Journal of Engineering Science and Technology
Vol. 3, No. 3 (2008) 265 - 271

A NEW CONGESTION MANAGEMENT MECHANISM FOR NEXT GENERATION ROUTERS

MOHAMMED M. KADHUM*, SUHAIDI HASSAN

Graduate Department of Computer Science, College of Arts and Sciences,
Universiti Utara Malaysia, 06010 UUM, Sintok, MALAYSIA
*Corresponding Author: kadhum@uum.edu.my

Abstract

While computer networks go towards dealing with varied traffic types with different service requirements, there is a necessity for modern network control mechanisms that can control the network traffic to meet the users' service requirements. Optimizing the network utilization by improving the network performance can help to accommodate more users and thus increase operators' profits. Controlling the congestion at the gateway leads to better performance of the network. Sending congestion signal sooner can be of great benefit to the TCP connection. In this paper, we propose Fast Congestion Notification (FCN) mechanism which is a new method for managing the gateway queues and fast sending of congestion signal to the sender. We tested our mechanism on Explicit Congestion Notification (ECN) packets which have higher priority; we achieved good results in terms of faster congestion signal propagation and better network utilization. Our analysis and simulations results show that the use of FCN over TCP connections sharing one bottleneck can improve the throughput, having less loss, less delay time, and better network utilization.

Keywords: Explicit Congestion Notification (ECN), Fast congestion notification (FN).

1. Introduction

1.1. Network management economics

Computer networks such as Internet are costly in term of building and action. Therefore, it is consequential to optimize the network utilization. Optimizing the network utilization allows us to increase as much as possible the number of users that the network resources can concurrently accommodate. The network performance is the main factor by which the users' satisfaction is determined. The user experiences the performance during using the network services. It is important to network operators and the owners as well, to improve efficiency the network performance as well as its utilization.

1.2. Problems of congestion

Congestion can happen in computer networks when the offered load and the demands for network resources exceed the network capacity. During congestion, router buffers are overflowed causing large queuing delays, and high packet loss. Congestion reduces the effective utilization of network resources and causes degradation in the performance experienced by the network users [1]. Therefore, it is worthwhile to reduce the occurrence of congestion situations in a network to optimize the utilization of network resources to provide the network users with suitable performance.

1.3. Queue management and congestion control

Congestion can be avoided when the traffic arrival rate to a gateway maintained close to the outgoing link capacities and the gateways' queue sizes kept small to guarantee the availability of buffer capacity for successful buffering and consequent forwarding of temporary traffic upsurges which could otherwise cause buffer overflows and packet loss [2]. Congestion management is the combined responsibility of network gateways and end-point hosts. Gateways are invested with the ability to delay or drop the packets inside the network. Gateways are responsible for congestion detection and notification delivery, queue's traffic arrival rate control, and queue size control. Traffic sources are responsible for the adjustment of their data transmission rates to enable the gateways to achieve their goals.

1.4. Congestion information

The congested gateway sends congestion information to a source in implicit or explicit manner. When sending is explicit, the gateway sends information in packet headers or in control packets such as Source Quench packets [3], choke packets [4], state-exchange packets [5,6], rate control messages [7,8], or throttle packets [9] to the source. An implicit sending happens when a source uses probe values [10], retransmission timers [11], throughput monitoring, or delay monitoring to indicate the occurrence of congestion.

Explicit congestion signalling imposes an extra burden on the network, since the network needs to transmit more packets than usual, and this may lead to a loss in efficiency if the signalling overhead is not controlled properly. On the other hand, with implicit signalling, a source may not be able to distinguish between congestion and other performance problems such as hardware problems.

1.5. Explicit congestion notification (ECN)

ECN was proposed for TCP/IP networks as a method of explicitly informing end-hosts of network congestion by marking packets instead of dropping them [12].

ECN, which has been proven to be better way of delivering congestion information to the source host [12], has a better transfer delay for short-lived flows than packet drop schemes [13,14]. In addition to reducing the number of timeouts for TCP flows, ECN mechanism does not require generation of additional traffic at the router and can be easily implemented in the data path of routers; it requires setting of a single bit.

ECN uses two bits in the IP header to carry information which indicates the router that the packet is ECN-capable or not (ECT1, ECT0 and Not-ECT codepoints). This information allows a congested router to mark the packet (by set EC codepoint) instead of dropping it as an indication of congestion. Also, ECN uses two bits in TCP header, ECE bit (ECE-Echo) for negotiating ECN-capability and inform the sender about the congestion, CWR bit is used to enable the TCP receiver to determine when to stop setting the ECN-Echo or whether it has reduced its congestion window.

Floyd [14] shows the benefits of using ECN with TCP and proposes guidelines for TCP's response to ECN fields in packet header. And she also verified that the performance of TCP congestion control is affected by the feedback delay involved in the congestion information reaching the source host from the bottleneck node.

Generally ECN implementations use the mark-tail strategy [12], i.e. when congestion is detected the router marks the incoming packets that have just entered the buffer of each router. Nevertheless, a received marked packet can experience a queuing delay until all earlier buffered packets have been transmitted. Thus, to reduce this queuing delay, Liu and Jain [15] proposed the mark-front strategy where a packet is marked at the time it is sent, in that way providing a faster congestion information delivery and reflecting the up-to-date congestion information.

Nevertheless, as a packet can still have a queuing delay in the buffer at each transit (intermediate nodes); the congestion information enclosed in an incoming packet cannot be directly transferred to its destination [16].

After estimating head marking (a modification of routers behaviour which allows to faster propagating congestion signals) on a single congested link and more complex link, Malowidzki [17] inferred that either the RED mechanism should be tuned or it should be replaced by a new AQM-type approach since the aggregate goodput of all TCP sources was the same for both ECN and non-ECN cases.

Kadhum and Hassan [18] prove that ECN improves the performances of short TCP sessions in RED network in case of having variation of files size and number of senders sharing one bottleneck; also it shows that ECN increases the throughput and generally reduces the delay.

2. Motivations and Objectives

Optimizing the network utilization allows us to accommodate as many users as possible the number of users that the network resources can concurrently be allocated. The network performance is the main factor by which the user's satisfaction is determined. The user experiences the performance during using the network services. It is important to improve the network performance as well as its utilization to increase the profits. The future availability of cheaper buffers and faster links and processors will not alleviate network congestion completely.

3. Fast Congestion Notification (FN)

Sending congestion signal as soon as possible can be of great benefit to the TCP connection, avoiding the severe plenty of a retransmit timeout for a connection has not yet started placing load on the network. We built Fast Notification (FN) mechanism to allow sending congestion signal faster so that the sender can reduce its congestion window sooner. This technique includes (Fig. 1):

(a) Creating queues and assign packets to those queues depend on the packet priority.

(b) Prioritizing for different traffic such as delay sensitive, interactive transaction-based applications (like desktop video conferencing) and all traffic which has higher priority than the rest.

(c) Mixing different queuing methods to guarantee sending the higher priority traffic faster and guarantee fairness and treatment for the rest of traffic.

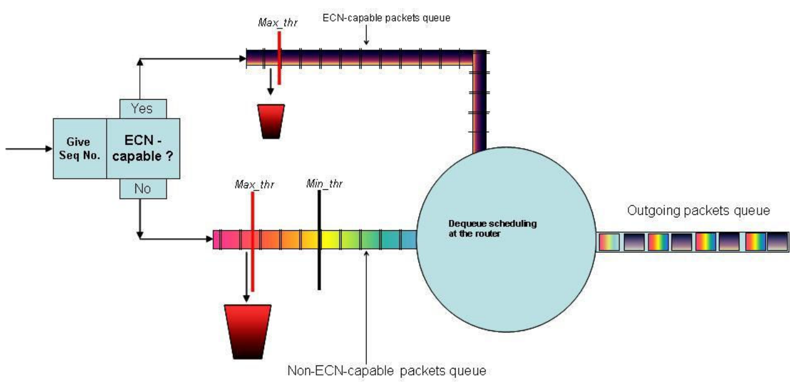

Fig. 1. FN in Operation.

What FN does?

FN monitors the queue size, and uses some measure of it, to detect congestion and to calculate the packet drop probability which is applied to the arriving packet. FN rules:

(a) The rate at which packets are discarded from the network in order to reduce the queue's traffic arrival rate to less than the outgoing link capacity.

(b) The rate at which congestion notifications are delivered to the source.

In response to congestion notifications, the source reduces its data rate, by making right adjustments to its end-to-end flow control algorithms which rule the rate at which it injects data into the network. The behavior of gateways in dropping packets, detection of congestion, and initiation and delivery of congestion notifications to sources, effects the data rate of traffic sources, the total traffic arrival rate to the gateway and the queues, and the size of the queues at the gateway.

4. Results

We tested our mechanism on Explicit Congestion Notification (ECN) packets which has higher priority. We achieved good results in term of faster congestion signal propagation and good network utilization.

Fig. 2 shows the throughput when we applied the proposed mechanism to ECN while Fig. 3 shows the throughput when use RED with ECN.

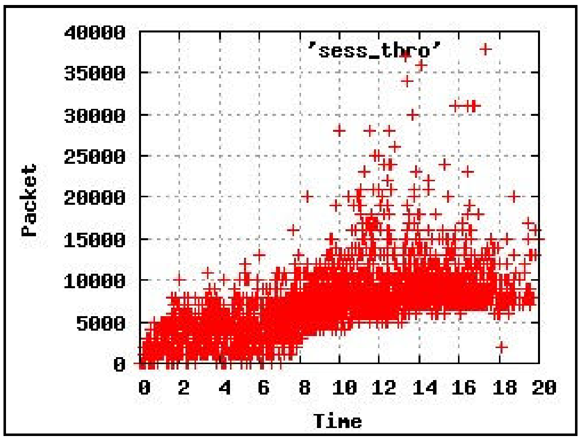

Fig. 2. Throughput with FN.

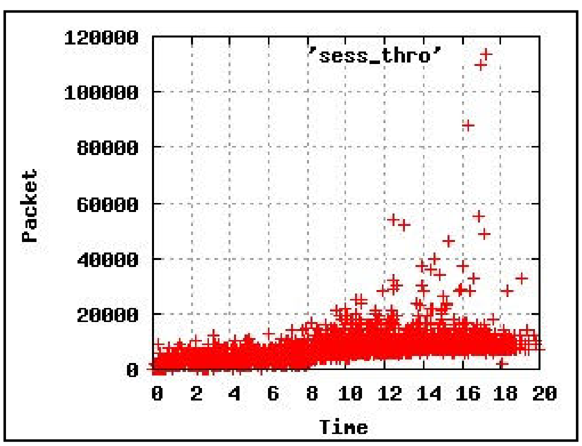

Fig. 3. Throughput without FN.

5. Conclusion and Future Work

In this paper, we have presented a queue management scheme, called Fast Congestion Notification (FN). CN enables ECN-TCP connections to send congestion information to the source quickly and before the congestion heppens. Thus, the source reacts depending on the feedback information and regulates its response to the congestion.

We evaluated the use of CFN in ECN-TCP connections using ns 2.29 simulator for a number of users sharing one bottleneck.

Our work verifies that:

(a) FN improves the performances of ECN-TCP connections in case of having number of senders sharing one bottleneck
(b) FN increases the throughput and reduces the delay.
(c) ECN is much more powerful than the simple packet drop indication.

We are applying FN over short TCP sessions which use small initial windows to investigate the effect of FN. Especially, when congestion loss occurs within this early time of TCP slow start, there are not enough packets in the network to generate the three duplicate ACKs which are necessary to initiate fast retransmit and fast recovery. The TCP sender has to wait till the end of the retransmission timeout (RTO). We also are going to apply FN on wireless network to see its effect and TCP behavior when use FN over wireless networks. In addition we are going to check the stability of CFN for high distance bandwidth networks.

References

1. Jacobson, V. (1988). Congestion avoidance and control. *Proceedings of SIGCOMM* 1988, 314-329.
2. Kuzmanovic, A., Floyd, S. and Ramakrishnan, K. (2006). Adding explicit congestion notification to (ECN) to TCP’s SYN/ACK packets. IETF Draft, draft-ietf-tcpm-ecnsyn-00.txt.
3. Prue, W. and Postel, J. (1987). Something a host could do with source quench: The source quench introduced delay (SQUID). *Request for Comments*, 1016, Network Working Group.
4. Majithia, J.C., Irland, M., Grang’e, J-L., Cohen, N. and O'Donnell, C. (1979). Experiments in congestion control techniques. Proc. *Int'l. Symp. Flow Control Computer Networks*, Versailles, France, 211–234.
5. Ko, K., Mishra, P.P. and Tripathi, S.K. (1991). Predictive congestion control in high-speed wide-area networks. In *Protocols for high speed networks II*, Elservier Science Publishers, North Holland.
6. Sabnani, K.K. and Netravali, A.N. (1989). A high speed transport protocol for Datagram/Virtual circuit networks. *Proc. ACM SigComm* 1989, 146–157.
7. ATM Forum (1996). Traffic management specification. Version 4.0, April 1996.
8. Williamson, C.L., and Cheriton, D.R. (1991). Loss-load curves: Support for rate-based congestion control in high-speed datagram networks. *Proc. ACM SigComm* 1991.
9. Kamoun, F. (1981). A drop and throttle flow control policy for computer networks. *IEEE Trans. on Commun*., Vol. Com-29, No. 4, 444–452.

10. Keshav, S. (1991). Congestion control in computer networks, Ph.D. thesis. University Of California.
11. Zhang, L. (1986). Why TCP timers don't work well? *Proc. SigComm* 1986, 397-405.
12. Floyd, S. and Ramakrishnan, K. (1999). A proposal to add explicit congestion notification to IP. *RFC* 2481.
13. Floyd, S. (1994). TCP and explicit congestion control. *ACM Computer Communication review*, 24(5), 10-23.
14. Hadi, J. and Ahmed, U. (2000). Performance evaluation of explicit congestion notification (ECN) in IP networks. *RFC* 2884.
15. Liu, C., and Jain, R. (2001). Improving explicit congestion notification with the mark-front strategy. *Computer Networks*, 35(2-3), 185-201.
16. Byung-Chul Kim and You-Ze Cho. (2002). Mark-relay strategy for explicit congestion notification in the internet. *Electronics Letters*, Vol. 38, No. 12, 612-614.
17. Malowidzki, M. (2003). Simulation-based study of ECN performance in RED networks. *SPECTS*'03.
18. Kadhum, M.M. and Hassan, S. (2007). The effect of ECN on short TCP sessions. Telecommunications and Malaysia International Conference on Communications, 2007. *ICT-MICC* , 708-712.