Understanding P2P-TV Systems Through Real Measurements

Delia Ciullo, Marco Mellia, Michela Meo, Emilio Leonardi
Politecnico di Torino, Italy, Email: {last name}@tlc.polito.it

***Abstract*—In this paper, we consider two popular peer-to-peer TV (P2P-TV) systems: PPLive, one of the today most widely used P2P-TV systems, and Joost, a promising new generation application of which no previous measurement study has been considered. Besides the traditional measurements like the amount of generated traffic for signaling and data transmission, the novel contribution of the paper consists in investigating the content distribution mechanisms. In particular, we evaluate the characteristics of both data distribution and signaling process for the overlay network discovery and maintenance. By considering two or more clients in the same sub-network, we observe the capability of the system to exploit the locality of peers. We also explore how the system adapts to different network conditions. The methodology we develop allows also to identify periodic behavior of the application, highlighting bursts of both data and signaling traffic.**

I. Introduction

In 1997 and 1998, Francis and Zhang respectively asserted that IP multicast was going nowhere, and that some form of application-level multicast was needed to bring multicast to the masses [1], [2]. This primogenial proposal of end-user multicast relied upon the peer-to-peer paradigm and foresaw a tree-based end-host overlay architecture able to easily scale to hundreds of nodes. Offering multicast at the application layer seemed the most flexible solution, given the lack of scalability, violation of the routers stateless principle and some practical issues preventing the broad offer of multicast at the IP layer.

The following years have witnessed the tumultuous diffusion of P2P systems for file sharing, with swarming mechanism offering the best performance. The novelty of such systems relies upon two main ideas: cooperation among peers and peer resource sharing, i.e., sharing of content, bandwidth, memory and processing power. The new concept of P2P users providing service to other users, while obtaining services from the system, has several beneficial effects on performance: it increases system capacity, improves network service reliability, makes the network more flexible and adaptive to the users needs. These features and the key attribute of a limited server infrastructure cost naturally made P2P technology emerge as a strong candidate to satisfy the demand for live or near-live streaming over the Internet. Undoubtedly the very last period testifies the sprouting popularity of several P2P video streaming systems, such as PPLive [6], Coolstreaming, PPStream, TVants. They all have attracted millions of users to watch live or on-demand video programs on the Internet [3], [4], [5]. Moreover a new generation of high-definition commercial P2P video applications, such as Joost [7], Babelgum, Zattoo, Sopcast, TVUnetworks, are at an advanced stage of prototyping and beta-testing with the aim of capturing the IPTV business opportunities. These systems are targeted to offer large bandwidth video streams (1-10 Mbit/s) to a very large population of users (up to tens of millions).

At the same time, the opportunity for P2P-TV systems to attract millions of users constitutes a worry for network carriers since the traffic they generate may potentially grow without control, causing a degradation of quality of service perceived by Internet users or even the network collapse (and the consequent failure of the P2P-TV service itself). Therefore, analyzing the behavior of P2P-TV systems and characterizing the traffic injected in the network are necessary steps toward the evaluation of the possible potential risks for the transport network.

Unfortunately, the majority of above described systems are proprietary, and thus their protocols/architectures/algorithms are inaccessible. Some measurement campaigns of P2P-TV systems have been already presented in the literature, including both active and passive characterization of P2P-TV system and protocols, users behavior and service quality. In particular, a measurement study of PPLive is undertaken in [4]. Authors measured indexes such as bitrate, contacted peers versus time, geographic distribution of peers, etc. A similar characterization was performed in [8], in which authors analyzed the peer churning rate, the system stability and provided a more detailed description of control protocols used by the application. Both previously cited papers focus on PPLive and, to the best of our knowledge, no previous measurement about Joost is available. However, to date, no common methodology has emerged to characterize and understand different systems. In this paper we therefore propose a measurement methodology which allows us to obtain a detailed characterization of traffic generated by P2P-TV applications. In particular, by running several experiments in which one or more clients access to the P2P-TV service, we observe each local peer behavior under different network conditions. Simple indexes, like the download/upload bitrate, are first considered. Then more complex measurement indices are defined. In particular, looking at the peer connection pattern, i.e., the amount of traffic the peer sends to each other peer in the system over time, we identify regular and periodic patterns that could be difficult to highlight from traditional measurements.

This work has been supported by the European Commission under the 7th Framework Programme STREP Project "Networks Aware Peer-to-Peer Application under WIse NEtwork" - NAPAWINE.

TABLE I
SUMMARY OF THE MAIN CHARACTERISTICS OF THE P2P-TV APPLICATIONS

	PPLive		Joost	
	signaling	data	signaling	data
Pkt size [B]	< 150	> 700	< 150	>900
L4-protocol	UDP/TCP	UDP	UDP	UDP
Video Stream Rate [kbps]	—	300-350	—	500-600
Total Bitrate [kbps]	100-150	400-450	1-3	550-650

Measurements are applied considering both PPLive and Joost. PPLive has been selected in light of the fact that nowadays it is one of most popular P2P streaming applications, while Joost has been chosen because it is the second generation system that reached the more advanced stage of protopying. PPLive offers both real-time and on-demand service, while Joost offers only on-demand streams. Clear differences can be observed through the methodology we are proposing.

II. EXPERIMENTAL SET-UP

P2P-TV (and P2P systems in general) rely on the fact that several nodes, called peers, are present at the same time and they act as both information destination and source. Information, i.e., the video stream or the file to be downloaded, is split into small pieces, called chunks. Each peer contacts other peers (called neighbors) to download chunks, while at the same time it can transmit already received chunks to other neighbors, so that an overlay topology is formed. The topology evolves over time since new neighbors can be contacted, while old ones could be dropped.

Our goal is to characterize the peer behavior, starting from simple measurements like the aggregate bitrate peers generate, and moving on more complex indexes to catch the overlay evolution over time. A "black-box" approach is considered, in which the traffic generated by P2P-TV clients during normal activity is observed. To this purpose, we setup a simple testbed composed by some PCs running the P2P-TV application. A Linux router connects the testbed PCs to the Internet via our University Campus LAN. It can enforce different network conditions, e.g., bandwidth or delay constraints. The PCs run the selected P2P-TV application in isolation, so that no other application is accessing the network. Packet sniffers are used to record i) all packets sent/received by the PCs to/from the Linux router, and all packets sent/received by the Linux router to/from the Internet. We will refer to the first and second case as *peer* traffic or *queue* traffic respectively.

III. SINGLE PEER MEASUREMENTS

We first run a set of experiments to characterize the aggregate amount of information a peer receives from and sends to the network with the two applications under analysis: PPLive and Joost. Starting from the packet level trace we focus on the amount of data exchanged by the local peer x with every other neighbor p.

First we derive the Cumulative Distribution Function (CDF) of the IP packet size; Fig. 1 reports it for the received traffic during a typical experiment. The packet size typically follows

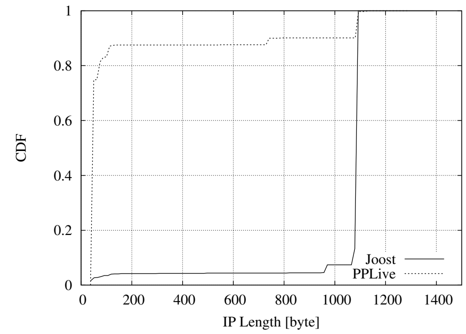

Fig. 1. Cumulative Distribution Function of the incoming traffic IP packet length.

a bimodal distribution, with either small or large packets. As already notices in [8], it can be assumed that small packets correspond to signaling traffic, while large packets carry content data so that a simple packet size threshold can be used to distinguish between signaling and data traffic. By observing the CDF in Fig. 1, we chose the following rule: packets smaller than 150B are considered signaling packets in both applications, packets larger than 700B or 900B are data packets for PPLive or Joost, respectively. First row of Table I reports the chosen thresholds; the rest of the Table shows information about the adopted transport protocol and the average reception and transmission bitrate. Both applications heavily rely on UDP as transport protocol, TCP being (very rarely) used by PPLive only. The download bitrate is similar for the two systems, but PPLive video rate is significantly smaller than Joost one, while PPLive signaling traffic is much larger. Notice that PPLive higher signaling traffic is due to the large amount of incoming signaling packets received from peers that download chunks from the local peer.

We are now interested in the bitrate and neighborhood evolution versus time. In the following, whenever needed, superscripts are used to state the particular information flow the measurement refers to: distinction will be made between transmitted (tx) or received (rx) traffic, and between signaling $(\cdot|\mathrm{S})$ or data $(\cdot|\mathrm{D})$ traffic.

Let $b(i)$ be the bitrate, i.e., the amount of exchanged information by peer x during time interval i of duration τ.

Let $p(t)$ be the function that returns the neighbor peer identifier of a packet received/transmitted at time t. Peers are identified by increasing numbers according to their arrival order, with N the total number of peers observed during the experiment. $p(t)$ represents the swarm evolution versus time t[1].

Finally, let $\mathbb{I}(p, i)$ be an indicator function that takes value 1 if peer p is active during the i-th time interval of duration τ.

From the previously defined variables, it is possible to measure:

- Number of *new peers* $n(i)$ at interval i, defined as the

[1] $p(t)$ is defined only for those time instants at which a packet is received.

number of peers that exchange packets with x during time interval i for the first time:

$$n(i) = \max_{t\in[0,i\tau)} p(t) - \max_{t\in[0,(i-1)\tau)} p(t)$$

- The vector of active peers at interval i,

$$\underline{s}(i) =< \mathbb{I}(1,i), \mathbb{I}(2,i), \ldots, \mathbb{I}(N,i) >$$

- Number of *active peers* $a(i)$: the number of different peers that exchange packets during the i-th time interval,

$$a(i) = \sum_{p=1}^{N} \mathbb{I}(p,i) =< \underline{s}(i) \cdot \underline{s}(i) >$$

where $< \underline{u} \cdot \underline{v} >$ is the scalar product between vectors $\underline{u}$ and $\underline{v}$.
- Normalized *peer correlation* $C(j)$:

$$C(j) = \frac{1}{kM} \sum_{i=1}^{M} < \underline{s}(i) \cdot \underline{s}(i+j) > \quad j \neq 0 \qquad (1)$$

$$k = C(0) = \frac{1}{M} \sum_{i=1}^{M} < \underline{s}(i) \cdot \underline{s}(i) >$$

$C(j)$ represents the average number of active peers at interval i that are also active at interval $i+j$. The peer correlation is defined averaging over M different time intervals. It is normalized to the average number of active peers $C(0)$. Intuitively, large numbers of $C(j)$ state that a large fraction of peers active in i are also active after j time intervals. Small values of $C(j)$ on the contrary state that after j time intervals the set of active peers is very different.

A. Experimental Results

We report a set of results that highlight the behavior of the two applications. While we report selected sample traces, similar results have been reproduced under several conditions and several times, using different network conditions, watching different channels, and during different hours of the day. Reported results represent therefore the typical behavior of the two applications. An extended set of results is available from [9].

1) Bitrate Measurements: Figures 2 and 3 report the evolution of the bitrate, $b^{(\mathrm{rx})}(i)$ and $b^{(\mathrm{tx})}(i)$, respectively, for PPLive and Joost; $\tau = 1s$ is used and no distinction between signaling and data traffic is made. The available bandwidth is unconstrained during the first part of the experiment; after about 380s, the available downstream bandwidth is constrained to values slightly below the average download rate that are 400Kbps and 550Kbps[2], respectively, for PPLive and Joost. Upstream capacity is not affected by any constraint. Figures report the received bitrate at the bottleneck queue and at the peer, as well as the upload bitrate.

In normal network conditions (before $t = 380s$), the bitrate $b^{(\mathrm{rx})}(i)$ is around 600 kbit/s for both the applications, even

[2]Thresholds were selected to match the video rate. Similar results are obtained with different values.

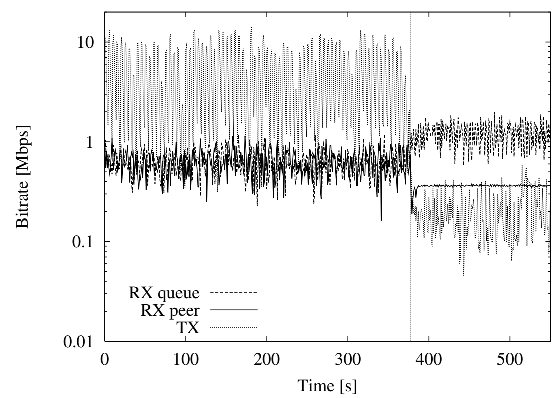

Fig. 2. PPLive: bitrate evolution versus time.

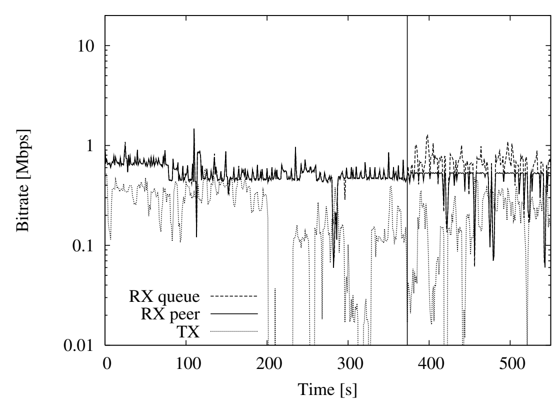

Fig. 3. Joost: bitrate evolution versus time.

if the video stream bitrate is different (see Table I), due to the PPLive higher signaling activity. Joost receiver rate seems more constant than PPLive one. The $b^{(\mathrm{tx})}(i)$ is instead rather different: PPLive exploits the available large upload bandwidth of our peer and transmits up to 15 times the amount of received information. On the contrary, Joost appears much less aggressive than PPLive, transmitting at fractions of the download bitrate and, again, in a less bursty way. The bitrate measured at the bottleneck queue and at the peer are the same, since no bandwidth constraint is present.

At $t = 380s$ the download bottleneck kicks in: PPLive reacts to the adverse situation becoming more aggressive, as shown by the increase of $b^{(\mathrm{rx})}(i)$ at the queue that is about twice the previously observed received bitrate. Joost appears less aggressive, but also in this case the offered load to the bottleneck queue is larger than its actual capacity. We repeated this experiment using different values of the bottleneck capacity, and in all cases we verified similar behaviors as those shown in these figures. We can therefore conclude that both PPLive and Joost keep pushing the video stream to the peer even if the download capacity does not meet the minimum required capacity. Notice that this behavior is not transient and continues for the whole experiment duration. Considering the transmitted bitrate, the PPLive peer cannot

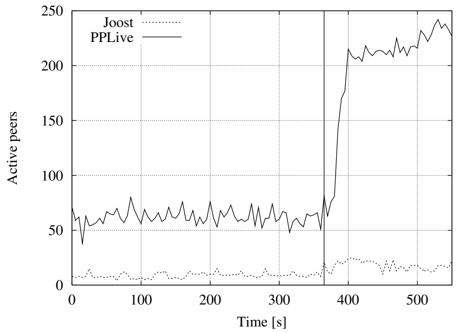

Fig. 4. Number of active peers considering data traffic observed every $\tau = 5s$.

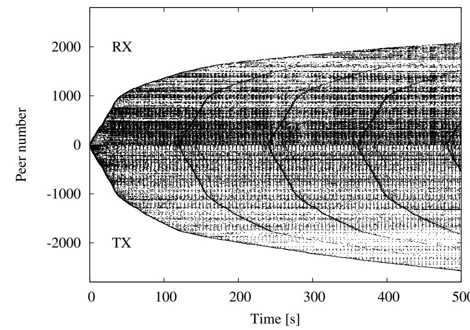

Fig. 5. Peer number vs time. PPLive signaling: $p^{(\mathrm{rx}|\mathrm{S})}(t)$ (positive values) and $p^{(\mathrm{tx}|\mathrm{S})}(t)$ (negative values).

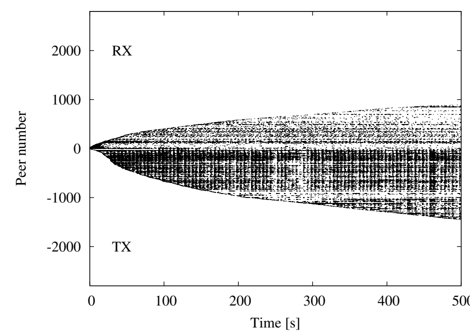

Fig. 6. Peer number vs time. PPLive data: $p^{(\mathrm{rx}|\mathrm{D})}(t)$ (positive values) and $p^{(\mathrm{tx}|\mathrm{D})}(t)$ (negative values).

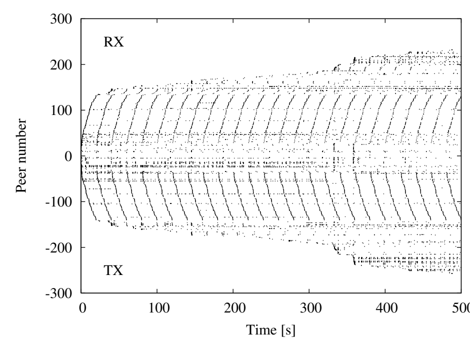

Fig. 7. Peer number vs time. Joost signaling: $p^{(\mathrm{rx}|\mathrm{S})}(t)$ (positive values) and $p^{(\mathrm{tx}|\mathrm{S})}(t)$ (negative values).

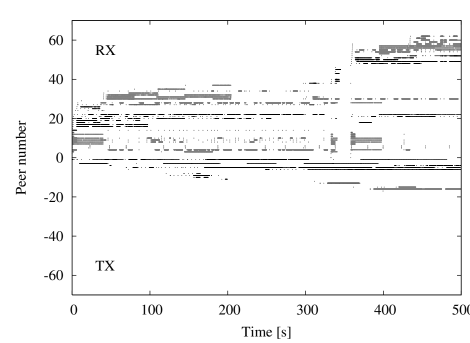

Fig. 8. Peer number vs time. Joost data: $p^{(\mathrm{rx}|\mathrm{D})}(t)$ (positive values) and $p^{(\mathrm{tx}|\mathrm{D})}(t)$ (negative values).

effectively upload content to other peers, and $b^{(\mathrm{tx})}(i)$ falls down almost immediately, the remaining upload bitrate is due to signaling traffic only. The Joost peer, instead, keeps transmitting information to other peers, since it serves "old" content that had been previously cached. This approach is due to the fact that Joost is a VOD system.

2) Peer Number and Type: For the same experiment discussed above, Fig. 4 reports the number of active peers in time intervals of duration $\tau = 5s$. During normal network conditions, PPLive is concurrently receiving data from more than 50 other peers, while only very few peers are used by Joost, probably due its VOD nature and the still low diffusion. After the bandwidth reduction, the number of neighbors increases to cope with the degraded performance; in the case of PPLive, in particular, the number of contacted peers reaches 240, i.e., more than 4 times the value during normal activity. Notice that the number of active peers is an important index that, if too large, can negatively affect the performance of stateful Layer-4 devices, like NAT boxes or Firewalls.

By performing DNS reverse lookup, and by using publicly available geolocation services, we discovered that PPLive peers are in more than 90% of cases located in China, with the second largest population in Japan; only few peers are actually located in Europe and in Italy. For Joost, most of the contacted peers are in Europe (60%) and US (17%), with the largest portion being in Italy (20%). In addition, 20% of Joost peers appears to be "server" peers located in the Netherlands and used mainly during the initial phase (first 100s). While this clearly reflects the customer base of the two applications, still Joost is probably more effective than PPLive in selecting "closer" neighbors.

3) Peer Contacts: A graphical representation of the PPLive swarm evolution as seen by the local peer x is reported in Figs. 5 and 6. In particular, Fig. 5 reports $p^{(\mathrm{rx}|\mathrm{S})}(t)$ and $p^{(\mathrm{tx}|\mathrm{S})}(t)$ while Fig. 6 represents $p^{(\mathrm{rx}|\mathrm{D})}(t)$ and $p^{(\mathrm{tx}|\mathrm{D})}(t)$; positive values are used to identify peers from which x is receiving packets, negative values peers to which x is transmitting. Plots refer to experiments during which no artificial bandwidth constraint is present.

PPLive induces a highly dynamic pattern of peer-to-peer interactions. For the whole activity period, the local peer continuously contacts other peers. As also shown in Fig. 9 which plots $n(i)$ considering a time interval of $\tau = 30s$, the rate of newly contacted peers is much faster during the initial phase, peaking to more than 800 newly contacted peer in the first 30s. After the initial phase, new peers are contacted at an almost constant rate of about 1 peer per second. The cardinality of contacted peers reaches the order of thousands within a few minutes. Moreover, most of the peers previously contacted are continuously polled, as suggested from the many horizontal lines in Fig. 5. Thus the local peer is continuously interacting with several hundreds of other peers. As previously noticed, this poses possible severe scalability issues in case of stateful Layer-4 devices. Some periodic information exchange can also be noted considering $p^{(\cdot|\mathrm{S})}(t)$: a periodic polling every 5s is performed by the local peer versus all neighbor peers (see the vertical patterns in negative part of Fig. 5) and a slower poll mechanism is adopted every 120s involving all the peers, as suggested by the diagonal darker patterns.

As indicated by Fig. 6 that reports $p^{(\cdot|\mathrm{D})}(t)$, not all peers to which some signaling is exchanged are used to download content: the number of data neighbors reduces to a few hundreds for received traffic and more than one thousand for transmitted data.

Joost swarm activity is, instead, represented in Figs. 7 and 8 considering signaling and data traffic, respectively. Even if the qualitative behavior of Joost resembles the one of PPLive, the number of contacted peers are scaled down of a factor of almost 10. This reflects directly into an average smaller number of new peers contacted every 30s, as shown in Fig. 9. Also in the Joost case, a clear periodic behavior is observed in the $p^{(\cdot|\mathrm{S})}(t)$ plot (as shown by the diagonal patterns about every 20s), while a more stable interaction between peers is suggested by the stronger horizontal patterns in $p^{(\cdot|\mathrm{D})}(t)$.

Finally, with respect to PPLive, Joost contacts new peers in a more bursty way, since bunches of new peers appear during very short periods (shown by vertical jumps in the plots and by the peaks in Fig. 9); this happens correspondingly to the transmission of commercial advertisements or ends of TV programs, during which the local peer is seeking new contents and sources. Since it is easy to envision a higher degree of overlay dynamics during these periods induced by users channel zapping, we suspect that the application prevents this by contacting bunches of new peers to guarantee service continuity.

We now focus on the normalized peer correlation defined in (1) and plotted in Figs. 10 and 11 for PPLive and Joost, respectively. We restrict the analysis to received packets only, i.e., $C^{(\mathrm{rx})}(j)$, considering $\tau = 1s$, since this case is representative of the typical application behavior. In both pictures, it is possible to observe several regular peaks, that reflect the periodic pattern of the received traffic . Considering Joost, signaling packets are cyclicly received from any other peer every 20s in a round-robin fashion (see also Fig. 7); in the correlation plot this corresponds to peaks for values of $j = 20, 40, 60, \ldots$. Similarly, for PPLive, two periodic patterns are shown: i) the diagonal curves in Fig. 5 that correspond to high peaks in the peer correlation at $j = 120, 240, 360, \ldots$; ii) a periodic trend of about 5s that corresponds to peaks at values of $j = 5, 10, 15, \ldots$. Interestingly, the long term periodicity is present on signaling traffic only, while the short term one is reflected by both signaling and data traffic. The second periodic behavior is caused by the fact that peer activity appears bursty, so that every 5 seconds new bulks of data are exchanged with other peers. Since UDP is used as transport protocol, we suspect that bursts of packets are sent by each source every 5 seconds, i.e., at the beginning of each transmission period. On the contrary, Joost correlation shows that no periodicity is present considering data traffic. Indeed, by further investigating the packet level traces, we noticed that a shaping algorithm is adopted by peers to send traffic at a constant rate, so that packets are uniformly spread during the activity period. This is reflected by the more constant bitrate previously observed in Fig. 3.

At last, the slow decay of the correlation functions shows that there is a group of peers that are contacted for long time, so that $C(j)$ does not go to zero even after quite large time lags, e.g., for $j > 250$. In particular, $C^{(rx|S)}(j)$ is larger than $C^{(rx|D)}(j)$, suggesting that the neighbor set is quite stable and built by several peers, but the number of peers with which data is exchanged is smaller and varies faster.

IV. Multiple peer measurements

In this section we consider experiments that involve two or more peers connected to the same local network and that are tuned on the same channel, thus forming a local cluster. Under this scenario, it is interesting to observe if the P2P-TV system exploits the peers locality and the overlay topology includes peers of the local cluster to exchange data. We run a set of experiments to observe if local peers are aware of each other. Results (not reported here due to lack of space) show that in PPLive the local peers start exchanging both signaling and data traffic almost immediately; in Joost, the local peers exchange only signaling traffic most of the time, while occasionally exchanging data traffic too. In both cases, the amount of data traffic between the local peers corresponds to a fraction of about 1/10 of the received bitrate (being the other 9/10 received by other neighbors). Intuition suggests that PPLive is more aggressive than Joost in exploiting peer local clustering.

It is interesting to observe how many common external peers are shared considering local cluster peers, i.e., how similar the set of neighbors of local peers are. Let $\mathcal{P}^{(x)}(i) = \{p|\mathbb{I}(p,i) = 1\}$ be the set of peers that are active at time i and that exchange data with peer x. Given any two peers, x_1, x_2, let $c^{(x_1,x_2)}(i) \in [0,1]$ be the ratio of common peers at time i, defined as

$$c_{(x_1,x_2)}(i) = \frac{\mathcal{P}^{(x_1)}(i) \cap \mathcal{P}^{(x_2)}(i)}{\mathcal{P}^{(x_1)}(i) \cup \mathcal{P}^{(x_2)}(i)} \qquad (2)$$

$c_{(x_1,x_2)}(i) = 0$ if no x_1 neighbor is a x_2 neighbor, while $c_{(x_1,x_2)}(i) = 1$ if x_1 neighbors are the same of x_2 neighbors.

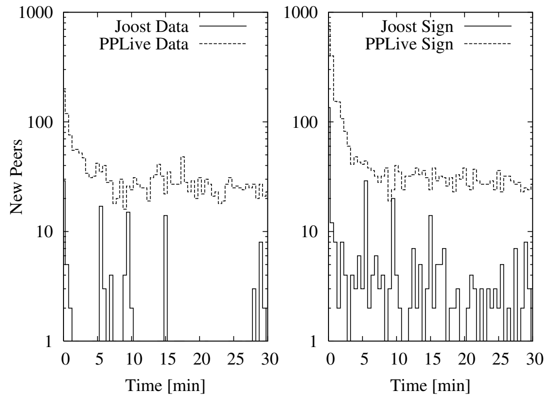

Fig. 9. Number of new peers per 30 second, $n(i)$, $\tau = 30$s.

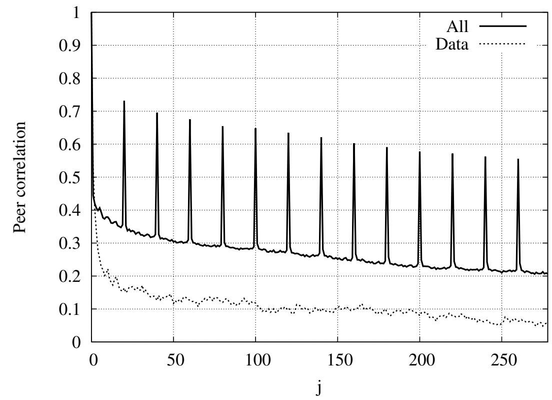

Fig. 11. Joost: normalized peer correlation, $C(j)$.

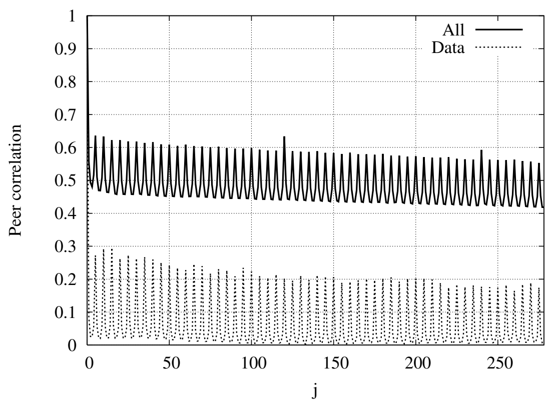

Fig. 10. PPLive: normalized peer correlation, $C(j)$.

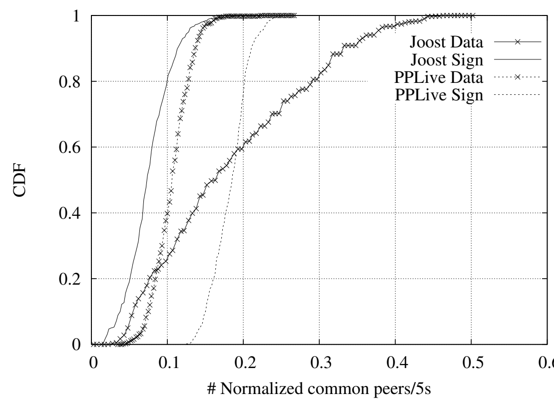

Fig. 12. Distribution of the normalized number of common peers, $c_{(x_1,x_2)}$, $\tau = 5s$.

Fig. 12 refers to an experiment with two active peers in our subnetwork tuned on the same channel. In particular, the figure shows the CDF of $c_{(x_1,x_2)}$, $\tau = 5s$ is used. Distinction is made between peers from which signaling or data traffic is received. When considering Joost, the fraction of common peers tends to be larger for data traffic, suggesting that a similar neighbor selection policy is applied by the two local peers. Note that the fraction of common peers that are contacted for signaling is, on the contrary, smaller, suggesting that the local peers have different neighbor sets. For PPLive, the fraction of common peers for signaling is significantly larger than the fraction of common peers from which data is received. This hints to more similar neighbor sets, but a more independent peer selection mechanism.

V. Conclusions

In this paper we have investigated the main characteristics of traffic originated by PPLive and Joost, two popular and well-known commercial P2P-TV systems. We defined several metrics that allow us to investigate the main time and spatial properties of P2P-TV traffic, the periodic behavior of sent and received traffic, the speed at which the neighborhood is formed and its dynamic variability. Moreover, by artificially constraining the available bandwidth of our test peer, we showed that probably neither PPLive nor Joost implement congestion control mechanisms, so that the local peer keeps trying to download a large amount of traffic even if the actual bandwidth is not large enough to support the service. By performing experiments in which some peers are present in the same local network, we also investigated the neighbor selection policy.

References

[1] P. Francis, "Yoid: Extending the Internet Multicast Architecture", Technical Report, April 2000.

[2] Y. Chu, S. Rao, H. Zhang, "A case for end system multicast", Sigmetrics 2000, Santa Clara, Ca, June, 2000.

[3] S.Q. Zhuang, B.Y. Zhao, A.D. Joseph, R.H. Katz, J.D. Kubiatowicz, "Bayeux: an architecture for scalable and fault-tolerant wide-area data dissemination", ACM NOSSDAV 2001, New York, NY, June 2001.

[4] X. Hei, C. Liang, J. Liang, Y. Liu, K.W. Ross, "A Measurement Study of a Large-Scale P2P IPTV System,". IEEE Transactions on Multimedia, December 2007.

[5] C. Huang, J. Li, K.W. Ross, "Can Internet VoD be Profitable?", ACM Sigcomm 2007, Kyoto, August 2007.

[6] PPLive, http://www.PPLive.com.

[7] Joost, http://www.joost.com.

[8] S. Ali, A. Mathur, H. Zhang, "Measurements of Commercial Peer-To-Peer Live Video Streaming", Workshop on Recent Advances in Peer-to-Peer Streaming, August, 2006.

[9] *Network-Aware P2P-TV Application over Wise Networks* - European Framework Project 7 (FP7) http://www.napa-wine.eu/cgi-bin/twiki/view/Public/WP31Results