# Generalized Multicast Congestion Control:
## An Efficient Multi-rate Scheme Using Single-rate Control *

Jiang Li, Shivkumar Kalyanaraman

Rensselaer Polytechnic Institute, 110 Eighth Street, Troy, NY 12180
{lij6@cs, shivkuma@ecse}.rpi.edu

**Abstract.** In this paper, we propose a multicast congestion control called GMCC. It provides multi-rate features at low complexity by encompassing a set of *independent* single-rate sub-sessions (a.k.a layers). Various receivers can subscribe to different subsets of these layers to achieve different throughput. The sending rate in each layer is adjusted without boundary by a single-rate multicast congest control algorithm. The set of layers offered to receivers is also dynamically adapted to need. In summary, GMCC is *fully adaptive*.

**Key words:** Multicast, congestion control, multi-rate,single-rate

## 1 Introduction

In multicast [5], the congestion control issue is complicated because we need to consider the congestion on a tree instead of that along a path. Intensive research has been conducted in this area, and researchers have proposed two categories of multicast congestion control protocols: single-rate and multi-rate.

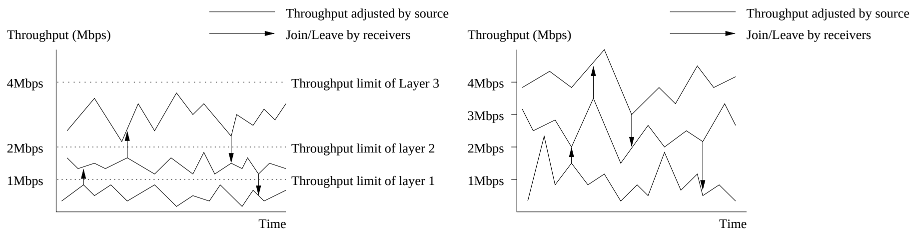
In single-rate protocols such as ORMCC [12], PGMCC [19] and TFMCC [21], the source sends data to all receivers at a dynamically adjusted rate. The rate has to be adapted to the slowest receiver to avoid consistent congestion at any part of the multicast tree. Therefore, faster receivers suffer. Still, single-rate protocols have advantages because they are simple.

In a multi-rate multicast session, there are a set of sub-sessions, also called layers. Each layer has multicast traffic and uses a separate multicast group address. Examples are RLM [10] PLM [11], RLC [20], FLID-DL [1], FLGM [3], STAIR [2] and WEBRC [15]. Recipients have to increase or decrease their receiving rates by joining or leaving some layers.[1] Since the sending rates of these layers are not adapted to network status, the receivers have to perform join/leave operations very frequently to adapt their throughput to real time congestion. However, according to IGMP [8], join and leave operations (especially leave) need time to take effect, leading to coarse control. Moreover, a large volume of control traffic is introduced into the network, and the routers are heavily loaded because all the rate control burden has been shifted to them. These schemes are also called *receiver-driven* schemes.

* This work was supported in part by a grant from Intel Corporation.

[1] Joining a layer is also called subscription, leaving a layer is also called unsubscription. In this paper we will use both sets of terms exchangeably.

A recently proposed scheme SMCC [9] is a hybrid of single-rate and multi-rate multicast congestion control. It combines a single-rate scheme TFMCC[21] with the receiver-driven idea. In each layer, the source adjusts sending rate *within a certain limit* based on TFMCC, and receivers join or leave layers cumulatively according to their estimated maximum receiving rates using TCP throughput formula [18]. Since the flows in each layer are adaptive to network status, the number of join and leave operations are greatly reduced. The congestion control is more effective. However, since SMCC requires static configuration of the maximum sending rates for each layer, it may require more layers than necessary for a receiver to achieve desire throughput, or fail to differentiate receivers desiring for different throughput.




(a) SMCC overview
(with per-layer throughput limit)

(b) GMCC overview
(no per-layer throughput limit)

**Fig. 1.** Qualitative Comparison of SMCC and GMCC

Our proposed scheme GMCC solves these problems while having the merits of SMCC. Figure 1 shows the difference between SMCC and GMCC visually. In general, GMCC has the following advantages:

(1) It is *fully* adaptive. The sending rate in each layer can be adjusted without rigid limits. Together with the automatically adjusted number of layers, it always allows heterogeneous receivers to receive at different rates.
(2) The number of layers used is just enough to accommodate the differences among the throughput desired by receivers. No redundant layers are used.
(3) The source can control the overall throughput of a multicast session by limiting the number of layers to be used. In particular, if only one layer is allowed, GMCC works as a single-rate multicast congestion control scheme, which is the reason it is so named.
(4) It is not coupled with equation-based rate control mechanism such as TFMCC. The rate control mechanism at source can be replaced by others based on representative (the most congested receiver).

GMCC also addresses the issue of starting and stopping traffic within layers depending on whether there are receivers in the layers.

In the rest of this paper, we will describe the details of GMCC, and show some simulation results to demonstrate the performance of GMCC.

## 2 Generalized Multicast Congestion Control

The two key issues of GMCC are (1) How the source controls the throughput in each layer, (2) How and when a receiver join or leave layers to adjust its total throughput. The basic ideas of solutions are the following:

- In each layer, the source chooses a most congested receiver as *congestion representative* (CR) and adjusts the sending rate of this layer according to the CR's feedback (Section 2.2).
- The source starts traffic in a layer when the first receiver joins and stops traffic in a layer when the last receiver leaves (Section 2.3).
- Each receiver joins layers cumulatively, and is allowed to be the CR of at most one layer.
- When a receiver detects that it is much less congested than the most congested receiver (i.e. the CR) in the highest layer it has joined, meaning it can potentially receive at a higher rate, it joins an additional layer *successively* (Section 2.4).
- When a receiver detects that it is the most congested receiver in more than one layer, which means it confines or can potentially confine the sending rates of more than one layer, it leaves the highest joined layer (Section 2.5).
- Receivers make decisions of join and leave based on statistics. Statistics can be used only if (1) At least a certain number of samples have been collected, and (2) Every layer has a CR.

As shown in the above ideas, it is important for a receiver to detect whether it is more congested than another. We propose to use *Throughput Attenuation Factor* (TAF) for this purpose described in next section. After that, we describe various aspects of the GMCC scheme.

### 2.1 Throughput Attenuation Factor

*Throughput Attenuation Factor* (TAF) [2] is a metric *measured at the receiver side* to indicate how congested the receiver is. It comprises two parts, *Individual Throughput Attenuation Factor* (ITAF) and *Congestion Occurrence Rate* (COR), each describing a different aspect of congestion.

**Individual Throughput Attenuation Factor** ITAF is defined as

$$1 - \frac{\mu}{\lambda}$$

measured only in congestion epochs (A congestion epoch is an event when one or more consecutive packets are lost.[3]) $\mu$ is the instantaneous output rate and $\lambda$ is

[2] This section is a self-contained overview of our technical report [13] which covers more details.

[3] We assume that packet loss is due to congestion only.

the rate of input generating this portion of output. It shows how much proportion of input is lost during an instance of congestion, and therefore indicates *how serious* this instance of congestion is.

ITAF may be measured in the following way in implementation: Each data packet carries the instantaneous sending rate information, assumed to be $\lambda_n$ for the packet of sequence number $n$. When a packet of sequence number $n$ arrives, the receiver divides this packet size by the latest packet arriving interval and gets the instantaneous receiving rate $\mu_n$. If the receipt of sequence number $n$ indicates a packet loss, a ITAF is obtained as $1 - \frac{\mu_m}{\lambda_m}$ where $m$ is the received sequence number immediate prior to $n$.

**Congestion Occurrence Rate** COR is defined as the reciprocal of the interval between two consecutive congestion epochs. For instance, if the loss of packet $n$ and $n+i$ ($i > 1$) is detected at time $t_1$ and $t_2$ respectively (with the packets from $n+1$ to $n+i-1$ received), then a sample of COR would be $\frac{1}{t_2-t_1}$. COR shows how frequently congestion happens.

With ITAF and COR defined, TAF is the product of these two factors, i.e.

$$TAF = ITAF \times COR$$

The larger TAF, the more congested is a receiver. To avoid unnecessary oscillation, the average values over a certain number (30 in our simulations) of ITAF and COR samples are used for TAF calculation. In GMCC, each receiver measures its own TAF *for each joined layer* and maintains the mean $\Theta$ and standard deviation $\Theta^\sigma$ of the latest $N$ TAF samples for the purpose of TAF comparison.

**Table 1.** Some Key Symbols In Section 2

| Symbol | Meaning |
|---|---|
| $\Theta_i$ | Average TAF of receiver $i$ |
| $\Theta_i^\sigma$ | Standard deviation of receiver $i$'s TAF |
| $\theta$ | Average ITAF of a receiver's highest joined layer measured during periods without bandwidth shifting |
| $\theta'$ | Average ITAF of a receiver's highest joined layer measured during band-width shifting periods |
| $N$ | Number of TAF/ITAF samples kept for calculation |
| $J$ | Number of positive TAF/ITAF comparison results required to join an ad-ditional layer |

### 2.2 Sending Rate Control Within A Layer

Given a layer with active receivers, the source chooses a most congested receiver in this layer as congestion representative (CR) and uses its feedback for rate adaptation. [4] When the CR detects packet loss, it sends feedback packets called *congestion indications* (CIs) back to the source that decreases the sending rate by half. To avoid reducing rate too much, the source decreases the sending rate at most once per SRTT (smoothed RTT). The samples of RTT are collected by the source at the receipt of CIs. The value of a sample is the time difference between the CI arrival and the departure of the data packet triggering the CI. SRTT is calculated by exponential weighted moving average formula: SRTT = (1 - $\varepsilon$) SRTT + $\varepsilon$ RTT ($0 < \varepsilon < 1$, we use 0.125). At the absence of CIs, the sending rate is increased by $s/SRTT$ each SRTT, where $s$ is the packet size.

To update CR, the source checks the following condition based on statistical inference [16], and switch CR from receiver $j$ from $i$ if the condition is true.

$$\Theta_i > \alpha_1 \Theta_j + \alpha_2 \sqrt{\frac{{\Theta_i^{\sigma}}^2 + (\alpha_1 \Theta_j^{\sigma})^2}{N}} \tag{1}$$

The symbol meanings are shown in Table 1, and $\alpha_1$, $\alpha_2$ are configurable parameters. We set $\alpha_1$ as 1.25 since we want to bias toward the current choice of CR to avoid unnecessary oscillation, $\alpha_2$ as 1.64 for a 90% confidence level.

On the receiver side, CIs are suppressed if receivers find out that condition (1) is true. The information of CR is broadcast to all receivers for the calculation.

Notice that unlike SMCC, the sending rate in each layer can be adjusted to any level required for adaptation. Besides, other rate control mechanisms such as those in PGMCC [19] and TFMCC [21] can be used in place of the current one, as long as the transmission rate is controlled by the source based on the feedback packets from the most congested receiver.

### 2.3 On-and-Off Control of Layers (by Source)

In any GMCC session, there is always a basic layer in which the source keeps sending packets subject to rate control. All other layers must be turned on (i.e. start traffic) or turned off (i.e. stop traffic) at right time to avoid bandwidth waste. When a receiver joins a layer which did not have any receiver yet, the source needs to start sending packets in this layer, i.e. *turn on* this layer. In an active layer, the CR keeps sending heartbeat packets to the source. If the source has not received heartbeat packets from the CR for a certain period, and no new CR is chosen, the source stops sending data in this layer, i.e. *turn off* this layer. For more details, please refer to our technical report [14].

### 2.4 Joining An Additional Layer (by Receiver)

Whenever a receiver enters a GMCC session, it subscribes to the basic layer of GMCC and stays there till it quits the session. Beyond this basic layer, the receiver must perform join operations to increase its total throughput rate at right time. A receiver joins an additional layer *successively* when it detects that its throughput rate can be potentially increased. There are three types of join:

[4] The concept of CR here is similar to the representative receiver in DeLucia & Obraczka's work [6] and TFMCC [21].

**Type 1: Join Operations Triggered by Normal TAF** This is the common type of join. Assume we observer receiver $i$, and the CR is receiver $j$. Receiver $i$ measures TAF for the *the highest layer* it is in as we defined before. Once there are at least $N$ TAF samples of this layer, it check the following condition:

$$\Theta_j > \beta_1 \Theta_i + \beta_2 \sqrt{\frac{(\beta_1 \Theta_i^{\sigma})^2 + \Theta_j^{\sigma\,2}}{N}} \tag{2}$$

$\beta_1$ and $\beta_2$ are parameters. We are conservative about join, and heuristically choose $\beta_1 = 2$, $\beta_2 = 2.58$ for a 99% confidence level. If the condition in (2) is true for $J$ $(= 30)$ consecutive times, the receiver will join an additional layer. The reason to use relatively small $N$ for samples and $J$ for TAF comparison results, instead of to use a single large $N$ for samples, is that calculating the mean and deviation of a large set of samples is expensive. Meanwhile, this method can catch the dynamics of networks.

Although the TAF comparison in other layers can also stimulate the receiver to join more layers, restricting it in the highest joined layer has equivalent effect and simplifies the design.

**Type 2: Join Operations Triggered by Hypothetical TAF** Sometimes the link between the source and a receiver is underutilized and the receiver does not have any TAF samples. In this case, we use *hypothetical* TAF. When a non-CR receiver notices that the CR of its highest joined layer has updated TAF statistics, this receiver gets a *hypothetical* COR sample by assuming congestion at this moment. Using this COR sample together with *unchanged* ITAF (1 for no loss), it calculates a *hypothetical* TAF and check the condition in (2). Once there are $J$ *consecutive* positive results, a join operation is triggered.

Note that the test version of COR and TAF are not accepted as permanent samples since they are not true samples. Once used, they are discarded. Consequently, the join operations of type 1 are not interfered.

**Type 3: Join Operations Triggered by Probabilistic Inter-layer Bandwidth Shifting** Under some special cases, there can be no type 1 and 2 join but join operations are still plausible. Consider a topology in Figure 2 containing two bottlenecks. Assume $R1$ is in one layer, and $R2$ is in two layers. When $R3$ enters the session, since the bandwidth in Bottleneck 1 has already be fully utilized, it will stay in only one layer. The reason is that the congestion generated by *intra-session flows* of other layers is not distinguished from that by *inter-session flows*, whereas the congestion of the former kind can actually be ignored in the context of deciding whether to join. This problem also occurs in SMCC, but the paper [9] did not consider it.

A solution can be that, for the above example, sometimes we try to send a little more in the first layer, while sending less in the second layer. If R3 does not see any increased congestion, it will know that a portion of the congestion is incurred by intra-session flows, therefore can join the second layer.

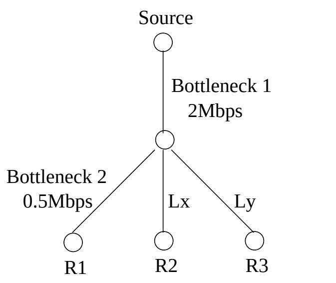



**Fig. 2.** A Topology Example Where Probabilistic Inter-layer Bandwidth Shifting Is Needed

Certainly the above method should be carefully managed because sending more in a layer might cause more severe congestion on some paths. We developed the following technique called *probabilistic inter-layer bandwidth shifting* (PIBS). Assume multiple layers (layer 1 to $n$, $n > 1$) are used in a multicast session. Let the period between two consecutive rate reductions (in the same layer) be a *rate control period* (RCP). At the beginning of each RCP at layer $i$ ($1 \leq i < n$), with probably $\rho$, the source decides that it will send data at the ($1+\delta$) level (otherwise send normally). That is, at any moment during this whole RCP, if the calculated sending rate is $\lambda_i$, the source will actually send packets at the rate of $\lambda_i + \min(\delta\lambda_i, \lambda_{i+1})$. At the same time, at layer $i+1$, the actual sending rate is adjusted to $\max(0, \lambda_{i+1} - \delta\lambda_i)$. Briefly, the source "shifts" some bandwidth from layer $i+1$ to layer $i$. To avoid significant unfairness to non-GMCC flows, $\rho$ and $\delta$ must be small (both are 0.1 in our simulations). Also, at any moment, no two layers are allowed to perform bandwidth shifting simultaneously.

Given a receiver $R$ and a multicast session the receiver is in, assume $\ell$ layers go through the bottleneck on the path between the source and $R$. If $\ell > 1$, for any layer $i < \ell$, according to the definition of ITAF (Section 2.1), the average ITAF measured by $R$ at layer $i$ during bandwidth shifting periods ($\theta'$) should be approximately the same as that measured during periods without bandwidth shifting ($\theta$). On the contrary, if $\theta'$ is larger than $\theta$, it means shifting bandwidth to layer $i$ cause more congestion, indicating that no layer above $i$ goes through the same bottleneck.

Assume $R$'s highest joined layer is $k$, and the highest layer with traffic for the whole multicast session is $L$. If $k < L$, $R$ will check the following condition at layer $k$ once it has at least $N$ samples for both $\theta$ and $\theta'$.

$$\theta - \gamma\sqrt{\frac{\sigma^2 + \sigma'^2}{N}} \leq \theta' \leq \theta + \gamma\sqrt{\frac{\sigma^2 + \sigma'^2}{N}} \qquad (3)$$

$\sigma$ and $\sigma'$ are the standard deviations corresponding to $\theta$ and $\theta'$ respectively. If condition (3) is true, the receiver $R$ will join layer $k+1$. It should be noticed that although ITAF samples are distinguished under this situation, they are treated as the same for TAF calculation under situation 1 and 2.

**Two Exceptional Cases** To prevent spurious join, there are two exceptional cases to be checked before the join operation really occurs.

(1) If any layer does not have a CR yet, meaning the session has not reached its sending rate limit, the join attempt should be canceled.
(2) If a receiver is already a CR for some layer, or detects that it may become a CR in any of its joined layers, meaning it will unnecessary confine the

sending rates of other layers if joined, it also refrains itself from join. The detection is done by checking the following condition, assuming this receiver is $i$ and the CR is $j$,

$$\Theta_i > \Theta_j + \omega\sqrt{\frac{{\Theta_i^\sigma}^2 + {\Theta_j^\sigma}^2}{N}} \tag{4}$$

$\omega$ decides confidence level, and we used 3.5 for 99.99% level.

It is worth mentioning that we do not have "join attempt" as SMCC does. We believe that in GMCC, since both the sending rates in each layer and the number of layers can be dynamically adjusted, as a multicast session goes on, the combination of sending rate settings and the choice of layer number will evolve to the extent that will accommodate the heterogeneity among the receivers, so that a join won't cause abrupt severe congestion. Moreover, omitting join attempts significantly simplifies the design.

### 2.5 Leaving a Layer

A receiver always unsubscribes from the highest joined layer. After a receiver joins a layer, it needs to wait for some time to allow the network stabilize. This is achieved by collecting $N$ more samples for TAF statistics in *all* joined layers before it checks whether to leave. Then, if the receiver is the CR or satisfies the condition in (4) in more than one layer, it leaves the highest layer it is in. The reason is the same as explained in the second exceptional case of join at the end of Section 2.4.

## 3 Simulations

We have run several $ns$-2 [17] simulations to test different aspects of GMCC. In these simulations, drop-tail routers are used, router buffer size is set to 20K bytes, and Reno TCP is used for background traffic.

### 3.1 Effectiveness of the adaptive layering

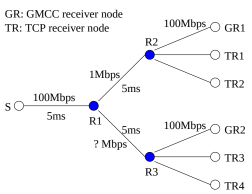
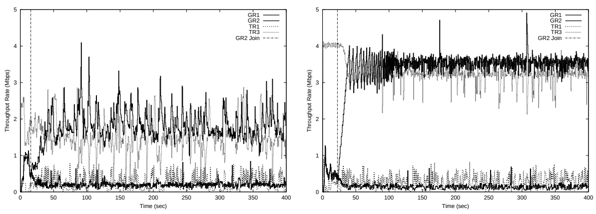



**Fig. 3.** Topology for layering effectiveness test



(a) Throughput when (R1,R3) is 5Mbps

(b) Throughput when (R1,R3) is 10Mbps

**Fig. 4.** Effective Layering Test Result

GMCC does not require redundant layers to satisfy heterogeneous receivers, as shown in this simulation. In Figure 3, four TCP flows go from node S to

TR1∼4 respectively, and a GMCC session originates at S and ends at GR1, GR2. The bandwidth of the link between R1 and R3 is set to 5Mbps in the first simulation, and 10Mbps in the second. Ideally, two layers as the minimum are expected for both simulations, with GR1 in one layer and GR2 in two.

The throughput of the flows in these two simulations are shown in Figure 4. GR2 joined an additional layer at 15.8-th second and at 22.4-th second in the first and second simulations respectively, and stayed in two layers till the end of simulations. In contrast, GR1 only subscribed to the basic layer. This conforms to the expectation above and shows that the GMCC does not use more layers than necessary.

### 3.2 Responsiveness to traffic dynamics

There are two types of response to traffic dynamics. The first type of response is by the source that adjusts sending rates within layers. GMCC's rate adaption by source is almost the same as that in our single-rate work ORMCC [12]. Therefore, we omit the examination of source response to traffic dynamics here, and refer readers to [12]. The second type of response is by receivers by means of joining and leaving layers. It can be considered as a complementary measure of the first type response limited by CRs.

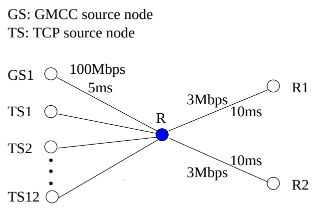



**Fig. 5.** Star topology for testing responsiveness to traffic dynamics

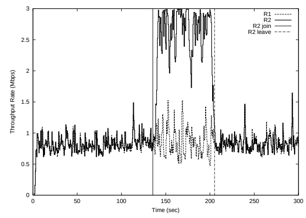



**Fig. 6.** Test Results of Responding to Traffic Dynamics

We used the star topology in Figure 5 to test the receivers' responsiveness to the dynamics of crossing traffic on the bottleneck. A GMCC session has GS1 as the source node and R1, R2 as the receiver nodes. On each of the links of (R,R1) and (R,R2), there are six TCP competing flows at the beginning of the simulation. During the period between 100-th and 200-th second, five TCP flows on the link (R,R2) pause, leaving one TCP flow as the only competing flow.

As shown in Figure 6, receiver R2 joined an additional layer at 135.412-th second. After those five TCP flows pause, the link (R,R2) became much less congested than (R,R1). Therefore, this join operation is appropriate. There is 35-second gap between the pause and the join operation, though. That is relatively long because GMCC is conservative about join and therefore requires enough number of samples and positive TAF comparison results (see Section 2.4). However, GMCC is quicker when making decisions about unsubscription. In this simulation, R2 left the layer at 205.178-th second. On the other hand, since there is no traffic dynamics on the link (R,R1), receiver R1 remains in one single layer.

### 3.3 Effectiveness of probabilistic inter-layer bandwidth shifting

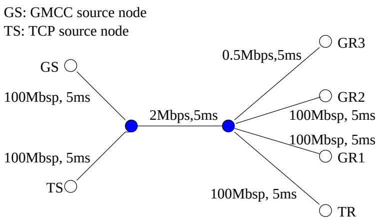



**Fig. 7.** Topology for testing prob. inter-layer bandwidth shifting

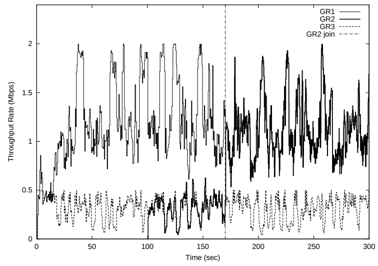



**Fig. 8.** Test Results of PIBS: Throughput of All GMCC Receivers

To verify that PIBS (Section 2.4) is a valid technique, we ran a simulation on the topology in Figure 7. A TCP flow originates at TS and ends at TR as background traffic. The GMCC flows in a multicast session go from GS to GR1,2,3. The 2Mbps bottleneck is shared by all three GMCC receivers, and the 0.5Mbps bottleneck only affects GR3. At the beginning of the simulation, only GR1 and GR3 are in the session. At 100th second, GR2 enters the session. Figure 8 shows that in one simulation instance, GR2 subscribed to an additional layer at 170.146-th second based on bandwidth shifting. Again, there is long delay because GMCC receivers need to collect enough samples before making decisions.

We noticed that in some other instances of this simulation, a join operation for another reason (in particular, under situation 2 in Section 2.4) happened before the results of bandwidth shifting took effect, and the joini operations triggered by bandwidth shifting were suppressed. This is not unexpected because the flows are dynamic and the comparisons in GMCC are all probabilistic. It is possible that during some random periods the condition in situation 2 becomes true and triggers a join operation.

### 3.4 Throughput Improvement

The topology in Figure 9 contains six bottlenecks and is used to test how GMCC improves the throughput of heterogeneous receivers with relatively slight difference of expected throughput. All the links are of 5ms delay. The bandwidths of the bottlenecks are from 1Mbps to 6Mbps. On each of them, there are two TCP flows as competing traffic. A GMCC session is held between the source GS and six receivers (GR1 to GR6). Simulation time is 600 seconds.

Figure 10 shows the over time average throughput rate of all receivers. Over time average throughput rate at time $t$ is defined as the total throughput through time $t$ divided by the total run time. We can see that the six GMCC receivers do achieve different throughput rates, with GR6 being the highest and GR1 being the lowest. Six layers are used in this simulation.

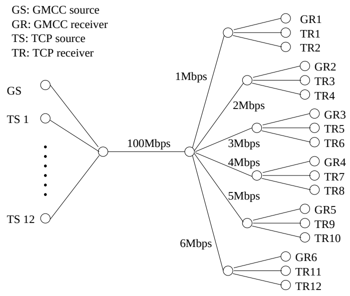
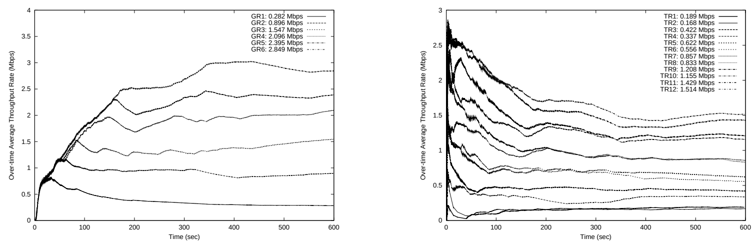



**Fig. 9.** Topology for testing throughput improvement

We need to mention that in this simulation, GMCC receivers achieve higher throughput than TCP correspondents. The reason is that each flow in a GMCC layer is a single-rate congestion control flow independent of other flows. It competes for bandwidth like any other flow does. For example, when GR2 subscribes to two layers, there are then two TCP flows and two GMCC flows on the 2Mbps bottleneck. The throughput of GR2 is the sum of both GMCC flows, and therefore can be approximately twice as much as each of the TCP flows. However, due to the limit by CRs in lower layers, assuming there are $n$ TCP flows and $m$ GMCC flows on a bottleneck, a receiver may not get the share of $m/(m+n)$. GR6 here is an example. Although what we observed for GMCC in this simulation is different from traditional TCP-friendliness concept, we don't consider it as a serious problem, because each GMCC flow within a layer still competes in a TCP-friendly manner. This is more or less the same as people open multiple TCP connections to transmit a single object over the Internet. Moreover, using independent GMCC flows of this kind greatly simplifies the task to achieve multi-rate for multicast. Still, we will explore this issue more carefully in the future.




(a) Over-time Avg. Throughput Rate of GMCC receiver's

(b) Over-time Avg. Throughput Rate of TCP receivers

**Fig. 10.** Throughput Improvement Test Result

# 4 Conclusion and Future Work

We have presented a multi-rate multicast congestion scheme called GMCC. By combining single-rate congestion control and traditional multi-rate techniques (mostly joining and leaving layers by receivers) in a novel way, it provides a simple design for a perplexing problem of which most previous solutions are complicated. While having the merits of a similar previous scheme SMCC [9], it is *fully* adaptive and surmounts the limits posed by SMCC's static configurations. A new technique called *probabilistic inter-layer bandwidth shifting* is proposed as the solution to a problem not addressed by SMCC. Besides, the rate control mechanism at source can be replaced by other representative-based mechanisms.

This paper includes the first step study of GMCC. In the paper, we have seen several parameters needed at receiver side in GMCC. To study how receivers' uniformly or differently changing these parameters affects GMCC performance is important. We would also like to conduct simulations in more complex topologies and with different types of buffer management (e.g. RED) on routers as well as with different flavors of TCP. We are now in the process of developing large scale simulations (with several thousand receivers) for GMCC on the simulator of ROSS [4].

## References


1. J. Byers, et al, "FLID-DL Congestion Control for Layered Multicast," *NGC 2000.*
2. J. Byers, G. Kwon, "STAIR: Practical AIMD Multirate Multicast Congestion Control", *NGC 2001.*
3. J. Byers, et al, "Fine-Grained Layered Multicast", *Infocom 2001*
4. Christopher D. Carothers, David Bauer, Shawn Pearc, "ROSS: A High-Performance, Low Memory, Modular Time Warp System", *14th Workshop on Parallel and Distributed Simulation (PADS 2000)*, May 2000
5. S. Deering, "Host Extensions for IP Multicasting", *RFC 1112*, August 1989
6. Dante DeLucia, Katia Obraczka, "A Multicast Congestion Control Mechanism Using Representatives", *Proceedings of the IEEE ISCC 1998*
7. Kevin Fall, Sally Floyd, "Simulation-based Comparisons of Tahoe, Reno, and SACK TCP", *Computer Comm. Review*, Vol. 26, No. 3, Jul. 1996, pp. 5-21
8. W. Fenner, "RFC 2236: Internet Group Management Protocol, Version 2", *IETF*
9. Gu-In Kwon, John Byers, "Smooth Multirate Multicast Congestion Control", *IEEE INFOCOM '03*, April 2003
10. S. McCanne, et al, "Receiver-driven Layered Multicast," *SIGCOMM '96*, Aug '96
11. A. Legout, E. Biersack, "PLM: Fast Convergence for Cumulative Layered Multicast Transmission Schemes", *Proc. of ACM SIGMETRICS*, 2000
12. Jiang Li, Shivkumar Kalyanaraman, "ORMCC : A Simple And Effective Single-Rate Multicast Congestion Control Scheme", *submitted work*, 2002
13. J. Li, et al, "Using Average Attenuation Factor to Locate the Most Congested Path for Multicast Congestion Control", *Technical Report, CS, RPI*, 2003, available at http://www.cs.rpi.edu/~lij6/Research/my_papers/ormcc-tr.ps.gz
14. J. Li, et al, "Generalized Multicast Congestion Control: An Efficient Multi-rate Scheme Using Single-rate Control", *Technical Report, CS, RPI*, 2003, available at http://www.cs.rpi.edu/~lij6/Research/my_papers/gmcc-tr.ps
15. Michael Luby, Vivek K. Goyal, Simon Skaria, Gavin B. Horn, "Wave and Equation Based Rate Control Using Multicast Round Trip Time," *SIGCOMM*, 2002.
16. William Mendenhall, "Introduction to Probability and Statistics: Third Edition", *Duxbury Press*, 1997
17. The Network Simulator - ns-2, available at http://www.isi.edu/nsnam/ns
18. Jitendra Padhye, et al, "Modeling TCP Throughput: A Simple Model and its Empirical Validation," *SIGCOMM 1998*, Aug. 1998.
19. L. Rizzo, "PGMCC: A TCP-friendly Single-Rate Multicast Congestion Control Scheme", *SIGCOMM '00*, Aug '00.
20. L. Vicisano, L. Rizzo and J. Crowcroft, "TCP-like congestion control for layered multicast data transfer," *INFOCOM*, Apr '98.
21. Jorg Widmer, Mark Handley, "Extending Equation-based Congestion Control to Multicast Applications", *SIGCOMM 2001*, Aug. 2001.