为一个整体在本地或 MEC 服务器上进行处理, 相应参数设置为: $L_i^d = 0, m_j^d = 0$ 。

如图 3 所示, 系统总成本随着 TSDs 个数增加而趋于上升, 所提卸载方案总成本小于传统部分卸载方案、中继辅助卸载方案和协同计算卸载方案。随着 TSDs 个数增

加, 由于系统待执行任务总数据量增加, 更多计算资源和通信资源被充分挖掘利用, 任务对应执行成本也将逐步增长。传统部分卸载方案总成本最高, 而中继辅助卸载方案和协同计算卸载方案次之。这是因为传统部分卸载方案没有考虑有限时间内最大通信传输限制, 无法在短时间内调度更多边缘算力。中继辅助卸载方案和协同计算卸载方案兼顾传统部分卸载方案的优势和不足, 借用 D2D 通信技术优化通信链路传输, 而协同计算方案为系统补充更多空闲设备算力。相对于传统部分卸载方案、中继辅助卸载方案和协同计算卸载方案, 所提卸载方案系统总成本平均减少 27.62%、25.58% 和 19.98%。

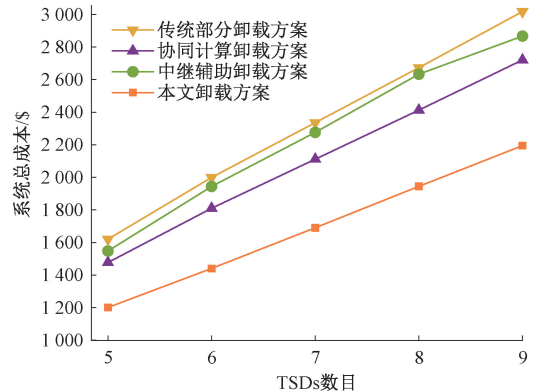


图 3 不同卸载方案下 TSDs 个数与系统总成本关系
Fig. 3 The number of TSDs versus the total system cost under different offloading schemes

传统部分卸载方案、中继辅助卸载方案和协同计算卸载方案及所提卸载方案中, 最大可容忍时延与系统总成本之间的关系如图 4 所示。无论采用何种卸载方案, 最大可容忍时延减少都促使系统总成本降低, 这是因为时延约束增大导致本地计算获得更优性能趋势, 从而被更多系统用户青睐, 降低整体任务执行成本。所提方案相较于传统部分卸载方案、中继辅助卸载方案和协同计算卸载方案总成本分别平均下降约 43.35%、38.19% 和 36.79%。所提卸载方案在低时延场景下成本更低, 且受时延限制影响最小。这是由于 D2D-MEC 系统在低时延上可以同时借助 TSDs 和 SSDs 双重算力支持, 从而获得更优系统性能。

考虑到任务数据量对所提卸载方案影响, 图 5 描述 TSDs 数目 $I = 10$ 时不同卸载方案下总成本随着任务数据大小变化情况。随着任务数据大小增加, 卸载方案系统成本也逐渐增大, 而所提卸载方案具有更低系统成本。这是因为任务数据量增加使任务执行所需计算资源和通信资源不断消耗, TSDs 更趋向于任务卸载而非本地计算, 所以此时执行成本也在不断增加。所提卸载方案相对于其他卸载方案总成本更低, 相对于协同计算卸载方

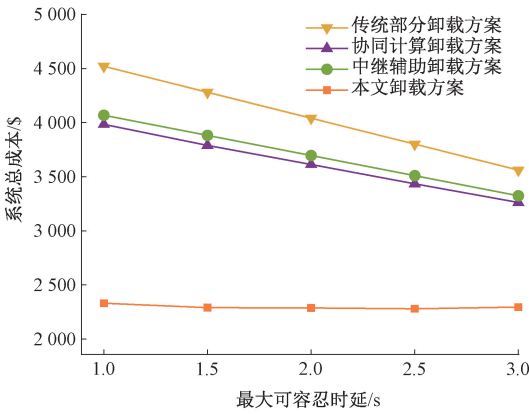


图 4 不同卸载方案下最大可容忍时延与系统总成本关系
Fig. 4 The maximum tolerable delay versus the total system cost under different offloading schemes

案降低 29.15%，相对于中继辅助卸载方案降低 30.60%，而相对于传统部分卸载方案降低 36.47%。

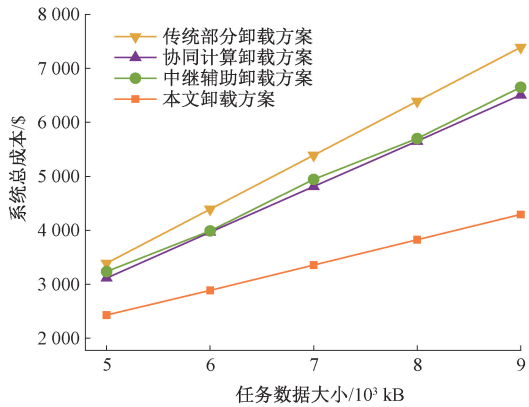


图 5 不同卸载方案下任务数据大小与系统总成本关系
Fig. 5 The task size versus the total system cost under different offloading schemes

3.2 算法性能比较

基于 BCD 的任务卸载和资源分配算法的复杂度主要取决于求解子问题采用的 CVX 方法。假设外层和内层的迭代次数分别为 L_1 和 L_2 。由于 CVX 工具箱使用内点法进行求解,因此内层复杂度取决于内点法的复杂度,可表述为 $L_2(IJ)^{3.5}$ 。故而算法的总计算复杂度为 $O(L_1 \cdot (1/l_1^2 + L_2 \cdot (IJ)^{3.5}))$,其中 l_1 是块坐标下降法的容错精度。

本节对比 BCD 算法和贪婪算法在联合优化任务卸载和资源分配策略上的性能差异。基于贪婪策略的任务卸载与资源分配算法通过逐步选择当前最优的 SSDs 设备来完成任务卸载和成本优化。为此,通过仿真实验分析两种算法在相同卸载场景下的系统执行总成本差异,并进一步考察其在设备数目、最大可容忍时延和任务数

据量变化下的表现。

图 6 展示贪婪算法与 BCD 算法随着 TSDs 数目变化的系统总成本优化性能差异。随着 TSDs 设备数量的增加,两种算法的总成本均随之上升,这主要是由于系统资源竞争加剧和任务负载加重。然而,BCD 算法始终维持更低的系统执行成本,相较于贪婪算法降低 7.59%,表明在任务卸载和资源优化上具有更高的效率和性能。这种优势归结于 BCD 算法可通过全局优化的方式有效整合资源并减少资源浪费,从而提高系统性能。

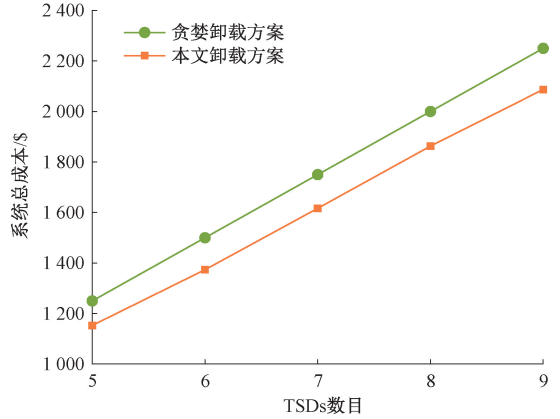


图 6 不同算法下 TSDs 个数与系统总成本关系
Fig. 6 The number of TSDs versus the total system cost under different algorithms

图 7 展示两种策略在不同最大可容忍时延下的总成本对比。最大可容忍时延增加后,两种策略的总成本都显著下降,主要是因为系统在较大时延范围内有更多的优化空间,可减少计算和通信成本。所提卸载方案性能始终优于贪婪卸载方案,成本降低 0.39%,这是因为块坐标下降法能通过全局迭代在任务卸载和资源分配间实现更好的综合优化。而贪婪策略虽然计算简单,但缺乏全局视角,因此其在减少系统总成本方面的效果有限。

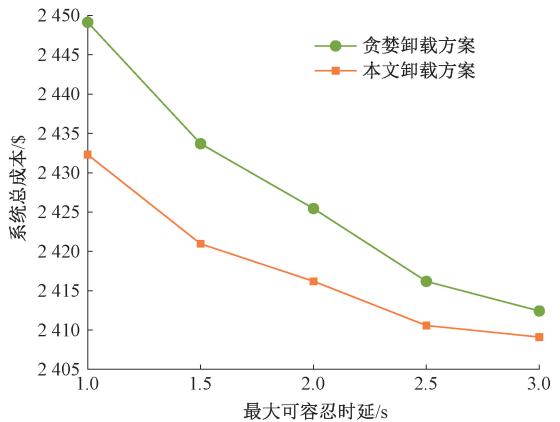


图 7 不同算法下最大可容忍时延与系统总成本关系
Fig. 7 The maximum tolerable delay versus the total system cost under different algorithms

图 8 比较基于块坐标下降法和基于贪婪策略的任务卸载和资源分配方案在不同任务数据量下的总成本。随着任务数据量增大,两种策略的总成本均呈增长趋势。在任务数据较小时,两种策略的差距不大,但随着数据量增加,所提卸载方案逐渐表现出更低的成本。这是因为在任务数据量较小时,系统开销相对较小,两种算法都能有效处理;随着数据量增加,贪婪卸载方案因局部最优选择的局限性,无法有效处理更大的任务数据量。而所提卸载方案能够通过迭代优化保持低系统成本,相较于基线方案,系统总成本降低 3.10%。

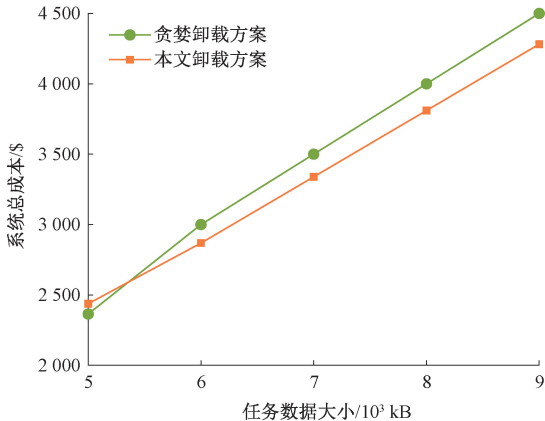


图 8 不同算法下任务数据大小与系统总成本关系

Fig. 8 The task size versus the total system cost under different algorithms

4 结 论

通过引入资源定价机制,提出一种基于 D2D-MEC 系统的资源分配和多任务卸载方案,以提高在时延、计算资源和通信资源等多重限制下的资源分配效率。采用 RLT 方法实现卸载比例的最优划分,并结合 BCD 算法对任务卸载和资源分配进行联合优化。仿真结果表明,所提卸载方案在相同条件下相较于传统部分卸载方案、中继辅助卸载方案和协同计算卸载方案更具系统性能优势,有效降低系统成本,并优于贪婪卸载方案。未来将考虑进一步探索高密度异构网络和多种通信技术共存场景下的任务卸载策略优化。

参考文献

[1] 黄永明, 郑冲, 张征明, 等. 大规模无线网络移动边缘计算和缓存研究 [J]. 通信学报, 2021, 42(4): 44-61.
HUANG Y M, ZHENG CH, ZHANG ZH M, et al. Research on mobile edge computing and caching in massive wireless communication network [J]. Journal on Communications, 2021, 42(4): 44-61.

[2] JIANG H, MUKHERJEE M, ZHOU J, et al. Channel modeling

and characteristics for 6G wireless communications [J]. IEEE Network, 2021, 35(1): 296-303.

- [3] KONG L H, TAN J L, HUANG J Q, et al. Edge-computing-driven internet of things: A survey [J]. ACM Computing Surveys, 2023, 55(8): 1-41.
- [4] 王瑶, 卢先领, 沈义峰. 移动边缘计算中依赖型任务的调度模型研究 [J]. 电子测量与仪器学报, 2022, 36(8): 60-68.
WANG Y, LU X L, SHEN Y F. Research on scheduling model of dependent tasks in mobile edge computing [J]. Journal of Electronic Measurement and Instrumentation, 2022, 36(8): 60-68.
- [5] 杨文杰, 巨涛, 杨阳, 等. 面向边缘计算的人工鱼群搜索任务调度 [J]. 电子测量与仪器学报, 2022, 36(11): 149-159.
YANG W J, JU T, YANG Y, et al. Artificial fish swarm search task scheduling for edge computing [J]. Journal of Electronic Measurement and Instrumentation, 2022, 36(11): 149-159.
- [6] 吕洁娜, 张家波, 张祖凡, 等. 移动边缘计算卸载策略综述 [J]. 小型微型计算机系统, 2020, 41(9): 1866-1877.
LYU J N, ZHANG J B, ZHANG Z F, et al. Survey of mobile edge computing offloading strategies [J]. Journal of Chinese Computer Systems, 2020, 41(9): 1866-1877.
- [7] 林峰, 罗钺文, 丁鹏举, 等. 车联网中自适应联合计算卸载资源分配算法 [J]. 计算机工程与设计, 2021, 42(7): 1824-1830.
LIN F, LUO CH W, DING P J, et al. Resource allocation algorithm for adaptive joint computing offloading in C-V2X [J]. Computer Engineering and Design, 2021, 42(7): 1824-1830.
- [8] HE Y H, REN J K, YU G D, et al. D2D communications meet mobile edge computing for enhanced computation capacity in cellular networks [J]. IEEE Transactions on Wireless Communications, 2019, 18(3): 1750-1763.
- [9] 陈澈, 郑艺峰, 杨敬民, 等. 移动边缘计算中基于中继辅助计算方法综述 [J]. 计算机应用研究, 2022, 39(8): 2241-2248.
CHEN CH, ZHENG Y F, YANG J M, et al. Survey on relay-assisted computing methods in mobile edge computing [J]. Application Research of Computers, 2022, 39(8): 2241-2248.
- [10] ZHU X Y, LUO Y Y, LIU AN F, et al. Multiagent deep reinforcement learning for vehicular computation offloading in IoT [J]. IEEE Internet of Things Journal, 2021, 8(12): 9763-9773.
- [11] 陈韩, 张晶, 董俊, 等. 基于 NOMA 的多用户协作

- MEC 卸载方案[J]. 南京邮电大学学报(自然科学版), 2022, 42(6): 52-60.
- CHEN H, ZHANG J, DONG J, et al. A multi-user cooperative MEC offloading scheme based on NOMA [J]. Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition), 2022, 42(6): 52-60.
- [12] 余雪勇, 邱礼翔, 宋家宁, 等. 无人机辅助边缘计算中安全通信与能效优化策略[J]. 通信学报, 2023, 44(3): 45-54.
- YU X Y, QIU L X, SONG J N, et al. Security communication and energy efficiency optimization strategy in UAV-aided edge computing[J]. Journal on Communications, 2023, 44(3): 45-54.
- [13] ZHAO Z C, ZHAO R, XIA J J, et al. A novel framework of three-hierarchical offloading optimization for MEC in industrial IoT networks[J]. IEEE Transactions on Industrial Informatics, 2020, 16(8): 5424-5434.
- [14] JIANG H, ZHANG Z CH, GUI G, et al. A novel estimated wideband geometry-based vehicle-to-vehicle channel model using an AoD and AoA estimation algorithm[J]. IEEE Access, 2019, 7: 35124-35131.
- [15] 李斌, 徐天成. 终端直通辅助的移动边缘计算任务动态卸载方案[J]. 无线电工程, 2022, 52(12): 2101-2108.
- LI B, XU T CH. D2D-assisted dynamic task offloading scheme in mobile edge computing [J]. Radio Engineering, 2022, 52(12): 2101-2108.
- [16] 方韬, 杨昉, 陈佳馨. D2D 辅助移动边缘计算下的卸载策略优化[J]. 计算机科学, 2022, 49(S1): 601-605.
- FANG T, YANG Y, CHEN J X. Optimization of offloading decisions in D2D-assisted MEC networks[J]. Computer Science, 2022, 49(S1): 601-605.
- [17] 杜利俊, 李陶深, 黄翊芯, 等. 基于 SWIPT 的 D2D 通信辅助移动边缘计算任务卸载策略[J]. 广西科学, 2023, 30(4): 754-763.
- DU L J, LI T SH, HUANG Y X, et al. Task offloading strategy of D2D communication assisted MEC based on SWIPT [J]. Guangxi Sciences, 2023, 30(4): 754-763.
- [18] LI M L, ZHOU X B, QIU T, et al. Multi-relay assisted computation offloading for multi-access edge computing systems with energy harvesting[J]. IEEE Transactions on Vehicular Technology, 2021, 70(10): 10941-10956.
- [19] LI Y, XU G CH, YANG K, et al. Energy efficient relay selection and resource allocation in D2D-enabled mobile edge computing [J]. IEEE Transactions on Vehicular Technology, 2020, 69(12): 15800-15814.
- [20] 董思岐, 吴嘉慧, 李海龙, 等. 面向优先级任务的移动边缘计算资源分配方法[J]. 计算机工程, 2020, 46(3): 18-23.
- DONG S Q, WU J H, LI H L, et al. Resource allocation method for priority task in mobile edge computing [J]. Computer Engineering, 2020, 46(3): 18-23.
- [21] ZHANG L, ANSARI N. Optimizing the operation cost for UAV-aided mobile edge computing[J]. IEEE Transactions on Vehicular Technology, 2021, 70(6): 6085-6093.
- [22] WANG ZH Q, RONG H G. An energy-efficient multi-stage alternating optimization scheme for UAV-mounted mobile edge computing networks[J]. Computing, 2024, 106: 57-80.
- [23] 胡东旭, 甘敏. 基于块坐标下降法的神经网络学习算法[J]. 青岛大学学报(自然科学版), 2023, 36(3): 64-69, 75.
- HU D X, GAN M. Neural network learning algorithm based on block coordinate descent method [J]. Journal of Qingdao University (Natural Science Edition), 2023, 36(03): 64-69, 75.
- [24] 龙隆, 刘子辰, 陆在旺, 等. 移动边缘网络下服务缓存与资源分配联合优化策略[J]. 通信学报, 2023, 44(1): 64-74.
- LONG L, LIU Z CH, LU Z W, et al. Joint optimization strategy of service cache and resource allocation in mobile edge network [J]. Journal on Communications, 2023, 44(1): 64-74.
- [25] LIU ZH J, TAN X, WEN M W, et al. An energy-efficient selection mechanism of relay and edge computing in UAV-assisted cellular networks [J]. IEEE Transactions on Green Communications and Networking, 2021, 5(3): 1306-1318.
- [26] 薛建彬, 安亚宁. 基于边缘计算的新型任务卸载与资源分配策略[J]. 计算机工程与科学, 2020, 42(6): 959-965.
- XUE J B, AN Y N. A new task offloading and resource allocation strategy based on mobile edge computing [J]. Computer Engineering & Science, 2020, 42(6): 959-965.
- [27] LI Q P, ZHAO J H, GONG Y. Computation offloading and resource allocation for mobile edge computing with multiple access points [J]. IET Communications, 2019, 13(17): 2668-2677.

作者简介



占慧芳, 2021 年于江南大学获得学士学位, 现为江南大学硕士研究生, 主要研究方向为移动边缘计算。

E-mail: 6221927007@stu.jiangnan.edu.cn

Zhan Huifang received her B. Sc. degree from Jiangnan University in 2021. Now she is a

M. Sc. candidate at Jiangnan University. Her main research interest includes mobile edge computing.



李正权, 1998 年于吉林工业大学获得学士学位, 2000 年于上海理工大学获得硕士学位, 2003 年于上海交通大学获得博士学位, 现为江南大学教授, 主要研究方向为 6G 移动通信系统物理层关键技术和物联网关键技术。

E-mail: lzq722@jiangnan.edu.cn

Li Zhengquan received his B. Sc. degree from Jilin University of Technology in 1998, M. Sc. degree from University of Shanghai for Science and Technology in 2000, and Ph. D. degree from Shanghai Jiao Tong University in 2003, respectively. Now he is a professor at Jiangnan University. His main research interest includes physical layer in 6G mobile communication system and Internet of Things.



武贵路(通信作者), 2018 年于东南大学获得博士学位, 现为海南大学副教授, 主要研究方向为智能通信、绿色通信和智能计算。

E-mail: guilu.wu@hainanu.edu.cn

Wu Guilu (Corresponding author) received his Ph. D. degree from Southeast University in 2018. Now he is an associate professor at Hainan University. His main research interest includes intelligent communications, green communications and intelligent computing.



聂高峰, 2016 年毕业于北京邮电大学, 现为北京邮电大学讲师, 主要研究方向为 5G 组网技术研究和网络体系架构研究。

E-mail: niegaofeng@bupt.edu.cn

Nie Gaofeng graduated from Beijing University of Posts and Telecommunications in 2016. Now he is a lecturer at Beijing University of Posts and Telecommunications. His main research interests include 5G networking technology and network architecture.