

## 基于自适应遗传算法的 MEC 任务卸载及资源分配\*

闫伟, 申滨, 刘笑笑

(重庆邮电大学 通信与信息工程学院, 重庆 400065)

**摘要:** 移动边缘计算(Mobile Edge Computing, MEC)是指将具有计算和存储能力的云服务器部署在移动网络边缘, 从而为用户带来低能耗、低时延的优势。针对单小区-多用户场景下的计算任务卸载问题, 设计系统总开销优化函数, 并提出任务卸载决策和资源分配的联合优化问题。首先采用自适应遗传算法制定卸载决策及后续的更新操作, 在每一次卸载决策更新的情况下将原问题分解为功率分配和计算资源分配两个子问题; 然后根据凸优化及准凸优化理论, 利用二分搜索法和拉格朗日乘子法分别求出功率分配和计算资源分配的最优解。仿真结果表明, 所提出的方案在保证用户时延约束的同时, 降低了系统总开销, 有效地提升系统的性能及用户服务质量。

**关键词:** 移动边缘计算; 自适应遗传算法; 任务卸载; 资源分配

中图分类号: TN929.5

文献标识码: A

DOI: 10.16157/j.issn.0258-7998.200113

中文引用格式: 闫伟, 申滨, 刘笑笑. 基于自适应遗传算法的 MEC 任务卸载及资源分配[J]. 电子技术应用, 2020, 46(8): 95-100.

英文引用格式: Yan Wei, Shen Bin, Liu Xiaoxiao. Offloading and resource allocation of MEC based on adaptive genetic algorithm[J]. Application of Electronic Technique, 2020, 46(8): 95-100.

## Offloading and resource allocation of MEC based on adaptive genetic algorithm

Yan Wei, Shen Bin, Liu Xiaoxiao

(School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications,  
Chongqing 400065, China)

**Abstract:** Mobile Edge Computing(MEC) provides its users with low energy consumption and low latency by providing IT service environment and cloud computing capabilities at the edge of mobile networks. This paper addresses the computing tasks offloading in a single-cell MEC scenario, and designs the optimization function of system total cost, and proposes a joint optimization problem of offloading decision and resource allocation. Firstly, the adaptive genetic algorithm is used to make the offloading decision and subsequent update operations. Based on updating the offloading decision, the two sub-problems of power allocation and computing resource allocation are solved correspondingly. The binary search method and the Lagrange multiplier method are used to obtain the optimal solutions for power allocation and computation resource allocation, respectively. Simulation results show that the proposed scheme can meet the goal of minimizing total system overhead for users under the requirement of delay constraints, and effectively improve system performance and user service quality.

**Key words:** mobile edge computing(MEC); adaptive genetic algorithm; task offloading; resource allocation

## 0 引言

近年来,随着移动互联网和智能终端的快速发展,越来越多的新型应用出现在人们的日常生活当中,例如,人脸识别、虚拟现实和增强现实等<sup>[1-2]</sup>。然而,由于移动设备的计算能力和电池续航能力有限,难以有效地处理这些应用。若将计算任务卸载到距离用户较远的云服务器可能会产生更高的时延从而影响任务的执行效率。针对以上问题,业界提出了移动边缘计算的概念。移动边缘计算(Mobile Edge Computing, MEC)是将具有存储和

计算能力的网络设备实体部署在移动网络边缘,从而为移动网络提供 IT 服务环境和计算能力<sup>[3]</sup>。MEC 的关键技术主要包括任务卸载和资源分配两个方面,任务卸载是指将计算密集型或延迟敏感型任务卸载到资源相对丰富的计算机或服务器中执行,以解决移动设备在存储、计算等方面存在的缺陷<sup>[4-5]</sup>。此外,在多个用户将其各自的计算任务从本地卸载到 MEC 服务器进行处理时,由此涉及有限的 MEC 服务器资源在各个用户之间的分配问题。因此,MEC 服务器的资源分配致力于解决移动设备在实现卸载后如何高效公平地分配资源以实现任务处理的问题。

\* 基金项目: 国家自然科学基金项目(61571073)

# 通信与网络 Communication and Network

针对 MEC 的任务卸载和资源分配问题,文献[6]考虑了云服务器计算资源受限的约束,提出了多用户场景下基于在线和离线的任务卸载决策算法,对计算任务进行云端处理和本地处理的划分,并分别处理。文献[7]提出了一种考虑前向链路和后向链路联合优化的卸载决策问题,并通过改进的人工鱼群算法对任务卸载过程中所产生的能量消耗进行优化。文献[8]分别研究了在单小区和多小区场景下的任务卸载和资源分配问题,设计了系统总开销函数(时延-能耗加权),并将用户设备的电池剩余能耗引入到加权因子当中,通过凸优化理论与迭代搜索相结合方式的寻找最优解。然而文献[6]、[7]仅关注用户任务卸载决策,缺少优化资源分配或者考虑资源约束;文献[8]只考虑了本地设备计算资源及无线资源的分配,而忽略了 MEC 服务器有限计算资源对任务卸载的影响。因此,无法完全保证能够高效地利用 MEC 服务场景的对于众多用户而言仍显得稀缺有限的无线资源和计算资源。

针对上述问题,本文提出首先采用自适应遗传算法获取卸载决策及后续的更新操作。在每一次卸载决策更新的情况下,将原问题分解为上行链路功率分配和计算资源分配两个子问题,然后根据凸优化和准凸优化理论,利用二分搜索法和拉格朗日乘子法分别求出功率分配和计算资源分配的最优解。最后,通过迭代求解得到最优的卸载决策和资源分配结果。仿真结果表明,所提出的方案在保证用户时延约束的同时,降低了系统总开销,有效地提升系统的性能。

## 1 系统模型

### 1.1 网络模型

考虑一个应用 MEC 的单小区-多用户网络模型,如图 1 所示。所有移动用户的集合表示为  $\mathcal{N}=\{1, 2, \dots, K\}$ 。其中,MEC 服务器部署在基站(Base Station)附近,MEC 服务器可以处理来自用户卸载的计算任务。假设每个用户  $k$  都有一个计算任务  $\sigma_k=\{b_k, s_k, T_k^{\max}\}$  需要执行,  $b_k$  表示输入数据量大小,  $s_k$  表示完成该计算任务所需的 CPU 周期,  $T_k^{\max}$  表示用户  $k$  所能容忍的最大时延。本文考虑的是全部卸载方案(即任务不可再分仅能被整体进行处理)。令  $a_k \in \{0, 1\}$  表示用户  $k$  的卸载决策。当  $a_k=1$  时,用户  $k$  选择将任务卸载到 MEC 服务器执行;否则,用户选择本地执行任务。将选择卸载计算任务的用户集合记为  $\mathcal{N}^e$ ,集合的势为  $|\mathcal{N}^e|=K^e=\sum_{k=1}^K a_k$ ,表示其包含的用户数。类似地,将选择本地执行的用户集合记为  $\mathcal{N}^l$ ,集合的势为  $|\mathcal{N}^l|=K^l=K-K^e$ 。

### 1.2 通信模型

在本模型中假设用户采用彼此正交的上行传输信道,因此用户在计算卸载过程中彼此之间不会产生同频

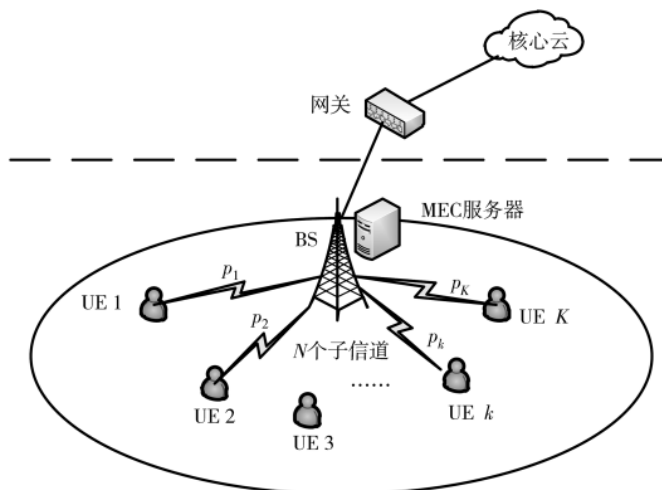


图 1 网络模型

干扰。则用户  $k$  以  $p_k$  功率传输时的上行链路速率  $R_k(p_k)$  表示为:

$$R_k(p_k)=W_k \log_2 \left( 1 + \frac{p_k h_k}{\sigma_{0,k}^2} \right) \quad \forall k \in \mathcal{N} \quad (1)$$

其中,  $W_k$  表示用户  $k$  对应的上行链路带宽,  $B=\sum_{k=1}^K W_k$  表示系统上行链路总带宽,  $h_k$  表示用户  $k$  到 BS 的上行链路功率增益,  $\sigma_{0,k}^2$  表示 BS 对应于用户  $k$  的上行链路噪声功率。

### 1.3 计算模型

(1)当用户选择将任务本地执行时,令  $f_k^l$  表示用户  $k$  的本地计算能力(CPU 周期数),本地计算任务完成的时间  $T_k^l$  为:

$$T_k^l = \frac{s_k}{f_k^l} \quad (2)$$

由文献[8]可知,本地计算时的能量消耗为  $E_k^l$ :

$$E_k^l = \kappa s_k (f_k^l)^2 \quad (3)$$

其中,能耗系数  $\kappa$  是与移动设备的芯片结构相关的常数。

本地计算的开销包括能量消耗和本地执行时延加权的无单位组合,根据式(2)、式(3),可表示为:

$$z_k^l = \gamma_k^e E_k^l + \gamma_k^l T_k^l \quad (4)$$

其中,定义  $\gamma_k^e$  和  $\gamma_k^l$  分别表示用户  $k$  在任务执行过程中对能耗和时延的权重系数,满足  $\gamma_k^e, \gamma_k^l \in \{0, 1\}$ ,  $\gamma_k^e + \gamma_k^l = 1$ 。用户可以根据任务对设备能耗和任务完成时延的要求动态地调整。

(2)当用户  $k$  选择将任务卸载到 MEC 服务器执行时,MEC 服务器会为其分配计算资源  $f_k$  以便及时处理该任务,令  $T_k^e(f_k, p_k)$  表示任务在对应于用户远端的 MEC 服务

# 通信与网络 Communication and Network

器的处理时延,其可表示为:

$$T_k^c(f_k, p_k) = T_k^{ul}(p_k) + T_k^{exe}(f_k) \quad (5)$$

其中,  $T_k^{ul}(p_k)$  和  $T_k^{exe}(f_k)$  分别表示任务的输入数据通过上行链路上传至 MEC 服务器以及任务在 MEC 服务器处理所对应的时延,并且有:

$$T_k^{ul}(p_k) = \frac{b_k}{W_k \log_2(1 + \omega_k p_k)} \quad (6)$$

其中,  $\omega_k = h_k / \sigma_{0,k}^2$ 。任务处理时延  $T_k^{exe}(f_k)$ , 表示为:

$$T_k^{exe}(f_k) = \frac{s_k}{f_k} \quad (7)$$

用户  $k$  选择将任务卸载到远端 MEC 服务器处执行的能耗  $E_k^c$ , 表示为:

$$E_k^c(p_k) = \frac{p_k}{\zeta} T_k^{ul}(p_k) = \frac{p_k}{\zeta} \frac{b_k}{W_k \log_2(1 + \omega_k p_k)} \quad (8)$$

其中,  $\zeta$  是设备传输功率放大器的效率。

因此,当用户  $k$  选择将任务卸载到 MEC 服务器进行处理时,其总开销包括能量消耗和远端执行时延,根据式(5)~式(8),可表示如下:

$$z_k^c = \gamma_k^e E_k^c(p_k) + \gamma_k^l T_k^c(f_k, p_k) \quad (9)$$

由于在 MEC 服务器执行后的结果数据量较小,因此本文忽略了计算结果在返回到用户过程中的时延和能耗。

## 2 问题形成

通过上述分析可得,用户的代价函数表示为:

$$z_k = (1 - a_k) z_k^l + a_k z_k^c \quad (10)$$

本文的主要目标是最小化所有用户的代价函数。根据上面的讨论,在单小区-多用户的场景下,计算任务卸载的优化目标函数表示:

$$\begin{aligned} \min_{A, F, P} Z &= \sum_{k=1}^K (1 - a_k) z_k^l + a_k z_k^c \\ \text{s.t. } C1: &a_k \in \{0, 1\}, \forall k \in \mathcal{K} \\ C2: &0 < p_k \leq p_{\max}, \forall k \in \mathcal{K}^c \\ C3: &f_k > 0, \forall k \in \mathcal{K}^c \\ C4: &\sum_{k \in \mathcal{K}^c} f_k \leq f_{\max} \\ C5: &\sum_{k \in \mathcal{K}^c} a_k W_k \leq B \end{aligned} \quad (11)$$

其中,  $C1$  表示用户的卸载决策;  $C2$  表示卸载用户传输时的上行链路功率不得超过最大传输功率;  $C3$  表示 MEC 服务器分配给卸载用户的计算资源是非负的;  $C4$  表示分配给卸载用户的计算资源不能超过 MEC 服务器所拥有的最大计算资源;  $C5$  表示由于系统带宽限制,假设

各个用户上传时分配的链路带宽皆为  $W_k$ , 则有  $\sum_{k \in \mathcal{K}^c} a_k \leq \lfloor B/W_k \rfloor = N$ , 表示小区内仅可同时允许不超过  $N$  个用户

进行数据上传。

## 3 卸载决策与资源分配的联合优化

由于整数约束与连续变量同时存在,因此式(11)所描述的优化问题属于混合整数非线性规划问题,求解难度大。因此,本文将原问题分解为卸载决策、上行链路功率分配和计算资源分配 3 个子问题,并依次进行求解。

### 3.1 基于自适应遗传算法的卸载决策

由于式(11)是涉及卸载决策  $A$  的 0-1 规划问题,因此考虑采用自适应遗传算法进行求解,该算法在传统遗传的基础上进行了改进,通过设置随适应度值做动态变化的交叉概率和变异概率,从而避免了算法陷入局部最优解,提高了算法精度。具体步骤如下:

(1) 编码和初始化种群。本文使用二进制编码方式对基因进行编码,指定种群规模  $I$ 。使用  $K$  个  $\{0, 1\}$  二进制位(基因)随机生成第  $i$  条染色体为  $A_i = \{a_1^i, a_2^i, \dots, a_K^i\}$ ,  $i \in \{1, 2, \dots, I\}$ 。得到初始的染色体种群  $I(0)$ 。

(2) 适应度评估。传统的遗传算法是选取适应度值较大的个体遗传到下一代。因此本文选取用户总开销的倒数作为评价染色体优劣的适应度函数,表达式如下所示:

$$\text{fitness} = \frac{1}{\sum_{k=1}^K (1 - a_k) z_k^l + a_k z_k^c} \quad (12)$$

(3) 选择操作。采用基于适应度比例选择的轮盘赌选择法从父代中选取一部分优良个体。

(4) 交叉和变异操作。本文根据交叉概率和变异概率来进行单点交叉和基本位变异操作,产生具有新基因的染色体。其中根据个体的适应值做动态变化的交叉概率  $P_c$  和变异概率  $P_m$  表示为:

$$P_c = \begin{cases} \frac{\beta_1(f_{\max} - f)}{f_{\max} - f_{\text{avg}}}, & f \geq f_{\text{avg}} \\ \beta_2, & f < f_{\text{avg}} \end{cases} \quad (13)$$

$$P_m = \begin{cases} \frac{\beta_3(f_{\max} - f')}{f_{\max} - f_{\text{avg}}}, & f' \geq f_{\text{avg}} \\ \beta_4, & f' < f_{\text{avg}} \end{cases} \quad (14)$$

其中,  $f_{\max}$  表示种群个体中的最大适应度值;  $f_{\text{avg}}$  表示整个种群的平均适应度值;  $f$  表示种群中选择交叉操作的两个个体中较大的适应度值;  $f'$  表示种群中选择变异操作个体的适应度值;  $\beta_1, \beta_2, \beta_3$  和  $\beta_4$  为常数。

(5) 遗传终止条件。通过上述的遗传操作,能够得到新一代的种群。然后根据式(12)计算出新一代种群个体的适应度值。若该种群的最大适应度值与平均适应度值变化不大,或者已经达到预设的最大迭代次数,则算法停止,输出最优适应度值以及所对应的卸载决策;否则继续执行算法,直到满足停止条件。

### 3.2 问题分解

通过上述自适应遗传算法制定用户的卸载决策后可得到卸载用户集合  $\mathcal{K}^c$ , 原目标函数(11)可以简化为:

# 通信与网络 Communication and Network

$$\begin{aligned} \min_{P, F} \quad & \sum_{k \in \mathcal{K}} \gamma_k^c E_k^c + \gamma_k^t T_k^c \\ = \quad & \sum_{k \in \mathcal{K}} \frac{\eta_k p_k + \tau_k}{\log_2(1 + \omega_k p_k)} + \frac{\gamma_k^t s_k}{f_k} \\ \text{s.t.} \quad & C2, C3, C4 \end{aligned} \quad (15)$$

其中,  $\tau_k = \frac{\gamma_k^t b_k}{W_k}$ ,  $\eta_k = \frac{\gamma_k^c b_k}{W_k \zeta}$ 。从式(15)可以看出, 原优化问题可以分解成上行链路功率  $P$  分配和计算资源  $F$  分配两个子问题, 而且不存在变量联合约束的情况, 因此考虑对这两个子问题分别进行求解。

### 3.3 卸载用户功率分配

通过上述问题分解, 可以将卸载用户的上行链路功率分配问题表示为:

$$\begin{aligned} \min_{p_k} \quad & \mathcal{J}(p_k) \\ \text{s.t.} \quad & C2 \end{aligned} \quad (16)$$

其中:

$$\mathcal{J}(p_k) = \frac{\eta_k p_k + \tau_k}{\log_2(1 + \omega_k p_k)} \quad (17)$$

通过对  $\mathcal{J}(p_k)$  求二阶导数  $\mathcal{J}''(p_k)$  可以看出,  $\mathcal{J}''(p_k)$  在定义域内并非总为正, 因此  $\mathcal{J}(p_k)$  是非凸函数。根据拟凸函数的二阶导数判定条件, 可以看出  $\mathcal{J}(p_k)$  是在定义域内的拟凸函数<sup>[9]</sup>。当  $\mathcal{J}'(p_k)=0$  时, 即:

$$\phi(p_k) = \eta_k \log_2(1 + \omega_k p_k) - \frac{w_k}{\ln 2} \frac{\tau_k + \eta_k p_k}{1 + \omega_k p_k} = 0 \quad (18)$$

可以观察到式(18)的一阶导数为  $\phi'(p_k) = \omega_k^2 (\tau_k + \eta_k p_k) / [\ln 2 (1 + \omega_k p_k)^2] > 0$ , 其中  $\phi(0) = -\omega_k \tau_k / \ln 2 < 0$ , 表明  $\phi(p_k)$  是在定义域内的单调递增函数且在初始点小于 0, 因此本文采用二分搜索法, 计算满足  $\phi(p_k)=0$  的点来表示最优的上行链路功率分配。

### 3.4 卸载用户计算资源分配

根据式(16)可以将计算资源分配问题表示为:

$$\begin{aligned} \min_{f_k} \quad & g(f_k) \\ \text{s.t.} \quad & C3, C4 \end{aligned} \quad (19)$$

其中:

$$g(f_k) = \sum_{k \in \mathcal{K}} \gamma_k^t s_k / f_k \quad (20)$$

可以观察到  $g(f_k)$  的定义域为凸集, 且 Hessian 矩阵为半正定矩阵, 故  $g(f_k)$  为凸函数。定义在不等式约束条件  $C3$ 、 $C4$  下的拉格朗日函数表达式为:

$$L(f_k, \lambda, \mu) = \sum_{k \in \mathcal{K}} \frac{\gamma_k^t s_k}{f_k} + \lambda \left( \sum_{k \in \mathcal{K}} f_k - f_{\max} \right) + \mu (-f_k) \quad (21)$$

其中,  $\lambda$  和  $\mu$  分别是与约束条件  $C3$  和  $C4$  相对应的拉格朗日乘子, 且  $\lambda, \mu \geq 0$ 。通过求取式(21)的 KKT(Karush-Kuhn-Tucker Conditions)条件<sup>[10]</sup>, 可得计算资源配的最优解为:

$$f_k^* = \frac{\sqrt{\gamma_k^t s_k}}{\sum_{k \in \mathcal{K}} \sqrt{\gamma_k^t s_k}} f_{\max} \quad (22)$$

## 4 仿真验证与分析

### 4.1 仿真参数设置

考虑一个应用 MEC 的单小区-多用户场景, 其中 MEC 服务器位于 BS 附近同址部署, 用户均匀分布在小区的覆盖范围内。具体仿真图参数如表 1 所示。

表 1 仿真参数设置表

仿真参数	参数取值
小区覆盖半径/m	500
系统带宽 $B$ /MHz	20
用户带宽 $W_k$ /MHz	1
上行链路带宽内噪声功率 $\sigma_0^2$ /dBm	-100
用户最大发送功率 $p_{\max}$ /dBm	23
计算数据量 $b_k$ /KB	400~1 200
移动用户所需 CPU 周期 $s_k$ /GHz	0.2~1
用户容忍最大时延 $T_k^{\max}$ /s	1~4
用户的计算能力 $f_k^l$	0.1 GHz~1 GHz/周期
MEC 服务器的计算能力	4 GHz/周期
能耗和时延消耗因子 $\gamma_k^c, \gamma_k^t$	0.25~0.75
$\beta_1, \beta_2, \beta_3, \beta_4$	0.2, 0.4, 0.3, 0.6

### 4.2 对比方案设置

为便于仿真结果对比与分析说明, 做如下方案设置: 方案一: 本地执行计算任务; 方案二: 全部卸载计算任务; 方案三: 文献[8]算法, 单小区中多用户依据任务卸载决策选择任务上传 MEC 服务器执行, 并联合优化计算和无线资源; 方案四: 本文方案, 基于自适应遗传算法的多用户任务卸载决策与资源分配联合优化算法。

### 4.3 仿真结果对比分析图

图 2 描述的是卸载用户数与网络总用户数的关系

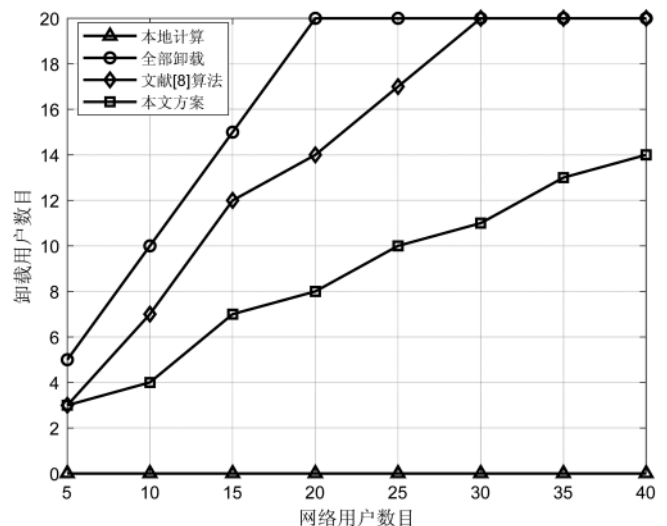


图 2 卸载用户数和网络总用户数的关系



# 通信与网络 Communication and Network

图。从图中可以看出,方案一选择全部本地处理任务,因此卸载用户数目一直为0;方案二选择卸载所有任务,直至 $K>20$ 时用尽系统带宽,剩余未卸载的用户只能选择本地处理任务;方案三未考虑到系统带宽的限制,当 $K>30$ 时系统带宽接近饱和,用户不能选择卸载任务;而本文方案能够最大限度地利用系统带宽,适当地选择卸载用户数目,从而提高系统效用。

图3描述的是用户总开销与用户数量的关系。从图中可以明显地看出本文所提方案与其他方案相比有更低的总开销。其中相比于文献[8]的算法未涉及卸载用户的MEC计算资源分配问题,本文算法综合考虑了卸载用户的上行链路发送功率与计算资源的分配问题,在满足用户在时延的约束条件下对用户进行了有效的功率分配和计算资源分配,因此在系统性能上有了较为明显的提升。

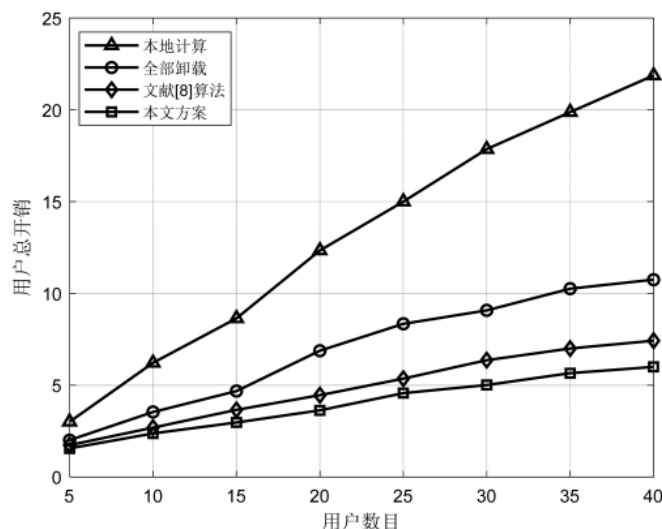


图3 用户总开销与用户数目的关系

图4所示为当用户数 $K=40$ 时,用户时延偏好权重与用户平均时延和能耗的关系。从图中可知,当用户时延偏好权重 $\gamma_k^i$ 增加时,说明此时对时延比较敏感,用户的平均时延随之降低。同时,用户的平均能耗增加,说明用户以牺牲能耗的代价来换取对时延的要求。表明了用户能够针对计算任务的需求作出动态的调整以提高系统性能。

图5所示为用户数 $K=40$ 时,3种固定的交叉概率和变异概率与自适应的交叉和变异概率对算法收敛性的影响。从图中可以看出,给出固定的交叉和变异概率会使该算法陷入局部最优解,而且会使搜索过程变长。自适应的交叉和变异概率能够随适应值做动态调整,防止进入局部最优解。相对于固定的交叉概率和变异概率的设定,本文所提方案具有明显的优势。

## 5 结论

本文讨论了单小区-多用户场景下基于移动边缘计

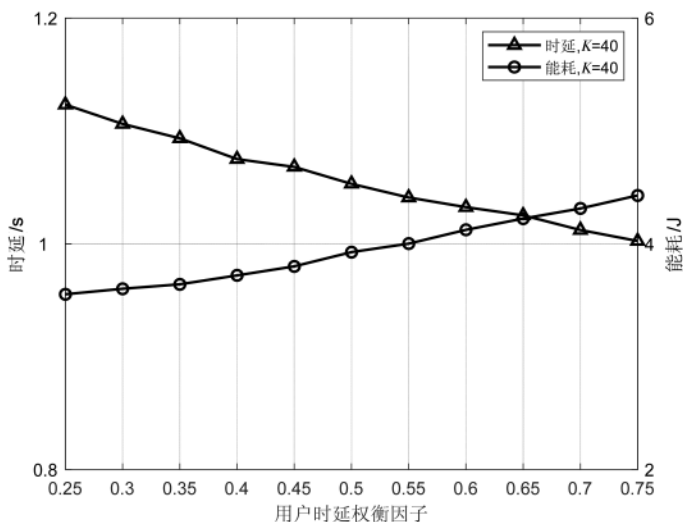


图4 用户时延偏好权重 $\gamma_k^i$ 与用户平均时延和能耗的关系

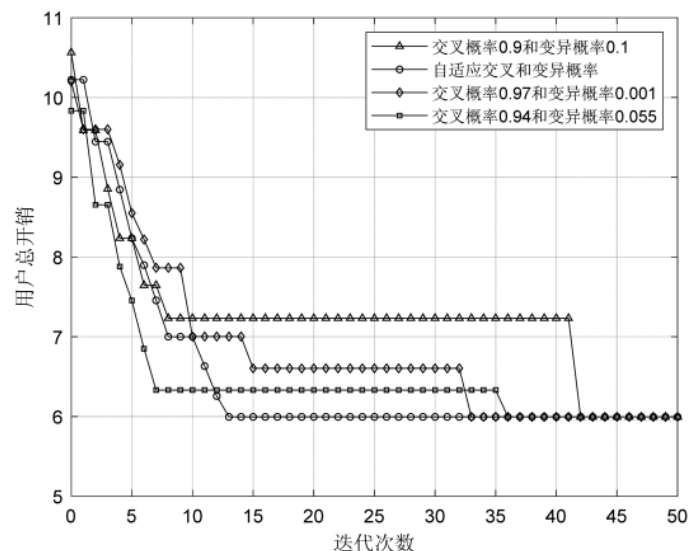


图5 交叉和变异概率对算法收敛性的影响

算的任务卸载和资源分配问题,提出了以最小化用户的总开销(能耗-时延加权和)为目标的优化问题。首先采用自适应遗传算法制定卸载决策及后续的更新操作,在每一次卸载决策更新的情况下将原问题分解为功率分配和计算资源分配两个子问题,然后根据凸优化及准凸优化理论,利用二分搜索法和拉格朗日乘子法分别求出功率分配和计算资源分配的最优解。实验证明,该方案可通过较少的迭代次数达到收敛,有效地降低系统的总开销,提升系统的整体性能。

## 参考文献

- [1] 朱雪田,夏旭,齐飞.5G网络关键技术和业务[J].电子技术应用,2018,44(9):1-4.
- [2] 杨鑫,时晓厚,沈云,等.5G工业互联网的边缘计算技术架构与应用[J].电子技术应用,2019,45(12):25-28.
- [3] MACH P,BECVAR Z.Mobile edge computing: a survey on architecture and computation offloading[J].IEEE Commun-

- 2633–2645.
- [9] BOYD S, VANDENBERGHE L, FAYBUSOVICH L. Convex optimization[J]. IEEE Transactions on Automatic Control, 2006, 51(11): 1859–1859.
- [10] LYU X, TIAN H, SENGUL C, et al. Multiuser joint task offloading and resource optimization in proximate clouds[J]. IEEE Transactions on Vehicular Technology, 2017, 66(4): 3435–3447.

### 作者简介：

刘笑笑(1994-),女,硕士研究生,主要研究方向为:雾  
计算机网络。

[illegible]

微纳卫星简易地面接收设备,由天线、射频前端、基带部分和 MATLAB 后处理软件组成。卫星数据经由天线接收、射频放大、滤波、射频下变频、ADC 数字量化、FPGA 数字下变频和降采样后通过网卡传输到上位机上,再通过 MATLAB 处理,能够正确解调出卫星的遥测数据。结果表明,该设备可用于微纳卫星地面遥测接收,同时当前工作可为后续解调算法 FPGA 工程化实现打下坚实基础。

[1] 陈利虎, 陈小前, 刘鹏飞. 适用微纳卫星的便携测控站设

- [1] 计[J]. 飞行器测控学报, 2015, 34(1): 16-23.
- [2] 韦明川. 紫丁香 2 号卫星 VHF/UHF 星地测控链路设计与实现[D]. 哈尔滨: 哈尔滨工业大学, 2015.
- [3] 户凌志, 张海勇, 贺寅. 卫星通信链路计算方法研究[J]. 舰船电子工程, 2019, 39(11): 72-75.
- [4] 周星星, 翟继强. 基于 FPGA 配置的 AD9361 处理器应用方法研究[J]. 科技创新与应用, 2017(36): 6-7.
- [5] 沈俊. 基于 Xilinx FPGA+AD9361 的软件无线电平台设计与实现[J]. 电子元件与信息技术, 2018(9): 58-60, 64.
- [6] 张俊辉. 接收机噪声系数对接收灵敏度影响[J]. 电子技术与软件工程, 2014(20): 53.
- [7] 杨芳, 傅伟廷, 秦天凯, 等. 高频数字抽取滤波器的设计[J]. 电子技术应用, 2017, 43(12): 25-28.
- [8] 郑天宇, 尹达一. 千兆 UDP/IP 协议栈 FPGA 实现及其在高速成像系统中的应用[J]. 现代电子技术, 2018, 41(18): 1-5, 11.

### 作者简介:

何林飞(1988-),男,硕士,主要研究方向:卫星测控、通信。

李晓飞(1987-),男,硕士,主要研究方向:卫星测控、通信。

韩俊博(1987-),男,硕士,主要研究方向:卫星测控、通信。

## 版权声明

经作者授权，本论文版权和信息网络传播权归属于《电子技术应用》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、DOAJ、美国《乌利希期刊指南》、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《电子技术应用》编辑部

中国电子信息产业集团有限公司第六研究所