面向移动边缘的组合服务选择及优化

陈昊崴 邓水光 赵海亮 尹建伟

(浙江大学计算机科学与技术学院 杭州 310027)

摘 要 移动边缘计算作为新型的计算范式,为降低网络延迟、能耗开销提供了新的思路.其将中心云的强大算力下沉至网络边缘,使得用户能够将计算任务卸载至物理位置更近的边缘服务器执行,从而节省经由核心网的时延与能耗开销.然而,由于移动边缘计算技术通常受到计算资源、网络传输带宽、设备电量等因素的制约,如何在有限的资源中获取最大的利用率成为亟待解决的难题.此外,复杂的网络服务可以被抽象为由若干个子服务按照一定拓扑结构组成的组合服务,然而紊乱多变的移动网络环境为用户策略赋予了时空特性、决策耦合、边缘节点异构以及计算复杂度高的特性,使得传统的基于 QoS(Quality of Service)的算法不再适用.本文建立由异构边缘节点以及装配有能量收集组件的移动设备组成的移动边缘系统,基于李雅普诺夫优化以及马尔科夫近似提出一种多项式计算复杂度的分布式算法,提出 CSS(Composite Service Selection)框架,旨在联合优化服务选择策略以及能量存储策略,以此最小化整体组合服务请求的总体响应时间,并将设备电量稳定在一个可靠的水平.本文选取四种基准算法,实验结果表明 CSS 框架具备更加良好的性能,在时延上优于其他算法 7.76%~28.88%,并能够最快实现电量稳定.随着场景规模的扩大,CSS 将体现更优的性能.

关键词 移动边缘计算;服务组合;服务选择;李雅普诺夫优化;马尔科夫近似

中图法分类号 TP301 **DOI 号** 10.11897/SP.J.1016.2022.00082

Composite Service Selection and Optimization for Mobile Edge Systems

CHEN Hao-Wei DENG Shui-Guang ZHAO Hai-Liang YIN Jian-Wei

(*College of Computer Science and Technology, Zhejiang University, Hangzhou* 310027)

Abstract Mobile edge computing has emerged as the promising paradigm, devoted to provide innovative ideas in reducing latency and saving energy consumption. It enables users to offload computation tasks to edge servers in the proximity by pushing powerful functionalities of central cloud to network edge, so as to reduce overhead spent on transmission via core network, in terms of latency and energy. However, due to real-life restrictions, such as computing resources, caching capabilities and network transmission bandwidth of edge servers, edge nodes could only provision stringent-limited services for users. Moreover, battery capability of user equipment is hardware-constrained. It is crucial to explore the potential of obtaining full utility under insufficient resources with mobile edge computing technology. With the concept of microservice proposed, polybasic network services can be abstracted as composite services consist of multiple subservices with complex structure. As mobile devices evolved to be capable of processing more diversified mobile services, people tend to exploit it to handle daily business. On the other hand, network state in mobile environment is volatile and location-based which endows spatial and temporal properties on users' service-oriented strategies, making traditional QoS-aware algorithm is not applicable any more. Addressing this problem faces challenges of decision coupling, edge

收稿日期:2020-09-10;在线发布日期:2021-03-19.本课题得到国家自然科学基金(No. U20A20173, No. 61772461)和浙江省自然科学基金(No. LR18F020003)资助.**陈昊崴**,博士研究生,主要研究领域为移动边缘计算、服务计算.E-mail:haowei98@zju.edu.cn.**邓水光**(**通信作者**),博士,教授,计算机学会(CCF)高级会员,主要研究领域为移动边缘计算、服务计算.E-mail:dengsg@zju.edu.cn.**赵海亮**,博士研究生,主要研究领域为移动边缘计算、服务计算.**尹建伟**,博士,教授,主要研究领域为服务计算、遥感大数据.

node heterogeneity and high computation complexity. Most existing work does not take into consideration all of the above factors. To bridge this gap, in this paper, we consider a mobile edge system composed of heterogenous edge nodes and mobile users who generate composite services requests randomly at the begin of each time slot and are equipped with energy harvest components for sustainable computing with external energy conversion. To optimize latency and guarantee that battery power is maintained at a stable and reliable level, we have to determine the selection and offloading strategies of composite service requests and energy harvesting policies for mobile devices in each time slot, which has non-polynomial complexity. We formulate it as a stochastic optimization problem and proposes a new framework named CSS (Composite Service Selection) aiming at minimizing the overall response time of composite service requests generated by users in a mobile community and stabilizing battery energy in a reliable level, with the above constraints considered. First, it transforms the stochastic optimization problem over time slots into normal optimization problem in a given time slot, by introducing Lyapunov optimization. Then this problem could be decoupled into two subproblems as energy harvest problem and service selection problem. The closed-form expression of the energy harvest problem can be derived directly and next we develop a distributed method to solve the service selection problem, based on Markov approximation which has polynomial computation complexity. We take four baseline algorithms as benchmarks which are adapted from previous works. The experimental results demonstrate the superiority of our proposed framework compared with other baseline algorithms based on real-world dataset. It can obtain asymptotic optimality with low complexity and outperform other algorithms from 7.76% to 28.88% in term of latency. Moreover, mobile devices utilizing CSS are the fastest to reach stability. As problem scales, CSS outperforms other algorithms from 3.7% to 55.24%.

Keywords　mobile edge computing; service composition; service selection; Lyapunov optimization; Markov approximation

1　引　言

近年来,移动云计算在应对数据爆炸、时延要求突增的局面中开始展露疲势.尽管由许多大型服务器构成的中心云能够视作相对资源无限的数据中心,具备极短的计算延迟,但其与用户之间的远距离通信往往会造成延迟驱动型应用所无法承受的过大传输时延.伴随着移动边缘计算技术(Mobile Edge Computing, MEC)的兴起,用户能够选择将计算任务卸载到附近的边缘服务器上运行[1].尽管边缘服务器的轻量级特性使得服务社区甚至用户具备自行部署的能力,但计算资源受限性这一固有缺陷限制着用户资源调度的决策.在学术界,MEC 技术被广泛应用于其他学术领域,其提倡的边－端协作形式为网络服务发展注入了新的活力.

随着移动设备的智能化,人们通过智能手机、手提电脑等设备户外办公已是普遍的现象.每一次移动应用的使用,比如打车服务、扫码服务,均是智能设备调用网络服务的体现.在过去,单体应用将所有功能打包在一个独立单元,一个应用程序负责实现所有的业务功能,比如在线购物包括用户登录、商品查询、下单支付等业务环节.然而,随着用户数据量的激增与在线业务的扩展,单体应用架构的非弹性机制(修改某一个业务环节时必须对整个应用进行更新,某环节错误可能导致整个系统宕机),其面临着负载难以均衡、开发更新困难、应用容错性差等难题.随着"微服务"概念的提出,单体应用逐渐过渡到以实现的功能为依据将服务进行分类的微服务架构(比如将在线购物划分为用户服务、商品服务、订单服务等).每个微服务职责单一,将业务能力封装并提供服务.当用户试图体验一系列复杂的网络服务时,比如一次看电影体验,包括购票服务、支付服务、约车服务等,实质上为若干个微服务被整合为一个链式的组合服务.为完成一次组合服务请求,用户可能面临多种服务选择方案(比如支付服务可以通过

支付宝、微信、信用卡等多种不同方案完成),不同的子服务选择方案将直接影响用户的服务质量.对于网络服务而言,足够低的响应时间是服务质量的基本保障.

在复杂的移动环境下,网络质量(信道状态)动态多变,先序子服务的选择策略将影响后继子服务的网络环境.由于边缘服务器存在着一定的通信范围,其只能为无线局域网(WLAN)范围内的用户提供服务调用,用户在移动过程中,可能被迫切换与边缘服务器的通信,某一子服务的不同选择方案可能导致响应时间发生改变,进而影响后续任务的决策动作集合,因此用户的服务选择策略具有时空特性并且是一个全局优化问题.在实际场景中,相同服务在异构的边缘服务器上运行所获得的响应时长不尽相同,异构的存储资源意味着可部署的网络服务的数量不同.相比于一般的单任务调度问题而言,组合服务选择策略需综合连续时间轴上全局优化的考虑以及异构网络环境对子服务序列的影响.因此,如何在多用户的异构云-边-端协作场景下,为组合服务请求制定最优的服务选择策略已成为保证用户服务质量的关键问题.

除去时延优化外,降低能耗也是移动服务的重点研究对象.尽管 MEC 技术为服务调用提供了相对强大的计算资源,但是对于目前依靠电池供电的设备而言,一旦剩余电量不足以完成服务调用过程,其获得的计算性能可能因此受到损害.在工业界,一般采取使用大容量电池以及定期充电的形式来延长电池寿命,但同时无法避免地增加了硬件成本以及人工成本.此外,在偏远的户外工程中,供电尚且是难以实现的服务,更无需提及长期定时充电.文献[2]阐明,随着无线传感网络(WSN)以及信息通信技术(ICT)中物联网设备能耗的急剧增长,绿色计算技术因强烈的可持续发展需求得以不断发展.EH(Energy Harvesting)技术作为新兴的绿色计算技术能够在一定程度缓解充电困难问题[3][4].其通过 EH 组件捕获周围可再生能源并转换为电能储备,从而维持设备工作与持续运转.然而,当电量超出一定阈值时,电池的寿命可能受到损耗.因此,如何在特定场景下制定能量存储策略使得设备在可靠电量水平下持续工作是亟需解决的难题.

本文将用户的移动模式建模为随机的交替更新过程[5],引入能量收集模型[6],并模拟终端设备在数据传输、任务卸载时所花费的能耗开销以及新能源存储收益.针对用户在连续时间轴上接连生成的组合服务请求(单个用户可能生成多个组合服务请求),将用户的 QoS 模型表述为指数级求解空间的 NP-hard 联合随机优化问题.基于李雅普诺夫优化,为连续时间轴上的约束条件构造控制队列将随机优化问题具体化为单个离散时间片内的一般优化问题,使得策略空间得到缩减,求解难度降低.基于新的优化目标,将其解耦为能量存储问题以及服务选择问题.在利用数学工具求解出凸的能量存储问题后,针对难度为 NP-hard 的服务选择问题,设计场景特定的马尔科夫链,基于马尔科夫近似框架进行分布式求解,使得系统在低时间复杂度内收敛至近似最优解.

本文所做出的主要贡献以及创新点如下:

(1)将用户的移动模式建模成交替更新过程,聚焦用户移动过程中的时间特性来仿真用户移动时的服务调用和卸载通信过程;引入能量收集模块,综合考虑设备的能源收益与能源开销.

(2)对于求解上述问题提出了一种组合服务选择框架 CSS,旨在考虑边缘异构性、资源受限性、电池受限性以及用户移动性的情况下,基于随机优化模型,解决大规模(多用户、多服务器、多种服务请求)场景下的总体服务质量优化问题.

(3)基于真实数据,将提出的服务选择框架与四种服务选择策略进行对比.仿真实验结果证明,提出的新型服务选择框架具备更优异的性能,并随着系统规模的扩大能够展现出更加优良的鲁棒性.

本文第 2 节介绍相关工作;第 3 节引入系统模型并数学表述研究问题;第 4 节阐述求解算法并展示仿真实验结果;第 5 节为总结与未来工作的展望.

2 相关工作

随着移动边缘计算的兴起,移动服务计算得到相应发展,网络服务架构不断多元化.文献[7]指出,从服务对象的角度出发,目前的三大主体可以分为:云,边,端三类."云"指代能够部署和供应服务的云中心服务器,"边"指代与云服务类似,但是计算能力和计算资源相对受限的边缘服务器,"端"则指代提出服务需求,并能完成服务交互的终端智能设备.在移动边缘计算的研究中,通常涉及到诸多变量,设备能耗、总体时延、资源调度一般作为优化目标,此外又需满足电池容量约束、时延约束以及计算资源约束等条件.

在以往对云-端,端-端模式的研究中,Sard-

ellitti 等人在文献[8]中基于连续凸近似技术,以最小化 MIMO 小区中带有延迟约束的用户的能量消耗为目标,在迭代算法中实现优化. Zhao 等人以无线电资源和用户计算资源为决策变量,采用启发式算法实现移动设备能耗降低[9]. Peng 等人研究众包场景下的服务选择问题,并提出了名为 CrowdService 的选择框架[10]. Wu 等人针对移动社区中用户间资源共享的服务问题提出收益驱动型机制[11]. 在端一端协作的 D2D 形式场景中,Deng 等人利用磷虾群智能算法,来模拟移动社区内用户既作为服务供应者又作为服务请求者的近似最优服务交互[12].

近年来针对 MEC 场景下边一端架构的服务计算探索也取得了不错的进展. 在单任务服务计算优化中,Zhou 等人针对多用户多边缘服务器下以卸载策略及资源分配策略为变量的联合优化问题,在满足用户 QoE 约束的情况下,最小化系统的总体时延,但是其构造的为静态场景,没有考虑到用户的移动性[13]. Chen 等人考虑用户间信道干扰因素,以瑞利衰弱模型对信道建模,进行了传输功率,服务器选择协同优化,但对于边缘服务器的资源调度只是单纯地平均分配[14]. Zhao 等人基于李雅普诺夫优化对跨边缘协作计算提出了新的卸载框架,在将设备电池能量稳定在可靠级别的同时最小化时间开销[6]. 然而对于复杂的组合服务研究,当前正处于亟待探索的境地. Wu 等人在 MEC 场景中提出组合服务选择策略,提出组合服务的响应时间计算方式,以最小化服务响应时间为目标,结合遗传算法与退火算法,以启发式思想在低时间复杂度内收敛到近似最优解,但其采用的移动模型在实际场景中存在较大的局限性[15]. Deng 等人采用 TLBO 算法在移动环境下进行组合服务选择以得到最短的响应时间,然而没有考虑设备在通信过程中所产生的能耗影响[16].

在以往动态环境下的研究文献中,用户的移动可从空间特性以及时间特性两个角度进行分析. 文 Deng 等人采用 Random Waypoint Model 对用户移动路径进行切片[16],离散成若干个位置点,位置点之间的移动速率服从特定的随机分布,从而构造出时间与位置点之间的映射函数,在知晓时间的情况下能通过映射获得用户的具体位置. Zhao 将用户的移动模式建模为 Gauss-Markov Mobility Model[6],以一定周期进行位置更新. 文献阐明[17],在现实生活中用户的具体位置点是极难预测的,它可能受到社交关系、用户职业(比如学生,公司职员)以及其它社会属性的影响. 在时间特性的建模中,文献认为,用户与用户间的间隔通信时段分布能够被指数曲线良好拟合[18]. 用户与用户间的通信时段被证明可以近似为一个指数分布[5].

3　系统模型

本节将从网络模型,组合服务模型,能量模型以及移动模型四个方面详细介绍本文研究的场景与问题. 首先,构造网络环境框架,提出组合服务的具体定义,并给出服务质量的计算方法,最后引入能量收集机制与用户移动模式.

3.1　网络模型

在一个服务社区中存在着 n 个携带智能移动设备并以一定速度移动的用户,标记为集合 $U=\{u_1,u_2,\ldots,u_n\}$. 系统中部署着云服务器与边缘服务器两种能为用户提供服务调用的服务器类型. 假设存在 m 个基站(BS)作为无线接入点,每个基站旁均配有一个边缘服务器处理计算任务,将边缘服务器标记为集合 $E=\{e_1,e_2,\ldots,e_m\}$. 网络运营商可在服务器上部署若干种实现一定功能的网络服务. 由于基站辐射信号存在一定局限性,所以边缘服务器只为其对应的基站通信范围内的移动用户提供服务. 基站间通过 X2 链路两两相连,使得边缘服务器之间能够相互两两通信与协作,定义 B_{ij} 为 i-th 与 j-th 边缘服务器间的链路带宽.

根据文献,云服务器的设计存在两种类型[19],第一种为与边缘服务器相似的计算资源有限服务器,第二种为能够部署并处理各种服务,具备足够的计算能力,本文将云服务器视为第二种类型. 所有基站与云服务器无线接入,定义 L_{ic} 为 i-th 边缘服务器到云数据中心的时延,为保持模型的科学性,该值要大于边缘服务器之间的传输数据的时间开销.

具体场景如图 1 所示,用户 1 与用户 2 在移动过程中加入或退出仅部署有部分服务的边缘服务器通信范围,但无论用户如何移动均能够与云中心服务器进行通信. 将时间轴离散化为长度为 τ 的连续时间片,表示为 $\tau=\{1,2,3,\cdots\cdots\}$. 用户携带配有 EH 组件的智能移动设备,在每个时间片起始时刻从外界获取可再生能源并转换为一定大小的能量,以此保障设备在下一时间片处理服务请求时具备足够的能量用以服务调用.

用户服从一定的概率模型在边缘社区中移动,随机切换与基站之间的通信,并在每个时间片开始

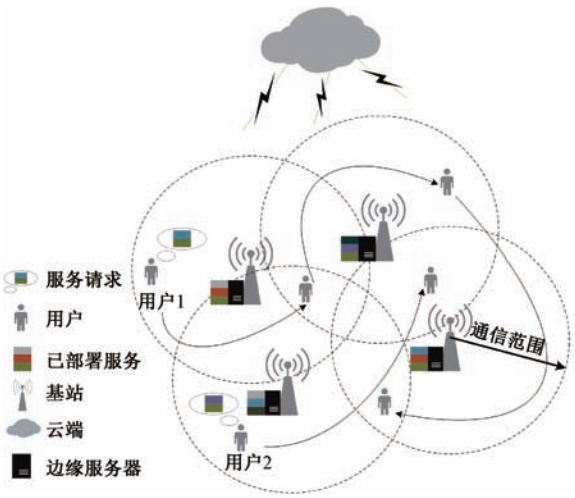

图 1 场景示意图

阶段以一定概率生成服务请求.为保证获得尽可能短的响应时间,用户需对组合服务请求制定合适的选择策略,并将服务请求卸载到通信范围内的服务器上执行,最后由服务器返回计算结果.

3.2 组合服务模型

为一目了然组合服务请求结构,对于与服务相关的变量定义,本节均采用元组形式表示.

定义 1. 任意组合服务请求均用一个三元组表示 $sr=(cs,t,u)$,其中:

(1)cs 表示该服务请求对应的组合服务类型;

(2)t 表示生成服务请求时的时间片位序;

(3)u 表示产生该请求的用户.

将一次组合服务依据用户所需体验的功能类型进行分类(如看电影服务包括在线购票,付款与打车).在一个离散化的时间轴上,为每次生成的服务请求定义 t 与 u 分量有助于对特定时间片与特定用户所获得的服务质量进行深入分析.

定义 2. 对任意组合服务采用一个三元组 $cs=(TA,R,S)$表示,其中:

(1)TA 为该类型组合服务所需要完成的子任务集合,$TA=\{ta_1,ta_2,\ldots,ta_h\}$;

(2)R 是表示 TA 中子服务之间关系的集合,$R=\{r(ta_i,ta_j)\,|\,ta_i,ta_j\in TA\}$;

(3)S 为该类型组合服务所对应的所有候选服务集合,$S=\{S_i\,|\,S_i=\{s_j\}_{j=1}^{M(i)}\}_{i=1}^{h}$ 对应的候选服务集合,$M(i)$为 S_i 所包含的候选服务个数.

组合服务本质为若干个存在依赖关系的子任务的组合形式,其中每个子任务实现特定的业务功能,对于每个子任务而言存在着若干种候选服务供予选择.所以组合服务的选择问题能够被具象化为对一系列的单个子任务进行候选服务选择.子任务之间存在着以下四种基本关系:序列,迭代,并行以及选择[11].

定义 3. 候选服务采用一个四元组 $s=(e,ex,p_i,p_o)$表示,其中:

(1)e 为部署有该候选服务 s 的服务器;

(2)ex 为 s 在该服务器 e 上对应的执行时间;

(3)p_i 为 s 在该服务器 e 上对应的输入数据参数;

(4)p_o 为 s 在该服务器 e 上对应的输出数据参数.

在异构的云-边-端协作系统中,服务器的不同决定了相同候选服务在执行时间、输入输出参数数值上的互异性.

对于组合服务请求 cs 中的每个子任务 ta_i,$\forall i\in h$ 而言,在给定候选服务 s 的选择策略的情况下,计算时延可以被划分为以下三个阶段:

(1)将服务请求卸载到服务器:

$$t_{off}=\sum_{j=1}^{m}x_{i,j}\frac{d_u}{r_{e_j}}+(1-\sum_{j=1}^{m}x_{i,j})\frac{d_u}{r_{e_c}} \tag{1}$$

其中 d_u 为上载数据量大小,r_{e_j} 与 r_{e_c} 分别指代 $j-$th 边缘服务器与云服务器的传输速率,$x_{i,j}=\{0,1\}$ 为二进制变量表示服务对服务器的卸载选择:

$$x_{i,j}=\begin{cases}1,i-\text{th 用户选择卸载 }j-\text{th 边缘服务器}\\0,\text{否则}.\end{cases}$$

单个候选服务不允许协作执行,即任意时刻单个候选服务只能在一个服务器上执行,即

$$\sum_{j=1}^{m}x_{i,j}\leqslant 1 \tag{2}$$

假定各用户与服务器间通信信道正交,卸载时无相互干扰.

(2)服务器执行:根据定义 3,在异构服务器上相同候选服务存在着不同的执行时间.对于给定的服务器 e,通过对当前服务 s 进行调用可以获得所需花费的执行时间开销 t_{ex}.

(3)服务器结果返回:在服务器 e_i 上执行完毕之后,会产生该服务的计算结果.如果此时用户在该服务器的通信范围内,可直接从 e_i 下载结果.否则,从覆盖当前用户的服务器集合中挑选下载开销最小的服务器 e_j 作为媒介获取结果.

$$t_{down}=\begin{cases}\dfrac{d_{res}}{r_{e_c}},\text{服务在云端执行}\\[2ex]\min\limits_{e_j\in E\cup e_c\setminus e_i}(t_{trans}^{e_i}+\dfrac{d_{res}}{r_{e_j}}),\text{否则}\end{cases} \tag{3}$$

其中 $t_{trans}^{e_i}$ 代表当前服务器与边缘服务器 e_j 之间的

通信开销,计算方式如下:

$$t_{trans}^{e_i}=\frac{d_{i,j}}{b_{i,j}} \tag{4}$$

其中 $d_{i,j}$ 为 e_i 与 e_j 之间的传输数据量大小,$b_{i,j}$ 为 e_i 与 e_j 间链路比特传输速率.

综上,对于组合服务 cs 中单个服务 s 的响应时间 t_s 计算方式可表示为

$$t_s=t_{off}+t_{ex}+t_{down} \tag{5}$$

对于不同结构的组合服务而言,其整体的服务质量计算方式是不同的,在本文中,定义运算符 $\vee$ 来整合各子任务的时间开销.所以,对于给定的组合服务 cs 而言,其全局服务质量计算公式可表示为

$$QoS_{cs}=\bigvee_{s\in S} t_s \tag{6}$$

每当生成组合服务请求,用户需从候选服务中选择具体的最优服务,以此获得全局最优的服务质量.

3.3　能量模型

移动设备处理服务请求的能力受限于本身的电池能量,而移动设备无法在使用过程中频繁地充电或者更换电池.随着万物智能化对可持续性工作的需求日益增加,物联网工程对绿色计算的需求催生了 Energy Harvesting (EH) 技术.它能够采集外界的可再生能源[3,4],如自然界的风能,潮汐能,光能以及人类行为能源,如人体热能,运动能,并转换为设备可储备的电能提高电池寿命.

EH 技术可以被建模为连续能源包到达的进程.在每个时间片开始时,i-th 用户设备中的 EH 组件生成上限为 $E_i^{h,\max}$ 的随机大小的能源包,标记为 $E_i^h(t)$,有

$$0\leqslant E_i^h(t)\leqslant E_i^{h,\max} \tag{7}$$

$E_i^h(t)$ 对不同的时间片服从独立同分布.部分能量包能够被设备以电能的形式存储,为下一时间片进行服务调用提供大小为 $\mu_i(t)$ 的能量开销,满足:

$$0\leqslant \mu_i(t)\leqslant E_i^h(t) \tag{8}$$

在给定一个组合服务选择的情况下,用户在进行组合服务调用时,对于其中某一确定服务组件 s 而言,负责三部分的能源开销:卸载,下载以及待机.

$$\varepsilon_{i,s}(t)=\varepsilon_{i,s}^{off}(t)+\varepsilon_{i,s}^{st}(t)+\varepsilon_{i,s}^{down}(t) \tag{9}$$

其中:

(1)$\varepsilon_{i,s}^{off}(t)$ 代表设备 i 卸载服务请求 s 以及相关数据至服务器过程中产生的能耗,可由以下公式计算得出:

$$\varepsilon_{i,s}^{off}(t)=p_i^{tx}t_{off} \tag{10}$$

其中 p_i^{tx} 为 $i-$th 用户的固定传输功率.

(2)$\varepsilon_{i,s}^{st}(t)$ 为设备 i 待机所花费的能耗,其计算结果与响应时长成正比:

$$\varepsilon_{i,s}^{st}(t)=p_i^{sp}t_s \tag{11}$$

其中 p_i^{sp} 为用户 i 的待机功率.

(3)$\varepsilon_{i,s}^{down}(t)$ 为用户 i 从服务器下载执行结果所花费的能耗,可由下式计算得出:

$$\varepsilon_{i,s}^{down}(t)=p_i^{tx}t_{down} \tag{12}$$

与服务质量整合规则同理,用户处理组合服务所产生的总体能量开销 $\varepsilon_i(t)$ 计算公式为

$$\varepsilon_i(t)=\bigvee_{s\in S}\varepsilon_{i,s}(t) \tag{13}$$

定义 $\chi_i(t)$ 为 $i-$th 用户在 t 时间片开始时的电池能量,其在当前时间片内处理服务请求花费的能耗不能超过当前设备所储备的能源,从 EH 组件中采集到的能源包部分将转换为电能处理下一时间片的服务请求,则有

$$\chi_i(t+1)=\chi_i(t)+\mu_i(t)-\varepsilon_i(t) \tag{14}$$

其中:

$$\chi_i(t)\geqslant\varepsilon_i(t) \tag{15}$$

与电池供电的计算系统相比,在采用 EH 技术的系统中进行服务选择策略设计将更加复杂,被赋予时间属性的电池能量使得用户决策在不同时间片内耦合.此外,$\mu_i(t)$ 作为新的决策变量将在一定程度上提高问题的求解难度.

3.4　移动模型

随着移动设备的智能化,移动服务不断得到普及,当今的研究开始重视用户的移动模式对系统决策带来的影响.在以往文献中,用户的移动可从两个角度进行分析:空间特性以及时间特性.

空间特性一般指通过建模用户移动轨迹可视化用户的移动方式,从而获取与用户地理位置相关的系统属性.在以往的工作中,用户的位置变化可通过一些算法进行建模,从而获得用户的移动轨迹[6,12,16,17].但是在现实生活中用户的具体位置点是极难预测的,它可能受到社交关系,用户职业(比如学生,公司职员)以及其它社会属性的影响.

时间特性指用户在移动过程中与时间相关联的变量,比如用户与基站之间的通信时长.为进行后续分析,本节为每对用户－基站对一一构造通信时间线如图 2 所示,由通信时段与间隔通信时段两部分组成,分别对应用户被该基站覆盖的时长,即该用户－基站对可相互通信的时段;以及两个连续通信时

段之间的时间间隔，即该用户－基站对无法通信的时段. 本文基于文献采用随机过程对用户移动模式的时间特性进行建模[5,20]，以此来获取用户的移动信息.

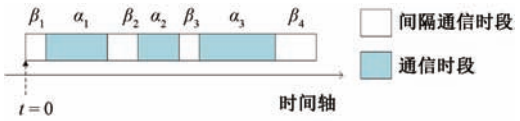

图 2 用户－基站对通信图

定理 1. 假设存在一个状态空间为{0,1}的随机过程，其在连续的时间轴上在 0 和 1 之间转换. 将系统停留在 0 和 1 的时间长度分别表示为 α_k 和 β_k，$k=\{1,2,3,\ldots\}$. 如果两个随机变量 α_k 和 β_k 均遵循独立同分布，则该随机过程被称为交替更新过程.

引理 1. 用户－基站对的通信过程是一个交替更新过程[5].

证明. 将用户－基站对的通信过程看作一个随机过程，用户在基站通信范围内记为状态 1，反之记为状态 0. 根据定理 1，将通信时段和间隔通信时段分别与 α_k 和 β_k 对应，如图 2 所示. 具体而言，当用户进入基站通信范围时，系统状态为 1，停留 α_k 时间后用户离开基站通信区域，状态切换成 0，并在 β_k 时间后用户再次与基站相遇，系统状态还原为 1，如此反复. 间隔通信时段分布能够被对数－正态分布曲线良好拟合，通信时段被证明可以近似为 Weibull 分布[20]. 因此，α_k 和 β_k 可以看作均服从独立同分布，所以用户－基站对的通信过程为一个交替更新过程.

证毕.

4 问题定义

用户在每个时间片开始阶段以一定概率生成服务请求，为了能够成功在该时间片内完成组合服务请求，令系统服务以下约束：

$$\tau_d \geqslant \max_{u\in U} QoS_{cs} \tag{16}$$

其中 τ_d 为组合服务 cs 的执行截止时间，为了不丧失一般性，τ_d 的最小取值应不小于当前组合服务均在云中心服务器上执行获得的响应时间. 此外，设置 $\tau_d \leqslant \tau$ 以保证每个服务请求均能在当前时间片内执行完毕.

当用户生成服务请求时，除对组合服务进行选择外，还需卸载到能带来更优体验的服务器上. 用户在移动过程中，会加入或离开某个基站的通信范围，直接影响用户能够选择卸载的服务器集合，进而影响最优服务选择策略. 因此，用户在制定服务选择策略时，必须考虑到移动性所带来的影响. 显然，这是一个多变量的全局联合优化问题. 将其用下列式子表述：

$$P1: \min_{\boldsymbol{\gamma},\boldsymbol{\varphi},\boldsymbol{\mu}} \lim_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\left[\sum_{i\in U} F(\boldsymbol{\gamma}_i(t),\boldsymbol{\varphi}_i(t))\right]$$
$$s.t.\ (7),(8),(15),(16) \tag{17}$$

其中 $\boldsymbol{\gamma}_i(t)$ 为 h 维向量，指代 i-th 用户在 t-th 时间片下针对服务请求做出的选择决策；$\boldsymbol{\varphi}_i(t)$ 为 h 维向量，对应 $\boldsymbol{\gamma}_i(t)$ 下的服务器卸载决策；$\mu_i(t)$ 为用户在当前时间片下的电量存储量；$F(\boldsymbol{\gamma}_i(t),\varphi_i(t))$ 代表该决策下对应的总体响应时间. $P1$ 中存在 $\boldsymbol{\gamma}_i(t),\boldsymbol{\varphi}_i(t),\mu_i(t)$ 三个决策变量，在给定 $t\in\tau$ 的情况下，具备指数级的策略搜索空间，并且 $P1$ 的优化目标为长期时间轴 τ 上的全局优化，具有非常高的求解难度，并且为 NP-hard 难度的优化问题，无法在多项式级别时间内得到最优解. 随着用户设备，组合服务，子任务数量的扩大，问题规模将呈指数级增加.

5 算法设计与分析

在原问题 $P1$ 中，系统可行解可以被解构成三个部分：服务选择决策，服务器卸载决策以及能源存储决策. 作为 NP-hard 问题，除枚举法外不存在任何算法能够求解得到最优解. 考虑到 $P1$ 作为随机优化问题(EH 组件的可再生能源随机过程，用户的随机移动模型)以及对连续时间轴直接优化具备很高的复杂度，本节首先基于李雅普诺夫优化构造虚拟队列将其具体化为单个时间片上的一般优化问题. 然后利用马尔科夫近似框架的分布式特性使得系统能够在无需先验知识以及未来时间片信息的情况下快速收敛得到近似最优解.

5.1 算法设计

对当前时间片内所有用户的电池能量情况进行整合，并将其构造为系统的虚拟队列 $\boldsymbol{\Theta}(t)\triangleq\{\chi_1(t),\chi_2(t),\ldots,\chi_n(t)\}$. 因为系统决策取决于电池能源的多少，是一个与时间相关联的变量，所以系统的决策集合具备时间依赖性，而原始版本的李雅普诺夫优化[21]只能在系统决策集合对于时间片呈独立同分布的情况下使用. 此处，引入 Weighted Perturbation Method 解决该问题[22]. 定义非负的摄动参数 $\boldsymbol{\theta}\triangleq\{\theta_1,\theta_2,\ldots,\theta_n\}$，并将虚拟队列相应更新为

$$\widetilde{\chi}_i(t)=\chi_i(t)-\theta_i \tag{18}$$

且满足:

$$\theta_i \geqslant \widetilde{E}_{\max}+\frac{V}{E_i^{\min}},$$

其中 $\widetilde{E}_{\max}\triangleq \min\{p_i^{sp}\tau_d+p_i^{tx}(\tau_d-ex),E_i^{\max}\}$,$E_i^{\max}$ 与 $E_i^{\min}$ 分别表示当前时间片该智能设备允许的放电量上下界.

接下来,引入李雅普诺夫函数,通过调整 $\widetilde{\chi}_i(t)$ 的权重和控制系数 V,在将电池能源维持在 $\boldsymbol{\theta}$ 左右的同时,求解原问题 $P1$.

$$L(\boldsymbol{\Theta}(t))\triangleq\frac{1}{2}\sum_{i\in U}(\chi_i(t)-\theta_i)^2=\frac{1}{2}\sum_{i\in U}\widetilde{\chi}_i(t)^2 \tag{19}$$

(19)为问题特定的李雅普诺夫函数,对其定义李雅普诺夫漂移为

$$\Delta(\boldsymbol{\Theta}(t))\triangleq E\left[L(\boldsymbol{\Theta}(t+1))-L(\boldsymbol{\Theta}(t))\,\middle|\,\boldsymbol{\Theta}(t)\right] \tag{20}$$

其中 $\Delta(\boldsymbol{\Theta}(t))$ 为虚拟队列储备量的变化量,其物理含义为下一时间片增加的电池能量,实质意义为系统对能耗开销处理能力的改变量. 由公式(14)可知:

$$\begin{aligned}\widetilde{\chi}_i(t+1)^2\leqslant&\widetilde{\chi}_i(t)^2+2\widetilde{\chi}_i(t)(\mu_i(t)-\varepsilon_i(t))\\&+(E_i^{h,\max})^2+(E_i^{\max})^2\end{aligned} \tag{21}$$

可以得到:

$$\Delta(\boldsymbol{\Theta}(t))\leqslant\sum_{i\in U}\widetilde{\chi}_i(t)(\mu_i(t)-\varepsilon_i(t))+B \tag{22}$$

其中 B 为常数值.

为将电池能源维持在一个稳定可靠的级别,即遵循当前时间片内的能耗约束(15),并保证原本的优化目标,即尽可能短的服务响应时间,构造新的目标函数为 $\Delta(\boldsymbol{\Theta}(t))+V\sum_{i\in U}F(\boldsymbol{\gamma}_i(t),\boldsymbol{\varphi}_i(t))$,其中 V 为控制系数. 在每一个时间片内,$P1$ 的近似最优解可以通过最小化其上界得出:

$$\begin{aligned}\Delta(\boldsymbol{\Theta}(t))+V\sum_{i\in U}F(\boldsymbol{\gamma}_i(t),\boldsymbol{\varphi}_i(t))\leqslant&\lambda(t)+\\\sum_{i\in U}\widetilde{\chi}_i(t)(\mu_i(t)-\varepsilon_i(t))+B\end{aligned} \tag{23}$$

其中 $B\triangleq\frac{n}{2}[(E_i^{h,\max})^2+(E_i^{\max})^2]$,至此,原问题 $P1$ 转化为问题 $P2$:

$$P2:\min_{\boldsymbol{\gamma},\boldsymbol{\varphi},\boldsymbol{\mu}}\sum_{i\in U}\widetilde{\chi}_i(t)(\mu_i(t)-\varepsilon_i(t))+B+\lambda(t)$$
$$s.t.\ (7),(8),(16) \tag{24}$$

其中 $\lambda(t)\triangleq V\sum_{i\in U}F(\boldsymbol{\gamma}_i(t),\varphi_i(t))$. 由于摄动参数的引入,约束(15)可被忽略. 接下来,提出 MA 算法进行求解,如算法 1 所示. 注意到此时连续时间轴上的随机优化问题 $P1$ 已经转换为在给定时间片上一般优化问题,策略搜索空间因为时间片变量 t 的剔除得到大幅降低.

算法 1. Drift-plus-penalty 算法.

1. 时间片 t 开始阶段,更新系统当前状态,获取独立同分布的随机事件信息,即用户生成的服务请求 $SR(t)$,EH 组件收集的能量包大小 $E^h(t)$ 以及用户的移动信息

2. 求解问题 $P2$ 以确定 $\boldsymbol{\gamma}_i(t),\boldsymbol{\varphi}_i(t),\mu_i(t),\forall i\in U$

3. 更新系统虚拟队列更新系统虚拟队列

4. $t\leftarrow t+1$

分析 $P2$ 可知,系统动作集可分为两类,一类为面向能源的动作,即能量存储策略;一类为面向服务的动作,包含服务选择策略以及服务器卸载策略. 两类动作互不干扰,因此 $P2$ 可被解构成两个子问题,对应为最优能量存储问题以及最优服务选择问题.

能量存储问题:通过对 $P2$ 进行分解,得到关于能源存储的子问题部分如下:

$$P_2^{EH}:\sum_{i\in U}\widetilde{\chi}_i(t)\mu_i(t) \tag{25}$$

显然,对于每个用户 i 的最优能量源存储的策略为

$$\mu_i^*(t)=E_i^{h,\max}\cdot\mathbb{R}\{\widetilde{\chi}_i(t)\leqslant 0\} \tag{26}$$

其中 $\mathbb{R}\{\cdot\}\in\{0,1\}$ 为逻辑函数,当且仅当 $\{\cdot\}$ 成立时 $\mathbb{R}\{\cdot\}=1$.

服务选择问题:相应的,$P2$ 剩下的子问题即为服务选择问题:

$$P_2^{SS}:\min_{\boldsymbol{\gamma},\boldsymbol{\varphi},\boldsymbol{\mu}}-\sum_{i\in U}\widetilde{\chi}_i(t)\varepsilon_i(t)+\lambda(t)+B$$
$$s.t.\ (7),(8),(16) \tag{27}$$

P_2^{SS} 是高度非凸的 NP-hard 问题,无法在多项式时间内得到最优解. 对于启发式搜索和进化算法而言,系统的搜索空间依然为 $\prod_{i\in U}\prod_{j\in h}M_i^t(j)$ 指数级,其中 $M_i^t(j)$ 指代时间片 t 时用户 i 的第 j 个子任务所对应的候选服务集合,并且每个用户的组合服务可能异构,导致启发式搜索和进化算法无法取得良好的性能. 利用马尔科夫近似框架的分布式特性,用户仅需知晓本地信息,从而避免在系统层面求解时用户间组合服务的异构性造成的影响. 此外,系统搜索空间被降维为单个用户的搜索空间,使

得系统能够快速收敛获得近似最优解.

Chen 等人针对求解网络中的联合优化问题,提出了马尔科夫近似框架[23].首先将原问题转换为等价的 MWIS(Minimum Weight Independent Set)问题,再利用 log-sum-exp 函数对其近似化处理,并构造时间可逆且问题特定的马尔科夫链,使系统在状态转移中收敛到渐进最优状态.

根据本文场景,定义 $f=\{\boldsymbol{\gamma},\boldsymbol{\varphi}\}$,$\forall f\in F$ 为系统状态,F 为满足约束条件的系统可行状态集合.为方便下文展开书写,将 P_2^{SS} 表示为一般的优化问题 $\min_{f\in F}x_f$,与 P_2^{SS} 等价的最小权重配置问题可以写成:

$$P_2^{SS}-EQ:\min_{\boldsymbol{\pi}\geqslant 0}\pi_f x_f \tag{28}$$

引入 log-sum-exp 函数[23]对 $P_2^{SS}-EQ$ 近似处理,得到 $P_2^{SS}-EQ$ 的带有一个多余熵项的近似优化问题 $P_2^{SS}-\beta$:

$$\begin{aligned}&P_2^{SS}-\beta:\min_{f\in F}\sum_{f\in F}\pi_f x_f+\frac{1}{\beta}\sum_{f\in F}\pi_f\log\pi_f\\&s.t.\ \sum_{f\in F}\pi_f=1\end{aligned} \tag{29}$$

其中 β 作为影响近似精确度的控制系数,所带来的近似差距为

$$\begin{aligned}&\min_{f\in F}x_f-\frac{1}{\beta}\log|F|\\&\leqslant-\frac{1}{\beta}\log\Big(\sum_{f\in F}\exp(-\beta x_f)\Big)\\&\leqslant\min_{f\in F}x_f\end{aligned} \tag{30}$$

当 $\beta\to\infty$ 时,近似差距为 0,能够得到:

$$\min_{f\in F}x_f\approx-\frac{1}{\beta}\log\Big(\sum_{f\in F}\exp(-\beta x_f)\Big) \tag{31}$$

虽然随着 β 的增长,log-sum-exp 函数的精确度越高.但是存在着一些收敛性考虑认为 β 取值应在一定范围之内.在实际中,越大的 β 值可能导致系统收敛速度越低.此外,对于非凸问题的求解而言,当 β 超过一定的取值时,系统将更容易陷入局部最优解.对于 $P_2^{SS}-\beta$,基于 KKT(Karush-Kuhn-Tucker)条件获得上述问题的最优解为

$$\pi_f^*(\boldsymbol{x})=\frac{\exp(-\beta x_f)}{\sum_{f'\in F}\exp(-\beta x_{f'})} \tag{32}$$

对上文构造的系统设计特定的且时间可逆的马尔科夫链.定义 $\pi_f^*(x)$ 为目标马尔科夫链的平稳分布,以此实现不同状态的分时处理.随着该马尔科夫链收敛至 $\pi_f^*(x)$,P_2^{SS} 能够以 $\log|F|/\beta$ 的优化差距被近似求解.至少存在一个时间可逆的可遍历马尔科夫链,其稳态分布为 $\pi_f^*(x)$[23].

$\forall f,f'\in F$,定义 $q_{f,f'}$ 为非负实数表示 $f\to f'$ 的系统转移速率.该马尔科夫链的设计必须满足以下两个条件[23]:

(1)任意两个状态相互可达;

(2)下述细致平衡等式恒成立:

$$\pi_f^*(\boldsymbol{x})q_{f,f'}=\pi_{f'}^*(\boldsymbol{x})q_{f',f} \tag{33}$$

$$\exp(-\beta x_f)q_{f,f'}=\exp(-\beta x_{f'})q_{f',f} \tag{34}$$

在文献中存在着许多种转移速率的设计方法[13,14,24],本文采用的设计为

$$q_{f,f'}+q_{f',f}=\exp(-\kappa) \tag{35}$$

其中 κ 为正常数.基于(33)—(35),得到转移速率的具体表达式为

$$q_{f,f'}=\frac{\exp(-\kappa)}{1+\exp[-\beta(x_{f'}-x_f)]} \tag{36}$$

$$q_{f',f}=\frac{\exp(-\kappa)}{1+\exp[-\beta(x_f-x_{f'})]} \tag{37}$$

显然,当 f 与 f'之间的性能差异越大,系统停留在更优配置 f'的概率 $q_{f,f'}$ 越大,这说明对于 $q_{f,f'}$ 的设计利于系统收敛到更优的状态.

算法 2 阐述了基于马尔科夫近似的具体算法步骤,并应用在前文的 CSS 框架之中求解问题 $P2$.在每个时间片开始阶段,系统获取发生的所有随机事件,网络环境以及队列储备量.首先系统进行状态初始化,对于每个用户,根据其生成的服务请求 $sr_i(t)$ 随机选择一种服务序列 $\boldsymbol{\gamma}_i(t)$并同时确定服务器卸载策略 $\boldsymbol{\varphi}_i(t)$,之后生成均值为 $\exp(\kappa)/N_i$ 的随机倒数时间,其中 N_i 为用户 i 的服务选择策略空间,并计算系统当前状态下的目标函数值 x_f(算法第 1 行到第 7 行).随后令所有用户开始时间倒数,一旦有用户停止则该用户告知其他用户停止倒数并改变自身状态,即随机执行另一种服务选择策略,用户根据设计的跳转概率判断是否跳转,并将更新状态(算法第 10 行至第 15 行).接下来为所有用户重新分配随机倒数时间(算法 17 行到 19 行).循环执行直到系统收敛.

算法 2. 马尔科夫近似(MA)算法.

输入:用户的服务请求集合 $\boldsymbol{SR}(t)$,虚拟队列储备量,网络环境以及移动信息

输出:$\forall i\in U$,服务选择决策 $\boldsymbol{\gamma}_i(t)$,卸载决策 $\varphi_i(t)$

1. FOR 用户 i IN U DO

2. 对组合服务 $sr_i(t)$随机候选服务选择决策来初始化 $\boldsymbol{\gamma}_i(t)$

3. 随机初始化服务器卸载策略 $\boldsymbol{\varphi}_i(t)$

4. 生成服从指数分布均值为 $\exp(\kappa)/N_i$ 的倒数计时器

5. END FOR

6. 当前状态标记为 f

7. 计算目标函数值 x_f

8. WHILE 系统尚未收敛 DO

9. 所有用户开始倒计时

10. IF 存在用户倒计时结束 THEN

11. 该用户标记为 i,广播告知其他用户停止倒计时

12. 用户 i 随机选择另一种选择方案以及卸载策略,并保证与旧状态只有一个变量不同

13. 标记当前状态为 f',并计算 $x_{f'}$

14. 用户 i 以 $q_{f,f'}=\exp(-\beta x_{f'})/\exp(-\beta x_{f'})+\exp(-\beta x_f)$ 的概率停留在 f'

15. 将当前状态标记为 f

16. END IF

17. FOR 用户 i IN U DO

18. 生成服从指数分布均值为 $\exp(\kappa)/N_i$ 的倒数计时器

19. END FOR

20. END WHILE

算法复杂度分析:在原问题中,策略搜索空间为 $\sum_{t\in\tau}\prod_{i\in U}\prod_{j\in h}M_i^t(j)$.通过李雅普诺夫优化对时间片因果约束构造虚拟队列,引入李雅普诺夫函数,$P1$ 被具体为单个时间片上的动作策略,状态空间缩减为 $\prod_{i\in U}\prod_{j\in h}M_i^t(j)$.在算法 2 中,每个最先倒计时结束的用户计算目标函数值只需要本地的配置信息而非其他用户的状态信息,使得每个时间片 t,每个用户在给定新的状态的情况下能够在以 $O(h)$的时间复杂度分布式执行.因此,通过引入李雅普诺夫优化以及马尔科夫近似,搜索空间得到大幅降低,使得系统能够快速收敛.

5.2　合理性分析

定理 2.上述算法设计满足构造时间可逆,平稳分布为(32)的马尔科夫链[29].

证明.定义当前状态为 f,用户为生成的组合服务请求制定选择决策以及卸载决策,同时生成服从均值为 $\exp(\kappa)/N_i$,$\forall i\in U$ 的指数分布随机倒数时间.假设用户 i 最先倒计时完成,并实现 $f\to f'$.

$$Pr(f\to f')$$

$$=\frac{1}{N_i}\cdot\frac{\exp(-\beta x_{f'})}{\exp(-\beta x_{f'})+\exp(-\beta x_f)}\cdot\frac{\frac{N_i}{\exp(\kappa)}}{\sum_{j\in U}\frac{N_j}{\exp(\kappa)}}$$

$$=\frac{1}{\sum_{j\in U}N_j}\cdot\frac{\exp(-\beta x_{f'})}{\exp(-\beta x_{f'})+\exp(-\beta x_f)}.$$

在该马尔科夫链中,用户 i 以 $\exp(\kappa)/N_i$ 的速率进行倒计时,即系统以 $\sum_{j\in U}N_i/\exp(\kappa)$ 的速率离开当前配置.此后,以 $Pr(f\to f')$ 的概率实现跳转.因此,$f\to f'$ 间的转移速率为

$$q_{f,f'}=\sum_{j\in U}\frac{N_j}{\exp(\kappa)}\cdot\frac{1}{\sum_{j\in U}N_j}\cdot\frac{\exp(-\beta x_{f'})}{\exp(-\beta x_{f'})+\exp(-\beta x_f)}$$

$$=\frac{\exp(-\kappa)}{1+\exp(-\beta(x_f-x_{f'}))}.$$

该式与式(36)一致,因此算法设计成立.将该式代入细致平衡等式中,平衡等式成立,所以该特定马尔科夫链时间可逆且平稳分布为式(32)所示.证毕.

5.3　扰动分析

在实际中,算法执行可能受到局部估计和移动环境中的噪声影响,无法获得精确的预期值 x_f,所以基于实际值进行跳转的马尔科夫链将收敛到与目标稳态分布 $\pi_f(\boldsymbol{x})$ 存在一定误差差距的稳态分布 $\bar{\pi}_f(\boldsymbol{x})$.基于工作[29]的贡献,本节以 $\bar{\pi}_f(\boldsymbol{x})$ 为研究对象,分析现实因素影响下的扰动马尔科夫链的优化差距.

定理 3.扰动误差的优化差距上界为(完整证明请见文献[29]附录 C):

$$2x_{\max}(1-\exp(-2\beta\Delta_{\max}))$$

证明.假设每个状态 $f\in F$ 的 x_f 均存在有界误差 Δ_f[25],所以实际值将扰动在 $[x_f-\Delta_f,x_f+\Delta_f]$ 范围内,将其离散化为 $2n_f+1$ 个取值:

$$\left[x_f-\Delta_f,\ldots,x_f-\frac{\Delta_f}{n_f},x_f,x_f+\frac{\Delta_f}{n_f},\ldots,x_f+\Delta_f\right].$$

假设状态 x_f 以 $\bar{\omega}_{j,f}$ 的概率取值为 $x_f+j\Delta_f/n_f$,并且满足:

$$\sum_{j=-n_f}^{n_f}\bar{\omega}_{j,f}=1 \tag{38}$$

以上述拓展后的实际值集合为系统状态集合构造新的马尔科夫链,原马尔科夫链的每个状态 $f\in F$ 将分别拓展为 $2n_f+1$ 个新状态 f_j.则 f 状态下的拓展马可夫链函数值可表示为 $(f,x_f+j\Delta_f/n_f)$,在经过倒计时后获得的新状态 f' 目标函数值为 $x_{f'}+j\Delta_{f'}/n_{f'}$.由(33)~(37)可得

$$\frac{\pi_{f_j}}{\pi_{f'_{j'}}}=\frac{\bar{\omega}_{j,f}\exp(-\beta x_f)}{\bar{\omega}_{j',f'}\exp(-\beta x_{f'})} \tag{39}$$

根据马尔科夫链状态间可达的性质,任意两个

状态 $f,f'\in F$ 间至少存在一条路径,则有

$$\frac{\pi_{f_0'}}{\bar{\omega}_{0,f'}\exp(-\beta x_{f'})}=\frac{\pi_{f_0}}{\bar{\omega}_{0,f}\exp(-\beta x_f)}=C \tag{40}$$

$$\frac{\pi_{f_j}}{\pi_{f_0}}=\frac{\bar{\omega}_{j,f}}{\bar{\omega}_{0,f}}\cdot\exp(-\beta\frac{j}{n_f}\Delta_f) \tag{41}$$

其中 f_0,f_0' 分别为 f,f'的无扰动状态且 C 为正常数.此外:

$$\sum_{f\in F}\sum_{j=-n_f}^{n_f}\pi_{f_j}=1 \tag{42}$$

根据式(39)~(42),因此拓展马尔科夫链中系统配置的平稳分布为

$$\bar{\pi}_f=\sum_{j=-n_f}^{n_f}\tilde{\pi}_{f_j}=\frac{\xi_f\cdot\exp(-\beta x_f)}{\sum_{f'\in F}\xi_{f'}\cdot\exp(-\beta x_{f'})} \tag{43}$$

其中 $\xi_f\triangleq\sum_{j=-n_f}^{n_f}\bar{\omega}_{j,f}\cdot\exp(-\beta j\cdot\Delta_f/n_f)$.定义 $\pi^*:[\pi_f^*,f\in F]$ 与 $\bar{\pi}:[\bar{\pi}_f,f\in F]$,二者的总变异距离为

$$\begin{aligned}d_{TV}(\pi^*,\bar{\pi})&=\frac{1}{2}\sum_{f\in F}|\pi_f^*-\bar{\pi}_f|\\&=\sum_{f\in\bar{F}}(\pi_f^*-\bar{\pi}_f)\end{aligned} \tag{44}$$

其中 $\bar{F}\triangleq\{f\in F\mid\pi_f^*\geqslant\bar{\pi}_f\}$,即

$$\frac{\pi_f^*}{\bar{\pi}_f}=\frac{\bar{\xi}}{\xi_f}\geqslant 1 \tag{45}$$

其中 $\bar{\xi}\triangleq\sum_{f'\in F}\xi_{f'}\exp(-\beta x_{f'})/\sum_{f'\in F}\exp(-\beta x_{f'})$,得到 $\forall f\in\bar{F}$:

$$\pi_f^*-\bar{\pi}_f=\frac{\exp(-\beta x_f)}{\sum_{f'\in F}\exp(-\beta x_{f'})}\cdot(1-\frac{\xi_f}{\bar{\xi}}) \tag{46}$$

定义 $\Delta_{\max}\triangleq\max_{f\in F}\Delta_f$,得到 $\forall j\in\{-n_f,\dots,n_f\}$:

$$\exp(-\beta\Delta_{\max})\leqslant\exp(-\beta\frac{j}{n_f}\Delta_f)\leqslant\exp(\beta\Delta_{\max}) \tag{47}$$

因此:

$$\begin{aligned}1-\frac{\xi_f}{\bar{\xi}}&\leqslant 1-\frac{\exp(-\beta\Delta_{\max})}{\exp(\beta\Delta_{\max})}\\&=1-\exp(-2\beta\Delta_{\max})\end{aligned} \tag{48}$$

得到:

$$\begin{aligned}d_{TV}(\pi^*,\bar{\pi})&=\sum_{f\in\bar{F}}(\pi_f^*-\bar{\pi}_f)\\&\leqslant\sum_{f\in F}\frac{\exp(-\beta x_f)}{\sum_{f'\in F}\exp(-\beta x_{f'})}\cdot(1-\exp(-2\beta\Delta_{\max}))\\&=1-\exp(-2\beta\Delta_{\max}).\end{aligned}$$

所以扰动误差的优化差距为

$$\begin{aligned}|\pi^*x^T-\bar{\pi}x^T|&=\Big|\sum_{f\in F}(\pi_f^*-\bar{\pi}_f)x_f\Big|\\&\leqslant 2x_{\max}\sum_{f\in F}|(\pi_f^*-\bar{\pi}_f)|\\&\leqslant 2x_{\max}(1-\exp(-2\beta\Delta_{\max}))\end{aligned} \tag{49}$$

证毕.

6 实验评估

基于上述提出算法及场景,本文在 Windows 系统 2.2GHz 英特尔 Core i7-8750H,8GB 内存下采用 Python 3.6 语言进行编程仿真.基于 EUA 数据集[26]中边缘站点的信息来模拟边缘服务器地理分布,每个基站的覆盖范围服从[150,400]内的正态分布[6].参考相关文献的随机生成参数设定[15],每个候选服务的输入和输出数据大小取值在 50 到 150 之间,服务器执行时间在 10 到 30 之间取值,数据传输速率在 1 到 5 之间变化,并且服务器间的传输速率为 1.待机功率 p^{sp} 设置为 0.002,发射功率 p^{tx} 设置为 0.025.根据文献,得到对真实数据 Dartmouth traces[27]的移动模型拟合参数设定.为更好地阐述 CSS 算法的优越性,以下将从三个方面展开,并相应地为三个问题做出解答:

(1)为什么需要考虑用户的移动模式?考虑用户的移动性能够将对优化目标造成多大的影响?

(2)为什么采用边缘计算架构而非云计算模式?

(3)CSS 算法对于各参数设定的鲁棒性如何?

对于组合服务的选择而言,过去采用的决策往往没有考虑用户移动性带来的影响,即用户在生成服务请求时,并不考虑移动模式所增添的决策可能,其只在当前时刻做出最优解决方案.在云计算中,对于一个组合服务,用户只单纯地选择执行时间最短的候选服务将其卸载到云端进行处理.因此,本文拟采用以下四种基准算法进行对比实验.

(1)遗传算法(GA):根据文献,以可行的系统决策作为染色体,采用遗传算法在移动环境下对组合服务进行选择,并基于交叉,变异以及轮盘赌的选择方式更新染色体群[28].

(2)遗传-退火算法(GA-AN):根据文献,以可行的系统决策作为染色体,采用使用退火算法改进的遗传算法在移动环境下对组合服务进行选择[15].不同于的轮盘赌方式[28],对于通过交叉和变异操作生成的新染色体以及旧染色体,以一定的退火概率进行选择并更新.

(3)贪心选择当前最优算法(GSCR):在当前服务计算与边缘计算领域,大多数研究并不将移动性纳入考虑范围,这对制定决策造成了时空局限性.每当用户开始处理新的子任务时,其最优决策为静态地制定当前时刻最短响应时间策略(思想与问题一一致).

(4)贪心选择服务器执行最快算法(GSE):对于每个子任务而言,用户选择执行时间最短的候选服务,即在计算能力最强的服务器(云服务器)上进行(思想与问题二一致).在过去,能够通过网络传输的数据报大小通常是很小的值,所以相对于考虑用户的网络质量改变造成的影响,选择执行时间最短的候选服务往往能够在直观上保证响应时间处在极短的水平.

6.1　基准实验对比

(1)最优性及稳定性评估

为了观察五种算法在长期时间轴上体现出的优劣,对各算法在连续 100 个时间片上的性能进行仿真.如图 3 所示在第 0 个时间片时,初始化网络环境参数,用户在时间片开始阶段随机生成服务请求,并根据四种算法得到相应结果.图中表明,随着时间的进行,在连续 100 个时间片上,MA 算法始终优于其它算法,说明该算法与其他算法相比不仅是最优途径,而且从长期时间轴的角度来看,该算法具备时间稳定性.对于问题一而言,仅选取当前最优的 GSCR 的平均延时比 MA 超出 21.20%,对于问题二而言,采取云计算模式的 GSE 算法所获得的时延要高于 MA 算法 28.88%.在 GA 算法以及 GA-AN 算法中,GA 算法要比 MA 算法收获的时延更高 14.19%,综合了退火算法的 GA-AN 展现出更优的性能,因为其添加了一定的试错机制防止陷入局部最优解,该策略与 MA 算法一致,但是仍然要比 MA 算法高出 7.76%的时延.而云计算 GSE 以及忽略用户移动性的 GSCR 将造成更高的时延.之后,下文将通过实验证明随着任务数量的增加,MA 算法的优势将更加明显,相对 GA-AN 以及 GA 在连续时间轴上将体现出更高的优化率.

同时,观察用户总体电源在各种算法下的变化趋势.从图 4 中可以观察到,随着时间的进行,用户从 EH 组件中获取新的能源包并转换为自身电能,同时在时间片开始阶段处理新生成的服务请求.越长的响应时间将从直观上造成更大的能耗损失.由于获得的能源与能耗相差不大,采用云计算的 GSE 算法始终维持在一个增加趋势很小的状态.在图 3 中,时延的优化效果强弱排序分别为 MA,GA-AN,GA,GSCR 以及 GSE,MA 算法具备最短的响应时间,其映射到电能消耗图中会具备更明显的增长趋势.相对采用云计算架构的 GSE 以及忽略移动性的 GSCR 而言,MA 算法无论是时延还是能耗都能够实现更小的开销.随着设备电量不断增多,根据(26),当电量超出 θ_i 时,即 $\bar{\chi}_i(t) > 0$ 的情况下,设备从 EH 组件中存储的能量将为 0,从而使电池能量维持在 θ_i 附近.从图 4 可看出,MA 算法比其他四种算法更快使系统达到稳定状态.

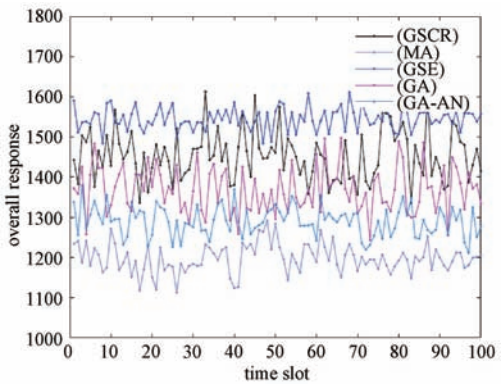

图 3　时间轴上五种算法响应时间对比图

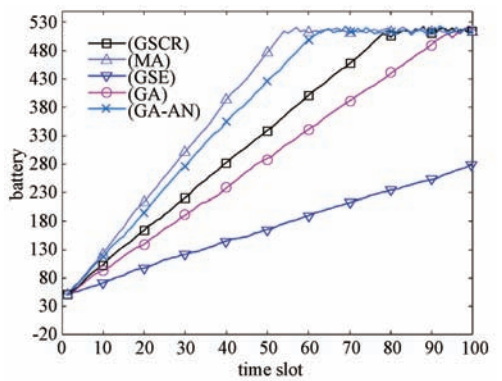

图 4　时间轴上五种算法电池能量对比图

(2)规模性评估

为了观察子任务数量对服务响应时间带来的影响,对上述四种算法在任务数量为 2～6 的范围内进行仿真.如图所示,四种算法计算得到的总体相应时间与任务数量均呈正相关.显然,在其他参数固定的情况下,任务数量的增加意味着用户需要处理更多的候选服务,因此响应时间逐渐增加.从图 5 中可以得到,本文提出的 CSS 框架在任务数量变化的情况下始终优于其余四种算法.在任务数量很小时,系统策略空间偏小,GA,GA-AN 以及 MA 算法的性能

差异不大，均能达到一个优良的收敛值. 而随着任务数量的增加，MA 算法相对于其他算法的优化率不断增加，在初期任务数量仅为 3 时，MA 算法相对于 GA-AN 算法的优化率仅为 3.7%，直到任务数量为 6 时，优化率增加到 16.83%，对于 GSE 算法优化率增加到 55.24%，这恰好表明了 MA 算法在大规模场景下展现出的优越性. 这是因为 GA 以及 GA-AN 算法的系统策略空间呈指数级增长，导致收敛十分困难，而且在任务数量为 6 时，GA 算法的性能已经被 GSCR 算法超越.

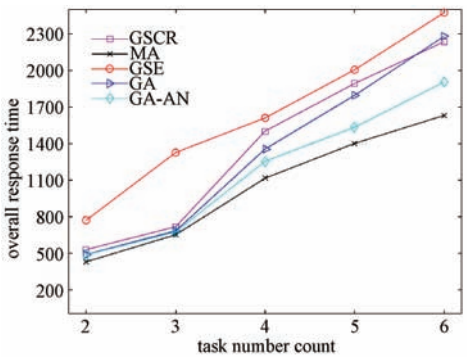

图 5 子任务数量对响应时间的影响

接下来，为了观察候选服务数量对算法带来的影响，在保持其他参数不变的情况下，控制候选服务数量在 1～5 之间变化. 分析图 6 可以发现，随着候选服务数量的改变，MA 算法不仅始终优于其余四种算法，而且展现出超出其他算法 7.33%～32.76%的优化水平. 值得注意的是，在图 5 中已经论证随着场景规模的扩大，MA 算法将展现出相对其他算法更高的性能. 在图 6 中候选服务很少的时候，比如候选服务数量 1～3，所有算法的时延全部稍微地下降，这是因为更多的候选服务意味着更多的选择，在一定阈值内时有利于带给用户更短的时延体验，而超出一定阈值时，过大的策略搜索空间可能导致系统并不能收敛. 此外，MA 算法的总体响应时间的波动最小. 在实验设定中，每个候选服务所对应的执行时间以及输入输出参数大小的生成均服从独立同分布. MA 算法结果对应的标准差仅是其它算法的 6.14%～18.38%. 而在一个随机场景下，随着参量的变化，算法体现出的波动性更弱，正是其鲁棒性更优的论证.

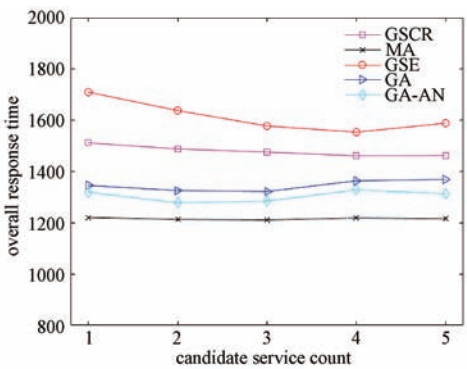

图 6 候选服务数量对响应时间的影响

6.2 内部参数对比

上述实验的设计目的为展现 MA 算法与其它算法的优劣性. 接下来，分析 MA 算法自身的一些控制参数影响.

(1)李雅普诺夫优化过程控制系数 V 的影响

首先，保持其余参量不变，控制 V 在一定范围内变化，从而分析 V 对于算法收敛的影响程度. 从图 7 可以观察到，V 在变化初期所产生的影响最大. 这是因为在马尔科夫近似过程中，V 作为响应时间的权重系数直接影响目标函数值，从而影响跳转概率，进而影响算法收敛结果. 在跳转概率的设计中，V 与跳转概率呈 e^{-V} 负相关，换言之，映射在生成的响应时间上，其应与 V 一条近 e^{-V} 曲线. 如图所示，仿真实验获得的结果与分析相契合，随着 V 的增大，其对收敛结果造成的影响将趋近于 0.

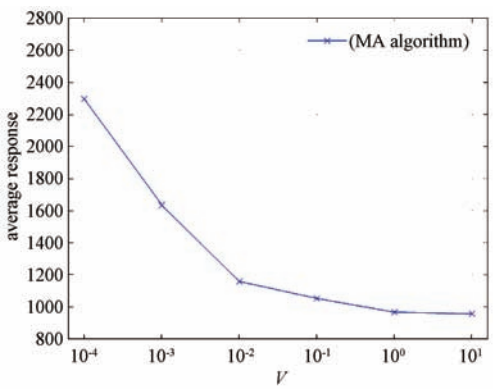

图 7 V 对响应时间的影响

在观察 V 对响应时间产生影响的过程中，记录了在连续时间轴上用户总体电量的变化曲线. 如图 7 所示，随着 V 的增大，用户的能耗变小，在连续时间轴上体现为电量的逐渐提升. 在 $V=1$ 时，电能的增长趋势最明显，斜率最大，这是因为 $V=1$ 的取值越大，系统将以更大的概率跳转到更优的状态. 与前述同理，当 V 很小时，其变化趋势能对收敛性造成更大的影响. 下图 8 表明，当 V 从 10^{-3} 增加到 10^{-2} 时，用户电能的变化差异最明显.

(2)马尔科夫近似过程控制系数 β 的影响

在马尔科夫近似过程中，控制系数 β 直接影响着跳转概率的取值. 当 $\beta \to \infty$ 时，系统将以趋近 1

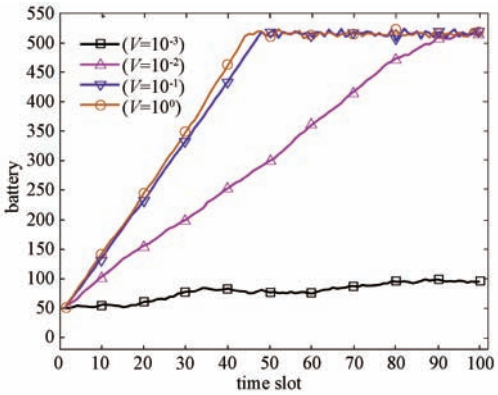

图 8　对电池能耗的影响

的概率往更优的状态跳转.但是在实际中,存在着许多因素限制着 β 的取值.为了观察其对收敛情况的影响,记录系统在不同 β 取值下,每次迭代后生成的结果.如下图 9 所示,系统在 $\beta=1$ 时将获得平滑的收敛曲线,而随着 β 的减小,系统产生的扰动将不断增大.$\beta=0.01$ 时,系统在 500 次迭代时产生收敛趋势,并在某值的一定范围内波动,这是因为试错概率设置过大,并且收敛值在 2200 以上.在 $\beta=0.05$ 时,系统在 1700 次之后体现收敛趋势,波动范围变小,但收敛值扔超出 $\beta=1$ 收敛值的 13.34%.在 $\beta=1$ 时,系统虽然在 2500 次迭代左右收敛,但收敛值仅有 1800.

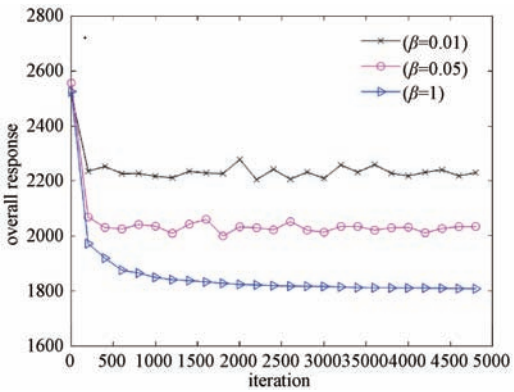

图 9　β 对收敛情况的影响

7　总　结

本文致力于研究应用了移动边缘技术的异构云—边—端服务计算场景中,在考虑用户移动模式以及电池有限容量的情况下,如何对组合服务请求制定最优选择策略这一关键难题.对于本文所做出的具体的研究探索,主要可以从以下三个方面进行总结:(1)构造复杂的大规模异构云—边—端协作网络,从直观上加大了用户制定最优的选择策略的难度;(2)考虑用户的移动性与移动设备的电池可持续性.将用户的移动性建模成交替更新过程,以此模拟用户与基站通信过程;并采用 EH 技术流程建模成生成能量包的随机过程.(3)针对研究问题提出了新的选择框架,将模型表述为多变量联合优化问题,结合李雅普诺夫优化与马尔科夫近似算法,将原问题结构成两类子问题,并构造问题特定的马尔科夫链,使系统在低时间复杂度下收敛得到近似最优解.本文不仅是对当前工作的总结,也是对未来工作的展望.在未来的研究中,将考虑更为严苛的 deadline 约束,以及如何制定边缘服务器服务请求队列调度策略,以此实现系统全局响应优化.

参　考　文　献

[1] Liu L, Huang H, Tan H, et al. Online DAG Scheduling with on-demand function configuration in edge computing//Proceedings of the International Conference on Wireless Algorithms, Systems and Application. Hawaii, USA, 2019: 213-224

[2] Lambert S, Van Heddeghem W, Vereecken W, et al. Worldwide electricity consumption of communication networks. Optics Express, 2012, 20(26):513-24

[3] Ulukus S, Yener A, Erkip E, et al. Energy harvesting wireless communications: A review of recent advances. IEEE Journal on Selected Areas in Communications, 2015, 33(3): 360-381

[4] Sudevalayam S, Kulkarni P. Energy harvesting sensor nodes: Survey and implications. IEEE Communications Surveys and Tutorials, 2011, 13(3): 443-461

[5] Wang R, Zhang J, Song S H, et al. Exploiting mobility in cache-assisted D2D networks: Performance analysis and optimization. IEEE Transactions on Wireless Communications, 2018, 17(8): 5592-5605

[6] Zhao H, Deng S, Zhang C, et al. A mobility-aware cross-edge computation offloading framework for partitionable applications//Proceedings of the 2019 IEEE International Conference on Web Services (ICWS). Milan, Italy, 2019:193-200

[7] Deng S, Huang L, Wu H, et al. Toward mobile service computing: Opportunities and challenges. IEEE Cloud Computing, 2016, 3(4): 32-41

[8] Sardellitti S, Scutari G, Barbarossa S. Joint optimization of radio and computational resources for multicell mobile-edge computing. IEEE Transactions on Signal and Information Processing over Networks, 2015, 1(2):89-103

[9] Zhao Y, Zhou S, Zhao T, et al. Energy-efficient task offloading for multiuser mobile cloud computing//Proceedings of the

International Conference on Communications in China, Shenzhen, China: IEEE, 2015. 789-803

[10] Peng X, Gu J, Tan T H, et al. CrowdService: Optimizing mobile crowdsourcing and service composition. ACM Transactions on Internet Technology, 2018, 18(2):1-25

[11] Wu H, Deng S, Li W, et al. Revenue-driven service provisioning for resource sharing in mobile cloud computing//Proceedings of the International Conference on Service-Oriented Computing. Malaga, Spain, 2017:625-640

[12] Deng S, Huang L, Taheri J, et al. Mobility-aware service composition in mobile communities. IEEE Transactions on Systems Man & Cybernetics, 2017, 47(3):555-568

[13] Zhou W, Fang W, Li Y, et al. Markov approximation for task offloading and computation scaling in mobile edge computing. Mobile Information Systems, 2019, 2019: 8172698: 1-8172698:12

[14] Chen H., Liu M., Wang Y., Fang W., Ding Y. A Markov approximation algorithm for computation offloading and resource scheduling in mobile edge computing. Ning H. eds. Cyberspace Data and Intelligence, and Cyber-Living, Syndrome, and Health. Singapore: Springer, 2019. 1138. https://doi.org/10.1007/978-981-15-1925-3_1

[15] Wu H, Deng S, Li W, et al. Mobility-aware service selection in mobile edge computing systems//Proceedings of the 2019 IEEE International Conference on Web Services (ICWS). Milan, Italy, 2019:201-208

[16] Deng S, Huang L, Hu D, et al. Mobility-enabled service selection for composite services. IEEE Transactions on Services Computing, 2016, 9(3): 394-407

[17] Wang D, Pedreschi D, Song C, et al. Human mobility, social ties, and link prediction//Proceedings of the 17th International Conference on Knowledge Discovery and Data Mining, San Diego, USA, 2011: 1100-1108

[18] Conan V, Leguay J, Friedman T, et al. Characterizing pairwise inter-contact patterns in delay tolerant networks//Proceedings of the 1st International Conference on Autonomic Computing and Communication Systems, Autonomics 2007. Rome, Italy, 2007:19

[19] Meng J, Tan H, Li X, et al. Online deadline-aware task dispatching and scheduling in edge computing. IEEE Transactions on Parallel and Distributed Systems, 2020, 31(6): 1270-1286

[20] Jain R, Shivaprasad A, Lelescu D, et al. Towards a model of user mobility and registration patterns. ACM Sigmobile Mobile Computing & Communications Review, 2004, 8(4):59-62

[21] T. Ouyang, Z. Zhou, and X. Chen. Follow me at the edge: Mobility-aware dynamic service placement for mobile edge computing. IEEE Journal on Selected Areas in Communications, 2018, 36(10):2333-2345

[22] Mao Y, Zhang J, Letaief K B. Dynamic computation offloading for mobile-edge computing with energy harvesting devices. IEEE Journal on Selected Areas in Communications, 2016, 34(12):3590-3605

[23] Chen M, Liew S C, Shao Z, et al. Markov approximation for combinatorial network optimization. IEEE Transactions on Information Theory, 2013, 59(10): 6301-6327

[24] Moon S, Thant Zin O O, Kazmi S M A, et al. SDN-based self-organizing energy efficient downlink/uplink scheduling in heterogeneous cellular networks. IEICE Transactions on Information and Systems, 2017, E100.D(5):939-947

[25] Hajiesmaili M H, Mak L T, Wang Z, et al. Cost-effective low-delay cloud video conferencing//Proceedings of the International Conference on Distributed Computing Systems, 2015, 2015:103-112

[26] Lai P, He Q, Abdelrazek M, et al. Optimal edge user allocation in edge computing with variable sized vector bin packing. CoRR, 2019, abs/1904.05553

[27] Kotz D, Essien K. Analysis of a Campus-Wide Wireless Network. Wireless Networks, 2005, 11(1/2):115-133

[28] Deng S, Wu H, Tan W, et al. Mobile service selection for composition: an energy consumption perspective. IEEE Transactions on Automation ence and Engineering, 2017, 14(99):1-13

[29] Zhang S, Shao Z, Chen M, et al. Optimal distributed P2P streaming under node degree bounds. IEEE/ACM Transactions on Networking, 2013, 22(3): 717－730.

CHEN Hao-Wei, Ph. D. candidate. His research interests include mobile edge computing and service computing.

DENG Shui-Guang, Ph. D., professor. His research interests include mobile edge computing and service computing.

ZHAO Hai-Liang, Ph. D., candidate. His research interests include mobile edge computing and service computing.

YIN Jian-Wei, Ph. D., professor. His research interests include service computing and remote sensing big data.

Background

Mobile edge computing technology provides new ideas in reducing latency and relieving network traffic load, as a promising paradigm. Compared with mobile cloud computing, it offloads computing tasks generated by mobile devices

to edge servers in the vicinity of users to obtain lower data-transmission latency, instead of dispatching them all to remote cloud server. However, there are some restrictions in mobile edge computing scenarios, in practice, such as battery, computing resources and capabilities of edge servers, network transmission bandwidth and so on. The above limitations provide a challenge for users to obtain the best QoS (Quality of Service) due to different selection strategy resulting in different latency. As mobile devices evolved to be capable of processing more diversified mobile services, people tend to exploit smart phones or some other mobile devices, to handle daily business. It's notable that network in mobile environment is volatile and location-based which endows spatial and temporal properties on users' service- oriented strategies, and traditional QoS-aware algorithm is not applicable any more. This paper aims to minimize the overall response time of composite service requests generated by users in a mobile community, while the above constraints considered. We propose a probabilistic pattern to model user's mobility as an alternating renewal process. To be more real-life, this paper adopts energy harvest model based on EH (Energy Harvest) components to simulate how battery changes in successive time slots. We formulate it as a stochastic optimization problem and proposes a new framework named CSS (Composite Service Selection) aiming at minimizing the overall response time of composite service requests generated by users in a mobile community and stabilizing battery energy in a reliable level, with the above constraints considered. First, it transforms the stochastic optimization problem over time slots into normal optimization problem in a given time slot, by introducing Lyapunov optimization. Then this problem could be decoupled into two subproblems as energy harvest problem and service selection problem. The closed-form expression of the energy harvest problem can be derived directly and next we develop a distributed method to solve the service selection problem, based on Markov approximation which has polynomial computation complexity. We take four baseline algorithms as benchmarks which are adapted from previous works. The experimental results demonstrate the superiority of our proposed framework compared with other baseline algorithms based on real-world dataset. It can obtain asymptotic optimality with low complexity and outperform other algorithms from 7.76% to 28.88% in term of latency. Moreover, mobile devices utilizing CSS are the fastest to reach stability. As problem scales, CSS outperforms other algorithms from 3.7% to 55.24%.

This research was partially supported by the National Science Foundation of China (No. U20A20173, No. 61772461) and Natural Science Foundation of Zhejiang Province (No. LR18F020003).