

## 基于综合信任的边缘计算资源协同研究

邓晓衡<sup>1</sup> 关培源<sup>1</sup> 万志文<sup>1</sup> 刘恩陆<sup>1</sup> 罗杰<sup>1</sup> 赵智慧<sup>2</sup> 刘亚军<sup>1</sup> 张洪刚<sup>3</sup>

<sup>1</sup>(中南大学信息科学与工程学院 长沙 410075)

<sup>2</sup>(中南大学软件学院 长沙 410075)

<sup>3</sup>(马萨诸塞大学波士顿分校工程系 波士顿 02125-3393)  
(dxh@csu.edu.cn)

## Integrated Trust Based Resource Cooperation in Edge Computing

Deng Xiaoheng<sup>1</sup>, Guan Peiyuan<sup>1</sup>, Wan Zhiwen<sup>1</sup>, Liu Enlu<sup>1</sup>, Luo Jie<sup>1</sup>, Zhao Zhihui<sup>2</sup>, Liu Yajun<sup>1</sup>, and Zhang Honggang<sup>3</sup>

<sup>1</sup>(School of Information Science & Engineering, Central South University, Changsha 410075)

<sup>2</sup>(School of Software, Central South University, Changsha 410075)

<sup>3</sup>(Engineering Department, University of Massachusetts Boston, Boston 02125-3393)

**Abstract** Edge computing, as a new computing paradigm, is designed to share resources of edge devices, such as CPU computing ability, bandwidth, storage capacity and so on, to meet the requirements of the real-time response, privacy and security, computing autonomy. With the development of Internet of things (IoT) and mobile Internet technology, edge computing is of great potential of being widely used. This paper investigates the basic features, concepts and definitions, the latest state of art, and the challenge and trends of edge computing. Based on the key challenge of guarantee of users' quality of experience (QoE), privacy and security in edge computing, we focus on the requirement of users and consider the quality of experience of users to optimize the edge computing system. We integrate three aspects of trust properties, which are identity trust, behavior trust and ability trust, to evaluate resources and users to ensure the success of resource sharing and collaborative optimization in edge computing. This paper also investigates various computing modes such as cloud computing, P2P computing, C/S and grid computing, and constructs a multi-layer, self-adaptive, uniform computing model to dynamically match different application scenarios. This model has four contributions: 1) reveal the mechanism of parameters mapping between quality of service (QoS) and quality of experience; 2) construct identity trust, behavior trust of resources and users evaluation mechanisms; 3) form an integrated trust evaluation architecture and model; 4) design a resource scheduling algorithm for stream processing scenario, considering the computing ability, storage capacity and dynamical channel capacity depends on mobility to improve the quality of experience of users. Through this model and mechanism, resources in the end point, edge network, cloud center three levels are expected to be trusted sharing and optimized using, and the users' QoE needs are well satisfied. At last, simulation results show the validity of the model.

**Key words** distributed computing; edge computing; fog computing; mobile edge computing; Internet of things; integrated trust

收稿日期:2017-10-23;修回日期:2017-12-11

基金项目:国家自然科学基金项目(61772553, 61379058)

This work was supported by the National Natural Science Foundation of China (61772553, 61379058).

**摘要** 边缘计算旨在共享利用边缘设备的计算、通信资源,满足人们对服务的实时响应、隐私与安全、计算自主性等需求,随物联网的发展将有广阔应用前景.调研了边缘计算的特征、基本概念和定义、最新研究进展以及边缘计算研究的挑战与发展趋势.基于边缘计算的核心挑战之一——隐私信任与安全保障问题,针对用户应用需求特征,充分考虑用户体验质量(QoE)来优化边缘计算系统.通过集成用户和资源的身份信任、行为信任、能力信任3个方面为综合信任度,利用信任评估保障对边缘计算资源管理与协同优化.针对终端的动态性、边缘设施能力受限、边缘与终端的邻近性、云中心功能强和距离远的特征,融合云计算、P2P计算、C/S与网格计算模式,构建多层自适应的统一计算模型,实现对应用场景动态匹配;研究以用户体验质量为目标的综合资源/用户信任评估体系与模型,实现资源QoS向QoE的指标映射,构建资源和用户的身份信任、行为信任评价机制,形成综合信任评估体系与模型;根据应用需求,研究面向计算能力、移动性与可用服务时间、剩余能量、带宽等多重约束的边缘计算的任务卸载、资源调度算法和优化方案,实现资源在终端、边缘、云中心3层级可信共享和优化利用,更好满足用户QoE需求.最后通过流计算任务分配的边缘计算场景验证了模型框架的有效性.

**关键词** 分布式计算;边缘计算;雾计算;移动边缘计算;物联网;综合信任

**中图法分类号** TP393

## 1 计算模式的演进与资源协同

计算的硬件形态、计算体的连接形式与应用的需求发展相互影响,使得计算范式经历了从大型机+终端的集中计算模式;到服务器+PC+计算机网络的自主网络计算模式,其表现为集群计算、网格计算,期间还有对等计算等形式;进一步的发展使计算再次回归集中,形成云平台、数据中心+终端的中心计算模式,表现为云计算与数据中心服务<sup>[1-2]</sup>.云计算通过将ICT资源或信息服务的基础设施中心、平台和应用服务器等虚拟化直接提供基础设施服务,如各种服务器、数据库、存储设备、并行分布式计算系统等;或者形成平台,如运营、支撑和开发平台等;或者构建服务应用,如提供的软件、数据和信息等各种应用.云计算的核心思想是通过将超大容量的资源集中统一管理,根据用户的需求与请求,动态地分配,由于集中的管理模式有良好商业模式,包括亚马逊、Google、微软等IT、互联网和通信等行业大企业纷纷进入云计算领域,云计算也因此蓬勃发展,应用和服务于各行各业<sup>[3-4]</sup>.云计算高度发展,完全集中化,也带来不可避免的局限性.比如电子商务平台、评级服务、搜索引擎、在线社交网络和位置服务等集中式服务释放个人和社交数据而导致隐私泄露;从用户到云完全授权的应用程序和系统控制,要求客户端到云的单方面信任,阻碍了用户之间建立更细粒度的信任;同时,还失去和浪费了现代个人设备的计算、通信和存储能力;最后,云计算亦无法满足强实

时性需求、大数据量交互与处理、本地化位置相关计算的需求.边缘计算(edge computing)应运而生,代表了云计算之后回归的新的计算范例,其中计算和存储资源放置在互联网的边缘,靠近移动设备、传感器和最终用户.边缘计算不是要替代云计算,而是与云计算有机融合,进行权衡,使用户获得最好的体验.边缘计算又被称为或者表现为雾计算(fog computing)、移动边缘计算(mobile edge computing)等<sup>[5-7]</sup>.研究者使用诸如“朵云”、“微数据中心”和“雾”的术语来指代小的边缘定位节点<sup>[8-10]</sup>,代表着和云计算、大数据中心概念体系中相对应的另一面.

从计算模式的演进,我们可以发现其中有一个基本的因素就是模式演化都是为了更加充分、有效地协同各种资源,包括计算、存储、通信资源等.物联网、移动互联网的不断发展,意味着网络的边缘有海量的数据产生和巨大边缘就近计算的需求,产生大量的边缘流量可能对核心网络带宽产生挑战.卡内基梅隆大学计算机系Satyanarayanan教授指出从发展的成功程度来看,云计算正在让位于边缘计算,未来边缘计算应用以及终端用户的体验都将不断提升<sup>[11]</sup>.边缘计算将有着广泛的应用前景,业内甚至将边缘计算和云计算看作未来2种并存、对等的计算模式.学术界的研究者们已经针对这种计算模式的体系结构、关键理论与科学问题、技术挑战等问题进行探索,并形成许多初创成果.为推动边缘计算的深入研究<sup>[12-13]</sup>,美国自然科学基金会于2016年首次以边缘计算为主题设立研究计划资助科学家研究<sup>[14]</sup>;Future Generation Computer Systems, IEEE

Networks 系列的边缘计算专辑与综述论文也于 2016 年、2017 年出版,IEEE 和 ACM 于 2016 年联合发起召开了首届边缘计算研讨会. 2016 年 11 月 30 日边缘计算产业联盟由华为技术有限公司、中国科学院沈阳自动化研究所、中国信息通信研究院、Intel、ARM、软通动力信息技术(集团)有限公司在北京宣布成立<sup>[15]</sup>;工业界 Cisco 公司提出了“Fog Computing”<sup>[16]</sup>,其理念和边缘计算完全切合,其更关注硬件设施的部署和技术上解决方案的推广;当前通信领域的 5G 标准正在制定与形成中,意味着边缘网络的通信速度将进一步大幅提升,边缘计算的解决方案和相关标准也将成为未来 5G 标准的重要组成部分. 未来边缘计算的研究和应用正处一个即将兴起阶段,我国的学术和工业界也紧盯研究前沿,推动边缘计算的科研和产业创新与发展,为未来我国能制胜于物联网、移动互联网和云计算融合的时代奠定坚实基础.

边缘计算注重广泛密集分布资源受限的终端、局部汇聚具有较强资源能力的边缘服务器、资源超级巨大云计算、数据中心服务器通过多种异构无线/有线网络连接,实现安全、可靠、高效率的资源共享与管理,达到资源的优化利用,同时满足用户的高 QoE 体验以及可信安全的需求. 要真正充分发挥边缘设施和终端节点的效能,资源的科学管理和调度是核心,用户的信任和安全是保障. 这需要设计以用户为中心的面向用户体验质量(quality of experience, QoE)、安全信任的边缘计算模型框架,实现高效安全的资源管理、负载合理地分配,实现数据合理快速地移动、服务友好地加载和迁移,实现设备与用户有机地协同. 这些是边缘计算能否和云计算有机融合,协同形成一个高效、安全可信的计算系统,能否有效满足用户需求而被广泛实际应用的关键因素和根本条件.

## 2 边缘计算的概念

### 2.1 边缘计算有关定义

1) 边缘计算. 虽然边缘计算的计算形式已经在一些实际应用中表现出来了,但是对于边缘计算基础理论的研究正在逐步开展,产业界也还正在推出不同形式的解决方案,尚未形成统一标准. 边缘计算至今也还没有一个统一的标准定义,现在有不少学者发表了关于边缘计算的看法.

美国韦恩州立大学施巍松教授认为,边缘计算是指网络边缘执行计算的一种新型计算模型,边缘

计算中边缘的下行数据表示云服务,上行数据表示万物互联服务,而边缘计算的边缘是指从数据源到云计算中心路径之间的任意计算和网络资源<sup>[17]</sup>. 如图 1 所示,从数据生产与消费的角度将云计算和边缘计算对等地对待,边缘设备可以根据实际需要进行数据存储、缓存、分析、传输等相关计算操作,从而实现边缘和云的资源统一协调.

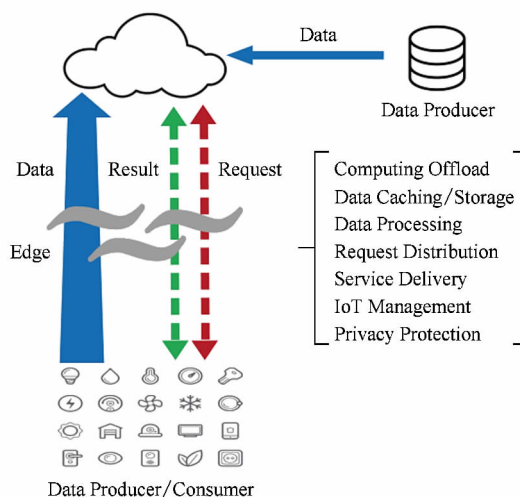


Fig. 1 Weisong Shi's edge computing paradigm

图 1 施巍松教授提出的边缘计算模型

美国卡内基梅隆大学的 Satyanarayanan 教授<sup>[8]</sup>认为边缘计算是一种新的范式,通过将小型的计算和数据中心的资源放置在网络的边缘,使得资源接近移动设备、传感器和最终用户,从而满足用户实时响应、安全隐私、高带宽等需求,如图 2 所示. 并通过 Cloudlet<sup>[8,10,18-19]</sup>作为一种示例形式,展示了一种 2 层云与微云架构的边缘计算模型,并通过包括利用谷歌眼镜在内视频应用展示出良好的应用效果.

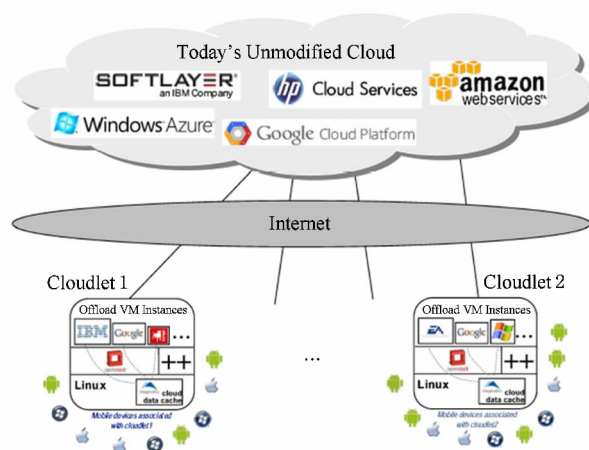


Fig. 2 Cloud and cloudlet infrastructure with two layers

图 2 2 层结构的云和微云结构

Lopez 等认为<sup>[12]</sup>以边缘为中心的计算作为一种新的范式,将计算应用、数据和服务的前沿从集中式节点推向网络边缘,即边缘中心计算.如图3所示,展示了中心计算的云计算和边缘计算的不同模式.

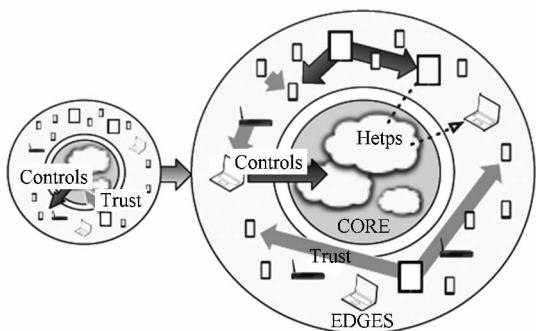


Fig. 3 Comparison between central and edge computing  
图3 中心计算与边缘计算模式对比

2) 移动边缘计算. 移动承载网络当前功能仅提供语音、数据通道,应用的业务占比却不断增加,电信标准组织和运营商拟通过未来5G网络将移动互联网及物联网业务融合,提供一体化服务提升移动网络带宽的价值.因此,欧洲电信标准协会 ETSI 提出移动边缘计算<sup>[20-21]</sup> (multi-access edge computing, MEC),即在“在移动网边缘提供 IT 服务环境和云

计算能力”<sup>[22]</sup>,将资源靠近移动用户,以减少网络操作和服务交付的时延,提高用户体验.从而实现了基于移动5G网络,将移动接入网与互联网业务有机融合,提供第三方应用集成,为移动边缘入口的服务创新提供平台,拓展移动通信运营商的市场领域,有效应对各种 OTT (over the top) 应用,被看成了电信运营商应用创新的基石.未来5G网络采用的接入方式,如 C-RAN (centralized/cloud radio access network) 或者 D-RAN (distributed radio access network) 都会将移动边缘计算模式作为5G标准组成实现对新的应用创新机制支持. MEC 也具有业务本地化、近距离及低时延的业务交付、为用户提供用户位置感知及其他网络能力的基本特征,如图4所示,将云计算和云存储拉近到网络边缘后,可以创造出一个具备高性能、低延迟与高带宽的计算环境,提升网络中各项内容、服务及应用的 QoS,用户也拥有更高的 QoE<sup>[23]</sup>. 一系列的新技术将被应用于 MEC 中,比如网络功能虚拟化 (network function virtualization, NFV)、软件定义网络 (software defined network, SDN)、边缘计算存储、网络切片急速、绿色网络通信技术等,其源于数据中心,扩展到网络边缘,可以提供更好的灵活性、扩展性以及面向终端的用户体验和性能.

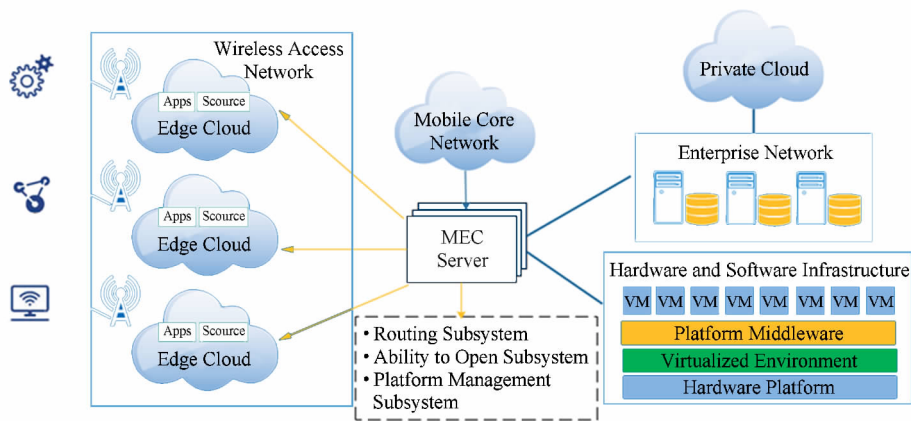


Fig. 4 Mobile edge computing architecture  
图4 MEC 体系结构示意图

3) 移动云计算. 移动云计算是基于云计算,将移动计算和云计算紧密结合,它将云计算集成到移动环境中,支持用户以按需、易扩展的方式获得所需的基础设施、平台、软件(或应用)等的一种 IT 资源或(信息)服务的交付与使用模式,移动云计算体现的是云计算技术在移动互联网中的应用<sup>[24]</sup>. 移动云计算的本质就是云计算,有效地利用了智能移动终端的处理能力、存储能力弱小、接入的灵活性和云中

心处理、存储能力超强的特点. 移动云计算是非常典型的中心计算模式,对于终端限制极小,只要能联网访问服务即可,其结构如图5所示.

移动云计算帮助移动终端降低硬件要求,便捷完成数据存取,利用云平台实现智能均衡负载,通过集中管理还可以降低管理成本和按需服务. 移动云计算会随着云计算的不断发展而扩大应用,尤其是提供基于大平台的面向终端用户的信息访问与服务



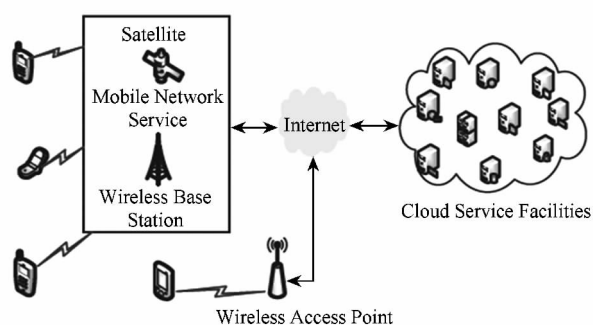


Fig. 5 System infrastructure of mobile cloud computing  
图5 移动云计算的系统结构示意图

请求的相关业务<sup>[25]</sup>.但是移动云计算的这种中心强大、边缘弱小的结果,对于具有大数据吞吐量和频繁数据交互、位置和延迟敏感的边缘实时大量计算与存储业务则会表现得无能为力,无法满足用户的即时响应的需求.当然,随着云计算的不断发展,由于对于信息查询等应用服务,天然符合了移动云计算的特征,因此也会进一步发展.

4) 雾计算.雾计算<sup>[26-28]</sup>是由思科公司最早提出的一个相对于云计算的概念,指的是将云计算扩展到企业或个人网络的边缘,有助于终端设备和云计算数据中心之间的计算,存储和网络服务的运行.雾计算更多强调基于硬件基础设施部署后的关于通信、计算和存储资源及服务在内的系统,涵盖终端用户或接近设备和系统的管理控制协同等,雾计算展示出比终端强、但比云弱的计算能力.边缘计算则更多地体现对于抽象硬件之后的应用系统层面的资源表示、协同、管理以及面向位置、时间、数据敏感的各类实例化的服务与应用<sup>[29-30]</sup>.如图6所示雾计算能对接终端用户、密集的地理分布进行移动性的支持.雾服务可以由网络托管或甚至在终端设备(例如机顶盒或接入点)中托管.主要的好处是边缘的邻近与智能的结合,可获得多个应用的实时或可预测的等待时间.因此雾计算适合实时数据处理和分析.

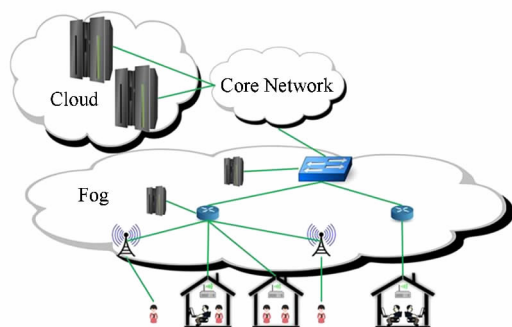


Fig. 6 Fog computing architecture  
图6 雾计算的结构示意图

## 2.2 边缘计算的特征

从互联网的角度看系统中的组成节点,核心层是由数据中心和云平台组成,分布于云和数据中心则是下一层中较小的数据采集与信息发布、局部计算服务器和内容分发网络;再下一层则是边缘,由诸如台式PC、平板电脑、智能电话、无线接入热点和微型数据中心、无线网络基站等具有一定计算能力的设备组成;再下一层则是各类传感器或者各种计算能力受限的具有联网功能的电子产品与器件,其数量巨大.由多层能力和结构不同的设备组成的互联系统,形成一个协同的整体,为用户提供服务,能在设备、网络、位置、服务能力等各方面都异构的条件下对用户透明、安全、可靠,体现了资源协同与优化的无缝、自适应和智能化的特点.边缘计算主要具有5个特点:

1) 边缘具有广泛的连通性.边缘计算环境中终端设备数量巨大,感知环境,形成广泛普遍的互联互通性.而接近网络边缘的计算、存储、通信的资源在提供服务时具有明显的优势,尤其对于大流量数据频繁交换的场景,这种计算模式比对等计算(P2P)和内容分发网络(CDN)具有更强的适应性,是两者的统一,将逻辑和物理的邻近保持了完美的一致,同时可以和中心协同.

2) 边缘实现智能感知与决策.随着物联网的迅猛发展,终端设备越来越小型化,计算能力也在不断增加,边缘的传感器和设备变得越来越强大,能够按需地感知周围的环境,并可以通过多个边缘设备的协同完成对本地数据进行处理,并形成基于时间和地点等上下文环境的智能感知的同时还能在本地进行智能决策,从而为边缘计算广阔的应用前景开启了大门.

3) 基于边缘的信任与安全保障.云计算通过将资源与信息汇聚到中心,形成了用户信息、用户行为的完全掌控,而现实社会中个人并不希望自己隐私信息完全被公司所掌握,而边缘计算完全可以将个人和社会的敏感数据暂时保存于边缘设备,而只根据应用需求将最终结果提交给服务器端,从而能够有效保护个人隐私.因此,以安全和专用的方式控制信任关系和敏感信息流的管理也必须属于边缘.

4) 边缘交互与控制融合.未来的应用是交互式,终端除了完成数据采集和局部计算还会根据应用的要求和管理的需要,对云端的资源进行利用,同时还需要根据计算结果和决策方案对本地的设备和装置和应用进行控制.资源的协同与控制需要来自

边缘节点以及云端服务的相互协同,选择性地部署、调度计算,同步或存储委托给其他节点或核心。

5) 边缘实现用户与服务统一。未来的地球、社会、城市、工厂等都会在丰富的感知节点帮助下,以穿戴移动设备为支撑,实现面向用户需求的、以人为中心的应用部署,用户在边缘可以提出各种服务请求,也可以接受各类服务,人类游走于边缘,全面融入整个社会和环境的信息空间。

因此,随着技术的发展,边缘计算逐步会演化为以用户为中心,实现广泛互联、具有隐私信任和安全保障、智能地感知和反馈控制的综合信息服务。边缘计算既会保留传统的 P2P、集群计算、云计算等分布式系统的部分特征,同时又具有上述新的特点,面向社会实际需求,伴随电子技术、通信技术、计算技术而不断演化发展。

### 3 边缘计算应用场景

云计算通过提供广泛的服务和几乎无限的可用资源,形成了大资源平台,得到广泛应用<sup>[1,31-32]</sup>。1999 年物联网被提出后迅速发展,联网终端产生大量的数据,大量的数据汇聚到云计算平台的数据中心,同时,很多部署在边缘的应用程序将使用这些数据。思科全球云指数估计:到 2019 年由人、机器和联网终端产生的数据将达到 500 ZB,而全球数据中心 IP 流量只能达到 10.4 ZB<sup>[33]</sup>。45% 由物联网产生的数据的存储、处理、分析和运行将靠近或者放在网络的边缘进行<sup>[34]</sup>。Cisco 公司预测到 2020 年将会有 500 亿终端连接到网络<sup>[35]</sup>。面向海量数据分析和处理的 IoT 应用可能要求很短的响应时间,有的可能涉及私人数据以及有些处理可能应对并发的超大流量的数据。由于存在大量 IoT 终端的海量数据的传输,即使云计算有着超大的资源池,也不能有效支持这些应用<sup>[36]</sup>。这是因为许多云计算应用是面向用户的,会经常遭遇大数据分析处理的场景不一定能满足本地实时、安全的要求,在物联网广泛应用的情况下,终端节点需要相互交互和共享数据,使用云计算中心作为一个集中的服务器,云计算中心仅仅是为用户之间、物联网设备之间,或者用户和物联网设备之间提供数据通信的一个通道、媒介和中间节点,去满足日益增加的用户设备之间通信的需求,如智能手机、平板电脑、智能家电、可穿戴电器和智能传感器等。这些设备在地理上处于网络的边缘,远离云数据中心。这种数据的通信或者边缘数据处理的需求

会造成大量的网络通信资源的开销以及云计算资源的开销,有些应用往往有很强的实时响应,云计算往往难以满足需求,造成通信和计算资源巨大浪费<sup>[37]</sup>。同时随着技术的进步,例如部署在大多数家庭中强大的专用连接盒、高容量移动终端用户设备和强大的无线网络,以及用户对信任、隐私和自主性的与日俱增的担忧。因此,有可能和必要寻找云中心相对应的网络边缘来完成有关的计算、数据分析或者数据传输,即如前面提到的边缘计算<sup>[18,38]</sup>。边缘网络不仅仅是数据消费者,也是数据生产者,将数据和计算以紧邻的方式存在,天然地满足这些需求,具有很好的优势。并且,随着物联网的不断发展,终端产生数据量不断急剧增涨,本地位置相关应用不断丰富,边缘计算可能和云计算对我们的社会有着同等重要的影响。边缘计算未来将在云端任务卸载、数据实时分析处理、边缘数据采集和本地内容分发、边缘交互与协同等领域具有很好的应用前景<sup>[39-45]</sup>。

#### 3.1 计算云端卸载

首先,在计算云端卸载方面,由于边缘有一定的计算资源,提供了从云端卸载部分工作量到边缘的机会<sup>[19]</sup>。一方面,传统的内容分发网络(CDN)中只是将云中心的数据在边缘服务器上缓存一份,从而节省云服务器的计算资源和网络路径上的带宽消耗,在边缘计算场景中依旧有效。更重要的一方面,边缘计算可以完成应用和服务的所有中间过程的处理,而将最后结果提交到云上,实现计算卸载。比如:基于边缘计算的移动网上购物服务,客户操作购物车,购物车的状态是动态变化的,避免了云计算模式下,大量用户购物车的变化产生巨大量的并发访问云中心,造成云服务器的负载和通信网路拥塞急剧增加。在城市交通管理中,传感器节点可以实时观察和感知交通运动,可以利用边缘服务器为车辆提供实时的本地交通状况信息服务,避免了 WAN 高延迟,海量数据传输与分析使得使用云服务的不可行<sup>[46]</sup>。智能家居中,物联网给家庭环境带来颇多好处,在家里用一个边缘网关来管理联网终端,数据在本地被采集和处理来减轻网络带宽的负担,同时,能更好地管理、交付与隐私保护。

#### 3.2 数据边缘实时分析

在数据边缘实时分析方面,如视频分析、移动手机和网络摄像机的广泛使用日益普及,由于较长时间的数据传输延迟和隐私问题,很多场景下云计算不再是适合于需要视频分析的应用程序。比如通过

视频寻找丢失的孩子、协同追捕嫌犯.在我国大部分县级以上城市通过实施天网工程,做到城区主要部分都部署有摄像头,市民也通过智能手机拥有各种照相机,并通过3G,4G以及WiFi网络连通起来.通过边缘计算模式,搜索一个孩子的请求可以在云端生成,然后发送到目标区域的所有联网终端.每一个联网终端,如智能手机,可以执行请求并搜索本地摄像头数据,最后将结果报告给云端.相比于云计算,边缘计算模式更可能借助每个终端的数据和计算能力从而更快地获得结果.比如海洋气候、城市安防、环境污染排放监测等.海洋监测通过探测气候变化应对像海啸等灾难性事件,这需要快速预测机制和迅速的信息传播,从而立即实时采取应急行动.部署在海洋上的传感器设备产生需要高计算资源和高带宽传输通道的大量数据,广域网海量数据传输延迟、云上数据处理延迟都将无法满足实时响应的需求,因此在边缘网络服务器上收集并处理数据有助于进行快速响应.

### 3.3 本地内容分发

在本地内容分发方面,移动互联网和在线社会网络的发展促成了本地信息分享的广泛应用,比如:终端用户通过照片、视频捕捉到本地安全或其他信息,可以直接向连接的通信基站的边缘服务器共享此信息.又如:在单个移动小区的用户中广播在音乐会中捕获的实况视频.在这些内容的传播之前,服务提供商可以向视频添加广告和其他相关信息.在这种情况下,服务提供商可以利用部署在Cell基站上的MEC服务器的计算资源来将信息添加到视频流.该信息被直接添加到基站上的视频,而不将视频重定向到另一个服务器进行相应处理并将其发送回基站,减少了处理的视频的传输延迟.

### 3.4 边缘的交互与协同

边缘的交互与协同是未来边缘计算的高级阶段,比如在反恐、健康管理领域等.恐怖分子一般特别善于隐藏自己,要识别恐怖分子往往需要最广泛的信息监控,这时广泛存在和部署的边缘网络成为了最重要的信息采集的基础部分,这些边缘网络往往地理上处于不同区域,逻辑上分属于不同的应用系统,比如视频监控、酒店管理、交通监控等等,反恐边缘服务器和其他各类应用的边缘服务器相互连接或者集成在一起.这样一旦有异常信息被捕获,反恐反应系统可以立即从触发点做出反应,而不会在云上去分析日志和统计信息等.

## 4 边缘计算的研究进展

随着物联网进一步普及与推广,边缘计算被广泛看好,边缘计算将成为一种普遍的计算模式与云计算共存与有机融合.学术界和产业界开展了广泛而深入的探索.

### 4.1 计算模式和系统框架

首先,在计算模式和系统框架方面,研究者面向终端-边缘-中心3个层次协同的资源管理与共享的应用场景,形成了各种计算模型与系统框架.Farris等人<sup>[47]</sup>提出了一种新颖的联合移动物联网即服务的边缘计算模型,利用终端节点和边缘服务器实现私有/公共移动IoT云的动态联合,兼顾考虑了面向物联网云提供商的合作,形成公共/私有云所有者相关联的所有设备联盟,同时将物联网云供应商分组选择的问题被建模成合作博弈的纳什均衡求解问题;Satria等人<sup>[48]</sup>依据过载/故障边缘服务器的传送范围内可用的相邻边缘计算服务器的数量,提出对应的恢复方案来解决边缘计算中的失效问题,通过邻近可用的移动设备用于中继过载/故障边缘计算服务器的数据,将分配最优ad hoc中继节点的问题建模为计算最优中继移动节点的优化问题,求解获得更多的边缘计算设备连接的可能性;Gomes等人<sup>[49]</sup>提出了一种端到端的云架构部署信息中心网络和移动Follow-me云组件,进而将内容迁移建模为优化问题,算法结合不同路由器的内容大小,内容流行度和缓存容量等因素,通过多属性决策(MADM)算法来寻找近似解,提出端到端云架构,M-FMC采用移动性预测算法优化内容迁移,具有很好的适应性;Tärneberg等人<sup>[50]</sup>提出了一种在移动云网络(MCN)中动态应用部署的系统模型,包括数据中心模型、网络模型、应用模型和用户模型,对用于管理MCN中的资源的应用布置的多个目标进行建模,目标函数包括节点的开销惩罚、执行成本和边缘资源,关键优化目标是最小化执行成本,评估和部署应用则通过穷举搜索算法决定任务的最佳位置,并确定其参考上界,通过可追踪迭代局部搜索算法求解;Li等人<sup>[51]</sup>根据雾计算与云计算相互关系与适用场景,提出3层系统架构,通过对每层的能量消耗和延迟进行形式化分析对框架进行建模,表示了物联网(IoT)环境中边缘(雾)和核心(云)之间的相互作用和合作;Satyanarayanan等人<sup>[10]</sup>提出了一种移动计算和云计算融合的架构Cloudlets,使得移动

设备的感知和交互能力可以与云中计算密集型和数据密集型处理无缝融合, Cloudlets 成为了移动设备和云中心之间的中间层, 设计一种应用自动感知的即插即用的架构, 并通过 Google 眼镜增强人类的感知和认知验证了其设计理念; Gosain 等人<sup>[52]</sup> 在一个部署有移动 4G 边缘网络、高速校园网络对计算和存储资源互联网的校园环境中, 构建了一个基于 GENI 边缘云计算网络的架构, 通过第 2 层网络互连, 并在美国的 50 个校园中应用, 研究者在 Wayne 州立大学校园里, 在警察巡逻车上运行的车辆传感和控制应用允许终端用户和研究人员收集丰富的数据集, 用于公共安全监控, 车辆内部状态感测和建模, 仿真下一代车联网技术, 这种边缘计算云基础设施非常适用于具有对延迟和带宽的高灵敏度的新型车联网; Liu 等人<sup>[53]</sup> 提出 ParaDrop 边缘计算平台, 在网络的边缘提供计算和存储资源, 允许第三方开发人员灵活地创建新类型的服务, 平台能够面向 WiFi 接入点有效应对终端设备的独特的上下文知识(例如邻近度、信道特性); Bhardwaj 等人<sup>[54]</sup> 认为后端驱动加载到边缘是一种解决未来终端设备与云交互时的带宽使用和延迟问题的有效方法, 进而提出一个快速、可扩展和安全加载具有边缘功能的平台; Nastic 等人<sup>[55]</sup> 提出了一个中间件为物联网云系统提供多层次服务的结构, 中间件采用通用的轻量级资源抽象机制, 支持 Edge 设备的特定应用定制, 通过动态管理的 API 以逻辑集中的方式自动配置 Edge 资源和应用程序组件, 采用灵活的供给模型, 实现 Edge 资源的自助服务, 按需消费, 并成功应用于楼宇管理系统领域; Amento 等人<sup>[56]</sup> 提出 FocusStack 的边缘计算管理系统, 基于位置的情境感知, 在多层地理寻址网络上实现, 解决低效的感知消息传递和混合主动性控制的问题, 并实现共享视频应用, 并展现出很好的效果; Orsini 等人<sup>[57]</sup> 利用移动边缘计算(MEC)或雾计算的基础设施组件创建 ad hoc 移动云, 为网络的逻辑边缘提供必要的资源, 设计一种安全、可靠和适应性的编程模型和框架, 满足设计弹性和可扩展的基于边缘的移动应用程序的需求; Bonomi 等人<sup>[58]</sup> 面对爆炸性扩张物联网终端, 提出一个分层的分布式架构, 涵盖从网络边缘延伸到边缘网络服务器, 用于处理海量物联网数据; Grieco 等人<sup>[59]</sup> 提出了一个可扩展和动态的中间基础设施 SECS(scalable edge computing services), 实现对异构机器的服务开发和部署高级 Edge 计算服务, 满足下一代网络的可扩展性、高可用性、容错和可靠性

的需求; 针对办公室、家庭及其他场所联网设备的数量不断增加, 形成大量潜在的计算基础设施可用于为终端用户提供与其与网络物理世界的交互的资源和能力, Bhardwaj 等人<sup>[60]</sup> 提出一种 ECC(edge cloud composites)的虚拟计算平台, 实现最终用户利用动态可用的本地和远程计算基础设施, 而不需要为每个场景显式地重写和/或重新配置应用, 并且以最小的最终用户干预; Lyu 等人<sup>[61]</sup> 生成容忍过时网络知识的渐近最优调度, 从而减轻对反馈的严格要求, 扰动的 Lyapunov 函数被设计为随机最大化效用, 以平衡网络吞吐量和公平性. 背包问题按照最佳时间表的每个时隙解决, 为所有设备的数据和能源积压提供最新的知识, 根据最新的网络知识将最佳时间表封装, 部分过时知识下的解决方案保持渐近最优, 并允许设备自行提名反馈; Rimal 等人<sup>[62]</sup> 探讨授权集成光纤的无线(WiFi)接入网络, 提供 ECM 功能的可能性, 然后分析了基于以太网的 WiFi 网络在边缘设备的延迟、响应时间效率和电池寿命方面的性能; Luong 等人<sup>[63]</sup> 开发了一个基于深度学习的边缘资源分配的最优拍卖模型, 构建了一个基于最优拍卖解析解的多层神经网络架构. 神经网络首先执行投标的单调变换, 然后为计算分配有条件的支付规则, 网络功能虚拟化(NFV)技术是实现 5G 网络目标的基本功能; 在文献[64]中 Salsano 等人首先介绍了 Superfluidity 项目提出的动态部署和虚拟功能组合的架构, 接下来考虑基于典型 5G 场景的案例研究, 并详细介绍了利用移动边缘计算(MEC)功能的视频流服务的设计和实现.

前述的研究在边缘计算系统架构与系统组成方面的研究做了很好的尝试, 并且在特定应用范例上取得较好的效果. 不过, 目前的体系结构总体上都具有很强特定应用的针对性, 而对于系统中云层、边缘层、终端层在带宽、计算、存储、能量方面的异构性、行为动态性、信任 and 安全性、优化目标的综合性等的考虑都需要进一步深入, 如何在面向异构多样的终端节点的可扩展性、面向用户不同需求的适应性以及面向参与者、资源的动态变化的透明性都有待深入研究, 最终构建一个安全、高效、可靠的边缘计算架构.

#### 4.2 云端计算卸载

在云端计算卸载方面, 当系统中存在数量巨大、动态变化的各类性能、特征不一的终端节点, 如何将计算或者传输的任务进行合理卸载迁移具有重要的意义, 往往能充分利用资源, 降低开销, 因地制宜显



著提高用户的体验效果.因此,边缘计算和云计算技术应该优势互补、相互融合.计算卸载和数据绑定是促进弹性增强低功耗终端移动设备(如智能手机)功能的关键技术之一.终端可以以面向服务的方式调用云服务来异步地完成资源密集、适于集中处理的任务;相反,在数据密集、交互密集、计算简单的任务适于被分割和分析,分配到边缘服务器或者终端节点进行计算和分析,达到系统效率的最优化.Flores等人<sup>[65]</sup>设计实现了一种移动云计算模型的卸载和委托框架,进行计算卸载和数据绑定,在资源一定条件下提高了系统性能;Lyu等人<sup>[66]</sup>提出了一种卸载请求和许可框架,考虑延迟和资源利用相关的个人卸载策略,激励移动设备参与边缘计算的任务中,同时,将多个用户之间共享有限资源的问题建模为混合整数线性问题,实现延迟约束下优化卸载准入和资源分配,还实现一个基于动态规划的能量最优卸载和资源优化方案;Lin等人<sup>[67]</sup>提出一种广域数据复制协议,保证数据完全一致性,获得边缘计算的低延迟和高可扩展性的优点,实现动态内容传输,且对被带到边缘的应用是完全透明的;Khan等人<sup>[68]</sup>提出一个Avatar移动云计算平台的中间件Moitree,允许未修改的应用程序通过移动端和云端无缝地执行以及在两端动态卸载计算和通信,还可以通过终端分组实现终端设备的自主协同;Mao等人<sup>[69]</sup>提出一个具有能量收集设备的绿色移动边缘计算系统,基于执行成本(其解决执行等待时间和任务失败)的性能度量设计有效的计算卸载策略,提出一种低复杂度在线算法,即基于Lyapunov优化的动态计算卸载(LODCO)算法,共同决定卸载决策,移动执行的CPU周期频率和计算卸载的发射功率;Tatemura等人<sup>[70]</sup>通过边缘计算加速Web服务 workflow 执行,提出覆盖Web服务网络(OWSN),基于CDN(内容交付网络)的集成Web服务管理和执行,引入服务水平要求规范提供多个服务水平偏好处理;分布式数据密集型 workflow 执行产生数据传输的网络瓶颈,Ramakrishnan等人<sup>[71]</sup>使用边缘资源加速分布式 workflow,通过使用代理网络减轻网络瓶颈,提高TCP/UDP传输速率,降低延迟和抖动;面对移动数据流量对蜂窝网络性能和经济带来的挑战,Gao等人<sup>[72]</sup>提出了一种基于议价的数据流量卸载机制,通过第三方WiFi或微微蜂窝接入点(AP)的数据卸载降低操作和资本支出,并有效地减轻蜂窝网络拥塞,使用纳什讨价还价理论来模拟和移动蜂窝网络运营商(MNO)和多个AP所有者(APO)的合作和协议.

此外,Han等人<sup>[73]</sup>提出利用绿色内容代理卸载移动流量方案,通过启用设备到设备(D2D)通信,基站(BS)通过业务卸载来降低其能量消耗,移动用户通过从其邻近对等端而不是远程BS检索内容来提高其QoS,同时,对利用绿色能量和带宽的量的约束最大化流量卸载问题建模,并提出一种启发式流量卸载(HTO)算法求解近似最优解;Sun等人<sup>[74]</sup>经用户设备(UE)与其云端代理保持交互,降低端到端(E2E)延迟,根据收益与开销进行建模,在Cloudlet网络提出最大化收益位置选择策略,选择性将代理迁移到最佳位置获得最优收益,针对数据流从多个源和地理位置产生,通过广域网将原始数据流传送到中央数据中心引起的大量终端节点产生的网络边缘突发数据量的突发,无法满足具有可预测QoS和低延迟要求应用的需求;于是Sajjad等人<sup>[75]</sup>提出一种统一流处理方法SpanEdge,通过在中央和近端数据中心分发流处理应用程序来减少或消除WAN链路所产生的延迟,在地理分布式基础设施中最佳地部署流处理应用程序,显著减少带宽消耗和响应延迟;Satyanarayanan等人<sup>[76]</sup>针对高速率传感器数据采集与上传的需求,构建了一个互联网规模的存储库GigaSight,包含强制实施隐私偏好和访问控制的众包视频内容,通过基于VM的云与边缘的联合架构,在互联网的边缘执行视频分析;Satyanarayanan<sup>[77]</sup>通过对现实世界事件进行概念性的理解,将这些事件升华为更高层次的概念结构,将其综合为一致的上下文敏感视图,并探讨视频感应、众包和边缘计算中的趋同趋势如何被用来创建适于无人驾驶和驾驶车辆的车辆系统中的情景感知的共享实时信息系统;Taleb等人<sup>[78]</sup>提出了一种在智慧城市中增强用户流式视频体验的方法.依靠MEC的概念作为提高QoS的关键因素,通过确保应用/业务遵循用户的移动性,实现“Follow Me Edge”概念,维持QoS.该方案强调MEC服务的自主创建,以允许随时随地以最佳的QoE和较少的延迟访问数据.移动边缘计算(MEC)和无线电力传输(WPT)已被确认在物联网(IoT)的有前途的技术是为用户提供增强的计算能力和可持续的能源供应大量的低功耗无线设备.Wang等人<sup>[79]</sup>考虑无线供电的多用户ECM系统,其中,多天线接入点(AP)(与ECM服务器集成的)广播无线电力给多个用户,并且每个用户节点凭借充电提出了一种统一的MEC-WPT设计在收获的能量上执行计算任务.使用MEC,这些用户可以

自己在本地执行各自的业务,或者基于时分多址(TDMA)协议将全部或部分用户卸载到 AP。

由于边缘服务器和数量庞大的终端节点具备可观的带宽和计算能力,同时还是数据的产生和采集者,研究者在蜂窝网络通信、工作流优化、计算卸载、内容卸载等方面做了很好的探索,并取得了好的效果。然而,在多条件约束,多目标优化的边缘计算卸载问题建模与求解,设计高效的算法充分利用边缘的资源有效解决问题,实现优化卸载对象选择与任务分配,减少云-端间无用数据传输,以及通过经济手段激励等方面,还需要突破性的进展。

#### 4.3 资源管理与协同

在资源管理与调度方面,边缘计算中的资源优化调度是核心问题之一,只有实现了资源的科学管理、合理调度与分配,才能面向实际需求,充分发挥出云、边缘服务器、终端的节点优势,实现利用率、能耗、带宽、存储等全方面的优化,最好地平衡 3 层资源的使用,最大限度地节省资源,提高收益,最好地满足用户体验,当前的资源调度研究主要集中在能耗、延时等单个指标的优化调度。Ahmed 等人<sup>[80]</sup>在移动边缘计算系统中提出一种联合调度算法协调地分配无线电资源和计算资源,通过在正交频分复用接入(OFDMA)的多个移动设备系统中,构建一个边缘计算云层用于任务卸载的子载波分配和用于在该云层中的任务执行的 CPU 时间分配,比单独频谱和计算的优化能接纳更多的卸载请求,同时实现节能效果显著;Sardellitti 等人<sup>[81]</sup>通过对 MIMO 多小区系统的多个移动用户(MU)计算卸载到公共云服务器的问题转化为无线电资源(MU 的发射预编码矩阵)和计算资源(由云分配给每个 MU 的 CPU 周期/秒)的联合优化问题,最小化总体用户的能量消耗,同时满足延迟约束,并提出一种基于新颖的连续凸近似技术的迭代算法,收敛到原始非凸问题的局部最优解,并且跨越多个无线接入点 AP 实现分布式和并行计算;Kiani 等人<sup>[82]</sup>提出了一种新的层次模型,通过引入领域、浅层和深层云的概念,在终端设备、边缘层、云层 3 个层次级别中进行设计,以基于拍卖的利润最大化方式提供计算资源,然后分配通信资源以满足用户的 QoS 要求;Tran 等人<sup>[83]</sup>介绍和研究了从移动边缘编排,到协同缓存和处理,以及多层干扰消除 3 个代表性的使用案例,证明了所提出的方法在促进向 5G 网络演进方面的有前景的好处,并讨论需要解决的关键技术挑战和开放性研究问题,以便将 MEC 有效整合到 5G 生态系统中;

Sun 等人<sup>[84]</sup>开发了一种新的以用户为中心的能量感知移动性管理(EMM)方案,以便在用户的长期能耗限制下优化无线接入和计算两者的延迟。基于 Lyapunov 优化和多武器强盗理论,EMM 以在线方式工作,没有未来的状态信息,有效地处理了不完善的系统状态信息;Mao 等人<sup>[85]</sup>开发了一个多用户 MEC 系统的在线联合无线电和计算资源管理算法,目的是最小化移动设备和 MEC 服务器的长期平均加权和功耗,受任务缓冲器稳定性约束;面向具有时间约束应用以及多运营商提供服务环境,Katsalis 等人<sup>[86]</sup>将服务重要性作为关键指标,研究边缘网络部署服务的调度,基于时间约束服务 SLA 对虚拟机(VM)调度决策,同时保证服务提供商之间的公平。通过 Lyapunov 优化框架来求解该随机优化问题,最大化具有时间约束 SLA 的多网络运营商共享环境中物理基础设施所有者的收益。面向移动设备上提供满足要求的增强现实(AR)问题,通过 AR 移动应用在上行链路中的数据收集,边缘上的计算和下行链路中的数据传递方面具有固有的协作属性,Al-Shuwaili 等人<sup>[87]</sup>提出一种在发送、接收和处理的数据在用户之间部分共享的资源分配方法,以获得通信和计算资源的有效利用,与常规独立卸载相比,通过连续凸函数近似(SCA)方法在移动能量消耗方面显著优化;针对计算任务调度策略设计的大小时间尺度的随机优化问题,Liu 等人<sup>[88]</sup>采用 Markov 决策过程方法,计算任务调度基于任务缓冲器的排队状态,本地处理单元的执行状态以及传输单元的状态,通过分析每个任务的平均延迟和移动设备的平均功耗,设计功率约束延迟最小化问题,提出了一种高效的一维搜索算法来找到最优任务调度策略。

#### 4.4 信任与安全管理

在信任与安全管理方面,在边缘计算中,涉及到边缘服务器、大量终端参与到资源访问、提供资源完成计算与数据传输等,这些资源具有动态性、异构性的特点,而边缘计算往往缺乏像云计算集中控制的安全机制,也使得参与者的安全风险加大。因此,边缘计算中的信任管理也成为了边缘计算任务能否有身份可信、行为可信、QoS 满足要求的能力可信的用户来协同完成的问题,成为实现一个安全可信的资源共享的计算环境的关键要素之一,关系到边缘计算环境是否真正走向应用,也是负载卸载、资源调度与共享得以实施的根本条件。信任管理在其他包括云计算、对等计算中得到了广泛关注和研究<sup>[89-90]</sup>,但目前在边缘计算领域研究不多,且大多数研究只

集中在移动云计算的领域,分析用户之间的信任关系。Petri 等人<sup>[91]</sup>研究了各种节点如何创建一个值得信赖的对等点云,其中反馈聚合用于识别自我利益的用户;Chen 等人<sup>[92]</sup>通过分析呼叫模式推导人类用户之间的信任关系;Hussain 等人<sup>[93]</sup>对如何计算边缘数据中心的声誉进行了研究,采用集中式信任管理方式,其存储 LTE 部署的云的信誉,通过此系统,用户可以匿名匿名评估 cloudlet 的服务;Kantert 等人<sup>[94]</sup>提出一个自我维护的信任管理系统,实现了来自不同管理域的自治服务器在类似网格的情况下共享其资源,与其他网格部署相反,假设存在自私的个体或恶意服务提供者,因此,设计了以自主和分布式方式计算一组信任度量,实现计算边缘数据中心之间的信任评估和管理。Figueroa 等人提出定量信任管理组件,并集成到不可信网络设备的(SOUND)安全平台<sup>[95]</sup>上,该平台由几个信任社区组成,其中包含各种主机。每当来自不同社区的 2 个主机交互时,他们不仅考虑到他们的相互信任,而且考虑他们社区之间的信任以及社区和另一个主机之间的信任。由于信任社区和边缘信任域之间的相似性,可对在边缘数据中心中部署信任管理系统提供参考。Bennani 等人<sup>[96]</sup>定义了用于混合云计算环境中的基于贝叶斯网络的信任模型,私有云可以评估和跟踪公共云提供的各种服务的声誉,该模型能对边缘计算的整个生态系统的可用服务的动态声誉进行管理。Jang 等人<sup>[97]</sup>提出 SOUL,通过关联的执行器透明和统一地访问设备和环境传感器完成相关任务,任务不仅可以在智能手机上处理,也可以利用边缘服务器和远程云资源,通过访问控制允许用户轻松定义传感器的访问权限,利用用户的社交关系并捕获访问请求的上下文,从而使得 Android 平台上的应用程序扩展传感器的使用性能和能源效率;Echeverría 等人<sup>[98]</sup>为非连接的边缘计算环境提出了基于安全密钥生成和交换的可信身份的解决方案,并通过开源实现的战术云计算系统进行了验证,大规模智能环境(LSE)是开放和动态的系统,通常在广阔的区域延伸,并且包括大量具有异构性质的交互设备。因此,在部署期间,可扩展性和互操作性是需要考虑的关键要求。为此,鉴于新设备和功能不断加入 LSE,必须增加服务和对象的发现和声誉评估。Cicirelli 等人<sup>[99]</sup>提出了一个基于代理的方法,利用边缘计算和社会物联网范例来解决上述问题。通过涉及商业道路环境的样本案例研究来评估这种方法的有效性。

面对边缘计算环境的异构资源的系统共享需要考虑资源性能的匹配、交互双方行为信任、身份信任、资源提供能力信任等因素,才能有效面对物联网中无处不在的终端节点之间相互交互、共享的挑战,比如像智能手机等资源受限设备上运行的应用,面临着如何被授权不同层次地动态自如地访问需要的可信的各类节点的问题,以及防止被不可信节点欺诈的问题,基于用户体验质量的信任可靠的边缘计算资源调度和共享问题都需要深入开展研究。

## 5 边缘计算的挑战与未来

云计算和数据中心应用也不断推广,各类传统应用系统都部署在云、数据中心,形成了成熟的云端的计算模式,表现为高度集中;同时,物联网技术迅猛发展,物联网应用、终端数量与种类都急剧增殖,体现出极大分散的特征。云计算面对海量分散的终端的一些应用在延迟约束、信任隐私保护等诸多方面需求难以应对,并且运行时整个系统开销大,还造成边缘资源的浪费,而边缘计算技术在网络的边缘执行计算,减少了响应延迟,充分利用了边缘节点有限资源,系统效率提升。下行数据代表云服务,上行数据代表物联网服务,“边缘”则是连接起数据源和云数据中心之间的任意计算和网络资源。边缘计算、雾计算、移动边缘计算、移动云计算、朵云计算这些概念都是属于边缘计算范式,其出发点相互间有些区别,雾计算更关注于边缘网络基础设施以及部署计算的平台,移动边缘计算则更多关注联网的终端的移动性,移动云计算与朵云计算则考虑的主要是基于边缘服务器条件下的集中处理问题。实际上,如何与云中心进行协同也是边缘计算需要考虑的问题,因为边缘计算不能完全脱离云计算而独立存在。数据的采集传输处理,计算任务的卸载、装载与调度,还包括信任和安全,才能支撑系统可扩展性、适用性。

边缘计算的应用前景已经得到学术界和工业界的广泛关注,探索性的研究已经广泛开展。目前,关于边缘计算的研究工作主要是针对一个特定需要解决的问题应用提出对应的解决方案,并进行了示例论证,比如移动终端计算卸载、传输卸载,物联网终端采集的数据的传输卸载与边缘实时处理、终端节点协同完成任务等,应用的领域涉及到智慧交通管控、智能家居、实时视频图像处理、环境监测等,一般都取得比较好的效果。不过,未来边缘计算真正在

工业界和市场得到广泛普及的应用,还面临诸多的挑战.

### 1) 多层次资源、对象的表示与管理

随着物联网的不断迅猛发展,接入的终端设备的数量会不断增加,种类更加繁多,设备接入的方式也是异构多样,提供服务的方式也会因为场景不同而变化,因此需要对终端的实体与对象、边缘实体与对象,以及云端的实体与对象等多层次的资源和对象进行有效地表示与管理,因此需要面向边缘计算场景建立命名机制、寻址与路由机制等,从而使所有的资源与对象能统一在边缘计算平台之中.

### 2) 高效自适应的计算模式与体系结构

云计算之所以得到迅猛发展在于工业界努力提供了一种高可用、高可靠、高性能且自由选择的计算服务,这种服务可以满足不同层次和不同需求的用户,可以是基础硬件设施,可以是虚拟的计算环境,也可以是某种应用服务,能全方位满足各种用户的需求.而边缘计算在计算模式与计算系统结构方面面临巨大的挑战.巨量的异构终端接入后成为系统的组成部分,既可以是资源和服务的提供者,也可以是服务和应用的请求者;同时,这些节点有些是固定不变和能力巨大的,有些则是可以动态变化和移动,且上线和下线的时间也是动态的,如何协同起系统中各异构对象,并确保系统在规模动态变化等状态保持高度自适应性、高鲁棒性和高可扩展性也成为边缘计算的计算模式与系统体系结构设计的另一个重要挑战.

### 3) 隐私信任与安全保障问题

以边缘为中心的计算超越了以前尝试使用E2E(端到端)加密和试图保护云中的用户信息的以用户为中心的隐私系统.以边缘为中心的架构将以新的方式挑战研究人员.除了加密以保护私人信息之外,将需要更多的安全代理用于使用诸如重新加密或基于属性的加密等不同技术的会合,通信和访问控制.此外,必须创建用于隐私感知信息共享的新型安全中间件,以促进以边缘为中心的系统.

云安全上的许多现有工作,例如加密数据存储、对加密数据的查询,同态系统可以有助于创建新的以边缘为中心的服务.与传统云安全研究的一个重要区别是:以边缘为中心的计算可以假定存在可信的或部分可信的稳定资源,执行一些通信、持久性、查询甚至是在边缘部署和控制的应用的计算.以边缘为中心的计算还可以考虑在基于分布式边缘的覆盖中的可信节点与恶意节点的共存.这将重新需要

安全路由、冗余路由、信任拓扑和以前的P2P研究应用于这种新颖的设置.另一个关键区别是:与集中式计算相比,以边缘为中心的计算防止了信息的集中.以前的关于信息碎片化与加密的云安全研究可以与分散覆盖技术相融合,以确保敏感数据的适当数据保护.此外,安全的云查询和对覆盖网络中的分段数据和索引的计算可以创建关于敏感信息的隐私的全新模型.

在这些边缘计算的研究中对于解决问题面对的场景进行建模,构建了合适的边缘计算体系结构,也包含资源的管理,具体来说涉及到任务的卸载和资源调度协同来共同完成计算任务2个方面.对于资源优化利用建立优化模型进行求解,不过问题求解的优化目标总体比较单一,而未来物联网的迅速发展意味着终端的硬件、软件、计算能力、通信能力、存储能力都是异构的,用户的需求也会是多样的<sup>[100]</sup>.因此,未来边缘计算的核心问题还是面向用户QoE的资源协同管理、调度优化问题,进而能够资源完美匹配、充分利用,用户获得最满意的QoE.而在这方面的探索期待进一步的推进;另外,很重要的一个方面是边缘计算的信任管理问题,这是保证边缘计算实际应用的重要制约因素,在网格计算、P2P计算中相关的研究较多,而面对边缘计算场景,个人终端直接参与到计算中来,跟人的社会生活紧密相连,只有保障了个人隐私与信息安全,边缘计算才能得到全面的发展<sup>[101-103]</sup>.

边缘计算正在引起研究者的广泛关注,学术界和工业界都在体系机构、计算模型、任务卸载、资源调度以及信任管理等方面开展了系列探索研究和试验系统的部署.基于用户的体验质量QoE,从系统的信任安全角度出发,充分考虑终端、边缘、云三方的资源管理与协同优化还需要深入研究,为边缘计算的广泛普及奠定坚实的理论基础.

## 6 基于信任的边缘计算模型

由于云计算具有计算、存储等资源巨大丰富、安全信任集中管理的特点,但是具有不可靠的延时、缺乏移动支持与位置感知;边缘计算的资源有限,缺少集中的信任管理,但数据和计算相互非常接近,同时还具有动态移动特性.因此,如何对包括海量终端节点、边缘服务器、云中心所有计算参与者在内的计算系统设计科学合理的计算架构、合理的任务卸载与迁移策略与机制、高效的资源调度与优化机制、有



效的信任管理与保障成为边缘计算需要解决关键问题。

本文第 6~9 节对边缘计算场景进行分析,对云、边缘、终端资源进行建模,实现对系统中的对象进行命名与标识,表示资源的 QoS 属性如计算能力、存储能力、传输能力等,同时还要考虑不同层次的属性比如移动性、行为可靠性以及身份属性等因素。类似地,需要对任务 and 用户进行建模,构建基于综合信任的表示模型,通过身份信任、行为信任、能力信任来综合表达资源关于前述的全面的的信息,构建融合云、边缘、终端的 3 层计算模型。然后,在计算模型的基础上,基于用户 QoE 要求,任务和资源基于综合信任度的匹配问题进行评估,重点考察延迟、带宽、存储、能耗、成本的因素,确定任务卸载策略。最后通过流计算任务处理实例,考察在边缘执行任务时资源的调度与分配问题,通过实验展示出基于综合信任的模型与系统具有比较好的适应性和性能。

边缘计算涉及到 3 个层次的对象,包括:唯一远程的资源丰富的云中心、一定数量资源相对丰富的本地边缘服务器、数量庞大的资源受限的终端节点。计算任务的完成可以在任意一层对象中完成,也可以通过多层协同完成,而在此过程中对象还处在动态变化之中。边缘计算的目标是要满足用户 QoE 的要求,同时从整个大系统的角度来看要节省计算、传输、存储、能源的开销。边缘计算的场景相比集中管理的云计算、单纯的 P2P 计算、传统的客户/服务器计算抑或资源异构的网格计算等计算模式的结构都要复杂,不属于其中一种模式,而是他们中的一种或多种模式的组合,设计的计算系统应该具有灵活性、可扩展性、适应性、鲁棒性。

### 6.1 多层的集中与对等混合的计算模式

根据上述的分析我们知道,从参与计算的实体对象来说包括了从云到边缘再到终端 3 个层次不同物理特征的设施,还包括了异构的连接网络。从应用场景和任务请求来看,其表现为多样性。有些任务具有很强的局部、实时(local, realtime)的特性,计算任务的完成仅仅需要一些邻近的终端交互和处理即完成,是一种完全的 P2P 计算模式<sup>[104]</sup>;有些任务则具有一定的计算资源需求,终端往往无法完全承担,同时具有较大交互频率和数据传输量,则通过采用 C/S 结构的自主计算模式,让边缘服务器来完成计算;如果计算任务与更大的邻近区域关联,需要更多邻近的终端节点和边缘服务参与,此时的需要采用网格计算模式;另一种情况是计算任务属于计算密集

型的,需要强大的云中心来完成计算,这就变成了一种云-端结构的云计算模式,但是如果处理的数据和发起的计算涉及多个终端节点和多个边缘服务器,任务还需要被合理分配到终端、边缘服务器、云中心以获得最优的资源利用率,从而演变成一种集 P2P, C/S、云计算混合计算模式。这是物联网技术、云计算、移动通信计算不断发展和业务需求演变的未来的实际情况。

### 6.2 多层可靠信任的边缘计算模型

通过计算模式的研究我们知道边缘计算模型中的组成对象具有数量巨大、种类繁多、形式多样、差异巨大的特征。同时,实体之间由于计算能力的强弱还存在明显的层次关系,在不同的层次之间还表现出动态性的特征,终端节点相对来说具有移动的动态性,而边缘服务器和云中心则在负载和可用资源状态上表现出动态性。因此,1)要将所有参与对象进行虚拟化表示,并通过合理的命名机制保证模型中的对象实体的唯一性;2)对于资源、用户等模型中的对象进行性能属性、信任性等进行评估;3)需要对接入模型的计算请求进行用户 QoE 要求评估,并对优先级以及资源匹配进行计算,实现准入控制和资源合理分配;4)模型需要具有可扩展性,对边缘计算环境中,由于终端的移动性,以及物联网终端不断增多,需要感知新的终端并接入系统;5)鲁棒性需求,当系统中有节点失效需要能够探测到并进行隔离,除了硬件故障,服务还可能由于各种原因而无法响应,模型应该有对应失效解决方案,保证系统的可靠性。

### 6.3 基于异构的统一边缘计算模型构建

以综合信任度为基础的资源与用户的身份、行为以及各种 QoS 性能特性为边缘计算的运行模式、资源管理调度策略选择思想,确定了以人的体验质量 QoE 为优化目标的方案设计准则。

边缘计算中,人是其中非常重要的因素,移动智能手机现在成为了每个人基本必备的边缘设备,具有丰富的传感器以及多种网络连接接口。同时,人的应用需求直接影响到计算模式,比如发起一个局部的实时视频分析或者视频分享的请求,又比如发起一个本地相关信息查询请求,还比如发起一个科学计算服务请求。个人通过移动端发起了 3 类不同特征的请求,第 1 个请求需要邻近的节点进行协同完成,第 2 个请求则需要边缘服务器进行响应,第 3 个则需要云计算中心提供服务支持。因此,要以人的需求为中心进行应用场景进行服务分类,根据服务类

型匹配对应的计算模式,同时对关联的各种计算实体进行虚拟化建模表示,图7可以看出各实体的所处的层次<sup>[105]</sup>,亦可以通过不同场景关联不同层的不同设备。

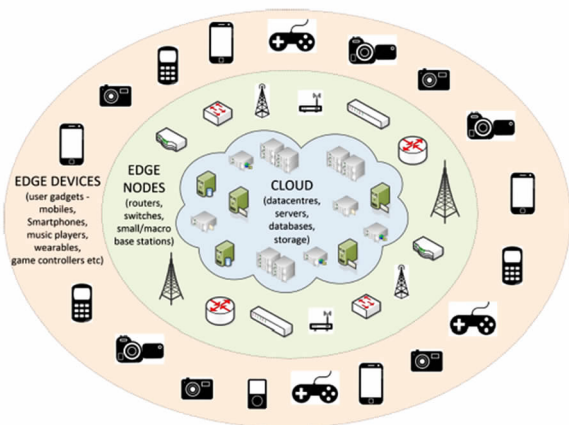


Fig. 7 Computing component distribution from centric to edge

图7 边缘到中心的边缘计算组件分布

进一步要对边缘计算环境(edge computing environment, ECE)中实体的性能、行为特点构建边缘计算模型,通过边缘计算单元(edge computing unit, ECU)、边缘计算执行体(edge computing executor, ECX)、边缘计算联合体(edge computing combo, ECC)来进行边缘计算建模,并构建边缘计算体系结构,实现资源按场景需要进行共享与协同优化,提高用户 QoE。我们通过对应用服务与应用

场景分类,同时考虑边缘计算单元的网络连接实际情况,设计边缘计算模型如图8所示。模型中可以看出,边缘计算单元有3种不同的类型,边缘设备(edge device, ED)、边缘服务器(edge server, ES)、中心节点(central node, CN),资源共享与协同有4种基本的模式,通过执行体将协同的资源关联起来,执行体1表示的边缘设备与边缘服务器之间协同;执行体2表示边缘设备通过P2P方式自主协同;执行体3则是一个边缘终端直接请求云计算中心服务的资源;执行体4则表示在邻近区域通过边缘服务器互联将所覆盖的区域的边缘设备协同起来共同完成某项实时位置相关的任务。该模型只体现了资源协同的方式,其中信任管理与资源共享与协同优化的算法并未能直接体现出来。

边缘计算的体系结构如图9所示,其中,服务应用层主要体现出系统中用户发起的服务请求;应用场景驱动层则是针对边缘计算的场景建模后,将服务请求关联到某种服务场景,以便与相关资源进行协同;资源管理与协同层主要负责用户资源请求和系统资源提供的调度与匹配工作;计算模式匹配层则是将按照适当的计算模式完成计算处理;资源表示层主要负责对于物理资源的抽象和表示,便于建模。边缘计算中,通过边缘计算单元对计算系统中的资源和用户进行抽象和封装,其结构如图10所示,主要包含与外部交互部分有服务请求接口、服务提供接口、资源状态发布/请求接口,同时内部则包含

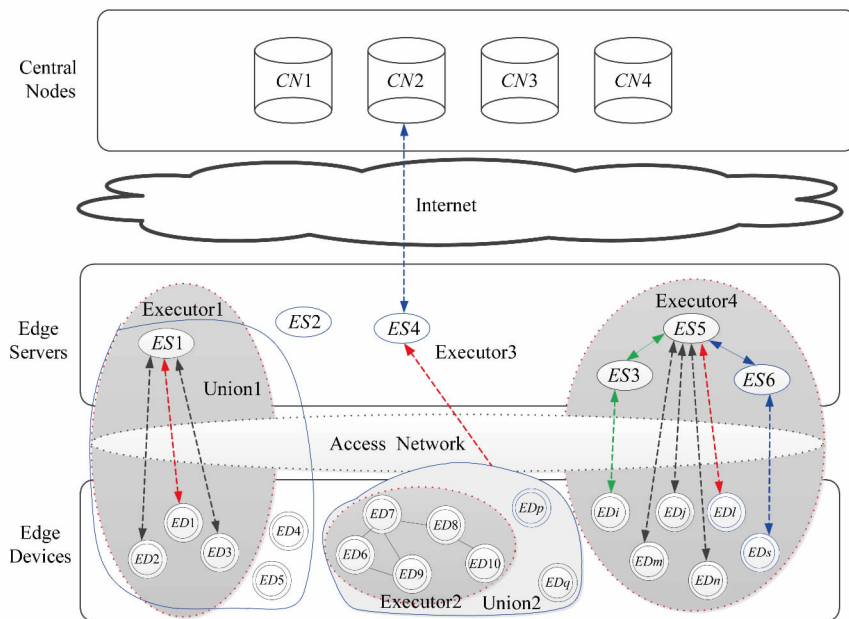


Fig. 8 Multi computing mode in edge computing

图8 多模式的边缘计算模型

有身份信息、状态信息、局部环境与邻居信息等,还包括资源共享与调度策略、信任评估机制等。不过,根据节点的硬件性能,节点可能只具备其中部分的功能。

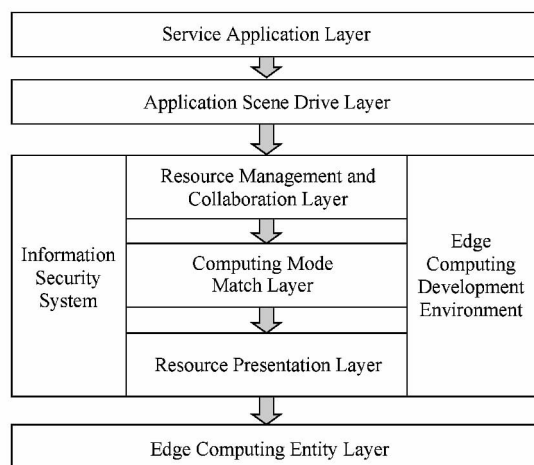


Fig. 9 Architecture of edge computing

图9 边缘计算体系结构示意图

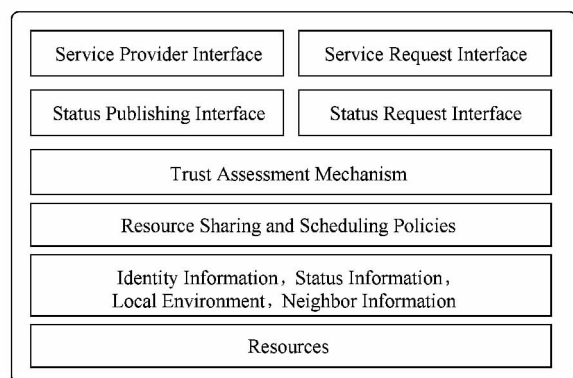


Fig. 10 Infrastructure of edge computing unit

图10 边缘计算单元结构示意图

通过对边缘计算建模实现对其组成结构、边缘架构、中间件兼容原有计算系统,我们还要考虑新架构可能依赖于部署在云计算、数据中心中的实际表现为以边缘为中心的分布式服务,为边缘节点的会合、通信、计算、内容分发和存储建立实现参考与标准,提供更为适用的分布式服务,也支持跨云计算数据中心和边缘设备分发应用程序,同时确保最终用户的控制和隐私。

## 7 边缘计算的 QoE 与信任评估体系

### 7.1 综合信任评估体系建立的动机

各种性能各异、身份不一、行为不同的边缘计算

中的各种基础设施,通过虚拟化被建模成系统独立的节点,通过有线或无线不同带宽与可靠性网络连接起来,最终形成了一个综合互联的大系统。当有计算任务发起或一个服务请求发起时,系统面临一个需要解决的问题就是如何评判这个任务或服务请求的发起者是否可信,该任务或者请求对于资源的要求如何评估,从而为资源的分配和调度提供依据。同时,又如何保证声称提供资源的节点能够真正提供对应的资源,这需要研究边缘计算的信任保障体系。可信问题涉及到资源共享与协同的安全、信任、服务质量及其保障机制,已经成为边缘计算需要深入研究的关键技术问题之一,我们需要对资源和用户的信任度进行综合性的评估,需要研究用户、资源的身份信任、行为信任以及能力信任,从而建立详细的信任评估体系,为资源共享和调度提供支持。

随着物联网、移动互联网以及在线社会网络的迅猛发展,其实线下社会中的许多事务已经完全置于线上进行处理,展示出一个虚拟的真实社会的运行体系。边缘计算在这方面的应用也会越来越多。而边缘计算有其特殊性,涉及到云、边缘和终端3个层面,在云计算系统中,身份信任是建立可靠的身份认证和审核基础上的,具有很高的身份信任度;边缘服务器作为一种比较固定安放的基础设施其身份信任度也是非常可靠的;而终端节点有些经常处于移动状态,具有较强的动态性,因此,其身份信任往往难于确定。当然也可以通过注册后,进入边缘计算环境中进行自动登录确认,有些终端可能出于隐私考虑不愿意透露身份信息,则需要新的身份认证机制。另外对于行为信任对于边缘计算也非常重要,终端节点对于有些实时性要求高的应用非常重要,移动终端协同完成这类任务比分配到云计算中心来完成具有更好的用户体验和更小资源开销,如果移动终端节点只期望从边缘计算中获得好处,而不愿意贡献资源,又或者不能贡献声称的资源,造成协商的任务不能按要求完成,因此需要借鉴社会学以及以往P2P计算系统中关于个人信任演化的规律,设计有效的行为信任机制。我们将能力信任定义为在边缘计算被定义为对服务请求的响应能力的总称,可以包括可用服务时长、CPU计算能力、能量维持能力、单位服务时间价格、存储空间大小、传输带宽大小等。在边缘计算中,由于不同类型的节点其能力信任度相差很大,但是能力信任度要能充分发挥作用还与其他诸多因素相关,比如位置的邻近性,通信带宽

的限制性,因此能力信任度有绝对值和相对值的区别.另外,应该依据应用场景的分类以及实际一个计算任务或者请求的实际上下文特征进行科学评估,才能有效评价,并为资源共享提供有力的支撑,需要对根据节点资源属性与应用需求相似度和场景相关性进行互信任都评价.

总之,在边缘计算中,需要建立一套综合的信任度评估体系,系统的安全与系统可靠高效运行提供保障.

## 7.2 基于综合信任度的评估体系构建

面对多样的应用需求情况,以及边缘计算实体的资源情况和相互关系,本文提出以身份信任、行为信任、能力信任为核心内容的边缘计算综合信任保障体系,根据系统状态和应用需求形成对系统的信任预期,据此构造保障边缘计算单元的身份信任、行为信任、能力信任的基本框架,协调和控制边缘计算单元的活动,支持系统服务目标的实现,信任保障体系结构如图 11 所示:

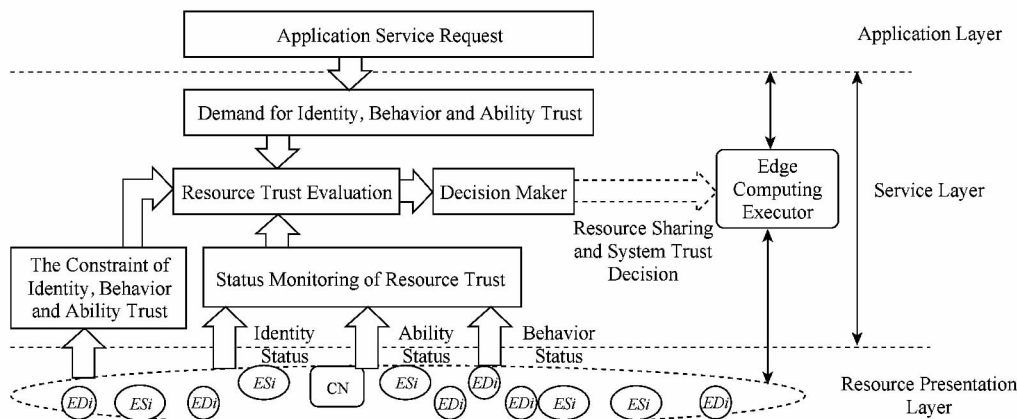


Fig. 11 Infrastructure of trust gaurantee in edge computing

图 11 边缘计算信任保障体系示意图

从图 11 中,可以看到用户发起的服务请求实际上是对资源的身份、行为的信任以及对于资源性能 QoS 等能力指标是有要求的,信任保障体系则实现需求到资源的协同与匹配,因此边缘计算信任保障体系要对资源的信任评估,确保符合条件的资源参与到请求服务的任务完成中来,资源抽象表示时信任属性是需要表示的重要指标,同时还需要信任状态监测保持相对比较新的状态,每一个资源其所属类型互不相同,个体差异大,资源供给能力也不相同,具有不同的约束门限.根据边缘计算环境的基于信任的资源共享与协同过程中的边缘计算单元、边缘计算执行体的相互关系以及内在的特征与属性,结合有关的计算模式,建立基于综合信任保障的边缘计算模型.

边缘计算的信任模型包括边缘计算环境、边缘计算联合体、边缘计算执行体、边缘计算单元等概念.边缘计算环境由一个三元组表示  $ECE = (ECU, ECC, ECX)$ ,其中  $ECU$  是指边缘计算单元  $ecu$  的集合,表示边缘计算环境中有不同应用需求和不同应用场景驱动的各种资源的总和; $ECC$  是指边缘计算联合体  $ecc$  的集合,表示为完成某应用服务而符合信任条件边缘计算单元集合; $ECX$  是指边缘计算执

行体  $ecx$  的集合,即为完成特定计算应用或任务通过信任评估符合条件,并且参与资源共享的边缘计算单元  $ecu$  的集合.

边缘计算单元  $ecu$  通过公共属性、信任属性来建模, $ecu = (cp, tr)$ ,其中,  $cp$  表示  $ecu$  的公共属性,所有边缘计算元素均有的属性描述,如  $ecu$  的  $id$ ,名称  $name$  及描述  $des$  等,  $ecu$  的功能  $fun$ ,类别  $class$  等;信任度  $tr$  表示  $ecu$  的信任属性,在边缘计算联合体以及构建边缘计算执行体时信息交互过程中表现出来的信任度.信任度  $tr$  则由身份、行为、能力 3 部分进行建模,即可表示为  $tr = (it, bt, ct)$ ;  $it$  即为  $ecu$  所具有的身份信任度,指  $ecu$  的关于身份认证、授权以及授权委托等身份可靠性保障机制所确定的身份合法性,其取值往往有二值性,  $it_{\alpha u_i} = (0, 1)$ ;  $bt$  即为行为信任度,指  $ecu$  在边缘计算环境中的各不同联合体中所要遵守的组织约定,既包含边缘计算环境对于边缘计算单元的特定行为约束,也包括在边缘计算元素相互间自主协同过程中所应遵循的交互规范,如共享资源时不能未经同意退出或减少共享的资源,还有移动性、持续时间等社会行为的属性,令

$$bt_{\alpha u_i} = w_1 bt_{\alpha u_i}^1 + w_2 bt_{\alpha u_i}^2 \cdots w_n bt_{\alpha u_i}^n, \text{ 其中 } \sum_{i=1}^n w_i = 1;$$



行为信任展示  $ecu$  的历史行为表现特征,该特征与时间演化过程、资源用户互评等指标一起,指示反馈信任度、演化信任度. 反馈信任度指资源提供的反馈是否真实可信,可表示为

$$bt_{ij}(t_{ecu_i}^n) = \lambda \times bt_{ij}^d(t_{ecu_i}^n) + (1 - \lambda) \times \sum_{ecu_r \in N(i)} \frac{bt_{ir}^f(t_{ecu_i}^n) \times bt_{rj}^d(t_{ecu_i}^n)}{\sum_{ecu_r \in N(i)} bt_{ir}^f(t_{ecu_i}^n)}, \quad (1)$$

其中,  $bt_{ij}^d$  为时间段  $n$  内  $ecu_i$  对  $ecu_j$  的直接信任评估,  $bt_{ir}^f$  为  $ecu_i$  对  $ecu_r$  的反馈信任度,  $N(j)$  为时间段  $n$  内和  $ecu_j$  进行交互的节点集合, 不包括  $ecu_i$ ,  $\lambda$  为信任评价的可靠性因子, 取值在  $0 \sim 1$  之间, 与发生交互的数目有关, 数目越多则  $\lambda$  取值越大. 演化信任度指边缘计算节点随时间变化的动态演进, 可通过平滑指数更新, 表示为

$$bt_{ij}^n = (1 - \beta) \times bt_{ij}^{n-1} + \beta \times bt_{ij}^n, \quad (2)$$

其中,  $bt_{ij}^n, bt_{ij}^{n-1}$  分别为第  $n, n-1$  时间段内的行为信任度,  $\beta$  为信任平滑更新因子.  $ecu_i$  的最终行为信任度  $bt_{ecu_i}^n$  则通过各多个行为信任度加权求平均可得.  $ct$  即为能力信任度, 是一组 QoS 相关指标的向量, 可以表示为

$$ct_{ecu_i} = \langle ct_{ecu_i}^1, ct_{ecu_i}^2, \dots, ct_{ecu_i}^k \rangle, \quad (3)$$

可以包括  $ecu$  在边缘计算中声称的其功能、功能相关性能属性以及提供服务所具备的可用性、可访问性、响应时间、吞吐量、可用带宽、存储空间大小、CPU 计算能力等信息. 可以设定  $U = \{ecu_i | i \in N\}$  为边缘计算元素集,  $ecu_i$  的 QoS 相关的能力隶属度矩阵定义为

$$\rho_{ecu_i} = (ct_{QoS_1}, ct_{QoS_2}, \dots, ct_{QoS_k}), \quad (4)$$

$ecu_i$  的 QoS 属性的权重系数具有归一化特征, 其向量量定义为

$$W_{ecu_i} = (W_{QoS_1}, W_{QoS_2}, \dots, W_{QoS_k}, \dots, W_{QoS_N}),$$

$$\text{s. t.} \quad \sum_{k=1}^N W_{QoS_k} = 1, \quad (5)$$

边缘计算元素  $ecu_i$  的能力信任度向量由  $ecu_i$  的 QoS 属性  $QoS_k$  的信任度向量和  $ecu_i$  的 QoS 属性的权重系数向量确定:

$$ct_{ecu_i} = W_{ecu_i} \times \rho_{ecu_i} = W_{QoS_1} ct_{QoS_1} + W_{QoS_2} ct_{QoS_2} + \dots + W_{QoS_k} ct_{QoS_k}. \quad (6)$$

实际的信任评估是非常难基于所有信息来完整、精确评估  $ecu$  的综合信任度, 要根据应用请求对于资源的信任需求情况, 将资源与请求的匹配度或相似度进行计算, 从而确定  $ecu$  是否属于某一  $ecc$  的  $ecu$  的集合, 进一步确定是否可以成为  $ecx$  的一部分

被调度去协同完成任务. 边缘计算联合体表示为  $ecc = (ECU, req)$ , 其中,  $ECU$  为面向某一特定服务请求  $req$  而查询和满足  $req$  要求的所有的边缘计算单位  $ecu$  的集合. 它们实质上一般为参与到某一互联网服务中来的边缘设备  $ed$ 、边缘服务器  $es$  或其他中心节点  $cn$ ;  $req$  为成功实现某一边缘计算应用服务对于资源的功能、服务质量、可信安全等需求的集合.

边缘计算执行体表示为  $ecx = (ECU, sat)$ , 指为响应用户的请求, 提供网络应用服务, 达到一组综合信任满意度  $sat$  要求而相互协同, 共同完成任务的一组边缘计算单元  $ecu$  的集合, 同时, 信任满意度  $sat$  是指通过服务请求的信任约束条件, 能够协同参与完成某一计算任务、服务达到的可信约束条件的程度, 单个资源或用户的满意度表示为  $sat = (is, bs, cs)$ , 信任满意度反映了  $ecu$  之间交互信息后对对方的可信任状况作出的满意度评判, 由 3 个部分组成, 其中:  $is$  为身份信任满意度, 表示请求  $ecu$  对目标  $ecu$  的真实身份、所拥有的授权或授权委托符合了自己预设的要求, 其取值为 0 或 1, 要么满意, 否则不满意;  $bs$  为请求  $ecu$  对目标  $ecu$  的行为的信任程度的一种主观感受, 它表现为对  $ecu$  的完全不信任到完全信任过程主观感受为完全不满意到完全满意, 其取值为  $0 \sim 1$  之间的任意值, 与历史的、当前的记录的行为信任度有关, 还与调度情况有关,  $ecu$  没有被调度, 则行为满意度维持不变, 如  $ecu$  被调度, 则任务处理的好坏将成为更新当前行为满意度的依据, 可以通过声誉的评价机制进行局部评价、全局评价以及基于时间的演化评估, 获得客观的评价结果. 我们通过模糊多目标决策方法确定边缘计算元素之间模糊信任度的高低, 解决边缘计算元素选择的决策问题,  $ecu_i$  关于评价等级  $ERS = \{V_j | -L \leq j \leq L\}$  的  $V_j$  的隶属度为  $\delta_{V_j}(ecu_i)$ , 记为  $\mu_{ij}$ , 决策矩阵为  $V = (\mu_{ij})_{N \times (-L, L)}$ , 模型需要考虑正、负理想方案的目标相对隶属度科学计算问题. 用综合信任度向量来表示一个边缘计算元素的可信程度能够充分反映用户的评价意见, 但也存在如何比较多个信任度向量的问题, 即当多个边缘计算元素可以满足一个  $ecc$  关于 QoE 要求时, 应该如何选择的问题. 我们采用改进的相对比值法, 该方法按照相对比值  $\xi(ecu_i)$  从大到小的顺序给出  $U = \{ecu_i | i \in N\}$  中的边缘计算元素的优劣排序.

基于上述信任保障体系结构和模型, 进一步设计边缘计算单元的信任度度量与局部信任度更新方

法,利用边缘计算联合体共享信任度信息的信任覆盖网以及全局信任度更新方法,从而实现基于综合信任的高效边缘计算。

## 8 基于综合信任度的资源管理与协同优化

### 8.1 基于综合信任的资源协同优化的思路

对于资源综合信任度进行科学客观地评估后,能够为资源、用户进行身份、行为和服务能力定量评价,为安全高效地实现资源共享调度以及整个系统优化提供了基本指标体系。不过,由于边缘计算环境中包含有多重特征的资源,计算模式可能包含多种形式的组合,需要设计资源管理与协同优化解决方案。首先,确定边缘计算应用场景和服务类型,不同的场景和应用类型决定了计算模式的组合与资源共享的方式。比如请求服务为计算密集型的任务,那么边缘服务和终端节点没有足够的资源不适合处理此类型的任务,应该将负载从边缘卸载,提交到云计算中心完成,从而获得良好的用户 QoE 效果;如果请求的任务与位置关联度高,信息交互频繁,且具有一定的计算量,那么,要求边缘服务器协同移动终端节点共同完成;如果仅仅是移动终端节点希望感知邻近环境的状态,或者自由随意的状态或信息请求,或计算请求,则只需终端节点自主协商共同完成任务即可。面对纷杂的边缘计算场景,要建立轻量级的资源信任状态维护和资源拓扑构建机制,为资源调度和协同提供支持;为了保证资源的高效利用和最低使用成本,采用基于综合信任度的资源与任务匹配度评估与之对应的资源调度算法,并通过计算场景分类以及任务卸载与装载评估与实现机制,实现计算模式的智能匹配,从而达到资源应用于合适场景;需要研究带有移动预测的 QoE 保证的资源调度机制,确保在终端节点在移动条件能在有效传输范围内将任务完成;需要研究系统过载失效的恢复机制,因为边缘计算环境的动态性比较强,部分节点失效或状态不一致是不可避免的,要设计鲁棒的资源调度算法;广泛部署的物联网终端成为边缘计算的重要组成部分,而此时数据传输往往通过无线网络完成,无线资源也成为制约和影响边缘计算性能的重要因素,因此需要研究无线频谱资源与计算资源的协同优化调度机制,从而取得更优的系统性能;边缘计算既是一种开放的系统也是可以基于经济收益驱动的系统,需要研究基于成本与收益优化的资源共享与调度算法,激励资源的充分利用。为了有效改进

和提高边缘计算的资源利用的效率,需要研究任务卸载算法、资源调度算法等的性能评价模型,进行分析,从而更好地指导调度算法的设计。

### 8.2 基于综合信任的用户 QoE 优化框架

边缘计算模型和信任保障体系研究方案确定了边缘计算系统工作的基本规范,为保障用户 QoE 效果,在模型的支持下设计边缘计算中任务卸载和资源调度的算法。首先,根据服务请求确定计算模式,结合信任保障模型,建立边缘计算资源调度问题的模型,进而根据用户请求的 QoE,实现对可用有效资源的筛选,再行设计高效的算法,进行资源调度,最后对算法进行性能评价。在这个过程中,首先要解决任务卸载的临界点的判定,即确定任务的完成是在边缘还是放在云计算中心来完成。我们根据服务请求的类型,如数据敏感型、计算密集型、带宽密集型,确定边缘对应资源是否满足,做出评判,从而确定资源共享模式。

针对任务卸载问题,对用户 QoE 进行分析,然后确定任务卸载的粒度,也可以根据任务完成时间来进行确定,在用户可以接受的 QoE 基础,以最少的支付成本完成提交的任务。如果边缘设备是智能移动终端,我们考虑移动性对于卸载任务的影响,尽量确保一次移动切换之内完成,从而获得高的性价比。任务卸载需要考虑同云计算透明访问提高系统效能。

对于确定在边缘完成的服务请求,则在边缘进行资源基于信任的调度。边缘服务器具有较好的计算能力、宽带连接能力、一定的存储能力。终端节点具有移动性、动态性、资源有限性以及地理的邻近性。有些任务需要边缘服务器提供计算资源的支持,同时结合终端节点的计算和服务能力;有些任务的完成则只需要终端节点的自主协同,这种系统过程既要满足用户 QoE 要求,同时,还需要考虑移动终端节点移动性、服务持续时间、电量与能耗诸多因素。其资源的调度与迁移难度很大,问题的建模也相对更加复杂。本文充分借鉴网格计算、P2P 计算领域优秀的资源调度算法,同时可以结合 P2P 环境中基于博弈的系统平衡优化的方法<sup>[106]</sup>。

边缘计算系统的资源共享与协同优化问题如图 12 所示,可以看出其实边缘计算的资源共享与协同问题是一个非常复杂的问题,我们将其设计为一个优化的框架,问题可以规约成一个多约束多目标优化问题,对于这类问题已经有很多研究,在实际中如此复杂的问题有时难于求解,对应到实际应用场景时,问题求解过程中的输入、约束因素、优化目标都

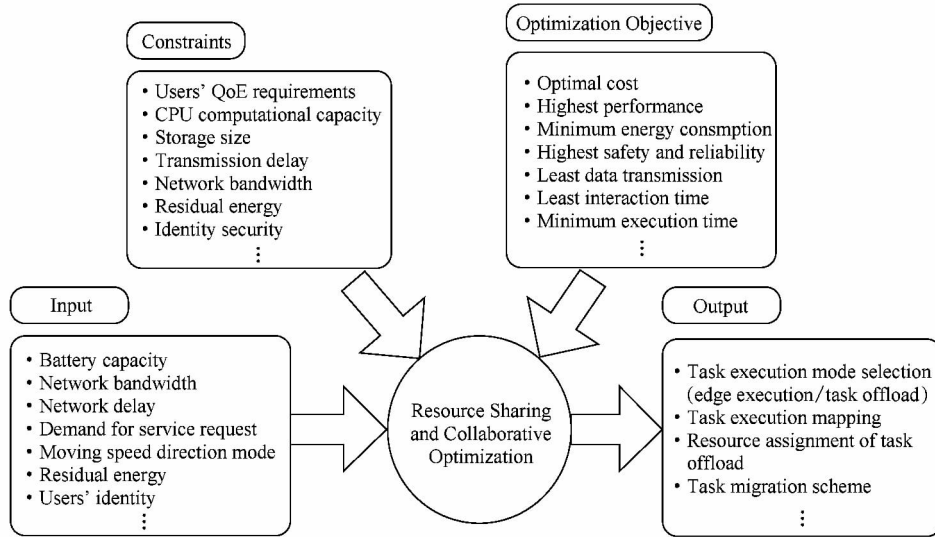


Fig. 12 Resource sharing and cooperation optimizing problem in edge computing

图 12 边缘计算系统的资源共享与协同优化问题示意图

会进一步地明确和简单化,因此问题求解相对变得可行.目前可以采用资源调度优化算法方面的启发式/非启发式算法方法已有很多,本文考虑实际情况设计系列任务卸载评判、卸载资源分配、边缘资源调度算法满足信任安全约束和成本约束条件去优化用户 QoE 效果.

基于综合信任的资源调度可以描述为一个多维目标优化问题,边缘计算执行体  $ecx = (ECU, sat)$ ,边缘计算中,部分  $ecu$  产生的服务请求  $T = \{t_1, t_2, \dots\}$ ,问题的目标是在可信决策空间  $sat(is, bs, cs)^T$  分别对应身份信任满意度、行为信任满意度、能力信任满意度,以  $f_1(is), f_2(bs), f_3(cs)$  作为可信优化调度的目标函数,分别代表身份可信满意度函数,能力可信满意度函数和行为可信满意度函数.因此,希望在边缘计算元素可信的可行集  $(is, bs, cs)$  上得到满足条件的优化调度方案  $\max f_1(is), \max f_2(bs), \max f_3(cs)$ ,边缘计算元素可信映射目标函数  $f_i(sat_i)$ ;对于某个可信指标  $i$ ,某一个提供服务的边缘计算元素决策空间为  $sat(is, bs, cs)^T$  经过服务可信映射目标函数  $f_i(sat_i)$  映射后可得到该维目标的值.问题的求解我们可以采用约束法、加权法等.由于约束法相当于穷尽求解过程,不宜于实时的调度系统,但往往能获得比较好的优化效果;对于动态变化的实际边缘系统则可以采用加权法.加权法的一般形式为

$$\max \sum_{j=1}^3 \omega_j^o f_j(sat), f_j(sat \geq \epsilon_j), \\ sat \in SAT, \quad (7)$$

其中,  $\omega_j^o$  为根据调度目标给定的一组权值,它反映

了请求对每个优化目标的权重,  $\epsilon_j \geq (sat_{i1}, sat_{i2}, sat_{i3})$  是第  $j$  维目标函数必须满足的决策空间约束,可以实现资源面向身份信任、行为信任、能力信任(多维 QoS 指标关联)的综合信任在用户需求的 QoE 约束下使系统的加权目标函数最大.对于多目标优化问题的研究表明,多目标规划一般不存在所有目标函数共同的极大点,因此可以采用基于综合信任求解非劣解.对于边缘计算执行体的可信优化调度同样不可能存在一个所有目标函数共同的极大点,需要根据用户 QoE 要求和资源 QoS 以及身份、行为信任的决策空间和权值向量预设,调度最优的边缘计算元素.

## 9 基于综合信任的资源优化调度的实例

网络边缘产生的数据正在急速增加,移动设备用户不仅是数据的消费者,也是数据的生产者.数据的产生和处理往往是实时和持续不断地进行,流处理在当前的大数据时代扮演着一个重要的角色.它可以在数据可用时对大量数据进行分析,并支持实时决策.在流处理中,数据流是分布在各个地理位置的多个源生成的连续的数据流.流处理的常见方法是将原始数据流传输到数据中心进行统一处理.然而,对于流处理应用,在网络边缘处理数据可以明显降低应用程序的响应延迟,提升用户体验.在网络边缘,我们利用移动的边缘节点来执行流处理应用.为更好说明算法的效果,我们将节点的身份信任和行为信任假定其取值都相同,提出了一种适用于移动

边缘环境的调度算法——移动资源感知 (mobile resource awareness, MRA) 调度算法. 我们提出基于资源的多维空间距离的请求与服务的匹配方法, MRA 能够根据智能终端移动和位置状态以及资源需求状态在边缘计算环境中找到具备合适的计算、存储、通信资源的节点进行协同, 完成对应的流任务的处理; 最后, 我们在 Storm 中实现了我们的调度算法, 在 Core 平台中进行了边缘节点的部署, 通过与其他调度算法相比, MRA 算法可以更有效地适应节点资源有限的移动边缘环境, MRA 算法优于其他 2 种调度算法.

### 9.1 问题描述和定义

由于流处理系统主要是主-从架构, 对于流处理应用, 边缘服务器可以看作是接收用户请求和分配任务的主节点, 同时边缘节点 (如智能手机、路由器等) 可以看作是执行任务的从节点. 对于边缘节点, 存在一些特征, 例如可用资源有限、节点可能是移动的等. 在这种移动边缘环境中, 我们需要一个调度算法来调度流处理任务, 同时可以提高整个流处理系统的吞吐量.

我们试图解决的问题是如何最好地将任务分配给节点. 每个任务都有一组特定的资源需求, 每个节点都有一组可用资源, 同时每个节点根据一个移动模型移动. 考虑到这些资源需求、资源可用性以及节点的移动模型, 从而提高流处理系统的吞吐量, 同时满足每个任务的资源需求. 因此, 这个问题就变成了如何将任务映射到节点, 以便满足每个任务的资源需求, 同时不能超出节点资源的可用性的问题.

我们首先考虑了 2 种不同类型的资源: CPU 和内存. 我们将 CPU 资源定义为可用的 CPU 百分比, 内存资源被定义为可用的兆字节数量. 我们将节点资源分为 2 类: 1) 必须完全满足的强制约束资源; 2) 不需要完全满足的松散约束资源. 对于松散约束, 我们的目标是尽量减少超过约束的资源数量. 划分强制性和宽松约束的基础是: 强制性约束的资源过度使用会导致系统性能显著下降, 而宽松性约束的资源过度使用则不会导致系统性能显著下降. 在本文中, CPU 被认为是一种具有松散约束的资源, 因为它们可以被过载; 内存被认为是一种具有强制性约束的资源, 因为我们不能超过节点上可用的内存总数.

进一步, 由于流数据协同处理需要将数据传送到参与协同的节点, 而在节点移动环境下, 节点间距离会因为边缘节点的移动性而动态变化, 从而影响

到数据传输的无线信道和传输的带宽. 我们的调度算法还需要考虑节点之间的距离变化带来的影响. 图 13 描述了移动边缘环境中执行流处理任务的示例. 在图 13 中, 流处理系统由一些边缘节点组成. 用户提交拓扑之后, 流处理系统根据调度器调度拓扑中的任务.

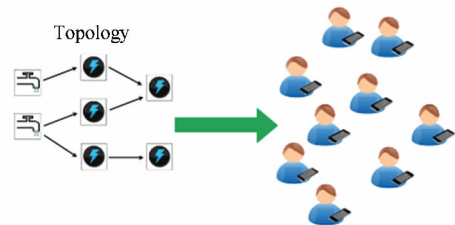


Fig. 13 Streaming tasks processing in edge computing

图 13 流处理任务在边缘环境中分配

我们假设每个节点对这些资源有特定的有限预算. 同样, 每一个任务指定每种资源所需求数量. 设  $T = \{\alpha_1, \alpha_2, \alpha_3, \dots\}$  为拓扑中所有任务的集合. 每个任务  $\alpha_i$  有 CPU 资源需求  $c_{\alpha_i}$ , 以及内存需求  $m_{\alpha_i}$ . 同样, 设  $N = \{n_1, n_2, n_3, \dots\}$  为所有节点的集合, 所有节点对于 CPU 和内存的总预算分别为  $W_1$  和  $W_2$ . 另外, 节点根据移动模型移动, 为了减少网络延迟, 执行任务的节点应该彼此接近. 为了定量建模, 我们将汇聚节点的吞吐量贡献  $Q_{n_i}$  定义为在节点  $n_i$  上元组的处理速率. 我们的目标是将任务分配给一个节点的子集, 通过最大化资源利用率和最小化节点之间的距离来增加流处理系统的总吞吐量, 同时该调度也不能超出节点总的可用预算  $W_1$  和  $W_2$ .

$$\begin{aligned} & \max_{(N' \subseteq N)} \sum_{n_i \in N'} Q_{n_i} \\ & \text{s. t. } \sum_{\alpha_i \in T} c_{\alpha_i} \leq W_1, \\ & \sum_{\alpha_i \in T} m_{\alpha_i} \leq W_2 \end{aligned} \quad (8)$$

这种选择和分配问题是背包优化问题的复杂和特殊变化. 为了解决各种背包问题, 已经提出了一些算法, 如动态规划<sup>[107-108]</sup>、树搜索<sup>[109-110]</sup>、近似算法<sup>[111]</sup>等. 然而, 这些算法的时间复杂度很高, 需要大量时间来计算分布式数据流系统中的可用调度, 不适宜于边缘计算场景. 由于诸如 Storm 之类的流处理系统需要尽可能接近实时地响应事件, 所以需要快速、简单地进行调度决策. 计算调度所需的时间越长, 应用程序的停机时间就越长. 此外, 如果 Storm 集群中发生故障, 需要重新调度任务, 则调度程序必须能够快速生成新的调度. 如果任务不能被快速地重



新安排,整个拓扑可能会被堵塞,甚至会发生消息队列溢出等灾难性故障.因此,我们将所有任务调度到多个节点,同时满足所有任务的资源需求,另外,这个调度算法应该在彼此靠近的节点上调度任务,以减少网络延迟.该算法必须快速、简单,以满足流处理应用的实时性要求.

## 9.2 MRA 算法的设计

存在由一组任务 $\{\alpha_1, \alpha_2, \alpha_3, \dots\}$ 组成的集合 $T$ ,以及一组节点 $\{n_1, n_2, n_3, \dots\}$ .对于每个任务,我们需要从 $\{n_1, n_2, n_3, \dots\}$ 中找到一个节点来执行任务.我们通过计算节点和任务之间的匹配度来选择节点,这个匹配度如图13中的 $distance$ 所示.在图13的3维坐标系中, $x$ 轴表示任务 $\alpha_i$ 的内存需求和节点 $n_j$ 的可用内存之间的差值, $y$ 轴表示任务 $\alpha_i$ 的CPU需求与节点 $n_j$ 的可用CPU之间的差值, $z$ 轴表示节点 $n_j$ 和参考节点(在节点选择部分讨论)在2维平面上的欧氏距离,距离近则相互间的通信信道好,可用带宽高.在图14中,点 $A_{ij}(x, y, z)$ 和原点 $O$ 之间的3维欧氏距离表示任务 $\alpha_i$ 和节点 $n_j$ 之间的匹配程度.如果这个3维欧氏距离越小,则任务 $\alpha_i$ 与节点 $n_j$ 之间的匹配程度较大.

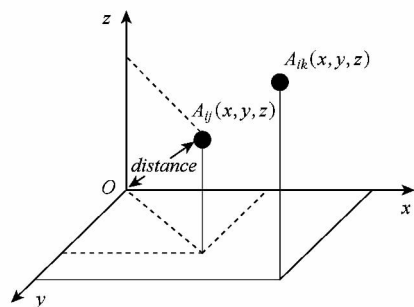


Fig. 14 The degree of matching between  $\alpha_i$  and  $n_j$

图14  $\alpha_i$  与  $n_j$  之间的匹配程度示例

MRA 算法处理过程如下:获得任务选择过程中需要被调度的任务列表,将这个有序任务列表中的每个任务分配到一个合适的任务执行节点,而这个任务分配过程中需要进行任务的CPU资源、存储资源、网络通信的资源需求与节点对应的可用资源的匹配评估;完成分配和任务计算,并且同步更新节点的可用资源(限于篇幅算法伪代码就不列出).

### 9.2.1 任务选择

任务选择首先是对流处理组件排序.在Storm的术语中,流处理图被称为 topology,其中包含2种类型的组件,spout 和 bolt. spout 是流处理图的源节点,它从外部数据源读取数据,然后将数据转换成

元组发送到 topology 中. bolt 消耗从 spout 或者其他 bolt 发送过来的元组,它可以是一个操作者节点也可以是一个汇聚节点.拓扑中的每个节点都可以根据其并行性参数执行一个或多个任务.通过遍历拓扑并对所有组件进行排序.组件排序的目的是尽可能将相互通信的组件调度到同一个节点上,这可以减少组件之间的通信延迟.我们从一个 spout 组件开始遍历拓扑,使用广度优先搜索(BFS)来遍历拓扑,其中相邻组件将被放置在一起.组件排序之后,再对任务进行排序.首先,我们得到所有组件的最大任务数,这代表每个组件需要迭代的次数.对于从有序的组件列表迭代出来的每个组件,我们从该组件获取一个任务,并将这个任务添加到任务列表中.然后,我们执行下一次迭代并且不断迭代直到这个最大值减到零为止.

### 9.2.2 节点选择

在获得有序的任务列表之后,我们需要将任务安排到节点上.如果需要调度的任务是任务列表中的第1个任务,则参考节点为空,那首先需要确定这个参考节点.参考节点是指节点集合中最靠近其他节点的节点.由于边缘节点可能是移动的,通过从 $\{n_1, n_2, n_3, \dots\}$ 中选择离其他节点最近节点,我们可以最大化传输带宽并加快元组的处理速度.在找到参考节点之后,我们可以计算得到如图13所示的可行节点和原点之间的匹配距离.注意:为了消除内存,CPU 和距离  $d$  之间不同衡量标准和量纲的影响,我们对内存,CPU 和距离  $d$  进行了归一化.我们得到匹配距离之后,将匹配距离  $distance_{n_j}$  和节点  $n_j$  存储到  $TreeMap$  中.循环完成之后,  $TreeMap$  值的集合的第1个元素就是我们正在寻找的  $selectedNode$ .同时,为了确保执行任务的节点可以彼此靠近,我们需要将参考节点更新为  $selectedNode$ .然后再完成任务的调度传输和处理,从而完成整个流处理过程.

## 9.3 调度算法的实验与分析

### 9.3.1 MRA 算法的实验设置

我们的目标是在一个模拟的边缘计算环境中评估我们的调度算法,在这个边缘计算环境中,Storm 集群由来自网络边缘的多个移动边缘节点组成.我们使用 CORE<sup>[112]</sup>来运行我们的实验. CORE 是美国海军研究实验室(NRL)制造的网络模拟工具,可以用来模拟由多个节点组成的网络.在 CORE 中,用户可以使用各种不同类型的虚拟节点来创建自定义网络拓扑. CORE 中的虚拟节点包括 PC、服务器、路

由器、交换机等<sup>[95]</sup>. 节点可以通过有线链路或无线网络进行连接. 有线链路可以配置诸如最大带宽, 延迟和抖动等参数. 对于无线网络, 我们可以使用可扩展的移动自组织网络仿真器(EMANE)模块来模拟真实的无线网络场景.

我们进行了 2 组实验: 组 1 实验的实验设置如图 15 所示. 在组 1 实验中, Storm 集群由 13 个节点组成. 一个节点被指定为主节点, 即运行 Storm Nimbus 和 Zookeeper, 而其他 12 个节点是从节点.

这些节点通过无线网络连接, 无线网络模型是 EMANE 中的 IEEE802.11abg 模型. 在实验过程中, 主节点保持静止, 其他工作节点根据随机路径点模型<sup>[113]</sup>进行移动. 我们的实验评估受到雅虎 Storm 性能测试<sup>[114]</sup>的启发. 整个实验中使用的拓扑是线性拓扑, 拓扑的任务数量设置为 60 个.

组 2 实验的实验设置如图 16 所示. 节点数量为 7 个, 其中一个节点为主节点, 其他 6 个节点为从节点, 采用线性拓扑, 任务数量设置为 30 个.

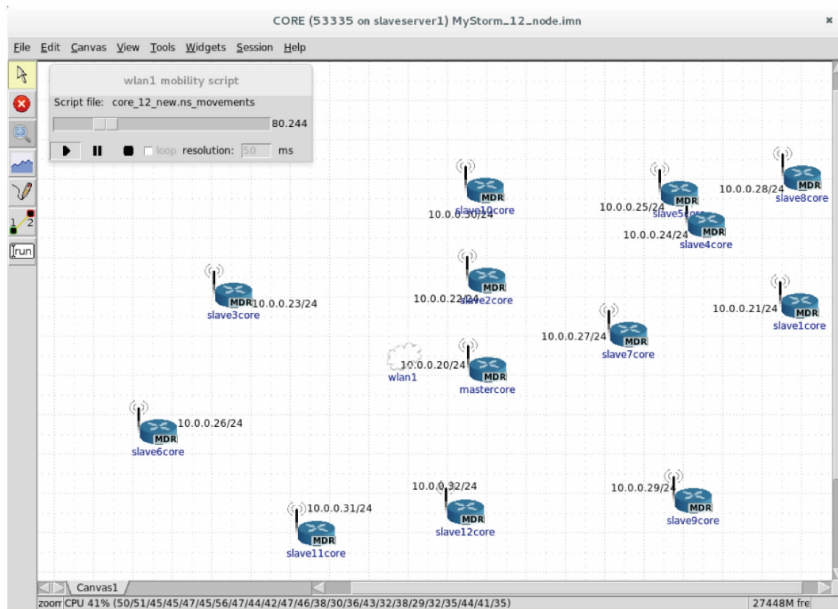


Fig. 15 The experimental setup of the first experiments

图 15 组 1 实验的实验设置

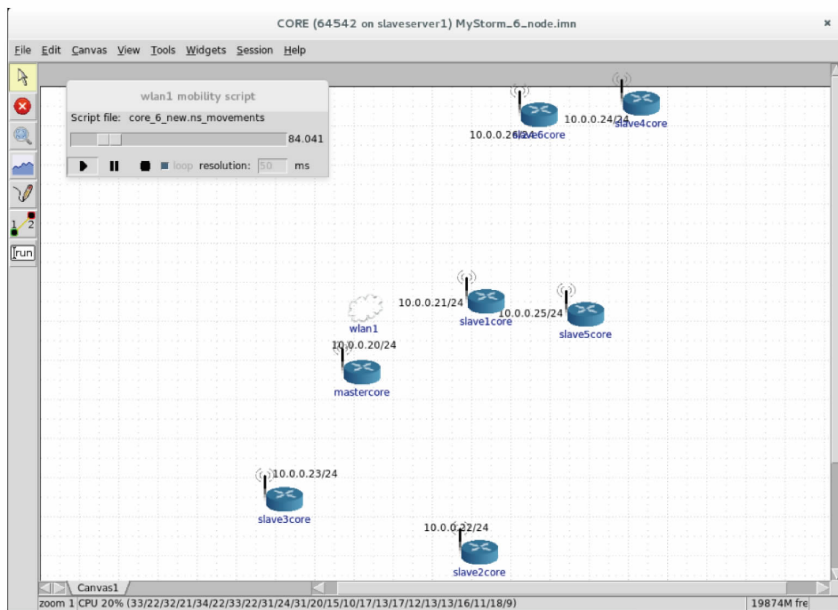


Fig. 16 The experimental setup of the second experiments

图 16 组 2 实验的实验设置

### 9.3.2 调度算法的实验结果

我们通过比较拓扑的总吞吐量和单位吞吐量来评估 MRA 算法的性能. 对于我们的评估, 拓扑的总吞吐量是输出 bolt 处理的元组总数, 拓扑的单位吞吐量是输出 bolt 每 10 s 处理的元组数量. 我们通过将我们的算法与 Storm 中的默认调度算法以及 RAS 调度算法<sup>[115]</sup>进行比较来评估我们的算法的性能.

在组 1 实验中, 1) 单位吞吐量. 图 17 显示了组 1 实验中拓扑的单位吞吐量指标的比较. 从图 17 中可以看出, 在大约 98% 的时间内, 我们的 MRA 算法的单位吞吐量是高于其他 2 种算法的, 其他调度算法在一小段时间内可能会高于我们的算法. 出现这种现象的原因是节点在不断移动, 在短时间内, 单位吞吐量可能会出现激增的情况. 另外, 默认调度算法的平均单位吞吐量约为 16 000, RAS 调度算法约为 13 000, MRA 算法约为 39 000. 从这些数据可以

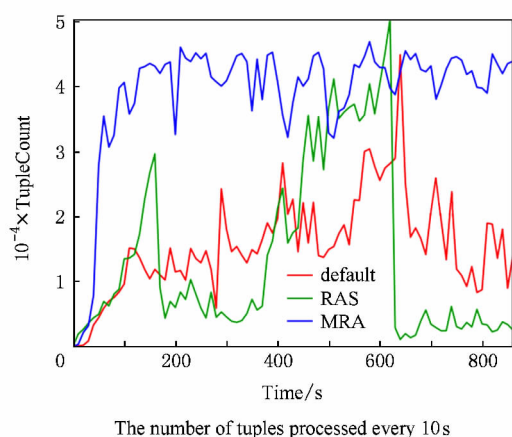


Fig. 17 Comparison of unit throughput in the first group of experiments

图 17 组 1 实验中单位吞吐量的比较

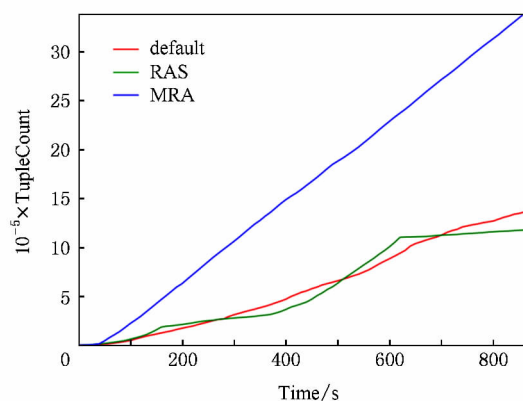


Fig. 18 Comparison of total throughput in the first group of experiments

图 18 组 1 实验中总吞吐量的比较

看出, 在元组处理速度方面, 我们的 MRA 算法比其他 2 个调度算法都要快很多. 2) 总吞吐量. 拓扑的总体吞吐量如图 18 所示. 从图 18 中可以看出, 我们的 MRA 算法明显优于其他 2 种调度算法, 并且随着时间的推移, MRA 的优势将变得越来越明显.

在组 2 实验中, 1) 单位吞吐量. 图 19 显示了组 2 实验中拓扑的单位吞吐量指标的比较. 从图 19 中可以看出, 和组 1 实验的实验结果类似, 我们的 MRA 算法明显优于其他 2 种算法. 另外, 默认调度算法的平均单位吞吐量约为 27 000, RAS 调度算法约为 20 000, MRA 算法约为 43 000, 大约是其 2 种调度算法的 2 倍. 2) 总吞吐量. 第 2 组实验中拓扑的总体吞吐量如图 20 所示. 从图 20 中可以看出, 我们的 MRA 算法在节点数量较少的情况下也是明显优于其他 2 种调度算法的.

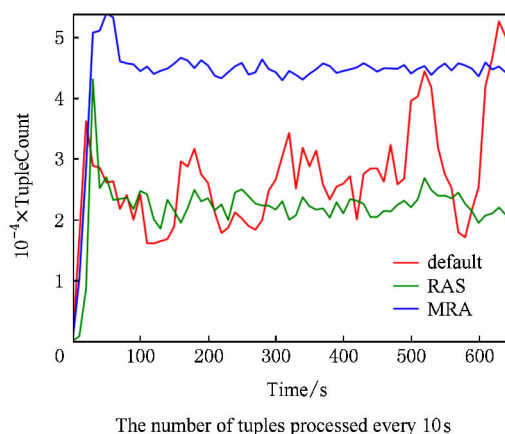


Fig. 19 Comparison of unit throughput in the second group of experiments

图 19 组 2 实验中单位吞吐量的比较

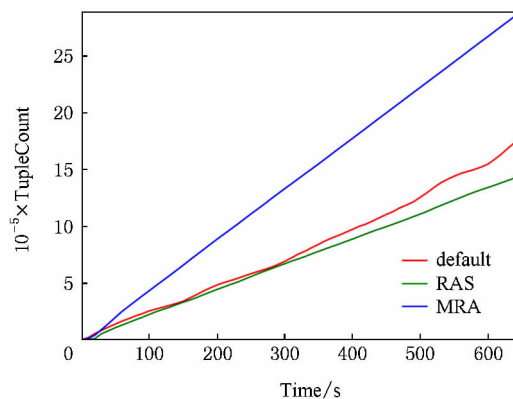


Fig. 20 Comparison of total throughput in the second group of experiments

图 20 组 2 实验中总吞吐量的比较

从实验结果可以看出 MRA 算法可以在多个可

用的边缘节点中根据资源的动态性,总是选取最适合的节点来完成流,从而获得更好的性能。

## 10 总 结

本文通过对计算模式演化发展的特征与规律的分析,结合物联网、移动互联网以及电子技术的发展指出边缘计算兴起的缘由。探讨和对比了主流的边缘计算的概念和定义;结合当前业内研究进展,分析了边缘计算的特征,归纳出边缘计算主要的应用场景,并分类详细调研了边缘计算的研究现状,指出了边缘计算的挑战与前景。本文针对边缘计算的隐私信任和安全以及资源协同优化的关键问题,提出了基于综合信任的边缘计算多层自适应的边缘计算模型与体系结构,并基于综合信任对边缘计算的资源协同优化问题进行了建模,形成了统一的多目标优化解决方案。最后通过边缘计算场景中的流计算任务面向计算、存储以及动态移动的通信资源进行任务调度,优化了资源共享效率,仿真实验验证了方法的有效性,表明该方法能够提升用户的 QoE。

## 参 考 文 献

- [1] Armbrust M, Fox A, Griffith R, et al. A view of cloud computing [J]. *Communications of the ACM*, 2010, 53(4): 50-58
- [2] Montresor A. Reflecting on the past, preparing for the future: From peer-to-peer to edge-centric computing [C] // *Proc of the 36th IEEE Int Conf on Distributed Computing Systems (ICDCS)*. Piscataway, NJ: IEEE, 2016: 22-23
- [3] Zeng Jiandian, Wang Tian, Jia Weijia, et al. A survey on sensor-cloud [J]. *Journal of Computer Research and Development*, 2017, 54(5): 925-939 (in Chinese)  
(曾建电, 王田, 贾维嘉, 等. 传感云研究综述[J]. *计算机研究与发展*, 2017, 54(5): 925-939)
- [4] Yu Ruiyun, Wang Pengfei, Bai Zhihong, et al. Participatory sensing: People-centric smart sensing and computing [J]. *Journal of Computer Research and Development*, 2017, 54(3): 457-473 (in Chinese)  
(于瑞云, 王鹏飞, 白志宏, 等. 参与式感知: 以人为中心的智能感知与计算[J]. *计算机研究与发展*, 2017, 54(3): 457-473)
- [5] Luan T H, Gao L, Li Z, et al. Fog computing: Focusing on mobile users at the edge [EB/OL]. [2017-10-23]. <http://sitic.org/wp-content/uploads/Fog-Computing.pdf>
- [6] Bonomi F, Milito R, Zhu Jiang, et al. Fog computing and its role in the Internet of things [C] // *Proc of Edition of the MCC Workshop on Mobile Cloud Computing*. New York: ACM, 2012: 13-16
- [7] Dastjerdi A V, Buyya R. Fog computing: Helping the Internet of things realize its potential [J]. *Computer*, 2016, 49(8): 112-116
- [8] Satyanarayanan M, Bahl P, Cáceres R, et al. The case for VM-based cloudlets in mobile computing [J]. *IEEE Pervasive Computing*, 2009, 8(4): 14-23
- [9] Gai Keke, Qiu Meikang, Zhao Hui, et al. Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing [J]. *Journal of Network & Computer Applications*, 2016, 59(C): 46-54
- [10] Satyanarayanan M, Chen Zhuo, Ha K, et al. Cloudlets: At the leading edge of mobile-cloud convergence [C] // *Proc of the 6th Int Conf on Mobile Computing, Applications and Services*. Los Alamitos, CA: IEEE Computer Society, 2014: 1-9
- [11] Satyanarayanan M. Edge computing: A new disruptive force [EB/OL]. [2017-10-23]. <http://acm-ieee-sec.org/2017/2017-10-12-sec2017-keynote.html>
- [12] Lopez P G, Montresor A, Epema D, et al. Edge-centric computing: Vision and challenges [J]. *ACM SIGCOMM Computer Communication Review*, 2015, 45(5): 37-42
- [13] Boulevard B. Edge Computing [OL]. [2017-10-23]. <http://vis.pnnl.gov/pdf/fliers/EdgeComputing.pdf>
- [14] National Science Foundation. NSF/Intel partnership on information-centric networking in wireless edge networks (ICN-WEN) [EB/OL]. [2017-10-23]. <https://www.nsf.gov/pubs/2016/nsf16586/nsf16586.html>
- [15] Yuan Yixue. Cloud computing is heating up, edge computing is up now [N]. *China Science*, 2016-12-16(4) (in Chinese)  
(袁一雪. 云计算正热, 边缘计算又起[N]. *中国科学报*, 2016-12-16(4))
- [16] 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*, 2014, 44(5): 27-32
- [17] Shi Weisong, Sun Hui, Cao Jie, et al. Edge computing—An emerging computing model for the Internet of everything era [J]. *Journal of Computer Research and Development*, 2017, 54(5): 907-924 (in Chinese)  
(施巍松, 孙辉, 曹杰, 等. 边缘计算: 万物互联时代新型计算模型[J]. *计算机研究与发展*, 2017, 54(5): 907-924)
- [18] Renkonen K O, Seppala M. Edge analytics in the Internet of things [J]. *IEEE Pervasive Computing*, 2015, 14(2): 24-31
- [19] Satyanarayanan M. A brief history of cloud offload: A personal journey from odyssey through cyber foraging to cloudlets [J]. *Getmobile Mobile Computing & Communications*, 2015, 18(4): 19-23
- [20] Antipolis S. Multi-access edge computing [OL]. [2017-10-23]. <http://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing>



- [21] Garvelink B. Mobile edge computing: A building block for 5G [EB/OL]. [2017-10-23]. <https://www.telecompaper.com/background/mobile-edge-computing-a-building-block-for-5g-1092281>
- [22] Antipolis S. Mobile edge computing—A key toward 5G [EB/OL]. [2016-04-26]. <http://www.etsi.org/technologies/clusters/technologies-edge-computing>
- [23] Yu Yifan, Ren Chunming, Yuan leifeng, et al. Analysis of mobile edge computing research [J]. Telecommunication Network Technology, 2016, 11: 59-62 (in Chinese)  
(俞一帆, 任春明, 阮磊峰, 等. 移动边缘计算技术发展浅析 [J]. 电信网技术, 2016, 11: 59-62)
- [24] Dinh H T, Lee C, Niyato D, et al. A survey of mobile cloud computing: Architecture, applications, and approaches [J]. Wireless Communications & Mobile Computing, 2013, 13 (18): 1587-1611
- [25] Xie K, Wang X, Xie G, et al. Distributed multi-dimensional pricing for efficient application offloading in mobile cloud computing [EB/OL]. [2017-10-23]. <http://ieeexplore.ieee.org/document/7792154/>
- [26] Byrne J, Casari P, Eardley P, et al. Reliable capacity provisioning for distributed cloud/edge/fog computing applications [C] //Proc of the 26th European Conf on Networks and Communications. Piscataway, NJ: IEEE, 2017: 1-6
- [27] Yi Shanhe, Li Cheng, Li Qun. A survey of fog computing: Concepts, applications and issues [C] //Proc of the 2015 Workshop on Mobile Big Data. New York: ACM, 2015: 37-42
- [28] Bonomi F, Milito R, Zhu Jiang, et al. Fog computing and its role in the Internet of things [C] //Proc of Edition of the MCC Workshop on Mobile Cloud Computing. New York: ACM, 2012: 13-16
- [29] Zhang Zhaosheng, Li Shuyu. A survey of computational offloading in mobile cloud computing [C] //Proc of the 4th IEEE Int Conf on Mobile Cloud Computing, Services, and Engineering. Piscataway, NJ: IEEE, 2016: 81-82
- [30] Chiang M, Zhang Tao. Fog and IoT: An overview of research opportunities [J]. IEEE Internet of Things Journal, 2017, 3(6): 854-864
- [31] Shuja J, Gani A, Rehman M H U, et al. Towards native code offloading based MCC frameworks for multimedia applications: A survey [J]. Journal of Network & Computer Applications, 2016, 75(C): 335-354
- [32] Yaqoob I, Ahmed E, Gani A, et al. Mobile ad hoc cloud: A survey [J]. Wireless Communications & Mobile Computing, 2016, 16(16): 2572-2589
- [33] Networking C V. Cisco global cloud index: Forecast and methodology, 2014—2019 [EB/OL]. [2017-10-23]. <https://docmia.com/d/285608>
- [34] MacGillivray C, Turner V, Yesner R, et al. IDC futurescape: Worldwide Internet of things 2016 predictions [OL]. [2017-10-23]. <http://www.idc.com/research/viewtoc.jsp?containerId=259856>
- [35] Evans D. The Internet of things: How the next evolution of the Internet is changing everything [EB/OL]. (2011-01-01) [2017-10-23]. [https://www.cisco.com/c/dam/en\\_us/about/ac79/docs/innov/IoT\\_IBSG\\_0411FINAL.pdf](https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf)
- [36] Shi Weisong, Cao Jie, Zhang Quan, et al. Edge computing: Vision and challenges [J]. IEEE Internet of Things Journal, 2016, 3(5): 637-646
- [37] Shi W, Dustdar S. The promise of edge computing [J]. Computer, 2016, 49(5): 78-81
- [38] Lopez P G, Montresor A, Epema D, et al. Edge-centric computing: Vision and challenges [J]. ACM SIGCOMM Computer Communication Review, 2015, 45(5): 37-42
- [39] Rudenko A, Reiher P, Popek G J, et al. Saving portable computer battery power through remote process execution [J]. ACM Sigmobile Mobile Computing & Communications Review, 1998, 2(1): 19-26
- [40] Yi Shanhe, Hao Zijiang, Qin Zhengrui, et al. Fog computing: Platform and applications [C] //Proc of the 3rd IEEE Workshop on Hot Topics in Web Systems and Technologies. Piscataway, NJ: IEEE, 2015: 73-78
- [41] Ha K, Chen Zhuo, Hu Wenlu, et al. Towards wearable cognitive assistance [C] //Proc of the 12th Int Conf on Mobile Systems. New York: ACM, 2014: 68-81
- [42] Chun B G, Ihm S, Maniatis P, et al. CloneCloud: Elastic execution between mobile device and cloud [C] //Proc of the 6th European Conf on Computer Systems. New York: ACM, 2011: 301-314
- [43] Hunt G C, Scott M L. The Coign automatic distributed partitioning system [C] //Proc of EDOC '98. Piscataway, NJ: IEEE, 1999: 252-262
- [44] Kumar K, Lu Y H. Cloud computing for mobile users: Can offloading computation save energy? [J]. Computer, 2010, 43(4): 51-56
- [45] Kosta S, Aucinas A, Hui P, et al. ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading [C] //Proc of IEEE INFOCOM 2012. Piscataway, NJ: IEEE, 2012: 945-953
- [46] Zhang Huyin, Wang Jing, Tang Xing. Joint routing and scheduling in cognitive radio vehicular ad hoc networks [J]. Journal of Computer Research and Development, 2017, 54 (11): 2445-2455 (in Chinese)  
(张沪寅, 王菁, 唐星. 认知无线车载自组织网络中的联合路由调度 [J]. 计算机研究与发展, 2017, 54(11): 2445-2455)
- [47] Farris I, Militano L, Nitti M, et al. MIFaaS: A mobile-IoT federation-as-a-service model for dynamic cooperation of IoT cloud providers [J]. Future Generation Computer Systems, 2017, 70: 126-137

- [48] Satria D, Park D, Jo M. Recovery for overloaded mobile edge computing [J]. *Future Generation Computer Systems*, 2017, 70: 138-147
- [49] Gomes A S, Sousa B, Palma D, et al. Edge caching with mobility prediction in virtualized LTE mobile networks [J]. *Future Generation Computer Systems*, 2017, 70: 148-162
- [50] Tärneberg W, Mehta A, Wadbro E, et al. Dynamic application placement in the Mobile Cloud Network [J]. *Future Generation Computer Systems*, 2017, 70: 163-177
- [51] Li Wei, Santos I, Delicato F C, et al. System modelling and performance evaluation of a three-tier Cloud of Things [J]. *Future Generation Computer Systems*, 2017, 70: 104-125
- [52] Gosain A, Berman M, Brinn M, et al. Enabling campus edge computing using GENI racks and mobile resources [C] // *Proc of the 1st IEEE/ACM Symp on Edge Computing*. Piscataway, NJ: IEEE, 2016: 41-50
- [53] Liu P, Willis D, Banerjee S. ParaDrop: Enabling lightweight multi-tenancy at the network's extreme edge [C] // *Proc of IEEE/ACM Symp on Edge Computing*. Piscataway, NJ: IEEE, 2016: 50-57
- [54] Bhardwaj K, Shih M W, Agarwal P, et al. Fast, scalable and secure onloading of edge functions using AirBox [C] // *Proc of IEEE/ACM Symp on Edge Computing*. Piscataway, NJ: IEEE, 2016: 14-27
- [55] Nastic S, Truong H L, Dustdar S. A middleware infrastructure for utility-based provisioning of IoT cloud systems [C] // *Proc of IEEE/ACM Symp on Edge Computing*. Piscataway, NJ: IEEE, 2016: 28-40
- [56] Amento B, Balasubramanian B, Hall R J, et al. FocusStack: Orchestrating edge clouds using location-based focus of attention [C] // *Proc of IEEE/ACM Symp on Edge Computing*. Piscataway, NJ: IEEE, 2016: 179-191
- [57] Orsini G, Bade D, Lamersdorf W. Computing at the mobile edge: Designing elastic android applications for computation offloading [C] // *Proc of the 8th IFIP Wireless and Mobile Networking Conf (WMNC)*. Piscataway, NJ: IEEE, 2015: 112-119
- [58] Bonomi F, Milito R, Natarajan P, et al. *Big Data and Internet of Things: A Roadmap for Smart Environments* [M]. Berlin: Springer, 2014: 169-186
- [59] Grieco R, Malandrino D, Scarano V. A scalable cluster-based infrastructure for edge-computing services [J]. *World Wide Web*, 2006, 9(3): 317-341
- [60] Bhardwaj K, Sreepathy S, Gavrilovska A, et al. ECC: Edge cloud composites [C] // *Proc of the 2nd IEEE Int Conf on Mobile Cloud Computing, Services, and Engineering*. Piscataway, NJ: IEEE, 2014: 38-47
- [61] Lyu X, Ni W, Tian H, et al. Optimal schedule of mobile edge computing for Internet of things using partial information [J]. *IEEE Journal on Selected Areas in Communications*, 2017, 35(11): 2606-2615
- [62] Rimal B P, Van D P, Maier M. Mobile edge computing empowered fiber-wireless access networks in the 5G era [J]. *IEEE Communications Magazine*, 2017, 55(2): 192-200
- [63] Luong N C, Xiong Zehui, Wang Ping, et al. Optimal auction for edge computing resource management in mobile blockchain networks: A deep learning approach [EB/OL]. [2017-10-23]. <https://arxiv.org/abs/1711.02844>
- [64] Salsano S, Chiaraviglio L, Blefari-Melazzi N, et al. Toward superfluid deployment of virtual functions: Exploiting mobile edge computing for video streaming [C] // *Proc of the 29th IEEE Int Teletraffic Congress (ITC 29)*. Piscataway, NJ: IEEE, 2017: 48-53
- [65] Flores H, Srirama S N, Buyya R. Computational offloading or data binding? bridging the cloud infrastructure to the proximity of the mobile user [C] // *Proc of the 2nd IEEE Int Conf on Mobile Cloud Computing, Services, and Engineering*. Piscataway, NJ: IEEE, 2014: 10-18
- [66] You Changsheng, Huang Kaibin, Chae H, et al. Energy-efficient resource allocation for mobile-edge computation offloading [J]. *IEEE Trans on Wireless Communications*, 2017, 16(3): 1397-1411
- [67] Lin Y, Kemme B, Patino-Martinez M, et al. Enhancing edge computing with database replication [C] // *Proc of the 26th IEEE Int Symp on Reliable Distributed Systems*. Piscataway, NJ: IEEE, 2007: 45-54
- [68] Khan M A, Debnath H, Paiker N R, et al. Moitree: A middleware for cloud-assisted mobile distributed apps [C] // *Proc of the 4th IEEE Int Conf on Mobile Cloud Computing, Services, and Engineering*. Piscataway, NJ: IEEE, 2016: 21-30
- [69] Mao Yuyi, Zhang Jun, Letaief K B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices [J]. *IEEE Journal on Selected Areas in Communications*, 2016, 34(12): 3590-3605
- [70] Tatemura J, Hsiung W P, Li W S. Acceleration of web service workflow execution through edge computing [C] // *Proc of the 13th Int World Wide Web Conf (WWW)*. Berlin: Springer, 2003
- [71] Ramakrishnan S, Reutiman R, Chandra A, et al. Accelerating distributed workflows with edge resources [C] // *Proc of Parallel and Distributed Processing Symp Workshops & PhD Forum*. Piscataway, NJ: IEEE, 2013: 2129-2138
- [72] Gao Lin, Iosifidis G, Huang Jianwei, et al. Bargaining-based mobile data offloading [J]. *IEEE Journal on Selected Areas in Communications*, 2014, 32(6): 1114-1125
- [73] Han T, Ansari N. Offloading mobile traffic via green content broker [J]. *IEEE Internet of Things Journal*, 2014, 1(2): 161-170

- [74] Sun Xiang, Ansari N. Primal: Profit maximization avatar placement for mobile edge computing [C] //Proc of 2016 IEEE Int Conf on Communications. Piscataway, NJ: IEEE, 2016: 1-6
- [75] Sajjad H P, Danniswara K, Al-Shishtawy A, et al. SpanEdge: Towards unifying stream processing over central and near-the-edge data centers [C] //Proc of IEEE/ACM Symp on Edge Computing. Piscataway, NJ: IEEE, 2016: 168-178
- [76] Renkonen K O, Seppala M. Edge analytics in the Internet of things [J]. IEEE Pervasive Computing, 2015, 14(2): 24-31
- [77] Satyanarayanan M. Edge computing for situational awareness [C] //Proc of 2017 IEEE Int Symp on Local and Metropolitan Area Networks. Piscataway, NJ: IEEE, 2017: 1-6
- [78] Taleb T, Dutta S, Ksentini A, et al. Mobile edge computing potential in making cities smarter [J]. IEEE Communications Magazine, 2017, 55(3): 38-43
- [79] Wang Feng, Xu Jie, Wang Xin, et al. Joint offloading and computing optimization in wireless powered mobile-edge computing systems [EB/OL]. [2017-10-23]. <http://ieeexplore.ieee.org/document/8234686/>
- [80] Ahmed E, Rehmani M H. Joint subcarrier and CPU time allocation for mobile edge computing [EB/OL]. [2017-10-23]. <http://pdfs.semanticscholar.org/302e/3d46b0de4ca9dc65447c1469eea0f800e9.pdf>
- [81] Sardellitti S, Scutari G, Barbarossa S. Joint optimization of radio and computational resources for multicell mobile-edge computing [J]. IEEE Trans on Signal and Information Processing over Networks, 2015, 1(2): 89-103
- [82] Kiani A, Ansari N. Towards hierarchical mobile edge computing: An auction-based profit maximization approach [J]. IEEE Internet of Things Journal, 2017, 4(6): 2082-2091
- [83] Tran T X, Hajisami A, Pandey P, et al. Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges [J]. IEEE Communications Magazine, 2017, 55(4): 54-61
- [84] Sun Yuxuan, Zhou Sheng, Xu Jie. EMM: Energy-aware mobility management for mobile edge computing in ultra dense networks [J]. IEEE Journal on Selected Areas in Communications, 2017, 35(11): 2637-2646
- [85] Mao Yuyi, Zhang Jun, Song S H, et al. Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems [J]. IEEE Trans on Wireless Communications, 2017, 16(9): 5994-6009
- [86] Papaioannou T, Katsalis K, Nikaein N, et al. SLA-driven VM scheduling in mobile edge computing [C] //Proc of the 1st IEEE Int Conf on Cloud Computing. Piscataway, NJ: IEEE, 2016
- [87] Al-Shuwaili A, Simeone O. Optimal resource allocation for mobile edge computing-based augmented reality applications [EB/OL]. [2017-10-23]. <https://arxiv.org/abs/1611.09243v1>
- [88] Liu Juan, Mao Yuyi, Zhang Jun, et al. Delay-optimal computation task scheduling for mobile-edge computing systems [C] //Proc of 2016 IEEE Int Symp on Information Theory. Piscataway, NJ: IEEE, 2016: 1451-1455
- [89] Gong Shangfu, Zhu Jianlei. A survey of reputation and trust mechanism in peer-to-peer network [C] //Proc of 2012 Int Conf on Industrial Control and Electronics Engineering. Piscataway, NJ: IEEE, 2012: 116-119
- [90] Yan Zheng, Zhang Peng, Vasilakos A V. A survey on trust management for Internet of things [J]. Journal of Network & Computer Applications, 2014, 42(3): 120-134
- [91] Petri I, Rana O F, Rezgui Y, et al. Trust modelling and analysis in peer-to-peer clouds. [J]. International Journal of Cloud Computing, 2012, 1(2/3): 221-239
- [92] Chen Shuhong, Wang Guojun, Jia Wenjia. A Trust Model Using Implicit Call Behavioral Graph for Mobile Cloud Computing [M]. Berlin: Springer, 2013: 387-402
- [93] Hussain M, Almourad B M. Trust in mobile cloud computing with LTE-based deployment [C] //Proc of 2014 IEEE Int Conf on Ubiquitous Intelligence and Computing, Int Conf on Autonomic and Trusted Computing, Int Conf on Scalable Computing and Communications and ITS Associated Workshops. Piscataway, NJ: IEEE, 2014: 643-648
- [94] Kantert J, Edenhofer S, Tomforde S, et al. Representation of trust and reputation in self-managed computing systems [C] //Proc of 2015 IEEE Int Conf on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM). Piscataway, NJ: IEEE, 2015: 1827-1834
- [95] Figueroa M, Uttecht K, Rosenberg J. A SOUND approach to security in mobile and cloud-oriented environments [C] //Proc of 2015 IEEE Int Symp on Technologies for Homeland Security. Piscataway, NJ: IEEE, 2015: 1-7
- [96] Bennani N, Boukadi K, Ghedira-Guegan C. A trust management solution in the context of hybrid clouds [C] //Proc of the 23rd IEEE Int WETICE Conf. Piscataway, NJ: IEEE, 2014: 339-344
- [97] Jang M, Lee H, Schwan K, et al. SOUL: An edge-cloud system for mobile applications in a sensor-rich world [C] //Proc of IEEE/ACM Symp on Edge Computing. Piscataway, NJ: IEEE, 2016: 155-167
- [98] Echeverria S, Klinedinst D, Williams K, et al. Establishing Trusted Identities in Disconnected Edge Environments [C] //Proc of IEEE/ACM Symp on Edge Computing. Piscataway, NJ: IEEE, 2016: 51-63

- [99] Cicirelli F, Guerrieri A, Spezzano G, et al. Edge computing and social Internet of things for large-scale smart environments development [EB/OL]. [2017-10-23]. <http://ieeexplore.ieee.org/document/8115262/>
- [100] Xie Kun, Cao Jiannong, Wang Xin, et al. Optimal resource allocation for reliable and energy efficient cooperative communications [J]. IEEE Trans on Wireless Communications, 2013, 12(10): 4994-5007
- [101] Deng Xiaoheng, He Lifang, Liu Qiang, et al. EPTR: Expected path throughput based routing protocol for wireless mesh network [J]. Wireless Networks, 2016, 22(3): 839-854
- [102] Pan Yan, Deng Xiaoheng, Shen Hailan. Credit distribution for influence maximization in online social networks with time constraint [C] //Proc of 2015 IEEE Int Conf on Smart City/SocialCom/SustainCom. Piscataway, NJ: IEEE, 2015: 255-260
- [103] Deng Xiaoheng, He Lifang, Li Xu, et al. A reliable QoS-aware routing scheme for neighbor area network in smart grid [J]. Peer-to-Peer Networking and Applications, 2016, 9(4): 616-627
- [104] Xie Kun, Wang Guanxin, Wen Jigang, et al. Cooperative routing with relay assignment in multiradio multihop wireless networks [J]. IEEE/ACM Trans on Networking, 2016, 24(2): 859-872
- [105] Varghese B, Wang Nan, Barbhuiya S, et al. Challenges and opportunities in edge computing [C] //Proc of IEEE Int Conf on Smart Cloud. Piscataway, NJ: IEEE, 2016: 20-26
- [106] Deng Xiaoheng, Luo Jie, He Lifang, et al. Cooperative channel allocation and scheduling in multi-interface wireless mesh networks [EB/OL]. [2017-10-23]. <https://rd.springer.com/article/10.1007/s12083-017-0619-8>
- [107] Wang Huamin, Tang Yangbin, Yin Gang, et al. Trust mechanisms of Internet software [J]. Science in China (Series E): Information Sciences, 2006, 36(10): 1156-1169 (in Chinese)  
(王怀民, 唐扬斌, 尹刚, 等. 互联网软件的可信机理[J]. 中国科学 E 辑: 信息科学, 2006, 36(10): 1156-1169)
- [108] Andonov R, Poirriez V, Rajopadhye S. Unbounded knapsack problem: Dynamic programming revisited [J]. European Journal of Operational Research, 2000, 123(2): 394-407
- [109] Sarkar U K, Chakrabarti P P, Ghose S, et al. Reducing reexpansions in iterative-deepening search by controlling cutoff bounds [J]. Artificial Intelligence, 1991, 50(2): 207-221
- [110] Xu Jielong, Chen Zhenhua, Tang Jian, et al. T-storm: Traffic-aware online scheduling in storm [C] //Proc of the 34th IEEE Int Conf on Distributed Computing Systems. Piscataway, NJ: IEEE, 2014: 535-544
- [111] Fayard D, Zissimopoulos V. An approximation algorithm for solving unconstrained two-dimensional knapsack problems [J]. European Journal of Operational Research, 1995, 84(3): 618-632
- [112] Ahrenholz J. Comparison of CORE network emulation platforms [C] //Proc of MILCOM 2010. Piscataway, NJ: IEEE, 2010: 166-171
- [113] Danniswara K, Sajjad H P, Al-Shishtawy A, et al. Stream processing in community network clouds [C] //Proc of the 3rd Int Conf on Future Internet of Things and Cloud. Piscataway, NJ: IEEE, 2015: 800-805
- [114] Shi Rui, Yang Xiazong. Research on the node spatial probabilistic distribution of the random waypoint mobility model for ad hoc network [J]. Journal of Computer Research and Development, 2005, 42(12): 2056-2062 (in Chinese)  
(时锐, 杨孝宗. 自组网 Random waypoint 移动模型节点空间概率分布的研究[J]. 计算机研究与发展, 2005, 42(12): 2056-2062)
- [115] Evans R. Yahoo's storm performance test [OL]. [2017-10-01]. <https://github.com/yahoo/storm-perf-test>



**Deng Xiaoheng**, born in 1974. PhD. Professor in the School of Information Science and engineering of Central South University. Senior member of CCF. His main research interests include wireless communications and networking, congestion control for wired/wireless network, cross layer route design for wireless mesh network and ad hoc network, online social network analysis.



**Guan Peiyuan**, born in 1987. PhD candidate in the School of Information Science and engineering of Central South University. His main research interests include edge computing, opportunity network, and big data.



**Wan Zhiwen**, born in 1994. Master candidate in the School of Information Science and Engineering of Central South University, Changsha, China. His main research interest is edge computing.



**Liu Enlu**, born in 1994. Master candidate in the School of Information Science and Engineering of Central South University, Changsha, China. Her main research interest is edge computing.



**Luo Jie**, born in 1994. Master candidate in the School of Information Science and Engineering of Central South University, Changsha, China. Member of CCF. His main research interests include wireless network and edge computing.



**Zhao Zhihui**, born in 1993. Master candidate in software engineering at Central South University. Her main research interests include edge computing, reputation mechanism and trust mechanism.



**Liu Yajun**, born in 1995. Master candidate in the School of Information Science and Engineering of Central South University, Changsha, China. Her main research interest is edge computing.



**Zhang Honggang**, born in 1970. PhD. Associate professor of computer engineering in the Engineering Department at University of Massachusetts Boston. His main research interests include computer networks and distributed systems, mobile computing, Internet of things, data sharing/swarming, and delay/disruption tolerant networks.