Optimization - based Congestion Control for Internet Multicast Communications

Ms. Nguyen Thi Thu Hang* and Mr. Taipio Erke **
Faculty of Telecommunications 1, Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
Tel : +84-34-515 484 Fax : +84-34-511 405 E-mail: thuhangbc@yahoo.com
**Telecommunications Program, School of Advanced Technologies, Asian Institute of Technologies, Thailand
Tel : +66 (0)2 524 5737 Fax : +66 (0)2 524 5737 E-mail: erke@ait.ac.th

This paper presents a combination of optimization concept and congestion control for multicast communications to bring best benefit for the network. For different types of Internet services, there will be different utility functions and so there will be different ways to choose on how to control the congestion, especially for real time multicast traffic. Our proposed algorithm OMCC brings the first implementation experiment of utility-based Multicast Congestion Control. Simulation results show that OMCC brings better network performances in multicast session throughput while it still keeps a certain fairness of unicast and multicast sessions, and thus, provides better benefit for all network participants.

Congestion control, multicast, optimization-based, utility function, fairness, internet communications

1. INTRODUCTION

The internet has witnessed an incredible growth that has in turn encouraged the enrichment of new techniques and services from dual sides of the network and the users. Together with the advent of broadband, wireless, and Web technologies, it is becoming simply viable to implement large-scale heterogeneous networks that can easily support content distribution, video broadcasting, distributed data bases, distant education, and teleconferencing. As part of the trend, multicasting has become an enabling technology because it saves time and bandwidth of the resources in the network [1].

Despite the fact that multicast brings the benefit to the network, especially to the service providers, it seems not enable the network operators to implement multicast for the reason of traffic congestion [2]. Besides, there is still existence of unicast that shares network resource with multicast [3].

The basic conflict in that case of congestion is this: It is desirable to increase the use of multicast where appropriate to reduce the overall bandwidth demand of applications that transmit high bandwidth data to many receivers, but the introduction of multicast sessions into the network must not deteriorate the performance of existing unicast traffic [3, 4].

The remainder of this paper is organized as follows. In part 2, two types of utility function are described. Multicast congestion control is presented in part 3 together with a review of TCP-friendly Multicast congestion control (TFMCC) which based on rate control mechanism. Part 4 describes the new algorithm combining utility concept and TFMCC named OMCC. Simulation configurations and results of OMCC in the internet communication are analyzed and summarized in part 5. Part 6 ends this paper with conclusions.

2. UTILITY FUNCTIONS

Utility function describes of what utility or usefulness a particular transmission rate is to the user. Utility function is a continuously differentiable, strictly concave, increasing function in the interval $(0, \infty)$ and it is assumed that utility function $U_r(x_r)$ is unbounded as $x_r \to 0$ [5].

2.1. Characteristics

Consider a network modeled as a set of directed links with fixed but not, generally, equal capacities. The work load for the network is generated by a set of sessions. A session is described by the subset of network links over which it transmits data and by a variable transmission rate, denoted x. The aggregate data rate on any link, then, is the sum of rates for all sessions using that link [2].

- Utility function U(x) is to characterize a session; U(x) presents the value of bandwidth to the session
- Two important properties of utility functions that capture natural intuitions about session behavior
  - Utility is an increasing function of transmission rate because of the assumption that each session would prefer as high a rate as possible.
  - The utility function has decreasing marginal returns, which models the idea that the value of a small increase in transmission rate is high for a session currently transmitting at a low rate, but decreases as the session's rate increases.
- Two widely used utility functions in congestion control models [3]:
  - Logarithmic utility $$U(x) = \log x \quad (1)$$
  - TCP(Transmission Control Protocol) utility [5]: $$U(x) = -\frac{1}{x} \quad (2)$$

Figure 1 shows two utility functions for quite different applications: Voice over IP and File Transfer Protocol.

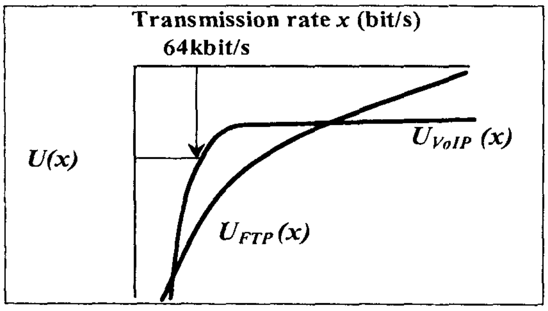

Fig. 1: Utility functions [5].

- The utility function for a voice over IP application is described by curve $U_{VoIP}(x)$. For uncompressed voice, acceptable voice quality requires a transmission rate of about 64 kbit/s to transmit an 8 kHz 8 bit signal. It is possible to cope with a little less quality than this, but as Figure 1 shows:
  - The utility of transmitting voice at less than 64 kbit/s reduces sharply as the transmission rate is decreased.
  - The quality of voice does not increase much if the transmission rate is increased beyond 64 kbit/s, so the utility function remains almost flat above 64 kbit/s, indicating that only little utility is gained by transmitting faster than 64 kbit/s.

2.2. Sender- and Receiver-Oriented Utility Functions

The concepts of sender- and receiver-oriented utility functions have been suggested in [3]. In the paper, a single-rate multicast session s with rate x and receiver set $\Re$ with size R has been considered.

In the sender-oriented approach, session utility function is a concave increasing function us of the session rate:

$$U_{snd} = u_s(x) \tag{3}$$

In the receiver-oriented approach, each receiver $i \in \Re$ has a utility function $u_i(x)$, which is concave and increasing. The session utility function is the sum of receiver utilities.

$$U_{rcv} = \sum_{i \in \Re} u_i(x) \tag{4}$$

Then, conversion of the definitions can be implemented by introducing two requirements.

- The first requirement is that all receivers in a session have identical utility functions.

  $$u_i(x) = u_r(s) \quad \forall i \in \Re \tag{5}$$

  The utility functions are typically thought as representing application characteristics and sometimes as being imposed by network mechanisms.

  Take $u(x) = -\frac{1}{x}$ to model TCP-style congestion as in [6]. To the extent that receivers within a session share the same application requirements, it is also reasonable to assume they share a utility function. So, this is natural assumption in the case of single-rate multicast.
- For sender-oriented equation, the number of receivers is ignored by treating all sessions equivalently. The receiver-oriented one reflects the idea that multicast session utility is itself a social welfare function [3], representing the aggregate utility of the receiver set.

3. MULTICAST CONGESTION CONTROL

3.1. Congestion Control in the Internet

The increasing number of group communication applications such as teleconference and distributed content services has led to a great deal of interest in the development of multicast transport protocols layered on top of IP multicast. However, these new transport protocols could cause congestion collapse if they are widely used but do not provide adequate congestion control [7].

The success of the Internet relies on the fact that TCP sessions respond to congestion by reducing their load presented to the network. The congestion control component of a transport protocol has two main objectives [8]:

- Avoid congestion collapse: A network congestion collapse occurs when the network is increasing busy, but there is little effective work getting done. Three scenarios that cause congestion collapse have been identified:
  - Unnecessarily-retransmitted packets
  - Fragmentation
  - Packets that are discarded before they reach their receivers
- Achieve fairness with competing traffics: There are two popular concepts of fairness. One is the max-min fairness and another is global fairness. Under the second concept of fairness, each entity has an equal claim to the network's scarce resources. For example, an entity going through many congested links is using more scarce resources than an entity going though only one congested link.

3.2. Multicast Congestion Control

There are several multicast congestion control protocols proposed recently. The approaches can be classified into three categories: single rate, replicated stream and layered [7, 9].

- Single rate (unirate) approach: Only one rate sent to the whole group. So this is fixed rate for all receivers (or destinations) in one multicast session.
- Replicated stream approach: Receiver will be portioned into groups and each receiver joins one group, so receivers can receive the same amount of service data at different rates.
- Layered approach: The data stream is organized in an incremental way, and a receiver incrementally joins higher groups according to its available bandwidth. In this case, may be both data and rate to each receiver are different with others.

  Multirate approach is considered as the composition of replicated stream and layer approaches. It has flexible rates for any receivers in a section according to their bandwidths of processing capabilities [9]. It is suitable for many services with dynamic users and resources such as video and voice multicast in teleconference, distance learning.

There are three key problems for multicast congestion control as depicted in [7]: feedback implosion, congestion indicator filtering and fairness. So, when building an algorithm for multicast congestion control, one should put all the three problems into consideration.

3.3. Review of TFMCC

Single rate multicast congestion control is considered to the basic problem that all three above approaches have to solve. So, the congestion control in [4] is considered as the basic protocol to solve the problem of multicast congestion control.

Figure 2 shows the basic multicast congestion control mechanism of TFMCC.

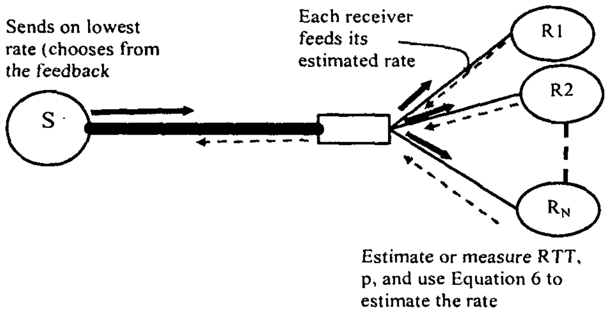

Fig. 2: General multicast congestion control mechanism [4].

In [4], TCP-Friendly Multicast Congestion Control (TFMCC) is a single rate multicast congestion control protocol designed to provide smooth rate change over time. The protocol extends the basic control mechanisms of unicast congestion control in [8] into the multicast domain, using the equation-based methods.

Its fundamental idea is to have each receiver a control equation (6) derived from the model of TCP's long term throughput [10] and use this to directly control the sender's transmission rate:

$$T_{TCP} = \frac{s}{t_{RTT}\left(\sqrt{\frac{2p}{3}} + \left(12\sqrt{\frac{3p}{8}}\right)p\left(1+32p^2\right)\right)} \quad (6)$$

where

T_{TCP} : expected throughput of a TCP flow, calculated as a function of p

p : steady-state loss event rate (measured by each TFMCC receiver)

t_{RTT} : the TCP round-trip time (estimated or measured by each TFMCC receiver)

s : the packet size (in bytes)

There are some main overview points of TFMCC functionality as follows [11]:

- Each receiver measures the packet loss rate.
- The receiver measures or estimates the round-trip time (RTT) to the sender.
- The receiver uses the control Equation (6) to calculate an acceptable transmission rate from the measured loss rate and round-trip time.
- The receiver sends the calculated transmission rate to the sender.
- A feedback suppression scheme is used to prevent feedback implosion while ensuring that feedback from the slowest receiver always reaches the sender.
- The sender adjusts the sending rate from the feedback information.

In TFMCC, the receiver that the sender believes currently has the lowest expected throughput of the group is selected as the current limiting receiver (CLR). The CLR is permitted to send continuous, immediate feedback to the sender without any form of suppression, so the sender can use the CLR's feedback to adjust the transmission rate (reduce or increase the transmission rate to the feedback's indicated rate). This avoids the problem of lacking feedback information when the slowest receiver's feedback path may be congested or lossy. In addition, any receiver whose expected throughput is lower than the sender's current rate sends a feedback message, and to avoid feedback implosion, biased feedback timers in favor of receivers with lower rates are used [4].

a) Measuring the Loss Event Rate (p)

One critical detail of TFMCC which is very important for the thesis is the method it uses to measure packet loss. In TFMCC, a receiver aggregates the packet losses into loss events, defined as one or more packets lost during a round-trip time. The numbers of packets between consecutive loss events is called a loss interval. The average loss interval size (Equation 6) can be computed as the weighted average of the m most recent loss intervals $l_k, \ldots, l_{k-m+1}$:

$$l_{avg}(k) = \frac{\sum_{i=0}^{m-1} w_i l_{k-i}}{\sum_{i=0}^{m-1} w_i} \quad (7)$$

The weights w_i are chosen so that very recent loss intervals receive the same high weights, while the weights gradually decrease to zero for older loss intervals. Large values for m improve the smoothness of the estimate, but a very long loss history also reduces the responsiveness and thus the fairness of the protocol. Values around 8 to 32 seem to be a good compromise.

The loss event rate p used as an input for the TCP model is then taken to be the inverse of l_{avg}. The interval since the most recent loss event is incomplete, since it does not end with a loss event, but it is conservatively included in the calculation of the loss event rate if doing so reduces p:

$$p = \frac{1}{\max\left(l_{avg}(k), l_{avg}(k-1)\right)} \quad (8)$$

See [8] for further detail of Equation (8).

b) Round-Trip Time Measurements

Each receiver starts with initial RTT and this initial RTT is used until a real measurement is made. A receiver measures the RTT by sending timestamped feedback to the sender, which then echoes the timestamp and receiver's identification (ID) in the header of a data packet. An exponentially weighted moving average (EWMA) is used to prevent a single large RTT measurement from greatly impacting on the sending rate. Equation (9) is used to calculate the RTT of each receiver after getting the instantaneous RTT:

$$t_{RTT} = \beta \cdot t_{RTT}^{inst} + (1-\beta) \cdot t_{RTT} \quad (9)$$

t_{RTT}^{inst} is the instantaneous RTT, $\beta_{CLR} = 0.05$ is set for the CLR while $\beta_{non-CLR} = 0.5$ is used for non-CLR receivers due to infrequent RTT measurements. One-way delay RTT adjustments are used by non-CLR receivers between the real measurements.

c) Slowstart

TFMCC uses a slowstart mechanism that quickly approaches its fair bandwidth share at the start of a session. The sending rate in the slowstart period will increase exponentially while normal congestion control allows only a linear increase. An exponential increase can usually lead to heavy congestion, so TFMCC has been designed a safe increase mechanism. For this safe mechanism, a multiple *d* is decided. The target sending rate of the sender will be calculated as *d* times the minimum rate that the sender has received from any of the receivers (in simulation *d* has value 2). Since a receiver can never receive at a rate higher than its link bandwidth, this effectively limits the overshoot to d times that bandwidth.
The slowstart ends as soon as any one of the receivers experiences its first packet loss.

4. PROPOSED ALGORITHM OMCC

This paper gives a brief overview of OMCC. The Optimization-based Multicast Congestion Control (O-MCC) simulation is the combining work that deals with the congestion control for Internet multicasting based on the optimization conditions of the network.

4.1. Optimization of Total Utility

The optimization conditions are placed for on all participants in the whole system. Take Figure 3 as a study case to solve the optimization problem. There are six links (l_1, l_2, l_3, l_4, l_5 and l_6) with capacities $c_1 = 3$, $c_2 = c_3 = c_4 = c_5 = c_6 = 1$ accordingly. Suppose that all links have same length.

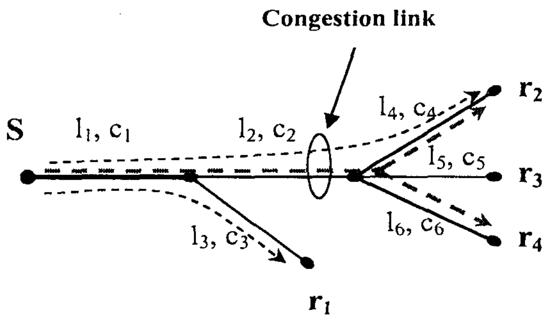

Fig. 3: Simple model for optimal calculation.

In the figure, S is the source (sender) that sends traffic data in three sessions to users. The first session is the unicast session to receiver r_1. The second one is the unicast to receiver r_2 and the last session is the multicast to two receivers r_3 and r_4.
To find the solution for the system's optimization, the total utility has to be considered together with constraints of the link capacities and all session's rates as follows:

maximize $$U_x(x) + U_y(y) \quad (10)$$

subject to $$x + y \le 1 \quad (11)$$

over $$x, y \ge 0 \quad (12)$$

where x and y are the unicast and multicast session rates on the congestion link

All three conditions (10), (11) and (12) have been simplified from the model in [14] because once the above three conditions are satisfied, then the whole system's optimization is also fulfilled. Table 1 shows all results when optimization conditions 10), (11) and (12) are applied with x and y are the session rates going in link c_2 (Figure 3).

Table 1: Different utility functions and system optimization rates

	Sender oriented logarithmic utility	Receiver oriented logarithmic utility	Receiver oriented TCP utility
$U_{unicast}$	$\frac{\log x}{3}$	$\log x$	$-\frac{1}{x}$
$U_{multicast}$	$\frac{\log y}{4}$	$2\log y$	$-\frac{2}{y}$
U_{total}	$\frac{\log x}{3} + \frac{\log y}{4}$	$\log x + 2\log y$	$-\frac{1}{x} - \frac{2}{y}$
U'_{total}	$\frac{1}{3x_{opt}} + \frac{1}{4y_{opt}} = 0$	$\frac{1}{x_{opt}} + \frac{2}{y_{opt}} = 0$	$\frac{1}{x^2_{opt}} + \frac{2}{y^2_{opt}} = 0$
x_{opt}	0.57	0.33	0.41
y_{opt}	0.43	0.67	0.59
Fairness	0.75	2.03	1.44

In Table 1, the sender oriented and receiver oriented utility functions are calculated with two typical utility functions (logarithmic and TCP utility). The two sessions should fill up the bottleneck bandwidth to bring maximum benefit for the telecommunication carrier.

Table 1 shows that using receiver oriented utility is suitable for multicast transferring because the optimum value favors multicast session rate. Also different utility functions are used and show the difference in the optimum value. It seems that using TCP utility, the fairness between unicast and multicast session rates is nearly one or better fairness.

4.2. OMCC

Based on the TFMCC rate control for multicast but the feedback rate to the source is modified according to optimization condition of the network, OMCC is built as the followings:

The TFMCC's rate estimation Equation (6) is applied to build up the OMCC rate estimation equation. In Equation (7), function *f(N)* is the scaling function which depends on the number of receiver in a traffic session. *N* is the number of multicast receivers. For TFMCC, *f(N)* is not considered, or just considered as a constant value.

$$T_{TCP} = f(N) \times \frac{s}{t_{RTT}\left(\sqrt{\frac{2p}{3}} + \left(12\sqrt{\frac{3p}{8}}\right)p\left(1+32p^2\right)\right)} \quad (13)$$

All mechanisms in TFMCC are applied to avoid congestion in the network bottleneck links: The slowstart, CLR, Loss Event Rate measurement and RTT measurement.

5. SIMULATION CONFIGURATION AND RESULTS

5.1. Assumptions

The simulation deals only with one congestion link in a heterogeneous network with different conditions in bandwidth, TCP window size, queue length, link delay and number of receivers in a session. There will be some scenarios that deal with different traffic flows but there is no change in sates of multicast session members during one simulation.

The O-MCC has been implemented by extending the Network Simulator version 2 (NS-2). The design and the implementation of O-MCC are supported by the TFMCC design and the Optimization approach.

5.2. Simulation Descriptions

In simulation, three kinds of traffic are used: CBR (Constant Bit Rate for traditional voice), FTP (File Transfer Protocol for bulk transfer) and TFMCC (multimedia traffic). The data packets are sent from their corresponding agents: UDP (User Datagram Protocol), TCP and TFMCC.

The general simulation topology is shown in Figure 4. This is the single-bottleneck topology where a number of sending nodes are connected to many receiving nodes through a common bottleneck. The bottleneck has bandwidth values of 1 Mbit/s, 2 Mbit/s or 4 Mbit/s; its queue length is of 10, 20 or 40 packets, and its link delay is 20 ms (modified from [4]). Other links have capacity of 5 Mbit/s, 5 ms link delay. For study the affect of link delay, all time delay of links are changed in some simulations into 10 ms, 1 ms and 0.1 ms.

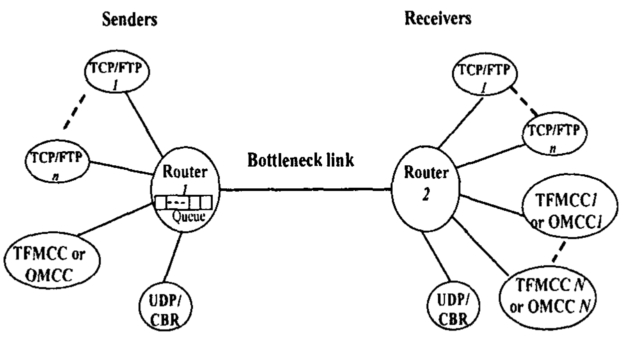

Fig. 4: General topology for simulation

There are FTP's application traffics attached to TCP agents which send unicast traffic (point to point) and which can change traffic rate adaptively to the network condition using window congestion control. The UDP agent sends CBR traffic packets and receives packets without controlling the rate, this traffic is used as the basic flow in the internet, which has no effect on the optimization total utility (the rate is nearly constant). The multicast source is attached to the TFMCC agent and sends its packets to multiple receivers (point to multipoint). The multicast rate is also changed according to the optimization network conditions.

5.3. Performance Parameter Evaluation

In this paper, the performances of the congested network with multicast communication are considered as the total throughput of congestion link and the fairness. The total maximum throughput over a bottleneck link achieved is also the best value for the network provider (all resources are sold). By changing numbers and types of flows, controlling TCP window size and queue length in the network, different values for throughput are achieved over the congestion link.

a) Throughput

This parameter is observed at the congestion link (after going out of the queue). It can be defined separately for each traffic flow or the total traffic over the congestion link as follows:

$$TP = \frac{N_r \times S \times 8bit}{T_{observed} \times 10^6} \quad [Mbit/s] \quad (14)$$

where

- N_r :Total received packets (UDP, TCP or TFMCC packets)
- S :Packet size in bytes
- TP :Throughput of a (all) flow(s) measured in Mbit/s
- $T_{observed}$:Observation time in second, can be a short duration to be used in drawing or total simulated duration to take the average throughput

b) Fairness

It is another performance parameter to be considered because of its importance in the heterogeneous network. In this thesis, only the fairness of throughput is considered, other quality of service (QoS) parameters will not take into account.

$$Fairness = \frac{n \times TP_{multicast}}{\sum_{i=1}^{n} TP_{unicast}(i)} \quad (15)$$

where

- n : Total number of unicast flows (number of TCP flows, can be 1, 2 or 3 in simulation
- $TP_{multicast}$: Throughput of multicast session over the congested link
- $TP_{unicast}(i)$: Throughput of unicast session number *i* over the congested link

5.4 Simulation Results

a) TFMCC Performances

TFMCC Performances and Fairness in Different Bandwidths

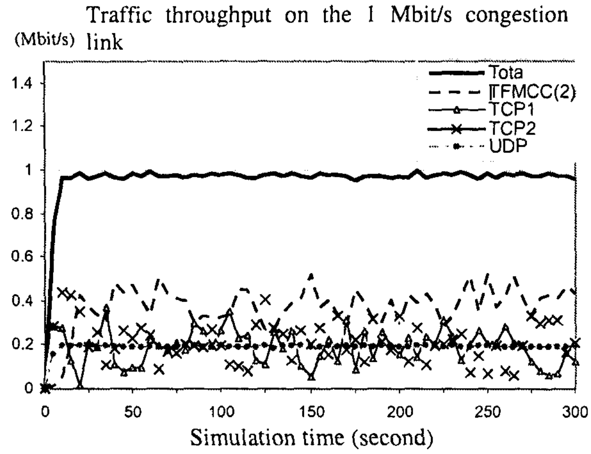

Fig. 5: TFMCC (2 receivers) flow over 2 TCP flows, total BW = 1Mbit/s.

The simulation is taken for three cases of bandwidth. The number of flows and the queue capacity are constant. There are four fixed flows: one TFMCC flow with two receivers, two TCP flows and one CBR flow aggregated in the network; all flows (single rate) go on a congestion link with bandwidth of 1, 2 and 4 Mbit/s.

Figure 5 shows a general view for case of the bottleneck link has 1 Mbit/s capacity. The figure brings a graphical view of all flow throughputs: TFMCC flow takes quite much more bandwidth comparing to the TCP flows; the UDP traffic line is almost constant. The total throughput over the congestion link has the highest value so it is the thick line with approximate value of 1 Mbit/s in upper position.

Table 2: Simulation results for BW = 1 Mbit/s

Traffic	Average throughput in the 1 Mbit/s link	
	In 300 seconds	In last 250 seconds
Total BW	0.971± 0.002	0.974 ± 0.002
TFMCC (2)	0.389 ± 0.011	0.410 ± 0.014
TCP1	0.195 ± 0.019	0.190 ± 0.017
TCP2	0.196 ± 0.020	0.184 ± 0.017
UDP	0.191 ± 4E-04	0.192 ± 5E-04
Fairness	1.995 ± 0.113	2.200 ± 0.157

In Table 2, (±) sign shows the standard deviation of the five running values in Figure 4 simulation. From the calculated values, one can see the fairness of multicast rate over unicast rate is approximately equal to two.

In other cases of congestion link bandwidth of 2 and 4 Mbit/s, the fairness values are approximately 1.5 and 1.1 accordingly.

TFMCC with Different Number of Receivers in Large Range and with Different Queue Capacities

In this part, TFMCC is simulated with different number of TFMCC receivers *(N)*. Figure 6a views the general throughputs of all flows in the network in case of congestion link queue capacity (Q) is 10 packets, the total link delay is 30 ms (three links from sender to receiver).

Figure 6b is the summary of fairness value with queue capacities 10, 20, and 40 packets which is the ratio of multicast throughput over unicast average throughput. With the increase in queue capacity, the total throughput over the congested link increases while the fairness decreases.

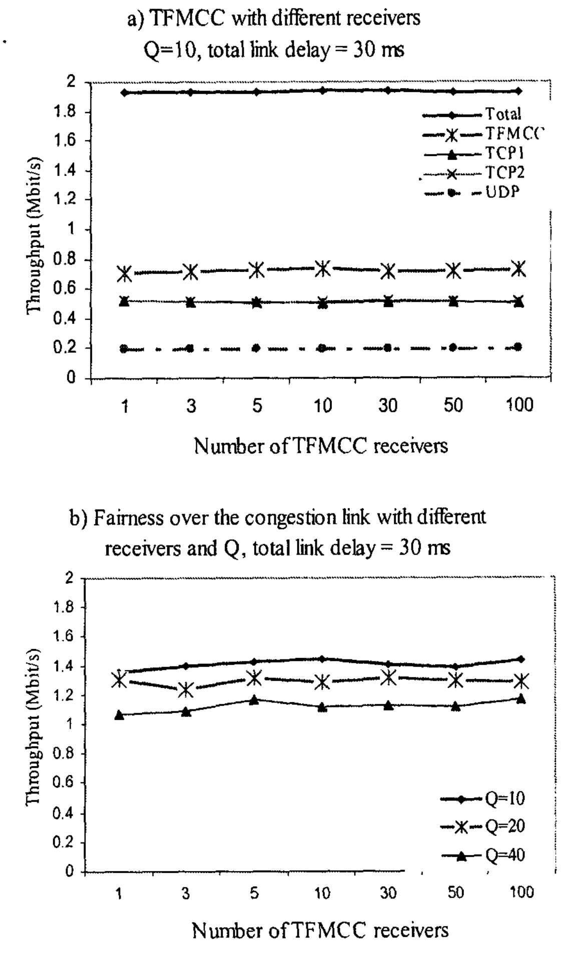

Fig. 6: TFMCC with different receivers, different queue capacities (Q), and total link delay = 30 ms.

TFMCC with Different Link Delays

Previously, all the simulations run with fixed total link delay of 30 ms in total three links for TCP and TFMCC. In this part, each link delay (D) is tuned to be 1 ms and 0.1 ms, and all flows are tested with different queue capacities and different number of TFMCC receivers.

Figure 7 shows a general view when observing the fairness of TFMCC and TCP in a variable environment of different round trip time. In the figure, with queue equals 10 packets, the fairness varies so much regardless of the numbers of receivers.

Comparing to Figure 6, it seems that with long distance (30 ms delay equals 9,000 km (can be considered as the end-to-end propagation time)) the fairness does not change much with different queue capacities. But when the distance is shorter, the fairness keeps quite stable with only large queue capacities.

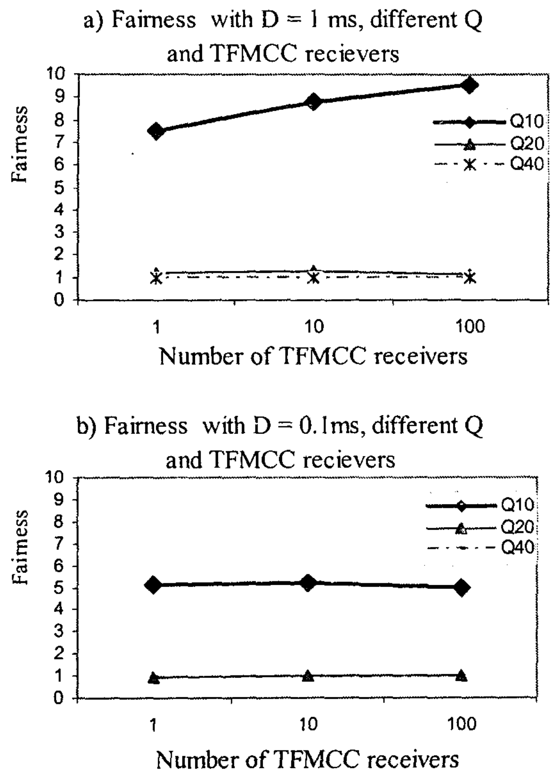

Fig. 7: Fairness with different TFMCC receivers, different queue capacities (Q) and different link delays (D).

b) OMCC Performances

From the view of optimization, if there are unicast and multicast sessions share the same bottleneck link, it is better for the multicast session takes more bandwidth as the utility takes the number of receivers into consideration. Using different definition of utility function, there would be different bandwidth sharing ratio to reach the optimal utility value.

Therefore, the target of this part is to modify TFMCC in to proposed Optimization-based Multicast congestion control. That means the multicast will get more favor in bandwidth sharing when the number of its receivers increases comparing to other competition flows, but it still keeps a certain lever of fairness and keeps congestion avoidance depending on the network conditions.

OMCC with Different Utility Functions

In the proposed OMCC and the simulation, the scaling function definition is based on the number of multicast receiver, means $f(N) = N$ for Logarithmic utility function and $f(N) = \sqrt{N}$ for TCP-utility function [13].

In Figure 8a, the general view for all throughputs on the congestion link (2 Mbit/s) is shown with one proposed Logarithmic- type OMCC flows having N receivers, two TCP flows and one UDP flow with fixed rate of 0.2 Mbit/s, path delays for the proposed OMCC and TCP flows are 30 ms, queue capacity is 20 packets. Figure shows that the throughput of OMCC is quite stable in the large receiver range. Figure 8b presents the comparison between the TFMCC and the two proposed OMCC fairness when they run in separate simulation with two TCP and one UDP flows. In the figure, the Log-OMCC seems to get most stable in fairness compared to others.

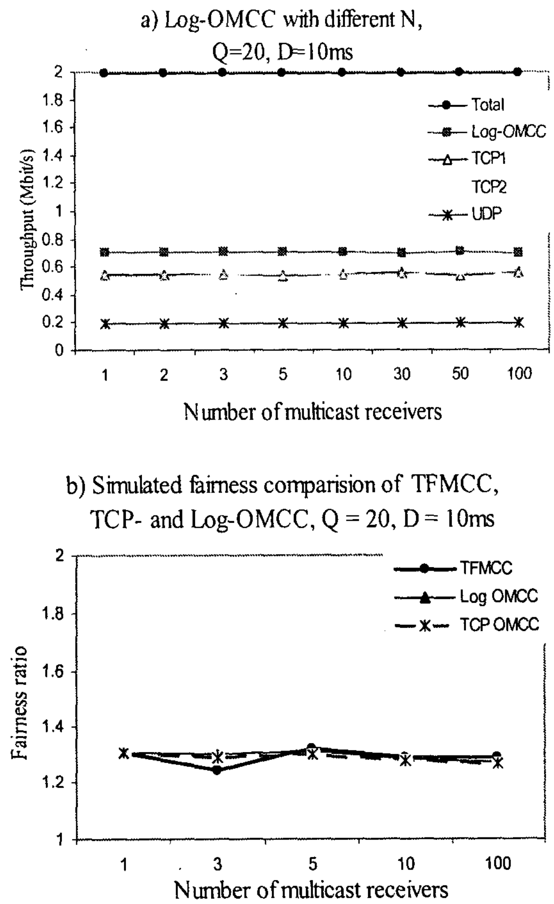

Fig. 8: Total view of the proposed Log-OMCC and OMCC fairness compared to TFMCC with changes in number of receivers in 2Mbit/s congestion link.

Effect of Different Link Delays and Queue Capacities to OMCC Fairness

In this part, simulation is run using different link delays and Q is 10 packets. Only the fairness is considered and summarized in Figure 8 and in [13].

In Figure 9a, with each 10 ms link delay, the fairness for TCP-OMCC seems to get litter higher value compared to TFMCC fairness and even better than the fairness of Log-OMCC. In Figure 9b the link delay is tuned to 1ms. With small number of multicast receivers N, the Log-OMCC gets higher value more than TFMCC but reduces when N increases. The TCP-

OMCC fairness increases when N is large but seems to fluctuate much. The situation results for Log OMCC in Figure 9c even worse with large number of multicast receivers: Its fairness is smaller compared to TFMCC fairness when the number of its receivers increases. For TCP-OMCC, its fairness still increases following the increase of N.

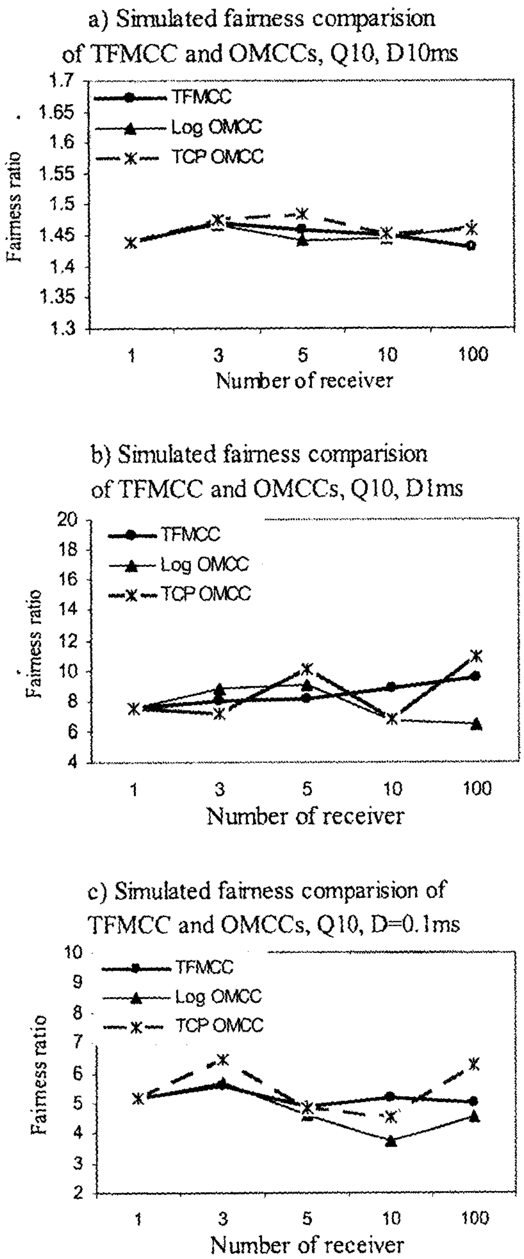

Fig. 9: The fairness of Log-OMCC and TCP-OMCC compared to TFMCC in the 2Mbit/s congestion link, Q = 10 packets.

6. CONCLUSIONS

This paper studies and presents a review on multicast congestion control combined receiver-oriented utility function in a simulated network. For real time multicast transmission, it is not easy to control the congestion without deteriorating the performance of other existing sessions in the network. Besides, the feedback from many multicast receivers should also be controlled strictly.

From the simulation results, it can be concluded that TFMCC provides the best fairness to TCP in the network with plentiful network resources, i.e. large available bandwidth and long queue. It supports a good smooth throughput in a non-random loss network for a large number of receiver ranges. TFMCC has larger bandwidth portion than the other aggregating flows in the network, with fixed number of TCP flows, TFMCC bandwidth does not increase much when the number of its receivers increases.

To improve the TFMCC with multicast utility trend, a modification called Optimization-based Multicast Congestion Control (OMCC) is proposed to take receiver- oriented utility function into the traffic control mechanism.

The results of the proposed OMCC simulation in NS2 simulator show a better level of stable throughput and improvement in the receiver-oriented utility optimization concept. These make a future guild for multicast congestion control based on the utility functions of different types of users and various types of services in the real network to get best benefit to all internet participants.

References

[1] IEEE Network, "Multicasting: An Enabling Technology", Vol.17, No.1, January 2003.

[2] Yang, Y. R. and Lam, S. S., "Internet Multicast Congestion Control: A Survey", *Proc. of ICT 2000*, Acapulco, Mexico, May 2000.

[3] Shapiro, J. K., Towsley, D. and Kurose J., "Optimization-Based Congestion Control for Multicast Communications", IEEE Communications Magazine, September 2002.

[4] Widmer, J. and Handley, M., "TCP-Friendly Multicast Congestion Control (TFMCC): Protocol Specification", IETF INTERNET-DRAFT I, November 2002.

[5] Wydrowski, B. and Zukerman, M., "QoS in Best-Effort Networks", IEEE Communications Magazine, Vol. 40, No.12, pp. 44 – 49, December 2002.

[6] Kunniyur, S. and Srikant, R., "End-to-End Congestion Control Schemes: Utility Functions, Random Losses and ECN Marks", *Proc. of IEEE Infocom*, March 2000.

[7] Yang, Y. R. and Lam, S. S, "Internet Multicast Congestion Control: A Survey", *Proc. of ICT 2000*, Acapulco, Mexico, May 2000.

[8] Floyd S., "Congestion Control Principles", RFC 2914, September 2000.

[9] Li, B. and Liu, J., "Multirate Video Multicast over the Internet: An Overview", IEEE Network January/February 2003, pp. 22 - 33.

[10] Padhye, J., Firoiu, V., Towsley, D. and Kurose, J., "Modeling TCP Reno Performance: A Simple Model and Its Empirical Validation", IEEE/ACM Transactions on Networking, 8(2), pp. 133 – 145 April, 2000.

[11] Kwon, G. and Byers, J. W., "Smooth Multirate Multicast Congestion Control", BUCS Technical Report 2002-025, September 2002.

[12] Low, S. H. and Lapsley D. E., "Optimization Flow Control-I: Basic Algorithm and Convergence", IEEE/ACM Transaction on Networking 7(6), pp. 861 - 874, 1999.

[13] Hang, N. T. T., "Optimization- based Congestion Control for Internet Multicast Communications," Master Graduate Thesis Study, AIT, Thailand, December 2003.

[14] Kelly, F., "Charging and Rate Control for Elastic Traffic", European Transactions on Telecommunications, Vol. 8, pp. 33 – 37, 1997.