IMPACT OF BURSTIFICATION TIME OUT ON THE PERFORMANCE OF DELAY-BASED TCP VARIANTS OVER OBS NETWORKS

Ratna Pavani. K, N. Sreenath
Department of Computer Science,
Indira Gandhi College of Arts and Science, Puducherry, India

ABSTRACT

Due to increasing demand for higher bandwidth application on the Internet, Optical networks with WDM technology has become a de-facto standard. Optical Burst Switching (OBS) is an established switching paradigm for the initiation of core optical networks. OBS layer will collect data from TCP layers based on the burst assembly mechanism (time based or quantity based). In either situation, there can be contention when two or more bursts arrive concurrently in two varied wavelengths and requests for the same outgoing wavelength. Consequently, one of the contending burst is dropped. These random burst losses (RBL) in an OBS network can cause TCP to timeout. Apart from this, TCP sources can also be classified based on their access bandwidth into fast, medium, and slow classes. Considering these circumstances, if a fast class TCP source is used over OBS network and a burst is lost due to RBL, TCP sender will consider the underlying network to be congested and react to it by initiating congestion control mechanism. Each variant of TCP differs from the other in the way they handle congestion control mechanism. In this backdrop, an experimental study was made to compare the performance of popular delay-based variants (TCP-Vegas, TCP-NewVegas, and FAST TCP) over OBS networks with varying metrics and traffic scenarios using Network Simulator version-2 (NS-2).

KEYWORDS: *Transmission Control Protocol (TCP), Optical Burst Switching (OBS) Network, Random burst losses (RBL,) TCP-Vegas, TCP-NewVegas, FAST TCP, Network Simulator version-2 (NS-2)*

I. INTRODUCTION

Increasing advancement of the Internet's traffic and by the evolution of the optical technology, TCP/IP over wavelength division multiplexing (WDM) networking seems to be an imminent architecture to protract the expected huge bandwidth demand of next generation networks. Optical technology used in WDM networks is capable of supporting a bandwidth demand up to 50Tbs [1]. Low attenuation of signal, exceptionally lower error bit rates, and minimum signal distortion as light rarely radiates away from fiber, are the key features of WDM networks. Optical Circuit Switching (OCS), Optical Packet Switching (OPS) and Optical Burst Switching (OBS) are three eminent switching paradigms in these networks. If there is an end-to-end lightpath between the source and the destination nodes for the entire session to evade optical-to-electronic (OEO) conversations at the intermediary node in a network, then that sort of switching technology is called OCS. Setting up a dedicated lightpath results in ineffective usage of bandwidth during high Internet traffic. In OPS networks this wastage of bandwidth can be avoided. In this network, IP packets coming from diverse sources are directly switched in the optical domain [2]. In OPS networks with fixed length packets, packet synchronization turns out to be crucial to minimize contention which is difficult to execute [3]. Pragmatic implementation of OPS stresses on fast switching times that are quite expensive [4]. To prevail with the problem of optical-buffering and optical-processing and still attain switching in optical domain, OBS networks have been proposed.

OBS is ascribed as a balance between the OCS and OPS networks [1]. OBS networks comprises of three main components, an ingress node, an egress node, and a network of core nodes. Ingress nodes and egress nodes can be collectively termed as edge nodes. In OBS, a burst is the central switching component. The edge nodes have to congregate IP packets and assemble them into bursts called burstification. Packets that are premeditated to the same egress node and that need same level of service are put into burst assembly queue. To avoid optical buffering and processing of the databurst at core nodes, a control packet also called burst header packet that holds the information about the length and arrival time of the databurst is sent ahead with an offset time. This offset time or time gap between the control packet and the databurst is sufficient to process the burst header packet and configure the switches at the core nodes. The switches along the route are configured only when the databurst arrives. This enables the databurst to cut through an all-optical path. At the egress node the databurst is disassembled back into IP packets.

Due to aggregation of packets into bursts at the ingress node, the characteristics of the IP traffic are changed inside the OBS network. There are two main components that change the characteristics of the input traffic: the burst assembly algorithm used and the burst assembly queue. While considering the modelling of the traffic in an OBS network, the burst formation time and the waiting time of the burst in the transmission queue have to be considered as they may increase end-to-end delay of the packets.

In these networks there are two types of protocols mainly used for wavelength reservation along the path in the core nodes, Tell-and-wait (TAW) and Tell-and-Go (TAG). In the TAW protocol, the control packet travels along the entire path from ingress node to egress node making reservation for the databurst. After successful reservation of the resource, an acknowledgement is sent back by the destination. If at any intermediate node the reservation fails along the path, a negative acknowledgement is sent back to release the obtained wavelength. In TAG protocol, databurst does not wait for an acknowledgement from the destination. The control packet first travels and reserves required bandwidth on the core network. Databurst follows it with an offset time on a different wavelength. In case the control packet fails to make the reservation along the path, the control packet along with the corresponding databurst is simply dropped or an appropriate contention resolution policy is considered when it reaches that link. Just-enough-time (JET) and Just-in-time (JIT) are two most imperative signalling mechanisms in TAG protocol.

In JIT, the wavelength that is taken by the control packet for the databurst is torn down using an explicit control message. The control packet and the databurst travel on different wavelengths. The control packet needs to inform the core node only about the wavelength on which the databurst is intended to arrive. In JET, the bandwidth is reserved only for the duration of the databurst; no explicit message is required to release the acquired resource. This enhances the utilization of the wavelength but increases the processing time of the control packet.

In JET protocol mechanism; there is a possibility of the databurst being contended by another burst that is intended to arrive later because the offset time varies according to the path length. This phenomenon is called retro-blocking owing to which the bursts with a greater offset time are successful in reserving the wavelength before the bursts with smaller offset time. Hence with OBS networks when JET signalling mechanism is used having an appropriate offset time is important to avoid contention which may lead to a burst loss.

Contention among two bursts arises due to their overlap in time when they arrive concurrently on two varied links or wavelengths and require the same out-going wavelength. Subsequently, when multiple bursts contend for the same wavelength at the core node, all but one of them is dropped. Burst loss owing to contention is a major source of trepidation in OBS networks. Such contention losses, which are temporary in nature, can degrade the performance of the applications using OBS networks. Apart from this, another issue of concern is, OBS layer will collect data from TCP layers based on burst assembly mechanism (time based burst assembly or quantity based burst assembly). In either case RBL will degrade the performance of TCP layers due to the presence of multiple segments in the same burst.

So in this scenario it becomes essential to study the behaviour of TCP over OBS networks. Based on the type of congestion control mechanism implemented, TCP variants are broadly classified into loss-based (TCP-Reno, TCP-Newreno and etc) and Delay-based (TCP-Vegas, TCP-NewVegas, FAST-TCP and etc). In this work, a performance study of popular delay-based flavours of TCP, TCP-

Vegas, TCP-NewVegas, and FAST-TCP are considered for evaluation to identify an apt variant and the metrics under which their performance will be optimal over OBS networks. This paper is organised as follows: In section-II, an analysis of the congestion control mechanism adapted by TCP-Vegas, TCP-NewVegas, and FAST-TCP is discussed (in correlation over OBS networks), Section-III deals with the motivation and problem definition. Section-IV deals with systems study, simulation environment, results of simulations along with discussion of the results. Conclusion and future work forms section-V and VI respectively.

II. TCP VARIANTS OVER OBS NETWORKS

TCP-Vegas is a delay based TCP variant that uses the measured value of round trip time (RTT) to estimate the available bandwidth in the network. TCP-Vegas use the difference between the estimated throughput and actual throughput to get an estimate of the congestion in a network [5]. TCP-Vegas first computes the Base-RTT, which is the minimum measured RTT (which is obtained by the summation of propagation delay and the queuing delay). The expected throughput is derived as

$$\text{expected} = \frac{\text{congestion window or CW}}{\text{Base}-\text{RTT}}$$

The actual throughput is computed for every round as

$$\text{actual} = \frac{\text{congestion window or CW}}{\text{RTT}}$$

where RTT is the most recent measured RTT. Then TCP-Vegas sender then computes

$$\text{diff} = (\text{expected - actual}) * \text{Base-RTT}$$

where diff is non-negative and is used to adjust the next Congestion Window (CW) size. Additionally TCP-Vegas defines two threshold values, α and β, for controlling the size of CW [6,7,8].

$$\text{CW} = \begin{cases} \text{CW} + 1 & \text{iff diff} < \alpha \\ \text{CW} & \text{iff } \alpha \le \text{diff} \le \beta \\ \text{CW} - 1 & \text{iff diff} > \beta \end{cases}$$

If α is larger than the diff, the CW is linearly increased in the next round and if β is less than the diff, the CW is linearly reduced. TCP-Vegas always tries to uphold the surfeit of packets between α and β. TCP-Vegas approximates that the network is congested if the actual throughput is less than expected throughput, hence reduces the transmission rate. The CW in TCP-Vegas is initialized to two segments and is exponentially augmented for every RTT in a slow start state. Slow start threshold is estimated by the value of γ.

The value of the CW remains unaffected between two consecutive rounds to make a valid comparison between expected throughput and actual throughput. Whenever a duplicate acknowledgement is received, it verifies to see the difference in current time and the timeout of the packet if is larger than the RTT; then it straight away retransmits the segment without waiting for triple duplicate acknowledgements. At the same time after a non-duplicate acknowledgement is received, TCP-Vegas checks for its timeout value from the time the packet was sent and retransmits if the segment time surpasses the timeout value without waiting for the duplicate acknowledgement. In this way TCP-Vegas adjusts its CW during multiple packet losses. This adjustment of CW is done by incrementing the size of the CW by 1 for every successful acknowledgement till the size of CW reaches a predefined threshold value. In an OBS network such linear growth of CW for every successful acknowledgement may lead to underutilization of the existing bandwidth.

TCP-NewVegas is an enhancement of TCP-Vegas and implements three sender-side changes, which aim to improve the performance of its predecessor. They are packet-pacing, packet-pairing, and rapid-window-convergence [9]. Primarily in packet-pacing, the transmission of packets is extended over the entire RTT. This prevents packets from being injected into the network, which may result in transient queues and distorted RTT measurements. In case of packet-pairing, TCP-sender increments the size of CW by sending two packets at a time for every second acknowledgement received. So during congestion, when retransmitting a packet, every retransmitted packet will be paired with a new packet till slow-start-threshold or a triple duplicate acknowledgement is received. Thereby, the TCP-NewVegas sender will increase the size of CW more swiftly than TCP-Vegas sender. Rapid-Window-Convergence is the third sender side modification done by TCP-NewVegas. It is executed in order to increase the CW more rapidly during the first part of the congestion-avoidance phase after the slow-start phase. In this process, CW grows and congregates to the optimal size far more rapidly than it would in case of other TCP variants like TCP-Vegas, at the same time continues to work in such a way that the underlying network is not over congested [10].

FAST-TCP (Fast AQM Scalable TCP where AQM represents Active Queue Management for TCP), is a recursive acronym for FAST-TCP. This protocol is generally considered as a high-speed adaptation of TCP-Vegas that aims to sustain a constant number of packets in queues throughout the network. Similar to TCP-Vegas, FAST-TCP utilizes both average and the minimum measured RTT to estimate the number of packets in the network queues. The number of packets in the network queues is taken by the TCP-sender to opt whether it should increase or decrease the sending rate, and the way of sending rate adjustment discriminates FAST-TCP from TCP-Vegas.

FAST-TCP makes use of the minimum measured RTT (baseRTT), the estimated average queuing-delay (RTT), as well as the packet loss, to accurately estimate the available network bandwidth and the network congestion. It cleverly reacts to both packet loss and packet delay in network congestion identification. It has loss and delay components that amalgamate to modify the number of packets on-the-fly according to the network condition. Therefore, FAST-TCP can be used as a reference model for dropping-based and delay-based TCPs. From the application perspective, FAST-TCP is considered as high-speed variant that operates for a relatively long period, which is expected to take an important role in some mission-critical applications such as grid and cloud computing [11,12].

To begin with, queuing-delay can be more precisely approximated than loss probability because packet loss presents one-bit of information to the sender, while queuing-delay provides multi-bit information to the sender with respect to the network status. This makes the implementation of congestion control easier to stabilize the current state of congestion in the network .In the second place; the working of queuing-delay seems to have an accurate measure with respect to network capacity. This helps preserve constancy as a network scales up in capacity [13, 14]. The working of FAST-TCP is characterised by approximating how far the current state of CW is from the equilibrium value of 1. Then congestion control scheme can impel the system rapidly, yet in an appropriate and fair behaviour, towards the equilibrium. Here the window convergence will be less when the current state is closer to equilibrium and more otherwise, irrespective of the equilibrium value. This feature of FAST-TCP enhances its performance over OBS networks when evaluated in comparison with other delay-based variants like TCP-Vegas and TCP-NewVegas.

III. MOTIVATION

In an OBS network, during burstification, there may be a delay caused due to the burst assembly mechanism used, and this may have an impact on end-to-end delay of the packets. Apart from this, RBL also can influence the performance of TCP over OBS network. Variants of TCP differ from one other in the way they handle the congestion control mechanism.

In [11] it is mentioned that during FAST-TCP flows where all packets in the window of TCP flow are assembled into a single burst, it suffers from the false congestion detection when the burst collides with any other burst and is dropped due to contention. This may trigger a timeout retransmission and FAST-TCP enters a slow start phase. During such situation, it is to be noted that FAST-TCP will also increase its CW to reach equilibrium state very rapidly. Also it may be possible to increase the performance by avoiding false timeout through providing more delay between the header burst and the corresponding databurst, so that the header burst can wait at the intermediate node for an appropriate wavelength. This reservation mechanism will affect conversely when there is high traffic. Hence it is expected that at low and medium traffic levels FAST-TCP should provide better results.

Therefore, it becomes essential that the TCP reacts to understand the actual situation of the underlying network and reinstate the size of the CW to suit the existing bandwidth. Thus, the issue of focus in this work is to analyse the impact on CW and identify an appropriate variant among the three popular delay-based flavours that gives better performance in an OBS networks.

IV. Simulations And Discussions

4.1. System Study

Simulations are being done using Network simulator version 2.27 (NS-2.27) to evaluate the three delay-based TCP variants, TCP-Vegas, TCP-NewVegas and FAST-TCP. NS-2.27 with OBS patch was used for simulation with random uniform burst distribution algorithm. Topology used is NSFNet. There are 14 optical core nodes in our topology with 28 TCP/IP nodes and 10 TCP connections. The IP packets are aggregated into burst at the edge nodes and transmitted all optically from source to destination. Packet processing in the core network is done by the optical classifier. The packets that are assembled in a single burst are defined in burst size. The size of the burst is varied to evaluate the performance of the three TCP variants. Burst size varies from a minimum of 10 packets per burst to a maximum of 11000 packets per burst. The hop delay between the control burst and databurst is 2ms. There are 100 assembly buffers at the ingress node. Burstification period is varied from 0.001 and 1.0 to estimate its affect on throughput of TCP variants.

To eliminate the problems in all optical processing of packet headers, the data plane and the control plane are separated in OBS simulator. MAX-PACKET-NUM is a variable used in the simulation to count the number of packets in a burst. JET signalling mechanism is used in this network, where the control packet tries to reserve resources for the burst just sufficient enough for transmission on each link it traverses. The control packet has all the vital information so that each intermediary optical switch in the core OBS network can transfer the databurst and also configures its switching matrix in order to switch the burst all-optically. The conversion of electrical-optical-electrical is taken care by the edge node in the core OBS network. They generate and forward the control packets followed by the databurst. There is no burst segmentation, fiber delay lines or deflection routing in the core network. The node entrance has classifier that separates TCP segments form the optical bursts. Latest Available Unused Channel with Void Filling (LAUC-VF) [15] and Minimum Starting Void (Min-SV) [16] are the scheduling algorithms that are presently implemented in OBS.

4.2. Simulation

The topology used for the simulation is NSFNet with 14 core nodes with finite buffer. This Network consists of 14 optical core nodes and 28 electrical nodes. This core network in our simulation is modelled as a single network with 1Gbps bandwidth and 10ms propagation delay. The access links have bandwidth of 155Mbps with link propagation delay of 1ms. Maximum number of packets in a burst varies between from 10 to 11000.

In order to identify an apt variant over OBS networks, the three delay-based variants were simulated and the figure-2, figure-3, and figure-3 epitomize the same. All the three variants belong to the same family of TCP-Vegas. TCP-Vegas implements Additive Increase and Multiplicative decrease (AIMD) mechanism during congestion control, thereby incrementing the size of CW linearly by 1 for every RTT.

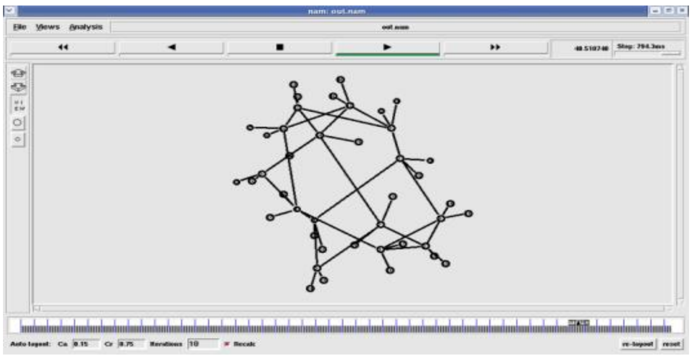

Figure.1. NSF Topology with 14 optical core nodes and 28 electric nodes

Table.1. Simulation parameters

Topology	: NSFNet
Number of optical core nodes	: 14
Number of electronic nodes	: 28
Number of TCP/IP connection	: 10
Packets per burst	: 10 to 11700
Max lambda	: 20
Link Speed	: 1GB
Hop delay	: 0.0001 to 1ms
Burstification period	: 0.001ms to 1.0ms

TCP-NewVegas improves over TCP-Vegas by making several changes to the sender side implementation of congestion control. TCP-NewVegas shows a superior performance over TCP-Vegas in OBS networks with its additional feature like packet-pairing and rapid-window-convergence during fast retransmit and fast recovery phases of congestion control mechanism. TCP-Vegas and TCP-NewVegas consider delay as a mode to identify congestion. FAST-TCP extends this feature and considers queuing-delay as a means to identify congestion in the network. This protocol constantly tries to maintain the size of CW close to the equilibrium value which is obtained from the most recent acknowledged transmission (latest RTT value). The figure: 2, 3 and 4 shows the simulation results of all the three delay-based variants with varying time-out values. We have considered CBR and FTP type of traffic. This core network in our simulation is modelled as a single network with 1Gbps bandwidth and 10µs propagation delay. The access links have bandwidth of 155Mbps with link propagation delay of 1µs.

Our results show that FAST-TCP performs uniformly better when compared with TCP-Vegas and TCP-NewVegas when burstification-time-out (BTO) value is 1µs. With BTO value equal to 0.01µs, it is observed that TCP NewVegas and FAST-TCP performance is almost equal. From the above simulations, it is concluded that FAST-TCP performs better than its counterparts and is more suitable for OBS networks under medium traffic conditions.

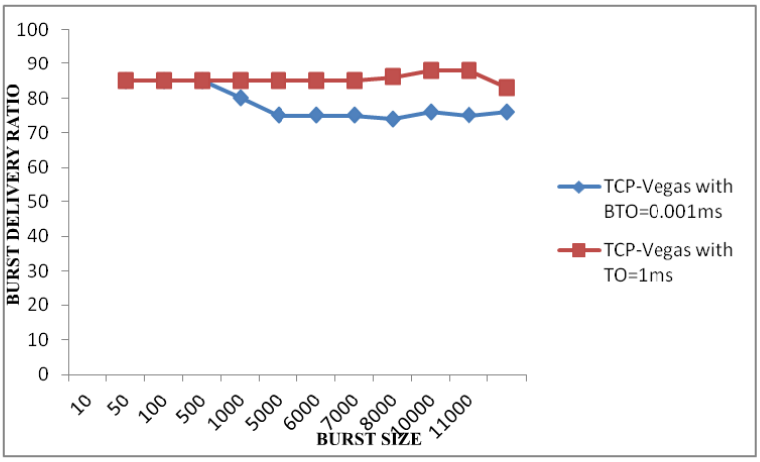

Figure: 2 Throughput of TCP-Vegas with BTO 0.01 µs and 1.0 µs

TCP-Vegas always try to adjust the size of CW to a value between α and β. When there is a multiple packet loss, TCP-Vegas based on the time-out value retransmits the lost packet without waiting for a triple duplicate acknowledgement. While considering fast flows of TCP-Vegas, it may be noted that there is an amount of delay encountered during burstification. This inference is made from the results obtained while simulating TCP-Vegas with BTO values 0.001µs, 0.01µs, 0.1µs, and 1µs as shown in figure:2. The throughput of TCP-Vegas improved when BTO was 1µs. When databurst size increased the performance of TCP-Vegas, it showed a constrained throughput with time-out value less than 1µs. After 10,000 packets there was a decrement in the performance of the protocol. Simulations were even run by increasing the time-out values, but after 1µs performance showed insignificant growth.

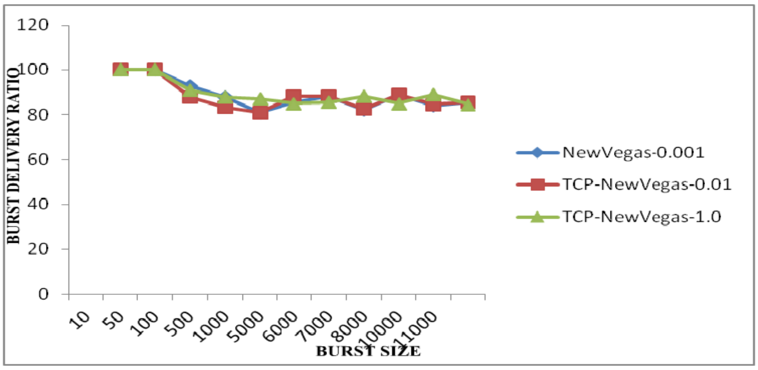

Figure: 3 Throughput of TCP-NewVegas with BTO 0.001 µs, 0.01 µs and 1.0 µs

Simulations in figure: 3 show the results of TCP-NewVegas. The overall throughput of TCP-NewVegas is between 80% and 90%. There is not much deterioration in its performance with varying BTO values. After a burst size of 500 packets and with BTO values equal to 0.001 there is a decline in the performance of TCP-NewVegas. It can be concluded that the impact of BTO over TCP-NewVegas is very limited and the protocol performance has enhanced over its predecessor TCP-Vegas over OBS networks. In case of FAST-TCP during simulation it is observed that its performance is high with higher BTO values with maximum throughput and minimum delay. Overall performance with the given parameters it is clearly understood that FAST-TCP shows nearly 91 percent throughput consistently, though there is some amount of delay in end-to-end delivery after 6000 packets with BTO value less than 1ms.

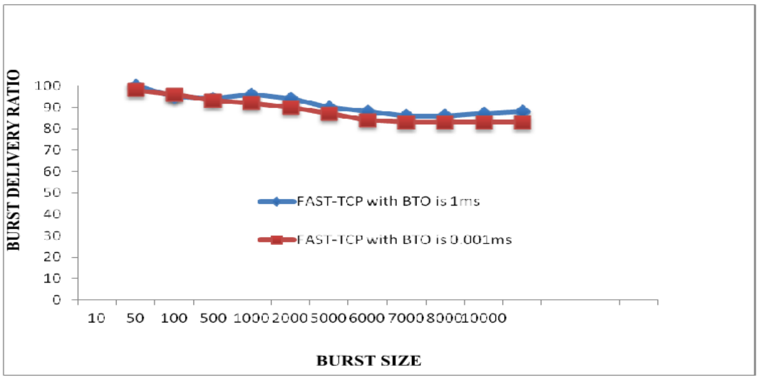

Figure: 4 Throughput of FAST-TCP with BTO 0.001 µs and 1.0 µs

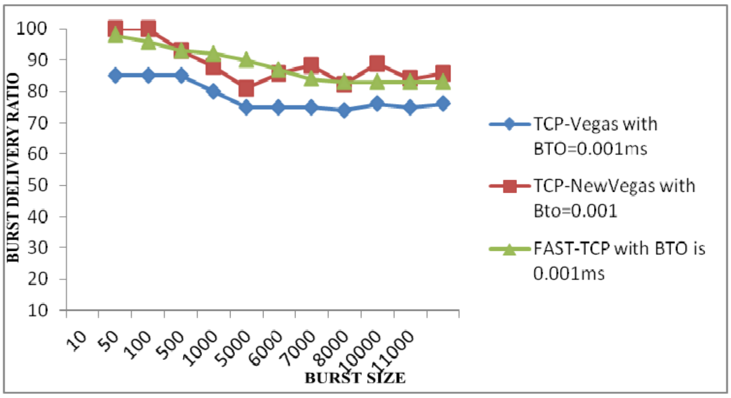

Figure: 5 Comparison of TCP-Vegas, TCP-NewVegas and FAST-TCP when BTO is 0.001 µs

TCP-NewVegas improves over TCP-Vegas by making several changes to the sender side implementation of congestion control. With all these amendments, TCP-NewVegas shows a superior performance over TCP-Vegas in OBS networks. TCP-Vegas and TCP-NewVegas consider delay as a mode to identify congestion. FAST-TCP extends this feature and considers queuing-delay as a means to identify congestion in the network. This protocol constantly tries to maintain the size of CW close to the equilibrium value which is obtained from the most recent acknowledged transmission (latest RTT value). Figure: 5 show the simulation results of all the three delay-based variants with varying time-out values. We have considered CBR and FTP type of traffic. This core network in our simulation is modelled as a single network with 1Gbps bandwidth and 10μs propagation delay. The access links have bandwidth of 155Mbps with link propagation delay of 1μs. Our results show that FAST-TCP performs uniformly better when compared with TCP-Vegas and TCP-NewVegas when BTO value is 1μs. With BTO value equal to 0.01μs, it is observed that TCP NewVegas and FAST-TCP performance is almost equal. From the above simulations, it is concluded that FAST-TCP performs better than its counterparts and is more suitable for OBS networks under medium traffic conditions.

V. CONCLUSION

Amid increasing online applications, most of the Internet traffic contains multimedia messaging and real-time traffic. Therefore, loss of bursts which causes simultaneous loss of multiple packets has a severe impact on TCP. One of the major issues of research in OBS is to identify the parameters for efficient transport layer protocol at higher layers despite the presence of frequent contention losses. To achieve this we have made an in-depth study of popular delay-based TCP variants considering various parameters and analyzed them using NS-2. In case of TCP-Vegas and TCP-NewVegas there is a decline in throughput when burstification time varied from 1.0ms to 0.001ms when compared to FAST-TCP. Therefore in case of Protocols that considered end-to-end delay as the measure of congestion, FAST-TCP proved to more consistent in its burst delivery ratio with more than 90% throughput when compared to TCP-Vegas and TCP-NewVegas when the method of burstification is either quantity-threshold mechanism or time-threshold mechanism.

VI. FUTURE WORK

In this research, we worked on delay-based variants of TCP and the way they react when used over OBS networks. We have also identified suitable parameters for the protocols at which we have maximum throughput. All the variants of TCP are simulated using only NSFNet topology with 14 optical nodes. To enhance the work, different topology with more number of optical and electrical connections can be used with increased simulation time to clearly obtain the throughput in case of random contention leading to false timeout. As an extension to this work we can make a detailed study to identify the cause for negative delay and decreased throughput of delay-based variants like TCP-Vegas, TCP-NewVegas and FAST-TCP. Channel scheduling and priority queuing [17] can be used in the OBS core network with burst retransmission to analyse the performance of TCP variants over OBS networks.

REFERENCES

[1]. T. Venkatesh, and C. Siva Ram Murthy, “An analytical approach to optical Burst Switched Networks,” e-ISBN 978-4419-509-2, Springer New York Dordrecht Heidelberg London, 2010.
[2]. B.Mukherjee, “Optical WDM networks,” ISBN 978-0-387-29188-8, Springer, USA 2006.
[3]. D. J. Blumenthal, P. R. Prucnal, and J. R. Sauer, “Photonic packet switches: Architectures and experimental implementation,” Proceedings of the IEEE, vol. 82, no. 11, pp. 1650–1667, November 1994.
[4]. R.Ramaswam, and K.N.Sivarajan, “optical networks a practical perspective,” ISBN: 978-0-12-374092-2,Morgan Kaufmann Publishers, Elsevier, USA ,2002.
[5]. B.Shihada and P-H.Ho, "Transport Control Protocol (TCP) in Optical Burst Switched Networks: Issues, Solutions, and Challenges", IEEE Communications Surveys and Tutorials, pp.70-86 , Second quarter 2008.
[6]. L. Brakmo, S. Brakmo, and L. Peterson, “TCP Vegas: New techniques for congestion detection and avoidance,” Proceedings of SIGCOMM, 1994.

[7]. L. Brakmo and L. Peterson, "TCP Vegas: end-to-end congestion avoidance on a global Internet," Journal on Selected Area, vol. 13, no. 8, pp. 1465-1480, October 1995.
[8]. Basem Shihada, Qiong Zhang, P-H-Ho, "Threshold-based TCP Vegas over Optical Burst Switched Networks," Computer Communications and Networks, 2006. ICCCN 2006. Proceedings.15th International Conference on 9-11 Oct. 2006.
[9]. Sing, Joel, and Ben Soh. "TCP New Vegas: performance evaluation and validation." In *Computers and Communications, 2006. ISCC'06. Proceedings. 11th IEEE Symposium on*, pp. 541-546. IEEE, 2006.
[10]. Sing, Joel, and Ben Soh. "TCP New Vegas revisited." In *Networks, 2005. Jointly held with the 2005 IEEE 7th Malaysia International Conference on Communication., 2005 13th IEEE International Conference on*, vol. 2, pp. 6-pp. IEEE, 2005.
[11]. BShihada, S.El-Ferik, and P-H.Ho, "FAST TCP over Optical Burst Switched Networks: Modelling and Stability Analysis" Optical Switching and Networking , Elsevier, Science Direct, Volume 10, Issue 2, pp.107–118, April 2013.
[12]. C. Jin, D. Wei, and S. Low, "FAST TCP: Motivation, Architecture, Algorithms, Performance," Networking, IEEE/ACM Transactions on Volume: 14, Issue: 6, pp.1246 - 1259, Dec. 2006
[13]. J. Wang, D.X.Wei, and S.Low, "Modelling and Stability of FAST TCP," Proceedings of IEEE International Conference on Computer Communication, Miami, FL, pp. 938-948, March 2005.
[14]. C. Jin, D. Wei, et al, "FAST TCP: From Theory to Experiments", IEEE Network, 19, pp.4-11, February 2005.
[15]. Y. Xiong, M.Vandenhoute, and C.Cankaya, "Control Architecture in Optical Burst-Switched WDM Networks," IEEE Journal on Selected Areas in Communications, 18, no 10, pp. 1838–1851, Oct. 2000.
[16]. J.Xu, C.Qiao, J.Li, and G.Xu, " Efficient Channel Scheduling Algorithms In Optical Burst Switched Networks," INFOCOM Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies (Volume: 3), San Francisco, pp. 2268 - 2278, April 2003.
[17]. Figueiredo, Gustavo B., and Nelson LS da Fonseca. "Channel reusability for burst scheduling in OBS networks." *Photonic Network Communications* (2013): 1-11.

AUTHORS

N. Sreenath is a professor in the Department of Computer science and Engineering at Pondicherry Engineering College Pillaichavady, Puducherry – 605014, India. He received his B.Tech in Electronics and Communication Engineering (1987) from JNTU College of Engineering, Ananthapur – 515002, Andra Pradesh, India. He received his M.Tech in Computer science and Engineering (1990) from University of Hyderabad, India. He received his Ph.D in Computer science and Engineering (2003) from IIT Madras. His research areas are high speed networks and Optical networks.

Ratna Pavani.K is an Assistant Professor in Indira Gandhi College of Arts and science, A government of Puducherry undertaking. She is pursuing a Doctoral Degree in Computer science Mother Theresa Women's University Kodaikanal. She received her MCA degree from Nagarjuna University Nagarjuna Nagar, Guntur, AP, 522510. She completed her MPhil (Computer science) from Manonmanian sundarnar University Tirunalveli. Areas of research include high speed networks and optical networks.