计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210100157

基于 NOMA-MEC 的车联网任务卸载、迁移与缓存策略

张海波　张益峰　刘开健
重庆邮电大学通信与信息工程学院　重庆 400065
(zhanghb@cqupt.edu.cn)

摘　要　在移动边缘计算(MEC)与非正交多路接入(NOMA)技术相结合的车联网系统中,针对用户处理计算密集型和时延敏感型任务时面临的高时延问题,提出了一种基于博弈论和Q学习的任务卸载、迁移与缓存优化策略。首先,对基于NOMA-MEC的车联网任务卸载时延、迁移时延与缓存时延进行建模;其次,采用合作博弈算法获得最优用户分组,以实现卸载时延优化;最后,为避免出现局部最优,通过Q学习算法优化用户分组中的迁移缓存联合时延。仿真结果表明,所提方案相比对比方案,能有效提升卸载效率并降低约22%～43%的任务时延。

关键词:车联网;移动边缘计算;非正交多路接入
中图法分类号　TN929

Task Offloading, Migration and Caching Strategy in Internet of Vehicles Based on NOMA-MEC

ZHANG Hai-bo, ZHANG Yi-feng and LIU Kai-jian
School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

Abstract　In the internet of vehicles systems that combining mobile edge computing (MEC) with non-orthogonal multiple access (NOMA) technology, to solve the high latency problem when user processes computationally intensive and latency-sensitive task, a strategy of task offloading, migration and cache optimization based on game theory and Q learning is proposed. Firstly, the model of offloading delay, migration delay and cache delay of the internet of vehicles task based on NOMA-MEC is established. Secondly, we use the cooperative game method to obtain the optimal user group to optimize the offloading delay. Finally, in order to avoid local optima, the Q learning algorithm is utilized to optimize the joint delay of the migration cache in the user group. The simulation results show that compared with other solutions, the proposed algorithm can effectively improve the offloading efficiency and reduce the task delay by about 22% to 43%.

Keywords　Vehicular network, Mobile edge computing, NOMA

1　前言

随着无线网络和车联网的快速发展,预计未来几年移动数据流量将呈现爆炸式增长。为了在各种应用中实现低延迟和高可靠性,车到万物(Vehicle to Everything,V2X)技术需要合适的协议,为我们未来的生活提供可靠的安全服务[1-2]。然而在用户密集的环境中,车联网存在严重的数据拥塞,难以为用户提供高质量的服务[3-4]。为此,在车联网中引入移动边缘计算(Mobile Edge Computing,MEC)技术,用户可以将计算密集型任务卸载到网络边缘,缓解计算压力,进而减少能耗和任务延迟。但移动边缘计算场景下依然存在多用户卸载时延较高以及迁移缓存联合时延未得到优化等问题。

为了提高多用户场景下的资源利用率,非正交多路接入(Non-Orthogonal Multiple Access,NOMA)技术[5-6]被提出并应用于车联网,有不少学者对基于NOMA技术的车联网展开了研究。在V2X系统中,文献[7-8]考虑到车辆的集中式和分布式功率分配问题,提出了基于NOMA的V2X广播系统的混合集中式/分布式方案,实现了最优功率分配。文献[9]提出了D2D(Device to Device Communication)通信增强的V2X网络架构,对NOMA-V2X系统中的资源共享方案

到稿日期:2021-01-21　返修日期:2021-05-06　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(61801065,61601071);长江学者和创新团队发展计划基金资助项目(IRT16R72);重庆市基础与前沿项目(cstc2018jcyjAX0463);重庆市留创计划创新类资助项目(cx2020059)
This work was supported by the National Natural Science Foundation of China (61801065,61601071), Program for Changjiang Scholars and Innovative Research Team in University (IRT16R72), General Project on Foundation and Cutting-Edge Research Plan of Chongqing (cstc2018jcyjAX0463), Chongqing Innovation and Entrepreneurship Project for Returned Chinese Scholars(cx2020059).
通信作者:张益峰(1582042470@qq.com)

进行了优化。文献[10]在基于 NOMA 的车联网系统中，提出了一种基于能效的动态资源分配算法，在保证网络稳定性的前提下提高了系统能效。但是，上述研究均没有考虑到，在 NOMA 系统中，如果没有最优化子载波的用户分配，容易出现多用户卸载时延较高的问题。因此，本文的研究工作针对基于 NOMA-MEC 的车联网用户分组优化展开。

在基于 NOMA-MEC 的车联网用户任务卸载过程中，当多个用户产生相同的任务卸载请求时，没有必要对其进行重复计算，为此引入了一种新的计算模式，称为计算(或任务)缓存[11]。它将用户卸载到边缘云的计算结果存储在位于网络边缘的缓存池中，从而简化卸载过程，减少卸载时延并提高资源利用率。缓存系统的主要性能参数是命中率，即用户在缓存池中找到请求文件的概率。目前，大量学者开始研究在小区基站中的数据缓存问题。文献[12]根据文件的流行程度随机缓存文件。文献[13]基于每个链路的传输效率来优化缓存命中率。文献[14]提出了一种内容中心网络中能耗优化的隐式协作缓存机制，其可以获得较优的缓存命中率及平均路由跳数。上述缓存研究工作旨在最大程度地提高命中率，但是忽略了数据缓存方面的时延成本。

资源虚拟化是车联网资源分配和管理的关键支持技术之一。资源虚拟化技术以虚拟机(Virtual Machine，VM)的形式在共享的硬件平台上同时执行各种用户的任务。较低的 VM(任务)迁移时延[15-16]可以促进负载平衡、方便故障管理和服务器维护。但是，基于路边云(Roadside Cloud，RSC)的车辆云计算(Vehicular Cloud Computing，VCC)具备云资源高度分散和用户高度动态等特点，使任务迁移时延成本较高，用户服务质量较差。而软件定义网络(Software Defined Network，SDN)有助于解决这一难题[17]。文献[18]充分利用 SDN 的灵活性，构建了拓扑自适应数据中心网(Topology-adaptive Data Center Network，DCN)。基于这种结构，可以最小化迁移成本和因多次迁移而产生的通信成本。文献[19-20]利用 SDN 制定了全局规则和特定规则，来增强 VM 的迁移效率。SDN 将数据平面与控制平面分离，使网络基础设施只处理数据转发问题，而控制逻辑被移动到中央控制器。通过集中的 SDN 控制器，可以实时监视和收集网络状态，使支持 SDN 的网络设备变得透明且可控，进而可以更加合理地分配资源，以优化缓存时延。因此，SDN 架构有助于优化任务迁移缓存联合时延。

虽然关于缓存与任务迁移的研究工作较多，但是共同考虑任务卸载、迁移与缓存的工作较少。由于车辆的移动性以及车联网中可下载内容的不断增长，会出现多个用户同时进行任务卸载、迁移与缓存的场景，造成数据拥堵，进而导致时延过高。因此，本文的主要工作为优化此场景下的任务时延，具体如下：

(1)构建一个多小区多用户的软件定义车联网网络场景，并引入 NOMA 系统。利用 SDN 控制器可以集中调度车辆与基站信息，其中，基站配置 MEC 服务器和缓存容器。

(2)研究用户在车辆移动期间的任务协作卸载时延以及任务迁移缓存联合时延优化的问题，将任务时延建模为一个混合整数非线性问题(Mixed Integer Nonlinear Problem，MINLP)。

(3)为了解决上述问题，提出了一种两阶段算法，将任务时延优化问题分为两个步骤来解决。首先，通过合作博弈建模，对车辆用户进行分组处理，提高了卸载效率，优化了任务卸载时延。然后，对用户分组后上传至 MEC 服务器的任务进行空间状态集的建模，并使用 Q 学习算法优化其迁移缓存联合时延。

2 系统模型

图 1 给出了基于 MEC 的软件定义车联网系统框架。其中，MEC 服务器、基站、车辆和缓存容器之间的数据传输代表数据平面，车辆可以通过蜂窝网络接口定期向 SDN 控制器更新其状态，包括位置和运动信息等内容。SDN 控制器代表控制平面，基于给定的服务，其可以执行带宽分配、路由协议等决策。基于接收到的控制信息，如 MEC 服务器的设备将进行对应的操作。SDN 的应用平面由车辆的服务请求构成。

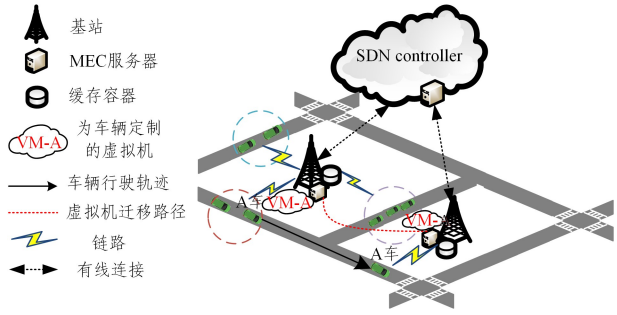

图 1 系统模型图

Fig. 1 System model diagram

2.1 网络模型

网络模型有 M 个基站，每个基站部署一个 MEC 服务器，MEC 服务器可以提供缓存或迁移服务。假设附近有 A 个服从泊松分布的车辆用户，表示为 $V=\{v_1,v_2,\cdots,v_A\}$，且 $A\geqslant M$。车辆的任务模型为 $\Gamma_i(D_i,C_i)$。其中，D_i 表示计算任务 Γ_i 的输入数据大小，C_i 表示完成任务 Γ_i 所需的 CPU 周期数。

2.2 信道模型

在基于 NOMA 技术的车联网中，多个车辆用户可以共享同一个子载波。MEC 服务器接收到的用户信号不仅包含该用户的信号，还包含使用同一子载波的其他用户的干扰信号。假设有 N 个子载波，并把车辆用户分成 N 组，每组用户共享同一子载波。车辆用户分组集为 $S=\{S_1,S_2,\cdots,S_n,\cdots,S_N\}$，其中，$S_n$ 表示使用第 n 个子载波的车辆用户分组，该分组中车辆用户数为 A_n，并且满足 $\sum_{n=1}^{N}A_n=A$。

信道增益为 $h_n=[h_{1,n},h_{2,n},\cdots,h_{A_n,n}]$，其中，$h_{a,n}$ 表示

车辆用户 $a(a\in S_n)$ 在子载波 n 上的信道增益。

2.3 传输模型

假设基站具有 N 个子载波的所有全信道状态信息(Channel State Information,CSI)。在用户分组 S_n 中,根据信道增益对用户进行排序,即 $|h_{1,n}|^2\leqslant\cdots\leqslant|h_{A_n,n}|^2$。

服务器应用连续干扰消除(Successive Interference Cancellation,SIC)来解码每个用户的信号。在第一阶段,解码信道增益最高的用户数据,将所有来自其他用户的信号视为干扰。接收机对该用户的数据解码后,就可以重构出其信号,并从接收到的总信号中减去该信号。下一阶段解码增益次高的用户数据,并重复此步骤直到最后一个用户,即最后一个用户只有高斯噪声干扰。因此,当用户 a 将信号上传给服务器后,用户的信干噪比(SINR)描述为:

$$SINR_a=\frac{p_a|h_{a,n}|^2d_a^{-\alpha}}{\sum\limits_{i=1}^{i\neq a}p_i|h_{i,n}|^2d_i^{-\alpha}+\sigma^2} \tag{1}$$

其中,p_a 为用户 a 的发射功率,p_i 为其他用户的发射功率;σ^2 是汽车终端的高斯白噪声;$d_a^{-\alpha}$ 表示路径损耗,其中 d_a 表示车辆用户 a 与 MEC 服务器的距离,$\alpha>2$ 为路径损耗指数。

用户 a 的上行传输速率为 $R_a=B\log_2(1+SINR_a)$,B 表示用户的信道传输带宽。因此其上行传输时间为:

$$T^L=\frac{D_i}{R_a} \tag{2}$$

2.4 计算模型

车辆本地计算任务模型为 Γ_i,车辆本身的计算能力为 F_i,且 $F_i>0$。则本地计算的时延为:

$$t^l=\frac{C_i}{F_i} \tag{3}$$

MEC 服务器计算任务 Γ_i 时,MEC 服务器分配给该任务的计算资源为 F_M。则服务器的计算时延为:

$$t^M=\frac{C_i}{F_M} \tag{4}$$

2.5 迁移模型

当车辆将任务上传给路边单元的服务器时,服务器会划分出一个虚拟机为其服务,帮助其进行任务计算。在服务期间,随着车辆的行驶,为了保证服务的连续性,服务器可以跨路边单元传输定制的虚拟机,此过程称为虚拟机迁移。由于内存预复制迁移是最常见、使用最广泛的方法,因此本文主要采用基于预复制迁移的虚拟机迁移模型[21]。

虚拟机迁移时间分为两个部分。第一部分为内存迁移时间:

$$T_{\text{mig}}=\frac{D_i}{L}\times\frac{1-\lambda^{E+1}}{1-\lambda} \tag{5}$$

其中,$\lambda=\frac{R}{L}$,R 为迁移期间的脏页率;L 为虚拟机的数据中心的网络带宽;E 为数据拷贝的迭代次数。

第二部分为迁移过程中的停机拷贝时间:

$$T_{\text{down}}=\frac{D_i}{L}\cdot\lambda^E+t_{\text{res}} \tag{6}$$

其中,t_{res} 为虚拟机在目的物理机的重启时间。

2.6 缓存模型

当车辆用户向 MEC 服务器请求任务内容时,若请求的内容 MEC 服务器已缓存,则可以减少回程延迟和回程带宽。基于文献[22]可知,任务请求内容的流行程度满足 Zipf 分布。对于任务 Γ_i,其缓存命中率为:

$$P(\Gamma_i)=\frac{\Gamma_i^{-\zeta}}{\sum\limits_{\Gamma_{i=1}}^{\Gamma_N}\Gamma_i^{-\zeta}} \tag{7}$$

其中,Γ_N 表示内容总数;ζ 为 Zipf 分布参数,一般设置为 0.56[23]。

用户请求的内容缓存的回报(减轻的回程带宽)可以表示为:

$$R_C=\bar{\omega}P(\Gamma_i) \tag{8}$$

其中,$\bar{\omega}$ 为用户平均数据传输速率。

当任务内容的大小为 D_i 时,其缓存回报时延为:

$$t^b=\frac{D_i}{R_C} \tag{9}$$

2.7 问题形成

本文的目的是优化用户的任务卸载时延、任务迁移时延与任务缓存时延。当多个用户同时进行任务卸载、迁移与缓存时,用户首先会检查自身能力是否可支持计算,若不能,则会通过 NOMA 系统将任务上传给附近的 MEC 服务器。然后 MEC 服务器将会检查缓存池,若任务结果存在,则 MEC 服务器会将任务结果回传给用户。这一过程中,由于车辆用户的移动性,因此 MEC 服务器同时也会进行任务迁移,根据 SDN 控制器提供的车辆位置信息,将任务迁移到车辆目的地附近的服务器进行任务回传。整个任务时延包括任务卸载时延、任务缓存时延与任务迁移时延,具体如下:

$$\min U(p_a,R_A,\lambda,N,\bar{\omega})=\sum_{S=\{S_1,S_2,\cdots,S_N\}}\sum_{i=1}^{i=A}\sum_{i=1}^{i=M}(1-\ell_M)\left[s_n\frac{D_i}{B\log_2\left(1+\frac{p_a|h_{a,n}|^2d_a^{-\alpha}}{\sum\limits_{i=1}^{i\neq a}p_i|h_{i,n}|^2d_i^{-\alpha}+\sigma^2}\right)}+\right.$$
$$\left.(1-\eta)\frac{C_i}{F_M}+\eta\left[\left(\frac{D_i}{L}\times\frac{1-\lambda^{E+1}}{1-\lambda}+\frac{D_i}{L}\cdot\lambda^E+t_{\text{res}}\right)+\frac{D_i}{\bar{\omega}\frac{\Gamma_i^{-\zeta}}{\sum\limits_{\Gamma_{i=1}}^{\Gamma_N}\Gamma_i^{-\zeta}}}\right]\right]+\ell_M\frac{C_i}{F_i}$$

$C1:s_n\in\{0,1\},\sum\limits_{n=1}^{N}s_n=1$

$C2:\ell_M\in\{0,1\}$

$C3:\eta\in\{0,1\}$

$C4:0<F_M<F_B$

$C5:0\leqslant p_a\leqslant P_{\text{MAX}},\forall a\in A$

$C6:D_i<A_{\text{LINK}}$

$C7:0<\lambda<1$

$C8:0<E<E_{\text{MAX}}$

$$C9:\sum_{V=1}^{A}\eta\Gamma_i\leqslant C_{\mathrm{MEC}} \tag{10}$$

C1 表明 s_n 是二进制变量，每个用户只能选择一个子载波。C2 和 C3 表示卸载决策与迁移缓存决策为 0-1 决策。C4 表示 MEC 服务器分配给该任务的计算资源小于 MEC 服务器总的计算资源 F_B。C5 表明发射功率 p_a 在允许的最大范围内且为非负。C6 表明在虚拟机迁移时传输的数据量小于链路的容量 A_{LINK}。C7 和 C8 表明虚拟机迁移时间模型参数 λ 和迭代次数 E 的范围。C9 表明缓存内容总和不宜超过服务器的存储范围。

3 基于 NOMA-MEC 的软件定义车联网任务卸载、迁移与缓存策略

本文工作主要针对任务时延的优化展开。任务时延主要分为任务卸载时延与任务迁移缓存联合时延两个部分，因此将优化算法分为两个阶段。第一阶段为基于 NOMA 的合作博弈卸载决策算法，其通过子载波之间的协作来提供用户分组方案，从而优化卸载时延。第二阶段，在用户分组后，当任务传至服务器时，使用 Q 学习算法优化任务迁移缓存联合时延。

3.1 基于 NOMA 的合作博弈卸载决策算法

为了将多个用户进行高效分组，即为每个子载波合理地分配用户，使 NOMA 系统的整体效率最高，实现多用户卸载时延最小，本文采用合作博弈算法，建立博弈论模型(V,S,ψ)。V 是车辆用户集合；$S=\{S_1,S_2,\cdots,S_n,\cdots,S_N\}$为每个子载波的用户分组集合；$\psi$ 为效用函数，是用户 a 在分组 S_n 的回报价值函数，与卸载时延成反比，其效益越大，卸载时延就越小。合作博弈可实现效用函数的最大化，进而优化卸载时延。假设有 N 个子载波，子载波集合为 Y。分组集合 S 满足 $n,\vartheta\in Y,S_n\cap S_\vartheta=\varnothing,\forall n\neq\vartheta$。

效用函数的公式为：

$$\psi(S_n)=\sum_{a=0}^{A_n}\frac{1}{T^L+t^M},\forall a\in S_n \tag{11}$$

为了解释说明合作博弈过程，对其进行相关定义与证明。

定义 1(拆分与合并) 在两个分组 S_ϑ 和 S_n 中，如果用户 a 离开 S_n 加入 S_ϑ，其中 $\vartheta\neq n,\vartheta,n\in Y$，则有：

$$\{S_\vartheta,S_n\}\rightarrow\{S_\vartheta\cup\{a\},S_n\backslash\{a\}\} \tag{12}$$

定义 2 $\succ_a$ 表示车辆用户 a 对分组 S_ϑ 和 S_n 的偏移度。用户 a 倾向于某个分组取决于其效用的提升程度。

$$S_\vartheta\succ_a S_n\Leftrightarrow\psi(S_n\backslash\{a\})+\psi(S_\vartheta\cup\{a\})>\psi(S_n)+\psi(S_\vartheta) \tag{13}$$

定义 3(交换) 当定义 2 的关系式成立时，会产生交换行为，并相应地更新分组。

$$\{S_\vartheta,S_n\}\rightarrow\{S_\vartheta\cup\{a'\}\backslash\{a\},S_n\{a\}\cup\{a\}\backslash\{a'\}\} \tag{14}$$

本节基于 NOMA 的合作博弈卸载决策算法中，一开始可用的子载波会随机分配给车辆用户。在每次迭代中，再次随机选择子载波，并且与上次子载波不同，随之产生不同的分组，相继进行上述定义中的某些分组操作，直到出现最佳的分组，实现多用户卸载时延的优化。

引理 1 从随机分组开始，算法 1 的合作博弈保证收敛于最终分组状态。

算法 1 合作博弈卸载决策算法

1. 用户初始化，采用随机分组 S_{START}
2. 定义当前分组 S_{NOW}
3. 重复迭代
4. 从分组 $S_n\in S_{\mathrm{NOW}}$ 随机挑选用户 a
5. 从分组 $S_\vartheta\in S_{\mathrm{NOW}}$ 随机挑选用户 a′
6. if 假设交换用户形成分组 S_{new}
7. if $S_{\mathrm{new}}\succ_{a'}S_{\mathrm{NOW}}$ then
8. 用户 a 离开 $S_n\in S_{\mathrm{NOW}}$ 加入 $S_\vartheta\in S_{\mathrm{NOW}}$
9. 用户 a′离开 $S_\vartheta\in S_{\mathrm{NOW}}$ 加入 $S_n\in S_{\mathrm{NOW}}$
10. 更新 S_{NOW}
11. $S_{\mathrm{NOW}}\leftarrow\{S_{\mathrm{NOW}}\backslash\{S_n,S_\vartheta\}\}\cup\{S_\vartheta\cup\{a\}\backslash\{a'\},S_n\cup\{a'\}\backslash\{a\}\}$
12. else 假设用户 a′加入 $S_n\in S_{\mathrm{NOW}}$ 形成 S_{new}
13. if $S_{\mathrm{new}}\succ_{a'}S_{\mathrm{NOW}}$ then
14. 用户 a′离开 $S_\vartheta\in S_{\mathrm{NOW}}$ 加入 $S_n\in S_{\mathrm{NOW}}$
15. 更新分组 S_{NOW}
16. $S_{\mathrm{NOW}}\leftarrow\{S_{\mathrm{NOW}}\backslash\{S_n,S_\vartheta\}\}\cup\{S_\vartheta\backslash\{a'\},S_n\cup\{a'\}\}$
17. Until

证明：为了达到效用函数 ψ 的最优值，车辆用户会不断地进行交换、拆分与合并操作，进而不断地改变分组。在第 n 次和第 $n+1$ 次迭代中，分组从 S_n 变为 S_{n+1}。这种操作当且仅当博弈效用 ψ 严格增加时发生，即：

$$S_n\rightarrow S_{n+1}\Leftrightarrow\psi(S_n)<\psi(S_{n+1}) \tag{15}$$

因此博弈效用值 ψ 总是递增的，即：

$$S_{\mathrm{START}}\rightarrow S_1\rightarrow S_2\rightarrow\cdots\rightarrow S_{\mathrm{END}} \tag{16}$$

其中，S_{START} 和 S_{END} 表示初始分组集和最终分组集。A 个用户在每个新分组集上的时延均有所降低。这是因为车辆用户是有限的，所以分组集也是有限的且基于贝尔数[24]。因此，上述递增顺序会收敛于最终态 S_{END}。

引理 2 最终态 S_{END} 是趋于稳定的。

证明：假设最终态 S_{END} 不是趋于稳定的。那么，必定存在用户 a 会趋向于离开当前所在分组而加入新的分组，形成新的分组集 S_{current}，即 $S_{\mathrm{current}}\succ_a S_{\mathrm{END}}$。这与 S_{END} 是最终态相矛盾。因此最终态 S_{END} 是趋于稳定的。

3.2 基于 Q 学习的任务迁移与缓存优化算法

通过上一节的合作博弈，得到最佳车辆用户分组，在此基础上，用户会将任务上传给服务器，本节采用 Q 学习算法来优化服务器上的任务迁移缓存联合时延。Q 学习 (QL)[25] 是一种增强学习方法，首先代理通过与环境的交互获得环境的信息，然后将所采取的行为得到的奖励存储在 Q 表中。通过更新 Q 表，代理能够采取一系列行动来最大化累积奖励，并将奖励设置为任务迁移缓存联合时延的倒数，通过奖励最大化得出最小时延。任务迁移缓存联合时延函数为：

$$\chi(\lambda,E,\bar{\omega})=\sum_{S=\{S_1,S_2,\cdots,S_N\}}\sum_{i=1}^{i=A}\sum_{i=1}^{i=M}(T_{\mathrm{mig}}+T_{\mathrm{down}})+t^b \tag{17}$$

引理 3 $\chi(\lambda,E,\bar{\omega})=\dfrac{1-\lambda^{E+1}}{1-\lambda}+\lambda^E+\dfrac{1}{\bar{\omega}}$ 为非凸优化问题。

证明：

$$\frac{\partial^2\chi}{\partial\lambda^2}=\frac{2\lambda^{(E-1)E(E+1)}-\lambda^{E(E-2)(E-1)}}{\lambda-1}-\frac{2(\lambda^E-2\lambda^{E+1}+1)}{(\lambda-1)^3}+\frac{2\lambda^{E(E-1)}-2\lambda^{E(E+1)}}{(\lambda-1)^2} \tag{18}$$

$$\frac{\partial^2\chi}{\partial\lambda\partial E}=2\lambda^E-\lambda^{E-1}-\lambda^{(E-1)E\ln(\lambda)}+\frac{2\lambda^{E(E+1)\ln(\lambda)}}{\lambda-1}-\frac{2\lambda^{(E+1)\ln(\lambda)}-\lambda^{E\ln(\lambda)}}{(\lambda-1)^2} \tag{19}$$

$$\frac{\partial^2\chi}{\partial E\partial\lambda}=2\lambda^E-\lambda^{E-1}-\lambda^{(E-1)E\ln(\lambda)}+\frac{2\lambda^{E(E+1)\ln(\lambda)}}{\lambda-1}-\frac{2\lambda^{(E+1)\ln(\lambda)}-\lambda^{E\ln(\lambda)}}{(\lambda-1)^2} \tag{20}$$

$$\frac{\partial^2\chi}{\partial E^2}=-\frac{\lambda^{E\ln\lambda^2}-2\lambda^{(E+1)\ln\lambda^2}}{\lambda-1} \tag{21}$$

$$\frac{\partial^2\chi}{\partial\omega^2}=\frac{2}{\omega^3} \tag{22}$$

其 Hessian 矩阵为：

$$\boldsymbol{H}=\begin{pmatrix}\frac{\partial^2 f}{\partial\lambda^2} & \frac{\partial^2 f}{\partial\lambda\partial E} & 0\\ \frac{\partial^2 f}{\partial E\partial\lambda} & \frac{\partial^2 f}{\partial E^2} & 0\\ 0 & 0 & \frac{\partial^2\chi}{\partial\omega^2}\end{pmatrix} \tag{23}$$

在 $0<\lambda<1$，$0<\omega<\omega_{\max}$，$0<E<E_{MAX}$ 且 E 为整数的条件下，$\frac{\partial^2 f}{\partial E^2}<0$，$\frac{\partial^2 f}{\partial\lambda^2}>0$，$\frac{\partial^2 f}{\partial\lambda\partial E}=\frac{\partial^2 f}{\partial E\partial\lambda}$，$\frac{\partial^2\chi}{\partial\omega^2}>0$。得出：

$$|\boldsymbol{H}|=\frac{\partial^2 f}{\partial E^2}\times\frac{\partial^2 f}{\partial\lambda^2}\times\frac{\partial^2\chi}{\partial\omega^2}-\frac{\partial^2 f}{\partial\lambda\partial E}\times\frac{\partial^2 f}{\partial E\partial\lambda}\times\frac{\partial^2\chi}{\partial\omega^2}<0 \tag{24}$$

因此优化式(17)为非凸优化问题。采用 Q 学习算法来优化此问题，建立状态模型 $\xi=\{R,L,S_{END}\}$，动作模型 $a=\{\lambda,E,\omega\}$，奖励设为：

$$r=\frac{1}{\chi(\lambda,E,\omega)} \tag{25}$$

这样 Q 学习算法奖励最大化即为时延最小化。

Q 学习的数学模型为：

$$Q(\xi,a)\leftarrow(1-\alpha)Q(\xi,a)+\alpha[r(\xi,a)+\gamma\max Q(\xi',a')] \tag{26}$$

其中，ξ 表示当前的状态；a 表示选择的行为；α 是学习率，值在 0～1 之间，本文设置为 0.01；$r(\xi,a)$ 表示在状态 ξ 下执行动作 a 将得到的回报；γ 表示折扣因子，决定时间的远近对回报的影响程度，本文设置为 0.9。

为了提高算法的稳定性，采用状态集和行为集分别收敛的方法，MEC 服务器记录状态-动作对应的执行次数及其状态-动作值。当这个状态-动作值收敛时，把之前的值的平均值视为状态-动作值，并停止对该状态-动作值的更新，即：

$$|Q_{n+1}(\xi,a)-Q_n(\xi,a)|\leqslant\upsilon \tag{27}$$

$$\phi_n(\xi,a)=\sum_{m=1}^{n-1}Q_m(\xi,a) \tag{28}$$

$$Q(\xi,a)=\frac{\phi_n(\xi,a)}{n-1} \tag{29}$$

其中，$Q_n(\xi,a)$ 是第 n 次迭代的价值回报。$\upsilon\approx1$ 为收敛误差参数。$Q(\xi,a)$ 为收敛后固定的状态行为回报。$\phi_n(\xi,a)$ 为记录 n 次迭代的效益累计值。当某个状态-动作值与上一次值的差异足够小时，就认为该状态-动作值达到收敛状态。

由算法 1 可得车辆用户的最佳分组 S_{END}，在此基础上，利用算法 2 建立状态集 ξ，通过行为集 a 的不断更新得到奖励集 r。最后不断地迭代学习，系统会得到最佳的行为参数指标、最优的奖励，从而优化任务迁移缓存联合时延。

算法 2 基于 Q 学习的任务迁移与缓存优化算法

输入：根据合作博弈算法得到最后分组 S_{END}，更新状态集 ξ，加入行为集 a

输出：奖励 r

1. 初始化 **Q** 矩阵
2. for 每一次探索
3. 随机选择一个初始状态 ξ
4. 随机选择行为
5. 利用选择的行为进入下一个状态 ξ'
6. $\xi\leftarrow\xi'$
7. 多次探索后，**Q** 矩阵趋于收敛
8. Until 到达最终状态

3.3 时间复杂度

本文通过一个两阶段算法来优化任务时延函数。第一阶段利用基于 NOMA 的合作博弈卸载决策算法来优化卸载时延，每次迭代进行 n 次计算操作，当算法迭代 n 次时，时间复杂度为 $O(n^2)$。第二阶段通过 Q 学习算法来优化分组用户的任务迁移缓存联合时延，算法的时间复杂度为 $O(n)$。整个算法的时间复杂度为 $O(n^2+n)=O(n^2)$。

4 仿真分析

本节基于 MATLAB 平台对所提方案进行验证。文中相关参数是参考 MEC 白皮书的规定和 IEEE 802.11p 标准规定下的车联网场景建立进行设定的。相关参数如表 1 所列。

表 1 仿真参数表

Table 1 Simulation parameter

参数值	
输入数据的大小/kB	1～700
噪声功率/dBm	−100
任务所需 CPU 周期数/GHz	0.1～1
MEC 服务器 CPU 周期频率/GHz	6
车辆用户的 CPU 周期频率/GHz	0.5～1
传输带宽/MHz	50～80
最小脏页率/Mbps	50
车辆最大传输功率/dbm	20
RSU 覆盖半径/m	250
最小 λ	0.2
最大迭代次数 E_{MAX}	30
路径损耗指数 α	3
子载波个数 N	4
缓存内容总数 Γ_N	10～50
平均单用户数据传输率 $\bar{\omega}$/Mbps	50～100

图 2 是迭代次数与本文方案、全部本地计算方案和全部卸载方案的平均时延对比图。由于车辆本地计算不涉及迭代

运算，因此不受迭代次数的影响。迭代次数较少时，全部卸载到 MEC 服务器的时延最高。其原因是车辆自身有计算能力可以应对目前的计算程度。随着迭代次数的增多，车辆用户计算能力的弊端会凸显出来。因此，卸载计算可以降低任务时延。本文方案不是单一的全部卸载，通过合作博弈，车辆用户关联会发生变化，迭代次数越多，分组集越趋于最优，时延会小于单一的全部卸载。

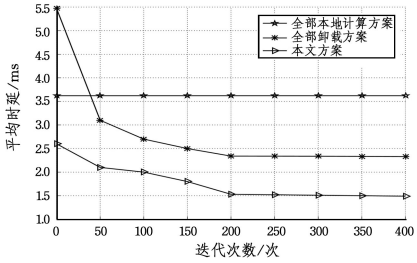

图 2　迭代次数与平均时延的对比

Fig. 2　Comparison of number of iterations and total delay

图 3 给出了基于 NOMA-MEC 系统部分卸载方案（本文方案）、基于 NOMA-MEC 系统全部卸载方案及基于 OFDM（正交频分复用）-MEC 系统部分卸载和全部卸载方案的性能对比，即平均时延与用户数量的关系。首先，可以观察到，对于所有方案，计算所涉及的网络延迟都随着用户数量的增加而增加。最重要的是，在基于 NOMA 的方案中，可以观察到，与部分卸载方案相比，将计算完全卸载到 MEC 服务器会导致更大的延迟。因此，将任务计算划分为卸载计算和本地计算有助于其更快地完成任务。

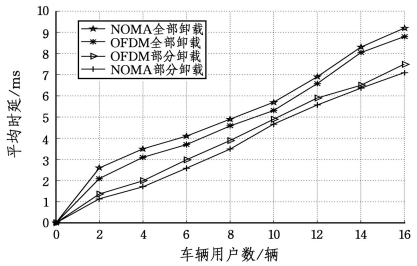

图 3　车辆用户数量与平均时延的对比

Fig. 3　Comparison of number of vehicle users and average delay

图 4 是关于传统 OMA 方案与 NOMA 方案用户容量的对比图。

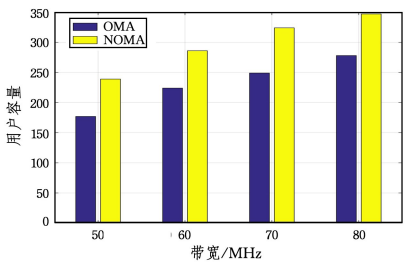

图 4　带宽与用户容量的关系

Fig. 4　Relationship between bandwidth and user capacity

在 NOMA 方案中，由于调度程序会在一个信道分配多个用户进行同时传输，因此需要多用户调度算法，本文采用合作博弈算法对用户进行分组。由图 4 可知，与 OMA 方案相比，NOMA 方案在相同带宽下用户容量有所提高，获得了大约 40％的增益。

图 5 是任务数据大小与平均时延的对比图。其中，对比方案为无迁移缓存方案、仅迁移无缓存方案以及基于贪婪算法的迁移缓存方案。

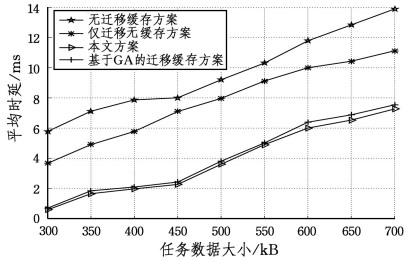

图 5　数据大小与平均时延的关系

Fig. 5　Relationship between data size and average delay

首先，从图 5 可知，随着数据大小的增加，时延会增多。无迁移缓存方案的平均时延是最大的，因为它没有合理地利用 MEC 服务器的计算资源。其次，若只迁移无缓存，会增加服务器的计算量，导致时延增多。本文方案通过 Q 学习算法来优化任务迁移缓存联合时延，通过查看 Q 表找到最佳策略，使累积奖励最大化。基于贪婪算法（GA）的方案选择当前状态下的最佳决策，而不考虑整个系统的最佳决策，容易陷入局部最优，因此其性能相比本文方案较差。

图 6 是 NOMA 模式和 OFDM 模式中，缓存内容总数与用户平均时延的关系图。从图 6 可知，随着缓存内容总数的增多，用户平均时延会减少，这是因为缓存内容的增多，避免了重复执行相同任务，从而简化了卸载过程，减少了 MEC 服务器计算资源的消耗，因此用户的时延会逐渐减少。从图 6 中还可以看出，NOMA 模式的用户平均时延小于 OFDM 模式的用户平均时延，这是因为在相同条件下，NOMA 可以承载更多用户，所以相同内容的访问概率会增加，进而减少了重复运算。

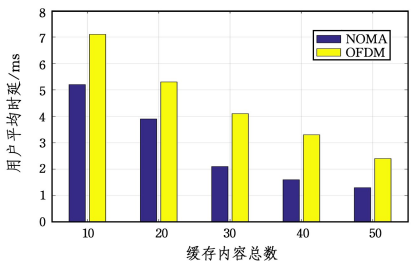

图 6　缓存内容总数与用户平均时延的关系

Fig. 6　Relationship between total number of cached content and average user delay

将本文方案与如下策略进行比较，任务缓存卸载方案（TCO）[26]、协作移动边缘计算方案（CMEC）[27]和安全任务卸载方案（STO）[28]。TCO 方案将流行任务缓存于基站，关键任务缓存于就近的 MEC 服务器，提高了缓存命中率。CMEC 方案通过 MEC 服务器协作计算来减少能量消耗。STO 方案

利用加密算法对任务进行加密解密，提高了任务的安全性。但是它们都忽略了任务时延的影响。

图 7 给出了本文方案与 TCO 方案、CMEC 方案、STO 方案在不同任务数据大小下的卸载时延对比。由图 7 可知，与其他方案相比，本文方案的时延消耗较小，这是因为本文通过合作博弈得到的最佳车辆分组集来进行任务卸载，使卸载过程更加智能高效，因此其效率高于对比方案。

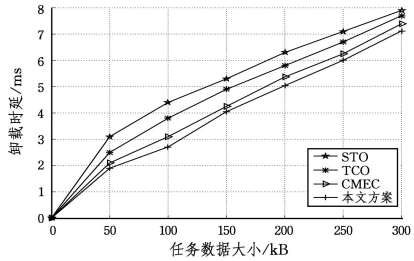

图 7 任务数据大小与任务卸载时延的对比

Fig. 7 Comparison of task data size and task migration cache delay

图 8 给出了本文方案与 TCO 方案、CMEC 方案、STO 方案在不同车辆用户数量大小下的总体时延对比。由图 8 可知，随着车辆用户的增加，本文方案与 CMEC 方案的时延低于其他方案，这是因为 CMEC 方案采用了 MEC 服务器之间的协作计算技术，在用户增多的情况下，其效率会高于其他方案，但是其时延依然不是最优，原因在于本文方案在多用户场景下，不仅考虑到基站之间的协作计算，还采用了任务迁移技术，使得服务器计算资源得到充分合理的分配，因此本文方案相比其他方案，时延降低了约 22%～43%，且时延性能优于其他方案。

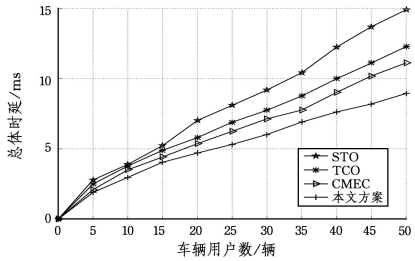

图 8 车辆用户数量与总体时延的对比

Fig. 8 Comparison of number of vehicle users and overall delay

结束语 本文研究了基于 NOMA-MEC 的车联网任务卸载、迁移与缓存策略问题，研究目标是降低针对基于 NOMA-MEC 的软件定义车联网中的任务卸载、迁移与缓存时延。本文通过合作博弈得出最佳车辆用户分组，从而优化卸载时延，并通过 Q 学习算法来优化 MEC 服务器中的任务迁移缓存联合时延。仿真结果表明，本文算法效果良好，任务时延得到优化。

参 考 文 献

[1] KARAGIANNIS G, ALTINTAS O, EKICI E, et al. Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions[J]. IEEE Communications Surveys & Tutorials, 2011, 13(4): 584-616.

[2] YANG H, ZHENG K, ZHANG K, et al. Ultra-Reliable and Low-Latency Communications for Connected Vehicles: Challenges and Solutions[J]. IEEE Network, 2020, 34(3): 92-100.

[3] FAN Y F, YUAN S, CAI Y, et al. Deep Reinforcement Learning-based Collaborative Computation Offloading Scheme in VehicularEdge Computing[J]. Computer Science, 2021, 48(5): 270-276.

[4] SONI T, ALI A R, GANESAN K, et al. Adaptive numerology A solution to address the demanding QoS in 5G-V2X[C]// Proc. IEEE Wireless Communications and Networking Conference (WCNC). Barcelona, Spain: IEEE Press, 2018: 1-6.

[5] SAITO Y, KISHIYAMA Y, BENJEBBOUR A, et al. Non-orthogonal multiple access (NOMA) for cellular future radio access[C]// 2013 IEEE 77th Vehicular Technology Conference (VTC Spring). Dresden: IEEE Press, 2013: 1-5.

[6] DI B, SONG L, LI Y, et al. V2X meets NOMA: Non-orthogonal multiple access for 5G-enabled vehicular networks[J]. IEEE Wireless Communation, 2017, 24(6): 14-21.

[7] DI B, SONG L, LI Y, et al. Non-orthogonal multiple access for high-reliable and low-latency V2X communications in 5G systems[J]. IEEE Journal on Selected Areas in Communications, 2017, 35(10): 2383-2397.

[8] DI B, SONG L, LI Y, et al. NOMA-based low-latency and high-reliable broadcast communications for 5G V2X services[C]// IEEE GLOBECOM. Singapore: IEEE Press, 2017: 1-6.

[9] WANG B, ZHANG R, CHEN C, et al. Interference hypergraph-based 3D matching resource allocation protocol for NOMAV2X networks[J]. IEEE Access, 2019, 7: 90789-90800.

[10] TANG L, XIAO J, ZHAO G F, et al. Energy Efficiency Based Dynamic Resource Allocation Algorithm for Cellular Vehicular Based on Non-Orthogonal Multiple Access[J]. Journal of Electronics and Information Technology, 2020, 42(2): 526-533.

[11] ABDELHAMID S, HASSANEIN H S, TAKAHARA G. On-Road Caching Assistance for Ubiquitous Vehicle-Based Information Services[J]. IEEE Transactions on Vehicular Technology, 2015, 64(12): 5477-5492.

[12] BLASZCZYSZYN B, GIOVANIDIS A. Optimal geographic caching in cellular networks[C]// 2015 IEEE International Conference on Communications. London: IEEE Press, 2015: 3358-3363.

[13] HIEU N T, FRANCESCO M D, YLAJAASKI A. Virtual Machine Consolidation with Multiple Usage Prediction for Energy-Efficient Cloud Data Centers[J]. IEEE Transactions on Services Computing, 2020, 13(1): 186-199.

[14] YI P, LI G, ZHANG Z. Energy Optimized Implicit Collaborative Caching Scheme for Content Centric Networking[J]. Journal of Electronics and Information Technology, 2018, 40(4): 770-777.

[15] LIU F, MA Z, WANG B, et al. A Virtual Machine Consolidation Algorithm Based on Ant Colony System and Extreme Learning Machine for Cloud Data Center[J]. IEEE Access, 2020, 8: 53-67.

[16] SADIO O, NGOM I, LISHOU C. Design and Prototyping of a Software Defined Vehicular Networking[J]. IEEE Transactions on Vehicular Technology, 2020, 69(1): 842-850.

[17] DEHGHAN M, JING B, SEETHARAM A, et al. On the complexity of optimal routing and content caching in heterogeneous networks[C]// 2015 IEEEConference on Computer Communications (INFOCOM). Hong Kong: IEEE Press, 2015: 936-944.

[18] CUI Y, YANG Z, XIAO S, et al. Traffic-Aware Virtual Machine Migration in Topology-Adaptive DCN[J]. IEEE/ACM Transactions on Networking, 2017, 25(6): 3427-3440.

[19] BOUGHZALA B, ALI R B. OpenFlow supporting interdomain virtual machine migration[C]// 2011 Eighth International Conference on Wireless and Optical Communications Networks (WOCN). Paris: IEEE Press, 2011: 1-7.

[20] LIU J, LI Y, JIN D. SDN-based live VM migration across datacenters[C] // SIGCOMM Conference. Chicago: IEEE Press, 2014: 583-584.

[21] WANG H, LI Y, ZHANG Y, et al. Virtual Machine Migration Planning in Software-Defined Networks[J]. IEEE Transactions on Cloud Computing, 2019, 7(4): 1168-1182.

[22] LI J, CHEN W, XIAO M, et al. Efficient Video Pricing and Caching in Heterogeneous Networks[J]. IEEE Transactions on Vehicular Technology, 2016, 65(10): 8744-8751.

[23] 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.

[24] ROTA G C. The number of partitions of a set[J]. The American Mathematical Monthly, 1964, 71: 498-504.

[25] HUANG Y, TAN T, WANG N, et al. Resource Allocation For D2D Communications With A Novel Distributed Q-Learning Algorithm In Heterogeneous Networks[C]// 2018 International Conference on Machine Learning and Cybernetics (ICMLC). Chengdu: IEEE Press, 2018: 533-537.

[26] HAO Y, CHEN M, HU L, et al. Energy Efficient Task Caching and Offloading for Mobile Edge Computing[J]. IEEE Access, 2018, 6: 11365-11373.

[27] MISRA S, SAHA N. Detour: Dynamic task offloading in software-defined fog for IoT applications[J]. IEEE Journal on Selected Areas in Communications, 2019, 37(5): 1159-1166.

[28] ELGENDY I A, ZHANG W, TIAN Y, et al. Resource allocation and computation offloading with data security for mobile edge computing[J]. Future Generation Computer Systems-The International Journal of Escience, 2019, 100: 531-541.

ZHANG Hai-bo, born in 1979, Ph.D, associate professor. His main research interests include vehicular networks and edge computing.

ZHANG Yi-feng, born in 1995, postgraduate. His main research interests include vehicular networks and edge computing.

（责任编辑：柯颖）