Simulation of Buffering Mechanism for Peer-to-Peer Live Streaming Network with Collisions and Playback Lags

Yuliya Gaidamaka, Ivan Vasiliev,
Andrey Samuylov, Konstantin Samouylov
Telecommunication Systems Department
Peoples' Friendship University of Russia
Moscow, Russia
E-mail: ygaidamaka@sci.pfu.edu.ru, iuvasiliev@gmail.com, asam1988@gmail.com, ksam@sci.pfu.edu.ru

Sergey Shorgin
The Institute of Informatics Problems
The Russian Academy of Sciences
Moscow, Russia
E-mail: sshorgin@ipiran.ru

***Abstract*—In this paper, an approach to the peer-to-peer live streaming network simulation is presented. The model considers collisions because of limitation of peer's upload capability and takes into account the data transfer delays, which cause playback lags between peers. As a basis of simulation the mathematical model is considered, which describes in terms of discrete Markov chain the data exchange process between buffers of users in peer-to-peer network. Joint analysis of the two models — simulation and mathematical — leads to better understanding the impact of collisions and playback lags on playback continuity which is necessary when designing the effective peer-to-peer live streaming network.**

Keywords—P2P live streaming network; buffer occupancy; playback continuity; Markov chain model; playback lags

I. INTRODUCTION

Peer-to-peer (P2P) network is a kind of overlay content delivery network which consists of users who make their resources (computing power, memory, and bandwidth) available to other users without central coordination. In P2P networks, users not only download data, but also simultaneously distribute the downloaded data to other users, thus, peer-to-peer networking differs from client-server networking.

There are two types of P2P networks: file-sharing and streaming P2P networks [1][2]. In both cases, users download content as small blocks of data called chunks and each user downloads the missing chunks from other users, who have already downloaded them. In file-sharing networks (known as BitTorrent-like networks), users have to download the entire file before they begin to use it, so that a user is not restricted by time to obtain any chunk. In streaming networks, users simultaneously download and play the video stream, so a limit for download time of a chunk is crucial, since every chunk has its playback deadline. To provide smooth playback in streaming P2P networks the buffering mechanism is utilized. Each user has a buffer for caching the most recently downloaded data chunks. Moreover, only the chunks that are yet to be played will be downloaded. In both cases, in order to select which chunk to download next, a download strategy, such as Rarest First, Latest Useful Chunk First (LF), Greedy, Rarest Random, Naive Sequential, Cascading, and Hybrid download strategies, is applied [3][4].

Peer-to-peer network performance measures are usually analyzed via using different mathematical models. The so called fluid models are used to analyze file sharing networks [5-8]. One of the main performance metrics of file-sharing networks is how long it takes to download the whole file (the file download time or latency). Streaming P2P networks are stricter in respect to performance measures, and their distinctive feature is that they are generally analyzed in discreet time [9-16] with much attention paid to investigation of the buffering mechanism [11-16]. In streaming networks, the main performance measures are the startup delay (or latency), playback continuity (or skip-free playout probability) and the probability of universal streaming.

In this paper, a simulation model for analyzing the data exchange process in P2P live streaming network is presented. Like the model of [16], our model is built on the scheme of chunk exchange between peers' buffers introduced in [11]. The model takes into account limitation of peer's upload capability the result of which becomes violation of playback smoothness. The corresponding performance measure is probability of playback continuity, one of Quality of Experience (QoE) parameters in P2P networks. Unlike [11] and [16], our model also considers the playback lags between peers, which were first discussed in [12]. As a basis of simulation the mathematical model is considered, which describes in terms of discrete Markov chain the data exchange process between buffers of users in P2P live streaming network. The mathematical model is based on the model introduced in our previous work [14] and was modified for LF download strategy in order to take into account collisions and playback lags. Joint analysis of the two models — simulation and mathematical — provides advantages in the development of the algorithm for modeling and allows to improve accuracy of the calculations. Two main characteristics were investigated as a function of peer's upload capability — the probability of collision and the probability of playback continuity. By numerical example, the impact of collisions and playback lags on playback continuity is illustrated. Our main results are the following.

- The rigorous mathematical model of the download strategy in terms of a discrete Markov chain, that takes into account collisions and playback lags unlike [11] and [16].
- The exact formula for the index of peer's buffer position to download a chunk according to LF

 ISBN: 978-1-61208-318-6

download strategy considering the playback lags between peers unlike [13] and [14].

- The detailed algorithm of chunk exchange between buffers of peers considering collisions unlike [13] and [14].
- The exact formula for the probability of playback continuity considering the playback lags unlike [11-16].

This paper is organized as follows. In Section II, a video data distribution in a P2P live streaming network with buffering mechanism is described, and the mathematical model of the download strategy, in terms of a discrete Markov chain with a rigorous mathematical description, is considered. Also, the detailed algorithm of chunk exchange between buffers of peers in P2P live streaming network is developed and main performance measures are defined. In Section III, performance analysis and some case study is performed. The conclusion of this paper is presented in Section IV.

II. MODEL

In this section, a video data distribution in a P2P live streaming network with buffering mechanism is studied. Consider a P2P network with N users present in the network, and a single server, which transmit only one video stream. The process of video stream playback is divided into time slots, the length of each time slot corresponds to the playback time of one chunk. Each user has a buffer designed to accommodate M+1 chunks, where the buffer positions are numbered from 0 to M: 0-position is to store the freshest chunk just received from the server, other m -positions, $m=1,...,\mathrm{M}-1$, are to store chunks, already received during the past time slots or will be downloaded in the coming time slots, and buffer M-position is to store the oldest chunk that will be moved out from the buffer for playback during the next time slot.

Let us specify the actions that the server and users perform during each time slot. At the beginning of each time slot the server randomly selects a user from the network and uploads the newest chunk into his buffer 0-position. Any other user, not chosen by the server during the current time slot, will perform the following actions. If there are empty positions in the user's buffer (i.e., there are missing chunks in his buffer) the user will randomly choose another user, called a target user, from the predefined group of his neighbors in order to download one of the missing chunks from him. The number of chunks that a target user can upload is restricted by its upload capability. So, if the number of users that chose the same user as a target user exceeds the target user upload capability then a collision occurs. In case of collision, the number of users that successfully download missing chunks corresponds to the target user upload capability and the others don't download anything at all. If no collision occurs and the target user has one of the missing chunks, then the attempt to download from the target user will be successful. If the target user has more than one of the missing chunks, then download strategy will define which chunk to download. One of the simplest used strategies is LF strategy. With the LF strategy during any time slot each user tries to download the appropriate chunk with minimum index [12]. A user will not download any chunk in the current time slot at all, if in the current time slot all positions of his buffer are occupied (there are no empty positions) or if the target user he have chosen does not have any of the missing chunks. At the end of each time slot, chunks in the buffer of each user shift one step forward, i.e., the chunk in M-position moves to the player for playback, the buffer 0-position gets free to accommodate a new chunk from the server at the beginning of the next time slot. The remaining chunks in other positions shift one position to the right (towards the end of the buffer) to replace the position freed by its predecessor.

Below a mathematical model for chunk exchange between user's buffers is developed in the form of discrete Markov chain. The model of user behavior, proposed in [13][15], is extended by taking into account data transfer delays called playback lags that affect the video data exchange process between users as it is shown in Figure 1.

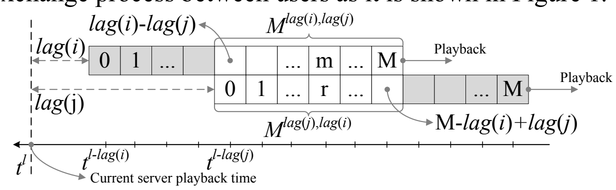

Figure 1. Buffers states mapping with playback lags

For a given network with N users and the single server, vector $\mathbf{z}(n)=\left(lag(n),u(n),\mathbf{x}(n)\right)$ defines the state of each user (n -user), where $lag(n)$ is the data transfer delay from server (playback lag), $u(n)$ is n -user upload capability and $\mathbf{x}(n)=\left(x_0(n),x_1(n),\cdots,x_\mathrm{M}(n)\right)$ is the state of n -user's buffer. Here $x_m(n)$ is the state of n -user's buffer m -position: $x_m(n)=1$, if n -user's buffer m -position is occupied with a chunk, otherwise $x_m(n)=0$, where m is the index of position in user's buffer, $m\in\{0,1,...,\mathrm{M}\}$. Each user in the network uses buffer positions $m=1,...,\mathrm{M}$ to store the chunks downloaded from the other users, and uses 0-position only to store the chunk downloaded from the server. Note that, if during any time slot M-position is occupied, then n -user watches the video stream without any pause.

Thus, the state of the system is defined by $\mathbf{Z}=(\mathbf{lag},\mathbf{u},\mathbf{X})$, where $\mathbf{lag}=\left(lag(1),...,lag(N)\right)$ and $\mathbf{u}=\left(u(1),...,u(N)\right)$ are vectors that define the playback lag, and the upload capability for each user, and the n -th row of the matrix $\mathbf{X}$ corresponds to the buffer state of n -user, $\dim\mathbf{X}=\mathrm{N}(\mathrm{M}+1)$.

Denote by $M^0\left(\mathbf{x}(n)\right)$ and $M^1\left(\mathbf{x}(n)\right)$ the set of indexes of all empty (1) and occupied (2) positions in n -user's buffer respectively:

$$M^0\left(\mathbf{x}(n)\right)=\left\{m:\ x_m(n)=0,\ m=1,...,\mathrm{M}\right\}, \qquad (1)$$

$$M^1\left(\mathbf{x}(n)\right)=\left\{m:\ x_m(n)=1,\ m=1,...,\mathrm{M}\right\}. \qquad (2)$$

Here $M^0\left(\mathbf{x}(n)\right)\subseteq\{1,...,\mathrm{M}\}$, $M^1\left(\mathbf{x}(n)\right)\subseteq\{1,...,\mathrm{M}\}$, and $M^0\left(\mathbf{x}(n)\right)\cup M^1\left(\mathbf{x}(n)\right)=\{1,...,\mathrm{M}\}$ is the set of indexes of all positions in n-user's buffer available for download from other user, not from the server. Note that due to playback lags not the entire buffer of a user is available for chunk exchange, see Figure 1. As in (3) for arbitrary i-user and j-user, $i,j\in\{1,...,\mathrm{N}\}$, the set $M^{lag(i),lag(j)}$ determines the indexes of i-user buffer positions that are available for chunk exchange with j-user:

$$M^{lag(i),lag(j)}=\begin{cases}\{0,1,\ldots,\mathrm{M}-lag(i)+lag(j)\}, & \text{if } lag(i)\ge lag(j),\\ \{lag(i)-lag(j),\ldots,\mathrm{M}\}, & \text{if } lag(i)<lag(j).\end{cases} \qquad (3)$$

Then, for n-user and h-user the intersection $M^0\left(\mathbf{x}(n)\right)\cap M^{lag(n),lag(h)}$ is the set of the indexes of n-user's empty buffer positions, one of which could be filled in with data from h-user. And the intersection $M^1\left(\mathbf{x}(h)\right)\cap M^{lag(h),lag(n)}$ is the set of the indexes of occupied positions in h-user's buffer, that h-user can upload corresponding chunk to n-user.

Due to data transfer delays, one and the same data chunk in the buffers of users with different playback lags is located in positions with different indexes. In order to establish a correspondence between these positions, the following operation is used: $m=r-lag(n)+lag(h)$. Here m is an index of buffer position for n-user, and r is a corresponding index of buffer position for h-user, $m\in M^{lag(n),lag(h)}$, $r\in M^{lag(h),lag(n)}$. Thereby, the index $m_{LF}\left(\mathbf{x}(n),\mathbf{x}(h),lag(n),lag(h)\right)$ of n-user's buffer position to which n-user according to LF download strategy should try to download a chunk from h-user is determined by the following formula:

$$\begin{aligned}&m_{LF}\left(\mathbf{x}(n),\mathbf{x}(h),lag(n),lag(h)\right)=\\&=\min\Big\{\left(M^0\left(\mathbf{x}(n)\right)\cap M^{lag(n),lag(h)}\right)\cap\\&\cap\Big\{\ m:m=r-lag(n)+lag(h),\\&r\in\left(M^1\left(\mathbf{x}(h)\right)\cap M^{lag(h),lag(n)}\right)\Big\}\Big\}.\end{aligned} \qquad (4)$$

Denote by $S\mathbf{x}(n)$ the shifting operator of vector $\mathbf{x}(n)$, meaning if $\mathbf{x}(n)=\left(x_0(n),x_1(n),\cdots,x_{\mathrm{M}-1}(n),x_{\mathrm{M}}(n)\right)$, then $S\mathbf{x}(n)=\left(0,x_0(n),\cdots,x_{\mathrm{M}-1}(n)\right)$. Let t_l be the shifting instant of buffer contents. When constructing the model in a discrete time, it is assumed that if at the instant t_l-0 a buffer is in the state $\mathbf{x}(n)$, then at the instant t_l+0 it will be in the state $S\mathbf{x}(n)$.

According to the protocol for the data distribution in P2P live streaming network with a buffering mechanism, in the interval $\left[t_l,t_{l+1}\right)$, which corresponds to the l-th time slot, the server and users perform the following actions.

1) At the instant t_l for all users the shift of the buffer content takes place:

a) Chunk in buffer M-position if present will be sent for playback;

b) All other chunks in other buffer positions will be shifted one position to the right, i.e., towards the end of the buffer;

c) Buffer 0-position will be emptied.

2) At the instant t_l+0 server chooses one user randomly and uploads a chunk for the current time slot to his buffer 0-position. If server has chosen i-user, then $x_0(i)=1$ at the instant $t_{l+1}-0$.

3) Each user (n-user), not chosen by the server, randomly chooses one of his neighbors (h-user). Let $C^l(h)$ be the number of users, which chose h-user as a target user at the l-th time slot.

a) If $C^l(h)\le u(h)$ (case "no collision") then n-user tries to download one of the missing chunks from h-user in its buffer's $m_{LF}\left(\mathbf{x}(n),\mathbf{x}(h),lag(n),lag(h)\right)$ position according to LF download strategy.

b) If $C^l(h)>u(h)$ (case "collision") then h-user chooses $u(h)$ users from $C^l(h)$ users randomly and each of chosen users tries to download one of the missing chunks from h-user in its buffer's $m_{LF}\left(\mathbf{x}(n),\mathbf{x}(h),lag(n),lag(h)\right)$ position according to LF download strategy. The other $C^l(h)-u(h)$ users go flop with downloading during the l-th time slot.

Denote by $\mathbf{Z}^l=\left(\mathbf{lag},\mathbf{u},\mathbf{X}^l\right)$ the network state at the instant t_l-0 and then the set $\left\{\mathbf{Z}^l\right\}:=\left\{\mathbf{Z}^l,\ l\ge 0\right\}$ forms a Markov chain over state space Ω with one class $\tilde{\Omega}$ of essential states, $\tilde{\Omega}\subset\Omega$. Let $\pi^l(\mathbf{Z})$ be the probability that Markov chain $\left\{\mathbf{Z}^l\right\}$ during l-th time slot is in state $\mathbf{Z}$, i.e., $\pi^l(\mathbf{Z})=P\{\mathbf{Z}^l=\mathbf{Z}\}$, $\mathbf{Z}\in\Omega$. The probability distribution $\pi^l(\mathbf{Z})$ is obtained in [14], the analytical formulas for calculating transition probability matrix of Markov chain $\left\{\mathbf{Z}^l\right\}$ is obtained in [15].

III. PERFORMANCE ANALYSIS AND SOME CASE STUDY

One of the main performance measures of P2P live streaming network is the probability $PV(n)$ of playback continuity, which is the probability that buffer M-position of n -user is occupied with the corresponding chunk for playback by the end of any time slot. To find this probability, the function

$$H_n^m(\mathbf{Z}) = \sum_{\substack{h=1,...,N,\\ h\neq n}} \delta_{m_{LF}(\mathbf{x}(n),\mathbf{x}(h),lag(n),lag(h)),m}\,, \quad (5)$$

$m=1,...,\mathrm{M}$, is introduced. The function $H_n^m(\mathbf{Z})$ corresponds to the number of h -users who have a chunk in their buffer m -position, from which n -user can download in accordance with the LF download strategy when the network is in the state $\mathbf{Z}\in\Omega$. Here

$$\delta_{i,j} = \begin{cases} 1, & i=j, \\ 0, & i\neq j. \end{cases} \quad (6)$$

Now, the probability $Q_n^l(m)$ that during the l-th time slot the chunk which n -user can download to his buffer m-position is available in the network is defined. Due to the dependency of this probability on the downloading strategy the function $Q_n^l(m)$ can be interpreted as the probability that n-user will select m-position and successfully download a chunk from the target user during the l-th time slot. If $N\geq 2$, then one can obtain the following formula:

$$Q_n^l(0)=0,$$

$$Q_n^l(m) = \frac{1}{N-1}\sum_{\mathbf{Z}\in\Omega}\pi^l(\mathbf{Z})\cdot H_n^m(\mathbf{Z})\,,\ m=1,...,\mathrm{M}. \quad (7)$$

Denote by $p_0^l(n,m)$ ($p_1^l(n,m)$) the probability that m-position of n-buffer is empty (occupied) during l-th time slot. Then, we can obtain a recursive relation for calculating the buffer state probabilities in a following form:

$$p_1^l(n,0)=1/N\,, \quad (8)$$

$$p_1^{l+1}(n,m+1) = p_1^l(n,m) + p_0^l(n,m)Q_n^l(m)\,,\ m=0,...,M-1\,.$$

Assume that the equilibrium distribution of the Markov chain $\{\mathbf{Z}^l\}$ exists. Denote by $p_1(n,m)=\lim_{l\to\infty}p_1^l(n,m)$ the probability that m-position of n-buffer is occupied and by $p_0(n,m)=\lim_{l\to\infty}p_0^l(n,m)$ the probability that m-position of n-buffer is empty, and $Q_n(m)=\lim_{l\to\infty}Q_n^l(m)$. Then, the following equation can be obtained:

$$p_1(n,0)=1/N\,,$$

$$p_1(n,m+1) = p_1(n,m) + p_0(n,m)Q_n(m)\,,\ m=0,...,M-1\,. \quad (9)$$

Thus, the probability $PV(n)$ that n-user is watching video without pauses during playback, i.e., the probability of playback continuity, is defined by the following formula:

$$PV(n) = p_1(n,M)\,. \quad (10)$$

Let us denote by $PC(h)$ the probability of collision for h -user, i.e., the situation when the number $C(h)$ of users that chose h -user as a target user exceeds the value of upload capability $u(h)$ of h-user. Thus, the formula is obtained:

$$PC(h) = \lim_{l\to\infty}P\{C^l(h)>u(h)\},\ h=1,...,N\,. \quad (11)$$

On the basis of the above results, a simulator was developed for analysis of a P2P live streaming network with following values of parameters: N=300, M=40, and the number of neighbors is equal to 60. It is assumed that all users have the same upload capability $u(h)=u, h=1,...,N.$ Therefore $PC(h)=PC,\ h=1,...,N.$

As it is shown in (12), the set of all users is split into three equal-sized non-overlapping groups for simplicity, assuming that the playback lags for all users in one group are the same, i.e.,

$$N=\bigcup_{k=1}^{3}N_k\,,\ lag(n)=lag(n')\,,$$

$$n,n'\in N_k\,,\ k=1,2,3\,. \quad (12)$$

The playback lag of the first group is set to zero and the playback lag of the second and third groups are 10 and 20 time slots respectively.

Then, the simulation was conducted according to the algorithm described in Section II. The simulation runs for a certain amount of simulation time equal to 1 000 000 time slots, as extending the simulation time did not affect the results. The statistics was gathered starting with the 50 000-th time slot in order to negate the non-steady state time interval.

The graphs in Figure 2 and Figure 3 show how the probability of collision and the probability of playback continuity depend on the user's upload capability. The corresponding 95% confidence intervals are not shown in the figures because of the scale. The confidence intervals are given in Table 1.

The graph in Figure 2 shows that the probability of collisions decreases with increasing the user's upload capability. The graphs in Figure 3 show that the users of the group with the largest value of the playback lag (the third group) have the greatest probability of watching video stream without pauses in playback, e.g., without freezes and reboots.

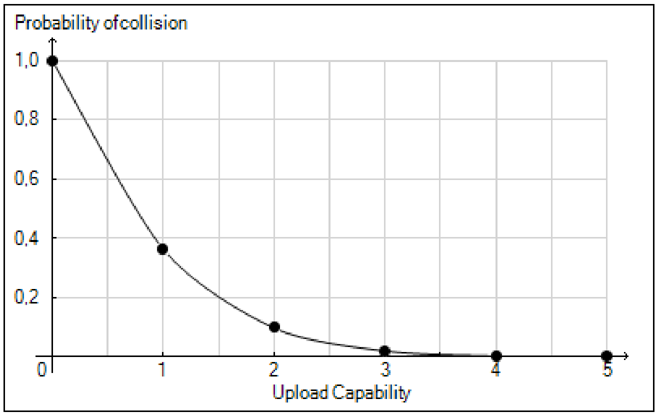

Figure 2. Probability of collision

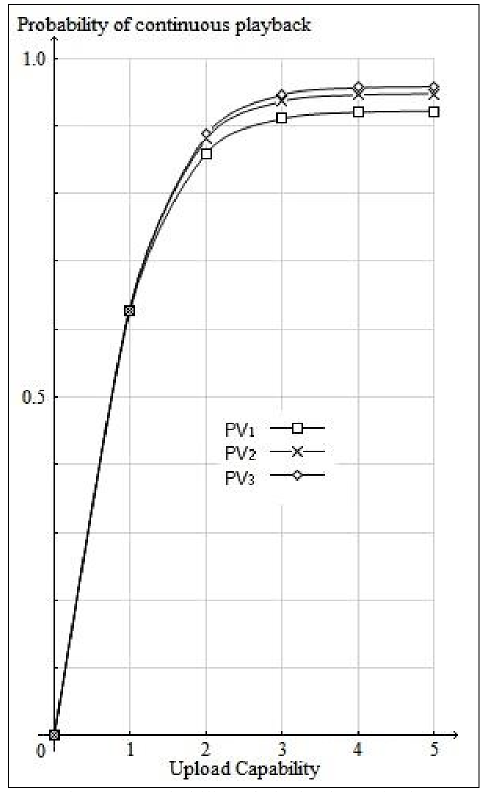

Figure 3. Probability of playback continuity

TABLE I. THE 95% CONFIDENCE INTERVALS

Upload capability u	***Probability of playback continuity*** PV_1	PV_2	PV_3	***Probability of collision*** PC
1	[0.627228, 0.628382]	[0.631887, 0.633776]	[0.631704, 0.633594]	[0.362913, 0.364799]
2	[0.85844, 0.859804]	[0.879786, 0.881058]	[0.885195, 0.886442]	[0.097177, 0.098341]
3	[0.909756, 0.910876]	[0.935353, 0.936314]	[0.945026, 0.945916]	[0.020048, 0.020601]
4	[0.919197, 0.920262]	[0.945085, 0.945974]	[0.955367, 0.956173]	[0.00343, 0.003663]
5	[0.920827, 0.921882]	[0.946741, 0.947618]	[0.957117, 0.957907]	[0.000486, 0.000577]

The reason is that any data chunk becomes highly available among users of the first and second groups by the time it is requested by the users of the third group.

IV. CONCLUSION

In this paper, the approach to simulation of the buffering mechanism in P2P live streaming network with collisions and playback lags is presented. The advantages of the mathematical model were used to develop the simulator for the performance evaluation of the QoE parameters including the probability of collision and the probability of playback continuity.

The direction of future research is simulation and comparison of most popular download strategies, such as Rarest First, Latest Useful Chunk First, and Greedy.

ACKNOWLEDGMENT

The reported study was partially supported by the Russian Foundation for Basic Research, research project No. 13-07-00953-a, 14-07-00090-a, and the Ministry of education and science of Russia, project 8.7962.2013.

REFERENCES

[1] E. Setton and B. Girod, "Peer-to-Peer Video Streaming. Springer," 2007, 150 p.

[2] Xuemin Shen, Heather Yu, John Buford, and Mursalin Akon, "Handbook of Peer-to-Peer Networking. Springer," 2010, 1421 p.

[3] Guangxue Yue, Nanqing Wei, Jiansheng Liu, Xiaofeng Xiong, and Linquan Xie, "Survey on Scheduling Technologies of P2P Media Streaming," Journal of Networks, Vol. 6, No. 8, August 2011, pp. 1129-1136.

[4] B. Fan, D. Andersen, M. Kaminsky, and K. Papagiannaki, "Balancing Throughput, Robustness, and In-Order Delivery in P2P VoD," In Proc. ACM CoNEXT, Dec. 2010.

[5] F. Clevenot and P. Nain, "A Simple fluid model for the analysis of the squirrel peer-to-peer caching system," Proc. of the IEEE INFOCOM, 2004, pp. 1-10.

[6] François Baccelli, Fabien Mathieu, and Ilkka Norros, "Performance of P2P networks with spatial interactions of peers. Networks and Telecommunications Networks, Systems and Services, Distributed Computing," Equipes-Projets GANG, TREC, Centre de recherche INRIA Paris, Rapport de recherche, n°7713, August 2011, pp. 1-23.

[7] Lei Guo, Songqing Chen, Zhen Xiao, Enhua Tan, Xiaoning Ding, and Xiaodong Zhang, "A performance study of BitTorrent-like peer-to-peer Systems," Proc. of the IEEE Int. Conf. on Communication, Vol. 25, № 1, 2007, pp. 155-169.

[8] R. Srikant and D. Qiu, "Modeling and performance analysis of BitTorrent-like peer-to-peer networks," Proc. of the ACM SIGCOMM Computer Communication Review, 2004, pp. 367-378.

[9] L. Kleinrock and S. Tewari, "Analytical model for bittorrent-based live video streaming," Proc. of the IEEE CCNC, 2007, pp. 976-980.

[10] R. Kumar, Y. Liu, and K. W. Ross, "Stochastic fluid theory for P2P streaming systems," Proc. of the IEEE INFOCOM, 2007, pp. 919-927.

[11] Yipeng Zhou, M. Dah, Chiu, and J. C. S. Lui, "A Simple Model for Analyzing P2P Streaming Protocols," Proc. of IEEE Int. Conf. IN Network Protocols (ICNP 2007), Oct. 19, 2007, pp. 226-235.

[12] Y. Zhao and H. Shen, "A simple analysis on P2P streaming with peer playback lags," Proc. of the 3rd International Conference on Communication Software and Networks (IEEE ICCSN 2011), May 27-29, 2011. Xi'an, China, pp. 396-400.

[13] A. Adamu, Yu. Gaidamaka, and A. Samuylov, "Analytical modeling of P2PTV network," Proc. of the 2nd International Congress on Ultra Modern Telecommunications and Control Systems (IEEE ICUMT 2010), Oct. 18-20, 2010, Moscow, Russia, pp. 1115-1120.

[14] A. Adamu, Yu. Gaidamaka, and A. Samuylov, "Discrete Markov Chain Model for Analyzing Probability Measures of P2P Streaming Network," Lecture Notes in Computer Science. Germany, Heidelberg: Springer. 2011. Vol. 6869. pp. 428-439.

[15] Yuliya Gaidamaka and Andrey Samuylov, "Analytical Modeling of Playback Continuity in P2P Streaming Network with Latest First Download Strategy" // Lecture Notes in Computer Science, Germany, Heidelberg, Springer-Verlag, 2013, Vol. 8121, pp. 363-370.

[16] L. Ying, R Srikant, and S Shakkottai, "The Asymptotic Behavior of Minimum Buffer Size Requirements in Large P2P Streaming Networks", Information Theory and Applications Workshop (ITA), 2010, pp. 1-6.