

移动边缘计算中分布式异构任务卸载算法

夏士超 姚枝秀 鲜永菊 李云*

(重庆邮电大学移动通信技术重庆市重点实验室 重庆 400065)

摘要: 随着物联网(IoT)迅速发展, 移动边缘计算(MEC)在提供高性能、低延迟计算服务方面的作用日益明显。然而, 在面向IoT业务的MEC(MEC-IoT)时变环境中, 不同边缘设备和应用业务在时延和能耗等方面具有显著的异构性, 对高效的任务卸载及资源分配构成严峻挑战。针对上述问题, 该文提出一种动态的分布式异构任务卸载算法(D2HM), 该算法利用分布式博弈机制并结合李雅普诺夫优化理论, 设计了一种资源的动态报价机制, 并实现了对不同业务类型差异化控制和计算资源的弹性按需分配, 仿真结果表明, 所提的算法可以满足异构任务的多样化计算需求, 并在保证网络稳定性的前提下降低系统的平均时延。

关键词: 边缘计算; 物联网; 博弈论; 李雅普诺夫优化; 异构任务卸载

中图分类号: TN92

文献标识码: A

文章编号: 1009-5896(2020)12-2891-08

DOI: 10.11999/JEIT190728

A Distributed Heterogeneous Task Offloading Methodology for Mobile Edge Computing

XIA Shichao YAO Zhixiu XIAN Yongju LI Yun

(Chongqing Key Laboratory of Mobile Communication Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: With the rapid development of the Internet of Things (IoT), Mobile Edge Computing (MEC) becomes increasingly effective in improving processing capacity and providing low-latency computing services. However, in the time-varying MEC-IoT environment, heterogeneous devices and applications cause serious challenges on efficient task offloading and resource allocation. A Distributed Dynamic Heterogeneous task offloading Methodology (D2HM) algorithm is proposed in this paper by exploiting game theory and Lyapunov optimization, which can achieves heterogeneous control and allocation of computation resources by dynamic quote price mechanism. Simulation results show that the proposed algorithm can meet the diverse computing needs of heterogeneous tasks and reduce the average delay of the system while ensuring network stability.

Key words: Mobile Edge Computing (MEC); Internet of Things (IoT); Game theory; Lyapunov optimization; Heterogeneous task offloading

1 引言

移动互联网和物联网快速发展和融合, 促使移

动终端设备和数据流量呈指数式增加, 与此同时, 新兴物联网应用不断涌现, 进一步促进计算密集型服务及异构业务爆炸性增长。移动边缘计算作为一种新兴和有前景的计算范式, 在物联网业务进一步驱动下, 在应用数据类型、期望延迟、设备类型等方面存在着显著的异构性, 为此, 移动边缘计算要实现高效实时的资源配置及出色的用户体验, 还需要具备多样化业务处理和差异化流量特征支持的能力。此外, 据CISCO VNI最新报告预测^[1], 到2022年全球物联网设备将达到260亿台, 对未来海量IoT设备信息的收集与传统的集中式管理控制已变得极不现实, 分布式计算、存储及其资源优化配置也已成为当前研究的热点问题。

近年来, 业务云端化、连接泛在化和数据多样

收稿日期: 2019-09-19; 改回日期: 2020-05-30; 网络出版: 2020-07-09

*通信作者: 李云 liyun@cqupt.cn

基金项目: 国家自然科学基金(61671096), 重庆市“科技创新领军人才支持计划”(CSTCCXLJRC201710), 重庆市基础科学与前沿技术研究项目(cstc2017jcyjBX0005), 重庆邮电大学博士研究生高端人才培养项目(BYJS201806)

Foundation Items: The National Natural Science Foundation of China (61671096), The Chongqing Science and Technology Innovation Leading Talent Support Program (CSTCCXLJRC201710), The Chongqing Research Program of Basic Science and Frontier Technology (cstc2017jcyjBX0005), The Chongqing University of Posts and Telecommunications PhD High-end Talent Training Project (BYJS201806)

化的面向物联网应用的移动边缘计算(Mobile Edge Computing for Internet of Things, MEC-IoT)得到了广泛的研究^[2,3]。其中,在任务卸载迁移、节能等方面,文献[4]提出了一种基于阈值的任务卸载策略,采用集中式优化方式,将计算资源分配建模为凸优化问题,以实现最小化系统能耗的目的。文献[5]提出一种分层的边缘云架构,并以最小化设备能耗和任务处理时延为目标,提出一种基于乘子法的任务卸载和资源分配机制。而针对移动网络流量分布不均的问题,文献[6]为了降低边缘服务器的长期能耗,结合李雅普诺夫优化理论提出一种边缘服务器休眠控制策略。文献[7]则以车载通信网络为场景,将资源分配过程建模为一种多轮顺序组合拍卖机制,使请求车辆卸载效益提高的同时最大化边缘服务节点的效益。为了优化边缘服务器的负载及边缘节点任务的资源差异化需求,文献[8]通过结合强化学习为任务匹配选择合适的边缘服务器,并结合终端用户的社会属性,进一步实现网络的负载均衡。一些研究则利用分布式方式来解决资源分配的问题^[6-8],其中,文献[9]为了减轻本地MEC服务器负载,用户可以通过宏基站之间的高速链路将任务卸载到临近区域的空闲MEC服务器上,并将任务卸载问题建模为一种势博弈,以优化整个系统的卸载能耗为目标,提出了一种分布式任务卸载及资源分配策略。而文献[10]为了激励云端共享资源,优化云端服务器计算能耗,提出了一种基于组合拍卖的计算资源分配模型,相似地,文献[11]为了提高云端计算资源利用率并减少用户卸载时延,将用户建模为买方,云端建模为卖方,设计了一种计算资源双向拍卖分配机制。

虽然上述研究工作对传统云计算环境中任务卸载策略进行了充分研究,然而,不同边缘设备和应用业务在时延和能耗等方面的不同需求,对异构IoT业务采用传统优化方式已无法满足日益明显的差异化需求;此外,由于实际MEC-IoT中复杂时变的网络环境及流量特征,需要对不同应用及业务的能耗和时延进行动态优化以提高用户的业务体验。

本文通过分析动态的MEC-IoT环境中不同业务类型卸载时延和计算资源需求的异构性,建立多样化的任务卸载模型,设计了差异化任务下的分布式资源分配分析方法;为实现对任务卸载和计算资源分配的弹性控制和按需分配,运用博弈论与李雅普诺夫优化理论,在用户收益与时延之间进行折中,设计一种资源的动态报价方法,并提出一种动态的分布式异构任务卸载算法(Distributed Dynamic Heterogeneous tasks offloading Method-

ology, D2HM),最后,通过仿真验证了本文所提算法的有效性。

2 系统模型

图1所示为一种典型的MEC-IoT异构任务卸载排队模型。假设该模型共有 M 个不同类型的物联网设备或异构类型的物联网业务(IoT Devices or Applications, IDAs), N 个边缘服务器(MEC Servers, MSs)。定义 m_i 表示第 i 个IDAs, $\forall i \in \{1, 2, \dots, M\}$, n_j 表示第 j 个MSs, $\forall j \in \{0, 1, \dots, N\}$, 特别地, $j=0$ 表示本地计算。

假设每个IDAs任务量到达过程服从泊松分布,定义 $a_i(t)$ 表示 m_i 在时隙 t 内到达的任务量。用 $Q_i(t)$ 表示 m_i 在时隙 $t \in \{0, 1, \dots\}$ 的任务队列积压, $\mathbf{Q}(t) = \{Q_1(t), Q_2(t), \dots, Q_M(t)\}$ 表示所有用户队列积压的集合。

2.1 队列积压模型

在第 t 时隙内, m_i 队列特征可以用三元组表示 $\Lambda_i = \langle Q_i(t), b_i(t), L_i \rangle$, 其中, L_i 表示 m_i 队列的单位处理密度, $b_i(t) = \sum_{j=0}^N b_{ij}(t)$ 表示 m_i 在第 t 时隙内卸载的总任务量, 特别地, $b_{i0}(t)$ 表示在本地的计算任务量

$$b_{ij}(t) = \frac{f_{ij}(t)}{L_i} \tau \quad (1)$$

其中, $f_{ij}(t)$ 表示第 j 个服务器分配给 m_i 的CPU处理速率, τ 为单位时隙长度。进一步, 可得 m_i 任务队列积压在第 $t+1$ 时隙为

$$Q_i(t+1) = \max [Q_i(t) - b_i(t), 0] + a_i(t), \quad \forall t \in \{0, 1, \dots\} \quad (2)$$

假设系统运行 T 时间长度, 为了保证队列稳定性, 要求队列积压在时间平均意义上满足式(3)条件

$$\overline{Q_i} = \lim_{T \rightarrow \infty} \sup \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \{Q_i(t)\} \leq \infty \quad (3)$$

2.2 卸载效用模型

在MEC-IoT环境中, 任务的卸载效用模型定义为非递减、凸的或两次可微函数^[12]。根据文

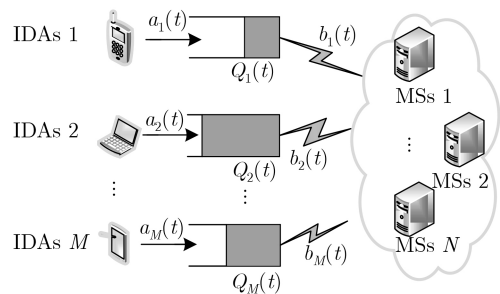


图1 一种典型的异构任务卸载模型

献[13,14], 定义第 t 时隙中, m_i 的任务卸载效用函数为 $u_i(t) = \sum_{j=0}^N u_{ij}(t)$

$$u_{ij}(t) = \alpha_i \lg(1 + b_{ij}(t)) \quad (4)$$

其中, α_i 是 m_i 的任务卸载权重系数。

2.3 通信成本模型

将任务卸载到云端需要经过3个过程: 任务上传, 任务计算处理及结果返回。由于结果返回时的任务量很小, 因此, 忽略计算结果返回时的通信成本。根据文献[14], 在数据的上传过程中, 任务的传输通信成本正比于任务卸载量, 即

$$C_{ij}^{\text{bw}}(t) \propto h(b_{ij}(t)) \quad (5)$$

其中, $h(b_{ij}(t))$ 是 $b_{ij}(t)$ 的函数, C_{ij}^{bw} 为通信成本, 本文假设数据通信成本为

$$C_{ij}^{\text{bw}}(t) = \sigma_{ij} b_{ij}(t) \quad (6)$$

其中, σ_{ij} 表示单位任务通信成本, 特别地, $\sigma_{i0} = 0$ 表示任务在本地计算不产生通信成本。

2.4 计算能耗成本模型

CPU动态调频调压(Dynamic Voltage Frequency Scaling, DVFS)技术允许处理器动态调节主频, 使CPU处于合适的主频设置, 达到节省功耗的目的。本文根据文献[15,16], 定义 m_i 在时隙 t 内将任务发送到 n_j 的计算能耗成本为

$$E_{ij}^{\text{cp}} = \kappa_j f_{ij}^2(t) \tau \quad (7)$$

其中, E_{ij}^{cp} 为计算成本, κ_j 是与芯片架构相关的有效能量成本系数。

3 分布式动态异构任务卸载算法

在时变的MEC-IoT环境下, 海量的IoT设备和应用业务在时延和能耗等方面具有显著的异构性。为提供差异化的计算服务, 将服务提供方(边缘服务器)作为卖方, 通过差异化的资源报价为移动终端提供不同的计算、存储资源; 而将移动终端作为买方, 通过自身的时延、能耗需求购买云端资源。卖方对单位计算资源报价越高, 获得的单位资源收益越高, 然而会降低买方的购买意愿, 从而将更多的计算任务在本地终端处理, 而造成云端资源的空闲和移动终端的队列积压。因此, 买卖双方之间是一种非合作博弈, 移动终端在卸载收益和卸载时延之间存在着折中关系。本节基于买卖博弈理论对边缘服务器的资源分配和价格策略进行建模, 并利用李雅普诺夫优化理论优化移动终端的卸载时间。

3.1 基于Lyapunov优化的买卖博弈模型分析

首先, 本节引入一种有偿的任务卸载模型[17], 将任务卸载端IDAs定义为买方, 任务处理代理端

MSs定义为卖方。买方向卖方购买CPU计算频率, 同时卖方根据自身的负载、成本等条件对计算资源进行报价。

对于IDAs, 定义在时隙 t 内向MEC服务器 n_j 购买计算资源 $f_{ij}(t)$ 所支付的费用成本为

$$c_{ij}(t) = p_{ij}(t) f_{ij}(t) \tau \quad (8)$$

其中, $p_{ij}(t)$ 为单位计算频率资源支付价格。

3.1.1 买方/IDAs博弈模型设计

IDAs卸载任务到MSs的过程中, 不仅需要向卖方支付一定的费用, 还会产生额外通信开销, 根据式(5), 式(7)和式(9), 可定义买方最大化收益问题为

$$\max U_{bi}(t) = \sum_{j=0}^N (u_{ij}(t) - C_{ij}^{\text{bw}}(t) - c_{ij}(t)) \quad (9a)$$

$$\text{s.t. } 0 \leq b_i(t) \leq Q_i(t), \forall i \in \{1, 2, \dots, M\} \quad (9b)$$

$$f_j^{\min} \leq f_{ij}(t) \leq f_j^{\max}, \forall j \in \{0, 1, \dots, N\} \quad (9c)$$

$$\bar{Q}_i = \lim_{T \rightarrow \infty} \sup \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{Q_i(t)\} \leq \infty \quad (9d)$$

其中, 式(9b)表示 m_i 卸载任务量不能超过其任务队列积压量, 式(9c)表示 m_i 向MSs请求的CPU计算频率不能低于MSs最小CPU频率 $f_j^{\min}$ 且不能超过MSs最大CPU频率 $f_j^{\max}$, 式(9d)保证队列稳定性。

通常, 云端服务器具有较高的CPU计算性能, 买方将任务卸载到云端服务器, 可有效提高任务的处理效率, 但需要向卖方支付一定的费用, 还会产生额外的通信开销, 往往会造成买方收益减少, 而将任务完全在本地计算处理, 又会导致队列长度过大, 不能保证队列稳定性。根据Little定律, 平均排队时延与平均队列长度成正比[18], 即 $\bar{W}_i = \bar{Q}_i / \lambda_i$, 其中 $\bar{W}_i$ 表示任务的平均排队时延, 因此买方收益和排队时延存在着折中关系。此外, 由于物联网环境下不同IDAs卸载时延和能耗的异构性, 需要针对不同IDAs进行资源的差异化分配。针对上述问题, 本文在买方利用Lyapunov优化理论, 通过对队列长度的异构控制, 实现不同IDAs的差异化需求, 该方法的优点在于不需要知道IDAs任务达到的分布信息, 只需要知道当前队列积压长度。

首先, 定义Lyapunov函数 $L(Q_i(t))$ 作为 m_i 任务队列积压度量指标

$$L(Q_i(t)) = \frac{1}{2}(Q_i(t))^2 \quad (10)$$

为了表征从一个时隙到下一个时隙Lyapunov函数 $L(Q(t))$ 的变化程度, 定义条件Lyapunov漂移为 $\Delta(Q_i(t)) = \mathbb{E}\{L(Q_i(t+1)) - L(Q_i(t)) | Q_i(t)\}$ (11)

根据Lyapunov优化理论, 本文引入drift-plus-penalty函数用以权衡 m_i 的卸载收益和时延之间的关系, 同时, 为了满足IDAs队列积压稳定性的同时最大化其卸载收益, 需要最小化drift-plus-penalty函数的上界

$$\Delta(Q_i(t)) - V_i \mathbb{E}\{U_i(t) | Q_i(t)\} \quad (12)$$

其中, $V_i \geq 0$ 是一个非负可控参数, 通过参数 V_i 可以对 m_i 队列长度和卸载收益进行折中。根据文献[18], 可得drift-plus-penalty表达式的上界。即

定理 1:

$$\begin{aligned} \Delta(Q_i(t)) - V_i \mathbb{E}\{U_{b_i}(t) | Q_i(t)\} &\leq B_i + Q_i(t) \lambda_i \\ &- \mathbb{E}\{Q_i(t) b_i(t) | Q_i(t)\} - V_i \mathbb{E}\{U_{b_i}(t) | Q_i(t)\} \end{aligned} \quad (13)$$

其中, $B_i = \frac{1}{2} \left((b_i^{\max}(t))^2 + \lambda_i^2 \right)$, $b_i^{\max}(t)$ 表示在时隙 t 中 m_i 能传输的最大任务量, 且 $b_i^{\max}(t) = Q_i(t)$ 。

根据定理1, 结合机会主义最小化期望值(opportunistically minimizing an expectation)的理论, 可得到买方的收益函数为

$$\begin{aligned} \max \quad & u_{b_i}(f_{ij}(t)) = V_i U_{b_i}(t) + Q_i(t) b_i(t) \\ \text{s.t.} \quad & \text{式(9b), 式(9c)和式(9d)} \end{aligned} \quad (14)$$

进一步, 根据文献[18]可得到如下推理:

推理1: 假设任务的到达率 λ 始终在网络系统容量 Λ 的范围内, 即 $\lambda + \epsilon \mathbf{1} \in \Lambda$ 。其中, $\epsilon \geq 0$ 表示任务到达率 λ 和网络系统容量 Λ 之间的距离度量, $\mathbf{1}$ 表示有 M 个元素的单位向量。假设系统运行经过了 T 时间长度, 则 m_i 卸载收益的时间平均边界, 以及队列积压的时间平均边界满足式(15)和式(16)

$$-\bar{U}_i = -\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{U_i(t)\} \leq -U_i^* + \frac{B}{V_i} \quad (15)$$

$$\bar{Q}_i = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\{Q_i(t)\} \leq \frac{B - V_i U_i^*}{\epsilon} \quad (16)$$

其中, U_i^* 表示 m_i 时间平均收益的最优值。式(15)和式(16)说明了买方的卸载收益和卸载时延是一种 $[O(1/V), O(V)]$ 形式的折中。

3.1.2 卖方/MSs博弈模型设计

对于MSs, 定义在时隙 t 内向 m_i 提供计算频率资源所获得的收益为 $c_{ji}(t)$, 根据式(8)可得

$$c_{ji}(t) = c_{ij}(t) = p_{ij}(t) f_{ij}(t) \tau \quad (17)$$

IDAs将任务卸载到MSs, 并根据MSs的报价支付一定的费用, MSs通过确定最优卖价来最大化自身的收益, 可得到卖方的最大化收益问题为

$$\max_{p_{ij}} u_{s_j}(p_{ij}(t)) = \sum_{i=1}^M (c_{ij}(t) - E_{ij}^{\text{cp}}) \quad (18a)$$

$$\text{s.t. } p_{ij}(t) \geq 0 \quad (18b)$$

3.2 策略分析

3.2.1 买方/IDAs策略分析

买方为了最大化自身收益, 会根据控制参数 V_i 、队列积压 $Q_i(t)$ 及卖方的报价等状态信息决定购买策略。根据式(14)可得 $\frac{\partial^2 u_{b_i}(f_{ij}(t))}{\partial (f_{ij}(t))^2} = -V_i \frac{\alpha_i \tau^2 L_i^{-2}}{(1 + \tau f_{ij}(t) L_i^{-1})^2 \ln 2} \leq 0$, 因此 $u_{b_i}(f_{ij}(t))$ 是关于 $f_{ij}(t)$ 的凸函数。又因为式(9b)、式(9c)和式(9d)是仿射函数, 因此买方优化问题可以利用拉格朗日乘子法求解, 定义拉格朗日函数为

$$\begin{aligned} L_{b_i}(f_{ij}(t), \lambda, \nu, v) \\ = u_{b_i}(f_{ij}(t)) + \sum_{j=0}^N (\lambda_j (f_{ij}(t) - f_j^{\max}(t))) \\ - \sum_{j=0}^N \nu_j (f_j^{\min}(t) - f_{ij}(t)) \\ + v \left(\sum_{j=0}^N b_{ij}(t) - Q_i(t) \right) \end{aligned} \quad (19)$$

利用KKT(Karush-Kuhn-Tucker)条件[19], 以求得最优的计算资源 $f_{ij}^*(t)$ 为

$$f_{ij}^*(t) = \frac{\alpha_i}{\Theta} - \frac{1}{\tau L_i^{-1}} \quad (20)$$

其中 $\Theta = \left(\frac{-Q_i \tau L_i^{-1} - \lambda_j^* + \nu_j^* - v^* \tau L_i^{-1}}{V_i} + \frac{\sigma_{ij} \tau}{L_i} + p_{ij} \tau \right) \ln 2$ 。

3.2.2 卖方/MSs策略分析

根据式(18), 单位计算频率资源报价 $p_{ij}(t)$ 越高, n_j 收益就会越高。然而对于IDAs, 就要付出更大的成本从MSs购买资源, 显然, 随着MSs报价的逐渐增大, IDAs从 n_j 购买计算资源的意愿将会降低, 转而购买其他服务器的资源, 进而影响到 n_j 的收益。因此MSs存在一个最优的报价使得IDAs和MSs的收益都能达到最优, 将式(20)代入式(18)可得

$$\frac{\partial^2 u_{s_j}(p_{ij}(t))}{\partial (p_{ij}(t))^2} = - \left(\frac{\Phi \Theta \ln 2 + \kappa \tau \alpha_i}{\Theta^2} + \frac{\kappa \tau f_{ij}^*}{\Theta} \right) \ln 2 \leq 0 \quad (21)$$

其中, $\Phi = \left(\frac{-Q_i \tau L_i^{-1} - \lambda_j^* + \nu_j^* - v^* \tau L_i^{-1}}{V_i} + \frac{\sigma_{ij} \tau}{L_i} + \kappa \tau f_{ij}^* \right)$ 。

由式(21)可知, MEC服务器 n_j 的收益是关于 $p_{ij}(t)$ 的凸函数, 且式(18a)是仿射函数, 因此, 卖

方优化问题可以运用拉格朗日乘子法来求解，构造拉格朗日函数为

$$L_{s_j}(p_{ij}(t), \boldsymbol{\mu}) = u_{s_j}(p_{ij}(t)) - \sum_{i=1}^M \mu_i p_{ij}(t) \quad (22)$$

利用KKT条件可求得 n_j 对 m_i 的最优报价策略为

$$p_{ij}^*(t) = (f_{ij}^*(t)\tau - \mu_i^*) \frac{\Theta^2}{\alpha_i \tau^2 \ln 2} + 2\kappa f_{ij}^*(t) \quad (23)$$

3.3 均衡存在性分析

本节证明了最优解 $(f_{ij}^*(t), p_{ij}^*(t))$ 是SE均衡解，并通过定义1及引理1说明了均衡解存在性。

定义1 当MEC服务器 n_j 报价 $p_{ij}(t)$ 固定时，如果

$$L_{b_i}(f_{ij}^*(t), \lambda^*, \nu^*, v^*) = \sup (L_{b_i}(f_{ij}(t), \lambda^*, \nu^*, v^*)) \quad (24)$$

同时，当IDAs向MEC服务器 n_j 申请的CPU计算频率 $f_{ij}(t)$ 固定时，如果

$$L_{s_j}(p_{ij}^*(t), \boldsymbol{\mu}^*) = \sup (L_{s_j}(p_{ij}(t), \boldsymbol{\mu}^*)) \quad (25)$$

那么， $(f_{ij}^*(t), p_{ij}^*(t))$ 是SE均衡解，即 $(f_{ij}^*(t), p_{ij}^*(t)) = (f_{ij}^{SE}(t), p_{ij}^{SE}(t))$ ，可由如下3个引理证明。

引理1 当MEC服务器 n_j 报价 $p_{ij}(t)$ 固定时， m_i 的效用函数 $L_{b_i}(f_{ij}(t), \lambda, \nu, v)$ 在 $f_{ij}^*(t)$ 处取最大值。

证明：由于 $\frac{\partial^2 L_{b_i}(f_{ij}(t), \lambda, \nu, v)}{\partial (f_{ij}(t))^2} \leq 0$ ，所以 $L_{b_i}(f_{ij}(t), \lambda, \nu, v)$ 是关于 $f_{ij}(t)$ 的凸函数，因此， $L_{b_i}(f_{ij}(t), \lambda, \nu, v)$ 在 $f_{ij}^*(t)$ 处取最大值，根据定义1， $f_{ij}^*(t)$ 是SE均衡解 $f_{ij}^{SE}(t)$ 。证毕

引理2 IDAs向MSs购买的最优CPU计算频率 $f_{ij}^*(t)$ 随着MEC服务器的报价 $p_{ij}(t)$ 的增加而降低。

证明：对式(20)求1阶导数可得

$$\frac{\partial f_{ij}^*(t)}{\partial p_{ij}(t)} = -\frac{\alpha_i}{\Theta^2} \ln 2 < 0 \quad (26)$$

式(26)表明， $f_{ij}^*(t)$ 是 $p_{ij}(t)$ 的单调递减函数，即当服务器报价增加时，IDAs购买的CPU计算资源会降低，导致MEC服务器收益变小。本文通过 $\partial L_{s_j}(p_{ij}(t), \boldsymbol{\mu}) / \partial p_{ij}(t) = 0$ 来求解MEC服务器的最优报价^[20]。证毕

引理3 当IDAs向 n_j 购买CPU计算频率 $f_{ij}^*(t)$ 固定时， $L_{s_j}(p_{ij}(t), \boldsymbol{\mu})$ 是关于 $p_{ij}(t)$ 的凸函数。

证明：对式(22)有 $\frac{\partial^2 L_{s_j}(p_{ij}(t), \boldsymbol{\mu})}{\partial (p_{ij}(t))^2} \leq 0$ ，所以MEC服务器 n_j 收益函数是关于报价 $p_{ij}(t)$ 的凸函数，因此， $L_{s_j}(p_{ij}(t), \boldsymbol{\mu})$ 在 $p_{ij}^*(t)$ 处取最大值，根据定义1， $p_{ij}^*(t)$ 是SE均衡解 $p_{ij}^{SE}(t)$ 。证毕

综上所述， $(f_{ij}^*(t), p_{ij}^*(t))$ 是最优卸载决策，同时为SE均衡解，即 $(f_{ij}^*(t), p_{ij}^*(t)) = (f_{ij}^{SE}(t), p_{ij}^{SE}(t))$ 。

4 仿真结果及分析

在本节利用MATLAB对资源动态报价，队列时延及收益等方面进行仿真，最后，与OCA算法^[13]，DGA算法^[17]及OMP算法进行对比，验证本文所提D2HM算法性能。

OCA(Online Centralized task Allocation): 采用在线的集中式任务卸载策略，并基于Lyapunov优化理论在系统收益和队列积压之间进行折中；

DGA(Decentralized Greedy task Allocation): 采用贪婪式先进先出的任务卸载模型，并基于买卖博弈激励机制的一种分布式任务卸载算法；

OMP(Only MEC server Process): 仅采用云端资源进行异构任务卸载及CPU计算资源分配。

4.1 仿真场景设置

根据文献^[13]，本文主要仿真参数设置如下，效用收益因子 $\alpha_i \in \{2, 1, 0.5\}$ ，本地最小CPU频率 $f_{i0}^{\min}(t)$ 为100 MHz，MEC服务器最小CPU频率 $f_{ij}^{\min}(t)$ 为150 MHz， $j \in \{1, 2, \dots, N\}$ ，本地最大CPU频率 $f_{i0}^{\max}(t)$ 为2000 MHz，MEC服务器最大CPU频率 $f_{ij}^{\max}(t)$ 为10000 MHz， $j \in \{1, 2, \dots, N\}$ ，单位时隙长度 $\tau = 1$ s，价格更新步长设置为 5×10^{-8} 。

4.2 性能分析

图2描述了1个IDAs和3个MSs的MEC-IoT场景中CPU计算频率随时隙数的动态报价及动态资源分配。由**图2(a)**可知，随着时隙变化，CPU频率逐渐收敛，这是因为在队列积压稳定之前，任务积压量随着时隙数的增加而增加，为了保证IDAs任务队列的稳定性，所需要处理的任务量也将随着增加，因此，需要向MSs购买更多的CPU计算频率，直到队列积压达到稳定状态。从**图2(b)**中可以看出，由于IDAs购买的CPU计算频率随着时隙数的增加而增加，MSs为了最大化自身收益，报价也将不断增加，直到队列积压达到稳定状态。

图3描述了多用户场景下，CPU频率、任务队列积压及卸载收益随时隙数的变化过程。场景中包含有3个不同属性的卖方MSs($\sigma_1 = 0.06, k_1 = 1 \times 10^{-7}$; $\sigma_2 = 0.06, k_2 = 4 \times 10^{-7}$; $\sigma_3 = 0.24, k_3 = 4 \times 10^{-7}$)。由**图3(a)**中可知，随着时隙数的增加，IDAs所购买的CPU频率逐渐增加，这是由于IDAs的队列积压随着时隙数的增加而增加，如**图3(b)**所示，直到收敛到稳定值。此外，从**图3(c)**中可知，随着队列积压收敛到稳定值，IDAs卸载收益也将达到稳定状态。根据IDAs在时延和计算需求方面的异构性，通过调整控制参数 V_i 可以控制IDAs队列积压的长度，因此设置不同的控制参数 V_i 可以表征IDAs在时

延方面的异构性。图3(b)描述了队列积压随着控制参数 V_i 的增加而增加,对于处理密度和时延要求较高的IDAs,可以通过调整控制参数 V_i 来满足IDAs对卸载时延的要求,然而,根据图3(c)可知,虽然通过减小控制参数 V_i 值满足IDAs的低时延要求,但其卸载收益却明显降低,这也验证了IDAs的卸载收益和卸载时延的折中关系。

图4描述了不同算法的平均队列积压和系统卸载收益的性能比较。场景中包含4个IDAs

($V_1 = 50, L_1 = 1000; V_2 = 50, L_2 = 1400; V_4 = 150, L_4 = 1800$) 和 3 个 M S s ($\sigma_1 = 0.06, k_1 = 1 \times 10^{-7}; \sigma_2 = 0.06, k_2 = 4 \times 10^{-7}; \sigma_3 = 0.24, k_3 = 4 \times 10^{-7}$)。由于贪婪DGA算法将每个时隙到达的任务全部卸载出去,因此DGA算法在每个时隙都不会有任务积压,但不能获得更高系统收益;OCA采用集中式的任务卸载算法,并在队列积压与收益之间进行折中,在牺牲了部分队列积压的情况下,在队列稳定之前获得高于DGA的收益;由于OMP仅采用云

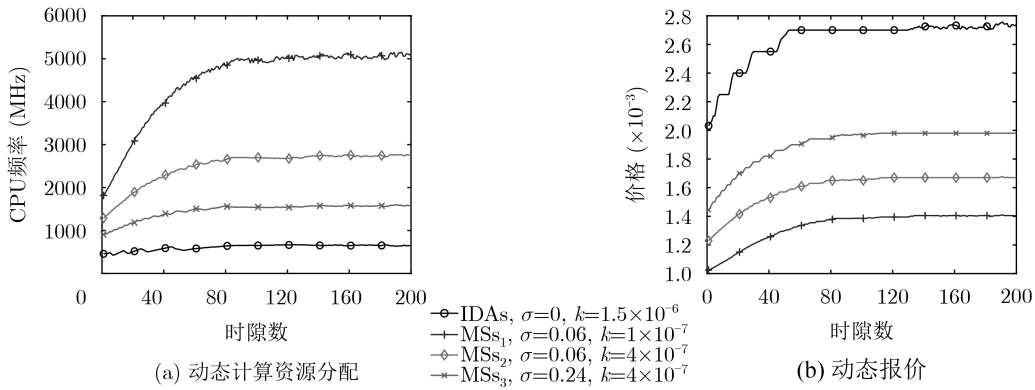


图2 CPU计算频率动态报价过程

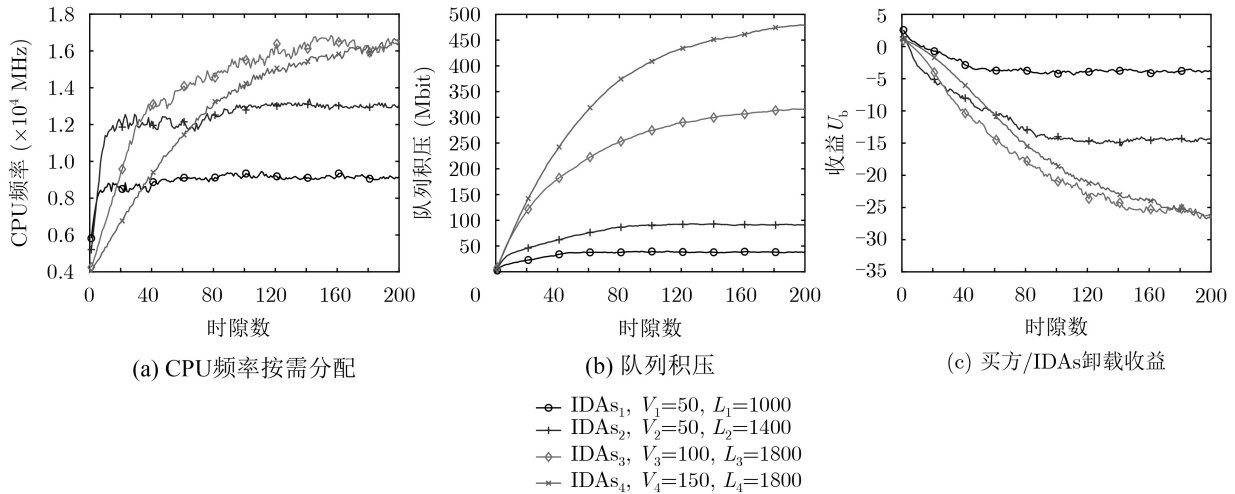


图3 队列积压及CPU频率的按需分配

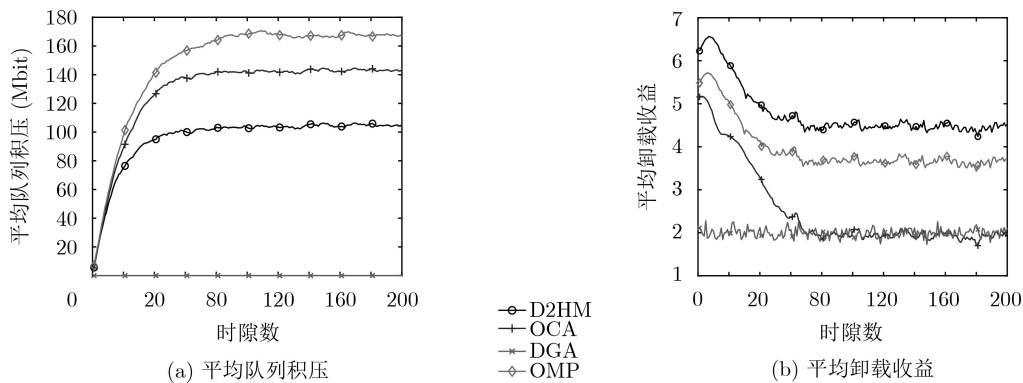


图4 不同算法性能对比

端资源进行任务的处理, 造成平均队列积压最大, 但根据任务的异构性进行差异化任务卸载及资源分配, 可获得的收益较高; 同理, D2HM算法通过差异化的资源报价在队列积压和平均卸载收益得到了更好的性能。

5 结束语

本文针对不同边缘设备和应用业务在时延和能耗等方面具有差异化需求的特点, 建立多样化任务卸载模型, 并为实现对任务卸载和计算资源分配的弹性控制和按需分配, 利用博弈论与李雅普诺夫优化理论, 设计了一种计算资源的动态报价方法, 并在用户收益与时延之间进行折中。通过实验仿真表明本文提出的分布式动态异构任务卸载算法的任务卸载弹性控制和资源按需分配的有效性和合理性。在MEC-IoT环境中, 无线网络环境的复杂性也是影响IoT设备任务卸载和资源分配的重要因素, 在未来工作将进一步研究在时变无线传输环境中, 对任务传输和资源分配的联合优化。

参考文献

- [1] CISCO. Cisco visual networking index: Forecast and trends, 2017–2022[R]. 2018.
- [2] MOURA J and HUTCHISON D. Game theory for multi-access edge computing: Survey, use cases, and future trends[J]. *IEEE Communications Surveys & Tutorials*, 2019, 21(1): 260–288. doi: [10.1109/COMST.2018.2863030](#).
- [3] YU Wei, LIANG Fan, HE Xiaofei, *et al.* A survey on the edge computing for the internet of things[J]. *IEEE Access*, 2018, 6: 6900–6919. doi: [10.1109/ACCESS.2017.2778504](#).
- [4] YOU Changsheng, HUANG Kaibin, CHAE H, *et al.* Energy-efficient resource allocation for mobile-edge computation offloading[J]. *IEEE Transactions on Wireless Communications*, 2017, 16(3): 1397–1411. doi: [10.1109/TWC.2016.2633522](#).
- [5] 代美玲, 刘周斌, 郭少勇, 等. 基于终端能耗和系统时延最小化的边缘计算卸载及资源分配机制[J]. 电子与信息学报, 2019, 41(11): 2684–2690. doi: [10.11999/JEIT180970](#).
DAI Meiling, LIU Zhubin, GUO Shaoyong, *et al.* A computation offloading and resource allocation mechanism based on minimizing devices energy consumption and system delay[J]. *Journal of Electronics & Information Technology*, 2019, 41(11): 2684–2690. doi: [10.11999/JEIT180970](#).
- [6] WANG Shuo, ZHANG Xing, YAN Zhi, *et al.* Cooperative edge computing with sleep control under nonuniform traffic in mobile edge networks[J]. *IEEE Internet of Things Journal*, 2019, 6(3): 4295–4306. doi: [10.1109/JIOT.2018.2875939](#).
- [7] 张海波, 栾秋季, 朱江, 等. 基于移动边缘计算的V2X任务卸载方案[J]. 电子与信息学报, 2018, 40(11): 2736–2743. doi: [10.11999/JEIT180027](#).
ZHANG Haibo, LUAN Qiuji, ZHU Jiang, *et al.* V2X task offloading scheme based on mobile edge computing[J]. *Journal of Electronics & Information Technology*, 2018, 40(11): 2736–2743. doi: [10.11999/JEIT180027](#).
- [8] 王汝言, 聂轩, 吴大鹏, 等. 社会属性感知的边缘计算任务调度策略[J]. 电子与信息学报, 2020, 42(1): 271–278. doi: [10.11999/JEIT190301](#).
WANG Ruyan, NIE Xuan, WU Dapeng, *et al.* Social attribute aware task scheduling strategy in edge computing[J]. *Journal of Electronics & Information Technology*, 2020, 42(1): 271–278. doi: [10.11999/JEIT190301](#).
- [9] WANG Chang, DONG Chongwu, QIN Jinghui, *et al.* Energy-efficient offloading policy for resource allocation in distributed mobile edge computing[C]. 2018 IEEE Symposium on Computers and Communications, Natal, Brazil, 2018: 366–372. doi: [10.1109/ISCC.2018.8538612](#).
- [10] HUU T T and THAM C K. An auction-based resource allocation model for green cloud computing[C]. 2013 IEEE International Conference on Cloud Engineering, Redwood City, USA, 2013: 269–278. doi: [10.1109/IC2E.2013.21](#).
- [11] JIN Along, SONG Wei, and ZHUANG Weihua. Auction-based resource allocation for sharing cloudlets in mobile cloud computing[J]. *IEEE Transactions on Emerging Topics in Computing*, 2018, 6(1): 45–57. doi: [10.1109/TETC.2015.2487865](#).
- [12] ZHOU Chongyu, THAM C K, and MOTANI M. Online auction for truthful stochastic offloading in mobile cloud computing[C]. 2017 IEEE Global Communications Conference, Singapore, 2017: 1–6. doi: [10.1109/GLOCOM.2017.8254630](#).
- [13] LYU Xinchun, NI Wei, TIAN Hui, *et al.* Optimal schedule of mobile edge computing for internet of things using partial information[J]. *IEEE Journal on Selected Areas in Communications*, 2017, 35(11): 2606–2615. doi: [10.1109/JSAC.2017.2760186](#).
- [14] CAO Bin, XIA Shichao, HAN Jiawei, *et al.* A distributed game methodology for crowdsensing in uncertain wireless scenario[J]. *IEEE Transactions on Mobile Computing*, 2020, 19(1): 15–28. doi: [10.1109/TMC.2019.2892953](#).
- [15] MAO Yuyi, ZHANG Jun, and LETAIEF K B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices[J]. *IEEE Journal on Selected Areas in Communications*, 2016, 34(12): 3590–3605. doi: [10.1109/JSAC.2016.2611964](#).
- [16] 张海波, 李虎, 陈善学, 等. 超密集网络中基于移动边缘计算的

- 任务卸载和资源优化[J]. 电子与信息学报, 2019, 41(5): 1194–1201. doi: [10.11999/JEIT180592](https://doi.org/10.11999/JEIT180592).
- ZHANG Haibo, LI Hu, CHEN Shanxue, *et al.* Computing offloading and resource optimization in ultra-dense networks with mobile edge computation[J]. *Journal of Electronics & Information Technology*, 2019, 41(5): 1194–1201. doi: [10.11999/JEIT180592](https://doi.org/10.11999/JEIT180592).
- [17] CAO Bin, XIA Shichao, LI Yun, *et al.* An incentive-based workload assignment with power allocation in ad hoc cloud[C]. 2017 IEEE International Conference on Communications, Paris, France, 2017: 1–6. doi: [10.1109/ICC.2017.7997026](https://doi.org/10.1109/ICC.2017.7997026).
- [18] NEELY M. Stochastic Network Optimization with Application to Communication and Queueing Systems[M]. Morgan & Claypool, 2010: 25–38.
- [19] BOYD S and VANDENBERGHE L. Convex Optimization[M]. Cambridge, UK: Cambridge University Press, 2004: 326–327. doi: [10.1017/CBO9780511804441](https://doi.org/10.1017/CBO9780511804441).
- [20] WANG Beibei, HAN Zhu, and LIU K L R. Distributed relay selection and power control for multiuser cooperative communication networks using stackelberg game[J]. *IEEE Transactions on Mobile Computing*, 2009, 8(7): 975–990. doi: [10.1109/TMC.2008.153](https://doi.org/10.1109/TMC.2008.153).
- 夏士超: 男, 1991年生, 博士生, 研究方向为物联网、边缘计算.
- 姚枝秀: 女, 1994年生, 硕士生, 研究方向为物联网、边缘计算.
- 鲜永菊: 女, 1973年生, 副教授, 主要研究领域为无线移动通信、物联网和边缘计算.
- 李 云: 男, 1974年生, 教授, 博士生导师, 主要研究领域为无线移动通信、物联网和边缘计算.

责任编辑: 马秀强