Scalable Layered Multicast with Explicit Congestion Notification

Li Zhu, Nirwan Ansari
Advanced Networking Lab
New Jersey Institute of Technology
Newark, NJ 07102, USA
{lz6, nirwan.ansari}@njit.edu

Zafer Sahinoglu, Anthony Vetro, Huifang Sun
Mitsubishi Electric Research Laboratories
Murray Hill, NJ 07974, USA
{zafer, avetro, hsun}@merl.com

Abstract

Many Layered Multicast (LM) congestion control schemes have been proposed to serve different users with heterogeneous bandwidth requirements. Most of the LM algorithms suffer from a large IGMP leave latency and a sudden rate increase. Layered Increase/Decrease with Dynamic Layering (FLID-DL) was introduced to address these challenges. Nevertheless, FLID-DL can neither provide very good bandwidth fairness with TCP nor work well in the wireless environment with link loss. We propose the Extended FLID-DL (E-FLID-DL), by incorporating Explicit Congestion Notification (ECN) into FLID-DL, to address these issues. E-FLID-DL not only improves bandwidth fairness with TCP, but also provides high throughput in a lossy network environment.

Keywords: *Layered Multicast, FLID-DL, Explicit Congestion Notification (ECN)*

1. Introduction

IP multicast provides a scalable and efficient way to transmit data to multiple receivers over a large scale network. One of the major challenges faced by IP multicast is the design of suitable congestion control algorithms, which can satisfy the requirement of different users with heterogeneous network bandwidth requirements. Layered Multicast (LM) algorithms [1][2][3][4] have been proposed to address this issue. In LM, data are transmitted in multiple layers in the form of multicast groups. There are two types of LMs: receiver-driven layered multicast and sender-adaptive layered multicast. In both algorithms, receivers continue to estimate the available bandwidth by tracking their packet loss ratios. In receiver-driven layered multicast, the receivers dynamically join and leave corresponding layers based on the estimated bandwidth. If more bandwidth is available, they try to receive extra data by joining additional layers; if congestion happens, receivers leave a certain number of layers to reduce their reception rates. In sender-adaptive approach, the receivers feedback their estimated bandwidth to the sender, and the sender adjusts the sending rate of each layer accordingly so that the overall reception rate at all receivers is optimized. Meanwhile, receivers may also join or leave layers according to their estimated available bandwidths.

In LM, the receivers should make correct leave/join decisions. It is easier for the receivers to leave one layer properly since congestion can be indicated by packet losses. It is more difficult for the receivers to decide if the spare bandwidth is enough for adding a new layer. One solution to this problem is to add a new layer in the active join-experiment [1]. If congestion happens during the join-experiment, the new added layer is dropped immediately; otherwise, the new layer is fully joined. One drawback of this approach is that when join-experiment from one user causes a packet loss, other users may interpret the loss as over-subscription of their current layer, and drop one layer unnecessarily. A more serious problem with this approach is the large IGMP (Internet Group Management Protocol) leave latency. In current IGMP, it is very quick to join one multicast group, but it can take several seconds or even longer time to leave a multicast group. As a result, when congestion happens, the network cannot stop transmitting the layers that receivers want to drop at the bottleneck.

Another difficulty in the development of multicast congestion control algorithms is how to achieve TCP-friendliness since most of the LM schemes use the UDP protocol. TCP-friendly flows can increase their transmission rates if spare bandwidth is available and should decrease their transmission rates when

0-7695-1916-4/03 $17.00

congestion happens. Several works tried to provide TCP-friendly congestion control for Layered Multicast [2][3][4], but they cannot alleviate the large IGMP leave latency problem.

Fair Layered Increase/Decrease with Dynamic Layering (FLID-DL) [5] was proposed to solve the IGMP slow leave problem and to achieve TCP-friendliness at the same time. In FLID-DL, time is divided into slots with length T (500 ms in default). Each multicast group changes the transmission rate in every time slot so that IGMP slow leave difficulty can be avoided. Though FLID-DL flows can share bandwidth with TCP flows rather fairly, it was shown that the average throughput of FLID-DL flows is still significantly less than that of TCP flows under certain network conditions [5].

IP is becoming a promising network layer protocol for wireless networks. It is an open issue on how to provide multicast in wireless networks such as LEO satellite networks. The wireless links can exhibit high bit error rates, which cause high packet losses due to link errors. In TCP-friendly congestion control, packet loss is used as the indicator of network congestion. Therefore, if we adopt these approaches in a wireless network directly, the transmission rate will be decreased significantly as the link loss probability increases. TCP-Peach [6] and TCP-Peach+ [7] addressed this challenge by introducing the use of dummy packets with lower priority, which will be dropped first when congestion happens. RCS [8] extended the idea of TCP-Peach to real time data transmission. In these schemes, when a packet loss is detected, dummy packets are sent to probe the network bandwidth. If a receiver receives dummy packets, it sends ACKs to the sender to indicate that spare bandwidth is available. Upon receiving ACKs for the dummy packets, sender can know whether the packet loss is due to congestion or bad link condition. Nevertheless, all these approaches require frequent feedbacks from receivers and are very difficult to be adopted in the multicast environment.

In this paper, we propose our Extended FLID-DL (E-FLID-DL) scheme. The distinction between E-FLID-DL and other works is two fold. First, we introduce Explicit Congestion Notification (ECN) [9] to improve fairness between FLID-DL traffic and TCP traffic. Second, we use ECN to perform TCP-friendly congestion control for Layered Multicast in wireless environment. To the best of our knowledge, this is the first study on this subject. The remainder of this paper is organized as follows. Section II presents our E-FLID-DL. Section III provides the simulation results. Conclusions are drawn in Section VI.

2. Extended FLID-DL (E-FLID-DL)

We first briefly review the FLID-DL scheme. Readers are referred to [5] for more details. FLID-DL deploys a fast leave scheme instead of relying on the slow IGMP leave. FLID-DL achieves this goal by using a dynamic layering approach, in which the transmitting rate in each layer decreases over time. Thus, the receiver can reduce its reception rate quickly by not joining any additional layer. Suppose there are L layers in an ordinary static layering scheme with rate $r_0, r_1, \ldots, r_{L-1}$ such that $r_i < r_{i+1}$ for $0 \le i < L$. Let D_L denote the leave latency. The time is divided into time intervals with length T (500 ms in default). The corresponding dynamic layer scheme has $L+s$ dynamic layers $d_0, d_1 \ldots d_{L-1} \ldots d_{L+s-1}$, where s is an integer such that $T < D_L < (s-1)T$. For FLID-DL, $r_L = r_{L+1} = \ldots = r_{L+s-1} = 0$. In the dynamic scheme, layer d_j transmits at rate $r_{(L+j-i)\bmod(L+s)}$ during time slot i. In other words, each layer d_j starts transmitting from the highest rate r_{L-1} and drops to the next lower rate in the next time slot till it reaches r_0. Then, layer d_j stop transmitting any data for the next s time slots. After this period of length of $L+s$ time slots, d_j repeats the same transmission scheme again. Hence, the leave delay is no longer than the length of the time interval T. If packet loss is observed in one time slot, the receiver will reduce its reception rate in the next time slot.

ECN was originally proposed for TCP unicast. There are two parameters Min_th and Max_th associated with ECN. If the average queue length at a router is between Min_th and Max_th, the router marks the CP bits in the packets' IP headers. If marked packets are received, the receiver sends feedback to inform the sender about this event, and the sender reduces its window size.

Our extended FLID-DL (E-FLID-DL) requires ECN functionality to be supported by the network. In E-FLID-DL and most of other LM schemes, the type of traffic is usually UDP instead of TCP. How to incorporate ECN with UDP traffic has not been fully studied. The sender behavior of E-FLID-DL is the same as that of FLID-DL except that it sets the ECT bit in the IP header of every packet as the ECN capable TCP sender. The ECN enabled network treats E-FLID-DL packets the same way as ECN capable TCP packets. The E-FLID-DL receivers use the reception of marked packets as the indication of congestion and track the number of marked packets, while the detection of lost packets is used to indicate the congestion in E-FLID-DL. If marked packets are received in the current time slot, each E-FLID-DL receiver drops the subscribed highest layer at the next time slot even if no packet is lost.

In a lossy wireless network, the FLID-DL receivers cannot distinguish between the packet loss due to congestion and the loss due to link errors. If the receiver drops one layer when a packet loss occurs because of link errors, the throughput decreases significantly. On the other hand, the E-FLID-DL receivers can avoid such kind of ambiguity because only the reception of marked packets is used to indicate the congestion. Therefore, the throughput cannot drop too much even when the link loss probability is high.

3. Simulation Results

In this section, we use simulations to demonstrate the advantages of E-FLID-DL over FLID-DL. In all of our simulations, we use the same parameter settings for both FLID-DL and E-FLID. The rate of the base layer is 24 Kbps, and the data are encoded into 20 layers with the cumulative transmission rate increasing multiplicatively by a factor of 1.3 per layer. The packet size is 1000 bytes, the slot time T is 500 ms, and the simulated RTT is 300 ms.

The first experiment is to show that E-FLID-DL can have fairer share of the bandwidth with TCP than FLID-DL in the network without a link loss. The simulation topology is shown in Figure 1. We change *n*, the number of FLID-DL and TCP sessions from 2, 4, …, to 16. The bottleneck bandwidth between router G_1 and router G_2 is set to be $2n$ Mbps accordingly. TCP_1, …, TCP_n are the *n* TCP senders, and $R_1 \ldots R_n$ are the corresponding *n* TCP receivers. This experiment consists of two steps. First, we set $S_1, S_2, \ldots, S_n$ and $H_1, H_2, \ldots, H_n$ as the *n* FLID-DL senders and receivers, respectively, and assume that no ECN functionality is supported by G_1 and G_2. In the second step, we set $S_1, S_2, \ldots, S_n$ and $H_1, H_2, \ldots, H_n$ as the E-FLID-DL senders and receivers respectively; G_1 and G_2 are ECN capable with $Min\_th = 2n$ packets and $Max\_th = 5n$. All simulations are 100 minutes long. All receivers start receiving data randomly in the first 10 seconds, and the throughput of all receivers are computed based on the last 50 seconds of the simulations.

The results are shown in Figure 2. We can see that although FLID-DL flows share relatively fair bandwidth with TCP flows at the bottleneck, TCP flows consume noticeably more bandwidth than FLID-DL flows. If ECN functionality is supported, we can achieve better fairness between E-FLID-DL and TCP.

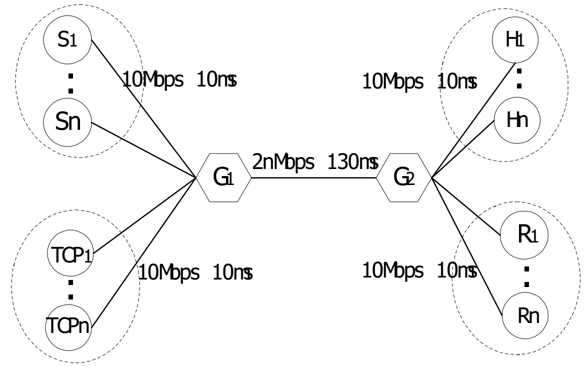

Fig. 1 The network topology for the first experiment

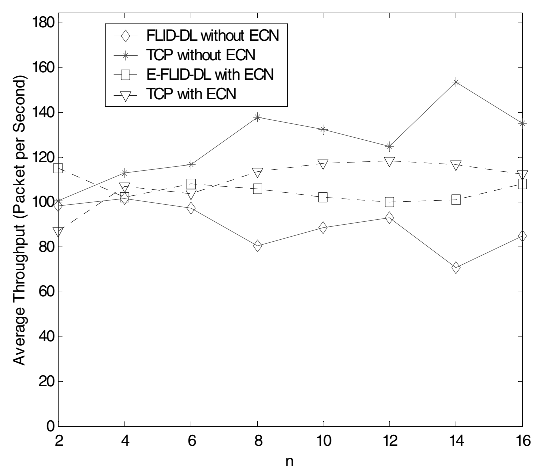

Fig. 2 Throughput comparison between FLID-DL and TCP with and without ECN support

The second experiment is to compare the throughput of E-FLID-DL and FLID-DL in a lossy wireless network. Figure 3 shows the topology of the simulation. S_1, S_2, …, S_{16} are 16 different multicast senders, and $H_1, H_2, \ldots, H_{16}$ are the corresponding receivers. The bottleneck between router G_1 and router G_2 is a lossy wireless link. In the first step, we set all the senders and receivers as ordinary FLID-DL senders and receivers. In the second step, the router G_1 and the router G_2 are ECN enabled with $Min\_th = 32$ packets and $Max\_th = 80$ packets. All the senders and receivers are E-FLID-DL capable. The simulation results are shown in Figure 4. We can see that the reception rate of ordinary FLID-DL decreases significantly as the link loss probability increases. Nevertheless, the reception rate of the E-FLID-DL remains almost constant, regardless of the link loss probability.

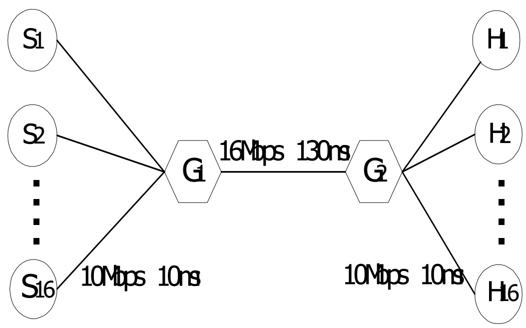

Fig. 3 The network topology used to compare the ordinary FLID-DL and our extended FLID-DL in wireless network

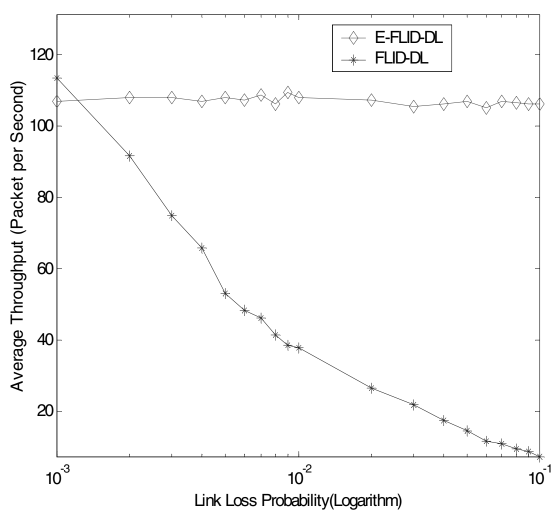

Fig. 4 Throughput comparison between FLID-DL and E-FLID-DL in a lossy wireless network

The simulation topology of the third experiment is shown in Figure 5. In this case, the connections between router G_1 and $H_1, H_2, \ldots, H_{16}$ are lossy wireless links. All other links (including the link between router G_1 and router G_2) have zero link loss. $S_1, S_2, \ldots, S_{16}$ and $H_1, H_2, \ldots, H_{16}$ are E-FLID-DL senders and receivers, respectively. Router G_1 and router G_2 are ECN enabled with $Min\_th = 32$ packets and $Max\_th = 80$ packets. This experiment shows how fair the E-FLID-DL can share the bandwidth with TCP flows at the bottleneck between G_1 and G_2. Designing TCP congestion control in a lossy wireless network is beyond the scope of this paper; therefore, all TCP flows in all the simulations in this paper do not traverse lossy links. The results are shown in Figure 6. We can see that although E-FLID-DL flows traverse lossy links, their throughput stay almost constant regardless of the increase of loss probability. At the same time, they can fairly share the bandwidth with TCP flows at the bottleneck.

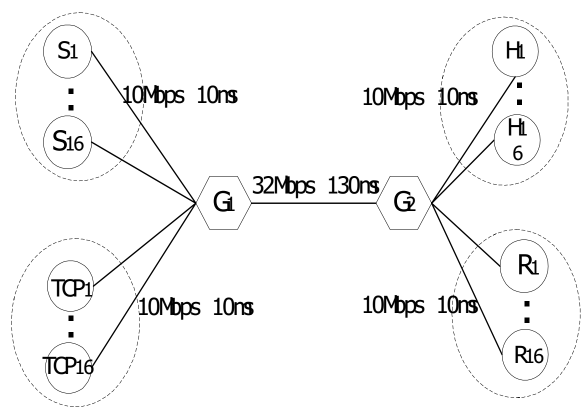

Fig. 5 The network topology of experiment 3

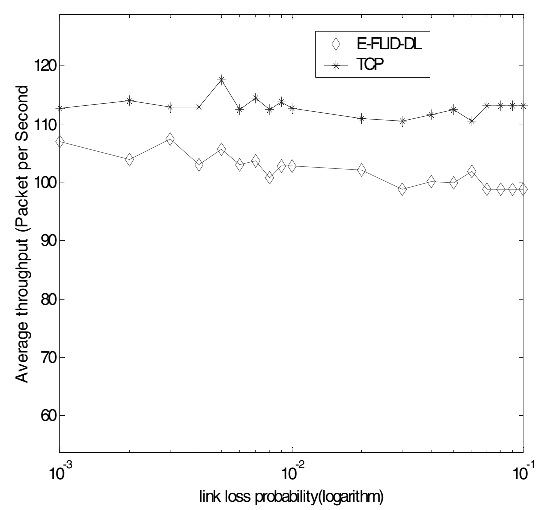

Fig. 6 Fairness performance of E-FLID-DL (in lossy network only) and TCP

In the fourth experiment, we consider the set-up shown in Figure 7. We have 16 E-FLID-DL sessions competing for bandwidth with 16 TCP flows at the bottleneck between router G_1 and router G_2. The

Proceedings of the International Conference on Information Technology: Computers and Communications (ITCC'03)
0-7695-1916-4/03 $17.00 © 2003 IEEE

connections between router G_2 and $H_1, H_2, \ldots, H_8$ are lossy wireless links; all other links have zero link loss probability. This means that among the 16 E-FLID-DL flows, 8 flows traverse wireless lossy links, and other 8 flows and all 16 TCP flows do not. We set $Min\_th = 32$ packets and $Max\_th = 80$ packets. The result is shown in Figure 8. We can notice three significant aspects for the performance of E-FLID-DL. First, E-FLID-DL flows have similar average throughput, regardless of traversing a lossy or lossless link. Second, the average throughput of E-FLID-DL traversing lossy links remain nearly unchanged with the increase in the link loss probability. Third, all E-FLID-DL flows can share the bandwidth with TCP flows at the bottleneck in a fair manner, although their average throughput is a little less than that of TCP flows.

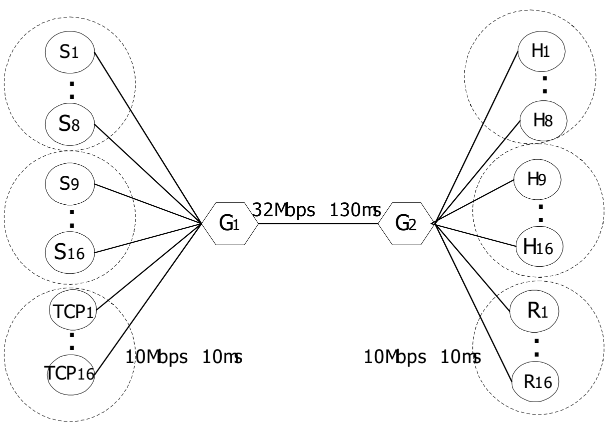

Fig. 7 The network topology of experiment 4

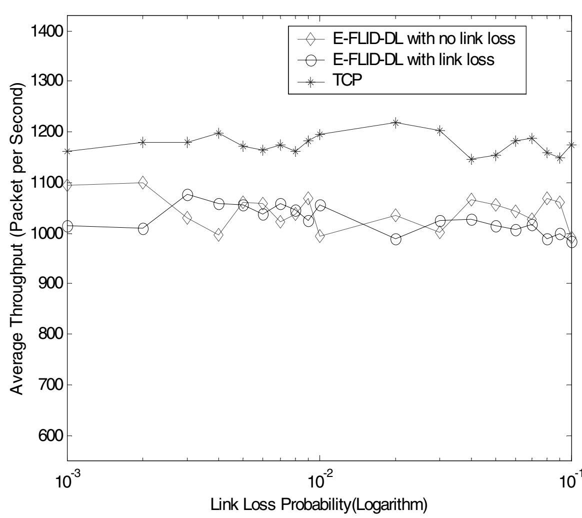

Fig. 8 Fairness performance of E-FLID-DL (with and without link loss) and TCP

4. Conclusions

In this paper, we have proposed Extended FLID-DL by introducing ECN into layered multicast. E-FLID-DL can improve the fairness between the TCP and FLID-DL flows. More importantly, our scheme works very well in the wireless environment. We have used simulations to demonstrate that our algorithm can sustain high throughput even for the case with very high link loss probability. We believe our approach can be deployed not only in all wire line networks, but also in all wireless networks such as LEO satellite networks.

References

[1] S. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," *Proceedings of ACM SIGCOMM'96*, 1996, pp. 117-130.

[2] L.Vicisano and J. Crowcroft, "TCP-like Congestion Control for Layered Multicast Data Transfer," *Proceedings of IEEE INFOCOM'98*, vol. 3, 1998, pp. 996-1003.

[3] B. Vickers, C. Albuquerque, T. Suda, "Source-Adaptive Multilayered Multicast Algorithms for Real-Time Video Distribution," *IEEE/ACM Transactions on Networking*, vol *8, no.6*, pp. 720-733, Dec 2000.

[4] J. Liu, B. Li, Ya-Qin Zhang, "A Hybrid Adaptation Protocol for TCP-Friendly Layered Multicast and Its Optimal Rate Allocation," *Proceedings of IEEE INFOCOM'02*, vol. 3, 2002, pp. 1520-1529.

[5] J. W. Byers, G. Horn, M. Luby, M. Mitzenmacher, and W. Shaver, "FLID-DL: Congestion Control for Layered Multicast," *IEEE Journal on Selected Area in Communications*, vol. 20, no. 8, pp. 1558-1570, Oct. 2002.

[6] I. F. Akyildiz, G. Morabito, and S. Palazzo, "TCP-Peach A New Congestion Control Scheme for IP Networks," *IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 307-321, June 2001.

[7] I. F. Akyildiz, X. Zhang, and J. Fang, "TCP-Peach+ Enhancement of TCP-Peach for Satellite IP Networks, *IEEE Communications Letters*, vol. 6, no. 7, pp. 303-305, July 2002.

[8] J. Tang, G. Morabito, I. F. Akyildiz, and M. Johnson, "RCS: A Rate Control Scheme for Real-Time Traffic in Networks with High Bandwidth-Delay Products and High Bit Error Rates," *Proceedings of IEEE INFOCOM'01*, vol. 1, 2001, pp. 114-122.

[9] K. Ramakrishnan, S. Floyd, and D. Black, "The addition of explicit congestion notification (ECN) to IP," RFC3168, Sep. 2001.

0-7695-1916-4/03 $17.00 © 2003 IEEE