A Novel Rate-Based Hop By Hop Congestion Control Algorithm

Shu-Ching Chen
School of Computer Science
Florida International University
Miami, FL, USA
chens@cs.fiu.edu

Mei-Ling Shyu
Department of Electrical & Computer Engineering
University of Miami
Coral Gables, FL, USA
shyu@miami.edu

Chengjun Zhan
School of Computer Science
Florida International University
Miami, FL, USA
czhan002@cs.fiu.edu

Srinivas Peeta
School of Civil Engineering
Purdue University
West Lafayette, IN, USA
peeta@ecn.purdue.edu

Abstract

As the development of Internet continues, congestion control has become a big issue to the computer network society. Most of the congestion control schemes fall into two categories: end-to-end and hop-by-hop schemes. In this paper, we propose a novel hop-by-hop algorithm that originates from a classical traffic control algorithm. The experimental results show that our proposed algorithm can achieve short delays and quick responses to the congestion situations and cause no packet loss. It can also minimize the bandwidth requirement and achieve very high buffer usage level for nodes along the transmission path.

1. Introduction

Nowadays, our capabilities to obtain multimedia information through the Internet have been increasing rapidly. With more and more real-time multimedia applications being deployed, the problems of quality degradation and packet losses become deteriorated. Hence, a good congestion control algorithm dedicated for real-time multimedia transmission is very important and demands great attentions.

End-to-end congestion control mechanisms have been intensively studied [1][2][3][4][5][6] and are the typical congestion control mechanisms on the Internet. An end-to-end congestion control mechanism usually uses some explicit or implicit information about the network conditions sent by the receiver to adjust the sending rate or window size at the sender [7]. End-to-end congestion control mechanisms are easy to be implemented and proven to be efficient. However, it takes at least one round-trip time (RTT) for the sender to get the feedback information, detect congestion, and make adjustments accordingly. In the current Internet environment, this process may take a long time, e.g., up to several seconds. In that time period, congestion may be aggravated, which thus causes packet losses and wastes network resources.

Recently, alternative mechanisms called Hop-by-Hop (HBH) congestion control mechanisms [8][9][10][11][12][13] have emerged and may provide better solutions to the problems mentioned above. In HBH mechanisms, traffic control is used at each node along the traffic flow path. As a consequence, congestion along the traffic flow path could be detected immediately and appropriate control could be carried out (e.g., buffering packets) at the immediate upstream node to alleviate the congestion condition. Finally, this congestion-alleviating process propagates to the sender to adjust its sending rate. HBH mechanisms could effectively ease network congestion and minimize network resource waste.

In this paper, a novel hop-by-hop congestion control algorithm is introduced based on a well-known ramp-metering algorithm in transportation networks. At the first glance, transportation networks and computer networks are completely different: they belong to different domains and handle different things. Basically, transportation networks deal with objects like roads, intersections, traffic signals, vehicles, traffic control center, etc.; while in computer networks, efficient and reliable transfer of electronic data is the major concern. However, transportation networks and computer networks do have many similarities. For example, the vehicles running on the roads could be mapped to the packets transferred through computer networks, and the road intersections could be mapped to the switches/routers in the computer networks, etc. Hence, the algorithms and theories proved to be efficient in transportation networks could be potentially beneficial to the computer network research.

In the recent years, ramp metering has emerged as an effective freeway control measure to ensure efficient freeway operations [14] in transportation networks. Ramp metering can be defined as a method by which the traffic seeking to gain the access to a busy highway is controlled at the access (merge) point via traffic signals [15]. To effectively use ramp metering as a control measure, various algorithms have been proposed and developed which fall into two categories: those algorithms that are based on local information (ALINEA [16]) and those algorithms that consider the area-wide factors (FLOW [17]).

In this paper, a framework applying the ideas in ALINEA to computer network research is proposed. The proposed framework has the advantage of a quick response to delays, and thus it could have better performance in comparison with the end-to-end congestion control algorithms such as the TCP flow control scheme [1]. As shown in the paper, if our framework is deployed on all the nodes along the traffic flow path, no packet loss occurs on the network.

The rest of the paper is organized as follows. The ALINEA algorithm and how to modify it to fit into computer network congestion control are presented in Section 2. Experiments have been conducted and the effectiveness of the algorithm is shown in Section 3. Section 4 gives the concluding marks and future work.

2. Control algorithm design

2.1 ALINEA introduction

ALINEA was developed based on the classical feedback theory [14][16]. The key point of the algorithm is to maintain an optimal mainline occupancy, while at the same time, to maximize the throughput. The basic idea of the algorithm can be stated using the following formula:

$R(k) = R(k-1) + W * (O - O(k))$, where:

- R(k) means the metering rate of a mainline section at time interval k;
- R(k-1) is the metering rate at k-1;
- O is the predefined desired occupancy;
- O(k) is the measured occupancy at k; and
- W is the weight factor.

According to this formula, if at a time interval k, O(k) is greater than O, then (O-O(k)) will be negative and thus R(k) will be decreased comparing to R(k-1). On the other hand, if O(k) is smaller than O, then R(k) will be increased comparing to R(k-1) since (O-O(k)) is positive. ALINEA attempts to get the desired occupancy regardless of the upstream traffic volume. It could automatically adjust to achieve a desired occupancy in cases of the congested traffic and light traffic.

2.2 The proposed framework

To apply ALINEA to computer network congestion control, the following mappings are made. A mainline section is mapped to a network node; the mainline occupancy is mapped to the buffer occupancy of a node; and apparently, the traffic rate is mapped to the network transmission rate.

Although ALINEA has been proved to be successful in transportation network [14][15][17], significant changes are needed to apply it to the computer networks. First, the proposed framework assumes that one buffer is associated with each incoming interface of a node, which corresponds to one or several upstream nodes. A node could have several incoming and outgoing interfaces, and it reports its network condition (i.e., buffer usage, bandwidth, etc.) to its neighboring nodes, either periodically or triggered by events.

The meanings of the parameters are also redefined. R(k) and R(k-1) mean the outgoing transmission rates of a node at time intervals k and (k-1), respectively. O is defined as the desired incoming interface buffer utilization of the immediate downstream node; while O(k) is the actual buffer usage level of the downstream node during time interval k. W is the weight factor.

Unlike the original ALINEA algorithm, in the proposed framework, the values of O and W are adaptable. When a connection starts, O is defined as a small value O_{min}, e.g., 15% of the buffer size; while a large W is used to rapidly increase the transmission rate to the optimal value. If at k, O(k) is greater than O, O(k) will be assigned to O until a predefined O_{max} is reached. At the same time, W is decreased until it is equal to a predefined W_{min}. The pseudo code for node control is shown as follows.

```
// Node initialization
$O = O_{min}$;
$W = W_{max}$;
// after initialization
while (1)
    Waiting for the control message from a downstream node
    $R(k) = R(k-1) + W * (O - O(k))$;
    if ($R(k) > R_{max}$)  $R(k) = R_{max}$;
    if (($O < O(k)$) and ($O(k) < O_{max}$))
    {$O = O(k)$; $W = W / 2$;}
    if ($W < W_{min}$)  $W = W_{min}$;
    if ($O(k) > O_{max}$) {$O = O_{max}$; $W = W_{min}$;}
end while;
```

In the proposed framework, if the calculated rate exceeds a threshold value (e.g., network capacity), the new rate is set to the threshold value so that the sender does not overrun the network. As can be seen from the pseudo code, in the start phase, the rate increases rapidly because (O – O(k)) and W are large. As O(k) increases and W decreases, the rate variation becomes smaller. Finally, a high buffer usage level can be reached (O_{max}), and the rate variation is very small.

3. Experimental results and analysis

The algorithm has been implemented and tested using the NS-2 [18] network simulator. Experiments have been conducted for the networks with 1, 2, 3, and 5 router(s) to test the transmission rate variations and buffer usage of the network nodes. In this paper, only the results for 1 router and 5 routers are shown and discussed since all the other experiments share similar characteristics.

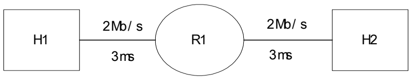

Figure 1: Configuration for 1-router scenario

Figure 1 gives the network configuration with one router, where R1 is the router between hosts H1 and H2. The packet size is 1,000 bytes. All the nodes have a buffer size of 200 packets. The bandwidth limit for all the links is 2Mb/s and the propagation delay is 3ms.

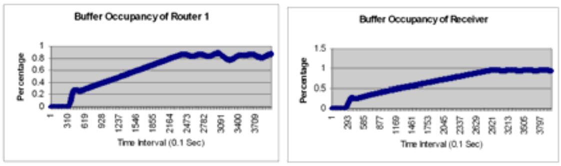

Figure 2: Buffer usage percentages of R1 and H2

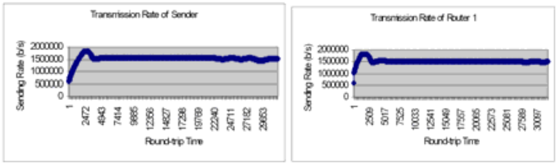

Figure 3: Transmission rate variations of H1 and R1

As shown in Figure 2, both R1 and H2 could achieve high buffer utilization and the actual buffer utilizations are very close to the desired buffer usage levels (for R1, 80% of the buffer size; for H2, 95% of the buffer size) with small variations. A predefined maximal rate could be set for a connection; either it is the bandwidth limit of the network or a user-defined value. As shown in Figure 3, when a connection starts, both the transmission rates of H1 and R1 increase dramatically. After the connection becomes stable, the transmission rate is close to the optimal rate with very small variations. As we can see from Figure 3, although the link bandwidth limit is 2Mb/s, the maximal actual transmission rate of R1 is 1.8Mb/s, which is a predefined maximal value.

The network configuration for the 5-router scenario is shown in Figure 4. The connection of hosts H1 and H2 traverses five routers. All the links have a propagation delay of 3ms. The bandwidth limits of the link R2 to R3 and the link R3 to R4 are 3Mb/s, while the other links have a bandwidth limit of 2Mb/s. The traffic packet size is 1,000 bytes. Each node has a buffer size of 200 packets.

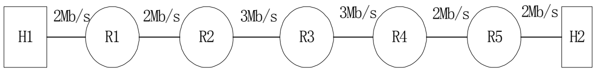

Figure 4: Configuration for 5-router scenario

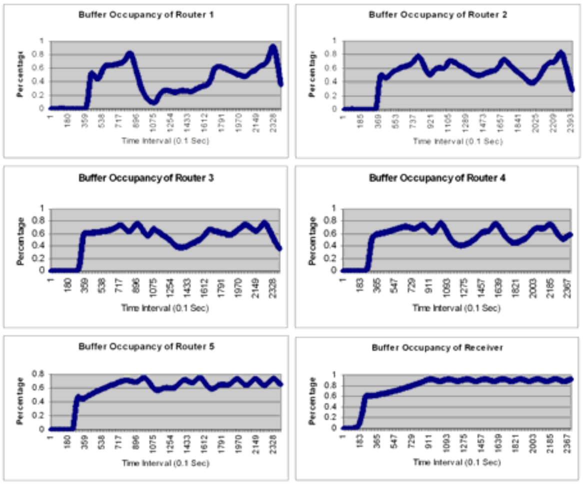

Figure 5: Buffer usage percentages of network nodes

As shown in Figure 5, each network node could achieve a high buffer usage level that is close to the predefined optimal buffer usage level (for the routers, 70% of the buffer size; for the receiver H2, 95% of the buffer size) after the network transmission becomes stable. There is no packet loss since none of the buffer utilization reaches 100%.

In the TCP flow control scheme, the detection of packet loss is the only event that leads to a reduction in the window size [8]. In contrast, as shown in Figure 2 and Figure 5, the proposed framework causes no packet loss. When the congestion occurs, in the TCP flow control scheme, no response is made until at least one RTT; while in the proposed framework, the response is given after only the transmission delays between two adjacent nodes (by suggesting a smaller transmission rate). For end-to-end delays, the TCP control scheme has sharp oscillations; while the proposed framework shows almost no oscillation when the connection becomes stable. The result figure is similar to that in [8] and is omitted here.

Figure 6 shows that initially, the transmission rates of all the nodes increase rapidly. However, after the nodes find out the desired optimal transmission rate for the connection, all the transmission rates are close to the optimal value with small variations. Another point needs to mention is that in a period of time, the transmission rates of H1, R1, and R5 are equal to 2Mb/s; while the transmission rates of R2 and R3 exceed 2Mb/s. The reason is that they have different bandwidth limits and the proposed algorithm makes sure the transmission rate will not exceed the bandwidth limit.

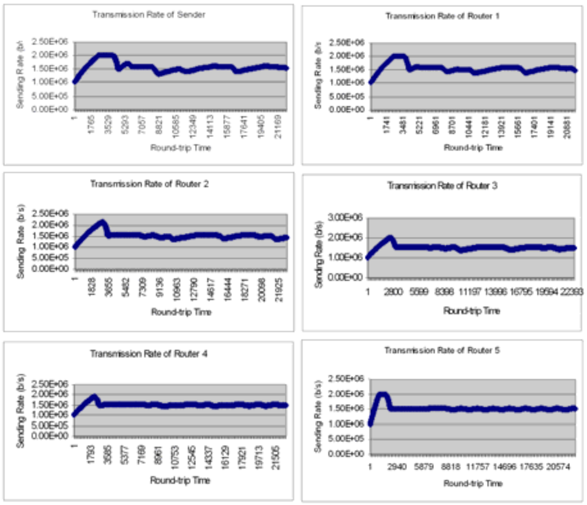

Figure 6: Transmission rates of network nodes

4. Conclusions and future work

In this paper, a novel hop-by-hop algorithm originating from the traffic control theory has been discussed for the packet-switched networks, and several experiments were conducted and analyzed. The experimental results demonstrated the effectiveness of our proposed algorithm since when the algorithm is applied, very high buffer usage level and optimal rate for multimedia applications could be achieved along the transmission path without packet losses. By using the proposed algorithm, a smooth play rate could be provided at the receiver side. It also has the advantages of shorter delays and quicker responses for congestion conditions.

Our future work will focus on comparing the algorithm with other well-known hop-by-hop algorithms such as HBH [8][11]. Another topic is to apply the algorithm to the wireless network area.

5. References

[1] V. Jacobson, "Congestion Avoidance and Control," *Proc. of ACM SIGCOMM*, Stanford, California, Aug. 1988.

[2] R. Rejaie, M. Handley, and D. Estrin, "RAP: An End-to-end Rate-based Congestion Control Mechanism for Realtime Streams in the Internet," *Proc. of IEEE INFOCOM'99*, New York, NY, March 1999.

[3] S. Jacobs and A. Eleftheriadis, "Real-time Dynamic Rate Shaping and Control for Internet Video Applications," *Proc. of the Workshop on Multimedia Signal Processing*, Princeton, NJ, June 1997.

[4] S. Floyd, M. Handley, J. Padhye, and J. Widmer, "Equation-based Congestion Control for Unicast Applications," *Proc. of ACM SIGCOMM 2000*, Stockholm, Sweden, Aug. 2000.

[5] M. Handley, J. Pahdye, S. Floyd, and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification," Internet Draft draft-ietf-tsvwg-tfrc-02.txt, Work in Progress, May 2001.

[6] M.-L. Shyu, S.-C. Chen, and H. Luo, "Optimal Resource Utilization in Multimedia Transmission," *Proc. of IEEE ICME*, pp. 880-883, Tokyo, Japan, Aug. 2001.

[7] K. K. Ramakrishnan and R. Jain, "A Binary Feeback Scheme for Congestion Avoidance in Computer Networks," *ACM Trans. Comp. Sys.*, 8(2), pp. 158-181, May 1990.

[8] P. Mishra, and H. Kanakia, "A Hop by Hop Rate-based Congestion Control Scheme," *Proc. of ACM SIGCOMM*, pp. 112-123, Aug. 1992.

[9] H. T. Kung and A. Chapman, "The FCVC Proposal for ATM Networks: A Summary," *Proc. of Int. Conf. Network Proctocols*, Oct. 1993.

[10] C. Ozveren, R. Simcoe, and G. Varghese, "Reliable and Efficient Hop-by-Hop Flow Control," *Proc. of ACM SIGCOMM*, London, UK, Aug. 1994.

[11] P. Mishra, H. Kanakia, and S. Tripathi, "On Hop-by-Hop Rate-Based Congestion Control," *IEEE/ACM Trans. on Networking*, 4(2), pp. 224-239, Apr. 1996.

[12] L. Costa, S. Fdida, and O. Duarte, "Hop by Hop Multicast Routing Protocol," *Proc. of ACM SIGCOMM*, San Diego, CA, Aug. 2001.

[13] B. Pang, X. Liu, Y. Xiang, and W. Gao, "A Novel Red-Based Hop-by-Hop Congestion Control," *Proc. of IEEE ICCNMC*, Beijing, China, Oct. 2001.

[14] M. Hasan, M. Jha, and M. Ben-Akiva, "Evaluation of Ramp Control Algorithms Using Microscopic Traffic Simulation," *Transportation Research Part C 10*, pp. 229-256, 2002.

[15] Ramp Metering Study, http://www.scotland.gov.uk /library5/transport/ramp_summ.pdf.

[16] M. Papageorgiou, H. Hadj-Salem, and J. M. Blosseville, "ALINEA: A Local Feedback Control Law for On-Ramp Metering," *Transportation Research Record 1320*, pp. 58-64.

[17] L. Jacobson, K. Henry, and O. Mehyar, "Real-time Metering Algorithm for Centralized Control," *Transportation Research Record 1232*, Transportation Research Board, National Research Council, Washington, DC, pp. 17-26.

[18]http://www.isi.edu/nsnam/ns/