软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2021,32(2):300−326 [doi: 10.13328/j.cnki.jos.006151]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

数据中心负载均衡方法研究综述*

刘敬玲, 黄家玮, 蒋万春, 王建新

(中南大学 计算机学院,湖南 长沙 410083)

通讯作者: 黄家玮, E-mail: jiaweihuang@csu.edu.cn

摘 要: 随着云计算的发展,数据中心网络成为近年来学术界和工业界关注的研究热点.现代数据中心网络通常采用胖树等多根树拓扑结构,存在多条可用路径来提供高对分带宽.由于等价多路径路由等传统的负载均衡方法无法适应数据中心网络中高动态和强突发的流量特性,多种针对数据中心的负载均衡方法不断涌现.围绕数据中心中负载均衡的基本问题,介绍了当前国际国内的研究现状,包括基于中央控制器、基于交换机和基于主机的负载均衡方法,并展望了数据中心网络负载均衡的发展趋势.

关键词: 数据中心网络;负载均衡;异构流量;拥塞控制;延迟敏感

中图法分类号: TP316

中文引用格式: 刘敬玲,黄家玮,蒋万春,王建新.数据中心负载均衡方法研究综述.软件学报,2021,32(2):300−326. http://www.jos.org.cn/1000-9825/6151.htm

英文引用格式: Liu JL, Huang JW, Jiang WC, Wang JX. Survey on load balancing mechanism in data center. Ruan Jian Xue Bao/ Journal of Software, 2021,32(2):300−326 (in Chinese). http://www.jos.org.cn/1000-9825/6151.htm

Survey on Load Balancing Mechanism in Data Center

LIU Jing-Ling, HUANG Jia-Wei, JIANG Wan-Chun, WANG Jian-Xin

(School of Computer Science and Engineering, Central South University, Changsha 410083, China)

Abstract: With the development of cloud computing, recently data center network has been a hot research topic in both academia and industry. Modern data center network is commonly organized in multi-rooted tree topology, such as fat-tree, with multiple available paths to provide high bisection bandwidth. Since the traditional load balancing scheme such as equal-cost multipath routing is not suitable for highly dynamic and bursty traffic in data center network, many load balancing mechanisms have been proposed. In this study, based on the fundamental research problems of load balancing in data center network, the international and domestic research progress of this area is introduced, including central controller-based, switch-based, and host-based load balancing schemes, and then the research trend of load balancing is prospected in data center network.

Key words: data center network; load balancing; heterogeneous traffic; congestion control; delay-sensitive

在计算机技术和互联网应用迅猛发展的推动下,用户对数据访问的需求日益增多,信息对储存容量的要求日益提高.云计算使用户可以按需地享受高质量服务和无处不在的网络访问[1],但是用户将数据外包给云服务器,使数据脱离了物理控制,随之带来了数据隐私泄露的问题.

随着云计算技术的兴起,数据中心(data center)作为云计算的硬件基础架构也在不断普及和应用.为了构建

* 基金项目: 国家自然科学基金(61872387, 61572530, 61972421); 中南大学创新驱动计划(2020CX033); 中南大学中央高校基本科研业务费专项资金(2020zzts142)

Foundation item: National Natural Science Foundation of China (61872387, 61572530, 61972421); Innovation-driven Project of Central South University (2020CX033); Fundamental Research Funds for the Central Universities of Central South University (2020zzts142)

收稿时间: 2019-06-04; 修改时间: 2019-10-08, 2020-02-01; 采用时间: 2020-09-15; jos 在线出版时间: 2020-10-12

高可用、高性能、低成本的云计算基础存储和计算设施[1−5],数据中心通常部署了大量商用交换机和服务器.数据中心网络连接了大规模服务器集群,是传递计算和存储数据的桥梁[6].为了提供超高带宽,数据中心网络的拓扑结构普遍采用 CLOS 结构[7],在主机之间提供了多条可用路径.在网络高负载状态下,为了降低链路拥塞和数据包丢失的概率[8],数据中心负载均衡机制将网络流量分配到所有可用路径上,充分利用了网络中存在的冗余链路,提高网络传输性能.

由于数据中心之间的流量相对稳定,通常采用中心控制器统计数据中心间的流量状态信息,并下发负载均衡控制规则以均衡流量.而数据中心内部流量占数据中心网络流量的绝大部分,且数据中心内部流量具有高动态和强突发的特性[9,10],给负载均衡工作带来了更大的挑战.因此,本文仅对数据中心内部的负载均衡工作进行讨论.本文主要综述了数据中心负载均衡的相关研究工作.具体来说,本文根据负载均衡方案部署的位置将它们划分成了三大类.

- 基于中央控制器的负载均衡方案:采用集中式的思想,基于中央控制器的负载均衡方案,通过引入中央控制器,收集各个交换机上的流量信息,得到关于网络路径和动态流量的全局视图,并根据中央控制器上的全局视图对网络拥塞作出快速反应,自动调整网络流量的转发路径.
- 基于主机的负载均衡方案:基于主机的负载均衡方案将负载均衡操作转移到分布式的主机上进行.主机依据端到端的拥塞信息,通过修改 TCP/IP 协议栈或引入虚拟软件交换机技术,以重路由、切分流量或细粒度调度的方式实施负载均衡.
- 基于交换机的负载均衡方案:多对一的通信模式使得瓶颈交换机上的负载过高,形成链路拥塞.基于交换机的负载均衡方案在交换机上感知网络拥塞,并采用不同的调度粒度将网络流量发送到不同的路径,实现快速的流量均衡.

本文第 1 节描述数据中心的背景和负载均衡研究的意义.第 2 节介绍数据中心网络结构与流量特征.第 3 节具体介绍相关的负载均衡方案,并对比分析各方案的基本思想、实现方法及均衡效果.最后探讨未来值得关注的研究方向.

1 研究背景和意义

近年来,随着云计算、大数据、分布式存储等新兴技术飞速发展,越来越多的企业和政府部门搭建了大型数据中心来提供金融、电商、交通等各种各样的在线服务.作为下一代互联网应用服务的基础架构,数据中心吸引了工业界和学术界的关注,成为了研究的热点领域.为了充分利用数据中心强大的计算和存储能力,网页访问、即时通信、财经金融、在线游戏等延时敏感型服务和数据分析、科学计算、网页内容索引等计算密集型服务被迁移到数据中心.随着数据中心在线应用服务规模的不断扩大,对数据中心网络带宽和性能提出了更高的挑战.为了向用户提供满意的服务质量,数据中心网络(data center networks)的传输性能备受关注[11−13].

数据中心网络通过交换机连接数据中心内部的大量主机,获取规模效应.为了提高网络传输性能,在数据中心网络架构设计方面,出现了胖树(fat-tree)[14]、VL2[15]、DCell[16]、BCube[17]等新型的“富连接”的网络拓扑结构.这些新型网络拓扑结构在源和目的主机之间提供了多条可用传输路径,可以利用并行多路径传输来提升数据中心的网络吞吐率和可靠性[18−20].

在数据中心网络特有的网络流量和应用需求背景下,针对日益增长的用户规模,数据中心供应商不断升级硬件设备,使用 10Gb/s,100Gb/s 级别的高带宽和微秒级别的低延时链路来提升传输速率.学术界提出了 DCTCP[1]、D^3[21]、D^2TCP[22]等新型传输控制协议和 PDQ[23]、PIAS[24]等交换机调度算法,同时利用新型的多路径网络架构来提高并发传输速度和网络整体健壮性.这些方案在一定程度上提升了网络性能,但无法解决数据中心网络中并行路径传输的流量不均衡问题.研究数据表明:不同类型的数据中心通常只有不到 25%的核心链路利用率,其余链路总是处于空闲状态,存在着严重的负载不均衡问题[25].

负载均衡的首要目标是将流量均匀地分配到各条并行路径上,提升网络链路利用率,避免高负载流量引发网络拥塞.作为数据中心负载均衡的标准方案,等价多路径路由(equal-cost multipath routing,简称 ECMP)[26]策略

采用静态哈希机制,依据数据包头的源 IP 地址、目的 IP 地址、源端口号、目的端口号和协议这五元组信息,利用哈希函数将不同的数据流分散到等价多路径上.ECMP 虽然实现部署简单,但并不感知路径的拥塞状态和流量特征,容易造成多条数据流(特别是长流)在路径上发生哈希冲突,导致链路拥塞和应用性能下降.

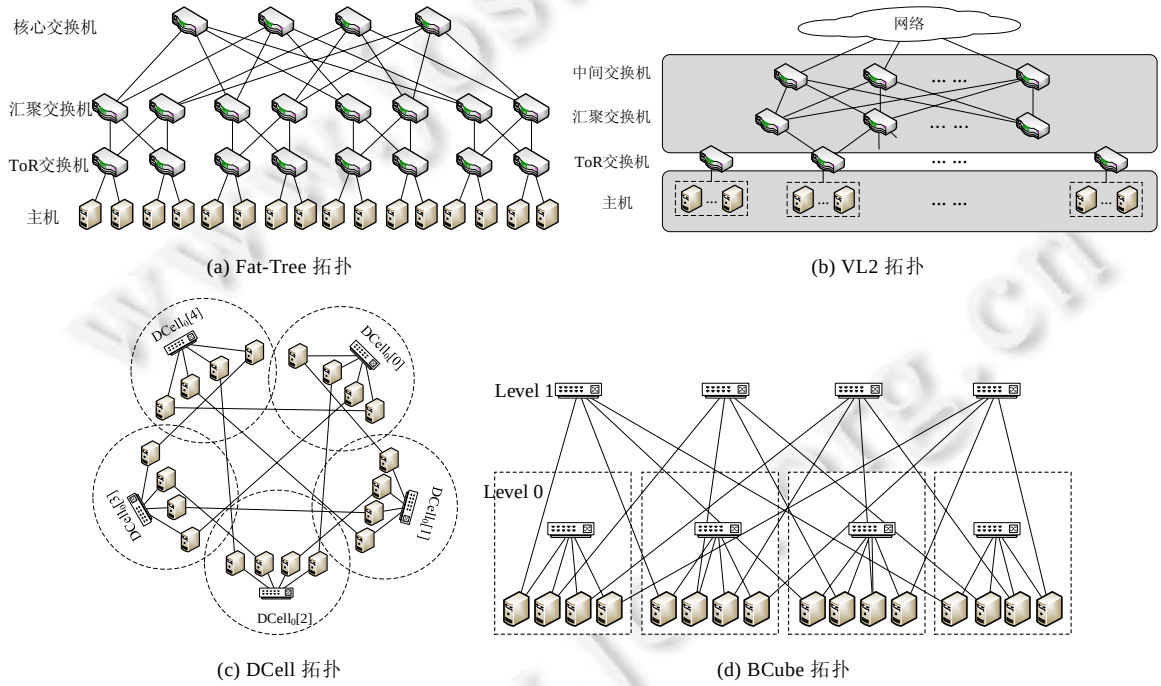
显然,在数据中心多路径网络环境和特有的网络应用下,如何设计高效的网络流量负载均衡机制,就成为了一个重要的研究问题.

2 数据中心网络结构与流量特征

为保证数据中心网络传输和应用服务的性能的关键技术,需要结合以下数据中心的拓扑结构和流量特性开展数据中心负载均衡机制的设计和优化.

2.1 数据中心网络结构

数据中心利用交换机和路由器等网络设备将大量服务器连接起来,以建设高性能的计算和存储基础设施.随着用户需求迅猛发展,数据中心的服务器数目在高速增长.由于传统基于树状的 3 层数据中心网络结构难以满足大规模数据中心的要求[7,27],如图 1 所示的 Fat-Tree[14]、VL2[15]、DCell[16]、BCube[17]等新型数据中心网络结构不断涌现,以提供超高带宽,降低大规模部署的开销,并适应不同应用需求.新型数据中心网络结构具有以下特点:(1) 主机之间存在大量可用并行路径,增加了网络的容错性和带宽;(2) 拓扑结构规则、对称,利于网络布线、自动化配置和扩展升级;(3) 对分带宽随着网络规模的扩展而增大,能为数据中心提供高速传输服务.



(a) Fat-Tree 拓扑 (b) VL2 拓扑

(c) DCell 拓扑 (d) BCube 拓扑

Fig.1 Typical architecture of modern data center network

图 1 典型的新型数据中心网络结构

2.2 数据中心流量特征

数据中心中运行着网页搜索、零售和广告等大量不同的应用系统,这些应用是数据中心的建设和研究的主要驱动因素,也是数据中心运营商的主要业务,它们会产生各种类型的数据流量.不同于传统的广域网,数据中心的网络流量和应用需求具有以下特点.

- 数据中心不同应用产生的数据流具有各不相同的网络传输性能需求[21,22].例如,在线搜索和实时推荐等应用的数据流往往有严格的延时期限需求,而虚拟机迁移和文件备份等应用的数据流则要求很高的网络吞吐率.
- 数据中心网络数据流的长度呈现重尾分布特性.据统计,大部分应用中数据流的数据量很小,90%的数据流属于小于 100KB 的短流,其数据传输在 10ms 内完成.数据量大于 100KB 的长流只占数据流总量的 10%,但这些长流却发送了超过 80%的数据量[1].其中,0.1%的长流发送时间超过 200s,其数据量占了总数据量的 20%[28].长流的大量数据包容易占据交换机缓存队列,对短流数据包形成线头阻塞,增大短流的延时.
- 数据中心网络流量呈现较强的突发性[29,30].在硬件设计上,目前高速网卡普遍采用了负载卸除设计来降低 CPU 的系统开销,以支持超高的链路带宽,但这使得大量数据包在极短时间内发送而形成突发流量;在传输控制协议 TCP 的拥塞控制机制下,发送方也会成簇发送拥塞窗口内的所有数据包;在应用层,数据中心通常采用划分聚合(partition/aggregate)的并发通信模式,即向服务器群发送请求后,服务器会同步地返回数据并汇聚生成响应结果,进一步加大了并行数据流的突发强度[30,31].

3 数据中心负载均衡方法研究进展

在本节中,我们将根据部署位置将数据中心网络流量负载均衡机制分成了 3 大类,包括基于中央控制器、基于主机和基于交换机的负载均衡机制.本节将具体介绍相关的负载均衡方案,并对比分析各方案的设计思想和工作原理.

3.1 基于中央控制器的负载均衡机制的研究

采用集中式的思想,基于中央控制器的负载均衡方案利用控制器对网络设备进行集中控制.中央控制器收集拥塞信息,并基于全局视图为数据流分配传输路径.控制器与交换机之间采用 OpenFlow 等协议通信.控制器定期测量各个交换机拥塞情况,根据全局链路利用率和流量信息向交换机下发转发表,以此决定数据流的转发路径,从而实现整体网络的负载均衡.近年来,学术界提出了多种基于中央控制器的方案,见表 1,它们的解决思路包括:① Hedera[32]、MicroTE[33]、Mahout[34]、OmniFlow[35]、Shafiee[36]等重路由方法;② Fastpass[37]、SAPS[38]、MSaSDN[39]等细粒度控制方法;③ RAPIER[40]、FDALB[41]、Freeway[42]、OFLoad[43]、AuTO[44]、SOFIA[45]等区分长短流的调度方法;④ LBDC[46]等多控制器间的负载均衡方法.

Table 1 Comprehensive comparison of central controller-based load balancing mechanisms

表 1 基于中央控制器的负载均衡机制的综合对比

文献名称	工作原理	拥塞信息	调度粒度	优势	劣势
Hedera	采用中央控制器估计带宽需求,重路由遭遇拥塞的长流	全局拥塞信息	流级别	解决长流哈希碰撞问题	控制周期过长,长流识别不准确
MicroTE	中央控制器为长流分配转发路径,剩余带宽则按加权等价多路径路由策略分配给难以预测的短流	全局拥塞信息	流级别	提升了长流的性能	控制周期过长,无法避免排队延时
Mahout	主机端通过套接字缓存判断长流,利用带内信号(DCSP 字段)通知中央控制器对长流进行重路由	全局拥塞信息	流级别	流量检测效率高,交换机开销小	需要修改主机内核
OmniFlow	结合负载均衡和流控来优化数据中心传输	全局拥塞信息	流级别	减少了短流的延时,同时保证了长流的吞吐率	阈值影响策略的选择,由于流量的动态性,难以选择合适的阈值
Shafiee 等人[36]	根据链路权重调节流量的传输路径	全局拥塞信息	流级别	算法的复杂度较低,不受网络结构的影响	对链路权重更新速度的要求较高
FastPass	使用中央控制器为每个数据包分配其发送时隙和传输路径来提升网络利用率和缓解拥塞,降低拥塞路径的排队延时	全局拥塞信息	包级别	实现了低排队延时	为每包分配时隙和路径,难以大规模部署和扩展

Table 1 Comprehensive comparison of central controller-based load balancing mechanisms (Continued)

表 1 基于中央控制器的负载均衡机制的综合对比(续)

文献名称	工作原理	拥塞信息	调度粒度	优势	劣势
SAPS	借助中央控制器检测链路故障,并建立对称的虚拟拓扑,各条流在虚拟拓扑中散射	全局拥塞信息	包级别	适用于非对称网络	控制器建立拓扑周期长
MSaSDN	控制器依据多播阻塞模型选择最小阻塞代价的链路作为最优路径	全局拥塞信息	流级别	解决组播流量引起的负载不均衡问题	控制器开销大
RAPIER	结合路由和调度提升任务性能,利用中央控制器确定大任务中流的转发路径、转发时间和服务速率	全局拥塞信息	流级别	降低平均任务完成时间	中央控制器的开销大
FDALB	在主机端依据流大小分布区分长短流,短流的路径由交换机决定,长流的路径由全局拥塞感知的中央控制器决定	全局拥塞信息	流级别	减少了流碰撞事件,提升了控制器的扩展性	仅感知发送端本地的流量分布,没有考虑其他发送端的流量,无法计算最优的阈值
Freeway	将长短流隔离在不同的路径上传输,短流采用 ECMP 路由策略,长流使用中央控制器调度	全局拥塞信息	流级别	解决长短流资源竞争问题	流级别的调度方式,无法灵活地使用多路径
OFLoad	利用 OpenFlow 交换机为长短流执行不同的路由规则,长流分配在最小利用率路径上传输,短流分配在带权重的路径上传输	全局拥塞信息	流级别	解决短流被长流阻塞的问题	控制器信息收集不及时,无法解决突发拥塞
AuTO	采用机器学习方法调节长短流的优先级、长流的发送速率与发送路径	全局拥塞信息	流级别	解决了短流被长流阻塞的问题,同时也保证了长流的吞吐量	流量的多样性加剧复杂性
SOFIA	依据流大小分布计算区分长短流的最佳阈值,以减少控制器的开销	全局拥塞信息	流级别	减少了控制器的内存和延时开销,同时减轻了链路拥塞	阈值的计算取决于流大小分布,流大小分布不准确极大地影响性能
LBDC	监控分布式控制器的拥塞情况,将流量最多的交换机移交给拥塞最轻的控制器管理	全局拥塞信息	交换机	解决多个控制器间负载不均衡的问题	控制器间存在同步性问题

3.1.1 基于重路由的中央控制器负载均衡机制

Hedera[32]方案是一个集中式的主动负载均衡算法,如图 2 所示.中央控制器从 ToR 交换机上收集流信息以检测长流,计算冲突路径后,通告各交换机为遭遇拥塞的长流重新切换转发路径.为了实现最大最小公平的带宽分配,Hedera 建立长流的带宽需求矩阵,通过多次调整速率,以收敛至公平带宽.Hedera 采用了全局最先匹配和模拟退火算法两个调度算法,以增加对分带宽.全局最先匹配算法遍历所有可能路径,将长流分配到第 1 条能够满足带宽需求的路径上.同时,Hedera 为各目标主机而不是为各流分配核心交换机,以缩小搜索空间,再利用模拟退火算法为长流选择路径.Hedera 利用全局信息避免了长流间的冲突,但是对短流并不友好.

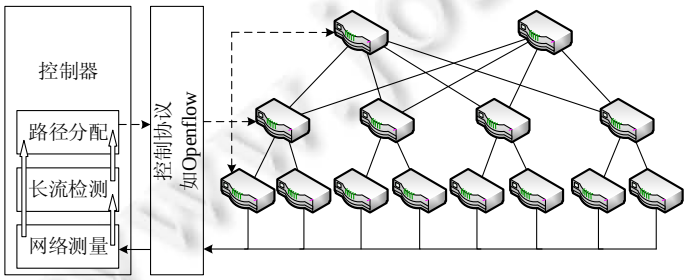


Fig.2 Schematic diagram of Hedera

图 2 Hedera 原理图

针对流量相对稳定的长流和突发性强的短流,MicroTE[33]方案使用中央控制器来跟踪长流,并优先分配转发路径,剩余带宽则按加权等价多路径路由策略分配给难以预测的短流.MicroTE 包含 3 个组件:监测组件,用于监控 ToR 间的流量模式和确定流量的可预测性;路由组件,基于控制器提供的网络信息确定路由;网络控制器,收集服务器的流量需求,并下发转发表给交换机.

Mahout[34]方案首先在主机端依据套接字缓存占用情况探测长流,当套接字缓存占用超过阈值时,判定该主机所发送的流为长流.主机端利用数据包包头 DSCP 字段通知中央控制器对该流进行重路由.具体来说:当交换机接收到带标记的数据包时,将其转发给中央网络控制器,中央控制器为相应长流分配最轻负载的路径;对于其他流量,交换机则采用 ECMP 对它们进行转发.Mahout 利用主机检测长流,提高了检测效率同时降低了交换机的开销,但同时也需要修改主机内核.

由于数据中心负载均衡器容易将长流的数据包分散到各条路径上,造成短流经历较长的排队延时;主动降低流速的流控方案虽减小了缓冲占用,但易引起多路径的带宽损失.OmniFlow[35]方案是一个在传输层结合负载均衡和流控来优化数据中心传输的方法.基于不同的网络状况,OmniFlow 动态地调整流量转发路径以充分利用带宽,或主动调整流速限制队列长度.OmniFlow 包括 3 个模块.

(1) 队列监视器:首先测量主机间多路径的排队延时,然后根据测量结果评估当前路径的质量,并相应地选择优化策略(负载均衡或流量控制)以改善传输.若负载适中,采用负载均衡充分利用网络带宽;当网络严重饱和时,使用流控方法快速排空队列.

(2) 负载均衡模块:重新路由流量到缓存队列长度低于给定阈值的路径.

(3) 流量控制模块:检测到所有路径的队列长度超过给定阈值时,降低流速以减轻网络拥塞.OmniFlow 在不损失长流吞吐率的同时降低了短流的延时.

Shafiee 等人[36]提出一种基于流级别的拥塞感知的数据中心网络负载均衡方法,该方法根据链路利用率为链路动态调整权重,并将新生成的流量放置在网络中的最小权重路径上传输.作者在理论和实践上证明了该算法具有良好的负载均衡性能,并证明了该算法能渐近地最小化网络开销.该算法不受流量模式的限制,且在不同的数据中心网络状态下良好地运行.虽然该算法的复杂度较低,但其性能会受到权重更新和最小权重路径计算速度的影响.

以上方案利用中央控制器实现网络流量的全局负载均衡,容易造成两个问题:一方面,由于控制周期过长,无法应对毫秒级甚至微秒级的突发流量;另一方面,难以保证长流识别准确性,无法避免数据包排队问题.

3.1.2 细粒度控制的中央控制器负载均衡机制

Fastpass[37]是一种细粒度的集中式传输控制架构,实现了多路径传输的低排队延时和高利用率.Fastpass 使用中央控制器为每个数据包分配发送时隙和传输路径来提升网络利用率和避免拥塞,有效降低了排队延时.收到终端主机向控制器发送的请求后,控制器为终端主机分配传输数据时隙;然后,控制器采用 ToR 着色算法为已分配时隙的数据包分配路径.但 Fastpass 的性能取决于发送时隙的利用率.当发送时隙被浪费时,很容易降低链路利用率.另外,Fastpass 为每包分配时隙和路径,难以大规模部署和扩展.

SAPS[38]方案是一个基于软件定义网络的包分散机制.SAPS 借助中央控制器检测链路故障,并建立对称的虚拟拓扑.虚拟拓扑对于一条流来说,其源端和目的端之间的所有路径具有对称的带宽和延时.每条流的数据包被分散到一个虚拟拓扑中.每条流依据其流大小与虚拟拓扑的二分带宽选择虚拟拓扑.其中,短流和长流分别映射到拥有更小和更大的二分带宽的虚拟拓扑中.SAPS 可在非对称拓扑中实现高性能.

组播不仅能降低任务完成时间,也能提高吞吐量.但由组播流量引起的网络阻塞严重影响数据中心的性能.在胖树拓扑的数据中心中,组播流量更容易造成负载不均衡和突发的流量冲突.MSaSDN[39]方案提出了一种面向胖树数据中心的组播调度算法.MSaSDN 方案首次构建了软件定义胖树数据中心中组播阻塞模型,并通过链路权重定义链路阻塞代价.SDN 控制器在接收到组播请求后,根据收集的网络信息计算链路阻塞代价,选择具有最小阻塞代价的链路作为最优路径.MSaSDN 方案借助 SDN 控制策略可以防止对其他子网中组播流量的干扰,降低网络阻塞概率.

3.1.3 区分长短流调度方式的中央控制器负载均衡机制

数据中心绝大部分数据流量是由少量的长流产生,大量的短流则通常都有严格的完成期限.吞吐量敏感的长流和延时敏感的短流之间存在资源竞争问题.

RAPIER[40]方案是一个任务敏感的优化系统.RAPIER 为了降低平均任务完成时间,结合路由和调度机制,构建了一个联合优化模型.RAPIER 使用中央控制器确定任务中每条流的转发路径、转发时间和服务速率,以最优化平均任务完成时间.当新任务到达时,RAPIER 为新任务中每条流计算路由路径和传输速率.当已存在的任务完成时,网络资源被释放,RAPIER 再重新分配带宽.RAPIER 主要处理数据敏感的大任务,而延时敏感的独立流和小任务直接以 ECMP 的方式路由.相比于只关注流调度的方案,RAPIER 提升了任务的性能.

FDALB[41]方案是一个感知流大小分布的负载均衡机制.FDALB 在主机端测量流大小分布,检测并标记长流.当一条流的已发送字节数超过一个阈值时,其数据包包头将被标记.一旦一条流结束,FDALB 重新计算长短流区分的阈值.交换机在接收到被标记的包时,将其转发给中央控制器.中央控制器收到被标记的包后,基于全局的拥塞信息为其做负载均衡决策.为了防止长流同步到达交换机时,中央控制器为它们选择相同的路径带来的碰撞问题,FDALB 使用了一个贪婪轮询算法调度长流到有最大共享剩余带宽的路径上传输.交换机在接收到短流的包时,直接以 ECMP 的方式传输.FDALB 方案减少了流碰撞事件数量,同时实现了高可扩展性.但是 FDALB 仅感知发送端本地的流量分布,没有考虑其他发送端的流量,难以计算得到最优的区分长短流阈值.

Freeway[42]方案是一个区分长短流调度方式的机制.为了避免长短流间的资源竞争,Freeway 将长短流分别隔离到高带宽路径和低延时路径上传输.Freeway 利用控制器收集流量和路径状态信息,动态地调整低延时路径和高带宽路径的数量,以保证短流的完成期限和长流的吞吐率.Freeway 的路径分割算法依据实时短流负载情况变化调节低延时路径的数量.Freeway 基于 M/G/1 排队理论为短流建立了队列模型,以 SLA(service-level agreement)定义的短流完成期限为约束,分析了低延时路径的最高延时阈值.当低延时路径的平均延时超过 2 倍阈值时,增加低延时路径数量;否则,将低延时路径调整为高吞吐量路径,分配给长流以增加可用带宽.在 Freeway 中,短流采用本地调度策略 ECMP 算法,而长流用中心调度的方式调度.Freeway 将长短流隔离在不同的路径上传输,能有效地解决长短流的资源竞争问题.但由于采用了流级别的调度方式,无法灵活使用多路径.

OFLoad[43]是一种基于 OpenFlow 的数据中心网络动态负载平衡方案,可实现数据中心的自动路由配置和负载优化.OFLoad 在主机上利用应用层信息区分和标记长流和短流.在 OpenFlow 交换机上首先执行 ECMP 路由策略,然后定期向控制器发送流信息.控制器为长流和短流采用不同的路由规则.对于长流,控制器将其分配到一条利用率最低的路径上传输.同时,控制器收集到同一目的 ToR 交换机的短流,并依据路径负载为这些短流分配路径,使得负载重的路径被分配较少量的短流.属于同一条短流的数据包在一条路径上传播,以避免乱序问题.OFLoad 可有效解决长流阻塞短流的问题.

除了将长短流分配到不同的路径上传输,学者们也提出将长短流隔离在不同的优先级队列中来保证长短流的性能[24,47].然而,区分优先级队列的阈值直接影响网络性能.由于流量的动态性和流大小的多样性,无法采用固定的优先级阈值,AuTO[44]采用了强化学习的方法实现自动的流量优化.AuTO 的中央系统收集各主机上的流信息,使用了深度强化学习技术来确定长短流的优先级和长流的发送速率与发送路径.为了最小化短流的延时,短流被给予高的优先级,同时直接以 ECMP 的方式传输;为了保证长流的吞吐率,中央系统单独给长流分配发送速率和发送路径.AuTO 解决了短流被长流阻塞的问题,同时也保证了长流的吞吐量,但 AuTO 依赖于机器学习方法的准确性,高动态和强突发的流量会增加 AuTO 的复杂性.

数据中心中,SDN 控制器通常被用来为流量分配路径.新流或需要重路由的流在交换机上无法匹配流表项时,会向控制器发送 packet-in 消息请求分配路径.每条流的路由建立会消耗控制器的内存,同时,频繁的 packet-in 事件易引起过大的延时.为了降低控制器的内存和延时开销,控制器通常只重路由流大小大于给定静态阈值的长流来减轻 ECMP 的哈希冲突问题.但理想情况下,任何一条流都应该被路由到最佳路径上传输.SOFIA[45]设计了一种最佳流量分割控制策略.控制器利用在线学习算法根据流大小分布计算区分长短流的最佳阈值,并统一通告给所有交换机.交换机根据分割阈值判定可重路由的流.SOFIA 减少了 packet-in 消息延迟,降低了路由开销

和链路拥塞程度.

3.1.4 针对多个中央控制器的负载均衡机制

大型数据中心网络通常被分割成多个区域,每个区域分别由一个控制器监测和重路由流量.多个控制器能有效分担负载,但也带来了控制器间负载不均衡的问题.LBDC[46]首次提出数据中心中分布式控制器的负载平衡问题,并证明该问题是 NP 完全的.同时,LBDC 给出了集中式和分布式的贪婪方法来迁移交换机的控制权,以解决控制器间负载不均衡的问题.集中式和分布式方案分别利用了全局网络视图和本地信息,在控制器间流量不均衡时迁移交换机的控制权,将具有最大流量的交换机移交给负载最轻的控制器管理.

3.2 基于主机的负载均衡机制的研究

虽然集中式的负载均衡方案可利用中央控制器收集全局流量信息,根据全局信息实现最优的负载均衡决策,但是获取和维护全局信息需要一定的部署开销,较大的反馈和控制延时也会降低动态突发流量下的集中式负载均衡性能.另外,还需要部署额外的网络组件,增加实施成本.基于主机的负载均衡方案将负载均衡操作转移到分布式的主机上进行,降低了部署开销.近年来,学术界提出了多种基于主机的方案,见表 2,它们的解决思路可分为:① FlowBender[48]、CLOVE[49]、ALB[50]、Hermes[51]、ELAB[52]等重路由方法;② MPTCP[53]、FUSO[54]、DCMPTCP[55]、DC^2-MTCP[56]、MMPTCP[57]、Presto[58]、VMS[59]、DumbNet[60]、Flicr[61]等流量切分协议;③ DRB[62]、JUGGLER[63]、NDP[64]、CAPS[65]、MP-RDMA[66]等细粒度调度机制.

Table 2 Comprehensive comparison of host-based load balancing mechanisms

表 2 基于主机的负载均衡机制的综合对比

文献名称	工作原理	拥塞信息	调度粒度	优势	劣势
FlowBender	通过修改哈希函数,重路由遭遇拥塞的流	感知路径拥塞	流级别	有效减轻哈希碰撞问题	选路具有随机性和被动性
CLOVE	使用 ECN 或者 INT 技术来检测路径的拥塞状况,在软件交换机上改变包头五元组,将流切换到最佳路径	感知全局拥塞	包簇级别	第 1 个虚拟化的数据平面负载均衡器,不用修改物理交换机或者虚拟机	INT 技术通用性不强,而仅仅使用 ECN 来做拥塞检测不够准确
ALB	测量 RTT 和单向延时,为包簇选择拥塞最轻的路径传输	感知全局拥塞	包簇级别	消除了时间不同步问题,准确测量了路径延时	软件交换机的处理时间可能影响延时检测的准确性
Hermes	根据路径状况和流状态做重路由决策	感知全局拥塞	短流以流级别,长流以包级别	适用于非对称网络	重路由决策过于保守
ELAB	主机端依据路径可用带宽比例均衡负载	全局拥塞信息	流级别	无需修改硬件,可动态感知路径拥塞	发生突发流量后,可能导致链路利用率低
MPTCP	在主机对间建立多条子流,每条子流拥有自己独立的拥塞窗口	感知全局拥塞	子流级别	提升了长流的性能	需要修改主机 TCP/IP 协议栈,对短流不友好
FUSO	利用轻拥塞子流的空闲拥塞窗口,快速重传其他子流没有确认的数据包	感知全局拥塞	子流级别	降低了拖尾延时	需要修改主机 TCP/IP 协议栈
DCMPTCP	标识 ToR 内的流量,消除不必要的子流,短流使用一条子流传输数据减轻突发拥塞,ToR 间的子流采用同一拥塞信息,改善拥塞控制的性能	感知全局拥塞	短流是流级别,长流是子流级别	改进了 ToR 内流量和 ToR 间并发短流的性能	需要修改主机协议栈
DC^2-MTCP	属于多路径传输协议,利用网络编码技术减少数据包丢失的影响	感知全局拥塞	子流级别	降低短流的延时,提高长流吞吐量	编码和解码带来系统开销
MMPTCP	初始阶段将数据包随机散射到所有可用路径上,当已发送字节超过阈值,则切换成 MPTCP 的模式	感知全局拥塞	短流以包级别,长流以子流级别	加快了短流的传输,提升了长流的吞吐量	需要修改主机 TCP/IP 协议栈

Table 2 Comprehensive comparison of host-based load balancing mechanisms (Continued)

表 2 基于主机的负载均衡机制的综合对比(续)

文献名称	工作原理	拥塞信息	调度粒度	优势	劣势
Presto	控制器收集拓扑信息,主机上的软件交换机将每个等大小(64KB)的包簇轮询地发往所有可用路径,接收端修改 GRO 机制来减轻乱序	不感知拥塞	固定大小的包簇级别	不需要定制交换机,不需要修改主机协议栈	部署复杂
VMS	利用软件交换机近似实现 MPTCP 的性能.通过在发送端改变五元组,将一条流分散到多条路径,估计路径的可用带宽设置流速	感知全局拥塞	子流级别	无需修改网络协议栈和主机底层协议	两端都需支持虚拟交换机,同时,软件交换机的处理延时大,影响延时敏感流的性能
DumbNet	主机探测和发现可用路径,并将所选的路径信息写入数据包包头,以源路由的方式对数据包进行转发	全局拥塞信息	包簇级别	降低交换机的复杂性,实现网络数据平面无状态化	依赖中央控制器,无法及时感知路径变化
Flicr	通过将流量从直连网络中拥塞的最短路径转移到非最短路径,实现负载均衡	全局拥塞信息	固定大小的包簇级别	适用于非对称网络	需要修改主机内核
DRB	为数据包轮询地选择路径,避免连续的数据包选同一条路径	不感知拥塞	包级别	在对称网络下实现最佳负载均衡效果	非对称网络下乱序严重
JUGGLER	在 GRO 层缓存乱序数据包一段时间再提交,以减少乱序包的影响	不感知拥塞	包级别	大大降低数据包乱序影响	修改网络协议栈
NDP	发送方对所有可用路径随机排序,然后按随机的顺序在每条路径上发送一个数据包,避免了多个发送方同时选择相同的路径传输	不感知拥塞	包级别	实现低延时和高吞吐量	交换机存在切包操作开销
CAPS	采用编码技术对短流编码,恢复乱序数据包	感知路径拥塞	短流以包级别,长流在包级别与流级别间切换	有效缓解了数据包乱序问题,提升了短流的性能	增加网络冗余,长流依旧存在乱序问题
MP-RDMA	采用多路径 ACK-clocking 和乱序感知的路径选择机制,为 RDMA 数据包选择最佳传输路径	感知全局拥塞	包级别	相比于单路径 RDMA,提升了网络利用率	需要改变 RDMA 网卡

3.2.1 基于重路由的主机负载均衡机制

FlowBender[48]是一个基于主机的负载均衡机制.针对 ECMP 的哈希碰撞问题,FlowBender 动态地重路由遭遇拥塞的流.FlowBender 利用 ECN 和 TCP 超时信号检测拥塞和链路故障.类似于 DCTCP 的标记机制,当交换机队列长度超过一个给定阈值时,所经过的数据包的 ECN 字段被标记.若一条流中被标记的包的数量超过给定阈值时,FlowBender 重路由该流.FlowBender 在交换机中配置哈希函数,将 TTL(time to live)的值作为哈希函数的额外输入值.当 FlowBender 检测到拥塞时,在发送主机上修改包头的 TTL 字段,重新计算哈希值来实现重路由.FlowBender 感知路径拥塞,实现对拥塞流的重路由,有效减轻了哈希碰撞的影响.但由于路径选择的随机性和被动性,重新选择的路径难以取得最优效果.

CLOVE[49]是一个部署在软件交换机上的负载均衡算法,软件交换机部署在源端主机的虚拟机管理器上.CLOVE 在物理交换机上使用 ECMP 路由,利用源路由机制发现等价多路径.CLOVE 使用 ECN 或 INT 技术[67−69]来检测路径的拥塞状况,在软件交换机上,通过改变包头五元组将流换到最好的路径上.为了防止换路带来的数据包乱序问题,CLOVE 采用包簇作为调度粒度.CLOVE 是数据中心中,第 1 个虚拟化技术下感知拥塞的数据平面负载均衡器,不用修改物理交换机或虚拟机,但是 INT 技术通用性不强,仅使用 ECN 来检测拥塞也不够准确.

针对 CLOVE-ECN 拥塞检测不准确的问题,ALB[50]提出了一个基于准确拥塞反馈的自适应负载均衡方案.

ALB采用基于延时的拥塞检测,准确地将包簇转发至轻拥塞的路径.ALB的源端主机上部署了虚拟交换机,在虚拟交换机上采用源路由机制实现多路径传输.ALB 利用 DPDK 技术[70]在 TCP 数据包头可选字段写入时间戳,准确地测量每一条流的 RTT 和单向延时,并且解决了源和目的主机时间不同步问题.但是,软件交换机的处理时间可能影响延时检测的准确性.

Hermes[51]采用主动的拥塞检测和谨慎的路由决策以降低数据包乱序影响.Hermes 利用了 ECN、延时和探测包来检测路径拥塞和失效状态,预估切换路径的收益来决定是否切换路径,以缓解盲目切换路径导致的乱序影响,避免链路状态和拥塞窗口不匹配的问题.Hermes 主要由感知模块和重路由模块两个模块组成,如图 3 所示.感知模块利用 RTT 和 ECN 两种拥塞信号感知网络拥塞和链路故障.当 RTT 和 ECN 信号都显示拥塞轻时,该路径被认为是非拥塞路径;当检测的 RTT 和 ECN 信号都显示重拥塞时,该路径被认为是拥塞路径;其他情况,该路径被认为是灰色路径.另外,当 Hermes 检测到某条路径上的流超时事件超过 3 次或者收不到任何 ACK 包时,该路径被认为是故障路径,重路由时会忽略此类路径.重路由模块采用包级别的调度粒度以及时反应网络拥塞.为了防止拥塞不匹配和数据包乱序问题,Hermes根据路径状态和流状态来做重路由决策.只有新流出现或流遭遇超时,或者当前路径被认定为拥塞路径时,Hermes 才触发重路由机制.而且,Hermes 只重路由低于一定发送速率的长流,同时要保证切换的路径是有收益的.Hermes 适用于非对称网络,但是重路由的决策过于保守,易导致链路利用率低.

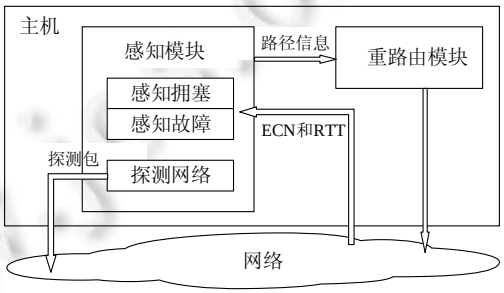


Fig.3 Schematic diagram of Hermes

图 3 Hermes 原理图

ELAB[52]是一种基于主机端的负载均衡机制,不但易于部署,而且能动态感知网络拥塞.ELAB 部署在主机系统协议栈的网络层和传输层之间,首先将数据包进行封装,加入隧道包头;然后,通过改变隧道包头的五元组在主机上实现主动路由.ELAB在主机进行第1次通信时探测路径状态,再以较长的周期反复探测以感知网络拓扑的变化.ELAB 主机端维护路径上限带宽、实际发送速率和可用带宽这 3 个状态.其中,上限带宽初始化为链路物理带宽,然后依据是否收到 ECN 标记包,以实际发送速率来更新上限带宽.在收到 ECN 标记包的一段时间后,主机端在拥塞路径上发送一组探测包以重新探测上限带宽.实际发送速率由接收端的数据包接收速率决定.可用带宽为上限带宽和实际发送速率的差值.最后,ELAB 主机端按照可用带宽比例,以比例轮询的方式在各条路径上分配流量.ELAB 机制无需修改交换机,实现了路径拥塞状态敏感的负载均衡.

3.2.2 流量切分的主机负载均衡机制

MPTCP[53]是一种基于主机的多路径传输控制协议,在发送主机上,将原本的单条数据流划分成多条并行的子流在不同路径上传输.MPTCP 在发送端和接收端之间建立多条子流,各子流拥有独立的序号空间和拥塞窗口,执行类似 TCP 的加性增窗乘性减窗策略,可自适应地将流量从拥塞路径上转移到非拥塞路径上,从而实现多路径的负载均衡.尽管 MPTCP 提升了长流的性能,但在丢包情况下,它严重加剧了延时敏感短流的拖尾时间.当路径发生拥塞的时候,短流在拥塞路径上的子流很容易发生全窗丢失而触发超时事件.而 MPTCP 的子流仅仅处理自身的丢包事件,无法快速解决超时问题,导致长的拖尾延时,从而影响整体的流完成时间.

FUSO[54]改进了 MPTCP 协议,能更快速地恢复丢包.FUSO 利用轻拥塞子流的空闲拥塞窗口快速重传其他子流没有确认的数据包,避免了拖尾子流的影响,减少流传输的总体完成时间.当发送端发现应用层没有新的数

据包发送,同时传输层拥有空闲拥塞窗口的时候,拥塞最轻的子流会重传拥塞最重子流的未确认的包.此外,MPTCP 的接收端有一个共享缓冲区,每条子流拥有一个映射到该共享缓冲区的虚拟接收缓冲区.FUSO 的接收端直接将恢复的包放在共享缓冲区,加快数据的提交.

MPTCP 使用子流均衡网络中的流量,每条子流维护独立的拥塞窗口,最大化每条路径的吞吐量,提升了数据中心中数据传输性能.但 MPTCP 不适用于 ToR 内的流和多对一模式下的短流.一方面,同一 ToR 内的主机对之间只有一条路径,多条子流同时发送到一条路径上会增加链路拥塞;另一方面,增加多对一短流的子流数会加重网络拥塞,短流的子流过小又容易遭遇丢包超时.DCMPTCP[55]方案是对 MPTCP 的改进:首先,使用 TCP 协议控制 ToR 内流量,消除不必要的子流减少开销;然后,DCMPTCP 依据已发送字节数评估流大小.短流用一条子流传输所有的数据以减轻多条子流造成的突发丢包问题.对于 ToR 间的流量,当一条路径上发生拥塞时,发送端的所有子流共享拥塞信息并减小拥塞窗口.

为了充分利用数据中心网络的链路带宽,多路径 TCP 将一条流切分为多条子流,在多条并行路径上传输.但是:当短流被切分为多条子流时,每条子流的拥塞窗口过小,在遭遇丢包时易发生超时,严重影响短流的延时;而当长流被切分为多条子流时,拥塞路径上的子流也容易阻塞整体传输.DC^2-MTCP[56]作为一个多路径传输协议,利用网络编码技术来快速恢复丢失和被阻塞的数据包.发送端首先通过监测数据包的丢失事件,推测路径质量,以选择合适的包进行编码:若路径间质量差异小,使用前向纠错编码技术对总发送窗口中最后若干个未被确认的包进行编码,以加快数据包恢复;若路径间质量差异大,选择总发送窗口中最先若干个未被确认的包进行后向纠错编码,以减少编码开销.然后,将编码包分配给不同的子流.DC^2-MTCP 依据路径的质量调整长流的编码冗余度:轻拥塞的子流主动传输更多的冗余包,以恢复重拥塞子流的丢包;在高质量路径上传输更多的编码包,以均衡冗余负载.对于无法准确感知路径质量的短流,通过历史统计信息来调整编码冗余度.DC^2-MTCP 降低了短流的延时,同时提高了长流的吞吐量.

针对数据中心长流和短流各自的性能需求[71],MMPTCP[57]采用包分散和子流传输相结合的方式,提高网络链路利用率.在初始阶段,使用随机包分散方法快速利用所有可用路径资源,有效降低了延时敏感短流的完成时间;当已发送字节大于一定阈值时,采用 MPTCP 协议来传输,MMPTCP 的长流被分割成多条子流,保证了较高的吞吐率.包分散的方法易带来数据包乱序问题,可通过动态调节快速重传门限值来防止由数据包乱序引起的虚假重传.MMPTCP 根据拓扑信息调节快速重传门限值,当源端发送的流量需要通过核心层交换机时,采用较高的快速重传门限;当流量仅在同一 ToR 交换机内传输时,采用较低的快速重传门限.MMPTCP 使用胖树寻址方案作为设置快速重传门限的基础,每个源端通过检查源 IP 地址和目的 IP 地址来推断其流量将穿越的网络拓扑层.当主机对间的流量在同一 ToR 交换机下传输时,采用默认的快速重传阈值 3;当主机对间的流量必须经过汇聚层和核心层交换机时,则使用较高的阈值.

在虚拟化多租户数据中心,客户通常会部署自定义虚拟机.各虚拟机管理器(如 hypervisor)下的虚拟交换机可利用多条等价路径实现负载均衡,而无需物理网络中的特殊功能或修改客户虚拟机 TCP/IP 协议栈,有利于移植与扩展.

Presto[58]使用软件交换机将流切分成 64KB 的数据切片.Presto 利用中央控制器收集拓扑信息,并将信息转发给在主机上的软件交换机.控制器首先将网络分割成一组生成树,每个软件交换机对应一个生成树.然后为每一个生成树中的每一个软件交换机分配一个唯一的转发标签,并安装相关的转发规则.为了实现有效的负载均衡,软件交换机以轮询的方式访问影子 MAC 地址,使得数据切片均匀地分布在网络中.虽然小于 64KB 的流不会发生乱序,但大于 64KB 的流仍然可能乱序.Presto 修改的 GRO(generic receive offload)算法区分了丢包和乱序.当发生乱序时,因为乱序的包还在网络中,修改的 GRO 不直接提交原来的段,而是等待乱序包到达后再提交给上层.由于数据切片内不会发生乱序,当属于同一个数据切片中的包出现空缺时,Presto 认为这是丢包引起.此时,GRO 立即向上层提交数据段.当流切片的边界出现空缺,无法判断是丢包还是乱序时,Presto 则依据是否超时判断丢包.

虚拟多信道散射 VMS[59]方案是一个基于虚拟交换机的负载均衡机制.VMS 通过在发送端改变五元组,将

一条流的包分散到多条转发路径上;接收端接收到数据包后,改回原来的五元组,并重新排序数据包以避免乱序.VMS 估算每一条转发路径的可用带宽,用虚拟窗口大小表示,在 TCP 包头中设置窗口大小,调节流传输速率.VMS 根据不同路径的虚拟窗口大小,自适应地为数据包选择转发路径.VMS 在虚拟机交换机上部署,不需要修改网络协议栈和主机底层协议,可近似实现 MPTCP 的性能.但是 VMS 两端都需支持虚拟交换机,同时,软件交换机的处理延时大,影响延时敏感流的性能.

为了简化数据平面交换机的操作,DumbNet[60]在主机上实现拓扑发现、网络路由和故障处理.主机使用源路由对数据包进行转发,数据包的包头包含一组代表其传输路径的端口号;交换机仅依据数据包携带的端口号转发数据包,完全实现了网络数据平面无状态化.DumbNet 使用基于主机的机制来收集路径信息,按照广度优先搜索算法发送探测包到其他主机来发现拓扑结构.然后,主机将路径信息存储在中央控制器中,由中央控制器维护全局拓扑信息和优化转发路由.同时,主机从中央控制器获得所有的可用路径,并存储在主机缓存中,以避免由链路故障引起的通信中断问题.最后,主机在数据包的包头写入其传输路径信息.DumbNet 的每台主机维护了流信息,并存储了多条可选路径信息,实现了包簇级别的负载均衡.

Flicr[61]是一个针对直连网络的主机端负载均衡方案.直连网络是指 ToR 交换机之间直接相连的网络拓扑,其包含了多条不同长度的路径.Flicr 利用 ECN 和 TCP 重传超时检测路径拥塞和故障.每条流开始发送时被路由到最短路径,而当发生拥塞时被重新路由.Flicr 以 RTT 为周期做路由决策,并以固定大小的包簇作为重路由的粒度以均衡负载.具体来说,Flicr 使用了交换机上的扩展哈希功能和 VLAN 标签,将交换机上的可用端口分配给最短路径和非最短路径的两个转发表,然后将两组 VLAN 标签值映射到这两个转发表.交换机依据包头的 VLAN 标签值,在对应的路由表中查找转发端口.Flicr 部署在主机端,根据拥塞信号更新数据包包头字段的 VLAN 标签值,以重路由流量.Flicr 考虑了端到端的拥塞信息,可适用拓扑和流量不对称场景.

主动式的流量切分负载均衡可以将任意长度的流切分成多段数据来实现负载均衡,可在一定程度上减少长流堵塞短流的问题,同时提高了网络链路的利用率.但是,流量切分方案容易发生拥塞路径上的拖尾问题.

3.2.3 细粒度调度的主机负载均衡机制

大多数负载均衡方案都以提高网络利用率为目标,没有同时考虑低延时和高吞吐量的需求.细粒度的负载均衡方案较好地平衡两个目标.DRB[62]是一个部署在胖树和 VL2 拓扑的分布式负载均衡机制,以数据包作为调度粒度.DRB 选择交错路径传输数据包,以避免连续的数据包被转发到同一条链路引起的队列堆积问题.例如,在胖树拓扑中,发送主机将第 1 个数据包随机发送至一个汇聚交换机,然后为下一个数据包以轮询的方式选择下一个汇聚交换机,以避免连续的数据包转发至同一个核心交换机.DRB 将数据包均匀地分配到各个路径上,充分利用了网络资源,避免了交换机队列堆积,降低了拖尾延时.

JUGGLER[63]是用来解决数据中心中负载均衡机制下数据包乱序问题的方案.GRO(generic receive offload)合并接收的有序数据包提交给上层,以减少每个数据包的 CPU 处理开销.但数据包乱序会引起很高的 CPU 开销.JUGGLER 在网络堆栈的入口 GRO 层尽可能多地对数据包进行排序.JUGGLER 先缓存活跃流的乱序数据包以等待乱序包达到,再有序地提交给上层.JUGGLER 可解决严重乱序问题,减少 CPU 的开销,但其需要修改网络协议栈.

NDP[64]采用包级别的转发方式,均匀地将流量分布到所有平行路径上传输,加快传输速度,实现了低延时和高吞吐量的目标.NDP 的发送端对所有可用路径随机排序,然后按随机的顺序在每条路径上发送一个数据包,发完一轮数据后重新排序路径,有效避免了多个发送方同时选择相同的路径传输.NDP 采用了剔除数据包负载保留包头的方式来避免数据包丢失.当交换机的队列长度超过一定阈值(比如 8 个数据包)时,数据包的负载被剔除,只保留其数据包头,有效降低了排队延时.数据包头优先正常数据包出队,驱动快速重传,有效降低了重传延时.同时,NDP 是一个接收端驱动的方案.接收端根据接收的数据包驱动响应包(PULL 包),发送端以接收的响应包来发送数据包或重传包,可保证数据包的发送速率与接收端的链路速率匹配,实现了高吞吐率.

为了消除数据包乱序的影响,CAPS[65]提出了基于编码的自适应包分散机制,在传输层和网络层之间增加了数据包编码层,利用编码包恢复乱序的数据包.为了利用多路径和防止短流遭遇线端阻塞,CAPS 对短流的数

据包进行编码,并将其散射到所有路径上.同时,为了提高长流的吞吐量,长流在短流的 ON 阶段用 ECMP 传输;而在短流 OFF 阶段,则使用所有路径以包粒度传输.

CAPS 由 3 个模块组成.

- 第一,编码模块.发送端使用前向纠错编码技术(forward error correction,简称 FEC)[72]对短流的数据包进行编码,每 k 个数据包被编码成 $k+r$ 个编码包,其中产生了 r 个冗余包.冗余包影响解码速率和网络负载开销,因此,CAPS 根据实时网络状况调节冗余包的数量,以实现收益的最大化.
- 第二,包散射模块.为了减轻长短流间的相互影响,CAPS 对长短流采取不同的调度策略:短流使用随机包散射(RPS)方案随机散射到所有路径上;而长流与短流共存时,长流使用 ECMP 路由机制避免乱序,否则使用 RPS 机制.
- 第三,解码模块.接收端在网络层获得任意 k 个编码包后,解码出 k 个原始数据包,提交给上层.

基于编码技术的负载均衡方案可有效地减轻数据包乱序的问题,但是会增加网络中冗余包的数量,加重网络拥塞.

RDMA 因其低时延、高吞吐量和低 CPU 开销的特性被广泛部署使用,但目前的 RDMA 采用单条路径传输机制,无法利用数据中心中丰富的并行路径资源.MP-RDMA[66]方案为 RDMA 提供了一种多路径传输机制,采用了 3 种新的技术解决 RDMA 网卡芯片内存容量有限的问题.

(1) 采取一个多路径的 ACK-clocking 机制来分配流量.具体来说,发送方依据接收的每个 ACK 所携带的 ECN 标志位来调整拥塞窗口:若 ECN 标志位为 0,则拥塞窗口增加 1;若 ECN 标志位为 1,则拥塞窗口减半.

(2) 使用一个乱序感知的路径选择机制,主动剔除慢路径并自适应地选择一组快速且延迟相似的路径.

(3) 利用同步机制,以确保在需要时按顺序更新内存.

与单路径 RDMA 相比,MP-RDMA 仅为每个连接状态增加 66B 额外内存大小,可以显著提高异常情况下的健壮性,并改善整体网络利用率.

3.3 基于交换机的负载均衡机制的研究

集中式负载均衡方案部署开销大,反馈和控制延时长;基于主机的负载均衡方案则需要修改主机协议栈.因此,目前主流的数据中心网络负载均衡策略是通过在交换机上快速感知网络拥塞,采用不同的调度粒度将网络流量发送到不同的路径上.近年来,学术界提出了多种基于交换机的方案,见表 3,它们的解决思路可按调度粒度划分为:① WCMP[73]、Expeditus[74]、TinyFlow[75]、DiFS[76]、DFFR[77]、Beamer[78]、iLoad[79]、Al-Tarazi[80]、Dart[81]等流级别的调度方法;② Detail[82]、RPS[83]、DRILL[84]、GRR[85]、QDAPS[86]、RMC[87]、OPER[88]等包级别的调度方法;③ Flare[89]、CONGA[90]、HULA[91]、Multi-hop CONGA[92]、LetFlow[93]、Luopan[94]和 MLAB[95]等包簇级别的调度方法;④ AG[96]、TLB[97]等自适应粒度的调度方法.

Table 3 Comprehensive comparison of switch-based load balancing mechanisms

表 3 基于交换机的负载均衡机制的综合对比

文献名称	工作原理	拥塞信息	调度粒度	优势	劣势
WCMP	根据链路容量分配路径权重,根据权重公平地将流量哈希到每一条路径	感知全局信息	流级别	可有效缓解链路故障问题	容易发生哈希碰撞,切换不灵活
Expeditus	交换机监测本地链路负载,采用两阶段路径选择机制汇总跨层交换机的拥塞信息,为每流做路径选择	全局拥塞信息	流级别	可实时监测3层网络架构的拥塞状态	需要硬件支持
TinyFlow	将长流切分成多条 10KB 的短流,通过动态地改变出端口实现短流的 ECMP 路由	局部拥塞信息	流级别	解决长流线端阻塞和哈希碰撞问题	易带来长流乱序问题

Table 3 Comprehensive comparison of switch-based load balancing mechanisms (Continued 1)

表 3 基于交换机的负载均衡机制的综合对比(续 1)

文献名称	工作原理	拥塞信息	调度粒度	优势	劣势
DiFS	交换机将所有流量均匀地分配在所有路径上,避免本地流量冲突,依据远端交换机发送的显式适应请求更改转发路径,以避免远端拥塞	本地拥塞信息和全局拥塞	流级别	具有很好的扩展性	收敛时间长
DFFR	维护每个交换机上每个出端口的流量信息,所有交换机将流量均匀分配给到同一目的 ToR 交换机的所有路径	局部拥塞信息	流级别	理想情况下可实现网络利用率最大化	难以适用于非对称网络
Beamer	利用已经存储在后台服务器中的连接状态,确保连接不会丢失.当服务器接收到一个没有状态的中间连接数据包时,会将其转发到另一台保持了该数据包状态的服务器	全局拥塞信息	流级别	能处理负载均衡器和服务器的变化问题	服务器的存储开销大
iLoad	利用可编程交换机,依据服务器的可用能源为服务器分配负载	可用能源信息	流级别	充分利用各个服务器的能源	随着服务器数量增加,需要增加交换机存储内存
Al-Tarazi 等人[80]	使用最少的交换机和链路转发流量以节省能源,并随机选择交换机和链路以均衡流量	全局拥塞信息	流级别	在节能的同时,实现了链路的负载均衡	交换机和端口的控制复杂
Dart	Dart 将拥塞分为接收端拥塞和网内拥塞.对于接收端拥塞,直接为发送方分配速率;对于网内拥塞,采用按序流偏转机制为短流重新路由	局部拥塞信息	流级别	收敛速度快	需要硬件支持
Detail	根据出端口队列占用信息,动态地为每个数据包选择拥塞最轻的下一跳	本地队列信息	包级别	有效解决拖尾问题	跨层处理增加部署的复杂性
RPS	为每一个数据包随机地选择路径	不感知拥塞	包级别	充分利用所有路径资源	容易发生数据包乱序问题
DRILL	每个数据包根据本地队列长度选择转发端口	本地队列信息	包级别	快速响应拥塞,实现微秒级的负载均衡	容易发生数据包乱序问题
GRR	GRR 为每个主机建立从服务器到一个主干交换机端口的路由路径,利用直通转发交换机转发数据包	全局拥塞信息	包级别	实现网络满利用率	难以适用于大规模的交换矩阵
QDAPS	根据同一条流的上一个数据包的剩余排队时间,为该流当前数据包选择合适的出端口队列	本地队列信息或者全局延时	包级别	使数据包有序到达接收端	交换机计算开销大
RMC	显式反馈包乱序,并利用编码技术减少拖尾时间	全局拥塞信息	包级别	避免不必要的快速重传,减小流完成时间	大规模拓扑存储开销大
OPER	在包级别负载均衡的基础上实现了可替换编码机制,保证传输性能	基于丢包/乱序率的全局拥塞信息	包级别	降低了乱序影响,同时保证流量负载均衡	需要收发两端支持
FLARE	两个包簇间的时间间隔大于路径延时的最大差异时,重新换路	全局延时信息	包簇级别	没有乱序问题	交换机计算开销大
CONGA	根据实时网络拥塞情况,为包簇分配最佳路径	全局拥塞信息	包簇级别	依据全局信息使得流量分配均匀,适用于非对称网络	反馈延时过大,且需要定制交换机,可扩展性差
HULA	转发包簇至最佳下一跳	下一跳的拥塞信息	包簇级别	只需维护最佳下一跳的拥塞信息,转发表的开销小	易发生羊群效应,导致最佳下一跳拥塞

Table 3 Comprehensive comparison of switch-based load balancing mechanisms (Continued 2)

表 3 基于交换机的负载均衡机制的综合对比(续 2)

文献名称	工作原理	拥塞信息	调度粒度	优势	劣势
Multi-hop CONGA	在单跳路径发生拥塞时,使用两跳路径上的空闲链路带宽提升性能	全局拥塞信息	包簇级别	适用于非对称网络	路径探测开销大
LetFlow	采用固定的包簇时间间隔,随机地为每个包簇选择转发路径	感知路径拥塞	包簇级别	可适用于非对称网络	选路具有随机性
Luopan	采样部分路径的拥塞情况,将固定大小的包簇转发至最轻拥塞的路径	感知路径拥塞	固定大小包簇级别	降低了拥塞信息探测和存储的开销,可适用非对称网络	只根据部分路径的拥塞情况,不能保证最优性能
MLAB	将高层网络分割成多个域,域间路由算法独立.域外 ToR 交换机只需获取各个域的出口负载信息,以降低探测开销	全局拥塞信息	包簇级别	适用于高层网络拓扑的负载均衡,便于部署和模块化升级	简化了远端的负载信息,无法得到全局最优的路径
AG	依据拓扑的不对称程度调节路径切换的粒度	全局拥塞信息	自适应粒度	降低了乱序的影响,同时保证了高的链路利用率	复杂拓扑的测量开销大
TLB	依据短流强度调节长流路径切换的粒度	本地拥塞信息	自适应粒度	依据短流强度调节长流路径切换的粒度	在非对称拓扑下会有乱序

3.3.1 流级别的交换机负载均衡

由于 ECMP 不感知路径的拥塞状态和流量特征,容易造成多条数据流在路径上发生哈希冲突.为了避免路径流量的不均衡,加权等价多路径路由 WCMP[73]利用中央控制器得到的网络拓扑和流量信息,根据路径的带宽容量为每条路径分配权重,以加权公平方法将流量哈希到不同的路径.WCMP 可以改善 ECMP 中哈希冲突的问题,同时适用不对称网络.但是作为一种流粒度的负载均衡方法,WCMP 不够灵活,不能依据网络拥塞信息对流进行重路由.

拥塞感知负载均衡方案要求知道源目的端之间所有路径的实时拥塞信息,通常由 ToR 交换机维护从其自身到其他 ToR 交换机之间所有路径的端到端拥塞状态信息.虽然可以通过数据包携带拥塞状态,但这种信息反馈方法难以在复杂的 3 层 CLOS 拓扑中使用.Expeditus[74]方案是一个适用于 3 层网络架构的分布式拥塞感知的负载均衡协议.Expeditus 首先通过监测本地出端口和进端口的链路负载,确保拥塞状态的实时性.然后,Expeditus 使用两阶段路径选择机制来汇总跨交换机的拥塞信息并做负载均衡决策.

- 第 1 阶段,只有源和目的 ToR 交换机选择从 ToR 交换机到汇聚层的最好的路径.源 ToR 交换机发送其出端口的拥塞信息给目的 ToR 交换机,目的 ToR 交换机将接收的拥塞信息与本地进端口拥塞信息相结合,选出到汇聚层的最佳路径.
- 第 2 阶段,被选择的汇聚层交换机以同样的方式根据 2 层和 3 层的拥塞信息选择最佳的核心层交换机.最终,ToR 和汇聚层交换机保存路径选择的结果.

Expeditus 为每条流在其 TCP 三次握手期间做路径选择,可防止数据包乱序问题,同时也不会引入延时开销.

针对 ECMP 不区分长短流带来的线端阻塞和拖尾延时的问题,以及哈希碰撞带来的带宽利用不足的问题,TinyFlow[75]将长流分割成多条 10KB 的短流,让所有的短流以 ECMP 的路由方式均匀分布在所有路径上.TinyFlow 主要包括两个部分:长流检测和动态随机路由.在 TinyFlow 中,ToR 交换机通过采样与主机相连的端口的流量检测流信息.TinyFlow 利用 OpenFlow 实现动态随机路由.当监测到长流时,交换机随机选择一个不同的出端口传输该流,同时修改流表、重置字节数.TinyFlow 可降低短流的完成时间,同时提高长流的吞吐率.但是,TinyFlow 易造成长流乱序的问题.

DiFS[76]方案将数据中心中的流量冲突分为本地冲突和远端冲突.在交换机一条出口链路上遭遇的并且可以通过本地调整传输路径解决的冲突,称为本地冲突;从多个核心交换机经过同一汇聚交换机到同一 ToR 交换机引起的流量冲突,称为远端冲突.DiFS 的每个交换机采用分布式贪婪路径分配算法将流量均匀分配到所有出口链路,以避免本地流量冲突.同时,每个交换机运行不均衡检测算法监测进口链路的流量.如果检测到流量冲突,交换机将发送显式适应请求(EAR)消息给流的发送交换机,建议其更改转发路径.收到 EAR 消息后,发送交换

机运行显式适应算法,避免远端流量冲突.DiFS 作为分布式流量调度算法,可扩展性强,但也存在路径振荡问题,需要较长的收敛时间.

DFFR[77]方案是一种分布式自适应负载均衡算法,从理论上最大化汇聚层网络利用率.已存在的分布式算法利用重路由流量来达到负载均衡的效果,这会引起路径振荡问题或者 TCP 乱序问题.DFFR 通过维护每个交换机上每个出端口的流量信息,不采用重路由和流量分割,而是所有交换机将流量均匀分配给到同一目的 ToR 交换机的所有路径,从而使得网络达到最佳平衡,最大程度地利用所有带宽.DFFR 在理论上和理想的情况下可以实现最优平衡,但其分布式的方案难以适应高动态和强突发的数据中心流量.同时在不对称网络下,难以只根据本地信息实现理想的负载均衡.

数据中心负载均衡器(MUX)通过一组动态后台服务器实现了请求的分发,均衡多个服务器之间的请求分布,避免单个服务器出现过载.现在,负载均衡方法在 MUX 上保存了每个连接所选服务器的状态信息,保证每个连接的数据都发送到同一个服务器,但是存在 MUX 与服务器之间的状态不匹配问题.当 MUX 和服务器数量同时发生变化时,会使连接中断.当请求连接数量很多时,负载均衡器的吞吐量下降.Beamer[78]不需要在 MUX 中保存每条流的状态,而是利用存储在后台服务器中的连接状态确保连接不会丢失.当服务器接收到一个无连接状态的数据包时,会将其转发到另一台保持该数据包所在连接状态的服务器.Beamer 实现了稳定哈希.连接先被哈希映射到一组固定的桶上,每个桶能映射到任何一个服务器.当服务器池发生变化时,Beamer 利用控制器更新并保存服务器和桶之间的新映射,MUX 再把流量分配到新服务器,保证了不中断、平滑的流量迁移.Beamer 能高效处理增加、移除 MUX 和服务器的情况.

iLoad[79]是一种基于可编程数据平面的绿色数据中心负载均衡体系结构,其中部署了 iLoad 的服务器向可编程交换机发送能源可用性信息.交换机根据该信息构建哈希表,以均衡服务器的工作负载与可用能源,使得可用能源多的服务器能承担更多的工作负载.

数据中心现有的大多数节能方案侧重于最大程度地降低能耗,而忽视了网络传输性能.Al-Tarazi 等人[80]首先证明了数据中心的最小化能耗问题是一个混合整数线性规划问题,提出了启发式节能算法,然后通过负载均衡机制提升网络传输性能.该节能算法尽可能地关闭所有空闲的端口和交换机,最小化活跃交换机和链路的数量,以最大程度地降低能耗.该方案同时使用了负载均衡机制在活跃链路上分配负载.该机制将流量负载分配到一个随机选择的活跃交换机,同时确保该交换机出口链路的可用带宽能满足流量负载的带宽需求,又不会造成该链路的利用率过载.该机制可有效缓解延时、丢包和链路拥塞问题.

在基于 RDMA 的数据中心中,现有的大部分拥塞控制机制需要通过多次迭代才能调节速率至目标速率,收敛速度较慢.Dart[81]方案将拥塞分为接收端拥塞和网内拥塞,以分治的思想快速缓解拥塞.其中,接收端拥塞的产生原因是同一接收端的并发流汇聚于连接接收端的最后一跳链路,网内拥塞是由不同接收端的流在网络中其他链路上发生碰撞而产生.对于接收端的拥塞,接收端通过显式分配速率的方式快速调节发送方的发送速率;对于网内拥塞,Dart 在硬件交换机部署按序流偏转机制对流进行重路由,以快速响应拥塞.当某条流遇到拥塞时,交换机为该流选取其他替代路径,以避开拥塞链路.为了避免乱序和减少 CPU 开销,Dart 只对部分短流进行路径偏转.对于发生网内拥塞的长流,则采用 DCQCN[98]机制缓解拥塞.

流级别的负载均衡方法将同一条流的数据包在相同路径上传输,能有效地避免乱序问题.但在高动态和强突发的网络流量下,静态负载均衡方法不但容易发生拥塞冲突导致拖尾,而且无法灵活切换路径,难以充分利用网络带宽.

3.3.2 包级别的交换机负载均衡

为了充分利用网络带宽,许多负载均衡方案以数据包为调度粒度.Detail[82]是一个跨层的方案,如图 4 所示,通过快速检测下层的拥塞状态,驱动上层的路由决策,以减少拖尾流完成时间.Detail 利用链路层信息减少丢包,在网络层执行包级别的负载均衡以均匀流量来减少拖尾程度,在传输层关闭 TCP 协议的快速恢复和重传机制来抵抗包乱序带来的虚假拥塞通知,在应用层提升延时敏感流的优先级以保证其性能.

具体来说,在链路层,Detail 使用优先级流控机制(PFC)[99]构建了一个无损结构.交换机监控入口队列占用情

况,当队列长度超过一定阈值时,交换机发送暂停消息(pause)给上一跳,要求其停止发送数据包;当队列长度减小,交换机发送重启消息(unpause)给上一跳,要求其恢复数据包传输.通过这种快速响应,无损结构可确保数据包不会因拥塞而丢失.在网络层,Detail 根据出端口队列占用信息,动态地为每个数据包选择拥塞最轻的下一跳.由于链路层的无损结构避免了网络拥塞丢包,Detail 关闭 TCP 协议的快速恢复和重传机制来避免数据包乱序触发的虚假重传.在应用层设置延时敏感流和延时不敏感流的优先级,确保高优先级的延时敏感流先被转发.

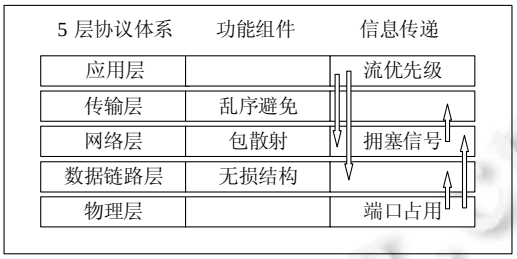


Fig.4 Schematic diagram of Detail

图 4 Detail 原理图

相对于复杂的跨层方法,RPS[83]方案实现了一个简单的包级别负载均衡,在交换机上随机地将数据包发送到所有的可用发送路径,提高了链路利用率.RPS 不要求修改主机,得到了很多商用交换机的支持.RPS 可以提高网络利用率,保证负载均衡,但是它无法适应非对称拓扑,易遭受数据包乱序问题.由于 TCP 无法区分乱序包和丢失的包,乱序包会导致拥塞窗口降窗,从而降低 TCP 的性能.为了解决非对称问题,RPS 采用了队列管理机制限制队列长度,减小队列间的延时差异,以减轻数据包乱序程度.

为了更快速地缓解突发流量引起的拥塞,DRILL[84]提出了一个针对微突发流量的负载均衡策略.由于基于全局流量信息的负载均衡系统的控制周期很长,可能无法检测和缓解生命周期短暂而又容易引起丢包的微突发流量.DRILL 仅利用本地信息快速做决策,通过比较当前随机选择的两个端口和上一轮最好端口的队列长度,选择最小队列长度的端口作为当前数据包的转发端口.DRILL 可以及时地解决微突发流量带来的拥塞问题,但是只依据本地交换机缓存队列长度判断路径状态,不能准确感知全局的拥塞信息,无法避免乱序问题.为了解决拓扑不对称带来的乱序问题,DRILL 将不对称的网络拓扑分割成多个对称的子拓扑,在每个子拓扑里执行基于本地队列的负载均衡决策.同时,DRILL 根据每个子拓扑的路径容量为其分配权重,按权重将流量负载分配到各个子拓扑,以均衡流量负载.DRILL 根据交换机本地队列长度快速地执行数据包级别的转发决策,可实现微秒级别的负载均衡,有效地解决了微突发流量带来的拥塞问题.

全局轮询 GRR[85]方案是一个适用于胖树拓扑的包级别负载均衡路由算法.假设每个时隙容纳一个数据包.在每个时隙中,GRR 为每个主机选择从服务器到一个主干交换机端口的路由路径,且任何路径都不会相互交叉.GRR 使用了一种快速转发且没有缓存队列的直通转发交换机.数据包通过直通转发交换机从源服务器发送到主干交换机.每个时隙内,路由路径以轮询的方式更新,实现了每包的负载均衡.GRR 方案可以保证 100%的吞吐量,但是它依赖于直通转发交换机.当交换机的端口变多,交换矩阵变得复杂时,难以实现理想的直通交换.

包级别的负载均衡方案可以充分利用多路径资源,提高网络的利用率.但在非对称拓扑下,包级别的方案的鲁棒性不强,极易发生乱序问题,导致 TCP 性能下降.QDAPS[86]方案设计了一种排队延时敏感的包粒度负载均衡方案,以对抗数据包乱序问题.QDAPS 在交换机上根据同一条流的上一个数据包的剩余排队时间为该流当前数据包选择合适的出端口队列,使数据包能够有序到达接收端.

QDAPS 由 3 个模块组成.

- 第一,估计队列延时.当交换机接收了一个新包时,QDAPS 根据该包所在队列的实时队列长度估计其排队延时.QDAPS 只需记录每条流最新到达包的排队延时.
- 第二,解决包乱序.QDAPS 为流的第 1 个数据包选择最短队列,然后在后续到达的数据包选择出端口时

保证了该出端口的排队延时大于该流上一个数据包的剩余排队时间,使得该流的数据包按序转发.

- 第三,重路由长流.由于QDPAS的每个数据包总是排在该流上一个数据包的后面,长流易经历较长的排队延时.当队列长度超过某一阈值时,QDAPS 重新选择最短出端口队列,防止出端口队列堆积的问题.重路由长流一方面降低长流的排队延时,另一方面增加乱序降窗的概率,因此,QDAPS选取了合适的重路由阈值以权衡排队延时收益和乱序开销.

QDAPS 作为包级别的负载均衡方案,增加了调度的灵活度,能快速地提升链路利用率;同时,根据交换机队列延时信息选择合适的出端口队列,有效地避免了乱序问题.但是 QDAPS 要为每条流最新到达包记录排队延时,受交换机有限的状态内存的限制.另外,QDAPS 需要为每个数据包计算非乱序路径,无法忽略大规模流量下的计算开销.

在数据中心的非对称网络拓扑下,负载均衡机制容易出现乱序问题.由于缺乏显式的乱序反馈,当发送端收到重复ACK的数量大于预先设定的阈值时,发送端误认为网络出现拥塞从而触发快速重传.这些不必要的快速重传势必降低链路利用率,增加流的完成时间.RMC[87]方案显式反馈包乱序,并采用编码技术来减少拖尾时间.交换机根据本地队列长度和全局路径延迟主动地标识乱序包,避免不必要的快速重传.发送端依据乱序包的占比计算冗余编码包的数量,减少拖尾时间.

数据包级别的细粒度负载均衡机制可以充分利用多条路径,从而实现高链路带宽利用率和均衡流量.但是网络拓扑不对称时的包丢失问题导致了较大的流完成时间,造成了传输性能的下降.虽然基于网络编码的解决方案可以有效地解决乱序问题,但也会引入过多的冗余编码包,带来过多的流量开销,导致较长的排队延迟甚至丢包.为了解决这个问题,OPER[88]提出了一种具有自适应能力的冗余编码包替换机制.在严重拥塞时,OPER 在交换机缓存中用新到达的数据包替换冗余编码包,来避免冗余包产生的额外排队延迟;否则,OPER 不替换或者少量替换冗余编码包,保证乱序情况下的健壮性.OPER 极大地减少了网络拥塞时编码包导致的排队延迟,提升了负载均衡机制的传输性能.

3.3.3 包簇级别的交换机负载均衡

流级别的负载均衡方案可有效避免数据包乱序,但易发生拥塞碰撞且无法充分利用多路径资源.包级别的负载均衡方案充分利用了多路径资源,但极易发生乱序问题.包簇级别负载均衡的切换粒度介于以上两类之间,可有效保证链路利用率和降低数据包乱序程度.包簇一般是由包间时间间隔区分的,当包间时间间隔大于某个值时,则产生一个新的包簇,由交换机来重新选择路由.

FLARE[89]为了避免乱序问题,仅当两个连续数据包间的时间间隔大于路径延时的最大差异时,才将下一包簇发送到其他的路径上.FLARE 实时测量所有路径的延时,保证流量切割的动态性和准确性,但同时也带来了很大的测量和计算开销.

CONGA[90]是一种拥塞感知的负载均衡方案,采用包簇作为调度粒度,如图5所示.

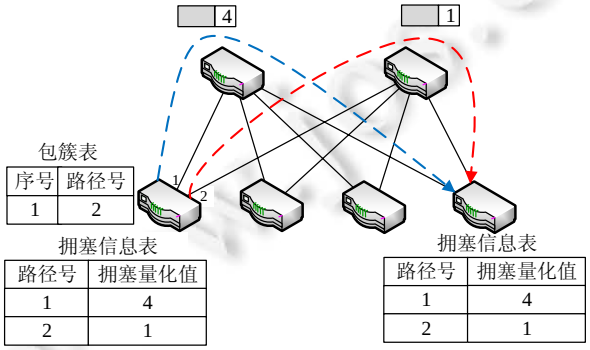


Fig.5 Schematic diagram of CONGA

图5 CONGA原理图

CONGA 利用 DRE(discounting rate estimator)技术测量和量化链路的拥塞程度.用 VXLAN[100]封装的数据包在网络传输过程中不断地依据路径上交换机的最重拥塞信息来更新包头携带的路径拥塞量化值.源和目的 ToR 交换机各自维护一张拥塞信息表,其中:目的 ToR 交换机解析正向数据包包头中的路径拥塞量化值,并记录在拥塞信息表中;由反向传输的数据包将各条路径的拥塞信息携带给源 ToR 交换机.源 ToR 交换机接收到反向数据包后,更新其拥塞信息表.此外,源 ToR 交换机还维护了一张包簇表,记录了活跃包簇的序号和所选的路径.源 ToR 交换机根据拥塞信息表和包簇表为每个包簇选择最轻拥塞路径.CONGA 基于端到端的路径状态反馈信息实现了全局拥塞感知的负载均衡,能够感知路径的拥塞和失效,可适用于非对称网络.但是 CONGA 需要存储大量路径信息,并使用定制的交换机,使得 CONGA 难以大规模部署;另外,从远端交换机得到的反馈可能无法准确反映实时路径状态.

HULA[91]采用可编程交换机实现拥塞感知的负载均衡,克服了 CONGA 的扩展性问题.一方面,HULA 交换机仅通过邻居交换机维护到目的交换机的最佳下一跳路径,无需记录所有路径的链路利用率,从而降低了维护路径状态的开销;另一方面,HULA 利用 P4 语言实现了在可编程交换机上运行的原型,不需要定制的交换机硬件,便于修改和扩展.具体地,HULA 定期发送探测包到所有可用路径,以收集全局链路利用率信息.基于探测包反馈信息,每个交换机选择路径利用率最小的下一跳路径,并将其通告给所有邻居节点.同时,每个交换机维护一张拥塞信息表用来存储到目的地的最佳下一跳路径,有效消除了路径爆炸对交换机的存储压力.另外,HULA 使用探测包主动获得路径拥塞信息,感知网络拓扑的变化.探测包由 ToR 交换机生成,它们经过的每个交换机会更新拥塞信息表.为了避免乱序,HULA 也选择包簇作为调度粒度,为每一个包簇选择最佳路径.HULA 虽然解决了 CONGA 的可扩展性问题,但由于 HULA 只选择最佳下一跳路径,容易发生羊群效应,导致在最佳路径上发生拥塞.同时,最佳路径的更新速度取决于探测频率,而频繁使用探测包会降低网络的有效利用率.

CONGA 方案主要均衡 Leaf-Spine 拓扑结构中单跳路径上的网络流量.单跳路径是指数据包首先从与发送端相连的 ToR 交换机上行链路向核心交换机传输,然后从核心交换机的下行链路向与接收端连接的 ToR 交换机传输.但当路径中的某条链路发生拥塞时,发送方误认为整条单跳路径拥塞.当所有单跳路径均发生拥塞时,发送端将认为所有链路都发生了拥塞.而实际上,此时仍然可能存在部分非拥塞链路.Multi-hop CONGA[92]在所有单跳路径负载重时,使用两跳路由路径绕过拥塞链路,通过利用相对较长传输路径上的非拥塞链路来缓解局部流量压力.Multi-hop CONGA 利用数据包携带的延时信息判定路径的拥塞程度:首先,在可选的单跳路径中选择最轻拥塞路径作为传输路径;当所有单跳路径的延时超过一定阈值时,搜索最轻负载的两跳路径作为传输路径;当所有两跳路径都拥塞时,则使用最轻拥塞的单跳路径作为传输路径.为了降低计算的复杂性和提高网络利用率,Multi-hop CONGA 采用包簇作为调度粒度.作为全局拥塞感知的负载均衡机制,Multi-hop CONGA 可解决非对称网络的问题,但是数据包中携带反馈信息过多,容易浪费带宽.

为了完全避免交换机检测路径拥塞的开销,LetFlow[93]利用数据包间的自然属性自动地感知路径拥塞.当流在某条路径上遭遇拥塞时,其数据包间的间隔会增长,自然形成具有时间间隔的包簇.LetFlow 按时间间隔阈值来区分包簇,并将包簇随机发送到其他路径.由于间隔阈值的选取一般大于最大路径差异,LetFlow 可以避免乱序,有效对抗非对称问题.LetFlow 不需要获得全局拥塞情况,相比于 CONGA 等全局拥塞信息感知的方案,可扩展性更好.但由于 LetFlow 调度的随机性,无法取得最优负载均衡性能.

以上依据包间隔来被动地划分包簇的负载均衡方法,难以适应快速变化的网络流量,而且不合理的间隔阈值容易造成链路空闲或过于频繁的路径切换.Luopan[94]是一个基于采样的负载均衡方案.Luopan 采用固定大小的包簇作为调度单元,包簇大小被定为服务器限制的最大 TSO 大小 64KB.针对在源目的交换机之间存在多条等价路径,Luopan 定期采样部分路径,然后将包簇直接转发到最小队列长度的路径上以实现负载均衡.

具体来说,源端交换机定期发送少量探测包到目的交换机,以探测几条随机路径的拥塞情况.探测包包头包括路径编号 PID、包类型 Type 和量化拥塞指标 QCM 这 3 个字段.每一个探测包遍历所在路径所有交换机出端口的队列长度,并将队列信息累计值存储在包头的 QCM 字段.目的端交换机接收到探测包时,立即生成对应 ACK 反馈包,将拥塞信息发送给源端交换机.当源端交换机依据收到的路径拥塞信息更新拥塞信息表,并为每个

包簇选择最佳路径传输.作为一个拥塞感知的方案,Luopan 降低了流完成时间,增加了非对称拓扑环境下的鲁棒性.同时,相比于CONGA 等感知整个网络拥塞的方案,Luopan 还减小了拥塞信息探测和存储的开销,具有更好的扩展性.

MLAB[95]方案是一个模块化负载感知的负载均衡方案.MLAB 首先提出了一个分布式数据驱动的反馈机制,在所有交换机的每个端口部署了一个负载检测器,由经过的数据包反馈给 ToR 交换机以检测远端负载.为了避免乱序问题,采用了包簇作为调度粒度,基于全局负载信息为包簇作路由决策.然后,MLAB 将网络分割为多个路由域,每个域中包含在两个 ToR 交换机之间同一条路径上的所有的汇聚交换机和核心交换机.从 ToR 交换机的角度看,每个域就是一个黑盒,用于传输它们的数据给另一个 ToR 交换机.每个域中的路由算法相互独立,域内的负载信息只能给域中的交换机使用,域外的交换机只能通过该域出端口获得其负载信息.因此,数据包只需携带域内最重的负载信息,减少了数据包的开销,降低了通信存储更新的难度.MLAB 方案便于部署和模块化升级.但其简化了远端的负载信息,每跳选择最轻的出端口不一定得到全局的最轻负载路径.

3.3.4 自适应调度粒度的交换机负载均衡

由于动态流量、链路故障和异构交换设备,数据中心网络容易出现拓扑不对称问题.在不对称的网络拓扑下,流级别和包簇级别的负载均衡方案采用较大的路径切换粒度,可有效防止乱序问题,但网络利用率较低.包级别的负载均衡器采用细的路径切换粒度,可充分利用各条路径,但易产生乱序问题,无法获得最优性能.AG 方案[96]根据拓扑不对称程度自适应调节路径切换的粒度:在拓扑不对称程度高的情况下,AG 增加切换粒度以减轻乱序影响;在拓扑不对称程度低的情况下,AG 降低切换粒度以获取高的链路利用率.AG 利用探测包定期测量交换机间的单向延时以获得准确的路径拥塞状态信息,计算最佳的调度粒度大小,并为每个调度单元随机分配一条传输路径以防止同步问题.

随着延迟敏感型和吞吐率导向型应用程序的流量需求不断增加,如何有效地平衡多路径之间的流量以提升用户的体验和服务的质量,成为大型数据中心网络中一个至关重要的问题.虽然近年在 DCN 中出现了很多负载均衡设计,现有的负载均衡方法并不感知长流和短流混合的流量特性,也没有考虑不同类型流的需求,为不同类型的流都使用相同的粒度切换路径.TLB[97]提出了一种数据中心网络中路径切换粒度自适应的负载均衡方法,交换机根据数据流的已发送数据量区分长流和短流:对于短流,以数据包为粒度选择队列长度最短的出端口转发其新到达的数据包;对于长流,则在满足短流延时截止期限的前提下,根据短流到达强度计算长流切换路径的队列长度阈值.若某条长流在交换机出端口的队列长度大于或等于长流切换路径的队列长度阈值,则选择队列长度最短的出端口转发该长流新到达的数据包.TLB 减小了短流平均完成时间,同时提高长流吞吐率,但在非对称拓扑下存在乱序问题.

3.4 小 结

基于中央控制器的方案通过中央控制器收集和分析全网的路径信息和流量信息,并集中式地选择传输路径.基于主机的负载均衡方案利用主机感知所有转发路径的拥塞信息,显式地控制每条流的路由路径.基于交换机的负载均衡机制依据交换机上的队列长度或者 ToR 交换机之间的拥塞信息,为各调度单元选择不同的出端口.总体上,基于中央控制器的、基于主机的和基于交换机的负载均衡机制各有优点和缺点,见表 4.

Table 4 Comprehensive comparison of different categories of load balancing mechanisms

表 4 不同类型负载均衡机制的综合对比

机制类型	优点	缺点
基于中央控制器的负载均衡机制	(1) 相比于分布式方案,集中式的基于中央控制器的方案拥有全局网络状态信息,并能准确感知链路故障,为数据流分配最佳传输路径,在流量相对稳定的场景下,能取得较好效果 (2) 基于中央控制器的方案将网络设备的控制平面与数据平面分离,实现了网络的灵活控制.交换机只需按照控制器下发的流表执行转发操作,简化了交换机的工作,降低了交换机的开销	(1) 获取和维护全局信息需要一定的部署开销,较大的反馈和控制延时也降低了动态突发流量下的负载均衡性能 (2) 在大规模的数据中心中,基于单个中央控制器的负载均衡方法的计算能力有限,控制器可能成为瓶颈;当采用多个控制器联动处理负载时,多个控制器之间通信是关键,存在同步问题

Table 4 Comprehensive comparison of different categories of load balancing mechanisms (Continued)

表 4 不同类型负载均衡机制的综合对比(续)

机制类型	优点	缺点
基于主机的负载均衡机制	(1) 相比于集中式的调度方案,基于主机的负载均衡方案更具有扩展性.主机具有独立性,各主机可按自身需求修改负载均衡机制并且不受其他节点的影响 (2) 相比于交换机的负载均衡机制,基于主机的负载均衡机制能感知端到端的拥塞状态,做出更加准确的路由决策;基于主机的负载均衡机制可依据全局信息协同传输控制协议实施流量的速率控制和转移	(1) 基于主机的负载均衡的端到端的反馈延时过大,难以适应高动态的突发流量.这类机制需要至少一个往返延时才能感知到突发流量引起的瞬时拥塞,难以避免突发流量带来的丢包问题 (2) 基于主机的负载均衡方案通常需要修改主机的协议栈,或所有主机需同时支持虚拟技术,在数据中心多租户的环境下难以升级部署
基于交换机的负载均衡机制	(1) 基于交换机的负载均衡机制作为一种数据平面方法,它独立于主机的网络堆栈,一旦部署,就立即服务于所有流量 (2) 基于交换机的负载均衡机制能实时感知网络中链路负载情况,可快速地在交换机出端口连接的路径上实现网络流量的均衡	(1) 基于交换机的负载均衡方案难以准确、快速地获取端到端的路径,特别是复杂的 3 层 CLOS 架构中的路径的拥塞状态信息,影响流量转移的准确性 (2) 基于交换机的负载均衡机制通常需要定制的交换机.虽然当前可编程交换机可实现负载均衡算法的原型,在一定程度上减少了成本开销,但目前,可编程交换机的原语无法支持复杂的计算,难以部署复杂的负载均衡算法

4 发展趋势和展望

针对数据中心网络的负载均衡问题,已经有了多种改进方案来缓解路径拥塞、提高利用率和降低传输延时,最终提升整体性能.但我们看到,在数据中心网络流量特性和应用需求下,目前改进方案还存在以下一些关键的问题.

- 在负载均衡的控制机理方面,设计拥塞状态的快速感知方法和切换粒度的动态调节机制仍然是关键问题.感知端到端路径上的拥塞状态,可获取准确的全局负载信息,但其反馈周期过长,难以适应快速突发的网络流量;感知交换机本地缓存状态可快速获取局部负载信息,但可能与全局拥塞状态不符,容易导致错误的负载均衡行为.同时,路由的切换粒度也影响负载均衡性能:当切换粒度过大时,易发生网络利用率不足的问题;否则,容易导致数据包乱序问题.因此,如何进一步优化负载均衡的控制机理仍然值得研究.
- 在拥塞控制和负载均衡的一体化设计方面,存在着传输层和网络层之间状态不匹配的问题.目前的拥塞控制算法仅依据当前路径上的拥塞状态调整速率.而负载均衡做重路由操作时,新路径状态可能与当前路径不一致,容易发生发送速率与路径拥塞状态不匹配的问题.例如,重路由到空闲路径,发送速率过低会导致链路利用率低;而重路由到拥塞路径时,发送速率过高又会进一步加剧网络拥塞.从跨层设计的角度,如何实现网络层和传输层的协调联动,是整体性能提升的关键.
- 随着软件定义网络技术在负载均衡机制中的不断普及应用,如何降低 SDN 中央控制器处理信息的开销和如何解决控制器间同步性问题变得十分重要.SDN 负载均衡的决策依托于网络状态采集,而现有的信息采集是通过节点主动发送状态信息或者通过探测报文探测状态信息.探测周期、报文格式和状态信息不统一,增加了控制器的处理难度[101].另外,SDN 控制器的 TCAM 内存有限,过多的探测信息不但增加了网络流量负载,同时增加了控制器处理的延迟[45].在大规模数据中心中,单个控制器的计算能力有限,需要多个控制器协同[102,103].当多个控制器做决策时,控制器间的同步性也是一个重要的问题.
- 目前,RDMA[98,104−108]已经被广泛部署到数据中心中,以提升传输效率.但一旦发生乱序,RDMA 的 go-back-to-zero[108]或 go-back-to-N[108]机制需重传全部数据包,使得 RDMA 只能执行流级别的负载均衡,容易导致链路利用率降低.虽然文献[66]在专用网卡使用类似于 MPTCP 的多条并行子流进行传输,但是专用网卡增加了大规模部署的难度.因此,如何在基于 RDMA 技术的数据中心中设计高效的负载均

衡方法,也是值得关注的研究方向.

References:

[1] Alizadeh M, Greenberg A, Maltz DA, Padhye J, Patel P, Prabhakar B, Sengupta S, Sridharan M. Data center TCP (DCTCP). In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2010. 63−74.

[2] Wang C, Wang CR, Wang XW, Jiang DD. Network architecture design for data centers towards cloud computing. Jounal of Computer Research and Development, 2012,49(2):286−293 (in Chinese with English abstract).

[3] Luo L, Wu WJ, Zhang F. Energy modeling based on cloud data center. Ruan Jian Xue Bao/Journal of Software, 2014,25(7): 1371−1387 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4604.htm [doi: 10.13328/j.cnki.jos.004604]

[4] Luo JZ, Jin JH, Song AB, Dong F. Cloud computing: architecture and key technologies. Journal on Communications, 2011,32(7): 3−21 (in Chinese with English abstract).

[5] Li D, Chen GH, Ren FY, Jiang CL, Xu MW. Data center network research progress and trends. Chinese Journalof Computers, 2014,37(2):259−274 (in Chinese with English abstract).

[6] Huang JW, Huang Y, Wang JX, He T. Packet slicing for highly concurrent TCPs in data center networks with COTS switches. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). Piscataway: IEEE, 2015. 22−31.

[7] Zhang J, Yu FR, Wang S, Huang T, Liu ZY, Liu YJ. Load balancing in data center networks: A survey. IEEE Communications Surveys and Tutorials, 2018,20(3):2324−2352.

[8] Yang Y, Yang JH, Qin DH, Wang YD, Ling X. DraLCD: Another traffic engineering method for data center networks. Acta Electronica Sinica, 2017,45(5):1261−1267 (in Chinese with English abstract).

[9] Shan DF, Ren FY. Improving ECN marking scheme with micro-burst traffic in data center networks. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2017. 1−9.

[10] Chen XQ, Feibish SL, Koral Y, Rexiford J, Rottenstreich O, Monetti SA, Wang TY. Fine-grained queue measurement in the data plane. In: Proc. of the Int'l Conf. on Emerging Networking Experiments and Technologies (CoNEXT). New York: ACM, 2019. 15−29.

[11] Deng G, Gong ZH, Wang H. Characteristics research on modern data center network. Journal of Computer Research and Development, 2014,51(2):395−407 (in Chinese with English abstract).

[12] Wang BF, Su JS, Chen L. Review of the design of data center network for cloud computing. Journal of Computer Research and Development, 2016,53(9):2085−2106 (in Chinese with English abstract).

[13] Wang YJ, Sun WD, Zhou S, Pei XQ, Li XY. Key technologies of distributed storage for cloud computing. Ruan Jian Xue Bao/ Journal of Software, 2012,23(4):962−986 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4175.htm [doi: 10.3724/SP.J.1001.2012.04175]

[14] Al-Fares M, Loukissas A, Vahdat A. A scalable, commodity data center network architecture. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2008. 63−74.

[15] Greenberg A, Hamilton JR, Jain N, Kandula S, Kim C, Lahiri P, Maltz DA, Patel P, Sengupta S. VL2: A scalable and flexible data center network. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2009. 51−62.

[16] Guo CX, Wu HT, Tan K, Shi L, Zhang YG, Lu SW. DCell: A scalable and fault-tolerant network structure for data centers. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2008. 75−86.

[17] GUO CX, Lu GH, Li D, Wu HT, Zhang X, Shi YF, Tian C, Zhang YG, Lu SW. BCube: A high performance, server-centric network architecture for modular data centers. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2009. 63−74.

[18] Zheng JQ, Zheng QM, Gao XF, Chen GH. Dynamic load balancing in hybrid switching data center networks with converters. In: Proc. of the Int'l Conf. on Parallel Processing (ICPP). New York: ACM, 2019. 1−10.

[19] Chen K, Wen XT, Ma XY, Chen Y, Xia Y, Hu CC, Dong QF. WaveCube: A scalable, fault-tolerant, high-performance optical data center architecture. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2015. 1903−1911.

[20] Wu DM, Sun XY, Xia YT, Huang X, Eugene TS. Hyperoptics: A high throughput and low latency multicast architecture for datacenters. In: Proc. of the USENIX Workshop on Hot Topics in Cloud Computing (HotCloud). Berkeley: USENIX, 2016. 1–6.

[21] Wilson C, Ballani H, Karagiannis T, Rowtron A. Better never than late: Meeting deadlines in datacenter networks. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2011. 50–61.

[22] Vamanan B, Hasan J, Vijaykumar TN. Deadline-aware datacenter TCP (D^2TCP). In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2012. 115–126.

[23] Hong CY, Caesar M, Godfrey PB. Finishing flows quickly with preemptive scheduling. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2012. 127–138.

[24] Bai W, Chen L, Chen K, Han D, Tian C, Wang H. Information-agnostic flow scheduling for commodity data centers. In: Proc.of the USENIX Symp. on Networked Systems Design and Implementation (NSDI). Berkeley: USENIX, 2015. 455–468.

[25] Benson T, Akella A, Maltz DA. Network traffic characteristics of data centers in the wild. In: Proc. of the ACM SIGCOMM Conf. on Internet Measurement (IMC). New York: ACM, 2010, 267–280.

[26] Hopps C. Analysis of an equal-cost multi-path algorithm. RFC 2992, 2000.

[27] Singh A, Ong J, Agarwal A, Anderson G, Armistead A, Bannon R, Boving S, Desai G, Felderman B, Germano P, Kanagala A, Provost J, Simmons J, Tanda E, Wanderer J, Holzle U, Stuart S, Vahdat A. Jupiter rising: A decade of clos topologies and centralized control in Google's datacenter network. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2015. 183–197.

[28] Roy A, Zeng HY, Bagga J, Porter G, Snoeren AC. Inside the social network's (datacenter) network. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2015. 123–137.

[29] Shan DF, Jiang WC, Ren FY. Analyzing and enhancing dynamic threshold policy of data center switches. IEEE Trans. on Parallel and Distributed Systems, 2017,28(9):2454–2470.

[30] Shan DF, Ren FY, Cheng P, Shu R, Guo CX. Micro-burst in data centers: Observations, analysis, and mitigations. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). Piscataway: IEEE, 2018. 88–98.

[31] Shan DF, Jiang WC, Ren FY. Absorbing micro-burst traffic by enhancing dynamic threshold policy of data center switches. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2015. 118–126.

[32] Al-Fares M, Radhakrishnan S, Raghavan B, Huang N, Vahdat A. Hedera: Dynamic flow scheduling for data center networks. In: Proc.of the USENIX Symp. on Networked Systems Design and Implementation (NSDI). Berkeley: USENIX, 2010. 1–15.

[33] Benson T, Anand A, Akella A, Zhang M. MicroTE: Fine grained traffic engineering for data centers. In: Proc. of the Int'l Conf. on Emerging Networking Experiments and Technologies (CoNEXT). New York: ACM, 2011. 1–12.

[34] Curtis AR, Kim W, Yalagandula P. Mahout: Low-overhead datacenter traffic management using end-host-based elephant detection. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2011. 1629–1637.

[35] Wen KY, Qian ZZ, Zhang S, Lu SL. OmniFlow: Coupling load balancing with flow control in datacenter networks. In: Proc. of the IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Piscataway: IEEE, 2016. 725–726.

[36] Shafiee M, Ghaderi J. A simple congestion-aware algorithm for load balancing in datacenter networks. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2016. 1–9.

[37] Perry J, Ousterhout A, Balakrishnan H, Shah D, Fugal H. Fastpass: A centralized "zero-queue" datacenter network. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2014. 307–318.

[38] Irteza SM, Bashir HM, Anwar T, Qazi IA, Dogar FR. Efficient load balancing over asymmetric datacenter topologies. Computer Communications, 2018,127(9):1–12.

[39] Li GZ, Guo ST, Yang YY. Multicast scheduling algorithm in software defined fat-tree data center networks. In: Proc. of the IEEE/ ACM Int'l Symp. on Quality of Service (IWQoS). Piscataway: IEEE, 2017. 1–9.

[40] Zhao YM, Chen K, Bai W, Yu ML, Tian C, Geng YH, Zhang YM, Li D, Wang S. RAPIER: Integrating routing and schedulingfor coflow-aware data center networks. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2015. 424–432.

[41] Wang S, Zhang J, Huang T, Pan T, Liu J, Liu YJ. FDALB: Flow distribution aware load balancing for datacenter networks. In: Proc. of the IEEE/ACM Int'l Symp. on Quality of Service (IWQoS). Piscataway: IEEE, 2016. 1–2.

[42] Wang W, Sun Y, Salamatian K, Li ZC. Adaptive path isolation for elephant and mice flows by exploiting path diversity in datacenters. IEEE Trans. on Network and Service Management, 2016,13(1):5−18.

[43] Trestian R, Katrinis K, Muntean GM. OFLoad: An OpenFlow-based dynamic load balancing strategy for datacenter networks. IEEE Trans. on Network and Service Management, 2017,14(4):792−803.

[44] Chen L, Lingys J, Chen K, Liu F. AuTO: Scaling deep reinforcement learning for datacenter-scale automatic traffic optimization. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2018. 191−205.

[45] De Pellegrini F, Maggi L, Massaro A, Saucez D, Leguay J, Altman E. Blind, adaptive and robust flow segmentation in datacenters. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2018. 10−18.

[46] Gao XF, Kong LH, Li WC, Liang WC, Chen YX, Chen GH. Traffic load balancing schemes for devolved controllers in mega data centers. IEEE Trans. on Parallel and Distributed Systems, 2017,28(2):572−585.

[47] Chen L, Chen K, Bai W, Alizadeh M. Scheduling mix-flows in commodity datacenters with karuna. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2016. 174−187.

[48] Kabbani A, Vamanan B, Hasan J, Duchene F. FlowBender: Flow-level adaptive routing for improved latency and throughput in datacenter networks. In: Proc. of the Int'l Conf. on Emerging Networking Experiments and Technologies (CoNEXT). New York: ACM, 2014. 149−160.

[49] Katta N, Hira M, Ghag A, Kim C, Keslassy I, Rexford J. CLOVE: How I learned to stop worrying aboutthe core and love the edge. In: Proc. of the ACM Workshop on Hot Topics in Networks (HotNets). New York: ACM, 2016. 155−161.

[50] Shi QY, Wang F, Feng D, Xie WB. ALB: Adaptive load balancing based on accurate congestion feedback for asymmetric topologies. In: Proc. of the IEEE/ACM Int'l Symp. on Quality of Service (IWQoS). Piscataway: IEEE, 2018. 1−6.

[51] Zhang H, Zhang JX, Bai W, Chen K, Chowdhury M. Resilient datacenter load balancing in the wild. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2017. 253−266.

[52] Chen G, Zhang WF. ELAB: End-host-based congestion aware load balancing for data center network. Journal on Communications, 2019,40(3):196−205 (in Chinese with English abstract).

[53] Raiciu C, Barre S, Pluntke C, Greenhalgh A, Wischik D, Handley M. Improving datacenter performance and robustness with multipath TCP. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2011. 266−277.

[54] Chen G, Lu YW, Meng Y, Li BJ, Tan K, Pei D, Cheng P, Luo LY, Xiong YQ, Wang XL, Zhao YJ. Fast and cautious: Leveraging multi-path diversity for transport loss recovery in data centers. In: Proc. of the USENIX Annual Technical Conf. (ATC). Berkeley: USENIX, 2016. 29−42.

[55] Dong EH, Fu XM, Xu MW, Yang Y. DCMPTCP: Host-based load balancing for datacenters. In: Proc. of the IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Piscataway: IEEE, 2018. 622−633.

[56] Sun J, Zhang Y, Wang X, Xiao SH, Xu Z, Wu HJ, Chen X, Han YN. DC^2-MTCP: Light-weight coding for efficient multi-path transmission in data center network. In: Proc. of the IEEE Int'l Parallel and Distributed Processing Symp. (IPDPS). Piscataway: IEEE, 2017. 419−428.

[57] Kheirkhah M, Wakeman I, Parisis G. MMPTCP: A multipath transport protocol for data centers. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2016. 1−9.

[58] He K, Rozner E, Agarwal K, Felter W, Carter J, Akella A. Presto: Edge-based load balancing for fast datacenter networks. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2015. 465−478.

[59] Li Z, Bi J, Zhang YR, Dogar AB, Qin CW. VMS: Traffic balancing based on virtual switches in datacenter networks. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). Piscataway: IEEE, 2017. 1−10.

[60] Li YR, Wei D, Chen XQ, Song ZH, Wu RH, Li YX, Jin X, Xu Wei. DumbNet: A smart data center network fabric with dumb switches. In: Proc. of the European Conf. on Computer Systems (EuroSys). New York: ACM, 2018. 1−13.

[61] Kabbani A, Sharif M. Flicr: Flow-level congestion-aware routing for direct-connect data centers. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2017. 1−9.

[62] Cao JX, Xia R, Yang PK, Guo CX, Lu GH, Yuan LH, Zheng YX, Wu HT, Xiong YQ, Maltz D. Per-packet load-balanced, low-latency routing for clos-based data center networks. In: Proc. of the Int'l Conf. on Emerging Networking Experiments and Technologies (CoNEXT). New York: ACM, 2013. 49−60.

[63] Geng YL, Jeyakumar V, Kabbani A, Alizadeh M. JUGGLER: A practical reordering resilient network stack for datacenters. In: Proc. of the European Conf. on Computer Systems (EuroSys). New York: ACM, 2016. 1−16.

[64] Handley M, Raiciu C, Agache A, Voinescu A, Moore AW, Antichi G, Wojcik M. Re-Architecting datacenter networks and stacks for low latency and high performance. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2017. 29−42.

[65] Hu JB, Huang JW, Lv WJ, Zhou YT, Wang JX, He T. CAPS: Coding-based adaptive packet spraying to reduce flow completion time in data center. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2018. 2294−2302.

[66] Chen G, Lu YW, Li BJ, Tan K, Xiong YQ, Cheng P, Zhang JS, Chen EH, Moscibroda T. MP-RDMA: Enabling RDMA with multi-path transport in datacenters. IEEE/ACM Trans. on Networking (TON), 2019,28(1):1−16.

[67] Kim C, Sivaraman A, Katta N, Bas A, Dixit A, Wobker LJ. In-band network telemetry via programmable dataplanes. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2015. 1−2.

[68] Li YL, Miao R, Liu HQ, Zhuang Y, Feng F, Tang LB, Cao Z, Zhang M, Kelly F, Alizadeh M, Yu ML. HPCC: High precision congestion control. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2015. 44−58.

[69] Pan T, Song E, Bian ZZ, Lin XC, Peng XY, Zhang J, Huang T, Liu B, Liu YJ. INT-path: Towards optimal path planning for in-band network-wide telemetry. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2019. 487−495.

[70] Intel DPDK. Data plane development kit. 2020. http://dpdk.org/

[71] Xu H, LiBC. RepFlow: Minimizing flow completion time with replicated flows in data centers. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2014. 1581−1589.

[72] Cui Y, Wang L, Wang X, Wang HY, Wang YN. FMTCP: A fountain code-based multipath transmission control protocol. IEEE/ACM Trans. on Networking (TON), 2015,23(2):465−478.

[73] Zhou JL, Tewari M, Zhu M, Kabbani A, Poutievski L, Singh A, Vahdat A. WCMP: Weighted cost multipathing for improved fairness in data centers. In: Proc. of the European Conf. on Computer Systems (EuroSys). New York: ACM, 2014. 1−14.

[74] Wang P, Xu H, Niu ZX, Han DS, Xiong YQ. Expeditus: Congestion-aware load balancing in Clos data center networks. IEEE/ACM Trans. on Networking (TON), 2017,25(5):3175−3188.

[75] Xu H, Li BC. TinyFlow: Breaking elephants down into mice in data center networks. In: Proc. of the 20th IEEE Int'l Workshop on Local and Metropolitan Area Networks (LANMAN). Piscataway: IEEE, 2014. 1−6.

[76] Cui WZ, Yu Y, Qian C. DiFS: Distributed flow scheduling for adaptive switching in FatTree data center networks. Computer Networks, 2016,105(8):166−179.

[77] Cheung CM, Leung KC. DFFR: A flow-based approach for distributed load balancing in data center networks. Computer Communications, 2018,116(1):1−8.

[78] Olteanu V, Agache A, Voinescu A, Raiciu C. Stateless datacenter load-balancing with beamer. In: Proc.of the USENIX Symp. on Networked Systems Design and Implementation (NSDI). Berkeley: USENIX, 2018. 125−139.

[79] Grigoryan G, Liu Y, Kwon M. iLoad: In-network load balancing with programmable data plane. In: Proc. of the Int'l Conf. on Emerging Networking Experiments And Technologies (CoNEXT). New York: ACM, 2019. 17−19.

[80] Al-Tarazi M, Chang JM. Performance-Aware energy saving for data center networks. IEEE Trans. on Network and Service Management, 2019,16(1):206−219.

[81] Xue JC, Chaudhry MU, Vamanan B, Vijaykumar TN, Thottethodi M. Dart: Divide and specialize for fast response to congestion in RDMA-based datacenter networks. IEEE/ACM Trans. on Networking, 2020,28(1):322−335.

[82] Zats D, Das T, Mohan P, Borthakur D, Katz R. DeTail: Reducing the flow completion time tail in datacenter networks. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2012. 139−150.

[83] Dixit A, Prakash P, Hu YC, Kompella RR. On the impact of packet spraying in data center networks. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2013. 2130−2138.

[84] Ghorbani S, Yang ZB, Godfrey PB, Ganjali Y, Firoozshahian A. DRILL: Micro load balancing for low-latency data center networks. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2017. 225−238.

[85] Qian ZM, Fan FJ, Hu B, Yeung KL, Li LY. Global round robin: Efficient routing with cut-through switching in fat-tree data center networks. IEEE/ACM Trans. on Networking (TON), 2018,26(5):2230−2241.

[86] Huang JW, Lv WJ, Li WH, Wang JX, He T. QDAPS: Queueing delay aware packet spraying for load balancing in data center. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). Piscataway: IEEE, 2018. 66−76.

[87] Zou SJ, Huang JW, Wang JX, He T. Improving TCP robustness over asymmetry with reordering marking and coding in data centers. In: Proc. of the IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Piscataway: IEEE, 2019. 57−67.

[88] Liu S, Huang JW, Jiang WC, Wang JX, He T. Reducing flow completion time with replaceable redundant packets in data center networks. In: Proc. of the IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Piscataway: IEEE, 2019. 46−56.

[89] Kandula S, Katabi D, Sinha S, Berger A. Dynamic load balancing without packet reordering. ACM SIGCOMM Computer Communication Review, 2007,37(2):51−62.

[90] Alizadeh M, Edsall T, Dharmapurikar S, Vaidyanathan R, Chu K, Fingerhut A, Matus F, Pan R, Yadav N, Varghese G. CONGA: Distributed congestion-aware load balancing for datacenters. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2014. 503−514.

[91] Katta N, Hira M, Kim C, Sivaraman A, Rexford J. HULA: Scalable load balancing using programmable data planes. In: Proc. of the Symp. on SDN Research (SOSR). New York: ACM, 2016. 1−12.

[92] Chen Y, Wu J. High network utilization load balancing scheme for data centers. In: Proc. of the IEEE Global Communications Conf. (GLOBECOM). Piscataway: IEEE, 2016. 1−6.

[93] Vanini E, Pan R, Alizadeh M, Taheri P, Edsall T. Let it flow: Resilient asymmetric load balancing with flowlet switching. In: Proc.of the USENIX Symp. on Networked Systems Design and Implementation (NSDI). Berkeley: USENIX, 2017. 407−420.

[94] Wang P, Trimponias G, Xu H, Geng YH. Luopan: Sampling based load balancing in data center networks. IEEE Trans. on Parallel and Distributed Systems, 2019,30(1):133−145.

[95] Fan FJ, Hu B, Yeung KL. Routing in black box: Modularized load balancing for multipath data center networks. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2019. 1639−1647.

[96] Liu JL, Huang JW, Li WH, Wang JX. AG: Adaptive switching granularity for load balancing with asymmetric topology in data center network. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). Piscataway: IEEE, 2019. 1−11.

[97] Hu JB, Huang JW, Lv WJ, Li WH, Wang JX, He T. TLB: Traffic-aware load balancing with adaptive granularity in data center networks. In: Proc. of the Int'l Conf. on Parallel Processing (ICPP). New York: ACM, 2019. 1−10.

[98] Zhu YB, Eran H, Firestone D, Lipshteyn M, Liron Y, Padhye JD, Raindel S, Yahia MH, Zhang M. Congestion control for large-scale RDMA deployments. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2015. 523−536.

[99] Priority Flow Control: Build Reliable Layer 2 Infrastructure. Cisco White Paper, 2015.

[100] Mahalingam M, Dutt D, Duda K, Agarwal P, Kreeger L, Sridhar T, Bursell M, Wright C. VXLAN: A framework for overlaying virtualized layer 2 networks over layer 3 networks. RFC 7348, 2014.

[101] Kim C, Bhide P, Doe E, Holbrook H, Ghanwani A, Daly D, Hira M, Davie B. In-band network telemetry (INT). P4 Language Consortium, 2016. https://p4.org/assets/INT-current-spec.pdf

[102] Poularakis K, Qin Q, Ma L, Kompella S, Leung KK, Tassiulas L. Learning the optimal synchronization rates in distributed SDN control architectures. In: Proc. of the IEEE Int'l Conf. on Computer Comunications (INFOCOM). Piscataway: IEEE, 2019. 1099−1107.

[103] Zhang ZY, Ma L, Poularakis K, Leung KK, Tuckerx J, Swami A. MACS: Deep reinforcement learning based SDN controller synchronization policy design. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). Piscataway: IEEE, 2019. 1−11.

[104] Mittal R, Lam VT, Dukkipati N, Blem E, Wassel H, Ghobadi M, Vahdat A, Wang YG, Wetherall D, Zats D. TIMELY: RTT-based congestion control for the datacenter. In: Proc. of the ACM Special Interest Group on Data Communication (SIGCOMM). New York: ACM, 2015. 537−550.

[105] Xiao MB, Wang H, Geng L, Lee R, Zhang XD. Catfish: Adaptive RDMA-enabled R-tree for low latency and high throughput. In: Proc. of the IEEE Int'l Conf. on Distributed Computing Systems (ICDCS). Piscataway: IEEE, 2019. 164−175.

[106] Xue JC, Chaudhry MU, Vamanan B, Vijaykumar TN, Thottethodi M. Fast congestion control in RDMA-based datacenter networks. In: Proc. of ACM SIGCOMM Posters and Demos. 2018. 24−26.

[107] Gao YX, Yang YC, Tian Chen, Zheng JQ, Mao B, Chen GH. DCQCN+: Taming large-scale incast congestion in RDMA over ethernet networks. In: Proc. of the IEEE Int'l Conf. on Network Protocols (ICNP). Piscataway: IEEE, 2018. 110−120.

[108] Guo ZH, Liu S, Zhang ZL. Traffic control for RDMA-enabled data center networks: A survey. IEEE Systems Journal, 2020,14(1): 677−688.

附中文参考文献:

[2] 王聪,王翠荣,王兴伟,蒋定德.面向云计算的数据中心网络体系结构设计.计算机研究与发展,2012,49(2):286−293.

[3] 罗亮,吴文峻,张飞.面向云计算数据中心的能耗建模方法.软件学报,2014,25(7):1371−1387. http://www.jos.org.cn/1000-9825/4604.htm [doi: 10.13328/j.cnki.jos.004604]

[4] 罗军舟,金嘉晖,宋爱波,东方.云计算:体系架构与关键技术.通信学报,2011,32(7):3−21.

[5] 李丹,陈贵海,任丰原,蒋长林,徐明伟.数据中心网络的研究进展与趋势.计算机学报,2014,37(2):259−274.

[8] 杨洋,杨家海,秦董洪,王于丁,凌晓.DraLCD:一种新的数据中心流量工程方法.电子学报,2017,45(5):1261−1267.

[11] 邓罡,龚正虎,王宏.现代数据中心网络特征研究.计算机研究与发展,2014,51(2):395−407.

[12] 王斌锋,苏金树,陈琳.云计算数据中心网络设计综述.计算机研究与发展,2016,53(9):2085−2106.

[13] 王意洁,孙伟东,周松,裴晓强,李小勇.云计算环境下的分布存储关键技术.软件学报,2012,23(4):962−986. http://www.jos.org.cn/1000-9825/4175.htm [doi: 10.3724/SP.J.1001.2012.04175]

[52] 陈果,张潍丰.ELAB:基于端系统的新型拥塞感知负载均衡机制.通信学报,2019,40(3):196−205.

刘敬玲(1994－),女,博士生,主要研究领域为数据中心网络.

蒋万春(1987－),男,博士,副教授,CCF 专业会员,主要研究领域为计算机网络,分布式系统.

黄家玮(1976－),男,博士,教授,博士生导师,主要研究领域为云计算,数据中心,软件定义网络,Web 优化,流媒体.

王建新(1969－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为计算机算法与优化,网络优化理论,大数据处理,深度学习,生物信息学,虚拟实验环境.