

一种多出口环境下的 HTTP 流量细粒度调度方法

吴家顺¹

WU Jiashun

摘要

DNS 进行多出口流量调度时, 常常以链路质量和链路带宽为依据进行引流, 但忽略了到 CDN 节点的链路质量和 CDN 的服务质量。为解决 DNS 在进行 HTTP 流量调度时随机性而引发的问题, 提出一种多出口环境下的 HTTP 流量细粒度调度方法, 并设计实现了 HttpTinyDispatcher 系统。首先, 从出口网络上采集与 HTTP 协议相关的数据包, 并搜集本地 DNS 域名解析缓存记录, 从出口路由器上面搜集 IP 归属地信息, 然后分析数据包, 选择 TCP 握手时延、HTTP 时延以及网页轮廓数据作为特征, 学习当前网络的特点, 确定调度参数, 并使用自动化的动态模拟分析进行不同链路的测试, 最终将流量调度到较低时延的链路上。通过实际出口网络测试, 发现了网络中存在的 HTTP 流量调度问题, 着重解决了超时、HTTP 响应慢等问题, 对需要调度的 HTTP 数据, 实际调度成功比率达到了 25%。

关键词

HTTP 流量调度; 多出口网络; DNS; 流量分析

doi: 10.3969/j.issn.1672-9528.2024.09.007

0 引言

随着互联网技术的日新月异, 网络已经成为大众日常生活中的一种必需品。高校用户作为网络的中坚力量之一, 具有形式多样的网络需求, 对网络带宽和网络服务质量提出了较高的要求。然而在综合带宽成本和线路质量的背景下, 高校往往会部署多个运营商的出口链路, 以这种性价比较高的方式来解决用户需求与带宽、链路质量方面存在的问题。在多出口链路中, 用户访问外网的流量路径选择、出口链路负载均衡等是一些迫切需要解决的问题, 在没有流量调度策略的干预下, 极易造成出口负载失衡, 部分链路极其拥塞, 而部分链路却相当空闲。

Web 是目前用户使用最多的基于 HTTP 协议的应用, 也自然成为用户判定网络好坏的一个试金石, 即使在移动应用领域, 大量的 APP 仍使用 HTTP 协议进行数据的交互。由此可见, 在多出口网络中, 实现网络负载均衡, 并进一步实现 HTTP 流量的细粒度调度, 对提升用户体验质量 (quality of experience, QoE) 具有重要的意义。

相关学者在改善多出口网络性能方面做了大量的研究。文献 [1] 和文献 [2] 提出了一种在出口网关部署负载均衡器的方法, 其基本原理是读取各个出口链路的负载情况, 然后将数据包从负载较小的链路转发出去。这种策略实现了线路的负载均衡, 但不能解决数据包最优选路问题, 即转发出去的

数据包在外部网络存在绕多个运营商线路的问题, 忽略了用户的网络使用体验。文献 [3] 提出一种基于策略路由多出口流量调度方法, 根据划分的内网区域, 将不同区域的数据包转发到不同的出口链路上。这种策略会造成不同区域的用户网络体验不一致, 且链路负载、数据包路径优化问题均未予以考虑。文献 [4] 提出了一种基于用户自主选择路由策略的流量调度方式, 即网络管理员设定若干基于目的 IP 地址的路由表, 由用户通过 Web 界面, 设置并决定所用对外通信时使用其中的哪一个路由表。文献 [5-7] 提出一种基于 DNS 的多出口流量调度方案, 当前越来越多的网络内容开始具有 CDN 节点, 即被访问内容在不同运营商链路上都有对应的 CDN 节点, 并且支持基于请求 IP 的 DNS 智能解析应答。在这一基础之上, 综合考虑出口链路当前流量、预测的流量变化趋势以及链路的服务质量进行调度, 根据调度结果返回域名在对应运营商的 IP 地址。

在 CDN 节点的支持下, 基于 DNS 的多出口流量调度方法目前得到了广泛的应用。该方法一方面最大化发挥了不同质量的出口链路的作用, 另一方面也提升了用户网络体验质量, 但目前仍存在如下一些问题: 虽然有 CDN 技术的支持, 但不同链路上的 CDN 节点到用户的时延以及服务质量是不一致的, 现有的流量调度存在一定的随机性, 不能实现调度、反馈的闭环持续优化; 部分资源可能仅在一个运营商处有 CDN 节点, 若被调度到其他运营商进行域名解析, 解析结果并不是最优结果; 部分 CDN 节点不稳定, 容易发生故障,

1. 南京理工大学信息化建设与管理处 江苏南京 210094

使得一旦使用某链路,将发生服务不可达的情况。在实际操作中,管理人员往往在用户进行报障时才能被动调整 DNS 解析策略,在成千上万的域名中标记这些异常数据并进行主动的人工干预难度非常大,需要一种自动化的、高效的算法来完成这一工作。

为了解决上述问题,本文在文献[4-5] DNS 多出口流量调度的基础上进一步研究,提出了一种多出口环境下的 HTTP 流量细粒度调度方法,来解决 DNS 调度、CDN 服务质量引发的用户体验下降问题。

1 方法设计与实现

研究发现,影响 Web 页面展现的因素有多个方面,既有网络层面因素,也有设计层面的因素,甚至用户电脑以及服务器当前的状态都会影响 Web 浏览的体验。本文从网络交互数据入手,分析用户使用网络产生的数据包,支撑定位故障原因,对进一步细粒度的调度 HTTP 流量起着至关重要的作用。

1.1 系统框架设计

本文提出的一种多出口环境下的 HTTP 流量细粒度调度方法框架结构如图 1 所示,由用户数据流量采集、域名解析缓存记录采集、IP 归属地采集、数据特征选择、学习评价算法、动态模拟分析、DNS 以及出口路由器接口等模块组成。

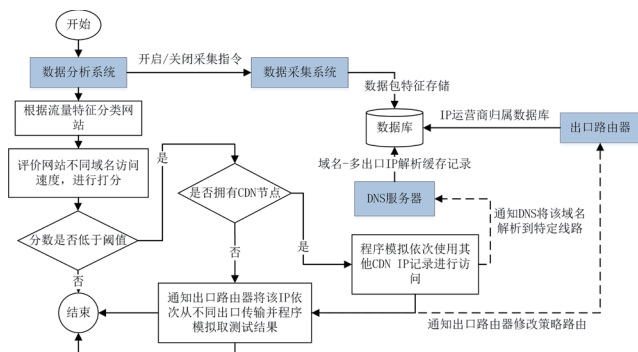


图 1 系统框架图

本框架通过信息采集、反馈、命令执行,将各个独立运行的网络设备转变为物联网中的一个节点,进一步实现网络设备间的信息感知、互联互通。

1.2 系统实现

本文在 Python 环境下实现了此系统,命名为 Http Tiny Dispatcher,通过使用 scrapy、scrapy_http、paramiko 等开源组件实现协议的解析、自动化测试等基础功能的构建,完成整个系统,具体模块实现如下。

1.2.1 用户数据流量采集

本文使用 TCPDUMP 工具进行数据报文的筛选与保存,设置如下筛选条件:发往 80 端口 TCP 首次握手包,即 SYN

包;从 80 端口发出的 TCP 握手响应包,即 SYN+ACK 包;发往 80 端口的 HTTP 请求,即 GET 包;从 80 端口发出的首个 HTTP 应答包。

1.2.2 域名解析缓存记录采集

从特定区域用户上网行为具有局部规律性这一社会工程学特征出发^[8-9],本文从本地 DNS 服务器中读取 DNS 解析的多出口缓存记录来表达该区域内的用户上网行为,缓存记录即同一个域名在不同链路运营商处解析出来的 CDN 节点 IP 信息,记录格式如下:

$$r = (\text{req}_{\text{domain}}, \text{ctc}_{\text{ip}}, \text{cucc}_{\text{ip}}, \text{cmcc}_{\text{ip}}, \text{cernet}_{\text{ip}}) \quad (1)$$

式中: $\text{req}_{\text{domain}}$ 表示被请求解析的域名, ctc_{ip} 表示 $\text{req}_{\text{domain}}$ 在中国电信链路且由中国电信 DNS 解析出来的 IP 地址, cucc_{ip} 表示 $\text{req}_{\text{domain}}$ 在中国联通链路且由中国联通 DNS 解析出来的 IP 地址, cmcc_{ip} 表示 $\text{req}_{\text{domain}}$ 在中国移动链路且由中国移动 DNS 解析出来的 IP 地址, $\text{cernet}_{\text{ip}}$ 表示 $\text{req}_{\text{domain}}$ 在教育网链路且由教育网 DNS 解析出来的 IP 地址。

1.2.3 IP 归属地采集

部分资源可能仅在一个运营商处有 CDN 节点,若被调度到其他运营商进行域名解析,解析结果并不是最优结果,因此切换到正确的运营商处进行解析是解决此问题的方法。通过采集出口路由器内置的 IP 运营商归属数据并结合公开的 IP 归属地查询系统,形成 IP 归属数据库,对比分析各个链路返回的域名解析结果 IP 地址是否属于各自运营商,对结果进行标记。

1.2.4 数据特征选择

通过解析 1.2.1 节采集的数据,选择并抽取可用的特征,本文选取了 TCP 握手时延、HTTP 时延、网页轮廓等特征来表达特定网页的画像。

TCP 握手时延: TCP 数据包是 Web 网站、应用数据传输的重要载体。在 TCP 协议进行数据传输之前,需要进行 TCP 三次握手,本文把从客户端发出 SYN 报文到接收到服务器响应的 SYN+ACK 报文之间时间间隔(两次握手)称为握手时延,记为 ΔTCP 。 ΔTCP 与具体 web 应用无关,更多的是反映端到端链路质量情况。

HTTP 时延: 从客户端发出 HTTP 请求到接收到首个 HTTP 应答的时间间隔,称为 HTTP 请求应答时延数据,记为 ΔHTTP 。 ΔHTTP 不仅包含了页面响应时间,也包括了链路质量时延。

网页轮廓: 一个网页通常由 html 页面、若干 css 文件、js 脚本、图片、音视频等元素组成,每一个元素都是客户端通过 url 从服务器获取,从 HTTP 协议层面分析,访问 html 页面中的每个子元素或者点击链接跳转到其他页面,HTTP 请求中都会用 Referer 字段来标记这个元素是哪个页面链接

过来的。本文以 HTTP 协议中的 (url、host、referer) 三元组来表示一个元素的信息, 具体对于一个网页而言, 是由多个 referer 值等于 html 页面 url 地址的元素组成。

综上所述, 本文选取的页面元素特征表达式为:

$$(destip, url, host, referer, \Delta TCP, \Delta HTTP, time) \quad (2)$$

式中: destip 表示访问的服务器地址, url 为访问的目标链接, host 为目标主机的域名, referer 表示元素是哪个页面链接过来的, ΔTCP 是链路时延, $\Delta HTTP$ 是页面响应时延, time 为访问时间戳。

1.2.5 学习评价算法

在对用户体验质量 (QoE) 测量方法上, 通常可以采用主动测试和被动监测。

被动监测: 从网络层面采集全网用户的网络数据包, 获取到的数据具有天然的大数据特征。

主动测试: 通过脚本或者程序记录页面中每个元素从请求到加载完成所耗的时间, 优点是测试环境完全可控, 但缺点是采集的样本点少。

本文通过被动监测获取数据, 进行数据的分析学习以获得网络现状, 通过正态分布模型来评估用户可接受的网络波动区间, 具体算法描述如下。

设置 host、destip 为分组条件, 对页面元素特征表记录进行分组:

$$H = (host_1, host_2, \dots, host_i, \dots, host_n) \quad (3)$$

$$host_i = (ele_{i1}, ele_{i2}, \dots, ele_{ij}, \dots, ele_{in}) \quad (4)$$

式中: H 表示分组集合, $host_i$ 表示第 i 个分组, ele_{ij} 表示 $host_i$ 分组中第 j 个元素。

对于 $host_i$, 分别升序排列集合中 ΔTCP 和 $\Delta HTTP$ 的值, 并根据公式 (5) 计算中位数, 结果分别记为 $M_{i-\Delta TCP}$ 和 $M_{i-\Delta HTTP}$ 。

$$M_e = f(x) = \begin{cases} x_{\frac{n+1}{2}}, & n \text{ 为奇数} \\ \frac{x_{\frac{n}{2}} + x_{\frac{n}{2}+1}}{2}, & n \text{ 为偶数} \end{cases} \quad (5)$$

$M_{\Delta TCP}$ 与 $M_{\Delta HTTP}$ 是连续随机变量, 正常状态下服从正态分布, 其公式为:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (6)$$

高时延和极小时延数据包在正常的网络中是一种小概率事件 (发生概率小于 5%), 结合假设检验的基本思想, 本文将以外的数据认为是不应该发生的事情, 进而将符合条件的流量数据列入分析调整集合中。

1.2.6 动态模拟分析

对于从 1.2.5 节中获取到的待分析调整对象, 进行测试从其他链路加载能否有较好的结果, 通过查询 1.1.2 节中记录

的解析记录, 使用脚本构建 Socket 依次连接到不同的 IP 地址, 记录 TCP 三次握手时延信息, 并计算出两次握手时延数据, 记为 ΔTCP_t ; 在 TCP 已经建立的基础上, 构建 HTTP 请求数据, 通过 Socket 发送出去, 计算发送请求到获取到 Web 服务器应答的时间间隔, 记为 $\Delta HTTP_t$; 分析需进行多次模拟, 取中位数作为结果, 对于满足要求的结果, 根据不同链路的当前负载, 将此域名调度到负载较轻的链路上。

1.2.7 DNS 以及出口路由器接口

本文中的调度方法在动态模拟分析阶段, 需要与 DNS 和路由器进行交互, 而 DNS 和路由器一般并未提供 API 供外部调动, 因此需要通过脚本模拟执行 shell 指令来控制 DNS 和路由器完成一些配置修改策略。

2 实验与数据分析

为了验证本文所提方案实际可用性以及产生的效果, 本文选取了南京理工大学实际出口网络进行了验证, 其网络拓扑如图 2 所示, 并接入了本文的数据采集分析系统。

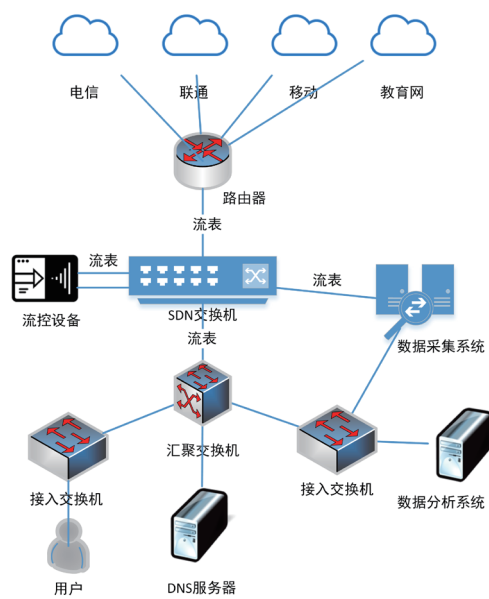


图 2 网络拓扑结构

图 3 是计算得到的当前网络 HTTP 流量的 TCP 和 HTTP 时延分布, 可以看出时延分布呈现出多个波峰。

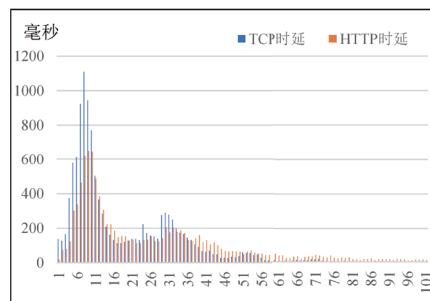


图 3 网络时延数据

针对图 3 数据，取出不同波峰里面的数据从运营商归属和地域位置进行了分析研究，分析结果如下：不同波峰中数据并未有明显分层次的运营商归属；不同波峰中数据具有比较明显的空间距离归属，如表 1 所示，即时延分布与 CDN 地域分布是强关联性，与运营商链路是弱关联性，而 CDN 节点并不是随机分布的，一般会集中在大城市或者数据交换节点城市中，故图 3 可以理解为不同地域空间距离上的正态分布叠加数据。根据数据，计算各自的均值和方差，并设置当前网络的 TCP 时延调度值为 60 ms，HTTP 时延调度值为 70 ms。

表 1 时延与地域关联信息

TCP 时延均值 /ms	地域	流量占比 /%
10	江浙沪	63.2
30	北京、广州、深圳等骨干节点	25.3
50	国内其他区域	6.4

表 2 分析比较了调度前后的 TCP 超时和 HTTP 应答超时数据占总请求的比率，经过调度后，超时的 TCP 连接和 HTTP 应答分别降低了 10% 和 25%。

表 2 超时请求调度前后占比数据

类别	调度前	调度后
TCP 超时连接占比 /%	3.9	3.5
HTTP 应答超时占比 /%	0.8	0.6

表 3 分析比较了调度前后 TCP 连接缓慢以及 HTTP 应答缓慢占总请求的比率数据，经过调度后，连接以及应答缓慢的数据分别降低了 23.5% 和 29.8%，未能调度的连接与请求在其他链路并未发现更好的低时延，故保持现状。

表 3 请求缓慢调度前后占比数据

	调度前	调度后
TCP 连接缓慢占比 /%	1.19	0.91
HTTP 应答缓慢占比 /%	1.44	1.01

在通过本文所设计的系统分析调整后，TCP 以及 HTTP 时延数据如图 4 所示，可以看出高时延区域的数据得到了一定的减少，靠近前部的波峰数据有所增加，同时 TCP 连接超时、HTTP 应答超时等问题也得到了一定的优化。

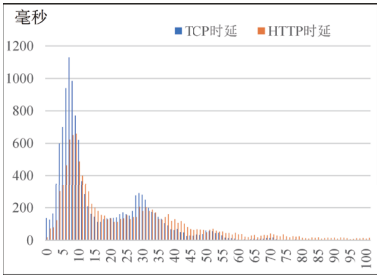


图 4 调度后 TCP 时延和 HTTP 时延数据

3 结论

针对多出口环境下链路负载均衡问题，基于 DNS 的流量调度发挥了很好的作用，但根本上是得益于不同运营商内部 CDN 节点的建设，本文提出的一种多出口环境下的 HTTP 流量细粒度调度方法通过数据监测、多 CDN 服务情况对比，解决了 DNS 调度流量时忽略端到端服务质量的问题。

本文选取实际网络出口进行了实验分析，结果表明，在当前的 DNS 调度下，存在一定的服务在某条链路中无法建立 TCP 连接或者 HTTP 请求应答缓慢、超时等问题，通过本文所提方法分析后，更换到合适的运营商链路后，问题得到了一定的缓解。

参考文献：

- [1] 梁可结,魏文红,王高才.一种有效的网络多出口流量调度方案[J].计算机工程,2010,36(5):116-118.
- [2] 秦勇.一种基于负载流量矩阵的均衡网关技术研究[J].中山大学学报(自然科学版),2002,41(6):28-31.
- [3] 罗伟雄,时东晓,刘岚.校园网多出口路由优化方案[J].计算机应用,2009,29(S1):41-43.
- [4] 张焕杰,夏玉良.用户自主选择的校园网出口策略路由实现[J].通信学报,2013,34(Z2):14-16.
- [5] 吴家顺,高静.多系统协同的网络负载均衡方法研究[J].中国教育信息化,2016(4):85-89.
- [6] 翁源,黄小红,李丹丹,等.基于 DNS 的网络多出口流量调度方法[J].东南大学学报(自然科学版),2017,47(S1):102-107.
- [7] 严格知,章勇.基于 DNS 的校园网多出口流量调度的构建[J].中国教育信息化,2020(15):83-89.
- [8] 李立耀,孙鲁敬,杨家海.社交网络研究综述[J].计算机科学,2015,42(11):8-21.
- [9] 宋竹,秦志光,罗嘉庆.电信数据中用户行为特征测量与分析[J].电子科技大学学报,2015(6):934-939.

【作者简介】

吴家顺（1984—），男，江苏扬州人，硕士，工程师，研究方向：网络安全、计算机应用技术。

（投稿日期：2024-06-13）