

系统仿真学报
Journal of System Simulation
ISSN 1004-731X, CN 11-3092/V

《系统仿真学报》网络首发论文

题目: 具有能量收集设备的移动边缘计算资源分配研究
作者: 李长云, 黎建波, 徐曦, 李亭立
DOI: 10.16182/j.issn1004731x.joss.21-0576
网络首发日期: 2021-09-14
引用格式: 李长云, 黎建波, 徐曦, 李亭立. 具有能量收集设备的移动边缘计算资源分配研究. 系统仿真学报. <https://doi.org/10.16182/j.issn1004731x.joss.21-0576>



网络首发: 在编辑部工作流程中, 稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定, 且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式 (包括网络呈现版式) 排版后的稿件, 可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定; 学术研究成果具有创新性、科学性和先进性, 符合编辑部对刊文的录用要求, 不存在学术不端行为及其他侵权行为; 稿件内容应基本符合国家有关书刊编辑、出版的技术标准, 正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性, 录用定稿一经发布, 不得修改论文题目、作者、机构名称和学术内容, 只可基于编辑规范进行少量文字的修改。

出版确认: 纸质期刊编辑部通过与《中国学术期刊 (光盘版)》电子杂志社有限公司签约, 在《中国学术期刊 (网络版)》出版传播平台上创办与纸质期刊内容一致的网络版, 以单篇或整期出版形式, 在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊 (网络版)》是国家新闻出版广电总局批准的网络连续型出版物 (ISSN 2096-4188, CN 11-6037/Z), 所以签约期刊的网络版上网络首发论文视为正式出版。

具有能量收集设备的移动边缘计算资源分配研究

李长云², 黎建波^{1,2}, 徐曦², 李亭立²

(1.湖南工业大学电气与信息工程学院, 湖南 株洲 412000; 2.智能信息感知及处理技术湖南省重点实验室, 湖南 株洲 412000)

摘要: 针对具有能量收集能力的移动边缘计算系统的计算资源分配问题, 提出一种基于李雅普诺夫贪婪优化(Lyapunov Greed Optimization, LGO)算法。构建在设备电池电量逐渐收敛下, 移动设备时延与能耗联合成本的动态最小化优化问题。利用李雅普诺夫动态优化理论, 将优化问题分解成每个时隙最佳本地执行、卸载执行和能量收集三个子问题, 并通过线性规划获得子问题最优解。然后通过在本地图执行、卸载执行和任务丢弃之间选择执行模式, 获得设备的时延与能耗联合成本最小结果。最后利用键值对设计贪婪策略程序, 以适应多用户多服务器系统。仿真结果证实, 在保证所有设备电池电量都在规定操作水平附近稳定情况下, 卸载率可达 99.9% 以上, 并能有效降低服务延时和系统能耗。

关键词: 能量收集; 李雅普诺夫优化; 边缘计算; 计算卸载; 资源分配

中图分类号: TP301. 文献标志码: A

DOI: 10.16182/j.issn1004731x.joss.21-0576

Research on mobile edge computing resource allocation with energy harvesting device

Li Changyun², Li Jianbo^{1,2}, Xu Xi², Li Tingli²

(1.School of Electrical and Information Engineering, Hunan University of Technology, Zhuzhou; 412000; 2. Key Laboratory of Intelligent Information Perception and Processing Technology of Hunan Province, Zhuzhou, 412000)

Abstract: In order to solve the problem of computing resource allocation of mobile edge computing system with energy gathering ability, an algorithm based on Lyapunov Greed Optimization (LGO) was proposed. This paper presents a dynamic optimization problem to minimize the combined cost of time delay and energy consumption of mobile devices under the gradual convergence of equipment battery power. By using Lyapunov dynamic optimization theory, the optimization problem was decomposed into three sub-problems: optimal local execution, unload execution and energy harvesting for each time slot, and the optimal solution of the sub-problems was obtained by linear programming. Then, by selecting the execution mode between local execution, unload execution and task discarding, the combined cost of time delay and energy consumption of the vehicle can be minimized. Finally, the greedy policy program is designed by using key-value pairs to adapt to multi-user and multi-server systems. The simulation results show that under the condition that the battery power of all equipment is stable around the specified operating level, the unloading rate can reach more than 99.9%, and the service delay and system energy consumption can be effectively reduced.

Keywords: energy harvesting; lyapunov optimization; edge computing; computation offloading; resource allocation

引言

5G 的大力发展和移动设备的日益普及, 为

湖南省创新平台开放基金项目 (19K026); 湖南省重点领域研发计划课题 (2020KF02)

作者简介: 李长云 (通信作者 1971-), 男, 博士, 教授, 研究方向: 智能信息处理、边缘计算、车间调度。E-mail: Lcy469@163.com。

黎建波 (1997-), 男, 硕士研究生, 研究方向: 边缘计算、智能信息处理。E-mail: w547j352@163.com。

<http://www.china-simulation.com>

VR、AR 和人脸识别等智能服务铺平了道路。然而,这种服务通常是计算密集、延迟敏感和耗能的,也是计算能力受限的移动设备所不能支持的。因此,需要研究新的范式来适应智能化的服务要求。移动边缘计算(Mobile Edge Computing,MEC)网络被认为是支持大规模能源受限和计算受限设备的有前途的范例。而且能量收集(Energy Harvesting,EH)进一步增强了有限的能量支持设备的运行能力^[1]。

近年来,具有 EH 设备的 MEC 系统的资源管理问题已引起广泛关注。在文献[2]中,为了优化具有 EH 的边缘计算系统的能效,提出了一种通过差分进化优化的节能任务卸载方法,该算法可以确定卸载时间、计算时间和频率。文献[3]中设计了一种分布式的面向纳什均衡的迭代最佳响应算法并应用于异构 MEC 网络中,有效降低了移动设备的平均响应时间。文献[4]中提出了一种基于李雅普诺夫优化理论的在线动态卸载和资源调度算法,有效降低了能耗成本。文献[5]中考虑了边缘计算环境中充电衰减的多样性,设计了基于马尔可夫决策过程的无线能量收集策略,做到了最大化长期平均效用。文献[6]中设计了一种考虑任务依赖性和动态能量收集的模型,同时定制模拟退火算法,以优化由所提出的贪婪算法生成的解决方案,有效降低了任务完成时间。文献[7]中,为具有 EH 设备的 MEC 系统提出了一种基于深度学习的分流方案,有效降低了能量消耗、计算时延和任务丢弃率。在文献[8]中,针对设备从环境射频源收集能量较少的问题,提出一种基于混合能量收集的 MEC 系统中最小化设备总能耗的资源分配策略。文献[9]将无线能量传输与 MEC 相结合,利用邻近边缘服务器的近端设备协助远端设备分流计算任务到边缘服务器计算,提出了一种最大化能量效率的资源分配策略。虽然目前围绕具有 EH 设备的 MEC 系统进行了大量工作,然而这些工作没有在使用时延和能耗成本作为评价指标的同时,考虑最大化利用边缘服务器的优势。

本文针对具有多台 EH 设备的边缘网络中资源

分配问题,提出 LGO 算法。在电池电量收敛情况下,构建长期动态服务延时与系统能耗联合成本最小化问题。本文旨在通过联合考虑能量采集、卸载决策和计算资源分配问题,在将每个设备电量稳定在预定值情况下,保证高卸载率和有效降低系统能耗和服务延时。

1 MEC 系统及优化问题建模

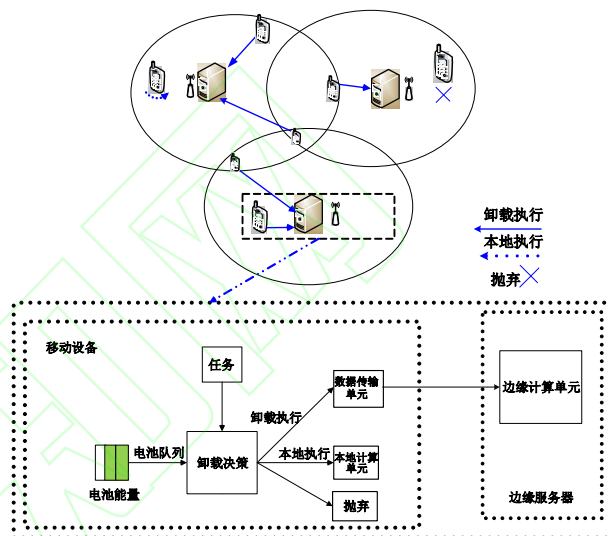


图1 系统场景和架构

Fig.1 System scenario and architecture

本文研究对象为多设备边缘计算网络。系统结构如图1所示,边缘系统包括 N 个带有 EH 能力的移动设备和 M 个边缘服务器。所有元件被限制在

特定区域, 其中服务器位于特定位置, 而设备的位置在服务器覆盖范围随意变化。 $d_{i,j}^t$ 表示各个时隙移动设备与 MEC 服务器之间距离。

每个计算任务可以在本地计算或者卸载到边缘服务器。如果两者都不行或电量不足时, 任务将被抛弃。所以:

$$I_{i,l}^t + I_{i,r}^t + I_{i,d}^t = 1 \quad (1)$$

式中, $I_{i,c}^t \in \{0,1\}$, $c = \{l, r, d\}$ 可以表示在第 t 个时隙计算模式指示器对于设备的状态。其中 $I_{i,l}^t = 1$, $I_{i,r}^t = 1$ 和 $I_{i,d}^t = 1$ 分别表示计算任务在本地、卸载和抛弃执行。

1.1 卸载计算模型

关注延迟敏感的计算任务, 其执行期限不超过每个时隙的长度^[10]。第 i 个设备在第 t 个时隙要处理的计算任务为 CT_i^t , 固定尺度为 L 位。

$C_{i,j}^t = 1, C_{i,j}^t \in \{0,1\}$ 表示第 i 个移动设备选择第 j 个边缘服务器进行卸载。当选择卸载计算模式时, 假设每个计算任务只分配给一个服务器, 即:

$$\sum_{j=1}^M C_{i,j}^t = 1 \quad (2)$$

式中, $\gamma_{i,j}^t$ 表示小尺度衰落信道功率增益, 根据通信理论, 信道功率增益表示为:

$$h_{i,j}^t = \gamma_{i,j}^t g_0 (d_0 / d_{i,j}^t)^\theta \quad (3)$$

式中, d_0 表示参考距离, θ 表示路径损耗指数, g_0 表示路径损耗常数。速率可以根据香农定理获得:

$$R(h_{i,j}^t, p_i^t) = \omega \log_2(1 + h_{i,j}^t / \sigma) \quad (4)$$

式中, ω 表示带宽分配, σ 表示每个云服务器的噪声功率, p_i^t 表示发射功率 (最大发射功率不超过 p^{\max}), 每个云服务器的带宽均分为 N 子带, 并且每个传感器被分配给一个子带 ω MHz。将每个移动设备和每个 MEC 服务器处理一位任务所需的 CPU 周期频率表示为 X_m 和 X_s 。假设 MEC 服务器的计算能力受到限制, 即:

$$\sum_{i \in N} I(I_{i,r}^t) \cdot C_{i,j}^t L X_s \leq f_s^{\max} \tau \quad (5)$$

式中, f_s^{\max} 表示每个 MEC 服务器的 CPU 周期频率的上限。 $I(\cdot)$ 表示指示器功能。

S_{UB} 表示移动设备同时连接到 MEC 服务器的最大数量。MEC 服务器的计算能力约束在式(5)中, 考虑到 S_{UB} 必须是整数, 所以:

$$S_{UB} \leq \left\lceil f_s^{\max} \tau / L X_s \right\rceil \quad (6)$$

本文没有考虑 MEC 服务器在执行过程中所消耗的延迟。对于第 i 个移动, 这种模式的总执行延迟等于输入任务的传输延迟。因此:

$$D_{i,r}^t = \frac{L}{R(h_{i,j}^t, p_i^t)} \quad (7)$$

相应消耗的能量是:

$$E_{i,r}^t = p_i^t \frac{L}{R(h_{i,j}^t, p_i^t)} \quad (8)$$

1.2 本地计算模型

通过应用动态电压和频率缩放技术(dynamic voltage and frequency scaling, DVFS)^[11], $x \in \{1, 2, \dots, L X_m\}$ 模式的总执行延迟表示为:

$$D_{i,l}^t = \sum_{x=1}^{L X_m} (f_{i,x}^t)^{-1} \quad (9)$$

相应的能耗是:

$$E_{i,l}^t = \sum_{x=1}^{L X_m} \delta (f_{i,x}^t)^2 \quad (10)$$

式中, δ 表示有效电容系数, 将每个移动设备的 CPU 周期频率的上限表示为 f_m^{\max} , 有:

$$f_{i,x}^t \leq f_m^{\max} \quad (11)$$

式中, $\forall x \in \{1, 2, \dots, L X_m\}$ 。

1.3 成本函数与问题建模

为了体现可再生能源过程的随机和间歇性质, 假设可收获的能源 e_i^t 是随机分布的^[12], 最大值为 $E_{i,H}^t$, 即:

$$0 \leq e_i^t \leq E_{i,H}^t \quad (12)$$

假设除了本地执行和远程执行之外, 其他类型的能量消耗足够小。利用 $E_x(I_i^t, f_i^t, p_i^t)$ 表示在第 t 个时隙移动设备的能耗。其中, $I_i^t \triangleq [I_{i,l}^t, I_{i,r}^t, I_{i,d}^t]$, $f_i^t \triangleq [f_{i,1}^t, f_{i,2}^t, \dots, f_{i,X_m}^t]$, 可以得到系统能耗成本:

$$E_x(I_i^t, f_i^t, p_i^t) = I_{i,l}^t E_{i,l}^t + I_{i,r}^t E_{i,r}^t \quad (13)$$

式中, $D_v(I_i^t, f_i^t, p_i^t)$ 表示在 t 时隙所需的总时间

为:

$$D_x(I_i^t, f_i^t, p_i^t) = I_{i,l}^t D_{i,l}^t + I_{i,r}^t D_{i,r}^t \quad (14)$$

系统延时是由执行延迟和放弃任务的惩罚组成。可以通过下式获得系统延时成本:

$$\cos t_{sum}^t \triangleq \sum_{i \in N} \cos t_i^t = \sum_{i \in N} [D(I_i^t, f_i^t, p_i^t) + \text{phi} \cdot I_{i,d}^t] \quad (15)$$

式中, phi 为任务丢弃成本。

在电池电量队列影响下, 服务延时和系统能耗联合成本长期最小化优化问题可以建模为:

$$\lim_{I_i^t, f_i^t, p_i^t} \lim_{T \rightarrow \infty} \frac{1}{T} E[E_x(I_i^t, f_i^t, p_i^t) + \cos t_{sum}^t] \quad (16)$$

$$s.t. (1), (2), (5), (12), (13)$$

$$0 \leq f_{i,v}^t \leq f_m^{\max} \quad (16)$$

$$\varepsilon(I_i^t, f_i^t, p_i^t) \leq E_{\max} \quad (17)$$

$$0 \leq p_i^t \leq p^{\max} \quad (18)$$

$$I_{i,c}^t \in \{0, 1\}, c \in \{l, r, d\} \quad (19)$$

式(16)和式(18)分别体现了移动设备 CPU 的循环频率和最大发射功率的约束。式(17)体现了出于安全原因的电池放电的上限, 在每个时隙输出能量的能量不能超过 E_{\max} 。式(19)描述任务选择的 0-1 指标约束。

2 低复杂度的联合代价和资源分配优化算法

首先, 需要将依赖于时间的原始问题通过李雅普诺夫优化转化为每个时隙确定性问题, 然后利用贪婪策略对算法进行升级和重构, 以适应多设备 MEC 系统。

2.1 电能队列构建

将第 t 个时隙的第 i 个移动设备的电池电量用 Q_i^t 表示。显然, 每个时隙的能量消耗不能超过电池水平, 即:

$$E_x(I_i^t, f_i^t, p_i^t) \leq Q_i^t \quad (20)$$

电池能量积压 Q_i^t 根据以下等式发展:

$$Q_i^{t+1} = Q_i^t + [e_i^t - \varepsilon(I_i^t, f_i^t, p_i^t)] \quad (21)$$

2.2 虚拟电能队列和漂移加惩罚公式构建

首先定义扰动参数和移动设备上的虚拟能量队列, 这是两个关键元素。李雅普诺夫优化要求允许的动作集是独立同分布, 这是本文建模的移动设备的时变队列所不能满足的。因此, 对于每个移动设备, 使用扰动参数 θ 定义虚拟队列 \tilde{Q}_i^t :

$$\tilde{Q}_i^t \triangleq Q_i^t - \theta$$

$$\theta = \tilde{E}_{\max} + V\eta \quad (22)$$

$$\tilde{E}_{\max} \triangleq \min\{\max\{\delta LX_m (f_m^{\max})^2, p^{\max} \tau\}, E_{\max}\} \quad (23)$$

$$\eta = \text{phi} / E_{\min} \quad (24)$$

式中, \tilde{Q}_i^t 为虚拟队列, 是移动设备实际电池能量水平的偏移版本; V 为惩罚权, 用来权衡延时与能耗。此外, 如果在第 t 个时隙请求的任务正在本地执行, 移动设备 CPU 处理任务的最佳周期频率相同, 即 $f_{i,x}^t = f_i^t, x \in \{1, 2, \dots, LX\}$, 它可以通过算术和几何方法的不等式得到。

根据上述分析, 将李雅普诺夫函数定义^[13]为:

$$L(t) = \frac{1}{2} \sum_{i \in N} (\tilde{Q}_i^t)^2 = \frac{1}{2} \sum_{i \in N} (Q_i^t - \theta)^2 \quad (25)$$

条件李雅普诺夫漂移可以写成:

$$\Delta(t) = L(t+1) - L(t) \quad (26)$$

这里, $\tilde{Q}_i^t \triangleq [\tilde{Q}_1^t, \tilde{Q}_2^t, \dots, \tilde{Q}_N^t]$, 则李雅普诺夫漂移加惩罚函数可以写成:

$$\Delta_v(t) \triangleq \Delta(t) + E[\cos t_{sum}^t | \tilde{Q}^t] \quad (27)$$

由式(21)可以获得:

$$(\tilde{Q}_{i+1}^t)^2 \leq (\tilde{Q}_i^t)^2 + 2\tilde{Q}_i^t(e_i^t - \varepsilon(I_i^t, f_i^t, p_i^t)) + (e_i^t)^2 + \varepsilon^2(I_i^t, f_i^t, p_i^t) \quad (28)$$

根据 Lyapunov 优化, 使用漂移加惩罚函数来平衡能耗和延时对优化结果的影响^[14], 漂移函数存在上限有:

$$\Delta(t) \leq \sum_{i \in N} E[e_i^t - \varepsilon(I_i^t, f_i^t, p_i^t) | \tilde{Q}^t] + C \quad (29)$$

这里, $C = \frac{N}{2} (\tilde{E}_{\max} + E_H^{\max})^2$, 结合式(27), 有:

$$\Delta_i(t) \leq \sum_{i \in N} \tilde{Q}_i^t (e_i^t - \varepsilon(I_i^t, f_i^t, p_i^t)) + V \cdot E[\cos t_{sum}^t | \tilde{Q}] + C \quad (30)$$

2.3 最佳计算卸载

2.3.1 问题简化与分解

根据系统模型可将计算卸载问题分解成三个子问题，即最优本地计算、计算卸载和能量收集。

在将 e_i^t 与目标函数解耦后，可以将每时隙问题的目标函数简化为以下优化问题 p_{CO} [15]：

$$p_{CO} : \min_{I_i^t, f_i^t, p_i^t} -\tilde{Q} \cdot \varepsilon(I_i^t, f_i^t, p_i^t) + V \cdot [D(I_i^t, f_i^t, p_i^t) + \text{phi} \cdot I(I^t = 1)] \quad (31)$$

2.3.2 最优本地计算问题

通过解决以下优化问题，可获得移动设备在本地执行任务的最佳中央处理器周期频率：

$$f_{ME} : \min_{f^t} -\tilde{Q} \delta W(f^t)^2 + V \frac{W}{f^t} \quad (32)$$

$$s.t. f^t \in [0, f_{\max}] \quad (32)$$

$$W / f^t \leq \tau \quad (33)$$

$$\delta w(f^t)^2 \in [E_{\min}, E_{\max}] \quad (34)$$

由式(32)所示，移动设备上的计算能力有限，本地执行不一定可行。为解决这一问题，本文给出上述优化的可行性条件，并求其最优解。

如果 f_{ME} 可行，其最优解为：

$$f^{t*} = \begin{cases} f_U & f_0^t > f_U \\ f_0^t & f_L < f_0^t < f_U \\ f_L & f_0^t < f_L \end{cases} \quad (35)$$

由于 CPU 能耗随着 f^t 的增加而增加，因此，电池输出能量约束可以等效表示为： $\sqrt{E_{\max} / \delta W} \leq f^t \leq \sqrt{E_{\min} / \delta W}$ 。通过合并式(32)，将可行的中央处理器周期频率设置为： $f_L = \max\{\sqrt{E_{\min} / \delta W}, w / \tau\}$, $f_U = \min\{\sqrt{E_{\max} / \delta W}, f_{\max}^{CPU}\}$ 。通过将优化函数的一阶导数设置为 0，可得到一个唯一的解，即 $f_0 = \sqrt[3]{V / (-2\tilde{Q}^t \cdot \delta)}$ 。

2.3.3 最优计算卸载

考虑任务由 MEC 服务器执行的情况，其中用于计算卸载的最佳发射功率可以通过解决以下优化问题来获得：

$$p_{SE} : \min_{p^t} -\tilde{Q} \frac{p_{i,j}^t L}{R(h_{i,j}^t, p_{i,j}^t)} + V \frac{L}{R(h_{i,j}^t, p_{i,j}^t)} \quad (36)$$

$$s.t. p^t \in [0, p_{tx}^{\max}] \quad (36)$$

$$\frac{p_{i,j}^t L}{R(h_{i,j}^t, p_{i,j}^t)} \in [E_{\min}, E_{\max}] \quad (37)$$

$$\frac{L}{R(h_{i,j}^t, p_{i,j}^t)} \leq \tau \quad (38)$$

首先确定发射功率的取值范围。在约束条件下，由式(37)中递增函数可得设备最小发射功率 $p_{E_{\min}}^t$ 和最大发射功率 $p_{E_{\max}}^t$ 。然后通过式(38)中递减函数可获得满足延迟需求时的最小功率 $p_{L,\tau}^t$ ，此时满足发射功率需耗能为 $E_{L,\tau}^t$ 。综上，发射功率需满足条件为：

$$p_L^t = \begin{cases} p_{L,\tau}^t & E_{L,\tau}^t \geq E_{\min} \\ \max[p_{L,\tau}^t, p_{E_{\min}}^t] & E_{L,\tau}^t \leq E_{\min} \end{cases}$$

$$(39) p_U^t = \begin{cases} \min[p_{\max}, p_{E_{\max}}^t] & E_{L,\tau}^t \leq E_{\max} \\ 0 & E_{L,\tau}^t \geq E_{\max} \end{cases} \quad (40)$$

只有当 $p_U^t \geq p_L^t$, p_{SE} 时才可行。为了求最优值，需要定义一个函数：

$$g(p^t) = -\tilde{Q} \frac{p^t L}{R(h_{i,j}^t, p^t)} + V \frac{L}{R(h_{i,j}^t, p^t)} \quad (41)$$

当 $\tilde{Q} < 0$ 时，对式(41)求导得：

$$\frac{d(g(p_{i,j}^t))}{dp^t} = \frac{-\tilde{Q} \log_2(1 + \frac{h_{i,j}^t p_{i,j}^t}{\sigma}) - \frac{h_{i,j}^t (-\tilde{Q} p_{i,j}^t + V)}{(h^t p_{i,j}^t + \sigma) \ln 2}}{w \log_2^2(1 + \frac{h_{i,j}^t p_{i,j}^t}{\sigma})} \quad (42)$$

$$= \frac{G(p_{i,j}^t)}{w \log_2^2(1 + \frac{h_{i,j}^t p_{i,j}^t}{\sigma})}$$

对函数 $G(p^t)$ 求导可得其为递增函数。当

$p^t = 0$ 时, $G(p^t) = \frac{-h_{i,j}^t V}{\sigma \ln 2} < 0$; 当 $p^t = +\infty$ 时, $G(p^t) = +\infty$ 。所以存在一个特解, 并设为 p_0^t 。综上可得, 最优频率为:

$$p^{t*} = \begin{cases} p_U^t & p_U^t < p_0^t \\ p_L^t & \tilde{Q}^t < 0, p_L^t > p_0^t \\ p_0^t & \tilde{Q}^t < 0, p_L^t < p_0^t < p_U^t \end{cases} \quad (43)$$

2.3.4 最佳能量收集

通过求解下面的线性规划问题, 可以得出每个移动设备最优的能量收获量:

$$\min_{0 \leq e_i^t \leq E_{i,H}^t} \sum_{i \in N} \tilde{Q}_i^t e_i^t \quad (44)$$

2.3.5 决定计算模式与贪婪策略

通过对丢弃任务、本地执行和 MEC 服务器执行三种计算模式的最优值进行评估, 得到最优的计算卸载决策, 为:

$$J_{CO}(I_i^t, f_i^t, p_i^t) = I(I_{i,l}^t) \cdot J_m(f_i^t) + I(I_{i,r}^t) \cdot J_{serve}^t(p_i^t) + I(I_{i,d}^t) \cdot V \cdot \phi_i \quad (45)$$

当选择卸载计算模式时, 任意两个移动设备之间存在相关性。因此, 李雅普诺夫在线算法不能直接处理多设备问题。针对多用户多服务器系统, 利用键值对方法设计贪婪策略。具体来说, 使用地图来存储每个移动设备及其选择的最佳 MEC 服务器。通过键值对方法可以做到在边缘服务器计算能力受限情况下服务器的最优选择。

算法 1 LGO 算法

1. 初始化系统参数, 令键值对为 false;
2. While $t \in [0, T]$ do;
3. For 每个移动设备 i to
4. 获取 $\xi_i^t, \tilde{Q}_i^t, d_{i,j}^t$;
5. 获取 f_i^{t*} , 如果 p_{ME} 是不可行的, 将键值对设置为 true。然后记录最优结果 $J_m^t(f_i^t)$;
6. For 每个 MEC 服务器 j ;
7. 获取 $h_{i,j}^t$ 和 $p_{i,j}^{t*}$ 。如果 p_{ME} 是不可行的, 将键值

对设置为 true。将 $\min J_s^t(p_{i,j}^t)$ 设置为 $J_s^t(p_i^t)$ 时, 并记录下此时 j 和 $p_{i,j}^{t*}$;

8. End for

9. 比较 $J_m^t(f_i^t)$, $J_s^t(p_i^t)$ 和 $V \cdot \phi_i$, 选择具有最小值的模式, 并设置相应的指示器变量 $I_{i,c}^t$ 作为 1;

10. If $I_{i,r}^t = 1$ then

11. 设置键值对 $i - j$ 进入地图;

12. End if

13. End for

14. If 键值对为 false

15. 调用键值对方法

16. End if

17. 更新 $t = t + 1$;

18. End while

算法 1 给出了所提出算法的主要部分, 子程序 1 展示的是键值对方法。

子程序 1 键值对方法

1. While map 不为空 do
2. if $I_{i,r}^t = 1$ then
3. 通过 $\min J_s^t(p_{i,j}^{t*})$ 时, 获取最优 j , 将对应的键值对 “ $i - j$ ” 放入地图;
4. If $flag[j] \leq S_{UB}$
5. 从 map 中删除键值对 “ $i - j$ ”, 并将 $J_s^t(p_{i,j}^{t*}) = \inf$
6. Else
7. 选择出前 S_{UB} 个 $J_{i,j}^s(p_{i,j}^{t*})$ 最小, 其余通过 $second\{J_s^t(p_{i,j}^{t*})\}$ 发现第二优 j , 将对应的键值对 “ $i - j$ ” 放入地图。然后返回步骤 4;
8. End if
9. else
10. 从本地执行和任务删除中选择最佳模式。然后从映射中移除相应的键值对;
11. End if
12. 保持对应 $I_{i,c}^t = 1$ 然后移除相应的键值对;
13. End while

3 仿真及对比分析

3.1 系统参数设置

本节建立了一个仿真平台来评估动态计算卸载决策与资源分配算法的性能。在 Windows 操作系统下搭建 Matlab 仿真环境，仿真参数如表 1 所示。

表 1 仿真参数

Tab.1 Experimental parameters

参数名称	参数符号	参数取值
移动设备数量	N	10
MEC 服务器数量	M	5
信道宽度	ω	1e6
完成任务所需周期数	W	737500
时隙长度	τ	2e-3
任务大小	L	1000
执行截止时间	τ_{max}	2e-3
CPU 计算能力	X	737.5
有效开关电容	k	1e-28
任务删除成本	ϕ	2e-3
接收机的噪声功率	σ	1e-13
设备最大发射功率	$p_{\text{tx}}^{\text{max}}$	1
设备最大 CPU 周期	f_{max}	1.5e9
采集能量上限	E_H^{max}	48e-6
电池最小输出能量	E_{min}	2e-5
惩罚权	V	1e-5
MEC 服务器	S_{UB}	4
最大连接数量		

为了确保移动性，假设每个移动设备与每个 MEC 服务器之间的距离 $d_{i,j}$ 在 $[0,70]$ 随机分配。为了比较，引入了三个基准策略算法，即贪婪能量分配的本地执行（本地执行）、贪婪能量分配的 MEC 服务器执行（全部卸载）和贪婪能量分配的动态卸载（动态卸载）。其工作原理如下：

本地计算：将最大可行中央处理器周期频率计算为 $f_U^t = \min\{f_{\text{CPU}}^{\text{max}}, \sqrt{\min(Q^t, E_{\text{max}})/\delta w}\}$ 。如果 $W/f_U^t \leq \tau$ ，计算任务将以 CPU 周期频率 f_U^t 在本地执行。此策略中禁用了计算卸载。

全部卸载：将最大可行发射功率计算为

$p_U^t = \min\{p_{\text{tx}}^{\text{max}}, p_{\min\{Q^t, E_{\text{max}}\}}^t\}$ 。当 $\sigma L \ln 2 / w h^t < \min\{Q^t, E_{\text{max}}\}$ 时， $p_{\min\{Q^t, E_{\text{max}}\}}^t$ 是 $pL = R(h_{i,j}^t, p) \min\{Q^t, E_{\text{max}}\}$ 的唯一解。在满足 $L/R(h_{i,j}^t, p_U^t) \leq \tau$ 条件下，计算任务将以 p_U^t 为发射功率卸载到 MEC 服务器上。否则，计算任务将被放弃。在此策略中禁用了本地计算。

动态卸载：分别计算本地执行和 MEC 服务器执行策略中的 f_U^t 和 p_U^t ，并检查其是否满足延迟要求。选择引起较小系统延时的可行计算模式。如果两种计算模式都不可行，计算任务将被放弃。

3.2 仿真结果分析

如图 2 所示，首先，可以观察到每个移动设备的能量水平在早期不断积累，最终稳定在扰动的能量水平附近，这是因为在所提出的算法中，李雅普诺夫漂移加惩罚函数在每个时隙均被最小化。

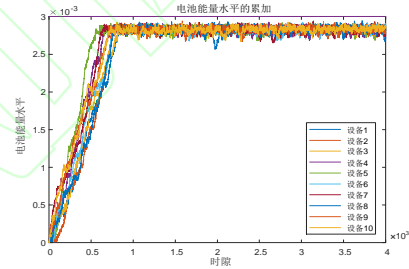


图 2 每个移动设备电池电量与时间的关系

Fig.2 The relationship between the battery power of each mobile device and time

图 3 展示了在多移动设备系统中，LGO 算法在选择计算模式的平均比率变化。在早期阶段，由于电池能量不足，计算任务倾向于被放弃。随着时隙的增加，电能队列开始收敛，算法使计算任务更倾向于 MEC 服务器执行。平均卸载比率达 0.99928，表明本文算法在最大化卸载计算任务数量方面有很好的性能。

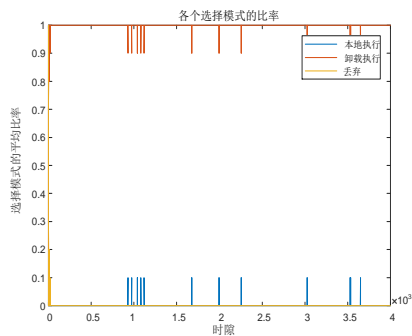


图3 选择模式的平均比率与时间的关系

Fig.3 The relationship between the average ratio of selection patterns and time

实际应用中主要关注两个指标：服务延时和系统能耗。本文所提算法与本地计算算法、全部卸载算法、动态卸载算法的比较结果如图4、图5所示。结果表明，LGO 算法在长期约束下可使系统平均延时和能耗保持稳定。虽然本地执行系统能耗较低，但能明显看出其延时较高，不适合表征用户体验感。与全部卸载算法相比，LGO 算法可以获得更优卸载发射功率，因此具有更低的系统能耗和服务延时。而动态卸载策略算法因无法管理采集的能量，导致任务经常因电池能量不足产生丢弃任务现象，同时仿真表明其平均任务丢弃率高达 36.09%。

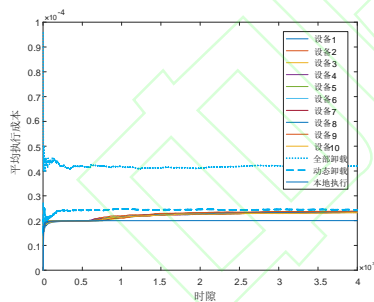


图4 系统能耗

Fig.4 System energy consumption

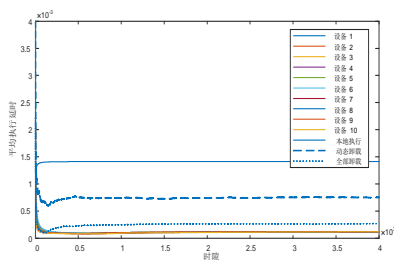


图5 服务延时

Fig.5 Service latency

由图6可以看出，随着最大距离的增加，卸载任务的平均比率逐渐降低。原因是信道功率增益随着每个移动设备和每个 MEC 服务器之间的距离而增加，这将导致更大的能耗和更长的执行延迟。因此，更多的移动设备将进行本地计算。

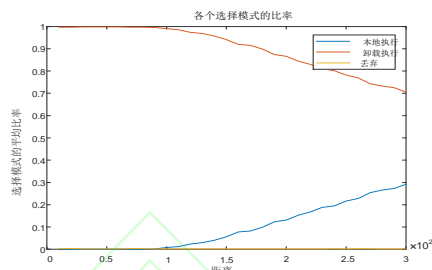


图6 距离与选择模式的关系

Fig.6 Relationship between distance and selection mode

为了方便观察惩罚权 V 对系统的影响，将场景设置为单移动设备单边缘服务器，其余不变。图7所示为惩罚权对延时和能耗的影响。系统能耗和服务延时随着惩罚权 V 的引入，实现了系统能耗和服务延时之间的平衡。随着惩罚权 V 增大，由于算法对能耗的要求逐渐提高，服务延时逐步上升，系统能耗逐步下降。图8表明，惩罚权 V 不会影响卸载率。总之，根据不同场景，可以通过寻找合适的惩罚权 V 值去获得合适的能耗与延时。

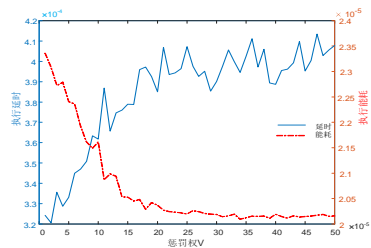


图7 惩罚权对延时和能耗的影响

Fig.7 Influence of penalty right on delay and energy consumption

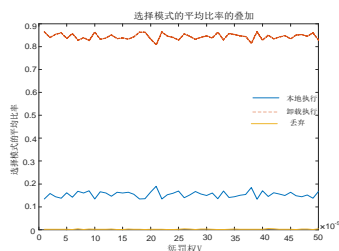


图8 惩罚权对选择模式的影响

Fig.8 Influence of penalty right on selection mode

4 结论

本文针对多台具有 EH 设备的边缘网络的资源分配问题,提出了 LGO 算法。在稳定电池电量前提下,以最小化服务延时和系统能耗联合成本为优化目标,并通过李雅普诺夫优化和贪婪策略理论获得最优解。仿真结果表明,所提算法可以将所有设备的电池水平稳定在预定阈值附近,同时可极大利用边缘服务器优势,并且在降低系统能耗和服务延时的效果上优于其他基准算法。本文研究没有考虑任务被请求概率,而实际情况上不可能每个任务都能被接收处理。考虑任务接收处理概率将是后续研究重点,以有效提升 LGO 算法的实际性能。

参考文献:

- [1] S. Sudevalayam, et al. Energy harvesting sensor nodes: survey and implications[J]. IEEE communications surveys & Tutorial, 2011, 13(3): 443-461.
- [2] Y. Sun, C. Song, S. Yu, et al. Energy-Efficient Task Offloading Based on Differential Evolution in Edge Computing System With Energy Harvesting[J]. IEEE Access, 2021, 9: 16383-16391.
- [3] T. Zhang, W. Chen, et al. Computation Offloading in Energy Harvesting aided Heterogeneous Mobile Edge Computing[J]. 2021 IEEE Transactions on Green Communications and Networking, 2021, 5(1): 552-565.
- [4] E. Zhao, et al. Dynamic Offloading and Resource Scheduling for Mobile-Edge Computing With Energy Harvesting Devices[J]. IEEE Transactions on Network and Service Management, 2021, 18(2): 2154-2165.
- [5] D. Zeng, S. Pan, et al. An MDP-Based Wireless Energy Harvesting Decision Strategy for Mobile Device in Edge Computing[J]. IEEE Network, 2019, 33(6): 109-115.
- [6] Y. Sun, et al. Latency Optimization for Mobile Edge Computing with Dynamic Energy Harvesting[C]. 2019 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCLOUD/SocialCom/SustainCom), 2019: 79-83.
- [7] MIN M H, XIAO L, CHEN Y, et al. Learning-based computation offloading for IoT devices with energy harvesting[J]. IEEE Transactions on Vehicular Technology, 2019, 68(2): 1930-1941.
- [8] 陈加法, 赵宜升, 高锦程, 陈忠辉. 基于混合能量收集的移动边缘计算系统资源分配策略[J]. 重庆邮电大学学报(自然科学版), 2021, 33(2): 193-201.
Chen Jiafa, Zhao Yisheng, Gao Jincheng, Chen Zhonghui. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2021, 33(2): 193-201.
- [9] JI L Y, GUO S T. Energy-efficient cooperative resource allocation in wireless powered mobile edge computing[J]. IEEE Internet of Things Journal, 2019, 6(3): 4744-4754.
- [10] 王凌, 吴楚格, 范文慧. 边缘计算资源分配与任务调度优化综述[J]. 系统仿真学报, 2021, 33(3): 509-520.
Wang Ling, Wu Chuge, Fan Wenhui. A Review of Edge Computing Resource Allocation and Task Scheduling Optimization[J]. Journal of System Simulation, 2021, 33(3): 509-520.
- [11] Y. Mao, et al. A survey on mobile edge computing: The communication perspective[J]. IEEE Communication Surveys & Tutorials, 2017, 19(4): 2322-2358.
- [12] L. B. Huang, et al. Utility optimal scheduling in energy harvesting networks[J]. IEEE/ACM Transactions on Networking, 2013, 21(4): 1117-1130.
- [13] 付主木, 王俊朋, 司鹏举, 陶发展. 基于李雅普诺夫随机优化的车辆边缘计算资源管理研究[J/OL]. (2021-02-04)[2021-06-28]. 控制与决策: 1-8, <https://doi.org/10.13195/j.kzyjc.2020.1211>.
Fu Zhumu, Wang Junpeng, Si Pengju, Tao Fengfa. Research on Vehicle Edge Computing Resource Management Based on Lyapunov Stochastic Optimization [J/OL]. Control and decision: 1-8, (2021-02-04)[2021-06-28]. <https://doi.org/10.13195/j.kzyjc.2020.1211>.
- [14] 徐思雅, 邢逸斐, 郭少勇, 杨超, 邱雪松, 孟洛明. 基于深度强化学习的能源互联网智能巡检任务分配机制[J]. 通信学报, 2021, 42(5): 191-204.
Xu Siya, Xing Yifei, Guo Shaoyong, Yang Chao, Qiu Xuesong, Meng Luoming. Energy Internet Intelligent Patrol Task Assignment Mechanism Based on Deep Reinforcement Learning [J]. Journal of Communications, 2021, 42(5): 191-204.
- [15] Y. Mao, et al. Dynamic computation offloading for mobile-edge computing with energy harvesting devices[J]. IEEE Journal of Selected Areas Communications, 2016, 34(12): 3590-3605.