



综述

## 边缘计算迁移研究综述

朱友康<sup>1</sup>, 乐光学<sup>1,2</sup>, 杨晓慧<sup>1</sup>, 刘建生<sup>1</sup>

(1. 江西理工大学, 江西 赣州 341000;

2. 嘉兴学院, 浙江 嘉兴 314001;

**摘要:** 边缘计算将网络边缘上的计算、存储等资源进行有机融合, 构建成统一的用户服务平台, 按就近服务原则对网络边缘节点任务请求及时响应并有效处理。由于边缘节点能力、资源、带宽、能源等受限, 计算迁移是边缘计算的一个关键问题。综述了移动边缘计算的起因、演进和发展趋势, 以从分布式计算、普适计算、云计算到边缘计算的演进历程为主线, 对比分析了各阶段计算迁移的特点并对经典模型进行了评述; 重点分析了边缘计算的最新研究进展及应用领域, 研究并对比分析了基于能耗优化管理的移动边缘计算模型。最后提出了一个面向 LTE 应用和基于时分多址 (time division multiple access, TDMA) 的多用户边缘移动终端计算迁移策略系统。

**关键词:** 边缘计算; 计算迁移; 物联网; 云计算; 移动终端

**中图分类号:** TP393

**文献标识码:** A

**doi:** 10.11959/j.issn.1000-0801.2019059

## A survey on edge computing offloading

ZHU Youkang<sup>1</sup>, YUE Guangxue<sup>1,2</sup>, YANG Xiaohui<sup>1</sup>, LIU Jiansheng<sup>1</sup>

1. Jiangxi University of Science and Technology, Ganzhou 341000, China

2. Jiaxing University, Jiaxing 314001, China

**Abstract:** Edge computing integrate the computing, storage and other resources on the edge of the network into a unified user service platform. According to the principle of nearest service, the network edge node task request is timely and correspondingly processed. Computational migration is a key issue in edge computing due to limited edge node capabilities, resources, bandwidth, and energy. The causes, evolution and development trends of mobile edge computing were reviewed from the distributed computing, pervasive computing, cloud computing to edge computing as the main line. The characteristics of computing migration at each stage were compared and analyzed, and the classical models were reviewed. The latest research progress and application fields of edge computing were focused on, mobile edge computing models based on energy optimization management were researched and compared. Finally, a time division multiple access based multi-user edge mobile terminal computing migration strategy system for LTE applications was proposed.

**Key words:** edge computing, computation offloading, IoT, cloud computing, mobile terminal

收稿日期: 2018-12-26; 修回日期: 2019-03-10

通信作者: 乐光学, guangxueyue@163.com

基金项目: 国家自然科学基金资助项目 (No.61462036); 浙江省自然科学基金资助项目 (No.LY16F020028)

**Foundation Items:** The National Natural Science Foundation of China (No.61462036), The National Natural Science Foundation of Zhejiang Province of China (No. LY16F020028)

2019059-1

## 1 引言

近年来,随着移动网络技术和智能终端的普及,人们需要用移动终端运行复杂的富应用程序,如计算机视觉、移动游戏、移动医疗和移动学习等。智能手机、平板电脑等移动设备正在成为学习、娱乐、社交、新闻更新和商业交流的重要工具。然而,由于移动终端的资源<sup>[1]</sup>(计算资源、电量资源、存储资源)是有限的,有限的计算资源与不断扩大的计算需求之间的矛盾日益突出。

随着互联网技术的发展,特别是智能移动终端的更新升级,智能手机拥有越来越强大的功能。移动设备要变得更小、更轻,电池寿命变得更长,意味着计算能力会受到限制。但是,用户对移动智能终端的期望越来越高,从而对计算和数据操作能力的要求也在提高,而实现这些功能则会损耗大量电池电量,如何协调这些矛盾是目前智能移动终端发展的技术瓶颈。

在云计算的发展热潮之后,许多云服务(如移动医疗、移动学习、移动游戏和移动管理等)都可以在移动设备上直接使用。移动边缘计算作为一种新技术出现,它在移动用户的近距离范围内提供云和IT服务。在移动边缘计算中,边缘云服务器被部署于每个基站,网络运营商负责转发和过滤数据分组。

计算迁移技术的兴起,为解决移动终端资源受限引入了新的方法。计算迁移在解决以下几种应用情况时作用尤为明显。

### (1) 将计算体迁移到其他资源丰富的终端上运行

智能移动终端可以实现大部分桌面终端上的应用,因此用户开始将手机视为个人信息处理工具,期望手机能够像常规信息处理设备(如工作站或笔记本电脑等)那样执行所有应用。但是,智能移动终端由于体积、输入操作等的限制,计算资源、存储资源、电池容量和网络连接能力始

终有限,无法支持一些大型应用的运行。此时,可以将部分计算体迁移到资源丰富的终端上运行<sup>[2]</sup>,只需返回计算结果给移动终端就可以达到预期的计算效果。

### (2) 跨移动终端任务同步

由于用户工作和生活的需要,要在多个不同的物理环境中使用不同的移动设备执行相同的任务,由此产生“一人多机”模式。但是,由于缺乏跨终端的任务迁移支持,任务迁移面临着连续性、异构性、透明性以及如何恢复任务的执行状态等挑战。当用户从一个环境移动到新的环境中时,需要重新手动配置运行环境,这会浪费用户的大量与任务执行无关的注意力资源。而计算迁移可以将一个移动设备上的任务执行状态迁移到其他移动设备<sup>[3]</sup>,实现跨移动终端的任务同步。

### (3) 移动设备资源共享的计算模型

新的物联网模式使资源受限的移动设备通过互联网相互连接。但是,这些终端设备处理能力和存储容量有限。计算迁移中,边缘设备将部分计算转移到远程云上执行,从而节省了移动设备的处理能力和能量。然而,将计算迁移到公共云需要公共云和边缘设备之间通过Internet进行数据交换<sup>[4]</sup>,由于网络带宽限制,数据交换过程可能会出现时延。为了克服这一问题,边缘计算的迁移方法被提出,即将移动设备的计算迁移到设备附近的其他移动设备上,它们之间通过Wi-Fi连接。这种在移动设备附近提供资源的其他移动终端称为cloudlet<sup>[5]</sup>。

### (4) 云数据中心计算任务下沉到边缘节点

云数据中心将计算和存储能力等资源“下沉”到网络边缘节点,因距离用户更近,用户请求不再需要经过漫长的传输网络到达核心网才能被处理,而由部署在本地的边缘服务器将一部分流量卸载,降低对传输网和核心网带宽的要求,直接处理并响应用户,满足移动网络高速发展所需的



高带宽、低时延的要求并减轻网络负荷,大大降低了通信时延<sup>[6]</sup>。

边缘计算为应用程序开发人员和供应商提供实时信息(如网络负载、用户位置),这些实时网络信息用于为移动用户提供上下文感知服务,从而丰富用户的使用体验,提升用户满意度。

## 2 引发计算迁移的原因

计算迁移本身是一个复杂的过程,而迁移是其最显著的特征之一,但是在不同的环境中引发迁移的原因也是不同的,本文总结了几种不同的计算迁移的原因,具体如下。

### (1) 移动终端资源受限

在复杂的 Internet 环境中,各种网络设备由于体积、质量等方面的千差万别,它们所承载的计算资源也有大有小,特别是现今智能手机普及度越来越高,人们对智能移动终端的依赖也越来越高,但智能手机由于体积的限制,不能像桌面计算机那样执行人们期望的所有应用。另一方面,移动终端的位置是根据人的需求随机变动的,不同的物理环境中,客户的网络质量不尽相同,这就产生了网络资源的动态变化<sup>[7]</sup>,需要计算迁移弥补网络连接差所带来的影响。

### (2) 均衡终端任务负载

当终端一次在本地执行的任务较多时,随着终端被占用的各种资源增加,设备的负载急剧增加,这时任务被执行效率就会变得很低,计算迁移可以减轻资源占用较大的设备的负载,并迁移到其他资源丰富的设备上。

### (3) 降低数据传输量

在计算任务执行的过程中,计算节点和本地客户端之间会产生大量需要传输的中间数据,这些数据的频繁传输需要耗费大量的网络资源。而将任务迁移到计算节点执行,这些执行计算所需的数据可以直接在计算节点调用,计算节点只需要返回给客户端计算结果,传输的

数据量大大减小。

### (4) 减少设备网络时延

当用户所在的物理环境网络不稳定、时延高时,将计算任务迁移到计算节点执行,客户端不必担心网络环境的变化,任务迁移之后用户不必保持时刻在线。用户可以在任务迁移到计算节点后暂时断开网络连接,待返回结果时连接即可。

## 3 计算迁移的发展

Mahadev<sup>[8]</sup>提出的 Cyber Foraging 思想是计算迁移的最早起源,他提出将资源受限的移动终端上的计算、存储等任务交给终端设备附近的计算、存储能力更强的服务器执行,可以减少移动终端的计算量,提升终端的性能。按照计算迁移的迁移方式,迁移从有限资源环境下的嵌入式计算,发展到小规模分布式计算,再到普适计算,2008 年以来进入云计算时代,2016 年则兴起基于终端附近设备的边缘计算。张文丽等<sup>[5]</sup>按照不同背景,将计算迁移分为分布式计算、普适计算和云计算 3 个阶段。本文基于参考文献<sup>[5]</sup>并按照时间的先后顺序,将计算迁移的相关研究工作划分为分布式计算阶段、普适计算阶段、云计算阶段和边缘计算阶段。表 1 总结了这几个阶段计算迁移研究的内容及实现方法。

### 3.1 分布式计算

在个人计算机和局域网取得飞速发展的背景下,单台设备的计算和存储能力不断加强,个人计算机大量资源大多数时间空闲,又由于局域网的发展,使得计算迁移成为可能。分布式计算应运而生。分布式计算的思想是将空闲的资源以服务的形式提供给他人使用,利用互联网上的计算机的 CPU 的闲置处理能力,从而将不同空间的设备的计算能力结合起来,解决大型的计算问题。

表1 计算迁移的发展

发展阶段	时间	研究动机	研究内容	实现方法
分布式计算	1995—2000年	便携式计算机的性能/能耗	迁移可行性研究	开发人员手动划分
普适计算	2001—2008年	智能手机性能/能耗	迁移系统架构	基于计算代理
云计算	2008—2016年	智能移动终端功能增强	基于云的迁移系统实现	基于云服务器
边缘计算	2016年至今	智能移动终端、资源互联共享	基于移动设备的迁移	基于WLAN/蓝牙的终端互联

但是,分布式计算的实现涉及的情形是非常复杂的,分布式计算对计算过程中的通信、数据获取以及计算结果的返回都有严格的要求。因此需要借助分布式计算框架,对计算细节进行封装,从而保证分布式计算正常运行。

通过使用分布式计算框架,程序员可以很容易地享受分布式计算带来的高速计算的好处,而且还不必对分布式计算过程中的各种问题和计算异常进行控制。本文对当前的分布式计算框架进行了系统的回顾与总结,任务执行过程给予计算迁移启发。分布式框架的介绍及对比见表2。

分布式计算出现于便携式计算机刚发展的阶段,受网络技术发展水平的限制,便携式计算机面临的主要问题是电源的待机时间短、续航能力较差,导致整体性能低。因此,在分布式计算阶段,计算迁移的研究主要是探讨将计算任务迁移到远程执行的可行性。

分布式计算阶段计算迁移研究最主要的贡献有以下3点。

(1) 基于负载共享的软件方法。电源管理是这一阶段计算迁移研究的一个重要方向,Othwan等<sup>[12]</sup>基于硬件方法,提出了一种基于负载共享概念的移动计算终端电源保护策略,用户任务从一个移动主机迁移到一个固定主机执行,可以延长电池寿命,缩短工作响应时间,以减少CPU的功耗。而CPU的利用率对传输过程有重要影响,参考文献<sup>[12]</sup>用式(1)中 $\text{CPU}_{\text{utilisation}}$ 来表示CPU的利用率:

$$\text{CPU}_{\text{utilisation}} = \frac{\sum J_i}{T} \quad (1)$$

其中, $J_i$ 表示CPU执行任务*i*的时间, $T$ 表示总的迁移周期。

(2) 无线环境下的远程执行可行性验证。

表2 分布式架构比较

框架名称	架构设计	存储	通信	任务执行过程
Hadoop <sup>[9]</sup>	JobTracker/ TraskTacker	HDFS	RPC/ HTTP	MapReduce Hadoop 把一个大的任务划分为多个小的计算任务并分配给集群的每个计算节点,跟踪每个计算节点的进度以决定是否重新执行,最后收集每个节点上的计算结果并输出
Storm <sup>[10]</sup>	Nimbus/ supervisor	实时的 输入流	zeroMQ 消息队列	向客户端提交一个声明好的拓扑,Nimbus 通过与 Zookeeper 交互获取适合的运行机器,把任务分配到具体的机器,机器上的 supervisor 根据分配到的任务启动相应进程。期间,无论是 supervisor 还是 worker 都与 Zookeeper 保持心跳联系
Spark <sup>[11]</sup>	master/ workers	内存、磁盘	共享、广播 变量	用户自己开发的程序以 driver 的方式连接 master,并指定数据集 RDD 的生成与转换,将 RDD 的操作发送至任务执行节点 workers。workers 即执行具体任务也存储计算所需数据,workers 收到操作定义对本地化数据进行操作,生成预期结果,对结果返回或者存储



本地运行成本高的任务,通过无线远程执行,这样可以通过固定终端连接来节省电力和电池寿命。Rudenko 等<sup>[13]</sup>比较了本地执行的进程和远程执行进程的功耗,权衡了通信电源开销与本地处理的电力成本,通过无线通信在便携式计算机设备中的使用,探讨了远程访问和远程执行的可行性。

(3) 任务自动划分理念。Hunt 等<sup>[14]</sup>提出了一个分布式应用程序自动分区系统 Coign, Coign 通过基于场景的分析构建应用程序的组件间通信的图形模型,Coign 应用图形切割算法在网络上划分应用程序,并最大限度地减少网络通信导致的执行时延。Coign 的核心思想是将计算任务的划分交给系统自身来完成<sup>[14]</sup>,图 1 表示 Coign 的运行时架构,使用 Coign,即使终端用户无法访问源代码,也可以将非分布式应用程序转换为优化的分布式应用程序,这样可以极大地减轻开发人员的工作负担。

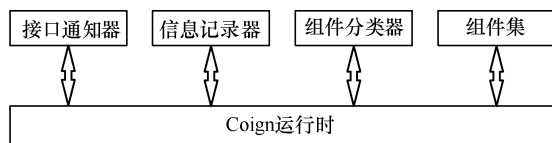


图 1 Coign 框架<sup>[14]</sup>

但是,分布式计算阶段关于计算迁移的研究主要是提出理论设想,并未提出具体的理论框架,也未进行可行性和有效性的验证,因此分布式计算阶段关于计算迁移的研究尚处于萌芽的状态,并未有实质性的进展。

### 3.2 普适计算

20 世纪 70 年代中期,随着个人计算机的普及,人与计算机之间的距离被极大地拉近了。尽管个人计算机并没有展现信息技术的全貌,但它迈出了让计算机普及的第一步。普适计算还是硬件技术、图形用户界面快速发展的一个重要推动力<sup>[15]</sup>。例如手持和可穿戴设备、无线局域网以及位置传感器都是普适计算阶段推动发展起来的。

清华大学徐光祐教授<sup>[16]</sup>等对普适计算的定义是:“普适计算是信息空间与物理空间的融合,在这个融合的空间中人们可以随时随地、透明地获得数字化的服务”。普适计算强调让计算机本身从人们的视线中消失,让人们注意的中心回归到完成的任务本身。

普适计算模式下计算迁移相关研究可以分为计算迁移架构、应用划分算法、资源预测方法、应用开发环境和技术几个类别<sup>[5]</sup>。普适计算的研究体系如图 2 所示。

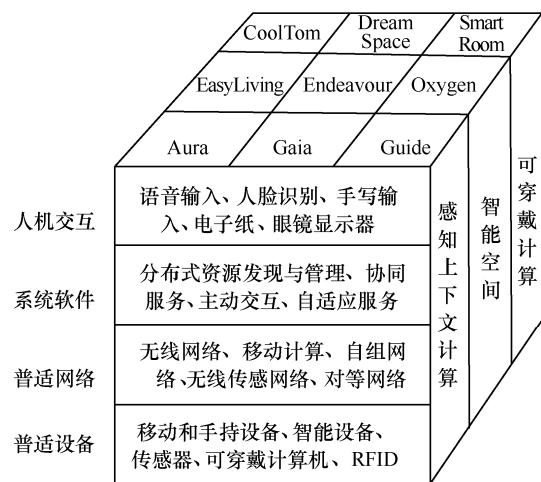
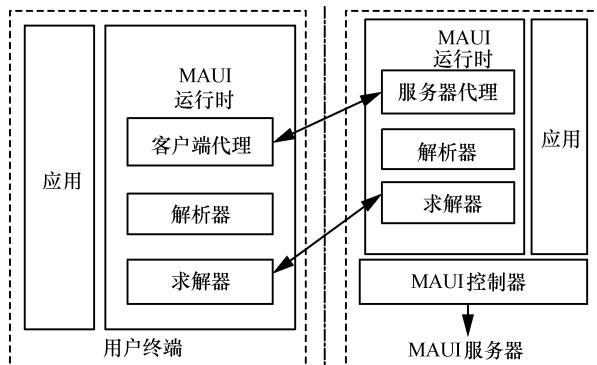


图 2 普适计算的研究体系

在计算迁移架构方面,Eduardo 等<sup>[17]</sup>提出了一个重要研究成果 MAUI,这是一种能够将基于能量感知的移动终端代码迁移到基础设施中的系统,MAUI 系统架构如图 3 所示。其中解析器负责收集终端的信息参数,提供给求解器,求解器根据参数的变化实时地求解线性规划问题。MAUI 利用托管代码的优势来减轻程序员处理程序分区的负担,同时最大限度地节省迁移代码产生的能耗。

Rajesh 等<sup>[18]</sup>最早提出了基于策略的远程执行系统 Chroma,该系统在保留特定应用程序的信息的同时,能够轻松地对这些应用程序进行分区以进行远程执行,可以在过度负荷的环境中自动使用额外的资源来提高应用程序性能。

图3 MAUI系统架构<sup>[17]</sup>

远程执行是普适计算的一项重要功能,它将小型设备的移动性与大型计算服务器的处理能力相结合。Jason 等<sup>[19]</sup>提出了一种基于普适计算的终端远程执行系统 Spectra, Spectra 通过匹配资源供应和需求来协助应用程序实现远程执行,为应用程序的功能该如何执行以及在何处执行提供建议。

Steven 等<sup>[20]</sup>提出了一种用于应用程序透明迁移的系统 Zap, Zap 在操作系统之上提供了一个虚拟化层,引入 pod, 这些 pod 是一组进程。Zap 可以在各个独立的计算机操作系统之间将一组进程 pod 作为一个单元进行迁移,而不会在迁移后留下任何残留状态,无需对操作系统进行任何更改。

Eli 等<sup>[21]</sup>提出了基于 Java 的应用程序自动程序分区系统 J-Orchestra, 该系统通过使用字节码重写、代理直接引用对象等方法,将字节码格式的 Java 应用程序作为输入,并将它们转换为分布式的应用程序,在不同的 Java 虚拟机上运行。J-Orchestra 在分布式系统中的使用方法很简单,用户只需要指定各种硬件和软件资源的网络位置及其相应的应用程序,因此 J-Orchestra 具有显著的通用性、灵活性和自动化的优势。

Su 等<sup>[22]</sup>提出了名为 Slingshot 的架构,该架构通过复制远程应用程序代理来缩短应用程序响应时间。每个应用程序有两类副本,第一类副本在终端用户连接的远程服务器上执行,Slingshot 在附近的代理上部署了第二类副本,代理将每个应

用程序的请求广播到所有副本,并返回它收到的第一个响应,这样可以缩短应用程序响应时间。

普适计算关于计算迁移的另一研究是动态分区应用程序,并将应用程序的一部分迁移到强大的代理上执行,即应用划分算法研究。Gu 等<sup>[23]</sup>基于模糊控制模型提出了一个迁移推理引擎,以自适应地选择应用程序分区策略。

在资源预测算法方面, Rich 等<sup>[24]</sup>通过预测本地和远程系统之间的带宽、信道条件等,并将迁移资源决策模型统一为一个统计决策问题,使用贝叶斯方法评估不同决策策略的效果,得出结论:在估计先验分布采用自动变点检测的贝叶斯方法性能最佳。

### 3.3 云计算

2007 年, IBM 和 Google 宣布在云计算领域展开合作, 云计算作为一种新兴的计算服务模式被产业界和学术界广泛关注<sup>[25]</sup>。云是一种并行和分布式系统, 由一系列相互连接的虚拟计算机组成, 它们根据服务提供商和消费者之间建立的服务等级协议, 动态地为用户呈现和供应一个或多个统一的计算资源<sup>[26]</sup>。基于参考文献[25-26]总结了云计算的演进历程, 见表 3。

云资源与传统的计算资源相比, 具有服务质量好、可用性高、资源的利用率和可扩展性强、数据段安全性和隐私性高的特点。但是云资源的访问也存在一些挑战, 由于用户与服务商之间的交互过程经过广域网和无线网, 云资源的使用存在以下 3 个问题<sup>[5]</sup>:

- 因广域网而带来的抖动、错误和时延;
- 无线网带宽有限、连接不稳定;
- 访问云资源的经济成本较高等。

在云计算阶段计算迁移按实现方法主要有 3 种, 分别为基于 cloudlet 的计算迁移、基于 surrogate 的计算迁移和基于云的计算迁移。

Tim 等<sup>[27]</sup>提出云应该以 cloudlet 的形式向移动用户的物理位置靠近, 建议使用一个更细粒度



表 3 云计算的演进历程

所处时期	特点
集群化时期	将分散的资源组合成了规模化的数据中心。这样的集中化时期也产生了不同的问题：数据传输中断、数据中心业务崩塌带来的数据中断等
网络计算时期	利用互联网把地理上广泛分布的各种资源连成一个逻辑的整体，就像一台超级计算机
虚拟化时期	将多台服务器整合，屏蔽了底层设备的差异之处转向统一提供处理能力，实现了物理服务器资源利用率的提升。在虚拟化技术出现之前，软件只能被绑定在静态硬件环境中，而虚拟化则打断了这种软硬件之间的依赖性
云计算时期	进入可配置的计算机资源共享池，这些资源能够被快速提供，只需投入很少的管理工作，或服务供应商进行很少的交互。按需自助服务，网络接入无处不在，与位置无关的资源池，快速弹性，按使用付费

的 cloudlet 概念来管理组件级别的应用程序，而不是将一个完整的虚拟机从云移动到 cloudlet。而且 cloudlet 不是固定在移动用户附近的固定的基础设施，而是以动态的方式给 LAN 中的移动用户提供可用的资源。这种物理位置上的邻近能有效地解决因广域网而带来的抖动、错误、时延以及网络连接不稳定等问题。

Ra 等<sup>[28]</sup>提出的 Odessa 是基于 surrogate 的计算迁移系统，Odessa 通过其在线分析器和简单的执行时间预测器，能够自适应地为移动用户的交互型应用程序提供计算迁移和并行处理决策，可以快速适应场景复杂性、计算资源可用性和网络带宽的变化。

CloneCloud<sup>[29]</sup>是基于云的典型的计算迁移系统，CloneCloud 采用静态分析和动态分析相结合的方法，以精细的粒度自动分区应用程序，同时优化移动终端在特定计算和通信环境下的执行时间和能源使用，能够在一些应用程序上实现高达 20 倍的执行速度，同时大大节省了移动设备的能耗。

表 4 对基于 cloudlet 的计算迁移、基于 surrogate 的计算迁移和基于 CloneCloud 的计算迁移这 3 种计算迁移系统进行了比较。

伴随移动互联网与云计算技术不断融合，基于移动终端的云计算模式迅速普及。移动终端计算、存储资源受重量、大小和散热等因素的限制，日益复杂的移动应用使得用户的体验质量降低。因此如何突破移动终端计算、存储和电池等资源的受限瓶颈，为它们提供增强计算能力和降低计算时延是至关重要的任务。为向移动用户提供更加丰富的应用，将云计算引入移动环境，移动云计算（mobile cloud computing, MCC）作为一种新的应用模式应运而生，计算迁移应用于云环境中，成为移动云计算的重要支撑技术<sup>[25]</sup>。

柳兴等<sup>[30]</sup>针对应用迁移到云端引起大量数据传输导致的能耗问题，提出一种任务联合执行策略（task collaborative execution policy, TCEP），通过把云端与移动终端联合移动应用的优化问题建模为最小化移动终端能耗问题，并采用一次迁

表 4 不同计算迁移实现方法比较

实现方法	迁移粒度	特点	缺点	目标	适用场景
cloudlet <sup>[27]</sup>	VM	支持以 VM 为粒度的应用划分，对系统进行设计和实现，从而使系统更灵活，应用性能也更高	安全性一般，且不支持对已有应用进行划分	响应时间	针对网络时延敏感的应用
Surrogate <sup>[28]</sup>	进程	支持自适应计算迁移，并基于 Android 为开发人员提供应用开发环境，能够根据移动终端周围的环境自动地进行计算迁移和并行化决策	安全性差	性能	针对网络时延敏感的应用
CloneCloud <sup>[29]</sup>	VM	自动转换所有运行在应用 VM 上的移动应用，完全不需要开发人员的参与	需要针对各种执行环境预先建立划分数据库	执行时间/能耗	针对网络时延不敏感或需要并行处理的应用

移最优特性来改进遗传算法进一步提高算法性能。仿真表明, TCEP 与现有的策略相比, 可有效减少最优解的运算时间, 同时最小化移动终端能耗。

### 3.4 边缘计算

近年来, 在物联网和 5G 通信的驱动下, 移动计算领域出现了一个范式的转变, 从集中式的云计算向分布式的边缘计算转变。为了解决云计算中的时延问题, 云服务应该被转移到与用户物理位置邻近的地方—移动网络的边缘, 也就是新出现的边缘计算范式。施巍松等<sup>[31]</sup>对边缘计算的定义是在网络边缘执行计算的一种新型计算模型, 而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。

计算迁移可以概括为代理发现、环境感知、任务划分、任务调度和执行控制等步骤<sup>[32]</sup>。然而, 不是所有计算迁移方案都包含计算迁移的全部步骤。具体迁移步骤如图 4 所示。

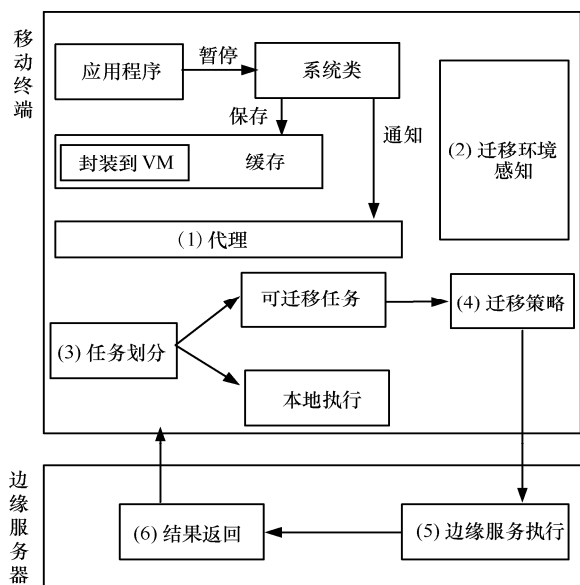


图 4 边缘计算迁移步骤

边缘计算可以被理解为云计算的一种特殊情况, 在传统的云计算中, 用户通过 Internet 连接访问云服务, 而在边缘计算中, 计算和存储资源分布于用户的近距离网络拓扑范围内<sup>[33]</sup>。因此, 与云计算相比, 边缘计算可以明显地降低时延和抖

动。云计算是以完全集中的方式部署, 服务器通常集中地放置在一个或几个位置, 但边缘计算完全以分布式的方式部署。边缘计算只能够提供有限的计算和存储资源, 表 5 中概述了云计算和边缘计算在关键技术方面的比较。

表 5 云计算与边缘计算关键技术的比较

评价指标	云计算	边缘计算
部署方式	集中式	分布式
与用户的距离	远	近
时延	高	低
抖动	大	小
计算能力	强	有限
存储能力	强	有限

边缘计算主要实现将云端的计算、存储等资源优势, 引入移动终端<sup>[25]</sup>, 使其突破终端资源限制, 提高运行应用程序的性能。它实现了将网络、计算、存储以及应用的融合, 通过开放网络能力来降低核心网络时延<sup>[34]</sup>, 使时延可以限制在 10 ms 级以内, 这对那些对时延要求高的应用意义深远, 特别是无人驾驶等领域。

边缘计算所涉及的范围非常广, 首先边缘计算与节点的信任以及资源协同研究<sup>[35]</sup>是密不可分的, 计算迁移技术是实现边缘计算的重要技术手段, 边缘计算中的数据安全与隐私保护问题则关系到边缘计算在各个应用领域的服务质量。笔者分别从技术体系、研究难点和应用方向 3 个层面对边缘计算的研究体系进行总结归纳, 图 5 为边缘计算研究体系。

## 4 边缘计算

### 4.1 边缘计算的研究进展

边缘计算平台运行在与移动终端物理位置相邻近的位置, 为移动终端提供 IT 和云服务。本节将描述边缘计算领域的几个最新研究成果。

#### 4.1.1 Femtoclouds

美国卡内基·梅隆大学计算机学院的 Habak 等<sup>[36]</sup>

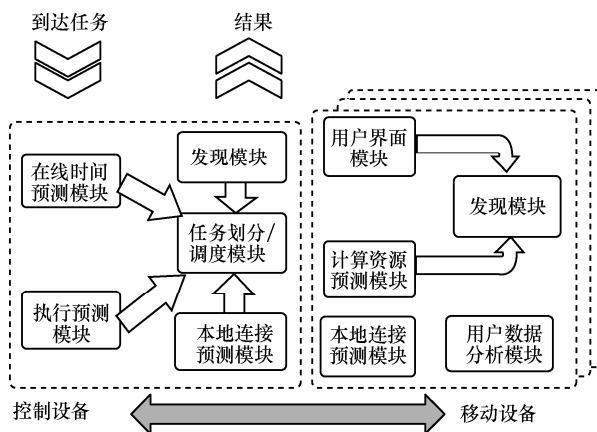




图5 边缘计算研究体系

提出 Femtoclouds 系统。Femtoclouds 是一个动态的、自配置的“多设备移动云系统”，通过协调多个移动设备来扩展 cloudlet 的计算资源。Femtoclouds 利用客户附近空闲的移动设备，为本地移动终端提供计算服务，因此减少了传统的将计算迁移到云数据中心过程中产生的网络时延。

Femtoclouds 系统架构<sup>[36]</sup>如图 6 所示，Femtoclouds 设备由 3 个部分组成：cloudlet 能够创建一个 Wi-Fi 接入点，并充当控制设备；计算机集群是一组愿意共享硬件和资源的移动设备；控制设备与集群中的移动设备协作，为迁移提供计算服务。

图6 Femtoclouds 系统架构<sup>[36]</sup>

控制设备的功能由执行预测模块、在线时间预测模块、任务划分和调度模块来实现。执行预

测模块负责估算每个计算任务所引入的执行负载，便于更有效地在不同的节点间分配任务。在线时间预测模块会收集区域中特定的数据，根据用户的集体行为构建一个通用的用户配置文件，预测 Femtoclouds 用户的在线时间。任务划分和调度模块执行预测模块和在线时间预测模块获得的信息，迭代地将任务划分给执行设备。

移动设备由用户界面模块、计算资源预测模块、用户数据分析模块组成。用户界面模块负责获取用户偏好、共享策略以及个人信息共享策略信息，将获得的信息发送给其他模块。计算资源预测模块负责分析移动设备的剩余计算能力，并将所获取的信息与控制设备共享。用户数据分析模块负责分析用户在不同场景中的行为以及偏好数据，构建一个用户数据概要文件，并只以用户可以接受的粒度与控制设备共享，以保护用户隐私。

控制设备与集群中的移动设备协作，执行在这两个设备中的本地连接预测模块、发现模块。本地连接预测模块负责计算控制设备和每个计算移动设备之间可用的带宽，笔者的方法是使用无线信号强度来获取带宽的初始估计。发现模块负责监测安装了 Femtoclouds 客户端的可用移动设备，一旦移动设备加入集群，它就会向控制设备发送一个注册分组。这个注册分组包含该设备之前对 Femtoclouds 系统贡献的计算能力的信息以及用于确定用户在线时间的配置文件信息。

Femtoclouds 的工作原理是：移动设备发送设备信息（移动设备的可用计算资源、可共享的计算资源、使用历史）和共享策略给 cloudlet，并通过 Wi-Fi 网络连接到计算集群。cloudlet 可以根据设备的计算资源可用性和电源情况，决定是否接受新设备接入计算集群中。移动终端通过发送计算代码、输入和输出数据到 cloudlet 上，迁移计算密集型任务。通过计算可用移动设备执行任务所需的时间，将迁移过来的任务在计算集群中进行

分配。

在移动设备的任务调度中, Femtoclouds 使用贪婪启发式方法来优化模型。在控制器中运行的调度算法对系统的性能至关重要。调度器必须将任务划分给可用的设备, 以最大化收益, 同时管理设备的运行。先将问题作为一个优化问题, 假设对设备能力(计算和带宽)和离开时间完全了解, 然后使用贪婪启发式方法来优化模型。对于每个任务, 调度程序决定将任务分配给哪个设备, 何时将任务发送给设备以及何时返回结果。任务执行调度器的总体目标是最大化整个集群的可用计算能力, 表示为<sup>[36]</sup>:

$$\text{Max}C = \sum_i \varepsilon_i \sum_k x_{ik} \quad (2)$$

其中,  $C$  是总的计算能力,  $\varepsilon_i$  是由第  $i$  个任务引入的计算负载。

系统的计算迁移性能取决于计算集群中可用的移动设备的数量以及设备的未使用资源量。

#### 4.1.2 Replisom

Abdelwahab 等<sup>[37]</sup>提出了 Replisom 系统, Replisom 这个名字的灵感来源于 DNA 复制, 它是一个移动边缘云架构, 当多个物联网设备想访问云资源时, Replisom 可以减少云的响应时间。

图 7 表示移动边缘计算端到端的内存复制架构, Replisom 将内存复制到各自的克隆虚拟机中, 而不是在边缘网络中增加新的设备。当多个物联网设备(假设第  $i$  个设备超过总量  $n$  个设备)有新的计算任务时, 每个设备使用端到端的通信技

术将新的任务发送给相邻设备。接收设备将接收的计算体压缩成一个副本。边缘云会定期给相应的物联网设备发送请求(使用预先计划好的授权)。当设备  $j$  接收来自边缘云的请求时, 作为响应, 它将压缩的副本推送到边缘云。

最后, 利用压缩采样构造算法, 将边缘云中的任务存储到相应的虚拟机中。当多个物联网设备将任务复制到附近的边缘云时, Replisom 体系结构降低了迁移期间的时延和成本。

#### 4.1.3 ME-VoLTE

Beck 等<sup>[38]</sup>提出了 ME-VoLTE (mobile edge computing enabled voice over LTE), 以减少视频通话中移动设备的电池消耗, 并为迁移策略的选择提供通信协议。ME-VoLTE 是基于视频电话的移动边缘计算系统, 可以在视频通话中减少终端能源消耗。

ME-VoLTE 系统将视频通话过程中的视频编码任务迁移到 MEC 边缘服务器上执行。图 8 表示 ME-VoLTE 的系统架构, ME-VoLTE 体系结构中的两个主要组件为: 服务调用控制模块(proxy/serving call state control function), 其任务是在移动设备和 VoLTE 网络之间发送信号; 媒体资源模块是 VoLTE 网络的一部分, 负责视频的混辑、存储视频的回放和视频转码。当一个移动设备想要进行视频电话时, 代理会将请求(连同一组代码块)发送到服务调用控制模块。服务调用控制模块根据当前可用的计算资源和上行强度来决定编码类型。在决定编码类型之后, 移动设备选择一

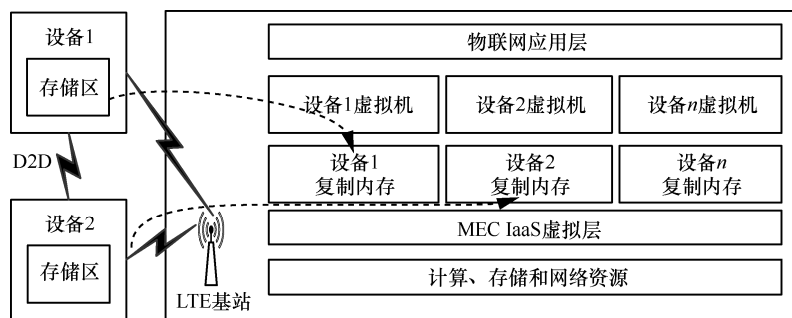


图 7 移动边缘计算端到端的内存复制架构<sup>[37]</sup>



个编解码器对视频进行编码, 这个编码器由边缘网络中的移动设备提供。

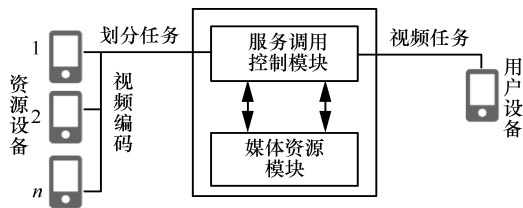


图8 ME-VoLTE 系统架构<sup>[38]</sup>

要实现将编码工作迁移到 MEC 服务器上的架构, 需要满足 3 个方面的要求。从用户的角度看, 新的 MEC 服务必须透明地集成到现有的体系结构中。第一个要求是, 迁移必须在用户和 MEC 服务器之间进行协商, 有两个原因: 首先, MEC 服务器的计算资源和上行线路带宽是多个用户共享的; 因此, 为了确保服务质量, MEC 服务器根据可用的计算和上行带宽资源, 来决定他们愿意提供多少资源。其次, 网络运营商不向移动用户收取额外流量费用, 这意味着在 ME-VoLTE 体系结构中, 网络计费功能作为协商过程的一部分, 必须被集成到架构中。

第二个关键要求是适应移动性。编码任务最初是被迁移到与用户非常接近的 MEC 服务器, 如果用户位置移动, ME-VoLTE 必须能检测用户位置发生的变化, 并验证编码参数是否需要调整以适应新的环境。

第三个要求是系统必须考虑网络接口的功耗。由于用户编码工作量减少, 视频数据的压缩率远低于传统的 VoLTE 系统。将这些仅微压缩的媒体流发送到边缘基站和 MEC 服务器时, 会导致用户的传输速率增加。由于 ME-VoLTE 旨在降低应用程序的功耗, 所以这里考虑的关键因素之一是对多媒体流的迁移能力。因此, ME-VoLTE 必须确保卸载不会增加 UE 的功耗。

#### 4.1.4 移动终端对等点之间的计算迁移

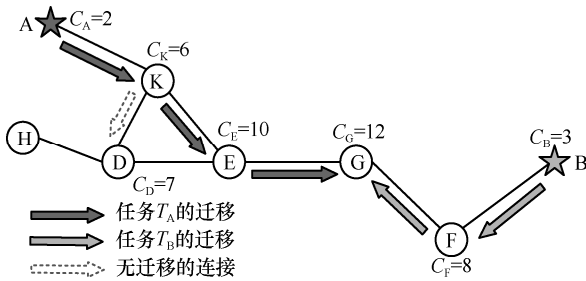
Gao<sup>[39]</sup>提出了一个计算迁移架构, 在网络边缘的移动设备之间迁移计算任务。在战区环境内,

处理区域环境的监测数据的应用需要大量的计算资源。该架构将部分应用程序迁移到附近的移动节点执行, 以减少计算时间和能源消耗。节点的迁移方案依赖于计算能力、相邻节点的能量级别以及节点之间未来可能发生的交互。通过分析移动节点间接触时间分布特性, 可以预测两个节点之间的可能发生的交互。根据这个框架, 在迁移发生之前, 系统会自动计算新节点上任务的执行时间、能量消耗和节点间未来的交互。如果在新节点上执行任务节省了时间和能耗, 并且新节点能确保在接触周期内完成任务, 则新节点将被确定为任务迁移节点。该框架具体做出的贡献如下。

- 通过分析确保了移动节点之间计算的迁移, 减少了能耗, 允许递归计算迁移进一步提高了网络资源的利用效率。
- 还研究了及时将计算结果返回给任务发起者, 以确保在大型网络范围内及时完成工作。
- 同时还考虑计算和操作数据的位置, 允许多个移动节点在计算执行期间相互协调, 以提高网络资源利用率。

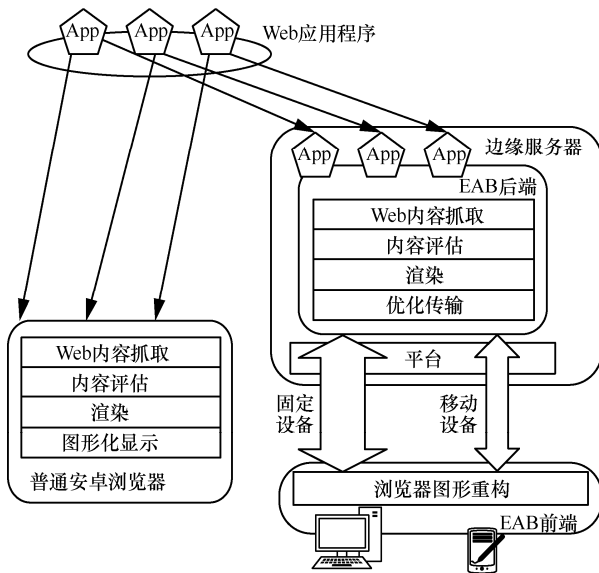
图 9 表示端到端的任务迁移框架, 每个移动节点的计算能力都已给出, 两个移动节点之间的连线表示两个节点之间的接触模式。当节点 A 连接的节点 K 的计算能力远高于 A 时, A 将计算任务  $T_A$  迁移到 K, 这样能以更短的时间、更少的能量消耗来完成任务  $T_A$ 。这种计算能耗的节省超过了 A 和 K 之间的数据传输成本, 从而提高了网络资源利用的整体效率。然后, K 作为任务  $T_A$  的“子发起者”, 递归地将  $T_A$  的工作量迁移到 E。而 K 与 D 连接时不会将  $T_A$  迁移到 D, 即使  $C_D > C_K$ 。主要原因是 K 和 D 之间的接触频率较低, 会降低  $T_A$  的计算结果及时交付到 A 的可能性。

该模型的性能取决于边缘网络中相邻节点的数量。如果节点网络中断, 模型则不会为其执行迁移任务。

图9 端到端任务迁移框架<sup>[39]</sup>

#### 4.1.5 边缘网页浏览加速模型

Takahashi 等<sup>[40]</sup>提出了 EAB (edge accelerated Web browsing) 的移动边缘计算原型, 以加速 Web 应用程序的执行。图 10 对比了普通安卓浏览器 (左) 与 EAB (右) 的架构。在 EAB 中, 边缘服务器被部署在移动客户端以及移动端附近的服务器之间。当移动 Web 浏览用户向 URL 页面发送请求时, 服务器端的响应首先在边缘服务器上被拦截。作为响应, 边缘服务器会过滤一些内容, 边缘服务器的任务一般是抓取 Web 内容、评估 Web 内容、Web 的重新布局和呈现。这个模型比普通的 Web 浏览模型具有更好的效果。

图10 普通安卓浏览器 (左) 与 EAB (右) 架构对比<sup>[40]</sup>

#### 4.1.6 上下文感知协同实时应用

Nunna 等<sup>[41]</sup>提出了基于 5G 技术和移动边缘计算的上下文感知协同实时应用架构。由于无线

网络的高时延, 在某些特定时间内的传统协同应用具有弊端。在笔者所提的架构中, 边缘服务器部署于每个边缘节点上。作者利用了近距离服务、上下文感知计算等 5G 技术的特性来实现协同。边缘服务器的中间件—边缘协作平台通过标准化的 API 收集用户详细位置、无线网络状况等重要信息。该架构将应用于道路事故场景和机器人远程工作场景中。

5G 网络与 MEC 集成在一起, 将提供一个强大的实时上下文感知协作平台。5G 网络的低传输时延, 与 MEC 服务的低应用程序响应时间相结合, 将使系统非常适合为实时场景服务。5G 的近距离服务、上下文感知计算这些特性可以通过安全 API 在基站部署到 MEC 服务器上, 从而使上下文感知的临时协作变得可行。

集成 MEC 与 5G 构建的协作平台有望在功能上实现通用, 平台应该支持不同行业领域所对应的不同服务子系统。为了实现敏捷和可靠的应用程序部署, 必须在这些服务子系统中提供安全访问接口。

由于时延和同步是协同计算模型中性能的重要体现, 而上述模型具有低时延特性, 因此适合于协同计算。目前, 上述模型仍然只是一个理论概念, 5G 技术也还处于发展阶段。

#### 4.1.7 CloudAware

Gabriel 等<sup>[42]</sup>提出了 CloudAware 系统, CloudAware 是一种移动边缘计算的编程模型, 应用于开发弹性的和可伸缩的移动边缘计算应用程序。CloudAware 不同于 MEC 领域相关的其他方法, 它的主要设计目标是支持与附近设备的临时和短时间交互, 以及在没有可用的代理或者用移动设备作为回退机制的连接终端时, 提供一个持续可用的应用程序。

图 11 表示 CloudAware 的执行平台和架构, CloudAware 采用 Jandex<sup>[43]</sup>中间件框架来获取分布式、并发执行和上下文感知计算等属性。在



CloudAware 中,发现服务负责监控可用的网络、网络强度以及每个服务器所提供的代理计算资源。上下文管理器收集用户的任务迁移状态,以估计所需的网络连接。分割器和求解器将应用程序分为不同的组件,在运行时根据时间优化(即最小化计算时间)和网络连接状态选择最优的迁移策略。还包括一个提供同步的协调器。CloudAware 的设计可以实现通过并行化加速计算、通过迁移计算节省能量或带宽并且支持多种移动场景的迁移。

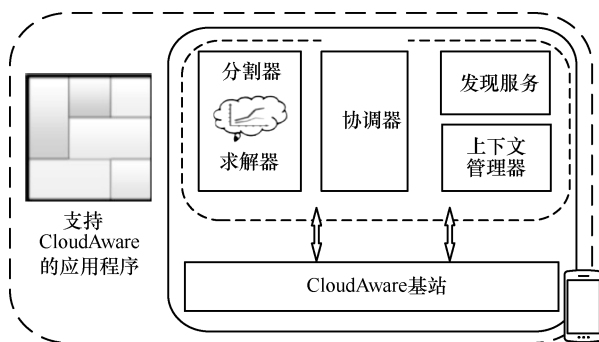


图 11 CloudAware 执行平台和架构<sup>[42]</sup>

表 6 对近年典型的边缘计算研究成果进行了比较。边缘计算的应用场景涉及互联网的方方面面,除了以上几个应用场景外,边缘计算在移动医疗、移动游戏、VR、AR、物联网传感数据异常实时检测<sup>[44]</sup>、边缘存储<sup>[25]</sup>以及物联网数据转发<sup>[45]</sup>等领域有广泛的应用。

学者们对边缘计算迁移系统架构的研究,主要目标集中在增强移动终端计算能力、降低能耗、缩短时延、提高网络资源利用效率等方面,采用的主要方法是优化迁移策略、优化传输效率等,特别是迁移策略的优化,是当前边缘计算研究的主要方向和重点领域。

#### 4.2 基于不同优化目标的边缘计算迁移研究

边缘计算迁移的节能研究需要移动边缘计算和无线通信技术的联合设计,其中最主要的部分是计算迁移策略及算法的优化。通过将计算迁移问题建模为优化问题,主要针对能耗、时延、传输代价、节点负载等。

随着移动设备的发展,终端设备寿命提升,降低能耗,提升用户体验质量。Zhang 等<sup>[46]</sup>研究了一个单用户 MECO 系统,通过比较本地计算(带有可变 CPU 周期)和优化后的迁移计算(可变传输速率)的能量消耗,得出最优迁移策略。Zhang 等<sup>[47]</sup>研究了用户终端移动模式在未知情况下的能耗优化,提出了基于卷积神经网络的深度 Q 网络的强化学习算法,从用户过去的状态中学习,实现能耗优化,缺点是没有对计算时间进行约束。Lyu 等<sup>[48]</sup>研究了移动边缘计算迁移面临的可扩展性问题,提出了一个轻量级的请求和准入框架来解决可伸缩性问题,设计了选择性迁移方案,以最小化设备的能量消耗。

表 6 典型的边缘计算研究成果比较

研究成果	发表时间	实现方法	设计目标
物联网传感数据异常实时检测 <sup>[44]</sup>	2018 年	建立基于边缘计算的分布式数据异常检测模型	提高物联网实时采集数据的质量
数据转发 <sup>[45]</sup>	2018 年	物联网数据转发模型	提升物联网数据转发性能
CloudAware <sup>[43]</sup>	2016 年	优化迁移策略	加速计算、节能
Replisom <sup>[37]</sup>	2016 年	克隆虚拟机、压缩	减少响应时间
Femtoclouds <sup>[36]</sup>	2015 年	优化迁移策略	增强计算能力
ME-VoLTE <sup>[38]</sup>	2015 年	优化传输	降低能耗
EAB <sup>[40]</sup>	2015 年	优化传输	加速资源访问
上下文感知协作实时应用 <sup>[41]</sup>	2015 年	上下文感知	在实时场景中降低时延
端到端计算迁移 <sup>[39]</sup>	2014 年	优化迁移策略	提高整个网络资源利用效率

移动终端增多的同时网络负载加重,可能引起网络时延。为了解决这一问题,结合能耗优化目标,参考文献[49-51]针对不同类型的多用户系统研究了 MECO 的资源划分。Stefania 等<sup>[49]</sup>研究了一个多用户的 MECO 系统,为了减少在迁移时延约束下的终端能量消耗,提出将无线和计算资源进行联合分配。Zhao 等<sup>[50]</sup>研究了在中心云和边缘云共存的情况下,迁移到不同云的最优用户调度问题,提出了一个基于任务负载的阈值迁移策略。Chen 等<sup>[51]</sup>研究了多用户 MECO 的分布式迁移,利用博弈论来实现能量和时延的最小化。Donghyeok 等<sup>[52]</sup>提出了一种将部分视频流量卸载到 Wi-Fi 网络来缓解蜂窝网络拥塞的软件定义网络架构,通过有效且公平地共享有限的蜂窝网络资源来提高所有用户的视频质量,提出了追求用户之间的全局系统效用和服务质量公平性的资源分配算法,该架构能够有效降低时延,但是对于终端能耗并没有有效的优化。Liu 等<sup>[53]</sup>提出了一种权衡迁移时延和可靠性的框架,设计基于启发式搜索、重构线性化技术和半定松弛 3 种算法来实现时延和可靠性的权衡,缺点是没有对能耗进行优化。

移动设备的增多使得网络带宽日益紧张。为解决这一问题,You 等<sup>[54]</sup>和 Mao 等<sup>[55]</sup>分别研究了自适应的计算迁移方法,并以无线能量传输和采集为目标,设计了基于无缝集成移动云计算和微波功率传输(microwave power transmission, MPT)两种技术的解决方案。Xiang 等<sup>[56]</sup>研究了单用户的 MECO 系统,将动态迁移与自适应的 LTE/Wi-Fi 链接选择进行集成,并提出了一种可扩展的近似动态规划(approximate dynamic programming, ADP)算法。

迁移算法优化方面,学者也做出努力。Chen 等<sup>[57]</sup>用 Lyapunov 技术开发了一种边缘基站之间的在线计算迁移框架,以最大化边缘计算系统的性能,但仅涉及了边缘基站之间的协作计算,并

没有优化终端的能耗。Zhang 等<sup>[58]</sup>研究了多流数据迁移的问题,一个终端有多个应用程序需要迁移,将多流数据迁移问题构建为限时空离散时间马尔可夫决策过程,通过基于动态规划的算法建立最优策略,但是该策略计算复杂度高,会在一定程度上损害性能。Muhammad 等<sup>[59]</sup>提出了一种基于激励的博弈论数据下载框架,实现了纳什均衡,但是该框架过分关注移动网络运营商的服务质量,而忽视了终端用户的能耗考虑。

Reza 等<sup>[60]</sup>探讨了迁移计算中的用户隐私安全问题,建议使用智能分区或动态迁移,应默认采用“本地优先”方法,在处理敏感数据时避免使用网络,但是缺乏对富应用程序复杂性的考虑。表7对相关的计算迁移策略的优缺点进行了总结。

综上所述,学者们关于边缘计算迁移策略的研究主要集中在算法设计上,大多数都是仅考虑了计算时延或终端能耗,一些研究的计算复杂度偏高,没有达到全局最优,并不能有效解决移动终端资源受限的问题,特别是能耗问题,因此当前研究中资源划分能耗优化的问题就凸显出来。

#### 4.3 边缘计算典型应用

边缘计算可应用于网页动态优化、物联网计算迁移、移动大数据分析、智能交通等场景。但是这些应用并不能在移动设备或便携式设备上有效地运行,这些应用是计算密集型的,对移动设备的处理能力和存储容量的要求很高。以下简要介绍边缘计算的几个应用场景。

##### 4.3.1 网页动态优化

传统的网页优化为了满足客户的期望,一般在 Web 站点上执行,并参考存储在数据库中的用户 Web 浏览历史<sup>[61]</sup>。网页优化也可以通过获取用户当前的地理位置并分析用户位置信息,根据用户的上下文感知来动态地<sup>[62]</sup>完成。而在移动边缘计算中,网页优化器可以托管在边缘服务器上,这样网页内容优化器能够动态地获取准确的位置和运行信息(网络负载、网络状态等),并根据获



表 7 相关迁移策略的优缺点

作者	发表时间	优化目标	创新点	存在不足
Zhang 等 <sup>[46]</sup>	2013 年	能耗	比较本地计算和迁移计算的能量消耗,得出最优迁移策略	基于单用户 MECO 系统,仅对比了迁移前后的能量消耗
Zhang 等 <sup>[47]</sup>	2016 年	传输代价	以无线能量传输为目标,设计了基于无缝集成移动云计算的解决方案	研究了低复杂度设备(如传感器),对于高计算复杂度的智能终端缺乏研究
Mao 等 <sup>[55]</sup>	2016 年	传输代价	以无线能量采集为目标,设计出了基于微波功率传输的解决方案	基于 CPU 频率和发射功率的优化,未综合考虑计算任务
Xiang 等 <sup>[56]</sup>	2014 年	算法优化	将动态迁移与自适应的 LTE/Wi-Fi 链接选择进行集成,提出了一种可扩展的近似动态规划算法	基于单用户的动态算法设计优化,缺乏全局最优考量
Stefania 等 <sup>[49]</sup>	2015 年	能耗	将无线和计算资源进行联合优化	缺乏严密的计算时延考量
Zhao 等 <sup>[50]</sup>	2015 年	负载均衡	研究了迁移到不同云的最优用户调度问题,提出了一个基于任务负载的阈值迁移策略	研究时延问题,未结合终端能耗研究
Chen 等 <sup>[57]</sup>	2015 年	节点负载	研究了多用户分布式计算迁移,运用博弈论来实现能量和时延的最小化	分布式的计算迁移算法得到局部最优
Zhang 等 <sup>[58]</sup>	2018 年	算法优化	基于马尔可夫决策和动态规划算法建立最优策略	计算复杂度较高
Reza 等 <sup>[60]</sup>	2018 年	隐私安全	使用智能分区和动态迁移研究了迁移计算中的用户隐私安全问题	未考虑富应用程序复杂性
Liu <sup>[53]</sup>	2018 年	时延	基于启发式搜索、重构线性化技术和半定松弛 3 种算法来实现时延和可靠性的权衡	缺乏对能耗的优化

取的信息进行内容优化<sup>[63]</sup>,如图 12 所示。基于边缘计算的内容优化提高了网络的性能,提高了体验的质量,并增加了新的服务。

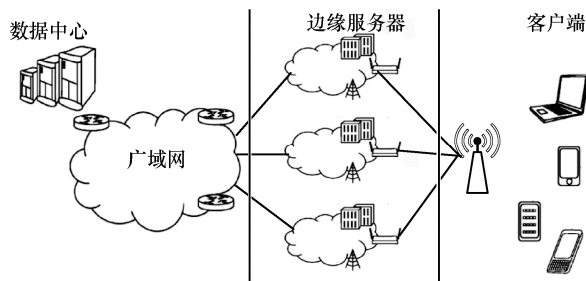


图 12 网页动态优化模型

#### 4.3.2 智能移动设备计算迁移

近年,可穿戴物联网设备流行起来,但是这些设备的处理能力非常低,传统的计算密集型应用(如 AR 和监视系统)无法在这些设备上执行。这个问题可以通过将应用程序拆分为小任务来解决,并且在云设备上执行拆分后的部分任务,但是这同时会引入网络时延和影响执行的准确性。

上述场景可以通过将任务迁移至边缘服务器上而非核心网络来优化,迁移至边缘服务器可以一定程度上减小时延<sup>[64]</sup>。移动边缘计算中的计算迁移面临着几个挑战:如何分割一个应用程序,如判断任务是否应该被迁移以及如何在用户地理位置发生变化时同步应用程序。

#### 4.3.3 边缘计算在大数据分析中的应用

大数据是大量数据的集合,对数据进行结构化和非结构化的分析,可以更好地做出商业决策和战略业务活动。近年来,手机上网成为主流,因此收集智能手机数据,并进行分析,是了解和分析市场的最有效途径<sup>[65]</sup>。在传统的大数据分析中,终端设备的数据被收集起来,然后转移到服务器上进行分析<sup>[66-67]</sup>,这个过程需要高带宽并且会引入时延。而移动边缘计算平台不再需要浪费大量带宽,就可以执行大数据分析,数据分析可以在网络边缘设备上,分析后,只需将结果返回服务器。因此,该场景将减少带宽消耗,提

高网络时延。

#### 4.3.4 智慧城市

现代化城市人口密集,城市生活面临着许多问题,智慧城市生活圈如图13所示。以交通为例,城市存在交通拥堵、道路状况不佳、停车位不足、公共交通能力不足、道路安全<sup>[68]</sup>等问题,智能交通是解决城市居民所面临的交通问题的有效途径。例如,在路边安装摄像头和传感器设备收集实时数据,可以通过边缘网络实现交通控制。传感器设备可以检测靠近设备的物体(如行人和车辆),并能测量物体的距离和速度,根据收集的数据,以合理的方式转换为智能红绿灯信号控制车流,从而实现交通管制。类似地,智能停车系统可以通过智能终端收集用户的位置信息,通过边缘网络上的设备查询附近的可用车库,用这些数据来建模,指导用户合理使用停车位。

#### 4.3.5 智慧海洋边缘计算

21世纪以来,随着我国对海洋空间资源的重视程度越来越高,对海洋空间的开发日渐频繁,但是海洋通信的相对落后严重制约了我国海洋战略的实施。传统的海上通信网络如海上无线通信、海洋卫星通信以及基于陆地蜂窝网络的岸基移动通信,存在通信制式互不兼容、通信带宽高低不

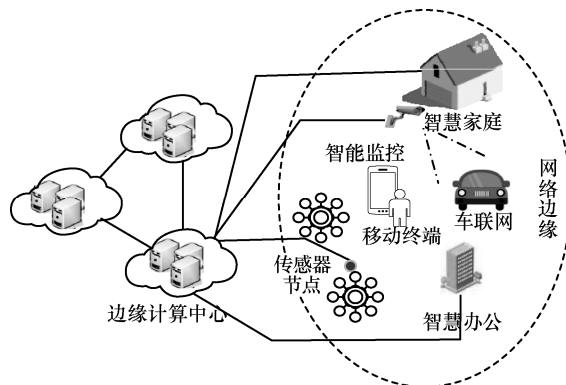


图13 智慧城市生活圈

一、覆盖范围存在盲区和缺乏高效统一的管理机制等缺点。新型智慧海洋通信系统在海上无线通信和卫星通信的基础上,以5G通信技术为核心,采用边缘移动通信,以岛礁等作为中继节点建立新的高速通信链路<sup>[69]</sup>。海洋边缘计算框架如图14所示。

### 5 TDMA多用户边缘移动终端计算迁移策略系统

基于移动终端资源受限问题的讨论与研究,笔者将进一步尝试提出边缘移动终端计算迁移的资源划分及迁移策略模型。

针对上文提出的当前边缘计算迁移策略研究中存在的如下几个问题:

- 大多数迁移策略都只考虑了计算时延或终

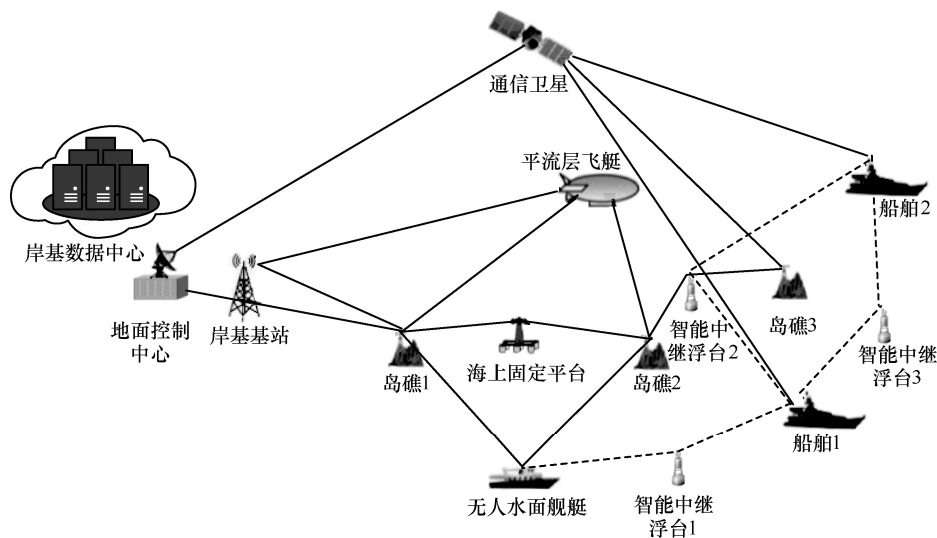


图14 海洋边缘计算框架<sup>[69]</sup>





端能耗;

- 部分研究的计算复杂度偏高;
- 没有达到计算时间和终端能耗的全局最优化。

针对以上问题, 本文尝试提出一个面向 LTE 应用的、基于时分多址 (time division multiple access, TDMA) 的多用户边缘移动终端计算迁移策略系统<sup>[70]</sup>, 该系统探讨了多个用户将任务迁移到一个边缘云基站的任务迁移模型, 系统模型如图 15 和图 16 所示。

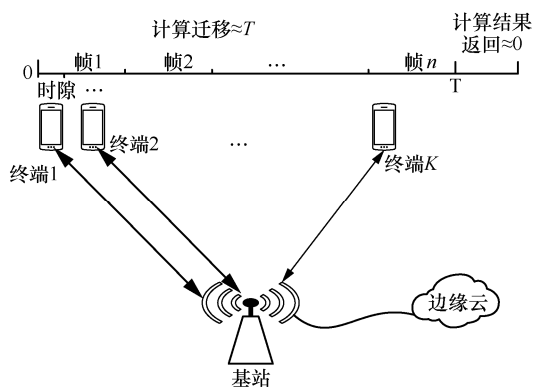


图 15 系统模型

系统模型为构建在蜂窝网络 (LTE) 环境下的基于时分多址的 MECO 系统, 将迁移时间  $T$  分割成互不重叠的时段 (帧), 再将帧分割成互不重叠的时隙 (信道), 每个时隙与一个用户具有一一对

应的关系, 系统依据时隙区分来自不同地址的用户信号, 从而完成多址连接的多用户单一边缘云的计算迁移, 该系统模型由  $K$  个索引为  $1, 2, \dots, k$  的终端用户和一个作为边缘云网关的基站构成。

当发生迁移计算时, 每个时隙包括两个连续的阶段:

- (1) 计算迁移或本地计算;
- (2) 云计算以及将边缘云的计算结果下载到移动终端。

由于边缘云服务器计算能力强, 数据在服务器上的计算时间几乎可以忽略; 并且计算结果相对较小, 结果返回的时间也可以忽略。由于以上原因, 与第一阶段相比, 第二阶段的持续时间可以忽略不计, 因此在资源划分时则不予考虑。

在任意时隙内, 基站基于 TDMA 对用户的一个任务子集进行完整/部分迁移调度。相比于博弈论的分布式任务调度方式, 基于基站的任务调度有一个显著的优点, 即任务信息集中到基站, 基站可以根据实时网络通信状态调整迁移策略, 从而避免分布式调度方式产生的额外系统开销。部分迁移或不迁移的用户分别使用本地 CPU 计算一部分或全部输入数据。

整个多用户迁移系统中的每个终端都相互联系, 由终端  $k$  来计算时隙内的  $D_k$  比特输入数据,

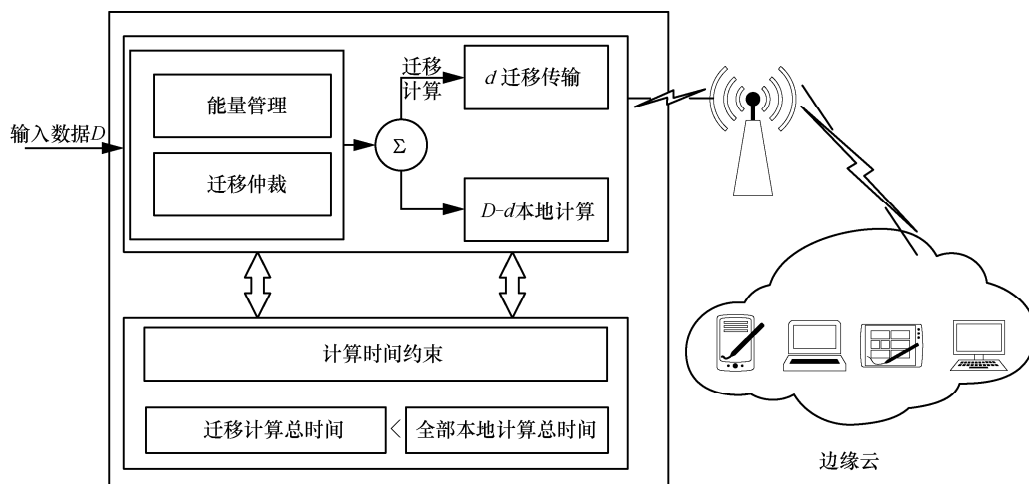


图 16 多用户边缘计算迁移系统

其中,  $d_k$  比特被迁移,  $(D_k - d_k)$  比特留在本地计算。用  $H$  表示信道增益 (无线信道对传输距离归一化后的衰落度),  $\sigma$  是复高斯白噪声的方差,  $p_k$  表示移动终端  $k$  的传输功率, 系统信道带宽为  $B$ 。根据香农定理, 构建终端  $k$  可达到的速率模型  $r_k$  为:

$$r_k = B \ln(1 + \frac{p_k H^2}{\sigma}), \forall k \quad (3)$$

将分配给终端  $k$  用于迁移的时隙长度表示为  $t_k$ ,  $t_k \geq 0$ , 其中  $t_k = 0$  对应于没有迁移的情况,  $a_k$  为迁移决策时间约束变量。用  $C_k$  表示在第  $k$  个终端计算 1 比特输入数据所需的 CPU 周期数,  $e_k$  表示用户本地计算的每个 CPU 周期的能量消耗。那么乘积  $C_k e_k$  就是每比特的计算所需能量。模型的详细介绍请参考参考文献[68]。资源划分模型可以构建为:

$$\begin{aligned} \min_{\{d_k, t_k\}} & \sum_{k=1}^K \left[ \frac{t_k \sigma \left( 2 \frac{r_k}{B} - 1 \right)}{H^2} + (D_k - d_k) c_k e_k \right] \\ \text{s.t.} & \begin{cases} a_k > 0' \\ \sum_{k=1}^K t_k \leq T, t_k \geq 0 \\ d_k \leq D_k, \forall k \end{cases} \end{aligned} \quad (4)$$

本文尝试将移动终端计算迁移的资源划分问题建模为一个凸优化问题, 采用拉格朗日乘子法进行求解。建立一个面向 LTE 应用的、基于 TDMA 的多用户边缘计算迁移系统, 研究了多个用户将任务迁移到一个边缘云基站的任务迁移模型。在系统的多用户计算迁移能耗凸优化问题中, 以计算时间最优为约束条件、以最小化终端本地能耗为目标来设计迁移方案, 最终确定一个最优的终端资源划分策略。本文提出的系统及其试验结果数据, 详情请参考参考文献[70]。

与已有的计算迁移策略相比, 本文提出的参考模型具有以下优势:

- 模型综合考虑网络带宽和通信质量的问

题, 可以根据实时网络状况选择迁移策略;

- 综合权衡计算时延和终端能耗, 保证了边缘迁移的计算质量;
- 拉格朗日方法求解凸优化问题, 可以显著减少迭代次数, 降低计算复杂度;
- 在模型的多用户计算迁移能耗凸优化问题中, 以计算时间最优为约束条件、以最小化终端本地能耗为目标来设计迁移方案, 可以实现计算时间和能耗的平衡, 达到系统最优。

## 6 结束语

在云计算发展的热潮之后, 许多云服务 (如移动医疗、移动学习、移动游戏和移动管理等) 都可以在移动设备上直接使用。移动边缘计算是一种新兴的技术, 它在移动用户的近距离范围提供云和 IT 服务。将用户的计算任务迁移到附近其他空闲的移动终端上, 达到节省终端能耗、减少网络传输费用等目的。

移动边缘计算平台通过在边缘网络上实现计算和存储来降低网络时延。它还允许应用程序开发人员通过使用实时无线访问网络信息, 提供上下文服务 (比如协作计算)。移动和物联网设备为计算密集型应用程序执行计算迁移, 如计算机视觉、移动游戏等, 使用移动边缘计算服务。本文介绍了边缘计算, 讨论了一些引发计算迁移的原因, 对计算迁移从分布式计算到边缘计算的发展概况进行了梳理, 介绍了几个移动边缘计算的应用场景。在此基础上, 尝试提出新的计算迁移策略模型以及未来的研究方向。

## 参考文献:

- [1] PAVEL M, ZDENEK B. Mobile edge computing-a survey on architecture and computation offloading[J]. IEEE Communications Surveys and Tutorials, 2017(99): 1.
- [2] 关沫. 复杂网络中的计算迁移问题[D]. 沈阳: 东北大学, 2005. GUAN M. Computing migration in complex networks[D]. Shenyang: Northeastern University, 2005.



- [3] 徐羽琼, 谌宗佳, 潘纲, 等. TaskShadow-V: 基于虚拟化的跨移动设备用户任务迁移[J]. 软件学报, 2011, 22(2): 129-136.  
XU Y Q, SHEN Z J, PAN G, et al. TaskShadow-V: user task migration across mobile devices based on virtualization[J]. Journal of Software, 2011, 22(2): 129-136.
- [4] HUERTA C G, LEE D. A virtual cloud computing provider for mobile devices[C]//ACM Workshop on Mobile Cloud Computing and Service: Social Networks and Beyond, June 15, 2010, San Francisco, California, USA. New York: ACM Press, 2010: 1-5.
- [5] 张文丽, 郭兵, 沈艳, 等. 智能移动终端计算迁移研究[J]. 计算机学报, 2016, 39(5): 1021-1038.  
ZHANG W L, GUO B, SHEN Y, et al. Mobile offloading on intelligent mobile terminal[J]. Chinese Journal of Computers, 2016, 39(5): 1021-1038.
- [6] 李子姝, 谢人超, 孙礼, 等. 移动边缘计算综述[J]. 电信科学, 2018, 34(1): 87-101.  
LI Z S, XIE R C, SUN L, et al. A survey of mobile edge computing[J]. Telecommunications Science, 2018, 34(1): 87-101.
- [7] ALLEN G, DRAM L T, FOSTER I, et al. Supporting efficient execution in heterogeneous distributed computing environments with cactus and globus[C]//2001 ACM/IEEE Conference on Supercomputing (CDROM), Nov 10-16, 2001, Denver, Colorado. New York: ACM Press, 2001: 52.
- [8] MAHADEV S. Pervasive computing: vision and challenges[J]. IEEE Personal Communications, 2001, 8(4): 10-17.
- [9] JEFFREY D, SANJAY G. MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008, 51(1): 107-113.
- [10] MARZ N. Twitter storm[Z]. 2012.
- [11] MATEI Z. Apache Spark[Z]. 2013.
- [12] OTHMAN M, HAILES S. Power conservation strategy for mobile computers using load sharing[J]. Mobile Computing and Communications Review, 1998, 2(1): 44-50.
- [13] RUDENKO A, REIHER P, POPEK G J, et al. Saving portable computer battery power through remote process execution[J]. Mobile Computing and Communications Review, 1998, 2(1): 19-26.
- [14] HUNT G C, SCOTT M L. The Coign automatic distributed partitioning system[C]//International Enterprise Distributed Object Computing Workshop, Nov 3-5, 1998, La Jolla, USA. Piscataway: IEEE Press, 1999: 187-200.
- [15] DEBASHIS S, AMITAVA M. Pervasive computing: a paradigm for the 21st century[J]. Computer, 2003, 36(3): 25-31.
- [16] 徐光祐, 史元春, 谢伟凯. 普适计算[J]. 计算机学报, 2003, 26(9): 1042-1050.  
XU G Y, SHI Y C, XIE W K. Pervasive/ubiquitous computing[J]. Chinese Journal of Computers, 2003, 26(9): 1042-1050.
- [17] EDUARDO C, ARUNA B, DAEKI C, et al. MAUI: making smartphones last longer with code offload[C]//International Conference on Mobile Systems, June 15-18, 2010, San Francisco, USA. New York: ACM Press, 2010: 49-62.
- [18] RAJESH K B, MAHADEV S, PARK S Y, et al. Tactics-based remote execution for mobile computing[C]//International Conference on Mobile Systems, May 5-8, 2003, San Francisco, USA. New York: ACM Press, 2003: 273-286.
- [19] JASON F, MAHADEV S. Balancing performance, energy, and quality in pervasive computing[C]//The 22nd International Conference on Distributed Computing System, July 2-5, 2002, Vienna, Austria. Piscataway: IEEE Press, 2002: 217-226.
- [20] STEVEN O, DINESH S, GONG S, et al. The design and implementation of Zap: a system for migrating computing environments[J]. ACM SIGOPS Operating Systems Review, 2002, 36(S1): 361-376.
- [21] ELI T, YANNIS S. J-orchestra automatic Java application partitioning[C]//16th European Conference on Object-Oriented Programming, June 10-14, 2002, Malaga, Spain. Heidelberg: Springer, 2002: 178-204.
- [22] SU Y Y, FLINN J. Slingshot: deploying stateful services in wireless hotspots[C]//The 3rd International Conference on Mobile Systems, Applications, and Services, June 6-8, 2005, Seattle, USA. New York: ACM Press, 2005: 79-92.
- [23] GU X H, KLARA N, ALAN M, et al. Adaptive offloading inference for delivering applications in pervasive computing environment[C]//1st IEEE International Conference on Pervasive Computing and Communications, Mar 26, 2003, Fort Worth, USA. New York: ACM Press, 2003: 107-114.
- [24] RICH W, SELIM G, CHANDRA K, et al. Using bandwidth data to make computation offloading decisions[C]//IEEE International Symposiums on Parallel and Distributed Processing, April 14-18, 2008, Miami, USA. Piscataway: IEEE Press, 2008: 1-8.
- [25] 崔勇, 宋健, 缪葱葱, 等. 移动云计算研究进展与趋势[J]. 计算机学报, 2017, 40(2): 273-295.  
CUI Y, SONG J, MIAO C C, et al. Mobile cloud computing research progress and trends[J]. Chinese Journal of Computers, 2017, 40(2): 273-295.
- [26] BUYYA R, YEO C S, VENUGOPAL S, et al. Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility[J]. Future Generation Computer Systems, 2009, 25(6): 599-616.
- [27] TIM V, PITER S, FILIP D T, et al. Cloudlets: bringing the cloud to the mobile user[C]//The 3rd ACM Workshop on Mobile Cloud Computing and Services, June 25, 2012, Low Wood Bay, Lake District, UK. New York: ACM Press, 2012: 29-36.
- [28] RA M R, SHETH A, MUMMERT L, et al. Odessa: enabling interactive perception applications on mobile devices[C]//The 9th International Conference on Mobile Systems, Applications and Services, June 28-July 1, 2011, Bethesda, USA. New York: ACM Press, 2011: 43-56.
- [29] CHUN B G, IHM S, MANIATIS, et al. CloneCloud: elastic execution between mobile device and cloud[C]//6th ACM

- EuroSys Conference on Computer Systems, April 10-13, 2011, Salzburg, Austria. New York: ACM Press, 2011: 301-314.
- [30] 柳兴, 李建彬, 杨震, 等. 移动云计算中的一种任务联合执行策略[J]. 计算机学报, 2017, 40(2): 364-376.
- LIU X, LI J B, YANG Z, et al. A Task collaborative execution policy in mobile cloud computing[J]. Chinese Journal of Computers, 2017, 40(2): 364-376.
- [31] 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.
- SHI W S, SUN H, CAO J, 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.
- [32] RAHIMI M R, REN J, LIU C H, et al. Mobile cloud computing: a survey, state of art and future directions[J]. Mobile Networks and Applications, 2014, 9(2): 133-143.
- [33] PAVEL M, ZDENEK B. Mobile edge computing-a survey on architecture and computation offloading[J]. IEEE Communications Surveys and Tutorials, 2017(99): 1.
- [34] 吕华章, 陈丹, 范斌, 等. 边缘计算标准化进展与案例分析[J]. 计算机研究与发展, 2018, 53(3): 487-511.
- LV H Z, CHEN D, FAN B, et al. Standardization progress and case analysis of edge computing[J]. Journal of Computer Research and Development, 2018, 53(3): 487-511.
- [35] 邓晓衡, 关培源, 万志文, 等. 基于综合信任的边缘计算资源协同研究[J]. 计算机研究与发展, 2018, 55(3): 449-477.
- DENG X H, GUAN P Y, WAN Z W, et al. Integrated trust based resource cooperation in edge computing[J]. Journal of Computer Research and Development, 2018, 55(3): 449-477.
- [36] HABAK K, AMMAR M, HARRAS K A, et al. Femto clouds:Leveraging mobile devices to provide cloud service at the edge[C]// 2015 IEEE 8th International Conference on Cloud Computing (CLOUD), June 27-July 2, 2015, New York, USA. Piscataway: IEEE Press, 2015: 9-16.
- [37] ABDELWAHAB S, HAMDAR B, GUIZANI M, et al. Replisom: disciplined tiny memory replication for massive IoT devices in LTE edge cloud[J]. Internet of Things Journal, 2015(99): 1.
- [38] BECK M T, FELD S, FICHTNER A, et al. Me-VoLTE: network functions for energy-efficient video transcoding at the mobile edge[C]// 2015 18th International Conference on Intelligence in Next Generation Networks (ICIN), Feb 17-19, 2015, Paris, France. Piscataway: IEEE Press, 2015: 38-44.
- [39] GAO W. Opportunistic peer-to-peer mobile cloud computing at the tactical edge[C]//Military Communications Conference (MILCOM), Oct 6-8, 2014, Baltimore, USA. New York: ACM Press, 2014: 1614-1620.
- [40] TAKAHASHI N, TANAKA H, KAWAMURA R. Analysis of process assignment in multi-tier mobile cloud computing and application to edge accelerated Web browsing[C]// 2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), March 30-April 2, San Francisco, USA. Piscataway: IEEE Press, 2015: 233-234.
- [41] NUNNA S, KOUSARIDAS A, IBRAHIM M, et al. Enabling real-time context-aware collaboration through 5G and mobile edge computing[C]// 2015 12th International Conference Information Technology New Generations (ITNG), April 13-14, 2015, Las Vegas, USA. Piscataway: IEEE Press, 2015: 601-605.
- [42] ORSINI G, BADE D, LAMERSDORF W. Computing at the mobile edge: designing elastic android applications for computation offloading[C]// Wireless and Mobile Networking Conference, Oct 5-7, 2016, Munich, Germany. Piscataway: IEEE Press, 2016: 112-119.
- [43] POKAHR A, BRAUBACH L. The active components approach for distributed systems development[J]. International Journal of Parallel, Emergent and Distributed Systems, 2013, 28(4): 321-369.
- [44] 张琪, 胡宇鹏, 嵇存, 等. 边缘计算应用: 传感数据异常实时检测算法[J]. 计算机研究与发展, 2018, 53(3): 524-536.
- ZHANG Q, HU Y P, JI C, et al. Edge computing application: real-time anomaly detection algorithm for sensing data[J]. Journal of Computer Research and Development, 2018, 55(3): 524-536.
- [45] 李继蕊, 李小勇, 高雅丽, 等. 物联网环境下数据转发模型研究[J]. 软件学报, 2018, 29(1): 196-224.
- LI J R, LI X Y, GAO Y L, et al. Review on data forwarding model in internet of things[J]. Journal of Software, 2018, 29(1): 196-224.
- [46] ZHANG W, WEN Y, GUAN K, et al. Energy-optimal mobile cloud computing under stochastic wireless channel[J]. Wireless Commun, 2013, 12(9): 4569-4581.
- [47] ZHANG C, LIU Z, GU B, et al. A deep reinforcement learning based approach for cost- and energy-aware multi-flow mobile data offloading[J]. IEICE Transactions on Communications, 2018.
- [48] LYU X C, TIAN H, JIANG L, et al. Selective offloading in mobile edge computing for the green internet of things[J]. IEEE Network, 2018, 32(1): 54-60.
- [49] STEFANIA S, GESUALDO S, SERGIO B. Joint optimization of radio and computational resources for multicell mobile-edge computing[J]. IEEE Trans on Signal and Information Processing over Networks, 2014, 1(2): 89-103.
- [50] ZHAO T, ZHOU S, GUO X. A cooperative scheduling scheme of local cloud and internet cloud for delay-aware mobile cloud computing[C]//IEEE GlobeCOM Workshops, Dec 6-10, 2015, San Diego, USA. Piscataway: IEEE Press, 2015: 1-6.
- [51] CHEN X, JIAO L, LI W. Efficient multi-user computation offloading for mobile-edge cloud computing[J]. IEEE/ACM Transactions on Networking, 2015, 24(5): 2795-2808.
- [52] DONGHYEOK H, GI S P, SONG H J. Mobile data offloading system for video streaming services over SDN-enabled wireless networks[J]. ACM Multimedia Systems Conference, 2018: 174-185.
- [53] LIU J H, ZHANG Q. Offloading schemes in mobile edge computing for ultra-reliable low latency communication[J]. IEEE Access, 2018(99): 1.
- [54] YOU C, HUANG K, CHAE H C. Energy efficient mobile cloud



- computing powered by wireless energy transfer (extended version)[J]. IEEE Journal on Select Areas in Communications, 2016, 34(5): 1757-1771.
- [55] MAO Y, ZHANG J, LETAIEF K B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices[J]. IEEE Journal on Select Areas in Communications, 2016, 34(12): 3590-3605.
- [56] XIANG X, LIN C L, CHEN X. Energy-efficient link selection and transmission scheduling in mobile cloud computing[J]. IEEE Wireless Communications Letters, 2014, 3(2): 153-156.
- [57] CHEN L X, XU J, ZHOU S. Computation peer offloading in mobile edge computing with energy budgets[C]//GlobeCOM IEEE Global Communications Conference, Dec 4-8, 2018, Singapore. Piscataway: IEEE Press, 2018: 1-6.
- [58] ZHANG C, GU B, LIU Z, et al. Cost-and energy-aware multi-flow mobile data offloading using Markov decision process[J]. IEICE Transactions on Communications, 2018.
- [59] MUHAMMAD A, SHAFI U K, RASHID A, et al. Game-theoretic solutions for data offloading in next generation networks[J]. Symmetry Open Access Journal, 2018, 10(8).
- [60] REZA R, TIMOTHY J P, RONALD P, et al. NoCloud: exploring network disconnection through on-device data analysis[J]. IEEE Pervasive Computing, 2018, 17(1): 64-74.
- [61] SIMMONS G, ARMSTRONG G A, DURKIN M G. An exploration of small business website optimization: enablers, influencers and an assessment approach[J]. International Small Business Journal, 2011, 29(5): 534-561.
- [62] ZHU J, CHAN D S, PRABHU M S, et al. Improving Web sites performance using edge servers in fog computing architecture[C]//2013 IEEE 7th International Symposium on IEEE Service Oriented System Engineering (SOSE), March 25-28, 2014, Redwood City, USA. Piscataway: IEEE Press, 2014: 320-323.
- [63] ARIF A, EJAZ A. A survey on mobile edge computing[C]//10th IEEE International Conference on Intelligent Systems and Control, Jan 7-8, 2016, Coimbatore, India. Piscataway: IEEE Press, 2016.
- [64] ATZORI L, IERA A, MORABITO G. The internet of things: a survey[J]. Computer Networks, 2010, 54(15): 2787-2805.
- [65] SARUKKAI R R, MENDHEKAR A. Method and apparatus for accessing targeted, personalized voice/audio Web content through wireless devices: US, EP1388096B1[P]. 2010-05-19.
- [66] CHEN M, MAO S W, LIU Y H. Big data: a survey[J]. Mobile Networks and Applications, 2014, 19(2): 171-209.
- [67] DEV D, PATGIRI R. Dr. Hadoop: an infinite scalable metadata management for hadoop how the baby elephant becomes immortal[J]. Frontiers of Electrical and Electronic Engineering in China, 2016, 17(1): 15-31.
- [68] MADAKAM S, RAMASWAMY R. The state of art: smart cities in India: a literature review report[J]. International Journal of Innovative Research and Development, 2013, 2(12): 115-119.
- [69] 夏明华, 朱又敏, 陈二虎, 等. 海洋通信的发展现状与时代挑战[J]. 中国科学: 信息科学, 2017 (47): 677-695.  
XIA M H, ZHU Y M, CHEN E H, et al. The state of the art and challenges of marine communications[J]. SciSin Inform, 2017(47): 677-695.
- [70] 乐光学, 朱友康, 刘建生, 等. 基于拉格朗日的计算迁移能耗优化策略研究[J]. 电信科学, 2018, 34(12): 10-23.  
YUE G X, ZHU Y K, LIU J S, et al. Research on optimizing strategy of computing offloading energy consumption based on Lagrangian method[J]. Telecommunications Science, 2018, 34(12): 10-23.

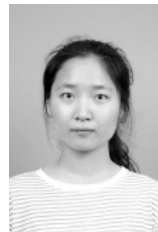
## [作者简介]



朱友康 (1992- ), 男, 江西理工大学硕士生, 主要研究方向为边缘计算和计算迁移。



乐光学 (1963- ), 男, 嘉兴学院教授, 主要研究方向为多云融合与协同服务、无线 mesh 网络与移动云计算、混成与嵌入式系统。



杨晓慧 (1996- ), 女, 江西理工大学硕士生, 主要研究方向为移动云计算。



刘建生 (1959- ), 男, 江西理工大学副教授, 主要研究方向为深度学习。