

控制与决策

Control and Decision

V2X异构车载网络下智能任务卸载策略研究

胡峰, 王文轩, 顾红

引用本文:

胡峰,王文轩,顾红. V2X异构车载网络下智能任务卸载策略研究[J]. *控制与决策*, 2022, 37(11): 3003–3011.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.0470>

您可能感兴趣的其他文章

Articles you may be interested in

[基于强化学习的边缘计算网络资源在线分配方法](#)

Reinforcement learning-based online resource allocation for edge computing network

控制与决策. 2022, 37(11): 2880–2886 <https://doi.org/10.13195/j.kzyjc.2021.0561>

[基于李雅普诺夫随机优化的车辆边缘计算资源管理](#)

Resource management of vehicle edge computing based on Lyapunov stochastic optimization

控制与决策. 2022, 37(3): 721–728 <https://doi.org/10.13195/j.kzyjc.2020.1211>

[基于深度强化学习的资源受限条件下的DIDS任务调度优化方法](#)

An optimization method for DIDS task scheduling under resource-constrained conditions based on deep reinforcement learning

控制与决策. 2022, 37(11): 3052–3057 <https://doi.org/10.13195/j.kzyjc.2021.0448>

[多无人机协同直播场景下自适应任务卸载决策](#)

Adaptive task offloading decision of multi-UAVs cooperation in live broadcasting scenario

控制与决策. 2021, 36(4): 974–982 <https://doi.org/10.13195/j.kzyjc.2019.1104>

[多无人机协同直播场景下自适应任务卸载决策](#)

Adaptive task offloading decision of multi-UAVs cooperation in live broadcasting scenario

控制与决策. 2021, 36(4): 974–982 <https://doi.org/10.13195/j.kzyjc.2019.1104>

V2X 异构车载网络下智能任务卸载策略研究

胡峰^{1†}, 王文轩¹, 顾红²

- (1. 南京信息职业技术学院 网络与通信学院, 南京 210023;
2. 南京理工大学, 电子工程与光电技术学院, 南京 210094)

摘要: 随着自动驾驶技术的迅速发展, 车辆日益增长的处理需求与资源受限的车载处理器之间的矛盾日渐突出. 车载边缘计算的出现解决了车载资源的物理限制, 增强了单个车辆的计算能力. 然而, 由于车载服务通常具有时延敏感性, 如何选择合适的通信接入技术, 更好地满足自动驾驶场景中时延要求便成为一个挑战性难题. 鉴于此, 综合考虑两种 V2X 通信接入技术, 即短距通信 (DSRC) 和基于蜂窝网的车载通信 (C-V2X), 提出一种 V2X 异构车载网络任务卸载模型. 首先分析车辆移动性特征, 并对车载资源进行虚拟化处理; 然后基于半马尔科夫决策过程原理对任务卸载问题进行建模, 分别制定状态、动作、奖励和转移概率; 最后基于强化学习智能算法获取最优任务卸载策略, 并通过大量数值仿真实验验证其任务卸载性能优于贪婪算法.

关键词: 车与万物互联; 车载边缘计算; 短距通信; 基于蜂窝网的车载通信; 任务卸载; 半马尔科夫决策过程; 强化学习

中图分类号: TP18 文献标志码: A

DOI: 10.13195/j.kzyjc.2021.0470

引用格式: 胡峰, 王文轩, 顾红. V2X 异构车载网络下智能任务卸载策略研究 [J]. 控制与决策, 2022, 37(11): 3003-3011.

Intelligent task offloading strategy in V2X heterogeneous vehicular networks

HU Feng^{1†}, WANG Wen-xuan¹, GU Hong²

- (1. School of Network and Communication, Nanjing College of Information Technology, Nanjing 210023, China;
2. School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China)

Abstract: With the rapid development of autonomous driving technology, the contradiction between the increasing processing requirements of vehicles and the resource-limited on-board processors is increasingly prominent. The emergence of vehicular edge computing solves the physical limitation of on-board resources and enhances the computing capacity of a single vehicle. However, due to the delay-sensitive of vehicular services in autonomous driving scenarios, how to choose the appropriate access technology to satisfy the delay constraint of vehicular services has become a challenge. In this paper, two kinds of V2X communication technologies, namely short range communication (DSRC) and cellular vehicular communication (C-V2X), are considered comprehensively, and a task offloading model of V2X heterogeneous vehicular network is proposed. Firstly, the characteristics of vehicle mobility are analyzed, and the on-board resources are virtualized. Then, the task offloading problem is modeled based on the principle of semi-Markov decision processes (SMDP), and the state, action, reward and transition probability are defined respectively. Finally, the optimal task offloading strategy is obtained based on the reinforcement learning intelligent algorithm, and the performance of the algorithm is proved to be better than the greedy algorithm through a large number of numerical simulations.

Keywords: V2X; vehicular edge computing; namely short range communication; C-V2X; task offloading; SMDP; reinforcement learning

收稿日期: 2021-03-21; 录用日期: 2021-08-09.

基金项目: 国家自然科学基金项目 (62004094); 江苏省自然科学基金项目 (BK20200334); 2019 年中国特色高水平高职学校和专业建设计划项目 (教职成函 [2019]14 号); 2021 年江苏高校青蓝工程优秀教学团队项目 (苏教师函 [2021]11 号 (立项公示文)).

责任编辑: 刘宝碇.

[†]通讯作者. E-mail: njcit_hufeng@163.com.

0 引言

智能车辆是构成ITS(intelligent transportation system, 智能运输系统)、智慧城市的核心部分. 为了保证驾驶安全性, 智能车辆需要装载各式各样的传感器(如GPS、摄像机、雷达和激光雷达)实时收集周围环境信息, 从而产生爆炸式增长的数据量^[1]. 据调查, 智能车辆每秒生成的数据量约1GB^[2], 日均所采集的数据量高达30TB^[3]. 面对如此庞大的数据量, 如何及时高效地计算处理以保证智能车辆的安全行驶成为了一个难题. 车辆边缘计算(vehicular edge computing, VEC)技术的出现, 解决了智能车辆物理资源受限的问题, 可利用资源虚拟化技术将分散的计算资源集中进行管理, 使资源得到高效利用. 与此同时, 将计算密集型任务卸载给周围车辆或边缘服务器, 从而满足自动驾驶场景中庞大数据量的处理需求^[4]. 在车载网络中, 车辆需要与周围车辆和路边单元(roadside unit, RSU)进行实时不断地信息交互和资源共享, 然而车载任务通常是时延敏感的, 如何选择合适的通信接入技术以满足任务处理时延要求变得尤为重要. V2X(vehicle to everything, 车与万物互联)通信技术被视为下一代车联网的关键技术, 为自动驾驶场景中时延敏感型任务处理提供了新的解决思路. V2X主要分为车辆之间(vehicle-to-vehicle, V2V)、车辆与路边基础设施之间(vehicle-to-Infrastructure, V2I)的互联技术. 在不久的将来, 车辆会在V2X通信技术的协助下实现任何场景下的高度连接, 通过实时获取动态的路况和周围环境信息, 增强车辆与环境的信息交互, 从而提高道路安全性、车辆感知能力和旅行舒适度, 减少交通拥堵、环境污染和基础设施的部署开销^[5].

V2X技术实现的挑战是选择合适的通信接入标准, 目前存在两种主流的V2X通信标准, 分别是专用短程通信(dedicated short range communication, DSRC)和基于蜂窝网的车载通信(cellular vehicle to everything, C-V2X)^[6]. DSRC主要通过一系列通信协议实现V2X通信, 其包括支持车载环境无线接入的IEEE 802.11p协议和支持网络服务和多信道操作的IEEE 1609.1.4协议^[7]. DSRC能够高效地组成动态的车载网络, 比较适应车辆移动性特征, 能够在快速移动的车辆间和路边设施实现动态稳定的连接. 尽管DSRC通常被认为是车载网络约定俗成的标准, 但最近C-V2X凭借自身诸多优点而备受关注. C-V2X是3GPP组织提出的基于蜂窝网车载通信技术, 可以借助LTE(long-term evolution)和5G等蜂窝网络实现车

辆通信, 具有覆盖范围广、容量大、服务质量高等优点^[6].

尽管两种V2X技术应用在车载环境中都有各自的潜力和优势, 但是也有各自的局限性. 由于DSRC起源于较早的IEEE 802.11标准, 最初是为低移动性的无线局域网而设计的, 从而会有一些缺陷, 如有限的覆盖、低数据速率和低服务质量. 采用DSRC技术虽能使得车辆组成动态的自组织网络, 但由于节点高速移动性, 从而会增加网络拓扑连接的不稳定性, 并进一步影响路由选择和寻址等功能^[8]. 此外, 城市中存在很多影响视距的障碍物, 极大地降低了DSRC通信性能^[9], 在一些恶劣的行驶环境中使用DSRC技术具有重大的安全隐患. 随着通信技术发展, 目前蜂窝网技术可以实现端到端时延低于100ms, 已能较好地支持车载应用, 如娱乐信息和交通管理信息共享. 但是, 在车辆密度较高时容易发生通信堵塞, 故当前蜂窝网发展状态还不能完全支持V2X车载通信^[10].

鉴于以上优缺点, 相关学者提出将C-V2X与DSRC相结合, 在异构车辆网络中集成各自优点同时克服其缺点, 利用异构接入技术更好地支持自动驾驶场景中V2X的应用^[5, 11-13]. V2X异构车载网络中, 车辆需要通过V2X通信方式和万物共享信息, 快速处理数据并作出决策. 随着智能巡航、交通管理和协作驾驶服务的出现, 更是迫切需要强大的计算资源和严格的时延容忍. 任务卸载作为VEC中的关键技术, 通过将计算量密集型任务发送给周围车辆帮助快速处理, 降低时延. 在具有异构V2X通信的VEC环境中, 卸载的可靠性受到接入技术和卸载实体的影响, 如何考虑车辆移动性特征和V2X通信方式来获取最优的任务卸载策略尤为重要, 然而目前该部分内容较少有学者进行研究.

马尔科夫决策过程(Markov decision process, MDP)用于描述强化学习场景下智能体与环境之间的交互关系, 并通过最大化长期收益找出动态随机过程中的最优决策. 当马尔科夫决策过程中决策时刻服从随机分布时, 则为半马尔科夫决策过程(semi-markov decision process, SMDP). 在V2X异构车载网络中, 车辆和任务到达或离开的时间点是随机的且间隔不固定, 在任务到达后又需尽快做出决策, 因此SMDP模型能较好地适用于V2X异构车载网络中任务卸载策略研究.

本文综合异构VEC中V2X技术和车辆移动性特征分析任务卸载过程, 获取智能的任务卸载方案. 其主要贡献总结如下:

1) 在VEC网络中,考虑车辆移动性特征以及DSRC、C-V2X相结合的异构通信接入技术分析任务卸载过程,并建立基于VEC的V2X异构车载网络系统模型;

2) 综合考虑任务卸载过程中能量和时延参数,根据不同的卸载方式制定合理的奖励;

3) 基于SMDP原理分析任务卸载模型,分别制定状态、动作、奖励、转移概率,并设计强化学习智能迭代算法,获取智能任务卸载策略。

1 相关文献

Xiong等^[5]提出了一种支持DSRC、C-V2X和毫米波3种通信方式的异构车载网络。首先利用随机网络模型推导出不同V2X通信方式下卸载时延上限,然后通过最小化通信和计算开销设计任务卸载优化算法,结果表明,该算法能有效降低资源消耗和卸载失败概率。Shen等^[11]基于蜂窝网和DSRC异构接入技术,提出了一种根据服务需求进行自适应选择的无线接入方案,并通过仿真表明了异构接入方案明显优于单纯采用DSRC的接入方案,能够同时满足低时延安全消息、高带宽视频数据流的传输需求。Huang等^[13]研究了SDN异构车载网络中的资源分配问题,通过最大化系统中所有车辆的服务质量获取最优调度方案,提高了网络中消息传输速率和资源利用率。Zhang等^[14]提出一种SDN辅助的MEC车载网络体系结构,并基于DSRC通信方式制定一个NP-hard多目标优化问题以最小化系统开销,最终通过将优化问题依次分解求出最优卸载决策和计算资源分布策略。Feng等^[15]提出一种MEC场景下的C-V2X任务卸载模型,综合考虑了链路稳定性和时延等因素,并通过最小化MEC服务器和车辆的功率消耗求解资源分配方案。赵海涛等^[16]提出一种由车辆、RSU与MEC服务器构成的车载边缘网络任务卸载模型,并考虑了计算任务类型、信道增益均不相同的特征,以最大化计算任务处理速率为优化目标来获取最优策略。然而,该文献仅考虑了DSRC单种通信方式,并未考虑高效的异构通信方式。

综上所述,关于V2X异构车载网络中不同接入方式下对任务卸载的影响,以及能量和时延对任务卸载策略的影响,至今较少有相关研究文献,故而激发本文开展相关研究。

2 任务卸载问题建模

图1展示了V2X异构车载网络系统模型,图中高速公路上行驶的车辆和路边的RSU形成一个动态的车载边缘网络,车辆支持DSRC和C-V2X两种通信

接入方式,进一步可细分为DSRC-V2V、DSRC-V2I、Cellular-V2I和Cellular-V2V。由于Cellular-V2V接入方式时延较大,而Cellular-V2I接入方式在覆盖率和可靠性方面都表现较好,故这里仅忽略了Cellular-V2V通信接入方式^[17]。假设每个车辆的计算能力均相同,即每个车辆提供一个单位计算资源(resource Unit, RU)。车辆可以通过Cellular-V2I方式接入核心网并访问远程云计算中心,从而获取充沛的计算资源。另外,网络覆盖范围内具有确定数目 N_R 个RSU,通常RSU具备较强的计算能力,若每个RSU具备 T_R 个计算单元,则RSU具备的资源总数目为 $M = N_R \cdot T_R$ 。在资源虚拟化技术的帮助下,所有的计算资源都可被整合到资源池中进行统一管理和分配^[18]。

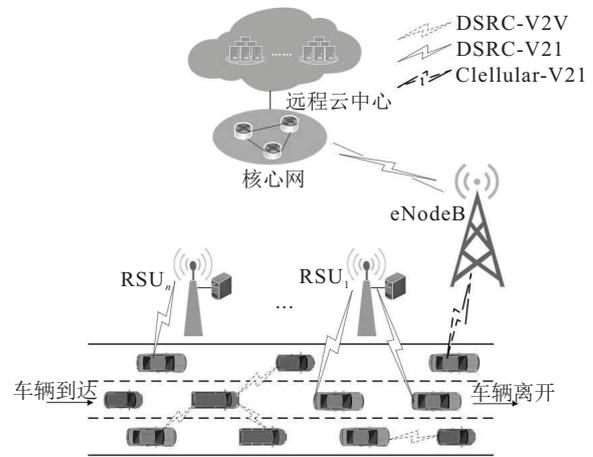


图1 V2X异构车载网络系统模型

在V2X异构车载网络中,计算资源的充沛程度受到事件的影响,如车辆到达和离开、请求到达和离开。计算资源的实时动态变化会导致任务卸载成功或失败,假设车辆到达和离开服从参数为 λ_v 和 μ_v 的泊松分布,任务到达和离开服从参数为 λ_t 和 μ_t 的泊松分布^[19]。考虑车辆移动性特征,采用半马尔科夫决策过程理论建模分析该场景下的任务卸载问题。半马尔科夫决策过程主要应用于决策、优化、控制和管理等领域,能够合理准确地建模基于时间序列的决策问题,并高效地寻找出最优策略。下面将参照SMDP原理,分别定义在V2X异构车载网络中系统状态、动作、奖励和转移概率的表达方式。

2.1 状态集合

在V2X异构车载网络中,系统状态应包含RU总数目、正在被处理的任务数目和当前发生的事件。对于任一事件, e 表示从事件集合 $\{V_{+1}, V_{-1}, A, L_1, \dots, L_M, Q_1, \dots, Q_N\}$ 中取值,其中 V_{+1} 和 V_{-1} 分别表示有车辆进入或离开系统, A 表示系统中某辆车发出一个请求, L_i 表示被本地 i 个RU处理的任務离开, Q_i

表示被RSU中*i*个RU处理的任务离开,*M*表示本地可以分配的RU最大数目,*N*表示RSU可以分配的RU最大数目.在特定事件*e*下系统状态可表示为 $(H, m_1, \dots, m_M, n_1, \dots, n_N, e)$.其中:*H*为网络中计算单元的总数目, m_i 为正在被本地*i*个计算单元处理的任务数目, n_i 为正在被RSU中*i*个计算单元处理的任务数目.状态集合*S*可表示为

$$S = \{s | s = (H, m_1, \dots, m_M, n_1, \dots, n_N, e)\}. \quad (1)$$

若用*H*表示系统中计算单元的总数目,则忙碌的计算单元数目应该满足数量关系限制,即 $\sum_{i=1}^M i \cdot m_i + \sum_{j=1}^N j \cdot n_j \leq H$,且*H*不能超过*K*,这里*K*表示V2X异构车载系统所能支持的最多RU数目,即最大计算负载能力.

2.2 动作集合

动作集合归纳了所有可能发生的决策行为,即系统在不同的事件下分配计算单元的数目.参照本模型得出,动作可从集合 $\{0, a_1, \dots, a_M, b_1, \dots, b_N, c\}$ 中选取.由于动作和状态两者相互关联,在状态*s*下采取动作*a*记录为 $a(s)$. $a(s) = 0$ 表示无需采取任何动作, $a(s) = a_i$ 表示通过DSRC-V2V方式将任务分配给本地*i*个计算单元来处理, $a(s) = b_i$ 表示通过DSRC-V2I方式分配给RSU中*i*个计算单元来处理, $a(s) = c$ 表示通过Cellular-V2I方式将任务上传给远程云.当系统中任务离开,车辆到达或离开时,不需要采取任何动作;当有任务到达,系统可能选择卸载给本地*i*个计算单元处理、卸载给RSU中*i*个计算单元处理或者上传给远程云处理.综上,动作集合*A_c*表示为

$$A_c = \begin{cases} \{0\}, & e \in \{V_{+1}, V_{-1}, L_1, \dots, L_M, Q_1, \dots, Q_N\}; \\ \{a_i, b_i, c\}, & e = A. \end{cases} \quad (2)$$

2.3 系统奖励

奖励表明系统在不同状态下采取不同动作时得到的好处,故本节将综合考虑能耗和时延以衡量系统中任务卸载的奖励.当采取一个确定的动作时,系统会获得一个立即收益 $I(s, a)$.此外,由于系统采取动作后需要经历一段时间才能到达下个状态,在这段时间内系统的开销为 $G(s, a)$.综上,在状态*s*采取动作*a*的系统奖励 $R(s, a)$ 可以表示为立即收益和系统开销的差值,即

$$R(s, a) = I(s, a) - G(s, a). \quad (3)$$

下面进一步描述该场景中的立即收益和系统开销.

2.3.1 立即收益

不同的动作和事件下立即收益也不同,由此将立即收益的计算方式划分为以下5类:

1) 当 $e = A, a = a_i$ 时:此时请求车辆产生一个任务,系统根据整体决策决定将该任务分配给本地*i*个RU处理并将计算结果返回.*E*表示任务在车辆自身处理时所消耗的能量,*P*表示车辆的发送功率, t_1 表示通过DSRC-V2V方式发送任务到本地RU所需要的时间,那么节省的能量可表示为 $(E - P \cdot t_1)$. T_{self} 表示任务在车辆自身处理所需要的时间, T_i 表示任务被*i*个RU处理所需要的时间,计算为 $1/(i \cdot \mu_t)$,那么节省的时延表示为 $(T_{\text{self}} - t_1 - T_i)$.若用 ω_e 和 ω_t 分别表示节省的能量和时间的权重因子,且满足 $\omega_e + \omega_t = 1$, ζ_e 和 ζ_t 分别为节省的能量和时延的价格, γ 为传输时延的价格,则可以综合考虑能量和时延来表示系统收益,即 $\omega_e \zeta_e (E - P \cdot t_1) + \omega_t \zeta_t (T_{\text{self}} - t_1 - T_i) - \gamma t_1$.

2) 当 $e = A, a = b_i$ 时:此时任务被分配给RSU中*i*个RU处理, t_2 表示通过DSRC-V2I方式发送任务到RU所需要的时间.参照上一种情况,得出此时的系统收益为 $\omega_e \zeta_e (E - P \cdot t_2) + \omega_t \zeta_t (T_{\text{self}} - t_2 - T_i) - \gamma t_2$.

3) 当 $e = A, a = c$ 时:此时任务被分配给RC处理, t_3 表示通过Cellular-V2I方式发送任务到BS所需要的时间, t_4 表示任务从BS发送到RC所需要的时间.*D*表示任务发送给RC处理时,车辆由于没有消耗本地计算资源而获取的收益,计算为 $\omega_e \zeta_e (E - P \cdot t_3) + \omega_t \zeta_t (T_{\text{self}} - t_3 - t_4)$,此时的系统收益为 $D - \gamma(t_3 + t_4)$.

4) 当 $e \in \{V_{+1}, L_1, \dots, L_M, Q_1, \dots, Q_N\}, a = 0$ 时:此时系统中发生的事件可能是车辆到达、被本地或RSU中*i*个RU处理的任务离开.在这些事件下,系统不会采取任何措施故不会有任何收益.

5) 当 $e = V_{-1}, a = 0$ 时:此时发生的事件是车辆离开.若当前忙碌的RU数目满足关系 $\sum_{i=1}^M i \cdot m_i +$

$\sum_{j=1}^N j \cdot n_j < H$,则此时车辆离开不会有任何收益.若

满足关系 $\sum_{i=1}^M i \cdot m_i + \sum_{j=1}^N j \cdot n_j = H$,则表明系统中

RU全部处于忙碌状态,此时任何车辆的离开都可能导致任务卸载失败,损害系统整体收益,那么应该追加一定的惩罚 η 避免这种情况.

综上,在不同的事件和动作下,系统的立即收益表示为

$$I(x, a) =$$

$$\left\{ \begin{array}{l} \omega_e \zeta_e (E - P \cdot t_1) + \omega_t \zeta_t (T_{\text{self}} - t_1 - T_i) - \gamma t_1, \\ e = A(i > 0), a = a_i; \\ \omega_e \zeta_e (E - P \cdot t_2) + \omega_t \zeta_t (T_{\text{self}} - t_2 - T_i) - \gamma t_2, \\ e = A(i > 0), a = b_i; \\ D - \gamma(t_3 + t_4), e = A, a = c; \\ 0, e \in \{V_{+1}, L_1, \dots, L_M, Q_1, \dots, Q_N\}, a = 0; \\ 0, e = V_{-1}, a = 0, \sum_{i=1}^M i \cdot m_i + \sum_{j=1}^N j \cdot n_j < H; \\ -\eta, e = V_{-1}, a = 0, \sum_{i=1}^M i \cdot m_i + \sum_{j=1}^N j \cdot n_j = H. \end{array} \right. \quad (4)$$

2.3.2 系统开销

$G(s, a)$ 表示系统在两次决策时间内的期望折扣开销。假设连续两次决策间时长服从参数为 α 的负指数分布,折扣后的系统开销^[18]表示为

$$G(s, a) = C(s, a) E_s^a \left\{ \int_0^\tau e^{-\alpha t} dt \right\} = \frac{C(s, a)}{\alpha + \sigma(s, a)}. \quad (5)$$

其中: $\sigma(s, a)$ 为状态 s 动作 a 下期望的事件速率; $C(s, a)$ 为状态 s 动作 a 下期望的开销速率,表示已分配的RU数目,即

$$C(s, a) = \sum_{i=1}^M i \cdot m_i + \sum_{j=1}^N j \cdot n_j. \quad (6)$$

在不同动作和事件下期望的服务速率计算为

$$\sigma(s, a) = \left\{ \begin{array}{l} X + (W + i)\mu_t, e = A, a = a_i/b_i; \\ X + W\mu_t, e = A, a = c; \\ X + (W - i)\mu_t, e = L_i/Q_i, a = 0; \\ Y + W\mu_t, e = V_{+1}, a = 0; \\ Z + W\mu_t, e = V_{-1}, a = 0. \end{array} \right. \quad (7)$$

其中

$$\begin{aligned} X &= (H - M)\lambda_t + \lambda_v + \mu_v, \\ Y &= (H + 1 - M)\lambda_t + \lambda_v + \mu_v, \\ Z &= (H - 1 - M)\lambda_t + \lambda_v + \mu_v, \\ W &= \sum_{i=1}^M i \cdot m_i + \sum_{j=1}^N j \cdot n_j. \end{aligned}$$

为了书写简洁,上式将重复的部分分别用 X 、 Y 、 Z 和 W 进行替换。

2.4 转移概率

在 SMDP 模型中,下一个状态取决于当前的状态和动作,系统在当前状态采取动作后到达下一个

状态的概率定义为转移概率。从当前状态 s 到下一个状态 s' 的转移概率记为 $P(s'|s, a)$ 。下面根据当前的事件和动作类型,分4种情况计算转移概率:1) 任务到达且系统卸载给 i 个计算单元处理时,若下一个发生的事件还是任务到达,则此时转移概率可以用任务到达的速率和事件总速率的比值表示,即 $M\lambda_t(i)/\sigma(s, a)$;2) 若下一个发生的事件是被 i 个计算单元处理的任務离开,则此时为处理该任务而忙碌的计算单元数目为 $i(n_i + 1)$,得出此时转移概率为 $(n_i + 1)i\mu_t/\sigma(s, a)$;3) 若下一个发生的事件是被 $j(i \neq j)$ 个计算单元处理的任務离开,则此时为处理该任务而忙碌的计算单元数目为 $(n_j + 1)j$,类推可得此时转移概率为 $(n_j + 1)j\mu_t/\sigma(s, a)$;4) 若下一个发生的事件是车辆到达或离开,则此时转移概率用车辆到达率或离开率与事件总速率的比值,即 $\lambda_f/\sigma(s, a)$ 和 $\mu_f/\sigma(s, a)$ 。综上,在事件 A 和动作 i 这种情况下,转移概率计算为

$$1) s = (H, m_1, \dots, m_M, n_1, \dots, n_N, A), a = a_i: \quad (8)$$

$$P(s'|s, a) = \left\{ \begin{array}{l} \frac{(H - M)\lambda_t}{\sigma(s, a)}, s' = (X_1, A); \\ \frac{(m_i + 1)i\mu_t}{\sigma(s, a)}, s' = (X_1, L_i); \\ \frac{m_j j \mu_t}{\sigma(s, a)}, i \neq j, s' = (X_1, L_j); \\ \frac{n_i i \mu_t}{\sigma(s, a)}, s' = (X_1, Q_i); \\ \frac{\lambda_v}{\sigma(s, a)}, s' = (X_1, V_{+1}); \\ \frac{\mu_v}{\sigma(s, a)}, s' = (X_1, V_{-1}). \end{array} \right.$$

其中 $X_1 = H, m_1, \dots, m_i + 1, \dots, m_M, n_1, \dots, n_N$ 。

以此类推其他6种情况下的转移概率计算方式。

$$2) s = (H, m_1, \dots, m_M, n_1, \dots, n_N, A), a = b_i: \quad (9)$$

$$P(s'|s, a) = \left\{ \begin{array}{l} \frac{(H - M)\lambda_t}{\sigma(s, a)}, s' = (X_2, A); \\ \frac{m_i i \mu_t}{\sigma(s, a)}, s' = (X_2, L_i); \\ \frac{(n_i + 1)i\mu_t}{\sigma(s, a)}, s' = (X_2, Q_i); \\ \frac{n_j j \mu_t}{\sigma(s, a)}, i \neq j, s' = (X_2, Q_j); \\ \frac{\lambda_v}{\sigma(s, a)}, s' = (X_2, V_{+1}); \\ \frac{\mu_v}{\sigma(s, a)}, s' = (X_2, V_{-1}). \end{array} \right.$$

其中 $X_2 = H, m_1, \dots, m_M, n_1, \dots, n_i + 1, \dots, n_N$ 。

$$3) s = (H, m_1, \dots, m_M, n_1, \dots, n_N, A), a = c:$$

$$P(s'|s, a) = \begin{cases} \frac{(H-M)\lambda_t}{\sigma(s, a)}, & s' = (X_3, A); \\ \frac{m_i i \mu_t}{\sigma(s, a)}, & s' = (X_3, L_i); \\ \frac{n_i i \mu_t}{\sigma(s, a)}, & s' = (X_3, Q_i); \\ \frac{\lambda_v}{\sigma(s, a)}, & s' = (X_3, V_{+1}); \\ \frac{\mu_v}{\sigma(s, a)}, & s' = (X_3, V_{-1}). \end{cases} \quad (10)$$

其中 $X_3 = H, m_1, \dots, m_M, n_1, \dots, n_N$.

4) $s = (H, m_1, \dots, m_M, n_1, \dots, n_N, L_i), a = 0$:

$$P(s'|s, a) = \begin{cases} \frac{(H-M)\lambda_t}{\sigma(s, a)}, & s' = (X_4, A); \\ \frac{(m_i-1)i\mu_t}{\sigma(s, a)}, & s' = (X_4, L_i); \\ \frac{m_j j \mu_t}{\sigma(s, a)}, & i \neq j, s' = (X_4, L_j); \\ \frac{n_i i \mu_t}{\sigma(s, a)}, & s' = (X_4, Q_i); \\ \frac{\lambda_v}{\sigma(s, a)}, & s' = (X_4, V_{+1}); \\ \frac{\mu_v}{\sigma(s, a)}, & s' = (X_4, V_{-1}). \end{cases} \quad (11)$$

其中 $X_4 = H, m_1, \dots, m_i - 1, \dots, m_M, n_1, \dots, n_N$.

5) $s = (H, m_1, \dots, m_M, n_1, \dots, n_N, Q_i), a = 0$:

$$P(s'|s, a) = \begin{cases} \frac{(H-M)\lambda_t}{\sigma(s, a)}, & s' = (X_5, A); \\ \frac{m_i i \mu_t}{\sigma(s, a)}, & s' = (X_5, L_i); \\ \frac{(n_i-1)i\mu_t}{\sigma(s, a)}, & s' = (X_5, Q_i); \\ \frac{n_j j \mu_t}{\sigma(s, a)}, & i \neq j, s' = (X_5, Q_j); \\ \frac{\lambda_v}{\sigma(s, a)}, & s' = (X_5, V_{+1}); \\ \frac{\mu_v}{\sigma(s, a)}, & s' = (X_5, V_{-1}). \end{cases} \quad (12)$$

其中 $X_5 = H, m_1, \dots, m_M, n_1, \dots, n_i - 1, \dots, n_N$.

6) $s = (H, m_1, \dots, m_M, n_1, \dots, n_N, V_{+1}), a =$

0:

$$P(s'|s, a) = \begin{cases} \frac{(H+1-M)\lambda_t}{\sigma(s, a)}, & s' = (X_6, A); \\ \frac{m_i i \mu_t}{\sigma(s, a)}, & s' = (X_6, L_i); \\ \frac{n_i i \mu_t}{\sigma(s, a)}, & s' = (X_6, Q_i); \\ \frac{\lambda_v}{\sigma(s, a)}, & s' = (X_6, V_{+1}); \\ \frac{\mu_v}{\sigma(s, a)}, & s' = (X_6, V_{-1}). \end{cases} \quad (13)$$

其中 $X_6 = H + 1, m_1, \dots, m_M, n_1, \dots, n_N$.

7) $s = (H, m_1, \dots, m_M, n_1, \dots, n_N, V_{-1}), a = 0$; 与上述情况类似,只需要将 $H + 1$ 变为 $H - 1$ 即可,此处不再赘述.

3 智能任务卸载策略

本节基于强化学习的思想,通过最大化SMDP的长期回报设计值迭代算法,从而获取最优任务卸载策略.智能体根据策略 π 在特定状态采取动作后的价值可以用一个期望函数 V 表示,又称值函数,它可以表明当前状态下策略 π 的长期影响,进一步得出值函数基于状态的递推关系.基于该递推关系逐步缩小差值以逼近最优决策,这便是值迭代算法的核心思想.在每次迭代过程中,都会产生一种策略,通过不断评估策略的好坏以确定当前环境下的最优策略.首先将迭代次数和每个状态的值函数初始化为零,然后在每一次迭代中,基于前一次迭代的值函数、奖励和转移概率,通过贝尔曼方程(即式(14)),计算出各状态的最大值函数.例如,在第 $k + 1$ 次迭代中,根据第 k 次迭代的值函数、奖励和转移概率计算当前环境下最大值函数,有

$$V_{k+1}(s) = \max_{a \in A_c} \left[R(s, a) + \gamma \sum_{s' \in S} P(s'|s, a) V_k(s') \right], \quad (14)$$

其中 γ 是折扣因子,取值为 $\gamma = \sigma(s, a) / (\alpha + \sigma(s, a))$.下面对奖励、转移概率、折扣因子进行归一化操作,将连续时间的SMDP模型转换为离散时间SMDP模型^[20],归一化方法如下:

$$\hat{R}(s, a) = R(s, a) \frac{\alpha + \sigma(s, a)}{\alpha + y}. \quad (15)$$

$$\hat{\gamma} = \frac{y}{(y + \alpha)}. \quad (16)$$

$$\hat{P}(s'|s, a) = \begin{cases} 1 - \frac{[1 - P(s|s, a)]\sigma(s, a)}{y}, & s' = s; \\ \frac{P(s'|s, a)\sigma(s, a)}{y}, & s' \neq s. \end{cases} \quad (17)$$

其中归一化因子 $y = K \cdot \lambda_t + K \cdot N \cdot \mu_t + \lambda_f + \mu_f$.那么,贝尔曼最优方程可以被重写为

$$\hat{V}_{k+1}(s) = \max_{a \in A_c} \left[\hat{R}(s, a) + \hat{\gamma} \sum_{s' \in S} \hat{P}(s'|s, a) \hat{V}_k(s') \right]. \quad (18)$$

在 $k + 1$ 次迭代中获取每个状态的最大值函数后,计算连续两次迭代过程中的差值.若最大的差值 $\|\hat{V}_{k+1} - \hat{V}_k\|$ 小于阈值 ϵ ,则有

$$\epsilon = \frac{\varepsilon(1 - \hat{\gamma})}{2\hat{\gamma}}, \quad (19)$$

值迭代算法停止, 此时卸载策略为最优策略, 记为 π^* . 在最优卸载策略下状态 s 的动作为

$$\pi^*(s) = \arg \max_{a \in A_c} \left[\hat{R}(s, a) + \hat{\gamma} \sum_{s' \in S} \hat{P}(s'|s, a) \hat{V}_{k+1}(s') \right]. \quad (20)$$

若迭代差值大于阈值, 则迭代次数累加 1, 算法继续, 直到找出最优方案. 综上, 智能迭代算法的伪代码如下所示.

算法 1 基于 SMDP 智能任务卸载策略.

step 1: 输入系统状态集合 S , 动作集合 A_c , 系统奖励 $R(s, a)$, 转移概率 $P(s'|s, a)$, 收敛速率 ϵ .

step 2: 进行初始化, 使所有状态 s 的奖励函数 $V(s) = 0$, 并使 $k = 0$.

step 3: 对状态集合 S 中的每个状态 s , 使

$$\hat{V}_{k+1}(s) = \max_{a \in A} \left[\hat{R}(s, a) + \hat{\gamma} \sum_{s' \in S} \hat{P}(s'|s, a) \hat{V}_k(s') \right].$$

step 4: 满足 $\|\hat{V}_{k+1} - \hat{V}_k\| < \epsilon$ 时, 进入 step 5, 否则将 k 自增 1 返回到 step 3.

step 5: 输出最优卸载策略 π^* .

4 仿真结果和分析

开展数值仿真实验以验证最优方案的性能, 仿真工具为 Matlab 2020 a, 仿真场景如第 2 节所述. 实验中首先初始化 SMDP 模型的四元组, 即由式 (1)~(3) 初始化系统状态集 S 、动作集 A_c 、系统奖励 $R(s, a)$, 由式 (8)~(13) 初始化转移概率 $P(s'|s, a)$. 利用第 3 节描述的值迭代算法, 通过最大化 SMDP 模型的长期收益获取最优方案. 最后将所提出方案与贪婪方案、智能粒子群方案进行性能对比和分析. 贪婪方案始终选择最大数目的可用资源来处理任务, 以便获取最大的当前收益. 智能粒子群方案中智能体能够与环境交互, 并通过学习和经验积累逐步改进自身策略, 最终形成一种最优策略^[19]. 这两种方案与所提出的 SMDP 方案有些类似之处, 较为适合与所提出方案作性能对比. 假设每个任务最多可以卸载给 2 个 RU, 即 $N = 2$, 动作 a_1 、 a_2 、 b_1 、 b_2 和 c . 具体仿真参数见表 1.

图 2 展示了 V2X 异构车载网络中动作概率伴随资源最大数目的变化趋势. 其中动作 a_1 和 b_1 的概率随着最大资源数目增加而减少, 这是因为资源数目逐步增多意味着网络中资源相对充足, 此时系统会避免采取保守的策略, 即动作 a_1 和 b_1 . 另外, 最大资源数目增加也意味着网络中容纳的车辆数目增多, 为了降低任务卸载时延, 获取更多收益, 采取动作 a_2 的概率会迅速增长. 然而, 随着车辆数目增多, 任务处理的请求

表 1 V2X 异构车载网络仿真参数^[18,20-23]

参数	取值	参数	取值
N	2	K	8~15
N_R	2	T_R	2~6
λ_t	5	μ_t	5
λ_v	10	μ_v	10
E	20	P	4
γ	2	η	18
α	0.1	T_{self}	20
ω_e	0.5	ω_t	0.5
ζ_e	2	ζ_t	2

数目也相对增多, 网络中的资源逐渐开始匮乏, 为了获得更多长期收益, 系统开始采取保守策略, 即降低动作 a_2 的概率. 与此同时增加了动作 c 的概率, 即将更多的任务通过 Cellular-V2I 方式卸载给远程云. 上述动作概率反映了任务卸载策略, 可以看出所提出的任务卸载策略能够自适应地随着计算资源的变化做出最佳决策, 体现其智能性.

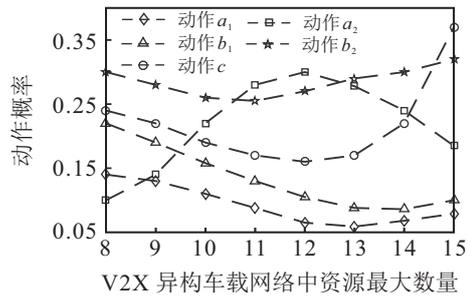


图 2 动作概率随资源最大数目变化趋势

图 3 展示了 V2X 异构车载网络中动作概率随单个 RSU 所携带资源的数目变化趋势. 可以看出, 动作 b_1 和 b_2 随着 RSU 所携带资源的数目增加而增加, 这是因为 RSU 所携带资源的数目增加代表 RSU 的计算能力增强, 此时车辆会首选将任务卸载给 RSU 处理来保留本地的计算资源, 即动作 a_1 和 a_2 减少. 同时, 为了获取更大的收益, 车辆会避免将任务卸载给远程云, 即动作 c 也减少.

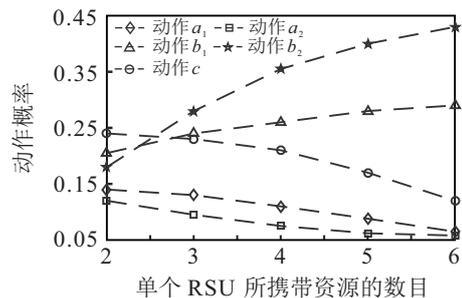


图 3 动作概率随单个 RSU 所携带资源的数目变化趋势

图4展示了V2X异构车载网络中长期收益伴随资源最大数目变化趋势。可以看出,系统的长期收益随着资源最大数目增加先增加后减少,本文SMDP策略获取的长期收益总是优于贪婪策略和智能粒子群策略。这是因为贪婪策略总是根据当前情况分配最多的计算单元而忽略未来的长期收益,可以将贪婪策略视为一种“目光短浅”的方案,而本节提出的方案则是一种有“长远目光”的方案,智能粒子群策略则属于这两者之间的一种过渡方案,通过自身的迭代进化,逐步逼近SMDP策略。另外,随着资源最大数目的增加,智能粒子群策略和SMDP策略差距越来越小,这是因为资源最大数目的增加代表SMDP模型中系统状态变多,即样本量逐渐变大,智能粒子群策略更容易找到全局最优解,但整体表现还是SMDP策略更胜一筹。随着车辆和任务请求数目的增多,网络负载加重,系统会选择将更多的任务卸载给远程云处理,从而减少了系统收益。

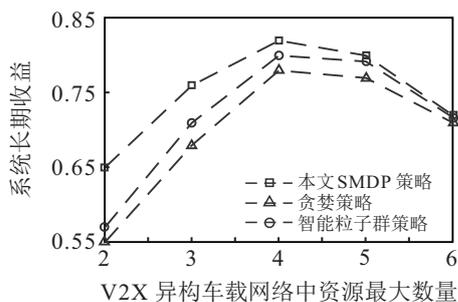


图4 系统长期收益随资源最大数目变化趋势

长期收益随单个RSU所携带资源的数目变化趋势如图5所示。可以看出,系统长期收益随单个RSU所携带资源数目的增加而增加,本文提出的策略同样优于贪婪策略和智能粒子群策略。这是因为贪婪策略倾向于基于当前系统环境寻找最优解,容易陷入局部最优解;智能粒子群策略通过不断学习进化,能够从局部最优解向全局最优解逼近,而SMDP策略根据系统长期收益,可获取每一步的全局最优解。另外,图5中长期收益性能提升的比率小于图4,这是因为随着单个RSU所携带资源数目增多,系统中资源越来越

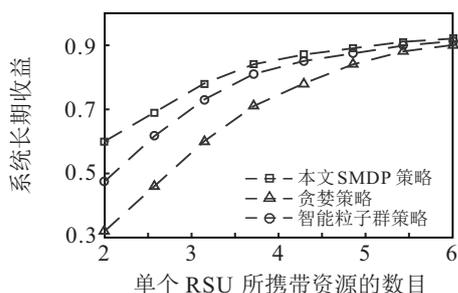


图5 长期收益随单个RSU所携带资源的数目变化趋势

越充沛,为获取更多收益,此时系统会尝试分配更多的计算资源处理任务,所以与贪婪策略的性能会越来越接近。

5 结论

自动驾驶场景中爆炸式增长的数据量,使得资源计算成为了瓶颈。车载边缘计算解决了车载资源的物理限制,增强了单个车辆的计算能力。然而,车载服务具有严格的时延敏感性,选择合适的通信接入技术才能满足车载服务的时延限制。本文综合考虑DSRC和C-V2X两种V2X通信接入技术,提出了一种V2X异构车载网络任务卸载模型,并基于半马尔科夫决策过程原理对任务卸载问题进行建模,分别制定了状态、动作、奖励和转移概率。最后基于强化学习思想设计智能值迭代算法以获取最优任务卸载策略,并通过数值仿真实验,对所提出的卸载策略性能进行分析和验证。在未来的工作中,将考虑任务的异构型特征,即不同优先级的任务对卸载时延要求不同,以探索满足异构时延要求的任务卸载策略。

参考文献(References)

- [1] Liu L K, Lu S D, Zhong R, et al. Computing systems for autonomous driving: State of the art and challenges[J]. IEEE Internet of Things Journal, 2021, 8(8): 6469-6486.
- [2] Zhou Z Y, Yu H J, Xu C, et al. BEGIN: Big data enabled energy-efficient vehicular edge computing[J]. IEEE Communications Magazine, 2018, 56(12): 82-89.
- [3] The connected vehicle: Big Data, big opportunities, SAS institute, white paper[EB/OL]. (2008-05-03) [2020-06-10]. <https://www.sas.com>.
- [4] Qiao G H, Leng S P, Maharjan S, et al. Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks[J]. IEEE Internet of Things Journal, 2020, 7(1): 247-257.
- [5] Xiong K, Leng S P, Huang C W, et al. Intelligent task offloading for heterogeneous V2X communications[J]. IEEE Transactions on Intelligent Transportation Systems, 2021, 22(4): 2226-2238.
- [6] Noor-A-Rahim M, Liu Z L, Lee H, et al. A survey on resource allocation in vehicular networks[J]. IEEE Transactions on Intelligent Transportation Systems, 2020(99): 1-21.
- [7] Kenney J B. Dedicated short-range communications (DSRC) standards in the United States[J]. Proceedings of the IEEE, 2011, 99(7): 1162-1182.
- [8] Yousefi S, Mousavi M S, Fathy M. Vehicular ad hoc networks (VANETs): Challenges and perspectives[C]. The 6th International Conference

- on ITS Telecommunications. Chengdu, 2006: 761-766.
- [9] Huang X, Zhao D, Peng H. Empirical study of DSRC performance based on safety pilot model deployment data[J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 18(10): 2619-2628.
- [10] Moubayed A, Shami A. Softwarization, virtualization & machine learning for intelligent & effective V2X communications[J]. IEEE Intelligent Transportation Systems Magazine, DOI: 10.1109/MITS.2020.3014124.
- [11] Shen X M, Li J, Chen L, et al. Heterogeneous LTE/DSRC approach to support real-time vehicular communications[C]. The 10th International Conference on Advanced Infocomm Technology. Stockholm, 2018: 122-127.
- [12] Mir Z H, Toutouh J, Filali F, et al. Enabling DSRC and C-V2X integrated hybrid vehicular networks: Architecture and protocol[J]. IEEE Access, 2020, 8: 180909-180927.
- [13] Huang W, Ding L H, de Meng, et al. QoE-based resource allocation for heterogeneous multi-radio communication in software-defined vehicle networks[J]. IEEE Access, 2018, 6: 3387-3399.
- [14] Zhang H B, Wang Z X, Liu K J. V2X offloading and resource allocation in SDN-assisted MEC-based vehicular networks[J]. China Communications, 2020, 17(5): 266-283.
- [15] Feng L, Li W J, Lin Y X, et al. Joint computation offloading and URLLC resource allocation for collaborative MEC assisted cellular-V2X networks[J]. IEEE Access, 2020, 8: 24914-24926.
- [16] 赵海涛, 张唐伟, 陈跃, 等. 基于DQN的车载边缘网络任务分发卸载算法[J]. 通信学报, 2020, 41(10): 172-178.
(Zhao H T, Zhang T W, Chen Y, et al. Task distribution offloading algorithm of vehicle edge network based on DQN[J]. Journal on Communications, 2020, 41(10): 172-178.)
- [17] Sial M N, Deng Y S, Ahmed J, et al. Stochastic geometry modeling of cellular V2X communication over shared channels[J]. IEEE Transactions on Vehicular Technology, 2019, 68(12): 11873-11887.
- [18] Zheng K, Meng H L, Chatzimisios P, et al. An SMDP-based resource allocation in vehicular cloud computing systems[J]. IEEE Transactions on Industrial Electronics, 2015, 62(12): 7920-7928.
- [19] Yang Y, Zhang Q, Wang Y, et al. Adaptive resources allocation algorithm based on modified PSO for cognitive radio system[J]. China Communications, 2019, 16(5): 83-89.
- [20] Lin C C, Deng D J, Yao C C. Resource allocation in vehicular cloud computing systems with heterogeneous vehicles and roadside units[J]. IEEE Internet of Things Journal, 2018, 5(5): 3692-3700.
- [21] Puterman M. Markov decision processes: Discrete stochastic dynamic programming[M]. New York: Wiley, 2005: 530-573.
- [22] Wu Q, Ge H M, Liu H X, et al. A task offloading scheme in vehicular fog and cloud computing system[J]. IEEE Access, 2019, 8: 1173-1184.
- [23] Wu Q, Liu H X, Wang R H, et al. Delay-sensitive task offloading in the 802.11p-based vehicular fog computing systems[J]. IEEE Internet of Things Journal, 2019, 7(1): 773-785.

作者简介

胡峰(1982-), 男, 副教授, 从事车载通信、光通信网络、物联网应用开发等研究, E-mail: njcit_hufeng@163.com;

王文轩(1982-), 女, 副教授, 博士, 从事通信网络、微波光子集成等研究, E-mail: wangwx@njcit.cn;

顾红(1967-), 男, 教授, 博士生导师, 从事噪声雷达通信理论和应用开发等研究, E-mail: guhong@mail.njust.edu.cn.

(责任编辑: 郑晓蕾)