Information Technology Journal, 2010
ISSN 1812-5638

Characterization and Evaluation of End-System Performance Aware Transport Schemes for Fast Long-Distance Optical Networks

[1,2]Weihang Wang, [1,2]Mingjie Tang, [1]Yongmao Ren and [1]Jun Li
[1]Computer Network Information Center, Chinese Academy of Sciences, Beijing, China
[2]Graduate University of Chinese Academy of Sciences, Beijing, China

Abstract: Fast long-distance optical networks are emerging for the purpose of transmitting large amounts of scientific data among research institutions quickly and reliably, standard TCP (Reno TCP) does not perform well because it fails to saturate link throughput in such high-bandwidth environment. Moreover, some novel study indicated that the network speed had outstripped the processing capacity of end nodes, thus several new protocols which are based on end-system performance monitoring has been proposed to take advantage of such networks. It is critical to provide a just and complete evaluation of these protocols. In this study, we compare two promising end-system performance aware rate adjustment schemes. First, a modified rate-based version of RBUDP which adjusts rate based on end-node packet loss ratio (Tsunami). Second, an end-system based scheme which takes buffer and CPU management into consideration (PA-UDP). Our experiments use a range of performance metrics, including throughput, intra-protocol fairness, TCP friendliness and rate adaptation speed to flow dynamics. The results provide insights into the effectiveness of these schemes and also for improvements in design and implementation of end-system performance aware protocols.

Key words: Transport protocol, performance evaluation, fast long-distance network, end-system performance

INTRODUCTION

With huge amount of data gathered in fields such as high energy and nuclear physics, astronomy and bioinformatics, such scientific applications need to be able to transfer increasingly large amount of data between remote locations. Toward this goal, Fast Long-Distance optical network (FLDnet) has emerged in the internet field. Examples of networks that enable FLDnet include DOE's Ultra Science Net (USN) (Rao *et al*., 2005), dynamic resource allocation via, GMPLS Optical Networks (DRAGON) (Lehman *et al*., 2006), Global Ring Network for Advanced Applications Development (GLORIAD) and others.

In contrast to shared packet-switched IP networks, fast long-distance networks are characterized by dedicated high speed links (e.g., 2.5, 10 Gbps), Large Round Trip (RTT) time and low packet loss ratio (10^{-9}~10^{-11}) (Martin-Flatin and Ravot, 2002) thus, providing an environment with no internal network congestion. Since, end-to-end dedicated link bandwidth matches or exceeds processing speeds in end systems (Smarr *et al*., 2003), contention and sharing bottlenecks are pushed to the end systems. These differences pose a new set of research challenges for network communication in FLDnet.

Fast long-distance networks typically span over large intra-continental or inter-continental distances, thus resulting in networks with large Bandwidth-Delay Products (BDP). Delivering high performance bulk data transfer in large BDP networks has been a long standing research challenge. Traditional TCP (Poster, 1981) and its variants (Brakmo and Peterson, 1995; Mathis *et al*., 1996; Floyd, 2003; Jin *et al*., 2004; Kelly, 2003; Leith and Shorten, 2004; Xu *et al*., 2004) were designed for shared, low bandwidth networks in which the bandwidth on internal links is a critical and limited sources and its performance is highly dependent on the bandwidth-delay product of the network (Jacobson *et al*., 1992). The TCP's slow start causes TCP to take a long time to reach full bandwidth when RTT is large and to recover from packet loss because of its Additive Increase Multiplicative Decrease (AIMD) (Allman *et al*., 1999) control law. As a result, a number of end-system performance aware schemes (Banerjee *et al*., 2006; Mark, 2002; Wu and Chien, 2004; Datta *et al*., 2006a, b; Eckart *et al*., 2008; Ren *et al*., 2009) have been proposed to overcome the limitations of TCP/IP in large BDP networks, an environment with no internal congestion but significant endpoint congestion. These solutions are rate-adaptive and able to fill high bandwidth-delay product networks by

Corresponding Author: Weihang Wang, CNIC 303. No. 4, 4th South Street, ZhongGuanCun, Haidian District, P.O. Box 349, Beijing, China Tel: (86)138 10345969

end-system performance monitoring. They also provide reliable transport services. We classify these protocols into three categories according to their congestion detection metrics, process schedule based, packet loss ratio and buffer occupancy.

In this study, we distinguish the key characteristics of FLDnet and categorize current end-system performance aware protocols. We will analyze the performance and fairness of two representative schemes under different communication patterns. With various workloads, we study these protocols using measurements on our testbed FLDnet (Yongmao *et al*., 2009). The primary contributions of this study are summarized below:

- Definition of the key characteristics of fast long-distance networks and communication challenges in FLDnet
- Classification and characterization of existing end-system performance aware transport schemes
- Evaluation of representatives of two promising end-system performance aware schemes (Tsunami, PA-UDP) for high speed single and parallel flows: all achieve high bandwidth when RTT is small, with a large round trip delay (80 m sec), only PA-UDP can maintain high speed (around 650 Mbps)
- Evaluation of these two protocols on intra-protocol fairness which shows all exhibit good intra-protocol fairness for 2 and 4 parallel flows
- Evaluation of these two protocols on TCP friendliness, which shows that Tsunami is more TCP friendly than PA-UDP
- Evaluation for workloads with rapid flow changes which shows that Tsunami does not capture the available bandwidth efficiently. The PA-UDP manages rapid flow transitions better and efficiently explores the available bandwidth

Present results suggested that probing receiver's capacity for flows in fast long-distance link is a challenging problem. Our experiments showed that buffer and CPU management based, of which PA-UDP is an exemplar, is a promising direction and deserves further investigation for the new networking environment of FLDnet.

COMMUNICATION IN FLDNET

Characteristics of FLDnet: A fast long-distance network is a set of distributed resources directly connected with fiber optic and Optical Cross-Connect (OXC), which is distinguished from traditional shared packet-switched IP network by its dramatic higher performance and quality of service. The key distinguishing characteristics of fast long-distance networks are:

- **High speed: (e.g., OC-192 = 10 Gbps):** It is dedicated links provide end-to-end light paths with enormous bandwidth, which can be used to meet the exigent requirement of transmission for huge amount of scientific data. The bottleneck in our experiments is a 1 Gbps dedicated link
- **Large round trip delay:** The FLDnet typically span large intra-continental or inter-continental distances, thus resulting in networks with large Bandwidth-Delay Products (BDP), which made TCP inefficient in such environment because of TCP's slow start and AIMD control law as described above
- **Low packet loss rate:** Compared with Bit Error Rate (BER) of other medium, fiber optic BER is significantly low, usually 10^{-9}~10^{-11} (taking error caused by network devices into account). In addition, no router exists in end-to-end light paths, so, no packets have to wait in queue and the packet loss ratio is very low. Traditional TCP's error control mechanism becomes redundant in such networks

In a fast long-distance network, one can view the optical connections between end systems as fast, dedicated connections, in contrast to shared links which are packet-switched by IP routers.

Communication challenges in FLDnet: Fast long-distance networks have distinguishing characteristics different from traditional IP networks. The standard TCP is designed for traditional packet-switched internet, thus resulting in its inability of saturating the bandwidth and introducing several urgent challenges in such environments are:

- **High throughput:** The most important requirement of e-science applications is high transfer speed. So throughput is main comparative metric in our study. Data transport protocols should be aggressive enough to employ all of the receiver's capacity
- **Intra Protocol Fairness Among Flows:** An important design goal for multi-flow communication is to provide predictable performance to each flow. This requires rate allocation of multiple flows to meet certain fairness criteria, such as max-min fairness (Bertsekas and Gallager, 1992). Intra-protocol fairness assures all flows following the same protocol receive the same level of service
- **TCP friendliness:** Since, most of application data still use TCP for transmission, an appropriate scheme

should be gentle enough to share available bandwidth with TCP. The notion of TCP friendliness (Mahdavi and Floyd, 1997) solves fair allocation between end-system performance aware transport protocols with TCP

- **Quick response to flow changes:** An ideal solution would react quickly to flows joining and departing. When other flows enter, the aggregate throughput should keep the same as the receiver's capacity. And when a flow leaves, the remaining flows should fast probe the available bandwidth and fairly share it

END-SYSTEM BASED CONGESTION DETECTION

The congestion detection schemes: According to congestion detection metrics, current end-system performance aware transport protocols mainly fall into three categories:

- **Process schedule monitoring:** As discussed by Datta *et al.* (2006a,b) and Banerjee *et al.* (2006) if large amount of CPU time allocated to processes other than the transmission process, sustaining coming packets will finally exceed receiver's processing ability. To avoid host congestion, the transmission process should be suspended when it obtains a relatively small portion of CPU time compared with other processes (Datta *et al.*, 2006a,b) or when runs at a significantly low dynamic priority (Banerjee *et al.*, 2006)
- **Packet loss ratio based:** Another area of active research is the use of a packet loss ratio based mechanism (Mark, 2002; Wu and Chien, 2004; Datta *et al.*, 2006a,b). In this approach, when packet loss ratio is increasing or relatively large compared to a dynamic threshold, it assumes receiver congestion happens or is aggravating, the sender slows down the sending rate in proportion to the packet loss ratio. The situation applies in reverse
- **Buffer and disk management considered:** The most recent emerging solution focuses on receiver's buffer and disk rate to achieve both high throughput and low burden on host. This is the approach taken by Performance Adaptive UDP (PA-UDP) (Eckart *et al.*, 2008) and RTsunami (Ren *et al.*, 2009). When high buffer occupancy and comparatively lower disk speed than CPU occurs, sender reduces transfer rate accordingly

Amongst three kinds of schemes described above, since OS schedule parameters are very difficult to be estimated accurately, meanwhile, only Tsunami and PA-UDP are both open source and high performance, so we compare these two protocols as representatives of the last two kinds of promising schemes.

Table 1: Summary comparison of Tsunami and PA-UDP

Characteristics	Tsunami	PA-UDP
Initial rate	Negotiated by sender and receiver	Negotiated by sender and receiver
Congestion detection	Comparison of error rate and adjustable threshold	Comparison of available buffer size and the remaining file size
Rate adaption	Rate adjustment with inter-packet delay compensation	Rate adaption with effective buffer and CPU management
Reliable transmission	Yes	Yes
Intra-protocol fairness	Yes	Yes
TCP friendliness	To some extent	No
Implementation	User level	User level

Each of these two protocols is different both in intended environment of use and performance characteristics. Among them, Tsunami targets faster file transfer over high speed links, the performance of which is limited by the I/O processing (and disk speed) of two end-nodes. It employs an inter-packet delay rate adjustment scheme to achieve high performance. PA-UDP is designed to efficiently maximize performance and minimize packet loss rate under different systems and exploits receiver's buffer occupancy and disk rate to manage receiver congestion.

Here, we give a brief overview of two end-system performance aware transport protocols: Tsunami and PA-UDP. A summary of the key characteristics of these two protocols can be found in Table 1.

Tsunami: Tsunami is an application-level protocol, intended for fast file transfer over high speed dedicated environments. Data blocks are transferred via UDP and control data are delivered through TCP. Tsunami communication starts by establishing a Tsunami session between Tsunami client (receiver) and server (sender). During session establishment, the client sends its target transfer rate, error threshold, inter-packet delay scaling factors and other parameters. After a Tsunami session has been established, the server is ready to transmit the file at the target transfer rate, which is specified by the client, via UDP. On every 50th iteration, the client calculates current error rate and sends it to the server, then the server examines the error rate. If it is over the threshold, the inter-packet delay (which is reciprocal of the transfer rate) is scaled upward. Otherwise, if the error rate is under the threshold, the inter-packet delay is scaled downward.

Tsunami performs rate control via adjustment of inter-packet delay rather than a sliding-window mechanism, thus dramatically improving the transfer rate in fast long-distance networks. However, Tsunami's congestion control views a relatively large error rate as a sign of receiver congestion, lacking the mechanism of distinguishing loss ratio caused by end system

congestion from other causes. Occasional packet loss does not mean low end system ability, thus it may lead to unnecessarily throttling throughput. We expect this problem to be solved in upcoming version of Tsunami.

PA-UDP: Based on a mathematical model and theoretical analysis, PA-UDP considers the effects of disk throughput and CPU latency to make sure receiver can support the required data rates. The authors propose a simple three way handshake to negotiate the initial target rate via TCP. Data could then be sent over UDP socket at the target rate. Receiver will send retransmission requests over TCP channel upon discovery of lost packets. The PA-UDP receiver periodically compares the size of file left to be transferred with remaining buffer size. If the left file size is smaller than the remaining buffer size, the sending rate will be set the maximum rate imposed by receiver. If remaining buffer size is smaller, the new sending rate will be a function of the remaining buffer size, the left file size and the speed CPU writes data to disk. New sending rate is updated via TCP channel.

The mathematical model, this protocol based on, assumes that the rate CPU moves data from buffer to disk is always slower than packets' coming rate. From the entire transmission process it is absolutely true. But due to the existence of buffer, processing rate can be faster than receiving rate. So, the mathematical model which is the heart of PA-UDP needs to be modified in their future work.

COMPARISON STUDIES

Here, our experimental studies for Tsunami protocol and PA-UDP protocol.

Experimental set-up: In order to emulate a fast long-distance optical circuit, we deploy the following experimental set-up. We connect two machines back-to-back with 1-Gigabit Ethernet adapters. The system configuration and the maximum bandwidth achieved by a TCP connection measured by Iperf 2.0.4 is shown in Table 2. The Iperf measurements are with a 1500-byte Maximum Transfer Unit (MTU). In particular, experiments are made under typical FLDnet configuration, in which RTT is 80 m sec.

Table 2: System configuration

	Configuration	
Characteristics	1	2
Processor	Intel (R) Xeon (R)	Intel (R) Xeon (R)
Processor speed	2.0 GHz	2.0 GHz
Cache size	4096 kb	4096 kb
RAM	3 Gb	4 Gb
OS	Linux 2.6.18	Linux 2.6.18
Iperf measurement	1.04 Mbps	16.7 Mbps

Throughput: Here, we compare these two end-system performance aware solutions with one UDP-based high performance data transfer protocol: RBUDP (He *et al*., 2002). Throughout our experiments, we use the latest versions of the protocols (RBUDP v1.0, Tsunami v1.1 and PA-UDP prototype) and use emulations with Netem delay router. We first present throughput of these end-system performance based protocols, as well as RBUDP, when transferring different sizes of files in Fig. 1. The file size is varying from 8 MB to 2.5 GB and the round trip delay is approximately 80 m sec. We also, measure throughput when transferring 512 Mb data under various round trip delays, from about 0 to 200 m sec (Fig. 2). For all the two scenarios described above, the bandwidth on each single point-to-point link is 1 Gbps.

Present results show that the throughputs of all three protocols do not change very much when delivering different size of files with a 80 m sec RTT. The PA-UDP

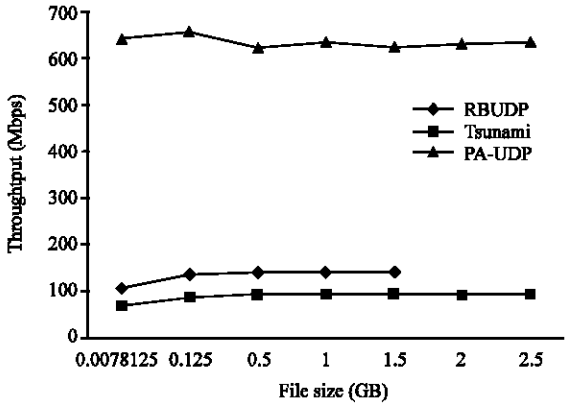

Fig. 1: Throughput when delivering different file sizes

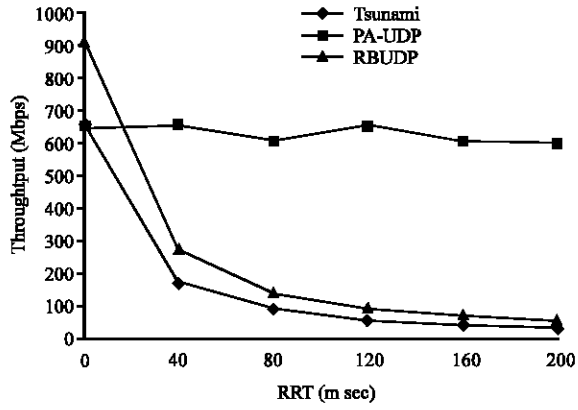

Fig. 2: Throughput with different round trip delays

is fastest with speeds higher than 600 Mbps, RBUDP is lower with rates between 100 and 150 Mbps and Tsunami transfers lowest with speed below 100 Mbps. Since, the sender of RBUDP has to keep all data it has sent until receiving feedback from the receiver, thus the delivering file size is limited by sender's buffer size. In our simulations, the available memory is about 1.62, so RBUDP can not work when delivering 2 and 2.5 GB files.

All three protocols show good performance (larger than 600 Mbps) when RTT is near 0 m sec. With increase of RTT, the throughputs of Tsunami and RBUDP reduce dramatically, only PA-UDP maintains high speeds approximately around 650 Mbps. This suggests PA-UDP effectively achieve high throughput while maintain low burden on host by monitoring host's buffer occupancy and disk rate. However, the situation on Tsunami and RBUDP is quite different. When round trip increases, Tsunami's slow rate adaption in large BDP environments lead to host's congestion, then receiver will discard packets and intensify sender's retransmitting, consequently causes low throughput. RBUDP maintains a fixed sending rate, regardless of receiver's capacity. When sender receives feedback from receiver, sender will retransmit lost packets. This part of cost will grow as round trip delay is increasing.

Intra-fairness: Although, in scientific application area, the probability of multi streams using the same high speed protocol in one single link is very rare, it is still necessary to compare the performance of parallel streams following the same protocol because this kind of competition among streams happens in point-to-point dedicated backbone networks which share the same source and sink.

The intra-protocol fairness is the fairness between two or more flows of the same protocol. We use the following definition for maximum-minimum fairness (Bertsekas and Gallager, 1992) which is widely used in networking fields: Given a set of K flows with rate allocation $R = \{r_1, r_2, \ldots, r_K\}$, where, r_{max} and r_{min} are the maximum and minimum rates of those flows, the Max-Min fairness index f_R of flow rate allocation R as:

$$f_R = \frac{r_{min}}{r_{max}}$$

In general, all these protocols achieve good fairness for a single link with 2 and 4 parallel flows with 80 m sec RTT except TCP (Fig. 3) throughputs among 4 TCP flows vary in a large scale (10x), the fairness index of 4 TCP flows is smaller than 0.2. This is because TCP's congestion control mechanism primarily focuses on

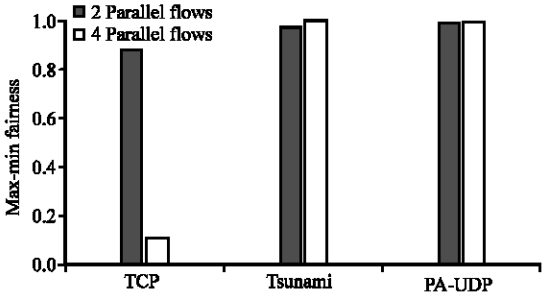

Fig. 3: Fairness index of 2 and 4 parallel flows on a single link

Table 3: RBUDP, Tsuanmi, PA-UDP each runs with a single TCP flow, point-to-point on a 1 Gbps link on the local cluster

Protocols	Throughput (Mbps)	TCP (Mbps)	Single TCP throughput (Mbps)	A
RBUDP	214.88	238	363	-0.051
Tsuanmi	477.62	309	363	0.214
PA-UDP	622.30	46.4	363	0.861

Table 4: RBUDP, Tsuanmi, PA-UDP each runs with a single TCP flow, point-to-point on a 1 Gbps link with 80 m sec RTT

Protocols	Throughput (Mbps)	TCP (Mbps)	Single TCP throughput (Mbps)	A
RBUDP	120.75	121	178	-0.001
Tsuanmi	72.56	116	178	-0.230
PA-UDP	612.33	9.47	178	0.970

congestion occurred in the internal networks rather than at the end systems, which makes TCP inefficient in such fast large RTT environment.

TCP friendliness: Since, currently most applications still employ TCP for transmission, a good transport protocol should be friendly enough to TCP. We study how the operation of these end-system performance aware transport protocols affects traditional TCP flows. We measure TCP throughput in the presence of each of these two end-system performance aware transport protocols and compare with TCP running alone. We desired that the protocol would neither be too aggressive nor too gentle towards TCP. In order to quantify how aggressive or gentle one protocol is, we introduce the following formula, which is the asymmetry between two throughputs:

$$A = \frac{\overline{x_1} - \overline{x_2}}{\overline{x_1} + \overline{x_2}}$$

where, x_1 and x_2 are the throughput averages of flow 1 and 2 in the cross-traffic (Bullot *et al.*, 2003).

Table 3 and 4 show the result of the cross-traffic between each of these two end-system performance aware transport protocols and TCP Reno. A value near one

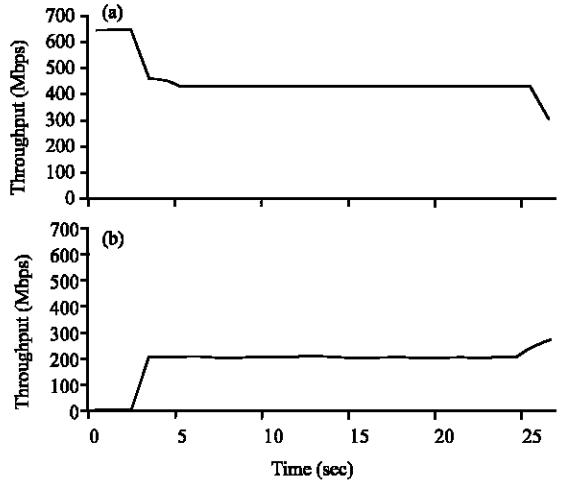

Fig. 4: Two parallel PA-UDP flows share a single link. Flow 2 begins about 3 sec later. (a) PA-UDP flow 1 and (b) PA-UDP flow 2

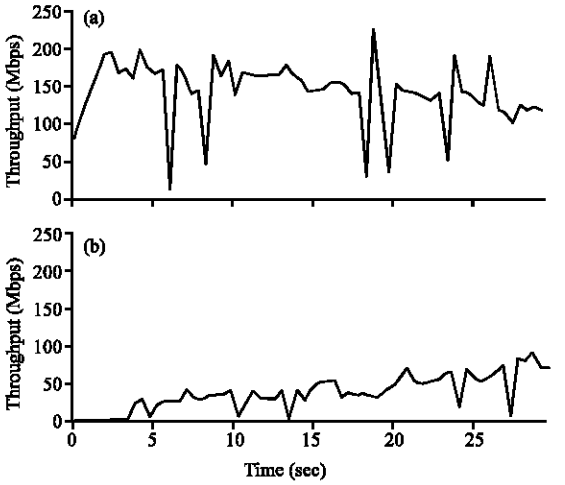

Fig. 5: Two parallel Tsunami flows share a single link. Flow 2 begins about 3 sec later. (a) Tsunami flow 1 and (b) Tsunami flow 2

indicates that the protocol is too aggressive towards the competing protocol. A value near minus one indicates a too gentle protocol. The optimal is to have a value near zero which indicates that the protocol is fair against TCP. Present results suggest that RBUDP, Tsunami flows show good sharing properties with TCP both in local cluster environment and on high bandwidth delay networks (80 m sec). The situation with parallel TCP and PA-UDP flows is quite different. In the presence of PA-UDP, TCP is not able to achieve the same level of throughput. This is because PA-UDP is too aggressive. PA-UDP employs a rate adaption scheme which considers host's buffer utilization and disk rate to obtain host's entire available capacity.

Transition management: The ability to respond quickly and stably to rapid flow changes is an important ability for transport protocols in high speed networks whose goal is to provide the maximum physical bandwidth to flows. When another flow enters, an ideal transport protocol would make the existing flow share the bandwidth equal with new flows, meanwhile, when flow leaves, a good transport protocol should rapid probe all available bandwidth. We use a two-stage scenario to evaluate Tsunami and PA-UDP. We begin with a single flow (flow 1) and a second flow (flow 2) begins around 3 seconds later. The trajectories for each flow's throughput are shown in Fig. 4 and 5.

Of these two protocols, PA-UDP achieves smooth transitions, but Tsunami gives erratic behavior. Tsunami's sensitivity to packet loss and delay produces rate oscillations in our Netem environment (Fig. 5a, b). Because Netem is an emulation tool, it is possible that its behavior is not true to real networks. PA-UDP performance is much better, producing clean transition and quick rate adaption to flow changes. When flow 2 enters, two flows share bandwidth steadily. Around 25 sec after beginning, the rate of flow 1 decreased because file transferred on flow 1 is pending finished (Fig. 4a, b). The PA-UDP has a fundamental advantage in its receiver's buffer and disk management based rate adjustment scheme, providing a global perspective across flows and thereby enabling dramatically better network performance.

CONCLUSIONS AND FUTURE WORK

Fast long-distance networks involve a new set of communication challenges where networks have plentiful bandwidth but limited end-system capacity. This change moves congestion from the internals of the network to the end nodes. We study the performance of two promising end-system performance aware transport protocols (PA-UDP and Tsunami) in a wide range of different circumstances. Present results showed that Tsunami and PA-UDP can achieve high performance for point-to-point connection with single or parallel flows in local cluster. However, when large round trip delays are considered, the situation is quite different. With its host's unique buffer occupancy and disk rate consideration, PA-UDP outperforms Tsunami providing lower loss, higher throughput and rapid, stable transitions as flows begin. These results suggest that receiver-driven buffer and CPU management schemes should be more broadly studied for FLDnet transport protocols.

Future work Major challenges remain for end-system performance aware transport protocols in FLDnet

environments. First, we need to explore more techniques to design experiments by using various NICs, different operating systems and host hardware. Results from both production networks and 10Gbps testbeds are also very helpful. Second, end-system performance aware transport protocols aim at effective transmission in high speed long distance networks, its efficiency in traditional low speed packet-switched networks needs to be systematically verified.

Finally, it is a critical and challenging topic to model these end-system performance aware transport protocols analytically, including a formal proof of their properties (e.g., convergence, fairness, TCP friendliness, etc.).

ACKNOWLEDGMENTS

We express our sincere gratitude to Dr. Jingguo Ge and Yuepeng E at Computer Information Network Center, Chinese Academy of Sciences for their helpful suggestions and to Ben Eckart at Tennessee Technological University for providing PA-UDP code for our simulation studies. This research is supported by National High Technology Research and Development Program of China (National 863 program) under Grant No. 2007AA01Z214.

REFERENCES

Allman, M., V. Paxson and W. Stevens, 1999. TCP congestion control. RFC 2581. http://portal.acm.org/citation.cfm?id=RFC2581.

Banerjee, A., W. Feng, B. Mukherjee and D. Ghosal, 2006. An end-system aware protocol for intelligent data transfer over lambda grids. Proceedings of 20th International Parallel and Distributed Processing Symposium, April 25-29, Rhodes Island, Greece, pp: 1-10.

Bertsekas, D. and R. Gallager, 1992. Data Networks. 2nd Edn., Prentice Hall, Englewood Cliffs, NJ.

Brakmo, L. and L. Peterson, 1995. TCP vegas: End to end congestion avoidance on a global internet. IEEE J. Select. Areas Commun., 13: 1465-1480.

Bullot, H., R. Les-Cottrell and R. Hughes-Jones, 2003. Evaluation of advanced TCP stacks on fast long-distance production networks. J. Grid Comput., 1: 345-359.

Datta, P., S. Sharma and W. Feng, 2006a. A feedback mechanism for network scheduling in lambdaGrids. Proceedings of the 6th International Symposium on Cluster Computing and the Grid, May 16-19, IEEE Computer Societ, Singapore, pp: 584-591.

Datta, P., W. Feng and S. Sharma, 2006b. End-system aware, rate-adaptive protocol for network transport in lambdagrid environments. Proceedings of the ACM/IEEE conference on Supercomputing, Nov. 11-17, Tampa, FL, USA., pp: 746-746.

Eckart, B., H. Xubin and W. Qishi, 2008. Performance adaptive UDP for high-speed bulk data transfer over dedicated links. Proceedings of IEEE International Symposium on Parallel and Distributed Processing, April 14-18, Miami, FL, pp: 1-10.

He, E., J. Leigh, O. Yu and T.A. de Fanti, 2002. Reliable blast UDP: Predictable high performance bulk data transfer. Proceedings of the IEEE International Conference on Cluster Computing, Sept. 23-26, Washington, DC., USA., pp: 317-317.

Floyd, S., 2003. HighSpeed TCP for large congestion windows. RFC3649, pp: 1-20. http://acs.lbl.gov/~evandro/hstcp/notes/hstcp-dsd.pdf.

Jacobson, V., R. Braden and D. Borman, 1992. TCP extensions for high performance. http://www.ietf.org/rfc/rfc1323.txt.

Jin, C., D.X. Wei and S.H. Low, 2004. FAST TCP: Motivation, architecture, algorithms, performance. IEEE/ACM Trans. Network., 14: 1246-1259.

Kelly, T., 2003. Scalable TCP: Improving performance in highspeed wide area networks. Comput. Commun. Rev., 33: 83-91.

Lehman, T., J. Sobieski and B. Jabbari, 2006. DRAGON: A framework for service provisioning in heterogeneous grid networks. IEEE Commun. Maga., 44: 84-90.

Leith, D. and R. Shorten, 2004. H-TCP: TCP for high-speed and long-distance networks. Proceedings of the 2nd International Workshop on Protocols for Fast Long-Distance Networks, February 2004, Argonne, Illinois USA., pp: 1-16.

Mahdavi, J. and S. Floyd, 1997. TCP-friendly unicast rate-based flow control. Technical Note Sent to the End-to-end Interest Mailing List. http://www.psc.edu/networking/papers/tcp_friendly.html.

Mark, R.M., 2002. Tsunami: A high-speed rate-controlled protocol for file transfer. http://steinbeck.ucs.indiana.edu/~mmeiss/papers/tsunami.pdf.

Martin-Flatin, P.J. and S. Ravot, 2002. TCP congestion control in fast long distance networks. California Institute of Technology, USA., Technical Report CALT-68-2398. http://netlab.caltech.edu/FAST/publications/caltech-tr-68-2398.pdf.

Mathis, M., J. Mahdavi, S. Floyd and A. Romanow, 1996. TCP selective acknowlegement options. RFC2018, Internet Engineering Task Force(IETF), http://www.ietf.org/rfc/rfc2018.txt.

Poster, J.B., 1981. Transmission control protocol. RFC 793. http://www.faqs.org/rfcs/rfc793.html.

Rao, N.S.V., W.R. Wing, S.M. Carter and Q. Wu, 2005. Ultrascience net: Network testbed for large-scale science applications. IEEE Commun. Maga., 43: S12-S17.

Ren, Y.M., H.N. Tang, J. Li and H.L. Qian, 2009. A novel congestion control algorithm for high performance bulk data transfer. Proceedings of the 8th IEEE International Symposium on Network Computing and Applications, July 9-11, IEEE Computer Society, Cambridge, pp: 211-218.

Smarr, L., A. Chien, T. De Fanti, J. Leigh and P. Papadopoulos, 2003. The Optiputer. Commun. Assoc. Comput. Mach., 46: 58-67.

Wu, X.R. and A.A. Chien, 2004. GTP: Group transport protocol for lambda grids. Proceedings of the 4th International Symposium on Cluster Computing and the Grid, April 19-22, Chicago, IL, USA., pp: 228-238.

Xu, L., K. Harfoush and I. Rhee, 2004. Binary increase congestion control for fast long-distance networks. 23rd Annu. Joint Conf. IEEE Comput. Commun. Soc., 4: 2514-2524.

Yongmao, R., T. Haina, L. Jun and Q. Hualin, 2009. Performance comparison of UDP-based protocols over fast long distance network. Inform. Technol. J., 8: 600-604.