

ZetaTCP®

基于智能学习的单边 TCP 传输加速技术



目 录

前言	2
1. TCP 加速技术的演进：三代技术.....	3
1.1. Loss-based 的 TCP 加速技术.....	3
1.2. Delay-based 的 TCP 加速技术.....	4
1.3. Learning-based 的动态 TCP 加速技术.....	4
2. ZetaTCP®的设计原则.....	5
2.1. 对应用及网络拓扑全透明.....	5
2.2. Learning-based 动态算法.....	6
2.3. 可扩展性.....	7
3. ZetaTCP®的主要加速机制.....	8
3.1. 更有效的拥塞判断及处理.....	8
3.2. 基于预测的快速丢包判断及恢复.....	11
4. 第三方比较测试.....	13
结论.....	14

前言

传输控制协议 *TCP* 是 *TCP/IP* 协议栈中的传输层协议。据统计，目前全球互联网数据流量 90% 以上通过 *TCP* 传输，通过 *UDP* 传输的不足 10%。而且 *TCP* 所占份额还在继续扩大，就连传统上通过 *UDP* 传输的多媒体应用也逐渐开始愈来愈多的采用 *TCP* 传输。

TCP 通过序列确认以及包重发机制，提供可靠的数据流发送。同时它对不同网络状况具有良好的自适应性，对互联网的迅速发展和普及立下了汗马功劳。但是这一设计于二十多年前的传输协议已经越来越不适应飞速发展的高速网络环境和新型应用的要求。当网络路径上存在一定的丢包和延时的情况下，*TCP* 连接的吞吐显著下滑，常常无法有效地利用带宽，从而造成带宽的闲置和浪费，并必然导致远程数据传输耗时太长，应用响应缓慢甚至无法使用等问题。

针对 *TCP* 在现代网络环境中传输效率的问题，学术界和产业界推出了一些 *TCP* 优化改进的算法和机制。从部署方式上分为双边 *TCP* 加速和单边 *TCP* 加速两类。双边 *TCP* 加速将 *TCP* 转化为私有协议，需要在传输的两端部署同样的技术，使用范围很窄，其应用仅限于公司跨分支机构的数据传输。而单边 *TCP* 加速保持和标准 *TCP* 协议的完全兼容，只需要在 *TCP* 连接的一端部署，具有非常广泛的适用性。对直接面向广大终端用户的互联网应用、移动应用和云计算应用，单边 *TCP* 加速是唯一的选择。

本篇白皮书介绍主流单边 *TCP* 加速机制的原理及优缺点，同时详细解析最新一代基于传输历史学习 (*Learning-based*) 的单边 *TCP* 传输加速技术 *ZetaTCP*[®]，及与其它主要 *TCP* 优化技术的比较分析。以下提到的 *TCP* 加速专指单边 *TCP* 加速。

1.TCP 加速技术的演进：三代技术

以拥塞判断及处理方式来区分，到目前为止，TCP 加速技术已经历了以下三代技术演进：

- *Loss-based* (基于丢包的拥塞判断及处理) – 以丢包来判断拥塞并调整传输速率；
- *Delay-based* (基于延迟的拥塞判断及处理) – 以往返延迟 (*Light TCP Tunnel,LTT*) 变化来判断拥塞并调整传输速率；
- *Learning-based* (基于学习的拥塞判断及处理) – 通过对路径上传输历史的学习动态判断拥塞并调整传输速率。

1.1.Loss-based 的 TCP 加速技术

Loss-based 的 TCP 加速技术沿袭了主流 TCP 以丢包来判断拥塞并调整传输速率的方式。其对传统 TCP 的改进主要是通过增大初始拥塞控制窗口(*Congestion Window, CNWD*)，同时在通过丢包判断出现拥塞后使用比传统 TCP 更激进的方式恢复 *CNWD*，以期减少拥塞对速率的影响。尽管这些改进在很多情况下确实能够提升速率，但 *Loss-based* 的 TCP 加速技术在原理上具有以下两个严重问题：

将丢包作为拥塞发生的信号很容易误判，导致传输速率下降，带宽得不到有效利用。现代很多网络会产生非拥塞因素的丢包，特别是对无线网络，如信号被干扰等因素导致的丢包并不意味有拥塞发生。

现代网络设备通常队列比较深，当拥塞发生时队列变长，延迟显著提高，但丢包迟迟不会发生。*Loss-based* 的 TCP 加速机制将继续高速传输直到队列完全充满，往往导致大量数据包丢失。这不但加重路径节点拥塞，而且需要花更长时间从大量丢包中恢复过来，经常会

导致传输阻滞。

*Loss-based TCP*加速包括在最新 *Linux* 系统中的 *Cubic TCP* 等。

1.2. Delay-based 的 TCP 加速技术

Delay-based 的 *TCP* 加速技术克服了 *Loss-based* 的主要缺陷，在原理上采用延迟的变化来判断拥塞程度并相应调整传输速度。这一机制更符合现代网络的特点，能够在发生拥塞网络节点的队列开始堆积时就及时下调传输速度，避免拥塞恶化，减少甚至避免丢包的发生。同时，*Delay-based* 的 *TCP* 加速技术不将丢包当作拥塞，在非拥塞因素发生的丢包时可以保持较高的速率。因此，设计优良的 *Delay-based TCP* 加速技术比 *Loss-based TCP* 加速技术在传输速度上有了显著的提升。尽管如此，*Delay-based TCP* 加速技术在使用中逐渐暴露出以下主要缺陷：

当 *TCP* 连接的路径上发生拥塞节点的队列很浅时，延迟并不提高，拥塞体现为阵发的丢包。*Delay-based* 的 *TCP* 加速技术感应不到这种拥塞，会继续高速发送数据包，从而导致持续大量丢包，之后的丢包恢复期会很长，导致传输速率显著降低。

当网络路径本身的延迟就变化很大时，*Delay-based* 的 *TCP* 加速技术会将非拥塞因素导致的延迟增加误判为拥塞并转入拥塞处理，从而导致没有必要的压低传输速率。包括移动互联网在内的无线网络延迟变化就很频繁；一些网络设备（特别是安全设备）也可能不定时地引入额外的数据包处理延迟。

FastTCP 是一个典型的 *Delay-based TCP* 加速。

1.3. Learning-based 的动态 TCP 加速技术

不论是 *Loss-based* 还是 *Delay-based* 的 *TCP* 加速技术都采用静态算法：基于对互联

网流量模型的假设前提采用固定的拥塞判断及恢复机制。但网络的发展趋势是流量特征越来越复杂并难以预测。因此，*Loss-based*和*Delay-based*的*TCP*加速技术常常只在其前提假设成立的特定网络场景下有效，并且随着传输的进行，网络路径特征发生变化，效果也会起伏不定，有时甚至出现反效果。为此，新一代*Learning-based TCP*加速技术采用网络路径特征自学习的动态算法，基于每一个*TCP*连接实时观察、分析网络特征，根据学习到的网络特征随时调整算法来更准确的判断拥塞程度、更及时地判断丢包，从而更恰当的进行拥塞处理并更快速的进行丢包恢复。这一设计从原理上克服了静态算法无法适应网络路径特征变化的问题，保证了在各种不同网络环境及频繁变化的网络延迟、丢包特征下加速效果的持续有效性。

本白皮书以下部分将详细解析最新一代*Learning-based TCP*加速技术*ZetaTCP*[®]。

2. *ZetaTCP*[®]的设计原则

从2006年以*Learning-based*的全新理念开始设计*ZetaTCP*[®]加速技术。设计思想及后续的开发都基于以下原则。

2.1. 对应用及网络拓扑全透明

ZetaTCP[®]作为一个传输增强模块与标准*TCP*协议完全兼容，并不替代操作系统中原有的*TCP*协议栈。由于应用程序仍然与所在操作系统的*TCP*协议栈交互，*ZetaTCP*[®]对应用来说完全透明。当流量经过*ZetaTCP*[®]模块时，*ZetaTCP*[®]在不改变任何数据内容及*TCP*封装格式的情况下通过改变数据包的传送时机及重传时机来达到加速目的。由于这种完全透明的工作方式，不仅任何应用感觉不到*ZetaTCP*[®]加速的存在，任何网络上的设备（如路由、安全设备等）也完全感觉不到*ZetaTCP*[®]加速的存在。这保障了*ZetaTCP*[®]对

应用及网络拓扑的全透明，在部署时并不要求应用和现有网络拓扑、网络配置有任何改动。当 *ZetaTCP*[®]以硬件产品形式部署时，内嵌 *ZetaTCP*[®]的设备以透明网桥的形式接入需要加速的网络路径上即可，不需要对现有网络设备做任何配置改动。当 *ZetaTCP*[®]以软件形式直接安装在应用服务器、智能手机或笔记本电脑等用户终端上时，*ZetaTCP*[®]加速引擎作为一个内核模块位于操作系统 *TCP*协议栈和网络接口驱动模块之间，如图 1 所示。这样，*ZetaTCP*[®]不需要与用户应用有任何交互，也对具体网络接口没有任何要求，实现全透明安装及工作模式。

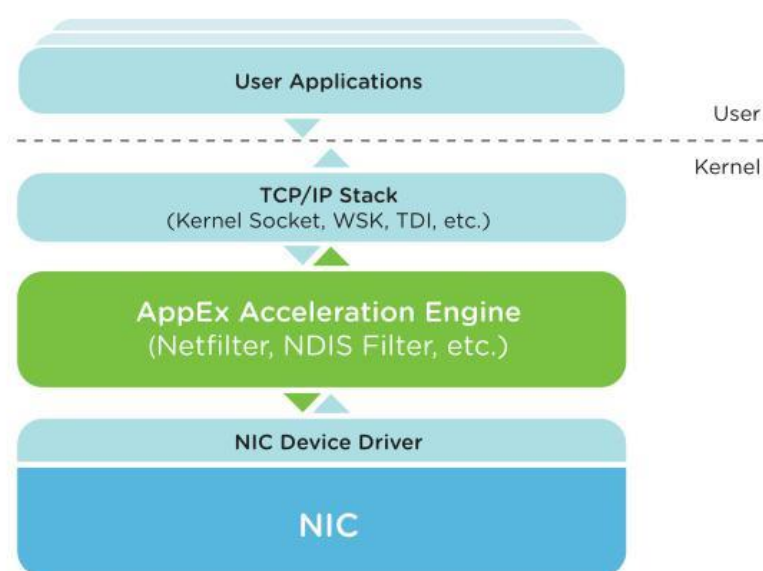


图 1 *ZetaTCP*[®]加速内核模块嵌入位置

2.2. Learning-based 动态算法

ZetaTCP[®]由一整套基于 *Learning-based*原则的专利算法组成。这些算法用于在传输过程中动态学习判断每个特定连接的网络路径特征，如端到端延迟及其变化特征、接收端反馈数据包 (*ACK*) 的到达间隔及其变化、数据包逆序程度及变化特征、可能由安全设备深度数据检测引起的延迟抖动、各种因素导致的随机丢包等。在实时跟踪这些特征的同时 *ZetaTCP*[®]随时综合分析这些特征并从中推导出在这个特定 *TCP* 连接网络路径上反映拥塞

及丢包的前兆信号，再根据这些动态智能学习的结果判断拥塞程度、决定与当前路径可用带宽相匹配的发送速度、拥塞恢复机制、并准确及时的进行丢包判断及恢复。

以上算法机制是通过一个自动学习状态机 (*Learning State-Machine*) 来实现，如图 2 所示。每一个 *TCP* 连接都对应一个 *Learning State-Machine*，记录该 *TCP* 连接的网络路径特征并动态决定适合该连接的拥塞判断及恢复机制和丢包判断机制。图 2 中的 *Flow Profiler* 主要负责直接提取网络路径外在特征并输入给 *Learning State-Machine*。*Learning State-Machine* 积累的智能被图 2 中所示的 *Loss Detector*、*Congestion Controller*、*Exception Handler* 及 *Window Controller* 模块用于调整相应 *TCP* 连接的传输行为。*Exception Handler* 和 *Window Controller* 模块也会动态反馈信息给 *Learning State-Machine* 用于进一步优化网络路径学习。

2.3. 可扩展性

ZetaTCP[®] 在设计上还具有非常好的可扩展性。这一可扩展性体现为两个方面：

- 向下扩展：最小化使用硬件资源（如处理器及内存资源），使 *ZetaTCP*[®] 可以嵌入在资源很少的低端的网络设备及终端，如低端家用路由器、智能手机、机顶盒等。
- 向上扩展：单个系统支持几百万甚至几千万的并发连接，使 *ZetaTCP*[®] 加速可以在运营商核心网络及大型数据中心出口进行部署。以上可扩展性主要是通过图 2 所示的 *Resource Manager* 模块的弹性资源管理来实现。*Resource Manager* 根据当前系统可用资源情况及 *Learning State-Machine* 模块输入的加速需求情况动态决定每一个 *TCP* 连接的可用资源，确保系统资源最合理的使用，达到加速效果和资源许可情况的平衡及最佳性价比。

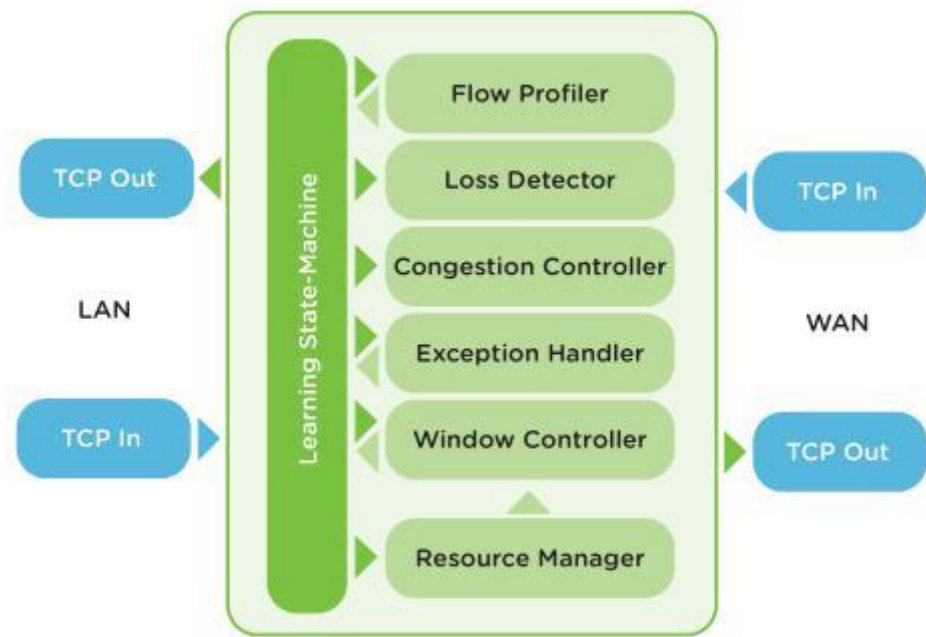


图 2 ZetaTCP[®]加速引擎架构

3. ZetaTCP[®]的主要加速机制

ZetaTCP[®]主要通过改善拥塞判断处理和更准确及时地进行丢包判断恢复来加速 TCP 传输并提升 TCP 连接的稳定性。

3.1. 更有效的拥塞判断及处理

标准 TCP 的拥塞判断及恢复是基于二十几年前的网络情况设计的，其基本假设是任何丢包都是拥塞的反映。这一假设与现代网络情况已完全脱节。现代网络丢包常常并不是由拥塞因素引起的。这一脱节导致标准 TCP 常常进入过于保守的传输状态。同时，当网络路径上都是深队列网络设备时，拥塞发生后很长时间内丢包都不会发生，标准 TCP 在这种情况下会对拥塞反应迟钝，导致过度传输，不仅加重了网络拥塞还造成大量丢包，从而陷入漫长的丢包恢复使传输阻滞。这些都导致标准 TCP 在现代网络环境下表现不佳。

以 FastTCP 为代表的 Delay-based TCP 加速技术将延迟的增加作为拥塞的信号，这更

适合现代网络的特点。但如本白皮书第一章的分析，*Delay-based TCP* 加速的拥塞判断及处理无法适应浅队列网络设备和设备处理引入延迟的场景，在现代网络环境下同样存在严重的适用性问题。

作为新一代 *TCP* 加速技术，*ZetaTCP*[®] 同时考虑丢包和延迟变化，并且创造性地引入了 *TCP* 连接路径网络特征自学习的动态算法机制来进一步提升拥塞判断的准确性和及时性。如本白皮书第二章所述，*ZetaTCP*[®] 通过动态学习分析的方式推导出特定 *TCP* 连接网络路径上反映拥塞的前兆信号，滤掉并非由拥塞导致的丢包和延迟变化，从而使拥塞判断更及时和准确。在此基础上，*ZetaTCP*[®] 算法也在随时总结逼近当前连接路径上的实际可用带宽，当拥塞发生时，使用当前最佳逼近结果来决定传输速率。

从算法语言上看，*ZetaTCP*[®] 拥塞判断算法是一个传输路径上各种网络特征的自动状态机。其功能是逐连接的智能学习并提高拥塞判断技能。所以用户常发现越是对长连接，加速效果越明显。该算法是自 2006 年以来在实际网络中不断研究、运行并持续优化的结果。目前仍在继续优化算法，这将是一个一直持续的优化过程。

ZetaTCP[®] 先进的拥塞判断及控制算法带来以下两点主要益处：

- **防止拥塞加剧**

拥塞加剧会导致两个问题：

- ◆ 由于产生了大量丢包，重传补洞的时间会很长，导致一段时间 *TCP* 传输窗口卡住，传输变得更慢甚至断线。
- ◆ 由于丢包增多，需要重传，重传率会提高，有效数据率降低。用户会发现线上的流量很大，但是实际应用速率没有对应提高。

ZetaTCP[®] 能够及时判断出拥塞，尽早减速，防止拥塞加剧并降低丢包数量，从而使传输变得更快更平滑，并提高了有效数据率。

- **防止拥塞后过度保守的传输**

当前 *TCP* 协议栈由于不清楚拥塞发生后连接路径上实际可用的带宽，往往采用大幅降低传输速率的方式进行恢复。这导致了路径带宽的闲置，是 *TCP* 传输效率低的主要原因之一。

ZetaTCP[®] 的拥塞判断算法在跟踪逼近当前实际可用带宽，发现拥塞后能够以最贴近当前路径可用带宽（使用稍低一些的传输速率以清空路径上的队列，贡献于拥塞网络节点的恢复）进行传输，从而防止带宽的闲置，使传输速率更高，更平滑。

- **Loss-based、Delay-based、和 Learning-based 拥塞控制比较分析**

本节通过常见的实际网络场景分析来比较 *Loss-based*、*Delay-based* 和 *Learning-based* 拥塞控制。图 3 显示了在四种常见网络场景下三种机制的表现。第一种场景是非拥塞因素产生的丢包，如无线网络中的信号干扰产生的数据包丢失等；第二种场景是在网络路径上某深队列设备节点产生了拥塞；第三种场景是在网络路径上某浅队列设备节点产生了拥塞；第四种场景是在网络路径上某网络设备在处理数据包时引入了额外的处理延迟，如网络安全设备在启动特殊策略或 *CPU* 突发高使用率时对数据包的处理时间加长。

如图 3 所示，*Delay-based TCP* 加速机制在前两种场景下能够避免误判拥塞，并及时判断出拥塞，显示出了比 *Loss-based TCP* 更高的性能，但在后两种场景下 *Delay-based* 算法的缺陷充分暴露：在第三种场景下 *Delay-based* 机制未能及时判断出拥塞，导致过度传输及大量数据包丢失，陷入缓慢的丢包恢复过程；在第四种场景下 *Delay-based* 机制将设备处理引入的额外延迟当作拥塞信号，导致没有必要的减速，从而降低了带宽利用率。而 *Learning-based ZetaTCP*[®] 能够根据对传输历史的动态学习掌握网络路径特征并准确判断不同丢包及延迟变化模式是否反映拥塞，所以在四种场景下都做出正确判断，并能最

快的从拥塞和丢包中恢复。因此 *Learning-based ZetaTCP*[®] 能够在这一传输时段取得最高的带宽利用率。

ZetaTCP[®]不只是在图 3 所示的四种场景下有明显优势，在大量其它网络场景下 *ZetaTCP*[®]通过动态学习积累的相应连接网络特征智能同样将帮助 *TCP*连接取得更好的性能及稳定性，而且一个连接传输持续的时间越长，*ZetaTCP*[®]的优势将越明显。

3.2. 基于预测的快速丢包判断及恢复

标准 *TCP*协议栈通过两种手段判断丢包：一是接收端连续重复确认包 (*Dup-ACK*) 的数量，二是 *ACK* 超时。当有较多丢包时，往往要靠 *ACK* 超时来判读超时并引发重传。现代网络的丢包经常是阵发，一个连接上有多个数据包同时丢失是常有的事。因此标准 *TCP* 经常要靠超时来重传补洞，往往导致几秒甚至上十秒的等待状态，让传输长时间停滞，甚至断线。这是影响标准 *TCP* 效率又一最主要问题。

ZetaTCP[®]的丢包判断除了采用标准 *TCP*的两种手段外，还创造性地引入了 *TCP* 连接路径网络特征自学习的动态算法机制来尽快地预测丢包。预测算法考虑的网络特征因素和 *ZetaTCP*[®]拥塞判断的自学习算法类似。通过传输历史的跟踪学习，*ZetaTCP*[®]丢包判断算法对每一个发出去尚未被对方 *ACK* 确认的数据包都给出一个已丢失概率，概率值会随着传输的进行不断改变，当概率值达到一个很高的阈值时，算法认为该数据包已丢，将立即启动重传。经过六年不断的改进算法，目前这一判断算法的准确率几乎达到 100% (准确率远高于标准 *TCP* 使用连续 *Dup-ACK* 判断丢包的准确率)。这一机制大大的降低需要依靠超时来判断丢包的机会，使 *TCP* 传输可以更快的补洞，平滑的传输，达到更高的平均速率。

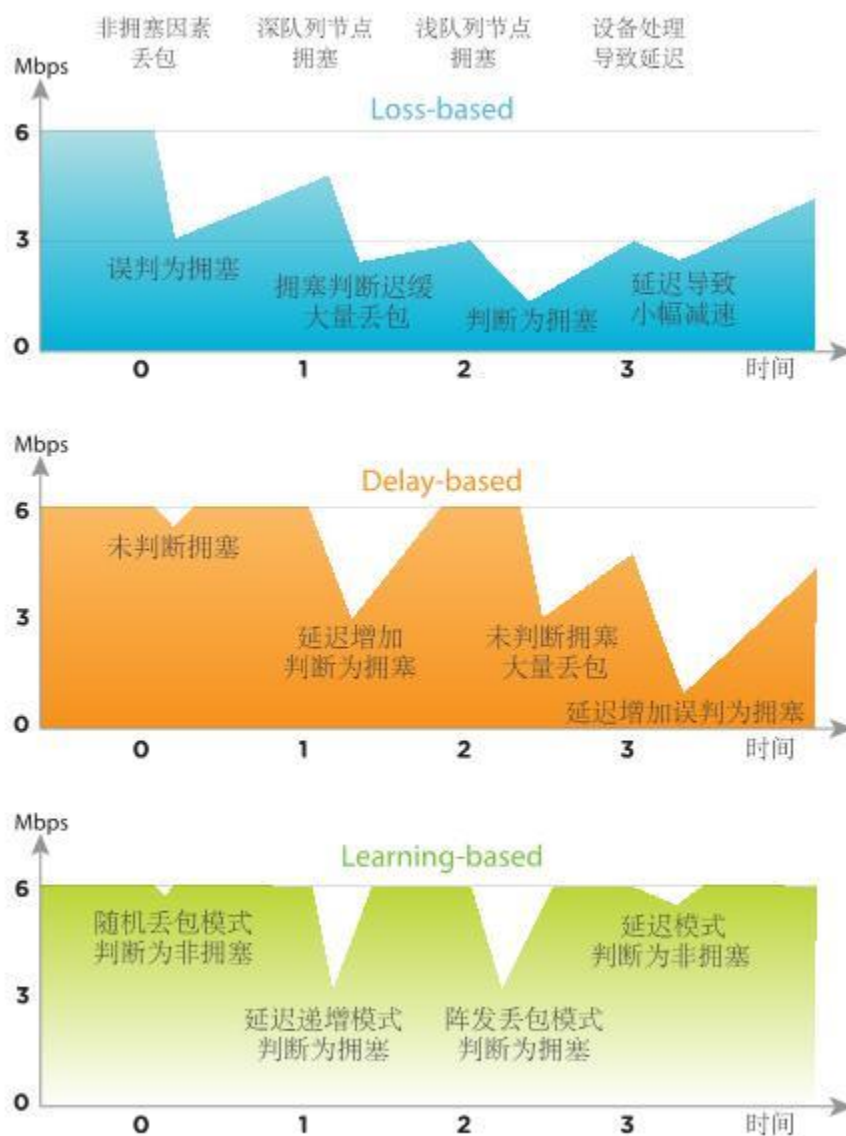


图 3 拥塞控制机制比较分析

图 2 加速引擎架构中所示的 *Loss Detector* 就是直接控制丢包预测的模块。*Loss Detector* 从 *Learning State-Machine* 中获得有关 *TCP* 连接路径网络特征，并用这一动态学习智能来预测并调整当前已发送数据包的丢失概率。

图 4 是真实测试中对经过 *ZetaTCP*[®] 加速和未经加速的标准 *TCP* 的传输速率比较结果。测试模拟了端到端 100 毫秒的延迟 ($RTT=100ms$) 和 1% 丢包率的网络路径。接收方 *TCP* 协议栈的最大接收窗口是 64K 字节，这一接收窗口值限制了最大可能带宽在 5.2Mbps。由于标准 *TCP* 丢包恢复的很慢，所以其传输速率远低于最大可能带宽，而经过

ZetaTCP[®]加速的 *TCP*连接能够快速从丢包中恢复，达到的传输带宽接近最大可能带宽。

快速丢包判断及恢复对目前使用日益普遍的无线网络，特别是移动无线网络有非常大的价值。移动网络经常容易因为信号损耗、干扰等原因产生丢包，其丢包率远大于有线网络。

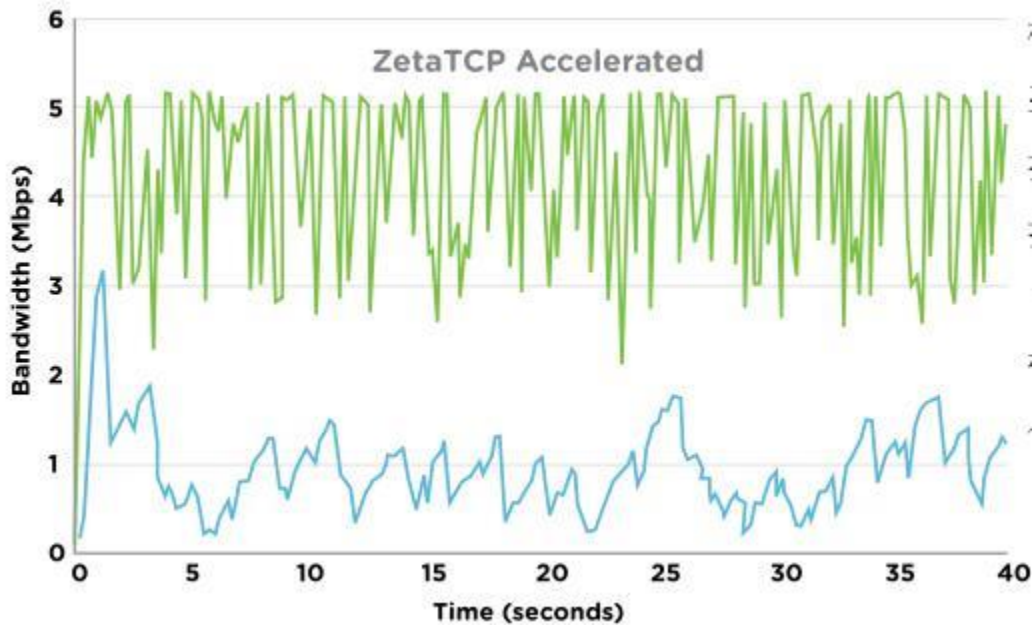


图 4 实测丢包环境下速率比较

标准 *TCP* 由于对这些丢包判断的不及时，传输效率常常很低而且传输质量不稳定，难以预测，严重影响用户体验。*ZetaTCP*[®]加速在无线网络环境下可以非常快的预测丢包并及时恢复，使传输更加平稳，速率更高，从而显著提升用户体验。

4. 第三方比较测试

图 5 实测比较了 *Loss-based*、*Delay-based*和 *Learning-based* *TCP*加速的效果。测试模拟了最高访问带宽 *16Mbps* 及各种常见网络延迟和丢包组合，并比较单个 *TCP*连接在不同组合下可以达到的吞吐率。为防止测试随机误差，每一个测试点是很多次测试所得吞吐率的平均值。*Loss-based*采用了在最新 Linux 内核版本中广泛使用的 *TCP Cubic*；*Delay-based*使用了目前广泛应用的一个商业化的 *TCP*加速产品；*Learning-*

*based*使用了 *ZetaTCP*[®]加速。从测试中可以看到 *Loss-based*在各种网络延迟和丢包组合下表现都最差，特别是当延迟和丢包率显著高的时候。*Delay-based*在性能上很显著的超越 *Loss-based*。而 *Learning-based ZetaTCP*[®]加速在三者中能提供最^高的传输速率，特别是在延迟和丢包较高的时候，*Learning-based ZetaTCP*[®]比 *Delay-based*加速技术显示出了非常明显的优势。

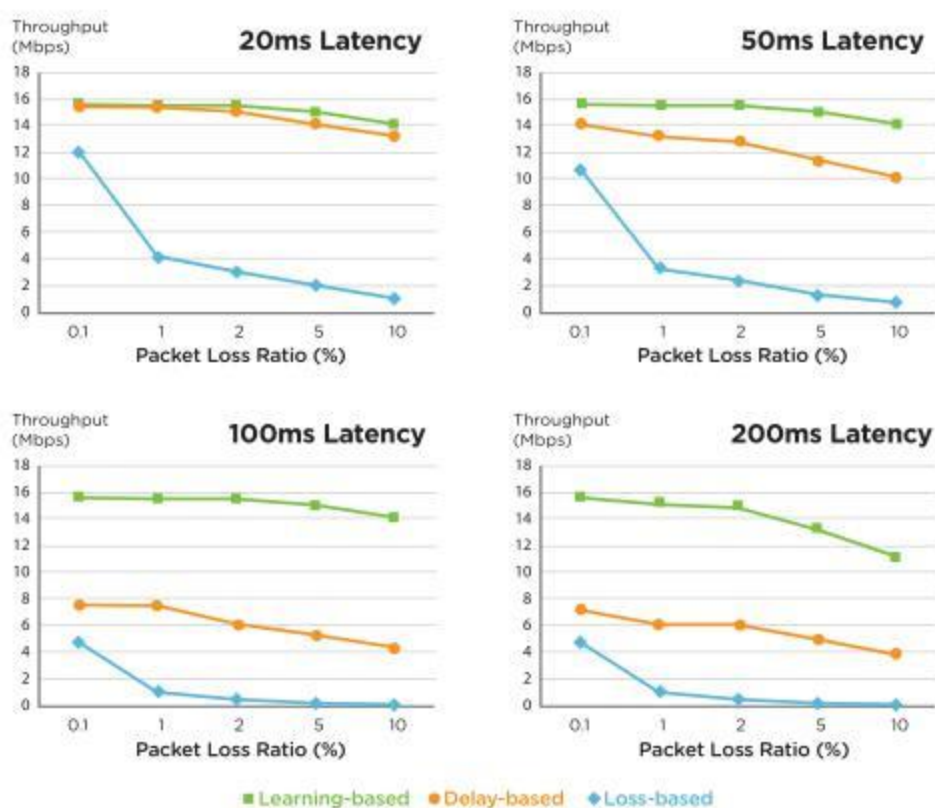


图 5 第三方比较测试结果

结论

互联网上超过 90%的流量通过 TCP 传输。而广泛使用的标准 TCP 协议已无法适应当前网络的复杂性和多样化。对 TCP 的加速优化经历了 *Loss-based*、*Delay-based* 和 *Learning-based* 三代技术演进。*Delay-based* 虽然比 *Loss-based* TCP 加速技术更符合

现代网络的特点，但其同属于静态拥塞判断算法，仍无法适应现代网络的多样化特征，特别是不适应无线网络信号的相关丢包特点。

ZetaTCP[®]作为最新一代 *Learning-based TCP* 加速技术从设计原理上解决了 *Loss-based*和 *Delay-based*的核心问题，通过针对每个 *TCP*连接传输动态学习来掌握其网络路径的各种特征，从而更及时的判断拥塞和丢包，并更快的进行拥塞恢复和丢包恢复。通过这一创新的动态学习算法机制，*ZetaTCP*[®]能够适应各种网络情况及网络特征的变化，达到更稳定及快速的传输效果，最终提升用户的体验。在架构设计上，*ZetaTCP*[®]不仅和标准 *TCP*全兼容，而且并不替代标准 *TCP*，只是作为一个传输增强模块，实现了对应用及网络拓扑的全透明部署。仅和标准 *TCP*全兼容，而且并不替代标准 *TCP*，只是作为一个传输增强模块，实现了对应用及网络拓扑的全透明部署。

北京华夏创新科技有限公司

地址：北京市海淀区北清路 68 号院 24 号楼 C 座 6 层 601 室

网址：<https://www.appexnetworks.com.cn/>

邮箱：marketing@appexnetworks.com

中国大陆区域：400-606-6118

北美区域：888-473-1388

韩国区域：003-0864-0156

其他区域：0086-29-81157308

