Journal of Computer Science 5 (12): 1055-1059, 2009
ISSN 1549-3636

Experimental Evaluation of Bottleneck Link Utilization with New-Additive Increase Multiplicative Decrease Congestion Avoidance and Control Algorithm

Hayder Natiq Jasem, Zuriati Ahmad Zukarnain, Mohamed Othman and Shamala Subramaniam
Department of Communication Technology and Networks,
Faculty of Computer Science and Information Technology,
University Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia

Abstract: Problem statement: As the Internet becomes increasingly heterogeneous, the issue of congestion control becomes ever more important. And the link utilization is one of the important things in term of congestion avoidance and control mechanisms. And we can define the utilization as simply the throughput divided by the access rate. And also all the developments for the congestion control and avoidance algorithms interest about the using of network resources and use the links capacity (utilization). **Approach:** In this research we continued to study the performances of the New-Additive Increase Multiplicative Decrease (AIMD) algorithm as one of the core protocols for TCP congestion avoidance and control mechanism, we want now to evaluate the effect of using the AIMD algorithm after developing it to find a new approach, as we called it the New-AIMD algorithm to measure the bottleneck link utilization and use the NCTUns simulator to get the results after make the modification of the mechanism. And we will use the Droptail mechanism as Active Queue Management (AQM) in the bottleneck router. **Results:** After implementation of our new approach with different number of flows, we will measure the bottleneck link utilization and we will get high utilization (more than 94%) for bottleneck link with using this mechanism and avoid the collisions in the link. **Conclusion:** Now and after got this results as high utilization for bottleneck link, we know the New-AIMD mechanism work as well under the giving network condition in the experiments.

Key words: Congestion control, TCP, AIMD, network utilization

INTRODUCTION

In a shared network such as the Internet, end systems should react to congestion by adapting their transmission rates to avoid congestion collapse and keep network utilization high[3]. The robustness of the current Internet is due in large part to the end-to-end congestion control mechanisms of TCP[12]. In particular, TCP uses an Additive Increase Multiplicative Decrease (AIMD) algorithm[6]; the TCP sending rate in congestion avoidance state is controlled by a congestion window which is halved for every window of data containing a packet drop and increased by one packet per window of data acknowledged.

The objective of these new congestion protocols is to address the needs of new multimedia applications and files transmitting. We notice that, like TCP, many of these proposals are based on the AIMD principle. Further, there is even a common belief that AIMD is optimal and is a necessary condition for a congestion control mechanism to be stable[2,4,5,16].

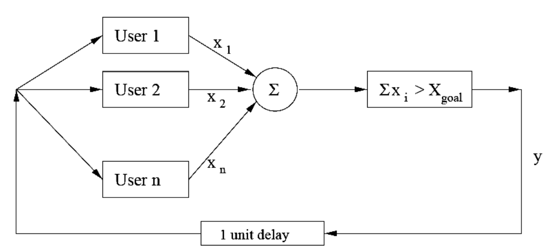

Fig. 1: Congestion system model[6]

AIMD congestion control was first studied by Chiu and Jain[6]. Figure 1 shows the system model they used to analyze a congestion control system. In this model, xi denotes the load generated by user i.

The congestion status of the network at time t is measured by:

$$X(t) = \underline{\Sigma}xi(t)$$

When $X(t) > X_{goal}$, the network is considered to be congested and the network sends a signal y = 1 to ask

Corresponding Author: Hayder Natiq, Department of Communication Technology and Networks, Faculty of computer Science and Information Technology, University Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia

all users to slow down; otherwise, the network indicates no congestion by sending y = 0. In this case, all users increase their load.

Formally, the dynamics of the system can be specified as:

$$\begin{aligned} X(t) &= \sum xi(t) \\ y(t) &= \begin{cases} 0 & \text{if } X(t) \le X_{goal} \\ 1 & \text{otherwise} \end{cases} \\ x_i(t+1) &= \begin{cases} a_I + b_I x_i(t) & \text{if } y(t) = 0 \\ a_D + b_D x_i(t) & \text{otherwise} \end{cases} \end{aligned} \quad (1)$$

It is important to notice that Eq. 1 assumes homogeneous delay for all users, with a unit delay on each feedback link.

And also, a file of size f with a total transfer time of Δ on a TCP connection results in a TCP transfer throughput denoted by r and obtained from Eq. 2:

$$r = f / \Delta \quad (2)$$

We can also derive the bandwidth utilization, pu, assuming that the link bandwidth is B, by Eq. 3:

$$pu = r / B \quad (3)$$

Generally, we have many versions of TCP in-use like (TCP-Tahoe, TCP-Reno, TCP-NewReno)[8-10], experimental TCP versions (TCP-SACK, TCP-Vegas)[11,12], as well as special-purpose TCP versions (T/TCP)[13]. And all these standards of TCP have AIMD as the core mechanism of congestion avoidance.

The Additive Increase/Multiplicative Decrease (AIMD), (AIMD-FC) and (New-AIMD) algorithms are described in details in[7,8,14,15].

Active Queue Management (AQM): The AQM algorithm runs on a router, which updates and feedbacks congestion information to end-users. The feedback is usually in the form of packet loss, delay, or marking. There are a many types of AQMs proposed, but in this research we will just explain and use the Droptail AQM[16,19].

Droptail AQM: Droptail (DT) is the simplest and most commonly used algorithm in the current Internet gateways, which drops packets from the tail of the full queue buffer. Its main advantages are simplicity, suitability to heterogeneity and its decentralized nature. However this approach has some serious disadvantages, such as no protection against the misbehaving or non-responsive flows (i.e., flows which do not reduce their sending rate after receiving the congestion signals from gateway routers) and no relative Quality of Service (QoS)[1,16]. But in this research we will use our approach[14] to reduce the sending rate by AIMD mechanism after receive signal from the router about the congestion occur.

The QoS is idea in the traditional "best effort" Internet, in which we have some guarantees of transmission rates, error rates and other characteristics in advance. QoS is of particular concern for the continuous transmission of high-bandwidth video and multimedia information. Transmitting this kind of content is difficult on the present Internet with DT.

Generally DT is used as a baseline case for assessing the performance of all the newly proposed gateway algorithms[17,18].

Network utilization: Network utilization is the ratio of current network traffic to the maximum traffic that the port can handle. It indicates the bandwidth use in the network. While high network utilization indicates the network is busy, low network utilization indicates the network is idle. When network utilization exceeds the threshold under normal condition, it will cause low transmission speed, intermittence, request delay and so on[1,10,16].

Networks of different types or in different topology have different theoretical peek value under general conditions. However, this doesn't mean that the higher network utilization is the better. We must make sure there is no packet loss when network utilization reaches a certain value. For a switched Ethernet, 50% network utilization can be considered as high efficiency. If using router or hub as core switch device in the network, the network utilization should be lower than the link bandwidth capacity to avoid the increasing collisions. Through monitoring network utilization, we can understand whether the network is idle, normal or busy. In our approach when we implement the New-AIMD mechanism we try to get high bottleneck link utilization more than 94% from link capacity (network resource). And we will make many experiments depends on number of flows using the link at same time, to show the different between them.

National Chiao Tung University network simulator (NCTUns): The NCTU network simulator is a high-fidelity and extensible network simulator and emulator capable of simulating various protocols used in both wired and wireless IP networks. The NCTUns can be used as an emulator, it directly uses the Linux TCP/IP protocol stack to generate high-fidelity simulation results and it has many other interesting qualities. It can simulate various networking devices. For example, Ethernet hubs, switches, routers, hosts, IEEE 802.11 wireless stations and access points, WAN (for purposely delaying/dropping/reordering packets), optical circuit switch, optical burst switch, QoS DiffServ interior and boundary routers. It can simulate various protocols for example, IEEE 802.3 CSMA/CD MAC, IEEE 802.11 (b) CSMA/CA MAC, learning bridge protocol, spanning tree protocol, IP, mobile IP, Diffserv (QoS), RIP, OSPF, UDP, TCP, RTP/RTCP/SDP, HTTP, FTP and telnet[20].

MATERIALS AND METHODS

We have implemented our evaluation plan for bottleneck link utilization on the NCTUns network simulator. The network topology used as a test-bed is the typical single-bottleneck dumbbell, as shown in Fig. 2.

For the simulation scenario as a general case we will have the following setup details:

The link's capacity at the senders, receivers and bottleneck link is 5Mbps. We used an equal number of senders and receivers nodes. The propagation delay was set to 30 ms for all links. All DT queues have 100 packet as queue length.

In this study as well we will make the experiment to implement the New-AIMD algorithm and then we can measure and evaluate the bottleneck link utilization. In our scenario TCP/FTP (using New-AIMD) flows are entering the system in the first second from the system start time.

RESULTS AND DISCUSSION

In Fig. 3-6, we observe the behavior of the using of bottleneck link capacity (utilization) with our mechanism (New-AIMD). In this experiment, we measure the link utilization in every second. We show that this mechanism works very well, under the given network conditions. And the shown results depend on the number of flows. In the Table 1 and Fig. 7, we show the average of bottleneck link utilization vs. number of flows use this link in same time and we measured this average after five second from the start time of the experiment, when the transport rate is stable.

In Fig. 3-6, we shown the result for experiment with (2-5) flows use the link in the same time and we can see the average of use the link (utilization) is more than 94%, we measured this average after 5 sec from start time. We showed this average in Table 1 and Fig. 7.

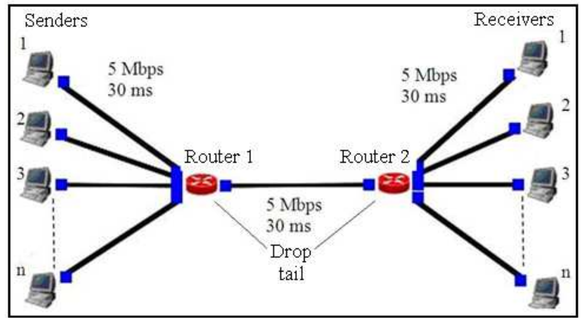

Fig. 2: Multiple flow experimental set-up for New-AIMD evaluation

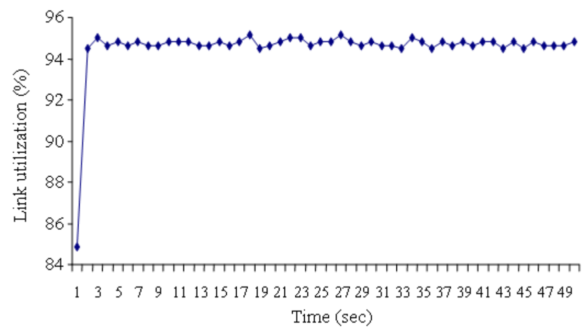

Fig. 3: Link utilization for bottleneck link with two flows

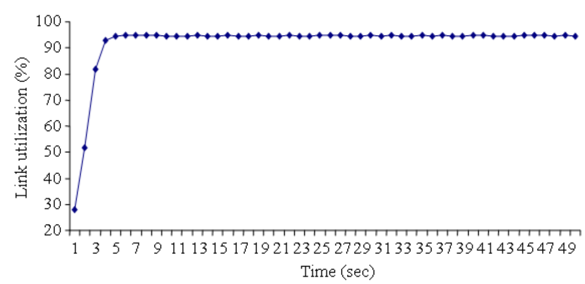

Fig. 4: Link utilization for bottleneck link with three flows

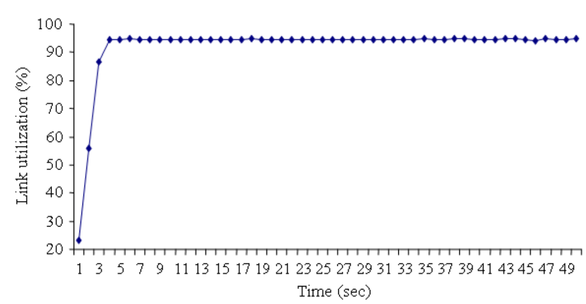

Fig. 5: Link utilization for bottleneck link with four flows

Table 1: No. of flows vs. average of bottleneck link utilization

No. of flows	Average of link utilization (%)
Two	94.73957
Three	94.68565
Four	94.59935
Five	94.48913

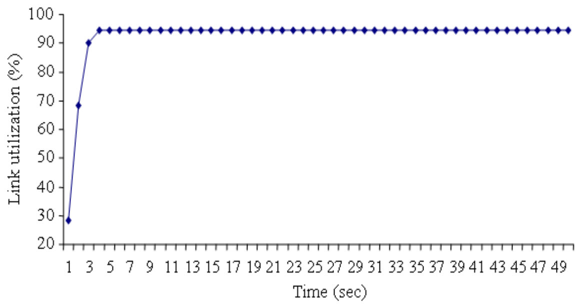

Fig. 6: Link utilization for bottleneck link with five flows

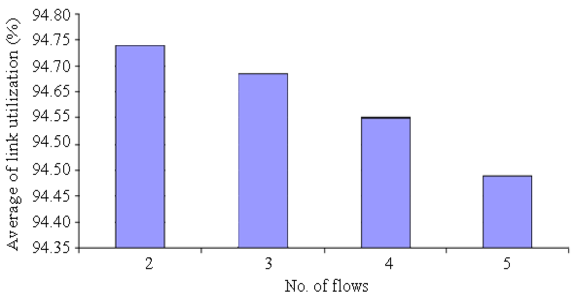

Fig. 7: Average of link utilization for bottleneck link with different numbers of flows

CONCLUSION

In this study we have experimental evaluation for one performance of New-AIMD mechanism it is the utilization of bottleneck link. We found the results after implement the New-AIMD algorithm and got the high utilization (more than 94%) for the link and avoid the congestion in this experiment for multi flows use the same link at same time. Then we can say this mechanism work as well under the conditions for network experiments above.

And the benefit from implementing the New-AIMD algorithm in this study, is to increase the average of using link bandwidth capacity for network resources (utilization), beside of avoids the network congestion as the major work for this algorithm.

REFERENCES

1. Kunniyur, S. and R. Srikant, 2004. Stable, scalable, fair congestion control and AQM Schemes that achieve high utilization in the internet. IEEE Trans. Automat. Control, 49: 2024-2029. http://comm.csl.uiuc.edu:80/~srikant/Papers/network
2. Handley, M., S. Floyd, J. Pahdye and J. Widmer, 2007. TCP Friendly Rate Control (TFRC): Protocol Specification. RFC 3448. http://www.icir.org/floyd/talks.html
3. Sally Floyd and Kevin Fall, 1999. Promoting the use of end-to-end congestion control in the internet. IEEE/ACM Trans. Network. 7: 458-472. http://cseweb.ucsd.edu/classes/wi01/cse222/papers/floyd-e2e-promote-ton99.pdf
4. Mahdavi, J. and S. Floyd, 1997. TCP-friendly unicast rate based flow control. Note sent to the end2end-interest mailing list. http://www.psc.edu/networking/papers/tcp_friendly.html
5. Peterson, L.L. and B.S. Davie, 1999. Computer Network: A Systems Approach. 2nd Edn,, Morgan Kaufmann Publishers.
6. Chiu, D. and R. Jain, 1989. Analysis of the increase/decrease algorithms for congestion avoidance in computer networks. J. Comput. Networks ISDN., 17: 1-14. http://pages.cs.wisc.edu/~suman/courses/740/papers/chiu89isdn.pdf
7. Lahanas, A. and V. Tsaoussidis, 2002. Additive Increase Multiplicative Decrease-Fast Convergence (AIMD-FC). Proceedings of the Networks 2002, Aug. 2002, Atlanta, Georgia.
8. Lahanas, A. and V. Tsaoussidis, 2003. Exploiting the efficiency and fairness potential of AIMD-based congestion avoidance and control. J. Comput. Networks, 43: 227-245. http://utopia.duth.gr/~vtsaousi/aimd-fc-comnet.pdf
9. James F. Kurose and Keith W. Ross, 2004 Computer Networking. 3rd Edn., Addison-Wesley, Boston, ISBN: 10: 0321227352.
10. William Stallings, 2004. Data and Computer Communications. 7th Edn., Prentice Hall, USA., ISBN: 10: 0131006819.
11. John Nagle, 1984. Congestion control in TCP/IP internetworks. SIGCOMM Comput. Commun. Rev., 14: 11-17. http://www.sce.carleton.ca/faculty/lambadaris/recent-papers/globecom2002.pdf

12. Van Jacobson, 1988. Congestion avoidance and control. Proceedings of SIGCOMM, Aug. 1988, ACM Stanford, CA., pp: 314-329. http://ee.lbl.gov/papers/congavoid.pdf
13. Mascolo, S., C. Casetti, M. Gerla, M.Y. Sanadidi and Ren Wang, 2001. Tcp westwood: Bandwidth estimation for enhanced transport over wireless links. Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, (MCN'01), ACM Press, New York, USA., pp: 287-297. http://www.cs.ucla.edu/NRL/hpi/tcpw/tcpw_papers/2001-mobicom-0.pdf
14. Hayder Natiq, Zuriati Ahmed, Mohamed Othman and Shamala Subramaniam, 2008. The TCP-based new AIMD congestion control algorithm. Int. J. Comput. Sci. Network Secur., 8: 331-338. http://paper.ijcsns.org/07_book/200810/20081051.pdf
15. Hayder Natiq Jasem, Z.A. Zukarnain, M. Othman and S. Subramaniam, 2009. Fairness of the TCP-based new AIMD congestion control algorithm. J. Theor. Applied Inform. Technol., 5: 568-576. http://www.jatit.org/volumes/research-papers/Vol5No5/6aVol5No5.pdf
16. Michael Welzl, 2005. Network Congestion Control Managing Internet Traffic. 1st Edn., John Wiley and Sons Ltd., ISBN: 10: 0-470-02528-X.
17. Haider, A., H. Sirisena, K. Pawlikowski and M.J. Ferguson, 2003. Congestion control algorithms in high speed telecommunication networks. Proceedings of the Annual ORSNZ Conference, University of Canterbury, Christchurch, NZ. www.orsnz.org.nz/conf36/papers/Haider.pdf
18. Eitan Altman, Chadi Barakat and V.M. Ramos, 2005. Analysis of AIMD protocol over path with variable delay. J. Comput. Networks, 48: 960-971. DOI: 10.1016/J.COMNET.2004.11.013
19. Koo, J., S. Ahn and J.W. Chung, 2005. A comparative study of queue, delay and loss characteristics of AQM schemes in QoS-enabled networks. Comput. Inform. J., 22: 1001-1019.
20. Wang, S.Y., C.L. Chou and C.C. Lin, 2007. The design and implementation of the NCTUns network simulation engine. Simulat. Model. Pract. Theor., 15: 57-81. http://nsl.csie.nctu.edu.tw/simpat2007.pdf