

## 一种能效优先的物联网任务协同迁移策略

周龙雨, 杨宁, 乔冠华, 张科, 郑其林

(电子科技大学信息与通信工程学院, 四川 成都 611731)

**摘要:** 移动边缘计算通过在数据源端执行通信和计算操作, 缩减了物联网业务的传输和处理时延。然而, 针对大量的物联网设备连接数, 海量碎片化的数据同时汇聚在边缘计算平台, 会显著地增加前传链路的流量负载和边缘服务器的计算负荷。为了应对这一挑战, 基于多样化的物联网应用需求, 通过最优化设备传输的选择控制, 设计了一种任务协同迁移策略, 以实现时延约束下的系统最小能量消耗。在缺少信道状态完美先验信息的条件下, 提出了一种基于深度增强学习的资源管理算法, 以较低的复杂度获得了最优的任务卸载决策。仿真结果表明, 与随机的传输选择策略相比, 所提出的算法能够显著地降低系统的能量消耗, 并且满足任务的服务时延。

**关键词:** 物联网; 边缘计算; 增强学习; 资源消耗; 任务协同

**中图分类号:** TN929.5

**文献标识码:** A

**doi:** 10.11959/j.issn.2096-3750.2019.00105

## Energy efficiency priority IoT task collaborative migration strategy

ZHOU Longyu, YANG Ning, QIAO Guanhua, ZHANG Ke, ZHENG Qilin

School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

**Abstract:** Mobile edge computing can reduce transmission delay and data processing delay for IoT applications by executing communication and computing operation in the edge network. However, for a large number of IoT device connections, massive service data is simultaneously gathered on the edge computing platform, which will significantly increase the traffic load of the forward link and the computing load of the edge server. In order to meet this challenge, based on diversified IoT application requirements, a task collaborative migration strategy was designed to realize the minimum energy consumption of the system under time delay constraints by optimizing the selection control of equipment transmission. In the absence of perfect channel state prior information, a resource management algorithm based on deep reinforcement learning was proposed to obtain the optimal offloading decision with lower complexity. The simulation results show that the proposed algorithm can significantly reduce the energy consumption of the system and meet the service delay of the task compared with the random transmission selection strategy.

**Key words:** Internet of things(IoT), edge-computing, reinforcement learning, resource consumption, task collaboration

### 1 引言

作为互联网的一种全域拓展模式, 物联网通过异构的通信网络实现设备间信息的传输和共享, 使得传感器、移动终端、智慧车辆等能够独

立寻址的物理设备实现互联互通。物联网技术演进为新型应用的开发和实施, 如环境监测、交通物流、城市管理和工业制造等, 提供了更加可靠的信息服务环境和平台。虽然物联网技术的兴起有效地促进了各种应用的不断涌现, 但是这些应用大多具有密集型计算处理需求和严苛的时延

收稿日期: 2019-04-06; 修回日期: 2019-04-20

基金项目: 国家重点研发项目 (No.2018YFC0807101); 四川省科技重点项目 (No.2018GZ0092)

**Foundation Items:** The National Key R&D Program of China (No.2018YFC0807101), Science & Technology Department of Sichuan Province (No.2018GZ0092)

要求,需要在较短时间内完成数据处理并提供实时决策的能力<sup>[1-2]</sup>。然而物联网设备大多由电池供电,并且设备配置的嵌入式计算模块处理能力较弱。如何在资源受限的物联网设备上执行频繁的计算和控制操作成为制约物联网发展的重要瓶颈。

移动边缘计算(MEC, mobile edge computing)技术为解决上述问题带来了新思路<sup>[3]</sup>。通过将计算资源部署在靠近用户端的 Sink 节点处,就近为应用提供所需的计算服务,有效地降低了系统能量消耗与处理时延。考虑物联网中应用需求的多样性、边缘服务器资源状态的动态性以及大规模物联网设备对通信和计算资源的竞争性等因素<sup>[4-5]</sup>, MEC 应用于复杂的物联网环境中面临新的挑战和问题,具体如下。

1) 如果海量数据同时上传到边缘服务器,不仅会产生数据“洪泛”效应,还会造成网络拥塞现象。因此,要在设备间设计一种任务传输选择机制。在当前时隙,基于设备的不同时延要求,选择最优的设备集合执行任务卸载操作。

2) 从边缘服务器的资源可用性角度来看,物联网设备之间存在资源的竞争关系,在满足设备的时延要求下,需要基于系统能量消耗实现资源可用性和业务需求的动态匹配。

目前,已有大量研究将 MEC 技术应用于各种复杂的场景中。参考文献[6]在社交网络中使用边缘计算技术使任务调度和资源分配的业务服务时延最小化。为了协同管理计算资源和缓存资源,参考文献[7]和参考文献[8]将最优化资源消耗问题转化为凸函数来求解。针对数据的调度协同问题,参考文献[9]基于 MEC 提出了一种集中式云和分布式 MEC 的混合光纤无线网络架构,支持多种网络技术,具有易扩展性、高容量和高可靠性等特点。

然而,以上研究工作仅考虑了资源有限性,忽略了任务差异性。任务差异性表现在传感器节点会产生不同种类的任务,如当不同设备执行任务卸载时,会有不同的服务时延要求。为了给设备提供更好的服务,需要在满足差异化的业务服务时延要求下,优化数据传输控制决策,进而使系统的能量消耗最小化<sup>[10]</sup>。

为了应对以上挑战,本文提出了一种基于增强学习的最优任务协同卸载策略。首先分析了现有的无线传感网络中密集节点的资源分配问题<sup>[11-12]</sup>,进

而将边缘计算引入物联网场景中,建立系统模型并分析了业务的差异性,描述了时延约束下能量最小化的优化问题,通过增强学习算法对提出的非线性模型进行求解,得出资源调度的优化方案。本文的主要贡献如下。

1) 本文基于多样化的物联网设备的性能要求,首次引入了不同任务间的协同选择卸载策略,建立了服务时延约束条件下的系统能量消耗模型。

2) 为了分析设备端的不同任务被迁移到服务器端所产生的能量消耗,采用模糊逻辑方法建立任务量和能量消耗之间的映射关系,填补了数据“洪泛”效应带来的网络拥塞影响的空白。

3) 在物联网场景中,首次引入深度增强学习算法来应对非线性优化模型所带来的时间复杂度的挑战,得到了最优任务协同卸载策略,并与随机的任务选择策略进行对比,验证了本文所提算法的有效性。

## 2 系统模型

典型的物联网系统模型如图 1 所示,包括智慧城市、智能交通、健康医疗等应用。面对各种应用多样化的性能要求和差异化的资源使用情况,不同类型的传感器通过任务协同卸载策略,实现了在时延约束条件下的能量消耗最小化。本文在满足时延约束条件下,以系统能量消耗为目标来建立优化模型,系统能量消耗涉及服务器计算消耗、链路传输消耗等。因此,本节将依次建立边缘计算系统的网络模型、通信模型、计算模型以及服务器任务队列模型。

### 2.1 网络模型

在一个由各种物联网设备和应用组成的边缘网络环境中,定义集合  $a = \{1, 2, \dots, N\}$  为传感器节点的索引集,  $N$  为传感器总数;  $b = \{1, 2, \dots, M\}$  为边缘服务器的索引集,  $M$  为边缘服务器总数。对于资源受限的传感器来说,传感器节点可以根据自身需求,将任务直接卸载到边缘服务器端,或者通过设备直通(D2D, device-to-device)技术由中继节点再卸载到边缘服务器端。为了描述传感器节点产生数据的特征<sup>[13]</sup>,本文将生成数据的连续时间划分成等间隔的离散时间。时隙划分如图 2 所示,  $T = \{1, 2, \dots, L\}$  为时刻点,  $\Delta t$  为数据生成周期。在每个等间隔的时隙内,传感器节点的数据到达率服从均值为  $\lambda$  的泊松分布,模型表示为

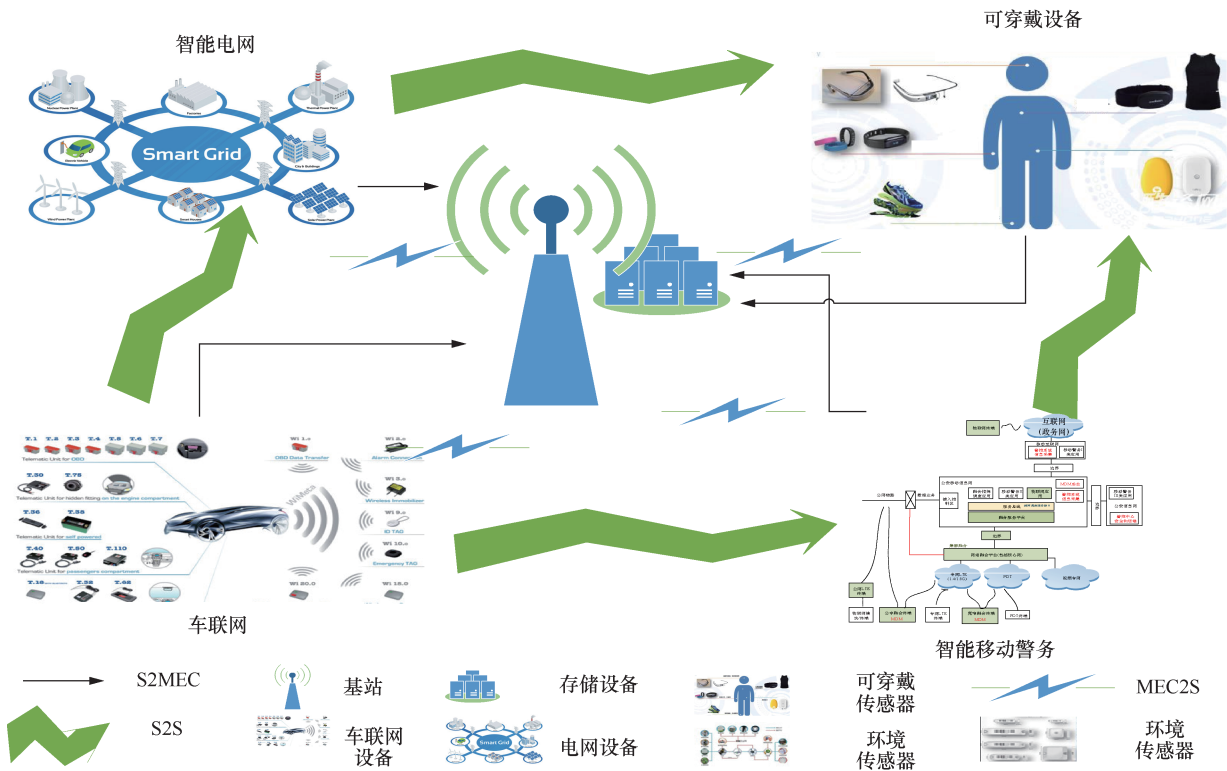


图1 典型的物联网系统模型

$$P_k(t_i) = \frac{(\lambda t_i)^k}{k!} e^{-\lambda t_i} \quad (1)$$

其中,  $P_k(t_i)$  表示传感器节点在  $t_i$  时刻产生的任务量为  $k$  的概率。



图2 时隙划分

## 2.2 通信模型

本节将从任务传输时延和能量消耗两个角度分析, 给出任务卸载的通信模型。

### 2.2.1 任务传输时延模型

假设在每个时隙内的网络拓扑恒定, 则在时隙  $t$  内, 假设有  $M$  个传感器节点需要执行任务卸载操作, 则传感器节点与所关联的边缘服务器间的信号与干扰加噪声比 (SINR, signal to interference plus noise ratio) 可表示为

$$r_i(t) = \frac{p_i g_i(t)}{\sum_{m=1, m \neq i}^M p_m g_m(t) + \delta_i} \quad (2)$$

其中,  $p_i$  为传感器节点  $i$  的发射功率;  $p_m$  为传感器节点  $m$  的发射功率;  $g_i(t)$  为传感器节点  $i$  在  $t$  时刻的功率增益;  $g_m(t)$  为传感器节点  $m$  在  $t$  时刻的功率增益;  $\delta_i$  为高斯白噪声。通过 SINR 进一步得到终端节点  $i$  与所关联的边缘服务器之间的数据传输速率为

$$v_i(t) = \omega_i \ln(1 + r_i(t)), \forall i \in a, j \in b \quad (3)$$

其中,  $v_i(t)$  为传感器节点  $i$  的传输速率,  $\omega_i$  为传感器节点  $i$  的传输带宽。

如果多个传感器节点在同一时隙内都将任务卸载到边缘服务器端, 会增加边缘服务器的计算负荷, 从而增加任务的处理时延。因此, 系统需要设计一种节点间的任务协同迁移策略, 从而决定在当前时隙下哪些节点发送数据、哪些节点保持静止状态, 以满足不同服务节点的时延应用需求。定义传感器节点  $i$  的传输时延为

$$t_i^{\text{trans}} = \frac{P_k(t_i) k_{\max}}{v_i(t)} \quad (4)$$

其中,  $k_{\max}$  为传感器节点在一个时隙内产生的最大任务量。

由任务传输时延模型可知, 被选择的节点  $i$  完

成数据卸载所消耗的时间  $t_i^{\text{trans}}$  与传输信道的 SINR 有关。为了尽可能地满足系统中节点的传输时延要求，本文提出了一种任务协同迁移策略，来降低系统的传输时延。

### 2.2.2 通信模型中的能量消耗模型

在传感器协同任务调度过程中，从传感器和边缘服务器的数据发送和接收所产生的能量消耗两方面进行分析<sup>[15-16]</sup>。从边缘系统角度考虑，将节点发送、接收所消耗的能量定义为附加能量消耗 (AEC, additional energy consumption)。根据参考文献[14]的一阶无线模型可知，传感器节点接收数据的能量消耗为

$$R_i = \varepsilon_{\text{elec}} \times P_k(t_i) k_{\text{max}} \quad (5)$$

传感器节点发送数据的能量消耗为

$$T_i = R_i + \varepsilon_{\text{amp}} \times P_k(t_i) k_{\text{max}} \times T^\alpha \quad (6)$$

其中， $R_i$  为接收的功率消耗， $T_i$  为发送的功率消耗， $\varepsilon_{\text{elec}}$  为接收或者发送单位比特的电路消耗， $\varepsilon_{\text{amp}}$  为发送单位比特时放大器的增益消耗， $T$  为边缘服务器与节点之间的传输距离， $\alpha$  为路径损耗指数。

### 2.3 计算模型

在本节中，将给出任务的计算模型，包括任务处理时延模型和能量消耗模型。

#### 2.3.1 任务处理时延模型

数据业务通过信道从数据源端传输到边缘服务器端，边缘服务器负责完成数据业务的计算任务，对于传感器节点来说，如果任务卸载过多，会增加边缘服务器的计算负荷，从而增加任务处理时延<sup>[17]</sup>。本文通过一种任务协同迁移策略来减少边缘服务器的数据处理时间。假设一个边缘服务器可以并发处理的任务量为  $\mu$ ，定义服务器的计算时延模型为

$$t_i^{\text{copt}} = \frac{P_k(t_i) k_{\text{max}}}{\mu \times \text{QPS}} \quad (7)$$

其中，QPS 为服务器的单位请求计算能力。

#### 2.3.2 计算模型中的能量消耗模型

本文对密集型任务进行层次划分，将任务量划分成不同的层次等级，采用经典的模糊逻辑方法定义每个等级产生的计算消耗<sup>[13]</sup>。具体来说，将节点产生的任务量模糊化成不同的层次等级，不同层次等级对应不同的计算消耗<sup>[18-19]</sup>。在一个时隙内，边缘服务器  $j$  与所关联的传感器节点之间的能量消耗

可表示为

$$c_j^{\text{copt}} = \sum_{i=1}^N f_i \times \text{Fuzzy}(P_k(t_i) \times k_{\text{max}}) \quad (8)$$

其中， $f_i \in \{0,1\}$  表示如果传感器节点在这一时刻被选择，则  $f_i$  为 1，否则  $f_i$  为 0。

服务器在空闲状态同样会产生供电消耗，对于边缘服务器  $j$  来说，可定义空闲时的能量消耗为

$$c_j^{\text{idle}} = \rho \times c_j^{\text{copt}} \quad (9)$$

其中， $\rho$  为边缘服务器空闲时的能量消耗与工作时的能量消耗的比值。

### 2.4 任务队列模型

从传感器节点的时延容忍角度分析，除了考虑数据业务的发送时延、计算时延外，当边缘服务器的计算资源不能满足较多节点的数据业务冲击时，会产生数据排队的情况。在  $t$  时刻，假设服务器  $j$  需要计算的任务量队列为  $q_j(t)$ ，下一时刻的任务量队列为前一时刻的计算剩余量和当前时刻任务产生量的和。定义边缘服务器的队列模型为

$$q_j(t+1) = \sum_{i \in \alpha} \{q_j(t) - c_j(t) + f_i(t) e_i(t)\} \quad (10)$$

其中， $c_j(t)$  表示服务器在  $t$  时刻的计算量， $e_i(t)$  表示传感器节点在  $t$  时刻的任务产生量。

传感器节点所期望的计算时间越少，数据排队情况发生的概率越大。单位时间内的任务到达率为  $f_i e_i(t)$ ，单位时间内的计算率为  $c_j(t)$ ，根据排队论（队列）模型可以得出任务的等待时间。定义传感器节点  $i$  的平均等待时间为

$$t_i^{\text{wait}} = \frac{f_i e_i(t)}{c_j(t)(c_j(t) - f_i e_i(t))} \quad (11)$$

## 3 问题分析

对于单个传感器节点来说，数据处理的总时延由数据传输时间、数据等待时间和服务器计算时间组成，因此，任务总处理时延可表示为

$$t_{c,i} = 2 \times t_i^{\text{trans}} + t_i^{\text{copt}} + t_i^{\text{wait}}, \forall i \in \alpha \quad (12)$$

本文所提的任务协同卸载策略的目标为最小化系统能量消耗。对于边缘系统来说，假定一个边缘服务器节点服务的终端节点的数量在一定时间内是恒定的。系统能量消耗包括节点能量消耗和服务节点计算消耗两部分，系统的正常工作开销主要体现在服务器的计算能力上，并且设定边缘服务

器空闲时的能量消耗与工作时的能量消耗之比  $\rho$  为定值。故时延约束下的能量消耗模型可表示为

$$\begin{aligned} \text{P1: } \min_{f(t)} & \sum_{i=1}^N \sum_{j=1}^M f_i \times (c_j^{\text{opt}} + c_i^{\text{AEC}}) \quad (13) \\ \text{s.t. C1: } & f_i = \{0,1\} \\ \text{C2: } & t_{c,i} \leq \omega_i, \forall i \in a \\ \text{C3: } & c_j^{\text{opt}} \leq c_{j,\max}, \forall j \in b \end{aligned}$$

其中,  $c_j^{\text{opt}}$  表示边缘服务器  $j$  的计算消耗;  $c_i^{\text{AEC}}$  表示传感器节点  $i$  的能量消耗;  $c_{j,\max}$  表示服务器  $j$  的最大计算容量。C1 表示传感器节点  $i$  是否在任务协同迁移集合中; C2 表示传感器节点  $i$  的总处理时延不能超过其期望时延; C3 表示服务器  $j$  所消耗的计算资源不能超过其最大资源容量。对于此问题, 从上述约束条件下的最优化模型分析, 此模型为 0-1 混合非线性整数规划模型, 不能从有限的时间复杂度内求得模型的最优解, 故本文采用人工智能算法求得该模型的次优解, 来逼近最优解。

#### 4 基于增强学习的协同调度解决方案

在优化问题 P1 中, 数据处理过程可以采用队列模型进行分析, 并且计算迁移系统需要实时感知边缘服务器的资源使用状态。因此, 优化问题 P1 可以转化为一个马尔可夫决策过程 (MDP, Markov decision process)<sup>[14]</sup>, 组成 MDP 的四元态构成如下。

1) 系统状态空间: 在时隙  $t$  内, 联合系统状态由队列和信噪比两方面构成, 表示为  $s(t) = (q(t), r(t))$ , 其中,  $q(t)$  表示边缘服务器在当前  $t$  时刻的数据队列, 如式(10)所示;  $r(t)$  表示物联网终端设备节点与边缘服务器通信链路之间的 SINR, 如式(2)所示。

2) 系统行动空间: 在时隙  $t$  内, 物联网设备节点是否将自己的任务量迁移到边缘服务器端, 表示为  $a_i(t) = f_i(t)$ 。

3) 代价函数: 边缘服务器系统在状态  $s(t)$  下, 采取行动  $a(t)$  所得到的代价为  $c(t) = k_1 e + k_2 d + k_3 q$ , 其中,  $e$  表示系统的能量消耗, 即本文的优化目标;  $d$  表示传感器节点的时延开销;  $q$  表示在不满足原有限域的情况下给予此行为的惩罚值。  $k_1$ 、 $k_2$ 、 $k_3$  分别表示 3 个指标的权重系数, 其物理意义为在满足时延和有限域的前提下, 需要最优化系统能耗, 则要赋予  $k_1$  较高的权值。

4) 状态转移方程: 由系统状态可知, 数据队列与信道状态都具有马尔可夫性, 所以系统的状态转移方程可表示为  $P(s'|[s,a]) = P(s'(b(t), r(t)) | s(b(t), r(t)), f_i(t))$ , 因为这两种状态是相互独立的, 所以, 更新系统状态转移方程为

$$P(b(t+1) | b(t), f_i(t)) * P(r(t+1) | r(t)) \quad (14)$$

通过对 MDP 四元态进行分析, 边缘系统的长期折扣累积代价函数可以表示如下

$$\text{P2: } v(s) = \frac{1}{T} E \left\{ \sum_{t=0}^T \gamma^t c(s(t), a(t) | s(0) = 0) \right\} \quad (15)$$

其中,  $v(s)$  表示系统代价函数, 即值函数;  $\gamma$  为折扣因子。如式(13)所示, 将原有的系统能量消耗最优转化为系统代价函数最小, 其中, 最优值函数可以表示为

$$v^*(s) = \min_{a \in A} \{c(s, a) + \gamma \sum_{s' \in S} P(s' | (s, a)) v^*(s')\} \quad (16)$$

通过定义上述值函数可以得到系统的最优迁移策略, 即值函数取最小值时的行为可表示为

$$\pi^*(s) = \arg \min_{a \in A} v^*(s) \quad (17)$$

其中,  $\pi^*(s)$  为值函数达到最小值时对应的迁移策略。

由于信道状态的随机动态性, 边缘系统无法获得关于信道状态的完全先验信息, 因此, 在使用 MDP 方法将模型转化为值函数后, 无法用传统的最优化算法通过直接求解上述值函数获得最优解。为了应对值函数求解的挑战, 本文提出了一种基于深度增强学习的计算迁移策略, 保证在原有可行域的前提下获得原问题的可行解。

由于无法获得先验信息, 所以在现有的增强学习算法中, 本文选择基于无模型的深度  $Q$  网络 (DQN, deep  $Q$ -network)<sup>[20-21]</sup> 算法。利用智能体不断地与外界环境进行信息交互, 通过学习  $Q(s, a)$  值去逼近真实值函数  $v(s)$ , 以获得相对最优解。

$Q(s, a)$  表示为

$$Q(s, a) = c(s, a) + \gamma \min_{a' \in A} v(s') \quad (18)$$

$Q(s, a)$  的更新过程表示为

$$Q(s, a) = (1 - \eta) Q(s, a) + \eta (c(s, a) + \gamma \min_{a' \in A} Q(s, a)) \quad (19)$$

其中,  $\eta$  为学习率。通过更新过程可以分析出智能体在学习过程中只需要基于当前状态学习就可以获得近似最优迁移策略。

为了查询任意给定状态下的  $Q$  值，系统需要建立一个 state-action 的二维数组，即用  $Q$ -table 来存储每次更新后的  $Q(s,a)$  值。但是该方法在矩阵维数很大的情况下，计算时间会明显上升，给计算时间复杂度带来了挑战，并且随着 state 和 action 元素的增加，系统可能会陷入维度灾难。为了解决上述问题，本文基于深度学习学习一个带参数的  $Q$  估计和  $Q$  现实，即  $Q_\theta$  和  $Q_\theta$ ，作为上述更新公式  $Q$  和  $\min_{a \in A} Q$  的近似值。基于 DQN 的计算迁移系统架构如图 3 所示。

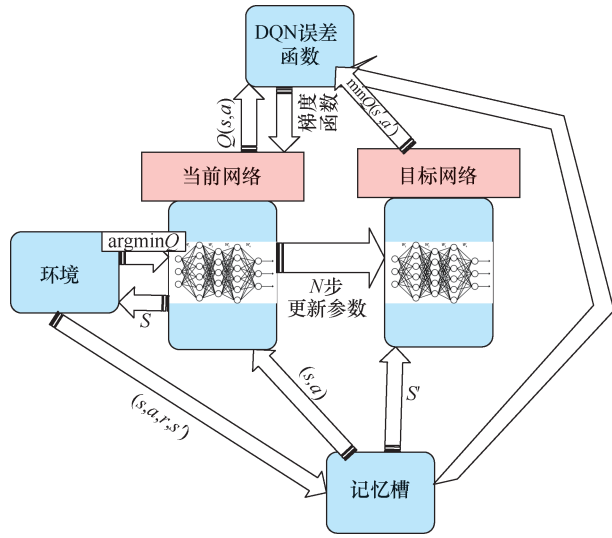


图 3 基于 DQN 的计算迁移系统架构

主网络和目标网络分别为更新参数  $\theta$  和  $\theta'$  的深度神经网络，而且这两个神经网络架构需要保持一致，只需要不同的参数。记忆槽用于存储历史数据，为主网络的参数  $\theta$  更新过程提供丰富的训练样本。基于 DQN 的计算迁移算法流程如下。

- 1) 输入：带权重的行为变量函数  $Q(\theta)$ ，容量  $N$ ，折扣因子  $\gamma$ ，学习速率  $\eta$ ，训练次数 step
- 2) 输出： $Q$  值
- 3) for episode=1,2,...,M do
- 4) 初始化状态  $s$  和 queue
- 5) for  $t=1,2,\dots,T$  do
- 6) 随机概率选择行为  $a_t$
- 7) 选择  $\arg \max Q$  并获得奖赏
- 8) 相应队列存储在缓存  $D$  中
- 9) 更新： $Q(s,a) \leftarrow R(s,a) + \gamma \max_a Q(s,a)$
- 10) 每  $N$  步重置  $Q$  估计  $\leftarrow Q$  现实
- 11) end for
- 12) end for

## 5 仿真结果分析

本节对上文提出的任务协同卸载策略进行了仿真实验验证。基于 Python+TensorFlow 的开发环境对 DQN 进行了搭建，如上所述，环境的搭建主要体现在行为集、状态集以及对应的代价函数 3 个方面。本文对仿真进行初始化，将随机选择有任务卸载和需求的 10 个终端节点，构建成具有节点行为特征的元胞矩阵。节点的决策时间间隔为 150 ms；节点在一个时隙内的传输功率为 10 mW，高斯白噪声  $\delta^2 = -173$  dBm，任务量服从一定的泊松分布，迁移系统的频谱带宽为 1 kHz；边缘服务器并发计算任务数量  $\mu=3$ ， $\rho=0.1$ ，路径损耗指数  $\alpha$  设为 2~6，传感器节点的时延容忍服从一定的泊松分布；学习率  $\eta$  为  $10^{-5} \sim 10^{-2}$ ，折扣因子  $\gamma$  定义为 0.6~0.9，记忆缓存  $N=500$ ，迭代训练次数设定为  $10^5$ 。最后将 DQN 选择迁移策略与随机选择迁移策略 (RCM) 在性能和收敛性方面进行了对比。

不同折扣因子下的  $Q$  值如图 4 所示，从图 4 可以看出，在不同的折扣因子下，系统能耗都可以在一定区间内收敛。说明了本文提出的基于 DQN 的任务卸载策略具有良好的收敛性能。另外，由于惩罚因子、学习速率和当前状态的权重不同，DQN 算法呈现出在一定区间内剧烈抖动的特性。在学习速率相同的情况下，折扣因子  $\gamma$  越小，意味着代价函数所占比重越大、在收敛区间内波动越剧烈，仿真结果显示，折扣因子  $\gamma$  在 0.8 时收敛最好。

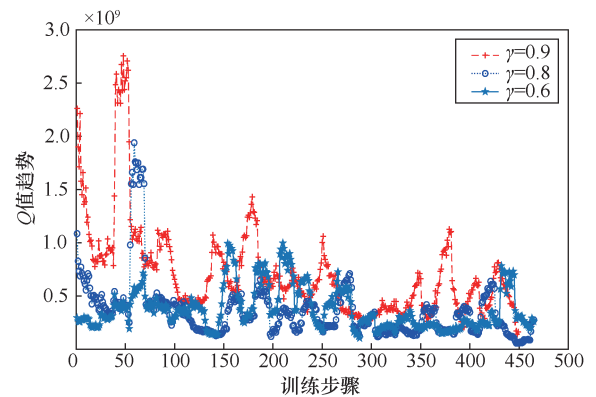


图 4 不同折扣因子下的  $Q$  值

系统代价的变化如图 5 所示，显示出了不同速率下的系统代价变化趋势。从图 5 中可以看出，随着学习迭代次数的增加，向赋予惩罚值小的方向逼近。从代价函数和式(18)的更新过程可以得出，赋

予代价函数的权值相同时,学习速率更快,对应的代价惩罚也更高,所以,在迭代过程中的收敛速度会相对较慢。

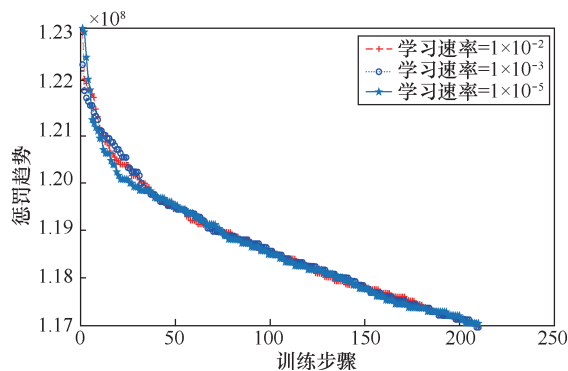


图 5 系统代价的变化

不同任务卸载策略下的能量消耗变化如图 6 所示。在图 6(b)中,学习速率和  $Q$  值有直接关系,系统能量消耗对应的权重越大,在一定区间内的数值变化起伏越平稳,从图 6 中可以看出,在迭代次数为 370 次左右的情况下,选择不同的学习因子  $\eta$  时,都能有较好的收敛效果,但是在前期的收敛速度有所区别,由于学习速率一般取值都相对较小,所以,在式(18)的  $Q(s,a)$  更新过程中,惩罚项的因子权重基本不会发生太大变化;在图 6(a)中,曲线表示 RCM 在不给长期累积函数并且无奖励惩罚的情况下,随着迭代次数的增加,系统能量消耗的变化。与 DQN 策略相比,在收敛性上,RCM 由于迁移具有高度随机性,所以在有限的迭代次数内不会有明显的收敛;在性能上,RCM 策略随着迭代次数的增加,能量消耗的最小值比 DQN 策略高。总之,无论是从性能、还是时间复杂度方面考虑,DQN 策略都比 RCM 策略更优,证明了 DQN 算法的有效性。

从上述分析可以看出,由于无线传感网络本身就是一个长期累积优化系统,所以,DQN 算法更合适。在 DQN 中,不同的折扣因子  $\gamma$  和不同的学习速率  $\eta$  都会直接或间接影响边缘系统能量消耗的收敛速度,但是从整体情况来看,在迭代次数可以接受的情况下,可以很好地得到该值函数的相对次优解,从而逼近原函数的最优解。

## 6 结束语

为了满足物联网应用多样化的性能要求,本文通过融合移动边缘技术和增强学习,在没有信道

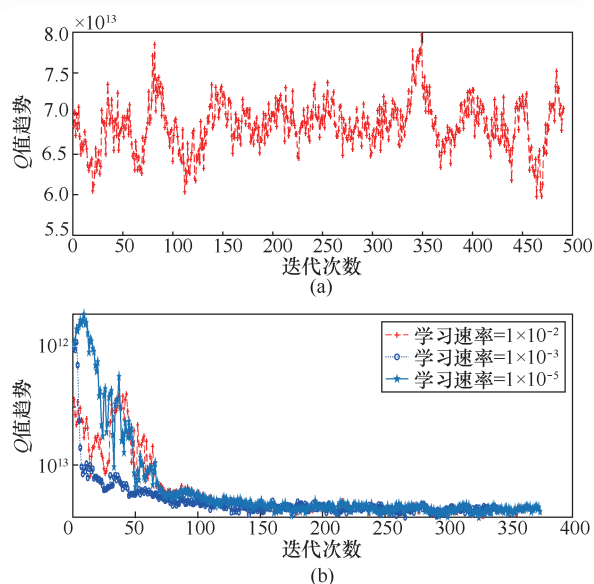


图 6 不同任务卸载策略下的能量消耗变化

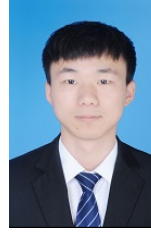
状态先验信息的情况下,优化了设备传输选择决策。与设备的随机传输选择策略相比,本文提出的基于深度增强学习的任务协同迁移策略可以显著地降低系统的能量消耗。此外,通过分析不同折扣因子和学习速率对能量消耗的影响,验证了本文所提算法具有良好的收敛性能。

## 参考文献:

- [1] GU Y, ZHENG C, MIAO P, et al. Joint radio and computational resource allocation in IoT fog computing[J]. IEEE Transactions on Vehicular Technology, 2018(99): 1.
- [2] SINGH A, VINIOTIS Y. An SLA-based resource allocation for IoT applications in cloud environments[C]//Cloudification of the Internet of Things. 2017.
- [3] ABBAS N, ZHANG Y, TAHERKORDI A, et al. Mobile edge computing: a survey[J]. IEEE Internet of Things Journal, 2017, (99):1.
- [4] BELLAVISTA P, CHESSA S, FOSCHINI L, et al. Human-enabled edge computing: exploiting the crowd as a dynamic extension of mobile edge computing[J]. IEEE Communications Magazine, 2018, 56(1):145-155.
- [5] ZHAO T, ZHOU S, GUO X, et al. Tasks scheduling and resource allocation in heterogeneous cloud for delay-bounded mobile edge computing[C]//IEEE International Conference on Communications. IEEE, 2017.
- [6] MAO Y Y, ZHANG J, SONG S H, et al. Power-delay tradeoff in multi-user mobile-edge computing systems[C]//2016 IEEE Global Communications Conference. IEEE, 2016.
- [7] YU Y H, ZHANG J, LETAIEF K B. Joint subcarrier and CPU time allocation for mobile edge computing[C]//2016 IEEE Global Communications Conference(GLOBECOM). IEEE, 2016:1-6.
- [8] WANG C, YU F R, CHEN Q, et al. Joint computation and radio resource management for cellular networks with mobile edge

- com-puting[C]//IEEE International Conference on Communications. IEEE, 2017.
- [9] WANG Y, MIN S, WANG X, et al. Mobile-edge computing: partial computation offloading using dynamic voltage scaling[J]. IEEE Transactions on Communications, 2016, 64(10): 4268-4282.
- [10] XU J, REN S. Online learning for offloading and autoscaling in renewable-powered mobile edge computing[C]//2016 IEEE Global Communications Conference (GLOBECOM). IEEE, 2017.
- [11] WANG C, LIANG C, YU F R, et al. Computation offloading and resource allocation in wireless cellular networks with mobile edge computing[J]. IEEE Transactions on Wireless Communications, 2017, 16(8): 4924-4938.
- [12] CHATTERJEE P, GHOSH S C, DAS N. Load balanced coverage with graded node deployment in wireless sensor networks[J]. IEEE Transactions on Multi-Scale Computing Systems, 2017, (99): 1.
- [13] MENDEL J M. Fuzzy logic systems for engineering: a tutorial[J]. Proc of the IEEE, 1995, 83(3):345-377.
- [14] LEVIN E, PIERACCINI R, ECKERT W. Using Markov decision process for learning dialogue strategies[C]//IEEE International Conference on Acoustics. IEEE, 1998.
- [15] JEONG S, SIMEONE O, KANG J. Mobile edge computing via a UAV-mounted cloudlet: optimization of bit allocation and path planning[J]. IEEE Transactions on Vehicular Technology, 2018, 67(3): 2049-2063.
- [16] XU C, PU L, LIN G, et al. Exploiting massive D2D collaboration for energy-efficient mobile edge computing[J]. IEEE Wireless Communications, 2017, 24(4): 64-71.
- [17] HE L, OTA K, DONG M. Learning IoT in edge: deep learning for the Internet of things with edge computing[J]. IEEE Network, 2018, 32(1): 96-101.
- [18] LIN C T. A neural fuzzy control system with structure and parameter learning[J]. Part I Fuzzy Sets & Systems, 1995, 70(2): 183-212.
- [19] CAO S G, REES N W, FENG G. Analysis and design of uncertain fuzzy control systems. Part I fuzzy modelling and identification[C]//IEEE International Conference on Fuzzy Systems. 1996.
- [20] RIEDMILLER M. Neural fitted  $Q$  iteration-first experiences with a data efficient neural reinforcement learning method[C]//European Conference on Machine Learning. 2005.
- [21] BOTVINICK M M, NIV Y, BARTO A C. Hierarchically organized behavior and its neural foundations: a reinforcement learning perspective[J]. Cognition, 2009, 113(3): 262-280.

## [作者简介]



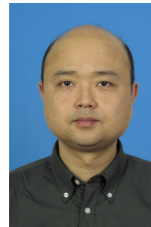
周龙雨（1995-），男，河北衡水人，电子科技大学硕士生，主要研究方向为物联网、移动边缘计算和存储以及人工智能技术。



杨宁（1974-），女，重庆人，电子科技大学副教授，主要研究方向为无线自组织网、物联网/无线传感器网络、下一代无线移动网络的网络协议设计、协议栈研发、网络系统集成与实现等。



乔冠华（1987-），男，山西长治人，电子科技大学博士生，主要研究方向为下一代无线网络资源分配、移动边缘计算和存储技术。



张科（1978-），男，重庆人，博士，电子科技大学讲师，主要研究方向为物联网、车联边缘计算与存储。



郑其林（1996-），男，四川眉山人，电子科技大学硕士生，主要研究方向为车联网和移动边缘计算。