高速网络环境下的传输性能优化研究进展*)

武航星[1] 慕德俊[1] 潘文平[1] 乔梅梅[2]
(西北工业大学自动化学院 西安 710072)[1] (北京交科公路勘察设计研究院有限公司)[2]

摘 要 目前,对下一代高速互联网络的研究正如火如荼地进行。然而,随着研究的不断深入,人们发现,在高速网络环境下,传统 TCP 存在着严重的不足,连接传输性能低下。针对这一问题,各种改善方案应运而生。本文对这些方案进行了大致分类,并重点介绍了各分类中的代表性算法,指出了其优缺点。最后,提出了几个有价值的研究方向。

关键词 下一代高速网络,性能优化

Recent Development of Transport Performance Optimization in High-speed Networks

WU Hang-Xing[1] MU De-Jun[1] PAN Wen-Ping[1] QIAO Mei-Mei[2]
(College of Automation, Northwestern Polytechnical University, Xi'an 710072)[1]
(Beijing Jiaoke Highway Surwaysing, Design and Research Institute)[2]

Abstract At present, research to the next generation high-speed networks is developed rapidly. However, with the in-depth study it was found that traditional TCP has severe defects and its' transport performance is poor in high-speed network. In order to solve this problem, various ideas were proposed. In this paper, we categorize these ideas, and then emphatically describe typical methods in each category and point out advantages and disadvantages of these methods. In the end, some valuable directions are provided.

Keywords Next generation high-speed networks, Performance optimization

1 引言

随着计算机网络的飞速发展,其应用领域不断拓宽,各种应用模式不断涌现,像音频和视频这样的对网络资源要求较高的多媒体应用更是呈爆炸性增长。目前,各种涉及海量数据传输的应用也不断涌现,如地球物理学、生物信息科学和射电天文学等领域的研究,均需要传输大量的数据,对网络传输性能提出了较高的要求。为了满足这些应用的的需求,为用户提供高速互联服务,下一代高速互联网的研究正在如火如荼地进行。在美国,从 1996 年开始,由 184 所大学和 70 家企业组成的 Internet 2 论坛,一直致力于开发新一代高速互联网,一种带宽更宽并且能够提供更好的 QoS 的 Internet,这就是我们所说的 Internet 2。Internet 2 的一个主要目标,就是为世界各地的大学提供接近实时的数据收集和分析能力,使科学家们通过千兆每秒的速度来传输数据。Abilene 是一个现已搭建好的 Internet 2 网络,它以 2.4Gbps 的速度连接着全世界的许多大学。Abilene 支持丰富的媒体类型以及端到端的 QoS。当前,已经有 180 所大学连接到 Abilene 上了。除了大学之外,还有一些设备提供商和政府的研究实验室也进入了 Abilene。在其他国家和地区也相继开展了下一代高速互联网络研究,包括加拿大的 CA 3NET、英国的 JANET2 以及亚太地区的 APAN 等。在中国,下一代高速互连网络"中国高速互连研究试验网络 NSFCNET"于 2000 年 9 月 25 日开通演示。该高速网络由六个骨干节点构成,网络传输带宽最高达 10G(100 亿)比特,部分骨干节点之间带宽达 2.5G 比特,并分别与中国教育和科研计算机网 CERNET、中国科技网 CSTNET,以及国际下一代互联网络 Internet 2 和亚太地区高速网 APAN 互联,是我国第一个与 Internet 2 实现互联的计算机互联网,为中国下一代互联网研究与世界接轨奠定了基础。

2 传统 TCP 在高速网络环境下的局限

随着下一代高速网络研究的不断深入,研究人员发现,在高速网络环境下,使用传统 TCP 的连接不能有效利用可用带宽,连接的传输性能低下。试验结果显示,在 Internet 2 中的带宽为 622Mbps 的链路上,对于大块数据(不少于 10MB)的传输,90%的 TCP 数据流传输速率小于 5Mbps,99%的 TCP 数据流传输速率小于 20Mbps[1]。通过深入的分析,文[2]指出,传统 TCP 在接收方使用通告窗口来对发送方的发送速率进行限制,对于大多数系统,该窗口的默认值为 64kB。虽然在 RFC1323[3]中对该值进行了扩展,但是目前仍然没有被广泛采用,正是这个 64kB 的默认值限制了可用带宽被有效利用。例如,对于带宽 600Mbps,往返时延 100ms 的链路,其利用率仅为 64kB/600Mbps×100ms≈0.8%。同时,文[2,4]指出,在同一端系统中,对于不同的连接,传统 TCP 使用相同的静态缓冲区分配参数。然而,在高速网络环境下,对于同一端系统的不同连接,其带宽延时积可能跨越几个数量级,变化较大,使用相同的静态缓冲区分配参数不可能适合所有的连接。这也是引起传统 TCP 连接传输性能低下的原因。此外,文[5]指出,传统 TCP 采用保守的和式增加和激进的积式减少(AIMD)的拥塞窗口调整策略,是造成连接性能低下的一个重要原因。例如,对于分组大小为 1500Bytes、往返时延

武航星 博士生,主要研究方向为网络拥塞控制和流量控制;**慕德俊** 教授,博士生导师,主要研究领域为并行计算、信息安全;**潘文平** 博士生,研究方向为网络服务质量;**乔梅梅** 工程师。

100ms 的连接，要达到 10Gbps 的传输速率，发送方的拥塞窗口要达到 83333 个分组大小，对于使用的 AIMD 拥塞窗口调整策略的传统 TCP，拥塞避免阶段将会持续 4167s，约 1.2h。这期间的可用带宽显然得不到完全有效的利用。

3 网络传输性能优化思路

针对传统 TCP 在高速网络环境下传输性能低下的不足，研究人员做了大量的工作，并提出了对其进行改善的一些思路。在本部分中，我们将这些思路大致分为四类：1）自动调整 TCP 连接参数；2）使用并行的 TCP 连接；3）在端系统中改进 TCP 的拥塞窗口调整策略；4）在中间节点提供精确的反馈信息。然后对各分类中的一些具有代表性的算法进行介绍和分析。

3.1 自动调整 TCP 连接参数

在高速网络环境下，研究人员发现，TCP 连接参数对数据传输性能具有很大的影响。一个网络专家可以通过调节优化 TCP 连接参数来取得较满意的连接性能。然而，对于一个普通用户来说，这种调节几乎是不可能的。并且，即便是对一个网络专家，这种调节也是繁琐而费时的。因此，研究人员开发了一些自动调整 TCP 连接参数的方法，以下是其中的几个代表性方法。

3.1.1 WAD[6]

WAD(Work Around Daemon)对连接参数的调整是基于 Web100 协议栈和网络工具分析框架 NTAF(Network Tool Analysis Framwork)实现的。Web 100 提供了一个管理信息库 MIB(Management Information Base)来记录 TCP 连接的统计数据，帮助 WAD 分析 TCP 连接所存在的具体问题。NTAF 是为了给具体的网络路径提供调节参数而开发的框架，主要功能就是在规则的时间间隔内运行测试工具，并将得到的结果保存到中央文档系统，备 WAD 设置 TCP 参数使用。WAD 的具体运行过程是这样的：当一个新的连接事件发生时，内核通知 Daemon 进程，Daemon 进程首先检查配置文件来判断该连接是否为参数可调节的连接。对于参数可调节的连接，要么使用配置文件表中的静态值来设置连接参数，或者使用 NTAF 中保存的调节参数来进行调节。

文[6]中，对连接缓冲区大小、虚拟 MSS、AIMD 参数、禁用延时确认、重新排序阈值等参数进行了大量的试验。结果表明，结合 Web100 的 MIB 和 NTAF，WAD 可对高延时高带宽环境下的海量数据传输的连接参数进行透明的调节，可使连接性能提高一个数量级以上。虽然如此，正如该文中一再强调，要真正决定那些连接调节参数应该标准化，仍然需要更多的试验。但是 WAD 存在的不足是：在一定的网络环境下，AIMD 参数的调节可能会影响到一些其他用户的公平性；有一些参数的调节效果不是很明确。而且对于那些参数应该被标准化，暂时也还没有好的结果。

3.1.2 Auto TCP Buffer Tuning[4]

随着高性能网络的不断发展，目前在单个端主机上同时存在的连接的带宽延迟积可能跨越 4 个不同的数量级。因此，对于这些 TCP 而言，默认的 TCP 缓冲配置参数不可能同时使多个连接都达到连接性能的优化。TCP 缓冲区自动调整的思想正是为解决这一问题而提出的。

TCP 缓冲区自动调整主要由两部分组成：1）接收方缓冲区的调整。当接收方缓冲区大多数情况下为空时，就认为这种低速率的传输可能是由于接收方缓冲过小造成的，因而增加缓冲区。当接收方缓冲区大于快速恢复时所需的缓冲区时，即认为峰值利用率已经达到，适当减小接收方缓冲。2）发送方缓冲区的调整。首先，TCP 连接根据网络条件，即连接的带宽时延积，设置连接发送方缓存 B_{send} 为带宽时延积的 2 倍大小。然后，发送方计算出平均每连接可分配到的缓冲区大小 $B_{fair}=B_{max}/N$，其中 B_{max} 为发送方可分配的总缓冲区阈值。N 为总的 TCP 连接的数量。当 $\sum B_{send}\geqslant B_{max}$ 时，设置 $B_{send}=\min(B_{send},B_{fair})$；当 $\sum B_{send}<B_{max}$ 时，将未能完全利用的缓冲分配给 B_{send} 大于 B_{fair} 的连接。TCP 缓冲区自动调整的实现只涉及到对基于 BSD 的套结字接口和 TCP 协议栈进行少量的改动，即可在性能上得到显著的提高。

3.1.3 DRS[2]

DRS(Dynamic Right-Sizing)对发送方的流量控制窗口和接收方的通告窗口进行动态调整，来提高发送方、接收方缓冲区的利用效率，可使连接效率显著提高，而且其实现对终端用户透明，并可和其他 TCP 实现共存。其具体实现伪代码如下。

发送方：

$fwnd=min(cwnd,awnd,maxwnd)$

接收方：

if (sender in slow start)

$awnd\geqslant 2\times cwnd$

if (sender in additive increase)

$awnd\geqslant cwnd+1$

其中：$fwnd$ 为发送方的流量控制窗口，$awnd$ 为接收方的通告窗口，$cwnd$ 为发送方的拥塞窗口，$maxwnd$ 为任一连接可用的最大缓存大小。显然，和传统的 TCP 相比，DRS 在发送方通过增加参数 $maxwnd$ 对流量控制窗口 $fwnd$ 进行限制；在接收方，通过推断发送方所处的状态和其拥塞窗口 $cwnd$ 大小来动态调整通告窗口 $awnd$。

试验结果显示，使用 DRS 可使连接效率提高一个数量级以上。然而，使用 DRS 的数据流和未使用 DRS 的数据流相比将得到更多的资源和带宽，存在公平性问题。

3.2 使用并行的 TCP 连接

并行 TCP 实质上相当于对传统 TCP 的拥塞窗口调整机制进行了修改。例如，对于 K 个并行连接，在无丢包时，一个 RTT 内，拥塞窗口增加 K；当其中一个连接发生丢包时，该连接拥塞窗口减半，但相对于 K 个连接，其总的拥塞窗口只减少了 $1/2K$。以下为该类中的一些代表算法。

3.2.1 XFTP[7]

XFTP 是对 FTP 进行功能上的扩展，增加了对多个 TCP 连接的支持和应用层拥塞避免的方法来提高连接吞吐量。XFTP 主要有两个特点：1）文件分块。由于 XFTP 认为，对于 K 个并行 TCP 连接，每个连接的吞吐量很可能会有所不同，所以简单地将要传输的文件分为和并行连接数量 n 相同的数据块数，吞吐量较小的连接会需要较长的时间才能完成传输任务，一直处于繁忙状态。而吞吐量较大的连接需要较短时间就会完成传输任务并马上处于空闲状态，这显然降低了连接效率。因此，XFTP 将待传输文件分为 m 个 8kbyte 的数据块($m\gg n$)，并行的任意连接一有资源可供利用就立即读取数据块并传输。这样，在文件的整个传输过程中，可保证所有的并行连接均始终工作，从而提高连接效率。2）应用层拥塞避免算法。如何决定最优的并行 TCP 连接的数量，一直是运用并行 TCP 来提高连接吞吐量的难点问题。对于这一问题，

XFTP 首先设置连接数量为默认的最大值，然后借助 TcpVegas 的思想，观测连接的 RTT 来决定对连接数量的增减。具体如下：设置 RTT 阈值 α、β，当观测到的 RTT 小于 α，增加连接数量。反之，当观测的 RTT 大于 β，减少连接数量。

然而，文[8]指出，XFTP 对文件的分块方法会引起较大的处理开销。同时，用 RTT 来预测网络的负载状况本身也存在着一定的问题。

3.2.2 PSocket[8]

PSocket(Parallel Socket)的基本思想和 XFTP 很相似，运用并行 TCP 连接传输分块的文件。二者的主要区别是对文件的分块方法。PSocket 认为对于源和目的端均相同的 TCP 连接，连接路径很可能也相同，所有并行连接的吞吐量应该相差不大，因而简单地将文件分为和并行连接数量相同的块数可以减小不必要的开销。然而，对于最优并行连接数量，PSoceket 是通过测试来决定，未给出较好的方法。

3.2.3 CPTCP[9]

显而易见，使用并行 TCP 连接的数据流和单连接的数据流在瓶颈资源的竞争上将占有绝对优势，存在严重的公平性问题。CPTCP 的优点就是针对这一问题，提出了一种公平性改善方案。CPTCP(Combined Parallel TCP)的总体思想如下：在网络未充分利用、轻度负载时，利用并行 TCP 连接增加连接吞吐量，提高网络性能；在网络重载时，对并行 TCP 连接进行限制，维护数据流兼有较好的公平性。对公平性的实现，CPTCP 是通过利用 TCP 拥塞避免的一个特点(具有较长 RTT 的数据流和较短的 RTT 的数据流相比，在资源的竞争上处于劣势地位)来改善公平性的。具体实现思想如下：CPTCP 对于并行的 TCP 连接，在拥塞避免阶段，使其增长 cwnd 所需要积累的 ACK 的数量加倍，其影响就等同于产生了一个虚拟的加倍的 RTT，从而降低了并行的 TCP 连接对带宽的竞争能力。试验结果显示，CPTCP 相对于其他的并行 TCP 连接，在网络拥塞时对公平性有一定的改善。

3.3 在端系统中改进 TCP 的拥塞窗口调整策略

针对传统 TCP 的保守增加和激进减少的拥塞窗口调整机制在高速网络环境下传输性能低的不足，研究人员提出了一些新的拥塞窗口调整方案来改进网络传输性能。其代表性算法如下。

3.3.1 HSTCP[5]

HSTCP(High Speed TCP)的主要特点是拥塞窗口的和式增加因子 $a(w)$ 与积式减小因子 $b(w)$ 都是拥塞窗口 w 的函数，随着 w 的变化而动态地变化，以更好地利用带宽。其具体实现如下：通过选择 log-log 坐标下 w 与丢包概率 p 之间为简单的线性关系，根据 NS2 下的试验数据，得出新的 HSTCP 响应函数：

$$w=0.12/p^{0.835} \tag{1}$$

选择积式减小因子 $b(w)$ 与 $\log w$ 为线性关系，得出

$$b(w)=0.69-0.12\log w \tag{2}$$

由(1)、(2)式可推出和式增加因子：

$$a(w)=\frac{0.11w^{0.8}-0.02w^{0.8}\log w}{1.31+0.12\log w}$$

同时考虑在低速环境下与传统 TCP 的兼容性，HSTCP 设置一个拥塞窗口阈值 W，文[5]取 $W=38$。当 $w\leqslant W$ 时，其 $a(w)$、$b(w)$ 与传统的 TCP 取相同的值，分别为 1，0.5。试验结果显示，HSTCP 在高速环境下可显著提高连接的吞吐量。然而 HSTCP 流对带宽的竞争能力远大于传统的 TCP 流，存在着严重的公平性问题。

3.3.2 Scalable TCP[10]

Scalable TCP 是建立在 HSTCP 基础之上的，二者的思想基本相同，所不同的只是 Scalable 基于对带宽的分配、数据流瞬时速率的变化、收敛速度和系统稳定性的考虑，选择拥塞窗口 *cwnd* 的调整算法如下：在拥塞避免阶段，当收到一个确认的 ACK 时，$Cwnd\leftarrow Cwnd+0.01$；当在一个 RTT 内首次检测到拥塞事件时，$Cwnd\leftarrow Cwnd-0.125\times Cwnd$。和 HSTCP 一样，考虑在低速环境下和传统 TCP 的兼容性，Scalable TCP 也设置一个拥塞窗口阈值 *lwnd*，文[10]取为 16。当 $Cwnd\leqslant lwnd$ 时，Scalable TCP 的拥塞窗口调整算法取为和传统 TCP 相同的调整算法。关于 *lwnd* 的取值，文[10]指出 16 并非最优值，*lwnd* 取值对 Scalable TCP 性能的影响还需进一步研究。同时，在高速环境下 Scalable TCP 流和传统 TCP 流间也存在着严重的公平性问题。

3.3.3 BI-TCP[11]

BI-TCP(Binary Increase TCP)的目标是在提高连接吞吐量的同时对 HSTCP 和 Scalable TCP 所存在的 RTT 公平性问题进行一定的改善。BI-TCP 与 HSTCP、Scalabel TCP 调整 Cwnd 的思路完全不同，它的主要特点表现在拥塞窗口的增加算法上。其拥塞窗口增加算法由两部分组成：二分搜索增加(binary search increase)、和式增加(additive increase)。所谓二分搜索增加，其含义如下：预设某一较大的最大窗口 max_*wnd*，设置当前窗口 *cwnd* 为最小窗口 min_*wnd*，当收到一个 ACK 时，调整 *cwnd* 如下：

$$Cwnd\leftarrow Cwnd+\frac{(max\_wnd-min\_wnd)/2}{Cwnd}$$

即在一个 RTT 内，直接将 *cwnd* 增加至 *max_wnd* 与 *min_wnd* 的中点处。此后若发生丢包事件，则设置 *max_wnd* 为当前 *cwnd* 大小，*min_wnd* 为减小后的 *cwnd* 大小，再重新计算中点值，继续执行二分搜索增加；相应地，若无丢包事件发生，则设置 *min_wnd* 为当前 *cwnd* 大小，*max_wnd* 不变，也同样计算中点值，继续执行二份搜索增加。这一过程重复进行，直到 $(max\_wnd-min\_wnd)/2$ 小于最小阈值 S_{min} 时为止。仅使用二份搜索增加方法存在的问题是当 *max_wnd* 与 *min_wnd* 相差较大时，直接将 *cwnd* 增加至二者的中点值处将给网络带来极大的压力。为了避免这一问题而提出了和式增加算法，设置阈值 S_{max}，当 $(max\_wnd-min\_wnd)/2\geqslant S_{max}$，在收到一个 ACK 时，$Cwnd\leftarrow Cwnd+S_{max}/Cwnd$。显然，BI-TCP 的一个很好的优点就是随着 *cwnd* 不断接近饱和点，*cwnd* 的增加速率不断减小，这就避免了在饱和点附近过大增加 *cwnd* 而引发的过多的不必要的丢包发生。同时，文[11]也在理论上和试验上证明了 BI-TCP 具有近似的线性 RTT 不公平性，即 $w_1/w_2\approx RTT_2/RTT_1$。和 HSTCP、Scalable 相比，性能有很大的改善。然而，当丢包率小于 10^{-8} 时，BI-TCP 流的发送速率的增长不如 HSTCP 和 Scalable TCP 快。

3.3.4 Fast-TCP[12,13]

Fast-TCP 是由加州理工大学开发的一种新的高速传输协议。Fast-TCP 认为，在高速网络环境下的丢包概率很小，通常在 10^{-8} 或更低的数量级上。在这样的情况下，使用排队时延对网络的拥塞状况进行预测将比使用丢包概率更加精确。因此 Fast-TCP 根据排队时延对 *cwnd* 进行调整。Fast-TCP 由两部分组成：1)对排队时延的估计。首先使用加权滑动平均算法对 RTT 进行估计，有

$$\overline{T}_i(k+1)=(1-\eta(t_k))\overline{T}_i(k)+\eta(t_k)T_i(k)$$

其中权值 $\eta(t)=\min\{3/w_i(t),1/4\}$，$w_i(t)$为 *cwnd* 在时刻 t 的值。$T_i(k)$为第 k 个 RTT 的采样值。再用得到的 RTT 均值估计排队时延：

$$\hat{q}_i(k)=\overline{T}_i(k)-base\_rtt_i$$

其中 $d_i(k)$为数据流 i 所观察到的最小 RTT。2)对 *cwnd* 的调整。Fast-TCP 根据估计得到的 $\overline{T}_i(k)$，$\hat{q}_i(k)$对 *cwnd* 进行调整：

$$w\leftarrow min\left\{2w,(1-\gamma)w+\gamma\left(\frac{base\_rtt_i}{\overline{T}_i(k)}w+\alpha(w,\hat{q}(k))\right)\right\}$$

其中 $\gamma\in(0,1]$，$\alpha(w,\hat{q}(k))$在文中暂取为常数。当检测到丢包事件时，减小因子同传统 TCP，有 $w\leftarrow 0.5w$。仿真试验表明，Fast-TCP 在吞吐量、公平性、快速响应、稳定性上均好于 HSTCP 与 Scalabe TCP。然而，文[14]却指出，Fast-TCP 存在着不能准确估计 RTT 的可能，这会导致引发不公平问题和路由器队列的振荡，文中同时给出了一种改进方法。

3.4 在中间节点提供精确的反馈信息

此类思路利用路由器为端系统提供及时、准确的网络拥塞状况的反馈信息，以供端系统更高效精确地调整发送速率，从而提高网络传输性能。其代表性算法如下。

3.4.1 XCP[15]

XCP(eXplicit Control Protocol)认为使用丢包事件作为主要依据对 *cwnd* 进行调整是造成传统 TCP 使用保守的和式增加和激进的积式减少算法，引起传统 TCP 在高速网络环境下传输性能较低的一个重要原因。网络拥塞信号应该能够精确地反映出网络的拥塞程度，而用“拥塞”或“不拥塞”这样的二进制变量来描述显然不够精确。因此 XCP 针对这一问题，对 IP 包头进行了扩展，在路由器上对网络拥塞程度进行精确的估计，并将结果反馈给发送端，以供其更好地调整 *cwnd* 来提高传输性能。增加的拥塞报头如图 1 所示。

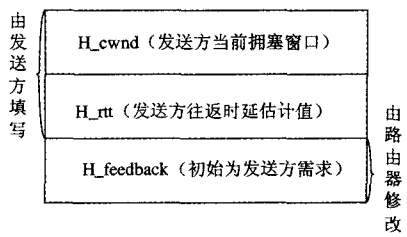

图 1 拥塞报头

XCP 的思想由 XCP 发送方、接收方和 XCP 路由器一起实现。XCP 发送方的功能主要是填写拥塞报头，供路由器计算 *H_feedback*。并根据反馈回的 *H_feedback* 对 *cwnd* 进行调整，调整策略为

$$cwnd\leftarrow max(Cwnd+H\_feedback,s)$$

其中 s 为一个数据包尺寸大小。XCP 接收方和传统 TCP 相似，所不同的只是当确认一个数据包时，要对拥塞报头进行复制，再回传给发送方。XCP 路由器主要完成对网络拥塞状况的估计，并将反馈信息 *H_feedback* 写入拥塞报头。在对 *H_feedback* 的计算过程中，XCP 的主要创新就是将效率控制器 EC(Efficiency Controller)和公平控制器 FC(Fairness Controller)分为两个独立的部分，这大大简化了分类器的分析和设计。EC 的目标是最大化链路的利用率，它将所有的输入数据流看作一个整体，根据链路带宽的利用状况，计算出一个总的反馈值，而不关心这些流之间的公平性。FC 的任务是根据计算得出的总反馈值，按照一定的规则，将反馈分配给每个数据包。仿真试验表明，与传统 TCP 相比较，XCP 在链路利用率、收敛速度、公平性、队列延时和丢包率等各方面性能均有很好的改善。然而，XCP 的实现需要扩展 IP 报头，这将直接影响到该协议的可扩展性。而且，显然也需要路由器的支持。对于当前复杂的网络环境，XCP 的部署上也存在着很大的困难。

3.4.2 VCP[16]

VCP(Variable-structure Congestion Control Protocol)针对 XCP 协议需要扩展 IP 报头这一问题，将网络拥塞状况进行分级，巧妙地利用 IP 报头中的 ECN 位来表示不同网络的拥塞状况。因此，VCP 的一个很好的优点就是无需对扩展 IP 报头。VCP 的具体思想如下：在路由器上计算链路利用率，并根据链路利用率将网络拥塞状况分为三级：轻负载、重负载和过负载状态，分别对应 ECN 位编码$(01)_2$、$(10)_2$、$(11)_2$。在终端系统，吸取 XCP 的效率和公平控制相分离的思想，根据 ECN 位反馈的不同而采取不同的控制算法。在轻载时，VCP 的主要目标是提高链路利用率，因此采用积式增加拥塞窗口算法。在重负载和过负载时，VCP 的主要目标是提高公平性，因此分别采用和式增加和积式减小拥塞窗口算法。可以看出，VCP 用 ECN 位来反馈网络拥塞状况的思想虽不及 XCP 的反馈精确，但比传统 TCP 的二进制反馈要好得多。同时 VCP 无需对 IP 报头进行修改，比 XCP 的可扩展性好。仿真试验表明，VCP 在链路利用率、队列延时和丢包率方面性能近似于 XCP，但公平性收敛时间要长于 XCP。

4 当前的几个热点研究问题

为了解决传统 TCP 在高速网络环境下传输性能低下的问题，研究人员已经做了大量的工作，并取得了一定的成果，提出了一些好的思路和算法。然而，从上面的分析中可以看出，这些思路和算法大都存在这样或那样的问题。目前，对高速网络环境下网络传输性能的优化研究还很不成熟，仍存在很多的问题值得深入研究。以下是我们认为比较有价值的一些热点研究问题。

• 公平性问题的研究。上面介绍的思路和算法，虽然在传输性能上有很大的改善和提高，但它们普遍存在着严重的公平性问题。公平性问题不仅存在于不同类型的数据流之间，而且存在于同类型而具有不同往返时延的数据流之间。而且，目前对公平性的评价指标大多采用文[17]的公平性指标，它将“公平”定义为瓶颈带宽在不同数据流之间绝对平均分配，而不考虑这些数据流各自一些属性的千差万别。对于该公平性定义是否完全合理，也可进行更近一步的研究和探讨。例如，文[2]就提出了一种相对公平性定义方法。

• 中间结点上的算法研究。目前，利用中间节点来改善和提高高速环境下网络传输性能的研究还不是很多，这方面好的算法也出现得较少。其原因很大程度上是许多研究者认为，中间结点上的算法很难在当今如此复杂的网络环境下部署。但是，中间节点上的算法有其自身的一些优点。在中间节点上可以及时、精确地感知网络拥塞状态，从而可以采取更精准的拥塞窗口调整算法来提高网络性能。同时，在中间节点上能根据不同数据流的信息，较好地改善公平性。从对 XCP、VCP 的试验可以看出，中间节点上的算法具有很好的传输性能。因此，我们认为，端系统对拥塞的感知方式是远远不够的，中间结点参与到拥塞控制中将是一个值得深入研究

的方向。至于可部署性问题，我们认为性能的改善和可部署性之间应该存在一个较好的折衷。

·控制理论、优化理论的应用研究。从以上的思路和算法的分析介绍中，不难看出，它们虽然对网络传输性能有所改善，但其自身几乎都存在着这样或那样的问题。其直接原因就是这些算法的设计都是针对局部的某一具体问题进行，依靠直觉的推断，根据经验改进算法，缺乏一套有效的、系统的理论分析工具对算法的设计进行指导。控制理论、优化理论作为相当成熟的系统理论，有相当多的方法可以借鉴到网络性能优化中来。近来，国内外的很多专家学者都认识到了可以应用控制理论、优化理论中的方法来解决网络中的问题，并做了一些尝试工作[18,19]。然而，由于网络自身的复杂性，这方面的研究还不成熟。所以，如何有效地将控制理论、优化理论的思想运用于日趋复杂的网络中，来指导目前单纯根据经验来改进算法的不足，将是一个未来研究的热点问题，也是一个难点问题。

结束语 本文对高速网络环境下的传输性能优化研究进行了分类，并重点分析了各分类中一些代表性的思路。同时在文中给出了几个较有价值的热点研究方向，希望能起到抛砖引玉的作用。

参 考 文 献

1 Internet2. Internet2 NetFlow: Weekly Reports, 2002. URL: http://netflow.internet2.edu/weekly/
2 Weigle E, Feng Wu-chun. Dynamic Right-Sizing: a Simulation Study. In: Proceedings of IEEE International Conference on Computer Communications and Networks (ICCCN), October 2001
3 Van Jacobson, Braden R, Bor-man D. TCP extensions for high performance. RFC1323, May 1992
4 Semke J, Mahdavi J, Mathis M. Automatic TCP Buffer Tuning. In: Proceedings of ACM SIGCOMM, October 1998
5 Floyd S. High Speed TCP for Large Congestion Windows. RFC 3649, December 2003
6 Dunigan T, Mathis M, Tierney B. A TCP Tuning Daemon. SuperComputing (SC), November 2002
7 Ostermann S, Allman M, Kruse H. An Application-Level solution to TCP's Satellite Inefficiencies. In: Workshop on Satellite-based Information Services (WOSBIS), November 1996
8 Sivakumar H, Bailey S, Grossman R. PSockets: The Case for Application-level Network Striping for Data Intensive Applications Using High Speed Wide Area Networks. In: Proceedings of Super Computing, November 2000
9 Hacker T, Noble B, Athey B. Improving Throughput and Maintaining Fairness Using Parallel TCP. In: Proceedings of IEEE Infocom 2004, March 2004
10 Kelly T. Scalable TCP: Improving Performance in HighSpeed Wide Area Networks. ACM Computer Communications Review, April 2003
11 Xu Lisong, Harfoush K, Rhee I. Binary Increase Congestion Control for Fast Long-Distance Networks. In: Proceedings of IEEE Infocom 2004, March 2004
12 Cheng Jin, Wei D X, Low S H. FAST TCP: motivation, architecture, algorithms, performance. IEEE Infocom, March 2004
13 Cheng Jin, Wei D X, Low S H, et al. FAST TCP: From Theory to Experiments [ED/OL]. http://netlab.caltech.edu/, 2003
14 Tan LS, Yuan C, Zukerman M. FAST TCP: Fairness and queuing issues. IEEE Communications Letters, 2005, 9 (8): 762～764
15 Katabi D, Handley M, Rohrs C. Congestion Control for High Bandwidth-Delay Product Networks. In: Proceedings of ACM SIGCOMM 2002, August 2002
16 Xia Y, Subramanian L, Stoica I, et al. One more bit is enough. ACM SIGCOMM Computer Communication Review, 2005, 35 (4): 37～48
17 Chiu D, Jain R. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks [J]. Computer Networks and ISDN Systems, 1989, 17(1): 1～14
18 Shor M H, Lik W J. Application of control theory of modeling and analysis computer system. http://www.cse.ogi.edu/～kangli/
19 Habibipou F, Khajepour M, Galily M. Application of control engineering methods to congestion control in differentiated service networks. Control Engineering Practice, 2006, 14(4): 425～435

(上接第 49 页)

[7,8]所采用的方法要好，可以有效地降低预测误差。这些证明，基于小波变换与自回归模型的方法，对网络流量预测是可行、有效的，并且比常用的预测方法具有更高的预测准确度。

结论 本文提出一种基于小波变换与自回归模型的网络流量预测方法，先将网络流量数据经过 Mallat 算法分解与单支重构，然后对重构后的近似部分和各细节部分别建立自回归模型，进而实现原始网络流量的预测。实验结果表明，该方法比传统的几种预测方法具有更高的预测精度，这表明小波变换在网络流量预测中的作用是显著的。对于这种方法还可以从如下两个方面进行改进：(1)在对单支重构得到的序列建立模型时，如果再辅以其它平稳化方法可以进一步提高预测准确度，比如：对数变换、差分、季节差分，甚至包括参考文[8]的平稳方法。(2)基于小波变换的预测方法，还可以跟其它常用模型结合使用，比如：ARMA 模型、ARIMA 模型等，由于这些模型比 AR 模型通常具有更好的预测效果，因此这类结合方法的预测准确度更高。

值得一提的是，在基于小波变换的预测中，还存在确定分解层数以及选用合适小波函数的问题。分解层数越多，分解后的信号在频率成分上越单一，各部分的平稳性也越好，但是随着分解层数的增加，所需的计算量也显著增大，因此分解层数的选择应适中。选用不同的小波函数，得到的预测效果会略有不同，比较常用的滤波器有：Battle 和 Lemarie 的 27-系数滤波器(简称 B-L 小波)，I. Daubechies 的 4-系数滤波器(简称 D-4 小波)，I. Daubechies 的 20-系数滤波器(简称 D-20 小波)，以及 Antonini 的一组双正交小波基对应的滤波器等。从定性、定量两个方面，对基于小波变换的预测方法进行误差分析，以进一步提高网络流量预测的准确度，是我们下一步的主要研究工作。

参 考 文 献

1 Yu I, Kim C. A Novel Short-Term Load Forecasting Technique Using Wavelet Transform Analysis [J]. Electric machines and power systems, 2000, 28: 537～549
2 Renaud O, Starck J L, Murtagh F. Wavelet-based Forecasting of Short and Long Memory Time Series [EB/OL]. http://www.unige.ch/ses/metri/cahiers/2002_04.pdf, May 2002
3 Paxson V, Floyd S. Wide-area Traffic: The failure of Poisson modeling [J]. IEEE/ACM Trans, Networking, 1995, 3: 226～244
4 Tsybakov B. Self-similar Process in Communications Networks [J]. IEEE Trans on Information Theory, 1998, 44(5): 1713～1725
5 韩良秀，从锁. 基于小波技术的网络流量特性刻画[J]. 小型微型计算机系统，2001，2(9)：1110～1113
6 从锁，韩良秀. 基于离散小波变换的网络流量多重分形模型[J]. 通信学报，2003，24(5)：3～8
7 邹柏贤，刘强. 基于 ARMA 模型的网络流量预测[J]. 计算机研究与发展，2002，39(12)：1645～1652
8 邹柏贤，姚志强. 一种网络流量平稳化方法[J]. 通信学报，2004，25(8)：14～23
9 王振龙. 时间序列分析[M]. 中国统计出版社，2002
10 Mallat S. A theory for multiresolution signal decomposition: The wavelet representation [J]. IEEE Trans Pattern Anal Mach Intel, July 1989, 11: 674～693
11 Antonini M, et al. Image coding using wavelet transforms. IEEE Trans. on Image Processing, 1992, 1(2): 205～220