

文章编号:1007-5321(2020)02-0110-06

DOI:10.13190/j.jbupt.2019-093

移动边缘计算中的时延和能耗均衡优化算法

景泽伟¹, 杨清海¹, 秦 猛²

(1. 西安电子科技大学 综合业务网理论及关键技术国家重点实验室, 西安 710071;
2. 鹏城实验室, 深圳 518055)

摘要: 为了提升移动边缘计算(MEC)网络中的任务卸载效用,提出了一种基于任务卸载增益最大化的时延和能耗均衡优化算法. 通过分析通信资源和计算资源对时延和能耗这 2 种性能指标的制约关系,将原问题分解为联合发射功率子信道分配子问题和 MEC 计算频率分配子问题. 通过 Karush-Kuhn-Tucker 条件,导出了最优的 MEC 计算频率闭式解. 此外,提出了一种基于二分法的发射功率分配算法和基于匈牙利二部图匹配的子信道分配算法. 仿真结果表明,提出的算法相比传统算法可以显著提升用户的任务卸载效用.

关 键 词: 移动边缘计算; 任务卸载; 子信道分配; 能耗; 任务完成时间

中图分类号: TP393

文献标志码: A

A Delay and Energy Tradeoff Optimization Algorithm for Task Offloading in Mobile-Edge Computing Networks

JING Ze-wei¹, YANG Qing-hai¹, QIN Meng²

(1. State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China;
2. Peng Cheng Laboratory, Shenzhen 518055, China)

Abstract: In order to enhance the task offloading utility in mobile-edge computing (MEC) networks, a delay and energy tradeoff optimization algorithm was proposed for maximizing the users' task offloading gains. The original optimization problem was decomposed into two sub-problems, i. e., the joint transmit power and sub-channel allocation sub-problem and the MEC computing frequency allocation sub-problem, upon the analysis of the restriction of communication and computation resources to the delay and energy consumption performances. The optimal MEC computing frequency was directly derived in the closed form by the Karush-Kuhn-Tucker condition. In addition, an efficient bisection method based transmit power allocation algorithm and a bipartite graph matching based sub-channel allocation algorithm were proposed, respectively. Numerical simulation results showed that, the proposed algorithm could improve the task offloading utility remarkably when compared with some traditional algorithms.

Key words: mobile-edge computing; task offloading; sub-channel allocation; energy consumption; task completion time

随着实时在线游戏、虚拟现实等新兴计算密集型 and 时延敏感型应用业务的不断涌现, 计算和能量

收稿日期: 2019-05-28

基金项目: 国家自然科学基金项目(61971327); 中国博士后科学基金项目(2019M663015, 2019TQ0210)

作者简介: 景泽伟(1993—), 男, 博士生.

通信作者: 杨清海(1976—), 男, 教授, E-mail: qhyang@xidian.edu.cn.

资源受限的智能用户终端正面临着严峻挑战. 为了缓解这一矛盾, 满足用户业务高带宽、低时延、低能耗的服务需求, 移动边缘计算 (MEC, mobile-edge computing) 技术已经演变为未来无线网络演进的一项关键技术^[1-2]. 区别于传统云计算, 由于 MEC 允许用户直接将计算任务卸载至接入网络边缘的 MEC 服务器中, 所以避免了日益严重的核心网络拥塞和漫长的任务传输时延问题. 随着无线接入网络与 MEC 系统的深度融合, MEC 将被赋予对无线网络参数的上下文感知能力, 从而使得接入网络的通信资源与 MEC 服务器计算资源的联合优化成为可能. 时延和能耗是衡量任务卸载和计算效用的两大性能指标, 任务在无线链路中的传输过程和在 MEC 服务器中的处理过程均会对这 2 种性能指标产生影响. 因此, 设计高效的通信和计算资源的联合优化算法是提升任务卸载和计算效用的关键^[3].

目前, 学术界已有一些关于移动边缘计算网络中通信和计算资源联合优化的研究工作. Wang 等^[4]分别以最小化任务完成时延和用户设备能耗为目标, 通过联合优化用户设备计算频率和上行发射功率, 探究了单个用户场景下的部分任务卸载问题. Mao 等^[5]利用李雅普诺夫优化技术研究了能量收集场景中的电池容量与能量收集速率对任务完成时延的影响. Wu 等^[6]以最小化任务完成时延为目标, 通过联合优化任务卸载和上行发射功率, 研究了非正交多址接入系统中的部分任务卸载问题. Chen 等^[7]提出了一种基于深度增强学习的在线任务卸载和能量分配决策算法, 以最小化长时期的任务完成时延和丢包率. Zhang 等^[8]通过将用户的任务卸载过程建模为一个势博弈过程, 提出了一个分布式的联合计算和通信资源分配算法.

与之前研究工作不同, 通过以最大化用户任务卸载的时延增益和能耗增益为目标, 提出了一种基于任务卸载增益最大化的时延和能耗均衡优化算法. 通过联合优化 MEC 计算频率、用户上行发射功率以及子信道, 最小化任务完成时间和用户能耗, 从而使得移动边缘计算中的任务卸载增益最大化.

1 移动边缘计算任务卸载模型

单个小区中 MEC 服务器服务多用户的场景如图 1 所示. MEC 服务器部署在基站侧, 为资源受限的移动用户终端提供计算任务卸载服务. 假设每个用户通过正交频分多址 (OFDMA, orthogonal frequen-

cy division multiple access) 技术接入基站, 用户集合采用 $\mathcal{N} = \{1, 2, \dots, N\}$ 表示, 整个系统频带被划分为 K 个子信道, 用集合 $\mathcal{K} = \{1, 2, \dots, K\}$ 表示, 定义子信道的关联矩阵为 $\mathbf{x} = [x_{n,k}]_{N \times K}$, 其中

$$x_{n,k} = \begin{cases} 1, & \text{若用户 } n \text{ 关联到子信道 } k, \\ 0, & \text{否则} \end{cases} \quad (1)$$

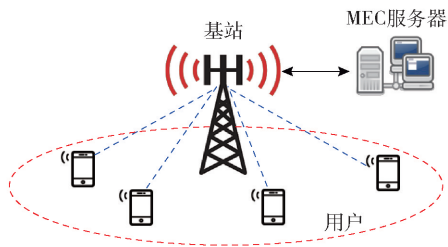


图 1 MEC 任务卸载系统框图

为避免同信道干扰, 假设每个子信道只能被分配给 1 个用户, 即 $\sum_{n \in \mathcal{N}} x_{n,k} \leq 1, \forall k \in \mathcal{K}$. 另外, 假设每个用户最多分配到 1 个子信道, 即 $\sum_{k \in \mathcal{K}} x_{n,k} \leq 1, \forall n \in \mathcal{N}$. 定义 $\mathbf{p} = [p_n]_N$ 为用户的上行发射功率向量, 用户 n 到基站的最高上行传输速率可用香农公式表示为

$$r_n = \sum_{k \in \mathcal{K}} x_{n,k} B_k \log(1 + p_n \Gamma_n^k) \quad (2)$$

其中: $\Gamma_n^k = |g_{n,k}|^2 / \sigma_k^2$ 表示基站接收端用户 n 在子信道 k 上的信号噪声比 (SNR, signal to noise ratio).

假设系统中的每个用户有一个计算任务需要处理, 表示为 $\mathcal{S}_n(\alpha_n, \beta_n)$, 其中 α_n 表示任务的总比特长度, β_n 为任务执行完成所需要的 CPU 周期数. 任务 \mathcal{S}_n 在移动终端计算时的本地计算时延和能量消耗分别表示为

$$t_n^l = \frac{\beta_n}{f_n^l} \quad (3)$$

$$e_n^l = \kappa (f_n^l)^2 \beta_n \quad (4)$$

其中: κ 为与 CPU 芯片结构相关的能量消耗因子^[4], f_n^l 为移动终端的本地计算频率. 如果此任务在 MEC 服务器完成, 则需要的上传时延和计算时延分别为

$$t_n^{\text{up}} = \frac{\alpha_n}{r_n} \quad (5)$$

$$t_n^s = \frac{\beta_n}{f_n^s} \quad (6)$$

其中 f_n^s 为 MEC 服务器分配给用户 n 的计算频率. 因此, 用户 n 的任务卸载所耗费的总时延为 $t_n^{\text{off}} = t_n^{\text{up}} + t_n^s$, 用户 n 的任务卸载所耗费的总能量为 $e_n^{\text{off}} =$

$\alpha_n \varepsilon_n p_n / r_n$, 其中 ε_n 为功率放大器效率因子。通常情况下, 由于任务计算输出的比特长度要远小于任务计算输入的比特长度, 所以省略了任务计算输出的下行传输时延和用户用于接收任务计算输出所消耗的能量^[8]。

2 通信和计算资源联合分配算法

2.1 问题描述

以网络总的任务卸载增益最大化为目标, 联合优化子信道、上行发射功率和 MEC 计算频率分配。首先, 将任务卸载增益定义为任务在边缘计算相比在本地计算所节约的时延和能耗加权量, 即

$$\mathcal{U}(\mathbf{x}, \mathbf{p}, \mathbf{f}^s) = \sum_{n \in \mathcal{N}'} \left(\lambda_n^e \frac{e_n^l - e_n^{\text{off}}}{e_n^l} + \lambda_n^t \frac{t_n^l - t_n^{\text{off}}}{t_n^l} \right) \quad (7)$$

其中: $\lambda_n^e, \lambda_n^t \in [0, 1]$, $\lambda_n^e + \lambda_n^t = 1$, 为用户 n 对于时延增益和能耗增益的偏好因子, 用户可以根据自身的剩余能量资源和任务时延要求灵活调节偏好因子, 从而达到对 2 种性能指标的不同侧重。 $\mathbf{f}^s = [f_n^s]_N$ 表示 MEC 计算频率分配向量。

至此, 网络任务卸载增益最大化问题可以由问题 \mathcal{P}_1 描述

$$\begin{aligned} \mathcal{P}_1: & \max_{\mathbf{x}, \mathbf{p}, \mathbf{f}^s} \mathcal{U}(\mathbf{x}, \mathbf{p}, \mathbf{f}^s) \\ \text{s. t.} \quad & \text{C1: } 0 \leq p_n \leq p_n^{\max}, \forall n \in \mathcal{N}' \\ & \text{C2: } \sum_{n \in \mathcal{N}'} x_{n,k} \leq 1, \forall k \in \mathcal{K} \\ & \text{C3: } \sum_{k \in \mathcal{K}} x_{n,k} \leq 1, \forall n \in \mathcal{N}' \\ & \text{C4: } \sum_{n \in \mathcal{N}'} f_n^s \leq f_s^{\max} \\ & \text{C5: } f_n^s \geq 0, \forall n \in \mathcal{N}' \\ & \text{C6: } x_{n,k} \in \{0, 1\}, \forall n \in \mathcal{N}', \forall k \in \mathcal{K} \end{aligned} \quad (8)$$

其中: C1 为用户的上行发射功率约束; C2、C3 和 C6 保证了每个子信道分配的正交性和用户的单信道接入约束; C4 和 C5 为 MEC 计算频率分配约束。不难发现, 问题 \mathcal{P}_1 是一个非凸的混合整数规划问题^[9], 且其复杂度随着用户和子信道的数目成指数性增长。为了使问题可解, 在下文中, 通过分析问题 \mathcal{P}_1 的结构, 将该问题分解为 2 个相互独立的子问题, 即联合子信道和功率分配子问题及 MEC 计算频率分配子问题, 进而提出了一种最优的通信和计算资源联合分配算法。

2.2 求解方案

首先, 重写问题 \mathcal{P}_1 的目标函数为

$$\mathcal{U}(\mathbf{x}, \mathbf{p}, \mathbf{f}^s) = \sum_{n \in \mathcal{N}'} (\lambda_n^e + \lambda_n^t) - \Lambda(\mathbf{x}, \mathbf{p}, \mathbf{f}^s) \quad (9)$$

$$\text{其中 } \Lambda(\mathbf{x}, \mathbf{p}, \mathbf{f}^s) = \sum_{n \in \mathcal{N}'} \left(\frac{\lambda_n^e e_n^{\text{off}}}{e_n^l} + \frac{\lambda_n^t t_n^{\text{off}}}{t_n^l} \right).$$

式(9)中的第 1 项是一个常数项, 去除第 2 项的负号, 可以将任务卸载增益最大化问题 \mathcal{P}_1 等价地转化为时延和能耗开销最小化问题 \mathcal{P}_2

$$\begin{aligned} \mathcal{P}_2: & \min_{\mathbf{x}, \mathbf{p}, \mathbf{f}^s} \Lambda(\mathbf{x}, \mathbf{p}, \mathbf{f}^s) \\ \text{s. t.} \quad & \text{C1, C2, C3, C4, C5, C6} \end{aligned} \quad (10)$$

由于变量 $\{\mathbf{x}, \mathbf{p}\}$ 和 \mathbf{f}^s 在问题 \mathcal{P}_2 的目标函数和约束中均可分, 所以它可以被分解为 2 个独立的子问题, 即联合子信道和功率分配子问题及 MEC 计算频率分配子问题。其中, 联合子信道和功率分配子问题可以表示为

$$\begin{aligned} \mathcal{P}_3: & \min_{\mathbf{x}, \mathbf{p}} \sum_{n \in \mathcal{N}'} \frac{\phi_n + \varphi_n p_n}{\sum_{k \in \mathcal{K}} x_{n,k} B_k \log(1 + p_n \Gamma_n^k)} \\ \text{s. t.} \quad & \text{C1, C2, C3, C6} \end{aligned} \quad (11)$$

其中: $\phi_n = \lambda_n^t \alpha_n / t_n^l$, $\varphi_n = \lambda_n^e \varepsilon_n \alpha_n / e_n^l$ 。为了求解问题 \mathcal{P}_3 , 首先假设给定任务 \mathcal{A}_n 在子信道 k 上传输, 即 $x_{n,k} = 1$ 。同时定义函数

$$\vartheta_n(p_{n,(k)}) = \frac{\phi_n + \varphi_n p_{n,(k)}}{B_k \log(1 + p_{n,(k)} \Gamma_n^k)} \quad (12)$$

其中 $p_{n,(k)}$ 表示任务 \mathcal{A}_n 在子信道 k 上传输时的用户发射功率。在此条件下, 问题 \mathcal{P}_3 转变为求解 N 个相互独立的功率分配子问题 \mathcal{P}_4

$$\mathcal{P}_4: \min_{p_{n,(k)}} \vartheta_n(p_{n,(k)}) \quad (13)$$

$$\text{s. t. } 0 \leq p_{n,(k)} \leq p_n^{\max}$$

子问题 \mathcal{P}_4 是一个非凸的分式规划问题, 在求解其之前, 首先给出如下引理。

引理 1 问题 \mathcal{P}_4 的目标函数是拟凸的, 并且该问题存在唯一的全局最优解。

证明 类似证明可以参考文献[10]。

根据引理 1 和一阶最优性条件, 问题 \mathcal{P}_4 的最优解 $p_{n,(k)}^*$ 要么在约束边界取到, 要么满足 $\vartheta'_n(p_{n,(k)}^*) = 0$, 即

$$\begin{aligned} \Omega_n(p_{n,(k)}) &= \varphi_n \log(1 + p_{n,(k)} \Gamma_n^k) - \\ & \frac{\Gamma_n^k (\phi_n + \varphi_n p_{n,(k)})}{(1 + p_{n,(k)} \Gamma_n^k) \ln 2} = 0 \end{aligned} \quad (14)$$

又因为 $\Omega'_n(p_{n,(k)}) > 0$, $\Omega_n(0) < 0$, 所以提出了一种基于二分法的高效功率分配算法。

算法 1 基于二分法的高效功率分配算法

1 根据式(14)计算 $\Omega_n(p_n^{\max})$;

2 如果 $\Omega_n(p_n^{\max}) \leq 0$ 则

3 $p_{n,(k)}^* = p_n^{\max}$;

4 否则

5 初始化收敛容限 τ ;

6 设置辅助变量 $p' = 0, p'' = p_n^{\max}$;

7 重复

8 设 $p_{n,(k)}^* = (p' + p'')/2$;

9 如果 $\Omega_n(p_{n,(k)}^*) \leq 0$ 则

10 设 $p' = p_{n,(k)}^*$;

11 否则

12 设 $p'' = p_{n,(k)}^*$;

13 直到 $p'' - p' \leq \tau$

14 设 $p_{n,(k)}^* = (p' + p'')/2$;

15 输出 $p_{n,(k)}^*$;

将算法1中得到的最优功率分配 $p_{n,(k)}^*$ 代入问题 $\mathcal{P}3$, 可以得到如下问题

$$\mathcal{P}5: \min \sum_x \sum_{n \in \mathcal{N}} \sum_{k \in \mathcal{K}} x_{n,k} \vartheta_n(p_{n,(k)}^*) \quad (15)$$

s. t. C2, C3, C6

问题 $\mathcal{P}5$ 是一个典型的二部图匹配问题, 因而可以用经典的匈牙利算法进行高效求解^[11].

MEC 计算频率分配子问题表示如下

$$\mathcal{P}6: \min \sum_{f^s} \sum_{n \in \mathcal{N}} \frac{\eta_n}{f_n^s} \quad (16)$$

s. t. C5, C6

其中 $\eta_n = \lambda_n^t f_n^l$. 显然, 问题 $\mathcal{P}6$ 是一个具有线性不等式约束的凸优化问题, 所以可以用 Karush-Kuhn-Tucker (KKT) 条件和拉格朗日乘子法^[9,12] 对其进行求解. 具体地, 引入拉格朗日乘子 δ , 问题 $\mathcal{P}6$ 的拉格朗日函数可以表示为

$$\mathcal{L}(f^s, \delta) = \sum_{n \in \mathcal{N}} \frac{\eta_n}{f_n^s} + \delta \left(\sum_{n \in \mathcal{N}} f_n^s - f_s^{\max} \right) \quad (17)$$

拉格朗日对偶函数表示为

$$\mathcal{G}(\delta) = \min_{f^s \geq 0} \mathcal{L}(f^s, \delta) \quad (18)$$

问题 $\mathcal{P}6$ 的对偶问题为 $\max_{\delta \geq 0} \mathcal{G}(\delta)$. 通过 KKT 条件, 可以得到 $f_n^* = \sqrt{\eta_n / \delta}$, $\forall n \in \mathcal{N}$. 将 f_n^* 代入拉格朗日对偶问题, 求得拉格朗日乘子的最优值为

$$\delta^* = \left(\sum_{n \in \mathcal{N}} \sqrt{\eta_n / f_s^{\max}} \right)^2 \quad (19)$$

因此最优 MEC 计算频率分配为

$$f_n^* = \frac{\sqrt{\eta_n} f_s^{\max}}{\sum_{n \in \mathcal{N}} \eta_n}, \quad \forall n \in \mathcal{N} \quad (20)$$

通过以上分析, 可以获得每个用户在任务卸载时的最优发射功率、子信道和 MEC 计算频率. 为了避免任意一个用户的任务卸载增益为负值, 每个用户可以根据式(21)进行任务计算模式选择.

$$b_n = \begin{cases} 1, & \text{若 } \mathcal{U}_n^*(\mathbf{x}_n^*, \mathbf{p}_n^*, f_n^*) > 0 \\ 0, & \text{若 } \mathcal{U}_n^*(\mathbf{x}_n^*, \mathbf{p}_n^*, f_n^*) \leq 0 \end{cases} \quad (21)$$

其中 $\mathbf{b} = [b_n]_N$ 为用户的任务计算模式选择向量, 当 $b_n = 1$ 时, 用户 n 采用 MEC 边缘计算模式; 当 $b_n = 0$ 时, 用户 n 采用本地计算模式. $\mathcal{U}_n^*(\mathbf{x}_n^*, \mathbf{p}_n^*, f_n^*)$ 为用户 n 的最优任务卸载增益.

2.3 算法步骤

通过以上分析, 提出了一种最优的联合通信和计算资源分配 (JCCRA, joint communication and computation resource allocation) 算法, 有以下步骤.

1) 网络参数输入. 初始化任务模型 $\mathcal{A}_n(\alpha_n, \beta_n)$ 、上行信道增益 $|g_{n,k}|^2$ 和噪声功率 σ_k^2 、计算能量消耗因子 κ 、功率放大器效率因子 ε_n 、用户能耗和时延偏好因子 λ_n^e 和 λ_n^t 、本地计算频率 f_n^l 等网络参数.

2) 用户上行功率分配. 采用算法1提出的二分法获得每个用户在每个子信道上的最优功率分配 $p_{n,(k)}^*$.

3) 最优子信道分配. 通过步骤2)得到的最优功率分配 $p_{n,(k)}^*$, 获得每个用户在每个子信道上的匹配权重 $\vartheta_n(p_{n,(k)}^*)$, 通过匈牙利算法进行最优的子信道分配.

4) 最优 MEC 计算频率分配. 通过式(20)获得每个用户的最优 MEC 计算频率分配.

5) 任务计算模式选择. 通过步骤2)~5)所获得的最优用户上行功率、子信道和 MEC 计算频率分配, 计算每个用户的最优任务卸载增益 $\mathcal{U}_n^*(\mathbf{x}_n^*, \mathbf{p}_n^*, f_n^*)$, 通过式(21)进行任务计算模式选择.

6) 资源优化结果输出: $(\mathbf{b}^*, \mathbf{x}_n^*, \mathbf{p}_n^*, \mathbf{f}^{s,*})$.

2.4 算法复杂度分析

由于最优的 MEC 计算频率分配可由 KKT 条件闭式导出, 所以提出的 JCCRA 算法复杂度主要取决于联合子信道和功率分配.

1) 功率分配. 为了获取匹配权重 $\vartheta_n(p_{n,(k)}^*)$, 需要针对每个用户和子信道运用二分法求解最优功率分配 $p_{n,(k)}^*$. 二分法复杂度为 $O(\text{lb}(p_n^{\max}/\tau))$, 步骤2)总的复杂度为 $O\left(\sum_{n \in \mathcal{N}} K \text{lb}(p_n^{\max}/\tau)\right)$.

2) 子信道分配. 由文献[11]可知, 运用匈牙利

算法进行子信道分配的复杂度为 $O(NK^2 + N\log N)$.

结合 1) 和 2), JCCRA 算法总的复杂度为 $O\left(\sum_{n \in N} K \log(p_n^{\max}/\tau) + NK^2 + N\log N\right)$. 不难看出, JCCRA 算法的计算复杂度为多项式复杂度, 适用于用户数目比较多的场景.

3 仿真结果

不失一般性, 考虑一个半径为 1 km 的圆形服务区域, 基站和 MEC 服务器同时部署在圆心, N 个用户均匀分布在该服务区域内. 假设每个用户具有相同的任务需要计算, 任务模型为 $\alpha_n = 5 \times 10^5$ bit, $\beta_n = 1 \times 10^9$ CPU 周期. 用户到基站的路径损耗模型采用 $L = 128.1 + 37.6 \lg d$ dB, 阴影衰落服从均值为零, 方差为 8 dB 的对数正态分布^[13]. 设置子信道的数目和用户数目相同, 并设置系统总的带宽为 20 MHz, 则每个子信道的带宽为 $B_k = 20/K$ MHz. 仿真中所用到的其他参数如下: $\kappa = 1 \times 10^{-26}$, $\varepsilon_n = 1/0.2$, $f_n^l = 1 \times 10^9$ Hz, $f_s^{\max} = 20 \times 10^9$ Hz, $p_n^{\max} = 23$ dBm, $\sigma_k^2 = -100$ dBm.

为了评估算法性能, 分别采取 2 个对比算法与所提算法 JCCRA 进行比较, 即基于最大 SNR 匹配的子信道分配算法 (maxSNR) 和基于穷举的最优子信道分配算法. 在 maxSNR 算法中, 每个用户采用最大发射功率传输, 并采用基于最大 SNR 匹配的方式接入子信道; 在穷举算法中, 系统将采用穷举的方式进行最优子信道分配.

图 2 说明了网络任务卸载增益随系统用户数目增大时的变化趋势. 本次仿真中, 将每个用户的偏好因子设置为 $\lambda_n^l = 0.3$ 和 $\lambda_n^e = 0.7$. 从图中可以看出, 提出的 JCCRA 算法在性能上明显优于 maxSNR 算法. 可观察到, 2 种算法的曲线均随用户数目的增多先上升后下降. 这是因为当用户数

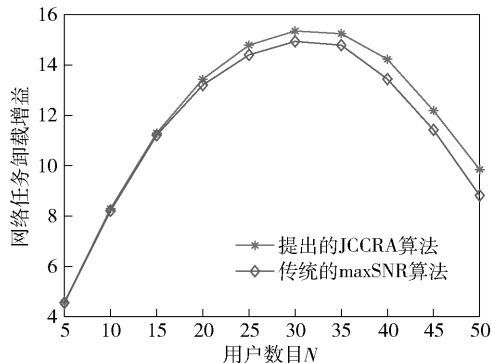


图2 网络任务卸载增益随用户数目的变化趋势

目比较少时, 网络的频谱资源和计算资源竞争不激烈, 因而用户的增多可以增加任务卸载增益. 然而当用户增加到一定数目时, 网络资源变得紧缺, 进而造成更大的任务卸载时延和能量消耗, 使得任务卸载增益下降.

图 3 说明了网络任务卸载增益随用户最大发射功率增大时的变化趋势. 本次实验中, 同时比较了穷举算法、提出的 JCCRA 算法和传统的 maxSNR 算法的性能. 由于穷举算法的复杂度极高, 故将用户数目设置为 $N=8$, 每个用户的偏好因子设置为 $\lambda_n^l = 0.3$ 和 $\lambda_n^e = 0.7$. 从图 3 可以看出, 提出的 JCCRA 算法和最优的穷举算法性能基本相同, 且 3 种算法的曲线均随用户最大发射功率增大而上升. 这是因为用户最大发射功率的增大使得每个用户有更多的机会采取最优的发射功率进行传输, 从而降低时延和能量开销, 进而增大网络任务卸载增益.

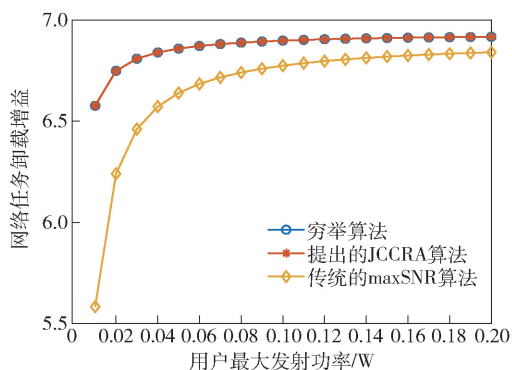


图3 网络任务卸载增益随用户最大发射功率的变化趋势

图 4 说明了网络平均时延和平均能耗随用户偏好因子变化时的均衡趋势. 在本次仿真中, 将用户数目设置为 $N=20$, 同时设 $\lambda_n^e = 1 - \lambda_n^l$. 从图 4 可以看出, 随着用户对时延偏好的增加, 网络的平均时延

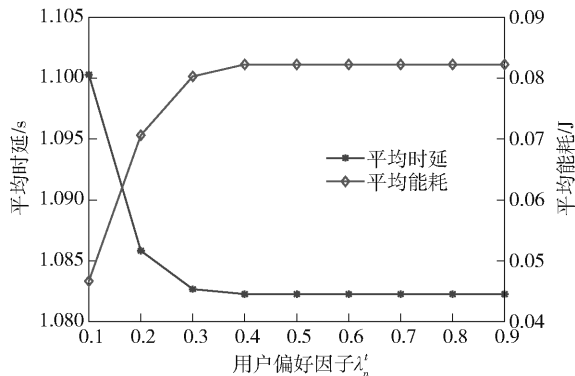


图4 平均时延和平均能耗的均衡曲线

逐渐减小,而网络的平均能耗逐渐增大,同时当用户的时延偏好因子 $\lambda'_n \geq 0.4$ 时,网络的平均时延和平均能耗开始保持稳定. 这是因为,当 λ'_n 数值比较小时, λ'_n 的增大使得系统以牺牲能耗性能来降低时延;然而当 λ'_n 数值比较大时,由于用户的发射功率限制,继续增大 λ'_n 并不能降低系统平均时延和增大平均能耗.

4 结束语

由于在移动边缘计算任务卸载的过程中,任务的卸载过程和计算过程会受到资源受限的上行信道和 MEC 服务器的双重挑战,从而产生额外的时延和能量开销. 为缓解这一矛盾,提出了一种基于任务卸载增益最大化的联合通信和计算资源分配算法. 通过联合优化用户上行发射功率、子信道分配和 MEC 计算频率,导出了针对每个用户的最优任务卸载增益,从而指导用户进行任务计算模式的选择. 仿真结果表明,提出的算法在性能上要优于传统对比算法.

参考文献:

- [1] Mach P, Becvar Z. Mobile edge computing: a survey on architecture and computation offloading [J]. IEEE Commun Surveys Tuts, 2017, 19(3): 1628-1656.
- [2] Wang C, Liang C, Yu F R, et al. Computation offloading and resource allocation in wireless cellular networks with mobile edge computing [J]. IEEE Trans Wireless Commun, 2017, 16(8): 4924-4938.
- [3] Mao Y, You C, Zhang J, et al. A survey on mobile edge computing: the communication perspective [J]. IEEE Commun Surveys Tuts, 2017, 19(4): 2322-2358.
- [4] Wang Y, Sheng M, Wang X, et al. Mobile-edge computing: partial computation offloading using dynamic voltage scaling [J]. IEEE Trans Commun, 2016, 64(10): 4268-4282.
- [5] Mao Y, Zhang J, Letaief K B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices [J]. IEEE J Sel Areas Commun, 2016, 34(12): 3590-3605.
- [6] Wu Y, Ni K, Zhang C, et al. NOMA assisted multi-access mobile edge computing: a joint optimization of computation offloading and time allocation [J]. IEEE Trans Veh Technol, 2018, 67(12): 12244-12258.
- [7] Chen X, Zhang H, Wu C, et al. Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning [J]. IEEE IoT J, 2019, 6(3): 4005-4018.
- [8] Zhang J, Xia W, Yan F, et al. Joint computation offloading and resource allocation optimization in heterogeneous networks with mobile edge computing [J]. IEEE Access, 2018(6): 19324-19337.
- [9] Boyd S, Vandenberghe L. Convex optimization [M]. Cambridge: Cambridge Univ Press, 2004.
- [10] Liu R, Sheng M, Wu W. Energy-efficient resource allocation for heterogeneous wireless network with multi-homed user equipment [J]. IEEE Access, 2018(6): 14591-14601.
- [11] Kuhn H W. The Hungarian method for the assignment problem [J]. Nav Res Log, 2005, 52(1): 7-21.
- [12] 吴伟华, 杨清海. 面向无线自主网络高效信息扩散的资源分配 [J]. 北京邮电大学学报, 2016, 39(2): 88-92.
- Wu Weihua, Yang Qinghai. Efficient resource allocation for information diffusion in wireless autonomic network [J]. Journal of Beijing University of Posts and Telecommunications, 2016, 39(2): 88-92.
- [13] Liu D, Wang L. Distributed energy efficient fair user association in massive MIMO enabled HetNets [J]. IEEE Commun Lett, 2015, 19(10): 1770-1773.