软件学报 ISSN 1000-9825, CODEN RUXUEW
*Journal of Software*,2016,27(Suppl.(2)):243−253

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

# 数据中心网络中的流量均衡*

李兆耕[1,2,3], 毕 军[1,2,3]

[1](清华大学 网络科学与网络空间研究院,北京 100084)

[2](清华大学 计算机科学与技术系,北京 100084)

[3](清华信息科学与技术国家实验室(筹)(清华大学),北京 100084)

通讯作者: 李兆耕, E-mail: li-zg07@mails.tsinghua.edu.cn

**摘 要**: 现代数据中心网络在任意两个主机之间都存在很多可选路径.如何在多个可选路径之间实现流量均衡,是数据中心网络中的重要研究课题.针对这一问题,已有研究者提出了很多解决方案.对多级Clos架构下的数据中心网络中的流量均衡问题做了深入的分析与总结.首先分析了数据中心网络的特点,然后定义流量均衡问题为最小化等价链路的最大潜在丢包率.之后总结了各种丢包产生的原因,并讨论了影响流量均衡设计方案的两个主要挑战:分组乱序与突发拥塞.在此基础上,把现有解决方案分为主动调度、切片散射、探测与调整及其他这 4 个大类,并对各解决方案逐一进行介绍,说明各自的优缺点.最后还对不同方案作了对比,指出了未来可能的研究方向.

**关键词**: 数据中心网络;流量均衡;ECMP;潜在丢包;突发拥塞

中文引用格式: 李兆耕,毕军.数据中心网络中的流量均衡.软件学报,2016,27(Suppl.(2)):243−253. http://www.jos.org.cn/1000-9825/16038.htm

英文引用格式: Li ZG, Bi Jun. Traffic balancing in data center networks. Ruan Jian Xue Bao/Journal of Software, 2016,27(Suppl.(2)):243−253 (in Chinese). http://www.jos.org.cn/1000-9825/16038.htm

## Traffic Balancing in Data Center Networks

LI Zhao-Geng[1,2,3], BI Jun[1,2,3]

[1](Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing 100084, China)

[2](Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

[3](Tsinghua National Laboratory for Information Science and Technology (TNList) (Tsinghua University), Beijing 100084, China)

**Abstract**: There are many optional paths between any server pair in modern data center networks. Traffic balancing among the paths is an important issue. Many solutions have been proposed to cope with this problem. This paper analyzes traffic balancing in data center networks with multi-tier Clos-based topology. The feature of data center networks is introduced and the traffic balancing problem is defined as minimizing the maximum potential packet loss among different equivalent links. Then the reasons of potential packet loss are summarized in data center networks and two main challenges are discussed related to traffic balancing: packet out-of-order and burst congestion. Ten existing solutions are classified into four categories: active scheduling, slice spraying, probing and adjusting, and others. The advantages of these solutions are described one by one in details, as well as their disadvantages. A comparison is made among these solutions and possible research directions are pointed out on traffic balancing in data center networks.

**Key words**: data center network; traffic balancing; ECMP; potential packet loss; burst congestion

今天,数据中心作为云计算的核心基础架构,其规模越来越大.当前一个大型的数据中心[1,2]通常可以容纳

* 基金项目: 国家自然科学基金(61472213)
Foundation item: National Natural Science Foundation of China (61472213)
收稿时间: 2016-06-05; 采用时间: 2016-10-18

至少几万台服务器,并且采用多层 Clos 网络将这些服务器连接起来.这种结构的数据中心网络在提高可靠性和可用带宽的同时,也引入了流量均衡问题,即如何将流量(主要是东西流量)均匀地分布在不同的可用路径上,以避免拥塞.流量均衡的实际效果对数据中心内部应用的数据传输性能影响巨大.

数据中心网络通常采用 ECMP 方法来实现流量均衡,即为每个不同的 TCP/UDP 流,根据其五元组哈希的不同,等概率地选择不同的等代价路径进行转发.ECMP 不要求交换机掌握网络中的流量信息和拓扑信息,也不需要记录自身的历史转发选择,因此最为简单.

但是,ECMP 方法有 3 个主要缺陷:缺乏全局信息;无状态;流和路径之间为单一映射.由于缺乏全局信息,ECMP 只能保证在完全对称的拓扑下其均衡是有效的,因此,ECMP 无法应对设备失效所导致的拓扑不对称.而无状态的本质属性在哈希算法的随机性以及流大小的随机性影响下会导致在一个具体的时刻,流量在可选路径集合中的分布是不均匀的.而一个流在其生命周期中,由于其与转发路径之间的单一映射关系,网络无法切换其选择的路径,因此灵活性大为降低.

针对这些问题,研究者们提出了一些更为高级的流量均衡方案.这些方案遵循了以下可能的修改思路:集中化的调度控制、分布式的负载信息探测、流的切分等.在具体实现上,不同的方案之间有很大的区别,有的选择修改交换机,有的选择修改终端服务器,还有的选择两者同时修改.所有这些流量均衡方案,其效果与 ECMP 相比都有很大的改进,但也都存在一些缺点.

本文尝试对数据中心网络中的流量均衡问题进行深入分析,定义流量均衡的目标并找出与方案设计相关的主要挑战,梳理现有的代表性解决方案并进行分类.对每一种流量均衡方案,本文会具体描述其核心思想与实现机制,并指出其优缺点.最后,本文还对这些方案进行了对比总结,并说明可能的未来研究方向.

本文第 1 节介绍数据中心网络的结构特征和流量均衡问题.第 2 节对影响流量均衡方案设计的关键挑战进行分析.第 3 节对各主要方案进行归类,并逐一介绍这些解决方案的具体设计以及优缺点.第 4 节给出不同方案的对比并提出未来可能的研究方向.第 5 节是本文论述的总结.

## 1 数据中心网络中的流量均衡

### 1.1 数据中心网络

为了降低成本并提高鲁棒性,现代数据中心网络大多采用普通交换机组成的多层 Clos 结构(多为 3 层,如图 1 所示).这种网络结构具有以下两个特征:对称性与分组性.对称性表明一个交换机与其他任何一个同层交换机在整个拓扑中都是等价的.分组性表明对任何两个同层交换机而言,与它们相连的上一层交换机要么完全相同(两者同组,如图 1 中的 $C_1$ 和 $C_2$),要么完全不同(两者不同组,如图 1 中的 $C_1$ 和 $C_5$).

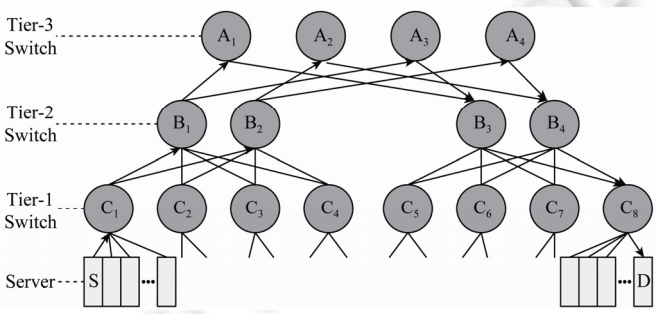



Fig.1 Data center network with multi-tier Clos topology

图 1 多层 Clos 结构数据中心网络

符合以上特征的网络结构可被定义为可折叠的 Clos 网络(FCN[3]),如 Fat Tree[4].现有大多数在实际应用中的数据中心网络都是这一拓扑的直接实例(如 Facebook 的数据中心[1])或间接变种(如 Google 的数据中心[2]).部

分规模较小的数据中心网络也都具备上述特征,如采用 Spine-Leaf 结构的数据中心网络(两层,单一分组).但是应该注意到,由于数据中心规模很大,因设备故障或升级所导致的链路临时失效问题是普遍存在的[5].因此,在很多情况下,数据中心网络实际运行的拓扑结构并不是严格对称的.

随着数据中心网络内部流量需求的逐步攀升,现有大多数数据中心网络都将服务器的接入带宽设为10Gbps,40Gbps 甚至是 100Gbps,而交换核心(fabric)的链路带宽也至少是同一大小或者更高.为了提供充足的对分带宽(bisection bandwidth),数据中心网络会尽可能地降低带宽收敛比(oversubscription,指一层交换机连接下层交换机或服务器的总带宽与连接上层交换机的总带宽之比).此外,多数大型数据中心网络是基于IP进行转发的,同一个机架下(即第 1 层交换机下)的服务器处于同一网段,位于不同机架之间的服务器则处于不同网段.

### 1.2 等价链路与流量均衡

在数据中心网络中,在一个源服务器和一个目的服务器之间存在数量众多的可选路径.这些可选路径在交换机的路由表中一般都具有同等的代价度量.因此,交换机通常采用 ECMP 的机制(五元组哈希)将流量分散在不同的可选路径中.这里定义等价链路的概念:对于两个有向链路 $I_1 \rightarrow J_1$ 和 $I_2 \rightarrow J_2$($I_1$ 和 $I_2$ 是指有向链路起始端的交换机,$J_1$ 和 $J_2$ 是指有向链路末尾端的交换机),如果 $I_1$ 和 $I_2$ 处于同层,且 $J_1$ 和 $J_2$ 也处于同层,同时对任意一个流,两条链路要么都可以用于承载该流,要么都不可以用于承载该流,那么这两条链路互为等价链路.由于等价链路这一关系具备自反性、对称性与传递性,因此,可以在此基础上进一步定义等价类的概念.

例如,在图 1 中,对于一个从服务器 $S$ 到服务器 $D$ 的流,在上行方向(低层交换机向高层交换机转发的方向),$C_1 \rightarrow B_1$ 与 $C_1 \rightarrow B_2$ 这两条链路是等价的,$B_1 \rightarrow A_1$,$B_1 \rightarrow A_3$,$B_2 \rightarrow A_2$,$B_2 \rightarrow A_4$ 这 4 条链路也是等价的.同样地,在下行方向(高层交换机向低层交换机转发的方向),$B_3 \rightarrow C_8$ 与 $B_4 \rightarrow C_8$ 这两条链路组成了一个等价类,$A_1 \rightarrow B_3$,$A_3 \rightarrow B_3$,$A_2 \rightarrow B_4$,$A_4 \rightarrow B_4$ 这 4 条链路也是属于一个等价类的.

需要注意的是,这里的等价链路并不一定是完全相同的.如果两个链路容量不同(如在端口聚合的情况下,其中一条逻辑链路有单个物理端口失效),但都满足上述定义,也应该将其视为等价链路,因为不同链路代表的路径选择在交换机路由表中的代价度量依然是相同的.

数据中心网络流量均衡问题就是研究如何将流量尽可能均匀地分布在这些等价链路之中.在传统的基于互联网主干网的流量工程研究中,处理多路径流量分布问题时,多采用“最大链路利用率最小化”这样的优化目标.然而,在数据中心网络流量均衡这一问题中,并不应采用这样的优化目标,因为链路利用率是一个统计平均值,需要流量在时间尺度上具有延续性,需要很高的复用度以消除总体流量的突发性,而数据中心网络流量特征并不保证满足这样的需求.

因此,本文将数据中心网络流量均衡的目标定义为最小化等价链路的最大潜在丢包率.这里的潜在丢包既包括真实丢包,也包括 ECN 所标识的显式拥塞通告,因为终端一般会根据 ECN 标识做发送速率的调整.之所以以潜在丢包率为目标,是因为其对数据中心网络中的应用影响巨大,会显著提高流完成时间的长尾分布.如果可以确保没有潜在丢包,那么应允许一个等价类中的链路其利用率相差很大,而不必强行迁移流(增加网络复杂度且可能增加分组乱序)以达到利用率均等的目标.

### 1.3 丢包的原因分析

如前所述,流量均衡的主要目标是尽可能降低等价链路的最大丢包率.通常而言,在数据中心网络中.丢包主要由 3 个方面的原因引起:路由问题、设备问题以及链路拥塞(如图 2 所示).路由问题是指在拓扑发生改变(如链路故障引发)后路由更新尚未完成时的临时性丢包;设备问题是指因设备缺陷所导致的永久性丢包,也可称为转发黑洞[6];链路拥塞是指当要在一条链路上转发的数据量超过了链路容量时,未及时转发的数据填满了交换机缓冲区而带来的随机性丢包.潜在丢包中的 ECN 标识与拥塞相关,因此包含在第 3 种丢包中.

链路拥塞有两种可能的原因,一是总体带宽短缺,通常由高带宽收敛比或多对一通信(如 TCP Incast)引起;二是流量分布不均,即局部带宽短缺.考虑这样一个简单的例子(本文后面以 ndmc 问题指代这一例子):有 $n$ 个峰值带宽需求为 $d$ 的长流,要分布在 $m$ 个容量为 $c$ 的等价链路中.如果 $nd>mc$,就会出现总体带宽短缺.如果 $nd \leqslant mc$,

则依然有可能有部分链路发生队列溢出,这时则属于流量分布不均.比如,当 $n$=3,$m$=4 且 $d$=$c$ 时,如果一个流被分配给每个链路的概率是相同的(以 ECMP 为例),那么依然有 62.5%的概率会发生拥塞(即将两个或更多流交给同一条链路转发).

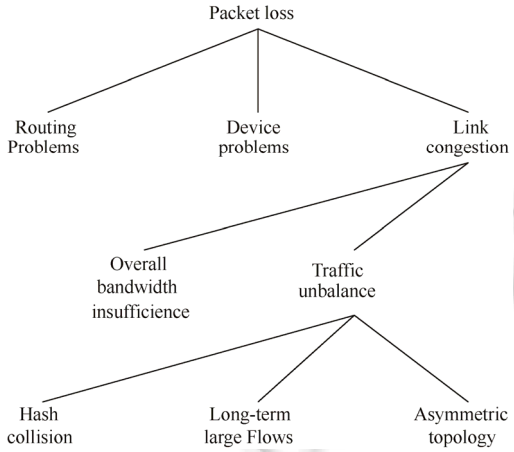



Fig.2 Reasons of packet loss in data center networks

图 2 数据中心网络中的丢包原因

上面的例子只是导致流量分布不均的一种可能,即哈希冲突.当使用 ECMP 时,还有另外两种原因会导致流量分布不均:大流持续和拓扑不对称.大流持续是指在有持续时间长的大流作为背景流量的情况下,依然采用恒定权重的哈希算法随机选路而导致的流量分布不均,其根本原因是 ECMP 的无状态本质.拓扑不对称是指当拓扑在下游失去对称性时,路径上游的交换机根据对称性做出的均匀分布选择在下游引发了不均匀的结果,其根本原因是 ECMP 无全局视图.这些问题并不能通过给不同的可选链路简单地加上权重值来彻底地解决(如文献[7]).

## 2 流量均衡面临的主要挑战

### 2.1 分组乱序

为了实现流量均衡,有时必须把一个流从中间打断,对打断前后的报文采用不同的路径进行转发.例如在 ndmc 问题中,当 $n$=3,$d$=6,$m$=2,$c$=10 时,只有打断流才有可能避免拥塞.最极端的情况下,甚至可以为每一个报文单独选路.但这种对流进行拆分的机制有可能会引起分组乱序(packet out-of-order).分组乱序对于 TCP 的性能有很大的影响.这体现在以下几个方面[8].

- 很多 TCP 拥塞控制算法会把分组乱序错误地识别为丢包从而发起快速重传,引发不必要的带宽消耗与拥塞窗口抑制,进而影响吞吐率.
- 分组乱序会让发送端错误地估计 RTT,从而可能低估 RTO,导致超时误判,同样引发不必要的重传与拥塞窗口抑制,影响吞吐率.
- 分组乱序会导致 ACK 时序混乱,加剧 TCP 流量的突发性.
- 分组乱序会导致传统的网卡负载剥离方案 GRO 发挥不出效果,导致 CPU 消耗高[9].

尽管有研究提出了可容忍乱序的 TCP 修改方案,但是都很难让吞吐率完全恢复到没有乱序时 TCP 的水平.同时,由于修改方案的计算复杂度相对更高,应用在数据中心网络高带宽的场景下会使 CPU 更容易成为瓶颈.因此,数据中心网络流量均衡方案应该尽量避免并着力处理分组乱序.

### 2.2 突发拥塞

由于数据中心网络中单个流可以具有很高的吞吐率,而所使用的交换机缓冲区却相对较小(shallow-buffer),因此,数据中心网络中会有很多突发拥塞,即队列在非常短的时间内迅速增长并发生溢出.假设交换机单端口可用的缓冲区大小为 1MB,而链路容量为 10Gbps,如果有 3 个端口进入的报文需要向另外一个端口转发,那么将该转出端口的可用缓冲区从 0 开始填满最快只需要约 500μs,相当于约两个 RTT 的时间(一个 RTT 大约在 200μs~300μs 之间[6]).这就给提前发现拥塞带来了很大的挑战.也正因如此,一般数据中心网络中所使用的 ECN 机制,要求交换机根据队列的瞬时大小而非主动队列管理(AQM)中常用的平均大小来决定是否进行 CE 标记[10].由于突发拥塞的存在,数据中心网络流量均衡方案应该尽可能在 1 个~2 个 RTT 时间内发现潜在的因不均衡导致的拥塞,从而执行相应的措施.

## 3 流量均衡的主要方法

解决流量均衡问题,主要有以下几种方法.

- 主动调度.将所有的流(主要是大流)单独调度,使得各个等价链路上的负载相对平均.
- 切片散射.直接以分组级别或近似于分组的级别的流切片为单位转发报文,使得各个等价链路上的负载相对平均.
- 探测与调整.探测不同路径上的实际负载,较多地使用探测到的负载相对较低的路径,较少使用探测到的负载相对较高的路径.

几乎所有的具体方案都可以归为以上 3 种主要方法中的一种.在每一种方法中,又会存在一些细分的子方法.例如,在切片散射中,又会分为基于分组的切片和基于分组集合的切片;在探测与调整方法中,会根据探测的执行者不同分为基于终端的探测和基于网络的探测.除此之外,还有一些方案,因为改动较多,难以给出单一的分类,因此本文将其统归为其他方案(如图 3 所示).

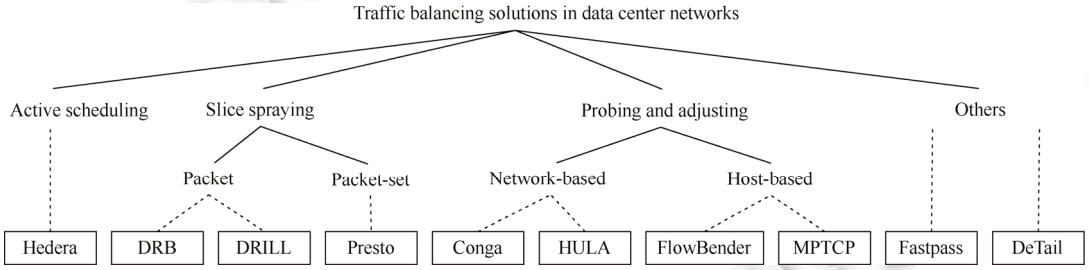



Fig.3 Categories of traffic blancing solutions in data center networks

图 3 数据中心网络流量均衡方案分类

### 3.1 主动调度

主动调度的代表性方案为 Hedera[11],其基本思想是,在默认情况下,交换机使用 ECMP 机制转发流量.交换机发现网络中的大流后,将大流信息发送给集中的调度器.调度器会对所有的大流做统一调度,并将调度结果以 Openflow 流表项的形式下发给交换机.交换机对于在流表中命中的流,使用流表项指示的出口转发,否则继续使用 ECMP 机制转发.

Hedera 的设计提出了流量真实需求预估的方法.大流的实际带宽需求并不一定等于检测到的大流的实际带宽占用,因为此时的带宽占用会受到拥塞影响.事实上,在带宽收敛比为 1:1 的数据中心网络中,大流的实际带宽需求只应受到同一接入链路上其他大流的竞争影响.因此,Hedera 要求调度器采用迭代松弛的方法来发现大流的真实带宽需求.在预估了真实流量需求后,调度器再利用全局优先适配或模拟退火两种算法来执行对大流的优化调度.

尽管 Hedera 集中调度的算法在理论上非常有效,但是这并不能彻底解决实际中的流量均衡问题.其主要原因是,Hedera 假设拥塞主要来自于大长流之间的竞争.事实上,较小较短的流由于瞬时速率也可以很高,也很有可能形成相互竞争.而基于大流识别汇报、集中统筹、下发规则这样的集中式解决方案出于可扩展性的考虑,是无法顾及非大长流的.因此,Hedera 对于动态性小的流量而言,其效果显著,但却很难适应动态性较强的流量.

### 3.2 切片散射

如果将流分成非常小的切片分别转发,流量就成为可任意切分的,那么可以证明最终的流量均衡效果近似最优[3].然而,这种方式会引入大量的分组乱序,从而影响终端的实际性能.因此,必须保证等价链路对应的队列长度严格接近.

根据切片的大小不同,方案可以分成两类:基于分组的切片和基于分组集的切片.

#### 3.2.1 基于分组的切片

基于分组的切片方式最容易实现.最简单的两种散射方式是随机散射和轮转(round-robin)散射.但这两种方式都会引入大量的分组乱序,即使是在没有拓扑不对称的情况下.DRB[12]和 DRILL[13]两个方案对散射方式进行了改进.

- DRB

DRB 即数位逆转散射,本质上是带状态的轮转散射,数位逆转只是在特定拓扑下有意义.DRB 通过在服务器上为每一个目的端维护一个状态值,依靠此状态值自增来依次选择封装报文使用的第 3 层交换机(类似于 VL2[14]的方式).

- DRILL

DRILL 方案本质上是最短队列散射.在转发报文时,从可选的出端口中随机挑选 $d$ 个,以及 $m$ 个在前一阶段拥有最低负载的出端口,在这最多 $d+m$ 个出端口中,选择队列最短的那个用于转发.事实上,通常 $d=2$,$m=1$ 即可满足需求.

无论是 DRB 所使用的带状态的轮转散射,还是 DRILL 所使用的最短队列散射,都能在很大程度上减少分组乱序.但是分组乱序并不能完全消除,因为分组本身的大小不等,且有拓扑不对称的可能[15].因此,两种方案都要求 TCP 协议栈做修改,即在接收端增加一个额外的重排缓冲区,在重排后再交给接收缓冲区来决定发送确认 ACK.但是,这种终端预重排的方式会引入额外的重排延时与重排开销.

像 DRB 或是 DRILL 这样的方案取得了非常好的流量均衡效果.但是方案在实际中却很难被部署,其原因包括:(1) 每一个流都被分散到所有可能的路径上,任何一个交换机出现问题,都会影响到几乎所有的流.(2) 尽管 TCP 可以增加重排,但原有的网卡用于 TCP 重组的负载剥离功能 GRO 却无法正常使用[6],因此在大带宽下会显著增加 CPU 的消耗,甚至 CPU 成为吞吐率的瓶颈.(3) 新的传输技术 RDMA 在很多情况下取代了 TCP,但 RDMA 传输是不容许乱序的[16].

#### 3.2.2 基于分组集的切片

基于分组集的切片方案可以认为是对基于分组的切片方案的一种改进.以 Presto[6]为例,每个流都被分成 64KB 大小**的分组集(也称为流胞(flowcell)),使用轮转散射的方式,依次切换所选路径.这样,在同一个分组集(如果 MTU 为 1 500 字节,那么一个分组集最少有 44 分组)中,分组乱序不会发生.分组乱序只有可能发生在不同分组集的边界处,因此极大地减少了分组乱序的出现时机.

为了实现基于分组集的路径切换,Presto 要求网络采用 Openflow 这样的交换机,并接受集中化控制器的管控.Presto 为每一个主机分配了一组影子 MAC 地址,针对每一个不同的影子 MAC 地址,交换机将使用不同的路径对报文进行转发.主机的操作系统(或虚拟化场景中宿主机的操作系统 hypervisor),对一个流每发送 64KB 数据就切换一次影子 MAC 地址,从而使网络切换流的转发路径,将流量均匀地分散在所有的等价链路上.

基于分组集的切片方案完全消除了小流(小于 64KB 的流)的乱序,但并没有彻底解决分组乱序问题.采用

** 64KB 是在多数操作系统启用 TSO 功能时,默认情况下,TCP 协议栈实际操作的分片大小.

64KB 的分组集,一旦发生乱序,其影响却比基于分组的切片方案更加严重(可能会接连出现很多个乱序报文,更容易被发送端识别为丢包从而导致拥塞窗口调整,增加终端重排所需的缓冲区大小).此外,基于分组集的切片没有基于分组的切片容易实现,需要额外设计来支持对分组所属的分组集进行归类(例如,Presto 中的集中管控与终端修改).

### 3.3 探测与调整

主动调度与切片散射都是希望各个等价链路的利用率尽可能地均匀.而正如前所述,既然潜在丢包才是流量均衡关注的重点,那么完全可以不对链路利用率做均衡,只在探测到有可能丢包(即拥塞)时再做出调整,这就是探测与调整的方法.根据执行探测的发起者的不同,可以分为基于网络的探测与调整和基于终端的探测与调整.

#### 3.3.1 基于网络的探测与调整

- Conga

Conga[17]是这种方案的代表.在 Conga 中,边界交换机维护一张记录负载信息的二维表,即由目的边界交换机和出口共同确定的路径上的负载程度.对任意一个报文,交换机将从该表中选择对应的目的边界交换机中负载程度最低的出口转发.

在 Conga 中,交换机需要为每一个出口维护一个寄存器,量化负载程度.当一个报文通过该出口转发时,交换机需要把当前的负载程度标记在用 VxLAN 封装后的头部中.一个报文在转发路径中会经过多个交换机,负载程度标记仅发生在当前负载程度比报文中已经携带的负载程度更大时.最终这个标记会到达目的边界交换机.目的边界交换机将把这一信息存在另一个表里,然后通过反向背负(piggyback)的方式,将这一信息反馈给源边界交换机,供后续转发参考.

由于报文的转发选择是即时指定的,为了确保一个流的报文能通过相同的路径,Conga 中的边界交换机维护一个流表,记录一个流的出口选择.为了能够对一个大流进行重路由,Conga 使用了流分段(flowlet)的概念,即如果一个流的两个连续报文到达时间差较大(如 500μs),则可以认为前后两个报文分属于两个流分段,对两个流分段分别路由不会引起分组乱序.这就需要流表中的表项具有很短的超时时间.注意,流分段与前面介绍的流切片有所不同.由于流分段之间不存在分组乱序,所以完全可以视为两个不同的流.

作为基于探测的方案,Conga 可以较好地应对突发拥塞.然而,Conga 方案主要为两层交换机构建的数据中心网络设计,无法直接应用于三层交换机构建的数据中心网络,原因有三:第一,多级选路导致路径与出口的对应关系复杂化;第二,随着目的边界交换机数量增多,每个交换机都需要维护更多的信息,面临扩展性问题;第三,使用反向背负的方式难以保证拥塞信息及时扩散到所有必要的地方.

- HULA

为了解决 Conga 存在的问题,有研究者提出了 HULA[18].HULA 把探测与反馈从数据报文中提取出来,由边界交换机单独地周期性发起,避免了反向背负的不确定性.这些探测报文在网络内部以类似于组播的形式复制转发,减少了探测报文本身的开销.此外,在边界交换机上,并不记录去向每一个目的边界交换机在所有端口上的负载信息,而只记录最优的下一跳对应端口的负载信息,从而减少了信息存储的压力.和 Conga 一样,HULA 利用了流分段,为每一个流分段按照最优的下一跳(即负载最低的端口)进行转发.

HULA 在很大程度上继承了 Conga 的思想,极大地提高了可扩展性.然而,由于 HULA 将探测报文单独提取出来,就必须面对探测开销问题.为了尽可能地降低探测开销,探测周期设置为 1ms,这就约等于几个 RTT 的时间,可认为与 Conga 近似.然而相比之下,Conga 的负载信息更新可以是循序渐进的,而 HULA 则具有跳跃性.因此,采用 HULA 时路由振荡的可能性更大.

#### 3.3.2 基于终端的探测与调整

- FlowBender

FlowBender[19]是一种基于终端的探测与调整方案.其思想非常简单,即在流预期即将遇到拥塞(通过 ECN 通告)时,交换机将对流进行重路由.为了达到这一目的,FlowBender 利用了 IP 报文头中的 TTL 字段.对一个 TCP

流,当主机检测到丢包或 ECE 反馈比例超过预设阈值时,主机将修改发送出去的 IP 报文中的 TTL.交换机在计算每一个报文的转发路径时,在传统的五元组之外,把 TTL 值也加入进来.这样,终端修改 TTL 后,交换机就会为后续的报文(有较大概率地)选择不同的路径转发.

FlowBender 在探测与调整方式上与 Conga 和 HULA 类似,都是根据之前一段时间的负载信息来进行之后的路径选择,其区别在于,Conga 和 HULA 是为每一个新的流分段确定性地选择负载最小的路径,而 FlowBender 则是为一个遇到拥塞的流随机性地重选路径.和 Conga 一样,FlowBender 可以较好地应对突发流量.不过由于选路的盲目性,当拥塞不可避免时,FlowBender 会出现路由振荡无法收敛的情况.此外,由于 FlowBender 的探测是跟随数据本身的,因此对于突发式的开关(on-off)流并没有较好的解决效果.

- MPTCP

MPTCP[20]是 TCP 协议的扩展,运行在主机协议栈中.MPTCP 不改变操作系统的系统调用,因此对应用而言是完全透明的.MPTCP 将一个 TCP 连接分成了多个不同端口(或地址***)的子流(subflow),每一个子流可以采用不同的端口,因此可以通过不同的路径(网络采用 ECMP).这样,即使某个子流经过的路径由于流量不均衡发生了拥塞,只要有其他子流经过的路径没有拥塞,TCP 连接的总体吞吐率就能得到保证.

MPTCP 完全不需要网络进行升级,在一个可以控制主机操作系统的数据中心中,是部署成本最低的方法.然而,MPTCP 也有很大的局限.首先,MPTCP 对子流的管理控制(主要是多子流数据的合并)会引入不小的额外计算开销[21];其次,MPTCP 的启用需要协商,如果对端不支持 MPTCP 则会增加额外延时;再次,MPTCP 会破坏 TCP 原有的公平性[22];最后,MPTCP 中的每个子流还是可能会遇到不均衡导致的拥塞,因此对小流而言,采用 MPTCP 的意义不大[19].这些问题都限制了 MPTCP 的实际部署.

### 3.4 其他方案

- Fastpass

Fastpass[23]是一个集中化的方案,不过这里并非直接对流进行调度.Fastpass 引入了时间槽的概念(一般是发送一个报文所用的时间,μs 级别),由一个逻辑上集中的仲裁器为所有的流分配发送报文的时间槽,保证每一条链路在同一个时间槽内,只有 1 个报文能通过.

Fastpass 是一个很复杂的设计,要求网络与主机协议栈同时修改.Fastpass 通过在主机协议栈与网卡间增加一层代理,来处理主机与仲裁器的交互.Fastpass 并不需要网络来预估流量需求,而是通过主机直接汇报(利用 send 这一系统调用).如果仲裁器在当前时间槽内允许一个流发送报文,则会把计算好的路径一并返回给此流的发送主机,主机使用这一路径发送报文.反之,如果仲裁器不允许一个流发送,此流的发送主机就不会发送任何报文.

由于 Fastpass 把多余的流量阻止在网络之外,因此网络内不会有任何多余的流量,即所有的队列都为 0.因此,主机可以关闭 TCP 拥塞控制,从而提高传输效率,也可以降低 CPU 的负载.Fastpass 的仲裁器通过主机汇报丢包来判断设备失效,因为在没有拥塞的情况下,丢包可以认为就是设备失效的结果.由于是集中化的处理方案,Fastpass 仲裁器在识别出设备失效后,会相应地少分配发送报文,并且在路径选择时避开失效设备.

Fastpass 在理论上完全解决了流量均衡问题.然而其缺陷也很明显.首先,集中化的设计必然要面临扩展性问题,对于 Fastpass 这样时间槽粒度下(μs 级)的调度,其面临的扩展性问题更加严峻.其次,由于所有的发送都需要仲裁器的授权,因此引入的额外延时是不可避免的.最后,Fastpass 的设计对全局时钟同步要求很高,而这对于大规模数据中心而言本身就是一个挑战.

- DeTail

DeTail[24]是一个多层间协作方案.首先,在链路层,DeTail 要求使用优先级流控(PFC),通过以太网的暂停帧(pause frame)来实现无损的二层网络.然后,网络层使用基于分组的切片随机散射.由于底层网络是无损的,传输层就不会视分组乱序为丢包,因此极大地简化了分组乱序处理问题.而应用层则需要给流指定优先级,以便使用优先级流控.

*** 只有在终端主机存在多宿主(multihoming)的情况下,才可以使用地址作为子流的区分.

DeTail 方案中也用到了分组散射,但是这种散射之所以可行,是因为整个网络从上到下都做出了很多改变.由于链路层优先级流控所使用的暂停帧存在固有的不公平问题,且需要应用层的参与,这就导致 DeTail 在实际中的应用受到限制.

## 4 各方案对比与待解决的问题

上一节介绍了 10 种具有代表性的流量均衡方案.每一种方案都有特定的优点与缺点.由于没有一种方法在所有情况下都优于其他方案,因此目前还没有形成统一的流量均衡解决方案.表 1 展示了这些方案在应对突发拥塞的能力、引入的分组乱序(与终端处理复杂度相关)和需要的修改这 3 个方面的对比.不同的数据中心网络可以根据实际需求选择合适的方案.

**Table 1** Comparison of traffic balancing solutions in data center networks

**表 1** 数据中心网络流量均衡方案对比

| 方案 | 应对突发拥塞的能力 | 引入的分组乱序 | 需要的修改 |
|---|---|---|---|
| Hedera | 弱 | 少 | 网络 |
| DRB | 强 | 较多 | 网络+主机 |
| DRILL | 强 | 较多 | 网络+主机 |
| Presto | 强 | 多 | 网络+主机 |
| Conga | 中等 | 无乱序 | 网络 |
| HULA | 中等 | 无乱序 | 网络 |
| FlowBender | 中等 | 较少 | 主机 |
| MPTCP | 中等 | 无乱序 | 主机 |
| Fastpass | 无拥塞 | 无乱序 | 网络+主机 |
| DeTail | 强 | 多 | 网络+主机 |

尽管数据中心网络中的流量均衡已经得到了较为充分的研究,但是有以下两个问题依然没有解决,这会成为数据中心网络流量均衡的未来研究方向.

(1) 瞬时性突发不均衡.即使是探测与调整方案也很难处理 1 个~2 个 RTT 时间内(次 ms 级)的突发拥塞所伴随的流量不均衡.而为了解决这种瞬时性的突发不均衡,目前只能采用特定的切片散射方式.但切片散射方式又会引入分组乱序等问题.因此,针对瞬时性突发不均衡还有待研究.

(2) 流量区分均衡.现有的流量均衡机制对不同流量并无特别区分,会把对丢包有不同要求的流量混合在一起处理.事实上,数据中心中的流量根据所属应用的不同,其对丢包的容忍程度也是不同的(一般而言,小流更不能容忍丢包).在维持总丢包不变的情况下,适当地增加高丢包容忍度的流的丢包,而减少低丢包容忍度的流的丢包,是流量均衡方案需要考虑的问题.

## 5 结 论

本文对数据中心网络中的流量均衡方案做了详细的分析.首先定义了数据中心网络流量均衡问题的目标,即等价链路中的最大潜在丢包率最小化,然后对数据中心网络流量均衡面临的两个重要挑战(分组乱序、突发拥塞)做了说明.之后,本文将流量均衡方案总结为 4 大类:主动调度、切片散射、探测与调整及其他.本文对当前主要的 10 种流量均衡方案(Hedera,DRB,DRILL,Presto,Conga,HULA,FlowBender,MPTCP,Fastpass 以及 DeTail)一一做了分析说明.最后,本文对这些方案做了对比,并提出了两个未来可能的研究方向:瞬时性突发不均衡与流量区分均衡.

**References**:

[1] Introducing data center fabric, the next-generation facebook data center network. 2014. https://code.facebook.com/posts/360346274145943/introducing-data-center-fabric-the-next-generation-facebook-data-center-network

[2] 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, Hölzle U, Stuart S, Vahdat A. Jupiter Rising: A decade of clos topologies and centralized control in Google's datacenter network. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2015. 183−197. [doi: 10.1145/2785956.2787508]

[3] Chiesa M, Kindler G, Schapira M. Traffic engineering with equal-cost-multipath: An algorithmic perspective. In: Proc. of the Int'l Conf. on Computer Communications. Piscataway: IEEE, 2014. 1590−1598. [doi: 10.1109/INFOCOM.2014.6848095]

[4] Al-Fares M, Loukissas A, Vahdat A. A scalable, commodity data center network architecture. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2008. 63−74. [doi: 10.1145/1402958.1402967]

[5] Gill P, Jain N, Nagappan N. Understanding network failures in data centers: Measurement, analysis, and implications. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2011. 350−361. [doi: 10.1145/2018436.2018477]

[6] Guo C, Yuan L, Xiang D, Dang Y, Huang R, Maltz D, Liu Z, Wang V, Pang B, Chen H, Lin Z, Kurien V. Pingmesh: A large-scale system for data center network latency measurement and analysis. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2015. 139−152. [doi: 10.1145/2785956.2787496]

[7] Zhou J, 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. New York: ACM, 2014. 1−14. [doi: 10.1145/2592798.2592803]

[8] Leung K, Li K, Yang D. An overview of packet reordering in transmission control protocol (TCP): Problems, solutions, and challenges. IEEE Trans. on Parallel Distribution System, 2007,18(4):522−535. [doi: 10.1109/TPDS.2007.1011]

[9] He K, Rozner F, Agarwal K, Felter W, Carter J, Akella A. Presto: Edge-Based load balancing for fast datacenter networks. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2015. 465−478. [doi: 10.1145/2785956.2787507]

[10] Alizadeh M, Greenberg A, Maltz D, Padhye J, Patel P, Prabhakar B, Sengupta S, Sridharan M. Data center TCP (DCTCP). In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2010. 63−74. [doi: 10.1145 /1851182.1851192]

[11] Al-Fares M, Radhakrishnan S, Raghavan B, Huang N, Vahdat A. Hedera: Dynamic flow scheduling for data center networks. In: Proc. of the Conf. on Networked Systems Design and Implementation. Berkeley: USENIX, 2010. 19−19.

[12] Cao J, Xia R, Yang P, Guo C, Lu G, Yuan L, Zheng Y, Wu H, Xiong Y, Maltz D. Per-Packet load-balanced, low-latency routing for clos-based data center networks. In: Proc. of the Conf. on Emerging Networking Experiments and Technologies. New York: ACM, 2013. 49−60. [doi: 10.1145/2535372.2535375]

[13] Ghorbani S, Godfrey B, Ganjali Y, Firoozshahian A. Micro load balancing in data centers with DRILL. In: Proc. of the Workshop on Hot Topics in Networks. New York: ACM, 2015. 17:1−17:7. [doi: 10.1145/2834050.2834107]

[14] Greenberg A, Hamilton J, Jain N, Kandula S, Kim C, Lahiri P, Maltz D, Patel P, Sengupta S. VL2: A scalable and flexible data center network. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2009. 51−62. [doi: 10.1145/1592568.1592576]

[15] Dixit A, Prakash P, Hu Y, Kompella R. On the impact of packet spraying in data center networks. In: Proc. of the Int'l Conf. on Computer Communications. Piscataway: IEEE, 2013. 2130−2138. [doi: 10.1109/INFCOM. 2013.6567015]

[16] Guo C, Wu H, Deng Z, Soni G, Ye J, Padhye J, Lipshteyn M. RDMA over commodity ethernet at scale. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2016. 202−215. [doi: 10.1145/2934872.2934908]

[17] Alizadeh M, Edsall T, Dharmapurikar S, Vaidyanathan R, Chu K, Fingerhut A, Lam V, Matus F, Pan R, Yadav N, Varghese G. CONGA: Distributed congestion-aware load balancing for datacenters. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2014. 503−514. [doi: 10.1145/2619239.2626316]

[18] 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. New York: HULA, 2016. 1−12. [doi: 10.1145/2890955.2890968]

[19] 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 on Conf. on Emerging Networking Experiments and Technologies. New York: ACM, 2014. 149−160. [doi: 10.1145/2674005.2674985]

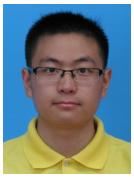
[20] Raiciu C, Barre S, Pluntke C, Greenhalgh A, Wischik D, Handley M. Improving datacenter performance and robustness with multipath TCP. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2011. 266–277. [doi: 10.1145/2018436.2018467]

[21] Raiciu C, Paasch C, Barre S, Ford A, Honda M, Duchene F, Bonaventure O, Handley M. How hard can it be? Designing and implementing a deployable multipath TCP. In: Proc. of the Conf. on Networked Systems Design and Implementation. Berkeley: USENIX, 2012. 399–412.

[22] Khalili R, Gast N, Popovic M, Upadhyay U and Boudec J. MPTCP is not Pareto-optimal: Performance issues and a possible solution. In: Proc. of the Int'l Conf. on Emerging Networking Experiments and Technologies. New York: ACM, 2012. 1–12. [doi: 10.1145/2413176.2413178]

[23] Perry J, Ousterhout A, Balakrishnan H, Shah D, Fugal H. Fastpass: A centralized "zero-queue" datacenter network. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2014. 307–318. [doi: 10.1145/2619239.2626309]

[24] Zats D, Das T, Mohan P, Borthakur D, Katz R. DeTail: Reducing the flow completion time tail in datacenter networks. In: Proc. of the Conf. on Special Interest Group on Data Communication. New York: ACM, 2012. 139–150. [doi: 10.1145 /2342356.2342390]



**李兆耕**(1989－),男,山东嘉祥人,博士生,主要研究领域为数据中心网络,内容中心网络.

**毕军**(1972－),男,博士,研究员,博士生导师,CCF 杰出会员,主要研究领域为软件定义网络,内容中心网络,互联网源地址验证.