Impact of Network Delay Variations on Multicast Sessions with TCP-like Congestion Control

Augustin Chaintreau[†] François Baccelli[‡] Christophe Diot[††]

[†]Ecole Normale Supérieure
45 rue d'Ulm
75005 Paris FRANCE
augustin.chaintreau@ens.fr

[‡]INRIA & Ecole Normale Supérieure
45 rue d'Ulm
75005 Paris FRANCE
francois.baccelli@ens.fr

[††]Sprint ATL
1 Adrian Court
Burlingame, CA 94010, USA
cdiot@sprintlabs.com

Abstract—**We study the impact of random noise (queueing delay) on the performance of a multicast session. With a simple analytical model, we analyze the throughput degradation within a multicast (one-to-many) tree under TCP-like congestion and flow control. We use the (max,plus) formalism together with methods based on stochastic comparison (association and convex ordering) and on the theory of extremes (Lai and Robbins' notion of maximal characteristics) to prove various properties of the throughput.**

We first prove that the throughput obtained from Golestani's deterministic model [1] is systematically optimistic. In presence of light tailed random noise, we show that the throughput decreases like the inverse of the logarithm of the number of receivers. We find analytically an upper and a lower bound for the throughput degradation. Within these bounds, we characterize the degradation which is obtained for various tree topologies. In particular, we observe that a class of trees commonly found in IP multicast sessions [9] (which we call umbrella trees) is significantly more sensitive to network noise than other topologies.

I. Introduction

TCP-friendly congestion control has been advocated by the IRTF Reliable Multicast Research Group in the past [13], where a TCP-friendly flow is a flow that competes "fairly" with TCP-connections. Several recent papers focused on a TCP-friendly solution for the control of multicast [10], [11], [12]. In particular, Golestani has made some fundamental observations on multicast flow and congestion control in [1] using a deterministic model.

The present paper goes a step forward from Golestani's in providing an understanding of further properties of TCP-like congestion control in a network with random noise. This step is of practical importance in that it establishes the dependence of a multicast session throughput on the number of receivers, and consequently refines observations realized with a deterministic model. Since multicast deployment will most probably be pushed by single source applications with high bandwidth requirements and a large number of receivers, it is important to check whether TCP-like congestion control does not in fact force multicast sessions to suffer very low bandwidth. Bhattacharyya, Towsley and Kurose [16] analyzed the impact of TCP-like congestion control on the throughput of a multicast session. They showed that for loss based additive-increase multiplicative-decrease algorithms, there is a severe degradation of throughput for large multicast groups.

We generalize the findings in [1] and [16] by showing that even in the case of an ideal TCP control where the flow control window size is kept equal to its maximal value, there is a severe throughput degradation within a one-to-many multicast tree when the group size grows. Intuitively, the session throughput is expected to decrease when the number of receivers increases for the following two reasons:

- Due to the stochastic assumptions, when a new receiver joins, it can add a new link whose bandwidth is less than that of any of the links already present in the tree.
- Due to the fact that the congestion control mechanism is based on informations stemming from all receivers, slow receivers will "slow down" the sender.

In other words, the higher the number of receivers, the higher the chance that one of them is slow enough to affect the global performance.

The influence of the tree topology on throughput that we establish analytically is another key contribution of this work.

We have chosen to model a multicast session as follows: packets are sent by a unique sender located at the root of a set of routers organized as a tree to a set of receivers located at the leaves of this tree. This tree will be referred to as the forward tree.

The transmission is controlled by a "TCP-like" congestion control mechanism where each receiver sends acknowledgements back to the sender, and where the sender throughput is controlled by a sliding window mechanism.

We have chosen to model a homogeneous tree, i.e. each receiver is equally distant from the source, and all routers with the same level in the tree have the same service time distribution. This assumption allows us to design a simpler model without losing the properties we want to observe.

The model captures congestion via the queueing delay that each packet experiences in each router it passes through. In particular, the fluctuations due to the processing of packets of other (unicast or multicast) connections sharing the same interface of the router are represented by random service times for packets of the reference multicast connection. Our random service times are assumed to be independent in time and space, and light-tailed (i.e. the tail decreases faster than a negative exponential function). The queuing strategy is assumed to be FIFO. Within this framework, the sender and the receivers are modeled as routers, possibly with different mean delays and different distributions.

For reasons that have been already explained (i.e. we are not interested in the effect of losses, but only in the effect of an ideal flow control having reached its maximal window size), we will assume that all routers have infinite buffers and consider that the network is lossless and that the window size is fixed.

All the assumptions that we make about the network (homogeneity), about transmission control (no losses, window size al-

0-7803-7016-3/01/$10.00 ©2001 IEEE

ways equal to its maximal value) and about noise (light tailed) have been carefully selected to provide an optimistic network environment. We will show that even in this favorable context there is a severe decrease of the throughput when the number of receivers increases.

To the best of our knowledge, this work is the first to address analytically the question of multicast session throughput degradation due to network noise (queueing delay), for different tree topologies, in a TCP-like (single-rate) control environment. Although we limited this first study to some simple cases, we believe that our mathematical methodology can be expanded to analyze more general cases (e.g. adaptive window, heavy tailed noise, non-homogeneous trees or windows) as discussed in the conclusion.

The paper is structured as follows: in Section II, we build our analytic model on the (max,plus) formalism [3], [4], [5]. In Section III, an algebraic simulator is derived from the model. Simulations show that the throughput obtained from Golestani's deterministic model [1] is systematically optimistic. We study throughput degradation for a large number of receivers and for different tree topologies. We further generalize our simulation results with the help of the (max,plus) model. In Section IV, we analyze the model using the notion of positive correlation (also called association), as well as the notion of maximal characteristics (Lai and Robbins). In the presence of a light tailed random noise, the throughput is shown to be upper and lower bounded by functions that decrease like the inverse of the logarithm of the number of receivers. This qualitative result explains the general shape obtained by simulation for throughput degradation. Within these bounds, we characterize the fine structure of degradation depending on the tree topology. We analyze three different families of tree topologies. First, we analyze classical binary trees. We then consider a class of trees commonly found in IP multicast sessions, [9], [14], which we call umbrella trees. We show that this class of trees is significantly more sensitive to network noise than other topologies, and that in some cases, these topologies reach the lower bound. We finally characterize the throughput degradation curve for a class of optimal trees called "reverse-umbrella" trees. The theorem proofs that cannot be found in this article are given in [17].

II. (MAX,PLUS) REPRESENTATION

We first introduce the (max,plus) algebra, show how it can be used to represent a network on a simple example, and apply this representation to multicast.

A. Introduction to the (max,plus) Algebra

We will first consider the scalar algebra, namely the set $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$, which we endow with two operations that are different from the usual ones : the max operation (denoted $\vee$) replaces the usual addition, and addition, with the convention ($\forall a \in \mathbb{R}_{\max}, -\infty + a = -\infty$), replaces the usual multiplication.

Note that this structure has all the properties required to make a commutative semi-ring : associativity, commutativity, identity elements[1], and distributivity ($\forall a, b, c \in \mathbb{R}_{\max}, a + (b \vee c) = (a + b) \vee (a + c)$).

Since $(\mathbb{R}_{\max}, \vee, +)$ is a semi-ring, we can construct matrix operations as in the conventional algebra, with the addition of matrices obtained by term by term maximization, and multiplication defined by the rule : $(\mathbb{A}\mathbb{B})_{i,j} = \max_k(\mathbb{A}_{i,k} + \mathbb{B}_{k,j})$. We will denote $\mathcal{E}$ the matrix filled with ε everywhere, and $\mathbb{I}$ the identity matrix (e on the diagonal and ε everywhere else).

Norm Let $\|.\|$ denote the matrix norm $\|\mathbb{A}\| = \max_{i,j}(\mathbb{A}_{i,j})$.

B. (max,plus) Representation of a Network

We illustrate the (max,plus) modeling of a network via a simple example : a point-to-point end-to-end connection through L routers (numbered 1 to L) with window flow control with a fixed size window W (this model and its multicast extension were introduced in [6]). The sender is incorporated into the first router, and the receiver into the last router. The multicast model we will present in section II-C is a simple extension of this preliminary example.

Each router is represented as a FIFO queue with an infinite buffer[2] and a random service time for each packet of the connection. As explained below, the service time includes the delay due to the processing of certain packets of other connections present in the router. Assuming that the connection under consideration stabilizes, it is natural to make the assumption that the service times of our router are identically distributed. We will also assume service time independence for the sake of simplicity, i.e. service times for different routers in the network are independent and the sequence of service times on a router is made of independent and identically distributed random variables. This assumption will be critical for the type of degradation that will be established in the present paper; however it can be significantly weakened for many other aspects like the representation of the network via products of random matrices and the subsequent characterization of throughput.

We will denote $s_m^{(i)}$ the service time of the m-th packet of the controlled connection on router i, and $x_m^{(i)}$ the time when router i has completed the processing and forwarding of packet m.

• Router $i > 1$ starts processing packet m as soon as it has finished processing packet $m - 1$, and the upstream router has forwarded packet m. After it has started processing this packet, $s_m^{(i)}$ units of time are still required for processing it. This processing time actually includes the processing time of all the packets of the other connections interleaved between packet $n - 1$ and packet n of the reference connection. So we have for $i > 1$:

$$x_m^{(i)} = (x_{m-1}^{(i)} \vee x_m^{(i-1)}) + s_m^{(i)}.$$

• The sender (considered as router $i = 1$) sends packet m as soon as it has finished with packet $m - 1$, and provided that the window control allows packet m to be sent (this translates the assumption that the source is saturated, namely it always has packets to send). So that we have :

$$x_m^{(1)} = (x_{m-1}^{(1)} \vee x_{m-W}^{(L)}) + s_m^{(1)}.$$

[1] for $\vee$, $-\infty$ that we will denote ε, and for $+$, 0 that we will denote e.

[2] Note that buffers being of infinite size, no loss occurs. As a result, the window size is assumed to reach its maximal value and to remain constant. The effect of congestion control is expressed by the variations of service times in routers.

Let X_m be the vector of dimension L with entries $(x_m^{(i)})_{1 \le i \le L}$, and Y_m be the block-vector of dimension LW with blocks $X_m, X_{m-1}, \dots, X_{m-W+1}$. We can capture the dynamics of the network by a (max,plus) linear recurrence

$$\begin{cases} Y_0 = \text{ the vector with all its coordinates equal to } e \\ Y_m = \mathbb{P}_m Y_{m-1} \text{ for } m > 0 \ , \end{cases} \tag{1}$$

where the matrix $\mathbb{P}_m$ has the following block structure (each block is a square block of dimension L):

$$\mathbb{P}_m = \begin{pmatrix} \mathbb{S}_m & \varepsilon & \dots & \varepsilon & \mathbb{W}_m \\ \mathbb{I} & \varepsilon & \dots & \varepsilon & \varepsilon \\ \varepsilon & \mathbb{I} & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \varepsilon & \varepsilon \\ \varepsilon & \dots & \varepsilon & \mathbb{I} & \varepsilon \end{pmatrix},$$

$$\mathbb{S}_m = \begin{pmatrix} s_m^{(1)} & \varepsilon & \varepsilon & \dots & \varepsilon \\ s_m^{(1)} + s_m^{(2)} & s_m^{(2)} & \varepsilon & \dots & \varepsilon \\ \vdots & & & \ddots & \vdots \\ s_m^{(1)} + \dots + s_m^{(L)} & & \dots & & s_m^{(L)} \end{pmatrix}$$

- $\mathbb{W}_m$ represents the window control mechanism. In this case we have $(\mathbb{W}_m)_{i,j}$ equal to ε if $j \neq L$ and to $s_m^{(1)} + \cdots + s_m^{(i)}$ for $i = 1, \dots, L$ and $j = L$.
- $\mathbb{S}_m$ represents the forwarding mechanism in the network, and $(\mathbb{S}_m)_{i,j}$ is more generally given by the maximum over all paths leading from i to j of the sum of service times for packet m on the path from router j to router i (including both i and j).

Note that if service times are independent and identically distributed, then the matrices $(\mathbb{P}_m)_m$ are also independent and identically distributed; we can then apply Corollary 1 (Appendix A), which gives the existence of $\lim_{m\to\infty} \frac{\|Y_m\|}{m} = \gamma$ both in expectation and with probability 1. γ is called the *Lyapounov exponent* of this sequence of matrices. Since $\|Y_m\|$ represents the epoch when packet m has arrived to its destination, $\frac{1}{\gamma}$ is therefore the *average throughput*, i.e. the total amount of data transmitted since the beginning of the session divided by the duration of the session. In the following section, we extend this model to multicast.

C. Representation of Multicast Flow Control

A single source broadcasts packets over a unidirectional tree to N receivers. Each node in the forward tree simultaneously duplicates and forwards each packet on the downstream branches. Acknowledgements are forwarded back to the source through a backward tree which is a mirror version of the forward tree, which we assume to be functionally independent of the forward tree. In the backward tree, the ack of a packet only arrives in router i when the latest of the acks of the same packet have been sent by the routers upstream. It is then transmitted by router i after some queuing delay. The aggregation of acknowledgements in each router of the backward tree allows one to take care of the implosion problem (see [1]). The case of a binary tree is shown in Figure 1.

The flow control is enforced by the sender; it is again based on a sliding window mechanism, of constant size W: the sender only sends packet $n + W$ when the ack of packet n has been received from the final router of the backward tree.

We have chosen to model homogeneous trees only. Homogeneity means that the path from the sender to each receiver is statistically the same for all receivers, i.e. there is the same number of router, and the service time distribution is the same for all routers of the same level[3].

We will still denote L the total number of routers in the network, and D the depth of the forward tree. For receiver i, we will denote $f(1,i), f(2,i), \dots, f(D,i)$ the different routers on the path from the sender to this receiver in the forward tree, and $f(D+1,i), f(D+2,i), \dots, f(2D,i)$ denote the different routers in the backward tree transmitting acknowledgments from receiver i to the sender. By definition, the *path* of receiver i is the sequence $f(1,i), f(2,i), \dots, f(2D,i)$. For router $f(d,i)$, we denote $s_n^{(f(d,i))}$ the service time of the n-th packet on this router.

With this notation

$$S_m^{(i)} = s_m^{(f(1,i))} + \cdots + s_m^{(f(2D,i))} \tag{2}$$

is the (minimal) round trip time (RTT) of packet m on the path that contains receiver i.

Homogeneity is an important difference with the assumptions in Golestani's model. Given his conclusion that receivers should have a window size proportional to their distance from the source, it makes sense to consider a homogeneous tree with a single window.

Note that homogeneity allows for quite complex tree structures. Homogenous trees are complex enough to illustrate the properties we want to stress. They also make the comparisons between different topologies easier.

The network model described above can be written in a way similar to that of Section II-B. Let X_m be the $\mathbb{R}_{\max}$ vector of dimension L where entry i is the departure time of packet m from router i. The first entry corresponds to the router of level 0 in the tree (the source), and the last entry corresponds to the router of the highest level ($2D$), at the end of the backward tree, which can be seen as that of the final aggregation. Let Y_m be the block vector of dimension LW built on top of $(X_m)_{m\in\mathbb{N}}$ and which captures the history of X_m in the same way as above. We have the same (max,plus) linear system for Y_m as in Equation (1), though with different matrices.

$\mathbb{P}_m$ has exactly the same block structure as before. The block $\mathbb{W}_m$ is defined as follows: if $f(d,i)$ is router l, then $(\mathbb{W}_m)_{l,L} = \max_{j\in r(l)} s_m^{(f(1,j))} + \cdots + s_m^{(f(d,j))}$, where $r(l)$ is the set of receivers whose path contains router l, and all other rows are ϵ.

The block $\mathbb{S}_m$ is again the maximum over all paths from l' to l of the sums of the different service times (of order m) along the path (with the maximum over an empty set equal to ε by convention).

The sequence $(\mathbb{P}_m)_{m\in\mathbb{N}}$ is again i.i.d., which allows us to deduce from Corollary 1 the existence of the Lyapounov exponent, which is given by $\lim_{m\to\infty} \frac{\|Y_m\|}{m}$ and which represents the inverse of the averaged throughput of the connection.

[3] two routers in the graph have the same level if they are equally distant from the sender.

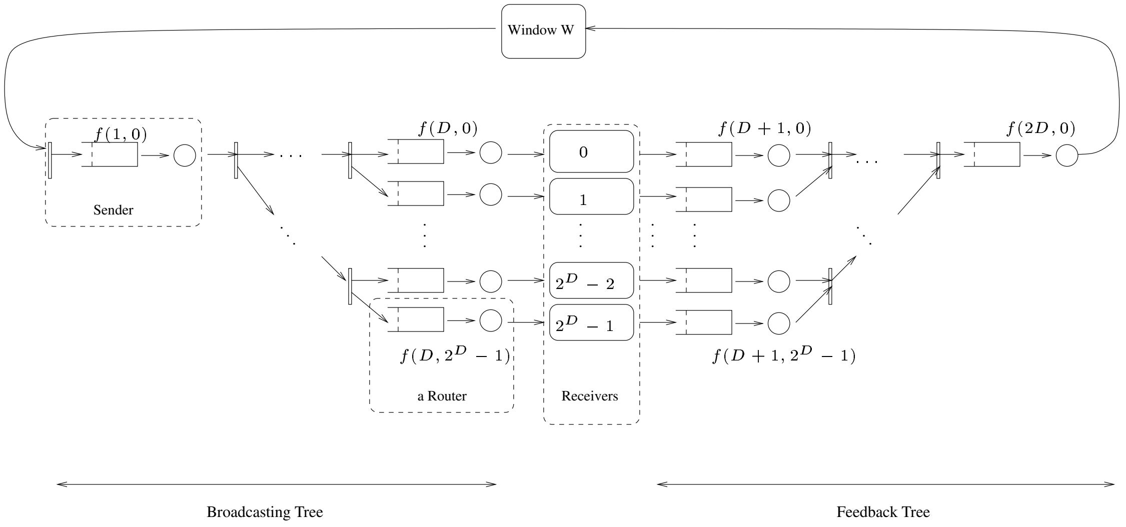

Fig. 1. The “forward and backward” graph

III. Algebraic Simulation

The algebraic simulator described below does the same job as a discrete event simulator. Its advantages are twofold: (a) since this type of simulation consists in products of matrices and vectors, it is of low complexity, which is important when the number of receivers becomes large, and (b) the same formalism is used in the simulation and in the analytical sections.

A. Description of the Simulator

We can compute the Lyapounov exponent which is the inverse of the average throughput for the connection, and which can be obtained from the simulator as the almost sure limit $\gamma = \lim_{m\to\infty} \frac{\|Y_m\|}{m}$. In practice, we can estimate γ by $\|Y_M\|/M$ for a large enough value of M. The algebraic simulator samples different random variables for service times in the routers, then builds the matrix $\mathbb{P}$, and multiplies the current value of Y by $\mathbb{P}$. After M steps, we have Y_M, and hence a reasonable approximation of γ (if M is chosen properly). Preliminary convergence studies that we made revealed that in most of the simulation runs, we can limit ourselves to $M = 400$ steps in order to have a good approximation of the Lyapounov exponent.

As far as the simulation is concerned, there is a natural computational trade-off between the accuracy of the estimation of the Lyapounov exponent and the simulation of multicast groups with a large number of receivers. The accuracy of the throughput estimator requires the simulation of a large number of packets, or equivalently the computation of the product of a large number of matrices, whereas large groups implies the manipulation of large matrices. In order to simulate large multicast groups, we had to accept moderately accurate estimates (i.e. rather large confidence intervals) for the Lyapounov exponents. This choice results in rather non-smooth shapes for most of the simulation curves produced below. However, as we see in the next section, this is sufficient to estimate the general shape and the relative ordering between the curves in question.

A.1 Modeling the Topology

In addition to the homogeneity assumption that we stressed in the last section, we will further assume that all service times in the backward tree are zero (i.e. as soon soon as all copies of packet m have reached their receivers, the sender instantaneously receives the acknowledgement of packet m).

Let us first consider a complete binary tree with height D and with total number of leaves 2^D).

We need first to vary the number of receivers of a multicast session to make it possible to study how the throughput varies with the size of the group. For every binary tree of size 2^D, we consider a set of N 'active' receivers which is a subset of the leaves of the complete binary (forward) tree ($N \leq 2^D$). For this we simply set the service times to be equal to zero in all the routers that do not forward packets to an active receiver. So we can use the general equations for the complete binary tree with these special values of the service times to analyze the sub binary tree corresponding to this subset of N leaves.

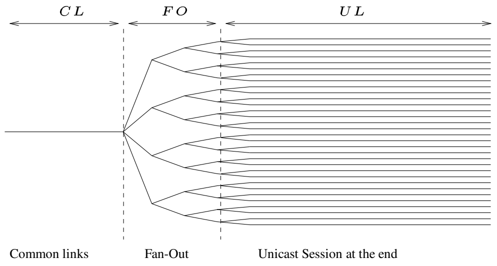

Fig. 2. Trees generic topology

The tree topologies we study are represented Figure 2. These topologies consist of three parts :
- A first set of CL links which is common to all receivers.
- A Fan-Out whose total depth is FO. The first step of this fan-out is k-ary (degree[4] k for the first node of the fan out), all the

[4] The method to emulate a k-ary fan out in binary trees consists in starting the binary expansion before the real fork and in using appropriate values for service time ensuring that this represents the desired fork.

other fan-outs are binary (degree 2 everywhere else).

- A unicast transmission of depth UL (unicast in the sense that there is no duplication of the packet in this part of the tree, and no link shared by different receivers).

Using this parametric representation of tree topologies, we simulate three types of trees represented on Figure 3. In addition to complete binary trees, we consider :

- Umbrella Trees : these trees end with a long unicast transmission after a short fan-out (large value of UL). The limiting case is that with one independent path from the source to each receiver. It is characteristic of a multicast tree where the receivers only share few links. This kind of topology is identified in [9], [14] as being often found in Mbone sessions.
- Reverse Umbrella Trees : packets are forwarded first along a long common path, and then a short fan out ends the transmission (large value of CL). Intuitively this kind of topology is optimal, as receivers behaviors differ only by few links.

These categories will be more precisely defined and analytically studied in Section IV-C.

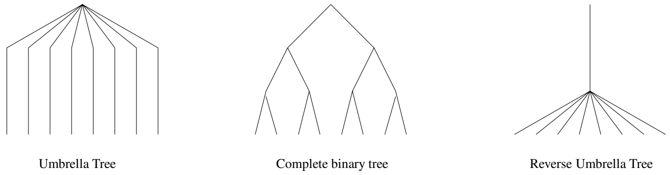

Fig. 3. Fundamental types of tree topologies observed

B. Average Throughput vs. Number of Receivers

This section focuses on the degradation of performance (i.e. increase of the Lyapounov exponent) when increasing of number of receivers in a multicast group.

We start the simulation by taking one active receiver in a binary tree, and by computing the associated (max,plus) linear recurrence on Y in order to estimate γ. Then we pick another receiver in the tree, add it to the current tree and compute the same simulation (which gives the value of γ for two receivers). Then we progressively fill the tree with more and more receivers.

We simulate different ways of filling in the tree. ”Best Filling” consists in starting from receiver 1 (numbers refer to Figure 1) and taking at each step the ”next” receiver in the order suggested by the numbering. We also consider ”Random Filling”, where each new receiver to join is chosen randomly.

Simulation results are shown on Figures 4 to 10. The service times in the routers follow an exponential law with the same parameter (λ) for each router in the network, so that the homogeneity condition is satisfied. In each simulation, λ is chosen in such a way that the sum of the service times along a path from the sender to any receiver has a mean value equal to 1 ($\lambda = D$).

Note that for homogeneous trees with deterministic service times, there is no dependency of the throughput on the number of receivers, as each receiver has the same round trip time and behaves synchronously with other receivers in the multicast group. For each plot, we have represented the value of the throughput in the deterministic case as found by Golestani, which will be equal to 1 for this choice of λ.

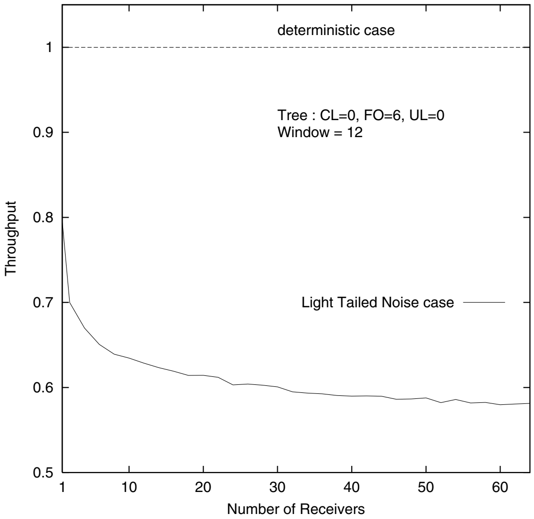

Fig. 4. Average throughput vs. number of receivers in a binary tree with light noises.

Figure 4 is obtained by simulating a complete binary tree of length 6 ($CL = 0, FO = 6, UL = 0$), with a window of size 12. Each router of the tree has an exponential service time with mean value 1. The feedback tree has a null service time on all routers.

The first important observation is that the average throughput decreases like the inverse of the logarithm of the number of receivers. This is completely different from what is obtained with a deterministic approach which seems to give a pretty optimistic evaluation of the throughput (represented by the horizontal dotted line). This observation will be verified analytically in the next section.

The important throughput drop between 1 and 2 receivers can be explained by the homogeneous nature of the tree that makes that the second receiver joins the tree with a path of length $CL + FO + UL$. Then the throughput keeps decreasing significantly until 20 participants. Between 40 and 60 receivers, the throughput stabilizes around 50% of the deterministic case.

Another important remark is that even when there is only 1 receiver, the throughput obtained by the stochastic model is significantly less than that of the deterministic model. For a theoretical explanation based on convex ordering, see section IV-A below.

The same kind of throughput degradation has also been observed in [16] under different assumptions.

B.1 Influence of the Noise on the Throughput

Before further investigating the shape of the throughput degradation, we have to verify that the shape of the degradation is not a direct consequence of the nature of the network noise.

Figure 5 gives throughput as a function of the number of receivers for service times belonging to the class of (bounded support) truncated exponential distribution functions. Since the mean values are not preserved by truncation of a given exponential density, the relative positions of the curves is not particularly meaningful.

So, the most interesting remark to be made bears on the shape

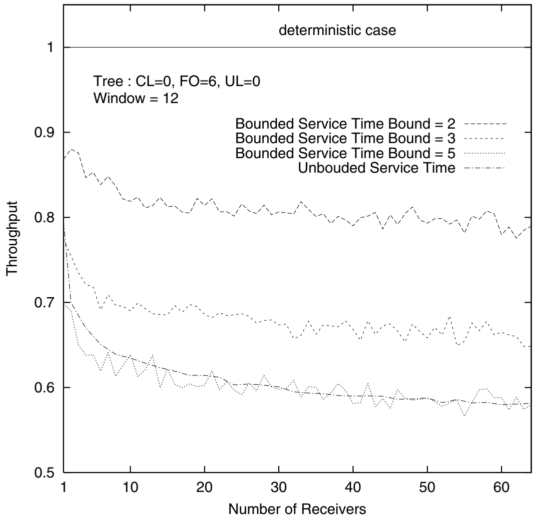

Fig. 5. Average throughput vs. number of receivers in a binary tree with bounded and unbounded light noises.

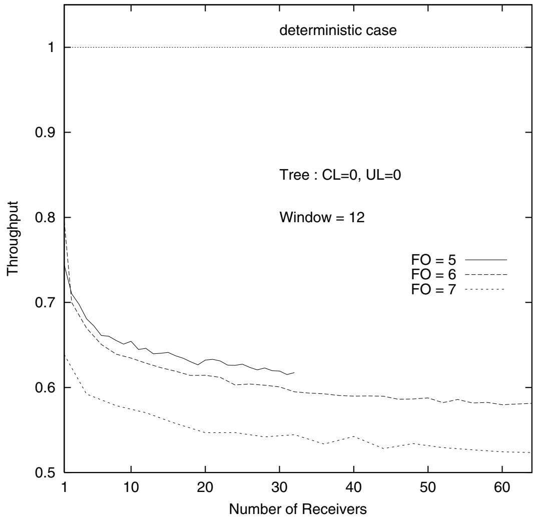

Fig. 6. Throughput vs. number of receivers for different tree depths.

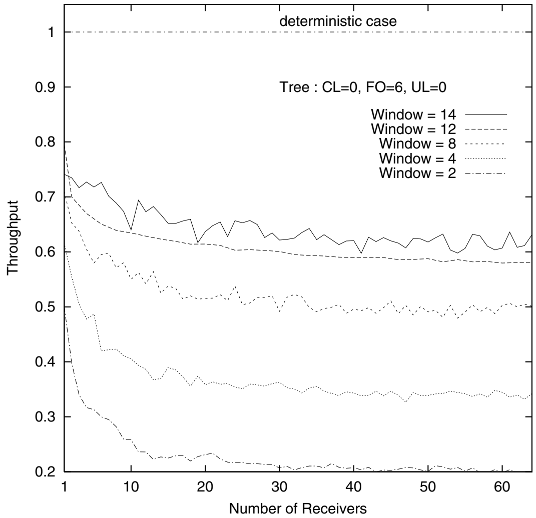

Fig. 7. Throughput vs. number of receivers with varying window size.

of the curves. We observe the very same logarithmic decrease as in the bounded case. This is particularly clear when looking at the case where the truncation threshold is large (i.e. equal to 5), which leads to throughputs that are quite close to the unbounded case. Thus, we can conclude from these curves that the shape of the degradation is not bound to the exponential assumption per se. Our conjecture is rather that all distribution functions with a tail bounded from above by a negative exponential function lead to such a decrease provided variance is large enough. For heavier tails, (e.g. Pareto tails) preliminary results seem to suggest that the growth of the Lyapounov exponent is polynomial. So, the bounded support and the light tail cases are qualitatively the same, at least when variance is not too small, and this generic case seems to be the most favorable one when compared to heavier Pareto type tails.

In the following simulations, we use unbounded service times.

B.2 Analysis of Various Network Parameters

In order to understand the impact of network parameters on the throughput degradation, we have varied network parameters. Figure 6 shows how the throughput decreases for various tree depth values. Trees being homogeneous, and service times being here all with the same distribution, tree depth influences the throughput by the fact that each receiver joins the tree with a path whose length is the maximum tree depth (i.e. $CL + FO + UL$). The deeper the tree, the faster the throughput decrease. We have chosen a default size of 6, which allows us to simulate a sufficiently high number of receivers.

Figure 7 focuses on the influence of the window size. We have chosen 12 as default value; this value is sensible and it keeps simulation times low enough.

We finally checked (Figure 8) the influence of the filling algorithm on throughput degradation. As expected, randomly filled trees suffer a more severe throughput decrease than best filling trees. This difference is easy to explain. In the random filling approach, adding a new receiver generally adds more network links than in the best filling approach, where a new receiver systematically adds the minimum possible number of links.

C. Analysis of the Tree Topology

We now simulate the tree topologies described in Section III-A.1 with window size equal to 12 and with a random filling technique. The total depth of the tree is always equal to 6. We only vary the value of CL, FO, and UL with the sum being 6. In a deterministic model, all trees of same length would have the same performance (depending only on the round trip time). Figure 9 plots various umbrella trees and Figure 10 plots various reverse-umbrella trees. In both case, the binary tree case is given as a reference.

First, varying the topology of the tree significantly influences (up to 20%) the throughput. The second observation is that reverse-umbrella trees perform systematically better than binary trees, themselves performing better than umbrella trees.

We also observe that the closer the fan-out from the receivers, the higher the throughput. Thus, trees where receivers share

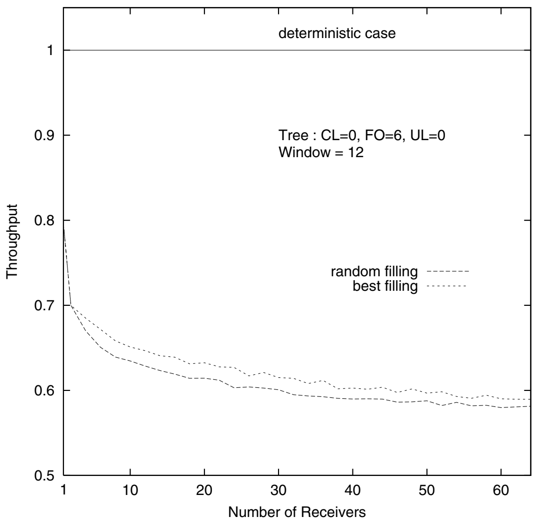

Fig. 8. Throughput vs. number of receivers with different tree construction approaches.

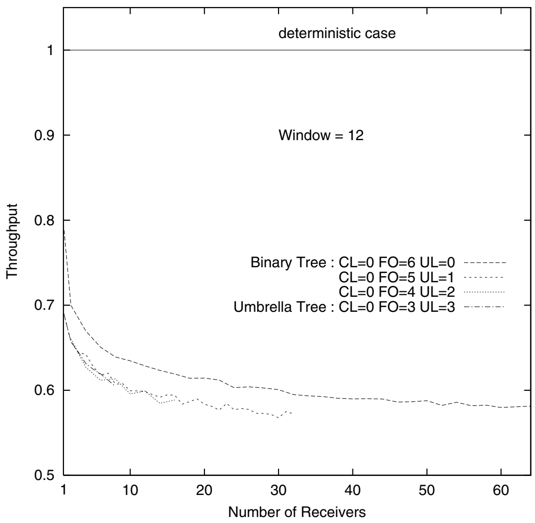

Fig. 9. Throughput vs. number of receivers in the case of umbrella trees.

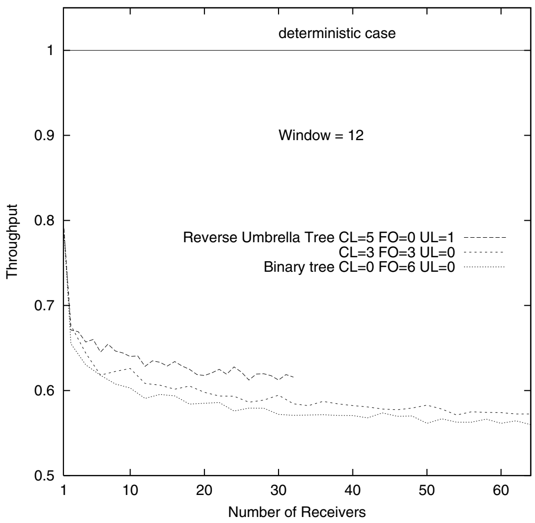

Fig. 10. Throughput vs. number of receivers in the case of reverse-umbrella trees.

few links are much more sensitive to the number of receivers in the group. The throughput of an umbrella tree has already decreased to 78% of the case with one receiver for a group made of 7 receivers, while a reverse-umbrella tree reaches the same throughput for 19 receivers. A throughput decrease of 81% (still compared to the case with one receiver) is reached with an umbrella tree for 3 receivers; with a binary tree, the same rate in obtained for 7 receivers and with a reverse umbrella tree for 10 receivers. For a group with three receivers, the throughput degradation (still compared to the case with one receiver) is 36% higher for an umbrella tree than for a binary tree. This observation is very important as the current Internet topology seems to favor umbrella trees. Note that such trees were also shown to be optimal in terms of network resource consumption [2].

IV. Mathematical Analysis of Throughput

In this section our goal is first to give mathematical arguments substantiating the growth of the Lyapounov exponent in $\ln(N)$ that was observed in the simulations. Second, we give mathematical arguments explaining why and how certain trees or certain situations should compare in a predictable way.

In order to characterize the throughput in our model, we use a few basic notions of stochastic comparison : *convex ordering* which will help us to compare the deterministic case and the random case; the notion of *association of random variables* which will help us to express the correlation between different receivers; lastly the notion of *maximal characteristics* which comes from the theory of extremes, and which will allow us to get explicit bounds on the performances.

A. Comparison with the Deterministic Case

Proposition 1: Let $\mathbb{A}$ and $\mathbb{B}$ be two random $\mathbb{R}_{\max}$ matrices. Then for all i and j : $(\mathbb{E}[\mathbb{A}\mathbb{B}])_{i,j} \geq (\mathbb{E}[\mathbb{A}]\mathbb{E}[\mathbb{B}])_{i,j}$, which implies $\|\mathbb{E}[\mathbb{A}\mathbb{B}])\| \geq \|\mathbb{E}[\mathbb{A}]\mathbb{E}[\mathbb{B}]\|$.

Proof: We just need to verify this formula for the two basic operations. This is clear for the addition; concerning the $\max$ operation, for all random variables X and Y with value in $\mathbb{R}_{\max}$, we have $\mathbb{E}[\max(X,Y)] \geq \max(\mathbb{E}[X], \mathbb{E}[Y])$ by a direct convexity argument. ■

Let now $\bar{\mathbb{P}}_m$ be the matrix describing the same network as above but this time when replacing each random service time by its mean value. We have $\bar{\mathbb{P}}_m = \mathbb{E}[\bar{\mathbb{P}}_m] \leq \mathbb{E}[\mathbb{P}_m]$ and so

$$\begin{aligned}\bar{Y}_m = \bar{\mathbb{P}}_m \ldots \bar{\mathbb{P}}_1 Y_0 &\leq \mathbb{E}[\mathbb{P}_m] \ldots \mathbb{E}[\mathbb{P}_1] Y_0 \\ &\leq \mathbb{E}[\mathbb{P}_m \ldots \mathbb{P}_1 Y_0] = \mathbb{E}[Y_m], \qquad (3)\end{aligned}$$

which implies $\bar{\gamma} \leq \gamma$. Hence, Golestani's deterministic model is actually proven to give the best possible throughput within the range of all stochastic models of the same class and with the same means.

B. Bounding Throughput Degradation

We now analyze the way throughput decreases when new receivers join the group.

B.1 Upper Bound

Theorem 1: Consider a network with exponential service times in routers; then γ is bounded from above by a function that can be expanded as

$$RTT\ln(N)(1+o(1)) \text{ for } N \to +\infty, \tag{4}$$

where N is the number of receivers, RTT is the average minimal round trip time of a receiver ($RTT = \mathbb{E}[S_1^{(1)}]$, where $S_1^{(1)}$ is defined in (2)).

Proof: We have

$$\frac{\|Y_m\|}{m} = \frac{\|\mathbb{P}_m \dots \mathbb{P}_1 Y_0\|}{m} \leq \frac{\|\mathbb{P}_m\| + \cdots + \|\mathbb{P}_1\|}{m}$$

The strong law of large numbers shows that with probability 1,

$$\gamma \leq \lim_{m\to+\infty} \frac{\|\mathbb{P}_m\| + \cdots + \|\mathbb{P}_1\|}{m} = \mathbb{E}[\|\mathbb{P}_1\|].$$

We now use the interpretation we have on the elements of $\mathbb{P}_1$. The largest element in $\mathbb{P}_1$ is the maximum of the sums of the service times of packet 1 along paths from the source to the last router, that is $\|\mathbb{P}_1\| = \max_{i=1\dots N}(S_1^{(i)})$.

The random variables $S_1^{(i)}$, $i = 1, \dots, N$ are *associated* [8], so that

$$\gamma \leq \mathbb{E}[\|\mathbb{P}_1\|] = \mathbb{E}[\max_{i=1\dots N}(S_1^{(i)})] \leq \mathbb{E}[\max_{i=1\dots N}(\tilde{S}_1^{(i)})],$$

where the random variables $\tilde{S}_1^{(i)}$ are independent, and for all i, $S_1^{(i)}$ and $\tilde{S}_1^{(i)}$ have the same law (for more details, see the appendix of [17]).

Using the homogeneity assumption (II-C), we have

$$\begin{aligned} \mathbb{E}[\max_i \tilde{S}_1^{(i)}] &= \mathbb{E}[\max_i(\tilde{s}_1^{(f(1,i))} + \cdots + \tilde{s}_1^{(f(D,i))})] \\ &\leq \mathbb{E}[\max_i \tilde{s}_1^{(f(1,i))} + \cdots + \mathbb{E}[\max_i \tilde{s}_1^{(f(D,i))}]]. \end{aligned}$$

For every $\max$ we can apply Corollary 2, so that we have the sum of D functions that can be expanded as $\ln(N)(1+o(1))$ multiplied by $\mathbb{E}[s_1^{(f(i,1))}], \dots, \mathbb{E}[s_1^{(f(D,i))}]$, respectively, so that the sum can be expanded in the same way with a multiplicative constant equal to $\mathbb{E}[s_1^{(f(1,i))}] + \cdots + \mathbb{E}[s_D^{(f(D,i))}] = \mathbb{E}[S_1^{(i)}]$. ■

This upper bound can be reached, this is the case when the window size is $W = 1$ and when we have an umbrella tree (tree made of very independent branches, which corresponds to the least aggregation (see Section IV-C)).

B.2 Lower Bound

Theorem 2: Consider a network with exponential noise in routers, and assume that receivers are distinct (ie. the last link for every receiver is different); then γ is bounded from below by a function that can be expanded as

$$\frac{\mathbb{E}[s^{(D)}]}{W}\ln(N)(1+o(1)) \text{ for } N \to +\infty, \tag{5}$$

where N is the number of receivers, W the window, and $s^{(D)}$ a typical service time in a router of level D (the last routers before receivers), which is by assumption the same for all receivers.

Proof: Let us consider the packets $W, 2W, 3W, \dots$. Since window has a fixed size, packet kW cannot start until the acknowledgements of packet $(k-1)W$ have arrived from all receivers (which is the definition of $\|Y_{(k-1)W}\|$). Since we then need to forward packet kW from the sender to all receivers to reach time $\|Y_{kW}\|$), we have :

$$\|Y_{kW}\| \geq \|Y_{(k-1)W}\| + \max_{i=1\dots N} S_{kW}^{(i)}.$$

So that we have, using $\gamma = \lim_{m\to+\infty} \frac{\|Y_{kW}\|}{kW}$,

$$\gamma \geq \frac{1}{W} \lim_{m\to+\infty} \frac{\max_{i=1\dots N}(S_W^{(i)}) + \cdots + \max_{i=1\dots N}(S_{mW}^{(i)})}{m}.$$

Now, as $(\max_i S_{mW}^{(i)})_{m\in\mathbb{N}}$ are i.i.d random variables, the law of large number gives us the inequality :

$$\gamma \geq \frac{\mathbb{E}[\max_{i=1\dots N}(S_1^{(i)})]}{W}. \tag{6}$$

If RTTs were independent for all receivers, we would be able to conclude immediately that there is an asymptotic behavior in $\ln(N)$. But this is not true as the RTTs of two receivers are made of a first common term which is the sum of the service times of the common routers they use from the source and of a second term, which is independent for each receiver.

Now for each receiver, there is at least a link that belongs only to the path from the source (this is indeed the last link). These links are supposed to have independent service times with the same exponential law s so that, applying Corollary 2, Equation (6) leads to the relation of the theorem. ■

It is possible to have a better lower bound for γ under some additionnal assumptions on the tree as we will see in the next subsection. Again this bound is reached by a category of trees, the perfect reverse umbrella tree (all links shared except the last one).

C. *Tree Topology Dependence : Description of Aggregation*

We have been able to bound γ, both from above and from below, by $\ln(N)$ functions. We now give a finer grain classification of tree topologies within these two bounds. The performance inside the interval defined by the bound found above depends essentially on the nature of the tree. We show that it is possible to create a partial order on the trees allowing one to achieve all throughputs within the interval between the lower and the upper bound. In this section, we consider trees with N receivers, under the assumptions of homogeneity and independence we described earlier.

C.1 Aggregation, Partial Order

In what follows we will assume that the service times on the backward tree are all equal to zero.

Consider two receivers i and j, with paths from the source to every of these two receivers. For every $l = 1, \dots D$, the service times $s_m^{(f(i,l))}$ and $s_m^{(f(j,l))}$ are either

- the same variable - when receivers i and j share their l-th link.
- two independent variables with the same law - when paths from the source to i and j are different at a given depth, and for all the following links in the tree.

Definition : Aggregation Due to the last statement, we can define aggregation of receivers i and j, that we will write $a(i,j)$, by

$$a(i,j) = \max\{l = 1 \dots D | s_m^{(f(i,l))} \equiv s_m^{(f(j,l))}\} \tag{7}$$

The aggregation is exactly the number of common service times in the two sums $S_m^{(i)}$ and $S_m^{(j)}$; all others terms of these sums are independent.

Aggregation indeed measures the correlation between two receivers: receivers with large aggregation appear to have similar performances. It is possible to show that a given aggregation characterizes a tree [5].

Definition : Aggregation order We will say that a tree T is less aggregated than another tree T' if their aggregation functions a and a' are such that $a \leq a'$. This is of course a partial order relation on trees. This order is compatible with the performance of the tree as shown in the next result.

Theorem 3: If T and T' are such that $a \leq a'$, then $\gamma_{a'} \leq \gamma_a$.

Proof: see the appendix of [17]. ■

This theorem gives us another proof for the upper and lower bounds of Section IV-B. The upper tree[6] (given by $a(i,j) = D-1$) provides the lower bound, the lower tree (given by $a(i,j) = 0$) provides the upper bound.

C.2 Umbrella Trees

Definition : Umbrella Tree of Class l A tree (given by its aggregation a) is said to be an umbrella tree of class l if we have $a \leq D - l$. It represents a tree that finishes for every receiver by a unicast connection of length at least l.

Umbrella trees of class l with l large typically correspond to a worst case situation, as receivers share few links. We have observed via simulation that for this type of trees, the throughput seems to degrade more severely when the number of receivers increases. In an umbrella tree, the lower bound for γ can be reached.

Theorem 4: Assume all service times are exponential random variables of parameter λ. Then the Lyapounov exponent of an umbrella tree of class l is bounded from below by a function that can be expanded as:

$$\frac{1}{W} R_l(N)(1 + o(1)) \text{ for } N \to +\infty, \tag{8}$$

where $R_l(N)$ is the unique solution (in X) of the equation

$$\exp(-\lambda X) \left(\sum_{k=0}^{l-1} \frac{\lambda^k X^k}{k!} \right) = \frac{1}{N} \tag{9}$$

located in the interval $(0, 1)$.

Proof: According to Theorem 3, we just need to verify this formula for the tree ($a = l$). The formula established in the proof of the lower bound, namely

$$\gamma \geq \frac{\mathbb{E}[\max_{i=1\ldots N}(S_1^{(i)})]}{W} \tag{10}$$

holds. Let us look at the performance of the tree ($\forall i, j$, $a(i,j) = l$). For all i, We have $S_1^{(i)} \geq s_1^{(f(i,D-l+1))} + \cdots + s_1^{(f(i,D))}$. We can then apply Corollary 3 to this sum of random variables which has a Gamma distribution of parameter (λ, l). ■

It is immediate to check that for all $l > 1$, the upper bound on the throughput based on R_l (9) improves on that of Theorem 2 (namely it is strictly smaller). However, as shown in the proof of Corollary 3, we have the equivalence $R_l(N) \sim \ln(N)$ as N tends to ∞, so that these bounds are asymptotically equivalent. At first glace, one may then think that there is no real improvement. Numerical evidence shows that for the range of the number of receivers considered in this paper, this improved bound is always much sharper than the previous one. For instance, for the umbrella tree (CL=0,FO=3,UL=3) of Figure 9, and for 8 receivers, the new upper bound (0.95) is much closer to the throughput provided by simulation (0.61) than the upper bound of Theorem 2 (2.89).

As we can see, in spite of all the optimistic assumption we made, an umbrella tree systematically results in a severe degradation of the throughput. The results observed in the algebraic simulation, as well as the intuition we had on tree topology impact, are confirmed and generalized by analytical results. Aggregation seems to be a key concept, as it gives us a parametric representation of the tree topology which allows direct performance comparison. An important result of this study is that umbrella trees, that are frequently encountered in the internet [9] suffer severe throughput degradation even in the case of light tailed fluctuation of the delay.

V. Conclusion

In this paper, we studied the impact of randomness (i.e. queueing delay) on the performance of a (one-to-many) multicast session in the presence of a "TCP-like" congestion control mechanism. With a simple analytical model, we analyzed the degradation of throughput when the size of the multicast group increases. In addition, we studied the impact of tree topology on the throughput of the multicast session.

In presence of a light tailed random noise, we show that the throughput decreases logarithmically when the number of receivers increases. We analytically find an upper and a lower bound for the throughput. Within these bounds, we characterize the degradation depending on the tree topology. A typical situation is that where the throughput severely decreases between 2 and 20 receivers; around 40 receivers, the throughput is only 50% of what it would be with a single receiver. In particular, we have identified a class of trees commonly found in IP multicast sessions [9], [14] as a worst case of throughput degradation. This observation is quantified by simulation and then explained analytically.

This work analytically proves that TCP-like congestion control might be harmful with reliable multicast transmission. Consequently, applications may prefer multi-rate control mechanisms to single rate reliable multicast transmission. This results extends to unreliable application that may find it difficult to manage a 50% drop in the average throughput when the number of receivers increases.

Multi-rate (layered) control mechanisms are best suited to multicast sessions with a large number of receivers. Rubenstein [11] has shown that multi-rate control can preserve TCP fairness with regard to TCP flows sharing the same congested node, while not penalizing all receivers in case of localized congestion. Subcasting (single group with filtering in nodes) is another

[5] a consequence from the study of the equivalence relation $i \approx_l^a i' \Leftrightarrow a(i,i') \geq l$ is that we can build the tree using only the aggregation function.

[6] when making the assumptions that the receivers are distinct.

area of investigation.

Another contribution of our work is a new analytical framework that can be used to study various problem related to flow and congestion control (in multicast and unicast environements).

In future works, we will extend and generalize our analytical framework. Extension to adaptive window size is possible based on a generalization of the (max,plus) representation of TCP Tahoe and Reno known for unicast [15].

In this framework, we will also study various sub-grouping approaches and try to define new classes of congestion control mechanism that might be applicable to unicast transmission as well. We will also analyze how TCP-like congestion control affect shared trees.

REFERENCES

[1] S. J. Golestani, K. K. Sabnani. "Fundamental Observations on Multicast Congestion Control in the Internet". Proceeding of IEEE ICNP '99. Ottawa. October 1999.

[2] F. Baccelli, D. Kofman and J. L. Rougier. "Self Organizing Hierarchical Multicast Trees and their Optimization". Proceedings of IEEE Infocom 1999. New York. April 1999.

[3] F. Baccelli, G. Cohen, G.J. Olsder and J. P. Quadrat. "Synchronization and Linearity". Wiley Editor. 1992.

[4] J.-Y. Le Boudec, P. Thiran and S. Giordano. "A short tutorial on Network Calculus I : fundamental bounds in communication networks". Proceedings of ISCAS'2000. Geneva. May 2000.

[5] J.-Y. Le Boudec, P. Thiran and S. Giordano. "A short tutorial on Network Calculus II : min-plus system theory applied to communication networks" Proceedings of ISCAS'2000. Geneva. May 2000.

[6] F. Baccelli and T. Bonald. "Window flow control in FIFO networks with cross-traffic". QUESTA. Special Issue on Stochastic Stability. May 1998.

[7] T. Lai and L. Robbins. "A class of dependent random variables and their maxima". Z. Wahrsch. 1978.

[8] R. E. Barlow and F. Proschan. "Statistical Theory of Reliability and Life Testing". Holt, Rinehart and Winston. New York. 1975.

[9] R. C. Chalmers and K. C. Almeroth. "Validating the Multicast Mystique". Submitted to IEEE Infocom 2001.

[10] L. Rizzo, L. Vicisano and J. Crowcroft. "TCP-like congestion control for layered multicast data transfer". Proceedings of IEEE Infocom 98. San Francisco. March 1998.

[11] D. Rubenstein, J. Kurose and D. Towsley. "The Impact of Multicast Layering on Network Fairness". Proceedings of ACM SIGCOMM 1999. Boston. August 1999.

[12] M. Handley and S. Floyd. "Strawman Congestion Control Specifications". IRTF RMRG report (available on the RMRG web site). December 1998.

[13] IRTF Reliable Multicast research group (RMRG). URL: www.east.isi.edu/RMRG/

[14] I. Stoica, T. S. Eugene Ng and H. Zhang. "REUNITE: A Recursive Unicast Approach to Multicast". Proceedings of IEEE INFOCOM 2000. Tel-Aviv. March 2000.

[15] F. Baccelli and D. Hong. "TCP is $(\max,+)$ Linear". Proceeding of ACM Sigcomm 2000. Stockholm. August 2000.

[16] S. Bhattacharyya, D. Towsley and J. Kurose. "The Loss Path Multiplicity Problem in Multicast Congestion Control" Proceedings of IEEE Infocom 1999. New-York. March 1999.

[17] A. Chaintreau, F. Baccelli and C. Diot. "An Analytical Framework for the Analysis of Multicast Congestion Control". INRIA report Nb. 3987, September 2000.

APPENDICES

A. Proof of the Main (max,plus) Representation Results

We start from the following result shown in [3].

Theorem 5: let $(\mathbb{A}_n)_{n\in\mathbb{N}}$ be a sequence of random square matrices in $\mathbb{R}_{\max}$ independent with same law and with coefficients which are either ϵ with probability 1, or with finite expectation; then we have :

$$\lim_{m\to\infty} \frac{\|\mathbb{A}_m \mathbb{A}_{m-1} \ldots \mathbb{A}_1\|}{m} = \gamma \tag{11}$$

in expectation and with probability 1, where γ is a constant called the (max,plus) Lyapounov exponent of this sequence of matrices.

Corollary 1: Under the assumptions of §II, the multicat model is such that

$$\lim_{m\to\infty} \frac{\|X_m\|}{m} = \gamma \tag{12}$$

in expectation and with probability 1, where γ is the Lyapounov exponent of the sequence $\{\mathbb{P}_n\}$.

Proof: The matrices $\mathbb{P}_n$ are i.i.d. so that they admit a Lyapounov exponent. In addition, here $\|X_m\| = \|Y_m\| = \|\mathbb{P}_m\mathbb{P}_{m-1}\ldots\mathbb{P}_1\|$. ■

B. Maximal Characteristics

We need an analytical tool, giving us the behavior of $\max_i^N X_i$ as a function of N and of the law of X_i, when the X_i's are independent and identically distributed. The maximal characteristics theory of Lai Robbins (described in [7]) provides such results, with a few assumptions on the law of X_i (verified by the exponential case and the Gamma law case).

Theorem 6 (Lai and Robbins maximal characteristics) Let $(\tilde{X}_m)_{m\in\mathbb{N}}$ be a sequence of $\mathbb{R}_+$-valued i.i.d. random variables. Assume that their common distribution function F satisfies :

$$\begin{array}{l} (\forall x \geq 0,\ F(x) < 1) \\ (\forall c > 1,\ \lim_{x\to+\infty} \frac{1-F(cx)}{1-F(x)} = 0) \end{array} \tag{13}$$

Let $m_N \hat{=} \inf\{x \geq 0, 1 - F(x) \leq 1/N\}$, then we have

$$\mathbb{E}[\max_{i=1\ldots N} \tilde{X}_i] = m_N(1 + o(1)),\ \text{for } N \to +\infty. \tag{14}$$

Here are two corollaries (the first of which is immediate):

Corollary 2 (exponential case) Let $(\tilde{X}_m)_{m\in\mathbb{N}}$ be a sequence of i.i.d exponential r.v.'s with parameter λ, then

$$\mathbb{E}[\max_{i=1\ldots N} \tilde{X}_i] = \mathbb{E}[\tilde{X}_1]\ln(N)(1 + o(1)),\ \text{for } N \to +\infty. \tag{15}$$

Corollary 3 (Gamma case) Let $(\tilde{X}_m)_{m\in\mathbb{N}}$ be a sequence of i.i.d Gamma random variables with parameter (λ, l) where $\lambda > 0$ and l an integer larger that or equal to 1; then for $N \to +\infty$,

$$\begin{aligned} \mathbb{E}[\max_{i=1\ldots N} \tilde{X}_i] &= R_l(N)(1 + o(1)) \\ &= \tfrac{1}{\lambda}\ln(N)(1 + o(1)), \end{aligned} \tag{16}$$

for $R_l(N)$ function defined by (9).

Proof: The distribution function is: $F : x \to 1 - (\sum_{k=0}^{l-1} \frac{\lambda^k x^k}{k!}) \exp(-\lambda x)$. F verifies the conditions (13), so that we can apply the previous result, and the formula for m_N gives the definition of R. The fact that $R(N) \sim \frac{1}{\lambda}\ln(N)$ is immediate from (9) when taking the logarithm on both sides. ■

0-7803-7016-3/01/$10.00 ©2001 IEEE