计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.200500098

边缘计算场景中基于虚拟映射的隐私保护卸载算法

余雪勇　陈　涛
江苏省无线通信重点实验室　南京 210003

摘　要　随着移动边缘计算(Mobile Edge Computing,MEC)和无线充电技术(Wireless Power Transmission,WPT)的诞生和发展,越来越多的计算任务被卸载至MEC服务器以进行处理,并借助WPT技术为终端设备供电,以缓解终端设备计算能力受限和设备能耗过高的问题。由于卸载的任务和数据往往携带用户个人使用习惯等信息,因此将任务卸载到MEC服务器进行处理会导致新的隐私泄露问题。针对上述问题,文中首先对计算任务的隐私量进行定义,并设计了能够降低用户在MEC服务器累积隐私量的虚拟任务映射机制;然后,综合考虑映射机制与隐私约束的优化,提出了一种具有隐私保护效果的在线隐私感知计算卸载算法;最后,对仿真结果进行分析发现,所提卸载方法能够使用户累积隐私量保持在隐私阈值内,达到了隐私保护的效果,同时提高了系统计算速率,降低了用户计算时延。

关键词:边缘计算;计算卸载;隐私保护;虚拟映射;神经网络
中图法分类号　TP393

Privacy Protection Offloading Algorithm Based on Virtual Mapping in Edge Computing Scene

YU Xue-yong and CHEN Tao
Wireless Communication Key Lab of Jiangsu Province,Nanjing 210003,China

Abstract　With the development of mobile edge computing (MEC) and wireless power transfer (WPT),more and more computing tasks are offloaded to the MEC server for processing. The terminal equipment is powered by WPT technology to alleviate the limited computing power of the terminal equipment and high energy consumption of the terminal equipment. However,since the offloaded tasks and data often carry information such as users' personal usage habits,tasks are offloaded to the MEC server for processing results in new privacy leakage issues. A privacy-aware computation offloading method based on virtual mapping is proposed in this paper. Firstly,the privacy of the computing task is defined,and then a virtual task mapping mechanism that can reduce the amount of privacy accumulated by users on the MEC server is designed. Secondly,the online privacy-aware computing offloading algorithm is proposed by considering the optimization of the mapping mechanism and privacy constraints jointly. Finally,simulation results validate that the proposed offloading method can keep the cumulative privacy of users below the threshold,increase the system calculation rate and reduce users' calculation delay at the same time.

Keywords　Edge computing,Computation offloading,Privacy protection,Virtual mapping,Neural network

1　引言

随着5G时代的到来,5G技术将更进一步地推动物联网、大数据、云计算的发展进程,智慧交通、智慧城市、位置服务、移动支付等新型服务模式和智能终端设备也逐渐普及,这导致数据规模呈指数增长,同时也对用户终端设备的数据处理与访问能力提出了更高的要求[1]。MEC的出现为此提供了可能的解决方案[2-4],通过将云计算中心的计算能力下沉,拓展至更靠近用户的边缘侧服务器,用户可以将计算任务卸载至较近的MEC服务器。随着卸载计算任务的距离缩短,时延与能耗也随之大幅降低;同时通过密集部署MEC服务器来满足海量设备同时连接的需求[5],也可以在时延敏感和计算密集型任务的场景中提供更好的服务[6]。WPT技术在通信领域一直备受青睐,近年来部分学者开始将WPT技术应用到物联网场景中。由于无需更换电池就可以通过服务器对设备进行无线充电[7],以缓解终端设备的能源限制,因此将WPT技术和MEC场景相结合有望解除物联网终端设备的两大基本限制[8-9]。

随着物联网设备的普及,不少学者围绕物联网隐私问题进行了研究[10-16],但多数研究都是在传统云计算场景的基础上[17]在数据加密、访问控制、身份认证等方面进行了探讨,关于MEC架构的物联网隐私保护的研究则较少。MEC支持的系统场景具备无线卸载功能,存在潜在的隐私泄露风险,即用户位置隐私和使用模式隐私[18]。文献[18]提出了一种基

到稿日期:2020-05-21　返修日期:2020-08-14　本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家自然科学基金资助项目(61871446);南京邮电大学自然科学基金项目(NY220047)
This work was supported by the National Natural Science Foundation of China(61871446) and Natural Science Foundation of Nanjing University of Posts and Telecommunications(NY220047).
通信作者:余雪勇(yuxy@njupt.edu.cn)

于马尔可夫决策过程的隐私感知任务卸载调度算法，可以在保持预先指定的隐私级别的同时实现较优的时延和能耗性能。在此基础上，文献[19]进一步分析了在医疗物联网场景中终端设备计算卸载的位置隐私和使用模式隐私，优化了终端能耗和计算时延，并引入了增强学习算法来求解最优的卸载速率和本地计算速率。文献[20]研究了用户的位置隐私保护，将少量用户的计算任务随机分流至距离用户较远的 MEC 服务器以达到保护用户位置隐私的目的，同时增加了终端的能耗，并采用经深度决策后的状态学习算法来快速求解最优的卸载决策。但上述研究都缺乏对用户卸载任务特征及频率所导致的隐私泄露问题的关注，因此无法解决恶意监听者通过 MEC 服务器获取用户终端卸载习惯和任务类型从而间接锁定用户而导致的隐私泄露问题。

本文提出了一种结合深度强化学习的虚拟映射卸载方法(Deep Reinforcement learning-based Mapping Offloading，DRMO)，该方法提高了边缘计算卸载场景中用户的隐私保护等级，同时优化了系统计算速率并降低了用户计算时延。为了能够在信道相干时间内快速解决复杂的组合优化问题，本文使用深度神经网络为在线卸载算法构建训练模型，并采用保序量化和自适应参数以实现快速收敛。训练模型将服务器与用户间的信道增益和用户终端生成的计算任务种类及生成概率作为输入信息，同时引入了隐私保护的训练约束，在保障系统计算速率最大化且满足隐私约束的条件下，训练输出多个用户的最优卸载决策。本文根据用户终端生成的计算任务种类及其概率定义了 MEC 服务器的用户累积隐私量，并采用虚拟任务映射机制合理减少卸载任务所包含的隐私量，当累计隐私量超出隐私阈值时，计算任务将移至本地进行处理，同时在 MEC 服务器上生成虚拟任务来降低累积隐私量。

本文第 2 节对系统模型和隐私保护问题进行了分析；第 3 节阐述了隐私量的定义、虚拟任务映射机制原理和在线隐私感知计算卸载方法的具体步骤；第 4 节对仿真结果进行了分析，并对比了不同算法的各项性能指标；最后总结了隐私保护卸载的研究结果，并对下一步的研究工作进行了展望。

2 系统模型及问题分析

2.1 系统模型

本文设定了一种由 N 个无线用户和单个 MEC 服务器组成的系统场景来模拟现实中的静态传感器网络或低功耗 IoT 系统，如图 1 所示，用户可通过连接 MEC 服务器进行无线充电和计算任务卸载。

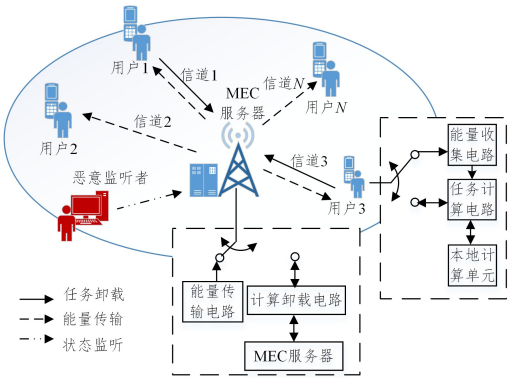


图 1 系统场景模型

Fig. 1 System scene model

MEC 服务器具有稳定的电力供应系统及用于无线充电的发射天线，可向其覆盖范围内的无线用户发射高频电磁波，且每个无线用户都拥有一个可充电的电池来存储能量，为设备计算提供支持。由于 MEC 服务器的计算能力是用户终端设备的三次量级[8,21]，因此无线用户可将本地计算任务卸载到 MEC 服务器中，以减少终端能耗，提升系统计算速率。为避免无线充电与任务卸载在同一频段执行时产生相互干扰，场景中的设备均采用时分复用电路。

本文将系统时间划分为连续的单位时间片段，每个时间片段时长为 T，T 小于信道的相干时间，即在静态物联网场景中信道保持恒定状态的最大时间差(一般为几秒钟，为了不失一般性，文中选取为 1s)[22-24]。单位时间 T 由用户终端设备的无线充电时间和计算卸载时间构成，由于 MEC 服务器的发射功率较大，因此返回计算结果的时间可忽略不计。采用时分复用方法，将单位时间 T 的前 aT 的时间用于 MEC 服务器为覆盖范围内的用户进行无线充电($a\in(0,1]$)，再划分 $\tau_i T$ 时间用于 MEC 服务器覆盖范围内的第 i 个用户进行计算任务卸载($\tau_i\in(0,1]$)。使用 h_i 表示单位时间 T 内 MEC 服务器与第 i 个无线用户间的信道增益，假定信道上行与下行状态始终相同，且在单位时间内保持不变，但在不同的单位时间内信道增益可能发生改变，再结合卸载决策将无线用户的计算模式分为本地计算模式和卸载计算模式。

2.1.1 本地计算模式

无线用户终端设备具有能量收集和本地计算的功能[8]，f_i 表示终端设备的处理器计算速度(cycle/second)，t_i 表示用于本地计算的时间($0\leqslant t_i\leqslant T$)，$\phi$ 表示终端处理器处理 1bit 计算任务所需的转数，则本地计算可处理的任务比特数为 $f_i t_i/\phi$。同时，终端设备本地计算所产生的能耗需满足 $k_i f_i^3 t_i\leqslant E_i$，其中，$k_i$ 表示计算能效系数[25]，E_i 表示用户 i 单位时间所收集的能量。为了处理尽可能多的计算任务，用户终端需将收集的能量全部用于计算，即最优计算时间 $t_i^*=T$，最优处理器计算速度 $f_i^*=(\frac{E_i}{k_i T})^{\frac{1}{3}}$，因此本地计算模式单位时间的计算速率如式(1)所示：

$$r_{L,i}^*(a)=\frac{f_i^* t_i^*}{\phi T}=(\frac{h_i}{k_i})^{\frac{1}{3}}\cdot\frac{(\mu P a)^{\frac{1}{3}}}{\phi} \tag{1}$$

其中，μ 表示能量收集效率，P 表示服务器无线充电的发射功率，L 表示当前时隙内用户的计算模式为本地计算模式(*Local*)。

2.1.2 卸载计算模式

在系统场景中，第 i 个用户用于计算卸载的时间用 $\tau_i T$ 表示，其中 $\tau_i\in[0,1]$，如图 2 所示。

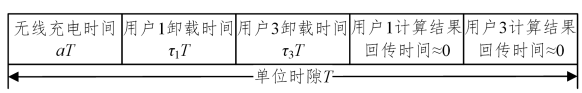


图 2 时分复用结构

Fig. 2 Time division multiplexing structure

服务器将计算结果返回用户所用的时间可忽略不计，因此 a,τ 需满足 $\sum_{i=1}^{N}\tau_i+a=1$。同理，为实现系统计算速率最大化，将收集的能量 E_i 全部用于任务卸载，根据能量收集效率和服务器无线充电的发射功率可得 $E_i=\mu P h_i a T$，此时卸载功率 $P_i^*=\frac{E_i}{\tau_i T}=\frac{\mu P h_i a}{\tau_i}$，卸载速率 $r_{O,i}^*(a,\tau_i)$ 则等于此时的信道

容量，根据香农公式可得：

$$r^*_{O,i}(a,\tau_i)=\frac{B\tau_i}{v_u}\log_2\left(1+\frac{\mu P a h_i^2}{\tau_i N_0}\right) \tag{2}$$

其中，O 表示当前时隙用户的计算模式为卸载计算模式(Off-loading)，v_u 表示通讯损耗系数(考虑到程序加密和打包，通常取 $v_u=1.1$)。

2.2 问题分析

MEC 服务器在计算用户卸载任务的同时也可统计用户卸载任务的种类及频率，从而为网络资源部署和服务器缓存内容提供数据分析依据。由于不同用户有不同的使用习惯，各个用户的卸载频率及计算任务种类也各不相同。恶意监听者通常会选择监听 MEC 服务器来获取用户的卸载情况，如果攻击者掌握了某用户的卸载习惯，就能通过监听 MEC 服务器各用户的卸载任务种类及频率来判断目标用户是否处于 MEC 服务器范围之内，如图 1 所示。

由于边缘节点防护资源有限，相比云计算中心，边缘网络更容易被攻击者发现和突破漏洞，且由于其同质性，攻击者一旦发现漏洞就可以从整个边缘网络监听并收集卸载结果，因此边缘网络具有攻击成本低、收益大的特点。

当无线用户选择将计算任务卸载至 MEC 服务器进行处理时，存在被恶意监听者攻击的风险。因此，在边缘计算场景中加强用户隐私保护的工作必不可少。

3 隐私感知的计算卸载方法

3.1 隐私度量及累积隐私量

利用目标用户卸载频率相对于该任务平均卸载概率的显著性，可量化计算任务的隐私属性[26]，即用量化的显著性来表示某种任务所蕴含的隐私量。本文采用卸载概率的比值关系来反映目标用户的显著性，对于时隙 T 内产生的卸载任务 $T(t)$，定义其隐私量 $q(t)$ 为：

$$q(t)=\ln\frac{p^A_{T(t)}}{\bar{p}_{T(t)}} \tag{3}$$

其中，$p^A_{T(t)}$ 为用户 A 卸载任务 $T(t)$ 的概率，$\bar{p}_{T(t)}$ 为所有用户卸载任务 $T(t)$ 的平均概率。

若 $p^A_{T(t)}>\bar{p}_{T(t)}$，则 $q(t)>0$，表示该时隙 T 内用户卸载该任务的隐私量为正；反之，则 $q(t)<0$，表示该时隙内用户卸载该任务的隐私量为负；当隐私量为正时，表明用户当前时隙处理的计算任务更具有个人特征，易暴露个人隐私；当隐私量为负时，表明用户当前时隙处理的计算任务有助于减少 MEC 服务器的累积隐私量，能降低用户个人隐私泄露的风险。随后，统计时间为 $0\sim t$，MEC 服务器统计到用户 A 的累积隐私量的计算公式如下：

$$Q(t)=\sum_{i=0}^{t}q(i)I_{MEC}(i) \tag{4}$$

其中，$I_{MEC}(t)=\{1,0\}$ 表示用户 A 在时隙 t 中的任务卸载情况。

从监听者的角度出发，在其监听 MEC 服务器中用户卸载规律和累积隐私量时，若某个用户的卸载规律与目标用户的相符，且该用户在 MEC 服务器的累积隐私量超过设定的阈值，则很大程度上可判定目标用户此时在该 MEC 服务器的服务范围内。虽然卸载的计算任务都包含一定的隐私量，且计算任务生成的概率不同，包含的隐私量有正也有负，但累积隐私量总体呈上升趋势。因此，监听者对上述判定的准确性与隐私量阈值有关，阈值越大，需要卸载的次数越多，判定的准确性越高；反之，需要卸载的次数越少，判定的准确性越低。

3.2 虚拟任务映射机制

在对计算任务所包含的隐私量进行量化并设置累积隐私量阈值后，用户终端可以感知到在当前环境卸载计算任务时暴露自身位置的可能性，从而增加卸载约束，在本地处理大部分计算任务，使 MEC 服务器监听到的累积隐私量始终保持较低的值，但这同时也会造成新的问题：1)在本地处理大量计算任务会增加用户终端的能耗，这违背了边缘计算的初衷；2)若监听者知道该隐私保护策略，将较低的卸载频率作为目标用户的判定标准，便可以优先锁定目标并进一步实施其他攻击手段。

本节提出了一种虚拟任务映射机制，通过适当调节用户卸载策略来控制用户在 MEC 服务器泄露的累积隐私量，从而减小被恶意监听者锁定的风险。在时隙 T 内，当用户选择将计算任务卸载至 MEC 服务器进行处理时，该机制先生成映射的虚拟任务，再计算用户累计隐私量，使得最终卸载次数与不考虑隐私时的卸载决策保持一致。假设目标用户生成的计算任务共有 5 种，其卸载概率和包含的隐私量如表 1 所列。由于任务 A 的卸载频率最高，其隐私量对 MEC 服务器的累积隐私量的影响最为显著。因此，根据用户卸载习惯在 MEC 服务器上预设虚拟任务映射机制 $f(x)$，生成映射后的虚拟任务 $task_{map}$，且 $task_{map}=f(x)\vdash\{x\mid x\in A,B,C,D,E\}$。当用户卸载任务 A 时，MEC 服务器正常处理任务 A，但在计算累积隐私量时将其映射为隐私量较小的任务 C。考虑到隐私量为负的情况，即将任务 A 映射为任务 B 会导致累积隐私量始终小于零，这反而使用户累积隐私量更具特征性，因此需选择合适的映射逻辑，才能合理降低用户在 MEC 服务器的累积隐私量。本文采用的映射逻辑如图 3 所示。当用户在 MEC 服务器的累积隐私量超过阈值时，将计算任务改为本地处理，并在 MEC 服务器生成隐私量最小的虚拟任务，以减少用户在服务器中的累积隐私量。

表 1 任务信息

Table 1 Information of tasks

任务种类	卸载概率	隐私量
A	0.4733	0.61221
B	0.21	−0.25587
C	0.1387	0.27013
D	0.1017	−0.40645
E	0.0763	−1.02943

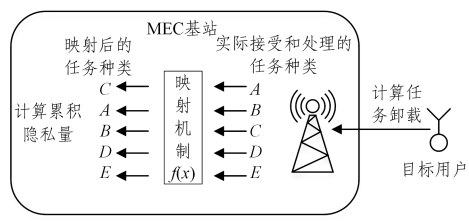


图 3 虚拟任务的映射逻辑

Fig. 3 Virtual task mapping logic

本文用指示函数 $I_F(t)$ 来表示用户 A 在时隙 T 内是否生成虚拟任务，由式(4)可得，MEC 服务器的累积隐私量的计算公式如下：

$$Q(t)=\sum_{i=0}^{t}(q(i)\cdot I_{MEC}(i)+q_0\cdot I_F(i)) \tag{5}$$

其中，q_0 表示用户卸载概率最低的任务所包含的隐私量。

3.3 算法描述

假设系统参数中只有信道增益 $h=\{h_i\mid i\in N\}$ 和用户卸载任务的种类具有时变性，其余参数均参考行业经验设置为定值。用户卸载决策取决于 MEC 服务器与用户间的信道增益 h_i，当信道增益较大时，用户倾向于将计算任务卸载至 MEC 服务器进行处理并获得计算结果，以达到节约终端能耗、降低计算时延和提升计算速率的目的。为了实现隐私保护，在 MEC 服务器上预设了虚拟任务映射逻辑，同时在训练过程中添加隐私约束，适时调整卸载策略，并在 MEC 服务器上生成虚拟任务。目标问题可描述为：

$$\begin{aligned}&Q^*(h)=\underset{x,\tau,a}{\text{maximize}}\ Q(h,x,\tau,a)\\&\text{subject to}\ \sum_{i=1}^{N}\tau_i+a\leqslant 1\\&\qquad a\geqslant 0,\tau_i\geqslant 0,\forall\, i\in N\\&\qquad x_i\in\{0,1\}\\&\qquad Q(t)\leqslant Q\_\text{target}\end{aligned} \tag{6}$$

其中，$Q(h,x,\tau,a)$ 为系统计算速率，结合式(1)与式(2)可以得出单位时间内的系统计算速率为多个用户(本地模式或卸载模式)的计算速率之和：

$$Q(h,x,\tau,a)=\sum_{i=1}^{N}w_i((1-x_i)r^*_{L,i}(a)+x_i r^*_{O,i}(a,\tau_i)) \tag{7}$$

其中，h 为信道增益，x_i 为第 i 个用户在当前时隙的卸载动作($x_i\in\{0,1\}$，0 表示本地处理，1 表示卸载到 MEC 服务器)，τ_i 为第 i 个用户在当前时隙分配的任务卸载时间占比($\tau_i\in(0,1)$)，a 为每个时隙无线充电的时间占比，ω_i 为第 i 个用户的权重系数，$Q(t)$ 和 Q_target 分别表示用户当前的累积隐私量和 MEC 服务器设置的隐私阈值，具体可见式(5)。

当在某个时隙 T，用户的卸载动作 $x_i=0$ 时，有 $\tau_i=0$，这是因为用户当前处于本地处理模式。若提前给出 x_i，那么原先难以解决的混合整数规划(Mixed Integer Programming，MIP)的非凸问题 P1 就可以转换为凸问题 P2，如式(8)所示：

$$\begin{aligned}&Q^*(h,x)=\text{maximize}\ Q(h,x,\tau,a)\\&\text{subject to}\ \sum_{i=1}^{N}\tau_i+a\leqslant 1\\&\qquad a\geqslant 0,\tau_i\geqslant 0,\forall\, i\in N\\&\qquad Q(t)\leqslant Q\_\text{target}\end{aligned} \tag{8}$$

因此，DRMO 卸载策略可分为两步求解：1)先根据信道状态训练生成卸载决策 x_i，再基于 x_i 优化时间分配问题以得到系统计算速率最优的卸载策略 x_t^*；2)计算卸载任务通过虚拟映射后的累积隐私量并调整卸载策略为 $\tilde{x}_t^*$，作为最终输出的卸载策略。DRMO 算法的框架图如图 4 所示。

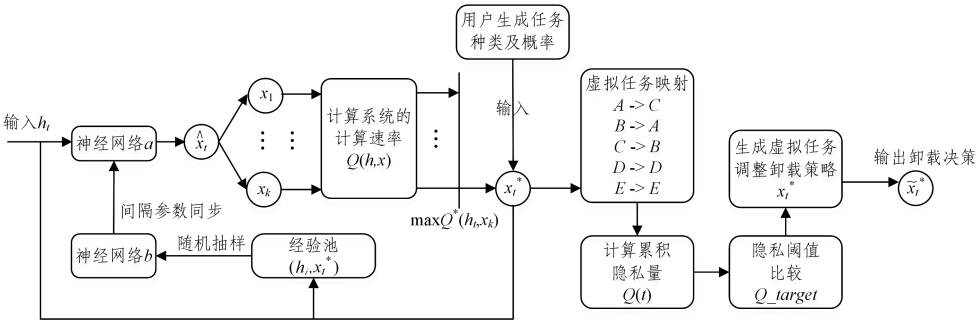


图 4 DRMO 算法的框架图

Fig. 4 Framework of DRMO algorithm

3.3.1 生成卸载决策 x_t^*

针对在线卸载方法需在信道相干时间内解决系统场景中无线充电和用户计算卸载时间的复杂组合优化的问题，DROO 算法在类似的 MEC 场景中构建了合适的神经网络并选取性能较优的激活函数，同时在训练过程中采用保序量化方法和自适应参数：首先，当获得初步训练结果时(某时隙内 N 个用户的二进制卸载决策 $\hat{x}_t$)，在保证用户间决策排列顺序不变的前提下进行有约束量化(量化为 $x_1,\cdots,x_k,k\in[1,K]$)，并减少不必要的随机探索，将状态搜索空间规模 K 从 $[1,2^N]$ 缩减至 $[1,N+1]$，随着 N 的增长，计算复杂度由原先的指数增长变为线性增长；然后，根据最优决策在量化决策中的排序规律，在计算系统计算速率时，设置固定的训练间隔 Δ，自适应地根据前 Δ 次训练经验动态调整当前时隙训练中量化决策状态的空间规模 K，即：

$$K_t=\begin{cases}N, & t=1\\ \min(\max(k^*_{t-1},\cdots,k^*_{t-\Delta})+1,N), & l \bmod \Delta=0\\ K_{t-1}, & \text{otherwise}\end{cases}$$

其中，t 为时隙，k_t^* 为 t 时隙中最优决策在量化决策中的索引排序，从而在保留随机性的同时进一步减小每次训练的计算量。

结合上述两种方法可以避免求解复杂的混合整数优化，并快速训练以生成时分复用参数及使系统计算速率最大化的用户卸载决策，降低了算法复杂度[27]。因此，本文引用 DROO 快速收敛的特性，采用同样的训练方法，以满足在线卸载的需求。

图 4 中，将信道增益 h_i 作为神经网络 a 的输入。训练过程中通过保序量化获得 K 个卸载策略 $x_i=\{x_i\mid i=1,\cdots,K\}$，进而使用一维二分式搜索获得 $Q^*(h,x)$ 对应的最优解(a^*，τ^*)及对应的卸载策略 x_t^*。同时将每一次迭代产生的最优卸载策略与信道增益 h_i 进行组合并存储至经验池 (h_t,x_t^*)，并以一定的训练间隔从经验池抽取样本来训练结构相同的神经网络 b，然后以一定的训练间隔同步更新神经网络 a 的参数，用于生成后续的卸载策略。

3.3.2 虚拟任务映射机制调整卸载策略 $x_t^*\rightarrow\tilde{x}_t^*$

当获得系统卸载速率最优的卸载策略时，为保证用户累积隐私量不超过阈值，在 MEC 服务器中采用虚拟任务映射

机制，将当前时隙内的用户卸载任务通过预设的逻辑映射为隐私含量相对较小的虚拟任务，再根据式(5)计算 MEC 服务器的累积隐私量 $Q(t)$。当用户在某个时隙的累积隐私量 $Q(t)$超过阈值 Q_targrt 时，将用户原先的卸载策略 x_t^* 改为本地处理，调整后的卸载策略记为$\widetilde{x}_t^*$，同时向 MEC 服务器发送控制指令，生成虚拟任务(虚拟任务为用户生成概率最小的任务)，以减小累积隐私量。累积隐私量的阈值可由其他用户在 MEC 服务器的累积隐私量取中位数获得，因此可有效隐匿用户的卸载特性，减小被监听者锁定的风险。DRMO 算法的具体步骤如算法 1 所示。

算法 1 DRMO 算法

输入：信道状态信息、用户计算任务种类和生成概率

输出：具有隐私保护效果的卸载策略

Step1 初始化训练参数；

Step2 将用户与 MEC 基站间的信道状态作为输入训练以生成初步卸载策略；

Step3 根据式(7)计算系统的计算速率，得到较优的卸载策略；

Step4 根据映射逻辑，计算用户在 MEC 服务器的累积隐私量；

Step5 根据虚拟任务机制调整卸载策略，得到最优策略。

4 仿真实验

本节利用 PyCharm 数值仿真并对比了 3 种卸载方法来验证上述模型和算法的有效性，选用与文献[27]相同的系统场景，其包含 10 个无线用户。为便于计算和比较，假设用户终端可能产生的计算任务类型有 5 种，统计型的经验规律表明内容型数据往往遵循少数内容经常被访问、多数内容很少被访问的规律。其中，内容被访问的频率与其被访问的排名成反比关系，称为 $Zipf$ 分布。因此实验中计算任务的产生概率使用 $Zipf$ 分布随机生成，即 $P(r)=\frac{C}{r^{\alpha}}$，其中，$r$ 表示某种任务出现频率的排序，C 与 α 均为常数，一般与数据信息内容相关，$P(r)$表示排序为 r 的任务的出现概率。

4.1 参数选取

本次实验中，用户任务卸载概率采用了符合 $Zipf$ 分布的随机概率(其中，$\alpha=1.1$，$C=0.1$)。用户 A 的任务卸载概率为[0.473 3，0.21，0.138 7，0.1017，0.076 3]。其他模型参数设置如表 2 所列。

表 2 参数设置

Table 2 Simulation parameters

参数	数值
MEC 服务器 WPT 充电功率 P/w	3
终端设备 WPT 接收效率 μ	0.51
终端设备计算能效系数 k_i	10^{-26}
终端处理 1 bit 数据所需要的转数 ϕ	100
信道噪声功率 N_0/w	10^{-10}
计算卸载带宽 B/MHz	2
神经网络训练间隔 δ/次	10
经验池抽取样本数量$\lvert\tau\rvert$/条	128
经验池大小/条	1024
优化器学习速率	0.01

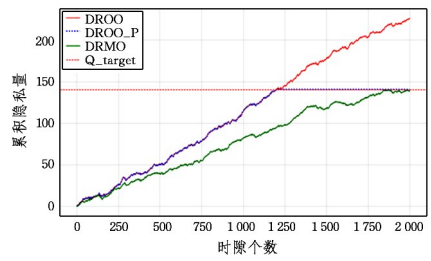
实验主要对比了 3 种算法：1)DROO 算法，不考虑用户隐私约束，在每个时隙 T 内，优化指标为 MEC 系统场景的计算速率，生成最优卸载策略；2)DROO_P 算法，在 DROO 算法的基础上考虑了用户的隐私保护问题，在每个时隙 T 内，在满足隐私约束的前提下，优化系统场景的计算速率，生成卸载策略；3)DRMO 算法，在 DROO 的基础上采用虚拟任务映射机制，并适时调整卸载策略，同时在 MEC 服务器中生成虚拟任务。

4.2 仿真结果分析

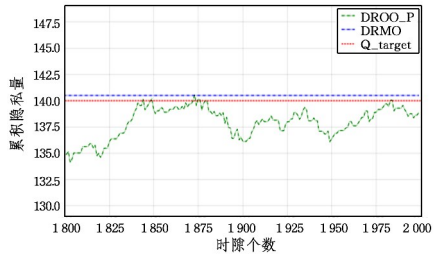
实验选取 MEC 系统场景中的某目标用户 A，根据信道增益状态，分别采用 DROO 算法、DROO_P 算法、DRMO 算法生成卸载策略。为了突出算法在隐私保护方面的有效性，实验对比了各算法的累积隐私量，同时验证算法在提升隐私保护性能的同时是否牺牲了太多其他指标，还对比了这几种算法在系统计算速率、用户计算时延、实际卸载次数以及算法执行时间方面的变化。

4.2.1 累积隐私量

实验多次统计了前 2 000 次卸载结果后 MEC 服务器的累积隐私量，变化趋势如图 5(a)所示。用户的累积隐私量随时间的增加总体呈上升趋势，但在个别时隙内也存在下降的情况(如第 1 500 个时隙处)。在第 1 250 个时隙左右，采用 DROO 算法与 DROO_P 算法时，用户在 MEC 服务器的累积隐私量已达到预设的隐私阈值 Q_targrt，此后 DROO 算法对应的累积隐私量继续攀升，DROO_P 对应的累积隐私量保持为隐私阈值范围的定值且不再改变。采用 DRMO 算法时，如图 5(b)所示，用户累积隐私量的增长趋势明显低于前两者，在第 1850 个时隙才达到隐私量阈值，且在之后的时隙用户累积隐私量开始在阈值内小范围波动，这说明 DRMO 算法具有更好的隐私保护效果。这是因为 DROO 算法没有考虑隐私保护约束，所以其对应的累积隐私量在达到隐私阈值后仍保持增长趋势。DROO_P 算法在 DROO 算法的基础上引入了隐私约束，当用户的累积隐私量达到阈值时会将该卸载的任务改为本地处理，当累积隐私量首次超过阈值后，后续的卸载策略因受到隐私约束而全部改为本地处理，因此累积隐私量不再增长，始终为定值。



(a)累积隐私量的变化



(b)累积隐私量的局部变化

图 5 累积隐私量的变化趋势

Fig. 5 Change trend of cumulative privacy

由于 DRMO 算法在 MEC 服务器上接收任务时采用了虚拟任务映射机制，减小了卸载概率最高的计算任务所包含的隐私量，因此其增长速率明显低于前者。当累积隐私量达到阈值后，DRMO 算法会将当前的卸载任务改为本地处理，并在 MEC 服务器上生成虚拟任务来降低累积隐私量，为后续的正常卸载提供了可能。

4.2.2 系统计算速率

图 6 给出了采用上述 3 种算法时，MEC 场景中系统计算速率随时隙的变化情况。当用户累积隐私量在第 1 250 个时隙达到阈值后，DROO_P 算法对应的系统计算速率发生明显衰落，仅为采用 DROO 算法的 80%左右，而 DRMO 算法对应的卸载速率仅在第 1 900 个和第 2 000 个时隙左右发生小幅变化。由系统计算速率公式可知，系统计算速率取决于卸载模式速率和本地计算速率，由式(1)可知本地计算速率与无线充电的时间占比有关，由式(2)可知卸载模式的计算速率与无线充电时间以及用户卸载时间占比有关，因此当 DROO_P 在累积隐私超过阈值后将后续所有任务都改为本地处理时，系统计算速率即为本地计算速率。MEC 服务器的算力明显强于用户终端，且训练得到的无线充电时间占比 a 也是时刻变化的，因此 DROO_P 算法的系统计算速率也随之发生变化。不同的是，DRMO 算法在达到隐私阈值后会适时地生成虚拟任务来降低累积隐私量，从而为后面的正常卸载做准备，仍发挥了 MEC 服务器高算力的优势，故其计算速率能够始终保持较高的水平。

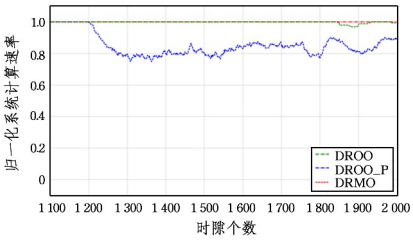


图 6 系统的计算速率对比

Fig. 6 Comparison of system calculation rates

4.2.3 实际卸载次数

为了进一步凸显 DRMO 算法的高效性，本文对 3 种算法的计划卸载次数和实际卸载次数进行了比较。最终得到的卸载策略为一连串的 0,1 卸载动作数组，因此通过统计其中元素 1 的个数可以获得实际卸载次数。计划的卸载次数为 MEC 服务器记录的卸载情况。

如图 7 所示，在 2000 个时隙内，若用户采用 DROO_P 算法，其计划卸载次数与实际卸载次数仅为 DROO 算法的 60%。

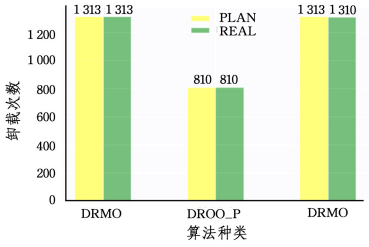


图 7 计划与实际卸载次数的对比

Fig. 7 Comparison of planned and actual offload times

通过多次统计发现，DRMO 算法对应的实际卸载次数与 DROO 算法相差不超过 10 次，这表明 DRMO 算法能在较高的计划卸载次数下保持较高的实际卸载次数。这是因为虚拟任务映射逻辑降低了高频任务的隐私量，从而减小了隐私约束对卸载决策的影响，同时累积隐私量达到隐私阈值后所采用的虚拟任务机制能适时调整卸载决策以降低累积隐私量，使得之后仍有正常卸载的可能。

4.2.4 用户计算时延

为了判断 DRMO 算法在隐私保护方面性能提升的同时是否大幅损失了其他性能，实验还比较了使用不同算法时计算时延的变化。根据图 2 所示的单位时间的结构图可知，当用户将计算任务卸载到 MEC 服务器时，计算时延为卸载任务所消耗的时间 $delay=\tau_i T$，若在本地处理，即为单位时间中减去无线充电剩下的时间 $delay=1-aT$。如图 8 所示，在选取的对比时隙中，DROO_P 的计算时延普遍最大，也存在在部分时隙三者的计算时延相同的情况，DRMO 算法对应的计算时延有 95%与 DROO 保持一致，仅在第 1 985 个时隙左右出现了其时延高于 DROO 的情况。这是由于用户卸载的时间往往短于本地处理的时间，因此用户卸载越多，计算时延越短。DROO_P 在累积隐私量达到阈值后，计算任务都为本地处理，因此其计算时延相比 DROO 会普遍增大，但当 DROO 和 DRMO 中用户因信道状态不佳而选择本地处理时，三者的计算模式相同，故计算时延也相等。由于累积隐私量达到隐私阈值后，DRMO 算法会适时地将原本的卸载任务改为本地处理并产生虚拟任务，以降低服务器的累积隐私量，会出现少数时隙时延增长的情况。

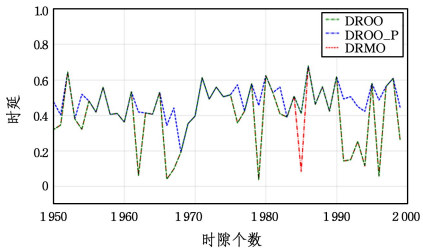


图 8 计算时延的对比

Fig. 8 Comparison of calculation delay

4.2.5 算法执行时间

最后比较了 3 种算法的执行时间。通过多次实验统计发现，取 30000 个时隙在个人 PC 上分别执行 3 种算法，DROO 算法的执行时间平均为 379.79 s，DROO_P 算法的执行时间平均为 468.12 s，DRMO 算法的执行时间平均为 407.09 s。3 种算法的执行时间都在一个量级，DROO_P 算法的执行时间相比 DROO 算法增加了 23%左右，而相同情况下 DRMO 算法的执行时间仅比 DROO 算法增加了 7%左右，考虑到 MEC 服务器的性能更强，算法执行的时间差距将会变得更小，这也使得通过调整卸载决策来保护用户隐私成为可能。

结束语 本文针对多用户单 MEC 服务器的场景，提出了一种基于虚拟映射的在线隐私感知卸载方法，通过降低用户在 MEC 服务器的累积隐私量来降低隐私泄露的风险。针对隐私保护的问题，提出了虚拟任务映射机制，降低了用户卸载任务所包含的隐私量，同时实现了适时降低 MEC 服务器

的累积隐私量以达到保护用户隐私的目的。实验结果表明，在相同信道条件下，本文提出的 DRMO 算法通过调节少量卸载决策就可以合理有效地降低用户在 MEC 服务器的累积隐私量，同时保持较高的系统计算速率和较低的计算时延。本文主要考虑了基础场景卸载策略中的隐私保护问题，而现实场景中的计算卸载任务种类繁多，真实卸载情况也受多种环境因素的影响，因此关于卸载策略中的隐私保护问题仍是未来研究的重点。

参 考 文 献

[1] LIN J，YU W，ZHANG N，et al. A Survey on Internet of Things：Architecture，Enabling Technologies，Security and Privacy，and Applications[J]. IEEE Internet of Things Journal，2017，4(5)：1125-1142.

[2] SUN X，ANSARI N. EdgeIoT：Mobile edge computing for the Internet of Things[J]. IEEE Communications Magazine，2016，54(12)：22-29.

[3] SABELLA D，VAILLANT A，KUURE P，et al. Mobile-edge computing architecture：The role of MEC in the Internet of Things[J]. IEEE Consumer Electronics Magazine，2016，5(4)：84-91.

[4] CORCORAN P，DATTA S K. Mobile-edge computing and the Internet of Things for consumers：Extending cloud computing and services to the edge of the network[J]. IEEE Consumer Electronics Magazine，2016，5(4)：73-74.

[5] ZHANG J L，ZHAO Y C，CHEN B，et al. Survey on data security and privacy-preserving for the research of edge computing[J]. Journal on Communications，2018，39(3)：1-21.

[6] YOUSEFPOUR A，ISHIGAKI G，GOUR R，et al. On reducing IoT service delay via fog offloading[J]. IEEE Internet of Things Journal，2018，5(2)：998-1010.

[7] BI S Z，HO C K，ZHANG R. Wireless powered communication：Opportunities and challenges[J]. IEEE Communications Magazine，2015，53(4)：117-125.

[8] WANG F，XU J，WANG X，et al. Joint offloading and computing optimization in wireless powered mobile-edge computing systems[J]. IEEE Transactions on Wireless Communications，2018，17(3)：1784-1797.

[9] BI S Z，ZHANG Y J. Computation Rate Maximization for Wireless Powered Mobile-Edge Computing with Binary Computation Offloading[J]. IEEE Transactions on Wireless Communications，2017，17(6)：4177-4190.

[10] YANG Y C，WU L F，YIN G S，et al. A survey on security and privacy issues in Internet-of-Things[J]. IEEE Internet Things Journal，2017，4(5)：1250-1258.

[11] REN H，LI H W，DAI Y S，et al. Querying in Internet of Things with privacy preserving：Challenges，solutions and opportunities[J]. IEEE Network，2018，32(6)：144-151.

[12] ALRAWAIS A，ALHOTHAILY A，HU C Q，et al. Fog computing for the Internet of Things：Security and privacy issues[J]. IEEE Internet Computing，2017，21(2)：34-42.

[13] WANGS，LUO Y，SUN L，et al. Design of Privacy Preservation Authentication Protocol in Pervasive Environment[J]. Computer Engineering，2012，38(6)：129-131.

[14] LIU Y，SHAREN G W，ZHANG L W. Secure Authentication Approach of Mobile Terminals with Privacy Protection in the Mobile Cloud Computing[J]. Journal of Chongqing University of Technology (Natural Science)，2019，33(10)：161-167.

[15] WANG Y，GE H B，FENG A Q. Computation Offloading Strategy in Cloud-Assisted Mobile Edge Computing[J]. Computer Engineering，2020，46(8)：27-34.

[16] HU J，YANG G，CHEN Z Y，et al. Research of Privacy-preserving Technology in Wireless Sensor Network Data Aggregation[J]. Computer Engineering，2012，38(15)：134-138.

[17] BOTTA A，DONATO W D，PERSICO V，et al. Integration of cloud computing and Internet of Things：A survey[J]. Future Generation Computer System，2016，56：684-700.

[18] HE X F，LIU J，JIN R C，et al. Privacy-aware offloading in mobile-edge computing[C]// IEEE Global Communications Conference. Singapore，2017：1-6.

[19] MIN M H，WAN X Y，XIAO L，et al. Learning-based privacy-aware offloading for healthcare IoT with energy harvesting[J]. IEEE Internet of Things Journal，2019，6(3)：4307-4316.

[20] HE X F，JIN R C，DAI H Y. Deep PDS-learning for privacy-aware offloading in MEC-enabled IoT[J]. IEEE Internet of Things Journal，2019，6(3)：4547-4555.

[21] YOU C S，HUANG K B，CHAE H. Energy efficient mobile cloud computing powered by wireless energy transfer[J]. IEEE Journal on Selected Areas in Communications，2016，34(5)：1757-1771.

[22] BULTITUDE R. Measurement，characterization and modeling of indoor 800/900 MHz radio channels for digital communications[J]. IEEE Communications Magazine，1987，25(6)：5-12.

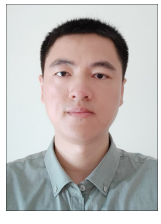
[23] HOWARD S J，PAHLAVAN K. Doppler spread measurements of indoor radio channel[J]. Electronics Letters，1990，26(2)：107-109.

[24] HERBERT S，WASSELL I，LOH T，et al. Characterizing the spectral properties and time variation of the in-vehicle wireless communication channel[J]. IEEE Transactions on Communications，2014，62(7)：2390-2399.

[25] GUO S T，XIAO B，YANG Y Y，et al. Energy-efficient dynamic offloading and resource scheduling in mobile cloud computing[C]// IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications. San Francisco，2016：1-9.

[26] ZHAO X，PENG J H，YOU W. A Privacy-aware Computation Offloading Method Based on Lyapunov Optimization[J]. Journal of Electronics & Information Technology，2020，42(3)：704-711.

[27] LIANG H，BI S Z，ZHANG Y J. Deep Reinforcement Learning for Online Computation Offloading in Wireless Powered Mobile-Edge Computing Networks[J]. IEEE Transactions on Mobile Computing，2020，19(11)：2581-2593.



YU Xue-yong，born in 1979，Ph.D，associate professor. His main research interests include Internet of Thing (IoT)，mobile edge computing and radio resource management on heterogeneous wireless networks.