Delay Asymptotics and Scalability for Peer-to-peer Live Streaming

György Dán, Member, IEEE and Viktória Fodor, Member, IEEE
ACCESS Linnaeus Centre, School of Electrical Engineering
KTH, Royal Institute of Technology
Stockholm, Sweden
E-mail: {gyuri,vfodor}@ee.kth.se

***Abstract*— A large number of peer-to-peer streaming systems has been proposed and deployed in recent years. Yet, there is no clear understanding of how these systems scale and how multi-path and multihop transmission, properties of all recent systems, affect the quality experienced by the peers. In this paper we present an analytical study that considers the relationship between delay and loss for general overlays: we study the trade-off between the playback delay and the probability of missing a packet and we derive bounds on the scalability of the systems. We present an exact model of push-based overlays and show that the bounds hold under diverse conditions: in the presence of errors, under node churn, and when using forward error correction and various retransmission schemes.**

***Index Terms*— C.2.4.b Distributed applications, C.2.6.c Multicast, C.4.e Performance attributes**

I. Introduction

In an overlay multicast system streaming content is delivered by utilizing the users' upload capacities. Consequently, such a system is promising for the cheap delivery of streaming media to a large population of users. The architectures proposed for overlay multicast (a.k.a. peer-to-peer streaming) generally fall into one of two categories: multi-tree-based or mesh-based. Solutions of both categories utilize multi-path transmission. Multi-path transmission offers two advantages. First, disturbances on an overlay path lead to graceful quality degradation in the nodes. Second, the output bandwidth of the peers can be utilized more efficiently.

Multi-tree-based overlays follow the traditional approach of IP multicast: nodes are organized into multiple transmission trees and relay the data within the trees. The streaming data is divided into packets and packets are transmitted at round-robin through the transmission trees, providing path diversity for subsequent packets in this way. The transmission trees are constructed at the beginning of the streaming session and are maintained throughout the session by a centralized or a distributed protocol. Node churn leads to the disconnection of the trees and hence to data loss, which is one of the main deficiencies of multi-tree-based overlays.

Mesh-based overlays (also called swarming) follow the approach of batch peer-to-peer content distribution: nodes know about a subset of all nodes (their neighbors); they both receive data from and forward data to their neighbors. There is no global structure maintained, hence the scheduling of data transmissions is determined locally. Mesh-based overlays are resilient to node churn as forwarding decisions are taken based on the actual neighborhood information, but their efficiency depends on the scheduling algorithm.

Several works deal with the management of multi-tree-based overlays ([1], [2] and references therein) and with scheduling algorithms for mesh-based overlays ([3], [4], [5] and references therein). There are also numerous proposals on how to improve the robustness of the overlays to errors using coding techniques such as forward error correction (FEC), multiple description coding (MDC) and network coding [6]. The evaluation of the proposed solutions is mostly based on simulations and small scale measurements; the analytical modeling of overlay multicast has not received much attention.

There are a number of commercial deployments of overlay multicast, e.g. [7], [8]. Commercial systems often serve hundreds of thousands of peers simultaneously [9], yet little is known how they would behave if the number of concurrent users increased to its tenfold. We argue that there is a need for an analytical understanding of the performance of large systems in order to be able to design systems that can provide controllable and predictable quality under a wide range of operating conditions.

The most important difference between overlay multicast systems and peer-to-peer content distribution, such as Bittorrent, is the delay aspect: data should be delivered to the nodes before their playout deadline. The probability that data arrive before their playout deadline depends on the playback delay b: the lag between the time of the generation of a packet at the source and the time of the playback at the peers, as shown in Fig. 1. The necessary playback delay for providing good streaming quality may depend on many factors: the overlay's architecture and size, which determine the nodes' distances from the source; the per-hop delay distribution, the packet loss probability between the nodes and the error control solutions used; the scheduling of packet transmissions in pull based overlays; and the frequency of node departures and the time needed to reconnect to the overlay in the case of multi-tree-based overlays.

Our aim is to define benchmarking metrics for the performance evaluation of overlay multicast systems. Specifically, we consider two questions related to the playback delay. First, how fast does the probability of missing a packet decrease as a function of the playback delay. Second, how fast should the playback delay be increased to maintain the probability of

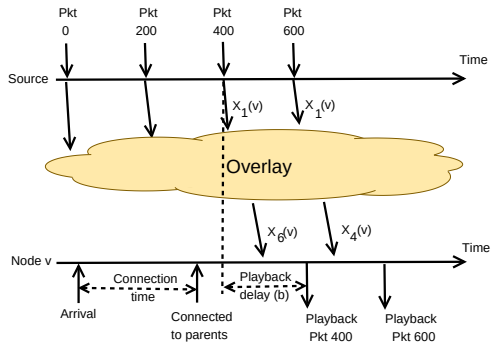

Fig. 1. The playback delay and the time needed to connect to the overlay.

missing a packet unchanged as the overlay's size increases. Through the derived bounds we can define the factors that influence the system scalability. The paper also presents an exact model of the temporal evolution of the data distribution in overlay multicast, and uses the exact model to illustrate how the derived bounds can be interpreted.

The rest of the paper is organized as follows. Section II gives an overview of the related work. Section III presents bounds on the playback delay and the scalability of the overlays based on the foundations of large deviation theory and on results on heavy-tailed distributions. We present an exact mathematical model of overlay multicast systems in Section IV. Section V discusses the delay bounds and the performance of the overlays based on the exact mathematical model, and we conclude our work in Section VI.

II. Related modeling work

The trade-off between the available resources and the number of nodes that can join the overlay was studied for overlay multicast systems utilizing a single transmission tree in [10]. The first models that describe the data distribution performance of multi-tree-based overlay multicast were proposed in [11], [12] and showed that these systems exhibit a phase-transition when using FEC. The effect of the forwarding capacity on multi-tree-based overlays was investigated in [13] using a queuing theoretic approach, and in [14] based on a fluid model. The delay characteristics of a mesh-based overlay were investigated in [4], and the authors showed an exponential relationship between the playback delay and the packet missing probability. The analytical results presented there are limited to a specific packet forwarding algorithm and to complete graphs. In [15] the authors considered a larger set of forwarding schemes, and showed the delay and throughput optimality of a fresh-data first forwarding scheme under certain conditions. In [16] the authors presented an analytical model of multi-tree-based overlays, which serves as the basis of the analytical model presented in this work. The focus in [16] was on identifying the primary sources of delay in overlay multicast and on comparing different prioritization schemes. We are however not aware of analytical results neither on the scalability of overlay multicast architectures in terms of delay, nor on the effects of the playback delay on the data delivery performance.

III. Delay bounds

We model the overlay as a directed graph $G = (V,E)$ with $N = |V|$ vertices. The set of vertices and edges can change over time due to node churn and due to the overlay management. We chose to omit the time dimension in our notation in order to ease understanding. Let us denote by s the source of the multicast, and by T_i the spanning tree rooted at the source, through which the copies of packet i reach the nodes in V. In a multi-tree-based overlay with τ trees the T_i are predetermined by the overlay maintenance entity and $\cup T_i = E$. In a pull-based (a.k.a. mesh-based) overlay the T_i are a result of local decisions taken in the nodes, such that all edges $(u,v) \in T_i$ are chosen from E. E is maintained by the overlay maintenance entity.

Let us denote by the random variable $L_i(v)$ the length of the simple overlay path from s to v in T_i, and the peer-to-peer per-hop delays by the non-negative random variables $X_h(v)$. For example, in Fig. 1, $L_{400}(v) = 6$ and $L_{600}(v) = 4$. The distribution of the $X_h(v)$ depends on many factors, e.g., the data distribution model (time spent for coordination between nodes), the probability of losses (due to churn and network congestion), the nodes' upload capacities, and the distance from the source (many proposed architectures place nodes with large upload capacities close to the source). Except for Theorem 1 we assume that the peer-to-peer per-hop delays on consecutive overlay hops are not correlated and follow the same probability distribution, i.e., the $X_h(v)$ are i.i.d. r.v.s. The possible reasons of correlation would be overlay optimization based on geographic locations or per-hop-delays. According to recent measurement studies (e.g., [9]) neighbor selection is however locality oblivious in practice, which supports the i.i.d. assumption.

The time it takes for packet i to reach node v from s, given that the length of the overlay path is l, is the sum of the per hop delays. Let us denote this conditional end-to-end delay by $D_i(v,l)$, $D_i(v,l) = \sum_{h=1}^{l} X_h(v)$. Based on the conditional end-to-end delays we can express the unconditioned end-to-end delay to node v

$$D_i(v) = \sum_{l=1}^{N-1} D_i(v,l) P(L_i(v) = l). \quad (1)$$

The probability that node v with playback delay b misses an arbitrary packet i is $P(D_i(v) > b)$. For our analysis we assume that every packet reaches every node after some finite amount of time, i.e., $\lim_{b\to\infty} P(D_i(v) > b) = 0$. Both multi-tree-based overlays with retransmissions and mesh-based overlays can fulfill this requirement.

A. Playback delay in stationary state

First, we consider an overlay in which N is a stationary process, that is, peers may join and leave the overlay but the average and the variance of the overlay's size does not change. In this case $L_i(v)$ is a stationary process as well. We describe the analytical results separate for the cases when the per-hop delays $X_h(v)$ follow distributions with finite moment generating functions (m.g.f.) and when they are heavy-tailed, that is, their m.g.f. is infinite.

Even though there is not much evidence of heavy-tailed end-to-end delay distributions in the Internet, we can identify three possible sources of heavy-tailed per-hop delay distributions. First, medium access control protocols used on multi-access broadcast channels often employ an exponential back-off retransmission scheme, e.g., CSMA/CA and CSMA/CD, which can lead to heavy-tailed delay distributions at the link layer [17], [18]. Second, interactions with cross traffic at the network layer can lead to heavy-tailed distributions in the presence of self-similar traffic [19]. Third, retransmission schemes that use an exponential back-off scheme at the transmission layer or at the application layer can lead to heavy-tailed per-hop delay distributions (e.g., long-range dependent like behavior observed in the case of TCP on timescales of practical interest [20]).

Nevertheless, in the case of delay sensitive applications, like streaming, per-hop delays with finite m.g.f. have practical significance. If the applications use some retransmission scheme at the transport layer or the application layer, then large delays originating in the network layer or in the link layer trigger retransmission requests at the transport or the application layer. The retransmissions cut the heavy tail of the lower layer delay distribution, and if the employed back-off scheme is slower than exponential, e.g., uniform or polynomial, then the resulting per-hop delay distribution will have a finite m.g.f.

The case of finite m.g.f.: We start the evaluation with the case when the per-hop delays $X_h(v)$ follow distributions with finite moment generating functions (m.g.f), i.e., a light-tailed distribution. First we show that if the per-hop delays $X_h(v)$ follow light tailed distributions then the end-to-end delay $D_i(v,l)$ from the source s to a node v on an overlay path has a light-tailed distribution as well, even if the per-hop delays are not i.i.d.

Lemma 1: Given non-negative random variables X_h ($h = 1 \dots n, n > 0$) with marginal p.d.f $f_h(x)$ and joint p.d.f $f(x_1,\dots,x_n)$ such that $E[e^{\theta X_h}] < \infty$, then $S_n = \sum_{h=1}^{n} X_h$ has $E[e^{\theta S_n}] < \infty$ as well, even if they are positively correlated.

Proof: For independent r.v.s the proof is trivial, $E[e^{\theta S_n}] = \Pi_{h=1}^{n} E[e^{\theta X_h}]$. For correlated r.v.s, we prove the lemma for $n = 2$, induction can be used for $n > 2$. Let us order X_1 and X_2 such that $\int_0^x f_1(t)dt \leq \int_0^x f_2(t)dt$ for $\forall\ x > x_0$ ($x_0 > 0$). Let us denote by X_2^{**} a random variable that is distributed as X_2 but is in perfect positive dependence with X_1 (see [21] for a definition), that is $x_2 = g(x_1)$ for some function g. $E[e^{\theta(X_1+X_2)}]$ is a convex, monotonically increasing function, hence [21]

$$E[e^{\theta S_2}] = E[e^{\theta(X_1+X_2)}] \leq E[e^{\theta(X_1+X_2^{**})}]. \quad (2)$$

Since there exists $\theta' > 0$ such that $E[e^{\theta' X_1}] < \infty$, for all $0 < \theta \leq \theta'/2$ it holds that

$$\begin{aligned} E[e^{\theta S_2}] &= \int_0^\infty \int_0^\infty e^{(x_1+x_2)\theta} f(x_1,x_2)dx_1dx_2 \\ &\leq \int_0^\infty e^{(x_1+g(x_1))\theta} f_1(x)dx_1 \quad (3) \\ &\leq a(x_0,\theta) + \int_{x_0}^\infty e^{2x_1\theta} f_1(x)dx_1 < \infty, \quad (4) \end{aligned}$$

where (3) holds because of (2) and (4) holds because $g(x) \leq x$ for $x > x_0$ due to the ordering of X_1 and X_2. ■

By Lemma 1 the distribution of $D_i(v,l)$ has finite m.g.f. The end-to-end delay $D_i(v)$ for an arbitrary packet and node is a linear combination of the $D_i(v,l)$ as given by (1), so that the following lemma applies to $D_i(v)$.

Lemma 2: Consider the non-negative random variables S_n ($n = 1 \dots N-1, N \geq 2$) such that $E[e^{\theta S_n}] < \infty$ for some $\theta > 0$. Let the r.v. S be a linear combination of the S_n, $S = \sum_{n=1}^{N-1} p_n S_n$ such that $\sum p_i = 1$, $p_i \geq 0$. Then $E[e^{\theta S}] < \infty$ for some $\theta > 0$, that is, the property of finite m.g.f. is preserved through the operation of linear combination.

Proof: Recalling one of the basic properties of m.g.f.s, for $S = \sum_{n=1}^{N-1} p_n S_n$, the m.g.f. of the r.v. S is

$$E[e^{\theta S}] = \sum_{n=1}^{N-1} p_n E[e^{\theta S_n}] \leq \sum_{n=1}^{N-1} E[e^{\theta S_n}] < \infty.$$

■

By Lemma 2, the distribution of $D_i(v)$ has finite m.g.f., that is, the end-to-end delay as seen by an arbitrary node for an arbitrary packet has finite m.g.f. We can prove the following theorem based on results from large deviation theory [22].

Theorem 1: The decrease of the probability that an arbitrary node with playback delay b misses an arbitrary packet in an overlay with N nodes is asymptotically at least exponential in b if the per-hop delays have finite m.g.f.

Proof: From Lemma 1 and 2 it follows that the distribution of $D_i(v)$ has finite m.g.f. In the following we show that the decrease of $P(D_i(v) \geq b)$ is asymptotically at least exponential in b.

Recall the Chernoff bound from large deviation theory. For the average of n i.i.d random variables X and $x > E[X]$

$$P\left(\frac{\sum_{j=1}^{n} X_j}{n} \geq x\right) \leq e^{-nI(x)}, \quad (5)$$

where $I(x)$ is the rate function given by

$$I(x) = max_{\theta>0}\theta x - ln(E(e^{X\theta})).$$

Fig. 2 shows the rate functions for two distributions with different parameters. The rate function $I(x)$ is convex for scalar random variables, is monotonically increasing on $(E[X], \infty)$ and $I(E[X]) = 0$ [22]. For non-negative r.v.s X with $E[X] \geq 0$, the derivative $\frac{\partial I(x)}{\partial x}|_{x_0} \geq I(x_0)/x_0$ for all $x_0 > E[X]$. Consequently, for $x > E[X]$ and $a > 1$ we can write

$$I(ax) \geq I(x) + (ax - x)I(x)/x = aI(x) \quad (6)$$

and hence

$$P\left(\frac{\sum_{j=1}^{n} X_j}{n} \geq ax\right) \leq e^{-nI(ax)} \leq e^{-anI(x)} = \left(e^{-nI(x)}\right)^a. \quad (7)$$

We can set $n = 1$ and apply (5) and (7) to the r.v. $D_i(v)$

$$P(D_i(v) \geq b) \leq e^{-I(b)},$$

and

$$P(D_i(v) \geq ab) \leq e^{-I(ab)} \leq \left(e^{-I(b)}\right)^a. \quad (8)$$

Eq. (8) holds for any $a > 1$, which proves the theorem. ■

The result is independent of the distribution $P(L_i(v) = l)$, and holds whenever there is enough forwarding capacity in the overlay. It is also independent of the number of packets in

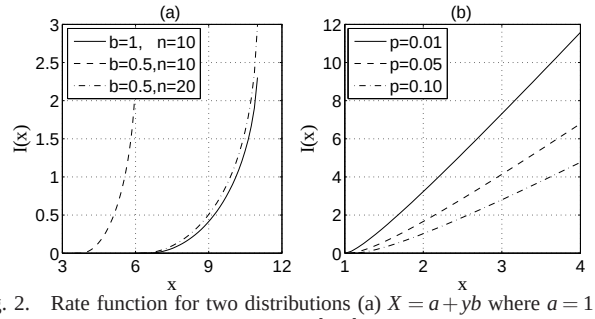

Fig. 2. Rate function for two distributions (a) $X = a + yb$ where $a = 1$ and y has discrete uniform distribution on $[1,n]$ and (b) geometric distribution with failure probability p.

the stream and does not make any assumption on the graph's connectivity or the distribution scheme, in particular, it does not assume a complete graph. The simulation results presented in [4] support our analytical result for pull-based overlays, and we show results later that support the theorem for multi-tree-based overlays.

The case of heavy-tailed distributions: Most practical heavy-tailed distributions, such as the Weibull, the Pareto and the log-normal distribution, belong to the class of subexponential distributions, which is a subclass of heavy-tailed distributions. Hence, we focus on the case when the per-hop delays $X_h(v)$ are i.i.d. and follow a subexponential distribution. We recall the definition of the subexponential property and its relevant consequences from [23].

Definition 1: (Subexponential distribution function) *Let X_i ($i \in \mathbb{N}$) be i.i.d. random variables with distribution function $F(x) < 1$ for all $x > 0$. Let us denote by $\overline{F}(x) = 1 - F(x)$ the tail of F and by $\overline{F}^{l*}(x) = P(X_1 + \cdots + X_l > x)$ the tail of the l-fold convolution of F. The distribution F is subexponential if*

$$\frac{\overline{F}^{l*}(x)}{\overline{F}(x)} \sim l. \tag{9}$$

As shown in Theorem 5.2 in [23], the random sums of subexponential distributions can be characterized as follows. Let p_l be a probability measure on $\mathbb{N}_0$. If

$$\sum_{l=0}^{\infty} p_l (1+\varepsilon)^l < \infty \tag{10}$$

for some $\varepsilon > 0$ and $G(x) = \sum_{l=0}^{\infty} p_l F^{l*}(x)$, then

$$\frac{\overline{G}(x)}{\overline{F}(x)} \sim \sum_{l=0}^{\infty} l p_l \tag{11}$$

If p_l expresses $P(L_i(v) = l)$ then the condition (10) holds because $L_i(v) < N$. We can substitute $\overline{G}(x)$ by $P(D_i(v) > b)$ and $\overline{F}(x)$ by $P(X_h(v) > b)$ and based on (11) we can write

$$P(D_i(v) > b) \sim E[L_i(v)] P(X_h(v) > b). \tag{12}$$

Consequently, in the presence of subexponential per-hop delays the packet missing probability is subexponential as well, i.e., the decrease of the packet missing probability is asymptotically slower than exponential as the playback delay increases. If the per hop delay statistics are known, then the packet missing probabilities can be predicted from (12).

B. Scalability

In this section we evaluate the effect of the increase of the overlay's size on the probability of missing a packet and on the necessary playback delay for keeping the packet missing probability constant. To decouple the problem of scaling in terms of playback delay and the problem of scaling in terms of overlay maintenance we consider the scaling of the path length distribution $L_i(v)$ given in our analysis: it is determined by the overlay maintenance entity.

In the following we show that, under certain conditions, if one would like to to keep the probability of packet missing unchanged then it is sufficient to increase the playback delay proportional to the the increase of $E[L_i(v)]$. Again, we treat the case of light-tailed and heavy-tailed per-hop delay distributions separate.

The case of finite m.g.f: Let the $X_h(v)$ be i.i.d. random variables with $M(\theta) < \infty$. We note that there are no asymptotic results available for correlated r.v.s, but we conjecture that the following theorem holds for correlated and for non identically distributed r.v.s as long as $E[X_h(v)]$ is bounded from above, and leave the proof to be subject of future work.

Theorem 2: The increase of the playback delay b needed to maintain the probability of missing an arbitrary packet unchanged is at most proportional to the increase of the path length $L_i(v)$.

Proof: To prove the theorem we look for a $d \geq 0$ such that for $a \geq 0$

$$P(D_i(v) \geq b | L_i(v) = l) = P(D_i(v) \geq b + d | L_i(v) = l + a). \tag{13}$$

We use Chernoff's bound (5) on the deviation of the average of i.i.d. random variables [22], hence we rewrite (13)

$$P(\frac{D_i(v)}{l} \geq \frac{b}{l} | L_i(v) = l) = P(\frac{D_i(v)}{l+a} \geq \frac{b+d}{l+a} | L_i(v) = l + a)$$

and express the upper bounds according to (5)

$$e^{-lI(\frac{b}{l})} = e^{-(l+a)I(\frac{b+d}{l+a})}. \tag{14}$$

We omit the base and rearrange the exponents to get

$$l\left[I(\frac{b}{l}) - I(\frac{b+d}{l+a})\right] = aI(\frac{b+d}{l+a}). \tag{15}$$

The right hand side of (15) is always positive. As the rate function is convex and monotonically increasing on $(E[X_h(v)], \infty)$ we have the condition

$$\frac{b+d}{l+a} \leq \frac{b}{l}, \tag{16}$$

that is

$$\frac{d}{b} \leq \frac{a}{l}, \tag{17}$$

which proves the theorem. ■

The following theorem establishes a similar results but with respect to the increase of the mean hop-count as seen by a node.

Theorem 3: The increase of the playback delay b needed to maintain the probability of missing an arbitrary packet unchanged is at most proportional to the increase of the mean path length $E[L_i(v)]$ if the standard deviation of the path length distribution does not increase faster than $E[L_i(v)]$.

Proof: Let us recall the Chebyshev inequality and apply it to the end-to-end delay $D_i(v)$

$$P(|D_i(v) - E[D_i(v)]| \geq A) \leq \frac{Var[D_i(v)]}{A^2}. \quad (18)$$

The end-to-end delay $D_i(v)$ is a compound random variable, hence its mean is $E[D_i(v)] = E[L_i(v)]E[X_h(v)]$ and its variance can be calculated as

$$Var[D_i(v)] = E[L_i(v)]Var[X_h(v)] + E[X_h(v)]^2 Var[L_i(v)]. \quad (19)$$

We can substitute (19) into (18) and introduce $b = A + E[D_i(v)]$. Then for $b > 2E[D_i(v)]$ we get

$$P(D_i(v) \geq b) \leq \frac{E[L_i(v)]Var[X_h(v)] + E[X]^2 Var[L_i(v)]}{(b - E[L_i(v)]E[X_h(v)])^2}. \quad (20)$$

Consider now that $E[L_i(v)|N = n_2] = (1+a)E[L_i(v)|N = n_1]$ and $Var[L_i(v)|N = n_2] \leq (1+a)^2 Var[L_i(v)|N = n_1]$ then

$$\begin{aligned} &P(D_i(v) \geq b(1+a)) \leq \\ &\leq \frac{(1+a)E[L_i(v)]Var[X_h(v)] + E[X]^2(1+a)^2 Var[L_i(v)]}{(b(1+a) - (1+a)E[L_i(v)]E[X_h(v)])^2} \\ &\leq \frac{(1+a)^2 E[L_i(v)]Var[X_h(v)] + E[X]^2(1+a)^2 Var[L_i(v)]}{(b(1+a) - (1+a)E[L_i(v)]E[X_h(v)])^2} \\ &\leq \frac{E[L_i(v)]Var[X_h(v)] + E[X]^2 Var[L_i(v)]}{(b - E[L_i(v)]E[X_h(v)])^2}, \end{aligned} \quad (21)$$

which is the same as the right hand side of (20). ■

In general, (17) shows that it is sufficient to increase the playback delay at the same pace as the depth of the spanning trees grows in order to maintain the probability of missing a packet unchanged. Theorem 3 generalizes the result to the growth rate of the mean hop count under certain conditions. E.g., if the nodes' distances from the source grow as $O(logN)$ then the playback delay should be increased proportionally to the logarithm of the growth of the overlay to keep the packet missing probability constant. Nevertheless, if the conditions of Theorem 3 are not satisfied then the playback delay might have to be increased faster than proportional to the increase of the mean hop-count. Consequently, one should not only look at the mean of the hop-count but also at its variance.

Unfortunately, the converse of the theorem cannot be proved: *there is no upper bound on the increase of the packet missing probability for constant playback delay as the overlay's size grows*, because the increase depends on the shape of the rate function itself. (For certain classes of per-hop-delay distributions and hop-count distributions one can derive asymptotic bounds based on the results shown in [24].)

The case of heavy-tailed distributions: For subexponential distributions, based on Theorem 5.2 in [23],

$$P(D_i(v) > b) \sim E[L_i(v)]P(X_h(v) > b), \quad (22)$$

which means that the increase of the packet missing probability is proportional to the increase of the mean path length. This result does not give however any bound on how much the playback delay has to be increased in order to maintain the probability of packet missing unchanged. In the following we show that for the class of distributions with a regularly varying tail, which is a subclass of subexponential distributions [23], a bound similar to Theorem 2 can be given. First we define the class of distributions with a regularly varying tail.

Definition 2: A positive measurable function f is said to be regularly varying with index α, denoted as $f \in \mathcal{R}(\alpha)$, for $\alpha \in \mathbb{R}$ if

$$\lim_{x\to\infty} \frac{f(tx)}{f(x)} = t^\alpha \quad \forall t > 0. \quad (23)$$

Definition 3: If a distribution function F has a regularly varying tail with index $-\alpha$, denoted as $\overline{F} \in \mathcal{R}(-\alpha)$, then

$$\overline{F}(x) = x^{-\alpha} m(x), \quad x > 0 \quad (24)$$

for some $m \in \mathcal{R}(0)$.

Distributions with regularly varying tail are, for example, the Pareto and the log-gamma distributions. For distributions with a regularly varying tail and finite mean the following scaling law applies, similar to Theorem 2.

Theorem 4: If the per hop delay distribution has regularly varying tail, then the increase of the playback delay b needed to maintain the probability of missing an arbitrary packet unchanged is proportional to the increase of the average path length in the overlay.

Proof: We prove the theorem by showing that the upper bound of the playback delay increases in proportion with the number of hops. We look for a $d \geq 0$ such that for $a \geq 0$

$$P(D_i(v) \geq b|E[L_i(v)] = l) = P(D_i(v) \geq b+d)|E[L_i(v)] = l+a). \quad (25)$$

Due to the subexponentiality and (22) we can find the upper bound on the necessary increase of the playback delay if we find a d such that

$$l\overline{F}(b) \geq (l+a)\overline{F}(b+d), \quad (26)$$

where $\overline{F} = P(X_h(v) > b)$. Since $\overline{F} \in \mathcal{R}(-\alpha)$, (26) becomes

$$lb^{-\alpha} m(b) \geq (l+a)(b+d)^{-\alpha} m(b+d) \quad (27)$$

for some $\alpha > 1$ and m. We can express d/b as

$$\frac{d}{b} \leq \sqrt[\alpha]{\frac{l+a}{l} \; \frac{m(b(1+d/b))}{m(b)}} - 1. \quad (28)$$

Since $m \in \mathcal{R}(0)$, for $b \to \infty$ we get

$$\frac{d}{b} \leq \sqrt[\alpha]{1 + \frac{a}{l}} - 1 \leq \frac{a}{l}, \quad (29)$$

which proves the theorem for $\alpha > 1$. ■

The asymptotic results of Theorem 1 indicate that *the exponential decrease* of the packet missing probability as a function of the playback delay *is not a good measure of the efficiency of a scheduling scheme* in overlay multicast: all scheduling schemes that manage to distribute the data to all nodes have this property if the per-hop delays have finite m.g.f. Alternatively, the fact that the packet missing probability decreases exponentially fast as a function of the playback delay shows that the per-hop delays are light-tailed, but does not show how the packet missing probability would change if the overlay's size increased (e.g., [4]). The results also show that it is necessary to control the asymptotic behavior of the per-hop delay at the application layer, in order to allow the efficient control of the playout positions in the overlay.

The scaling of the overlay path lengths with respect to the overlay's size is however a good measure of the scalability of the overlay maintenance and packet scheduling algorithms. Theorems 2 and 4 show that the playback delay does not have to be increased faster than proportional to the increase of the overlay path lengths to keep the packet missing probability constant. It also means however that nodes should adjust their playout positions (i.e., their playback delays) according to the size of the overlay in order to avoid buffer underruns. At the same time, one cannot draw conclusions on the scaling properties of the overlay by showing how the probability of packet missing increases for a fixed playback delay as a function of the overlay's size. We refer to [25] for an example where the authors showed the scalability according to Theorem 2, and to [26] for an example where they did not.

IV. Data distribution model

In the following we present an analytical model of an overlay multicast system and use the model to verify the delay bounds discussed in Section III. The model was developed with multi-tree-based overlays in mind (e.g., proposed in [1], [27], [28], [29]), but it can be extended to mesh-based overlays. We quantify the performance of the data distribution via the probability $\pi(b)$ that an arbitrary node receives or can reconstruct (i.e., possesses) an arbitrary packet in the overlay within the playback delay b. If we denote by $A_v(b)$ the number of packets possessed by node v in an arbitrary block of packets, then $\pi(b)$ can be expressed as the average ratio of the number of packets possessed in a block of n packets over all nodes, i.e., $\pi(b) = E[\sum_v A_v(b)/n/N]$. The probability of missing a packet is directly related to the possession probability: $P(D_i(v) \geq b) = 1-\pi(b)$.

A. System description

We denote the number of trees in the overlay by τ. We assume the existence of a tree maintenance entity (centralized [27] or decentralized [30], [31]) that finds suitable predecessors for arriving nodes and for nodes that lose their predecessors due to node churn or preemption [1], [29]. We denote by $\mathcal{L}_m(v)$ the level of node v in tree m. Packet i is distributed in tree $m = (i \bmod \tau)+1$. To simplify the notation, we introduce the notion of stripe, and say that packet i belongs to stripe m if it is distributed in tree m.

We consider two forms of error control: forward error correction and retransmissions. When forward error correction (FEC) is used, the source adds c redundant packets to every k packets, resulting in a block length of $n = k+c$. We denote this FEC scheme by FEC(n,k). Once a node receives at least k packets of a block of n packets, it may recover the remaining c packets, and forwards the reconstructed packets if necessary. Block based FEC can be used to implement PET and the MDC scheme considered in [27], where different blocks (layers) of data are protected with different FEC codes: the probability of reception for the different blocks depends on the strength of the FEC codes protecting them.

We consider three retransmission schemes. A node that detects a packet loss in stripe m requests the retransmission of the packet by one of these three strategies:

(RP) from its *predecessor* in tree m. If the loss is due to node churn then the node will have to wait until a new predecessor is found.

(RB) from a *backup* predecessor node, a node that is forwarding packets in tree m in the same level as the node's actual predecessor. This scheme assumes that every node maintains a list of backup nodes, but we do not model the overhead of maintaining such a list.

(RA) from a predecessor in *another tree*. A predecessor in another tree is likely to be far away from the source in tree m, hence the retransmission might take longer than using a backup list.

For the RP and the RB strategies, the level of node v in the spanning tree T_i through which packet i reaches it ($L_i(v)$) is the same as the level of node v in the tree in which packet i should be distributed ($\mathcal{L}_{i \bmod \tau}(v)$), i.e., $L_i(v) = \mathcal{L}_{i \bmod \tau}(v)$. For the RA strategy $L_i(v) \geq \mathcal{L}_{i \bmod \tau}(v)$, i.e., due to the retransmissions the spanning tree T_i can be deeper than the trees maintained by the overlay maintenance entity. We will discuss other aspects of retransmissions in Section V-C.

B. Analytical model

Let us denote by $L = max_{m,v}\mathcal{L}_m(v)$ the number of levels in the overlay. For simplicity, we assume that a node is in the same level in the trees in which it forwards data. Similarly, we assume that a node is the same level in the trees in which it does not forward data, and denote the level by $L_l = \mathcal{L}_m(v)$ for nodes that forward data in level l. Typically, $L-1 \leq L_l \leq L$ in well-maintained multi-tree-based overlays. The model builds on the simplifying assumption that the probability that a node is in possession of a packet is independent of whether another node in the same level is in possession of a packet. For brevity, we show equations for the case when n is a multiple of τ, and the output bandwidths are equal for all nodes and large enough to upload at the stream's rate. Consequently, the number of successors O_v is equal for all nodes and $O_v \geq \tau$. We model the behavior of the overlay in the presence of independent packet losses. We denote the loss probability on every overlay hop by p to ease understanding, but hop-dependent values of p can be used in the model. We show equations for the homogeneous case here to ease understanding, though in Section V we show results for heterogeneous output bandwidths and loss probabilities. Heterogeneous input and output bandwidths, loss correlations and heterogeneous losses can be modeled by following the procedure presented in [32].

We introduce random variables to model the one-way delay (the time it takes for a packet to travel between two nodes of the overlay if it is not lost, T_{dra}), the packet loss detection times (the time it takes for a node to detect that a packet will not arrive, T_{dld}, and the time it takes for a node to detect that a retransmission request or the retransmitted packet was lost, T_{rld}) and the time needed for retransmissions (the time it takes for a retransmission request to reach its destination, T_{rra}, and the time it takes for a retransmitted packet to reach the requestor, T_{rrb}). The subscripts are mnemonics, and we denote by f_{xxx} the probability density function (pdf) of T_{xxx} in

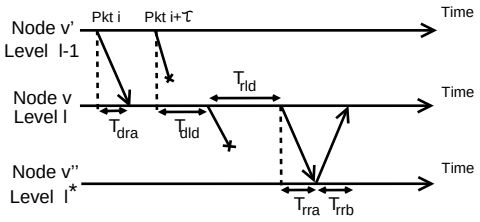

Fig. 3. Delivery of two packets to node v. The first packet is delivered directly ($X_l(v) = T_{dra}$). The second packet is received after two retransmission requests ($X_{l^*+1}(v) = T_{dld} + T_{rld} + T_{rra} + T_{rrb}$).

the paper. Fig. 3 shows some of the delays and their notations used in the model. The r.v.s X_h of the model used to obtain the delay bounds in Section III can be mapped to the sums of the T_{xxx} depending on the retransmission scheme used.

The key to the performance of the overlay is the probability $\rho_{j,l}(t)$ that a node in level l receives an arbitrary packet of stripe j no later than t time *after the first packet of the FEC block the packet belongs to is ready to be sent out from the source*. Let us introduce the binary random variable $R_{j,l}(t)$, such that $P(R_{j,l}(t) = 1) = \rho_{j,l}(t)$. The probability that nodes receive data from other nodes is determined by the probability that a node that forwards data in a tree can forward the data to its successors. Hence, we introduce the probabilities $\pi^f_{j,l}(t)$ that a node that is in level l in the tree where it forwards data possesses an arbitrary packet in stripe j no later than t. Fig. 4 illustrates $\rho_{j,l}(t)$ and $R_{j,l}(t)$ in an overlay with $\tau = 4$, $n = 4$ and two levels.

In the following we present a system of algebraic and differential equations of convolution type that describes the evolution of this probability. One can interpret the following equations and variables as the state equations and the state space of a system, and the solution is the response of the system to the input signal given by (39).

We describe the state of the nodes in level l with respect to a packet in stripe j by the following state variables. The probability that by time t a node has received the packet directly from its predecessor depends on the evolution of $\pi^f_{j,l-1}$, the probability that the predecessor (in level $l-1$ of the tree) possesses the packet and on the pdf of the forward per-hop delay, f_{dra}.

$$\frac{\partial \alpha_{j,l}(t)}{\partial t} = \int_0^t \frac{\partial \pi^f_{j,l-1}(t-\upsilon)}{\partial t} f_{dra}(\upsilon) d\upsilon, \tag{30}$$

where $\int_0^\infty f_{dra}(t)dt = 1-p$. If the packet is lost, the node has to detect that it will not receive the packet from its predecessor. The evolution of this state variable depends on f_{dld} ($\int_0^\infty f_{dld}(t)dt = p$), the pdf of the time necessary to detect packet loss

$$\beta_{j,l}(t) = \int_0^t \frac{\partial \pi^f_{j,l-1}(t-\upsilon)}{\partial t} f_{dld}(\upsilon) d\upsilon. \tag{31}$$

The node triggers a retransmission request at time t in three cases: if it detects a lost packet from its predecessor according to $\beta_{j,l}$, if it receives a message about unsuccessful retransmission according to $\phi_{j,l}$ or if it detects that its retransmission request message or the retransmitted packet had been lost according to $\psi_{j,l}$

$$\gamma_{j,l}(t) = \beta_{j,l}(t) + \phi_{j,l}(t) + \psi_{j,l}(t). \tag{32}$$

The retransmission request arrives to the corresponding predecessor (determined by the retransmission scheme used) depending on the pdf of the one way delay f_{rra}, ($\int_0^\infty f_{rra}(t)dt = 1-p$)

$$\delta_{j,l}(t) = \int_0^t \gamma_{j,l}(t-\upsilon) f_{rra}(\upsilon) d\upsilon. \tag{33}$$

Let us denote by l^* the level of the corresponding predecessor (e.g., l^* is $l-1$, $l-1$ and L_l for the *RP*, *RB* and *RA* retransmission schemes respectively). The retransmitted packet arrives to the node according to $\varepsilon_{j,l}$, depending on the probability that the node addressed by the retransmission request possesses the packet, π_{j,l^*}, and depending on the pdf of the one way delay, f_{rrb} ($\int_0^\infty f_{rrb}(t)dt = 1-p$).

$$\frac{\partial \varepsilon_{j,l}(t)}{\partial t} = \int_0^t \delta_{j,l}(t-\upsilon)\pi_{j,l^*}(t-\upsilon) f_{rrb}(\upsilon) d\upsilon. \tag{34}$$

If the node addressed by the retransmission request does not possess the requested packet, its negative acknowledgment arrives to the node depending on f_{rrb},

$$\phi_{j,l}(t) = \int_0^t \delta_{j,l}(t-\upsilon)(1-\pi_{j,l^*}(t-\upsilon)) f_{rrb}(\upsilon) d\upsilon. \tag{35}$$

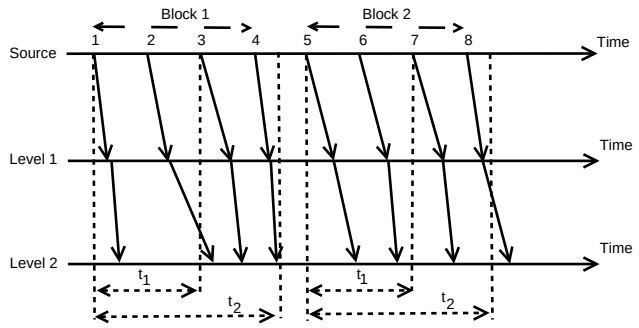

Fig. 4. $\rho_{j,l}(t)$ and $R_{j,l}(t)$ for t_1 and t_2 and two blocks of data. $\rho_{2,2}(t_1) = 0.5$, $\rho_{2,2}(t_2) = 1$, $\rho_{4,2}(t_1) = 0$, $\rho_{4,2}(t_2) = 0.5$.

τ	Number of trees in the overlay
n	FEC block length
k	Number of data pkts. in FEC block
L	Number of levels in the overlay
O_v	Outdegree of node v
C_v	Output capacity of node v
N	Number of nodes in the overlay
N_l	Number of nodes in level l of the overlay
a	Mean packet size
B	Stream bitrate
$f_{xxx}(.)$	PDF of one way delay
p	Pkt. loss probability on overlay hop
$\rho_{j,l}(t)$	Prob. of packet reception in stripe j in level l at time t
$\pi_{j,l}(t)$	Prob. of packet possession in stripe j in level l at time t
$\pi(b)$	Prob. of packet possession with playback delay b

TABLE I
LIST OF NOTATIONS USED IN THE DATA DISTRIBUTION MODEL.

The node detects the retransmission failure (due to a lost retransmission request or a lost retransmission) according to

$$\psi_{j,l}(t) = \int_0^t \gamma_{j,l}(t-\upsilon) f_{rld}(\upsilon) d\upsilon, \quad (36)$$

both $\phi_{j,l}$ and $\psi_{j,l}$ being input for (32), and $\int_0^\infty f_{rld}(t)dt = 1-(1-p)^2$).

Finally, the packet is received by the node either directly from its predecessor or through retransmission, that is

$$\frac{\partial \rho_{j,l}(t)}{\partial t} = \frac{\partial \alpha_{j,l}(t)}{\partial t} + \frac{\partial \varepsilon_{j,l}(t)}{\partial t}. \quad (37)$$

The probability of packet possession at time t for stripe j depends on the packet reception probability $\rho_{j,l}(t)$ and the possibility of reconstruction using FEC. A node possesses a packet of stripe j either if it receives it by time t or if it can reconstruct it using the packets received in the other stripes, i.e., it receives at least k out of the remaining $n-1$ packets,

$$\pi^f_{j,l}(t) = \rho_{j,l}(t) + (1-\rho_{j,l}(t))P(\sum_{i\neq j} R_{i,l_i}(t) \geq k), \quad (38)$$

where $l_i = l$ for stripes in which the node is fertile and $l_i = L_l$ for stripes in which the node is sterile. For simplicity we assume that nodes are in the same level in the trees in which they do not forward data.

The initial condition of the problem is given by the time packets are ready to be sent out from the root node. If the packets of an FEC block are sent out smoothed over na/B time then

$$\pi^f_{j,0}(t) = H(t-(j-1)a/B), \quad (39)$$

where a is the mean packet size, B is the stream's bitrate and $H(\cdot)$ is the unit step function.

We solve the above system of differential-algebraic equations numerically in an iterative way. For playback delay b the value of $\rho_{j,l}(t)$ has to be evaluated for $t \leq b+(n-1)a/B$.

Based on the probabilities $\pi^f_{j,l}(t)$ we can express $\pi_{j,l}(b)$ $(1 \leq l \leq L)$, the probability that a node that is l hops away from the source in the tree where stripe j is distributed possesses an arbitrary packet before its playout deadline given the playback delay b. The playout deadline for a packet in stripe j is $t_j = b+(j-1)a/B$, so that

$$\pi_{j,l}(b) \quad = \quad \rho_{j,l}(t_j) + (1-\rho_{j,l}(t_j)P(\sum_{i\neq j} R_{i,l_i}(t_j) \geq k).$$

The probability that an arbitrary node possesses a packet is

$$\pi(b) = \frac{1}{n}\sum_{j=1}^{n}\frac{1}{N}\sum_{l=1}^{L}\pi_{j,l}(b)N_l, \quad (40)$$

where N_l is the number of nodes l hops away from the source. We will show how to estimate N_l for multi-tree-based overlays in Section V-B.

The computational complexity of the calculation is $O(L|f_d|)$, where $|f_d|$ is the length of the vectors used to approximate the pdfs of the delay distributions. As L is $O(logN)$ in the considered overlays, the algorithm scales well with the number of nodes in the overlay.

C. Modeling node churn

Following the arguments presented in [32], the effects of node departures on a multi-tree-based overlay that employs FEC can be incorporated in the model in the following way. We denote by $E[\Xi]$ the mean time it takes for a node to find a predecessor, by $E[\Omega]$ the mean holding time of a predecessor, and by $E[M]$ the mean node lifetime. Measured values for these quantities were shown in, e.g., [1], [29]. Let us denote by $\kappa = E[\Omega]/E[\Xi]$ the ratio of the average time before the departure of a predecessor node and the average time to find a new predecessor as seen by a node. Furthermore, we denote by $\alpha = E[\Omega]/E[M]$ the ratio of the average time before the departure of a predecessor node and the average node lifetime. If nodes have i disconnected predecessors upon their arrival then the average ratio of their disconnected predecessors as seen by a random observer is

$$E[\Delta_i] = \frac{\tau + i\alpha}{\tau(\kappa+\alpha+1)}. \quad (41)$$

One can then use $p = E[\Delta_i]$ in the model to estimate the overlay's performance in the presence of node churn. Simulation results in [32] verify the accuracy of this approach for FEC.

D. Applying the model to pull-based systems

The model can be applied to pull-based systems given the distribution $P(L_i(v) = l)$ and the one way delay distributions $f_{xxx}(t)$. The distributions of both $L_i(v)$ and $f_{xxx}(t)$ depend on the scheduling algorithm used in the overlay, and it is outside of the scope of this paper to estimate them.

V. Numerical results

In the following we present numerical results obtained using the exact model derived in Section IV, and show that the bounds derived in Section III hold in the presence of various retransmission schemes and FEC.

A. System parameters

We model the propagation delay D_p by one of two distributions. The first distribution is light-tailed, a normal distribution truncated at 180 ms with mean $E[D_p] = 67$ ms and standard deviation $\sigma_{D_p} = 21$ ms extracted from a transit-stub network of 10^4 nodes generated with the GT-ITM topology generator [33]. The second distribution is heavy tailed, a shifted Pareto distribution with CDF $F_M(x) = 1-(1+x/b)^{-z}$, $z = 3$ and $b = 0.134$ ($E[D_p] = 67$ ms, $\sigma_{D_p} = 116$ ms). We consider the streaming of a $B = 400$ kbps data stream, and the capacity of the source node's output link is $C(s) = 100$ Mbps. The outdegree of the source, O_s, is set to 50 throughout the paper for easy comparison and to ensure that the overlay is feasible for all considered values of the number of trees [13], though the particular value of O_s does not affect the validity of our conclusions. The packet size is 1410 bytes. The distribution of the nodes' output capacities (C_v) and outdegrees (O_v) is as in [4] and is shown in Table II. Since the effect of the input capacity of the nodes is small on the results [16], we consider 10 Mbps for all nodes.

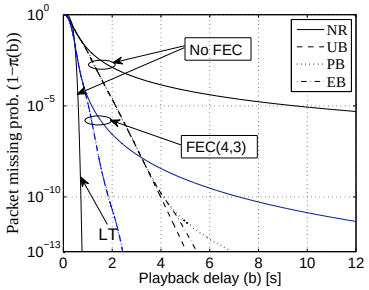

Fig. 5. Packet missing prob. vs. playback delay for light-tailed (LT) and for heavy tailed one-way delays without and with retransmissions, *RP*, *RTD*, $\tau = 4$, $N = 10^4$, $p = 0$.

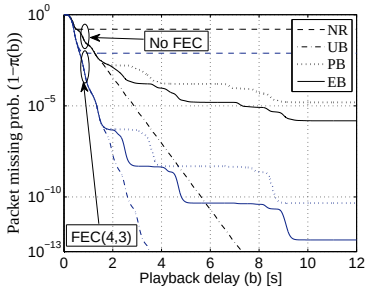

Fig. 6. Packet missing prob. vs. playback delay for heavy-tailed one-way delays and various back-off schemes, *RP*, *RTD*, $\tau = 4$, $N = 10^4$, $p = 0.05$.

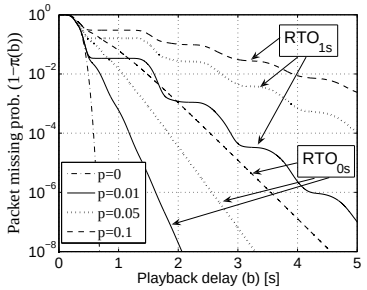

Fig. 7. Packet missing prob. vs. playback delay for light-tailed one-way delays, different packet loss probabilities, and packet loss detection times, $\tau = 4$, $N = 10^4$.

The inter-arrival times of nodes joining the overlay are exponentially distributed, this assumption is supported by several measurement studies, e.g., [34]. The session holding times M follow the log-normal distribution, the mean holding time is $E[M] = 306$ s ($\mu = 4.93$, $\sigma = 1.26$) [34]. The nodes are prioritized according to their outdegrees as proposed in [29], hence large contributors are closer to the source in the trees in which they forward data and reconnect faster to the trees.

B. Approximating the nodes' distance distribution

We approximate the number of nodes in level l in a tree via the recurrence $N_l = \sum_{v \in \mathcal{R}(l-1)} O_v$ with initial condition $N_1 = min(N, O_s)$, where $\mathcal{R}(l-1)$ denotes the set of nodes for which $\mathcal{L}_m(v) = l-1$. E.g., if the outdegrees were homogeneous, then $N_l = O_s \overline{O}_v^{\,l-1}$. A real overlay's structure differs from this approximation due to node churn, but as simulation results show [12], [16], the difference does not have a significant effect on the accuracy of the model.

C. Per-hop delay distribution and retransmissions

For the pdf of the one-way delay (f_{dra}) and the retransmission times (f_{rra} and f_{rrb}) we use the model described and validated in [16]. The model captures the delay distribution on the output links of the nodes as seen by the departing packets, the propagation delays (D_p) and the delay distribution on the input links of the nodes as seen by the arriving packets.

We consider three retransmission timeout calculation methods. The first method (denoted by *RTD*) calculates the retransmission timeout (RTO) dynamically based on the mean and the standard deviation of the one-way delay and the round-trip-time (i.e., $\text{RTO} = E[X_h] + 4\sigma[X_h]$), similar to the algorithm used by most TCP implementations. We consider three back-off schemes in combination with the *RTD* method to calculate the distribution of the time until a retransmission request is actually generated (the distributions of T_{dld} and T_{rld}). For the uniform back-off scheme (UB) the RTO does not increase after successive failed retransmission requests. For the polynomial back-off scheme (PB) after k successive failed retransmission requests the RTO is set to k^2 times its original value. For the exponential back-off scheme (EB) after k successive failed retransmission requests the RTO is set to 2^{k-1} times its original value.

Methods two and three are idealized retransmission timeout calculation methods. We denote by RTO_{0s} the case when the loss of a packet is detected at the instant when it should have arrived (if it had not been lost), i.e., an ideal loss detection algorithm, for which $f_{dld}(\upsilon) = p/(1-p)\, f_{dra}(\upsilon)$. We denote by RTO_{1s} when a retransmission is requested 1 s after the packet or the retransmission request has been sent out. We denote by NR when no retransmissions are used.

D. The case of packet losses

First, we show the effect of the tail of the distribution of the one-way delay on the playback delay distribution and how retransmission schemes can influence it. We change the tail of the one-way delay by choosing the distribution of D_p to be light-tailed or heavy-tailed.

a) Tail asymptotics: Fig 5 shows the packet missing probability as a function of the playback delay for light-tailed and heavy-tailed one-way delays. There are no packet losses between overlay nodes, hence retransmission requests are sent only due to late arriving packets. For heavy tailed one-way delays retransmissions follow the *RP* scheme with the RTD method, for light-tailed one-way delays we show results without retransmissions. The figure confirms the analytical results presented in Section III: for light-tailed one-way delays (LT) the playback delay decreases at least exponentially, for heavy-tailed one-way delays the tail of the playback delay distribution is heavy without retransmissions, even if FEC is used. In the presence of FEC the time until a packet is possessed is the minimum of two random variables: the time until the packet would be received through the corresponding

Ratio	15%	25%	40%	20%
C_v	10 Mbps	1 Mbps	384 kbps	128 kbps
O_v	2.5τ	2τ	0.75τ	0.25τ

TABLE II
DISTRIBUTION OF NODE OUTPUT CAPACITIES AND OUTDEGREES.

NR, RP, RB, RA	*R*etransmissions: *N*one, from *P*arent, from *B*ack-up parent or from *A*nother tree
UB, PB, EB	*B*ack-off scheme: *U*niform, *P*olynomial, *E*xponential
RTO_s	Idealized loss detection schemes

TABLE III
NOTATIONS USED IN THE FIGURES

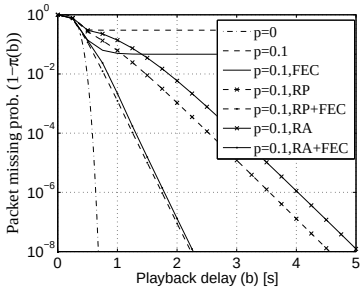

Fig. 8. Packet missing prob. vs. playback delay for different retransmission schemes, $\tau = 4$, $N = 10^4$.

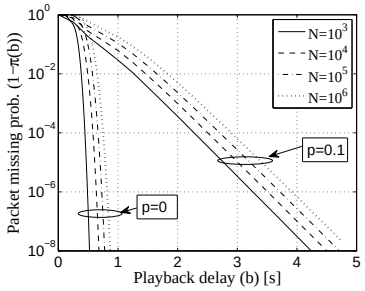

Fig. 9. Packet missing prob. vs. playback delay for light-tailed one-way delays and various overlay sizes.

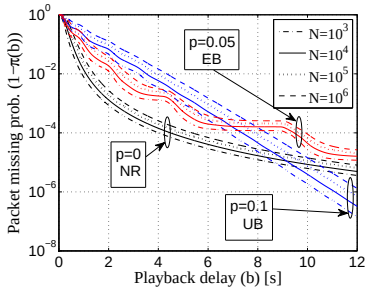

Fig. 10. Packet missing prob. vs. playback delay for heavy-tailed one-way delays and various overlay sizes.

parent and the time to FEC recovery (which is the k^{th} order statistic of $n-1$ random variables with finite m.g.f.). Hence the tail of the distribution of the packet missing probability is determined by the tail of the per-hop-delay distribution. Nevertheless, FEC effectively decreases the packet missing probability for given playback delay, and even though it cannot change the tail behavior, it can make the decay of the packet missing probability exponentially fast for practical purposes.

Our conclusion is similar for the different back-off schemes: the packet missing probability decays almost exponentially fast for all three schemes for packet missing probabilities of practical interest. The tails of the distributions have no practical importance in this case, and the trunks of the distributions are approximately light-tailed down to 10^{-10}. (A packet missing probability of 10^{-13} would lead to 1 packet missing its playout deadline every 158 *years* assuming an HDTV streaming channel at 20Mbps, i.e., approximately 2000pkts/sec). This phenomenon can be explained by that single retransmissions are sufficient to achieve very low packet missing probabilities, hence the effect of the back-off scheme cannot be observed. An interesting question is whether the distributions' tails become important as the overlay's size increases. We will answer this question when discussing Fig. 10.

Fig. 6 shows results for heavy-tailed one-way delays in the presence of packet losses without retransmissions and with retransmissions using the RP scheme, the RTD method and the three back-off schemes. Without retransmissions (NR) the analysis of the asymptotic behavior presented in [12] with respect to N,p and the FEC code applies to $lim_{b\to\infty}\pi(b)$: the packet reception probability converges to the asymptotically stable fixed point of the discrete dynamic system shown in [12], and $lim_{b\to\infty}\pi(b) < 1$. When using retransmissions we see however the effect of the back-off scheme on the tail behavior. The UB scheme leads to an exponential decay of the packet missing probability, while the PB and EB schemes show a slower decrease. The use of FEC does not change the tail behavior, but it can lead to an exponential decrease of the packet missing probability for practical values of interest. For the PB scheme there is a difference between the tail behavior (which is light-tailed) and the behavior for packet missing probabilities of practical interest (which shows a subexponential decrease): the figure suggests that PB leads to a heavier tail than EB, this is however only true for relatively small playback delays, and is explained by $k^2 \geq 2^{k-1}$ for $k \leq 6$. Nevertheless, for small delays a PB scheme might lead to a slower decay of the packet missing probability than an EB scheme.

Next we evaluate the packet missing probability as a function of the playback delay for the case of packet losses and light-tailed one-way delays. Fig. 7 shows results for the RP scheme with the two different idealized methods for RTO calculation. The figure shows results for $N = 10^4$ nodes organized in $\tau = 4$ trees. Despite the choice of a different RTO calculation method than the one used in Fig. 5, the curve for $p = 0$ in Fig. 7 looks almost identical to the curve for the light-tailed one-way delay distribution in Fig. 5: the decrease of the packet missing probability is faster than exponential. This is predicted by Fig. 2 (a), as the rate function for the discrete uniform distribution grows faster than linear. The rate function of the geometric distribution is however close to linear (Fig. 2 (b)), hence we expect that in the presence of losses the decrease of the packet missing probability is not much faster than exponential. This is supported by the curves that show results for $p > 0$. The slope of the curves is related to the slope of the rate function of the per-hop delay distribution, the steeper the rate function, the faster the decrease of the packet missing probability. Though for RTO_{1s} the RTO is big compared to the per-hop delays and hence the packet missing probability decreases almost in a stepwise manner, we still observe the exponential decay. The curves for different loss probabilities and loss detection times show similar properties, they only differ in the slopes of the curves.

Fig. 8 shows results without losses and with losses ($p = 0.1$) for the RP and the RA retransmission schemes for light-tailed one-way delays and RTO_{0s}. The $N = 10^4$ nodes are organized in $\tau = 4$ trees, and FEC(4,3) is used when indicated. We observe that in the presence of losses the exponential decay does not hold when retransmissions are not used just as in the case of heavy-tailed one-way delays. Again, the analysis of the asymptotic behavior presented in [12] with respect to N,p and the FEC code applies to $lim_{b\to\infty}\pi(b)$, i.e., $lim_{b\to\infty}\pi(b) < 1$. Consequently, the assumptions of Theorem 1 are not fulfilled, because all nodes do not receive all data. When using retransmissions, the decay is exponential, as shown by the results for both the RP and the RA retransmission schemes, with and without FEC. FEC decreases the necessary playback delay to achieve a certain packet missing probability, but the exponential decay still holds for the reason explained before.

Consequently, an alternative of increasing the playback delay in order to achieve a certain packet missing probability is to introduce FEC. Nevertheless, the ratio of FEC redundancy has to be adjusted dynamically based on feedback from the nodes. We observe a small difference between the results obtained with the two retransmission schemes. Using the RA scheme, retransmission of a packet in stripe m becomes possible only once nodes that do not forward data in tree m receive the packet. These nodes are in the last level of tree m, and hence, we observe a slow decay of the packet missing probability close to the point where the curves with and without retransmissions separate. Surprisingly, the difference between the results obtained with the RP and the RA schemes is small, especially when FEC is used, in which case the decrease of the decay close to the point where the curves with and without retransmissions separate is significantly smaller as well.

b) Scaling behavior: After the discussion of the behavior of the packet missing probability as a function of the payback delay, we turn to the problem of scaling. Through numerical examples we demonstrate the results of Theorem 2 and 4.

Fig. 9 shows results for the RP retransmission scheme for various overlay sizes, light-tailed one-way delays, RTO_{0s} and $\tau = 4$ trees. The figure supports Theorem 2: the horizontal gap between the curves is constant, that is, both in the presence of losses and in the absence of losses it is enough to increase the playback delay logarithmically in order to maintain the packet missing probability constant. Surprisingly however, the smaller the playback delay needed to achieve a certain packet missing probability for a given overlay size, the more sensitive is the overlay to the increase of the number of nodes. For $p = 0$ the packet missing probability increases by orders of magnitude if the overlay's size increases by a factor of ten, for $p = 0.1$ the increase is significantly smaller. Consequently, even if one could achieve a low packet missing probability with a small playback delay, the playback delay should be over-dimensioned to ensure that the packet missing probability does not become too high if the overlay suddenly grows.

Fig. 10 shows results for heavy-tailed one-way delays, various overlay sizes and loss probabilities using the RP retransmission scheme and the RTD method. For the UB scheme the packet missing probability exhibits a light tail, and consequently we observe the same scaling properties as in Fig. 9 for light-tailed one-way delays. For $p = 0$ and for the EB scheme the packet missing probability exhibits a heavy tail, but we can still observe the same scaling properties, i.e., an increase of the playback delay with a constant is sufficient to compensate for the tenfold increase of the overlay's size. This behavior was predicted by Theorem 4. Furthermore, as predicted by (22), the increase of the packet missing probability at a given playback delay is proportional to the logarithm of the ratio of the number of nodes $log(N_2/N_1)$ as the number of nodes increases from N_1 to N_2 because $E[L_i(v)] \sim O(logN)$. There is however no such asymptotic result if the packet missing probability follows a light-tailed distribution.

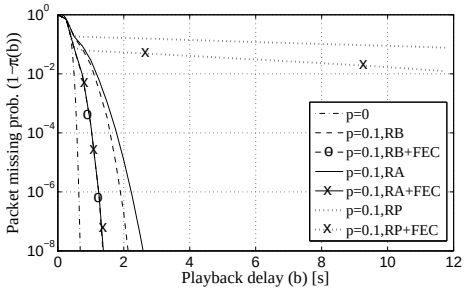

Fig. 11. Packet missing prob. vs. playback delay for $N = 10^4$, node churn and three retransmission schemes.

E. The case of node churn

In the following we show analytical results for the case of node churn. For the reconnection and the disconnection times we use values similar to the measured data presented in [1]: $E[\Xi] = 5$ s and $E[\Omega] = 200$ s in the tree where a node forwards data, and $E[\Xi] = 30$ s and $E[\Omega] = 100$ s in the trees where it does not. (Nodes are disconnected with a higher probability in the trees where they do not forward data.) In lack of a measured distribution we model the reconnection time Ξ with a normal distribution $N(E[\Xi], E[\Xi]/3)$. Based on these values the loss probability experienced by a node in a tree where it forwards data ($p = 0.024$) and where it does not forward data ($p = 0.1968$) can be calculated according to (41).

The distribution of the retransmission times depends on the retransmission scheme used. For the *RP* scheme retransmission occurs once the new predecessor is found, hence f_{rra} is the pdf of the forward recurrence time of a renewal process with inter-renewal time Ξ [35]. For the *RB* and *RA* schemes f_{rra} is as discussed in Section V-C. For all three schemes, we assume that retransmissions are asked from nodes that are present in the overlay, i.e., $\int_0^\infty f_{rra}(t)dt = \int_0^\infty f_{rrb}(t)dt = 1$.

Figure 11 shows results for the case of node churn and the three retransmission schemes. As expected, the RB scheme, which involves significant control overhead, performs best. Surprisingly however, the RA scheme performs nearly as good as the RB scheme, both without and with FEC. This is because under churn nodes experience more frequent losses in the trees in which they do not forward data, i.e., far from the source: when these losses occur, data is already available in large parts of the overlay, hence the additional delay introduced by the RA scheme is small. The RP scheme, due to the large retransmission delays, performs almost as bad as if there were no retransmissions at all. Nevertheless, we observe the exponential decay with a very slow decay rate. The bad performance of the RP scheme suggests that resilience to node churn in a multi-tree-based overlay requires retransmission schemes that abandon the rigid structure of the trees, and converge towards pull-based architectures, e.g., the RA and RB schemes.

VI. Conclusion

In this paper we presented analytical results on the data distribution and scaling behavior of overlay multicast systems in terms of the playback delay and the overlay size. We derived general bounds on the streaming efficiency and the overlay

scalability and gave a detailed model of peer-to-peer streaming systems to interpret the presented bounds.

Our asymptotic results show that the tail behavior of the end-to-end delay distribution, i.e., the evolution of the packet missing probability as a function of the playback delay, is determined by the tail behavior of the per-hop delay distributions. If the per-hop delays are light-tailed, then the packet missing probability shows an asymptotically at least exponential decrease as a function of the playback delay, while it exhibits a heavy-tailed distribution otherwise. The tail behavior of the per-hop delays is dominated by the back-off scheme used for retransmissions. Nevertheless as the detailed results show, back-off schemes that lead to a heavy-tailed per-hop delay distribution can still show an exponential decrease of the packet missing probability in the range of practical interest. Since the decrease of the packet missing probability reflects the characteristics of the per-hop delays, it is not a good measure of the efficiency of the overlay structure or of the stream distribution algorithms.

To assess the structure of the overlay one has to look at the scaling of the playback delay with respect to the overlay size. We showed that in an overlay in which the distance of the peers from the source is a logarithmic function of the overlay's size, the playback delay does not have to be increased faster than the logarithm of the overlay's size to keep the packet missing probability constant.

We presented a detailed model of a push-based overlay, and showed that the asymptotic scaling properties hold using various retransmission schemes and FEC. We concluded that even simple overlay management solutions can provide good scaling properties.

The results presented in the paper provide metrics to assess the scalability of peer-to-peer streaming systems and give a basic understanding of the dependencies between streaming performance, overlay and data transmission control.

REFERENCES

[1] Y-W. Sung, M. Bishop, and S. Rao, "Enabling contribution awareness in an overlay broadcasting system," in *Proc. of ACM SIGCOMM*, 2006, pp. 411–422.

[2] X. Liao, H. Jin, Y. Liu, L.M. Ni, and D. Deng, "Anysee: Scalable live streaming service based on inter-overlay optimization," in *Proc. of IEEE INFOCOM*, April 2006.

[3] N. Magharei and R. Rejaie, "PRIME: Peer-to-peer Receiver drIven MEsh-based streaming," in *Proc. of IEEE INFOCOM*, May 2007.

[4] L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "Randomized decentralized broadcasting algorithms," in *Proc. of IEEE INFOCOM*, 2007.

[5] Thomas Locher, Remo Meier, Stefan Schmid, and Roger Wattenhofer, "Push-to-pull peer-to-peer live streaming," in *Proc. of International Symposium on Distributed Computing*, Sept. 2007.

[6] V. Fodor and Gy. Dán, "Resilience in live peer-to-peer streaming," *IEEE Communications Magazine*, vol. 45, no. 6, pp. 116–123, June 2007.

[7] "PPLive," http://www.pplive.com/, May 2008.

[8] "OctoShape," http://www.octoshape.com/, May 2008.

[9] X. Hei, C. Liang, J. Liang, Y. Liu, and K.W. Ross, "A measurement study of a large-scale P2P IPTV system," *IEEE Trans. Multimedia*, vol. 9, no. 8, pp. 1672–1687, 2007.

[10] T. Small, B. Liang, and B. Li, "Scaling laws and tradeoffs in peer-to-peer live multimedia streaming," in *ACM Multimedia*, October 2006.

[11] Gy. Dán, V. Fodor, and G. Karlsson, "On the stability of end-point-based multimedia streaming," in *Proc. of IFIP Networking*, May 2006, pp. 678–690.

[12] Gy. Dán, V. Fodor, and I. Chatzidrossos, "Streaming performance in multiple-tree-based overlays," in *Proc. of IFIP Networking*, May 2007, pp. 617–627.

[13] Gy. Dán, V. Fodor, and I. Chatzidrossos, "On the performance of multiple-tree-based peer-to-peer live streaming," in *Proc. of IEEE INFOCOM*, May 2007.

[14] R. Kumar, Y. Liu, and K.W. Ross, "Stochastic fluid theory for P2P streaming systems," in *Proc. of IEEE INFOCOM*, May 2007.

[15] Thomas Bonald, Laurent Massoulié, Fabien Mathieu, Diego Perino, and Andrew Twigg, "Epidemic live streaming: optimal performance trade-offs," in *Proc. of ACM SIGMETRICS*, June 2008, pp. 325–336.

[16] Gy. Dán and V. Fodor, "An analytical study of low delay multi-tree-based overlay multicast," in *Proc. of ACM P2P-TV*, Aug 2007.

[17] Yang Yang and Tak-Shing Peter Yum, "Delay distributions of slotted ALOHA and CSMA," *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1846–1857, 2003.

[18] Taka Sakurai and Hai L. Vu, "MAC access delay of IEEE 802.11 DCF," *IEEE Trans. Wireless Commun.*, vol. 6, no. 5, pp. 1702–1710, 2007.

[19] M. Lelarge, Z. Liu, and C.H. Xia, "Asyumptotic tail distribution of end-to-end delay in networks of queues with self-similar cross traffic," in *Proc. of IEEE INFOCOM*, March 2004.

[20] Daniel R. Figueiredo, Benyuan Liu, Vishal Misra, and Don Towsley, "On the autocorrelation structure of TCP traffic," *Computer Networks*, vol. 40, no. 3, pp. 339–361, 2002.

[21] M Denuit, C. Genest, and É. Marceau, "Stochastic bounds of sums of dependent risks," *Insurance: Mathematics and Economics*, vol. 25, no. 1, pp. 85–104, Sept. 1999.

[22] A. Schwartz and A. Weiss, *Large Deviations for Performance Evaluation: Queues, communication and computing*, Chapman & Hall, 1995.

[23] Charles M. Goldie and Claudia Klüppelberg, "Subexponential distributions," *A Practical Guide to Heavy Tails: Statistical Techniques for Analysing Heavy Tails, Birkhauser, Basel*, 1997.

[24] D. B. Cline, "Convolutions of distributions with exponential and subexponential tails," *J. Austral. Math. Soc. Ser. A*, vol. 43, pp. 347–365, 1987.

[25] Viktória Fodor and Ilias Chatzidrossos, "Playback delay in mesh-based peer-to-peer systems with random packet forwarding," in *Proc. of IEEE Future Multimedia Networks*, September 2008.

[26] Fabio Picconi and Laurent Massouli, "Is there a future for mesh-based live video streaming?," in *Proc. of IEEE P2P*, Sept. 2008, pp. 289–298.

[27] V. N. Padmanabhan, H.J. Wang, and P.A Chou, "Resilient peer-to-peer streaming," in *Proc. of IEEE ICNP*, 2003, pp. 16–27.

[28] K. Sripanidkulchai, A. Ganjam, B. Maggs, and H. Zhang, "The feasibility of supporting large-scale live streaming applications with dynamic application end-points," in *Proc. of ACM SIGCOMM*, 2004, pp. 107–120.

[29] M. Bishop, S. Rao, and K. Sripanidkulchai, "Considering priority in overlay multicast protocols under heterogeneous environments," in *Proc. of IEEE INFOCOM*, April 2006.

[30] M. Castro, P. Druschel, A-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh, "SplitStream: High-bandwidth multicast in a cooperative environment," in *Proc. of ACM SOSP*, 2003.

[31] E. Setton, J. Noh, and B. Girod, "Rate-distortion optimized video peer-to-peer multicast streaming," in *Proc. of ACM APPMS*, 2005, pp. 39–48.

[32] Gy. Dán and V. Fodor, "Stability and performance of overlay multicast systems employing forward error correction," *Performance Evaluation*, Under review.

[33] Ellen W. Zegura, Ken Calvert, and S. Bhattacharjee, "How to model an internetwork," in *Proc. of IEEE INFOCOM*, March 1996, pp. 594–602.

[34] E. Veloso, V. Almeida, W. Meira, A. Bestavros, and S. Jin, "A hierarchical characterization of a live streaming media workload," *IEEE/ACM Trans. Networking*, vol. 14, no. 1, pp. 133–146, 2006.

[35] W. Feller, *An Introduction to Probability Theory and its Applications*, John Wiley and Sons, 1966.

György Dán received the M.Sc. degree in computer engineering from the Budapest University of Technology and Economics, Hungary in 1999 and the M.Sc. degree in business administration from the Corvinus University of Budapest, Hungary in 2003. He received his Ph.D. in Telecommunications in 2006 from KTH, Royal Institute of Technology, Stockholm, Sweden. He worked as a consultant in the field of access networks, streaming media and videoconferencing 1999-2001. Currently, he is an assistant professor at KTH, Royal Institute of Technology. His research interests include the design and performance evaluation of distributed and peer-to-peer systems.

Viktória Fodor received her M.Sc. and Ph.D. degrees in computer engineering from Budapest University of Technology and Economics, Hungary, in 1993 and 1999, respectively. In 1999 she joined the Laboratory for Communication Networks at KTH, Royal Institute of Technology, where she now acts as associate professor. Her current research interests include performance analysis and protocol design for multimedia communication and sensor networks.