Fair Allocation of Utilities in Multirate Multicast Networks

Saswati Sarkar and *Leandros Tassiulas*
Dept. of Electrical and Computer Engineering and Institute for Systems Research
University of Maryland, College Park, MD, USA

Abstract

We study fairness in a multicast network. We assume that the source hierarchically encodes its signal and the hierarchical structure is predetermined. We study fair allocation of utilities, where utility of a bandwidth can be the number of layers or the bandwidth itself, or any other function of the bandwidth depending on system requirements. The utility function is not strictly increasing in general. Fairness issues become vastly different in this case as opposed to that when the utility function is strictly increasing. Computation of lexicographically optimal utility allocation becomes NP-hard in this case, while lexicographically optimal utility allocation is polynomial complexity computable when the utility function is strictly increasing. Furthermore, maxmin fair utility allocation may not exist in the general case. We introduce a new notion of fairness, maximal fairness. We propose a polynomial complexity algorithm for computation of maximally fair utility allocation. Even though, maximal fairness is a weaker notion of fairness, it coincides with lexicographic optimality and maxmin fairness, when maxmin fair utility allocation exists. So the algorithm for computing maximally utility allocation computes maxmin fair utility allocation, when the latter exists.

1 Introduction

Multicasting poses some specific fairness challenges. The fairness objective is that every receiver receives service at a rate commensurate with its capabilities and the capacity of the path leading to it from the source. A single rate of transmission per session is likely to either overwhelm the slow receivers or starve the fast ones, in absence of additional provisions. Multirate transmission should be used to counter network heterogenity. Multirate transmission can be attained using a hierarchical or a layered transmission scheme. In this approach, a signal is encoded into a number of layers that can be incrementally combined to provide progressive refinement. The receivers adapt to congestion by adding and dropping layers.

We study fair allocation of utilities, where utility of a bandwidth can be the number of layers, the bandwidth itself or any other function of the bandwidth, e.g., perceptual quality. Maxmin fairness[1] is a well accepted notion of fairness. A utility allocation is maxmin fair, if no receiver can be allocated a higher utility without hurting another receiver having equal or lower utility. A maxmin fair utility allocation may not exist if the utility functions are not strictly increasing. However, a maxmin fair utility allocation always exists for strictly increasing utility functions. Lexicographic optimality is another notion of fairness. A lexicographically optimal utility vector is one which maximizes its minimum component in a feasible set, subject to this maximization, it maximizes the second minimum, etc[1]. In general, lexicographically optimal utility allocation exists even when the utility function is not strictly increasing. However, computation of lexicographically optimal utility allocation is an NP-hard problem for non-strictly increasing utility functions. Lexicographically optimal utility allocation is identical to the maxmin fair utility allocation, when the latter exists and is thus polynomial complexity computable for strictly increasing utility functions, though. We can compute a maximally fair utility allocation for non strictly increasing utility functions, instead. We introduce the concept of maximal fairness more formally later, but a utility allocation is maximally fair if no other utility allocation is "fairer" in some sense. That is, if a utility allocation is maximally fair, then to increase the utility received by a receiver s, we must lower that of another receiver j to a value less than the new utility of s. If a utility allocation is maximally fair, then any other utility allocation will be "unfair" or "less fair" to some receiver. Maximal fairness is a weaker notion of fairness as compared to maxmin fairness and lexicographic optimality. But, maximal fairness has various desirable fairness properties,e.g., it coincides with maxmin fairness and lexicographic optimality when maxmin fair utility allocation exists. We discuss other desirable fairness properties of maximally fair allocations later. In a nutshell, maximal fairness is probably the best we can achieve for non-strictly increasing utility functions, in view of the nonexistence of maxmin fair utility allocation and computational complexity of lexicographically optimal utility allocation. We will present a polynomial complexity algorithm for computing maximally fair utility allocation in this paper. This algorithm yields a maxmin fair utility vector, if it exists. Our algorithm for computation of maximally fair allocation does not assume any properties specific to internet or ATM. So it is applicable in a very general scenario. Keeping in mind ATM networks, we have incorporated minimum utility requirements and maximum utility constraints in our model.

We review related work in this area. Well known network protocols for layered transmission, RLM (Receiver-driven Layered Multicast)[8] and LVMR (Layered Video Multicast with Retransmissions)[5] do not handle fairness among sessions very well, when there are multiple sessions competing for bandwidth[6]. An alternative layer allocation scheme has been proposed in [6]. There is empirical evidence that this scheme improves fairness among sessions for networks with multiple video sessions sharing only one link. But, there is no experimental or analytical evidence that the scheme works well for more complex networks, with sessions sharing several links with each other. Besides [6] does not compute actual rates or the number of layers allocated to the receivers in an arbitrary network, under some well defined notion of fairness. An algorithm for computation of maxmin fair rates, when hierarchical signal structure is completely flexible, has been proposed in [10]. Recently, [11] addressed the scenario with hierarchical signal structure predetermined. But, it considers the special case in which bandwidth consumed by every layer is the same irrespective of the particular layer or the source. It points out that maxmin fair layer allocation may not exist and computation of lexicographically optimal layer allocation is NP-hard even in this special case. It also presents an algorithm for computation of maximally fair layer allocations. However this algorithm does not yield a maximally fair layer allocation, in a more general case i.e., if different layers consume different bandwidths, or if the network can allocate any rate in a particular bandwidth range and only a discrete set of bandwidths in another range. The situation becomes

[1]Maxmin fairness and lexicographic optimality will be defined formally later.

significantly different in this general case. The general case is relevant in the current day networking scenario. For example, every additional layer induces an exponential increase in the total bit rate in the codec of [7]. A study of fair allocation of utilities will capture this general scenario. Incidentally, [3] presents an algorithm for computation of maxmin fair utilities. However the algorithm in [3] has been presented for unicast network only, whereas we consider multicast network. More importantly, the algorithm attains maxmin fair utilities for strictly increasing utility functions only. The operation of this algorithm is ambiguous for nonstrictly increasing utility functions and does not even generate a maximally fair utility allocation in general[11]. Again there are many examples of practical importance in which the utility functions do not increase strictly.

2 Allocation of Utilities

We first motivate the study of fair allocation of utilities. The hierarchical signal structure is often predetermined and different for different sources. For example, a network can have two sessions and a single link. Source of the first session transmits unit bandwidth layers. Every layer of the second session consumes 2 units of bandwidth. Let the capacity of the link be 9 units. A bandwidth allocation allocating 5 units of bandwidth to the first session and 4 units to the second one is intuitively fair (predetermined layer bandwidths do not allow allocation of 4.5 units each). However, this allocation gives session 1, 5 layers and session 2, 2 layers. Intuitively, $(3, 3)$ is a "fairer" layer allocation. The corresponding bandwidth allocation, $(3, 6)$ does not seem fair. So, fair allocation of layers and bandwidths are different objectives. We think the fairness objective should depend on the specific requirements of particular networks. So we study fair allocation of utilities, where utility of a bandwidth can be the number of layers, the bandwidth itself or any other function of the bandwidth, e.g., perceptual quality.

There can be various possibilities in hierarchical coding. The hierarchical structure may be completely flexible, i.e., given the desired rates of various receivers, the source generates as many layers as distinct receiver rates and tunes the layer bandwidths so as to match the receiver rates. The hierarchical structure may be predetermined, i.e., layer bandwidths can not be tuned to match any desired receiver rates and receivers can not partially subscribe to layers. Also, it is possible that layer bandwidths can be fine tuned in certain ranges, while the granularity is coarse in other ranges. Hierarchical structure of different sources may be different, i.e., signal structure of some sessions may be flexible, while the signal structure may be predetermined for some others. Bandwidths consumed by layers may be different for different sources. Bandwidths consumed by different layers of the same source may also be different. We model all these scenarios by choosing the utility functions appropriately. We discuss the choice of utility functions later, but in general, the utility functions will not be strictly increasing. Fairness under this assumption is vastly different from the case in which the utility functions are strictly increasing.

Formally, "utility" is a function of the bandwidth. We assume that the utility function for receivers of the same session is the same. "Utility set" of session i, A_i is the range of the utility function of session i. Receivers of a session i can possibly receive a utility if it belongs to utility set of session i, A_i. We assume that the utility function satisfies the following technical assumptions for every i.

1. $A_i \cap (x, \infty) \neq \phi$, $\forall x \in R$.

2. Given any $y \geq 0$, $\max(A_i \cap [0, y])$ exists.

3. If $x \notin A_i$, then $\min(A_i \cap (x, \infty))$ exists.

4. Given any utility x, there always exists a minimum bandwidth required for session i to attain utility x, i.e., the set $\{y : u_i(y) = x\}$ has a minimum, where $u_i(y)$ is the session i utility corresponding to bandwidth y. The minimum bandwidth for utility x is denoted $\Upsilon_i(x)$.

The various scenarios described before, e.g., hierarchical structure flexible, partly flexible and partly predetermined, completely predetermined, etc. can be modelled by choosing the utility set suitably for every session i. We describe a few possible choices in Figure 1(a). The first assumption indicates that the utility set is not upper bounded. If necessary, we can impose maximum utility constraints separately. The other assumptions are more technical. Assumption (2) indicates that if there exists elements in a utility set arbitrarily close to but less than a positive real number z, then z is an allowable utility. Assumption (3) indicates that, if a real number x is not an allowable utility then the set of utilities greater than x has a minimum. The last three assumptions exclude some functions from being valid utility functions. Refer to Figure 1(b) for details.

Observe that the utility function is non strictly increasing in many cases(Figure 1(a)). Any utility function satisfying properties (1) to (4) is acceptable. The assumptions are general in nature and hence a large choice of utility functions are acceptable. Also the utility function of receivers of the same session must be the same. This assumption is reasonable, because utility structure, (e.g., hierarchical structure) often depends on source and there is only one source per session.

3 Network Model

We consider an arbitrary topology network with N multicast sessions. A multicast session is identified by the pair (v, U), where v is the source node of the session and U is the group of intended destination nodes. We assume that the traffic from node v is transported across a predefined multicast tree to nodes in U. The tree can be established during connection establishment phase if the network is connection oriented or can be established by some well known multicast routing protocol like DVMRP[4], CBT[2], etc. in an internet type network.

We call every source destination pair of a session a virtual session. For example, if a session n has source v_n and destination set U_n, where $U_n = \{u_{n1}, \ldots, u_{nt}\}$, then this session would correspond to t virtual sessions, $(v_n, u_{n1}), \ldots, (v_n, u_{nt})$. Refer to Figure 2 for an example of sessions and virtual sessions. To ensure fairness in a multirate network, we need to consider fair utility allocation for the virtual sessions separately, instead of those for the overall sessions.

An $M-$ dimensional vector $\vec{r}$ is a feasible utility allocation if

1. $r_s \in A_{\chi(s)}$, $\forall\ s \in \{1, \ldots, M\}$, where $\chi(s)$ is the session of virtual session s.

2. $\mu_i \leq r_i \leq p_i$ $\forall i$, where μ_i and p_i are respectively the minimum and maximum utiliities of virtual session i, $p_i \geq \mu_i \geq 0$,

3. Total bandwidth consumed in every link is less than or equal to the capacity of the link. Bandwidth consumed in a link is the sum of the bandwidths consumed by

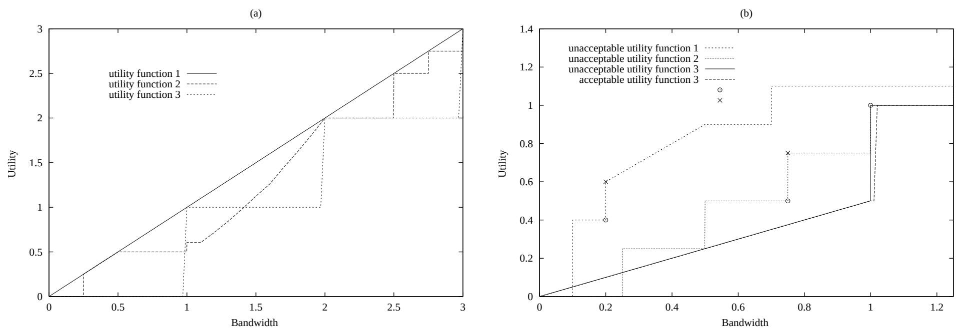

Figure 1: Figure (a) shows three utility functions. For first and third, utility of bandwidth x is the actual bandwidth which can be allocated when x units of bandwidth is available. Utility function 1 models the case when hierarchical structure is completely flexible and any bandwidth can be allocated, if it is available. The utility function is linear and the utility set is the nonnegative real line. Utility function 3 models the case when hierarchical structure is completely predetermined. Only a discrete set of rates can be allocated and the utility function is stair-case. Utility set is the set of nonnegative integers. Utility function 2 models the case when hierarchical structure is flexible in certain ranges of bandwidth and predetermined in other ranges. Here, the utility is the qualitative gain from the bandwidth. In the flexible region, function increases strictly, linearly or sublinearly. In the predetermined region, it is a stair-case function. Note that the step jumps are unequal, indicating unequal layer bandwidths for the same source. The utility set consists of closed intervals and some discrete points. In general, utility can also be the number of available layers when a ceratin bandwidth is available.
Figure (b) shows three unacceptable utility functions. The first utility function (unacceptable utility function 1) is not acceptable with data point 'o.' The utility set violates property (3). However the utility function becomes acceptable if the data point is replaced by the one marked, 'x'. Similarly, the second utility function (unacceptable utility function 2) is not acceptable with data point 'o', (property (4) is violated). Again the utility function becomes acceptable if the data point is replaced by the one marked, 'x'. The utility function for the third one (unacceptable utility function 3), with data point 'o' does not satisfy property (2). However, a small modification indicated by "acceptable utility function" satisfies property (2).

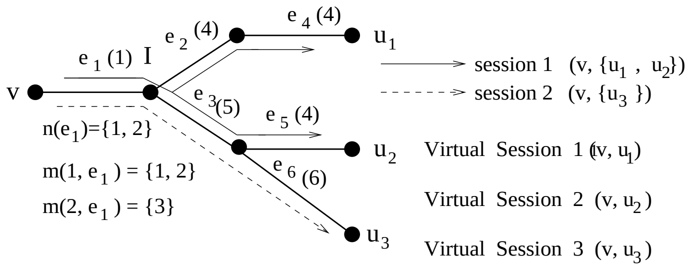

Figure 2: Session 1 includes virtual sessions, $1, 2$. Session 2 includes virtual session 3. The figure shows some sample $n(e_i)$ and $m(i, e_j)$s. The numbers in brackets, () denote the capacities of the respective links. Let the utility of bandwidth x be the actual bandwidth allocated, if x units of bandwidth is available, for each session. The capacity constraint for link e_1 is $\lambda_{1e_1} + \lambda_{2e_1} \leq 1$ and that for link e_3 is $\lambda_{1e_3} + \lambda_{2e_3} \leq 5$. Here $\lambda_{1e_1} = \max(r_1, r_2)$, $\lambda_{2e_1} = \lambda_{2e_3} = r_3$, $\lambda_{1e_3} = r_2$. The minimum and maximum utility constraints are $0.6 \leq r_1 \leq 5$, $0.3 \leq r_2 \leq \infty$ and $0 \leq r_3 \leq 5$. Let the utility sets be the nonnegative real line for both sessions. The utility allocation $(.6, .6, .4)$ is the maxmin fair, lexicographic optimal and maximally fair allocation. Now let the utility sets consist of nonnegative integers only for both sessions and let the minimum and maximum utility requirements not be there. There is no maxmin fair utility allocation. The utility allocation $(1, 1, 0)$ is lexicographically optimal. The utility allocations $(1, 1, 0)$ and $(0, 0, 1)$ are both maximally fair.

different sessions traversing the link. Bandwidth consumed by a session in a link is the bandwidth consumed by the *session link utility*, where session link utility is the maximum utility allocated to virtual sessions of the same session traversing the link. Let $n(l)$ denote the set of sessions traversing link l, $m(k, l)$ denote the set of virtual sessions of session k passing through link l, C_l denote the capacity of link l, λ_{il} denote the session link utility of session i in link l and $\Upsilon_i(x)$ denote the minimum bandwidth required for session i to attain utility x . Then $\lambda_{il} = \max_{j \in m(i,l)} r_j$ and

$$\sum_{i \in n(l)} \Upsilon_i(\lambda_{il}) \leq C_l \qquad \text{(capacity condition)}.$$

Figure 2 illustrates an example network with a few capacity and maximum and minimum utility constraints.

A feasible utility allocation is maxmin fair if it is not possible to maintain feasibility and increase the utility of a virtual session without decreasing that of any other virtual session which has equal or lower utility. More formally, a feasible utility allocation vector $\vec{r}^1$ is maxmin fair if it satisfies the following property with respect to any other feasible utility allocation vector $\vec{r}^2$: if there exists i such that the ith component of $\vec{r}^2$ is strictly greater than that of $\vec{r}^1$ $(r_i^2 > r_i^1)$, then there exists j such that the jth component of $\vec{r}^1$, r_j^1 is less than or equal to the ith component of $\vec{r}^1$, r_i^1 $(r_j^1 \leq r_i^1)$ and the jth component of $\vec{r}^2$ (r_j^2) is strictly less than the jth component of $\vec{r}^1$ $(r_j^2 < r_j^1)$. A maxmin fair utility allocation may not exist in general for non strictly increasing utility functions. Refer Figure 2 for an example.

Given a $M-$ dimensional vector $\vec{V}$, define its lexicographically ordered version $\hat{V}$ as follows: $\hat{v}_j = v_k$ for some k and $\hat{v}_1 \leq \hat{v}_2 \leq \ldots \hat{v}_M$. In other words, components of $\hat{V}$ are an ordered version of those of $\vec{V}$. A vector $\vec{r}^1$ is lexicographically greater than another $\vec{r}^2$ if there exists i such that $\hat{r}_i^1 > \hat{r}_i^2$ and $\hat{r}_j^1 = \hat{r}_j^2$ if $j \leq i$. Vectors $\vec{r}^1$ and

$\vec{r}^2$ are lexicographically equal if $\hat{r}^1 = \hat{r}^2$. Vector $\vec{r}^1$ is lexicographically less than $\vec{r}^2$, if $\vec{r}^2$ is lexicographically greater than $\vec{r}^1$. A utility allocation vector $\vec{r}$ is lexicographically optimal if it is feasible and if every feasible utility allocation vector is lexicographically less than or equal to $\vec{r}$. Refer Figure 2 for an example of a lexicographically optimal utility allocation vector. A lexicographically optimal utility allocation vector always exists, under our assumptions on the utility set. However, its computation is NP-hard. We proved this in [11], while considering layer allocation vectors in hierarchical networks with predetermined hierarchical structure and every layer consuming the same bandwidth irrespective of the source and the particular layer. This is a special case of what we consider here (we can look upon allocated layers as utility of a bandwidth). So the result holds in this case as well.

We suggest an alternate notion of fairness, maximal fairness. We use the concept of *relative fairness* introduced in [9]. A utility allocation vector $\vec{r}^1$ is *fairer* than another utility allocation vector $\vec{r}^2$, if for every virtual session i which has greater utility under $\vec{r}^2$ than under $\vec{r}^1$, there is some other virtual session j whose utility was already no more than that of i under $\vec{r}^1$, and has been decreased further by $\vec{r}^2$. A more formal definition of relative fairness follows.
A utility allocation vector $\vec{r}^1$ is *fairer* than another layer allocation vector $\vec{r}^2$ if

- $\vec{r}^1 \neq \vec{r}^2$ and
- if there exists an i such that $r_i^1 < r_i^2$, then there exists a j such that $r_j^1 \leq r_i^1$ and $r_j^2 < r_j^1$ (r_k^l is the kth component of $\vec{r}^l$, $l = 1, 2$.)

A utility allocation vector $\vec{r}$ is maximally fair if it is feasible and if no other feasible utility allocation vector is fairer than $\vec{r}$. Refer Figure 2 for an example of a maximally fair utility allocation vector.

As we show later, maximally fair utility allocation exists and can be computed in polynomial complexity. But, first we investigate, whether maximal fairness is a good notion of fairness. Note that from the definition of maximal fairness, it is clear that if a utility allocation is maximally fair, then we can increase the utility of a receiver s, only by decreasing that of another receiver to a value below that of the new utility value of receiver s. Thus we can increase the utility of s only by being unfair to some other receiver. Also, if maxmin fair vector exists in any feasible set, then it is the only maximally fair vector[11]. So any algorithm for computation of a maximally fair utility allocation will yield a maxmin fair utility allocation, if one exists. Besides, any lexicographically optimal vector is maximally fair in any feasible set[11]. So lexicographically optimal vector is one of the maximally fair ones, in any feasible set. Thus maximal fairness is a good notion of fairness.

Maximum utility constraints can be incorporated by adding artificial links between receivers with maximum utility constraints and the rest of the network. Capacity of such an artificial link is equal to the bandwidth consumed by the maximum number of utilities of the respective receiver. So henceforth we shall ignore the maximum utility constraints.

4 Algorithm for Computation of Maximally fair Utility Allocation

Now we present an algorithm for computation of maximally fair utility allocation. First, we introduce certain terminologies. $\chi(s)$, $n(l)$ and $m(i, l)$, are defined in page 4 respectively.

Ψ_i is a real valued function defined as $\Psi_i(x) = \inf_{y \in A_i \cap (x,\infty)} y$, $\forall\ i$. Loosely speaking, $\Psi_i(x)$ is the smallest session i utility greater than x. If the utility set of session i, is a set of integers, then $\Psi_i(x) = \lceil x \rceil$ if x is not an integer and $\Psi_i(x) = x + 1$, otherwise. This function is well defined by property (1).

$\Phi_i(x)$ is a real valued function defined as $\Phi_i(x) = \max_{y \in A_i \cap [0,x]} y$. Thus, $\Phi_i(x)$ is the largest session i utility not exceeding x. If the utility set of session i, is a set of integers, then $\Phi_i(x) = \lfloor x \rfloor$. This function is well defined by property (2).

The minimum session link utility, μ_{il} is the maximum of the minimum utilities of session i virtual sessions traversing l, i.e., $\mu_{il} = \max_{s \in m(i,l)} \mu_s$.

L_s is the set of links traversed by virtual session s.

A virtual session s is saturated w.r.t. a utility vector $\vec{r}$ if it traverses a link l in which

$$\begin{aligned} r_s &= \lambda_{\chi(s)l} \\ \text{If } r_s = \Psi_{\chi(s)}(r_s),\ \sum_{i \in n(l)} \Upsilon_i\left(\lambda_{il}\right) &= C_l \\ \text{If } r_s < \Psi_{\chi(s)}(r_s),\ \sum_{i \in n(l)} \Upsilon_i\left(\lambda_{il}\right) &> C_l - \Big(\Upsilon_i\left(\Psi_{\chi(s)}(r_s)\right) - \Upsilon_i\left(r_s\right)\Big) \end{aligned}$$

A session is *saturated* on a link l if all the virtual sessions of the session traversing the link l are saturated.

$\eta_l(k)$ denotes the link control parameter of link l at the end of the kth iteration. Link control parameter is the iterate which would be used in computation of the maximally fair utilities. An utility equal to $\eta_l(k)$ would have been allocated to the unsaturated virtual sessions traversing the link, if there were no bandwith constraints on other links, and feasible utility allocations need only satisfy the capacity constraints.

$\eta_{il}(k)$ denotes the session link parameter of session i traversing link l. It is the utility assigned to session i, if there were no bandwidth constraints for any of its virtual sessions on other links and feasible utility allocations need only satisfy capacity and minimum utility constraints (i.e., utility set is the nonnegative real line).

$\omega_s(k)$ is the utility that is assigned to virtual session s at the end of the kth iteration if it is restricted to receive no more than any of its session link control parameters, $\eta_{\chi(s)l}(k)$, on its path. Here, it is the largest valid utility in $A_\chi(s)$ not exceeding, its minimum session link control parameter on its path.

$\Omega_{il}(k)$ is the utility allocated to session i in link l, under the utility vector $\vec{\omega}(k)$. $\Omega_{il}(k) = \max_{j \in m(i,l)} \omega_j(k)$.

$r_s(k)$ is the utility allocated to virtual session s at the end of the kth iteration. $\vec{r}(k)$ denotes the utility vector at the end of the kth iteration, with components, $r_s(k)$.

$\lambda_{il}(k)$ is the utility allocated to the session i in link l at the end of the kth iteration. It is actually the maximum of the utilities allocated to the virtual sessions in $m(i,l)$ at the end of the kth iteration.

$\Lambda(k)$ is the set of virtual sessions which are saturated w.r.t. utility allocation $\vec{\omega}(k)$.

$S(k)$ denotes the set of unsaturated virtual sessions at the end of the kth iteration.

$\Xi_l(k)$ denotes the set of unsaturated sessions traversing link l at the end of the kth iteration.

$F_l(k)$ denotes the total bandwidth consumed by the saturated sessions traversing link l at the end of the kth iteration.

The algorithm follows.

1. $k = 0$, $\eta_l(0) = 0$, $F_l(0) = 0$, $\Xi_l(0) = n(l)$ $\forall$ link l, $S(0) = \{1, \ldots, M\}$, $r_j(0) = \mu_j$, $\forall j \in S(0)$, $\lambda_{il}(0) = \max_{j \in m(i,l)} r_j(0)$.
2. $k \rightarrow k+1$
3. For every link l in the network compute the link control parameter. For this, first compute a variable $\alpha_l(k)$. If $\Xi_l(k-1) \neq \phi$, then $\alpha_l(k)$ is computed as follows:

   $$\alpha_l(k) = \sup_\theta \{\theta : F_l(k-1) + \sum_{i \in \Xi_l(k-1)} \Upsilon_i \left(\max\left(\Phi_i(\theta), \lambda_{il}(k-1)\right)\right) \leq C_l\}$$

   (It can be shown that the above set is nonempty for all k.) If $F_l(k-1) + \sum_{i \in \Xi_l(k-1)} \Upsilon_i \left(\max\left(\Phi_i(\alpha_l(k)), \lambda_{il}(k-1)\right)\right) \leq C_l$, then link control parameter for a link l is $\eta_l(k) = \alpha_l(k)$, else link control parameter, $\eta_l(k)$ is chosen s.t. $\eta_l(k) < \alpha_l(k)$ and $\eta_l(k)$ satisfies some other technical properties. For all unsaturated sessions i passing through link l, $(i \in \Xi_l(k-1))$, $\eta_{il}(k) = \max(\eta_l(k), \lambda_{il}(k-1))$.
4. Compute $\omega_s(k)$ for all virtual sessions $s \in S$, where $\omega_s(k) = \Phi_{\chi(s)}(\min_{l \in L_s} \eta_{\chi(s)l}(k))$, if $s \in S(k-1)$, else $\omega_s(k) = r_s(k-1)$.
5. For every link l in the network, compute $\Omega_{il}(k)$ for every session i in $n(l)$, $\Omega_{il}(k) = \max_{s \in m(i,l)} \omega_s(k)$.
6. Compute the set of virtual sessions saturated during the kth iteration under utility vector $\vec{\omega}(k)$, $\Lambda(k)$.
7. If $\Lambda(k) \neq \phi$, compute the utilities allocated to the virtual sessions after the kth iteration, via, $r_s(k) = \omega_s(k)$, $\forall s$ and go to step (9).
8. Select a virtual session $s \in S(k-1)$ which satsfies *certain* technical properties and compute $r_j(k)$ for all virtual sessions j as

   $$r_j(k) = \begin{cases} \omega_j(k) & j \neq s \\ \Psi_{\chi(j)}\left(\omega_j(k)\right) & \text{otherwise.} \end{cases}$$

9. For every link l in the network compute the session link utility in link l, for every session in $n(l)$, $\lambda_{il}(k) = \max_{s \in m(i,l)} r_s(k)$.
10. Compute the set of virtual sessions unsaturated after the kth iteration, $S(k)$.
11. If $S(k) = \phi$, i.e., all virtual sessions are saturated, the algorithm terminates, else go to the next step.

12. For every link l, compute the set of unsaturated sessions passing through link l at the end of the kth iteration, $\Xi_l(k)$

13. For every link l, for which $\Xi_l(k) \neq \phi$, compute the bandwidth consumed by the saturated sessions passing through link l, $F_l(k) = \sum_{i \in \Xi_l(k)} \Upsilon_i \left(\lambda_{il}(k)\right)$.

14. Go to step (2).

At every iteration k, the algorithm computes a "fair utility share" for every session i, the session link parameter, $\eta_{il}(k)$. Since a virtual session can receive utilities in utility set A_i only, utility $\Phi_i\left(\eta_{il}(k)\right)$ is offered to all virtual sessions of session i traversing the link. A virtual session is allocated the minimum utility offered on its path. If no virtual session saturates in the current iteration, then residual bandwidth may be used to increment the utility of some virtual session traversing this link and satisfying the properties mentioned in step (8). The utility of such a virtual session is increased to the next higher value. This is done because otherwise, the algorithm can continue forever. This is because in the next iteration, the same link control parameter will be computed and the process repeats again and again. A session is saturated if all its virtual sessions are saturated. The bandwidth consumed by the saturated sessions, if any, are computed. This bandwidth is subtracted from the link capacity, and the link control parameters are recomputed at the beginning of every iteration as per step (3) and the process continues. The algorithm determines which virtual session and sessions are saturated. A new iteration starts if still there are unsaturated virtual sessions. Upon termination, we have a maximally fair utility allocation.

We assume the bandwidth function Υ_i is right continuous and strictly increasing ($\Upsilon_i(x)$ is the minimum bandwidth required to attain session i utility x). Continuity is intuitive because if valid utilities in A_i are very close, then the minimum required bandwidths should also be very close. Strictly increasing property of the minimum bandwidth is also intuitive. This does not preclude the same utility corresponding to a range of bandwidth and hence non strictly increasing utility functions, because we only require the minimum bandwidth necessary for attaining a particular utility to be strictly increasing. The following results hold under this assumption.

Theorem 1 (Generalized Maximal Fairness) *If the generalized maximal fairness algorithm terminates, then the output utility allocation vector is*

1. *maximally fair,*

2. *maxmin fair, if a maxmin fair utility allocation exists.*

Theorem 2 (Finite-Termination Theorem) *The algorithm terminates in at most $M + |\mathcal{L}|M^2$ number of iterations, where $\mathcal{L}$ is the set of links and M is the number of virtual sessions.*

Every step of this algorithm has a complexity of $O(|\mathcal{L}|M)$. The algorithm must terminate in $M + |\mathcal{L}|M^2$ iterations. Thus the overall complexity of this algorithm is $O(|\mathcal{L}|^2M^3)$.

5 Conclusion and Discussion

Our algorithm is amenable to distributed implementation. The criteria for determination of utility of a virtual session uses information along the path of the virtual session mainly. The only place where the algorithm uses global information is that $r_s(k) = \Psi_{\chi(s)}(\omega_s(k))$, for at most one virtual session, s. This feature of the algorithm is not crucial to the proof of maximal fairness of the output and is a matter of convenience. This increase in utility can be carried out for multiple virtual sessions, subject to feasibility and as long as they satisfy the criteria of step (8) and the algorithm will still output a maximally fair utility allocation.

In special cases, the algorithm terminates faster. If the utility functions are strictly increasing for all sessions, then $\Psi_i(x) = x$, for all i, x. In this case, the algorithm terminates in at most M iterations and yields a maxmin fair utility allocation. So the algorithm complexity is $O(\mathcal{L}M^2)$. If the utility function is a staircase function for every session i, and the ratio between the largest step jump for all sessions and the smallest step jump for all sessions is p, then the algorithm terminates in at most $M + p|\mathcal{L}|M^2$ iterations. So the algorithm complexity is $O(p\mathcal{L}^2M^2)$. We would like to point out that we made no assumptions on upper bound or lower bound for step jumps of utility functions.

References

[1] D. Bertsekas and R. Gallager, *Data Networks*, Englewood Cliffs, NJ:Prentice-Hall, 1987.

[2] T. Ballardie, P. Francis, and J. Crowcroft. Core based trees: an architecture for scalable inter-domain multicast routing, *Proceedings of ACM SIGCOMM,* Sept. 1993, pp. 85-95

[3] Z. Cao and E. Zegura. Utility Max-Min: An Application-Oriented Bandwidth Allocation Scheme *IEEE INFOCOM'*99

[4] S. Deering and D. Cheriton. Multicast routing in datagram internetworks and extended LANs, ACM Transactions on Computer Systems, vol 8, no. 2, pp. 54-60, Aug. 1994.

[5] X. Li, S. Paul and M. Ammar. Layered Video Multicast with Retransmission (LVMR): Evaluation of Hierarchical rate control, *Proceedings of IEEE Infocom' 98*, March 1998

[6] X. Li, S. Paul and M. Ammar. Multi-session Rate Control for Layered Video Multicast *Technical Report* GT-CC-98 – 21, College of Computing, Georgia Institute of Technology, '98

[7] S. McCanne, Scalable Compression and Transmission of Internet Multicast Video. *PhD thesis*, Univ. of California, Berkeley, December 1996.

[8] S. McCanne, V. Jacobson and M. Vetterli. Receiver-Driven Layered Multicast, *Proceedings of ACM SIGCOMM '96*, Stanford, CA, September 1996.

[9] S. Sarkar and K. N. Sivarajan: Fairness in Cellular Mobile Networks *Proceedings of the 34th Annual Allerton Conference on Communication, Control and Computing,*1996*, pp* 457 – 469

[10] S. Sarkar and L. Tassiulas: Distributed Algorithms for Computation of Fair Rates in Multirate Multicast Trees *Technical Report* TR 99 – 42, Institute for Systems Research and University of Maryland, 1999.

[11] S. Sarkar and L. Tassiulas: Fair Allocation of Discrete Bandwidth Layers in Multicast Networks *Technical Report* TR 99 – 43, Institute for Systems Research and University of Maryland, 1999.