

BBR-TCP 协议实验性能评价^①

李振涛^{1,2}, 任勇毛¹, 周旭¹, 周亚球^{1,2}

¹(中国科学院 计算机网络信息中心, 北京 100190)

²(中国科学院大学, 北京 100049)

通讯作者: 周旭, E-mail: zhouxu@cnic.cn

摘要: 随着网络技术的发展, 核心设备路由器的缓存容量在不断增加, 基于丢包的拥塞控制机制引发了 BufferBloat 和带宽利用率低等问题. 为此, Google 最近提出了一种基于瓶颈带宽和往返时延测量的拥塞控制协议 BBR, 引起了广泛关注. 由于 BBR 新近提出不久, 目前缺乏对其性能评估的工作. 本文通过实验对 BBR 协议进行了比较全面的性能分析和评估, 包括协议的传输效率、收敛性及公平性等, 并在此基础上, 提出了 BBR 协议下一步的改进思路.

关键词: 拥塞控制; 协议; BBR; 性能评价

引用格式: 李振涛, 任勇毛, 周旭, 周亚球. BBR-TCP 协议实验性能评价. 计算机系统应用, 2018, 27(9): 229-235. <http://www.c-s-a.org.cn/1003-3254/6515.html>

Performance Evaluation of BBR-TCP Based on Experiments

LI Zhen-Tao^{1,2}, REN Yong-Mao¹, ZHOU Xu¹, ZHOU Ya-Qiu^{1,2}

¹(Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China)

²(University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: With the development of network technology, loss-based congestion control mechanism has drawbacks because of low bandwidth utilization and BufferBloat problem. A congestion-based congestion control mechanism BBR has been adopted by Google and attracts wide attention. In this study, we evaluate the performance of BBR based on experiment, including protocol's transmission efficiency, convergence, as well as fairness, and provide future improving direction of BBR protocol.

Key words: congestion control; protocol; BBR; performance evaluation

1 引言

随着网络技术的发展, 网络应用场景呈现出多样化和复杂化的特点, 例如数据中心依赖带宽极高延时极低的网络, 监测自然环境的传感器网络带宽十分不稳定, 卫星通信网络则具备高时延特点, 而实际的网络业务可能同时需要多种网络的支撑. 与此同时, 网络核心设备路由器的缓存容量在不断增加, 这意味着路由

器可以缓存待转发数据包不断增加, 这也就是典型的 BufferBloat 问题. 随着终端应用对网络的可靠性和实时性需求的增加, BufferBloat 问题已经影响了网络的性能. 上述客观条件为网络协议的设计带来了挑战, 网络协议对不同场景的适用性不同, 在多种实际场景下的性能有较大的差别. 此外, 不同网络场景下带宽利用率和时延的优先级要求不同, 网络协议的适配性成为

① 基金项目: 国家自然科学基金 (61601443, 61672490, 61602436); 国家科技重大专项 (2017ZX03001015-004); 北京市自然科学基金 (4174110)

Foundation item: National Natural Science Foundation of China (61601443, 61672490, 61602436); National Science and Technology Major Project of China (2017ZX03001015-004); Natural Science Foundation of Beijing Municipality (4174110)

收稿时间: 2018-01-09; 修改时间: 2018-02-06; 采用时间: 2018-02-26; csa 在线出版时间: 2018-08-16

影响网络协议性能的重要因素。

TCP 作为互联网核心协议之一在不断演进,其最重要的一项功能就是对网络进行拥塞控制。随着网络应用场景的多样化和复杂化, TCP 中早期的拥塞控制机制已经不再适用于当前的网络环境,影响了网络的性能。在上述背景下, Google 于 2016 年 10 月提出了一种基于瓶颈带宽和往返时延测量的拥塞控制协议 BBR (Bottleneck Bandwidth and Round-trip propagation time),旨在提高网络链路利用率和降低网络时延^[1], BBR 协议已经在 Google 内部的数据中心广域网 B4 部署使用, B4 分为数据中心内部网络 (IDC Network) 及骨干网 (Backbone Network)^[2]。BBR 协议在 B4 中的性能已经得到了验证,但其应用范围还不够广泛,仅仅局限在谷歌内部,适用性还未被重复证明,其在更为复杂和多样的网络环境中的性能还有待验证。本文通过对 BBR 的测量实验,分析和评估和分析了 BBR 协议的性能并在此基础上提出了下一步的改进方向。

2 背景及相关工作

2.1 BBR-TCP 原理简介

BBR 是 TCP 中的拥塞控制机制,不再把丢包作为拥塞产生的信号,丢包检测基于 RACK^[3],发生丢包后只重传数据包,不再对发送端速率调整。网络链路状态检测是基于实时测量的,相较于基于丢包的反馈,实时测量能更准确地反馈出链路的状态。在 BBR 中,发送端不断地测量网络的传输速率 deliveryRate ,具体方法是用短时间内传输的数据量除以传输时间。发送端维持一个长度为 $10 \times RTT$ 时间窗口 Win_Max ,在 Win_Max 内把测量到的 deliveryRate 最大值近似为网络的瓶颈带宽 $BtlBW$ (Bottleneck Bandwidth)。当经过一个 Win_Max 后,原 $BtlBW$ 过期,重新测量确定 $BtlBW$ 的值。 RTT 也在不断测量,每收到一个数据包就计算该数据包的 RTT 。发送端维持一个长度为 10 s 的时间窗口 Win_Min ,发送端把测量到的最小 RTT 作为网络的传输时间 $RTprop$,如果在 Win_Min 内 $RTprop$ 没有更新,该 $RTprop$ 值过期,然后重新测量确定 $RTprop$;如果在 Win_Min 内 $RTprop$ 未更新,则重置 Win_Min 持续时间即 Win_Min 重新开始计时。通过上述方式,发送端测量得到近似的 $BottleBW$ 和 RTT ,这两个值是拥塞控制的关键参数。

BBR 协议根据测量到的 $BottleBW$ 和 RTT 调整发

送端速率 sendingRate ,传输过程分为四种状态: STARTUP、Drain、ProbeBW 和 ProbeRTT,四种状态分别根据条件触发和退出。BBR 通过 Pacing 方式控制发送速率 sendingRate ,增益系数 gain 作用是调整 Pacing,同时发送窗口的上限设置为带宽时延积与增益系数的乘积,带宽时延积 BDP (Bandwidth Delay Product) 是网络的重要参数,作用是控制网络中数据包的总量,其值为公式 (2) 和公式 (3)。

$$\text{sendingRate} = \text{gain} * BtlBW \quad (1)$$

$$BDP = BtlBW * RTT \quad (2)$$

$$Cwnd = \text{gain} * BDP \quad (3)$$

BBR 协议内公平性是通过 ProbeRTT 状态实现的,在一个 TCP 流进入 ProbeRTT 状态后会减少网络中的数据包,其他的流会更新 RTT 同时重置 Win_Min 窗口, Win_Min 开始重新计时,不同 TCP 流同步数据传输过程,从而实现协议内公平性。BBR 协议在设计上没有考虑与其他 TCP 拥塞控制协议的公平性,协议间公平性评估是本文的研究重点之一。

2.2 相关工作

对 TCP 协议的拥塞控制机制的改进大致可以分为基于丢包的拥塞控制和基于时延的拥塞控制机制^[4,5]。基于丢包的拥塞控制机制相对较多,典型的有 Reno-TCP^[6]、BIC-TCP^[7]和 Cubic-TCP^[8]等。基于时延的拥塞控制机制,典型的有 Vegas-TCP^[9]、FAST-TCP^[10]等。

Reno 是一种基于丢包反馈的拥塞控制机制, Reno-TCP 是目前使用最广泛的 TCP 拥塞控制版本。Reno 协议基于丢包反馈机制和 AIMD 的速率调整算法适用于浅队列表路由器,缺点是无法识别错误丢包和拥塞丢包,在丢包率高的链路上带宽利用率不高。

BIC-TCP 是基于二分搜索思想的拥塞控制机制。BIC-TCP 的缺点首先就是抢占性较强, BIC-TCP 的增长函数在链路带宽小时延短的情况下比起标准的 TCP 抢占性强; BIC-TCP 窗口控制阶段分为 binary search increase, max probing, 然后还有 S_{max} 和 S_{min} 的区分,这几个值增加了算法上的实现难度,同时也对协议性能的分析模型增加了复杂度。

Cubic 协议是对 BIC 的改进,在设计上简化了 BIC-TCP 的窗口调整算法,使用一个三次函数取代 BIC-TCP 的增长曲线。Cubic 最关键的点在于它的窗口

增长函数仅仅取决于连续的两次拥塞事件的时间间隔值,从而窗口增长完全独立于网络的时延 RTT ,从而解决 HSTCP 存在严重的 RTT 不公平性^[11],而 Cubic 的 RTT 独立性质使得 Cubic 能够在多条共享瓶颈链路的 TCP 连接之间保持良好的 RTT 公平性。

Brakmo 和 Peterson 提出了一种用 RTT 测量网络状况的拥塞控制算法,称之为 Vegas-TCP. Vegas 通过 TCP 连接中 RTT 值的改变情况来控制拥塞窗口. Vegas 协议用 RTT 的改变来判断,所以能较好地预测网络带宽使用情况,并且对小缓存的适应性较强,其公平性和效率都较好.但 Vegas 存在公平性问题,在与基于丢包的拥塞控制机制中会处于劣势。

由于 BBR-TCP 是一种新的协议,对于 BBR-TCP 性能评价工作还比较少,尤其缺乏全面系统的性能评价.已有的一些对 BBR-TCP 协议性能评价的工作,评价场景简单、性能指标不够全面.例如,Brakmo 通过发送连续的 RPC 的方式在 Netem 仿真环境下^[12],对 BBR 性能在 Lan 和 Wan 环境下的吞吐率、时延和公平性进行了评估,但网络参数设置的粒度太大,缺乏对实验结果的原因的深入分析.此外,Google 提出一种基于 UDP 的应用层传输协议 QUIC^[13],该协议也采用了和 BBR 类似的 Pacing 机制。

3 实验设计

本文通过 Netem 对网络环境进行仿真,Netem 是一个网络仿真平台^[14],通过配置延时、丢包率和队列长度等参数仿真局域网、广域网和高速长距离网络等网络环境.同时通过软件发包方式在网络中添加背景流量,进一步还原了网络场景的真实性.实验场景包括四台服务器和一台交换机,其中一台服务器作为数据接收端 (Client),TCP 配置为默认配置;Server01 和 Server02 部署 BBR 协议,Server03 部署 Cubic 协议.服务器与交换机直连,交换机对网络环境仿真.实验的网络拓扑如图 1. Server 和 Client 采用高配置规格的服务器主机,网络接口为千兆网卡,在实验中服务器负载不高,服务器性能不会成为影响传输性能的瓶颈。

Server 端作为数据发送端通过 Iperf 以 TCP 流的方式向 Client 端发送数据,Iperf 是一个网络带宽测量工具^[15],能够测量网络实时带宽.链路带宽默认为 1Gbps,瓶颈链路为交换机和 Client 之间的链路,通过改变仿真器配置网络参数,仿真多种网络环境,评估 BBR 协议在不同网络环境下的适用性。

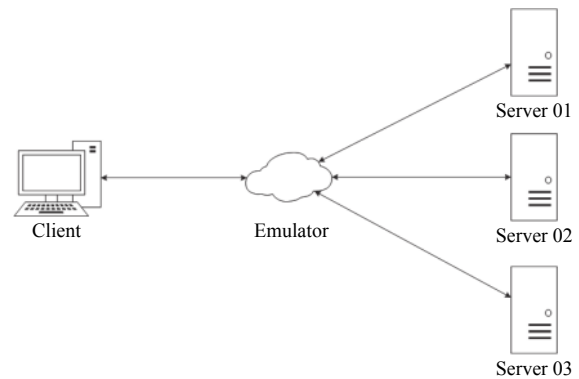


图 1 实验拓扑图

BBR 性能评价实验主要对协议的传输效率、收敛时间和公平性等指标进行评价,与默认的 TCP 拥塞控制协议 Cubic 进行比较.传输效率主要指网络传输速率和稳定性;收敛时间在带宽变化的场景下传输速率收敛到可用带宽的时间;公平性指不同 TCP 流对带宽的利用率公平性,包括协议内公平性和协议间公平性。

4 实验结果及分析

4.1 效率分析

本文首先对 BBR 协议的效率进行评价,在 RTT 为 10 ms,链路丢包率为 1×10^{-12} 的场景下,部署 BBR 协议的 Server01 与 Client 传输数据,传输时间持续 60 s,部署 Cubic 协议的 Server02 也以同样的方式传输 60 s 的数据.传输速率的变化如图 2,在实验结果中 BBR 协议的吞吐率大部分时间稳定在 900 Mbps 左右,实验结果中有几个时刻的传输速率明显小于 900 Mbps,原因是 BBR 每隔 10 s 进入一次 ProbeRTT 状态,该状态中发送端降低发送速率,因此会出现传输速率周期性降低的现象。

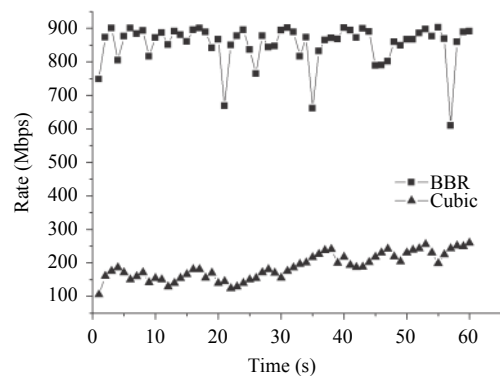


图 2 吞吐率随时间变化图

Cubic 协议的吞吐率在 200 Mbps 上下波动,原因是发送端在不断地进行最优发送窗口的探测,传输速率在最优值上下波动,而该值远低于物理带宽,原因是传输速率受到链路丢包率的影响,Cubic 协议中拥塞窗口最大值与丢包率 p 的平方根成反比^[15].丢包率会限制拥塞窗口的增长,丢包率过高会导致拥塞窗口收敛到一个较小的值,从而限制传输速率.而在操作系统内核对 TCP 拥塞窗口上限进行了限制,拥塞窗口的最大值用 tcp_wmem 参数来表示.Cubic 协议中在传输过程中拥塞窗口最大值 $Cwnd_{max}$ 同时受到丢包率和 tcp_wmem 影响,如公式 (5).

$$Win_{max} \propto \sqrt{1/p} \quad (4)$$

$$Cwnd_{max} = \text{Min}(Win_{max}, tcp_wmem) \quad (5)$$

$$Rate_{cubic} = \frac{1}{RTT} \sqrt{\frac{\alpha}{2} \frac{1+\beta}{1-\beta} \frac{1}{p}} \quad (6)$$

$$Rate_{bbr} = \text{BottleBW} * 0.98 * \min(1, \frac{tcp_rmem}{BDP}) \quad (7)$$

$$T_{probe} = \log_2 BW * RTT \quad (8)$$

Cubic 在稳定状态下传输速率 $Rate_{cubic}$ 如式 (6),其中 α 是 AIMD 模型中控制拥塞窗口线性增长的参数, β 是拥塞窗口乘性减小的参数,根据实验中的操作系统内核版本¹, α 的值为 1, β 的值为 0.8,带入式 (7) 可得 $Rate_{cubic} \approx 212$ Mbps,基本接近上述实验结果.BBR 在稳定状态下传输速率如式 (7),其中 0.98 含义是发送端有 98% 的时间完全利用带宽,而 2% 的时间处于 ProbeRTT 状态,ProbeRTT 状态下传输速率接近于零.在 RTT 为 10 ms, BandWidth 为 1 Gbps 的场景下, BDP 为 1025 KB,而 tcp_rmem 初始设置为 1000 KB,根据式 (7),理想状态下 $Rate_{bbr} \approx 960$ Mbps,上述实验结果低于这个值,但基本符合实际情况.上述实验其他条件保持不变,改变 RTT,分别测量 BBR 和 Cubic 传输速率,实验结果如图 3. Cubic 的传输效率在随着 RTT 变大而变低,原因是 Cubic 协议的传输速率与 RTT 成反比,RTT 增加传输速率会降低. BBR 传输速率在 RTT 大于 20 ms 后大幅度下降,原因是 BBR 协

议的传输速率受到对发送窗口的限制, tcp_rmem 的值限制了拥塞窗口的最大值.根据式 (7) 在 $tcp_rmem > BDP$ 的情况下,传输速率只受瓶颈带宽的影响;而在 $tcp_rmem < BDP$ 的情况下,传输速率受到 tcp_rmem 的限制.随着 RTT 增加, BDP 变大, BBR 的传输速率受到 tcp_rmem 影响,所以当网络的 RTT 大于 20 ms 后,传输速率下降.为了验证该结论,把 tcp_rmem 设置为 80 MB,此时 tcp_rmem 足够大,内核不会成为限制传输速率的因素,在其他条件保持一致的情况下实验结果如图 4.

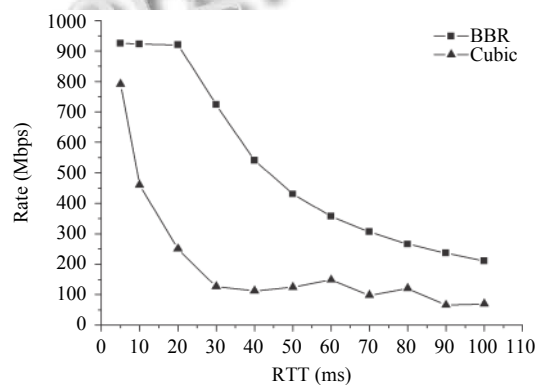


图3 吞吐率随 RTT 变化图

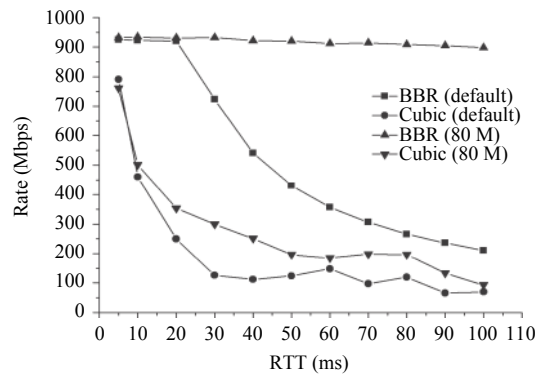


图4 吞吐率对比图

随着 RTT 的变大, BBR 协议传输速率稳定在 900 Mbps 以上,而 Cubic 在改变 tcp_rmem 后传输速率变化不大,说明内核中 TCP 窗口的限制不是影响 Cubic 性能的主要因素,丢包率是影响 Cubic 性能的主要因素,进一步验证了上文的结论.

上述分析中可知 BBR 协议性能受链路丢包率影响不大,而 Cubic 协议性能受丢包率影响较大,为了评价对丢包率对协议性能的影响,在不同丢包率的链路

上对 BBR 和 Cubic 性能进行了测试,为了评价不同的网络类型下丢包率对 BBR 和 Cubic 的影响,分别设置 RTT 为 10 ms、20 ms 和 50 ms 进行了实验,实验结果如图 5。从实验结果可以得出 BBR 对丢包率不敏感,丢包率变化对传输速率影响不大,而 Cubic 对丢包率比较敏感,传输速率随丢包率的增加变化较大,丢包率为 0.1% 的链路传输速率几乎为零。

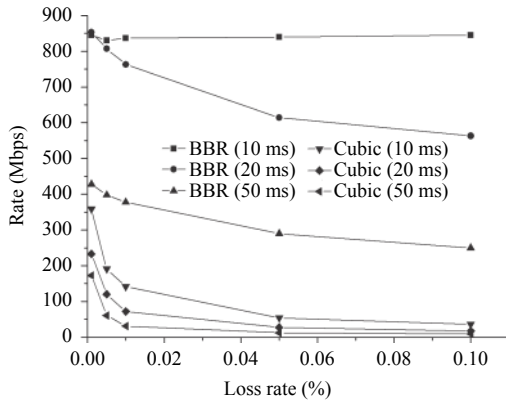


图 5 丢包率对协议性能的影响图

在传输效率上, BBR 协议性较 Cubic 等基于丢包的拥塞控制协议有了提升,主要原因是 BBR 通过自身的机制避免了链路拥塞产生丢包, BBR 对丢包不进行降窗处理,所以 BBR 的传输速率较高。

4.2 收敛性分析

BBR 协议通过 STARTUP 探测到最大的可用带宽,通过 ProbeBW 状态感知带宽的变化。本文对 BBR 协议的收敛性进行了评价,可变带宽通过背景流量的改变实现,背景流量为带宽抢占性很强的 UDP 流量。首先对 BBR 协议在 UDP 背景流量下进行了收敛性测试。首先启动 BBR-TCP 连接,5 s 过后背景流量出现,背景流量持续 10 s。实验结果如图 6, BBR 在 2 s 内完成了收敛,原因是 BBR 采用乘性增长方式探测可用带宽,因此收敛速度较快。

本文对 BBR 协议在可变带宽下的收敛性进行了实验分析,实验结果如图 7。瓶颈链路带宽的变化如图中所示,实验结果中 BBR 协议的收敛时间小于 1s。通过对实验结果分析可知, BBR 协议保证了协议的快速收敛,协议的收敛性较好。

4.3 公平性分析

协议公平性是指不同 TCP 流对带宽的占用率的公平性,是衡量协议性能的重要方面,本文对 BBR 协

议的公平性进行了评估,包括基本公平性、大小流公平性、RTT 公平性、协议间公平性。基本公平性指条件完全相同情况下不同 TCP 流之间的公平性,在基本公平性评价实验中,部署了 BBR 协议的 Server01 和 Server02 同时向 Client 传输 60 s 数据,瓶颈带宽为 1 Gbps,实验结果如图 8。两个 BBR 流的带宽占用率基本相同,但是总的带宽利用率稳定在 750 Mbps 左右,利用率不高,原因是两个 BBR 流竞争瓶颈带宽会导致 RTT 变大,从而导致发送端降低发送速率。

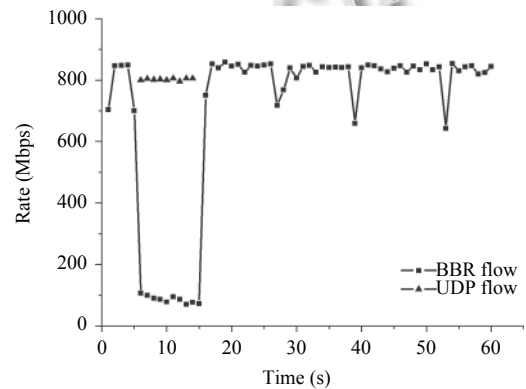


图 6 背景流量下的收敛性

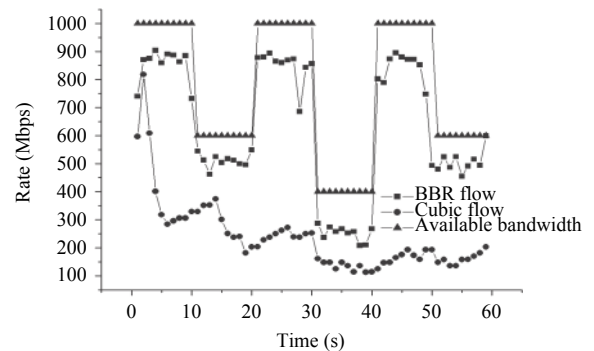


图 7 可变带宽下的收敛性

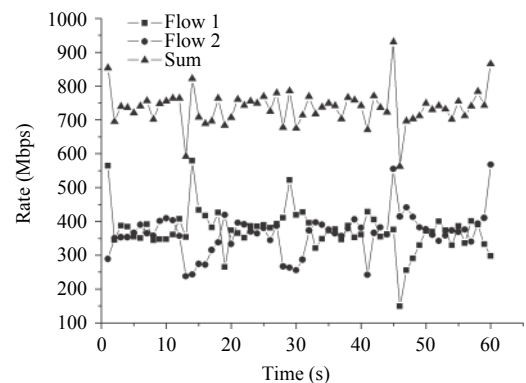


图 8 基本公平性

实际网络中 TCP 流是随机启动的, 本文设计了对随机启动 BBR 流测试实验. 在实验中一个 BBR 流先启动, 5 s 后启动另一个 BBR 流, 每个流持续 60 s. 实验结果如图 9, 两个 BBR 流的传输速率图像基本对称, 平均传输速率非常接近, 两个先后启动的 BBR 流公平性较好, 但还是存在总体带宽利用率不高的问题. 大小流公平性是值持续时间长的 TCP 流和持续时间短的 TCP 流之间的公平性. 本文设计了评估大小流公平性的实验, 其中大流持续 60 s, 小流持续 10 s, 实验结果如图 10. 实验结果表明小流的传输速率稳定在 400 Mbps, 小流的性能没有受到太大影响.

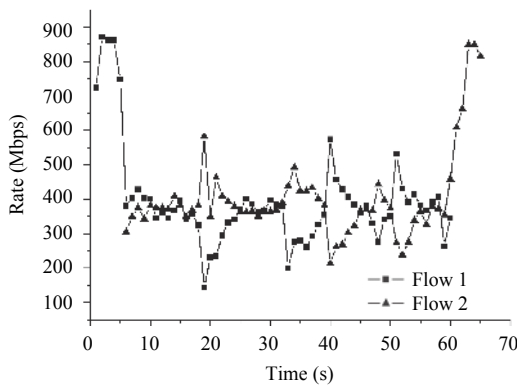


图 9 动态基本公平性

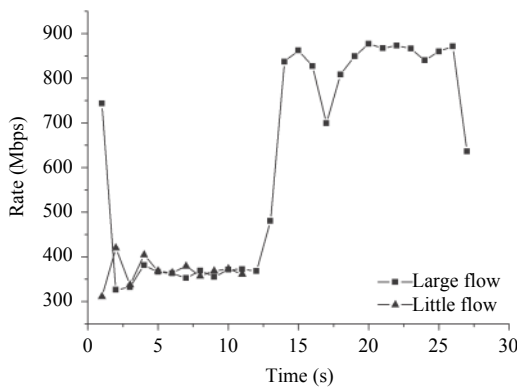


图 10 大小流公平性

RTT 公平性指不同 RTT 的 TCP 流之间的公平性, BBR 协议在设计上有类似慢启动的 STARTUP 状态, 带宽的探测时间 T_{probe} 与 RTT 成正比, 如式 (8). BBR 流的带宽探测速度与 RTT 成反比, RTT 小的 TCP 流探测速度快, 带宽抢占性强, 所以 RTT 小的 BBR 流会占据较多带宽. 本文对 RTT 公平性进行了评估实验, 实验中同时启动 Server01 和 Server02 与

Client 建立 TCP 连接, 两个 TCP 流记为 Flow 1 和 Flow 2. Flow 1 的 RTT 为 10 ms, Flow 2 的 RTT 为 20 ms, Flow 1 传输速率稳定在 600 Mbps, Flow 2 传输速率稳定在 300 Mbps, 符合上述带宽探测速率与 RTT 成反比的结论. 为了进一步验证 BBR 的公平性, 把 Flow 1 的 RTT 固定为 10 ms, 改变 Flow 2 的 RTT, 其他条件保持不变然后进行实验, 实验结果如图 11, 随着 Flow 2 的 RTT 增加, Flow 1 和 Flow 2 的性能差距在变大, 进一步验证了对 BBR 协议中 RTT 公平性分析.

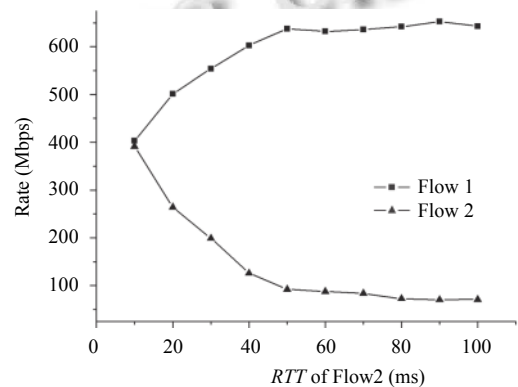


图 11 RTT 公平性扩展测试

协议间公平性是指不同协议间的公平性, 本文中主要是对 BBR 与 Cubic 之间的公平性进行评价. 评价实验分别在 RTT 为 50 ms 的环境下, 部署 BBR 的 Server01 和部署 Cubic 协议的 Server03 同时与 Client 建立 TCP 连接, 两个流分别为 Flow 1 和 Flow 2, 测试两个流的传输速率, 实验结果如图 12. 在 RTT 为 50 ms 的场景下 BBR 流占据了大部分带宽. 随着 RTT 的增加, Cubic 占用的带宽会逐渐增加, 原因是 RTT 增加 BDP 变大, 内核开始限制 BBR 的传输速率, Cubic 的传输速率开始增加, 但 Cubic 在与 BBR 的竞争中处于劣势.

通过对 BBR 公平性评估, 可以看出 BBR 协议在 RTT 公平性和协议间公平性上还存在改进空间.

5 BBR-TCP 协议改进思路

BBR 协议作为一种基于测试的拥塞控制机制在效率方面有了较大的提升, 但在收敛时间和公平性方面还有进一步优化的空间. 基于本文对 BBR 的评估, 我们提出了对 BBR 协议如下的改进思路: (1) 协议在收敛速度上的优化, 由于 ProbeBW 状态中 $pacing_$

$gain=1$ 的 phase 持续 6 个 RTT , 这将导致 BBR 在带宽探测阶段收敛速度偏慢并且可能出现新增加的可用带宽被其他 TCP 流占用的情况, 在下一把的工作中可以通过减少 $pacing_gain$ 为 1 的 phase 数量或者增加 $pacing_gain$ 的值对收敛速度进行优化; (2) TCP 窗口限制优化, 在本文中提到了 TCP 窗口限制了 BBR 的性能, 由于 BBR 对传输效率的大幅度提升, TCP 窗口上限成为了现在 BBR 协议的因素, 可以考虑适当扩大 TCP 窗口的上限但同时也要兼顾整个网络的负载量和公平性, 因此下一步可以通过扩大 TCP 窗口的方式对 BBR 协议效率进行优化; (3) BBR 协议的 RTT 公平性和协议间公平性优化, BBR 中带宽的探测过程与 RTT 成反比, 下一步可以通过借鉴 Cubic 中带宽探测与 RTT 无关的优点对 RTT 公平性进行优化; BBR 协议在与 Cubic 的竞争中处于绝对优势, 原因是 Cubic 在丢包后降低发送速率而 BBR 不进行速率调整, 下一步可以考虑通过降低 BBR 带宽占用率的方式对公平性进行改进。

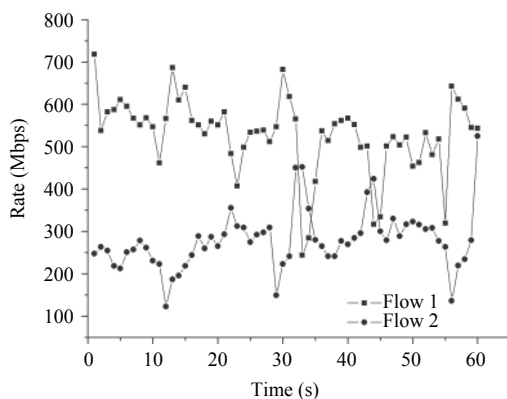


图 12 公平性实时测试

6 总结

本文基于仿真实验对 BBR-TCP 协议的性能进行了比较全面的分析和评估, 包括协议传输效率评价、收敛性评价和公平性评价, 分析了 BBR-TCP 协议较其他传输协议传输效率提升的关键原因, 对其传输速率在不同场景下的性能进行了实验评估。在协议公平性方面, 分析了 BBR 的协议内公平性和协议间公平性存在的问题, 然后通过实验的方式进行了评估。BBR 协议内公平性较好, 但与其他拥塞控制协议如 Cubic 协议

对比时会引发协议间公平性问题。此外, BBR 协议也存在 RTT 公平性较差的问题。在实验评估的基础上, 本文最后对 BBR-TCP 协议进一步改进提出了方向。

参考文献

- Cardwell N, Cheng YC. BBR: Congestion-based congestion control. *ACM Queue*, 2016, 14(5): 20–53.
- Jain S, Kumar A. B4: experience with a globally-deployed software defined WAN. *Proceedings of the 2013 ACM SIGCOMM Workshop on Future Human-centric Multimedia Networking*. Hong Kong, China, 2013. 3–14.
- Cheng YC, Cardwell N. RACK: A time-based fast loss detection algorithm for TCP. *Proceedings of the IETF 98*. Chicago, IL, USA, 2017.
- 任勇毛, 唐海娜, 李俊, 等. 高速长距离网络传输协议. *软件学报*, 2010, 21(7): 1576–1588.
- 王国栋, 任勇毛, 李俊. TCP 改进协议在高速长距离网络中的性能研究. *通信学报*, 2014, 35(4): 81–90. [doi: [10.3969/j.issn.1000-436x.2014.04.010](https://doi.org/10.3969/j.issn.1000-436x.2014.04.010)]
- Jacobson V. Congestion avoidance and control. *Proceedings of the SIGCOMM'88 Symposium*. 1988. 314–32.
- Xu LS, Harfoush K, Rhee I. Binary increase congestion control for fast long distance networks. *Proceedings of the 23rd Annual Joint Conference on Computer and Communications Societies*. Hong Kong, China, 2004. [doi: [10.1109/INFCOM.2004.1354672](https://doi.org/10.1109/INFCOM.2004.1354672)]
- Xu LS, Rhee I. CUBIC: A new TCP-friendly high-speed TCP variant. *ACM SIGOPS Operating System Review*, 2005, 42(5): 64–74. [doi: [10.1145/1400097.1400105](https://doi.org/10.1145/1400097.1400105)]
- Brakmo L, O'Malley S. TCP vegas: New techniques for congestion detection and avoidance. *Conference on Communications Architectures*, 1994, 24(4): 24–35.
- Cheng J, Wei D, Low SH. FAST TCP: From theory to experiments. *IEEE Network the Magazine of Global Internetworking*, 2005, 19(1): 4–11. [doi: [10.1109/MNET.2005.1383434](https://doi.org/10.1109/MNET.2005.1383434)]
- Floyd S. High Speed TCP for Large Congestion Windows. RFC 3649, December 2003.
- Brakmo L. BBR Evaluation, 2016. https://drive.google.com/file/d/0B4YZ_0yTgbJEa21CbUVLWFdrX2c/view.
- Wilk A, Iyengar J. QUIC: A UDP-based secure and reliable transport for HTTP/2. *IETF93: Prague*, July 16, 2016.
- Netem. <https://wiki.linuxfoundation.org/networking/netem>.
- Iperf. <https://iperf.fr/>.