

边缘计算可信协同服务策略建模

乐光学 戴亚盛 杨晓慧 刘建华 游真旭 朱友康

(嘉兴学院数理与信息工程学院 浙江嘉兴 314001)

(ygx_9018@163.com)

Model of Trusted Cooperative Service for Edge Computing

Yue Guangxue, Dai Yasheng, Yang Xiaohui, Liu Jianhua, You Zhenxu, and Zhu Youkang

(School of Mathematics and Information Engineering, Jiaxing University, Jiaxing, Zhejiang 314001)

Abstract With the development and widespread application of the Internet of things and 4G/5G wireless network technology, we have entered into the Internet of everything era. It is easier to connect the edge computing devices, such as mobile phones, PAD, etc., to the Internet. Thus, the number of data generated by edge computing devices is increasing significantly. However, the current network services cannot provide such demand, posed by edge computing, on high throughput, frequently connection, sensitive to location and latency. It is an efficient way to improve the quality of service by 1) considering the characteristics of the intelligence, diversity and flexibility of node at the edge of network, 2) locally aggregating computing, storage and network service resources, and 3) adaptively building trusted cooperative service system. The key to efficiently build a cooperative service system is quickly looking for and then dynamically organizing the edge computing nodes. In this paper, we propose a leader-based trusted cooperative service for edge computing (TCSEC). The main idea is the leader node selects its cooperative service node set with a self-adjustable clustering, which takes into consideration the features of a node, e. g. trust degree, influence, volume, bandwidth, and quality of the link, and realizes the rapid resource aggregation and computing migration. Based on our approach, it is fast to respond to the computing request and provide reliable service. The simulation shows TCSEC can efficiently speed up the ability to construct a cooperative service system and improve the quality of service.

Key words edge computing; trust evaluation; node influence; task offloading; trust cooperation

摘要 随着物联网、4G/5G 无线网络技术的发展和普及应用,万物互联已成为现实.移动计算、智能手机、Pad 和可穿戴等智能终端设备大量接入使网络边缘设备数量迅速增加,边缘设备所产生的数据呈指数增长,边缘计算面临大吞吐量、频繁交互、位置和延迟敏感等特征的实时业务服务需求挑战.充分发挥边缘计算节点智能、多样和灵活等特点,通过局部汇聚计算、存储、网络服务等共享方式实现边缘计算资源快速融合,构建可信协同服务系统是保障边缘计算 QoS 的一种有效方法.为了快速发现、动态组织、自主融合边缘节点进行协同服务,提出一种基于盟主的边缘计算协同服务(trust cooperative services for edge computing, TCSEC)模型.该模型基于信任度、影响力、容量、带宽、链路质量等表征节点特征属性,以任务驱动方式,由盟主节点根据策略和边缘节点的特征属性选择协同服务节点集,实现资源快速融合与计算迁移,为计算请求节点提供及时响应和可靠服务.仿真实验表明:TCSEC 能有效提高边缘计算协同服务能力和服务质量.

收稿日期:2019-02-13;修回日期:2019-11-18

基金项目:国家自然科学基金项目(61572014)

This work was supported by the National Natural Science Foundation of China (61572014).

关键词 边缘计算;信誉评价;节点影响力;任务迁移;可信协同

中图分类号 TP393

随着云计算、大数据、物联网等技术的迅速发展和智能移动设备的普及应用,开启了以边缘计算为基础的万物互联时代,网络边缘设备量及其产生的数据均在快速增长.高清视频、直播、物联网、虚拟现实和增强现实等大流量、大连接等应用已成为通信网络流量的主要业务^[1].文献[2-4]指出到2020年全球数据总量将超过40 ZB,边缘计算将处理其中45%的物联网数据.据思科视觉网络指数(Cisco visual networking index, VNI),从2016年开始更多的流量从蜂窝网络卸载到WiFi,到2021年时全球移动数据流量中仅视频流量将达到78%以上,全网流量中IP视频流量将达到82%,85%的流量为视频、游戏和多媒体数据流,其中内容分发网络(content delivery network, CDN)流量将占据全网流量的71%^[5].据中国互联网络信息中心(China Internet Network Information Center, CNNIC)第42次《中国互联网络发展状况统计报告》,我国网络用户数已达到8.02亿,其中手机网民规模达7.88亿,74.1%的网民使用短视频,仅网络直播用户规模达到4.25亿^[6].由于受数据处理的实时性、带宽、能耗、安全等约束,以集中式处理为核心的云计算模式将不能满足高效处理边缘设备产生的数据需求^[2-4].研究表明以云计算为核心的边缘计算是解决该问题的有效方法之一^[7-8].文献[2]指出由于边缘计算是移动设备位于无线接入网内,通过边缘服务器或移动设备提供信息和云计算服务的一种新的网络结构,具有高带宽、低延迟和安全性强的特点.由于边缘计算将计算任务下推至靠近数据源的位置,甚至迁移至数据源到云计算中心的链路节点上,边缘计算面临三大挑战:

1) 服务发现.由于计算服务请求者具有动态性,相对静态的域名服务协议(domain name system, DNS)服务发现机制不能满足大规模动态边缘计算需求^[9].如何准确发现周边服务是边缘计算面临的关键问题之一.

2) 快速配置.用户和计算设备的动态无约束性,计算设备的随意上下线导致大量服务迁移,引起浪涌式网络流量^[10].如何快速自适应服务快速配置是边缘计算面临的关键问题之二.

3) 负载均衡.边缘服务器提供服务时,其边缘设备同时产生大量数据^[2].如何动态有效地调度这些数据实现高效边缘计算服务是边缘计算面临的关键问题之三.

1 相关研究

万物互联与云计算和大数据技术相融合产生了边缘计算新技术.其核心思想是赋予边缘设备执行计算任务和数据分析处理使命,为实现负载均衡、低延迟和低开销的目的,构建以云计算为中心、边缘计算为辅的轻耦合混合计算模式,将云计算中心的计算任务适当迁移至边缘设备,以解决云计算集中式服务日渐形成的瓶颈问题,提高计算任务数据处理效率,有效保护用户数据安全及隐私.文献[11]指出当边缘计算特别是移动边缘计算在处理大吞吐量数据和数据频繁交互、位置和延迟敏感的实时业务服务请求时明显力不从心,如何进行边缘计算服务资源动态调度,实现协同服务、负载均衡和低延迟开销是边缘计算的关键问题;文献[12]提出一种无线通信资源与边缘计算资源协同调度算法,在移动边缘计算场景下通过优化频谱分配,提高频谱利用率,同时节能效果显著;文献[13]采用博弈论的方法以分布式方式实现高效的卸载计算,并应用于多径无线竞争信道的多用户计算迁移场景;文献[14]提出融合移动边缘计算的5G移动通信网络架构,自适应支持多通信模式,基于虚拟化技术实现资源高效共享;文献[15]以服务重要性指标分级,基于时空约束和虚拟机调度策略,研究边缘计算任务调度方法,保证服务提供的公平性,并利用Lyapunov框架求解最大时间约束的优化问题,均衡多运营商和设备所有者的收益.

文献[16]综述了移动边缘计算的演化和发展,对比分析了基于能耗优化管理的移动边缘计算模型;文献[17]以计算时间和能耗为目标,将计算迁移建模为凸优化问题,提出了一种基于拉格朗日的计算迁移能耗优化策略;文献[18]将任务迁移权重引入协同盟员调度和构建负载均衡适应函数,提出一种在多约束条件下边缘计算可信协同任务迁移策略.

文献[19]针对关联性任务数据引起大量迁移的问题,从日志信息入手,挖掘任务数据的依赖关系,通过优化数据存储、迁移调度尽可能减少了系统开销;文献[20]针对边缘设备与基站关联及基站睡眠优化问题,提出一种基于超密集网络的移动边缘计算框架(computing offloading framework based on

mobile edge computing and ultra dense network, COMED), 基于联合计算卸载、基站睡眠和用户-基站关联 (joint computing offloading base station sleeping and user-base-station association, JOSA) 的任务调度方法实现能耗优化; 文献[21]研究移动边缘编排优化协同缓存处理, 在多场景下应用证明了该方法能够有效地将移动边缘计算应用于 5G 网络; 文献[22]针对移动边缘计算中高频内容刷新导致能耗高、计算服务效率低的问题, 提出基于 Markov 的能耗优化模型, 提高内容刷新效率以降低计算能耗; 文献[23]根据用户请求的地理分布, 用整数线性规划对时变的业务需求进行建模, 优化分布式缓存部署策略; 文献[24]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, 在资源一定的条件下提高了系统性能; 文献[25]提出了一种移动边缘计算最优资源分配策略, 将资源分配转化为一个凸优化问题求解, 以计算延迟为约束实现能耗最小化, 通过设置任务迁移阈值实现多用户任务迁移资源优化分配, 有效降低系统整体能耗; 文献[26]提出一种基于近邻传播的迁移聚类 (transfer affinity propagation, TAP) 算法, 基于源域与目标域的相似性, 通过近邻特征学习改进消息传递机制, 实现在特征数据稀疏时具有较好的迁移性能; 文献[27-29]分别研究了 Web 服务 CDN 网络、蜂窝移动网络、流视频网络下, 通过云端卸载至边缘, 优化用户体验的问题。

文献[30]针对集群服务器中负载均衡问题, 提出基于集群架构的边缘流媒体服务器, 构建了缓存替换算法 (multilevel cache and load balance supported, MCLBS), 有效提高缓存命中率和降低服务带宽消耗; 文献[31]提出了最小化终端的传输延迟和最大化其满意度的分布式算法, 优化缓存文件分配, 提高分发性能; 文献[32]针对分布式并行处理架构中数据分布不均、数据流实时动态变化且不可预知等导致负载不均衡的问题, 提出通过 Key 粒度迁移和元组粒度拆分进行轻量级均衡调整以保证系统负载均衡。

文献[33]指出恶意用户、误操作、漏洞、不安全资源等是弹性移动云计算面临的主要安全威胁, 针对该问题提出了相应的防御措施; 文献[11]从边缘计算隐私与安全保障角度出发, 聚焦信任度构建、资源调度和协同问题, 构建综合信任评估体系, 保障边缘计算资源管理与协同优化, 构建多层自适应统一计算模型, 实现对应用场景动态匹配; 文献[34]基于用户群体感知机制, 利用布隆过滤和二元内积运算

估计用户间相似度, 形成匹配用户联盟保护个人隐私; 文献[35-38]分别研究了在 Web 服务网络、P2P 网络、机会网络和无线 Mesh 网络场景下的节点信誉度的建模方法, 构建基于信誉评价的节点协同机制, 通过信誉奖惩监测和管理节点行为, 有效保障了协同服务的安全性。

本文基于“负载均衡”理论, 融入节点特征属性构建边缘计算协同服务池, 为边缘计算中服务快速发现和计算迁移提供一个解决方案. 当边缘服务器 (本文称为盟主) 负载达到设定阈值时, 启动网络资源聚合机制, 将邻近节点扩展为协同服务节点, 构建一个动态虚拟边缘计算协同服务池, 实现计算迁移, 服务完成后系统释放资源, 自行解散. 这样不仅可解决拥塞、延迟和抖动的问题, 而且可大幅提高边缘计算服务效率和服务质量, 其中协同服务盟员节点的选取策略是关键。

2 边缘计算协同服务盟员节点组织策略

2.1 协同服务盟员节点选择策略

为保障边缘计算服务效用, 构建自适应协同服务系统是一种有效方法, 其核心是当边缘计算服务负载达到设定的阈值时, 根据自愿、公平的原则和激励策略, 启动资源聚合机制, 将边缘资源有机整合, 动态自适应构建可信协同服务系统^[39-40]. 为用户提供安全可信、灵活鲁棒的边缘计算服务. 协同盟员节点具有 2 个主要特点^[41-45]:

1) 自主性. 盟员节点对其提供的共享资源和协同服务具有完全控制决定权。

2) 任务驱动性. 基于任务迁移需求, 盟主根据策略选取满足需求的节点为盟员, 确定边界和约束条件, 自主融合构建边缘计算协同服务系统. 协同任务完成, 释放资源, 解除协同关系。

分析发现任务驱动性与自主性实际上是互斥的. 完全自主性, 有可能导致协同服务质量和效率不高的问题, 甚至无法实现协同融合。

在实际构建边缘协同服务时, 一般通过折中策略兼顾其目标驱动性和自主性特征, 使构建的协同服务系统达到预期的效率. 启动协同服务时, 盟主节点在保持盟员节点自主性的基础上, 根据任务需求启动网络资源聚合机制动态选取协同盟员节点, 定义约束条件、控制策略, 确定边界、盟主与协同服务盟员间的主从关系, 盟主节点对协同服务系统实施管理、监控和动态调整, 保障协同服务系统的正常运行^[46]。

设边缘计算协同服务系统有 n 个节点和 m 条连接边,其连接矩阵 $\mathbf{E}=(e_{ij}), e_{ij}=1$ 表示节点 i 和节点 j 相连,连接关系强度矩阵 $\mathbf{A}=(\rho_{ij}), \rho_{ij}$ 表示节点 i 和节点 j 的连接关系强度:

$$\rho_{ij} = \begin{cases} e_{ij} \frac{1}{\sqrt{Rtt(i,j)}}, & i \neq j, \\ 0, & i = j; \end{cases} \quad (1)$$

$$e_{ij} = \begin{cases} 1, & \text{节点 } i, j \text{ 相连,} \\ 0, & \text{节点 } i, j \text{ 不相连;} \end{cases}$$

其中, $Rtt(i,j)$ 为节点 i 到节点 j 的网络延迟.

设观察周期为 T ,在第 k 个观察期节点 i 向节点 j 提供协同服务成功和失败的次数分别为 S_{ij}, F_{ij} ;构造节点 j 对节点 i 在第 k 个观察期的局部信任度函数 $Trust(i)$:

$$Trust(i) = (1-\mu)^{1/(S_{ij}+\epsilon F_{ij})} S_{ij} / (S_{ij} + \epsilon F_{ij}), \quad (2)$$

$$\epsilon = \epsilon + F_{ij} / (S_{ij} + \epsilon F_{ij}),$$

$$\text{s.t. } \epsilon|_{t=0} = 0, 0 < \mu < 1, Trust(i)|_{S_{ij}+\epsilon F_{ij}=0} = 0,$$

其中, ϵ 为节点不诚信任务迁移处罚因子; μ 为局部信任度控制因子.

根据节点的协同交互操作,构建节点 i 对节点 j 的吸引力函数 $F(i)$:

$$F(i) = \rho_{ij} f(i) Q(i) Trust(i) e^{d_{\min,ij}+1}, \quad (3)$$

$$f(i) = h_i m_i \omega_i e^{\frac{1}{t_i}} \left(\sum_{k=1}^n h_k m_k \omega_k e^{\frac{1}{t_k}} \right),$$

$$Q(i) = (1-\beta_i^{(t)})(m_i - m_i^{(t)}) \times (\omega_i - \omega_i^{(t)}) / (m_i \omega_i),$$

其中, $f(i)$ 表示节点 i 的服务能力; $Q(i)$ 表示节点 i 的负载率; $d_{\min,ij}$ 表示节点 i 路由到节点 j 的最短跳数; h_i, ω_i, m_i 分别表示节点 i 可提供的计算能力、带宽、内存大小; t_i 表示节点 i 的平均在线时长; $e^{-d_{\min,ij}+1}$ 表示吸引力传播具有级联衰减性; $\beta_i^{(t)}, m_i^{(t)}, \omega_i^{(t)}$ 分别表示第 t 个观察周期内节点 i 已提供的计算负载、内存大小、带宽.

构建盟主节点影响力 $G(i)$ 模型:

$$G(i) = \frac{\alpha_i}{\sum_{k=1}^n \rho_{ik}} \sum_{j=1, i \neq j}^{n-1} F(i) \times jaccard(i, j), \quad (4)$$

$$\alpha_i = \cos^2 \left(\pi / \left(2 \sqrt{\sum_{k=1}^n \rho_{ik}} \right) \right),$$

$$jaccard(i, j) = \frac{\sum_{k=1}^n \rho_{ik} \rho_{jk}}{\sum_{k=1}^n \rho_{ik} + \sum_{k=1}^n \rho_{jk} - \sum_{k=1}^n \rho_{ik} \rho_{jk}},$$

其中, α_i 为离散化因子.

算法 1. 协同服务集构建算法.

Step1. 设初始边缘计算网络节点均为协同服务盟员,盟主对候选盟员分辨阈值系数 $\lambda, \lambda_{\min} \leq \lambda \leq \lambda_{\max}$, 令 $\lambda = \lambda_{\max}, 0 < \lambda \leq 1$.

Step2.

① 如果 $G_i \geq \lambda (i \neq j, i, j \leq n)$, 标记节点 i 和节点 j 彼此为盟友, 将节点 j 划入节点 i 的备选盟员集中.

② 否则, 转 Step3.

Step3. $\lambda = \lambda - s, s$ 为系数.

① 转 Step2, 将边缘计算节点划分为 m 个不相互的协同服务盟员集.

② 如果某个节点 j 不属于任何协同服务盟员集, 则加入相邻节点中影响力 G_i 最大的节点 i 所在的协同服务盟员集中.

③ 调整. 在划分的协同服务盟员集中, 将影响力小的节点加入到相邻节点对其影响力 G_i 最大的节点 i 协同服务盟员集中.

Step4.

① 边缘计算网络节点集无划分操作, 转 Step3.

② 边缘计算网络节点集被划分为 2 个以上协同盟员服务子集, 则停止分解, 令 $\lambda = \lambda_{\min}$.

③ 将划分的协同盟员服务子集视为 1 个新边缘计算网络, 令 $\lambda = \lambda_{\max}$, 转 Step2.

④ 直到所有的协同盟员服务子集不能被再划分为止.

Step5. 构建协同盟员服务子集间的关系映射图, 标识各协同盟员服务子集分辨系数阈值 λ 的界.

2.2 协同服务机制分析与建模研究

边缘计算协同服务的工作原理为: 当盟主节点发起协同任务时, 依据任务需求定义协同服务内容、协同服务目标、控制策略、协同边界和约束条件等参量, 遴选适宜、有效的协同服务盟员节点, 定义协同服务模式和协议, 建立其特征、顺序、从属等相互间的映射关系, 启动网络资源聚合机制, 快速完成盟员节点间的聚合, 构建边缘计算协同服务系统.

利用状态机描述边缘计算协同服务模型 TCSEC 为 D_{TCSEC} :

$$D_{TCSEC} = (Q, \Gamma, \delta, q_0, F). \quad (5)$$

其中:

1) Q 是有穷状态集, $Q = \{\text{边缘计算协同服务的状态集合}\}$;

2) Γ 是状态转移条件产生式;

3) δ 是状态转移函数, $Q \times \Gamma \rightarrow Q$;

4) $q_0 \in Q$, 是起始状态, 由盟主产生;

5) $F \subseteq Q$, 是结束状态集, $F = \{\text{结束}\}$.

D_{TCSEC} 状态转换如图 1 所示:

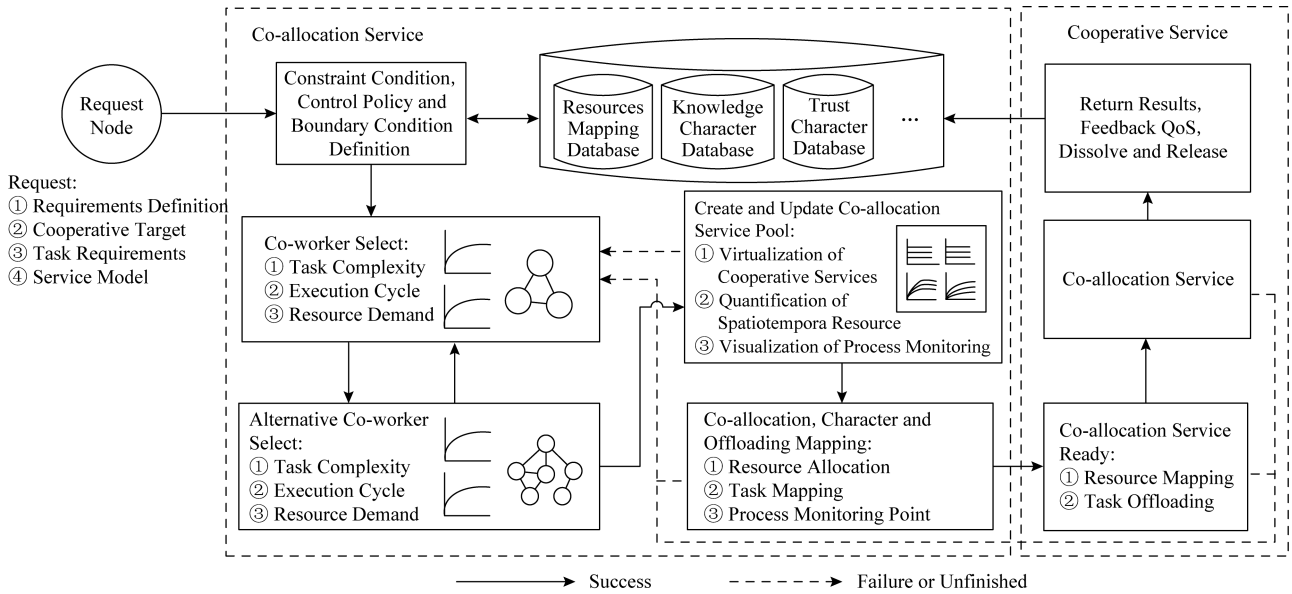


Fig. 1 Trust cooperative service model for edge computing

图 1 边缘计算可信协同服务模型

协同服务的业务事件状态以 T_i 表示, 利用状态机描述其业务事件处理机制 CSE (cooperation service event) 为 D_{CSE} :

$$D_{CSE} = (Q, \Gamma, \delta, q_0, F). \quad (6)$$

其中:

1) Q 是有穷状态集, $Q = \{T_0, T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$;

2) Γ 是状态转移条件产生式;

3) δ 是状态转移函数, $Q \times \Gamma \rightarrow Q$;

4) $q_0 \in Q$, 是起始状态, 由盟主产生;

5) $F \subseteq Q$, 是结束状态集, $F = \{T_7\}$.

一次协同操作过程:

Step1. 发布协同服务需求, 选取满足需求的候选协同服务节点.

Step2. 确定协同服务目标、边界、约束条件和策略.

Step3. 定义协同服务协议和建立角色映射关系.

Step4. 启动聚合机制构建协同服务系统.

Step5. 返回协同服务结果, 更新特征属性数据库. 协同服务的业务事件处理机制^[36]如图 2 所示.

由于目标计算任务分布不均, 引起边缘计算协同服务请求分布不均, 边缘计算协同服务节点可能会出现服务瓶颈, 导致协同服务质量下降, 若计算任务迁移和边缘节点调度策略不完善、节点抖动等引起边缘计算服务不稳定甚至终止服务, 则进一步加

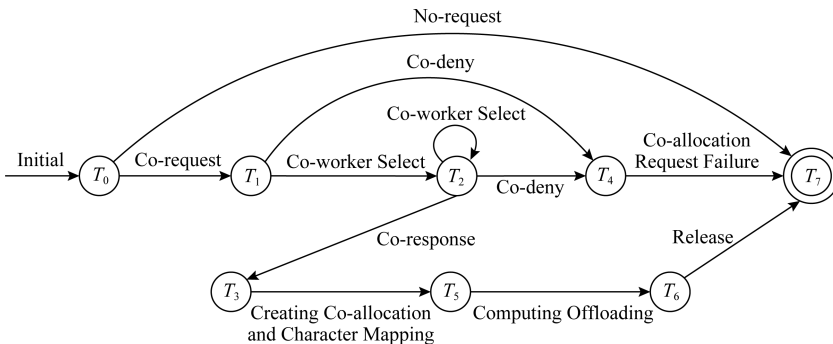


Fig. 2 State transition model for cooperative service event

图 2 一次协同服务状态转移模型

剧计算任务分布的不合理,导致网络的抖动,造成协同服务性能下降。

当节点状态和性质发生改变,无法满足协同服务需求时,通过服务迁移动态调整节点承载的计算任务,可有效加强系统服务性能。在构建的边缘计算协同服务中,当边缘计算协同服务需要执行任务迁移时,根据相关需求及约束策略启动任务迁移机制,将计算任务迁移至目标盟员,协同完成计算任务,这一过程对用户而言是透明无隙的。

协同服务迁移的业务事件状态以 T_i 表示,利

用状态机描述其协同服务任务迁移状态为

$$D_{\text{task-offloading}} = (Q, \Gamma, \delta, q_0, F), \quad (7)$$

其中:

- 1) Q 是有穷状态集, $Q = \{\text{边缘计算任务迁移状态集合}\}$;
- 2) Γ 是状态转移条件产生式;
- 3) δ 是状态转移函数, $Q \times \Gamma \rightarrow Q$;
- 4) $q_0 \in Q$, 是起始状态, 由盟主产生;
- 5) $F \subseteq Q$, 是结束状态集, $F = \{T_5\}$ 。

边缘计算协同服务任务迁移过程如图 3 所示:

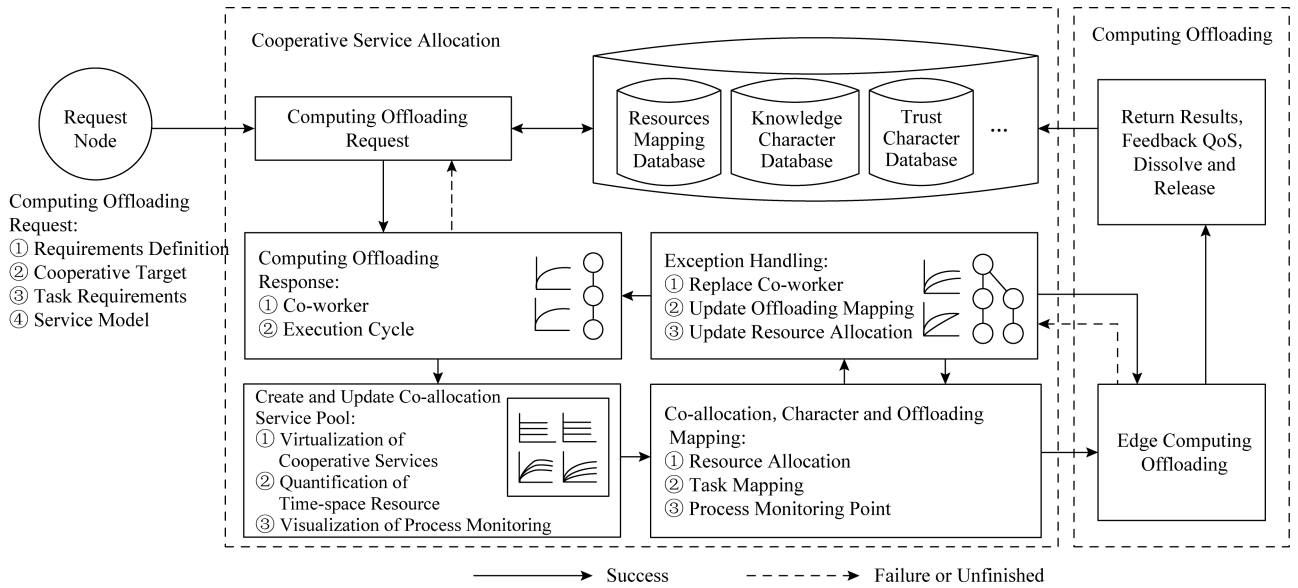


Fig. 3 State transition model for offloading task

图 3 任务迁移状态转移模型

3 边缘计算协同服务系统建模与性能分析

构建边缘计算协同服务需遵循 6 条规则^[47-51]:

规则 1. 构建的边缘计算协同服务系统是完备的,当且仅当协同服务盟员节点集不为空。

规则 2. 以协同盟员节点间特征属性为约束条件,按影响力择优原则在候选协同服务盟员节点集中选择协同服务盟员节点。

规则 3. 定义盟主节点、协同盟员节点间的逻辑约束、主从关系和职责。

规则 4. 盟主节点负责对边缘计算协同服务中的协同资源、状态进行监管和调度,定义协同服务盟员节点间的协同服务协议和约束条件,依据策略动态调整协同盟员角色,过滤性能差的协同盟员节点。

规则 5. 协同服务任务完成后,盟主节点对该盟员节点的协同服务质量和信任度做出公正的评价,更新特征属性数据库;释放资源,解除盟主节点和协同盟员节点间的协同服务关系。

根据策略规则,构建基于边缘计算可信协同服务系统模型 TCSEC 如图 4 所示,其中,②为聚合、判断、状态转移机等,①为策略、映射机制。

TCSEC 共享流服务系统状态表征:

$$M_{\text{TCSEC-streaming-service}} = (Q, \Gamma, \delta, q_{\text{start}}, q_{\text{accept}}), \quad (8)$$

其中:

- 1) Q 是有穷状态集, $Q = \{\text{加入,就绪,在线,协同,离线}\}$,其中,在线状态 $\in \{\text{边缘计算协同服务,计算任务迁移}\}$;
- 2) Γ 是状态转移条件产生式;
- 3) δ 是转移函数, $\delta = (Q \times \Gamma) \rightarrow f$;

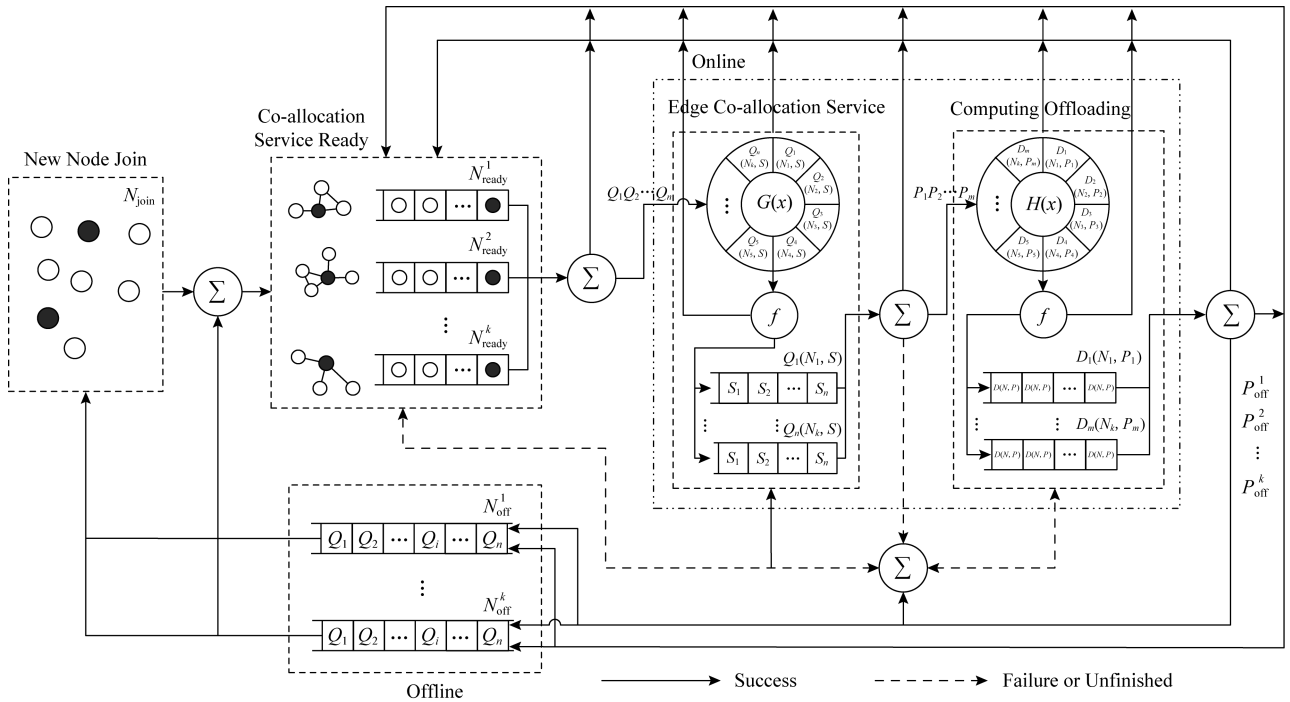


Fig. 4 System model of TCSEC

图4 边缘计算协同服务执行模型

- 4) q_{start} 是起始状态, $q_{start} = \{\text{加入状态}\}, q_{start} \in Q$;
- 5) q_{accept} 是接受状态, $q_{accept} \subseteq Q$.

在线边缘计算协同服务系统为服务请求节点提供服务策略:

- 1) 当节点获得服务后, 节点以概率 P 选择离开边缘计算协同服务系统, 进入离线阶段, 或以 $1-P$ 的概率选择留在 TCSEC, 进入就绪阶段.
- 2) 因异常原因, 节点直接进入离线阶段. 如果节点在系统允许的会话时间段 τ 内恢复正常则节点直接进入异常断点, 继续相应服务操作.
- 3) 当服务请求端异常或中断服务请求时, TCSEC 提供端撤销提供的任务迁移服务队列, 节点进入就绪状态.

协同服务质量评价策略. 在时间窗 τ 系统接收用户对服务质量的评价:

- 1) 接收用户对服务质量的正常评价.
- 2) 拒绝同一用户在时间周期 τ 内对同一服务质量进行多次评价, 并标记该用户, 系统以 $1-Trust(i)$ 的概率对用户信任做负面评价.
- 3) 以 $Trust(i)$ 的概率接收同一用户对协同服务质量评价, 并标记该用户.

由上可知, 系统具有很强的抗攻击和抑制协同作弊的能力.

以近海港口为例构建一个边缘计算服务系统, 如图 5 所示.

用户节点以轻量级泛洪在网络中搜索目标流数据, 按就近服务的策略返回搜索结果, 用户节点依据搜索结果与合适的邻居节点建立连接, 邻居节点将自己缓存的计算任务转发给用户节点, 用户端对获得的流数据进行可用性检测, TCSEC 系统通过联合协同方式为用户节点提供流数据实现流式服务, 用户节点同时接受来自网络其他节点的流服务请求, 并将自己缓存的流数据共享给相应的邻居节点. 其中, \otimes 为聚合、判断、状态转移机等.

TCSEC 服务系统中节点获取服务状态表征:

$$M^{TCSEC-request} = (Q, \Gamma, \delta, q_{start}, q_{accept}). \quad (9)$$

其中:

- 1) Q 是有穷状态集, $Q = \{\text{服务请求, 数据包有效性检测, 数据包聚集与分配, 进入缓存区, 编码与调度, 播放超时重传请求, 内容转发, 服务质量评价等}\}$;
- 2) Γ 是状态转移条件产生式;
- 3) δ 是转移函数, $\delta = (Q \times \Gamma) \rightarrow f$;
- 4) q_{start} 是起始状态, $q_{start} = \{\text{服务请求}\}, q_{start} \in Q$;
- 5) q_{accept} 是接受状态, $q_{accept} = \{\text{计算任务}\}, q_{accept} \subseteq Q$.

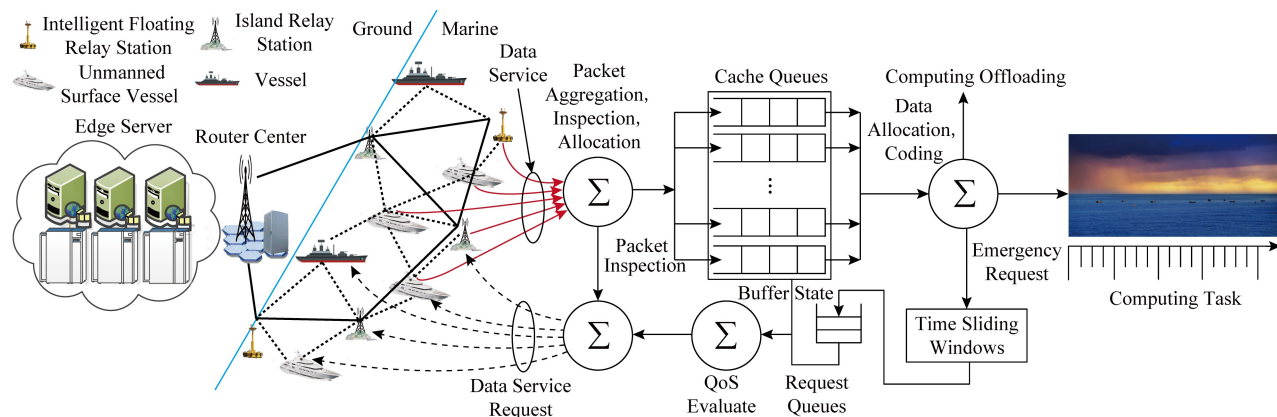


Fig. 5 State transition model of TCSEC offloading task stream

图 5 边缘计算协同任务迁移流数据状态转移模型

由式(9)可得,节点进入 TCSEC 系统获得服务的过程描述为:

Step1. 以轻量级泛洪方式搜索目标计算任务数据.

Step2. 系统返回搜索结果列表,通过服务协商,用户节点选择合适邻居节点建立连接关系,构成协同服务联盟.

Step3. 系统联合协同节点为用户节点提供自己缓存的任务数据.

Step4. 用户节点实施数据包汇聚、可用性检测和分配到相应的缓冲区队列.重新请求不可用的流数据,发表缓冲区状态.

Step5. 数据编码和调度,实施计算服务.如果出现任务缺失、计算异常等,则发起紧急调度请求,等待任务流数据.

Step6. 服务质量评价,请求流数据;接受邻居节点服务请求,转发本地缓冲的流数据.

4 实验与分析

由分析可知:边缘计算协同服务盟主以节点的信任度、影响力、可用带宽、链路质量等特征指标作为选择盟员的约束条件,以最小开销构建边缘计算协同服务池.对于小规模边缘计算协同服务,其构建开销可以忽略;一旦边缘计算协同服务节点达到一定规模,其开销必增,占用网络资源,降低服务质量.因此,在构建协同服务池生成规则中设定规模上下界,当其规模不满足设定域时,启动边缘计算协同服务簇分裂或合并机制,形成新的协同服务池.

近海港口构建的通信网络是一个典型的边缘计算服务应用场景,本文仿真实验场景如图 5 所示.

实验中假设 3 个条件成立:

1) 每个服务节点周围可能存在一个小规模的亚可信边缘计算协同服务节点集,其信任初始值均设为 0.5,这些亚信任节点曾经与盟主服务节点构成协同服务,以形成一定的信任协同关系,这种原始的亚可信边缘计算协同节点其信誉度随服务节点在网络中存活的时间、协同服务质量等而动态变化.

2) 盟主节点按相关策略从其候选协同服务盟员集中选取,以构建边缘计算协同服务.

3) 通信网络以岸基数据中心、岸基站为中心,由岛礁中继站、智能浮台中继站、船舰节点、智能移动节点等构成.

网络仿真环境:

1) 基于 Router View 公开数据集,构建边缘计算系统服务簇;Router View 公开数据集节点数为 6 474,连接边数为 13 895,平均度为 4.292 6,基尼系数为 0.608,相对边缘分布熵为 0.854,聚类系数为 0.959%;

2) 岸基站为根节点,与 7 个中继基站组建骨干网络,1 个岸基网络数据中心服务器为源服务发起节点,节点间自由交互;

3) 以每个基站为初始簇头,构建可信协同服务集.当边缘计算协同服务簇节点规模小于设定的下界时进行合并操作,原盟主节点降为盟员节点;当簇节点规模大于上界时,按策略选择盟员成为新的盟主节点,进行分裂操作;

4) 边缘计算可信协同服务系统由岸基数据中心服务器集群(edge server, ES)、岸基站(router center, RC)、中继基站(router node, RN)、盟主节点(leader node, LN)、盟员节点(edge node, EN)等组成;

5) 节点通信制式符合 IEEE 802.11 协议簇标准.

系统仿真环境:

服务器为 Intel Xeon E5, 4 核 16 线程, 2.5 GHz, RAM 为 4 GB, WinServer2010, 虚拟机带宽 1 MBps;

客户机为 Intel Core i5, 2 核 4 线程, 2.7 GHz, RAM 为 8 GB, Win7 系统;

服务端为 JDK1.8+Apache Tomcat 8.0, Mysql 5.7;

客户端为 Chrome 浏览器;

编程语言为 Java, JavaScript;

开发工具为 Eclipse Oxygen 3a;

后端框架为 JSP, JavaBean, JDBC;

前端可视为 SVG 绘图库, Ajax 异步前后端交互.

4.1 TCSEC 骨干网络系统性能测试分析

为保证骨干网络的鲁棒性, 对骨干网络实施负载压力测试. 测试时间为 120 min, 观察周期为 30 min. 设定岸基数据中心服务器集群的负载率达到 60% 启动边缘协同服务负载均衡机制; 在 30 min 时对岸基数据中心服务器集群实施突发浪涌式服务请求, 60 min 时服务请求达到其负载率 60%, 系统启动边缘协同服务负载均衡机制, 按策略将相应计算任务卸载至其协同盟员边缘服务盟主节点执行,

系统基本稳定在其负载率的 63%. 如图 6 所示. 实验表明, TCSEC 模型的骨干网络具有很好地鲁棒性和负载均衡能力.

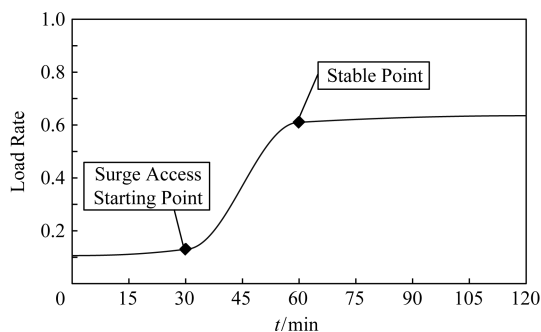


Fig. 6 Load rate of edge server

图 6 数据源服务器负载率

4.2 TCSEC 模型系统聚类性能测试分析

为分析边缘计算协同服务性能, 以聚类效率、服务流量、响应延迟、服务效率等为评价指标. 从静态和动态 2 种网络拓扑状态将本文提出的协同服务池算法 TCSEC 与 K -means^[52], KNN ^[53] 对比分析.

4.2.1 静态协同服务

协同服务池一旦构建完成并稳定后, 不再进行

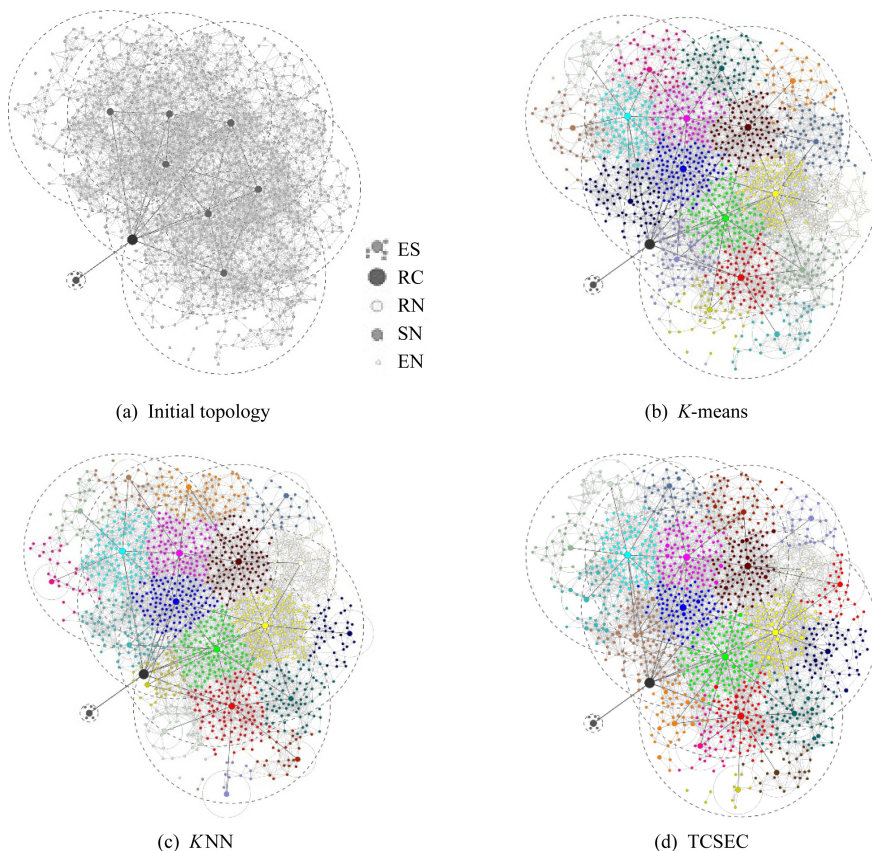


Fig. 7 Topology of static edge computing cooperative service

图 7 静态边缘计算协同服务拓扑结构

合并和分裂操作.其中节点颜色表示节点参与构建边缘计算协同服务系统的意向,节点大小表示节点的服务能力,其边缘计算协同服务拓扑如图 7 所示,聚类算法评价参数如表 1 所示,边缘计算协同服务

池性能评价参数如表 2 所示,协同服务簇、中继基站、盟主节点平均流量如图 8~10 所示,协同服务带宽如图 11 所示,任务接受率如图 12 所示,协同服务成功率如图 13 所示,协同服务响应延迟如图 14 所示.

Table 1 Performance Parameters of Static Edge Computing Cooperative Service Clustering Algorithm

表 1 静态边缘计算协同服务聚类算法性能参数

Algorithms	Number of Clusters	Number of RN Clusters	Size of RN Cluster	Std of RN Cluster Size	Candidate Cooperative Nodes in RN Cluster	Number of LN Clusters	Size of LN Clusters	Std of LN Cluster Size	Candidate Cooperative Nodes in LN Cluster	Number of Free Node	Iteration Times	RunTime /s	Max Hops	Degree of Aggregation
TCSEC	22	7	98.43	13.49	4.0	15	47.87	22.42	1.8	3	19	10.37	6	0.83
K-means	20	7	95.71	18.45		13	56.69	20.62		4	16	7.84	8	0.57
KNN	20	7	115.57	11.46		13	46.00	24.15		4	16	3.29	11	0.44

Note: Std means standard deviation.

Table 2 Performance Parameters of Static Edge Computing Cooperative Service Pool

表 2 静态边缘计算协同服务池性能参数

Algorithms	RTT in Edge Computing Server/ms	RTT in Cluster /ms	Stable Time /min	Number of Concurrency	Traffic of Cluster /(MB/min)	Traffic of RN /(MB/min)	Bandwidth of RN Cluster /Mbps	Traffic of LN /(MB/min)	Bandwidth of LN Cluster /Mbps	Free Resource Utilization
TCSEC	19	7	60	53	20 973.83	3 696.86	7.84	2 250.51	4.25	0.25
K-means	30	39	60	41	9 770.61	2 942.42	10.57	1 300.34	5.69	0.18
KNN	43	41	60	39	10 254.34	3 569.18	11.56	1 781.80	4.63	0.19

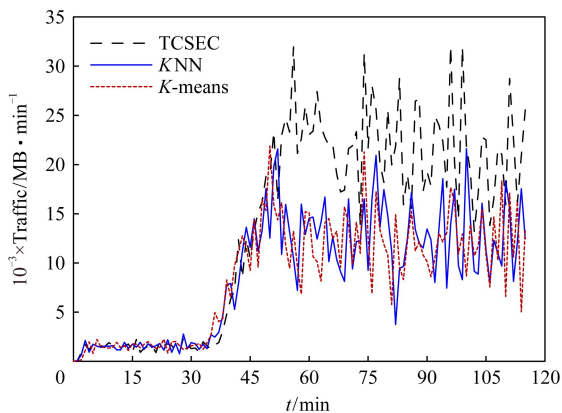


Fig. 8 Cooperative service cluster traffic for static edge computing

图 8 静态边缘计算协同服务簇平均流量

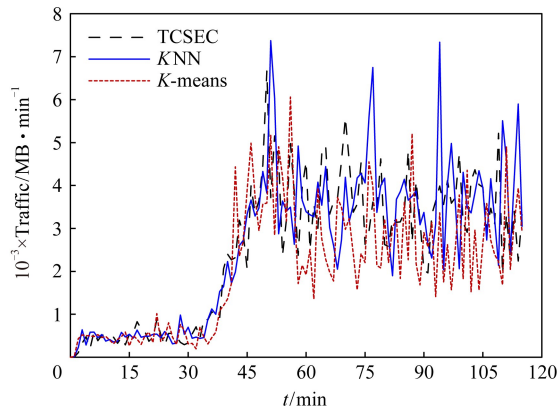


Fig. 9 Relay node traffic for static edge computing

图 9 静态边缘计算协同服务中继基站流量

静态边缘计算协同服务具有 5 个特征:

1) 超级节点聚类具有相似趋同性和马太效应,能力和性能相当的节点会聚在一起;为了保证系统服务提供的服务能力基本一致,能力相对弱的热心节点所构建的协同服务簇的簇内节点数相对较多.

2) KNN, K-means 算法形成 13 个超级节点,构成 20 个边缘计算服务池,分别耗时 3.29 s 和

7.84 s; TCSEC 算法形成 15 个超级节点,构成 22 个边缘计算服务池,耗时 10.37 s; 所构建边缘计算服务池平均规模在 46.00~56.69 之间,网络平均延迟分别为 43 ms, 30 ms, 19 ms, 池内交互平均延迟 41 ms, 39 ms, 7 ms.

3) KNN, K-means 算法构建边缘计算服务池时分裂 13 次,没有合并.聚合度分别为 0.44 和 0.57,协同效率分别 0.45 和 0.58.

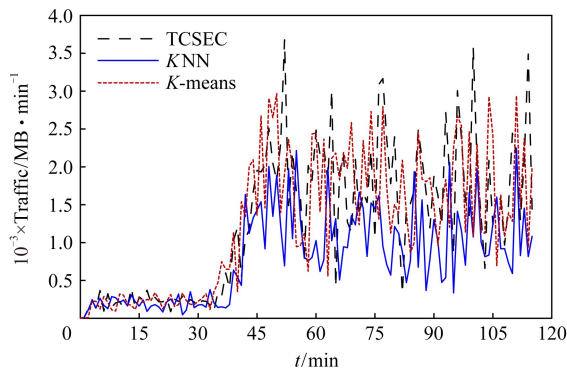


Fig. 10 Leader node traffic for static edge computing

图 10 静态边缘计算协同服务盟主节点流量

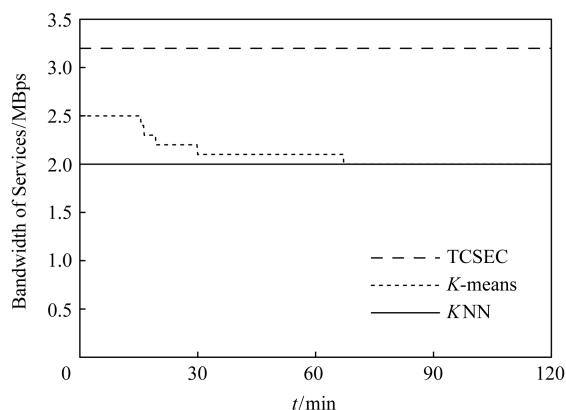


Fig. 11 Bandwidth of static edge computing service

图 11 静态边缘计算协同服务带宽

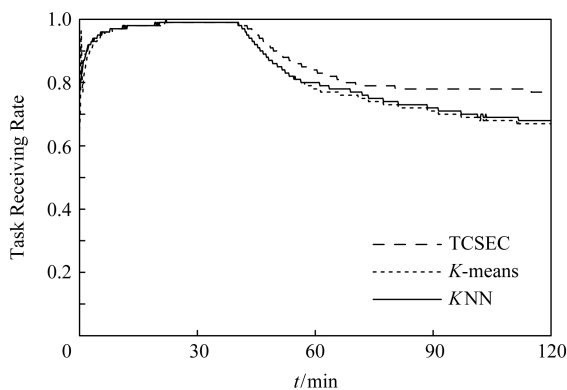


Fig. 12 Task receiving rate of static edge computing service

图 12 静态边缘计算协同服务任务接收率

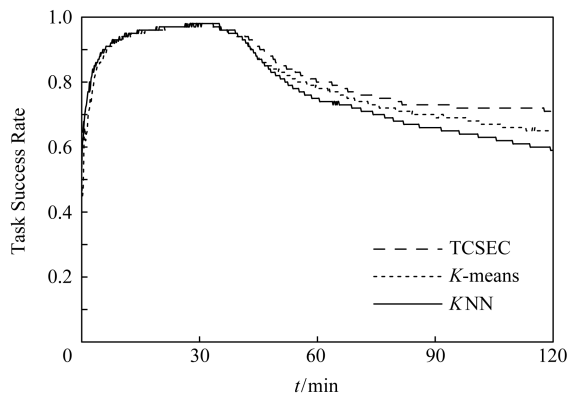


Fig. 13 Task success rate of static edge computing service

图 13 静态边缘计算协同服务成功率

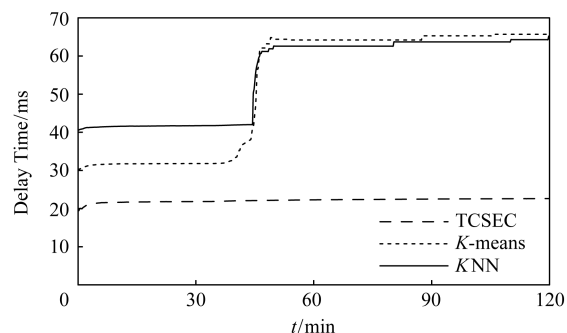


Fig. 14 Response delay of static edge computing service

图 14 静态边缘计算协同服务响应延迟

实验结果表明:

1) 盟主节点均由热心边缘服务器节点形成, 协同服务池构建具有马太效应。

2) TCSEC 算法构建的边缘计算协同服务池能有效均衡基站和盟主节点的负载, 能有效平滑浪涌效应。

3) TCSEC 算法构建的边缘计算协同服务池网络平均延迟分别比 KNN, K-means 算法减少 55.81% 和 36.67%; 边缘计算协同服务池内平均延迟 TCSEC 算法分别比 KNN, K-means 算法减少 82.93% 和 82.05%。

4) TCSEC 算法的协同效率分别比 KNN, K-means 算法的提高 77.78% 和 37.93%。

5) TCSEC 算法基站、盟主节点服务带宽分别比 KNN, K-means 算法提高 3%, 21.30% 和 32.25%, 45.50%, 系统平均并发数提高 35.90% 和 29.27%, 平均服务流量提高 1.15 倍和 1.05 倍。

4.2.2 动态构建协同服务

协同服务池构建完成后, 系统根据虚拟服务池的协同工作状态进行动态合并和分裂操作. 其边缘

4) TCSEC 算法构建边缘计算服务池时执行 20 次分裂, 合并 5 次, 聚合度 0.83, 协同效率 0.80。

5) 执行协同服务, 60 min 边缘计算协同服务池趋于稳态, TCSEC, KNN, K-means 算法平均并发数分别达 53, 41, 39, 平均服务流量分别为 20 973.83 MB/min, 10 254.34 MB/min, 9 770.61 MB/min。

计算协同服务拓扑如图 15 所示,聚类算法评价参数如表 3 所示,边缘计算协同服务池性能评价参数如表 4 所示,协同服务簇平均流量如图 16 所示,盟主节点

点数如图 17 所示,协同服务池分裂、聚合次数如图 18~19 所示,任务接受率如图 20 所示,协同服务成功率如图 21 所示,协同服务响应延迟如图 22 所示。

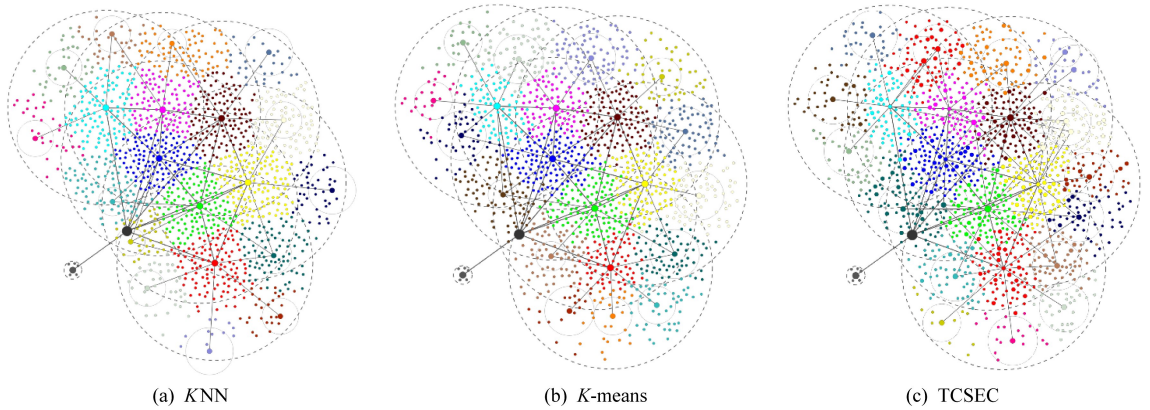


Fig. 15 Topology of dynamic edge computing cooperative service

图 15 动态边缘计算协同服务拓扑结构

Table 3 Performance Parameters of Dynamic Edge Computing Cooperative Service Clustering Algorithm

表 3 动态边缘计算协同服务算法性能参数表

Algorithms	Number of Clusters	Size of RN Cluster	Std of RN Cluster Size	Candidate Nodes in RN Cluster	Number of LN Clusters	Size of LN Clusters	Std of LN Cluster Size	Candidate Nodes in LN Cluster	Number of Free Node	Maximum Number of Hops	Stable Time /min	Aggregation Times	Divide Times	Degree of Aggregation
TCSEC	27	97.43	10.71	12.14	20	36.25	10.62	3.25	3	5	60	69	77	0.97
K-means	22	97.57	15.45		15	48.27	25.62		4	6	70	99	103	0.96
KNN	22	106.00	16.75		15	44.53	24.15		4	6	70	103	110	0.95

Note: Std means standard deviation.

Table 4 Performance Parameters of Dynamic Edge Computing Cooperative Service Pool

表 4 动态边缘计算协同服务池性能参数表

Algorithms	RTT in Edge Servers/ms	RTT in Cluster /ms	Number of Concurrency	Traffic of Cluster /(MB/min)	Traffic of RN /(MB/min)	Bandwidth of RN Cluster /Mbps	Traffic of LN /(MB/min)	Bandwidth of LN Cluster /Mbps	Free Resource Utilization
TCSEC	20	20	64	38 900.53	4 016.54	9.27	2 909.53	4.11	0.30
K-means	61	21	53	21 196.40	3 587.86	9.13	2 610.50	4.50	0.25
KNN	63	19	54	20 900.53	3 549.61	12.71	2 593.29	4.43	0.25

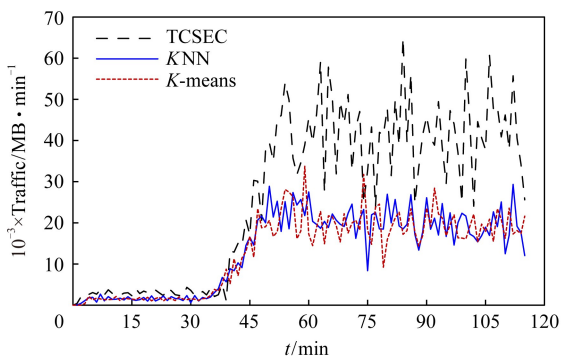


Fig. 16 Cooperative service cluster traffic for dynamic edge computing

图 16 动态边缘计算协同服务簇平均流量

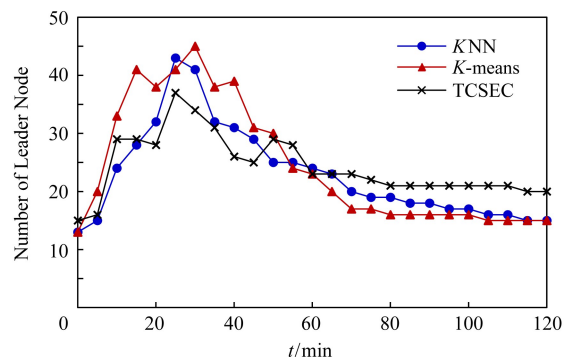


Fig. 17 Number of leader nodes in dynamic edge computing

图 17 动态边缘计算协同服务盟主节点数

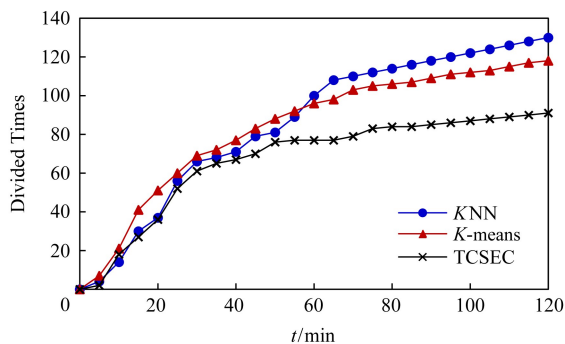


Fig. 18 Total divided times of dynamic edge computing

图 18 动态边缘计算协同服务总分裂次数

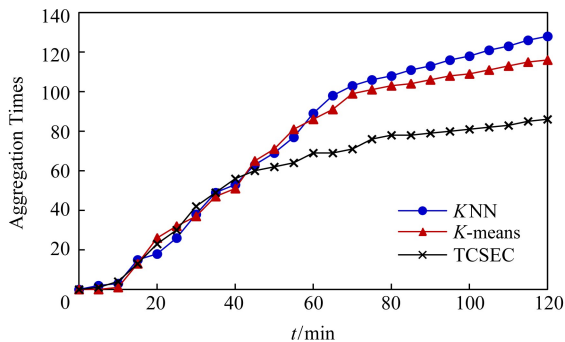


Fig. 19 Total aggregation times of dynamic edge computing

图 19 动态边缘计算协同服务总聚合次数

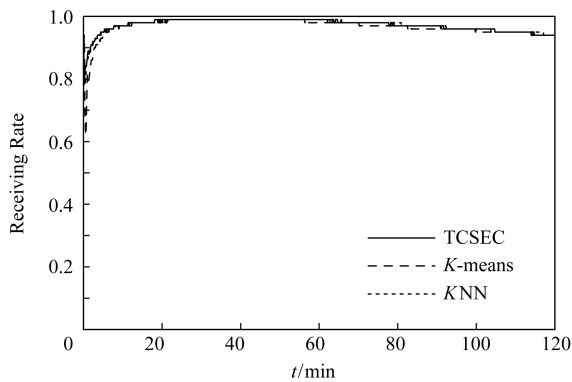


Fig. 20 Task receiving rate of dynamic edge computing

图 20 动态边缘计算协同服务任务接收率

动态边缘计算协同服务具有 3 个特征:

1) KNN, K-means 算法在 70 min 达到稳定状态, 分别形成 20 和 15 个盟主节点, 构成 27 和 23 个虚拟服务池, 平均规模分别为 44.53 和 48.27. TCSEC 算法在 60 min 达到稳定状态, 形成 23 个超级节点, 构成 30 个虚拟服务池, 平均规模为 36.25.

2) KNN, K-means, TCSEC 算法聚合度和协同效率基本一致, 但其分裂数分别为 110, 103, 77 次, 合

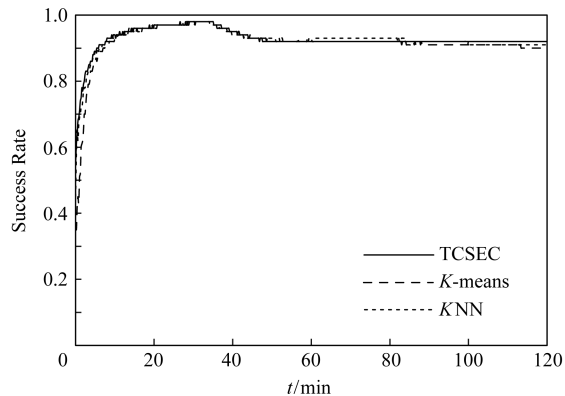


Fig. 21 Task success rate of dynamic edge computing

图 21 动态边缘计算协同服务成功率

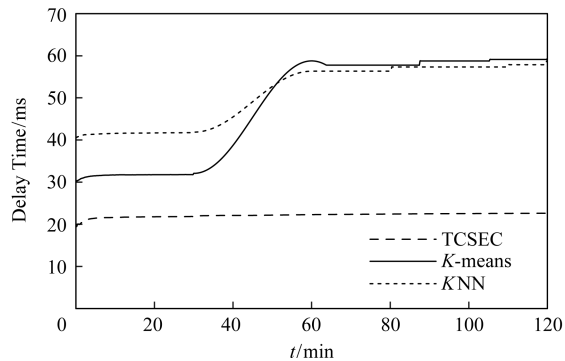


Fig. 22 Response delay of dynamic edge computing

图 22 动态边缘计算协同服务响应延迟

并数分别为 103, 99, 69 次; 中继基站平均服务带宽为 3 549.61 MB/min, 3 587.86 MB/min, 4 016.54 MB/min, 盟主节点平均服务带宽为 2 593.29 MB/min, 2 610.50 MB/min, 2 909.53 MB/min, 网络平均延迟分别为 63 ms, 61 ms, 20 ms, 协同服务池内交互平均延迟为 19 ms, 21 ms, 20 ms.

3) 边缘计算协同服务池进入稳态工作后, TCSEC, KNN, K-means 算法平均服务流量分别为 38 900.53 MB/min, 21 196.40 MB/min, 20 900.53 MB/min.

实验结果表明:

1) TCSEC, KNN, K-means 算法在达到相同聚合度和协同效率状态下, TCSEC 算法收敛时间比 KNN, K-means 快 14.29%, TCSEC 算法分裂合并总次数比 KNN, K-means 算法分别减少 31.46% 和 27.72%;

2) TCSEC 算法超级节点服务带宽分别比 KNN, K-means 算法提高 12.19% 和 11.46%, 协同服务池平均服务流量提高 83.52% 和 86.12%;

3) TCSEC 算法构建边缘计算协同服务的网络平均响应速度分别比 KNN, K-means 算法快 68.25% 和 67.21%。

小结:

1) 动态与静态构建边缘计算协同服务系统相比, TCSEC 算法聚合度达到 0.97, 协同服务系统总平均服务流量提高 41.53%, 协同效率提高 18.98%, 协同服务成功率提高 28.17%;

2) TCSEC 算法综合性能明显优于 KNN, K-means 算法。

4.3 TCSEC 模型系统仿真分析

为分析边缘计算协同服务系统交互特征, 以访问量、流量、负载、信誉度、任务接收率、协同成功率

等为评价指标, 基于 TCSEC 动态聚类对边缘计算协同服务系统进行仿真。

设数据源边缘服务器 (edge service, ES) 集群存在 100 个待迁移计算任务, 每个迁移计算任务量 $500 \leq M \leq 1024$, 对目标计算任务所在边缘服务器 ES 进行浪涌式服务请求测试。用户请求频率 $f \leq 4$ 次/s, 每次请求 5 个连续的计算任务, 实验测试时间 6 h, 访问量、流量采样周期为 30 s, 信誉度、任务接收率、协同成功率采样周期为 5 s。

4.3.1 TCSEC 模型系统访问量分析

为分析浪涌访问下, 边缘计算协同服务访问量变化特征, 数据源边缘服务器 ES, 中继基站 RN 和盟主节点 LN 的访问量变化如图 23 所示:

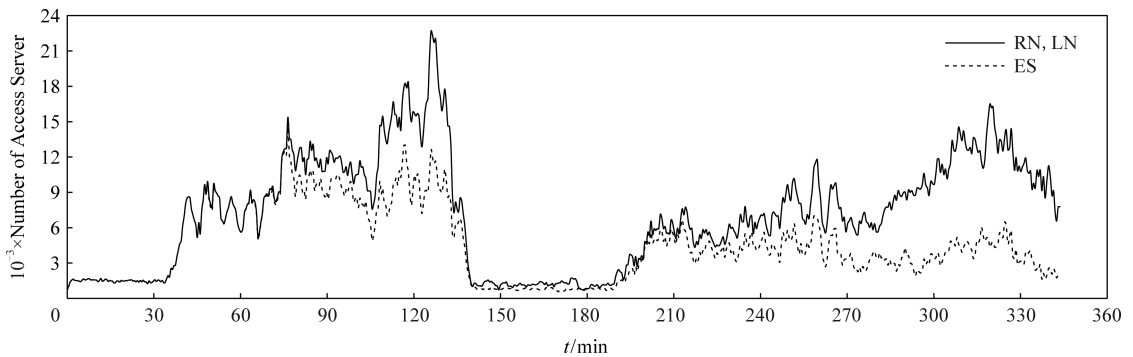


Fig. 23 Visits of cooperation service for edge computing

图 23 边缘计算协同服务访问量

由图 23 可知, 边缘计算协同服务访问量具有 2 个特征:

1) 34 min 系统出现第 1 次浪涌请求, ES 与 RN, LN 的访问量峰值分别达到 9919 和 9920, 平均访问量为 7745 和 7673; 71 min 系统出现第 2 次浪涌请求, ES 与 RN, LN 的访问量峰值分别达到 14567 和 15353, 平均访问量为 9498 和 11433; 106.6 min 系统出现第 3 次浪涌请求, ES 与 RN, LN 的访问量峰值分别达到 12994 和 22671, 平均访问量为 9441 和 15791; 124 min 达到系统设定的访问量上限, 触发协同服务池分裂和聚合, 135 min 结束第 1 阶段浪涌访问测试。

2) 180 min 时开始实施与第 1 阶段相同的浪涌访问。在 185.0 min 时, 出现第 1 次浪涌请求, ES 与 RN, LN 的访问量快速上升, 系统启动边缘计算协同服务机制, 实施浪涌访问均衡, ES 与 RN, LN 的访问量峰值分别为 6982 和 7734, 平均访问量为 3111 和 3814; 230 min 时, 出现第 2 次浪涌请求, ES 与 RN, LN 的访问量峰值分别为 6982 和 11765, 平均访问

量为 4265 和 7262; 280 min 时, 出现第 3 次浪涌请求, ES 与 RN, LN 的访问量峰值分别为 6477 和 16475, 平均访问量为 3704 和 10913; 340 min 结束浪涌访问测试。

TCSEC 模型系统访问量实验结果表明:

1) 第 1 阶段测试时, 第 1 次浪涌请求, ES 与 RN, LN 的访问量在可承受范围内, 访问量差异不大。第 2 次浪涌请求, 数据源边缘服务器访问量超过了阈值 12000 次, 任务迁移触发, ES 与 RN, LN 构建边缘计算协同服务系统, 并将大量任务迁移分发至中继基站和盟主节点, 访问量逐渐表现出差异。第 3 次浪涌请求, RN, LN 的访问量超过了阈值 18000 次, ES 的访问量超过了阈值 12000 次, 系统触发分裂聚合协同操作, 进行协同服务资源再度融合和边缘计算协同服务系统重构, 18 min 后达到稳定。

2) 第 2 阶段测试时, 实施浪涌访问均衡, 再次面临相同的浪涌请求时, ES 的访问量将维持稳定, RN, LN 的峰值访问量比第 1 阶段平均下降 21.31%, 平均访问量下降 30.89%。

3) 计算任务服务节点提供的计算任务因“新鲜”被访问节点传播吸引更多的节点访问,使计算任务服务节点的访问量随着访问节点的增加而增加,当访问量达一个极大值时,由于目标计算任务的“新鲜”度开始下降,访问节点随之减少,访问量下降。

4) ES 与 RN, LN 构建协同服务后,其访问量

变化具有联动放大效应.通过访问请求迁移分发和构建边缘计算协同服务,能够有效抑制访问拥塞。

4.3.2 TCSEC 模型系统性能分析

为分析浪涌访问下,边缘计算协同服务系统的服务能力及均衡性,ES, RN, LN 节点的负载率如图 24 所示,实时流量如图 25 所示。

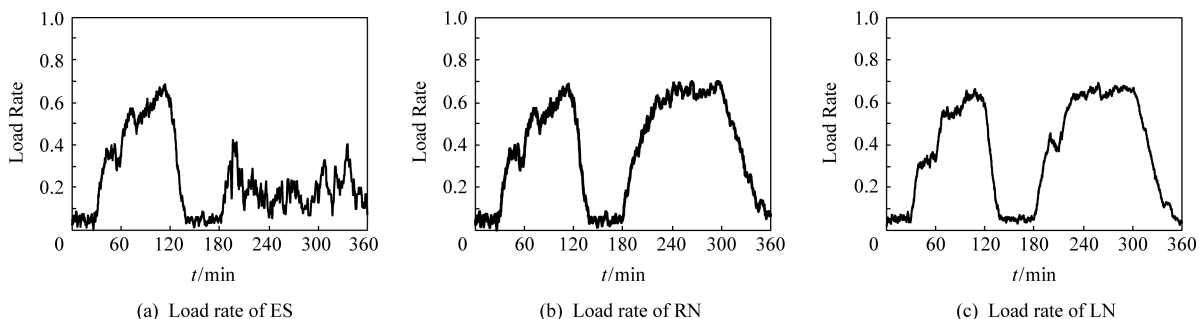


Fig. 24 Load rate of node in TCSEC

图 24 TCSEC 模型系统节点负载率

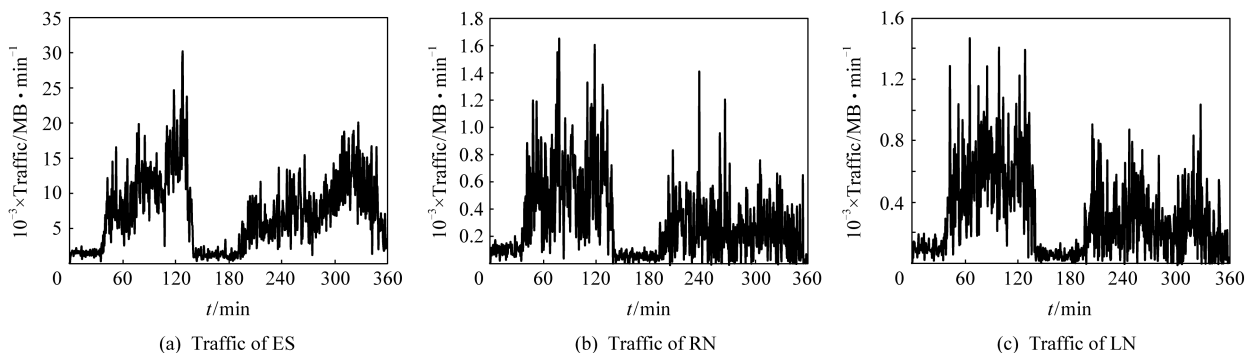


Fig. 25 Traffic of node in TCSEC

图 25 TCSEC 模型系统节点实时流量

由图 24、图 25 可知 TCSEC 模型系统节点流量具有 3 个特征:

1) 第 1 次浪涌请求, ES 峰值带宽峰值流量达到 15554.17 MB/min, 平均流量为 7642.50 MB/min; RN 节点和 LN 节点的峰值流量分别为 1373.71 MB/min 和 797.81 MB/min, 平均流量为 525.04 MB/min 和 521.01 MB/min. 第 2 次浪涌请求, ES 带宽峰值流量达到 20919.72 MB/min, 平均流量为 11746.31 MB/min; RN 节点和 LN 节点的峰值流量分别为 1970.95 MB/min 和 1368.94 MB/min, 平均流量为 660.36 MB/min 和 651.31 MB/min. 第 3 次浪涌请求, ES 峰值流量达到 26926.74 MB/min, 平均流量为 14416.21 MB/min; RN 节点和 LN 节点的峰值流量分别为 1670.90 MB/min 和 1307.68 MB/min, 平均流量为 810.70 MB/min 和 699.44 MB/min.

2) 构建边缘计算协同服务系统, 启动边缘计算协同服务后, ES 峰值流量为 3070 MB/min, 平均流量为 1758.08 MB/min; RN 节点和 LN 节点峰值流量为 1045.81 MB/min 和 924.62 MB/min, 平均流量为 307.66 MB/min 和 302.78 MB/min.

3) ES, RN, LN 流量分别于 41 min, 52 min, 62 min 达到任务迁移阈值, 启用请求服务转发迁移. 65 min 时, ES 流量超过额定阈值, ES 与 RN, LN 构建骨干网络边缘计算协同服务系统. 124 min, RN, LN 在自治域内构建网络边缘计算协同服务系统。

TCSEC 模型系统节点负载和流量实验结果表明:

1) 第 1 阶段浪涌访问测试, ES, RN, LN 是系统服务的主要提供者, 第 2 阶段浪涌访问测试, 以 RN 和 LN 为盟主构建的边缘计算协同服务是系统服务的主要提供者。

2) ES 负载达到设定阈值时,启用构建边缘计算协同服务,将任务迁移至盟员 RN 和 LN,其流量增速降低 69%,当 RN 和 LN 触发构建边缘计算协同服务后,ES 负载率降低 51%并趋于稳定。

3) 边缘计算协同服务盟主节点的流量波动,将直接带动其从属节点产生相同的流量波动,所构建的边缘计算协同服务具有马氏效应。

4) 所有节点均具有优先与综合性能优的节点连接趋势,盟员节点因综合性能差异而聚集到不同的边缘计算协同服务中,形成边缘计算协同服务的综合性能参数和服务质量也不尽相同。

5) 普通盟主节点聚类构建的边缘计算协同服务,由于其节点综合性能较弱,因此其边缘计算协同服务的服务能力也相对较弱。这类边缘计算协同服务在系统中主要承担保障网络连通性和为局域提供服务的作用。

4.3.3 TCSEC 模型系统信誉度验证

为验证边缘计算协同服务系统信誉度的有效性,RN 节点、LN 节点、EN 节点信誉度如图 26 所示。协同服务系统节点信誉度具有 3 个特征:

1) RN 节点和 LN 节点信誉度分别在 8.6 min 和 8.7 min 达到 0.8,此时 RN 节点和 LN 节点间信任关系初步建立,在 22.0 min 和 20.8 min 时达到 0.9,此时 RN 节点和 LN 节点间的主从关系得到确立。

2) 在 60~120 min 和 180~240 min,浪涌请求访问量达到极大值时,RN 节点和 LN 节点信任关系稳定于 0.92~0.94。

3) 第 1 阶段浪涌访问测试,边缘服务器协同节点于 50 min 信誉度开始逐步上升,在 130 min 时达到稳定。第 2 阶段浪涌访问测试,边缘服务器节点于 190 min 信誉度开始逐步上升,于 300 min 稳定于 0.94。

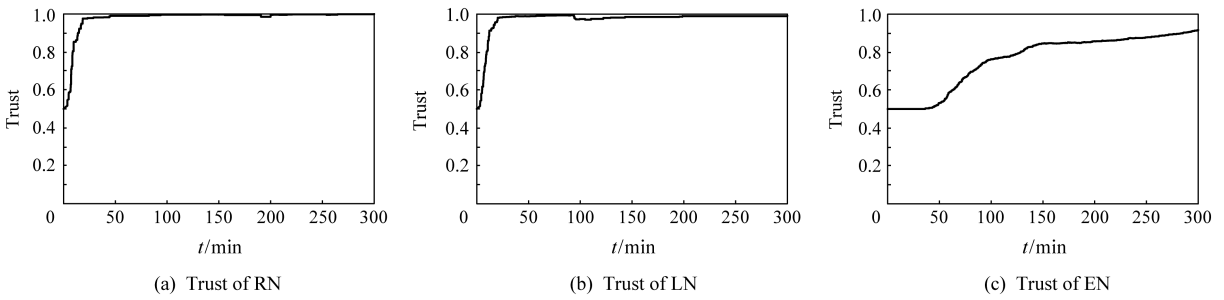


Fig. 26 Trust of node in TCSEC
图 26 TCSEC 模型系统节点信誉度

TCSEC 模型系统节点信誉度实验结果表明:

1) 节点的平均在线时长、服务能力、可用带宽、信誉度和链路延迟对边缘计算协同服务的综合性能产生重要影响。

2) RN 节点和 LN 节点平均信誉度相对稳定,有效激励了协同服务盟员参与协同的热情,高效完成协同任务。

4.3.4 TCSEC 模型系统用户服务质量评价

为分析浪涌访问下,用户服务质量服务参数,通过任务接收率、协同服务成功率、链路重寻次数、用户服务带宽、协同服务响应延迟和协同传输丢包率进行评价,如图 27~31 所示。

由图 27~31 可知,边缘计算协同服务参数具有 2 个特征:

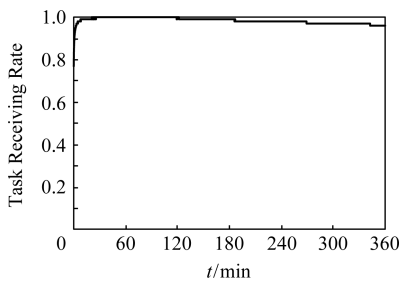


Fig. 27 Task receiving rate
图 27 任务接收率

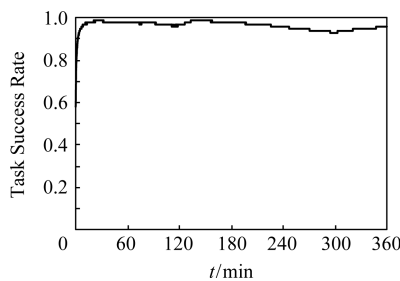


Fig. 28 Task success rate
图 28 交付成功率

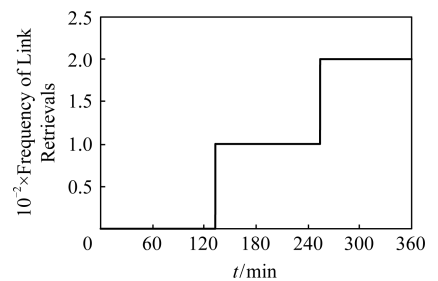


Fig. 29 Frequency of link retrievals of EN
图 29 盟员链路平均重寻次数

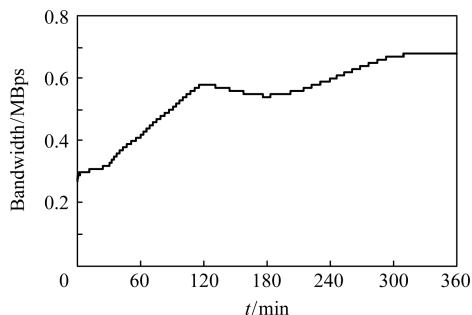


Fig. 30 Average service bandwidth of EN
图 30 盟员平均服务带宽

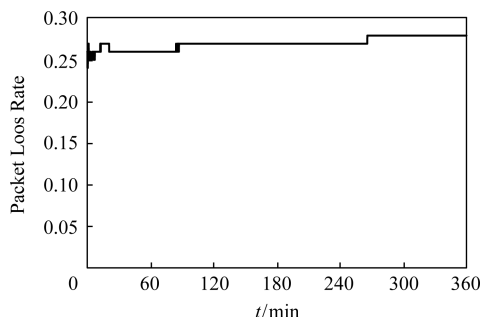


Fig. 31 Average packet loss rate
图 31 平均丢包率

1) 边缘计算协同服务任务接收率在第 8 min 稳定于 0.99, 协同成功率在第 10 min 稳定于 0.97, 丢包率在第 4 min 稳定于 0.3.

2) 第 1 阶段浪涌访问测试, 节点获得服务速率由 0.29 MBps 提高至 0.58 MBps, 链路平均重寻次数为 0. 启动协同服务机制, RN 节点和 LN 节点构建边缘计算协同服务系统, 节点获得服务速率由 0.56 MBps 提高至 0.68 MBps, 链路平均重寻次数 0.02.

节点信誉度实验结果表明:

1) 当信誉体系建立完成后, 在浪涌访问下, 边缘计算协同服务系统任务接收率稳定于 0.99, 协同成功率稳定于 0.97.

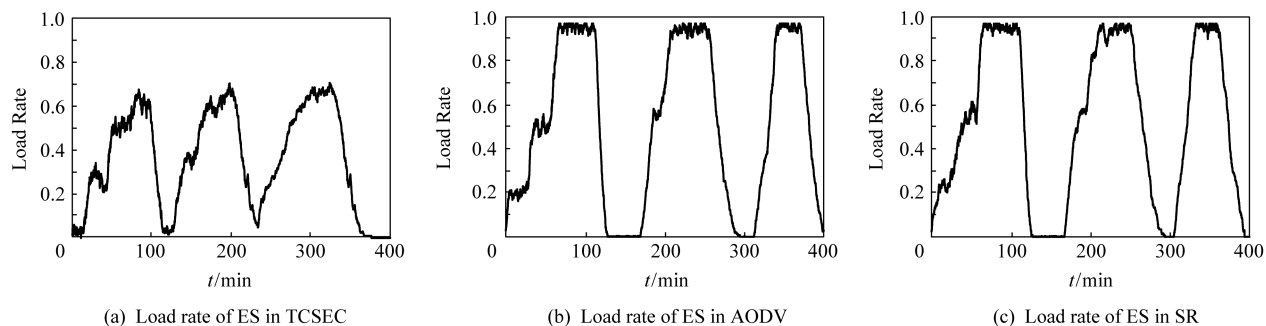


Fig. 32 Load rate of ES in edge computing cooperative service

图 32 边缘计算协同服务数据源服务器负载率

2) 随着边缘计算协同服务系统协同规模扩大, 链路重寻率上升 2%, 系统的服务效率提升 1.95 倍.

通过服务迁移和构建边缘计算协同服务系统, 能够有效解决无线网络中存在高拥塞、单点失效、低数据分发效率问题, 提高服务质量, 实现负载均衡, 为大规模可信边缘计算服务提供保障.

4.4 TCSEC 模型系统性能对比

为对比边缘计算协同服务系统性能, 以流量、负载、信誉度、任务接收率、协同成功率等为评价指标, 基于 TCSEC 动态聚类, 将本文提出的 TCSEC 与随机游牧 (stochastic routing, SR)^[54]、按需协同路由 (ad hoc on-demand distance vector routing, AODV)^[55] 进行对比分析.

设 ES 集群存在 100 个待迁移计算任务, 每个迁移计算任务量 $500 \leq M \leq 1024$, 对目标计算任务所在 ES 进行浪涌式服务请求测试. 用户请求频率 $f \leq 6$ 次/s, 链路重寻次数 ≤ 3 , 每次请求 5 个连续的流数据文件, 实验测试时间 10 h, 采样时间设定与 4.3 节实验一致.

TCSEC 与随机游牧、AODV 协作的 ES 的负载和流量如图 32, 33 所示, 中继基站 RN、盟主节点 LN 平均流量如图 34 所示, 信誉度实验结果如图 35 所示, 访问高峰段系统服务参数如表 5~表 6, 协同服务响应延迟如图 36 所示, 协同服务任务迁移丢包率如图 37 所示.

边缘计算协同服务系统具有 5 个特征:

1) 第 1 阶段浪涌访问测试访问高峰时段 60~120 min, TCSEC, SR, AODV 算法 ES 峰值流量分别为 28 201 MB/min, 13 182 MB/min, 15 754 MB/min, 峰值负载分别为 0.65, 0.97, 0.97, 平均流量 16 891 MB/min, 6 491 MB/min, 7 746 MB/min, 平均负载分别为 0.66, 0.96, 0.95, 基站、盟主节点峰值流量分别为 15 286 MB/min, 19 454 MB/min, 19 438 MB/min, 峰值负载分别

为 0.74,0.97,0.97,平均流量分别为6 589 MB/min, 6 579 MB/min,7 824 MB/min,平均负载分别为 0.68, 0.93,0.92.TCSEC,SR,AODV 算法链路平均重寻次数分别为 2.18,2.25,0.01,协同服务接收率和成功率均分别为 0.83,0.50,0.47,丢包率分别为 0.21, 0.39,0.30.

2) 第 2 阶段浪涌访问测试访问高峰段 210~270 min,TCSEC,SR,AODV 算法 ES 峰值流量分别为 10 230 MB/min,6 574 MB/min,9 633 MB/min,峰值负载分别为 0.70,0.97,0.97,平均流量分别为 5 179 MB/min,4 586 MB/min,5 171 MB/min,平均负载分别为 0.49,0.95,0.94,基站、盟主节点峰值流量分别为 7 854 MB/min,10 604 MB/min,16 890 MB/min,峰值负载分别为 0.67,0.97,0.97,平均流量分别为 2 020 MB/min,4 607 MB/min,5 108 MB/min,平均负载分别为 0.58,0.93,0.92.TCSEC,SR,AODV 算法平均链路重寻次数分别为 2.23,2.20,0.01,协同服务接收率和成功率均分别为 0.74,0.70,0.45,丢包率分别为 0.21,0.39,0.31.

3) 第 3 阶段浪涌访问测试访问高峰段 360~

420 min,TCSEC,SR,AODV 算法 ES 峰值流量分别为 12 700 MB/min,4 258 MB/min,2 961 MB/min,峰值负载分别为 0.70,0.97,0.97,平均流量分别为 6 714 MB/min,795 MB/min,870 MB/min,平均负载分别为 0.48,0.94,0.94,基站、盟主节点峰值流量分别为 9 752 MB/min,10 546 MB/min,15 459 MB/min,峰值负载分别为 0.65,0.97,0.97,平均流量分别为 1 610 MB/min,3 962 MB/min,3 899 MB/min,平均负载分别为 0.61,0.93,0.92;TCSEC,SR,AODV 算法平均链路重寻次数分别为 2.18,2.25,0.01,协同服务接收率和成功率均分别为 0.89,0.39,0.45,丢包率分别为 0.21,0.40,0.30.

4) 实验测试 20 min,TCSEC,SR,AODV 算法信誉度均为 0.92.访问高峰时段,TCSEC 算法 RN, LN 的信誉度稳定于 0.97~0.98,而 SR,AODV 算法 RN,LN 的信誉度趋于 0.

5) 系统边缘计算协同服务达到稳定后,TCSEC,SR,AODV 算法已获得服务的盟员平均带宽分别为 2.5 MBps,2.5 MBps,2.41 MBps,平均访问延迟分别为 24 ms,37 ms,29 ms.

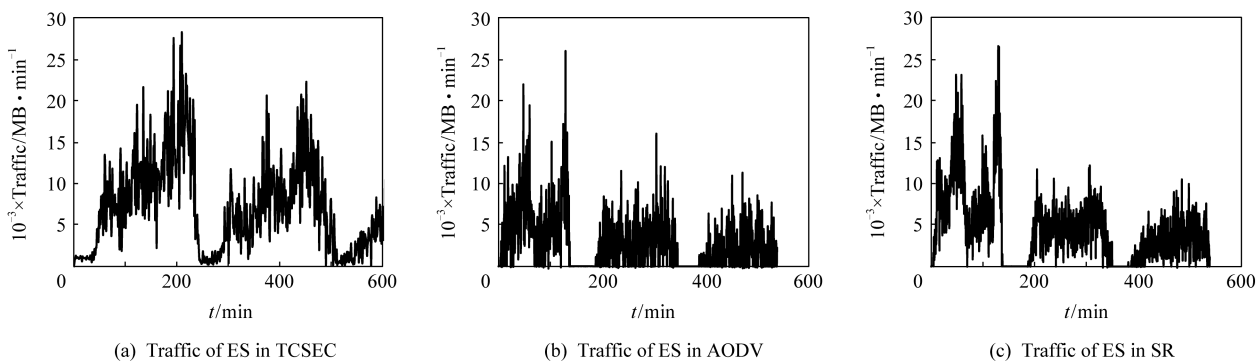


Fig. 33 Traffic of ES in edge computing cooperative service

图 33 边缘计算协同服务数据源服务器流量

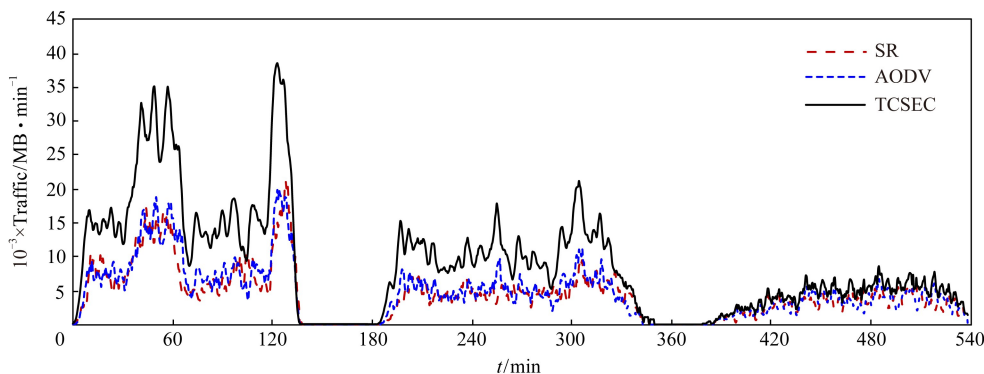


Fig. 34 Average traffic of RN and LN in TCSEC,AODV and SR algorithms

图 34 TCSEC,AODV,SR 算法中继基站和盟主节点平均流量

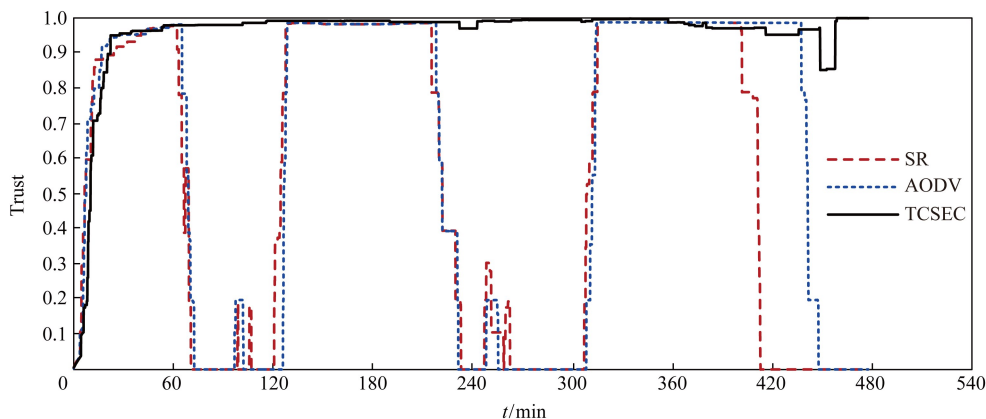


Fig. 35 Average trust of RN and LN in TCSEC, AODV and SR algorithms

图 35 TCSEC, AODV, SR 算法中继基站和盟主节点平均信任度

Table 5 Performance Parameters of Cooperation Service for Edge Computing During Peak Access Time

表 5 边缘计算协同服务访问高峰时段性能参数

Algorithms	First Time During Peak Access(60~120 min)						Second Time During Peak Access (210~270 min)					
	Peak Time/s	Task Receiving Rate	Task Success Rate	Frequency of Link Retrievals	Packet Loss Rate	Bandwidth of Service /Mbps	Peak Time/s	Receiving Rate	Task Success Rate	Frequency of Link Retrievals	Packet Loss Rate	Bandwidth of Service /Mbps
SR	3 940	0.50	0.50	2.18	0.39	2.50	13 280	0.39	0.39	2.23	0.39	2.50
AODV	4 148	0.47	0.47	2.25	0.30	2.34	13 119	0.45	0.45	2.20	0.31	2.44
TCSEC	2 285	0.83	0.83	0.01	0.21	2.50	15 063	0.89	0.89	0.01	0.21	2.50

Table 6 Performance Parameters of Cooperation Service for Edge Computing During Peak Access Time

表 6 边缘计算协同服务访问高峰时段性能参数

Algorithms	Third Time During Peak Access (360~420 min)					
	Peak Time/s	Task Receiving Rate	Task Success Rate	Frequency of Link Retrievals	Packet Loss Rate	Bandwidth of Service/Mbps
SR	23 564	0.40	0.40	2.39	0.40	2.50
AODV	24 520	0.40	0.40	2.44	0.30	2.47
TCSEC	23 999	0.90	0.90	0.01	0.21	2.50

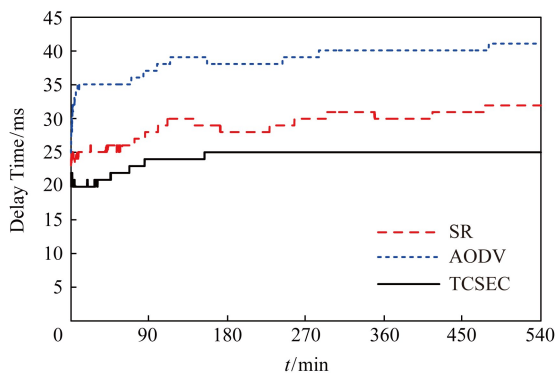


Fig. 36 Access delay in edge computing cooperative service

图 36 边缘计算协同服务访问延迟

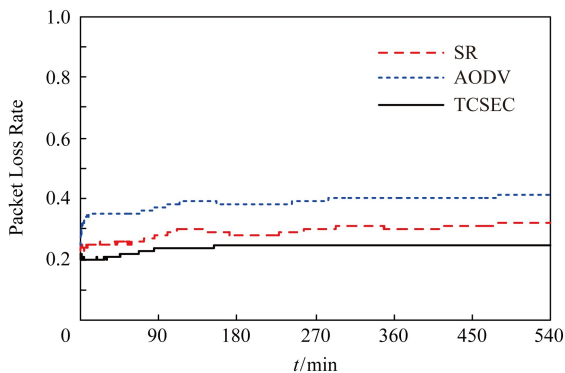


Fig. 37 Packet loss rate in edge computing cooperative service

图 37 边缘计算协同服务平均丢包率

实验结果表明:

1) 信誉评价与节点的负载、协同成功率相一致,能够有效反映协同节点实时状态.信誉度迅速下降直接反映了系统出现高拥塞、访问过载等现象.

2) TCSEC能够在浪涌式访问请求测试实现负载均衡,有效地完成协同任务,SR,AODV算法则因负载过大发生雪崩效应.

3) 在访问高峰时段,TCSEC与SR,AODV算法相比,ES峰值负载平均降低31.0%,平均负载平均降低40.6%,基站、盟主节点峰值负载降低27.7%,平均负载平均降低27.3%.

4) TCSEC与SR,AODV算法相比,协同任务接收率分别提高29.1%,36.4%,协同成功率分别提高29.0%,36.3%,协同传输丢包率分别降低18.3%,9.3%.

5) TCSEC与SR,AODV算法相比,平均服务带宽基本一致,请求接收率分别提高29.1%,36.4%,任务成功率分别提高29.1%,36.4%,接收延迟分别降低35.1%,17.2%.

小结:TCSEC与SR,AODV算法相比,在负载均衡能力上提高29%,系统服务能力分别提高29%和36%,服务质量分别提高35%,17%.TCSEC算法能够在浪涌式访问请求下有效地完成协同任务.

5 总 结

以数据源边缘服务器、岸基中心基站、岛礁/智能浮台中继基站、盟主节点、船舰、智能移动节点等构建的近海港口边缘计算协同服务面临网络拓扑动态、资源受限、抖动、搭便车、盲区效应、策略和恶意节点等诸多挑战,导致网络服务质量、容量和带宽的不稳定性,使网络效用下降.为保障网络服务效用,构建自适应协同服务系统是一种有效方法.从负载均衡的角度出发,根据边缘计算协同服务的负载情况,动态构造由协同服务盟员组成的边缘计算协同服务来解决目前近海港口边缘计算协同服务中存在拥塞、单点失效、盲区、效率和服务质量不高的问题.提出了一种基于盟主的边缘计算协同服务组织模型TCSEC,模型运用节点的信任度、贡献度、容量和带宽、链路质量等作为表征节点的特征属性,以任务驱动方式,由盟主节点基于节点的服务能力和相似性来选择协同服务的节点,构建边缘计算协同服务.给出了构建边缘计算协同服务的数学模型、约束条件

和构造规则,并进行了较为详细的分析.仿真实验表明,通过构建边缘计算协同服务,能有效解决边缘计算服务提供节点过载的问题,实现了协同服务、资源共享、负载均衡的目的,提高了近海港口环境下边缘计算协同服务的服务质量.

参 考 文 献

- [1] China Academy of Information and Communications Technology. 5G economic and social impact white paper [EB/OL]. (2017-06-13) [2019-10-10]. http://www.caict.ac.cn/kxyj/qwfb/bps/201804/t20180426_158438.htm (in Chinese) (中国信息通信研究院. 5G 经济社会影响白皮书[EB/OL]. (2017-06-13) [2019-10-10]. http://www.caict.ac.cn/kxyj/qwfb/bps/201804/t20180426_158438.htm)
- [2] Shi Weisong, Zhang Xingzhou, Wang Yifan, et al. Edge computing: State-of-the-art and future directions [J]. Journal of Computer Research and Development, 2019, 56(1): 69-89 (in Chinese) (施巍松, 张星洲, 王一帆, 等. 边缘计算: 现状与展望[J]. 计算机研究与发展, 2019, 56(1): 69-89)
- [3] 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)
- [4] Turner V, Gantz J F, Reinsel D, et al. The digital universe of opportunities: Rich data and the increasing value of the Internet of things [EB/OL]. (2018-11-26) [2019-10-10]. <https://www.emc.com/leadership/digital-universe/2014iview/high-value-data.htm>
- [5] Cisco. Cisco visual networking index: Global mobile data traffic forecast, 2016-2021 Q&A [OL]. (2018-07-31) [2019-10-10]. <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-forecast-qa.html>
- [6] China Internet Network Information Center. The 42nd China Statistical Report on Internet Development [EB/OL]. (2018-08-20) [2019-10-19]. <http://www.cnnic.net.cn/hlwfzyj/hlwzbg/hlwtjbg/201808/P020180820630889299840.pdf> (in Chinese) (中国互联网络中心. 中国互联网络发展状况统计报告[EB/OL]. (2018-08-20) [2019-10-19]. <http://www.cnnic.net.cn/hlwfzyj/hlwzbg/hlwtjbg/201808/P020180820630889299840.pdf>)
- [7] Hu Yunchao, Patel M, Sabella D, et al. Mobile edge computing-A key technology towards 5G [EB/OL]. (2015-09-01) [2019-10-10]. <https://www.etsi.org/media-library/white-papers>

- [8] Shi Weisong, Cao Jie, Zhang Quan, et al. Edge computing: Vision and challenges [J]. *IEEE Internet of Things Journal*, 2016, 3(5): 637-646
- [9] Cheshire S, Krochmal M. DNS-based service discovery [EB/OL]. (2013-02-01) [2019-10-10]. https://www.heise.de/netze/rfc/rfc/rfc_6763.shtml
- [10] Cao Jie, Xu Lanyu, Abdallah R, et al. Edge OS_H: A home operating system for Internet of everything [C] //Proc of the 37th IEEE Int Conf on Distributed Computing Systems (ICDCS2017). Piscataway, NJ: IEEE, 2017: 1756-1764
- [11] Deng Xiaoheng, Guan Peiyuan, Wan Zhiwen, et al. Integrated trust based resource cooperation in edge computing [J]. *Journal of Computer Research and Development*, 2018, 55(3): 449-477 (in Chinese)
(邓晓衡, 关培源, 王志文, 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)
- [12] Yu Yinghao, Zhang Jun, Letaief K B. Joint subcarrier and CPU time allocation for mobile edge computing [C] //Proc of Global Communications Conf. Piscataway, NJ: IEEE, 2017
- [13] Xu Chen, Lei Jiao, Li Wenzhong, et al. Efficient multi-user computation offloading for mobile-edge cloud computing [J]. *IEEE/ACM Transactions on Networking*, 2015, 24(5): 2795-2808
- [14] Qi Yanli, Zhou Yiqing, Liu Ling, et al. MEC coordinated future 5G mobile wireless networks [J]. *Journal of Computer Research and Development*, 2018, 55(3): 478-486 (in Chinese)
(齐彦丽, 周一青, 刘玲, 等. 融合移动边缘计算的未来 5G 移动通信网络[J]. *计算机研究与发展*, 2018, 55(3): 478-486)
- [15] Katsalis K, Papaioannou T G, Nikaein N, et al. SLA-driven VM scheduling in mobile edge computing [C] //Proc of IEEE Int Conf on Cloud Computing. Piscataway, NJ: IEEE, 2017: 750-757
- [16] Zhu Youkang, Yue Guangxue, Yang Xiaohui, et al. A survey on edge computing offloading [J]. *Telecommunications Science*, 2019, 35(4): 74-94 (in Chinese)
(朱友康, 乐光学, 杨晓慧, 等. 边缘计算迁移研究综述[J]. *电信科学*, 2019, 35(4): 74-94)
- [17] Yue Guangxue, Zhu Youkang, Liu Jiansheng, et al. Optimizing strategy of computing off loading energy consumption based on Lagrangian method [J]. *Telecommunications Science*, 2018, 34(12): 10-23 (in Chinese)
(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[J]. *电信科学*, 2018, 34(12): 10-23)
- [18] Yue Guangxue, Dai Yasheng, Yang Xiaohui, et al. Multi-constrained trusted cooperative task migration strategy for edge computing [J]. *Telecommunications Science*, 2019, 35(11): 36-50 (in Chinese)
(乐光学, 戴亚盛, 杨晓慧, 等. 边缘计算多约束可信协同任务迁移策略[J]. *电信科学*, 2019, 35(11): 36-50)
- [19] Wang Bin, Wu Yajing, Yang Xiaolong, et al. Dependency-driven task scheduling scheme of big data processing [J]. *Journal of Software*, 2017, 28(12): 3385-3398 (in Chinese)
(王玢, 吴雅婧, 阳小龙. 关联性驱动的大数据处理任务调度方案[J]. *软件学报*, 2017, 28(12): 3385-3398)
- [20] Yu Bowen, Pu Lingjun, Xie Yuting, et al. Joint task offloading and base station association in mobile edge computing [J]. *Journal of Computer Research and Development*, 2018, 55(3): 537-550 (in Chinese)
(于博文, 蒲凌君, 谢玉婷, 等. 移动边缘计算任务卸载和基站关联协同决策问题研究[J]. *计算机研究与发展*, 2018, 55(3): 537-550)
- [21] 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
- [22] Guo Yanchao, Gao Ling, Wang Hai, et al. Power optimization based on dynamic content refresh in mobile edge computing. *Journal of Computer Research and Development*, 2018, 55(3): 563-571 (in Chinese)
(郭延超, 高岭, 王海, 等. 移动边缘计算中基于内容动态刷新的能耗优化[J]. *计算机研究与发展*, 2018, 55(3): 563-571)
- [23] Tang Guoming, Wang Huan, Wu Kui, et al. Tapping the knowledge of dynamic traffic demands for optimal CDN design [J]. *IEEE/ACM Transactions on Networking*, 2019, 27(1): 98-111
- [24] 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
- [25] You Changsheng, Huang Kaibin, Chae H, et al. Energy-efficient resource allocation for mobile-edge computation offloading [J]. *IEEE Transactions on Wireless Communications*, 2017, 16(3): 1397-1411
- [26] Hang Wenlong, Jiang Yizhang, Liu Jiefang, et al. Transfer affinity propagation clustering algorithm [J]. *Journal of Software*, 2016, 27(11): 2796-2813 (in Chinese)
(杭文龙, 蒋亦樟, 刘解放, 等. 迁移近邻传播聚类算法[J]. *软件学报*, 2016, 27(11): 2796-2813)
- [27] Pei Ren, Qiao Xiuquan, Chen Junliang, et al. Mobile edge computing—A booster for the practical provisioning approach of web-based augmented reality [C] //Proc of 2018 IEEE/ACM Symp on Edge Computing (SEC). Piscataway, NJ: IEEE, 2018: 349-358
- [28] Lin Gao, Iosifidis G, Huang Jianwei, et al. Bargaining-based mobile data offloading [J]. *IEEE Journal on Selected Areas in Communications*, 2014, 32(6): 1114-1125
- [29] 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

- [30] Chai Yunpeng, Gu Lei, Li Sanli. Cluster-based edge streaming server with adaptive load balance in mobile grid [J]. *Journal of Computer Research and Development*, 2007, 44(12): 2136-2142(in Chinese)
(柴云鹏 顾雷, 李三立. 移动网格中负载均衡的集群式边缘流媒体服务器[J]. *计算机研究与发展*, 2007, 44(12): 2136-2142)
- [31] Liu Tingting, Li Jun, Shu Feng, et al. Distributed resource allocation in caching-enabled heterogeneous cellular networks based on matching theory [J]. *CCF Transactions on Networking*, 2019, 2(1): 57-68
- [32] Fang Junhua, Wang Xiaotong, Zhang Rong, et al. High-performance data distribution algorithm on distributed stream systems [J]. *Journal of Software*, 2017, 28(3): 563-578 (in Chinese)
(房俊华, 王晓桐, 张蓉, 等. 分布式数据流上的高性能分发策略[J]. *软件学报*, 2017, 28(3): 563-578)
- [33] Li Pengwei, Fu Jianming, Li Shuanbao, et al. Elastic mobile cloud computing: State of the art and security analysis [J]. *Journal of Computer Research and Development*, 2015, 52(6): 1362-1377 (in Chinese)
(李鹏伟, 傅建明, 李拴保, 等. 弹性移动云计算的研究进展与安全性分析[J]. *计算机研究与发展*, 2015, 52(6): 1362-1377)
- [34] Xiong Jinbo, Ma Rong, Niu Ben, et al. Privacy protection incentive mechanism based on user-union matching in mobile crowdsensing [J]. *Journal of Computer Research and Development*, 2018, 55(7): 1359-1370 (in Chinese)
(熊金波, 马蓉, 牛犇, 等. 移动群智感知中基于用户联盟匹配的隐私保护激励机制[J]. *计算机研究与发展*, 2018, 55(7): 1359-1370)
- [35] Wang Shangguang, Sun Qibo, Yang Fangchun. Reputation evaluation approach in Web service selection [J]. *Journal of Software*, 2012, 23(6): 1350-1367 (in Chinese)
(王尚广, 孙其博, 杨放春. Web 服务选择中信誉度评估方法[J]. *软件学报*, 2012, 23(6): 1350-1367)
- [36] Wu Xu. Enhanced stable group model-based trust evaluation scheme for mobile P2P network [J]. *Chinese Journal of Computers*, 2014, 37(10): 2118-2127 (in Chinese)
(吴旭. 基于增强稳定组模型的移动 P2P 网络信任评估方法[J]. *计算机学报*, 2014, 37(10): 2118-2127)
- [37] Li Feng, Si Yali, Chen Zhen, et al. Trust-based security routing decision method for opportunistic networks [J]. *Journal of Software*, 2018, 29(9): 2829-2843 (in Chinese)
(李峰, 司亚利, 陈真, 等. 基于信任机制的机会网络安全路由决策方法[J]. *软件学报*, 2018, 29(9): 2829-2843)
- [38] Lin Hui, Tian Youliang, Xu Li, et al. A novel privacy aware secure routing protocol for HWMN [J]. *Journal of Computer Research and Development*, 2015, 52(8): 1883-1892 (in Chinese)
(林晖, 田有亮, 许力, 等. HWMN 中一种新的隐私感知安全路由协议[J]. *计算机研究与发展*, 2015, 52(8): 1883-1892)
- [39] Wang Yang, Wang Ruzhuan, Yan Yuanting, et al. TCLM-P2P: A logic model of task cooperation for P2P community [J]. *Journal of Computer Research and Development*, 2012, 49(2): 270-277 (in Chinese)
(王杨, 王汝传, 严远亭, 等. TCLM-P2P: 面向 P2P 社区的任务协作逻辑模型[J]. *计算机研究与发展*, 2012, 49(2): 270-277)
- [40] Zeng Mingfei, She Shunzheng. P2P network server deployment scheme and its heuristic optimization algorithm [J]. *Journal of Software*, 2013, 24(9): 2226-2237 (in Chinese)
(曾明霏, 余顺争. P2P 网络服务器部署方案及其启发式优化算法[J]. *软件学报*, 2013, 24(9): 2226-2237)
- [41] Cheng Boliang, Guo Sun, Jie Anquan. Research on multi-agent trusted alliance based on self-organizing evolution [J]. *Journal of Computer Research and Development*, 2010, 47(8): 1382-1391 (in Chinese)
(程柏良, 国荪, 揭安全. 基于自组织演化的多 Agent 可信联盟研究[J]. *计算机研究与发展*, 2010, 47(8): 1382-1391)
- [42] Hu Jun, Zhang Zhenxing, Zou Li. The formation mechanism of distributed automated negotiation alliance based on collaboration degree [J]. *Journal of Computer Research and Development*, 2015, 52(5): 1080-1090 (in Chinese)
(胡军, 张振兴, 邹立. 基于协作度的分布式自动协商联盟形成机制[J]. *计算机研究与发展*, 2015, 52(5): 1080-1090)
- [43] Liu Jinglei, Zhang Wei, Liu Zhaowei, et al. The nature and application of alliance structure diagram [J]. *Journal of Computer Research and Development*, 2011, 48(4): 602-609 (in Chinese)
(刘惊雷, 张伟, 刘兆伟, 等. 联盟结构图的性质及应用[J]. *计算机研究与发展*, 2011, 48(4): 602-609)
- [44] Cheng Bailiang, Jie Anquan, Zeng Guosun, et al. Constrained evolution equation and probability calculation of trusted alliance under group random motion [J]. *Chinese Journal of Computers*, 2012, 35(12): 2479-2491 (in Chinese)
(程柏良, 揭安全, 曾国荪, 等. 群体随机运动下可信联盟的约束演化方程及其概率计算[J]. *计算机学报*, 2012, 35(12): 2479-2491)
- [45] Stuart M A, Gualtierio C, Roger M W. Cooperation through self-similar social networks [J]. *ACM Transactions on Autonomous and Adaptive Systems*, 2010, 5(1): 83-111
- [46] Wang Xiaofan, Li Xiang, Chen Guanrong. *The Theory and Application of Complex Network* [M]. Beijing: Tsinghua University Press, 2006 (in Chinese)
(汪小帆, 李翔, 陈关荣. *复杂网络理论及其应用*[M]. 北京: 清华大学出版社, 2006)
- [47] Satyanarayanan M. A brief history of cloud offload [J]. *ACM SIGMOBILE Mobile Computing and Communications Review*, 2015, 18(4): 19-23

- [48] Sladana J, György D. Decentralized algorithm for randomized task allocation in fog computing systems [J]. IEEE/ACM Transactions on Networking, 2019, 27(2): 85-97
- [49] Song Liumeng, Chai K K, Chen Yue, et al. Cooperative coalition selection for quality of service optimization in cluster-based capillary networks [J]. IEEE Systems Journal, 2018, 12(2): 1700-1708
- [50] Shao Yanling, Li Chunlin, Fu Zhao, et al. Cost-effective replication management and scheduling in edge computing [J]. Journal of Network and Computer Applications, 2019, 129(1): 46-61
- [51] Yu Fangxiaopi, Chen Haopeng, Xu Jinqing. DMPO: Dynamic mobility-aware partial offloading in mobile edge computing [J]. Future Generation Computer Systems, 2018, 89: 722-735
- [52] Huang Shuqian, Wang Gaocai, Zhang Zhen, et al. A method of geometric k -center gateway deployment of wireless mesh networks [J]. Chinese Journal of Computers, 2013, 36(7): 1475-1484 (in Chinese)
(黄书强, 王高才, 张震, 等. 一种无线 Mesh 网络几何 K 中心网关部署求解方法[J]. 计算机学报, 2013, 36(7): 1475-1484)
- [53] Ban Dongsong, Wen Jun, Jiang Jie, et al. Constructing k -barrier coverage in mobile wireless sensor networks [J]. Journal of Software, 2011, 22(9): 2089-2103 (in Chinese)
(班冬松, 温俊, 蒋杰, 等. 移动无线传感器网络 k -栅栏覆盖构建算法[J]. 软件学报, 2011, 22(9): 2089-2103)
- [54] Zhang Yiwen, Wang Kaibin, Yan Yuanting, et al. Service quality prediction based on covering random walk algorithm [J]. Chinese Journal of Computers, 2018, 41(12): 2756-2768 (in Chinese)
(张以文, 汪开斌, 严远亭, 等. 基于覆盖随机游走算法的服务质量预测[J]. 计算机学报, 2018, 41(12): 2756-2768)
- [55] Aashkaar M, Sharma P. Enhanced energy efficient AODV routing protocol for MANET [C] //Proc of Int Conf on Research Advances in Integrated Navigation Systems. Piscataway, NJ: IEEE, 2016: 1-5



Yue Guangxue, born in 1963. PhD, professor. Member of IEEE and CCF. His main research interests include cloudy convergence and collaborative services, wireless mesh networks and mobile cloud computing.



Dai Yasheng, born in 1993. Master candidate. His main research interests include wireless mesh network trusted cooperative communication mechanism.



Yang Xiaohui, born in 1996. Master candidate. Member of CCF. Her main research interests include mobile cloud computing.



Liu Jianhua, born in 1978. PhD, Associate professor. Member of IEEE. His main research interests include distributed computing, wireless communications, multimedia networking, wireless sensor networks, and machine learning.



You Zhenxu, born in 1993. Master candidate. Member of CCF. Her main research interests include trust network and social recommendation algorithm.



Zhu Youkang, born in 1993. Master candidate. His main research interests include computational migration based on energy consumption optimization.