A Model to Seize the Instantaneous Performance of P2P Streaming Platforms: Simulative and Experimental Validation

M.L. Merani, D. Saladino
Dept. of Information Engineering
University of Modena and
Reggio Emilia, Italy
Email: marialuisa.merani@unimore.it, daniela.saladino@unimore.it

G.P. Leonardi
Dept. of Applied Mathematics
University of Modena and
Reggio Emilia, Italy
Email: gianpaolo.leonardi@unimore.it

Abstract—**In recent years P2P technology has gained great popularity not only in file sharing applications but also in the field of video distribution.**

This work proposes a simple model to assess the performance a P2P system designed for video streaming can achieve, measured in terms of average video delivery rate. The model allows to compute system efficiency in a simple but accurate way through the partition of peers in two distinct populations: bad peers, that are not collaborative, and good peers, that contribute to share their video contents with others. As a meaningful example, it is employed to investigate the behavior of a real P2P prototype subject to high peer dynamics: its effectiveness is proved via experimental results, and sheds a new light on the way a streaming service overlay can be centrally and timely monitored.

I. Introduction and Related Work

P2P technology for video and television broadcasting has experienced a rising success among network users. The reasons of its success are the lack of necessity of network infrastructure support and the possibility of peers to cooperate with each other in order to diffuse contents. These factors allow to have the reduction of implementation costs and a great scalability.

An excellent tutorial on P2P-based broadcast is provided in [1], together with a critical discussion on the challenges accompanying its wide-scale deployment. A direct insight on a live video streaming architecture is offered in [2], discussing possible improvements and key trade-offs in system design. The contributions in [3], [4] and [5] reveal features and potentials of the P2P technology when deployed in the live streaming arena. Finally, IP-TV provisioning with service guarantees, traffic pressure on ISPs and security concerns are discussed in [6].

There are undoubtedly several relevant issues to face in the design of a P2P streaming architecture. In this work, we investigate how the performance of a P2P streaming system can be assessed in a simple yet accurate manner. The performance metric we consider is the average video delivery rate, definitely a good indicator to quantify the efficiency that the overlay achieves. An effective and tractable analytical model, that groups the overlay peers in two distinct populations, is proposed, leading to a closed-form expression for the evolution of such efficiency metric as a function of time. This analytical form is then employed to successfully predict the performance variations that the P2P overlay displays when different time trends for the number of peers in system are examined.

In literature there are some theoretical papers that venture into the modeling and analysis of P2P streaming systems. An interesting analytical study is presented in [7], where the authors highlight the effect of some factors, such as upload/download capacity heterogeneity and playback delay, on system behavior. In particular, the reference metric they adopt is the probability of universal streaming, a condition that occurs when all peers within the overlay receive the video at the full streaming rate and are therefore completely satisfied of the service they experience. Determining such probability relies on the *resource index*, a time-varying system parameter, defined as the ratio between the available overall capacity, provided by contributing peers and by the video server, and the total bandwidth required to achieve universal streaming. In [8], a significant and original modeling effort is performed to investigate the performance of a multi-channel P2P streaming system. In this work too, Wu et al. employ the resource index as the quantitative measure of channel streaming quality, in order to understand when the system reaches universal streaming. Also our simulations and experiments adopt this metric to determine if the system is working in an underloaded or overloaded status, as in [9]: in the former condition, there is excess capacity to fulfill all peers' download requirements, whereas in the latter, capacity is scarce. But the analytical model we present in this paper provides more, as it allows to evaluate to what extent system performance degrades in case of a sudden step join of new users. In [7] and [8], the only stochastic process is the number of peers in system, while the peers' upload contributions coincide with their capacities; in our model, instead, we consider the bandwidth that the peers actually provide, and therefore have two different stochastic processes: the actual upload bandwidth and the number of peers concurrently connected. As the results will show, this allows to capture some behaviors that otherwise would go unseen.

A P2P video streaming system is also analyzed in [10], where the authors evaluate the performance of the P2P architecture via its efficiency, defined as the probability of a peer being in the uploading status. This model proves that the upper bound of efficiency is tightly correlated to the average number of neighbors each peer owns. In our work, instead, system efficiency is more reasonably defined as the average video delivery rate that is achieved within the overlay, and it is computed as a function of the overall number of "good" contributing peers.

The soundness and effectiveness of the proposed model is proved resorting to a simulative reproduction of a P2P overlay: in particular, the analytically predicted values of the efficiency indicator and the values observed via simulation are extremely close. Tests are also conducted on a real P2P video streaming prototype: the PlanetLab [11] replica of analogous stresses on the examined system reveal that indeed the proposed approach closely reflects what happens in the overlay.

Additionally, the model quantitatively highlights when a successful distribution of the video can be achieved, via a condition on the ratio between the size of the peer population that uploads content and the totality of the active peers.

The rest of the paper is organized as follows. Section II introduces the efficiency indicator employed to evaluate the performance of the P2P video streaming system and describes the partition of peers into two different populations. Section III presents the derivation of the mathematical model and the closed-form expression for system efficiency. Section IV discusses experimental results, that validate the model and support its predictions. Finally, Section V concludes the paper and suggests possible future work.

II. Definitions and Assumptions

A. Efficiency Measure at System Level

We consider a general reference model for the P2P system, without introducing any specific assumption on the underlying overlay topology (i.e., mesh, tree, hybrid), and choose as the elementary building blocks of our analysis the upload and the download rates effectively exploited by the peer x at time t, that we term $u(t,x)$ and $d(t,x)$, respectively. A very reasonable efficiency indicator for peer x at time t is the ratio $\frac{d(t,x)}{d}$, where d represents the video streaming rate, so that this ratio takes on values between 0 and 1, unity indicating that the peer is able to download the whole stream. Notice however that a value strictly lower than one does not necessarily imply the peer is not satisfied: that will depend on the actual encoding scheme the system adopts.

Next, we introduce the efficiency $E(t)$ at time t of the whole system: denoting by $N(t)$ the set of peers in the overlay at time t, we set

$$E(t) = \frac{1}{N(t)\cdot d}\sum_{x\in N(t)} d(t,x)\,. \qquad (1)$$

In other words, $E(t)$ is defined as the average download rate normalized to the streaming rate d, and as such $0 \leq E(t) \leq 1$.

We next recall what we term "the information conservation law", that will be of use for $E(t)$ analysis. We first denote by $U_S(t)$ the server upload bandwidth being instantaneously utilized, and by S the server capacity, so that $U_S(t) \leq S$.

Information Conservation Law

In a generic P2P streaming system, at any given time t, the following relation holds

$$U_S(t) + \sum_{x\in N(t)} u(t,x) \geq \sum_{x\in N(t)} d(t,x)\,. \qquad (2)$$

This law simply states that the server bandwidth plus the overall upload bandwidth of the peers that contribute to the system good functioning is greater than or equal to the sum of all download rates. We further observe that the equal sign in (2) corresponds to the optimistic situation where neither packet losses are observed, nor duplicate packets are transmitted in the overlay. Intentionally, we focus our attention on this limit circumstance, so that we consider

$$U_S(t) + \sum_{x\in N(t)} u(t,x) = \sum_{x\in N(t)} d(t,x)\,. \qquad (3)$$

From (3) it follows that $E(t)$ can be equivalently defined in terms of upload rates as

$$E(t) = \frac{\sum_{x\in N(t)} u(t,x) + U_S(t)}{N(t)\cdot d}\,. \qquad (4)$$

We close this subsection observing that system efficiency $E(t)$ as defined in (4) is profoundly different from the resource index σ widely used in literature [7]-[9]. If we indicate by $u_{max}(x)$ the *constant* upload capacity of peer x, $u_{max}(x) \geq u(t,x)$, $\forall x$, $\forall t$, we recall that σ is defined as

$$\sigma = \frac{1}{N(t)\cdot d}\left(\sum_{x\in N(t)} u_{\max}(x) + S\right)\,. \qquad (5)$$

Incidentally, the P2P system is said to be *underloaded* when it displays a resource index greater than or equal to 1, meaning that – in principle – there is excess capacity to satisfy all peers' download requests (in a complementary manner, the P2P system is *overloaded* when $\sigma < 1$). Yet, in (5) the only stochastic process is $N(t)$. On the contrary, in (4) the terms $u(t,x)$ and $U_S(t)$ can fluctuate too, and are therefore responsible for the essential difference between $E(t)$ and the resource index σ. It is precisely this difference that allows to capture some system behavior that would otherwise go unseen, if σ is the only indicator considered.

The reader should simply think of the impact of a flash crowd, i.e., of a peculiar $N(t)$ instance, on system performance: in this case, the resource index σ reveals little, only the generic indication of either a system with scarce or excess capacity. $E(t)$ does: as it will be shown next, it correctly seizes the occurrence of a potential risk for the overlay and quantifies the transient worsening in performance that takes place.

III. Model Derivation

To capture the evolution in time of the overlay and to understand the dynamic behavior of its efficiency, we assume that system peers $N(t)$ can be divided into two populations, that we denote by $G(t)$ and $B(t)$: whenever there is no ambiguity, we will omit the time dependence to keep the notation slim.

G is the *good* population, whose salient feature is that a peer $x \in G$ uploads content to other peers, subject to the constraint that its upload capacity $u_{\max}(x)$ is not trespassed, i.e., $0 < u(t,x) \leq u_{\max}(x)$, for $x \in G$.

B is the population of "bad" peers, whose upload rate is null, $u(t,x) = 0$ for $x \in B$. The *free rider* population resides in B, as such nodes cannot upload content to other peers: this feature is structural, due to, e.g., the free rider lying behind a NAT or firewall. Also new peers entering the overlay belong to B, as initially they have no content to share.

Later on, peers in B that are not free riders might migrate from B to G: a properly behaving P2P system has to meet the goal of reducing their residence times in B by as much as possible, promoting their quick transition from B to G.

In our analysis we make the following hypotheses:

(H1) $\forall x \in G$, $d(t,x) = d_{\max}$, where $d_{\max} = d \cdot \min(1, \sigma)$. In other words, good peers successfully download content at the maximal rate that the network can support for a given resource index σ.

(H2) Peers in B that are free riders are not penalized by the system: hence, their download rate is $d(t,x) = d_{\max}$ too. New comers, however, have $d(t,x) < d_{\max}$.

(H3) The examined system is a "pure" P2P overlay: equivalently, the server percentually provides a marginal contribution to the system good functioning, the S/d ratio being far lower than $N(t)$. Moreover, its capacity is utilized in full, so that $U_S(t) = S$, $\forall\, t$.

(H4) M classes of nodes are present in $N(t)$, reflecting some *a priori* distribution. We denote by p_j, $j = 1, 2 \ldots, M$, the percentages of peers in each class and by $u_{\max -j}$ the corresponding capacities.

We first provide an $E(t)$ rewriting in terms of the unknown population of bad peers $B(t)$ and their average download rate $d_B(t)$ at time t, $d_B(t) = \frac{1}{B(t)} \sum\limits_{x \in B(t)} d(t,x)$. Owing to (H1) and (H2) we can express the sum of all download rates as

$$\sum_{x \in N(t)} d(t,x) = d_{\max} \cdot G(t) + d_B(t) \cdot B(t)\,. \tag{6}$$

Thanks to (6) and observing that $G(t) + B(t) = N(t)$, we can rewrite $E(t)$ given by (1) as

$$\begin{aligned} E(t) &= \frac{d_{\max}\, G(t) + d_B(t)\, B(t)}{N(t) \cdot d} \\ &= \frac{d_{\max}\, N(t) - (d_{\max} - d_B(t))\, B(t)}{N(t) \cdot d} \\ &= \min(1,\sigma) - \frac{(d_{\max} - d_B(t))\, B(t)}{N(t) \cdot d}. \end{aligned} \tag{7}$$

Next proposition deals with the average download rate of bad peers, $d_B(t)$.

Proposition 1: If we assume that the system works optimally, i.e., it satisfies all download requests up to its overall capacity constraint, we have that

$$d_B(t) = d_{\max} \cdot \varphi \left(\frac{S + (u_{\max}(t) - d_{\max}) G(t)}{d_{\max} B(t)} \right), \tag{8}$$

where $\varphi(r) = \max(0, \min(1, r))$ and $u_{\max}(t)$ is the average upload capacity of the peers in G.
To obtain (8), let us indicate by $u_G(t)$ the average upload rate of good peers at time t,

$$u_G(t) = \frac{1}{G(t)} \sum_{x \in G(t)} u(t,x)\,; \tag{9}$$

taking advantage of (H3), (9) and (6), we can rewrite (3) as

$$S + u_G(t) \cdot G(t) = d_{max} \cdot G(t) + d_B(t) \cdot B(t)\,, \tag{10}$$

whence

$$d_B(t) = \frac{S + (u_G(t) - d_{max}) \cdot G(t)}{B(t)}\,. \tag{11}$$

The hypothesis of a system that works optimally translates into

$$u_G(t) = u_{\max}(t)\,, \tag{12}$$

where we define $u_{\max}(t)$ as

$$u_{\max}(t) = \frac{1}{G(t)} \sum_{x \in G(t)} u_{\max}(x)\,; \tag{13}$$

once (13) is replaced in (11), it leads to

$$d_B(t) = \frac{S + [u_{\max}(t) - d_{\max}]\, G(t)}{B(t)}\,, \tag{14}$$

with $0 \leq d_B(t) \leq d_{\max}$ or, in more compact notation, to (8).

Let us exploit $d_B(t)$ expression for an interesting rewriting of system efficiency $E(t)$. In particular, for the special case when $d_B(t) = d_{\max}$, $E(t)$ in (7) turns into

$$E(t) = min(1, \sigma)\,, \tag{15}$$

whereas when $d_B(t)$ takes on the expression in (14), then

$$E(t) = min(1,\sigma) - \frac{(d_{\max} - \frac{S - [u_{max}(t) - d_{\max}](t) G(t)}{B(t)}) \cdot B(t)}{N(t) \cdot d} \tag{16}$$

that through a few, simple algebraic steps becomes

$$E(t) = \frac{u_{max}(t) G(t) + S}{N(t) \cdot d} \tag{17}$$

By our assumption (H4) on the classes of nodes present in G, we infer that, given the free rider percentage is not excessive, then for G large enough it is reasonable to state that its composition reflects $N(t)$ composition, so that $u_{\max}(t)$ can be well approximated by the *a priori mean*

$$u_{\max} = \sum_{j=1}^{M} u_{\max -j} \ \cdot p_j\,. \tag{18}$$

This fact justifies the replacement of $u_{\max}(t)$ by $u_{\max}$ in (17) without significantly affecting the evaluation of $E(t)$, and the goodness of such replacement is confirmed by the numerical tests presented in the next section. Thanks to this observation, we can rewrite (17) and merge it with (15) into

$$E(t) = \min\left\{\min(1, \sigma),\ \frac{u_{\max} G(t) + S}{N(t) \cdot d}\right\}. \quad (19)$$

Formula (19) shows that $E(t)$ can follow two different regimes. In the first, $E(t)$ coincides with the minimum between 1 and the resource index σ; in the second, $E(t)$ can significantly depart from the resource index σ.

IV. Numerical results

To validate the model and support its predictions, we have resorted to both simulations and real experiments, as detailed in the following subsections.

A. Simulations via PeerSim

For the numerical assessment we resorted to PeerSim [12], a Java based simulator whose basic version allows to simulate P2P systems for file sharing. For video streaming delivery, we have employed and tailored to our purposes an additional protocol, named Overlay Streaming Distribution Protocol [13], and its corresponding modules. This is a hybrid push-pull, mesh-based streaming protocol devised for real time content distribution.

In the examined framework, the video to be distributed to the peers within the overlay is divided into M substreams, each with a rate of d/M: in our simulations $M = 16$. All M substreams have to be received by each peer, in order to guarantee a proper reconstruction of the video. Upon joining the overlay, a new peer is immediately given a list of neighbors, and it is among them that the peer randomly selects its potential parent peers: once these are contacted, if they possess the desired substream and have not exhausted their upload capacity, they start providing the newcomer with video chunks. As the envisioned distribution scheme is push-pull, once a parent peer starts delivering video chunks to a child peer, it continues to do so until either the parent leaves the overlay or the child itself departs. Additionally, every peer is forced to provide each of its children peers, i.e., the peers that receive content from it, with one single substream, to avoid the very likely disruption in video quality that its sudden departure would cause.

Peers dynamically enter and leave the overlay: their interarrival time and lifetime can follow different distribution functions. We have employed the exponential cumulative distribution function to describe the lifetime of the peers, with an average value $\frac{1}{\mu}$ equal to the duration of the video stream, made of 10^4 chunks, and verified that, for a given average, modifying the choice of the actual statistics does not play a significant role. Moreover, the time unit the simulator adopts coincides with the time required to transmit a video chunk and the simulation time is equal to the total video stream duration.

	Residential	Institutional	Free Rider
Scenario 1	$70\% : 1/2d$	$20\% : 7/6d$	$10\% : 0d$
Scenario 2	$70\% : 1/2d$	$30\% : 7/6d$	–
Scenario 3	$40\% : 2/3d$	$60\% : 5/3d$	–

TABLE I
PEERS UPLOAD DISTRIBUTION

	$N = 100$	$N \geq 1000$
Scenario 1	$\sigma = 0.78$	$\sigma \simeq 0.58$
Scenario 2	$\sigma = 0.9$	$\sigma \simeq 0.7$
Scenario 3	$\sigma = 1.47$	$\sigma \simeq 1.27$

TABLE II
RESOURCE INDEX IN THE EXAMINED SETTING

In our simulations, according to [9], we have considered a heterogeneous population of nodes, with different classes of users, as reported in Table I: in the first examined scenario, given the video streaming rate is d, with probability 0.7 peers are classified as "residential" and exhibit an upload capacity of $\frac{1}{2}d$; with probability 0.2 they are "institutional", with an upload capacity of $\frac{7}{6}d$, while with probability 0.1 they are free riders, i.e., their upload capacity is null. In the second scenario, with probability 0.7 the peers have an upload capacity of $\frac{1}{2}d$ and with probability 0.3 can provide $\frac{7}{6}d$. Lastly, in the third scenario with probability 0.4 peers have an upload bandwidth of $\frac{2}{3}d$ and with probability 0.6 they can provide $\frac{5}{3}d$. As regards the streaming server, its upload bandwidth S is $k = 20$ times the streaming rate d, respecting the assumption that we are considering a pure P2P architecture. Moreover, every node has a download bandwidth exactly equal to the streaming rate d.

Depending on the number N of nodes in the overlay, the resource index σ takes on different values: for the scenarios described above, a few are reported in Table II, indicating that for the values of N that are of practical interest, the first two settings refer to an overloaded system ($\sigma < 1$), whereas the third to an underloaded one ($\sigma \geq 1$).

We have next recreated several instances for the evolution of the number in system $N(t)$, all exhibiting a sudden increase at some point in time: this is what is typically observed in P2P broadcasting system when a very popular television event occurs.

Here we report the results obtained imposing a disruptive ramp-like input of new nodes, but analogous outcomes have been obtained in different settings and for different system inputs. As an example, Fig.1 shows the evolution of $N(t)$ we have taken into consideration and its partition into $G(t)$ and $B(t)$ in the first examined scenario. The solid line corresponds to $N(t)$, the dashed line to $B(t)$ and the dotted line to $G(t)$. Note that the average of the peer interarrival time takes on different values in different time intervals, to guarantee $N(t)$ exhibits the behavior shown in the figure.

The lower set of curves reported in Fig.2 give the corresponding $E(t)$ evolution as witnessed by simulation and by

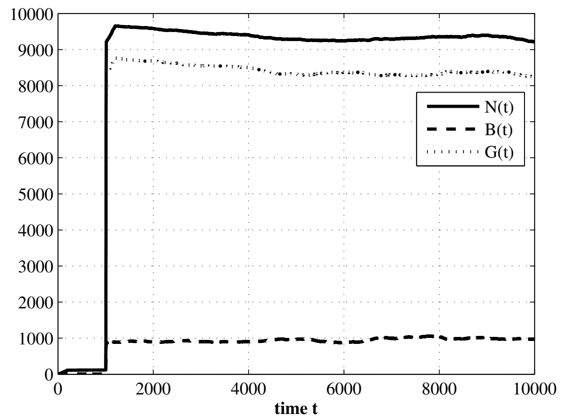

Fig. 1. Partition of $N(t)$ in $B(t)$ and $G(t)$ for the harsh step join in the first examined scenario (overloaded case)

analysis, namely, by eq.(19); two additional pair of curves are shown, that refer to the second and third scenario. Solid lines indicate $E(t)$ as obtained through simulation, dashed lines report $E(t)$ as analytically determined. By visual inspection we conclude that the analytical results are very close to the simulation outcomes: the model is successful in providing the correct time trend that system efficiency follows. Most importantly, it quantifies the worsening in system performance that occurs when the flash crowd hits the overlay, a phenomenon that goes completely unseen if the resource index σ only is considered: indeed, $E(t)$ coincides with σ only before - and well after – the step join takes place, but it significantly departs from it during the transient.

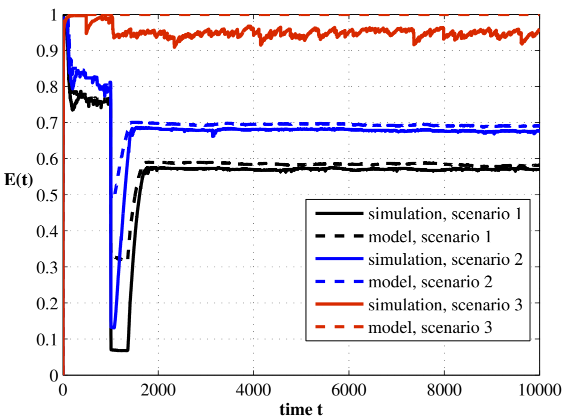

Fig. 2. $E(t)$ evolution corresponding to the input of Fig.1 in the three examined scenarios

B. Experimental Results via Gridmedia on PlanetLab

To check the soundness of $E(t)$ evolution we have also resorted to experiments via a real P2P streaming prototype, GridMedia [14][15], investigating its behavior in the presence of flash crowds on PlanetLab.

	PeerSim	PlanetLab
Initial peer population	˜100	˜5
After the "ramp-like" input	˜10000	˜500

TABLE III
MAIN PARAMETER VALUES FOR PEERSIM AND PLANETLAB EXPERIMENTS

GridMedia delivers video packets relying upon UDP at transport layer, whereas all control messages (e.g., buffer maps) are transmitted via TCP. The GridMedia version we installed on our streaming server divides the video to be distributed into $M = 16$ substreams and adopts a "hybrid" push-pull approach: whenever a GridMedia client peer requests a packet from a neighbor, then all packets belonging to the same substream are automatically pushed from the neighbor to the node.

We have repeatedly distributed an H.264 video stream coded at 500 kbits/s, with a duration of 10 minutes, within an overlay of PlanetLab nodes where we installed the GridMedia client software, and have performed numerous tests imposing an exponential distribution for the lifetime as well as for the off-line time of the client peers. Additionally, we have set the server upload capacity to a value slightly greater than the streaming rate.

The experiments involved a maximum of 500 nodes, an actual PlanetLab limit we had to face. Hence, we properly scaled down from the original simulation setup by a factor of 20, as summarized in Table III.

In detail, we have recreated the ramp-like input, that from the original population size of 5 exhibits a sudden boost of new peers, up to slightly less than 500. An instance of the $N(t)$ evolution in time that we have considered is PlanetLab is displayed in Fig.3, together with the size $G(t)$ of the good population we record in system. From an operational point of view, we have defined a peer to be "good" as soon as it starts providing an upload rate greater than the single substream rate, roughly around 30 kbit/s.

During each experiment, system efficiency $E(t)$ has been experimentally determined, i.e., (4) has been evaluated collecting the upload rates of all peers from the GridMedia logserver with a ten-seconds frequency. Fig. 4 displays the average $E(t)$ behavior, obtained by further averaging the results of 10 different experiments: the solid line refers to $E(t)$ values obtained from PlanetLab measurements, the dashed line to $E(t)$ as predicted by the model through (19), inferring the nodes' capacities from their maximum upload rates. The experimental results successfully validate the model. They also indicate that, when many peers enter the system all together in a small time window, $E(t)$ significantly decreases, then it gradually recovers.

As a conclusion, these outcomes demonstrate that the efficiency a real system achieves is the same as the one predicted by our model.

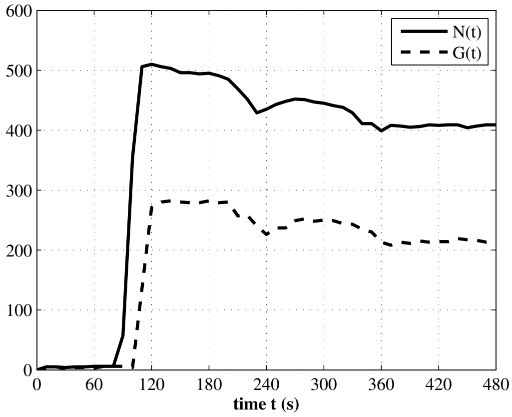

Fig. 3. An instance of $N(t)$ and $G(t)$ evolution corresponding to the harsh step join of new nodes in PlaneLab

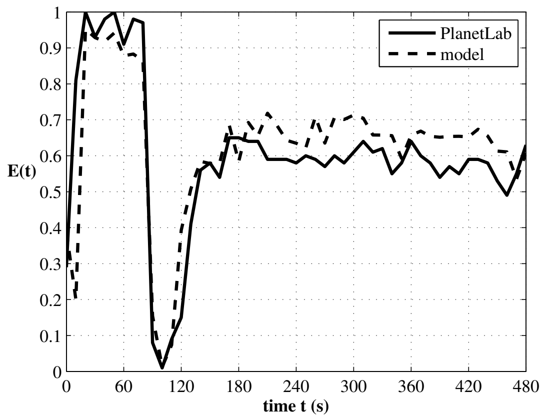

Fig. 4. Average $E(t)$ values for the harsh step join of new nodes

V. Conclusions

This paper has tackled the issue of measuring network efficiency in a P2P video streaming system and of investigating how swift peer dynamics affect such efficiency.

It has focused on an appropriate merit figure, the average normalized video delivery rate, and has put forth a simple model that partitions the overlay peers into two distinct populations: this has allowed to derive a simple, yet accurate closed-form expression for the examined metric.

The model can be employed to predict the efficiency variations displayed by the P2P overlay when stressed by different inputs of peers: in particular, it quantifies how the average delivery rate decreases when a flash crowd occurs, highlighting the remarkable dependence of this metric on the intensity of the input of new nodes. Its results have been successfully validated via both simulations and experiments: the model therefore allows to monitor system performance in a centralized, yet very light and cost-effective manner. It also represents an effective tool to rely upon, when proactively reacting to critical operating conditions that start building up within the P2P system: its real-time indications are essential to drive the adoption of suitable countermeasures, in order to re-establish satisfying operating conditions.

Acknowledgment

The authors would like to thank Daniele Gratteri for having performed most of the PlanetLab experiments discussed in this work.

References

[1] J. Liu, S. G. Rao, B. Li, H. Zhang, Opportunities and Challenges of Peer-to-Peer Internet Video Broadcast. Proceedings of the IEEE, Vol.96, Issue 1, January 2008, pp. 11 - 24.
[2] Y. Tang, J.-G. Luo, Q. Zhang, M. Zhang, S.-Q. Yang, Deploying P2P Networks for Large-Scale Live Video-Streaming Service, IEEE Communications Magazine, Vol. 45, Issue 6, June 2007, pp.100-106.
[3] 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, Vol. 9, Issue 8, December 2007, pp.1672 - 1687.
[4] B. Li, S. Xie, G.Y. Keung, J. Liu, I. Stoica, H. Zhang, X. Zhang, An empirical study of the Coolstreaming+ system. IEEE Journal on Selected Areas in Communications, Vol. 25, Issue 9, December 2007, pp.1627 - 1639.
[5] S. Xie, G.Y. Keung, B. Li, A measurement of a large-scale peer-to-peer live video streaming system. Packet Video 2007, November 2007, pp.153 - 162.
[6] X. Hei, Y. Liu, K.W. Ross, IPTV over P2P streaming networks: the mesh-pull approach, IEEE Communications Magazine, Vol. 46, Issue 2, February 2008, pp.86 - 92.
[7] R. Kumar, Y. Liu, K.W. Ross, Stochastic Fluid Theory for P2P Streaming Systems. Proc. of IEEE Infocom 2007, May 2007, pp.919 - 927.
[8] D. Wu, Y. Liu, K.W. Ross, Queuing Network Models for Multi-Channel P2P Live Streaming Systems, Proc. of IEEE Infocom 2009, April 2009, pp.73 - 81.
[9] Z. Liu, Y. Shen, K. W. Ross, S. S. Panwar, Y. Wang, LayerP2P: Using Layered Video Chanuks in P2P Live Streaming, IEEE Trans. on Multimedia, Vol. 11, No. 7, November 2009, pp.1340 - 1352.
[10] H. Liu, G. Riley, How Efficient Peer-to-peer Video Streaming Could Be? IEEE Consumer Communications and Networking Conference 2009, January 2009, pp.1 - 5.
[11] PlanetLab, "PlanetLab homepage", http://www.planet-lab.org/
[12] PeerSim P2P Simulator, "PeerSim homepage", http://peersim.sourceforge.net/
[13] D. Carra, R. Lo Cigno, E. Biersack, Graph Based Analysis of Mesh Overlay Streaming Systems, IEEE Journal on Selected Areas in Communications, December 2007, Vol.25, no.9, pp.1667-1677.
[14] M. Zhang, L. Zhao, Y. Tang, J.-G. Luo, S.-Q. Yang, Large-Scale Live Media Streaming over Peer-to-Peer Networks through Global Internet, ACM International Workshop on Advances in Peer-to-Peer Multimedia Streaming (P2PMMS), November 2005, pp. 2128.
[15] M. Zhang, L. Zhao, Y. Tang, J.-G. Luo, S.-Q. Yang, Understanding the Power of Pull-based Streaming Protocol: Can We Do Better?, IEEE Journal on Selected Areas in Communications, December 2007, Vol. 25, No. 8.