QoS in Peer to Peer Live Streaming through Dynamic Bandwidth and Playback Rate Control

Maria Efthymiopoulou, Nikolaos Efthymiopoulos
Department of Electrical and Computer Engineering, University of Patras
Patras, Greece
Email: mefthymiop@ece.upatras.gr, nefthymiop@ece.upatras.gr

***Abstract*—Current commercial live video streaming systems are based either on a typical client-server (cloud) or on a peer-to-peer (P2P) architecture. The former is preferred for stability and QoS while the latter is scalable with small bandwidth and management cost. In this paper, we propose a scalable and stable service management architecture for a cloud assisted P2P live streaming system. In order to achieve this we develop an analytical model and a hybrid control strategy that dynamically allocates from the cloud the exact amount of bandwidth that is required while simultaneously dynamically adapts the playback rate to the available bandwidth resources in order to guarantee the complete and on time stream distribution. To the best of our knowledge our proposed model is the first that copes up with a hybrid control strategy for simultaneous playback rate adaptation and auxiliary bandwidth allocation.**

Keywords - *peer to peer; live streaming; control theory; QoS*

I. INTRODUCTION

Video streaming has become a dominant part of today's internet traffic. As Cisco analyzes in [1] between 2012 and 2013, the highest growth happened on the Internet side in online video with 16 percent year-over-year growth. On the other hand the tremendous number of users leads even the major streaming service providers (e.g., YouTube) to suffer from high bandwidth costs and scalability issues. P2P live streaming and P2P video on demand architectures as: [5][10]-[13][16] have received a lot of research attention in the past few years. In order to reveal the importance of our study we highlight the major requirements from P2P live streaming systems which are: i) **Efficiency** of the media distribution in terms of utilization of peers' upload bandwidth, in order to minimize any additional bandwidth contributed by media servers (cloud) and/or maximize the playback rate of the stream which the system is able to deliver, ii) **Stability** of the distribution which is defined as the uninterrupted and complete stream delivery in each participating peer even in the presence of dynamic conditions (e.g., unrelated network traffic, system bandwidth changes, peer arrivals and/or departures) that affect the amount of the available upload bandwidth in the system, iii) **Scalability** which is determined by the amount of resources (bandwidth, storage, processing overhead) that cloud, which manages the system, has to contribute in order to sustain the uninterrupted delivery of the stream, as the number of participating peers grows.

In the literature, there are two strategies in order to dynamically harmonize the relationship between the playback rate and the total upload bandwidth that participating peers and cloud contribute and enable in this way efficiency, stability and scalability. The first [6] is the dynamic allocation of upload bandwidth from auxiliary sources (e.g., clouds) while the second [16] is the dynamic adaptation of the playback rate according to the existing upload bandwidth of participating peers. The selection of a strategy has to do, except the technical issues, with the desire of system's users and the business model that the service provider wants to follow. In case that users and the service provider desire a costless live steaming the second has to be selected. In case that they desire a live streaming with high stream quality the first has to be selected. In this work we propose a hybrid strategy that enables a flexible tradeoff between the advantages of the two.

Towards the first strategy, the research community has proposed monitoring systems, such as [4][6], that use statistical methods for the scalable monitoring of the total available bandwidth resources in a P2P overlay. These systems are scalable but suffer from three drawbacks which are: i) stochastic methods are suitable only for specific upload bandwidth distributions among participating peers, ii) their efficiency in terms of peer bandwidth exploitation is low due to the low confidence interval, iii) they are not stable as they do not capture the system dynamics in cases of sudden disturbances (e.g., underlying network traffic changes and/or peers' arrivals-departures).

In [14], the problem of stability is recognized and it is studied the impact of flash crowds on the stability. In [15] is studied the stability of a real P2P live streaming system and is highlighted that the server plays an indispensable role in the stability. All these works highlight the problems that occur in P2P live streaming without a QoS enabling system.

Motivated by the lack of an analytical model and a holistic study in this area, and based in our previous work [2][3], we develop a control strategy for a non-linear system that is able to dynamically allocate from the cloud the exact amount of bandwidth that is required and simultaneously adapt the playback rate to the available bandwidth resources in order to guarantee the complete and on time stream distribution for each peer.

The reminder of this paper is structured as follows: Section II presents our P2P live streaming system's architecture which is our background work. Section III presents the problems that we solve. In Section IV, is analyzed the proposed scalable bandwidth and playback rate control strategy and in Section V we conclude.

II. PROPOSED SYSTEM ARCHITECTURE

Our P2P live video streaming system consists of a media server in a cloud, (noted by S) and a set of peers (noted by N). S divides the stream into video blocks and is responsible for: i) the initial diffusion of blocks to a small subset of nodes among participating peers, ii) the tracking of the network addresses of a small set of participating peers in order to assist the bootstrap of the P2P overlay, iii) the dynamic and scalable monitor of the resources of participating peers, iv) the dynamic control of auxiliary bandwidth and playback rate.

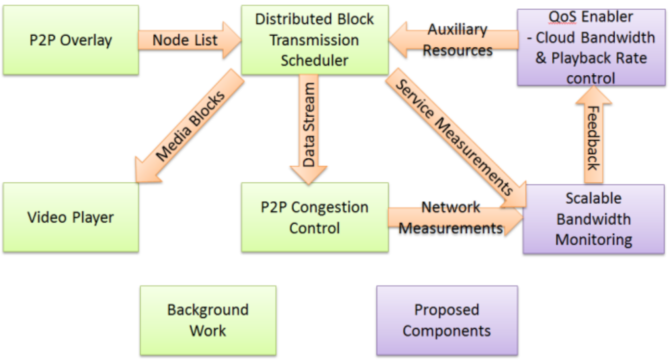

Figure 1. Proposed P2P live streaming architecture and major interactions

In order to allow peers to exchange video blocks, each peer in N maintains network connections with a small subset of other peers which will be noted as neighbors. The sets of these connections change dynamically and form a dynamic graph called the P2P overlay [2][3], which is a graph topology and P2P overlay management algorithms that each peer periodically executes. We use distributed optimization theory in order to dynamically ensure in a distributed (scalable) and dynamic fashion that: i) peers have connections proportional with their upload bandwidth, ii) peers have connections with other peers close to the underlying network, iii) our P2P overlay is adaptable to underlying network changes and peer arrivals and departures.

Distributed Block Transmission Scheduler (DBTS) [2][3] coordinates video block exchanges in a distributed fashion. In order to achieve this we developed a set of algorithms which executed by every peer in N which dynamically communicates with its neighbors in the P2P overlay. The major objective of DBTS is to ensure the timely delivery of every block to every peer by exploiting the upload bandwidth of participating peers and the additional bandwidth resources that media servers may contribute. DBTS sends the video blocks that have to be sent in the P2P congestion control component and the ordered stream with the blocks that it receives to DBTS.

These two components enhance our system with two properties that we exploit here. The first property (Property 1) is that if idle bandwidth exists it is derived from bandwidth surplus in the system and not from the inefficiency of the system to exploit it. In other words we guarantee that the presence of idle bandwidth implies the complete stream delivery. The second property is that the percentages of the idle resources among participating peers are almost equal (Property 2). We highlight here that in case of heterogeneous peer upload bandwidth various peers send with various bitrates (analog with their upload bandwidth capacity) but the percentage of their bandwidth utilization, and so the percentage of their idle time is very similar.

Our P2P congestion control mechanism [7] is able to manage sequential transmissions of video blocks to multiple locations that DBTS sends to it and to provide to the Scalable Bandwidth Monitoring the dynamic estimation of: i) the upload bandwidth capacity, ii) the idle bandwidth resources of each participating peer. In the rest of this paper, we analyze a Bandwidth Playback Rate Control (BPRC) component that acts as the QoS enabler of the P2P live streaming system. Its scalability properties are analyzed in detail in [16].

III. PROBLEM STATEMENT

We assume a set of peers N that receive the same video stream. In order to receive the stream all peers in N issue requests to their neighbors (a small subset of N) with a bit rate p_k which is the media object playback rate. The subscript k is an integer denoting the time instant.

The fulfillment of these requests generates the incoming flows. These requests are served from the same set of peers N which exploit their upload bandwidth. These are the outgoing flows in the system. By exploiting the outgoing flows P2P congestion control is able to calculate dynamically the upload bandwidth capacity, $u_{(i)k}$, and the idle percentage of the upload bandwidth capacity, $id_{(i)k}$, of each participating peer i in N.

The first problem is the development of an analytical model that connects, with an analytical relationship, the bandwidth that we have to allocate or release dynamically and the playback rate with the dynamic idle percentage of the upload bandwidth of the participating peers.

The second problem that we solve is to create a BPRC strategy with which we exploit our analytical model in order control $id_{(i)k}$ of each participating peer in N to a reference value id_{REF} by adapting dynamically, by the use of auxiliary resources (cloud), system's total upload bandwidth and playback rate. In this way, if the total upload bandwidth of the participating peers is greater than the required we dynamically estimate this surplus in order to be able to use it for other purposes (e.g., distribution of another stream) and/or increase p_k. Otherwise, if total system's upload bandwidth is less than the required, we dynamically estimate the amount of the deficit and we demand it from S in order to ensure the stability of the distribution and/or decrease p_k

IV. BANDWIDTH AND PLAYBACK RATE CONTROL (BPRC)

BPRC is a control functionality that is executed periodically, at a time instant k, with period T. It is executed in a centralized fashion by the server, S, who generates the media object that is streamed. Its objective is to set the idle time

 ISBN: 978-1-61208-444-2

percentage $id_{(i)k}$ of each peer i in N to a reference value id_{REF}, by periodically adjusting $U_{(S)k}$ and p_k. As $U_{(S)k}$ we define the total amount of upload bandwidth that should be added/removed from the P2P Overlay every time instant k that BPRC is executed. In the rest of this section we model this process analytically and we propose a control strategy with which we periodically calculate $U_{(S)k}$ and p_k. The symbols that we use are presented altogether in Table I below. Index i indicates a peer i that belongs to N and the index k represents a time instant k. In order to derive the system model we make two assumptions which are:

Assumption 1: According to Property 2 that we described in Section II we can write approximately $id_{(i)k}=id_k$, for each i belongs to N. We note that id_k represents the average $id_{(i)k}$ in N.

Assumption 2: Period T, with which BPRC is executed, is lower than the time interval that is needed for significant changes in the total upload bandwidth of participating peers. So we can do the approximation that total upload bandwidth remains similar between two consecutive executions of BPRC.

At any time instant k and in case that there are sufficient upload bandwidth resources (Property 1) is guaranteed the complete delivery of the stream to every peer in the set N and so the incoming flow to each participating peer is p_k. Thus the sum of the incoming flows of N peers is $N*p_k$. The sum of the incoming flows that peers receive is equal to the sum of the outgoing flows that peers in N contribute by using their upload bandwidth. The sum of the outgoing flows is the sum of their non-idle upload capacity $u_{(i)k}$ so we have:

TABLE I. NOTATION

Symbol	Definition
S	Generator (source) of the media object
N	Set of participating peers (in the equations below we use N as the number of participating peers)
p_k	Dynamic media playback rate at time instant k
$U_{(S)k}$	Amount of upload bandwidth that should be added/ removed from the P2P overlay at time instant k as it determined from BPRC
$u_{(i)k}$	Upload capacity (upper limit) of peer i at time instant k
$id_{(i)k}$	Idle time percentage of peer i at time instant k [0,1]
id_k	Average estimated idle time percentage of N peers at time instant k [0,1]
id_{REF}	Average idle time percentage reference value [0,1]
T	Period of execution of BPRC
m_k	System input that represents the change in the playback rate as determined from BPRC
m_{REF}	System input in the equilibrium point of BPRC

$$Np_k = \sum_{i\in N}(1-id_{(i)k})u_{(i)k} \tag{1}$$

Under Assumption 1 we can rewrite (1) as:

$$Np_k = (1-id_k)\sum_{i\in N}u_{(i)k} \tag{2}$$

Rewriting (2) for time instant k+1,we have:

$$Np_{k+1} = (1-id_{k+1})\sum_{i\in N}u_{(i)k+1} \tag{3}$$

By definition at time instant k+1, total system's upload bandwidth resources, can be expressed as the sum of total system's upload bandwidth resources at time instant k plus $U_{(S)k}$. Thus, holds that:

$$\sum_{i\in N}u_{(i)k+1} = \sum_{i\in N}u_{(i)k} + U_{(S)k} \tag{4}$$

We now define the playback rate at time instant k+1, as the sum of the playback rate at time instant k p_k and w_k which is the difference that BPRC will introduce to the playback rate. Thus, holds that:

$$p_{k+1} = p_k + w_k \tag{5}$$

By using (4),(5) in (3) we have:

$$N(p_k + w_k) = (1-id_{k+1})(\sum_{i\in N}u_{(i)k} + U_{(S)k}) \tag{6}$$

Now by dividing (2),(5) and by using Assumption 2 we have:

$$id_{k+1} = 1 + \frac{(id_k - 1)\sum_{i\in N}u_{(i)k}\left(p_k + w_k\right)}{\left(\sum_{i\in N}u_{(i)k} + U_{(S)k}\right)p_k} \tag{7}$$

We now set:

$$m_k = \frac{\sum_{i\in N}u_{(i)k}\left(p_k + w_k\right)}{\left(\sum_{i\in N}u_{(i)k} + U_{(S)k}\right)p_k} \tag{8}$$

From (7) by the use of (8) we have:

$$id_{k+1} = 1 + (id_k - 1)m_k \tag{9}$$

By setting $id_k=id_{k+1}=id_{REF}$ in (8) we obtain m_{REF} which is defined as the input, in the equilibrium point and is equal to 0. Thus, in this case arises that m_{REF} in the equilibrium point is equal to 1. In order to have a system which has as its equilibrium point (0,0) we now set:

$$x_k = id_k - id_{REF} \tag{10}$$

$$u_k = m_k - m_{REF} \tag{11}$$

The idle time percentage id_k belongs to the interval (0,1) by definition. Thus x_k ranges between ($-id_{REF}$, $1-id_{REF}$). By substituting (10),(11) in (9) we have:

$$x_{k+1} = 1 - id_{REF} + (x_k + id_{REF} - 1)(u_k + m_{REF}) \tag{12}$$

We observe that (12) is nonlinear for a linear closed loop system we use a feedback linearization [9] which introduces a state feedback such that the closed loop system becomes linear. To this end we select a control strategy u_k of the form:

$$u_k = \frac{(1-k_c)x_k}{k_c x_k + id_{REF} - 1} \tag{13}$$

In (12), k_c is a parameter that we choose. By combining now (10), (11) and (12) we have a system with eigenvalue k_c :

$$x_{k+1} = k_c x_k \tag{14}$$

In this way it is easy to observe from (13) that the series $\{x_k\}$ converges to 0, and so id_k converges to id_{REF} for any value k_c that belongs to (-1,1). Since k_c is a designer's choice we can explicitly set the eigenvalue of the system by just setting k_c. If we now combine (8),(10),(11) and (13) we have:

$$\frac{(1-k_c)(id_k - id_{REF})}{k_c(id_k - id_{REF}) + id_{REF} - 1} = \frac{\sum_{i\in N} u_{(i)k}\left(p_k + w_k\right)}{\left(\sum_{i\in N} u_{(i)k} + U_{(S)k}\right) p_k} - 1 \tag{15}$$

So each time that BPRC is executed, after measuring $u_{(i)k}$ and p_k, we have to select a pair of values for w_k and $U_{(S)k}$ such that (15) is satisfied. The selection of this pair has to do the desirable playback rate (p_k+w_k) and the auxiliary upload bandwidth that is allocated from the cloud or released from the P2P overlay which represented in (16). We keep the selection strategy open to any policy. As it can be seen from (15) a high playback rate will lead to a high auxiliary upload bandwidth and a low playback rate to a low auxiliary upload bandwidth.

$$\sum_{t\in(0,k-1)} U_{(S)t} + U_{(S)k} \tag{16}$$

The selection of id_{REF} has to do with the accuracy of our modeling (Assumption 1) and the adversity of the changes (disturbances) in the total upload bandwidth in the P2P overlay (Assumption 2). High inaccuracy and system disturbances need high id_{REF} (high degree of resource overprovisioning) that will guarantee uninterrupted stream delivery.

V. CONCLUSIONS

The proposed model is the first analytical model towards the simultaneous control of playback rate and auxiliary bandwidth provision in P2P live streaming. We leave the evaluation and a robust analysis of our model (to derive the minimum id_{REF} that guarantees uninterrupted stream delivery as a function of the model accuracy and the magnitude of disturbances) as future work.

ACKNOWLEDGMENT

This has been financed by STEER [7] which is an European Commission's FP7 project and by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund.

REFERENCES

[1] Cisco Visual Networking Index, "The Zettabyte Era—Trends and Analysis", 2015.

[2] N. Efthymiopoulos, A. Christakidis, S. Denazis, and O. Koufopavlou , "Liquidstream – network dependent dynamic P2P live streaming," Springer Peer-to-Peer Networking and Applications, vol. 4, pp. 50-62, Mar. 2011, doi: 10.1007 /s12083-010-0092-0.

[3] A. Christakidis, N. Efthymiopoulos, J. Fiedler, S. Dempsey, K. Koutsopoulos, et al., "VITAL++ a new communication paradigm: embedding P2P technology in next generation networks," IEEE Communications Magazine, vol. 49, pp. 84-91, Jan. 2011, doi: 10.1109/MCOM.2011.5681020.

[4] D. Ciullo, V. Martina, M. Garetto, E. Leonardi, and G. L. Torrisi, "Stochastic Analysis of Self-Sustainability in Peer-Assisted VoD Systems," IEEE INFOCOM, pp. 1539-1547, Mar. 2012, doi: 10.1109/INFCOM.2012.6195521.

[5] PPLive. [Online]. Available from: http://pplive.en.softonic.com/

[6] D. Ciullo, V. Martina, M. Garetto, E. Leonardi, and G. L. Torrisi, "Performance analysis of non-stationary peer-assisted VoD systems," IEEE INFOCOM, pp. 3001-3005, Mar. 2012, doi: 10.1109/INFCOM.2012.6195747.

[7] Social Telemedia Environment for Experimental Research. [Online]. Available from: http://www.steer.ics.ece.upatras.gr

[8] Opnet modeler. [Online]. Available from: www.opnet.com

[9] J.J. Slotine, and W. Li, Applied Nonlinear Control. Prentice Hall, 1991.

[10] GridCast. [Online]. Available from: http://www.gridcast.cn

[11] PPStream. [Online]. Available from: http://www.ppstream.com

[12] TVU. [Online]. Available from: http://www.tvunetworks.com

[13] SopCast. [Online]. Available from: http://www.sopcast.com

[14] Y. Chen, B. Zhang, C. Chen, and D. M. Chiu, "Performance modeling and evaluation of peer-to-peer live streaming systems under flash crowds," IEEE/ACM Networking, vol. 22, pp. 1106–1120, Aug. 2014, doi: 10.1109/TNET.2013.2272056.

[15] C. Wu, B. Li, and S. Zhao, "Diagnosing network-wide P2P live streaming inefficiencies," ACM Multimedia, Computing, and Communications, vol. 8, Article No 13, Feb. 2012, doi: 10.1145/2089085.2089090.

[16] M. Efthymiopoulou, N. Efthymiopoulos, A. Christakidis, N. Athanasopoulos, S. Denazis, and O. Koufopavlou, "Scalable playback rate control in P2P live streaming systems", Springer Peer-to-Peer Networking and Applications, pp. 1-15, Sept. 2015, doi: 10.1007/s12083-015-0403-6.