On Max-Min Intra-Session Fairness in Replicated Multicast Services *

Zongming Fei
Laboratory for Advanced Networking
Department of Computer Science
University of Kentucky
Lexington, KY 40506-0046
fei@cs.uky.edu

Mengkun Yang
Laboratory for Advanced Networking
Department of Computer Science
University of Kentucky
Lexington, KY 40506-0046
myang0@cs.uky.edu

ABSTRACT

Replicated multicast service deals with the well-known problem of heterogeneity by dividing receivers into multiple multicast groups. Most of current work focuses on optimizing the *social welfare* represented as a sum of some performance measures of receivers. In this paper we define a new concept called *intra-session fairness* and present an optimal solution to achieve fairness among receivers in the same session. The goal is to maximize the minimum fairness value of the receivers. The novelty of the framework is that it is independent of the specific definition of the fairness function. We illustrate a layering method to implement the max-min intra-session fair allocation and demonstrate the significant difference in fairness achieved by the maximal social welfare algorithm and the max-min intra-session fairness algorithm.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Modeling techniques

General Terms

Design, Performance, Algorithms

Keywords

fairness, multicast, replication

1. INTRODUCTION

Multicast communication achieves scalability by sending data to multiple receivers at the same time. Receivers in the same multicast session usually share the fate with each other. A conventional multicast session usually consists of a single multicast group and receivers have to get data at the same rate, even if their capabilities are quite different. In some applications the sender simply transmits at the lowest rate of all the receivers. It is not so fair, to some extent, to those receivers capable of receiving at a much higher speed. In other applications receivers can tolerate a certain degree of packet loss and we can set the sending rate higher than the lowest rate. For example, for a multicast video-on-demand server, setting its sending rate higher than the lowest rate may improve the *fairness* [1] among receivers in the same session. In Figure 1 where a single group multicast session consists of one server S and four receivers $R1, R2, R3$ and $R4$, we can set the group rate to 2 (the lowest rate), or set the rate somewhat higher (say 4). Here comes the first question.

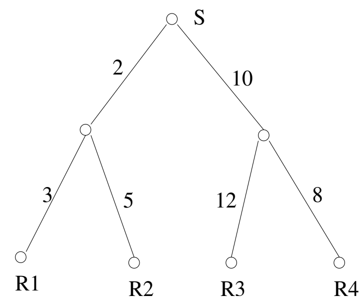

Figure 1: A multicast session with a single group

Question #1: In a session with a single multicast group, how should we set the group rate so that it is fair to some extent to both fast and slow receivers?

More recently, using multiple multicast groups has been proposed to deal with the heterogeneity problem among receivers in the multicast communication [2, 7]. These schemes tend to put those receivers with similar capacities to the same group and it is fairer to those receivers with a higher capacity. In Figure 1, we can let server S form two multicast groups and each group transmits at a different rate. For ex-

*This work was supported in part by the National Science Foundation under Grant CCR-0204304 and EIA-0101242. Any opinions, findings, and conclusions expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

NGC '02, October 23-25, 2002, Boston, Massachusetts USA

[1] Fairness is a vague concept. To some people, it means all parties involved get the equal share. To others, it means they get the share proportional to their capacities. We will give a formal definition of max-min intra-session fairness in Section 3.

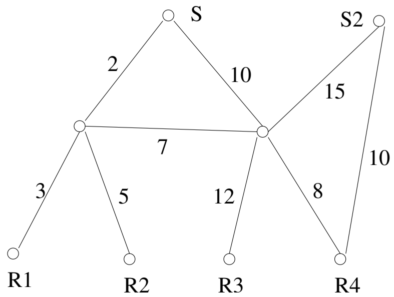

Figure 2: A multicast session with multiple groups

ample, receivers $R1, R2$ join one group sending at rate 2 and receivers $R3, R4$ join the other group sending at rate 8. Note that all receivers $R1, R2, R3$ and $R4$ are still in the same multicast session even though they are allocated to different groups. This is because they are still receiving the same content, and it is perfectly fine to move a receiver $(R1, \cdots, R4)$ from one group to another though there may be performance consequences. A more general form of replicated multicast service is the replication of multicast servers. For example, in Figure 2 we have the multicast server S replicated at another place $S2$, which sends out the same content as S, but possibly at a different rate (thus quality). Each of them can form one or more multicast groups and each receiver will be allocated to one of the groups from either server.

We need criteria to guide the allocation of receivers to one of the multicast groups in the session. The most widely used measure is the sum of utilities of receivers [7]. Receivers are divided into groups and the group rates are set in such a way that the sum is maximized. This may achieve better social welfare but, at the same time, has to sacrifice performance of some receivers and thus unfair. Here comes our second question.

Question #2: In a session with multiple multicast groups, how should we divide receivers into groups and set the rates of those groups so that it is fair to some extent to all receivers?

In this paper we give a formal definition of *intra-session fairness* among receivers in the same multicast session. The concept is based on a fairness function defined for individual receivers and the goal is to maximize the minimum value of fairness of receivers in the same session.

Starting with a single group multicast session, we give optimal solutions to the rate assignment problem (question #1) for a variety of fairness function definitions. Note the solution is specific to a fairness function and we believe there always exists an optimal solution to any fairness function. Then for a session with multiple groups, we give an algorithm that can generate an optimal solution to the problem of grouping and rate assignment (question #2). The novelty of the solution method is that it is a general framework and independent of the specific definition of the fairness function. It can be used to solve the grouping and rate assignment problem in a session with multiple multicast groups as long as we have a solution to the rate assignment problem for the fairness function in the single group session. We illustrate a method of implementing the max-min intra-session fair allocation by layering and demonstrate the significant difference in fairness achieved by the maximal social welfare algorithm and the max-min intra-session fair allocation.

The rest of the paper is organized as follows. Related work is discussed in Section 2. Section 3 gives a formal definition of intra-session fairness. In Section 4 we present optimal solutions to the max-min intra-session fairness problem in sessions with a single group and with multiple groups. We also give performance evaluation of the algorithms, in comparison with the maximal social welfare algorithms. Section 5 discusses the implementation issues and illustrates a layering method for video streaming applications. We conclude the paper in Section 6.

2. RELATED WORK

Fairness is one of the key issues in bandwidth allocation. Max-min fairness [5] aimed at maximizing the minimum bandwidth allocated to flows involved. The bandwidth is equally divided among all flows for each link. The rate of each flow is determined by the minimum rate allocated to it by the links from the source to the destination. The flow with minimum rate is assigned its bottleneck rate and the bandwidth it does not use in other links will be divided equally among other flows. The process is repeated until we determine the allocation for all flows. It is extended to the multicast case in [10], where all receivers in the same multicast group are allocated the same group rate, which is determined by all links involved from the sender to all receivers in the group. The goal is still to maximize the minimum rate of all receivers. The utility max-min fairness maximizes the minimum of the utility of receivers instead of the rate [3].

Multi-rate max-min fair allocation by Rubenstein et al extended the multicast max-min fairness further to the case where receivers in the same multicast session can receive at different rates [9]. However the allocation algorithm cannot be used to answer the two questions raised in Section 1 because its target is the *inter-session* fairness. Further, to implement multi-rate max-min fairness end-to-end, we have to use up to n groups if we have n receivers. If the number of groups we can use is smaller than n, we still have the problems of deciding the allocation of receivers into groups and choosing a rate for each group, in a way that is fair to receivers. This is the *intra-session fairness* problem we want to solve in this paper.

Another related work is the use of multiple multicast groups to deal with the problem of heterogeneous receiver capacities [1, 8, 2, 7]. In the scheme of destination set splitting [1] the sender would send to different groups at different rates controlled by the feedback from the receivers in each group. Multiple multicast channels were also used in a reliable multicast scheme to reduce end-host processing costs and network bandwidth usage [8]. Bhattacharyya et al designed a scheme to minimize the average transfer time by using multiple multicast groups in bulk data transfer [2]. More recently the work by Kar addressed the problem of achieving rates that maximize the total receiver utility for multirate multicast sessions [7]. Most of them aimed at optimizing some performance measure that is the summation (or average) of receivers' performance, at the cost of sacrificing performance of some receivers.

Most closely related work is the inter-receiver fairness which intends to solve the same intra-session fairness problem [6]. A fairness function is defined for individual receivers and the inter-receiver fairness is defined as the weighted sum of the fairness' of all receivers. Our observation is that the sum is optimized, but the fairness of each individual receiver is not taken care of. Therefore it is still a maximal social welfare allocation. The point is that the distinction between social welfare and intra-session fairness is not in the form of performance function defined for individual receivers, but in how they are combined to represent the final performance measure we want to optimize.

3. MODEL

3.1 Performance Measure of an Individual Receiver

Performance of receivers can be measured in different ways. For bulk data transfer we may be interested in the rate a receiver can get data from the sender. For multimedia applications we may define more complicated utility functions, such as the proportion of useful data.

Considering the fact that receivers may have different capacities (either due to the processing speed, or the rate of its access link, or its fair share at the bottleneck link), we define the measurement function with two parameters. One is the capacity (c) of the receiver and the other is the rate (r) of the multicast group it receives data from. Here are some examples.

1) $f_1(c,r) = min(c,r) = \begin{cases} r & \text{if } r \leq c \\ c & \text{if } r > c \end{cases}$

2) $f_2(c,r) = \frac{min(c,r)}{max(c,r)} = \begin{cases} r/c & \text{if } r \leq c \\ c/r & \text{if } r > c \end{cases}$

3) $f_3(c,r) = \begin{cases} r^2/c^2 & \text{if } r \leq c \\ c^2/r^2 & \text{if } r > c \end{cases}$

4) $f_4(c,r) = \begin{cases} 1-(1-r/c)^2 & \text{if } r \leq c \\ 1-(1-r/c)^2 & \text{if } c < r \leq 1.5c \\ 1.125c/r & \text{if } r > 1.5c \end{cases}$

The performance can be as simple as the actual receiving rate as in f_1, which targets at getting an equal share for each receiver. An alternative is the ratio of the group rate over the capacity such as f_2, which tries to achieve the goal that each receiver gets the rate proportional to its capacity. Performance measurement for individual receivers can be even more complicated (e.g., using functions f_3 and f_4). For most multimedia applications, even if a receiver has to drop some packets when the sending rate is higher than its capacity, there is still some value to the receiver. That is why we define $f(c,r) > 0$ when $r > c$ in these functions. However, the value to the receiver when it has to drop packets is not as good as in the case where the sending rate is equal to its capacity. As a matter of fact, as the sending rate (r) increases in the range $(c, +\infty)$, the fairness value decreases in f_2, f_3 and f_4. This implies that the higher the sending rate, the less value for the receiver in this range.

In some other applications, such as file or software distribution, date are no use at all for a receiver if the server is sending at a rate (r) greater than its capacity (c) and the receiver may lose part of a file. In this case we may prefer the following fairness functions that have value 0 if $r > c$.

1') $f_1'(c,r) = \begin{cases} r & \text{if } r \leq c \\ 0 & \text{if } r > c \end{cases}$

2') $f_2'(c,r) = \begin{cases} r/c & \text{if } r \leq c \\ 0 & \text{if } r > c \end{cases}$

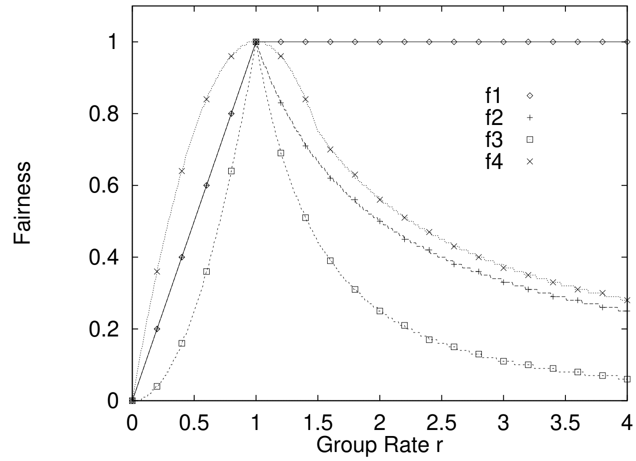

Figure 3: Fairness Functions

3') $f_3'(c,r) = \begin{cases} r^2/c^2 & \text{if } r \leq c \\ 0 & \text{if } r > c \end{cases}$

4') $f_4'(c,r) = \begin{cases} 1-(1-r/c)^2 & \text{if } r \leq c \\ 0 & \text{if } r > c \end{cases}$

We plot functions f_1, f_2, f_3 and f_4 in Figure 3 with $c = 1$. All these functions (including f_1', f_2', f_3' and f_4') have the following property: it monotonically increases over $(0, c)$ and decreases over $(c, +\infty)$. The intuition behind this is that the closer the group rate to the capacity the better its performance, and a receiver achieves the maximal utility or fairness if the group rate is equal to its capacity.

Based on measures defined for individual receivers, we will define *social welfare* and *intra-session fairness*. We want to emphasize that our definitions and algorithms do not depend on any specific form of functions defined for individual receivers and one can use whatever performance measure he/she thinks appropriate for a specific application.

3.2 Social Welfare

The social welfare emphasizes the total or average performance of all receivers. We define the social welfare of a session as the weighted sum of performance measures over all receivers, i.e.,

$$W = \sum_{i=1}^{n} w_i * f(c_i, r_{g(i)})$$

where c_i is the capacity of receiver i, $r_{g(i)}$ is the rate of the multicast group ($g(i)$) receiver i is allocated to.

The weights (w_i's) and function f will be determined by a specific application. Even if f is some fairness measure for each receiver (for example f_2), we still consider W is a social welfare measure, not an intra-session fairness measure.

3.3 Intra-Session Fairness

Intra-session fairness achieves the fairness among receivers in the same multicast session. Unlike the social welfare measure, the main focus is on each individual value of some performance function f for receivers. Therefore we define the intra-session fairness to be a vector of performance measures of receivers [2]. Given n receivers, the intra-session fairness

[2]Actually it can also be defined as a multi-set of values of

of an allocation is defined as a vector, i.e.,

$$F = \langle v_1, v_2, \cdots, v_n \rangle$$

where $v_i = f(c_i, r_{g(i)})$ and c_i is the capacity of receiver i, $r_{g(i)}$ is the rate of the multicast group ($g(i)$) receiver i is allocated to.

To compare the relative fairness between allocations, we define an ordering over the fairness vectors. It is hard to compare two vectors if their elements (v_i's) are not ordered. Our idea is that for each vector we define a canonical form in which elements in the vector are ordered. We then define an ordering over the canonical forms and finally extend that ordering to any fairness vectors.

Def 1: The canonical form of vector F is denoted as $\mathcal{C}(F)$. If $F = \langle v_1, v_2, \cdots, v_n \rangle$, then $\mathcal{C}(F) = \langle u_1, u_2, \cdots, u_n \rangle$, where $u_1, u_2, \cdots, u_n$ is a permutation of $v_1, v_2, \cdots, v_n$, and $u_i \leq u_{i+1}$ for all $1 \leq i \leq n-1$.

Def 2: We define an ordering $\prec_v$ over canonical forms. $\langle u_1, u_2, \cdots, u_n \rangle \prec_v \langle t_1, t_2, \cdots, t_n \rangle$ iff there exists an i $(1 \leq i \leq n)$ such that $u_i < t_i$ and for all $1 \leq j < i$, $u_j = t_j$.

Def 3: Two vectors $F_1 \prec_v F_2$ iff $\mathcal{C}(F_1) \prec_v \mathcal{C}(F_2)$.

To make it complete we define an equivalent relation over fairness vectors.

Def 4: Two vectors $F_1 =_v F_2$ if they have the same canonical form, i.e., $\mathcal{C}(F_1) = \mathcal{C}(F_2)$.

Def 5: Two vectors $F_1 \preceq_v F_2$ if $F_1 \prec_v F_2$ or $F_1 =_v F_2$.

The goal of the *max-min intra-session fair* allocation is to maximize the minimum of fairness value in the fairness vector. With the ordering defined, the optimal solution is the one that maximizes F over the ordering $\prec_v$, or to try to get $\max_{\prec_v} \langle v_1, v_2, \cdots, v_n \rangle$.

3.4 Summary

The social welfare and the intra-session fairness are two different measures in multicast services. Optimizing one of them does not necessarily joepardize the other and we do observe that in some cases we can improve both measures at the same time. We do not argue that the intra-session fairness is a better measure than the social welfare in allocating receivers to one of the replicated multicast groups and determining the rates of these multicast groups. Nor do we object to the ideas of using or adapting the algorithm maximizing one measure to get a (possibly approximate) solution to the other problem. However we want to point out that in general we cannot achieve the max-min intra-session fairness by maximizing the social welfare and *vice versa*.

4. A FRAMEWORK FOR MAX-MIN INTRA-SESSION FAIR ALLOCATIONS

4.1 Single Multicast Group

In a single group multicast session, all receivers get the same group rate no matter what their capacities are. The problem is to determine the rate of the group so that the fairness F is maximized with regard to $\prec_v$. This depends on the definition of individual fairness function f. Given n receivers with the capacity of receiver i being c_i and $c_i \leq c_{i+1}$ for all $1 \leq i \leq n-1$, Table 4.1 gives the optimal group rate to maximize the intra-session fairness F for the functions defined in Section 3. The computation can be complicated depending on the definition of the fairness function. For example, the solution for function f_4 needs to solve an equation $P(x) = 0$ if $c_n > 3c_1$, where $P(x) = x^3 - 2c_n x^2 + 1.125 c_1 c_n^2$. Note that if receivers cannot tolerate loss ($f(c, r) = 0$ if $r > c$, as in f_1'), we have to set the group rate to the slowest receiver to avoid having an element of 0 in the vector. Another observation is that the group rate depends only on rates of the slowest receiver and the fastest receiver. This is because we impose the requirement that the fairness function increases monotonically over $(0, c)$ and decreases monotonically over $(c, +\infty)$.

4.2 Multiple Multicast Groups

To deal with the heterogeneity problem of receivers, multiple multicast groups are used to send the content to receivers at different rates. Receivers with similar capacity join the same group so that slower receivers will not affect the performance of faster receivers. If the maximal number of groups we can have is m, the problem becomes grouping receivers with similar rates together and assigning them to one of the m groups with the goal of maximizing the intra-session fairness F.

The brute force solution to the problem involves trying every possible way to select m indices from n possible positions. We develop a framework that can solve the problem in polynomial time. The idea is to transform the original problem to the longest path problem and then use a modified Dijkstra's algorithm [4] to solve it. In the following discussion we assume $c_i \leq c_{i+1}$ for all $1 \leq i \leq n-1$.

Informally, we represent possible groupings of receivers and their association with a multicast group as a directed graph, with the vertices arranged in rows and columns as follows. For the problem of allocating n receivers to m groups, we have $n \times m + 1$ vertices. Figure 4 illustrates the case in which we have 6 receivers and 3 groups. There is one column of vertices for each group; there is one row of vertices for each receiver. Vertex $v_{i,j}$ in row i and column j represents the potential that the receivers assigned to group j will start from receiver i. An edge from $v_{i,j}$ to $v_{k,j+1}$ $(i \leq k)$ represents that receiver i to receiver $k-1$ will be grouped together and assigned as group j. The weight on this edge is the intra-session fairness vector of this multicast group. The length of a path is the concatenation of elements from the fairness vectors on the edges in the path. The comparison between the lengths of two paths is based on the ordering $\prec_v$ defined in Section 3.3. Note we only need to compare two paths with the same end points and therefore, they have the same number of elements in the vectors.

A solution is represented as a path that includes one vertex per column in the graph. With the edges and weights as described, the longest path corresponds to an optimal solution.

The *correctness* of the framework is straightforward. Any allocation can be mapped to a path in the directed graph, with the length of the path being the intra-session fairness of the allocation. The fact that the path found by our al-

the fairness function for each receiver and then we can define an ordering over multi-sets. However, we choose to use the vector notation for the ease of presentation.

Fairness function	Optimal group rate
$f_1(c,r) = \begin{cases} r & \text{if } r \le c \\ c & \text{if } r > c \end{cases}$	$r = c_n$
$f'_1(c,r) = \begin{cases} r & \text{if } r \le c \\ 0 & \text{if } r > c \end{cases}$ and $f'_2(c,r), f'_3(c,r), f'_4(c,r)$	$r = c_1$
$f_2(c,r) = \begin{cases} \frac{r}{c} & \text{if } r \le c \\ \frac{c}{r} & \text{if } r > c \end{cases}$	$r = \sqrt{c_1 * c_n}$
$f_3(c,r) = \begin{cases} \frac{r^2}{c^2} & \text{if } r \le c \\ \frac{c^2}{r^2} & \text{if } r > c \end{cases}$	$r = \sqrt{c_1 * c_n}$
$f_4(c,r) = \begin{cases} 1-(1-\frac{r}{c})^2 & \text{if } r \le c \\ 1-(1-\frac{r}{c})^2 & \text{if } c < r \le 1.5c \\ 1.125\frac{c}{r} & \text{if } r > 1.5c \end{cases}$	$r = \begin{cases} \frac{2*c_1*c_n}{c_1+c_n} & \text{if } c_n \le 3c_1 \\ x, \text{where } x^3 - 2c_n x^2 + 1.125 c_1 c_n^2 = 0 & \text{if } c_n > 3c_1 \\ \quad \text{and } c_1 \le x \le c_n. & \end{cases}$

Table 1: Optimal group rate in the case of single multicast group

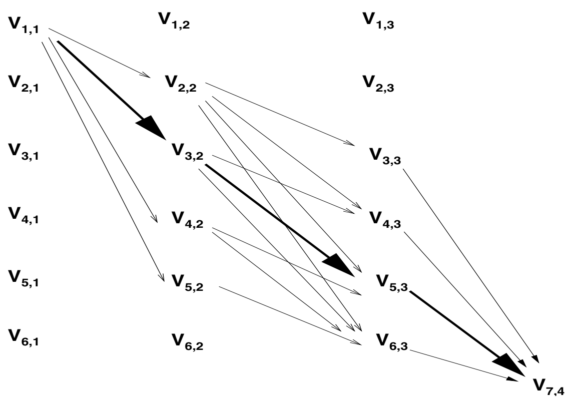

Figure 4: A graph for six receivers and three groups

gorithm is the longest implies that the solution achieves the maximal intra-session fairness. Therefore, it is an optimal solution.

4.3 Performance Results

We evaluate the performance of the max-min intra-session fair algorithm and the maximal social welfare algorithm. We have 100 receivers with capacities ranging from 0 to 1 following uniform, exponential and normal distributions. The exponential distribution has a mean of 0.5 and we only use those values between 0 and 1. The normal distribution has a mean 0.5 and variance 0.25. The function f_2 is used as the individual fairness function in the calculation of the intra-session fairness and the social welfare (equal weights). Figure 5 shows the minimum fairness for the max-min fair allocation algorithm and the maximal social welfare algorithm when the capacity of receivers follows the uniform distribution. As the number of groups in a session increases from 1 to 40, the minimum fairness of receivers increases from 0.05 to more than 0.96. However there are significant differences between the max-min intra-session fairness algorithm and the maximal social welfare algorithm. For example, when the number of groups is 8, the max-min fairness algorithm has a minimum fairness of 0.78 while the maximal welfare algorithm has a minimum fairness of 0.34.

Figure 6 shows the minimum fairness when the capacity follows the exponential distribution. The minimum fairness is a bit closer between two algorithms, but we still can

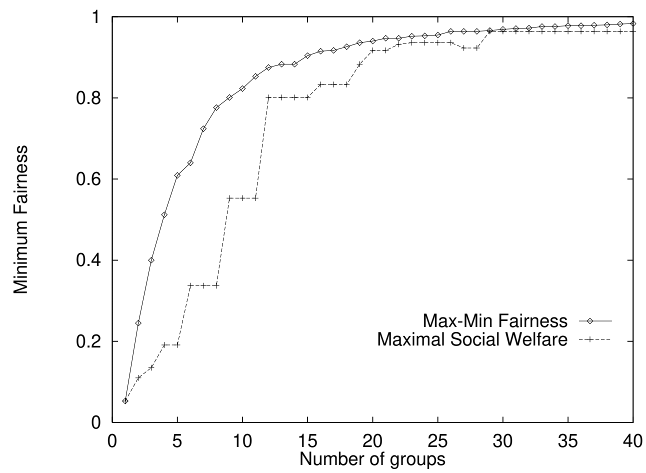

Figure 5: Minimum Fairness (Uniform)

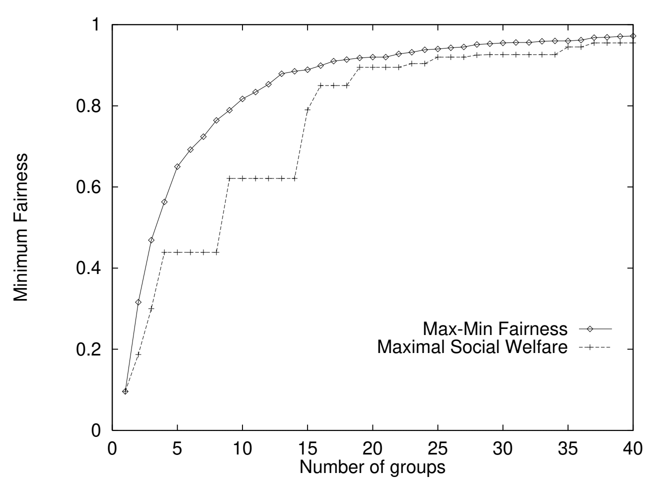

Figure 6: Minimum Fairness (Exponential)

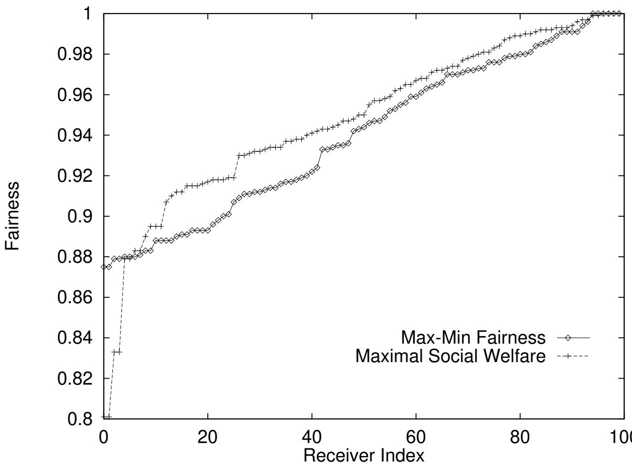

Figure 7: Fairness of Receivers (Uniform)

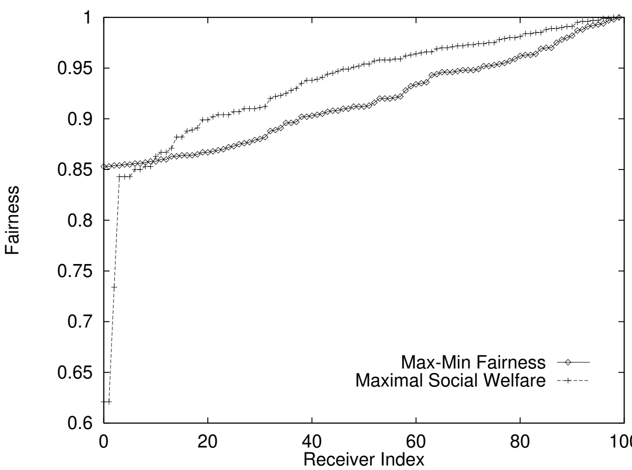

Figure 8: Fairness of Receivers (Exponential)

see the maximal welfare algorithm has a much lower minimum fairness than the max-min fairness algorithm, especially when the number of groups is between 4 and 14. We have similar results in the case of normal distribution.

To take a closer look at the fairness value for other receivers, we plot all elements of the fairness factor in Figures 7 and 8 when the number of groups is 12. We sort the fairness value of receivers in increasing order. The x-axis is the receiver index and the y-axis is the fairness of each receiver. Figure 7 is the result when the distribution of receiver capacity is uniform. The max-min fairness algorithm has larger fairness values than the maximal welfare algorithm when the receiver index is smaller than 5, and therefore it is fairer for those at the lower end of fairness values based on the fairness function used. It is at the expense of smaller fairness value of those receivers having a higher index. Figure 8 shows the result when the distribution is exponential. The max-min fairness has higher fairness values than the maximal welfare when the receiver index is smaller than 8. Though the max-min fairness has a smaller total average of fairness values than the optimal welfare, it is "fairer" based on the max-min criterion.

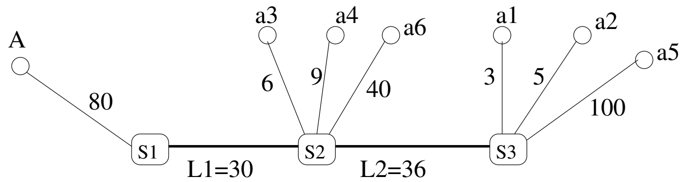

Figure 9: Sharing with receivers in the same multicast session

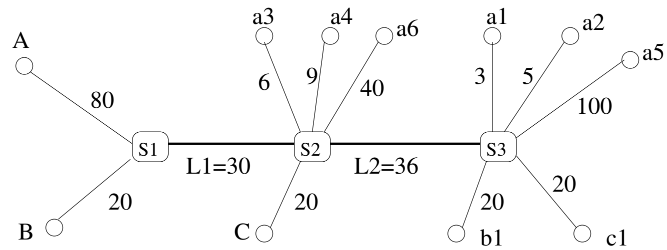

Figure 10: Links shared by receivers from other sessions

5. IMPLEMENTATION ISSUES

One of key issues in the max-min fair allocation is to determine the capacity for each receiver. When the links in the path from the source to a receiver are *not* shared with other unicast/multicast sessions, such as receivers in Figure 9, the capacity of a receiver is the bandwidth of the bottleneck link in the path because this is the maximal rate that the receiver can get data from the source. When the links from the source to a receiver are shared with other sessions, we can use the multi-rate max-min fair allocation to determine the bandwidth at each link that can be used by this multicast session [9]. For example, in Figure 10 we want to know the capacities of receivers in a multicast session with A as the sender and $a1$, $a2$, ..., $a6$ as receivers. The links ($L1$,$L2$) to receivers are shared with other two sessions, one has source B and receiver $b1$ and the other has source C and receiver $c1$. The bandwidth allocated to multicast session A by the multi-rate max-min fair allocation is shown in Figure 11. The capacities for $a1$ to $a6$ are 3, 5, 6, 9, 12 and 18, respectively.

The multi-rate max-min fair allocation can be implemented end-to-end by using multiple multicast groups for each session. If we have enough groups to use, we can achieve their capacity for all receivers. We need to use six multicast groups for session A. However, in practice we have to implement with a limited number of groups and usually this will inevitably result in an underutilization of bandwidth at various points in the network. This is a tradeoff bandwidth utilization for simplicity of management and implementa-

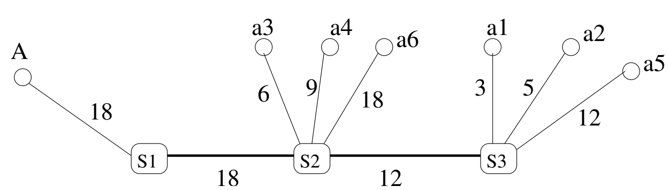

Figure 11: Bandwidth allocated for the session

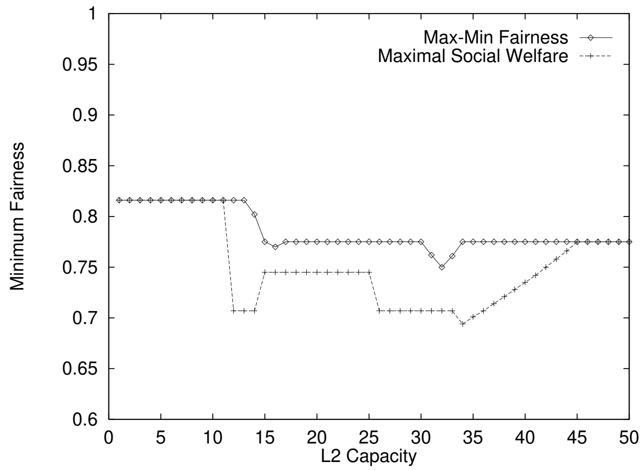

Figure 12: Minimum Fairness

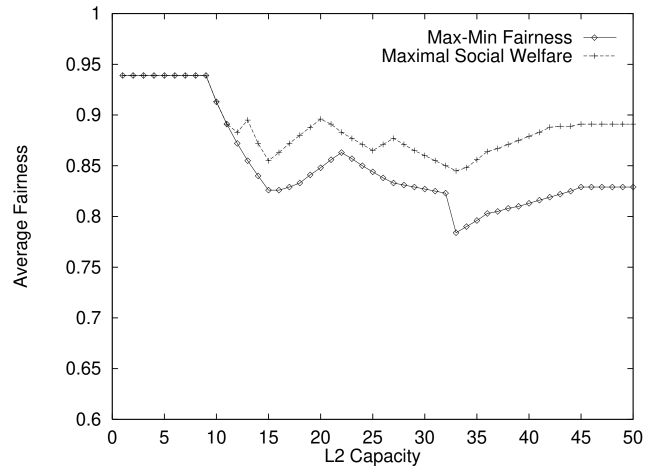

Figure 13: Average Fairness

tion. The problem is: with a given number of groups we can use, how should we divide receivers into groups and determine a rate for each group? We can use the algorithms in the previous section to find the max-min intra-session fair groupings and group rates. For example, if we can only use 3 groups, the optimal solution is to group $a1$ and $a2$ together, $a3$ and $a4$ together, and $a5$ and $a6$ together. Their rates will be set to 3, 6 and 12, respectively.

Setting three multicast groups with rates 3, 6, 12 and having receivers join corresponding groups will result in the total rates over shared links exceeding the bandwidth allocated for the session. For example, the total rate over link $L1$ is 3+6+12=21, higher than the bandwidth allocated to the session, which is 18. For real-time transmission where not all receivers need to get the same data, we propose a layering based solution. If we have m groups of receivers and the max-min intra-session fair allocation results in the group rates from r_1 to r_m, we can implement it as m layers and set the rate of layer i to $r_i - r_{i-1}$ (assume $r_0 = 0$). All the receivers in group i will listen to layer 1 up to layer i. The actual rate of each receiver is still equal to the optimal intra-session fair allocation while the total rate over a given link will be smaller than or equal to the bandwidth allocated. For the example in Figure 11, three layers will be set at rates 3, 3 ($= 6 - 3$), and 6 ($= 12 - 6$). Receivers $a1$ and $a2$ listen to layer 1, $a3$ and $a4$ listen to layers 1 and 2, $a5$ and $a6$ listen to layers 1, 2 and 3. They get the max-min intra-session fair rate and the overall rate at each link is not over the limit.

5.1 Performance Results

We evaluate the performance of the max-min intra-session fair allocation and the maximal social welfare allocation in the environment where links are shared with receivers from other unicast/multicast sessions. The topology used is as shown in Figure 10. The bandwidth for each link is labeled next to it, except that we vary the bandwidth of link $L2$ from 1 to 50. The goal is to use three layers to implement the multicast session with receivers $a1$ to $a6$. We show the results using the max-min fairness algorithm and the maximal welfare algorithm, with f_2 as the individual fairness function.

Figure 12 shows the minimum individual fairness value over all receivers. When the bandwidth of link $L2$ is less than 10, there is no difference because receivers $a1$, $a2$, and $a5$ are all constrained by link $L2$ and has the capacity of 3. When the bandwidth of $L2$ is between 10 and 45, we see significant differences between these two algorithms. Figure 13 shows the average fairness of receivers. We can see that the maximal welfare algorithm gets a higher average fairness value, or better social welfare. However, we find that better social welfare is achieved at the cost of less intra-session fair, especially for some unlucky receivers in the session. Through these two plots, we can see the tradeoff between the max-min intra-session fairness and the maximal social welfare. While social welfare is an important measure for grouping multicast receivers and rate assignment, we cannot ignore the intra-session fairness if we want to be fair among receivers in the same session.

6. CONCLUDING REMARKS

In this paper we gave a formal definition of intra-session fairness and illustrated its essential difference from social welfare. Intra-session fairness is a very important concept usually ignored in practice. In the case of single group sessions, we gave optimal solutions to the max-min intra-session fairness problem for many fairness function definitions to set an appropriate group rate. In the case of multiple group sessions, we presented an algorithm framework that can generate optimal solutions to the problem of grouping receivers to achieve the max-min intra-session fairness. The novelty of the solution is that it does not depend on any specific individual fairness function definition. We also discussed implementation issues and showed the difference in fairness achieved by the maximal social welfare and the max-min intra-session fair allocations.

The future work includes investigating the heterogeneous fairness function definitions. It is possible that different receivers want to use different fairness functions. The problem becomes much more complicated if the fairness function is in a complicated form, which may involve some special utility each receiver is interested in. Our perception is that we may not have a general solution and have to do an application specific analysis to find an intra-session fair solution to the problem.

Acknowledgment

We would like to thank anonymous reviewers for their valuable comments that help improve the paper.

7. REFERENCES

[1] M. H. Ammar and L.-R. Wu. Improving the throughput of point-to-multipoint ARQ protocols through destination set splitting. In *Proceedings of INFOCOM'92*, pages 262–71, May 1992.

[2] S. Bhattacharyya, J. Kurose, D. Towsley, and R. Nagarajan. Efficient rate-controlled bulk data transfer using multiple multicast groups. In *Proceedings of IEEE INFOCOM'98*, pages 1172–1179, March 1998. San Francisco, CA.

[3] Z. Cao and E. W. Zegura. Utility max-min: An application-oriented bandwidth allocation scheme. In *Proceedings of IEEE Infocom'1999*, 1999.

[4] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[5] J. M. Jaffe. Bottleneck flow control. *IEEE Transactions on Communications*, 29:954–962, July 1981.

[6] T. Jiang, M. H. Ammar, and E. W. Zegura. Inter-receiver fairness: A novel performance measure for multicast ABR sessions. In *Proceedings of ACM SIGMETRICS 98/Performance 98*, pages 202–211, June 1998. Madison, Wisconsin.

[7] K. Kar, S. Sarkar, and L. Tassiulas. Optimization based rate control for multirate multicast sessions. In *Proceedings of IEEE Infocom'2001*, April 2001. Anchorage, Alaska.

[8] S. Kasera, G. Hjalmtysson, D. Towsley, and J. Kurose. Scalable reliable multicast using multiple multicast channels. *IEEE/ACM Transactions on Networking*, 8:294–310, June 2000.

[9] D. Rubenstein, J. Kurose, and D. Towsley. The impact of multicast layering on network fairness. In *Proceedings of ACM SIGCOMM'99*, September 1999. Cambridge, MA.

[10] H.-Y. Tzeng and K.-Y. Siu. On max-min fair congestion control for multicast ABR service in ATM. *IEEE Journal on Selected Areas in Communications*, 15:545–556, April 1997.