Multi-Layer Utilization Maximal Fairness for Multi-Rate Multimedia Sessions

Paulo Mendes*# mendes@cs.columbia.edu

Henning Schulzrinne# schulzrinne@cs.columbia.edu

Edmundo Monteiro* edmundo@dei.uc.pt

*Department of Informatic Engineering
University of Coimbra
3030 Coimbra, Portugal

#Department of Computer Science
Columbia University
New York, NY 10027, USA

Abstract

We present a fairness model, called *Multi-Layer Utilization Maximal Fairness* (MLUM). The motivation of the MLUM model is to accomplish intra-session and inter-session fairness in the presence of multi-rate (multi-layer) multimedia sessions, distributing bandwidth between multicast trees of different sessions, considering the number of receivers in each session, and improving bandwidth efficiency. To accomplish this goal, the model is divided in three components: a fairness definition, a policy and a protocol. The MLUM definition guarantees fairness between receivers in a session and fairness between different sessions, considering the number of receivers in each session. The MLUM policy implements the MLUM definition in multicast aware border routers of Autonomous Systems (AS). It's functionality is divided in the control plane and the data plane. In the control plane, sessions fair rates are estimated based upon the MLUM fairness definition. In the data plane, a queueing discipline will fairly distribute bandwidth between sessions, considering their fair rates. The MLUM protocol allows the exchange of control information (sessions number of receivers and fair rates) between MLUM policy routers in order to accomplish a fair distribution of bandwidth between concurrent sessions. This paper aim to present the MLUM fairness definition, to describe the MLUM fairness policy control and data plane functionality and to evaluate how the policy data plane can fairly distribute bandwidth. We present some simulations that evaluate the performance of the MLUM policy data plane in different scenarios and that compare its performance to other queueing disciplines. We also briefly describe the MLUM protocol, due to its close relation with of the MLUM policy functionality.

Keywords: fairness, multicast, multi-rate, quality adaptation.

1 Introduction

There is an increase number of applications where sources send information to multiple receivers. These applications can be divided considering the number of receivers and their geographical location. For example, Internet TV , radio and video on demand are applications with large and dense groups of receivers, while video and audio conference, virtual classrooms and multiparty network games are applications with small and sparse groups of receivers. Multicast is the most appropriate delivery systems for these two types of applications since it creates a delivery tree that connects a source with multiple receivers, without duplicating information, thus resulting in better bandwidth usage. The IP multicast model, namely the recent Source Specific Multicast (SSM) approach [Holbrook01], was developed to achieve this scalability and efficiency in multi-party communication systems, being particularly adapted for applications with large and dense groups of receivers. To avoid the substantial infrastructure modifications of IP multicast, when applications with small and sparse groups of receivers are used, other approaches exist [Chu00][Pendarakis01], which position the multicast functionality at the end systems, creating an application-level multicast middleware, introducing however duplication of packets and larger end-to-end delays than IP multicast.

Independently of the multicast model used by multimedia applications, the Internet lacks mechanisms that fairly distribute bandwidth between multi-rate multimedia sessions. How to achieve a fair distribution of bandwidth between different sessions, allowing receivers to get the required quality level without wasting resources in the network, is still a challenging research topic.

Many fairness definitions for multicast networks assume that multimedia sessions are single-rate, in which receivers have equal capabilities. However multi-rate (multi-layer) approaches, like hierarchical video, are more realistic in a heterogeneous network as the Internet, since they allow receivers to get data at different rates accordingly to their capabilities and the capability of the path between them and the source. In multi-layer approaches, fairness can be divided in *intra-session fairness*, related to members of the same session and *inter-session fairness*, related to receivers of different sessions.

Intra-session fairness can be achieved using layered multicast protocols [McCanne96]. In such protocols, the multi-layer approach consists of dividing multimedia streams in several layers, each one with a different rate, and sent to different multicast addresses. The set of layers comprising a multimedia stream constitutes a *multimedia session.* These sessions can have a fixed or dynamic set of layers and layer rates. However, hierarchical video codes aren't usually amenable to use dynamic layer bandwidths due to their inherent structure. For example, in certain video codes sub-streams can be extracted to produce only a specific range of resolutions, like the 3D Subband Video Coding [Taubman94]. Although the fairness model presented in this paper can deal with any number of layers per sessions, this codes limitations were considered in the simulations, where we only use multi-layer sessions with three layers.

However, multi-layer approaches can include: i) significant and persistent quality instability; ii) problems with synchronization of receivers; iii) unfairness between sessions; iv) low bandwidth utilization efficiency, due to layers granularity. A receiver-driven adaptive mechanism, based upon the work described in this paper, can be used to solve the first two points. Trying to contribute to the solution of the last two points, we present in this paper a *Multi-Layer Utilization Maximal (MLUM)* fairness model, which is divided in three components: a fairness definition, a policy and a protocol. The MLUM definition guarantees intra and inter-session fairness. The MLUM policy implements the MLUM definition in multicast aware border routers of Autonomous Systems (AS). It's functionality is divided in two parts: the control plane and the data plane. In the control plane, sessions fair rates are estimated based upon the MLUM fairness definition. In the data plane, a queueing discipline will fairly distribute bandwidth between sessions, considering their fair rates. The MLUM protocol allows the exchange of control information (sessions number of receivers and fair rates) between MLUM policy routers. However, fairness policies based only upon the number of receivers could not lead to an optimal fair system due to bandwidth waste, when the bandwidth fair share assigned to a session is higher than the rate really used by that session. An example of this is when mobile phone or personal digital assistant (PDA) sessions, which have low rate requirements, have the highest number of receivers. Therefore, the MLUM fairness definition besides assigning bandwidth between sessions based upon the session number of receivers, also increase resources efficiency utilization trying to approximate to one the ratio between sessions rate and sessions fair rate.

The MLUM model has two main motivations. First, multicast delivery to multiple receivers requires less bandwidth than using unicast, mainly when considering applications with large and dense groups of receivers. Second, using an efficient fair distribution of bandwidth between multicast trees will increase Internet Service Providers (ISP) incentive to deploy multicast since their number of clients can increase maintaining a low utilization of network resources. Multimedia clients will have an increased incentive to choose ISP that deploy this kind of fairness policies, because clients can get a higher quality when they belong to large multimedia sessions, probably for a equal of lower price. And since clients have a higher incentive to form large groups, this mean a higher profit for multimedia content providers, like TV and radio stations.

We won't attempt to present an optimal fairness definition, since there isn't only one correct definition of fairness, because social and economic issues can influence this definition as much as technical ones. These issues normally lead ISP to implement different fairness policies, considering their business strategy. However we based the MLUM fairness definition in the number of receivers in each multimedia session and the bandwidth utilization efficiency, since these seems to us as a base fair utility function. The main idea behind this option is to use this definition as a base for a hierarchical fairness structure, which can be build adding more fairness utility functions, like for example price functions.

We used the network simulator NS [NS], version 2.1b7a, to evaluate the MLUM model behavior. The first results, presented in this paper, show the impact of the MLUM fairness policy data plane behavior (layers classification, layers average rate estimation, layers marking and queueing discipline) on sessions rate. The MLUM policy data plane performance was also compared with other queuing disciplines, like FIFO, RED, FRED, CSFQ and DRR.

The remaining of the paper is divided as follows. Section 2 presents a brief description of the fairness definitions analyzed. Section 3 describes the MLUM fairness definition and policy. Section 4 presents the first simulation results, concerning the MLUM policy data plane functionality. Section 5 briefly describes the MLUM protocol, and section 6 presents some conclusions and future work.

2 Related Work

To obtain an intra and inter-fairness definition well adapted to multi-rate (multi-layer) sessions, several fairness approaches were analyzed.

Between all the studied fairness definitions, max-min is the most well-accepted definition to guarantee fairness between receivers in single-rate sessions. Although the max-min definition was extended to include multi-layer sessions, it's efficient only when continuous set of rates are being used. [Rubenstein99] presents a study of how layering impacts max-min fair allocations within large-scale multicast networks. This study shows that in the presence of a discrete set of rates, as used in hierarchical video codes, max-min fairness can not exist. Besides the study of multi-layer sessions, the analyzed work doesn't mention the issue of inter-session fairness.

The maximal fairness definition presented in [Sankar00] exist in the presence of a discrete set of rates, but it doesn't consider the number of receivers served by each multimedia session. This way a policy implementing this fairness definition can't distribute available bandwidth in order to satisfy the quality requirements of as many receivers as possible.

[Legout99] presents a proposal to distribute bandwidth between sessions considering the number of receivers in each one. Although this proposal considers multi-layer sessions, the number of receivers in each session layer isn't considered in the bandwidth distribution. Therefore this proposal don't maximize the number of receivers that has its quality requirement satisfied. Another disadvantage of this proposal is that it assumes a fix number of flows in each link, static group of receivers and that all routers keep per-flow state.

A proposal to improve inter-session fairness between multilayer sessions is presented in [Li99]. This approach uses the max-min fairness definition, which was proved in [Rubenstein99] that can not exist in the presence of discrete set of rates, as is the case of multi-layer sessions. Besides this, this proposal only works when sessions share only one link and don't consider, neither the number of receivers in each session, when distributing the link bandwidth, nor the sessions layers different priorities, when dropping packets in the link.

Since none of the analyzed approaches provides intra and inter-fairness between multi-layer sessions with different number of receivers in a distributed environment, we became motivated to present a different fairness model to accomplish this goal.

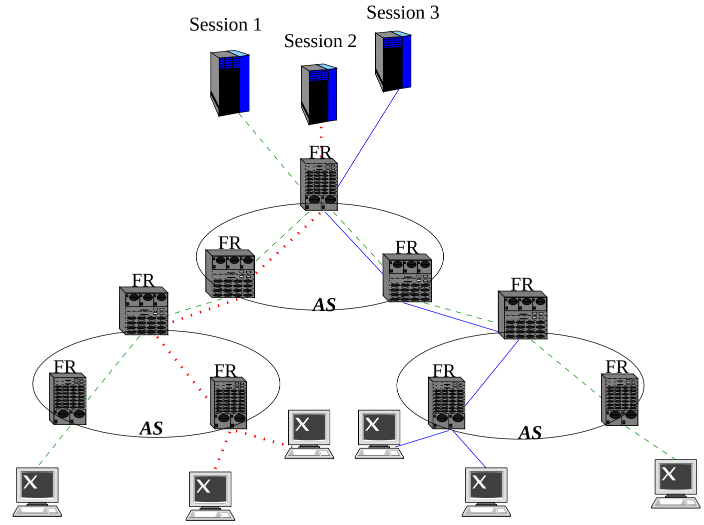

Figure 1: Position of fairness routers (FR)

3 MLUM fairness model

This section describes two parts of the MLUM model: the fairness definition and how it's implemented in routers, the MLUM policy. Sub-section 3.1 describes the MLUM definition and how it can guarantee intra-session and inter-session fairness. Subsection 3.2 describes the functionality of the MLUM policy data plane and control plane. In section 5 of this paper, we'll briefly describe the third component of the MLUM model, the MLUM protocol. Although, the aim of this paper don't include the study and evaluation of the protocol, its brief description will allow a better comprehension of how the MLUM model can achieve a fair distribution of bandwidth between concurrent sessions multicast trees.

The MLUM policy has two characteristics that improves its scalability. First, the policy is only implemented in Autonomous Systems (AS) border routers, which we'll denominate in the rest of this paper as fair routers (FR). Second, the policy doesn't require knowledge about all FR, needing only to know the fair rates of the previous upstream and downstream FR and the downstream number of receivers for each session. In the MLUM model we assume, as shown in figure 1, that all multicast branch points should only exist in FR routers, and not in the interior of an AS. Besides this, FR routers will consider as maximum bandwidth in each downstream interface, the minimum bandwidth of all links until the next FR router. The estimation of this bandwidth is beyond the scope of this paper.

If Differentiated Services networks (DiffServ) [RFC2475] are considered, it's assumed that all multicast layers from the same session use the same DiffServ service inside an AS. In this case each FR will distribute the bandwidth assigned to each DiffServ service between sessions that use that service. In DiffServ networks, sessions using services based upon Assured Forwarding classes (AF) can have their layers mapped to three different drop precedence. In the MLUM model packets are marked only with two drop precedences, as is explained in the sub-section 3.2.2.

3.1 MLUM Fairness Definition

The MLUM fairness definition is well-adapted to be used with multi-layered sessions, such as hierarchical video and audio sessions. However, it can also be used with non-hierarchical multicast sessions and with unicast sessions, which are treated as a particular case of the general functionality. In the former case, sessions are considered to have only one layer with several receivers, and in the latter case to have only one layer with one receiver[1]. This behavior makes it possible to use the MLUM model with any kind of sessions.

In the MLUM fairness definition, a session S_i is defined as having one sender and several receivers. The sender uses a multi-layer approach, being each layer sent to a different multicast address. Receivers try to subscribe to as many layers as they are allowed by the quality of the path between them and the sender. Therefore a session S_i can be represented in terms of its sender and its receivers by $(X_i, \{y^1_{(s_i,l_1)}, \dots, y^n_{(s_i,l_n)}\})$, where X_i is the session sender and $y^k_{(s_i,l_k)}$ is the session k^{th}receiver, which has subscribed l_k layers. The number of receivers that a session S_i has in a link j, a_{ij}, is equal to the number of receivers in the session base layer, since all receivers will subscribe at least the base layer, l_0, of the joined session, as shown in equation.

$$a_{ij} = \sum_k y^k_{(S_i,l_0)} \tag{1}$$

The session S_i fair rate in a link j, F_{ij}, is defined as the ratio between the session number of receivers in the link j, a_{ij}, and the total number of receivers in that link, considering the bandwidth capacity[2] until the next FR router, C_j. This is shown in equation 2, where n is the number of sessions that share the link j.

[1]All multicast sessions with more than one receiver will have a fair rate higher than unicast sessions.

[2]The bandwidth capacilty is equal to the link capability if the network isn't a DiffServ network, or the capacity of the used DiffServ classe otherwise.

$$F_{ij} = (\frac{a_{ij}}{\sum_{\delta=1}^{n} a_{\delta j}})C_j \quad (2)$$

In a multi-session environment the fair rate allocation between sessions in a link j, is represented by the vector $V_j(F_{1j}, ..., F_{nj})$. This fair rate allocation vector is only feasibility if the sum of the fair rates of all sessions in that link doesn't exceed the bandwidth capacity C_j.

The relation between a session fair rate, in the vector V_j, and its real rate defines the session quality in a link. The session S_i rate in the link j depends of the session number of layers and their rates. Equation 3 estimates the session rate in a link j, r_{ij}, considering the number of layers S_i has in the link, l_{ij}, and the rate of each session layer l_k has, $r_{(s_i,l_k)}$. This equation shows that the session rate variation is discrete, with a granularity dependent of the layers rate.

$$r_{ij} = \sum_{k=0}^{l_i} r_{(s_i,l_k)} \quad (3)$$

Therefore, considering a session S_i rate and fair rate in a link j, we can define that session quality factor in the link, Q_{ij}, as the difference between its fair rate and its rate.

$$Q_{ij} = F_{ij} - r_{ij} \quad (4)$$

A session is said to be efficient when it has a quality factor of zero. A quality factor higher than zero is an indication that the session fair share isn't being totally utilized, which means that there is the opportunity to increase receivers quality or to use the unused bandwidth to increase the quality of another session. A quality factor lower than zero is an indication of packet loss, because the session rate is higher than its fair share. In this case, since session S_i fair rate isn't enough to satisfy the quality requirements of its receivers in the less significant layer l_k, this layer rate will be used by other sessions. However, session S_i receivers in the layer l_{k-1} will maintain their quality level, i.e., the session quality factor without the less significant layer, Ql_{ij}, will be preserved. This quality valour is defined in equation 5 for a session S_i in a link j, being l_k its last layer.

$$Ql_{ij} = F_{ij} - (r_{ij} - r_{(s_i,l_k)}) \quad (5)$$

The MLUM fairness aim to maximize bandwidth utilization increasing the number of receivers with satisfied quality requirements. To accomplish this, sessions are classified by the number of receivers they have in their less significant layer, i.e., by their importance. We say that a session S_i is more important than a session S_k in a link j, if S_i has more receivers in the less significant layer than S_k. This is shown in equation 6, where a_i^i and a_k^k are the number of receivers in S_i and S_k less significant layers, and I_{ij} and I_{kj} are these sessions importance in the link j.

$$\forall i,k \in [1,n],\ I_{ij} > I_{kj} \Rightarrow a_i^i > a_k^k \quad (6)$$

Since the number of receivers in each layer decrease from the base layer to the less significant layer, if a session S_i is more important than a session S_k, it has always more receivers than session S_k.

Therefore to maximize the bandwidth utilization increasing the number of receivers with satisfied quality requirements, the estimated fair rate allocation vector $V_j(F_{1j}, \ldots, F_{nj})$ in each link has to comply with the following MLUM fairness definition.

MLUM fairness definition: Consider that F_{ij}, Q_{ij}, Ql_{ij} and I_{ij} are respectively the fair rate, quality factor, quality factor without the less significant layer and importance of a session S_i on a link j, as given by equations 2, 4, 5 and 6. A rate allocation vector $V_j^1(F_{1j}^1, \ldots, F_{nj}^1)$ is said to be *Multi-Layer Utilization Maximal* fairer, if it's feasible and if for any alternative feasible rate allocation vector $V_j^2(F_{1j}^2, \ldots, F_{nj}^2)$:

$$\begin{gathered}\forall i \in [1,n],\ F_{ij}^2 > F_{ij}^1 \wedge Q_{ij}^2 \geq 0 \Rightarrow \exists k \in [1,n], \\ F_{kj}^2 < F_{kj}^1 \wedge Ql_{kj}^2 < 0 \vee (I_{kj} > I_{ij} \wedge Q_{kj}^2 < 0 \leq Q_{kj}^1)\end{gathered} \quad (7)$$

This MLUM fairness definition allows the fair rate of a session S_i in a link j to be increased in order to get a higher quality factor is the following conditions are satisfied. First, if there aren't sessions S_k less important than S_i, which quality factor Ql_{kj} becomes lower than zero. Second, if there aren't sessions S_k more important than S_i, which quality factor Q_{kj} becomes lower than zero.

3.2 MLUM Fairness policy

In this section we'll describe the MLUM policy that implements the fairness definition in each FR router. The policy functionality is divided in two parts: the control plane and the data plane. In the control plane sessions fair rates are estimated based upon sessions current number of receivers. In the data plane, a queueing discipline will fairly divide bandwidth between sessions, considering their fair rates.

3.2.1 MLUM policy control plane

In the control plane, FR routers estimate the fair rate allocation vector $V_j(F_{1j}, \ldots, F_{nj})$ for each downstream link. For that, they need information about sessions number of receivers, sessions upstream and downstream fair rates and sessions layers average rate. They get the first two from their upstream and downstream FR neighbors using the MLUM protocol. Local average rates are collect from the policy data plane.

To estimate the fairer rate allocation vector, FR routers maintain, for each downstream link, a list with information about the sessions present on that link. For each session the local, downstream and upstream fair rates will be stored, as well as the number of receivers in each session layer. The position of each session in the list is determined by its importance, being the list ordered by decrease order of session importance. This ordering is useful for the estimation of the fairer rate allocation vector. Each FR will also maintain a list for each upstream link, only with the indication of what sessions have their senders in that upstream link. Each session stored information occupy 8 bytes in the upstream link list and 64 bytes in the list of each downstream link where the session is present. The information about each layer that a session has in each downstream link occupy 36 bytes in the downstream list. Therefore, if for example a FR router has 3 downstream links, the space occupied by 1000 sessions in each link, each session with 3 layers, will be 516 KB.

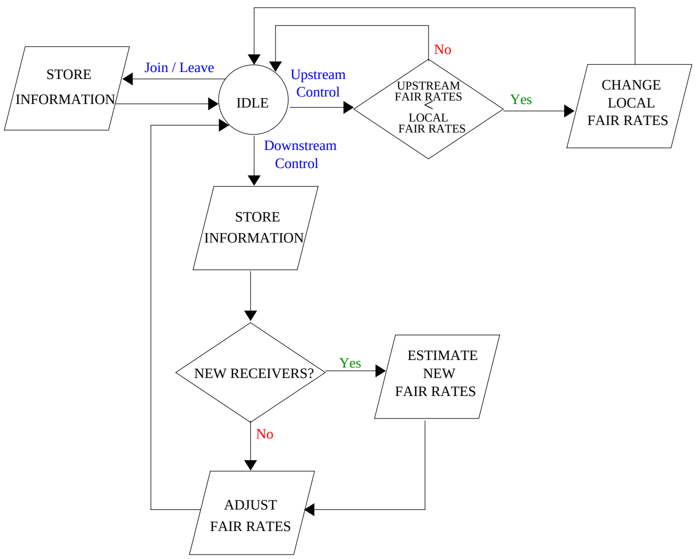

Figure 2: FR operations in the control plane.

Figure 2 presents a simplified description of the control plane operations a FR has to perform each time the downstream or upstream session information changes. Every time there are changes in the downstream number of receivers (receivers in the base layer of the downstream sessions), FR routers will use equation 2 to estimate the fair rate for each affected session. Then, sessions fair rate will be adjusted, in order to increase the efficient usage of bandwidth and the number of receivers with satisfied quality requirements. The adjustment is done in two steps. First, all sessions that have a fair rate higher than their downstream fair share will see their fair rate decreased to their downstream fair rate level, increasing the local available bandwidth. Second, the local available bandwidth is used to effectively increase the quality of sessions with local fair rate lower than their downstream fair rate, or lower than their estimated average rate. When the local available bandwidth is zero, and there are still sessions with a negative quality factor, equation 7 will be used to increase sessions quality, starting by the most important sessions. The fair rate of a session S_i can only be increased if its quality factor, Q_{ij}, is lower than zero and its fair rate, F_{ij}, is lower than the session downstream fair rate[3], $F_{i(j-1)}$. It should also be possible to increase Q_{ij} to positive values considering the link available bandwidth, the Q_{kj} of all sessions S_k more important than S_i and the Ql_{kj} of all sessions S_k less important than S_i. If S_i quality factor can only become positive if the quality factor of sessions less important than S_i become negative, S_i fair rate is only increased if the number of satisfied receivers increase by a pre-configured value. This way the scalability of the fairness policy is improved, because the number of adjustments is kept low[4]. If all theses conditions are satisfied, the fair rate of a session S_i is increased until its quality factor, Q_{ij}, becomes zero. The transference of bandwidth to session S_i is first done using the available bandwidth, then using the fair rate of less important sessions and finally using the fair rate of more important sessions, under the conditions shown in equation 8.

$$F_{ij} = \begin{cases} min(F_{ij-1},\, F_{ij} + ab_j)\,,\; ab_j > 0 \\ min(F_{ij-1},\, F_{ij} + Ql_{kj})\,,\; ab_j = 0 \wedge Ql_{kj} \geq 0 \wedge I_{kj} < I_{ij} \\ min(F_{ij-1},\, F_{ij} + Q_{kj})\,,\; ab_j = 0 \wedge Q_{kj} \geq 0 \wedge I_{kj} > I_{ij} \end{cases} \tag{8}$$

In this equation, ab_j is the available bandwidth on link j, F_{ij-1} is the fair rate of session S_i on the link *j-1* (downstream the local FR), Q_{kj} is the quality factor of session S_k in link j *and* Ql_{kj} is the quality factor of session S_k in link j without the session less important layer l_k.

When upstream fair rates change, the fair rate of the modified sessions S_i are updated with the minimum value between the local previous estimated fair rate, F_{ij}, and the upstream fair rate, F_{ij+1}. This is illustrated in equation 9. This way the local fair rate vector won't be higher than the upstream fair rate vectors, which leads to a better bandwidth utilization in all sessions multicast trees.

$$F_{ij} = min(F_{ij+1},\, F_{ij}) \tag{9}$$

After the adjustment shown in equation 9, the bandwidth made available is distributed between all sessions S_i with negative quality factor, Q_{ij}, starting with the most important ones. A session S_i fair rate is only increased if it's possible to increase Q_{ij} to positive values with the available bandwidth and if its final fair rate will be lower than S_i upstream and downstream fair rate.

In order to optimize the performance of the MLUM policy control plane, each time the local sessions information is updated, the number of operations made, in order to adjust sessions fair rates, isn't proportional to the total number of sessions and layers locally stored. When sessions downstream fair rates and number of receivers is updated, the number of operations is proportional to the number of sessions changed and the number of sessions with a negative quality factor. When upstream sessions fair rate is updated, the number of operations is proportional to the number of sessions changed. Also, when downstream changes in the number of receivers don't concern the base layers, FR routers won't estimate new fair rates, adjusting only the existing fair rates if the importance order of the local sessions is changed.

3.2.2 MLUM policy data plane

In the MLUM policy data plane, FR routers implement a queueing discipline to fairly distribute bandwidth between sessions in

[3] F_{ij} should always remain lower than $F_{i(j-1)}$

[4] There are no rate adjustment between sessions with approximatly the same number of receivers.

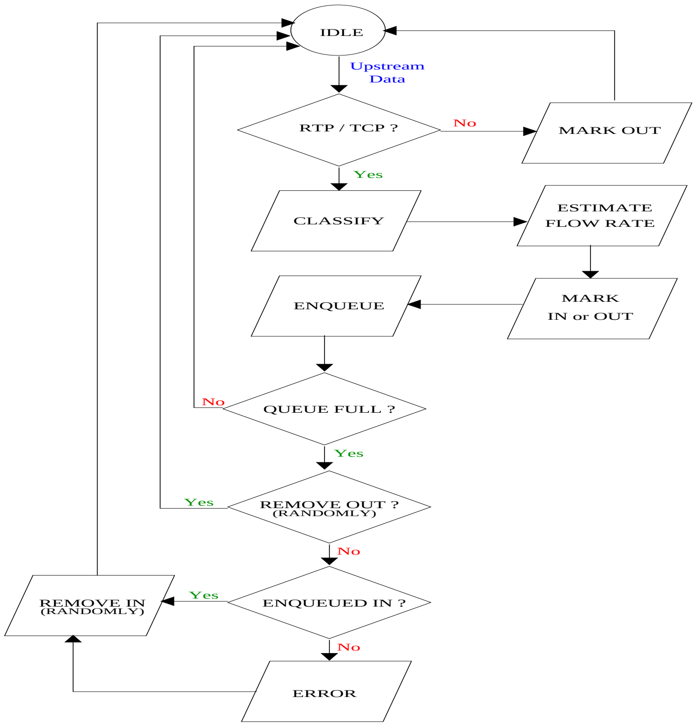

Figure 3: FR operations in the data plane

each downstream link. To accomplish this, FR routers estimate the sessions layers average rate and use the sessions fair rates, estimated in the control plane. The functionality of the policy data plane is illustrated in figure 3. Each time data arrives from one of the upstream links, this data is first classified in order to identify its session and layer. After this, the average rate for this session layer is estimated. The actualized average rate and the session fair rate are used to mark data packets in order to protect well-behaved traffic. This marking process is required since sessions with a rate higher than their fair rate, i.e., with a quality factor, Q_{ij}, lower than zero, contributes to the link congestion. In this situation, all packets corresponding to a rate higher than the session fair rate is said to be *out* of profile. Punishing sessions with *out* of profile packets will increase the incentive to use adaptation mechanisms on end hosts, in order to decrease rates to values lower than sessions fair rates. The use of rates lower than sessions fair rate, i.e., *in* profile rates, decrease the probability of having congested links and increase sessions quality. Several punishment strategies can be adopted, considering they effectiveness and complexity.

In what concerns the layers average rate estimation, the easier way is to ask for the layers rates directly to the sessions sources. However this could lead to an unfair situation, because sources can mislead the MLUM policy saying that they have a lower rate than they actual have, trying to get a higher fair rate. This misleading is possible since the adjustment of fair rates depends upon the actual layers rate. Therefore, FR will instead estimate the average layers rate, maintaining the policy independent from sources. The average rate of a layer l_j belonging to a session S_i, $r_{(S_i,l_j)}$, is updated every time a new packet arrives, considering $t^k_{(i,j)}$ and $s^k_{(i,j)}$ as the arrival time and size of the k^{th} packet. A linear (equation 10) or exponential (equation 11) equation can be used to update the arriving rate.

$$r^k_{(S_i,l_j)} = \frac{1}{2}\left(\frac{s^k_{(i,j)}}{T^k_{(i,j)}} + r^{k-1}_{(S_i,l_j)}\right) \qquad (10)$$

$$r^k_{(S_i,l_j)} = \left(1 - e^{-\frac{T^k_{(i,j)}}{Y}}\right)\frac{s^k_{(i,j)}}{T^k_{(i,j)}} + e^{-\frac{T^k_{(i,j)}}{Y}} r^{k-1}_{(S_i,l_j)} \qquad (11)$$

where $T^k_{(i,j)} = t^k_{(i,j)} - t^{k-1}_{(i,j)}$ and Y is a constant.

We use the exponential equation, introduced in [Stoica98], since using a factor of $e^{-\frac{T}{Y}}$ we can control the system stability and the time the average rate estimation takes to converge to the real rate. A small value of Y increases the system responsiveness to rapid rate fluctuations, decreasing the rate convergence time, being however more dependent of the packeting structure. A high value of Y avoids potential system instability, since its more independent of the packeting structure, increasing however the rate convergence time. [Stoica98] proposes that Y should have an average value between 100 and 500ms.

In what concerns marking packets *in* or *out*, FR routers implement a simple punishment strategy, which consists in using sessions fair rate as a punishment boundary. Starting by the base layer (high priority) and ending with the highest enhanced layer (low priority), sessions packets will be marked with high priority while the session rate is *in* profile. When sessions rate become *out* of profile, packets will start be marked with low priority. Equations 12 and 13 give the probability that a layer l_n of a session S_i has to be marked high priority, $P_h(i,n,j)$, and low priority, $P_l(i,n,j)$, in a link j. With this strategy there is also a differentiation between *out* of profile sessions, since *out* of profile sessions with higher rates are more punished.

$$P_h(i,n,j) = \begin{cases} 1 & , L_{in} \leq F_{ij} \\ \frac{M_{in} - L_{i(n-1)}}{r_{(S_i,l_n)}} & , L_{in} > F_{ij} \end{cases} \qquad (12)$$

$$P_l(i,n,j) = \begin{cases} 0 & , L_{in} \leq F_{ij} \\ \frac{L_{in} - M_{in}}{r_{(S_i,l_n)}} & , L_{in} > F_{ij} \end{cases} \qquad (13)$$

In the previous equations F_{ij} is the fair rate of session S_i in a link j and L_{in} and M_{in} are given by the equations 14 and 15.

$$L_{in} = \sum_{k=0}^{n} r_{(s_i,l_n)} \qquad (14)$$

$$M_{in} = max(F_{ij}, L_{i(n-1)}) \qquad (15)$$

A more aggressive punishment strategic could be chosen, being however more complex than the implemented on. This more aggressive strategy can be briefly described as follows. When a session S_i has an average number of *out* of profile packets higher than a pre-configure bound, measure during a period of time Δt, $F_{ij} - \Delta z$ become the fair rate available to that session during the next Δt period of time. The punish value Δz is doubled in the next Δt period of time if the number of *out* of profile packets remain higher than a pre-configure bound. Otherwise, the session fair rate becomes F_{ij} again.

After being marked, each packet is enqueued using the MLUM queueing discipline, which distributes bandwidth fairly between sessions layers, preferentially dropping *out* of profile packets. This enable us to use FIFO scheduling which is simple, easy to deploy and prevents packets reordering (which can happens since a session has packets in more than one layer). The MLUM queueing discipline as a simple functionality. If the queue become full after placing a packet, an *out* of profile packet will be randomly removed from the queue and discarded. If the packet placed in the full queue is an *in* profile packet and the queue don't have *out* of profile packets, then an *in* profile packet will be randomly removed from the queue and discarded. This randomly discard of packets guarantees that flows that have higher rates will be more severely punished, and that *out* of profile packets will be always dropped first.

4 Simulation of the MLUM fairness policy data plane

In this section we present the first simulations results, which allow us to evaluate the MLUM policy data plane. The FR policy control plane is dependent of the MLUM protocol to exchange control information between neighbors FR. Since the MLUM protocol implementation in NS is being finished, we couldn't include results from the MLUM policy control plane functionality in this paper. Therefore session fair rates are manually configured for each simulation made of the MLUM data plane functionality. These first simulations aim to evaluate how the policy data plane distribute bandwidth between sessions with different layer rates and different fair rates. These simulations results are showed in sub-section 4.2. First, the MLUM policy data plane behavior is compared, in sub-section 4.1, with five other queueing disciplines.

Simulations were made using one scenario with one single 10 Mbs congested link shared by N sessions. Each session has three layers, being each layer identified by a source address and a multicast group address, as used in the SSM multicast approach.

4.1 Comparison of MLUM data plane with different queueing disciplines

In this simulation we used FIFO, RED, FRED, DRR and CSFQ queueing disciplines, which can be briefly characterized as:

1. FIFO (First-In-First-Out) - The queue is managed by a drop-tail strategy, packets being served in a first-in-first-out order.

2. RED (Random Early Detection) - In this queue management strategy [Floyd93], packets are dropped probabilistically before the buffer become full, providing early congestion detection.

3. FRED (Flow Random Early Drop) - It's a RED extension that provides some fairness in the distribution of bandwidth [Lin97]. To achieve fairness, FRED maintain in all routers state about all flows that have at least one packet in the buffer. FRED drops packets from flows that either had many packets dropped, or have a queue bigger than the average queue size. We'll use two variations of FRED, which will be denominated FRED-1 and FRED-2. The latter guarantees a minimum number of buffers for each flow, having a better performance than FRED-1 only when there are a high number of flows in the router.

4. CSFQ (Core-Stateless Fair Queueing) - This algorithm [Stoica98] tries to achieve fairness between flows in a continuous region of the Internet, distinguishing between edge routers and core routers. The former computes per-flow rate estimates and label packets passing through them by inserting these estimates into the packet header. The latter use FIFO queueing, keep no per-flow state and use a probabilistic dropping mechanism using the information in the packet labels.

5. DRR (Deficit Round Robin) - This algorithm [Shreedhar95] is an implementation of the weighted fair queueing (WFQ) discipline. This algorithm is the only, from the five studied, that achieved a higher level of fairness, because is the only one to use a sophisticated per-flow buffer management. The buffer manager drops a packet from the flow with the longest queue, when the buffer is full.

These algorithms present different levels of complexity. DRR and FRED have to classify incoming flows while FIFO, RED and CSFQ don't. DRR has also to implement its own packet scheduling algorithm, while all the other use first-in-first-out scheduling. CSFQ has a complexity similar to FRED in the edge routers and to RED in core routers. Compared to these queueing disciplines, MLUM has a complexity lower than CSFQ since the queuing discipline MLUM uses in the edge routers is less complex than FRED.

Although MLUM is the only one to work with multi-layer (multi-rate) sessions with different number of receivers, we had to restrict its functionality in this first simulation, in order to be able to compare it with the other five algorithms in an equal scenario. Therefore in this simulation we used 33 source UDP flows indexed from 0, where $flow_i$ sends $i+1$ times more than its fair share of 303 Kbs. Thus, flow 0 sends at 303 Kbs, flow 1 sends at 606 Kbs and so on. In the MLUM case, we used 33 RTP sessions with 1 layer each. The simulation has the following parameters. The link has a capacity of 10 Mbs, a latency of 1 ms and a buffer of 64 KB. In the case of RED and FRED the first threshold is set to 16 KB and the second one to 32 KB. In MLUM the variable Y, used to estimate the average rate (equation 11), is set to 100 ms, being two times larger than the maximum queueing delay, i.e., $\frac{64\,KB}{10\,Mbs} = 51.2$ ms. In CSFQ the variable K (used to estimate the average rate), K_α (used to estimate the fair rate) and K_c (used in marking the decision of whether the link is congested or not) are also set to 100 ms.

In this simulation DRR is used as the fairness reference, FIFO and RED as the unfair examples and CSFQ as the reference edge/core approach. The goal is to obtain results as near as possible of DRR and similar to CSFQ.

Figure 4 shows the average flow throughput over a 10 s simulation. FIFO, RED and FRED-1 fail to insure fairness, with flows getting a share proportional to their incoming rate, while DRR is extremely effective in achieving a fair bandwidth distribution. CSQF and FRED-2 achieve a less precise level of fairness. We can observe that, even if MLUM isn't being used with multi-layer traffic, its behavior is reasonable fair, significantly closer to DRR. Although MLUM needs to classify flows, as DRR, MLUM uses a simpler queueing management scheme. Simulations show that MLUM also has an average performance similar to CSFQ, having flows throughput between -12% and +6% of the ideal value (DRR) while CSFQ flows throughput are between -11% and +5%. Although CSFQ is one percentile point near to DRR than MLUM, it doesn't work with multi-layer sessions neither consider the number of receivers in different sessions.

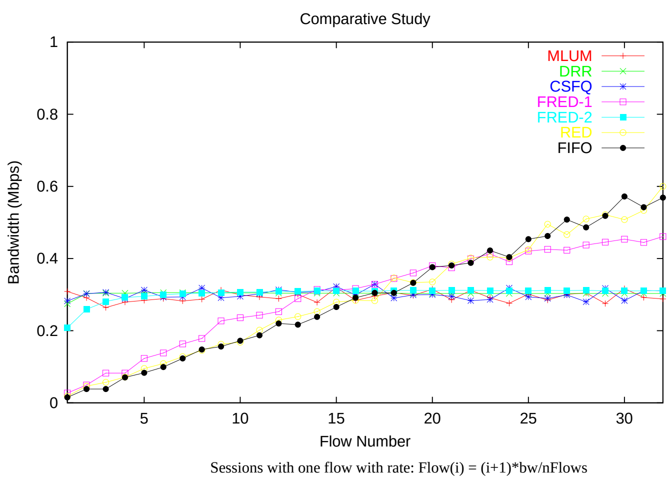

Figure 4: Results for a 10 Mbs link shared by 33 flows.

4.2 Simulation of the MLUM policy data plane with different multi-layer and fair rates schemes

For this simulations, we maintained the simulation parameters described in sub-section 4.1. Since the goal of this second set of simulations is to test the multi-layer, multi-receiver behavior of the MLUM policy data plane, we maintain the existence of 33 RTP sources, but group them in 11 sessions, each one with 3 layers.

We divided the simulations in nine scenarios considering the range of fair rates for the 11 sessions and the rate range of the 33 layers.

In what concerns the range of fair rates, we used three scenarios. FR-increase, in which fair rates for each session start at 659 Kbs, for session 1, and increase at 50 Kbs per session. FR-equal, in which all sessions have the same fair rate of 909 Kbs. And FR-decrease, in which fair rates for each session start at 1159 Kbs, for session 1, and decrease at 50 Kbs per session.

For each one of the fair rate scenarios, we used three types of sources rate. Rates-equal, in which sessions layers have the same rate equal to 403 Kbs. Rates-369, in which sessions have the same rate, but layers have different rates, with layer 0 having 303 Kbs, layer 1 having 606 Kbs and layer 2 having 909 Kbs. Rates-multiple, in which sessions rates are multiples of 50 Kbs starting with session 1. Inside each session a $layer_i$ as a rate equal to twice the rate of $layer_{i-1}$, being the rate of layer 0 equal to the session rate.

Figures 5 to 7 show the first group of simulation results, where the MLUM policy was configured with increasing fair rates for each session, starting at 659 Kbs for session 1, and increasing at 50 Kbs per session.

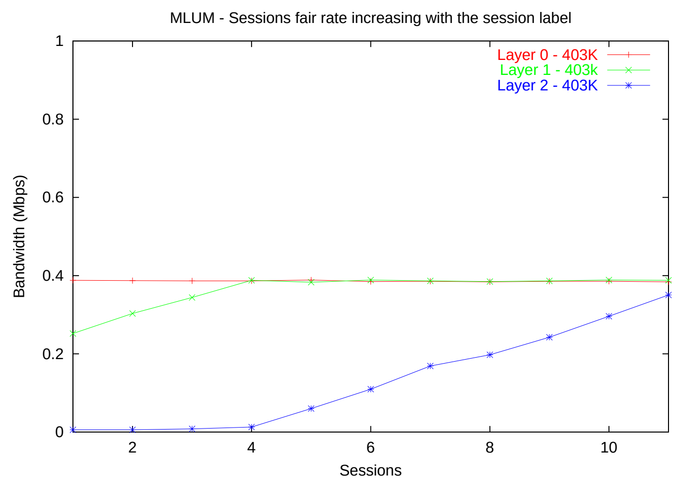

Figure 5: FR-increase and Rates-equal.

In figure 5 all sessions are punished since they have a rate of 1209 Kbs (3 layers with 403 Kbs), which is higher than any session fair rate. Session 1 is the most punished since it has the lowest fair rate, 659Kbs. Therefore, this session has all layer 2 (less priority layer) packets dropped, and layer 1 only have 256 Kbs of *in* profile traffic. Starting at session 4, all sessions have 0% dropping in layer 1, since their fair rates start to be higher than 859 Kbs (session 4 value).

In figure 6 sessions also have equal rates (1818 Kbs in this case), but layers have different rates, having the higher priority layer (layer 0) a rate of 303 Kbs, layer 1 a rate of 606 Kbs and the less priority layer (layer 2) a rate of 909 Kbs. Simulations results are similar to the ones showed in figure 5, but there are more sessions with a higher punishment level, since sessions rate are also higher. Besides the different layer rates, sessions continue to be punished respecting layers priorities. In this simulation layer 1 starts to have 0% dropping only after session 6, because the rate of layer 0 plus layer 1 is higher than the scenario of figure 5.

In figure 7 sessions rates increase, starting at 50 Kbs for session 1 and ending at 550 Kbs for session 11. Each layer has a rate two times higher than the previous layer rate. So, for

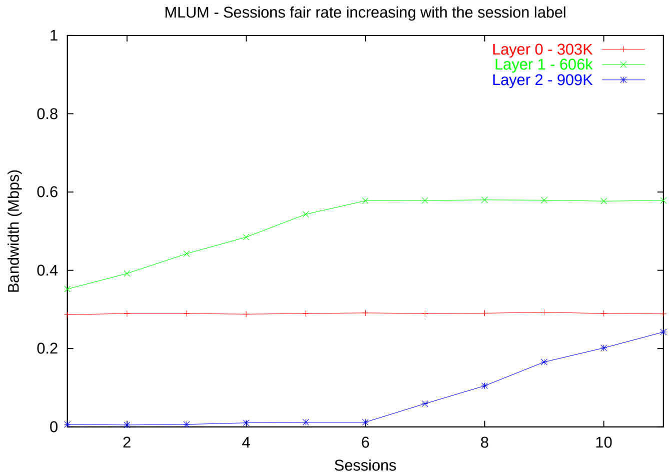

Figure 6: FR-increase and Rates-369.

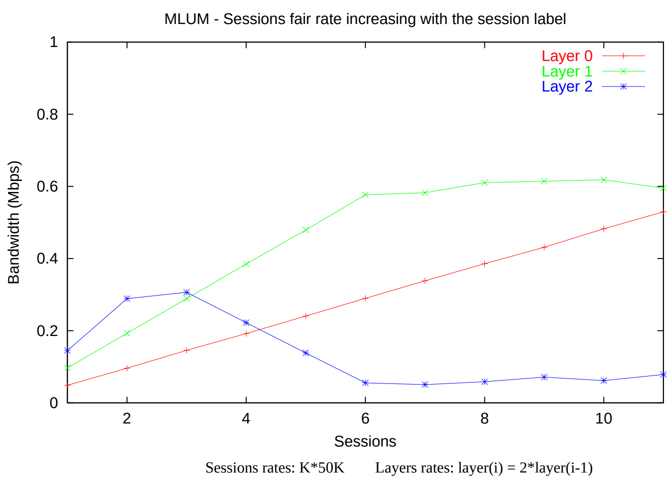

Figure 7: FR-increase and Rates-multiple.

example, in session 1, layer 0 has 50 Kbs, layer 1 has 100 Kbs and layer 2 has 150 Kbs, while in session 11, layer 0 has 550 Kbs, layer 1 has 1100 Kbs and layer 2 has 2200 Kbs. In this simulation session 1 and session 2 don't use all their fair rate. Session 1 don't use 359 Kbs of its 659 Kbs fair rate and session 2 don't use 9 Kbs of its 709 Kbs fair rate. This means that a total of 368 Kbs of bandwidth is available to be distributed between sessions with a negative quality factor, starting by the most important. Since all sessions are equally important (it was assumed in this simulation that all layers had the same number of receivers) this available bandwidth will be distributed probabilistically between session layers. Therefore layers with higher rate will get a higher percentage of this available bandwidth. This means that after session 5 (last session with a 0% dropping in layer 1), layers 1 and 2 will compete for the available bandwidth, which is allocated to layer 2 since it has a rate two times higher than layer 1. Besides the distribution of available bandwidth, sessions quality factor wasn't increased. As can be seen in figure 7, layers 1 and 2 of sessions 6 to 10 continue suffering losses.

Results in figure 7 show that the existence of a fair rate adjustment mechanism could increase sessions quality. For example, in this simulation, the available bandwidth would have been used instead to increase the fair rate of sessions with a quality factor, Q_{ij}, or a quality factor minus the last layer, Ql_{ij}, more close to zero, trying to turn it positive, increasing effectively the quality of some sessions. With a fair rate adjustment mechanism, session 6 and session 7 quality would have been increased, since their layer 1 traffic would have become all *in* profile. This because session 6 and 7 have the quality factors minus the last layer most close to zero from all sessions with a negative quality factor: $Ql_{6j} = -91\,Kbs$ and $Ql_{7j} = -191\,Kbs$. Considering the available bandwidth of 368 Kbs, those quality values would have become positive. The next session with Ql_{ij} more close to zero is the session 8 with $Ql_{8j} = -291\,Kbs$, however the remaining available bandwidth ($368 - 91 - 191 = 86\,Kbs$) isn't enough to increase its quality. In this example Ql_{ij} is used to increase sessions quality and not Q_{ij} since, due to the high rate of layer 2, the Q_{ij} of any session is to negative to be possible to turn it positive with an available bandwidth of 368Kbs. The session with the Q_{ij} more close to zero is the session 5 with $Q_{5j} = -891\,Kbs$.

Figures 8 to 10 show the second group of simulation results, where the MLUM policy was configured with equal fair rates of 909 Kbs for all sessions. Figure 8 and 9 show that sessions are treated equally when they have the same characteristics. In figure 10 the available bandwidth is higher than in the figure 7, what explains the higher rate of sessions 6 to 10 layer 2. Contrary of what happens in figure 7, in figure 10, layer 1 rate of sessions 6 to 10 decreases. This happens because these sessions fair rate is lower in figure 10 than in figure 7, and since fair rates are first used by layer 0, which rate increases from session 6 to 10, less bandwidth is available to layer 1 in figure 10.

Figures 11 to 13 show the third group of simulation results, where the MLUM policy is configured with fair rates starting at 1159 Kbs, for session 1, and decreasing by 50 Kbs per session. The first two figures have an explanation similar to the case of figures 5 and 6, respectively, considering the contrary evolution of the fair rates. In figure 13, layer 2 of sessions 6 to 10 has even a higher rate than in figure 10, because the fair rate made available by sessions 0, 1 and 2, is higher. Layer 1 rate, of sessions 6 to 10, decrease even more than in figure 10 since the fair rates of these sessions are lower in this third group of simulations.

4.3 Discussion of simulation results

We have simulated the MLUM policy data plane in several multi-layer, multi-receiver (different fair rates) scenarios in order to tests its ability to deal with a wide range of layers rates and to guarantee different ranges of fair rates. The simulation results show that the MLUM policy is adapted to deal with multi-layer sessions and can guarantee different fair rates set. The MLUM policy data plane marks more packets *in* profile for sessions with higher fair rate, but sessions with a rate higher

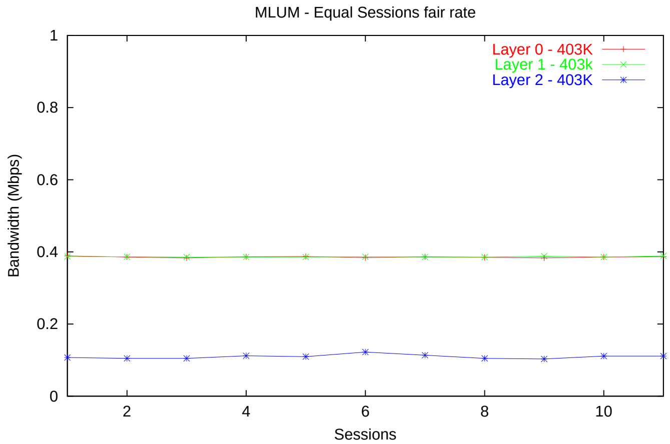

Figure 8: FR-equal and Rates-equal.

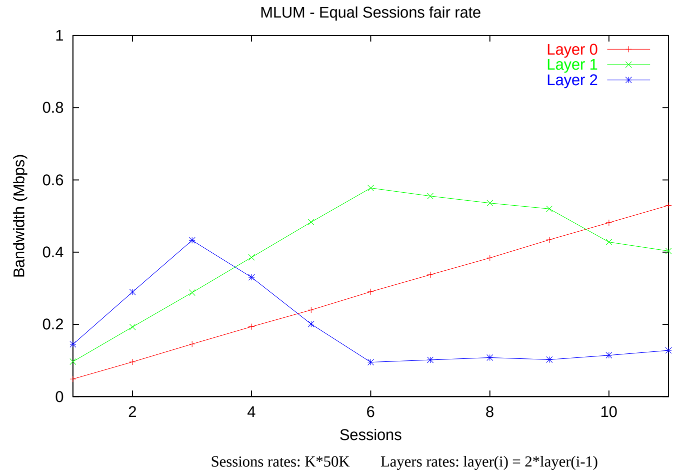

Figure 10: FR-equal and Rates-multiple.

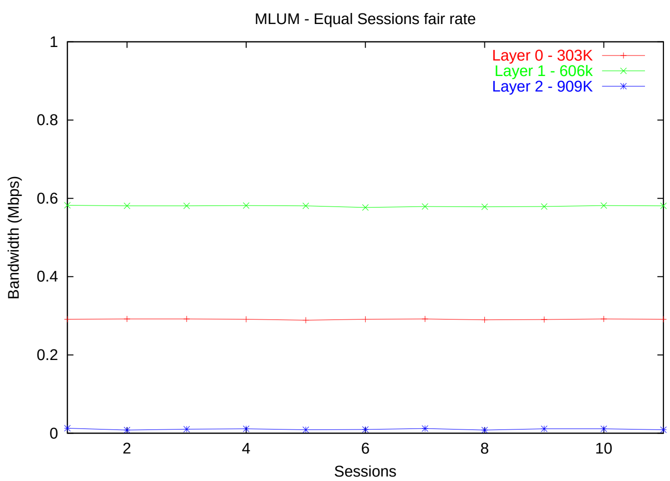

Figure 9: FR-equal and Rates-369.

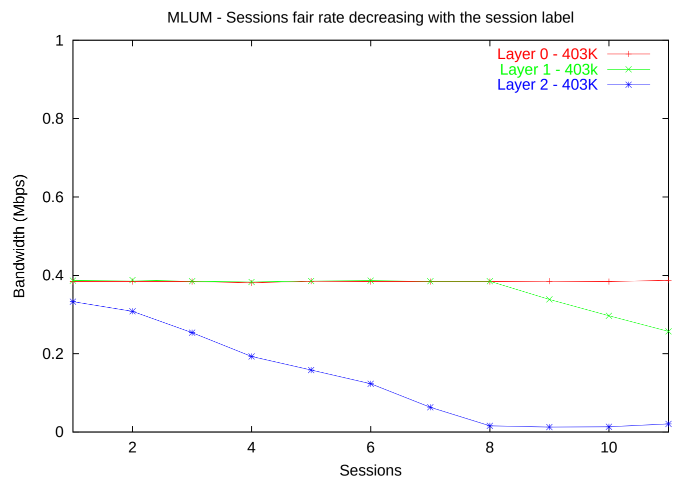

Figure 11: FR-decrease and Rates-equal

than their fair rate are punished. This punishment process respects the layers priority, i.e., dropping starts with lower priority layers. However, one important result of these simulations is the evidence that a fair rate adjustment mechanism can be helpful to increase the percentage of sessions with a positive quality level. Beside this, we compared the MLUM behavior with other fair and unfair queueing algorithms in an uni-rate, uni-receiver scenario. These simulations showed that the MLUM policy is far superior the FIFO and RED and is comparable to FRED. In what concerns the CSFQ approach, the MLUM policy performance had an average divergence of 9% from DRR, when CSFQ had an average divergence from DRR of 8%.

5 MLUM protocol

Although this paper aim to present and evaluate the MLUM fairness definition and policy, we'll describe briefly the functionality of the MLUM protocol. This is justified by the straight relation between the MLUM policy control plane and the MLUM protocol.

The MLUM protocol allows FR routers to exchange control information that MLUM policy needs to estimate sessions fair rates.

The MLUM policy needs to know the number of receivers in each layer used by the source. Although this information could be collected asking end hosts about the layers they want to subscribe, the MLUM policy shouldn't rely on receivers in order to guarantee fairness between all sessions. Therefore information normally given by end hosts, namely the number of receivers and the receivers layer scheme, is collected as follows. FR routers get the number of local receivers based upon the number of multicast subscription and unsubscription. FR routers determine layers priority based upon the order of subscriptions. This is possible because receivers can only subscribe to a $layer_i$ if they have already subscribed to $layer_{i-1}$. Receivers are motivated to subscribe in the correct order, because this is the only way for them to receive data with good quality.

In the protocol implementation being made in the NS simulator, receivers make two calls when subscribing or unsubscribing

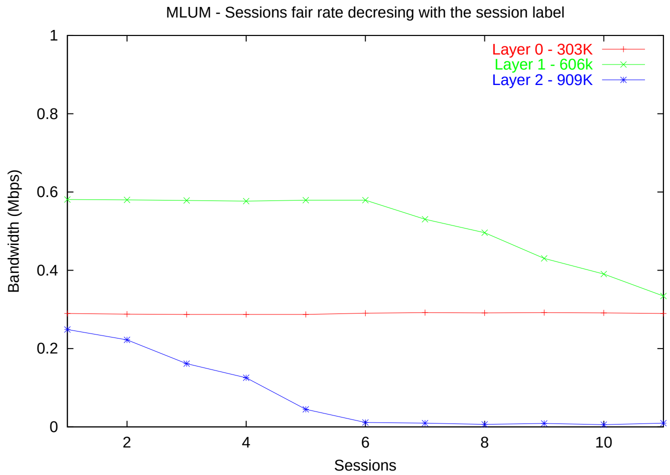

Figure 12: FR-decrease and Rates-369

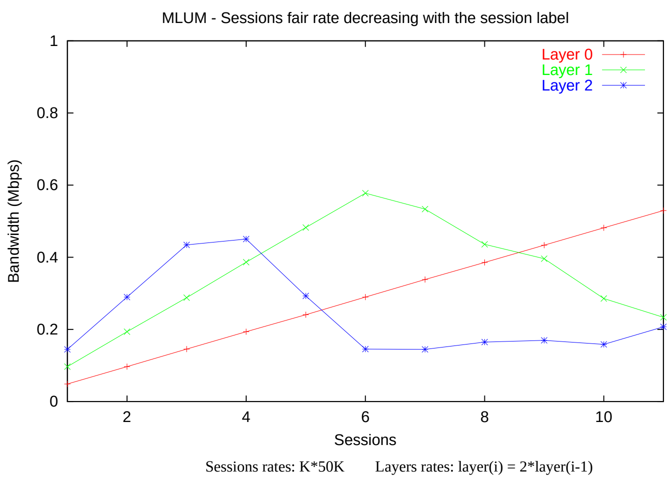

Figure 13: FR-decrease and Rates-multiple.

a session layer (multicast group). First they perform a MLUM subscription and then they perform a multicast subscription. The goal is to maintain the MLUM and multicast protocols independent. However in a future test-bed implementation, the MLUM protocol can get this information from the multicast aware FR router. This way receivers don't need to be changed, continuing to do only a multicast subscription / unsubscription operation.

The MLUM protocol has two types of messages. *Update* messages, which are sent upstream and *Sync* messages, which are sent downstream.

Update messages include information for each session that exist downstream that FR. That information include the maximum fair rate in all downstream links where the session has receivers and the total number of receivers in all represented downstream links. These messages are sent throughout the upstream link in the direction of the session source. *Update* message packets have a maximum size of *mtu*, which is pre-configured and have a default value of 576 bytes, in order to avoid fragmentation.

Two kind of *update* messages are generated by a FR. *Partial* messages, which have only information about layers that had a high variation in their number of receivers since the last sending period and *complete* messages, which contain information about all sessions layers in this FR. These messages are sent in pre-configured periods with duration given by the variable *partial_timeout*, in the former case, and by the variable *complete_timeout*, in the latter case. The *complete_timeout* value is superior to the *partial_timeout*, being by default two times bigger. The existence of partial messages decrease the amount of exchanged information helping to maintain the protocol scalable[5].

FR routers use another timeout variable in order to detect failure links. A failure in a link is detected if a FR don't receive any *update* messages through that link until this timeout expires. This timeout value should be equal or higher than the *complete timeout*, because if the downstream layers aren't very dynamic, FR routers will only get information about it in *complete* messages. Associating this timeout variable, not to each downstream link, but to each session layer, FR routers can use it also to detect layers without receivers, decreasing the time FR router have to keep those layers information stored. Basically, FR routers will send upstream information about layers without receivers in the next *update* message. If that information isn't included in the next *partial* message, it's locally deleted and the upstream routers will detect these layers absence in the next *complete* message, which will in this case have a smaller size.

Sync messages include the actualized minimum fair rate of each session. When a FR receives this kind of messages, it'll actualize the local fair rate of each session present in the message with the minimum value between the upstream and the local fair rates. *Sync* messages are sent downstream by each FR at periods with a pre-configured duration given by the variable *sync_timeout*. This value is by default equal to *complete_timeout*. The *sync_timeout* period defines the interval by which receivers can get information about their session fair rate, i.e., the interval at which receivers can adapt their reception quality.

6 Conclusion and Future Work

This paper presents and evaluates a *Multi-Layer Utilization Maximal (MLUM)* fairness definition, based upon the number of receivers each session has, and a MLUM policy that implements the fairness definition assigning more bandwidth to sessions with higher number of receivers, maintaining an efficient bandwidth utilization in each link. We presented some NS simulations that evaluate the impact of the MLUM fairness policy data plane, considering one link shared between sessions with three layers. The MLUM fairness policy data plane performance was compared with other queueing disciplines like FIFO, RED, FRED, CSFQ and DRR. The results show that, in a

[5]However a FR router can be configured to work only with complete messages.

multi-layer and multi-receiver scenario, the MLUM is the only approach that is adapted to deal with sessions with different rates per layer, maximizing the utilization of bandwidth and maximizing the number of receivers with their quality requirement satisfied. In an uni-rate and uni-receiver scenario, results show that the MLUM policy is superior to FIFO and RED. Moreover MLUM is comparable to FRED and CSFQ and its performance approximates the DRR behavior.

As future work we'll improve the MLUM policy data plane efficiency, mainly comparing the simulations results with the ones obtained using another queueing disciplines that allow MLUM FR routers to differentiate between layers marked *in* and *out*, namely the RIO approach described in [Clark98]. After this, we'll simulate the behavior of the MLUM policy control plane and the MLUM protocol in the same scenario used in this paper, i.e., only one link, in order to collect information about the computation charge inherent to the estimation of fair rates. This simulations will then be extend to a scenario with multiple links. After the evaluation of the complete MLUM model, we'll create a receiver driven adaptive mechanism, which will use the fair rates that receivers can get from the network in each *sync* period. On one hand, being this adaptation mechanism based upon the sessions fair rates it should solve the unfairness between multicast sessions and the leave latency problem [McCanne96]. On the other hand, since the adaptation will be trigged by the information carried in *sync* messages, this adaptation mechanism can be used to solve the significant and persistent quality instability of multimedia systems and the problems with synchronization of receivers. Considering this fairness model as the base of a hierarchical fairness structure, a future expansion of the model can be possible with the inclusion of additional utility functions, such as weighting receivers based upon a pricing or a priority approach and weighting sessions based upon their duration or distance between sender and receivers.

References

[RFC2475] Blake, S. and Black, D. and Carlson, M. and Davies, E. and Wang, Z. and Weiss, W. *"An Architecture for Differentiated Services". IETF, RFC 2475.*, December 1998.

[Clark98] Clark, D. and Fang, W. *"Explicit Allocation of Best-Effort Packet Delivery Service". IEEE/ACM Transactions on Networking, 6(4):362-373.*, August 1998.

[Chu00] Chu, Y. and Rao, S. and Zhang, H. *"A Case for End System Multicast". Proceedings of ACM Sigmetrics'00 , Santa Clara,CA, USA.*, June 2000.

[Floyd93] Floyd, S. and Jacobson, V. *"Random Early Detection gateways for Congestion Avoidance". IEEE/ACM Transactions on Networking, 1(4):397-413.*, August 1993.

[Holbrook01] Holbrook, H. and Cain, B. *"Source-Specific Multicast for IP". Internet - Draft.*, March 2001.

[Lin97] Lin, D. and Morris, R. *"Dynamics of Random Early Detection". Proceedings of ACM SIGCOMM'97, pages 127-137, Cannes, France.*, October 1997.

[Legout99] Legout, A. and Nonnenmacher, J. and Biersack, W. *"Bandwidth Allocation Policies for Unicast and Multicast Flows". Proceedings of IEEE INFOCOM'99, New York.*, March 1999.

[Li99] Li, X. and Paul, S. and Ammar, M. *"Multi-Session Rate Control for Layered Video Multicast". Proceedings of Multimedia Computing and Networking, San Jose, CA.*, January 1999.

[McCanne96] McCanne, S. and Jacobson, V. and Verteli, M. *"Receiver-driven Layered Multicast". Proceedings of ACM SIGCOMM'96, Stanford, CA.*, August 1996.

[NS] NS. *ucb/lbml/vint network simulator - ns (version 2.1b7a).*

[Pendarakis01] Pendarakis, D. and Shi, S. and Verma, D. and Waldvogel, M. *"ALMI: An Application Level Multicast Infrastruture". Proceedings of 3rd Usenix Symposium on Internet Technologies & Systems (USITS 2001), San Francisco, CA.*, March 2001.

[Rubenstein99] Rubenstein, D. and Kurose, J. and Towsley, D. *"The Impact of Multicast Layering on Network Fairness". Proceedings of ACM SIGCOMM'99, Cambridge MA.*, August 1999.

[Stoica98] Stoica, I. and Shenker, S. and Zhang, H. *"Core-Stateless Fair Queueing: A Scalable Architecture to Approximate Fair Bandwidth Allocations in High Speed Networks". Proceedings of ACM SIGCOMM'98, Vancouver, Canada, pp. 118-130.*, September 1998.

[Sankar00] Sankar, S. and Tassiulas, L. *"Fair Allocation of Discrete Bandwidth Layers in Multicast Networks". Proceedings of IEEE INFOCOM'2000, Tel-Aviv, Israel.*, March 2000.

[Shreedhar95] Shreedhar, M. and Varghese, G. *"Efficient Fair Queueing using Deficit Round Robin". Proceedings of ACM SIGCOMM'95, pages 231-243, Boston, MA, USA.*, September 1995.

[Taubman94] Taubman, D. and Zakhor, A. *"Multirate 3-D Subband Coding of Video". Proceedings of IEEE Transactions on Image Processing, Vol.3, No. 5.*, Vol. 35:pp. 572–588, September 1994.