基于 UDP 的高速网络传输协议研究*

岳兆娟[1,2]，任勇毛[1]，李　俊[1]
(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)

摘　要：分析了基于 UDP 的改进高速网络传输协议的基本原理，详细阐述了近年来提出的一些典型的基于 UDP 改进传输协议的主要设计思想。通过建立端到端的高速网络试验床，评价了 RTT 及丢包率对 RUBDP、Tsunami、UDT、PA-UDP 四种传输协议吞吐率的影响，实验结果表明丢包率对于四种传输协议的吞吐率影响比较大。

关键词：高速网络；基于 UDP 传输协议；性能评价
中图分类号：TP393　　**文献标志码**：A　　**文章编号**：1001-3695(2013)10-2887-04
doi:10.3969/j.issn.1001-3695.2013.10.002

Research on UDP-based high-speed network transport protocols

YUE Zhao-juan[1,2], REN Yong-mao[1], LI Jun[1]
(1. *Computer Network Information Center, Chinese Academy of Sciences, Beijing* 100190, *China*; 2. *University of Chinese Academy of Sciences, Beijing* 100049, *China*)

Abstract: Thia paper firstly analyzed the basic principles of UDP-based improved protocols, and then elaborated the design ideas of these improved protocols proposed in recent years. By establishing end-to-end test bed, it evaluated the effect of RTT and loss rate to the four typical improved protocols (RBUDP, Tsunami, UDT, and PA-UDP). The experimental results show the throughput of the four improved protocols decreases with RTT and loss rate increasing. Finally this paper proposed some further research directions.

Key words: high-speed networks; UDP-based transport protocols; performance evaluation

近年来，各种新型的网络应用对网络带宽提出了越来越高的需求，如量子物理、地球观测等领域的海量数据传输。国际科研组织纷纷开始研究部署高速的下一代互联网络，并建立了高速网络试验床，如 UltraScience net (USN)[1]、CHEETAH[2]、APAN[3]、CAnet4[4] 等。中国科技网参与发起的多个国家参与的中美俄环球科教网络 GLORIAD(global ring network for advanced application development)[5] 是为了满足不同国家科学数据共享所建设的。随着下一代互联网建设及发展，高速长距离网络将成为其主流网络[6]。高速长距离网络有三个典型特征：a)高带宽，为了传输新型网络应用所产生的大规模的海量数据，高速网络的带宽至少为 622 Mbps(OC-12)，常见的有 2.5 Gbps(OC-48)、10 Gbps(OC-192)、甚至 40 Gbps(OC-768)；b)高延时，通常在国内传输一个报文的往返时延(RTT)是小于 30 ms，在高速长距离网络中，分布在全球不同地区的终端节点之间的物理距离比较大，因此传送报文往返延迟比较大，可以达到 300 ms 以上；c)光通路(lightpath)连接，相对于传统 IP 网络通过路由器进行分组交换的多点连接，高速光网络核心网主要由光通路通过电路交换连接，网络中间几乎不发生拥塞。而接入网也由数量比传统 IP 网络少得多的终端节点(如 10^3)组成[7]。

虽然高速网络带宽呈现指数性增长，但是用户在实际的大数据传输过程中却未感受到传输速率的快速增加。研究人员通过实验和理论分析发现，传输协议是造成带宽利用率不高的原因之一。传统的 TCP 是针对低速、低延迟的网络设计的，随着网络带宽及延时的增加，其 AIMD(加性增加、乘性减小)拥塞控制算法过于保守而不适用于高速长距离网络。近年来，一些研究人员提出基于 TCP 的适用于高速网络的改进协议如 BIC TCP[8]、FAST TCP[9]、CUBIC TCP[10] 等。同时，研究人员也提出了基于 UDP 的适用于高速网络的传输协议，此类改进协议使用 TCP 传输控制信息，使用 UDP 传输数据信息从而对传统的 UDP 增加了可靠性保证。

1　高速网络传输协议原理分析

本文主要分析了 RBUDP[11]、Tsunami[12]、UDT[13]、PA-UDP[14]、PAT[15]、PLUT[16] 等典型的高速网络传输协议。这些协议都是由发送端和接收端组成。接收端通过各种拥塞检测参数，按照一定的拥塞控制算法对拥塞状况进行检测，并周期性地反馈给发送端；发送端根据反馈采取相应的速率调整策略。表 1 列举了这些协议所采用的拥塞检测参数及拥塞检测算法，它们可以反映拥塞水平是加剧还是减弱。

2　高速网络传输协议介绍

2.1　RBUDP 协议

RBUDP 协议的发送端首先以用户指定的速率用 UDP 传送完所有的数据包，接收端接收数据并保持一个 tally 来确定哪些数据包已经被接收。当数据包全部发送完毕，发送端用 TCP 发送 DONE 信号表明数据包传送完毕；当接收端收到

收稿日期：2013-02-19；**修回日期**：2013-04-12　　**基金项目**：国家“863”计划资助项目(2011AA01A101)；中国科学院研究生科技创新项目
作者简介：岳兆娟(1984-)，女，河南驻马店人，博士研究生，主要研究方向为高速网络传输协议(yuezhaojuan@cstnet.cn)；任勇毛(1981-)，男，副研究员，博士研究生，主要研究方向为高速网络、传输协议；李俊(1968-)，男，研究员，博导，主要研究方向为下一代互联网、高速网络.

DONE 信号后,反馈一个带有 bitmap tally 的确认包,发送端根据接收到的 tally 来重传丢失的数据。此过程重复进行,直到接收端成功接收所有数据。

表 1 所分析协议的拥塞检测参数和拥塞检测算法

传输协议	拥塞控制参数	拥塞加剧	拥塞减弱
RBUDP	不检测	不检测	不检测
Tsunami	周期内的丢包率	丢包率大于错误门限	丢包率小于错误门限
SABUL	周期内的丢包率	丢包率大于错误门限	丢包率小于错误门限
UDT	周期内的丢包率	丢包率大于错误门限	丢包率小于错误门限
PA-UDP	剩余文件的大小 bitleft,剩余的缓存空间 memleft,接收端硬盘的写速率 r(disk),接收端接收数据的速率 r(recv)	bitleft > memleft	bitleft ≤ memleft
HURRICAN	周期内窗口大小 $w_c(t)$ 及睡眠时间 $T_s(t)$,丢包率	丢包率增加	丢包率减少
PAT	周期内窗口大小 $w_c(t)$ 及睡眠时间 $T_s(t)$,重传率 $x(t)$,ACK 之间的间隔 $I(t)$,CPU-bound 进程加入或者离开 CPU 运行队列的个数 n	n 个 CUP-bound 进程加入到运行队列, $R_{br}(t)$ 减少($R_{br}(t)$ 为瓶颈带宽)	n 个 CUP-bound 进程离开运行队列, $R_{br}(t)$ 增加
PLUT	周期内窗口大小 $w_c(t)$ 及睡眠时间 $T_s(t)$,重传率 $x(t)$,ACK 之间的间隔 $I(t)$	重传率增加,吞吐率减少	重传率减少,吞吐率增加
Para-PLUT	周期内窗口大小 $w_c(t)$ 及睡眠时间 $T_s(t)$,重传率 $x(t)$,ACK 之间的间隔 $I(t)$,并行流的数量 m	$G(m) < G(m-1)$($G(m)$ 为多个流的聚合吞吐率)	$G(m) \geqslant G(m-1)$
FRTP	不检测	不检测	不检测

在高速长距离光网络环境中,RBUDP 协议没有拥塞控制机制,以恒定速率持续发送的策略极易引起接收端缓冲区溢出而产生大量丢包[17]。针对 RBUDP 协议未考虑终端性能对传输效率的影响,研究人员提出了 RBUDP + 协议[18]。

2.2 Tsunami 协议

Tsunami 协议摈弃了 TCP 对于收到数据发送 ACK 确认的滑动窗口机制,而是周期性地对未收到的数据分组发送 NAK(negative acknowledgement),其结构如图 1 所示。尽管该协议用在几乎不发生网络拥塞的高速网络环境中,但是为了避免发送速率超过终端的处理数据能力,影响网络传输性能,该协议采用了流控制机制。Tsunami 协议设置了一个可以动态调整的错误门限 errorthreshold,接收端周期性地计算当前分组错误率 lossrate 并反馈给发送端。当分组 lossrate > errorthreshold 时,发送端增加分组间的间隔,当分组 lossrate < errorthreshold 时,发送端减少分组间的间隔,直到达到目标速率。

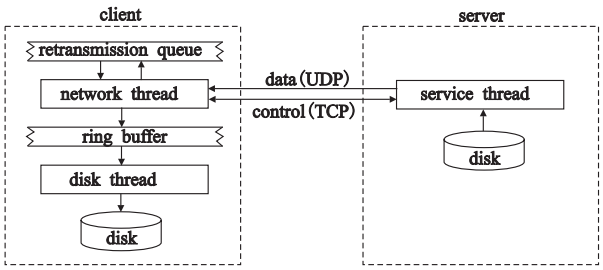

图1 Tsunami协议结构图

Tsunami 协议增加了基于丢包率的拥塞控制机制。但是其拥塞控制机制过于简单,丢包率不能准确反映拥塞状况,尤其是在 RTT 比较大的高速长距离网络中,当发送端基于反馈进行速率调整的时候,拥塞状况可能已经解除。其改进的 RT-sunami[19] 协议通过拥塞趋势提前预测拥塞水平并采取相应的拥塞避免策略。

2.3 PA-UDP 协议

通常硬盘的处理速率是不到网络传输速率的一半,并且高速数据的处理也更占用 CPU,因此,发送速率和终端处理速率不匹配将导致接收端的缓冲区溢出。PA-UDP 协议由发送端和接收端组成,发送端通过三次握手发送一个初始速率,接收端周期性地计算接收速率 r(recv)、硬盘处理速率 r(disk)及剩余缓存大小 m 来重置新的发送速率 newrate。newrate = αr(disk)(其中 $\alpha = 1/(1 - m/f)$,m 为剩余缓存大小,f 为剩余文件大小)。接收端把新的发送速率 newrate 反馈给发送端,发送端再根据新的发送速率调整分组间隔 $t_d = \dfrac{L + \sqrt{L^2 - 4\beta LR}}{2R}$($R$ 代表接收端要求的新的发送速率,β 为调整因子,L 为分组大小)。PA-UDP 协议建立一个有效利用缓存和 CPU 管理的数学模型,并且充分考虑了网络性能、CPU 性能及硬盘的读写性能,使其在不同的终端系统下动态和自动地最大化传输性能。

2.4 SABUL 协议

SABUL[20] 协议在 UDP 上增加了端到端的可靠性、基于速率的拥塞控制及流控制机制。SABUL 协议通过错误控制策略实现可靠性,它对于每个 UDP 数据包增加一个顺序码,当接收端检测到数据包丢失时,周期性地发送 NAK 数据包告知发送端,发送端重传丢失的数据包。发送端在内存中保存发送的数据包,当收到 ACK 时,再清除已经确认过的数据包。

SABUL 协议使用基于速率的拥塞控制策略。接收端周期性地发送 SYN 控制数据包,该数据包记录上一个 SYN 周期内接收到的数据包的数量。发送端用 SYN 控制数据包中的信息评估分组丢失从而评估网路的拥塞程度。当丢包率低于丢包门限值时,发送端减少分组间隔,增大发送速率;当丢包率高于门限值时,发送端增大分组间隔,减少发送速率。

SABUL 部署在用户空间,所以接收端不能区分出网络拥塞发生的丢包还是接收端缓存溢出发生的丢包[21]。用一个固定的窗口来限制网络中不被确认的数据包的数量,当发生拥塞时,限制数据包丢失的数量。

2.5 UDT 协议

UDT 协议是 SABUL 的最新版本,该协议把基于速率的拥塞控制和基于窗口的流控制机制相结合,速率控制周期性地更新发送分组间的间隔,流控制技术基于收到的 ACK 更新流窗口的大小。UDT 协议拥塞控制分为慢启动和拥塞避免阶段,在拥塞避免阶段采用 DAIMD(递减式增加的 AIMD)拥塞控制算法,在发送初期,带宽比较充足,发生拥塞的概率比较低,所以分组传输速率增加得比较快;当接近可用带宽时,发送端减少增加发送分组的传输速率,降低了拥塞发生的概率。

UDT 协议采用了 SACK 机制,即接收端不是收到一个数据包就发送 ACK,而是周期性(默认为 0.01 s)地发送 ACK,当检测到分组丢失时,就发送 NAK,这样保证数据传输的可靠性。UDT 协议主要是用在共享链路的网络环境,并考虑了有效性、公平性和稳定性。

2.6 FRTP 协议

FRTP(fixed rate transport protocol)[21] 是基于 SABUL 的改

进协议。FRTP 去掉了 SABUL 的速率控制策略,发送端不再基于丢包率调整分组间的间隔,从而改变发送速率,而是通过固定的分组间隔来保持一个固定的发送速率。

假定分组长度为 1500 Byte,1 Gbps 的发送速率要求分组间的间隔是 12 μs,而操作系统的时间粒度是 10 ms,所以 FRTP 采用 busy waiting 的方法控制分组间的间隔从而保证稳定的发送速率。当下个分组的发送时间是 t,若此时系统的当前时间 currenttime 小于下个分组的发送时间 t,即 currenttime < t 时,执行 NOP 指令。但是 NOP 是一个无操作的指令,浪费处理器资源,当处理器上运行多个进程的时候会相互干扰。当发送端以恒定的发送速率发送数据包,如果接收端不能以相匹配的速率接收数据包时,将导致接收端缓冲区溢出而发生重传,降低吞吐率。

2.7　Hurricane 协议

Hurricane[22] 协议主要应用于高速专用网络,使用该协议传输的流并不与其他流共享网络,其主要目标是最大化带宽利用率。其结构如图 2 所示,它与 SABUL 相似,由发送端和接收端组成,发送速率由拥塞窗口的大小 $w_c(t)$ 及睡眠时间 $T_s(t)$ 两个参数控制。在发送初期,发送端以一个比较小的发送速率发送数据包,接收端接收到数据包后,将有序的数据包直接写入硬盘,对无序的数据包在缓存中进行重组,同时把丢失的数据包放进丢包队列,并周期性地将丢包反馈给发送端。发送端重新加载丢失的数据包并进行重传。

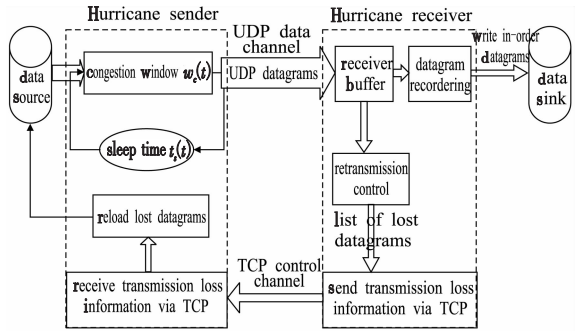

图2　Hurricane协议结构图

在最初速率调整过程中,发送端逐渐增加发送速率,此时丢包率增加较小,吞吐率呈线性增加,当增加到一定程度时,吞吐率保持稳定或者减少,此时的稳定吞吐率就接近网络最大传输速率,相应的发送速率为最优初始发送速率。一旦确定最优初始发送速率,发送端将根据最优初始速率调节拥塞窗口的大小 $w_c(t)$ 及睡眠时间 $T_s(t)$ 以获得较好的网络传输效率。该协议在开始传输的时候需要通过手动调节参数使传输速率逼近最优,而这个过程非常耗时。

2.8　PLUT 协议

高速网络传输协议需要手动调整初始速率,非常耗时和不方便使用,PLUT 协议采用自动的方式调整初始速率以使吞吐率最大化。其结构与 Hurricane 相似,由发送端和接收端组成,发送速率由拥塞窗口的大小 $w_c(t)$ 及睡眠时间 $T_s(t)$ 两个参数控制。接收端通过重传率、初始发送速率计算出接收数据的吞吐率,并反馈给发送端。发送端使用随机逼近的算法(RMSA)使发送速率逼近最大可用带宽并且使重传率比较低。在高速网络环境中传输时,过多地发送确认信息会占用接收端 CPU 时间,也会影响发送端发送数据的效率,PLUT 协议接收端使用 Keifer-Wolfowitz 梯度递减方法计算合适的间隔发送 ACK 确认以最大化吞吐率。

2.9　PAT 协议

PAT 协议是基于 UDP 的协议,使用性能自适应的流控制机制管理发送端和接收端以便获得高吞吐率。在传统网络中,分组丢失发生在与其他流竞争的情况,而在高速网络中,拥塞往往发生在终端。设发送端发送分组的速率是 λ,接收端分组的接收速率是 μ,当发送端发送分组的速率大于接收端接收分组的速率即 $\lambda > \mu$ 时,可能会导致接收端主机的缓存溢出,从而导致数据分组丢失。PAT 的流控制机制就是为了避免发送端发送速率过快,而接收端不能及时处理而导致的分组丢失。分组丢失可能发生在 ring buffer 或者是 socket receive buffer 或者同时发生[28]。PAT 协议主要考虑分组丢失发生在 socket receive buffer 的情况。该协议基于马尔可夫状态转换图对接收端的 socket receive buffer 及数据处理过程进行建模,接收端评估出瓶颈带宽 $R_{br}(t)$ 并反馈给发送端,发送端使用随机逼近的算法(RMSA)使发送速率逼近瓶颈速率。PAT 协议考虑有背景流量的情况,通过 PAT monitor 监测加入或者离开的 CPU-bound 进程,并以此调整瓶颈带宽。

2.10　Para-PLUT 协议

Para-PLUT[23] 是 PLUT 的改进协议。当终端主机是多核处理器时,该协议建立多个并行 UDP 连接进行传输,充分利用处理器资源,最大化聚合吞吐率。现有基于 UDP 的改进协议大多都是建立单个 UDP 连接,使用不同的线程接收、确认、存储分组。随着多核处理器的广泛使用,接收端只用一个线程处理到达的分组,缺乏并行处理的策略,不能充分利用终端的资源。

该协议通过数学模型比较了建立单个连接和多个连接的传输性能。假定分组到达率是 λ,分组的处理能力是 μ,当 $\lambda < \mu$ 时,两种方法有相同的吞吐率,平均响应时间的差别也非常小,但建立单个连接会浪费掉部分系统资源。当 $\mu < \lambda < 3\mu$ 时,建立单个连接的系统响应时间比较大,吞吐率比较低,建立多个连接的系统充分利用了终端 CPU 资源,获得较小的响应时间及较高的吞吐率。Para-PLUT 协议最初建立一个 UDP 连接,通过接收端速率评估,发送端逐渐增加并行连接的数量,直到接收端的聚合吞吐率 $G(m)$ 下降,此时的并行连接数量即为最大的并行流的值。发送端基于 ID 把数据分组分割成不同的部分,然后分配给不同的流传输。

其他的基于 UDP 的改进协议还有 RAPID[24]、RAPID +[25]、GTP[7]、FOBS[26] 等。RAPID 和 RAPID + 协议主要考虑终端性能对网络传输效率的影响;GTP 协议主要关注在多点到单点的通信模式下,多个流之间如何公平地分配带宽;FOBS 是一个轻量级的协议,其主要应用于网格计算。

3　性能分析

通过搭建端到端的试验床来评价这些传输协议的性能。高速长距离网络试验床由两台高性能终端服务器通过千兆以太网链路连接中间的 NETEM[27] 仿真器组成。网络仿真器 NETEM 可以改变网络的链路带宽、丢包率、延迟等网络参数。在实验中,网络带宽取 1000 Mbps、RTT 的取值范围为 2 ms ~ 320 ms、丢包率取值为 0.001% ~0.1%。

本文选取上述协议中的 RUBDP、Tsunami、UDT、PA-UDP 四种开源协议进行测试。PA-UDP 只提供了发送虚拟文件的传输方式,即由发送端自动生成指定大小的文件,所以在实验

过程中,使用 PA-UDP 传输时所传送的文件是由其发送端自动生成的;其余三种协议均采用了实际文件传输,传输文件大小取为 2 GB。

当带宽是 1000 Mbps、丢包率取 0.01% 时,测试不同 RTT 对传输性能的影响,实验结果如图 3[29] 所示。当带宽是 1000 Mbps,RTT 取 160 ms 时,测试不同丢包率对传输性能的影响,实验结果如图 4[29] 所示。

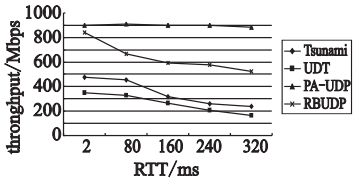

图3 RTT对吞吐率的影响

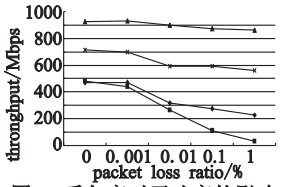

图4 丢包率对吞吐率的影响

实验结果表明,PA-UDP 获得了最佳传输性能,而 UDT 协议由于不需要设定发送速率、缓存大小等参数,使用最方便。RBUDP、Tsunami、UDT 协议随着 RTT 的增加、丢包率的增加,传输速率下降,尤其是受丢包率的影响比较大。

4 进一步研究方向

本文主要分析了近几年提出的基于 UDP 改进协议的基本原理,通过对这些改进协议的深入分析,笔者认为还有以下几个方面需要进一步研究:

a)用户友好性。通过协议分析及测试可以发现,大部分的协议一般需要用 Iperf 或者 Netperf 等工具测量网络的瓶颈带宽,然后由用户手动设定初始速率。初始速率设定过小,不能充分利用网络带宽;初始速率设定过大,则易造成网络拥塞,降低传输性能。因此在设计协议时,考虑自适应地设定初始速率将大大提高用户友好性,同时提高协议的稳定性。

b)内部流和外部流的公平性。随着 e-science 各种应用的发展,高速光网络的通信模式发生了转变,由点到点传输结构向多点到多点的传输结构转变[7]。因此,如何提高多流传输时的内部和外部流公平性,使相同的流获得相同的服务,同时保持较高的吞吐率及低的丢包率是多流传输面临的挑战。

c)终端性能影响。当分布在全球各地的数据共享时,由于核心网有足够的带宽,通常不会发生拥塞。而终端节点需要对大量的数据进行处理,如存储、计算等,其处理能力通常小于高速网络的传输能力。因此,在高速长距离网络中,网络拥塞发生在终端节点而不是网络中间,如何从提高终端性能角度考虑来提高网络的利用率将是进一步需要研究的方向。

参考文献:

[1] RAO N S V, WING W R, CARTER S M, *et al*. Ultrascience net: network testbed for large-scale science applications[J]. IEEE Communications Magazine, 2005, 43(11): S12-S17.

[2] ZHENG Xuan, VEERARAGHAVAN M, RAO N S V, *et al*. CHEETAH: circuit-switched high-speed end-to-end transport architecture testbed[J]. IEEE Communication Magazine, 2005, 43(8): S11-S17.

[3] APAN[EB/OL]. http://www.apan.net.

[4] CAnet4[EB/OL]. http://www.canarie.ca/canet4/index.html.

[5] GLORIAD[EB/OL]. http://www.gloriad.org/gloriad/index.html.

[6] 黄小猛, 林闯, 任丰源. 高速传输协议研究进展[J]. 计算机学报, 2006, 29(11): 1901-1908.

[7] WU R X, CHIEN A A. GTP: group transport protocol for Lambda-Grids[C]//Proc of IEEE International Symposium on Cluster Computing and the Grid. 2004: 228-238.

[8] XU Li-song, HARFOUSH K, RHEE I. Binary increase congestion control(BIC) for fast long-distance networks[C]//Proc of IEEE INFOCOM. 2004: 2514-2524.

[9] WEI D X, JIN Cheng, LOW S H, *et al*. FAST TCP: motivation, architecture, algorithms, performance[J]. IEEE/ACM Trans on Networking, 2006, 14(6): 1246-1259.

[10] HA S, RHEE I, XU Li-Song. CUBIC: a new TCP-friendly high-speed TCP variant[J]. ACM SIGOPS Operating System Review, 2008, 42(5): 64-74.

[11] HE E, LEIGH J, YU O, *et al*. Reliable blast UDP: predictable high performance bulk data transfer[C]//Proc of IEEE International Conference on Cluster Computing. Washington DC: IEEE Computer Society, 2002: 317-324.

[12] MEISS M R. Tsunami: a high-speed rate-controlled protocol for file transfer[EB/OL]. http://citeseerx.ist.psd.edu/viewdoc/download?-doi=10.1.1.113.4551.

[13] GU Yun-hong, GROSSMAN R L. UDT: UDP-based data transfer for high-speed wide area networks[J]. Computer Networks, 2007, 51(7): 1777-1799.

[14] ECKART B, HE Xu-bin, WU Qi-shi. Performance adaptive UDP for high-speed bulk data transfer over dedicated links[C]//Proc of IEEE International Symposium on Parallel & Distributed Processing. 2008: 1-10.

[15] LU Xu-kang, WU Qi-shi, RAO N S V, *et al*. On performance-adaptive flow control for large data transfer in high speed networks[C]//Proc of the 28th IEEE International Conference on Performance Computing and Communications. 2009: 49-56.

[16] WU Qi-shi, RAO N S V, LU Xu-kang. On transport methods for peak utilization of dedicated connections[C]//Proc of the 6th International Conference on Broadband Communications, Networks, and Systems. 2009: 1-8.

[17] 任勇毛, 唐海娜, 李俊, 等. 高速长距离网络传输协议[J]. 计算机学报, 2010, 21(7): 1576-1588.

[18] DATTA P, SHARMA S, FENG Wu-chun. A feedback mechanism for network scheduling in LambdaGrids[C]//Proc of the 6th International Symposium on Cluster Computing and the Grid. Washington DC: IEEE Computer Society, 2006: 584-591.

[19] REN Yong-mao, TANG Hai-na, LI Jun, *et al*. A novel congestion control algorithm for high performance bulk data transfer[C]//Proc of the 8th IEEE International Symposium on Network Computing and Applications. Washington DC: IEEE Computer Society, 2009: 288-291.

[20] GU Yun-hong, GROSSMAN R. SABUL: a transport protocol for grid computing[J]. Journal of Grid Computing, 2003, 1(4): 377-386.

[21] ZHENG Xuan, MUDAMBI A P, VEERARAGHAVAN M. FRTP: fixed rate transport protocol: a modified version of SABUL for end-to-end circuits[C]//Proc of Broadnet. 2004: 1-11.

[22] RAO N S V, WU Qi-shi, CARTER S M, *et al*. High-speed dedicated channels and experimental results with Hurricane protocol[J]. Annals of Telecommunications, 2006, 61(1-2): 21-45.

[23] LU Xu-kang, WU Qi-shi, RAO N S V, *et al*. On parallel UDP-based transport control over dedicated connections[C]//Proc of IEEE Global Telecommunications Conference. 2010: 1-5.

[24] BANERJEE A, FENG Wu-chun, MUKHERJEE B, *et al*. RAPID: an end-system aware protocol for intelligent data transfer over Lambda Grids[C]//Proc of the 20th International Parallel and Distributed Processing Symposium. 2006.

[25] DATTA P, FENG Wu-chun, SHARMA S. End-system aware, rate-adaptive protocol for network transport in LambdaGrid environments[C]//Proc of Supercomputing Conference. New York: ACM Press, 2006.

[26] DICKENS P M. FOBS: a lightweight communication protocol for grid computing[C]//Proc of the 9th International Euro-Par Conference. 2003: 938-946.

[27] NETEM[EB/OL]. http://www.linuxfoundation.org/collaborate/workgroups/networking/netem.

[28] RIO M, GOUTELLE M, KELLY T, *et al*. A map of the networking code in Linux kernel 2.4.20, DataTAG-2004-1[R]. [S.l.]: Data Tag, 2004.

[29] YUE Zhao-juan, REN Yong-mao, LI Jun. Performance evaluation of UDP-based high-speed transport protocols[C]//Proc of the 2nd IEEE International Conference on Software Engineering and Service Science. 2011: 69-73.