P2P vs. IP Multicast: Comparing Approaches to IPTV Streaming Based on TV Channel Popularity

Alex Bikfalvi[a,b,*], Jaime García-Reinoso[b], Iván Vidal[b], Francisco Valera[b], Arturo Azcorra[b]

[a] *Institute IMDEA Networks, Avenida del Mar Mediterraneo, 22, 28918, Leganés, Madrid, Spain*
[b] *University Carlos III of Madrid, Avenida de la Universidad, 30, 28911, Leganés, Madrid, Spain*

Abstract

Already a popular application in the Internet, IPTV is becoming, among the service providers, a preferred alternative to conventional broadcasting technologies. Since many of the existing deployments have been done within the safe harbor of telco-owned networks, IP multicast has been the desired streaming solution. However, previous studies showed that the popularity of the TV channels follows the Pareto principle, with the bulk of TV channels being watched only by a small fraction of viewers. Recognizing the potential scalability issues, we believe that multicast streaming approach may not be desirable for unpopular TV channels, especially when there are many such channels in the provider's service package. For this reason, the peer-to-peer content distribution paradigm is seen as an alternative, in particular for non-popular content. In order to analyze its viability, in this paper we perform a comparative analysis between IP multicast and a peer-to-peer overlay using unicast connections as streaming approaches, in the context of channels with different degrees of popularity. The analysis targets the bandwidth utilization, video quality and scalability issues, and our findings show that while multicast is always more efficient, peer-to-peer has a comparable performance for unpopular channels with a low number of viewers.

Keywords: IP multicast, unicast, peer-to-peer, IPTV, TV channel, popularity

1. Introduction

While television over the Internet Protocol or IPTV has been existing for some time, only in recent years it gained a significant attention from service providers. This interest has led to several commercial deployments, usually along telephony and Internet access service packages. However, as previous studies recognize, these commercial IPTV architectures have been implemented on a limited scale by telcos managing their own network and offering a limited set of TV channels.

Because these private networks do not suffer the restrictions of the public Internet, IP multicast has been the preferred technical solution as it can deliver the best performance and can be deployed with existing protocols and equipments. However, with the advent of the next-generation networks and the groups advocating for deregulation of the IPTV market, it is expected that in the future many of these telco-owned networks have to support a larger number of service providers, and consequently a large number of channels.

A comprehensive study on watching television [1] revealed that the TV channel popularity is distributed similar to the Pareto principle, the vast majority of channels having only a small fraction of viewers. Depending on the hour of the day, 90 percent of the channels are watched only by 20 percent of the total number of active subscribers. In addition to this findings, semi-interactive techniques such as Near Video on Demand (NVoD), where the same TV program is broadcast several times, will increase the number of TV channels as well. As a final argument, in today's Internet we witness a large growth of user generated video content, where most of it is published as content-on-demand. However, similar to the demand for live audio streaming, that spawned a large number of Internet radio channels, we argue that even residential users can be interested in generating their own live TV content.

These trends inherently expose the telcos to a number of issues. First, under these circumstances it is no longer affordable to use static multicast to stream all TV channels, as some telcos do in the present. Second, using dynamic IP multicast can lead to scalability problems that have been long studied [2, 3, 4, 5]. Finally, these scalability issues can translate to an increased cost for the service provider, especially when the number of users viewing a multicast channel is low.

For these reasons, in this paper, we explore the possibility of using a peer-to-peer (P2P) approach to complement the functionality of dynamic IP multicast for channels with a low popularity. This results in a *hybrid* IPTV system, as the one illustrated in Fig. 1 where a subset of channels are

[*] Corresponding author
Email addresses: alex.bikfalvi@imdea.org (Alex Bikfalvi), jgr@it.uc3m.es (Jaime García-Reinoso), ividal@it.uc3m.es (Iván Vidal), fvalera@it.uc3m.es (Francisco Valera), azcorra@it.uc3m.es (Arturo Azcorra)

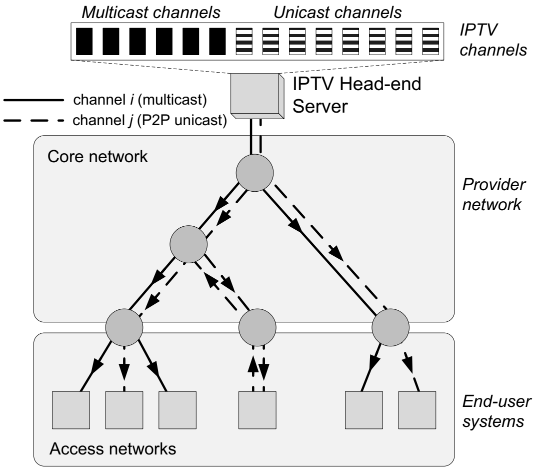

Figure 1: Concept of a hybrid IPTV system with multicast and P2P unicast channels.

streamed with IP multicast and the remaining with P2P unicast connections. Using P2P, in addition or in replacement of multicast for IPTV streaming, is an interesting problem that has been tackled in many recent papers and standardization drafts. In Sec. 2, we go deeper in some of this related work that we used as source of inspiration.

Our comparison focuses on three key parameters: the *total bandwidth* necessary to stream the TV content, the *quality* of the stream delivered to the user and the *scalability* issues. The analysis follows two approaches to ensure the accuracy and a correct explanation for our results: an *analytical model* for the network utilization and *computer simulations*. The analytical model uses the information from the network topology, the TV watching model, the organization of the P2P overlay and the channel popularity to derive a mathematical expression for the bandwidth utilization. Our proof shows that, in general, the bandwidth can be expressed as function of the network topology and the channel popularity. The effect of the network topology is measured by the average length of a P2P unicast connection and the size of a multicast tree, both in number of topological links traversed by the video stream.

In order to emphasize the effects of the channel popularity, we assume the TV channels are grouped only in two categories, where all the channels in a category have the same popularity. The channels from the category with the lowest popularity are called *unpopular*, while the rest are called *popular*. A selected set of channels is streamed using IP multicast, while the rest require IP unicast connections. We assume that IP multicast is more beneficial for the channels with a high popularity, and therefore these channels will be preferred when the number of available multicast groups is smaller than the number of channels. For channels that require unicast connections, we examine the possibility of using a *P2P overlay*. This approach tackles the scalability issue of a client-server solution, by preferring streaming connections to other set-top boxes, whenever possible. Since our study relies on existing P2P streaming algorithms, we will not cover implementation aspects of the P2P part of the system. Possible solution for a telco-managed IPTV infrastructure can be similar to the one described in [6], where the P2P functionality is distributed between the set-top boxes, or in [7], proposing a centralized P2P IPTV architecture for a next-generation-network. P2P proposals for Internet streaming include [8, 9, 10, 11, 12].

Our findings show that although dynamic IP multicast is always better than P2P unicast, the difference in terms of bandwidth utilization for unpopular channels can be negligible. Because we believe that bandwidth utilization alone is not a fair method of expressing the real cost of using multicast, we perform an empirical comparison between the multicast bandwidth utilization and its scalability measured as number of forwarding entries. Our results reflect that while there is a large gap in bandwidth savings between popular and unpopular channels, the difference is not that large in terms of forwarding entries, which means that using an additional multicast group for an unpopular channel brings almost the same cost in terms of scalability but only a small benefit in terms of saved bandwidth. In terms of delivered quality, the P2P design is always more challenging, especially for multi-channel streaming where channel-changes churn is added on the peer churn. Nevertheless, our results show that for a carefully designed system, acceptable levels of video quality are achievable.

2. Related Work

The P2P approach to IPTV streaming in the Internet has been an interesting topic for many years. Most of the proposals were justified by the lack of a viable alternative, such as IP multicast, dealing with the challenge of connecting a large user base to a single TV server. As IPTV started to penetrate in the commercial providers, usually having their own infrastructure in place, IP multicast became again the preferred choice. In addition, new standardization efforts by ETSI TISPAN concerning the next-generation mobile and fixed networks include IPTV delivered via multicast [13] [14].

However, in the existing deployments, telcos rely on a static multicast infrastructure, where all channels are streamed to the edge of their network, usually a DSLAM or a point-of-presence, allowing the users to change channels with minimum delay. This approach works well with a small number of channels, but can be a possible issue for a larger number. For this reasons many proposals, including early technical reports from TISPAN, still at the draft status, consider P2P as an alternative even for telco-owned networks [15]. Following this idea, in one of our previous papers, we presented an exclusive IPTV P2P system for an IMS-based next-generation network [7].

In [6] the authors use traces from a large commercial IPTV provider to show how P2P compares in terms of bandwidth costs to static IP multicast that is currently used by most providers. In addition, they study as well the effectiveness of topology-aware P2P, concluding that only topology-oblivious P2P performs worse than static multicast and only during prime-time. Their results show however, that dynamic IP multicast is always better and they put the preference for the P2P choice on the lack of proper support in the routers and not well understood multicast aggregation.

Using their inspiring results, we extend their work by assuming that dynamic multicast is always used in a hybrid multicast-P2P streaming system taking into account the channels popularity. We argue, that although unicast in general and P2P in particular cannot outperform multicast, a fair comparison between them should also include the multicast overhead and scalability issues.

From this perspective, since early papers like [16, 17, 18], researchers have studied the cost and proper pricing of multicast transmissions. Some proposals perceive each multicast group as a single resource suggesting a flat-rate charging approach, regardless on the number of viewers. While our work is not centered on the issue of multicast pricing, a topic that could be under debate, these studies are enough to trigger the question of whether a real multicast pricing should include more than just its bandwidth costs.

3. Analysis Setup and Assumptions

Our comparison between the multicast and P2P streaming follows two approaches. The first is an analytical analysis of the bandwidth utilization with the purpose of serving as a general case model. The second involves computer simulations applied to a set of scenarios based on several network topologies and P2P algorithms. Their goal is to ascertain the validity of the analytical approach including as additional prerequisites the selection of a model for the user behavior and a pattern for the channel popularity.

3.1. Watching the TV Channels

We begin by introducing the following notations, representing the most important parameters that we shall use throughout the paper:

U: the number of TV users (subscribers);

N: the number of TV channels in the provider's service package;

g: the number of channels streamed with IP multicast (the remaining $N - g$ channels use P2P unicast connections);

T: an observation interval of the IPTV system.

In addition, the TV channels are divided into a number of *channel categories*, where all TV channels in a category have the same popularity. We have:

K: the number of channel categories;

Q_j: the probability of selecting a channel from category j;

M_j: the number of TV channels in the category j.

Finally, an individual channel i is characterized by the following parameters:

$\mathscr{P}_i$: the channel popularity;

p_i: the channel probability;

v_i: the number of viewers.

3.1.1. Channel Holding Time

The behavior of the users of changing the channels is modeled according to an existing and comprehensive study of users watching TV based on data from a real IPTV service provider [1]. One of the findings from this study is a probability distribution for the channel holding time in a deployed IPTV system, i.e. the duration a user watches a given channel. We reuse these results in our experiments, under the assumption that *all channels from a category have the same selection probability* (i.e. the probability distribution for the TV channels in a category is uniform). This assumption is justified by the lack of any real data showing that channel holding times for different channels follow different probability distributions, and by our desire to simplify the channel watching model such that the effects on the network performance are easy to identify. In this context, an interesting result is that approximately 72% of holding times last less than one minute (and over 60% less than 15 seconds), a behavior that the authors of the study call *surfing*. Their data includes both churn generated by channel changes and the churn generated by users switching on and off their TV.

3.1.2. Channel Probability

In order to assess the impact of hybrid streaming mechanism on a number of channels with different levels of popularity, we propose a simple channel popularity model. The objective of the model is to devise a simple selection criterion of the *next channel* during a channel change, such that in long term each channel will have a deterministic popularity. For a channel change, the next channel is always different from the current channel. In 3.1.3 we give a rigorous definition of the channel popularity, and we explain its relationship to the model.

We start by introducing the following definitions.

Definition 1. *The* probability *of a channel i, denoted by p_i, is the probability that a user will select channel i during a channel change.*

Definition 2. *The* probability *of a channel category j, denoted by Q_j, is the probability that a user will select a channel from category j during a channel change.*

In the experimental evaluation, we examine the streaming performance where the channels are divided into two categories, the first having M_1 *popular* channels, and the second having M_2 *unpopular* channels, and where $M_1 + M_2 = N$. The probability of each category is Q_1, and Q_2 respectively, where $Q_1 > Q_2$ and $Q_1 + Q_2 = 1$. With these assumptions the channel probability is:

$$p_i = \begin{cases} \dfrac{Q_1}{M_1} & \text{if } 1 \leq i \leq M_1, \\ \dfrac{Q_2}{M_2} & \text{if } M_1 < i \leq N. \end{cases} \tag{1}$$

We choose M_i and Q_i such that the condition $p_i \geq p_{i+1}$ is always fulfilled. For a single channel category, in a scenario where all channels are equally popular, their probability is $1/N$. Therefore, in any different scenario, channels having $p_i > 1/N$ are *popular* and channels with $p_i < 1/N$ are *unpopular.* In appendix Appendix A we give a detailed explanation of the channel change model, proving the channel probability expressed in (1).

3.1.3. Channel Popularity and Channel Viewers

In addition to the channel probability, we introduce two metrics that assess the impact of a particular channel during an arbitrary observation period of the IPTV system, denoted by T.

Definition 3. *The* popularity *of a channel* i*, denoted by* $\mathscr{P}_i$*, is the total amount of time channel* i *is watched by any user throughout the observation period.*

Essentially, the channel popularity is a duration of time representing the sum of all its sessions for all users (e.g. if during the observation period T, two users watch channel i for x and y seconds, respectively, $\mathscr{P}_i|_T = x + y$). With this definition, the channel popularity captures the effect of both the number of users watching the channel (i.e. the user dimension represented by the channel probability) and the length of their respective sessions (i.e. the temporal dimension represented by the channel holding time).

Definition 4. *The* viewers *of a channel* i*, denoted by* v_i*, is the average number of subscribers viewing channel* i *during the observation period.*

In other words, the number of channel viewers is a measure of the channel's impact on the entire subscriber base, and consequently determines the effect of viewing the channel upon the network.

In addition to the previous metrics, we introduce the *total popularity*, denoted by $\mathscr{P}^*$, as the sum of the popularity for all channels:

$$\mathscr{P}^* = \sum_{i=1}^{N} \mathscr{P}_i. \tag{2}$$

The popularity of a channel can be expressed as relative to the total popularity.

Under the earlier assumptions on the channel changing model, between the model parameters, the channel probability, the channel popularity, the total popularity and the number of viewers there exists a set of important relations.

(i) During a given observation period T, the total popularity is:

$$\mathscr{P}^* = U \cdot T. \tag{3}$$

(ii) When the observation period approaches infinity, the channel popularity becomes proportional to the channel probability, that is:

$$\lim_{T \to \infty} \frac{\mathscr{P}_i}{\mathscr{P}^*} = p_i. \tag{4}$$

When the observation period is less than infinity, the popularity is only approximatively proportional to the channel probability.

(iii) The number of viewers equals the channel popularity divided to the observation period:

$$v_i = \frac{\mathscr{P}_i}{T}. \tag{5}$$

(iv) When the observation period approaches infinity, the number of viewers can be expressed in terms of channel probability:

$$v_i \to U \cdot p_i \quad \text{when} \quad T \to \infty. \tag{6}$$

In appendix Appendix B we extend the definitions of the previous metrics, and we prove the relationships between them.

3.2. Network Topology and Its Impact on Unicast and Multicast Streaming

For our comparison, we consider a telco-like network topology, comprising of a *core network* and a number of *access networks.* The access networks are connected to the core network through a set of *edge routers.* The access networks consist of direct links between the edge router and the set-top box from the customer premise, consistent to an xDSL access technology. The IPTV head-end server is randomly placed at any core router, and the users (or viewers) are uniformly distributed across the access networks.

From the perspective of this paper, the core network graph is described by a triplet measuring the connectivity and average distance between nodes in P2P unicast and multicast traffic scenarios:

m: represents the ratio between the number of routers and links in the network (half of the average node degree);

l_u: represents the average distance between two random edge routers;

$l_m(v)$: represents the average size of the shortest-path tree from a random source router to the edge routers of v set-top boxes (users).

According to previous studies like [16] and [19], real networks, such as routing and AS topologies of the Internet, have a value for m of two or higher.

In practice, the multicast tree size (l_m) with respect to the user group size (v) is difficult to determine with a good

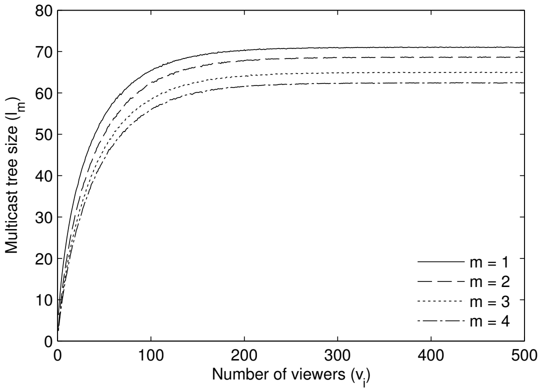

Figure 2: The multicast tree size (l_m) versus the number of users (v).

m	1	2	3	4
l_u	6.18 ± 0.002	3.30 ± 0.0009	2.70 ± 0.0007	2.40 ± 0.0006
l_m^{∞}	71.04 ± 0.04	68.65 ± 0.04	64.95 ± 0.04	62.43 ± 0.03
k	0.68	0.82	0.85	0.87

Table 1: The set of parameters describing each network topology.

accuracy. The *power scaling law* of Chuang and Sirbu [16] gives a relationship between unicast and multicast. This law introduces k, a multicast scaling factor depending on the network topology, and holds only when the number of users is lower or comparable to the number of edge network nodes (access routers). The *power scaling law* is expressed as:

$$l_m(v) = l_u \cdot v^k. \tag{7}$$

When the number of users is higher than the number of edge nodes, the tree size enters in a *saturation region*, where it does not increase even if more users are added. Figure 2 illustrates the relationship between l_m and v in the case of a Waxman-type core network with 100 routers, 50 access routers and $m \in \{1, 2, 3, 4\}$.

When comparing the obtained l_m with their findings, we can observe the average tree size increases with the number of viewers v up to a limit determined only by the topology and the selection of the edge routers. We denote the saturation limit of the multicast tree size by l_m^{∞}. For our network topologies, this limit is reached for a multicast group size of more than 250 users.

Table 1 summarizes the measured values for l_u, l_m^{∞} and k for each selected topology. The multicast scaling factor, k, is between 0.8 and 0.9 for the topologies with a higher m, similar to the findings from [16] and [19].

4. Analytical Estimation of Bandwidth Utilization

In this section, we perform an analytical estimation of the total bandwidth utilization, taking into account the previous assumptions and a variable number of channels streamed using IP multicast, while the rest are streamed using P2P unicast connections. We denote by B_0 the bitrate of one video stream (we assume that all channels require the same streaming bit rate). The total bandwidth utilized in the network, denoted by B, is the sum of the total bandwidth used in the access network (B_A) and the bandwidth used in the core network (B_C). The bandwidth in the access network can in turn be written as the sum of uplink ($B_{A,u}$)[1] and downlink ($B_{A,d}$) bandwidth, while the bandwidth in the core network can be written as the sum of the bandwidth used for P2P unicast ($B_{C,u}$) and multicast ($B_{C,m}$) streaming. We have:

$$B = B_A + B_C = B_{A,u} + B_{A,d} + B_{C,u} + B_{C,m}. \tag{8}$$

In a similar way, we define by $b(i)$ the bandwidth utilization for channel i, with the individual components being denoted by: $b_A(i)$, $b_{A,u}(i)$, $b_{A,d}(i)$, $b_C(i)$, $b_{C,u}(i)$ and $b_{C,m}(i)$. All bandwidth components are expressed as average over the observation period T. We note that the total bandwidth utilization does not include the bandwidth required between the network and the IPTV server belonging to the service provider.

4.1. Access Network Downlink Bandwidth ($B_{A,d}$)

Under the assumptions that all U users are continuously viewing a channel, the access network downlink bandwidth is the product between the number of users connected and the stream bit rate:

$$B_{A,d} = U \cdot B_0. \tag{9}$$

4.2. Access Network Uplink Bandwidth ($B_{A,u}$)

The access network uplink bandwidth is determined by the channels $i \in \{g+1, ..., N\}$ using P2P unicast streaming connections. For any P2P unicast channel i, a fraction of viewers stream directly from the head-end server, while the rest use the P2P overlay. If we consider the uplink bandwidth for both the server and the peers, we have:

$$\begin{aligned} B_{A,u} &= \sum_{i=1}^{N} b_{A,u}(i) = \sum_{i=1}^{g} B_0 + \sum_{i=g+1}^{N} v_i \cdot B_0 \\ &= g \cdot B_0 + \frac{B_0}{T} \sum_{i=g+1}^{N} \mathscr{P}_i. \end{aligned} \tag{10}$$

4.3. Core Network P2P Unicast Bandwidth ($B_{C,u}$)

The core network bandwidth utilization for P2P unicast streaming is determined by the last $N - g$ channels. For each of these channels, the average bandwidth is the

[1] We include the uplink bandwidth used by the head-end server in the total access uplink bandwidth.

product between B_0, the number of viewers v_i, and the average P2P unicast path length l_u introduced in Sec. 3.2. Depending on the P2P algorithm, the average P2P unicast path length can be the same for streams from both peers and the IPTV server. In these circumstances, we have:

$$\begin{aligned} B_{C,u} &= \sum_{i=g+1}^{N} b_{C,u}(i) = \sum_{i=g+1}^{N} B_0 \cdot l_u \cdot v_i \\ &= \frac{B_0 \cdot l_u}{T} \sum_{i=g+1}^{N} \mathscr{P}_i. \end{aligned} \tag{11}$$

4.4. Core Network Multicast Bandwidth ($B_{C,m}$)

In a similar manner, the multicast bandwidth in the core network for channel i is the product between B_0 and the average multicast tree size $l_m(i)$ introduced in Sec. 3.2. We have:

$$B_{C,m} = \sum_{i=1}^{g} b_{C,m}(i) = B_0 \sum_{i=1}^{g} l_m(v_i). \tag{12}$$

4.5. Total Bandwidth (B)

Using our analytical estimation of the bandwidth utilization, we studied the effects of three parameters affecting the network performance: (1) the network topology, (2) the channel popularity and (3) the P2P overlay. These results clearly demonstrate that, while dynamic IP multicast is always the most efficient, P2P can be an reasonable choice for unpopular channels even from the perspective of used capacity. The amount of bandwidth that is saved by using more multicast groups is very small compared with the amount saved for popular channels. We present our results from the perspective of the worst-case scenario, that is we tune one parameter while the rest would give the worst possible result.

4.5.1. Effect of the Network Topology

To study the effect of the network topology, we considered a simple popularity model[2] with $K = 2$ channel categories having $M_1 = 20$ and $M_2 = 80$ channels, with category probabilities Q_1 and Q_2, respectively. The system has $U = 1\,000$ users.

Figures 3 illustrates the total bandwidth utilization versus the number of channels streamed with multicast, determined with (8) for the chosen popularity model, and for different topology node degrees ($m \in \{1, ..., 4\}$). The figure shows that streaming more channels with IP multicast reduces the bandwidth utilization, suggesting that IP multicast is more desirable. However, there are two important observations. First, the network capacity saving is much smaller for unpopular channels, even when the number of viewers per channel is relatively high. For example, in Fig. 3, $Q_2 = 0.4$ corresponding to $U \cdot Q_2/M_2 = 5$ viewers per channel, whereas the results from [1] show that unpopular channels can have as little as 10 viewers. Second, better connected networks, with a higher m, reduce the absolute value of the saved capacity even further. Since, networks deployed in reality have an $m \geq 2$, we can see in both figures that streaming all unpopular channels with multicast connections saves only 50 percent of the bandwidth.

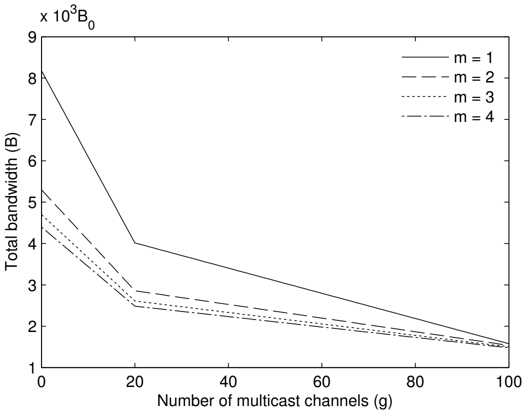

Figure 3: The effect of the network topology on the bandwidth utilization. The channel popularity model has 2 categories with $M_1 = 20$, $M_2 = 80$, $Q_1 = 0.6$ and $Q_2 = 0.4$.

4.5.2. Effect of the Channel Popularity

Second, we want to measure the impact of the channel popularity on the bandwidth utilization. Toward this end, we selected network topologies having $m = 2$, the typical case for deployed networks, and we modify the probability of the channel categories. Figure 4 shows the obtained results where the category probability of the popular channels category is $Q_1 \in \{0.6, 0.7, 0.8, 0.9\}$, with a category probability for the unpopular category of $Q_2 \in \{0.4, 0.3, 0.2, 0.1\}$, respectively.

The result from the figure summarizes our main claim, that as the channel popularity decreases their corresponding streaming bandwidth approaches that of IP multicast. This result can be explained further, by computing the average number of viewers for unpopular channels. When the category probability of the popular category is $Q_1 = 0.9$, we have $Q_2 = 0.1$ for $M_2 = 80$ unpopular channels. In this case, the number of viewers for an unpopular channel is $U \cdot Q_2/M_2 = 1.25$, making the size of the multicast tree comparable to the total length of the P2P unicast connections.

4.6. Bandwidth Versus the Channel Popularity

Expanding the previous bandwidth analysis for an individual channel, we can determine its variation with the

[2] We based the values for these models on the previous finding from [1] showing that 10 percent of the most popular channels account for almost 80 percent of the viewers. While we recognize that this model does not represent a real-life popularity distribution, our choice has the purpose of emphasizing the difference between popular and unpopular channels.

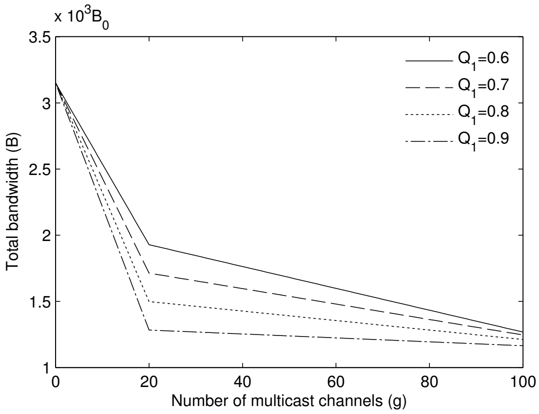

Figure 4: The effect of the channel popularity on the bandwidth utilization. The network topology has $m = 2$; the channel popularity model has two categories with $M_1 = 20$, $M_2 = 80$, a given Q_1 and $Q_2 = 1 - Q_1$.

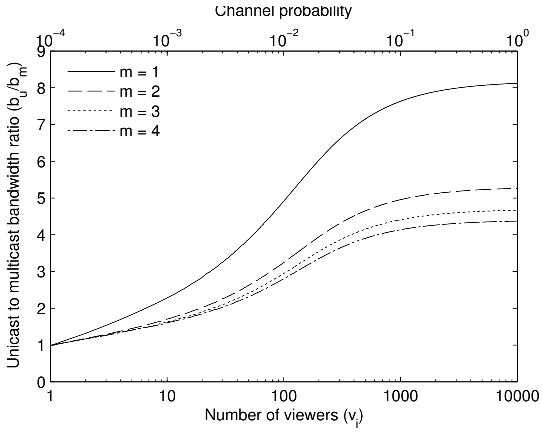

Figure 5: The ratio between P2P unicast and multicast bandwidth for one channel, depending on the channel probability or number of viewers.

channel popularity or channel probability. The P2P unicast bandwidth for channel i is:

$$\begin{aligned} b_u(i) &= b_{A,d}(i) + b_{A,u}(i) + b_{C,u}(i) \\ &= \frac{\mathscr{P}_i}{T} B_0 + \frac{\mathscr{P}_i}{T} B_0 + \frac{\mathscr{P}_i}{T} l_u B_0 \\ &= U B_0 p_i \left(2 + l_u\right). \end{aligned} \tag{13}$$

In a similar manner, the multicast channel bandwidth is:

$$\begin{aligned} b_m(i) &= b_{A,d}(i) + b_{A,u}(i) + b_{C,u}(i) \\ &= \frac{\mathscr{P}_i}{T} B_0 + B_0 + l_m(v_i) B_0 \\ &= B_0 \left(1 + U p_i + l_m(v_i)\right). \end{aligned} \tag{14}$$

Figure 5 illustrates the ratio between the P2P unicast and multicast bandwidth for a channel, $b_u(i)/b_m(i)$, versus its probability. This comparison of the streaming technologies, further emphasizes that for highly popular channels with probabilities higher than $1/N = 10^{-2}$, the unicast bandwidth required to serve the same number of users is four-to-six times higher than multicast for the least connected networks ($m \in \{1, 2\}$).

However, for channel probabilities much less than $1/N$, between 10^{-4} and 10^{-3} the unicast and multicast bandwidth becomes comparable. Note that under our assumed input data, according to (6) a channel probability of 10^{-4} corresponds to an average of one viewer per channel. The presented values have a small variation around 1 for $p_i \approx 10^{-4}$ due to the statistical measurement error of l_u and l_m. Theoretically, at this value b_u equals b_m, for there is one user per channel.

5. Results Validation and Interpretation

In this section, we validate the analytical analysis of the bandwidth utilization presented in Sec. 4, by comparing the previous results with the data obtained from computer simulation, and we offer some insights on the issues of delivered quality and scalability. Our experimental evaluation emulates a set of $U = 1\,000$ users watching TV over an IP network. Every user chooses from a pool of $N = 100$ available channels, with $M = 20$ channels being popular (a category probability of $Q = 0.6$). It results that a popular channel is being watched at any time by an average of $\frac{UQ}{M} = 30$ users, while an unpopular channel has an average of $\frac{U(1-Q)}{N-M} = 5$ users. A variable number of g channels are streamed using multicast while the rest use P2P unicast.

Based on these settings, our objective is to draw comparative conclusions between the multicast and P2P unicast channels in terms of bandwidth utilization, delivered experience and scalability issues. For this purpose, we developed a packet-level time-domain discreet event simulator, capable of accommodating large scale simulations of several video streaming algorithms. As we mentioned previously, user behavior and hence peer participation uses the data from [1] in conjunction with our channel popularity model.

5.1. Evaluation Settings and Simulation Setup

Network Topology: We used BRITE [20] to generate a set of medium-sized core networks based on a Waxman routing model and consisting of 100 routers and a links to nodes ratio $m = 2$, equivalent to a realistic but loosely-connected network. In order to diversify the core network topology, we generated 20 different network topologies using the same parameters. Hence, any result presented in this paper represents an average for all 20 network topologies. The IPTV head-end server is randomly placed at any core router, and the hosts (viewers) are uniformly distributed across 50 access networks with point-to-point links between a host and its corresponding edge router.

The core network and the link to the server are overprovisioned, with 1 Gbps links. In the access network, users' broadband connections are divided as follows: 15% with 1 Mbps downlink/256 kbps uplink, 20% with 3 Mbps/640 kbps, 50% with 6 Mbps/1 Mbps and 15% with 10 Mbps/1 Mbps, as a typical DSL access scenario in Europe. Since our goal is only to compare the streaming performance of multicast and P2P unicast TV channels (as opposed to examining their performance under various conditions), the evaluation does not consider additional third-party traffic.

Packet-level Simulator: Our in-house developed simulator draws concepts for well-established network simulators such as *ns*, but optimizes the simulation of certain network functions, in order to handle to large-scale video streaming application in a reasonable time[3]. It emulates the network functions (packet queuing, forwarding and routing), implements the network components (such as links, routers, hosts) and the media streaming server and client functions (such as coder/decoders, playback buffers). Some components, such as the client playback functions, can be used with both multicast and P2P streams with the goal of obtaining results as least biased as possible. The software receives as input the network topology, the model of the user behavior and channel popularity, and implements TV head-end servers and clients that can send and receive both P2P unicast and multicast channels.

Video Encoding and Video Traffic: We use a synthetic video source that generates an MPEG video streams with an I-to-I frame distance $d_{I-I} = 9$ and an I-to-P frame distance $d_{I-P} = 3$. Hence, every MPEG group-of-pictures (GOP) has one I frame, two P frames and six B frames. Every channel is encoded at a constant bit rate of 500 kbps and 25 frames per second, corresponding to an average quality video stream that enables the participation of most peers. In order to provide for a fair comparison that accommodates both multicast and unicast traffic, the video data is packetized in connectionless UDP datagrams transmitted over the best-effort IP network. In this manner, our results are presented as obtained from the simulator, with minimal post-processing[4]. We do not include any additional error correction, and hence packets lost or delayed due to congestion will result in missing frames at playback.

Multicast Channels: The end-hosts (i.e. set-top boxes) use IGMP to join or leave a multicast tree via their corresponding edge router. In the core network, the multicast trees are managed using the Protocol Independent Multicast - Sparse-Mode (PIM-SM), as it is one of the most common deployed. The PIM-SM rendezvous point (RP) router is manually configured as the router closest to the head-end server. In this way, we did not give the multicast channels any unfair disadvantage, although in a different setting where the RP is dynamically elected, the multicast will perform worse than in our findings.

P2P Unicast Channels and Streaming Algorithms: For P2P unicast streaming, the hosts (also called as *peers*) user a P2P live streaming protocol/algorithm. Because our intention is to rely on real-life P2P proposals, but in the same time be as general as possible without endorsing a particular proposal that might have its own advantages and disadvantages, we focused on the three main categories of P2P streaming protocols that have been proposed by the community.

(i) Single-tree streaming, also referred to as application-level multicast (ALM), and which tries to emulate the traditional network-level multicast by creating permanent connections between participating peers in a tree-like structure rooted at the source. The stream packets are seamlessly forwarded by every peer to its downstream neighbors.

(ii) Multiple-tree streaming, attempts to negate some of the disadvantages of a single tree where not all peers can participate (i.e. the leaves of the tree), and where the departure of a node will result in a temporary but total loss of the video for all its downstream peers. On the other hand, multiple trees are more challenging to manage and, in the case of video streaming, to synchronize. We divide the original stream into 8 stripes, and a client has the requirement of receiving minimum 4 stripes in order to start the playback.

(iii) Mesh streaming, uses dynamically-generated temporary connection through which peers exchange the video data. As opposed to the tree techniques, where peers typically *push* the data to their neighbors, mesh protocols work on-demand or *pull*, with receiver peers asking their neighbors for particular *chunks* or *segments* of the live stream. Segments are requested and delivered according to a scheduling strategy. In our evaluation we used the DoNET/Coolstreaming scheduling [9], an heuristic that tries to maximize the in-time delivery of segments.

For all three streaming protocols, the peer participation is managed centrally by a tracking server that stores the IP addresses of all peers watching a given channel. The necessity for the tracking server comes as compromise of using existing proposal that have not been designed to work well in a high churn environment. Peers may choose to advertise themselves to their neighbors based on their available uplink resources. In the tree techniques, peers simply keep track of their downstream neighbors, while in the mesh scenario, peers use a moving average to estimate the incoming request rate.

Finally, we acknowledge that our selected algorithms do not represent an exhaustive set, and that there are many other proposals with new optimizations. For instance, for multiple-channel streaming, a proposal for video-upload decoupling results in better performance at the expense of most peers receiving two channels at the same time [12]. Obviously, the behavior of such proposals do not fit well within out theoretical model since we assume

[3] The code for our simulator is freely available at the following URL: `http://enjambre.it.uc3m.es/~bikfalvi/projects/simstream`.

[4] The only post-processing of the measured transmitted traffic that we perform is to calculate the actual bitrate of an elementary stream (B_0) considering the additional overhead.

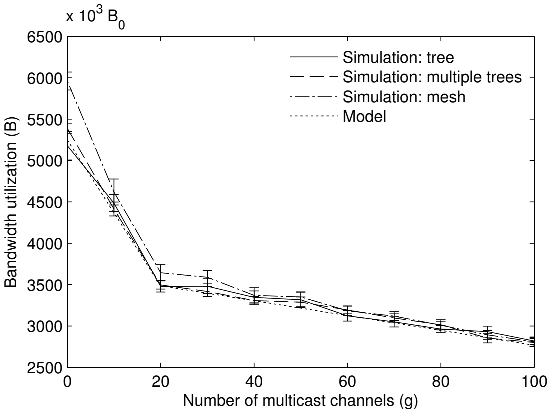

Figure 6: Comparison between the analytical model and simulation data. Simulation results were obtain for three P2P streaming algorithms; the channel popularity model has two categories with $M_1 = 20$, $M_2 = 80$, $Q_1 = 0.6$, $Q_2 = 0.4$; the network topology has $m = 2$.

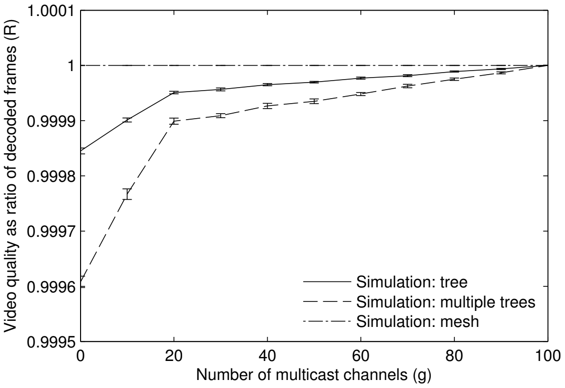

Figure 7: Comparison between the average video quality vs. the number of multicast channels. The quality is estimated at playback using the fraction of decodable frames criterion.

a client receives only one channel. For these reasons, we believe that classic P2P streaming techniques aided by a tracking server for peer management (commonly used in file-sharing), represents a fair selection for our experiments.

5.2. Bandwidth Utilization

The objective of this comparison is to examine whether a complex P2P algorithm can still be described by the equations from Sec. 4. In addition, the comparison shows that for a P2P algorithm the bandwidth utilization can be estimated if we know the P2P unicast path length (l_u) and the multicast tree size (l_m), under the assumption that contributing peers are uniformly distributed[5].

Figure 6 illustrates the match between the analytical and simulation data. The analytical model approximates very well the simulated P2P streaming algorithms, lying close to the 95% confidence interval of the experimental results, except when the mesh algorithm is used for popular channels. This result proves that l_u and l_m approximate with a good accuracy the effect of the network topology, for the typical usage case. The slight inconsistency for the mesh Coolstreaming algorithm is explained by the fact that we relied on the default settings, proposed by its authors, which are not well-suited to a multiple channel environment. In our case, we remind that over 60 % of the channel sessions hold for less than 15 seconds, whereas Coolstreaming is optimized for longer sessions: the video segments size has 1 second in duration (we rounded that value to 27 frames, i.e. 3 GOPs). With these settings, we observe that for popular channels, the stream segments are better distributed among contributing peers, but due to the high churn rate, their contribution is limited (i.e. peers leave without sending many segments to their neighbors) resulting in a lower amount of video traffic. On the other hand, for unpopular channels, where the average number of viewers is around five, content distribution is more limited, with a large fraction of peers streaming from the server, which is approximated by our model.

5.3. User Experience

The second criterion for our comparison is to estimate the impact on the user experience. Traditionally, IP multicast has been used reliably in many commercial deployments with acceptable levels of quality. On the other hand, due to their distributed nature, P2P techniques exhibit a lower performance. In order to assess the difference in terms of user experience, we focus on three main parameters: the *quality* of the decoded video, the channel *interruptions*, the channel *change delay* and the stream *synchronization delay*.

For the first two parameters we adopt the fraction of decodable frames criterion [21], that estimates the output quality as the ratio between decoded and expected frames:

$$R = \frac{N_{frames\_decoded}}{N_{frames\_expected}}. \tag{15}$$

With this metric, the quality is considered the best when $R = 1$ and the worst when $R = 0$. The video decoder calculates the number of expected frames based on the moment when the playback started, and the number of decoded frames considers the dependencies between different type of MPEG frames (e.g. the loss of an I frame will affect the entire GOP).

Figure 7 illustrates the average delivered quality versus the number of multicast channels, estimated using the previous criterion. We can observe that although average quality increases when the number of multicast channels

[5] When peers are not uniformly distributed (viewers for a channel are grouped in particular geographical area), the model can still be applied as long as we calculate new average distances between the peers based on the user distribution and their typical channel preferences. For simplicity, in our examination we assumed that both the users and their channel preference is uniformly distributed.

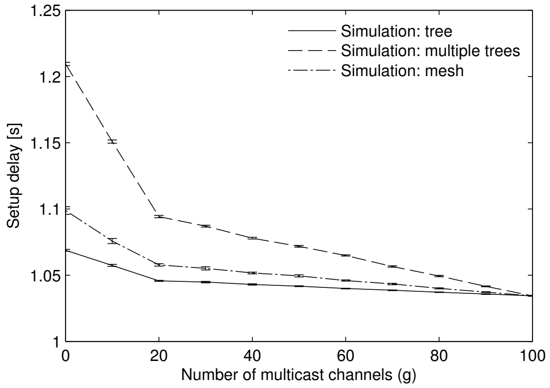

Figure 8: The channel change (setup) delay as a qualitative comparison between popular and unpopular channels.

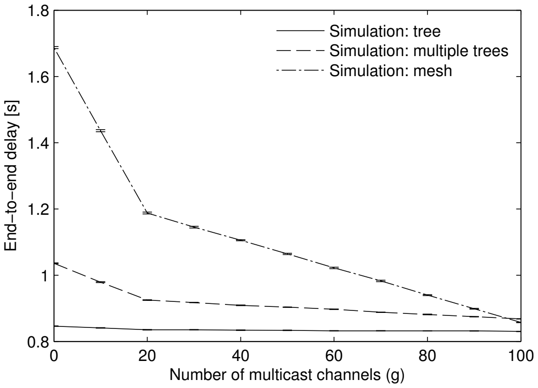

Figure 9: The channel end-to-end delay as a qualitative comparison between popular and unpopular channels.

increases, the difference is very small once we begin using P2P for unpopular channels. When using P2P for popular channels, the lower quality (due to frame losses and playback interruptions) is explained similar to our explanation for a lower bandwidth and is caused mainly by churn. The quality loss is higher for tree-based streaming, due to the lower complexity of peer participation management as opposed to the mesh structure. In addition, the on-demand nature of mesh streaming makes possible for peers to ask for missing packets, typically at the expense of a greater delay, control overhead (and bandwidth) and buffer requirements.

Figure 8 shows the *channel change* (or *channel setup* delay), which measures the difference between the moment the user changes the channel and the moment the playback commences. The *end-to-end* delay, illustrated in Fig. 9, measures the difference between a certain frame is transmitted at the server and its playback moment at the user and illustrates the end-to-end delay the video data traverses through the network. We intend the figures to illustrate qualitatively the several P2P streaming techniques and multicast rather than quantitative absolute values. This is because each P2P algorithm has it own requirements in terms of amount of buffering an synchronization, parameters which can be tuned (often at a trade-off with the quality) to increase or decrease the startup delay. In our case, it is worth noting that the mesh algorithm generates higher delays due to the higher degree of coordination that is required between peers, although the delay average is much less when using P2P only for unpopular channels. In addition, although multicast exhibits a superior performance, in absolute values the difference between the two is small suggesting that at the expense of this difference, which might be at the lower limit of perception to the user, the IPTV provider can benefit greatly in situations where multicast is not feasible.

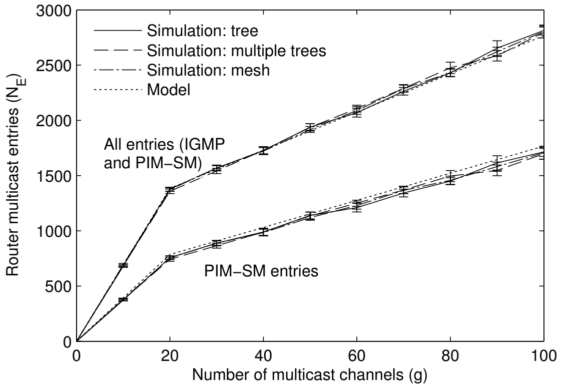

Figure 10: The number of multicast entries for each network topology and for a channel model with two categories ($M_1 = 20$, $M_2 = 80$, $Q_1 = 0.6$, $Q_2 = 0.4$). The figure emphasizes the small difference in number of multicast entries between popular and unpopular channels, rather than their absolute value.

5.4. Scalability Issues

The scalability is one of the multicast issues that has been widely recognized and intensively studied [2, 3, 4, 5, 19, 22]. When becoming a member of a multicast tree, every router in the network adds a new multicast forwarding entry. However, unlike for unicast routing, multicast addresses are not hierarchical and there is no natural way of consolidating multicast entries.

This problem is aggravated for core network routers that will have to handle a very large number of forwarding entries when many multicast groups are used. While a number of solutions have been proposed, such as forwarding entries aggregation (the multicast trees sharing the same interfaces and having a common address prefix are represented by a single entry), there is still no uniformly implemented solution. Furthermore, aggregation is not well suited for low popularity TV channels having few users and following many disjoint paths.

While we acknowledge that multicast scalability is only a performance problem, which in an IPTV scenario de-

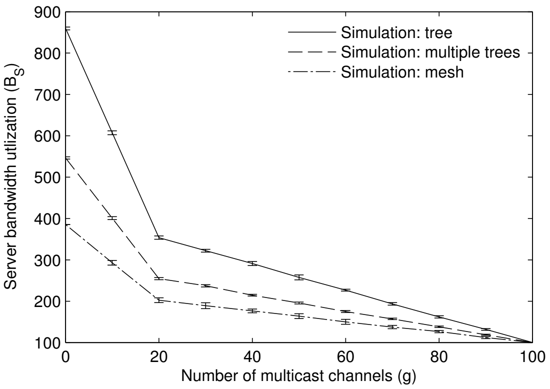

Figure 11: The server utilization as a comparison between the three classes of P2P streaming schemes.

pends on the number of channels and users, in this section we compare the multicast benefit in terms bandwidth and its drawback in terms of scalability. For this purpose, the following equations estimate the number of multicast entries for IGMP and PIM-SM routers, assuming the point-to-point connection from hosts to the edge router:

$$N_{E,\mathrm{IGMP}} = \sum_{i=1}^{g} v_i = U \sum_{i=1}^{g} p_i, \tag{16}$$

$$N_{E,\text{PIM-SM}} = \sum_{i=1}^{g} l_m(v_i) = \sum_{i=1}^{g} l_m(Up_i), \tag{17}$$

$$N_{E,all} = \sum_{i=1}^{g} (v_i + l_m(v_i)) = \sum_{i=1}^{g} (Up_i + l_m(Up_i)). \tag{18}$$

For IGMP, the number of entries equals the number of users, since no two hosts share the same network segment in our scenarios. For PIM-SM, the average number of entries equals the number of links for which the routers keep a state entry, and which is equal to the multicast tree size for each channel. Figure 10 illustrates the number of multicast entries versus the number of multicast groups obtained with both equations (17) and (18) and the time-domain simulation. Since the number of entries for a router represents the number of outbound interfaces belonging to the multicast tree, this result shows that there is a small difference in terms of multicast entries between popular and unpopular channels.

It is interesting to notice, that when we compare the previous result with Fig. 3 illustrating the bandwidth utilization for the same channel popularity model, we can observe that there is a large difference in terms of bandwidth between popular and unpopular channels. Therefore, using multicast for unpopular channels brings only a small gain in terms of bandwidth but has almost the same disadvantage for scalability as the popular channels. This finding is particularly essential when there are a large number of IPTV channels with very low popularity.

Finally, we look at the scalability issues in the P2P streaming, which, in our case, are represented by the utilization of the server as a last resort option for streaming content that cannot be served by peers. Toward this end, Fig. 11 shows the server load, in terms of average sent traffic. Similar to the previous figure describing the channel change delay, we intend this results as a qualitative one emphasizing a potential drawback of the P2P system. It is true that in our case the server utilization seems high, but this is due by the low ratio between users and channels we have selected (10:1) and the nature of the P2P schemes that do not cope very well with the high churn generated by the channel changes. Furthermore, the P2P algorithms can be improved to handle a specific system, while the scalability in IP multicast depends mainly on the network.

6. Conclusion

In this paper, we evaluated a hybrid IPTV system using IP multicast and P2P unicast. Our work comes in the context of an increasing number of service providers (telcos) moving into the IPTV market, but which, as recent papers suggest [6], use IP multicast to stream the TV channels to their users. This state of the facts combined with a possible increase of the number of TV channels in the near future, has raised the question of whether multicast alone is suitable to deliver a large number of channels, many having a very low popularity.

Our work compares analytically and through simulations the bandwidth utilization, quality and scalability aspects for a varying number of channels streamed by multicast and the rest by P2P unicast. The channels have popularity values around distinct levels dividing them into popular and unpopular. Our results demonstrate that while IP multicast is always the most efficient, for channels with very low popularity P2P can be an alternate choice because the amount of bandwidth that is saved by using more multicast groups is negligible. We emphasize the with a careful design, the quality impact can be small, when P2P is reserved for unpopular channels.

Appendix A. A flat model for channel changes

In this appendix, we describe a general model for channel popularity. The findings presented in our paper rely on two particular instances of this model.

Appendix A.1. Definitions

In addition to the definitions from 3.1.2, we introduce the following notion.

Definition 5. *The* change probability *for a pair of channel categories of indices i and j is the probability that a user will change from a channel belonging to category i to a channel belonging to category j. It is denoted by $q_{i,j}$.*

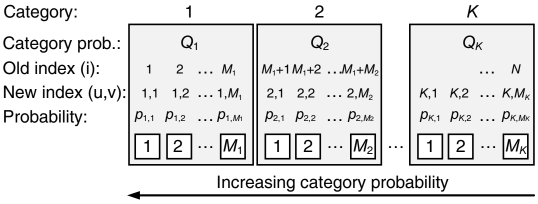

Figure A.12: All TV channels are grouped in K categories. Category (1) contains M_1 channels, category (2) contains M_2 channels and so on. All channels in the same category have the same probability of being selected: $p_{i,1} = \cdots = p_{i,M_i}$. The channel categories are ordered by their decreasing probability, category (1) has the highest category probability, category (K) has the lowest category probability: $Q_1 > \cdots > Q_K$.

Appendix A.2. The model

Our general model for the channel probability has K channel categories. We call the model *flat* because as defined previously, all the channels in the same category have the same probability. The model is: *during a channel change the next channel is determined by choosing a channel category with a given category probability; thereafter a channel in the selected category is chosen at random with a uniform distribution among the available channels.* The available channels are all channels from that category, eventually excluding the current channel from which the change is performed.

We denote by M_i be the number of TV channels belonging to category i. If the total number of channels is N, the following restriction applies:

$$\sum_{i=1}^{K} M_i = N. \tag{A.1}$$

The categories, illustrated in Fig. A.12, are ordered by their decreasing probabilities, i.e. $Q_1 > Q_2 > \cdots > Q_K$. For the sake of simplicity we substitute the channels numerical indices with pairs of two values representing the index of the category and the index of the channel within that category:

$$i \leftrightarrow (u, v). \tag{A.2}$$

Using this new channel index notation, the channel probability can be written as $p_{u,v}$, where u is the index of the channel category and v is the index of the channel within the category. This index variable change is illustrated as well in Fig. A.12, where the flat index i is mapped to the pair index (u, v). For example, we have: $p_1 = p_{1,1}$, $p_{M_1} = p_{1,M_1}$, $p_{M_1+1} = p_{2,1}$, etc. In addition, under the assumption the probabilities in the same category are equal, we have $p_{i,1} = \cdots = p_{i,M_i} = p_{(i)}$, where we introduced the new notation $p_{(i)}$ as representing the probability of any channel in category i.

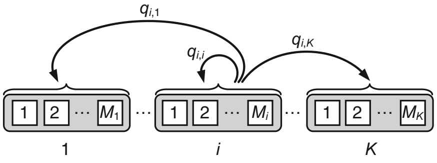

Figure A.13: The change probability $q_{i,j}$ is the probability of changing from any channel belonging to category i to any channel belonging to category j.

		From channel in the category					
		1	2	$\cdots$	i	$\cdots$	K
To channel in the category	1	$\frac{Q_1}{M_1-1}$	$\frac{Q_1}{M_1}$	$\cdots$	$\frac{Q_1}{M_1}$	$\cdots$	$\frac{Q_1}{M_1}$
	2	$\frac{Q_2}{M_2}$	$\frac{Q_2}{M_2-1}$	$\cdots$	$\frac{Q_2}{M_2}$	$\cdots$	$\frac{Q_2}{M_2}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
	i	$\frac{Q_i}{M_i}$	$\frac{Q_i}{M_i}$	$\cdots$	$\frac{Q_i}{M_i-1}$	$\cdots$	$\frac{Q_i}{M_i}$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
	K	$\frac{Q_K}{M_K}$	$\frac{Q_K}{M_K}$	$\cdots$	$\frac{Q_K}{M_K}$	$\cdots$	$\frac{Q_K}{M_K-1}$

Table A.2: The set of change probabilities

In order to determine the channels probability, we start from writing the changes probability. According to the definition, the change probability $q_{i,j}$ is the probability of changing the channel from a channel belonging to category i to a channel belonging to category j. The change probability is the same, regardless the channel, because in our flat model all channels in a category are assumed to have the same probability. Using the model definition stated above, the channel probability is the category probability divided to the number of available channels in that category:

$$q_{i,j} = \begin{cases} \dfrac{Q_j}{M_j} & \text{if } i \neq j \\ \dfrac{Q_j}{M_j - 1} & \text{if } i = j. \end{cases} \tag{A.3}$$

Figure A.13 illustrates the meaning of the change probabilities, starting from category i, while Table A.2 shows a set of typical values.

In order to calculate the channel probability, we can model the channel change pattern with a Markov chain (Fig. A.14). Therefore, the probability $p_{(i)}$ of a channel in category i depends on the probability of all the channels

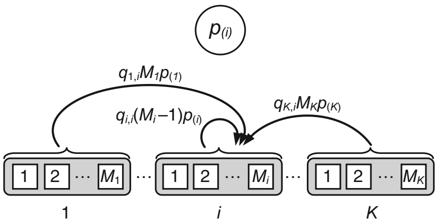

Figure A.14: Calculating the channel probability using a discrete Markov chain: the probability of a channel depends on all paths available to change to that channel. This example illustrates the probability $p_{(i)}$ of a channel from category i.

from which is possible to change. These probabilities are: $p_{(1)}$ for category 1, through $p_{(K)}$ for category K. There are M_1 such channels in category 1, M_2 in category 2, and so on. The only exception is category i, where only $M_i - 1$ channels are available (it is not possible to change between the same TV channel). The channel probabilities are multiplied by the change probabilities that represent the chance of changing between categories. Because the sum of all channel probabilities equals one, we can write the following system consisting of K unknowns and $K+1$ equations:

$$\begin{cases} p_{(i)} = q_{i,i}(M_i - 1)p_{(i)} + \sum_{\substack{j=1 \\ j \neq i}}^{K} q_{j,i} M_j p_{(j)} \\ \sum_{i=1}^{K} M_i p_{(i)} = 1. \end{cases} \tag{A.4}$$

It can be proved easily that out of the $K+1$ equations, only K are linearly independent. Replacing $q_{i,j}$ by their definition (A.3), we can rewrite the system as:

$$\begin{cases} p_{(i)} = \sum_{j=1}^{K} Q_i \frac{M_j}{M_i} p_{(j)} \\ \sum_{i=1}^{K} M_i p_{(i)} = 1. \end{cases} \tag{A.5}$$

The solution of the system is:

$$p_{(i)} = \sum_{j=1}^{K} Q_i \frac{M_j}{M_i} p_{(j)} = \frac{Q_i}{M_i} \sum_{j=1}^{K} M_j p_{(j)} = \frac{Q_i}{M_i}. \tag{A.6}$$

Appendix B. Channel probability, popularity and viewers

In this appendix, we include a set of proofs for the equations describing the relationship between channel probability, popularity and viewers, and which were used without a demonstration earlier in the paper.

Appendix B.1. Channel watch events and channel holding times

Definition 6. *The* channel watch event *is the action of one user watching continuously one TV channel. During a given observation period, we denote by n the total number of watch events. In an analog manner, we denote by n_i the number of watch events corresponding only to channel i, and by $n_{i,j}$ the number of watch events corresponding only to channel i being watched by user j.*

To facilitate the mathematical proof, here we extend the definition of the channel holding time, as presented in 3.1.1. We assume that every channel watch event has a finite holding time.

Definition 7. *Let $(\Omega_X, \mathcal{F}_X, P_X)$ be a probability space. The* channel holding time *is the continuous random variable $X : \Omega_X \to \mathbb{R}^+$, where X is finite.*

Corollary 1. *During an observation period with n channel watching events, then there exists a sequence of random variables X_k, with $1 \leq k \leq n$, where X_k represents the holding time of a channel being watched for the k-th time. The subset $X_k(i)$, with $1 \leq k \leq n_i$, represents the sequence of channel holding times for channel i. The subset $X_k(i,j)$, with $1 \leq k \leq n_{i,j}$, represents the sequence of channel holding times for channel i by user j.*

We assume the random variables from all these sequences, X_k, $X_k(i)$ and $X_k(i,j)$, are *independent* and *identically distributed.*

Corollary 2. *According to the law of large numbers, because the random variables representing the channel holding times are finite, independent and identically distributed, there exists a mean of the channel holding times, denoted by μ, and this mean is finite. That is, if $\overline{X_n} = \frac{1}{n}(X_1 + ... + X_n)$ is the average holding time for n watching events, we have:*

$$\overline{X_n} \to \mu \quad when \quad n \to \infty. \tag{B.1}$$

Similarly, we have $X_n(i) \to \mu$ and $X_n(i,j) \to \mu$, when $n \to \infty$.

We denote by $F_X : \mathbb{R}^+ \to [0,1]$ the *cumulative distribution function* of the continuous random variable X representing the channel holding time. In our paper, we use the cumulative distribution function F_X presented in [1].

Appendix B.2. Channel popularity

Definition 8. *The* popularity *of a channel i when watched by user j is the amount of time that channel is being watched by that user during an observation period, and it is denoted by $\mathscr{P}_{i,j}$.*

The popularity is a period of time.

Corollary 3. *If $X_k(i,j)$ are random variables representing the holding time for viewer i and channel j during an observation period, where $1 \leq k \leq n_{i,j}$, the popularity of channel i and viewer j is expressed as:*

$$\mathscr{P}_{i,j} = \sum_{k=1}^{n_{i,j}} X_k(i,j). \tag{B.2}$$

Definition 9. *The* popularity *of a channel i the amount of time that channel is being watched by any user during an observation period, and is denoted by $\mathscr{P}_i$. If we denote by U the number of users, we have:*

$$\mathscr{P}_i = \sum_{j=1}^{U} \mathscr{P}_{i,j}. \tag{B.3}$$

Corollary 4. *If $X_k(i)$ are random variables representing the holding time for any viewer and channel i during an observation period, where $1 \leq k \leq n_i$, the popularity of channel i is expressed as:*

$$\mathscr{P}_i = \sum_{k=1}^{n_i} X_k(i). \tag{B.4}$$

Definition 10. *For a given observation period, the* total popularity of all channels *is the sum of popularity of all channels during that observation period. The total popularity is denoted by $\mathscr{P}^*$ and if N is the number of channels, we have:*

$$\mathscr{P}^* = \sum_{i=1}^{N} \mathscr{P}_i. \tag{B.5}$$

Corollary 5. *If X_k are random variables representing the holding time for any viewer and any channel during an observation period, where $1 \leq k \leq n$, the total popularity is expressed as:*

$$\mathscr{P}^* = \sum_{k=1}^{n} X_k. \tag{B.6}$$

Theorem 1. *For a service provider with U subscribers that are always connected, the total popularity of all channels, $\mathscr{P}^*$, measured during an observation period of duration T satisfies the following equality:*

$$\mathscr{P}^* = U \cdot T. \tag{B.7}$$

Proof. *From the definition (B.5) of the total popularity we have:*

$$\mathscr{P}^* = \sum_{i=1}^{N} \mathscr{P}_i. \tag{B.8}$$

By replacing the popularity of channel i with its definition (B.2), we obtain:

$$\mathscr{P}^* = \sum_{i=1}^{N} \sum_{j=1}^{U} \mathscr{P}_{i,j} \tag{B.9}$$

where $\mathscr{P}_{i,j}$ is the popularity of channel i for user j.

Under the assumption that a user j stays connected (i.e. watching a channel) during the entire observation period T, from the definition of the channel popularity we obtain that the sum of the popularity of all channels for a user j is the observation period, T. Hence:

$$\sum_{i=1}^{N} \mathscr{P}_{i,j} = T \quad \forall j, 1 \leq j \leq U. \tag{B.10}$$

By replacing (B.10) in (B.9) we obtain:

$$\mathscr{P}^* = \sum_{j=1}^{U} T = U \cdot T. \tag{B.11}$$

Appendix B.3. Channel probability

Definition 11. *Let $(\Omega_Y, \mathcal{F}_Y, P_Y)$ be a probability space and N be the number of TV channels. Given the discrete random variable $Y : \Omega_Y \to \{1, ..., N\}$ representing a change to a TV channel, the* channel probability *is the probability mass function $p : \{1, ..., N\} \to [0, 1]$, where $p_i = \Pr(Y = j)$.*

Corollary 6. *Because Y is a discrete random variable we have:*

$$\sum_{i=1}^{N} p_i = 1. \tag{B.12}$$

Corollary 7. *If n_i is the number of watching events for channel i, and n is the total number of channel watching events during an observation period, we have:*

$$p_i = \lim_{n \to \infty} \frac{n_i}{n}. \tag{B.13}$$

Theorem 2 (Infinity limit theorem). *If a channel has a non-zero probability, when the observation period approaches infinity, the number of watching events for that channel, approaches infinity as well, and we have:*

$$\lim_{T \to \infty} n_i = \infty. \tag{B.14}$$

Proof. *Let n be the number of channel watching events for all channels and all users, and T be the observation period. The channel holding times are represented by the sequence of random variables X_k with $1 \leq k \leq n$. According to (B.6) and (B.7) we have:*

$$\sum_{i=1}^{n} X_i = U \cdot T. \tag{B.15}$$

When the observation period approaches infinity, $T \to \infty$. Because X_i is finite according to its definition, it results that n should approach infinity as well. Hence, $n \to \infty$.

Using (B.13), when $n \to \infty$, $p_i \neq 0$ if and only if $n_i \to \infty$.

Theorem 3 (Popularity theorem). *When the observation period, T, approaches infinity, the popularity of a channel i, $\mathscr{P}_i$ is proportional to the channel change probability, and we have:*

$$\lim_{T \to \infty} \frac{\mathscr{P}_i}{\mathscr{P}^*} = p_i. \tag{B.16}$$

Proof. *According to the previous theorem if $p_i \neq 0$, when $T \to \infty$ we have $n \to \infty$ and $n_i \to \infty$.*

By substituting $\mathscr{P}_i$ and $\mathscr{P}^*$ with (B.3) and (B.6), we obtain:

$$\lim_{T\to\infty} \frac{\mathscr{P}_i}{\mathscr{P}^*} = \lim_{\substack{n\to\infty \\ n_i\to\infty}} \frac{\sum_{k=1}^{n_i} X_k(i)}{\sum_{k=1}^{n} X_k}. \tag{B.17}$$

According to the definition of the mean of the sequences of random variables X_k and $X_k(i)$, we have:

$$\lim_{T\to\infty} \frac{\mathscr{P}_i}{\mathscr{P}^*} = \lim_{\substack{n\to\infty \\ n_i\to\infty}} \frac{n_i \cdot \mu}{n \cdot \mu} = \lim_{\substack{n\to\infty \\ n_i\to\infty}} \frac{n_i}{n} = p_i. \tag{B.18}$$

Appendix B.4. Channel viewers

Definition 12. *The* channel viewers *is the function $v : \{1, ..., N\} \to [0, \infty)$, where v_i represents the average of number of users viewing channel i during an observation period.*

Theorem 4. *During the observation period the number of channel viewers equals the channel probability divided to the observation period.*

Proof. *Let $T \in [0, \infty)$ be the observation period.*

In addition, let $v' : \{1, ..., N\} \times [0, \infty) \to \mathbb{N}$, where $v'_i(t)$ is the instantaneous number of users viewing a channel at the moment of time t.

According to the definition of v_i, the average number of viewers for channel i during the observation period is:

$$v_i = \frac{1}{T} \int_0^T v'_i(t)\,\mathrm{d}t. \tag{B.19}$$

On the other hand, the popularity of channel i is defined as the total amount of time i is being watched by any user. Hence:

$$\mathscr{P}_i = \int_0^T v'_i(t)\,\mathrm{d}t. \tag{B.20}$$

By substituting (B.20) into (B.19), we obtain:

$$v_i = \frac{\mathscr{P}_i}{T}. \tag{B.21}$$

Theorem 5 (Viewers theorem). *For a given observation period T, when the observation period approaches infinity, the average number of viewers for a channel is the product between the total number of users and the channel probability:*

$$v_i \to U \cdot p_i \quad when \quad T \to \infty. \tag{B.22}$$

Proof. *According to the previous theorem, we have:*

$$v_i = \frac{\mathscr{P}_i}{T}. \tag{B.23}$$

Using (B.7), the observation period can be written as:

$$T = \frac{\mathscr{P}^*}{U}. \tag{B.24}$$

Hence:

$$v_i = U \frac{\mathscr{P}_i}{\mathscr{P}^*}. \tag{B.25}$$

According to the popularity theorem (B.16), $\mathscr{P}_i/\mathscr{P}^* \to p_i$ when $T \to \infty$. Hence:

$$v_i \to U \cdot p_i. \tag{B.26}$$

Acknowledgment

This article has been partially supported by the Spanish Ministry of Science and Innovation through the CONPARTE project (TEC2007-67966-C03-03/TCM), and by the Madrid Community through the MEDIANET project (S2009-TIC1468).

[1] M. Cha, P. Rodriguez, J. Crowcroft, S. Moon, X. Amatriain, Watching television over an IP network, in: Proceedings of the 8th ACM SIGCOMM conference on Internet measurement conference, ACM New York, NY, USA, 2008, pp. 71–84.

[2] T. Wong, R. Katz, An analysis of multicast forwarding state scalability, in: Proceedings of the 8th IEEE International Conference on Network Protocols (ICNP 2000), 2000, pp. 105–115.

[3] D. Thaler, M. Handley, On the aggregatability of multicast forwarding state, in: Proceedings of IEEE INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies., Vol. 3, 2000, pp. 1654–1663.

[4] A. Fei, J. Cui, M. Gerla, M. Faloutsos, Aggregated Multicast: an approach to reduce multicast state, in: Proceedings of IEEE Global Telecommunications Conference 2001 (GLOBECOM '01), Vol. 3, 2001, pp. 1595–1599.

[5] B. Zhang, H. Mouftah, Forwarding state scalability for multicast provisioning in IP networks, IEEE Communications Magazine 41 (6) (2003) 46–51.

[6] M. Cha, P. Rodriguez, S. Moon, J. Crowcroft, On next-generation telco-managed P2P TV architectures, in: International workshop on Peer-To-Peer Systems (IPTPS), 2008.

[7] A. Bikfalvi, J. García-Reinoso, I. Vidal, F. Valera, A peer-to-peer IPTV service architecture for the IP multimedia subsystem, International Journal of Communication Systems.

[8] M. Castro, P. Druschel, A. Kermarrec, A. Nandi, A. Rowstron, A. Singh, SplitStream: high-bandwidth multicast in cooperative environments, in: Proceedings of the nineteenth ACM symposium on Operating systems principles, ACM, 2003, pp. 289–313.

[9] X. Zhang, J. Liu, B. Li, T. Yum, CoolStreaming/DONet: A data-driven overlay network for efficient live media streaming, in: Proceedings of IEEE INFOCOM, Vol. 3, 2005, pp. 13–17.

[10] V. Pai, K. Kumar, K. Tamilmani, V. Sambamurthy, A. Mohr, Chainsaw: Eliminating trees from overlay multicast, Lecture notes in computer science 3640 (2005) 127–140.

[11] N. Magharei, R. Rejaie, Prime: Peer-to-peer receiver-driven mesh-based streaming, IEEE/ACM Transactions on Networking (TON) 17 (4) (2009) 1052–1065.

[12] D. Wu, C. Liang, Y. Liu, K. Ross, View-upload decoupling: A redesign of multi-channel p2p video systems, in: Proc. of IEEE INFOCOM, Citeseer, 2009, pp. 2726–2730.

[13] ETSI-TISPAN, IPTV Architecture; IPTV functions supported by the IMS subsystem, TS 182.027, ETSI (2008).

[14] ETSI-TISPAN, IPTV Architecture; Dedicated subsystem for IPTV functions, TS 182.028, ETSI (2008).

[15] ETSI-TISPAN, Peer-to-peer for content delivery for IPTV services: analysis of mechanisms and NGN impacts, TR 182.010, ETSI (2009).

[16] J. Chuang, M. Sirbu, Pricing multicast communication: A cost-based approach, Telecommunication Systems 17 (3) (2001) 281–297.

[17] T. Henderson, S. Bhatti, Protocol-independent multicast pricing, in: Proceedings of the 10th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV), Citeseer, 2000, pp. 11–17.

[18] N. Kausar, B. Briscoe, J. Crowcroft, A charging model for sessions on the internet, in: Proceedings of ISCC, Springer, 1999, pp. 32–38.

[19] G. Phillips, S. Shenker, H. Tangmunarunkit, Scaling of multicast trees: comments on the chuang-sirbu scaling law, in: Proceedings of ACM SIGCOMM, ACM, New York, NY, USA, 1999, pp. 41–51.

[20] A. Medina, A. Lakhina, I. Matta, J. Byers, BRITE: An approach to universal topology generation, in: Proceedings of Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, Vol. 1, 2001, pp. 346–353.

[21] A. Ziviani, B. Wolfinger, J. De Rezende, O. Duarte, S. Fdida, Joint adoption of QoS schemes for MPEG streams, Multimedia Tools and Applications 26 (1) (2005) 59–80.

[22] P. Radoslavov, D. Estrin, R. Govindan, Exploiting the bandwidth-memory tradeoff in multicast state aggregation, USC-Dept. of CS, Tech. Rep.