On the Performance of Multiple-tree-based Peer-to-peer Live Streaming

György Dán, Viktória Fodor and Ilias Chatzidrossos
School of Electrical Engineering
KTH, Royal Institute of Technology, Stockholm, Sweden
Email: {gyuri,vfodor,iliasc}@ee.kth.se

***Abstract*— In this paper we propose and analyze a generalized multiple-tree-based overlay architecture for peer-to-peer live streaming that employs multipath transmission and forward error correction. We give mathematical models to describe the stability properties of the overlay and evaluate the error recovery in the presence of node dynamics and packet losses. We show how the stability of the overlay improves with the proper allocation of the outgoing bandwidths of the peers among the trees without compromising its error correcting capability.**

I. Introduction

The increasing input and output bandwidth of end-hosts in the Internet and the high incidence of flat rate charging gave rise to various peer-to-peer content delivery networks in recent years. Several peer-to-peer live streaming architectures have been proposed and also implemented following the tree based push or the mesh based pull approach (see [1], [2], [3], [4], [5] and references therein). In this paper we focus on push based streaming solutions where the streaming content is forwarded along distribution trees constructed at the beginning of the streaming sessions and maintained as the overlay membership changes. Robustness in these architectures is achieved by applying multiple distribution trees together with some form of multiple description coding based on forward error correction (FEC) [2], [1] or delay limited retransmission [3], and priority schemes [4]. While there are several designs proposed and also implemented the evaluation of these solutions is mostly based on simulations and measurements, giving little insight on general characteristics.

Previously, we proposed mathematical models to describe the behavior of CoopNet [2] like architectures in [6], [7]. Our results showed that the two architectures proposed in the literature, and used as a basis in recent works (e.g., [4]) are straightforward but not optimal. Minimum depth trees minimize the number of affected peers at peer departure, minimize the effect of error propagation from peer to peer, and introduce low transmission delay. Nevertheless, the overlay is unstable and may become disconnected when one of the trees runs out of available capacity after consecutive peer departures. Tree disconnection leads to data loss unless resolved by reconstructing the trees which, however, adds management overhead [2], [1]. Minimum breadth trees are stable and easy to manage, but result in long transmission paths.

[1]This work was in part supported by the Swedish Foundation for Strategic Research through the projects Winternet and AWSI.

In this paper, we evaluate how a generalized architecture performs and suggest parameters for further system design. Specifically, we evaluate the probability of tree disconnection in general multiple-tree-based overlays and investigate how the tree architecture affects the data distribution performance if forward error correction is applied.

The rest of the paper is organized as follows. Section II describes the proposed overlay structure and error correction scheme. We evaluate the feasibility of the overlay and give an analytical model describing the evolution of the available capacity in the overlay in the case of node dynamics in Section III. Section IV discusses the performance of the overlay based on the mathematical models and simulations and we conclude our work in Section V.

II. System description

A. Overlay structure

The overlay we propose in the following is a generalization of the multiple tree based overlays presented in [2], [1]. The aim of the proposed design is to improve the stability of the overlay in the case of node departures while keeping the length of the transmission paths in the overlay low.

The overlay consists of a root node and N peer nodes. The peer nodes are organized in t distribution trees, and in each tree they have a different parent node from which they receive data. We denote the maximum number of children of the root node in each tree by m, and we call it the multiplicity of the root node. We assume that nodes do not contribute more bandwidth towards their children as they use to download from their parents, which means, that each node can have up to t children to which it forwards data. (See Fig. 1.) Minimum depth trees are generated if nodes have children in one tree only and minimum breadth trees are generated if nodes have one child in each tree [2], [1].

In this work we assume that, instead of the two extreme cases, every node can have children in up to d of the t trees, called the fertile trees of a node. A node is sterile in all other trees, that is, it does not have any children. We say that each node has a total of t cogs in its fertile trees and has no cogs in its sterile trees. We distinguish three different policies that can be followed to allocate cogs in the fertile trees. With the *unconstrained* cog allocation (UCA) policy a node can have up to t cogs in any of its fertile trees. With the *reserving* cog allocation (RCA) policy a node can have up to $t-d+1$ cogs

a,

b,

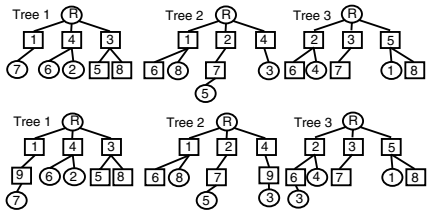

Fig. 1. a) Overlay with $N = 8, t = 3, m = 3$ and $d = 2$, b) the same overlay with $N = 9$. Identification numbers imply the chronological order of arrival, squares indicate that the node is fertile.

in any of its fertile trees (i.e., every node has at least one cog in every fertile tree). With the *balanced* cog allocation (BCA) policy a node can have up to $\lceil t/d \rceil$ cogs in any of its fertile trees. If we denote the maximum number of layers in the trees by L, then in a well maintained tree each node is $1 \leq i < L$ hops away from the root node in its fertile trees, and $L-1 \leq i \leq L$ hops away in its sterile trees.

B. Tree management

The results presented in this paper are not dependent on the particular tree management algorithm used, our focus is on the performance of the overlay rather than the efficiency of the tree building algorithm. Nevertheless, we describe briefly the centralized algorithm used for the simulations. We assume that the overlay is maintained by the streaming server, also the root of the distribution trees.

Before discussing the tree management we define the notion of eligible parent. For a node in its sterile tree, an eligible parent is a node that has at least one free cog and is not parent of the node in any other tree. For a node in its fertile tree an eligible parent is a node that either has a free cog or a sterile child and is not parent of the node in any other tree.

The objective of the tree management is to keep the number of free cogs balanced across the trees and to push the sterile nodes to the lower layers of the trees. To keep the number of free cogs balanced, the fertile trees of a node joining the overlay will be the d trees with the lowest number of free cogs. In the case of equal number of free cogs the trees are selected at random. To build minimum depth trees, the node connects in each tree to one of the eligible parents closest to the root. If a selected parent does not have a free cog but a sterile child, the new node takes the place of the sterile child and the sterile child is connected to the new node. In the case of a node departure, the disconnected children together with their subtrees try to reconnect following the same process. If the tree runs out of available capacity after node departures, some of the children cannot reconnect and the tree becomes disconnected. Disconnected children reattempt to connect to the tree after a reconnection interval T until they manage to reconnect. Fig. 1 shows an overlay for $t = 3$, $m = 3$ and $d = 2$, and shows how the overlay changes when a new node joins.

C. Data transmission and error resilience

The root uses block based FEC, e.g., Reed-Solomon codes [8], so that nodes can recover from packet losses due to network congestion and node departures. To every k packets of information c packets of redundant information are added resulting in a block length of $n = k + c$. If the root would like to increase the ratio of redundancy while maintaining its bitrate unchanged, then it has to decrease its source rate. We denote this FEC scheme by FEC(n,k). Using this FEC scheme one can implement UXP, PET, or the MDC scheme considered in [2]. Lost packets can be reconstructed as long as no more than c packets are lost out of n packets. The root splits the data stream into t stripes, with every t^{th} packet belonging to the same stripe, and it sends every t^{th} packet to its children in a given tree. Peer nodes relay the packets upon reception to their respective child nodes. Once a node receives at least k packets of a block of n packets it recovers the remaining c packets. If a packet belonging to a fertile tree is recovered, then it is sent to the respective children.

III. Feasibility and overlay stability

A. Evaluation of the overlay feasibility

We call an overlay feasible for given m, t and N if it can be constructed. A necessary and a sufficient condition for the overlay to be feasible was shown in [1]. Those conditions were based on the number of cogs that nodes want to use and are willing to offer. The condition shown here relates the parameters of the overlay to each other.

Proposition 1: If the overlay following the BCA policy is feasible for arbitrary N, then $m \geq \lceil t/d \rceil - 1$.

Proof: To show that this condition is necessary, we show that if the condition is not satisfied then there exists N for which the number of cogs in a particular tree can be less than N, in which case some nodes cannot connect to the tree. The total number of offered cogs in the t trees of the overlay is $mt + Nt$. Hence there has to be at least one tree where the number of cogs is $m + \lfloor Nd/t \rfloor t/d$. The offered cogs have to be enough for all N nodes of the overlay in that tree, so that

$$N \leq m + \lfloor \frac{N}{t/d} \rfloor \frac{t}{d}. \tag{1}$$

We can rearrange the inequality to get

$$m \geq N - \lfloor \frac{N}{t/d} \rfloor \frac{t}{d}. \tag{2}$$

Since $N - \lceil t/d \rceil + 1 \leq \lfloor \frac{N}{t/d} \rfloor \frac{t}{d} \leq N$, the right hand side of (2) is bounded from above by $N - (N - \lceil t/d \rceil + 1)$. ■

The condition is only sufficient for feasibility for well maintained trees. In the presence of node departures the overlay cannot always be kept well maintained and hence in general the condition is not sufficient.

B. Evolution of the available capacity

We define the available capacity in the overlay as the sum of the unused offered cogs of fertile nodes minus the number of disconnected nodes, hence it can be negative. For example, if there are f fertile nodes that have no children then the available capacity is ft. We use induction to calculate the available capacity in the overlay. Initially, the available capacity in the

overlay is mt, since the root node can support m nodes in each tree. Upon the arrival of a node the available capacity remains mt, since the node consumes one available cog in each of the t trees and adds a total of t available cogs in its d fertile trees. Similarly, a departure does not change the available capacity in the overlay. Since the available capacity in the overlay is mt, the available capacity per tree is m on average. In the following we investigate the evolution of the available capacity for the BCA policy. For simplicity we assume that t/d is an integer.

Upon departure of a node the available capacity decreases by $t/d-1$ in the departing node's d fertile trees and increases by one in its $t-d$ sterile trees. The available capacity in some of the departing node's fertile trees can decrease below zero, in which case the tree becomes disconnected. In the following we show how the probability of disconnection depends on the parameters t, m and d of the overlay.

We consider the stationary state of the system, when the arrival and departure rates are equal. We assume that the interarrival times of nodes are exponentially distributed, this assumption is supported by measurement studies [9]. We approximate the distribution of the session holding times by an exponential distribution. The distribution of the session holding times was shown to fit the log-normal distribution [9], however, using the exponential distribution makes modeling easier. For the simulations we use the log-normal distribution, and as we will see, the model gives a good match with the results of the simulations. For a given arrival intensity λ, the mean number of nodes in the overlay is $\overline{N}=\lambda/\mu$, where $1/\mu$ is the mean session holding time.

To model the evolution of the available capacity, we use a two-dimensional Markov process with state (ν,ι), corresponding to the number of nodes in the overlay and the available capacity in an arbitrary tree respectively. The state space of the process is $\{N_l \dots N_u\} \times \{c_l \dots c_u\}$. The parameters N_l and N_u are the lower and the upper bounds on the number of nodes in the overlay that the model considers. Similarly, c_l and c_u are the lower and the upper bounds on the available capacity that the model considers. We set $N_l = 0.9\overline{N}$, $N_u = 1.1\overline{N}$, $c_l = -(m-1)t$ and $c_u = mt$, so that the model is computationally feasible but the probability of $\nu \notin \{N_l \dots N_u\}$ and $\iota \notin \{c_l \dots c_u\}$ is negligible. The model is approximate, since the available capacity in an arbitrary tree is not independent of the available capacity in the other trees (since their sum is constant). A model that considers the evolution of all trees would be $t+1$ dimensional, and hence computationally not feasible. Another approximation is the use of a limited state space.

We denote by $q_{i,j}^{k,l}$ the transition intensity from state (i,j) to state (k,l) and $a(j)$ is the probability that an arriving node is assigned to be fertile in the chosen tree given that the available capacity is j in that tree. The transition intensities are then given as ($N_l \le i \le N_u$ and $c_l \le j \le c_u$)

$$\begin{aligned} q_{i,j}^{max(i-1,N_l),min(j+1,c_u)} &= (t-d)i\mu/t \\ q_{i,j}^{max(i-1,N_l),max(j-t/d+1,c_l)} &= di\mu/t \\ q_{i,j}^{min(i+1,N_u),max(j-1,c_l)} &= (1-a(j))\lambda \end{aligned}$$

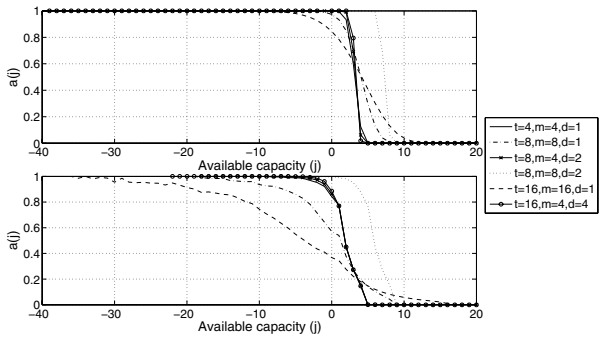

Fig. 2. $a(j)$ vs. available capacity. Analytical model (upper) and simulation results (lower) with the BCA policy.

$$q_{i,j}^{min(i+1,N_u),min(j+t/d-1,c_u)} = a(j)\lambda$$

For the distribution of $a(j)$ we use

$$a(j) = \sum_{u=1}^{d} \left(\begin{array}{c} t-1 \\ u-1 \end{array} \right) F(\frac{j-m}{t/(2d)-1})^{(u-1)} (1-F(\frac{j-m}{t/(2d)-1}))^{t-u}, \quad (3)$$

where $F()$ is the standard normal distribution function. The rationale behind the distribution of $a(j)$ is the following. An arriving node is chosen to be fertile in the d trees with least available capacity among all t trees. We assume independence of the available capacity in the trees and model their distribution by a normal random variable with mean m and standard deviation $t/(2d)-1$. A node is then assigned to be fertile in a tree with available capacity j if there are at least $t-d$ trees with available capacity higher than j.

Fig. 2 shows $a(j)$ from (3) and obtained via simulations. For $j<0$ the model assumes $a(j)$ to be higher, while for j close to m to be lower than it is according to the simulations. The probability of $j<0$ is however small, and hence (3) is a pessimistic estimate. It will be subject of future work to derive a more precise distribution for $a(j)$.

We can calculate the steady state distribution $\psi(i,j)$ of the Markov process using the transition intensity matrix $Q = (q_{i,j}^{k,l})$. The reconnection failure probability is then the ratio of the failed reconnection attempts and the total number of reconnection attempts per time unit,

$$p_f = \frac{1}{\lambda} \sum_{N_l \le i \le N_u} \sum_{c_l \le j < t/d} \psi(i,j) \frac{i\mu}{(t/d)} (\min(t/d, t/d-j)-1). \quad (4)$$

From the model we see that it is the ratio t/d and its relation to m that determine p_f: increasing t/d increases, increasing m decreases it.

IV. Performance evaluation

In the following we analyze the behavior of the overlay using the analytical models presented in the previous sections and via simulations. For the simulations we developed a packet level event-driven simulator. We assume that the session holding times follow a log-normal distribution with mean $1/\mu = 306s$, as shown in [9]. We use the arrival rate λ to

change the mean number of nodes in the overlay $\overline{N}$. We used the GT-ITM [10] topology generator to generate a transit-stub model with 10000 nodes and average node degree 6.2. We placed each node of the overlay at random at one of the 10000 nodes and used the one way delays given by the generator between the nodes. The delay between overlay nodes residing on the same node of the topology was set to 1 ms. Losses on the paths between the nodes of the overlay occur independent of each other with probability p. The reconnection interval T is 1 s unless otherwise stated.

We consider the streaming of a 112.8 kbps data stream to nodes with link capacity 128 kbps. The packet size is 1410 bytes. Nodes have a playout buffer capable of holding 140 packets, which corresponds to 14 s delay with the given parameters. Each node has an output buffer of 80 packets to absorb the bursts of outgoing packets in its fertile tree. To obtain the results for a given overlay size $\overline{N}$, we start the simulation with $\overline{N}$ nodes in its steady state as described in [11]. We set $\lambda = \overline{N}\mu$ and let nodes join and leave the overlay for 5000 s. The measurements are made after this warm-up period for 1000 s and the presented results are the averages of 10 simulation runs. The results have less than 5 percent margin of error at a 95 percent level of confidence.

A. Tree management

Fig. 3 shows the reconnection failure probability as a function of the number of trees for $\overline{N} = 10000$ and $m = t$. The model overestimates p_f, partially due to the pessimistic choice of $a(j)$. The results obtained with the model show however similar tendencies as the simulations. The reconnection failure probability increases as t increases. Increasing d for a given t and m decreases the reconnection failure probability significantly ($d = 1, m = t$ vs. $d = 2, m = t, BCA$). The decrease is orders of magnitude bigger when using the UCA policy according to the simulations. The reconnection failure decreases sharply as m/t increases ($d = 1, m = t$ vs. $d = 1, m = 2t$), and $d = 1, m = 2t$ gives the same p_f as $m = t, d = 2$ using the BCA policy. The reconnection failure does not change if t/d and m are kept constant using the BCA policy (e.g.,t=4 on $d = 1, m = t$ vs. t=8 on $d = 2, m = t/d$ vs. t=16 on $d = 4, m = t/d$). Hence, one can increase t and n without increasing p_f by keeping t/d constant. The UCA policy decreases p_f significantly compared to the BCA policy for the same t/d ratio and value of m.

Fig. 4 shows the reconnection failure probability as a function of the mean number of nodes and the reconnection interval for the three cog allocation policies. The figure shows that increasing the number of nodes and increasing the reconnection interval slightly decrease p_f for all policies. The reason for this phenomenon is that the longer a node waits between reconnection attempts or the higher the arrival intensity, the higher the probability that a fertile node arrives to the disconnected tree during the reconnection interval. A higher value of T means of course that nodes have to wait longer between reconnection attempts and hence loose more packets, so that there should be an optimal value of T for given m, t, d and FEC parameters. Among the three cog allocation policies the UCA policy performs best in terms of reconnection failure probability, with the RCA policy performing nearly as good. The results with BCA coincide with the results with $d = 1$ for the same t/d and m values.

B. Data distribution

We evaluate the performance of the data distribution in the overlay by considering the probability π that an arbitrary node possesses (i.e., receives or can reconstruct) an arbitrary packet as function of p, the probability that a packet is lost between two adjacent nodes. We have extended the mathematical model of minimum depth trees presented in [7] for the BCA case. While the model considers homogeneous, independent losses between the adjacent peers, it can be modified to deal with heterogeneous and correlated losses and also node departures as shown in [6]. As it was shown in [6], for every (n,k) there is a loss probability p_{max} above which π approaches 0 as the number of layers increases. We refer to the data distribution as stable if $p < p_{max}$ and call it unstable otherwise.

Fig. 5 shows π as a function of p obtained with the model for $m = 4$ and $n = t$. The vertical bars show the values $\pi(1)$ at the upper end and $\pi(L-1)$ at the lower end. We included them for $d = 1$ only to ease readability, but they show the same properties for other values of d as well. The figure shows that π remains high and is practically unaffected by N and d as long as $p < p_{max}$. It drops however as the packet loss probability crosses the threshold p_{max}. The drop of the packet possession probability gets worse as the number of nodes and hence the number of layers in the overlay increases. At the same time, for $p > p_{max}$, the difference between $\pi(1)$ and $\pi(L-1)$ (the packet possession probability of nodes that are fertile in the first and the penultimate layers, respectively) increases. Furthermore, increasing t (and hence n) increases π in a stable system, but the drop of the packet possession probability gets faster in the unstable state due to the longer FEC codes.

To validate our model we first present simulation results for the BCA policy. After the 5000 s warm-up period with node arrivals and departures we send 1000 packets through the overlay and measure the ratio of the possessed packets per block. Fig. 6 shows π as a function of p for the same scenarios as Fig. 5. Fig. 6 shows a good match of the simulation results with the analytical results.

To see how the cog allocation policy influences the results, we show π as a function of p in Fig. 7 for the same scenarios as in Fig. 6 but for the UCA policy. Comparing the figures we see that π is the same for $p < p_{max}$, but is higher in the unstable state of the overlay. The better performance of the UCA policy is due to that the overlay has less layers than using the BCA policy. With the UCA policy nodes tend to have more children in the fertile tree where they are closest to the root, so that the tree structure is similar to the $d = 1$ case and the number of layers is lower than using BCA.

Hence, the data distribution performance of an overlay with t trees and $d = 1$ can be closely resembled with an overlay with $d > 1$ and td trees by employing a mechanism that promotes

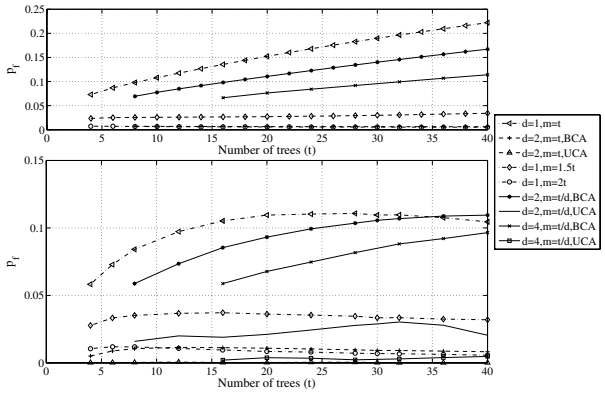

Fig. 3. p_f vs. t for $\overline{N} = 10000$. Analytical results (upper diagram, no curves for UCA) and simulations (lower diagram).

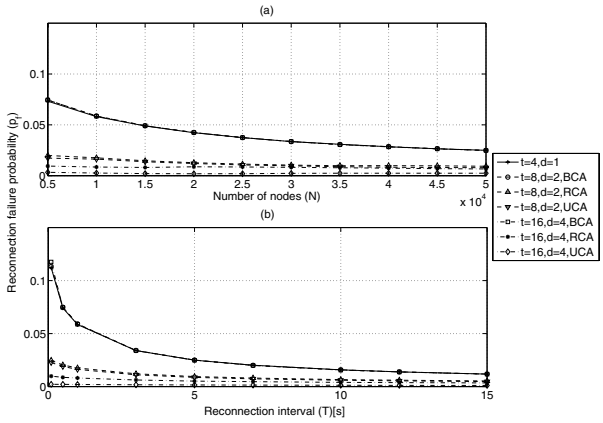

Fig. 4. (a) p_f vs. $\overline{N}$ for $T = 1$ and $m = 4$ (b) p_f vs. T for $\overline{N} = 10000$ and $m = 4$. Simulation results.

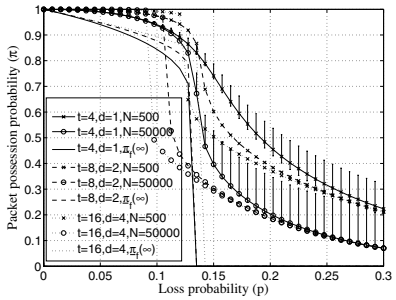

Fig. 5. π vs. p for $m = 4$, $n = t$, $k/n = 0.75$. BCA policy.

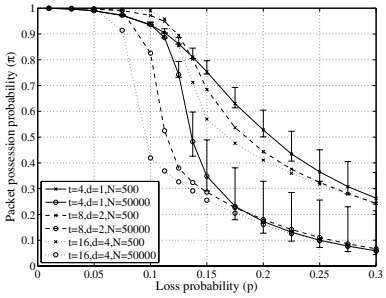

Fig. 6. π vs. p for $m = 4$, $n = t$, $k/n = 0.75$. BCA policy, simulation results.

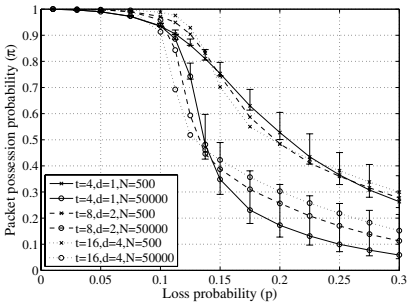

Fig. 7. π vs. p for $m = 4$, $n = t$, $k/n = 0.75$. UCA policy, simulation results.

parents close to the root, such as the UCA policy. Doing so allows the use of longer FEC codes and at the same time one can decrease the reconnection failure probability.

V. Conclusion

In this paper, we proposed and analyzed a peer-to-peer live streaming solution based on multiple distribution trees and FEC. We proposed the free allocation of the outgoing bandwidth of the peers across several trees. The aim of this design is to avoid tree disconnections after node departures, which can happen with high probability in overlays where all the peers can forward data in one tree only.

We analyzed the balanced (BCA), the reserving (RCA) and the unconstrained cog allocation (UCA) policies. Based on analytic models and simulations we concluded that increasing the number of fertile trees increases the overlay stability, with the UCA policy giving the highest increase. Furthermore, the number of trees of the overlay can be increased without worsening the overlay's stability if the ratio of the number of trees and the number of fertile trees is kept constant.

We showed that the number of fertile trees and the allocation policy does not influence the performance of the data transmission in the stable region of the overlay, and with the UCA policy the data distribution performance can be close to that of the minimum depth trees also in the unstable region. Our results indicate that adjusting the number of fertile trees can be a means to improve the overlay stability without decreasing the performance of the data transmission.

References

[1] 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.

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

[3] 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.

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

[5] 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.

[6] 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.

[7] Gy. Dán, I. Chatzidrossos, V. Fodor, and G. Karlsson, "On the performance of error-resilient end-point-based multicast streaming," in *Proc. of IWQoS*, June 2006, pp. 160–168.

[8] I.S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *SIAM J. Appl. Math.*, vol. 8, no. 2, pp. 300–304, 1960.

[9] E. Veloso, V. Almeida, W. Meira, A. Bestavros, and S. Jin, "A hierarchical characterization of a live streaming media workload," in *Proc. of ACM IMC*, 2002, pp. 117–130.

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

[11] J-Y. Le Boudec and M. Vojnovic, "Perfect simulation and stationarity of a class of mobility models," in *Proc. of IEEE Infocom*, March 2004.