Journal of System Simulation

Volume 33 | Issue 3 Article 1

3-18-2021

A Survey of Edge Computing Resource Allocation and Task Scheduling Optimization

Wang Ling
Department of Automation, Tsinghua University, Beijing 100084, China;

Chuge Wu
Department of Automation, Tsinghua University, Beijing 100084, China;

Wenhui Fan
Department of Automation, Tsinghua University, Beijing 100084, China;

A Survey of Edge Computing Resource Allocation and Task Scheduling Optimization

Abstract

Abstract: With the rapid development of Internet of Things (IoT) and mobile terminals, the concept of edge computing arises. By moving the computation and storage capacity to the edge of network, edge computing is able to deal with a large amount of data produced by IoT devices and the responsive request from IoT application. To improve the utility of edge resource, the quality of service and quality of user experience, resource allocation and task scheduling optimization problems under edge computing attract wide attention. It becomes more difficult due to the geographic separated and heterogeneous features of edge computing resource as well as the requirements of performance, energy consumption, cost and stability and introduces the system models of edge computing, IoT and cloud computing, presents the optimization metrics, scheduling models and solution optimization, including exact algorithms, heuristic methods and intelligent optimization algorithms. In addition, typical application cases, and points out the further research contents and directions are provided to promote the development of edge computing.

Keywords

edge computing, resource allocation, scheduling, optimization, modeling and simulation

Recommended Citation

Wang Ling, Wu Chuge, Fan Wenhui. A Survey of Edge Computing Resource Allocation and Task Scheduling Optimization[J]. Journal of System Simulation, 2021, 33(3): 509-520.

第 33 卷第 3 期 系统仿真学报© Vol. 33 No. 3
2021 年 3 月 Journal of System Simulation Mar. 2021

王凌 清华大学自动化系长聘教授、博士生导师、学位委员会副主席，国家杰出青年科学基金获得者。中国仿真学会理事，中国仿真学会智能仿真优化与调度专委会副主任；中国人工智能学会自然计算与数字智能城市专委会副主任；国际期刊 IJAAC 主编、CSMS 执行主编；IEEE TEVC、SEC、ESWA 等 SCI 期刊副主编；控制理论与应用、控制与决策、系统工程与电子技术等期刊编委。

长期从事复杂系统建模控制与优化调度的研究与应用，主持国家杰出青年科学基金及面上项目、国家重点研发计划课题等 20 余项；已出版专著 5 部、译著 1 部，在 IEEE Trans 等刊物上发表 SCI 论文 200 余篇、WOS 引用一万余次、Google Scholar 引用两万余次；获国家自然科学二等奖 1 项；省部级科技奖励 6 项；自动化学报、控制理论与应用、控制与决策等期刊优秀论文奖。获国家杰出青年科学基金、北京市科技新星、教育部新世纪优秀人才、清华大学学术新人奖、中国自动化学会青年科学家奖、中国高被引作者等。

边缘计算资源分配与任务调度优化综述

王凌，吴楚格，范文慧
(清华大学 自动化系，北京 100084)

摘要：随着物联网和移动终端的迅速发展，边缘计算技术应运而生，通过将计算和存储配置在互联网边缘，处理物联网终端产生的大量数据，应对时延敏感型应用请求。为提高计算资源使用效率，优化性能指标，边缘计算资源分配与任务调度优化问题受到了广泛关注。边缘计算资源的地理分散性、异构性以及对性能、能耗、费用、稳定性等的需求，增加了优化调度的复杂性。通过介绍边缘计算和物联网、云计算协同的系统模型，给出优化的指标、调度模型及其求解算法，包括精确算法、启发式方法及智能优化方法等，归纳典型应用案例，指出有待进一步研究的内容和方向，有助于促进边缘计算的发展。

关键词：边缘计算；资源分配；调度；优化；建模与仿真
中图分类号：TP391 文献标志码：A 文章编号：1004-731X (2021) 03-0509-12
DOI: 10.16182/j.issn1004731x.joss.20-0584

A Survey of Edge Computing Resource Allocation and Task Scheduling Optimization

Wang Ling, *Wu Chuge*, *Fan Wenhui*
(Department of Automation, Tsinghua University, Beijing 100084, China)

Abstract: With the rapid development of Internet of Things (IoT) and mobile terminals, the concept of edge computing arises. By moving the computation and storage capacity to the edge of network, edge computing is able to deal with a large amount of data produced by IoT devices and the responsive request from IoT application. To improve the utility of edge resource, the quality of service and quality of user experience, resource allocation and task scheduling optimization problems under edge computing attract wide attention. It becomes more difficult due to the geographic separated and heterogeneous features of edge computing resource as well as the requirements of performance, energy consumption, cost and stability and introduces the system models of edge computing, IoT and cloud computing, presents the optimization metrics, scheduling models and solution optimization, including exact algorithms, heuristic

收稿日期：2020-08-10 修回日期：2020-09-03
基金项目：国家杰出青年基金(61525304)，国家自然科学基金(61873328)
第一作者：王凌(1972-)，男，博士，教授，研究方向为智能优化调度理论与方法等。E-mail：wangling@mail.tsinghua.edu.cn

methods and intelligent optimization algorithms. In addition, typical application cases, and points out the further research contents and directions are provided to promote the development of edge computing.

Keywords: edge computing; resource allocation; scheduling; optimization; modeling and simulation

引言

随着移动设备的更新换代，物联网快速发展，地理上分散分布的终端设备向计算平台提出了低时延、高带宽、数据私密性等需求，在云计算基础上文献[1]提出边缘计算的概念，通过将数据推到互联网边缘来提升系统整体的可用性和可拓展性；文献[2]提出雾计算的概念，通过将计算和存储设备配置在互联网边缘来减少互联网数据传输量，从而降低时延、节省带宽及相关费用。目前，边缘计算和雾计算引起了学术界和工业界的极大关注[3]。文献[4]介绍了雾计算的关键技术，如计算、通讯、存储、命名和私密性等，列举了常见的雾计算应用。文献[5]综述了雾计算的概念、框架及编程模型、设计及规划、资源管理、操作、软件及工具、硬件及协议等。多个国家联合成立了 OpenFog 联盟[6]。雾计算和边缘计算紧密相关，相比于雾计算，边缘计算更强调边缘设备间资源的相互协同，是处理云上游数据或物联网下游数据的技术[7]，而雾计算服务的实现需要边缘技术的支撑。

资源分配和任务调度优化是计算系统的重要研究问题之一，其解决方案影响资源使用的有效性和用户的服务体验[8]。鉴于边缘计算资源的异构性、处理器的地理分散性以及电池耗电量等优化需求，对资源分配和任务调度优化形成了新的挑战。文献[9]梳理了边缘计算场景下的优化问题，总结了 40 多项优化指标，并按优化场景分类综述相关研究。针对边缘计算资源配置问题，文献[10]综述了优化算法类型、优化目标和约束。基于上述工作，本文将介绍“物联网-边缘计算-云计算”系统架构及相关描述模型，总结边缘计算场景下资源分配调度问题中的常见优化指标、约束及典型算法，并按抽象调度模型对研究工作进行分类，概述问题的数学模型与求解方法，介绍典型应用，并指出进一步研究的内容，有助于研究人员根据场景、目标、调度问题等建立调度模型，设计优化算法，获得有效调度方案。

1 系统模型

云计算中心协同计算场景下的资源调度优化，如图 1“云-边-端”3 层计算系统[11]。

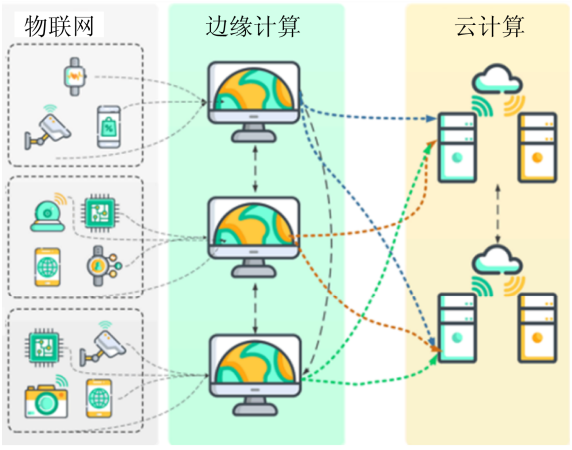

图 1 “云-边-端”计算系统结构
Fig. 1 Three-tier computing system architecture

DOI: 10.16182/j.issn1004731x.joss.20-0584

第 1 层是物联网层[12]，传感器、处理器根据应用需求感知、测量和收集原始数据，在本地处理大量数据或将其上传至计算节点。此外，物联网装置被划分为不同集合，相同集合的多个装置能够提高容错度。

第 2 层是边缘计算层，位于互联网边缘，靠近数据源。边缘计算节点连接物联网，边缘节点之间可以互相通信，提供计算和存储功能。文献[13]将该层分为抽象层和协同层。文献[14]将边缘节点分为计算和网关节点，其中计算节点包含控制、计算和通讯模块，负责接收和处理应用请求，网关节点包含期望评级和应用分配单元，负责评估应用请求的优先级并为应用分配处理节点。

第 3 层是云计算层。一般而言，云计算中心的计算和存储能力强于雾节点，提供高聚合度的集中计算、存储等服务，但云计算中心和本地的长距离往往会造成的传播和传输时延，导致数据型任务上传云节点时存在难以容忍的延时。

对于上述 3 层计算系统的架构模型，文献[15]综述了集成物联网+雾+云 C2F2T (Cloud to Fog to Things) 解析模型、Petri 网模型、整数规划模型等以及预测和分析系统的各项指标。文献[16]提出了解析模型，预测给定应用的计算资源需求和服务质量 QoS (quality of service)。文献[17]通过实例分析了物联网应用对边缘计算的需求，考察异构、结构分层以及大规模设施等特性，为边缘计算系统、服务质量、应用及操作建立解析模型。文献[18]建立了边缘计算节点的排队模型，给出各环节时延的解析模型，基于模型量化卸载策略对系统相关软硬件的影响，并观测系统参数变化。文献[19]利用 Petri 网模型评估其边缘计算架构的性能，并与标准移动众包架构进行对比。文献[20]利用 Petri 网模型评估数据的可追溯性。对系统的各组件及参数建模，可以辅助验证优化算法的可行性、有效性、优化性。

2 优化目标及约束

将用户、运营方等方面的需求建模为优化目标与约束，进而寻求问题的解决方案非常重要。文献[10]整理了 280 项雾计算下的优化指标，包括时延、能耗、收益与花费、设备相关和其他 5 大类。本节从性能、能耗、花费 3 个方面介绍边缘计算下优化目标和约束的建模与典型优化方法。

2.1 性能指标

2.1.1 时延

服务时延或反应时间，是指应用提交请求和收到回应间的耗时[18]。时延是边缘计算资源调度优化的重要指标。文献[10]将时延分为计算节点上耗时、节点间传输耗时及计算节点间迁移时间共 6 类。

针对边缘计算环境，文献[18]给出了时延解析模型，任务的服务时延由不同计算节点的处理时延按分配概率加权计算得到。文献[21]考虑架构中的网关数量以及缓冲区占用率等情况，定义了多个时延计算方程。除理论模型外，文献[22]对雾计算时延进行了实验统计分析，统计了不同应用、http 需求等在不同雾计算设备、不同环境下的时延，发现 II 型广义极值分布能较好地模拟时延，指出小型计算任务时延取决于通讯时间且不同环境对时延的影响最高可达 11 倍。此外，针对时延的稳定性实验表明，时延敏感型任务应将根据环境变化而调整分配方案。目前，以最小化时延为目标的算法很多，分布式优化策略、计算任务卸载、任务调度等方法或模型都旨在降低时延进而提升系统性能。

2.1.2 截止时间

除最小化时延外，任务的截止时间可表示任务的紧迫程度。实时系统[23]给出硬期限和软期限的定义。不同任务的时延敏感度不同，一些任务未能在期限前完成则会出现严重后果，于是被定义为硬期限约束

任务，否则为软期限约束任务。若完成时间大于期限，则存在拖期，即完成时间和期限之差。此外，时间相关效率函数也被用来描述任务完成时间的要求，譬如用阶跃函数刻画硬约束，用衰减指数函数作为延时函数，任务完成时间越晚则其完成效用越低。

文献[24]提出在线调度方法，最大化满足截止时间的任务个数。文献[25]将用户体验转化为任务截止时间并作为问题的约束，优化获得满足条件的可行解。文献[26]在定义适配值时对超过截止时间的应用添加惩罚项。文献[27]将工作流中的子任务按期限类型分为硬期限和软期限约束任务，分别考虑其合法性和拖期大小。文献[22]用概率分布函数模拟任务时延，对任务完成的时间函数与设定时间相关效率函数的积分作为任务完成效率，并最大化任务分配问题的任务完成效率。目前，根据任务特征及边缘计算场景需求选择合适的指标，设计特定优化算法，是截止时间相关优化问题的重点。

2.1.3 用户体验质量

相比于客观的时延等性能，QoE 的定义依赖于用户、环境和服务，较为主观。在国际电信联盟给出的定义基础上，文献[28]将 QoE 定义为用户在一定的客观环境中对所使用的服务或者业务对整体的认可程度。常见的 QoE 优化问题要求提高用户对音频质量、视频清晰度的满意程度等。随着移动端视频需求的增大，对 QoE 的量化、建模及优化的研究日益增多。边缘计算作为降低即时应用时延的解决方案之一，可提高用户满意度，通过部署雾节点提供雾计算服务，因而考虑 QoE 的边缘资源分配和任务调度策略可为用户提供更优质的服务。

为了优化用户玩交互游戏的体验，文献[29]综合反应时间、网络拥塞和服务覆盖度作为用户的 QoE，并设计 Cloud-Fog 系统，从而降低时延，提升终端用户的覆盖度。文献[30]利用 QoE 指标估计和指导下一步的资源分配。另外，文献[15]将 QoE 设为边缘计算资源配置算法的唯一优化目标，根据用户对应用的期待和边缘计算资源状态，匹配“特定应用-计算节点”，从而优化整体满意度。目前，如何与具体问题结合来量化 QoE 并设计优化算法，满足用户需求并提升满意度，是重点的研究工作。

2.2 能耗指标

能耗是数据中心的主要开销之一，包括计算机器、制冷散热设备耗电等。将计算设备移至网络边缘，保证资源有限的电池供电设备正常运行非常重要。文献[31]探讨了边缘计算是否能节省云计算能耗，讨论了下载、更新、预加载数据量等应用以及不同场景下用电量的变化。研究不同情况下的资源分配和任务调度对降低能耗很有意义。

2.2.1 电池电量

边缘计算环境下，移动终端设备的电池电量约束问题亟待研究，其中耗电量分为监测、计算、通讯和执行 4 部分[32]。监测耗电量与其数据包大小、时长相关；通讯分为上传和接收 2 部分，上传能耗与数据量及传输距离正相关，接收能耗与数据量正相关；计算能耗取决于具体硬件参数；执行能耗与具体执行应用和次数正相关。文献[33]在应用截止时间约束下，分配和排序任务最小化物联网设备总能耗。此外，文献[34]将系统的生存周期定义为所有传感器中最先耗尽电量的传感器的工作时长，进而延长系统生存周期成为一个优化目标。目前，如何有效的节约移动端电量，并维持系统稳定，是研究工作的重点。

2.2.2 其他能耗

文献[31]给出了路由器等多用户访问的网络设备和家用服务器等计算设备的能耗模型，其中网络设

https://dc-china-simulation.researchcommons.org/journal/vol33/iss3/1
DOI: 10.16182/j.issn1004731x.joss.20-0584

备主要依据数据流量计算能耗，服务器能耗则基于时间，同时指出微型服务器接入特定类型网络、活跃时间呈一定比例、下载活动占较大比例的应用等情形下边缘计算可节省云计算中心的能耗。

文献[35]将边缘计算节点能耗建模为关于计算量的单增严格凸函数，并用二次函数刻画其特性，用服务器开关状态以及处理器 CPU 频率的一次函数刻画云计算节点能耗。文献[36]将能耗分为空闲时和占用时能耗 2 类，可以通过对时间积分计算。目前，针对边缘计算和云计算节点的能耗优化，包括总能耗的优化、性能和能耗的多目标均衡优化。

2.3 花费指标

经济指标是用户选择资源以及运营商提供解决方案时需要考虑的重要因素。云计算的广泛应用一定程度上取决于其成功的“pay-as-you-go”商业模式。可见，服务的合理定价和对商业成本的优化是边缘计算亟待解决的重要问题。资源调度优化的花费主要集中于计算、存储和网络等资源成本。

文献[37]在考虑性能、虚拟机等约束下，将虚拟机的配置成本和数据通讯的花费之和作为优化目标，获得最优的用户基站选择、虚拟机-基站的配对等解决方案。文献[38]考虑城域网中的经济成本问题，将应用分为只需计算、只需存储以及存储与计算等类别，以计算节点计算、存储、链路容量和时延为约束，最小化支持网络拓扑结构下的经济成本，包括计算、存储、城域网带宽和核心计算节点上下游各项的成本。同时，参考谷歌云等的价格模型给出了不同计算节点的计算、存储的价格，价格差异将导致不同应用的资源选择方案不同。目前，如何将定价模型和实际应用结合，根据边缘计算特点定义花费和建立优化模型，是研究工作的重点。

3 调度优化模型及求解方法

本节针对“云-边-端”计算系统，介绍单一任务的卸载或分配决策、多任务分配及调度、边缘计算环境中的资源约束与和任务优先关系约束、相应的求解方法。体系结构如图 2 所示，其中决策变量为 $x_{i,j,r}$=1，表示任务 i 在计算节点 j 上第 r 个处理。

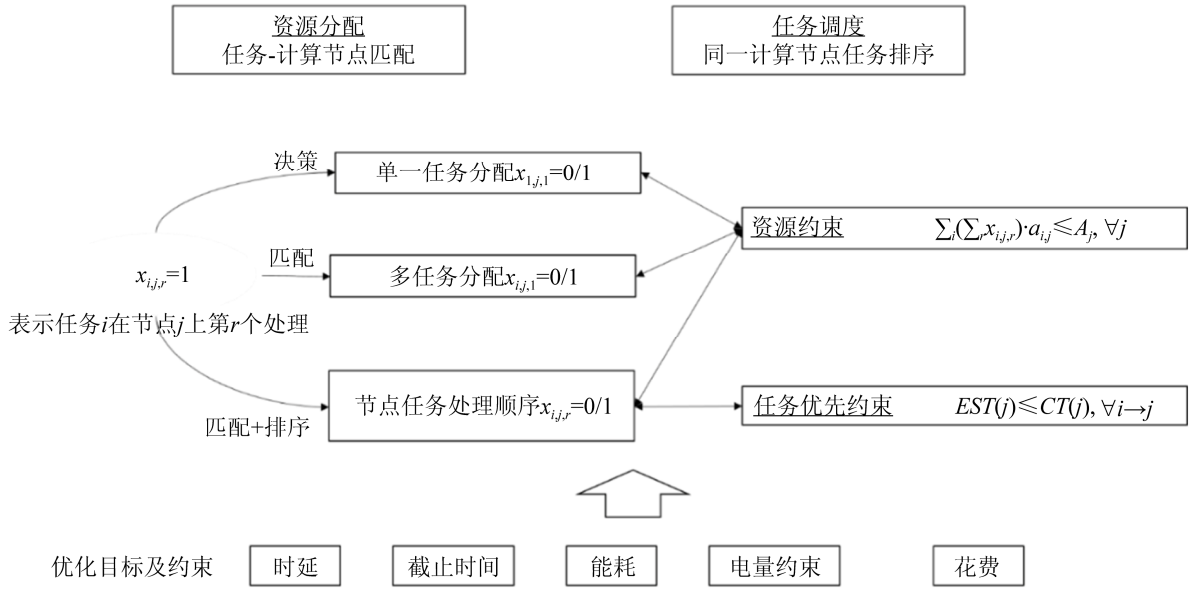

图 2 边缘计算资源分配与任务调度优化

Fig. 2 Edge computing resource allocation and task scheduling optimization

3.1 独立任务卸载/分配

3.1.1 问题与模型

边缘计算旨在减轻网络负载、分担终端计算任务，而独立任务的卸载决策则是基础。文献[39]探讨了移动终端的任务卸载决策问题，包括何时卸载、静止/动态、确定任务是否卸载、卸载收益等。文中指出卸载的目的是提高性能、节省能耗，并指出很有必要卸载高计算量和低数据量的任务。文献[40]在智能网关场景下，根据能量状态、数据类型和通讯频率进行卸载决策，同时介绍了物联网中间件支撑卸载决策的技术手段与常见的任务卸载场景。

独立任务的卸载或分配，可看作 $x_{1,j,1}$ 的 0/1 决策问题，如图 2 所示，即确定任务是否分配至节点 j。多个独立任务的卸载或分配，可看作 $x_{i,j,1}$ 的 0/1 决策问题，即任务 i 和节点 j 的匹配问题，以实现负载均衡。如果计算节点特定时间内最多只能处理一个任务时，可看作 $x_{i,j,r}$ 的 0/1 决策问题，即任务-节点匹配和节点上的任务排序问题。

排队模型、状态转移方程等解析模型可用于分析边缘节点任务卸载策略的有效性。文献[41]针对环形拓扑结构边缘节点间的负载均衡问题，建立了随机游走模型，分析了超负荷节点是否应向相邻节点卸载任务。文献[21]基于边缘节点和任务的数据、缓存状态、数据接收和计算能力等特点，建立了状态更新、网关和云计算节点排队模型，并提出了分布式算法优化整体时延。文献[42]将车辆视作边缘节点构建交通管理模型，利用排队论和马尔科夫转移方程对任务处理状态建模，进而决策节点间的计算转移。

3.1.2 启发式策略

智能终端设备具有决策能力，鉴于边缘计算环境的动态性、不确定性，研究边缘节点根据环境和自身状态的分布式在线决策算法十分必要。文献[43]对每个终端预先分别计算各项任务在边缘和云层的能耗，进而确定是否卸载任务。文献[18]提出了分布式卸载策略，边缘节点针对请求判断计算时延是否超过阈值，进而将请求加入队列，或获取周边节点状态选择最佳卸载对象，若无合适节点则上传云端处理。同时，文中提出解析模型分析了分布式算法的可行性和有效性。文献[44]讨论了全卸载、部分卸载等策略，利用改进交替方向乘子法同时优化用户的平均时延和能耗。

针对多任务分配问题，文献[24]分别给出了集中式和分布式在线优化算法，最大化满足截止期限的任务个数，其中分布式算法在接入节点和边缘节点实现。文献[15]对 QoE 建立模糊模型，给出了雾计算环境下最大化 QoE 的多任务分配方案。

3.2 资源约束下的任务分配

3.2.1 优化问题

由于计算系统的计算、存储、带宽、电池电量等资源有限，将有限资源作为约束给出调度优化模型，更符合现实。文献[45]提出了带 CPU 容量、电池电量约束的任务-节点分配模型。文献[11]将任务分配问题按约束分为网络、节点、能耗和应用 4 类，其中计算节点约束分为软件约束(如库、操作系统)和硬件约束(如内存、CPU、存储)。

求解资源约束下的任务分配问题即寻找满足约束的任务-计算节点分配方案。对于计算、内存、带宽有限的物理设备或虚拟机，可建立相应的多维背包[46]模型。

3.2.2 模型与算法

资源约束可表示为 $\Sigma_i(\Sigma_r x_{i,j,r})\cdot a_{i,j} \leqslant A_j$，如图 2 所示，其中 $\Sigma_r x_{i,j,r}=1$ 表示任务 i 分配至节点 j，$a_{i,j}$ 表示任

DOI: 10.16182/j.issn1004731x.joss.20-0584

务 i 占用的某类资源量，A_j 表示节点 j 该资源总量。当没有任务处理时间约束时，可建模为多个物品分配到若干多维背包的匹配问题。考虑边缘计算节点的容量，文献[22]给出容量有限情况的动态规划求解方法，通过计算任务分配收益、拒绝无法分配的任务，最大化整体处理效率。针对工业 4.0 场景下的任务-网关分配问题，文献[26]利用负载-风险模型定义任务分配的可靠性，提出相应的贪婪算法寻求同时满足时延和可靠性要求的方案。文献[47]建立混合整数规划模型，确定满足边缘节点 CPU、内存和存储容量约束的利用率最大化且的任务分配方案。文献[17]标记节点的硬软件参数，设计并实现了任务分配的回溯搜索算法。

对于一些复杂情况，资源约束下的任务分配问题仅是子问题之一。针对边缘节点的容量约束和云节点的 CPU 频率约束下的负载均衡问题，文献[35]将原始问题分解成边缘节点负载分配、云节点负载分配和最小化时延指派问题 3 个子问题，分别利用凸优化、Benders 分解和匈牙利算法求解。文献[48]约定存储服务器预存任务图像后才能处理相关的读写操作，即任务图像-存储器的匹配子问题，通过松弛 0–1 决策变量利用贪婪算法获得可行解。

3.2.3 智能算法

鉴于在诸多复杂问题上的优越性能，智能算法在本领域得到了推广应用。文献[49]采用分布式遗传算法(Genetic Algorithm，GA)求解 CPU、内存和带宽约束下混合云任务分配问题，利用有偏随机键和特定解码规则处理不可行解，最大化同时处理的任务数。文献[25]用 GA 求解任务-公交车的分配问题，在节点计算容量约束下寻求经济成本最低的方案。文献[50]提出了一种共生有机体搜索算法，在不同场景下向虚拟机分配任务，评价成本、网络占用和能耗指标。针对 CPU 算力约束下的边缘计算容器迁移问题，文献[51]将迁移策略建模为多维马尔科夫过程空间，并利用深度强化学习快速决策。

3.3 带优先约束任务调度/卸载

3.3.1 优化问题

为了加快计算进程，将分布式计算系统中的复杂任务分割为多个子任务形成任务流，进而分配到处理器上并行处理。由于部分任务的计算需要前序任务的结果数据，任务间存在优先约束关系，可利用有向无环图 DAG (directed acyclic graph)抽象和建模工作流，图节点代表子任务，节点间连线代表子任务间的优先约束关系。文献[52]给出了 DAG 调度问题的树形图，考虑任务间是否存在通讯时间、处理器资源是否有限、处理器是否完全连接等多种情况。

3.3.2 数学模型及相关算法

任务间优先约束表示为 $EST(j) \leqslant CT(i)$，$\forall i \rightarrow j$，如图 2 所示，其中 $EST(j)$表示任务 j 可开始处理的最早时间，$CT(i)$表示任务 i 的完成时间，$i \rightarrow j$ 表示任务 i 到 j 存在优先约束。

针对移动端卸载问题，设定处理器同一时间可以处理多个任务，文献[53]提出了基于动态规划的确定性时延约束任务分割算法，证明了其次优性。针对截止时间约束下的能耗最小化问题，不限制 DAG 的拓扑结构，但约定处理器在同一时间只能处理一个任务，文献[54]给出了边缘计算节点与远端云节点协同环境下的 DAG 混合整数规划模型。该问题包含具有 NP 难特性的一般性指派问题，目前没有多项式时间的最优化方法。文献[54]利用松弛整数规划模型中的 0–1 变量将问题转换为凸优化问题，进而设计启发式方法。针对多 DAG 的移动终端卸载问题，文献[55]提出了混合整数规划模型，对是否将任务上传云端进行决策，并在截止时间约束下优化能耗。

3.3.3 启发式方法

DAG 调度优化方法主要包括启发式方法、智能算法及混合算法[56]，其中启发式方法主要分为列表调度、聚类和任务复制 3 类。列表调度方法按优先级排列任务，然后从待调度的任务中选择最高优先级的任务分配给合适的处理器[57]；聚类方法将任务聚类，直到类别数与处理器数相等[58-59]。任务复制将传输数据量大的任务复制到多个处理器上处理，进而降低时延。

在“云-边-端”系统下，文献[60]将前向排名作为列表调度准则，同时优化时延以及云和边缘计算节点的计算与通讯成本。同时考虑截止时间和费用，文献[54]利用下界估计等方法为 DAG 的子任务分配截止时间并为任务分配计算节点。参照任务截止时间的顺序，文献[61]依据任务的最早完成时间插空分配虚拟机，进而优化整体时间性能。

3.3.4 智能算法

区别于启发式规则，智能算法力争全局的优化性能。文献[62]利用 GA 优化任务-边缘节点群的分配。文献[12]采用概率表征任务间的前后位置关系，在基于启发式方法的 DAG 预分割后，利用双变量相关的分布估计算法排序任务，同时优化应用整体完成时间和边缘节点能耗。文献[27]考虑任务截止时间信息，利用 EDA 优化总拖期。针对任务-节点分配问题，文献[63]利用粒子群算法优化成本和完成时间的加权目标。

针对边缘计算资源分配和任务调度问题，从任务卸载决策、资源约束下的任务分配、DAG 调度与分配的上述综述可见：一些建模工作比较系统，在剖析系统特性的基础上提出了精确方法，但算法复杂性高，占用计算资源多，不适用于大规模问题；大部分工作采用启发式策略分配资源和调度任务，尽管面对大规模任务和异构化的资源，这类方法设计方便、易于实现且不占用过多的计算资源，但难以取得很好的整体优化效果；智能算法适用于强约束、多目标的复杂问题，可拓展性强，但很难适用于分布式在线问题等实时性要求高的场景。

4 应用与仿真验证

边缘计算技术已在增强现实、图像识别、网站性能优化、智慧城市、车联网等诸多领域得到成功应用[64]。边缘计算资源分配和任务调度优化亦在相关领域发挥了巨大作用。为了对资源管理策略进行评价，边缘环境的实证分析十分必要。由于现实边缘计算环境搭建难度大、实体变更成本高，仿真已成为重要的工具。通过提供自定义接口，可重复评估实验，进而验证资源分配与任务调度优化的算法性能。

对边缘计算环境建模与实验运行的现有模拟器与工具包有 Edgecloudsim[65]，iFogSim[66]，DeFog[67]等。其中，DeFog 可建模边缘计算环境，搜集测试应用，譬如深度学习目标检测、语音识别、在线游戏等，标记测量并发数、成功率、平均时延和上传下载数据量等指标，辅助分析算法的有效性。文献[47，68]则利用 iFogSim 验证资源调度策略的有效性。

除了仿真验证，一些调度算法被用于真实边缘场景解决实际问题，表 1 列举了一些实际案例，包括制造业[26,36]、交通运输[25,42]、视频处理[20,55,68]等领域。当然，对于不同领域的应用，目标以及相应的建模与优化算法都有所不同。

对于任务卸载问题，算法主要决策任务的派发与卸载，譬如文献[68]将视频监控应用分为用户交互、目标探测和追踪等任务，算法主要解决任务分类、处理器类型匹配等问题。对于资源约束问题，文献[20，26]考虑带宽、CPU 等约束，文献[25]将完成时间和边缘节点存储作为约束，文献[69]则将实时功耗的上限作为约束。对于任务间无法忽视的优先约束，文献[17，55]将其设置为问题的一个维度进行调度优化。

DOI: 10.16182/j.issn1004731x.joss.20-0584

表 1 应用场景与调度问题
Tab. 1 Applications and scheduling problems

应用场景	调度问题
室内联网机器人[45] 糖果装配线[36] 城市交通车辆网络[42] 视频监控、目标追踪[68]	任务卸载
智能监控、老人行动探测[20] 工业 4.0 (智能网关)[26] 公交车网络[25] 智能家居[69]	资源约束下的任务分配
火情探测报警系统[17] 视频导航[55]	带优先约束的任务调度

5 结论

随着边缘计算的快速发展，资源分配和任务调度的研究面临诸多挑战，从 5 个方面进行讨论。

(1) 理论

在理论层面，面向复杂问题或应用场景，理论性成果很匮乏。譬如，研究截止时间的优化算法的可调度性、计算资源受限下调度算法的计算复杂度、分布式优化算法的收敛性。同时，针对简化模型的寻优算法的收敛性与鲁棒性、基于模型的问题下界，都是很有意义的理论研究内容。

另外，面对多目标优化问题，目标均衡策略、协同优化机制、决策空间与目标空间的分解策略、层次分析等方法与技术的研究、改进均有待相应的理论分析的支撑。

(2) 建模

在模型层面，大规模的边缘节点、有限的计算资源、不稳定的网络连接不稳定、异构性的平台，都给问题的建模带来挑战。针对大规模和异构等特点，可建立分布式模型；针对网络、计算资源、物联网应用的不稳定特性，可建立不确定模型；针对需求的实时性和动态性，可建立动态和重调度模型，进而设计相应的方法获得可行、有效、鲁棒的调度方案，保证满意的实时性、可扩展性、鲁棒性等。

(3) 方法

在方法层面，要同时注重基于数学模型和数据驱动的优化调度策略、算法研究。尽管精确求解方法因寻优效率低而不适用于大规模边缘计算节点与物联网应用，但问题的可调度性研究有理论意义。尽管启发式与贪婪搜索方法难以取得全局性能，但适用于对优化质量要求不高的实时边缘计算场景。智能算法是进一步研究的重点，但必须结合实际应用的需求兼顾计算效率和优化质量。

考虑到边缘计算环境的实时性与动态性、计算资源的有限性、缺少获取所有边缘节点状态的主节点等情况，分布式优化方法的研究值得关注与发展。另外，随着人工智能技术的发展，将优化方法与深度学习、强化学习等有机结合，利用已有调度案例数据实现模型的线下训练，进而再开展线上优化与决策，是当前研究的热点方向。

(4) 验证

理论分析、性能对比、仿真验证，都是调度优化算法验证层面可开展的工作。譬如，利用排队模型或状态转移模型建立边缘计算节点任务处理状态的模型并开展性能分析，利用 Petri 网模型评估系统的性能和可靠性，利用数学规划模型获得性能的界作为寻优算法的对比基准。

另外，搭建原型系统验证所提算法的可用性、有效性、高效性，是实际应用的必要前提，尤其是智能战场等复杂应用对边缘计算侧仿真提出了很高的实际需求[70]。只有通过面向实际系统的验证，才能为算法的可行性、有效性、稳定性提供充分说服力的依据。

(5) 推广与应用

边缘计算调度优化理论与方法的研究，是技术实现与应用的前提。通过实际系统的应用，一方面可验证理论与算法的有效性，另一方面可发现新的问题与需求，通过推广应用使得理论与方法的研究得到进一步完善与发展。先进理论成果在实际工业、生活场景下验证和应用，有利于加深对调度优化问题的理解，有效算法的成功应用则将促进边缘计算技术的落地与发展。

总之，边缘计算技术方兴未艾，资源分配和任务调度作为核心问题许多研究工作亟待深入，促进理论、方法、技术与应用的发展以及相关学科的交叉融合。

参考文献：

[1] Pang H H, Tan K L. Authenticating Query Results in Edge Computing[C]//20th International Conference on Data Engineering. Boston, MA, USA: IEEE Comput Soc, 2004: 560-571.
[2] Bonomi F, Milito R, Zhu J, et al. Fog Computing and its Role in the Internet of Things[C]//1st Edition MCC Workshop Mobile Cloud Comput. NY, USA: ACM, 2012: 13-16.
[3] Vaquero L M, Rodero-Merino L. Finding Your Way in the Fog: Towards a Comprehensive Definition of Fog Computing[J]. ACM SIGCOMM Computer Communication Review (S0146-4833), 2014, 44(5): 27-32.
[4] Hu P, Dhelim S, Ning H, et al. Survey on Fog Computing: Architecture, Key Technologies, Applications and Open Issues[J]. Journal of Network and Computer Applications (S1084-8045), 2017, 98(11): 27-42.
[5] Yousefpour A, Fung C, Nguyen T, et al. All One Needs to Know About Fog Computing and Related Edge Computing Paradigms: A Complete Survey[J]. Journal of Systems Architecture (S1383-7621), 2019, 98(1): 289-330.
[6] O. C. Architecture Working Group, Open Fog Reference Architecture for Fog Computing[R]. 2017, 1(1): 162.
[7] Shi W, Cao J, Zhang Q, et al. Edge Computing: Vision and Challenges[J]. IEEE Internet of Things Journal (S2327-4662), 2016, 3(5): 637-646.
[8] Hussain H, Malik S U R, Hameed A, et al. A Survey on Resource Allocation in High Performance Distributed Computing Systems[J]. Parallel Computing (S0167-8191), 2013, 39(11): 709-736.
[9] Bellendorf J, Mann Z Á. Classification of Optimization Problems in Fog Computing[J]. Future Generation Computer Systems (S0167-739X), 2020, 107(1): 158-176.
[10] Brogi A, Forti S, Guerrero C, et al. How to Place Your Apps in the Fog: State of the Art and Open Challenges[J]. Software: Practice and Experience (S0167-739X), 2019, 1(1): 1-8.
[11] Wu C, Li W, Wang L, et al. Hybrid Evolutionary Scheduling for Energy-efficient Fog-enhanced Internet of Things[J]. IEEE Transactions on Cloud Computing (S2168-7161), 2018, 1(1): 1-1.
[12] Atzori L, Iera A, Morabito G. The Internet of Things: A Survey[J]. Computer Networks (S1389-1286), 2010, 54(15): 2787-2805.
[13] Bonomi F, Milito R, Natarajan P, et al. Fog Computing: A Platform for Internet of Things and Analytics. N. Bessis, C. Dobre. Big Data and Internet of Things: A roadmap for smart environments[M]. Cham: Springer, 2014, 546: 169-186.
[14] Mahmud R, Srirama S N, Ramamohanarao K, et al. Quality of Experience (QoE)-aware Placement of Applications in Fog Computing Environments[J]. Journal of Parallel and Distributed Computing (S0743-7315), 2019, 132(1): 190-203.
[15] Buyya R., Srirama S. N. 雾计算与边缘计算：原理和范式[M]. NJ, USA: Wiley, 2019: 31-50.
[16] Wu H, Yue K, Hsu C, et al. Deviation-based Neighborhood Model for Context-aware QoS Prediction of Cloud and IoT Services[J]. Future Generation Computer Systems (S0167-739X), 2017, 76(1): 550-560.
[17] Brogi A, Forti S. QoS-Aware Deployment of IoT Applications Through the Fog[J]. IEEE Internet of Things Journal (S2327-4662), 2017, 4(5): 1185-1192.
[18] Yousefpour A, Ishigaki G, Gour R, et al. On Reducing IoT Service Delay Via Fog Offloading[J]. IEEE Internet of Things Journal (S2327-4662), 2018, 5(2): 998-1010.
[19] Merlino G, Arkoulis S, Distefano S, et al. Mobile Crowdsensing as a Service[J]. Future Generation Computer Systems (S0167-739X), 2016, 56(1): 623-639.
[20] Lomotey R K, Pry J, Sriramoju S. Wearable IoT Data Stream Traceability in a Distributed Health Information System[J].

https://dc-china-simulation.researchcommons.org/journal/vol33/iss3/1
DOI: 10.16182/j.issn1004731x.joss.20-0584

Pervasive and Mobile Computing (S1574-1192), 2017, 40(1): 692-707.
[21] Desikan K E S, Srinivasan M, Murthy C S R. A Novel Distributed Latency-aware Data Processing in Fog Computing-enabled IoT Networks[C]//Proc. of the ACM Workshop on Distributed Information Processing in Wireless Networks. Chennai, India: ACM, 2017: 1-6.
[22] Gorlatova M, Inaltekin H, Chiang M. Characterizing Task Completion Latencies in Fog Computing[J].Computer Networks (S1389-1286), 2020, 181:107526.
[23] Abbott R K, Garcia-Molina H. Scheduling Real-time Transactions: A Performance Evaluation[J]. ACM Transactions on Database Systems (S0362-5915), 1992, 17(3): 513-560.
[24] Meng J, Tan H, Li X Y, et al. Online Deadline-aware Task Dispatching and Scheduling in Edge Computing[J]. IEEE Transactions on Parallel and Distributed Systems (S1045-9219), 2020, 31(6): 1270-1286.
[25] Ye D, Wu M, Tang S, et al. Scalable Fog Computing with Service Offloading in Bus Networks[C]//3rd International Conference on Cyber Security and Cloud Computing. Beijing, China: IEEE, 2016: 247-251.
[26] Verba N, Chao K M, Lewandowski J, et al. Modeling Industry 4.0 based Fog Computing Environments for Application Analysis and Deployment[J]. Future Generation Computer Systems (S0167-739X), 2019, 91(1): 48-60.
[27] Wu C, Wang L. A Deadline-aware Estimation of Distribution Algorithm for Resource Scheduling in Fog Computing Systems[C]//IEEE Congress on Evolutionary Computation. Wellington, New Zealand: IEEE, 2019: 660-666.
[28] 林闯, 胡杰, 孔祥震. 用户体验质量(QoE)的模型与评价方法综述[J]. 计算机学报, 2012, 35(1): 1-15.
Lin Chuang, Hu Jie, Kong Xiangzhen. Survey on Models and Evaluation of Quality of Experience[J]. Chinese Journal of Computers, 2012, 35(1): 1-15.
[29] Lin Y, Shen H. Cloud Fog: Towards high quality of experience in cloud gaming[C]//44th International Conference on Parallel Processing. Beijing, China: IEEE, 2015: 500-509.
[30] Aazam M, St-Hilaire M, Lung C, et al. MeFoRE: QoE based Resource Estimation at Fog to Enhance QoS in IoT[C]//23rd International Conference on Telecommunications. Thessaloniki, Greece: IEEE, 2016: 1-5.
[31] Jalali F, Hinton K, Ayre R, et al. Fog Computing May Help to Save Energy in Cloud Computing[J]. IEEE Journal on Selected Areas in Communications (S0733-8716), 2016, 34(5): 1728-1739.
[32] Halgamuge M N, Zukerman M, Ramamohanarao K, et al. An Estimation of Sensor Energy Consumption[J]. Progress in Electromagnetics Research B (S1559-8985), 2009, 12(12): 259-295.
[33] Li W, Delicato F C, Zomaya A Y. Adaptive Energy-efficient Scheduling for Hierarchical Wireless Sensor Networks[J]. ACM Transactions on Sensor Networks (S1550-4859), 2013, 9(3): 1-34.
[34] Dietrich I, Dressler F. On the Lifetime of Wireless Sensor Networks[J]. ACM Transactions on Sensor Networks (S1550-4859), 2009, 5(1): 1-39.
[35] Deng R, Lu R, Lai C, et al. Optimal Workload Allocation in Fog-cloud Computing Towards Balanced Delay and Power Consumption[J]. IEEE Internet of Things Journal (S2327-4662), 2016, 1(1): 1-1.
[36] Wan J, Chen B, Wang S, et al. Fog Computing for Energy-aware Load Balancing and Scheduling in Smart Factory[J]. IEEE Transactions on Industrial Informatics (S1551-3203), 2018, 14(10): 4548-4556.
[37] Gu L, Zeng D, Guo S, et al. Cost Efficient Resource Management in Fog Computing Supported Medical Cyber-Physical System[J]. IEEE Transactions on Emerging Topics in Computing (S2168-6750), 2017, 5(1): 108-119.
[38] Sturzinger E, Tornatore M, Mukherjee B. Application- aware Resource Provisioning in a Heterogeneous Internet of Things[C]//International Conference on Optical Network Design and Modeling. Budapest: IEEE, 2017: 1-6.
[39] Kumar K, Liu J, Lu Y H, et al. A Survey of Computation Offloading for Mobile Systems[J]. Mobile Networks and Applications (S1383-469X), 2013, 18(1): 129-140.
[40] Aazam M, Zeadally S, Harras K A. Offloading in Fog Computing for IoT: Review, Enabling Technologies, and Research Opportunities[J]. Future Generation Computer Systems (S0167-739X), 2018, 87(1): 278-289.
[41] Fricker C, Guillemin F, Robert P, et al. Analysis of an Offloading Scheme for Data Centers in the Framework of Fog Computing[J]. ACM Transactions on Modeling and Performance Evaluation of Computing Systems(S2376-3639), 2016, 1(4): 16-34.
[42] Wang X, Ning Z, Wang L. Offloading in Internet of Vehicles: A Fog-enabled Real-time Traffic Management System[J]. IEEE Transactions on Industrial Informatics (S1551-3203), 2018, 14(10): 4568-4578.
[43] Zhao X, Zhao L, Liang K. An Energy Consumption Oriented Offloading Algorithm for Fog Computing[C]//12th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness. Cham: Springer, 2016: 293-301.
[44] Xiao Y, Krunz M. QoE and Power Efficiency Tradeoff for Fog Computing Networks with Fog Node Cooperation[C]// IEEE Conference on Computer Communications. Atlanta, USA: IEEE, 2017: 1-9.
[45] Kattepur A, Dohare H, Mushunuri V, et al. Resource Constrained Offloading in Fog Computing[C]//1st Workshop on Middleware for Edge Clouds & Cloudlets. Trento, Italy: ACM, 2016: 1-6.

[46] Pisinger D. Where Are the Hard Knapsack Problems[J]. Computers & Operations Research (S1873-765X), 2005, 32(9): 2271-2284.
[47] Skarlat O, Nardelli M, Schulte S, et al. Towards QoS-aware Fog Service Placement[C]//1st International Conference on Fog and Edge Computing. Madrid, Spain: IEEE, 2017: 89-96.
[48] Zeng D, Gu L, Guo S, et al. Joint Optimization of Task Scheduling and Image Placement in Fog Computing Supported Software-Defined Embedded System[J]. IEEE Transactions on Computers (S0018-9340), 2016, 65(12): 3702-3712.
[49] Mennes R, Spinnewyn B, Latre S, et al. GRECO: A Distributed Genetic Algorithm for Reliable Application Placement in Hybrid Clouds[C]//5thInternational Conference on Cloud Networking. Pisa, Italy: IEEE, 2016: 14-20.
[50] Rahbari D, Nickray M. Scheduling of Fog Networks with Optimized Knapsack by Symbiotic Organisms Search[C]// 21stConference of Open Innovations Association. Helsinki: IEEE, 2017: 278-283.
[51] Tang Z, Zhou X, Zhang F, et al. Migration Modeling and Learning Algorithms for Containers in Fog Computing[J]. IEEE Transactions on Services Computing (S1939-1374), 2019, 12(5): 712-725.
[52] Kwok Y, Ahmad I. Static Scheduling Algorithms for Allocating Directed Task Graphs to Multiprocessors[J]. ACM Computing Surveys (S1557-7341), 1999, 31(4): 406-471.
[53] Kao Y, Krishnamachari B. Optimizing Mobile Computational Offloading with Delay Constraints[C]// IEEE Global Communications Conference. TX, Austin: IEEE, 2014: 2289-2294.
[54] Sundar S, Liang B. Offloading Dependent Tasks with Communication Delay and Deadline Constraint[C]//IEEE Conference on Computer Communications. Honolulu, HI: IEEE, 2018: 37-45.
[55] Mahmoodi S E, Uma R N, Subbalakshmi K P. Optimal Joint Scheduling and Cloud Offloading for Mobile Applications[J]. IEEE Transactions on Cloud Computing (S2168-7161), 2019, 7(2): 301-313.
[56] Selvi S, Manimegalai D. DAG Scheduling in Heterogeneous Computing and Grid Environments Using Variable Neighborhood Search Algorithm[J]. Applied Artificial Intelligence (S0883-9514), 2017, 31(2): 134-173.
[57] Topcuoglu H, Hariri S, Min-You Wu. Performance-effective and Low-complexity Task Scheduling for Heterogeneous Computing[J]. IEEE Transactions on Parallel & Distributed Systems (S1045-9219), 2002, 13(3): 260-274.
[58] Liou J, Palis M A. A Comparison of General Approaches to Multiprocessor Scheduling[C]//11th International Parallel Processing Symposium. Switzerland: IEEE, 1997: 152-156.
[59] Xu Y, Li K, He L, et al. A DAG Scheduling Scheme on Heterogeneous Computing Systems Using Double Molecular Structure-Based Chemical Reaction Optimization[J]. Journal of Parallel and Distributed Computing (S0743-7315), 2013, 73(9): 1306-1322.
[60] Pham X Q, Huh E N. Towards Task Scheduling in a Cloud-fog Computing System[C]//18th Asia-Pacific Network Operations and Management Symposium. Kanazawa, Japan: IEEE, 2016: 1-4.
[61] Stavrinides G L, Karatza H D. A Hybrid Approach to Scheduling Real-time IoT Workflows in Fog and Cloud Environments[J]. Multimedia Tools and Applications (S1380-7501), 2019, 78(17): 24639-24655.
[62] Skarlat O, Nardelli M, Schulte S, et al. Optimized IoT Service Placement in the Fog[J]. Service Oriented Computing and Applications (S1863-2386), 2017, 11(4): 427-443.
[63] Xie Y, Zhu Y, Wang Y, et al. A Novel Directional and Non-local-convergent Particle Swarm Optimization based Workflow Scheduling in Cloud-edge Environment[J]. Future Generation Computer Systems (S0167-739X), 2019, 97(1): 361-378.
[64] 赵梓铭, 刘芳, 蔡志平, 等. 边缘计算: 平台、应用与挑战[J], 计算机研究与发展, 2018, 55(2): 327-337.
Zhao Ziming, Liu Fang, Cai Zhiping, et al. Edge Computing: Platforms, Applications and Challenges[J]. Journal of Computer Research and Development, 2018, 55(2): 327-337.
[65] Sonmez C, Ozgovde A, Ersoy C, et al. EdgeCloudSim: An Environment for Performance Evaluation of Edge Computing Systems[J]. Transactions on Emerging Telecommunications Technologies (S2161-3915), 2018, 29(11): 1-1.
[66] Gupta H, Dastjerdi A V, Ghosh S K, et al. iFogSim: A Toolkit for Modeling and Simulation of Resource Management Techniques in the Internet of Things, Edge and Fog computing environments[J]. Software - Practice and Experience (S0038-0644), 2017, 47(9): 1275-1296.
[67] Jonathan M, Nan W, Ashish T, et al. DeFog: fog Computing Benchmarks[C]//Proc 4th ACM/IEEE Symposium on Edge Computing. New York, USA: ACM Press, 2019: 47-58.
[68] Sharma S, Saini H. A Novel Four-tier Architecture for Delay Aware Scheduling and Load Balancing in Fog Environment[J]. Sustainable Computing: Informatics and Systems (S2210-5379), 2019, 1(1): 1-8.
[69] Xia C, Li W, Chang X, et al. Edge-based Energy Management for Smart Homes[C]//16th Dependable, Autonomic and Secure Computing. Athens, Greece: IEEE, 2018: 849-856.
[70] 翟岩龙, 孙文心, 包天虹, 等. 基于微服务的边缘侧仿真方法及框架研究[J]. 系统仿真学报, 2018, 30(12): 44-53.
Zhai Yanlong, Sun Wenxin, Bao Tianhong, et al. Edge-side Simulation Method and Framework Based on Micro-services[J]. Journal of System Simulation, 2018, 30(12): 44-53.

https://dc-china-simulation.researchcommons.org/journal/vol33/iss3/1
DOI: 10.16182/j.issn1004731x.joss.20-0584