A Preferential Attachment Model for Tree Construction in P2P Video Streaming

Marcio N. Miranda[1], **Daniel R. Figueiredo**[2]

[1] Universidade Federal de Alagoas (UFAL) – Instituto de Computação
BR 104 Norte Km 97 – 52072-970 – Maceió, AL – Brazil

[2]Universidade Federal do Rio de Janeiro (UFRJ)
COPPE/Programa de Engenharia de Sistemas e Computação
Caixa Postal 68511 – 21941-972 – Rio de Janeiro, RJ – Brazil

marcionmiranda@gmail.com, daniel@land.ufrj.br

Abstract. *Tree-based peer-to-peer (P2P) video streaming systems rely on a video dissemination tree to deliver video to peers. In order to offer good quality of service, two fundamental aspects should guide the construction of the video dissemination tree: low node degree and short distances to the server. In this paper, we propose a very simple growth process to construct the video dissemination tree. Our generative model is based on the preferential attachment principle, where preference is given in terms of node quality. The proposed model has a single parameter to weigh the relative importance of node degree and node distance on assessing node quality. We investigate our model through simulations and find that surprisingly good video dissemination trees are usually generated. In particular, topological properties of generated trees are never extreme and average tree quality is mostly comparable (and sometimes superior) to carefully designed dissemination trees. Our results indicate that the proposed model is capable of self-organizing nodes into good trees under various assessments of node quality.*

1. Introduction

Large-scale video streaming in the Internet is a problem that has received much attention for over a decade now. Most of the difficulties arise from the large amount of resources (e.g., network bandwidth) required by the application when serving hundreds or thousands of users. Such a high demand for resources requires a system architecture beyond centralized solutions in order to scale more efficiently.

During the past few years, proposals for large-scale video streaming applications have started to adopt the peer-to-peer (P2P) paradigm [Liu et al. 2008b, Liu et al. 2008a]. In the P2P paradigm, users of the application (known as peers) not only receive service, but also provide service to other users, which allows the system to scale more efficiently with the number of users.

An approach to apply the P2P paradigm to video streaming is to construct a *video distribution tree*, where the server (or servers) correspond to the root of the tree and the peers (users) to its internal nodes [hua Chu et al. 2002, Banerjee et al. 2004]. The server is responsible for generating (in case of live streaming) or storing (in case of pre-stored streaming) the video. As users arrive to the system, they can receive the video stream either from the server or from another peer already in the tree. Thus, peers receiving the video stream also offer service, that is, forward the video stream to other peers.

A major challenge within this system is the construction of the video distribution tree, as different trees will offer different quality of service to users. The main difficulty is that the server nor any other peer has a global view of the quality of the distribution tree. Even if all peers that would form the distribution tree were known in advance, arranging them in the *best* tree is not a trivial task, as information about connectivity between the peers (bandwidth, delay, etc.) is usually not readily available. Thus, most systems rely on probing, side information and randomization to construct the video distribution tree [Liu et al. 2008b, Liu et al. 2008a].

At any rate, two issues are fundamentally important when considering the quality of a video distribution tree:

- **Node degree.** The degree of a node in the tree (i.e., number of children) corresponds to the number of video streams being served by this node. Since nodes have finite resources (e.g., network bandwidth, memory, etc) and each video stream served consumes resources, node degree directly affects the quality of the video served by a node.
- **Node distance.** The node distance is the number of hops between the node and the root of the tree (equivalently, it is the level of the node on the tree). Since video is forwarded from node to node down the tree, node distance directly affects the quality of the video received, as the video stream is likely to experience larger delays and losses.

Thus, mechanisms to construct efficient video distribution trees usually consider these two characteristics. However, an important issue to assess the quality of the video distribution tree is the relative impact of these two characteristics. For example, if bandwidth is widely abundant, then node distance impacts video quality relatively more than node degree.

In this paper, we are interested in understanding the quality and topological properties of video distribution trees when a simple mechanism is used to construct the tree. In particular, we model the construction of the distribution tree using a simple probabilistic growth process that consider node degree and node distance. Our generative model is based on the idea of "preferential attachment" [Albert and Barabási 2002], where preference is given to nodes with higher *utility*, which is a measure for the quality of the video served by a given node in the tree. Our model has a single parameter that captures the relative importance of node degree and node distance when assessing the video quality.

Note that our goal is not to model any specific protocol, but to understand and characterize video distribution trees constructed through a simple, self-organizing mechanism based on the preferential attachment principle. We abstract all system-level details, such as bandwidth capacity and peer location, and consider just the fundamental aspects that determine video quality.

We evaluate the proposed model numerically through simulations and report the topological properties and quality of the trees constructed. Our findings indicate that the model generates relatively good quality trees when compared to trees that are carefully organized (e.g., complete k-ary trees). Moreover, we find that the topological properties among the trees are not extremely different even when comparing opposite ends of the balance between node degree and node distance. Intuitively, the probabilistic approach captured by the model avoids trees with extreme topological structures, such as a star.

The remainder of this paper is organized as follows. In Section 2 we present the related work and some discussion. Section 3 describes the proposed model in detail and the metrics to be investigated. Section 4 compares the quality of the trees generated by the proposed model with the quality of other interesting trees. Section 5 presents the numerical evaluation of the model. Finally, Section 6 concludes the paper with our final remarks.

2. Related work

Within the past few years, several solutions based on the peer-to-peer (P2P) paradigm have emerged to address large-scale video streaming. Such solutions can be classified roughly into tree-based systems or mesh-based systems [Liu et al. 2008b, Liu et al. 2008a]. In tree-based systems, peers are organized into a tree structure and information (i.e., video stream) flows down one or more distribution trees [hua Chu et al. 2002, Banerjee et al. 2004]. In mesh-based systems, there is no particular structure and peers exchange information directly with one another, dynamically changing their neighbors over time, similar to an epidemic dissemination [Li et al. 2007, Hei et al. 2007].

There has been recent work on mathematically modeling P2P streaming video systems. Small et. al presents a tradeoff study of system-level parameters in the scaling of the system [Small et al. 2006]. Kumar et. al propose a fluid-based modeling framework to evaluate the performance of general P2P streaming systems as a function of system-level parameters [Kumar et al. 2007]. Bonald et. al propose a model for epidemic-style dissemination in order to evaluate different dissemination strategies [Bonald et al. 2008]. The work of Carra et. al presents a graph-based model to capture the dynamics of video distribution tree [Carra et al. 2007]. This last work is the closest to the scenario we investigate in this paper, however, there are fundamental differences. We are concerned with the understanding properties of a distribution tree constructed through a very simple mechanism based on preferential attachment, where all system-level parameters are abstracted. For example, differently from prior work, our model has no notion of bandwidth capacity or any explicit limits on the maximum number of children a node can have. These constraints are inherently captured by the self-organizing nature of the proposed model, as we will soon discuss.

The principle of preferential attachment has been used to model the growth process of several different networks [Albert and Barabási 2002]. Barabasi and Albert have applied this principle to model networks that exhibit a power-law scaling behavior in their degree distribution [Barabási and Albert 1999]. In their model, preference is proportional to node degree, such that nodes with higher degree are mode likely to receive edges from arriving nodes. They show that this simple growth process leads to graphs with a power law degree distribution.

Motivated by other kinds of directed networks, Sevim and Rikvold have investigated a model where preference is *inversely* proportional to node out degree [Sevim and Rikvold 2006]. In their work, arriving nodes prefer to connect to nodes with lower out degree. They show that this model does not lead graphs with a power law out degree distribution, but rather to exponentially decreasing tail probabilities. Although similar, the model we investigate in this paper is inherently different, as preference is inversely proportional to both node degree and node distance. However, as we will soon describe, our proposed model can degenerate to the model investigated by Sevim and Rikvold (when the parameter $\alpha = 1$).

3. A tree construction model

We consider a video streaming system composed of a single server and some large number of homogeneous peers. Peers arrive to the system sequentially and connect to a single node in the video distribution tree to start receiving service (i.e., the video stream). Peers in the tree offer service to an arriving peer by simply forwarding to the arriving peer the video stream. We assume that peers always forward the video if they are chosen to be the parent of an arriving node (i.e., all peers are altruistic). Finally, we also assume that peers never leave the system nor move in the distribution tree, thus, their position in the distribution tree is determined at the time of their arrival. Figure 1 illustrates the construction of a video distribution tree.

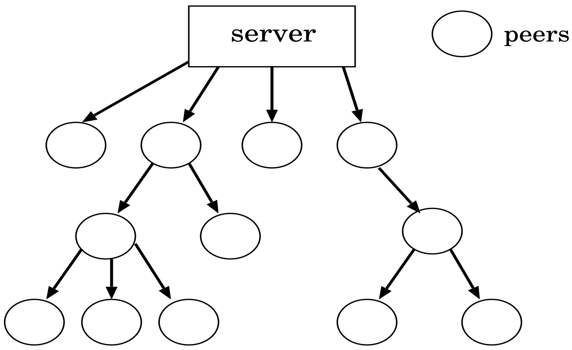


Figure 1. The video distribution tree

A tree construction mechanism determines where in the current distribution tree an arriving peer should connect to. The mechanism basically determines the parent node of an arriving peer. We assume that such mechanism is inherently an *online algorithm*, as it has no knowledge of the number nodes that will join the distribution tree nor can the mechanism rearrange the distribution tree, shuffling nodes around as they join the tree. This assumption is rather realistic when considering distributed, large-scale video P2P streaming systems.

We consider a very simple model for the tree construction mechanism. In particular, the mechanism takes into consideration the quality of the video stream that the arriving peer will receive when attaching itself to a peer already in the distribution tree (or from the server). Two properties are fundamentally important in determining the quality of a node in the tree[1]: node degree and node distance.

Intuitively, node quality is inversely proportional to its degree, as the finite resources of the node must be shared among all its children. Moreover, node quality is also inversely proportional to its distance to the server, as network characteristics that negatively impact video quality (e.g., delay and losses) are proportional to distances. Thus,

[1]We use the expression "node quality" to refer to the quality of service (i.e., video stream) provided by a node to other nodes.

node quality is determined by a combination of these two properties, and quality degrades as any of the two increases. This motivates the use of the following utility function to assess node quality:

$$u(d,l) = \frac{1}{\alpha(d+1) + (1-\alpha)(l+1)} \quad \text{for } 0 \leq \alpha \leq 1 \tag{1}$$

where d and l correspond to the (out) degree of the node and its distance in the tree to the server (measured in hops), respectively. The parameter α is used to weigh the relative importance of the two properties. Note that when $\alpha = 0$, then node quality depends only on distances. This would represent a system where (server or peer) bandwidth is rather unlimited (i.e., nodes have virtually infinite bandwidth). In the other extreme, when $\alpha = 1$, node quality depends only on degree. This would represent a system that has severe bandwidth limitations. Intuitively, α is a parameter that determines the kind of system being considered. Of course, α will have fundamental influence when assessing node and tree quality.

Equation 1 will be used to determine the parent node of an arriving peer, i.e., the node from which the arriving node will receive service. In particular, we consider a probabilistic approach, using the idea of "preferential attachment". Thus, an arriving peer randomly connects to the distribution tree using a probability that is proportional to the utility it will receive from its parent node. Let p_v denote the probability that an arriving node chooses as parent node v already in the tree, where v can also be the server. We have that:

$$p_v = \frac{u(d_v, l_v)}{\sum_{s \in S} u(d_s, l_s)} \tag{2}$$

where d_v and l_v correspond to the degree and the distance of node v, and S is the set of peers already in the distribution tree, including the server, at the time a new node arrives. Note that p_v varies with the number of nodes in the tree.

The above mechanism is suppose to model algorithms that perform an informed guess about where an arriving node should be placed in the distribution tree. Intuitively, the underlying algorithm does not require precise information about node quality, but rather just some idea of node quality. This is captured by the randomness in the above mechanism. Note that this is significantly different from finding and connecting an arriving node to an optimal parent in tree (i.e., a node that would yield the highest utility). Such algorithm would require much more precise information about the distribution tree.

It should be noted that our model do not takes into consideration any explicit restriction on node degree. This implicitly implies that nodes do not have any explicit bandwidth constraints. However, the model can be easily adapted to cope with an explicit limit on node degree in order to represent the constraints on node bandwidth. For example, new nodes could can be restricted to attaching themselves to nodes that have a small enough degree.

3.1. Metrics

In order to characterize the topological properties of the trees constructed by the proposed model we will use traditional graph theoretical metrics, such as maximum and average distance, maximum and average node degree, node degree distribution.

To assess the quality of the tree constructed by the mechanism we consider a metric to capture the average quality of a node in the tree. Let q_v denote the quality of the video received by node v. Thus, similarly to Equation 1, we have:

$$q_v = \frac{1}{\alpha d_{p(v)} + (1-\alpha) l_v} \quad \text{for } 0 \leq \alpha \leq 1 \tag{3}$$

where $p(v)$ denotes the parent node of node v. Note that the quality of the video received by a node depends not on the node's degree, but on the degree of its parent node. Moreover, α is identical in both equations, as quality must be assessed using the same relative importance between node degree and node distance.

Using Equation 3, we define the tree quality as the average node quality of a given tree as follows:

$$\bar{q} = \frac{\sum_{s \in S} q_s}{|S|} \tag{4}$$

where S is the set of nodes in the tree and $|S|$ the cardinality of this set.

One should note that the metric used to determine node quality (Equation 3) depends only on the degree of its parent node and on its distance to the server. However, the quality of a node could depend on other topological properties of the tree. For example, the quality could depend on the degree of all nodes on the path to the server (e.g., the average degree, as opposed to the parent degree).

4. Comparison trees

In order to compare the quality of the trees generated by the proposed model, we consider distribution trees constructed by an *offline algorithm*, thus, by an algorithm that has knowledge of the total number of nodes joining the tree. This algorithm outputs a final distribution tree indicating how nodes should be organized.

We consider two different offline tree construction algorithms. One considers the ensemble of complete k-ary trees, while the other attempts to construct the "optimal" distribution tree. Both are described in the following sections.

4.1. Complete k-ary trees

Recall that a complete k-ary tree is a tree where all non-leaf nodes have exactly k children. Intuitively, complete k-ary trees should yield a good tree quality, under the metric defined above, as they can tradeoff distances and degree by varying k. In particular, note that if $k = 1$ then we have line tree, and if $k = |S|$ (where $|S|$ is the number of nodes in the tree) we have a star tree.

Consider a complete k-ary tree with $n = |S|$ nodes, for some value k. Note that since n is fixed, it is possible that for a given k the tree is not complete to its last level. Let q_k^c denote the average node quality of this tree, as given by Equation 4. Now let q^c denote the best average node quality over all possible k-ary trees. Thus, we have:

$$q^c = \max_{1 \leq k \leq |S|} q_k^c \tag{5}$$

Note that q_k^c depends on α, as α determines the relative importance between node degree and node distance. Figure 2 shows the average tree quality of complete k-ary trees with 60000 nodes for all values of k. Each curve corresponds to an α value. Note that when α is zero, the best average tree value is obtained when $k = 60000$, as this leads to the star tree and all distances are 1. When α is one, the best average tree value is obtained when $k = 1$, as this leads to a line tree where all node degrees are 2. With $\alpha = 0.5$ we observe that the best average tree is obtained when $k = 4$, a relative small value.

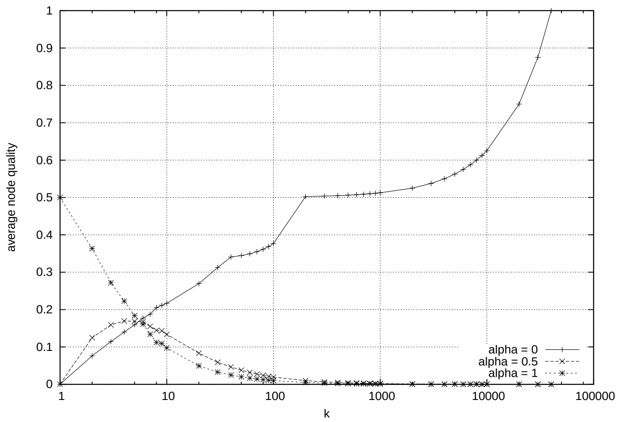


Figure 2. Average tree quality for complete k-ary trees with different α values

4.2. "Optimal" distribution tree

The complete k-ary tree has the drawback that all nodes in the tree must have the same degree. This is clearly not optimal, as nodes at larger distances (further down the tree) will experience lower quality. However, this reduction on quality could be compensated by smaller degrees. This is exactly the intuition behind the "optimal" distribution tree.

Consider a system with $n = |S|$ nodes. In the "optimal" distribution tree, we assume all nodes should have the same quality. Thus, maximizing node quality would maximize tree quality. Since all nodes have identical quality, all nodes in a given level of the tree must parents with the same degree. Thus, all nodes in a given level of the tree have a certain degree. Let d_i denote the degree of nodes in level i, where $i = 0, 1, \ldots, n$. Note that d_0 is the degree of the server. Thus, as given by Equation (3), the quality of a node in level i is given by:

$$q_i^o = \frac{1}{\alpha d_{i-1} + (1-\alpha)i} \quad \text{for } 0 \leq \alpha \leq 1 \tag{6}$$

Since we assume q_i^o is identical for all i, we can solve for d_i, and thus:

$$d_i = d_0 - \frac{i(1-\alpha)}{\alpha} \quad \text{for } 0 < \alpha < 1 \tag{7}$$

where d_0 is the degree of the server and necessarily greater than 0. Note that d_0 determines the degree sequence of the tree, as given by Equation (7). However, the tree sequence $d_0, d_1, \ldots, d_n$ must be such that it can form a tree that can hold n nodes. Let m_i denote the number of nodes in level i, with $i = 1, 2, \ldots, n$. Note that the tree has at most n levels (the case of a line tree). We have that:

$$m_i = \prod_{j=0}^{i-1} d_j \quad i = 1, \ldots, n \tag{8}$$

Since the tree must hold all n nodes, we have the following condition on the degree sequence:

$$\sum_{i=1}^{n} m_i \geq n \tag{9}$$

Since the quality of level 1 nodes depend only on d_0 (the degree of the server), this quality is maximized when d_0 is smallest possible (recall that α is fixed). Thus, we are interested in determining the smallest d_0 for which the sequence of degrees given by Equation (7) satisfy the condition in Equation (9). Note that such value would give the "optimal" tree quality assuming all nodes have identical quality. We can numerically obtain this value by iterating over d_0.

Finally, we have yet to establish that this tree is indeed the optimal tree that can be generated by any offline algorithm. The issue is that we assume here that all nodes have identical degree, a restriction that may not hold on the optimal tree. This is why we have been using the word optimal between quotes to refer to this tree.

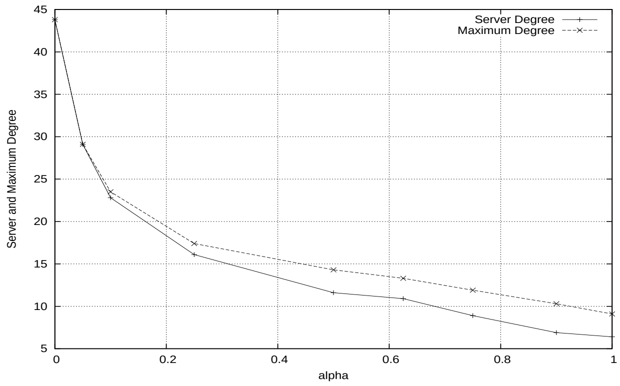
5. Results

In order to investigate the topological features of the distribution tree generated by the proposed model, we developed simulator to simulate the growth process. As with the model, the simulation starts out with the video server (i.e., root node). Peers are then sequentially added to the video distribution tree following the preferential attachment rule given in Equation 2. After each node is added to the tree, the attachment probabilities of all nodes in the tree are recomputed. For the results that follow, simulation stops after exactly $n = 60000$ nodes are added to the tree[2]. Each simulation scenario is executed 20 times and we report on the sample average of each metric. Finally, we are interested in the behavior of the model as a function of its sole parameter α, which determines the relative importance between small node degree and short distances when assessing the quality of nodes.

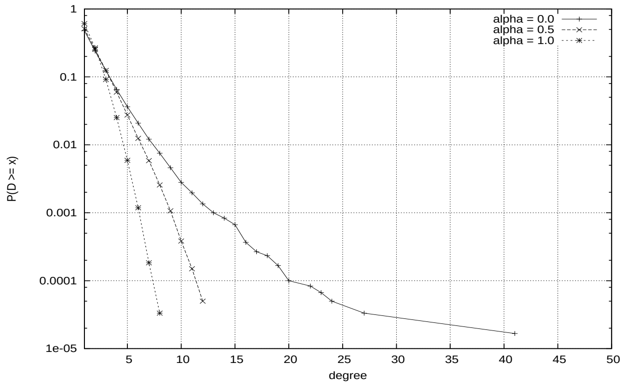
5.1. Node degree

We start by investigating the node degree. Figure 3(a) shows the degree of the server and the maximum degree of nodes in the distribution tree. We note that both server degree and maximum node degree decrease monotonically with α, varying from over 40 (when $\alpha = 0$) to less than 10 (when $\alpha = 1$). Moreover, both server degree and maximum node degree exhibit a similar trend and values. What is most surprising is that the server and maximum node degree do not vary significantly with α. Recall that as α tends to zero, the contribution of the degree to the node utility (Equation 1) tends to zero. Thus, one would expect to observe trees with much larger maximum degrees in the attempt to reduce distances. In the limit, when $\alpha = 0$, one would expect the server degree to be proportional to n, the number of nodes in the tree. However, this is not the case. Intuitively, this does not occur because the probability of choosing a specific node in the tree (Equation 2) is small and *decreases* with n when α is very small. Thus, it is extremely unlikely that a single node will attract all arriving nodes, leading to a star topology. We soon make this argument precise.

[2]Results using different number of nodes (but large) to construct the tree are qualitatively very similar.



(a) Degree of the server and maximum degree in the tree as a function of α



(b) CCDF of node degree (including server) for different values of α.

Figure 3. Numerical evaluation of node degree in the video distribution tree.

Figure 3(b) shows the complementary cumulative distribution function (CCDF) of the node degree for different values of α. We note that the tail of the degree distribution increases as α decreases. For larger values of α (i.e., $\alpha > 0.5$), the degree distribution drops very sharply. In any case, even for smaller values of α, the tail does not seem to follow a power law degree distribution (note the semi-log scale of the graph).

5.2. Node distances

We now consider node distances, which measures the distance in hops from the node to the root of the tree (i.e., the server). Figure 4 shows the average and maximum node distance as a function of α. As expected, we note that both average and maximum distances increase as α increases, with the average value ranging from around 6 (when $\alpha = 0$) to 15 (when $\alpha = 1$). Note that both maximum and average exhibit a similar trend. As with node degree, what is most surprising is that distances are not large, even when considering the maximum distance in the tree. Recall that as α approaches 1, the contribution of the node distance to the node's utility (Equation 1) tends to zero. Thus, one would expect to observe much larger distances, with values proportional to n in the limit (when $\alpha = 0$). Again, this is not the case. Intuitively, the probability of choosing a specific node decreases with n when α is close to 1. Thus, it is extremely unlikely that a leaf node will always be chosen by an arriving node, which would lead to a line tree and large distances.

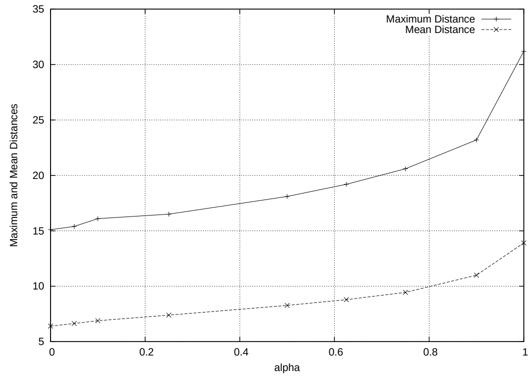


Figure 4. Mean and maximum node distances as a function of α.

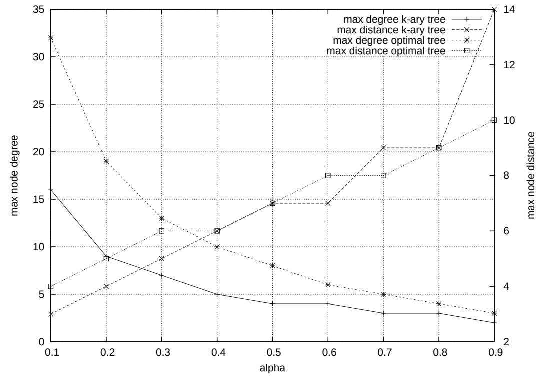


Figure 5. Maximum node degree and maximum node distance for both comparison trees as a function of α.

Figure 5 shows the maximum node degree and the maximum node distance for both comparison trees. For the complete k-ary tree, all nodes have the same degree, given by k (recall that this is the best k-ary tree). For the "optimal" tree, the server degree has the largest degree. The maximum distance measures the height of the two distribution trees. Observe that the maximum degree of the "optimal" tree is always larger than that of the complete k-ary tree. This is expected, as the "optimal" can vary the node degree to better accommodate the nodes, starting out with a larger degree for the server. We can also observe that for α near 0.5, these trees have similar degree and distance with the tree generated by the proposed model (see Figures 4 and 3(a)).

5.3. Tree quality

We now consider the quality of the video distribution tree generated by the model. In order to have a parameter for comparison, we consider the quality of the best complete k-ary tree and the quality of the "optimal" distribution tree, as described in Section 4. Thus, for a given α, we consider the complete k-ary tree of highest quality, as determined by Equation (5).

Figure 6 shows the tree quality as a function of α for both the tree generated by the model and the comparison trees. Surprisingly, for a wide range of α values, the quality of the of the tree generated by the model is similar to the quality of the best k-ary tree, being slightly superior in some cases. However, at the extremes, when either α approaches zero or one, the best k-ary tree becomes significantly superior to the tree generated by the model. Intuitively, this occurs because the model cannot generate extremely degenerate trees, like the line tree and the star tree, that yield the best tree quality when $\alpha = 1$ and $\alpha = 0$, respectively.

As illustrated in Figure 6, the "optimal" distribution tree has a quality superior to both the tree generated by the model and the best complete k-ary tree. However, the quality of the tree generated by the model is at most 1/3 lower than the "optimal" tree quality (for the example illustrated in the figure).

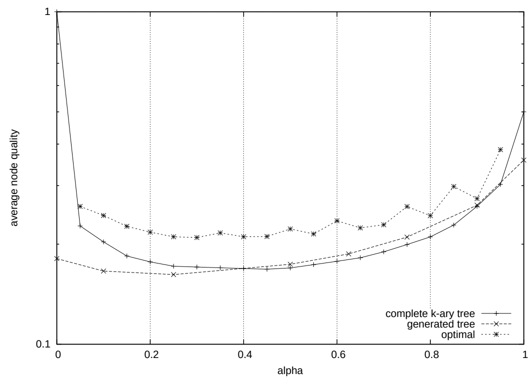


Figure 6. Tree quality for the best complete k-ary tree, the "optimal" tree and the tree generated by the model as a function of α.

5.4. Discussion

The numerical evaluation of the model reveals some interesting observation about its behavior. Contrary to the classical "preferential attachment" model, where a few nodes tend to dominate the graph, attracting most arriving nodes, our model leads to a self-organization of the nodes in the tree, without the appearance of any degenerate structure. Even at extremes, when α approaches zero or one, the model does not produce trees with degenerated structure. Of course, this may not be optimal at extremes, but otherwise it attests to the robustness of the model under various α values.

To make this argument precise, consider the case where $\alpha = 0$. Assume that after the arrival of n nodes to the system, we have a star topology, where the server is the center of the star and all n nodes are directly connected to it. In this case, the probability that the next arriving node connects to the server, as determined by Equation (2), is $p_s = 2/(2+n)$, while the probability it connects to a given node is $p_v = 1/(2+n)$, for any leaf node v. Note that the probability of attaching to the server is twice the probability of attaching to any given node v. However, the probability of attaching to *any* leaf node is $n/2$ larger than the probability of attaching to the server. Thus, it is very likely that the arriving node will break the star topology, connecting itself to a leaf node and not the server.

Another interesting observation is that the quality of the trees generated by the model are surprisingly good, comparable and sometimes superior to the quality of the best complete k-ary tree. In some cases (for some values of α), the quality is also comparable to the quality of the "optimal" distribution tree. This hints on the power of a self-organized mechanism to generate video distribution trees. Note that constructing a complete k-ary tree or an "optimal" distribution tree is much more entailed than using a simple probabilistic approach as proposed in the model. Nonetheless, the more rigid structure generated by the offline mechanisms do not necessarily lead to trees with much superior quality, at least when not considering the extreme cases for α. In these extremes ($\alpha = 0$ or $\alpha = 1$), it is clear that degenerate tree structures are significantly superior.

Finally, we note that extreme cases for the utility function, when α approaches zero or one, are likely not to be of interest. As video quality depends inherently on both node degree and node distance, these two aspects are likely to be present evaluation of system quality. Under this condition, the proposed model exhibit surprising properties, such as very good tree quality.

6. Conclusion and future work

Constructing efficient video dissemination trees is a real burden for tree-based P2P streaming systems. In this paper, we considered a simple mechanism based on the idea of "preferential attachment", where arriving nodes have preference for connecting to nodes with both low degree and short distances to the server. The single parameter α of the proposed model weighs the importance of these two fundamental aspects that used to define video quality received by nodes.

Through a numerical evaluation, we observe that the proposed model leads to an effective self-organization of the nodes, generating trees that do not exhibit extreme topological properties (e.g., a star topology) but that can deliver very good video quality under a wide range of quality measures (i.e., various ranges for α). In particular, the quality of the tree are comparable to offline tree construction mechanisms.

As for future work, we are currently investigating other online tree construction mechanisms in order to assess the benefits of informed guessing, as proposed in this paper. For example, compare the proposed mechanism to an online mechanism where nodes may attach themselves to parents chosen uniformly at random.

Another line of work we are investigating is the impact that selfish peers (i.e., peers that do not provide service to other peers) have on topological properties and quality of the video distribution tree. In the current study, all nodes are altruistic and contribute to the system as requested by the mechanism.

References

Albert, R. and Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97.

Banerjee, S., Lee, S., Braud, R., Bhattacharjee, S., and Srinivasan, A. (2004). Scalable resilient media streaming. In *Proc. of ACM International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286:509–512.

Bonald, T., Massoulié, L., Mathieu, F., Perino, D., and Twigg, A. (2008). Epidemic live streaming: optimal performance trade-offs. In *SIGMETRICS'08: Proc. of the 2008 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 325–336.

Carra, D., Lo Cigno, R., and Biersack, E. (2007). Graph based analysis of mesh overlay streaming systems. *IEEE Journal on Selected Areas in Communications (JSAC)*, 25(9):1667–1677.

Hei, X., Liang, C., Liang, J., Liu, Y., and Ross, K. (2007). A measurement study of a large-scale P2P IPTV system. *IEEE Transactions on Multimedia*, 9(8):1672–1687.

hua Chu, Y., Rao, S. G., Seshan, S., and Zhang, H. (2002). A case for end system multicast. *IEEE Journal on Selected Areas in Communications (JSAC)*, 20(8):1667–1677.

Kumar, R., Liu, Y., and Ross, K. (2007). Stochastic fluid theory for P2P streaming systems. *INFOCOM 2007: Proc. of the 26th IEEE International Conference on Computer Communications*, pages 919–927.

Li, B., Yik, K., Xie, S., Liu, J., Stoica, I., Zhang, H., and Zhang, X. (2007). An empirical study of the Coolstreaming system. *IEEE Journal on Selected Areas in Communications (JSAC)*, 25(9):1627–1639.

Liu, J., Rao, S., Li, B., and Zhang, H. (2008a). Opportunities and challenges of peer-to-peer internet video broadcast. *Proceedings of the IEEE*, 96(1):11–24.

Liu, Y., Guo, Y., and Liang, C. (2008b). A survey on peer-to-peer video streaming systems. *Peer-to-Peer Netw Appl*, 1(1):18–28.

Sevim, V. and Rikvold, P. A. (2006). Effects of preference for attachment to low-degree nodes on the degree distributions of a growing directed network and a simple food-web model. *Phys. Rev. E*, 73(056115).

Small, T., Liang, B., and Li, B. (2006). Scaling laws and tradeoffs in peer-to-peer live multimedia streaming. In *MULTIMEDIA'06: Proc. of the 14th annual ACM international conference on Multimedia*, pages 539–548.