E2E Blocking Probability of IPTV and P2PTV*

Yue Lu[1], Fernando Kuipers[1], Milena Janic[2], and Piet Van Mieghem[1]

[1] Delft University of Technology
{Y.Lu,F.A.Kuipers,P.F.A.VanMieghem}@tudelft.nl
[2] TNO Information and Communication Technology
Milena.Janic@tno.nl

Abstract. Increased Internet speeds together with new possibilities for tailor-made television services have spurred the interest in providing television via the Internet. Several television services are readily available and both IP-layer and application-layer (P2P) technologies are used. When disregarding commercial influences, customers will choose for IPTV or P2PTV based on the Quality of Experience (QoE). In this paper, we investigate one important QoE measure, namely the content blocking probability.

1 Introduction

The demand for digital television via a dedicated network or the Internet is growing fast. A digital television service can be provided in two different ways: either utilizing a mechanism on the IP layer (IPTV) or on the application layer (P2PTV).

IPTV is implemented in a dedicated network, which connects the end-users' television set-top boxes through Digital Subscriber Line Access Multiplexers (DSLAMs). The television programs are collected at a data centre and distributed towards the DSLAMs along an IP-layer multicast tree. A DSLAM replicates the received signal and sends it to the end users.

P2PTV can either make use of application-layer multicast trees or chunk-based P2P technology. The majority of existing P2PTV applications are chunk-based. They make use of the BitTorrent technology[1] and are currently free of charge.

Disregarding economic incentives, telecom operators will continue to deploy and extend IPTV in the dedicated network if its quality surpasses the quality of P2PTV. In contrast, if connecting end-users' set-top boxes in a P2P way can achieve better Quality of Experience (QoE), the operator should consider to deploy P2PTV technology instead. Hence, defining, computing and comparing the quality of these two television architectures is deemed essential. In this paper, we focus on the content blocking probability as our QoE measure of interest, and ask the following questions:

* This work has been partially supported by the European Union CONTENT NoE (FP6-IST-038423) and the triangular cooperation KPN-TNO-TUDelft.

[1] http://www.bittorrent.com/

A. Das et al. (Eds.): NETWORKING 2008, LNCS 4982, pp. 445–456, 2008.

- *What factors contribute to the blocking of IPTV and P2PTV, and how?*
- *Which technology incurs the lowest end-to-end blocking?*
- *In what situation can P2PTV offer users better QoE than IPTV?*

In order to answer these questions we have developed blocking models for IPTV and P2PTV, which we subsequently apply to two case studies. Our case study of IPTV is based on measurement data of an existing Dutch IPTV network. Our P2PTV case study is based on measurements of SopCast[2].

The remainder of this paper is organized as follows. Section 2 presents related work. In Section 3 the blocking of IPTV will be defined and computed, and applied to our case study. Section 4 adopts a same methodology for P2PTV. Section 5 compares the two architectures. Finally, we conclude in Section 6.

2 Related Work

Work on IPTV mainly focuses on designing protocols (e.g., [1]) and implementations by broadband network operators, router manufacturers, and television providers. For the performance of IPTV, video quality and packet loss were analyzed, but the end-to-end blocking of requests has not been computed from a users' point of view. Karvo *et al.* [2] set up a queuing model to calculate the end-to-end multicast blocking, but their work is not aimed at IPTV and is not based on realistic data. In this paper, we will define and compute the IPTV blocking probability based on realistic data.

The majority of P2PTV implementations (e.g., SopCast, PPLive [3], and CoolStreaming [4]) use chunk-based BitTorrent technology. Consequently, in this paper, we only consider chunk-based P2PTV. Related work on BitTorrent-like P2P blocking and performance [5], [6], [7], only targets the P2P file sharing systems. Empirical blocking results were provided [3], [4], but a definition of and a formula to compute the end-to-end blocking in P2PTV architectures are still missing. In this paper, we provide a first step in this direction.

3 IPTV Blocking Probability

3.1 Model with Assumptions

Figure 1 illustrates a possible realization of an IPTV architecture. As this figure indicates, end users are connected to DSLAMs which, on their turn, are connected to an edge router. We only model the blocking of a single television channel over a single DSLAM. We assume that C represents the capacity of a link from a particular DSLAM to the edge router. The maximum number of channels that can be transmitted simultaneously over a link with capacity C is $m = \left\lfloor \frac{C}{C_o} \right\rfloor$, where C_o is the capacity of one television channel.

We further assume that there are K available television channels that can be viewed. If a channel is being viewed, we say it is "on". Not all the channels are

[2] http://www.sopcast.org/

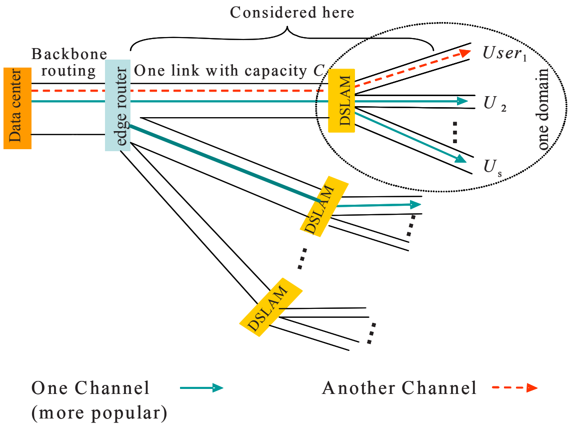

Fig. 1. A possible realization of an IPTV architecture

equally popular. In our model we assume that the popularity distribution α_i is given[3].

The arrival and departure process of TV users is assumed to be Poissonian. In reality, the request arrival rate $\lambda(t)$ is non-stationary. If a popular TV program starts, the arrival rate will be high, while once the advertisements come, the average arrival rate decreases. However, we consider the average case over a specific time period. Based on a TV-users market survey[4], the arrival process is approximately Poissonian in the period of 16:00 to 22:00. According to the measurement study of X. Hei *et al.* [3, Fig 13.], the CDF of TV viewing time follows an exponential distribution, which justifies our assumption that the TV user departure process is Poissonian.

3.2 Computation of the Blocking Probability

If a user, requesting a particular television channel i, cannot receive its content, we consider channel i to be blocked. Two main causes of blocking in IPTV are:

I *Limited processing capability of a DSLAM.* If many users simultaneously desire content from the same DSLAM, blocking might occur. The blocking probability of this case is denoted as B_{proc}.

II *Insufficient available capacity from the DSLAM to the edge router.* If a maximum number of m channels is transmitted to the DSLAM, a new user requesting a new channel i (not among the transmitted m channels) will find his request blocked. Since the more popular channels have a higher probability to be present among the already transmitted m channels, $B_{link}(i)$ depends on which television channel i is requested.

[3] We have used the TV channel popularity distribution in The Netherlands, which is available at http://www.kijkonderzoek.nl/ (in Dutch).

[4] Market Report of TV customers in The Netherlands, http://www.kijkonderzoek.nl/

Since blocking II can only occur if blocking I did not take place, for IPTV the end-to-end blocking probability $B(i)$ for channel i is

$$B(i) = B_{proc} + (1 - B_{proc})B_{link}(i) \tag{1}$$

B_{proc}: Assuming Poisson arrivals and departures, as explained in Section 3.1, we model the DSLAM as an $M/M/n/n/s$ queue, where s represents the number of users accessing the DSLAM and n the number of replications that the DSLAM can handle. In our model $\rho_{DSLAM} = \frac{\lambda_{DSLAM}}{\mu_{DSLAM}}$ is the same for each user, where λ_{DSLAM} represents the rate at which a user requests a TV service, and μ_{DSLAM} the rate at which the user turns his TV off. According to [8, pp. 512], we have

$$B_{proc} = \frac{\frac{(s-1)!}{(s-1-n)!n!}\rho_{DSLAM}^{n}}{\sum\limits_{h=0}^{n}\frac{(s-1)!}{(s-1-h)!h!}\rho_{DSLAM}^{h}} \tag{2}$$

$B_{link}(i)$: To compute $B_{link}(i)$ we introduce two new probability functions $P(i)$ and $B_{Engset}(i)$. $P(i)$ is the probability that channel i is "on" and $B_{Engset}(i)$ is the probability that the link from the DSLAM to the edge router is consumed by m channels other than the requested channel i. Our computation of $B_{Engset}(i)$ is given in the Appendix, and

$$B_{link}(i) = (1 - P(i))B_{Engset}(i) \tag{3}$$

$P(i)$: Disregarding possible blocking[5], channel i can be modeled as an $M/M/\infty$ queue (in steady state), with infinite positions available in the queue to store all requests for channel i. Hence, the probability that channel i is "on", in the condition that no requests are blocked, is equal to the probability that the $M/M/\infty$ queue [8, pp. 281] is not empty: $\Pr[N_s > 0]_i = 1 - \exp(-\rho_i)$, where $\rho_i = \lambda_i/u_i = \alpha_i\lambda/u_i$. λ_i is the users' arrival rate in channel i, and u_i is the number of users leaving from channel i per second. λ is the users' arrival rate in the IPTV system, which includes both the rate at which users switch on their television as well as the channel switching rate. α_i represents the popularity of channel i , where $0 \leq \alpha_i \leq 1$.

Channel i can be either "off" or "on". We therefore resort to a two-state Markov chain illustrated in Figure 2 to analyze its steady state. In Figure 2, "0" represents the state that channel i is "off" and "1" represents the state that channel i is "on". The probability to be in the state "off" is $1 - P(i)$ and consequently the probability to be in the state "on" is $P(i)$. To change from the "off" state to the "on" state, requests should exist for channel i (this event has probability $\Pr[N_s > 0]_i$) and these requests may not be blocked (this event has probability equal to $1 - B_{Engset}(i)$). The process remains in the "off" state if there are no requests in the queue for channel i ($\Pr[N_s = 0]_i$) or if the requests

[5] If a user's request enters when channel i is "on", the request will be grouped into multicast. However, if a user's request enters when channel i is "off", it has to open a new channel, which might not be possible in case of insufficient available capacity.

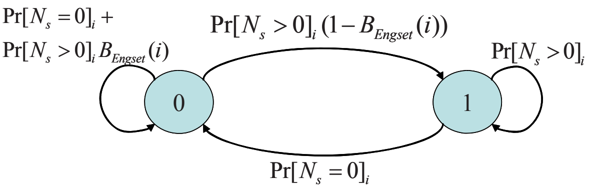
$\Pr[N_s = 0]_i +$
$\Pr[N_s > 0]_i B_{Engset}(i)$
$\Pr[N_s > 0]_i (1 - B_{Engset}(i))$
$\Pr[N_s > 0]_i$
0
1
$\Pr[N_s = 0]_i$

Fig. 2. Two-state Markov chain representing the status of channel i

are blocked (this event has probability $\Pr[N_s > 0]_i B_{Engset}(i)$). To change from the "on" state to the "off" state, all requests for channel i are served until the queue becomes empty (with probability $\Pr[N_s = 0]_i$). To remain in the "on" state, the queue may not be empty ($\Pr[N_s > 0]_i$).

According to the steady state probability of a two-state Markov chain [8, pp.176], the probability to be in the state "on" is

$$\begin{aligned} P(i) &= \frac{\Pr[N_s > 0]_i (1 - B_{Engset}(i))}{\Pr[N_s > 0]_i (1 - B_{Engset}(i)) + \Pr[N_s = 0]_i} \\ &= 1 - \frac{1}{\exp(\rho_i) - B_{Engset}(i) \exp(\rho_i) + B_{Engset}(i)} \end{aligned} \tag{4}$$

By substituting formulae (3), (2), (4), and (10) into formula (1), we obtain the end-to-end blocking probability $B(i)$ of IPTV.

3.3 Case Study

We focus on a Dutch IPTV network for which we have obtained the following parameter values:

The channel popularity distribution α_i for the channels ($1 \leq i \leq 23$) is obtained from a market survey[6]. We assume that the remaining $K - 23$ channels uniformly share a popularity of 5.9%. In reality, the channel popularity distribution changes more smoothly, but here we just classify channels as popular or unpopular.

The link capacity C is typically 155Mb/s in nowadays DSL systems and one MPEG4 coded TV channel consumes 2.5Mb/s. Hence, approximately $m = 60$ different TV channels can be transmitted simultaneously to the DSLAM. We further assume that the IPTV system consists of $500,000$ subscribers, who are uniformly distributed over 1200 DSLAMs and 41% of the subscribers are active in rush hours. This results in an average number of $s = 171$ active users connecting to a single DSLAM. However, only $n = 120$ channel replications can be handled by the DSLAM simultaneously. We define $Q_{IPTV} = \lambda / u_i$.

Figure 3 plots the IPTV blocking probability $B(i)$ as a function of the channel index i, with $m = 60$ and $n = 120$.

The less popular channels (higher channel index) have a considerably higher probability to be blocked than the more popular channels. In addition, we can

[6] Channel 1 has a popularity of 15.1% and channel 23 of 0.2%, covering in total 94.1%.

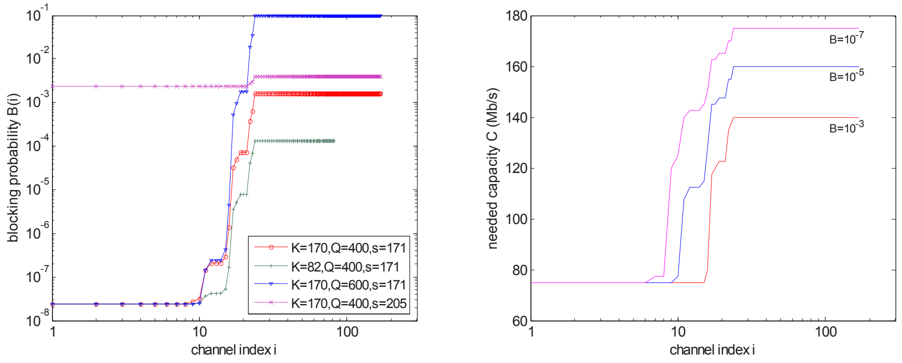

Fig. 3. (Left) IPTV end-to-end blocking $B(i)$ with $m = 60$, $n = 120$, (right) The required capacity C for meet different blocking requirements, with $K = 170$, $s = 171$, and $Q_{IPTV} = 400$

observe that the user's request is less likely blocked if there are less available TV channels (smaller K), if there are less users per DSLAM (smaller s), or if users leave more frequently (smaller Q_{IPTV}).

Figure 3 also plots the capacity C that is required to assure that the IPTV blocking $B(i)$, as a function of channel index i, does not exceed a certain level.

4 P2PTV Blocking Probability

4.1 Model with Assumptions

Figure 4 presents our model of chunk-based P2PTV applications. We focus on the blocking of a single television channel i for a particular user U. User U is viewing a television channel i. The playback rate of the channel i is v kbits/s. In the worst case, user U downloads the next second of content when displaying the current second of content. One second of television content is divided into R chunks. Hence, one chunk contains v/R kbits.

First, based on a peer list that user U obtains, a fixed amount of peers (referred to as partners) are randomly chosen to form a partner group $\mathcal{P}$ for that channel i. In this partner group, partners exchange information about which chunks are available at which partner. Based on this chunk-availability information, a user U chooses $M \leq |\mathcal{P}|$ peers from whom he downloads chunks. Those peers are called parents of user U. A parent supplies at least 1 chunk to user U. Similarly, the peers downloading the content from user U will be referred to as children of user U. A parent has Y children at the same time. We denote by N_i the number of available peers in the entire peer list corresponding to channel i and by bw_{up} the upload bandwidth of a parent. Simultaneously while downloading the chunks, user U is displaying the content stored in his buffer[7]. Here, we consider the

[7] There is a zero probability of buffer overflow, because the buffer window is sliding and the outdated chunks will be deleted automatically.

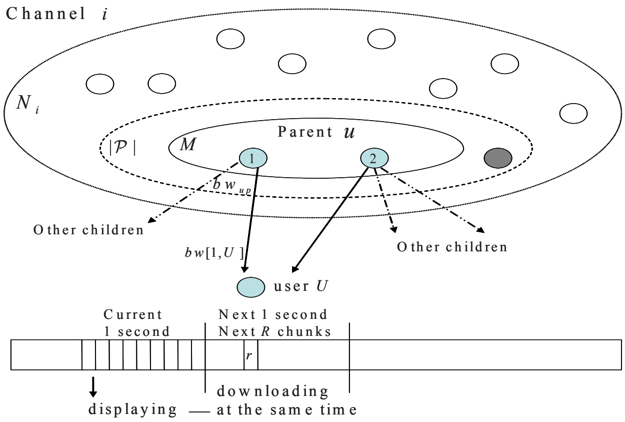

Fig. 4. P2PTV model when user U has 2 parents

worst case of P2PTV blocking where none of these R chunks were previously downloaded and the download of these R chunks has to be finished within the coming second.

4.2 Computation of the Blocking Probability

In IPTV, the limited available bandwidth in an infrastructure network is the main cause of IPTV blocking, while the P2PTV network built on top of the Internet has, in theory, more available capacity contributed by peers. Hence, the P2PTV blocking is mainly caused by content unavailability, peer selection randomness and peer dynamics. Contrary to IPTV, where blocking only occurs when requesting a television channel, blocking in P2PTV can also occur while watching television. We define the blocking probability in P2PTV as the probability that user U cannot successfully download the next R chunks before having displayed the current R chunks.

If the required R chunks cannot all be found in the partner group, which happens with probability $b_{chunk}(i)$, blocking will occur. Even when all the chunks can be found at the $|\mathcal{P}|$ partners, a chosen parent may upload his chunks too slowly. This occurs with probability $b_{time}(i)$. Finally, if a parent leaves during uploading, blocking will occur. The probability of this occurrence is $b_{dyn}(i)$. As a reasonable approximation, we assume that the different events are independent. The end-to-end blocking $b(i)$ of channel i in P2PTV can be presented as:

$$b(i) = b_{chunk}(i) + (1 - b_{chunk}(i))(b_{time}(i) + (1 - b_{time}(i))b_{dyn}(i)) \qquad (5)$$

In the sequel, we determine $b_{chunk}(i)$, $b_{time}(i)$ and $b_{dyn}(i)$.

$b_{chunk}(i)$: Choosing parents among partners is based on the chunk-availability information. Blocking arises when user U cannot find all R chunks from the *randomly chosen* partners. We let $1 - b_{chunk}(i)$ denote the probability that user U can find all R chunks in his partner group.

$$1 - b_{chunk}(i) = \mathop{\mathit{\Pi}}_{r=1}^{R} (1 - B_i(r)) \tag{6}$$

where $B_i(r) = [1 - \pi_i(r)]^{|\mathcal{P}|}$ represents the probability that user U cannot find chunk r successfully among $|\mathcal{P}|$ randomly chosen (and hence considered independent) partners. We use $\pi_i(r)$ to represent the probability that a peer is storing chunk r.

$b_{time}(i)$: Blocking arises when at least one parent u cannot upload his chunks to user U within 1 second due to insufficient bandwidth. The upload bandwidth that a parent u has available for user U is $bw[u, U]$, which has a distribution function $\Pr[bw[u, U] \leq x]$. Given that user U requests X_u chunks of v/R kbits from parent u, the required upload bandwidth is $bw_{X_u} = \frac{X_u v}{R}$. If the available bandwidth is smaller than the bandwidth required for uploading the requested chunks, blocking will occur.

In the following, we will make use of the theory of partitions [9]. A partition of a positive integer R is a collection of positive integers whose sum is R. If there are M terms in the sum, then R is said to be partitioned into M parts. We let $p_M(R)$ represent the number of partitions of R into parts not exceeding M, which has the generating function [9]: $\prod_{j=1}^{M} \left(1 - x^j\right)^{-1} = \sum_{R=0}^{\infty} p_M(R) x^R$. From the generating function the following recursion is obtained:

$$p_M(R) = p_{M-1}(R) + p_M(R - M)$$

with $p_M(0) = p_1(R) = 1$ and $p_M(j) = 0$ for $j < 0$.

In our P2PTV model, we need to find the number of partitions of R chunks over exactly M parents, which equals $p_M(R) - p_{M-1}(R) = p_M(R - M)$. The occurrence probability of having M parents has a density function $\Pr[M = k]$, however for partitions with a same number of M we assume that the occurrence probability of each possible partition is the same.

Blocking occurs only when the available bandwidth $bw[u, U]$ is smaller than the required bandwidth. Assuming independence,

$$b_{time}(i) = \sum_{k=1}^{|\mathcal{P}|} \frac{\Pr[M = k]}{p_k(R - k)} \sum_{j=1}^{p_k(R-k)} \left(1 - \prod_{u=1}^{k} (1 - \Pr[bw[u, U] < \frac{X_u(j) v}{R}])\right) \tag{7}$$

where the index j refers to one of the $p_M(R - M)$ possible partitions, and $X_u(j)$ refers to how many chunks user U requests from parent u. For instance, we have $p_M(R - M) = 2$ possible partitions of $R = 4$ chunks over $M = 2$ parents (u_1 and u_2), namely partition $j = 1$ consisting of $X_{u_1}(1) = 1$ and $X_{u_2}(1) = 3$ and partition $j = 2$ consisting of $X_{u_1}(2) = 2$ and $X_{u_2}(2) = 2$.

$b_{dyn}(i)$: We let $b_{dyn}(i)$ represent the probability that at least one parent leaves during his uploading period, which causes blocking.

In our P2PTV model, the peer departure process $Z(t)$ is considered Poissonian, which was explained in Section 3.1. The rate at which peers leave from channel i is denoted by θ_i. The departure rate of one peer is $\frac{\theta_i}{N_i}$, where N_i stands for the total number of available peers in channel i.

We denote by $P_u(j)$ the probability that parent u with $X_u(j)$ chunks (in partition j) leaves during uploading. $P_u(j) = \Pr[Z(t+t_0)-Z(t_0) = 1] = (\frac{\theta_i}{N_i}t)\exp(-\frac{\theta_i}{N_i}t)$, where $t = \min\{1, \frac{X_u(j)v/R}{bw[u,U]}\}$.

Hence, we can express $b_{dyn}(i)$ as:

$$b_{dyn}(i) = \sum_{k=1}^{|\mathcal{P}|} \frac{\Pr[M=k]}{p_k(R-k)} \sum_{j=1}^{p_k(R-k)} \left(1 - \prod_{u=1}^{k}(1-P_u(j))\right) \tag{8}$$

Substituting formulae (6), (7) and (8) into formula (5), we obtain the end-to-end blocking $b(i)$ of P2PTV.

4.3 Case Study

In this P2PTV case study, we analyze a chunk-based P2PTV application called SopCast. We have developed scripts and set-up a distributed measurement testbed via Planetlab. We installed SopCast and TcpDump on each (of in total 80) Planetlab nodes and performed various measurements, among which we believe three are unique, namely: (1) measurements on the topology - we have the parent distribution $\Pr[M = k]$ for popular as well as unpopular channels, (2) measurements on the bandwidth distribution $\Pr[bw[u, U] \leq k]$, and (3) the quality of experience (e.g., blocking) at the end user. We have used our measurement data as input to our model and then computed the blocking probability according to the formulae above. We also compared our analytical results with our measurement results on blocking probability.

Our experiments with SopCast indicated that the upload bandwidth of a parent bw_{up} is almost uniformly shared by his Y children in a 1 second period of time. Hence, for user U, we can define $bw[u, U] = \frac{bw_{up}}{Y}$. Since each parent can have a different upload rate and a different number of children at a given time, the value of $bw[u, U]$ may differ per parent. Later, we will use our empirically obtained $bw[u, U]$ distribution function into our model.

To allow for a fair comparison with the IPTV case, we will choose for our P2PTV case study similar values, where possible. The channel popularity distribution α_i is the same as for IPTV, and the user behavior is the same as for IPTV. We let $N_i = N\alpha_i$, where N is the number of concurrent active peers over all channels. N can be $1200s\alpha_i$ (the same size for IPTV) or $2,200,000\alpha_i$ (the current peak size of a P2PTV system). For other values, our case study is based on SopCast. Our measurements were run for 5 times on 5 different days and we have used the obtained results for our computations. Based on our measurement data, a minimum of 1 and a maximum of 3 partners are chosen as parents for both popular channels and unpopular channels in 1 second. The distributions of the number of parents $\Pr[M = k]$ differ for popular and unpopular channels.

Our measurement study has revealed that the channel under study has a playback rate $v = 300$kbits/s and one chunk size is 13kbytes. Hence, R is around 3 chunks/second. In our computation of the blocking probability $b_{chunk}(i)$, we assume $\pi_i(r)$ equals to 91% ([10]) for each channel. We set $v/R = 104$kbits, to

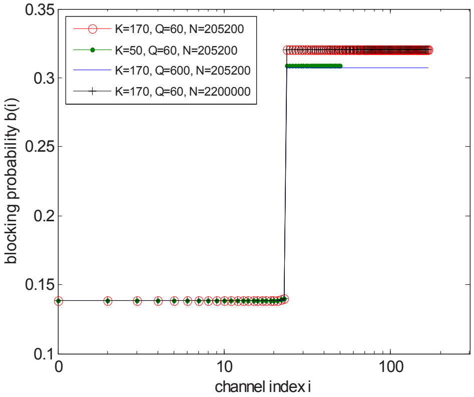

Fig. 5. P2PTV end-to-end blocking $b(i)$ with different values for the number of channels K, the number of active users in the system N, and $Q_{P2PTV} = \lambda/\theta_i$

retrieve the blocking probability $b_{time}(i)$. We assume that the download bandwidth of an end user is large enough to download $R = 3$ chunks in 1 second. In order to obtain the blocking probability $b_{dyn}(i)$, we define $Q_{P2PTV} = \frac{\lambda}{\theta_{i.}}$ and further assume[8] $\theta_i = 0.00038N/Q_{P2PTV}$, with θ_i fixed for each TV channel.

Given the above values we can get $b_{chunk}(i)$, $b_{time}(i)$ and $b_{dyn}(i)$, and subsequently compute the P2PTV blocking $b(i)$ from formula (5). Figure 5 plots the P2PTV blocking probability $b(i)$ as a function of the channel index i. Figure 5 illustrates that, the 23 most popular channels have a much smaller probability to be blocked than the remaining unpopular channels. The blocking probability of all channels is quite high, because what we have modeled is the worst case without any positive effects due to buffering. In order to validate our model, we measured the blocking at end users. We monitored the download speed at each user using 1 second as unit and we compared the download speed with the playback rate. The fraction of time the download speed is smaller than the constant playback rate gives the blocking probability defined in our model. When averaging the blocking probability over all channels, we find a blocking probability of 22% when excluding the buffer effect, which fits our mathematical results.

The P2PTV end-to-end blocking $b(i)$ shown in Figure 5 is mainly contributed by $b_{time}(i)$. Hence, the limited upload bandwidth of an end user's parent distributed to him mainly causes the large blocking. This clearly indicates the importance of the parent selection policy.

We can also observe that a user will face less blocking if the number of available channels K is smaller, or if users leave infrequently (high Q_{P2PTV}), while changing the amount of users did not significantly affect the blocking probability.

[8] As observed in [4, Fig.14], when there are on average $N = 1750$ active peers, the maximum arrival rate of peers is $\lambda = 0.67$ peers/s and $\frac{\lambda}{N} = 0.00038$.

5 Blocking Comparison

In this section, we compare IPTV and P2PTV based on the computations presented in Sections 3 and 4. We assess which technology (IPTV or P2PTV) incurs the lowest end-to-end blocking. Currently, if the operator decides to transmit television content to its residential users using P2PTV instead of IPTV, all channels will likely face more blocking than before. However, we can predict that when the number of end users increases, this will change. In a P2PTV system, the number of users has little effect on the blocking, while in an IPTV system the blocking would largely increase if the processing capability of the equipment (like DSLAM) cannot scale accordingly. When the total amount of users increases and the amount of DSLAMs and their processing capability remains the same, there will be a point at which the P2PTV system will start to outperform the IPTV system for popular channels. In our Dutch case study, this cross-over point is around $297,600$ users ($s = 248$ per DSLAM). Provided the appropriate data is available, our formulas allow for similar computations for different cases.

6 Conclusions

Two digital television technologies, namely IPTV and P2PTV, have been discussed in this paper. We have provided new blocking models for both technologies and have defined and derived formulae to calculate the end-to-end blocking probability based on our blocking models. Different contributions of the blocking have been analyzed. Moreover, the blocking of IPTV and the blocking of P2PTV have been compared under fair and realistic conditions. We found that when the amount of users increases, there will be a point at which the blocking in P2PTV will be less than in IPTV, unless the IPTV network is extended accordingly. Finally, our results cannot only be used to analyze the blocking in existing systems, but also to dimension them.

References

1. Lim, S.Y., Soek, J.M., Lee, H.K.: A path control architecture for receiving various multimedia contents. In: Proc. of ICACT 2006 (February 2006)
2. Karvo, J., Virtamo, J., Aalto, S., Martikainen, O.: Blocking of dynamic multicast connections in a single link. In: Proc. of IEEE BROADNETS (1998)
3. Hei, X., Liang, C., Liang, J., Liu, Y., Ross, K.W.: A Measurement Study of a large-Scale P2P IPTV System. IEEE Transactions on Multimedia 9(8), 1672–1687 (2007)
4. Zhang, X., Liu, J., Li, B., Yum, T.P.: CoolStreaming/DONet: A Data-driven Overlay Network for Peer-to-Peer Live Media streaming. In: Proc. of IEEE INFOCOM, March 2005, vol. 3, pp. 2102–2111 (2005)
5. Meo, M., Milan, F.: QoS-aware Content Management in P2P Networks. In: Proc. of HOT-P2P 2004 (2004)

6. Susitaival, R., Aalto, S., Virtamo, J.: Analyzing the dynamics and resource usage of P2P file sharing by a spatio-temporal model. In: Proc. of P2P-HPCS 2006 (2006)
7. Guo, L., Chen, S., Xiao, Z., Tan, E., Ding, X., Zhang, X.: A Performance Study of BitTorrent-like Peer-to-Peer Systems. IEEE J. Selected. Area in Comm. 25 (January 2007)
8. Van Mieghem, P.: Performance Analysis of communications Networks and Systems. Cambridge University Press, Cambridge (2006)
9. Rademacher, H.: Topics in Analytic Number Theory. Springer, Berlin (1973)
10. Vassilakis, C., Laoutaris, N., Stavrakakis, I.: On the Benefits of Synchronized Playout in Peer-to-Peer Streaming. In: Proc. of CoNEXT (September 2006)

A $B_{Engset}(i)$

The IPTV system with K available channels and m admitted channels can be modeled by an $M/M/m/m/K$ queuing system. In this Engset queue with different channel i arrival rate λ_i and channel i leaving rate μ_i, each channel request can arrive in random order. Karvo *et al.* [2] introduced a binomial probability generation function to deduce the probability $\pi_j^{(i)}$ (j positions are occupied by j channels other than channel i in an infinite capacity system) for this system:

$$\varphi_i(z) = \sum_{j=0}^{\infty} \pi_j^{(i)} z^j = \prod_{k=1}^{K} \frac{q_k + p_k z}{q_i + p_i z} \tag{9}$$

in which, $q_i = 1 - p_i$ and $p_i = 1 - e^{-\frac{\lambda_i}{\mu_i}}$. $B_{Engset}(i) = \pi_m^{(i)} / \sum_{j=0}^{m} \pi_j^{(i)}$. According to $\pi_j^{(i)} = \frac{1}{j!} \frac{d^j \varphi_i(z)}{dz^j}|_{z=0}$([8, pp.18]) and Eq. (9),

$$B_{Engset}(i) = \frac{\frac{1}{m!} \frac{d^m \left[\prod_{k=1}^{K} \frac{q_k+p_k z}{q_i+p_i z}\right]}{dz^m}|_{z=0}}{\sum_{j=0}^{m} \frac{1}{j!} \frac{d^j \left[\prod_{k=1}^{K} \frac{q_k+p_k z}{q_i+p_i z}\right]}{dz^j}|_{z=0}} \tag{10}$$