An optimal topology for a static P2P live streaming network with limited resources; static analysis, static and dynamic real-world results

Jonathan Stern*
RayV Research

Mira Gonen
Bar Ilan University

Omer Luzzatti†
RayV Research

Raphael Goldberg
RayV Research

Eran Weiss
RayV Research

ABSTRACT

In this paper we present a P2P live streaming topology, prove its optimality under common constraints, and match the analytical research with results from a running commercial network. We assume two types of nodes: viewers that consume the entire media, and amplifiers which are non-viewing nodes utilized for their upstream bandwidth. We analytically derive the minimum needed server upload capacity, for any topology, under the following assumptions: buffer time is limited, dynamics is low, and the total bandwidth required by the viewers exceeds the total upstream bandwidth of all peers. Then, we present a two-level topology and prove that it achieves the minimum possible server upload, up to a small fraction. Finally, the assumptions and derivation are backed up by performing several experiments on RayV's real-world commercial system, with varying network parameters. Namely, we show our predictions are valid while varying the viewers to amplifiers ratio, the stream bit-rate, and the country of the peers. These results not only verify the analytical static predictions but also evaluate the dynamic costs during the 'flash crowd' initial time when peers are joining the system.

1. INTRODUCTION

In recent years more and more users consume live streaming over the open Internet. However, providing a mass market live streaming service over the open Internet using the existing unicast solutions is highly inefficient. This inefficiency manifests as high operational costs and reduced scalability. These limitations can be overcome by using an approach based on peer-to-peer (P2P) technology, which can shift load from the server to the peers.

However, the P2P paradigm was initially designed for file sharing, which does not have the severe delay limitations of live streaming. Overcoming this limitation requires a new approach, including a thorough analysis of its abilities and limitations.

In recent years, several commercial live P2P streaming systems have emerged (like PPlive; www.pplive.com). The analysis presented in this paper is based on RayV's commercial solution (www.rayv.com).

Every analysis of a P2P system is performed with respect to one or more optimization goals and under various constraints. The division between goals and constraints varies between different works in the literature. (Li et al. [8])'s goal function is to minimize server upload while (Liu and Wu [11]) focus also on maximizing viewing quality. (Huang et al. [6]) attempt to minimize end-to-end delay, and (Sengupta et al. [15])'s goal is to maximize the supportable media streaming rate. As our system is commercially driven, our main goal is to reduce the distribution cost by minimizing server upload (maximizing peers upload contribution) while guaranteeing viewing quality for a certain bit-rate, and while not exceeding a reasonable delay.

Another basic aspect of a P2P solution is the topology of the grid. In the literature it is common to distinguish between two archetype solutions: 'tree-based' (e.g. *Dán* et al. [3]), and 'mesh-based' (e.g. [13]). Regardless of the topology, note that the flow of a specific data-chunk is never circular, which means it propagates as a directed acyclic graph (a tree). In a pure 'tree-based' solution, consecutive data-chunks tend to repeat the same paths, while in an ideal 'mesh-based' solution the paths of each data chunk are determined individually according to dynamically evaluated parameters.

Typically, tree-based solutions are 'predictable', in the sense that the data flow is relatively constant; however, these topologies tend to be sensitive to grid-dynamics due to the dependance of low layer nodes on the stability of their higher layer 'ancestors'. Herein we propose a well-defined topology, in which each node can rely on

*jonathan.stern@rayv.com
†omer@rayv.com

the amount of data it will receive, while minimizing the sensitivity to dynamics. This is achieved by a shallow two layer multi-tree approach and by the utilization of an information dispersal algorithm (IDA; [14]) to minimize packet duplications.

Another aspect to be addressed when designing a P2P solution is the tendency of the peers to join and leave the grid and the frequency in which they do so. This behavior is also known as peer dynamics. An extreme, though common, dynamic scenario is the tendency of peers to join the grid in a short period of time (typically at the beginning of a show) also known as 'flash crowd' (Liu et al. [10]). We measure the effect of this scenario on our system and its dependence on the parameters of the grid. These results can serve as a basis for an analytical description of a highly dynamic system.

Verifying analytical descriptions of P2P live systems is commonly done by simulations. One drawback of this approach is that it is highly dependant on the presumptions of the analysis. Our results are derived from testing on a commercial live P2P system, therefore unrealistic assumptions are kept to a minimum.

1.1 Our Contributions

We begin with a theoretical analysis of a P2P system under our assumed constraints (§2). After discussing the basic concepts, we add amplifiers (non-viewing relay nodes; Sengupta et al. [15]) and derive the maximal possible P2P for any static topology. Then, we propose a specific topology, and prove it is near-optimal. In order to validate our theoretical analysis we perform various tests on RayV's real-world system. The static and dynamic results of these tests are presented in §3. Our conclusions and suggested future research are given in §4.

2. THEORETICAL ANALYSIS

2.1 Preliminaries

We define two types of nodes, *Viewers* and *Amplifiers*. *Viewers* are nodes that download the whole stream and contribute their upload. Let N_{V} be the number of viewers in the network. *Amplifiers* are non-viewing nodes with available upload bandwidth that do not need to download the entire stream. Let N_{A} be the number of amplifiers in the network. The total number of nodes in the network is denoted by $N(\equiv N_{\mathrm{V}} + N_{\mathrm{A}})$. The *server* is the source of the entire live stream. Usually, in the real-world, the server receives the live stream from an encoder as it becomes available, and broadcasts it to the peers. For the purpose of this research, we ignore this division between the encoder and the server. Also, we assume that the server upload capacity is unlimited.

Let R be the bitrate of the broadcasted stream (in kbps). Let C_i be the upload capacity of peer i, and C be the average capacity of all peers. We assume C_i is independent of the specific acceptors a certain donor has, a common assumption in the literature. Quantitative analysis of this assumption is delayed to future work. By U_{s} we denote the data rate sent from the server (also in kbps).

Our goal is to minimize U_{s}. In order to measure the cost efficiency of our system we define the P2P ratio:

$$\mathit{P2P} = \mathit{1} - \frac{U_{\mathrm{s}}}{N_{\mathrm{V}} \cdot R}. \tag{1}$$

Namely, P2P equals one minus the server upload normalized by the stream data rate multiplied by the number of viewers. This normalization factor is equivalent to the server upload in a uni-casting solution which does not utilize peer upload, and therefore is a good benchmark to evaluate P2P effectiveness.

2.2 System Constraints

- **Available Upload**
  As noted by Kumar et al. [7], a crucial parameter describing the system is the relative difference between the total upload capacity of the peers and the required download. We denote this parameter by β:
  $$\beta = 1 - \frac{C}{R} \cdot \frac{N}{N_{\mathrm{V}}} \tag{2}$$
  If $\beta \leq 0$ then the peers can provide all the needed upload of the system. If $\beta > 0$, β represents the minimum normalized needed U_{s} in order to assure all viewers can view the stream. Thus, regardless of the topology used, $U_{\mathrm{s}} \geq \beta \cdot N_{\mathrm{V}} \cdot R$. We denote the case $\beta > 0$ by *under*, and the case $\beta \leq 0$ by *over*.

  Clearly, the server needs to upload the stream at least once, so $U_{\mathrm{s}} \geq R$. Therefore we get that
  $$\mathit{P2P} = \mathit{1} - \frac{U_{\mathrm{s}}}{N_{\mathrm{V}} \cdot R} \leq \mathit{1} - \max\left\{\frac{\mathit{1}}{N_{\mathrm{V}}}, \beta\right\} \tag{3}$$
  Then,
  $$\mathit{P2P}_{max} = \mathit{1} - \max\{\frac{\mathit{1}}{N_{\mathrm{V}}}, \beta\} \tag{4}$$

- **Maximum Buffer Time**
  As discussed in the introduction we assume a constraint in which the live streaming system has a maximum possible delay between broadcast time and viewing time. Typically this time is from 3 to 10 seconds. We denote this delay as D_{buffer}.

- **Packet Size**
  In principle, sending larger packets increases the efficiency of the system (protocol overheads, per-packet delays, etc). However, the maximum transfer unit (MTU) is an infrastructure upper limit

one does not want to surpass. Over the Ethernet, MTU size is 1500 bytes (including protocol overhead). For these reasons and the simplicity of the analysis, we define that each packet includes $P = 10kbits$ (1250 bytes) of *data*, and, aligned with industry standards, we don't analyze the option that this size can be changed.

2.3 P2P$_{max}$ with Amplifiers

By Equation 2, in a viewer-only system $\beta = 1 - \frac{C}{R}$. Current typical values of R requested by content owners range from 800 kbps to 2000 kbps. Current values for C range from 150 kbps for common ADSL networks, to 1500 kbps for common fiber networks. Therefore, current typical viewer-only systems have $\beta > 0$. As shown in Equation 3, when $\beta > 0$, reducing β enlarges P2P$_{max}$. A possible improvement is to add non-viewing nodes - amplifiers. Using the upload of the amplifiers, the total available upload capacity is increased by $N_\text{A} \cdot C$, and β is reduced appropriately. However, in order to fully utilize the upload of the amplifiers, they need to constantly receive relevant data, thus exploiting some fraction of the system's upload capacity.

Denote by s the download rate per amplifier.

LEMMA 2.1. *The minimum value of s, $s_{\min}$, is* $\frac{P}{D_{buffer}}$

Proof: Since every packet becomes obsolete after D_{buffer}, an amplifier must receive at least 1 packet every D_{buffer}, namely, its minimum download rate is $\frac{P}{D_{buffer}}$. $\square$

THEOREM 2.1. *The optimal configuration of amplifiers, when $\beta > 0$, is that amplifiers receive their data directly from the server.*

Proof: If an amplifier receives data directly from the server, U_s is increased by s. Alternatively, if an amplifier receives the data from any other node, that node's available upload is reduced and thus the amount of total upload available for the viewers is also reduced by s. In an *under* situation ($\beta > 0$), where deficient bandwidth is compensated by the server, this causes U_s to increase also by s. Therefore a configuration in which amplifiers download their data directly from the server is an optimal configuration. $\square$

A direct consequence of theorem 2.1 is that regardless of the topology, amplifiers increase U_s by at least $N_\text{A} \cdot s_{\min}$. Therefore, when the system contains amplifiers and $\beta > 1/N_\text{V}$, a more accurate expression for P2P$_{max}$ is:

$$P2P_{max} = 1 - \beta - \frac{N_\text{A} \cdot s_{\min}}{N_\text{V} \cdot R} = \frac{C \cdot N - N_\text{A} \cdot s}{R \cdot N_\text{V}}. \quad (5)$$

Note that if $C_i < s$ for amplifier i, then this amplifier can only reduce P2P, and therefore should not be used.

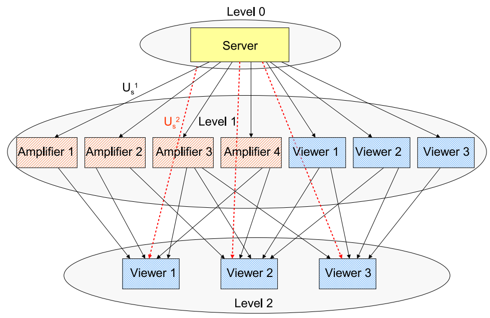

Figure 1: 2-level grid

2.4 2-Level Grid

We propose a topology for the P2P system and then prove its optimality under common conditions.

Description of the 2-level grid: Figure 1 depicts the 2-level grid. The nodes are presented in two levels. Nodes in the first level receive packets from the server (level 0) and forward them to their acceptors in level 2. All amplifiers and viewers are present in the first level; viewers also populate the second level. Whenever the viewers cannot receive the entire stream from the first level, they complete the missing data directly from the server. We denote the server upload to the first level by U_s^1 and to the second level by U_s^2.

The stream is divided into segments of size Sz packets, P bytes each, where $Sz = D_{buffer} \cdot \frac{R}{P}$. The server pushes to every 1st-level node one packet per segment, which is then forwarded to the node's 2nd-level acceptors. Therefore, $s = \frac{R}{Sz}$, which by Lemma 2.1 equals to $s_{\min}$. Note that for a grid with $D_{buffer} = 3$ sec and $R = 1000$ kbps we get $Sz = 300$. Thus, in a common under scenario,

$$\beta > \frac{N}{N_\text{V} \cdot Sz} \quad (6)$$

for any reasonable $\frac{N}{N_\text{V}}$.

To minimize the duplication of data we use an Information Dispersal Algorithm (IDA), such as Network Coding (NC), see [1, 5, 4, 2, 9, 12]. This enables us to assume that the amount of unnecessary packets each viewer receives is $\delta \cdot Sz$ per segment, where δ depends on the specific IDA that is used[1].

We next prove that the 2-level grid described above is near-optimal:

THEOREM 2.2. *A 2-level grid maximizes P2P up to δ.*

Proof: Since $s = \frac{R}{Sz}$, $U_s^1 = \frac{R}{Sz}N$. U_s^2 equals the remaining data viewers haven't directly received from the

[1]For example, in NC over GF2 with $Sz = 50$, $\delta = 0.04$ (see §3).

server in U_s^1 and from their level-1 donors. The number of packets the viewers got directly from the server is $\frac{R}{Sz} \cdot N_{\rm V}$, and the data they got from level-1 nodes is $C \cdot N$. Therefore,

$$U_s^2 = \left(R(1+\delta) - C \cdot \frac{N}{N_{\rm V}} - \frac{R}{Sz} \right) \cdot N_{\rm V} \tag{7}$$

and Equation 6 ensures that $U_s^2 > 0$. Thus,

$$\begin{aligned} P2P &= 1 - \frac{U_{\rm s}}{N_{\rm V} \cdot R} = 1 - \frac{U_s^1 + U_s^2}{N_{\rm V} \cdot R} \\ &= 1 - \frac{(R/Sz) \cdot N}{N_{\rm V} \cdot R} \\ &- \frac{\left(R(1+\delta) - C \cdot \frac{N}{N_{\rm V}} - \frac{R}{Sz} \right) \cdot N_{\rm V}}{N_{\rm V} \cdot R} \\ &= 1 - \frac{N}{N_{\rm V}} \cdot \frac{1}{Sz} \\ &- \left(1 + \delta - \frac{C}{R} \cdot \frac{N}{N_{\rm V}} - \frac{1}{Sz} \right) \\ &= 1 - \left(\frac{N}{N_{\rm V}} - 1 \right) \cdot \frac{1}{Sz} - (1+\delta) \\ &+ \frac{C}{R} \cdot \frac{N}{N_{\rm V}}. \end{aligned} \tag{8}$$

Replacing $\frac{R}{Sz}$ with $s_{\rm min}$, Equation 8 implies that

$$\begin{aligned} P2P &= 1 - \left(\frac{N}{N_{\rm V}} - 1 \right) \cdot \frac{s_{\rm min}}{R} - (1+\delta) \\ &+ \frac{C \cdot N}{R \cdot N_{\rm V}} \\ &= \frac{-N \cdot s_{\rm min} + s_{\rm min} \cdot N_{\rm V} + C \cdot N}{R \cdot N_{\rm V}} - \delta \\ &= \frac{C \cdot N - N_{\rm A} \cdot s_{\rm min}}{R \cdot N_{\rm V}} - \delta \\ &= P2P_{max} - \delta. \end{aligned} \tag{9}$$

□

3. SYSTEM TESTS

In order to solidify the assumptions and derivations in this paper, we conducted a performance analysis of a commercial P2P live-streaming system. We use RayV's commercial P2P streaming solution, a solution that broadcasts 200 active channels daily, with hundreds of millions minutes watched a month, to more than 100 countries. The results of the analysis are preceded by a description of the system.

3.1 System Description

Our system topology is based on the 2-level grid described in §2.4. When a new viewer joins the grid, its IP is enlisted at the server. Then, it receives from the server IPs of other listed peers. Upon receiving the peer list, the viewer initiates connections in which it is the acceptor. Thus, acceptors are always viewers, while donors are both either viewers or amplifiers. As seen in Figure 1, acceptors are in the 2nd level of the grid, while donors are in the 1st level. The peer list is updated by the server which sends every $L = 60$ seconds[2] a fresh peer list to each of the viewers.

When a new amplifier joins the grid, it simply connects to the server and waits for incoming connection requests.

When a 1st level node has acceptors, it receives packets from the server, and pushes them to all of its acceptors. The logic of 1st level nodes is such that whenever they do not have acceptors they stop fetching data from the server.

To effectively distribute the data, we use a simplified Network Coding technique in which only the server generates linear sums of the data packets over $GF2$. The stream of data is divided into segments of size Sz packets, where each packet is 10 kbits long. We term the segment that is currently being distributed the 'current segment' and the raw packets that it includes 'base packets'. As described in §2.4, for every segment the server sends one packet to each 1st level node. This packet is a random linear sum (over $GF2$) of the base packets of the current segment, together with the coefficients. Thus, there are $2^{Sz} - 1$ possibilities for each computed packet. When a 1st level node receives the computed packet, it pushes it to its 2nd level acceptors. When a 2nd level viewer receives all the packets from its 1st level donors, it checks whether reconstructing the original data is possible. This is done by checking whether the degree of the coefficients matrix is greater than Sz. Thus, if matrix inversion is possible, the original data is restored. Otherwise, needed base packets are requested directly from the server, and the original data is reconstructed upon their arrival. From information theory, the expected number of packets needed to receive Sz *independent* packets is $\lesssim S{\rm z} + 2$, so in our system $\delta \approx 2/Sz$, for the δ defined in §2.4.

We use $D_{\rm buffer} = 10$ seconds, which is significantly larger than the grid propagation time[3]. As mentioned in §2.2, $P = 10$ kbits, therefore by Lemma 2.1 and the fact that $s = \frac{P}{D_{\rm buffer}}$, the maximum possible Sz for a 1200 kbps stream is 1200. However, for practical CPU constraints due to the requirements of using NC over $GF2$ we are limited to Sz of 60-80. As can be seen in Equation 5, the P2P cost of this compromise is

$$\Delta{\rm P2P} = \frac{\rm N_A}{\rm N_V Sz} - \frac{\rm N_A}{\rm N_V \cdot 1200} \approx \frac{\rm N_A}{\rm N_V Sz} \tag{10}$$

For $S{\rm z} = 60$ and $\frac{N}{N_{\rm V}} = 2$, ΔP2P=0.017. This is the

[2] As shown below, L is significantly lower than the grid build-up time. Therefore, the results are robust to moderate changes in L.

[3] The propagation time in a 2-hop grid rarely exceeds 300 milliseconds

maximum ΔP2P for the tests described below.

An additional mechanism makes sure that 1^{st} level donors accept connection requests only as long as they do not experience losses in their upload. If a donor's upload capacity is C_{i}, it will start experiencing losses when more than $\frac{C_{\text{i}}}{R/S\text{z}}$ acceptors are connected to it. Therefore, to avoid losses, once this number of outgoing connections is reached, the donor will deny further connection requests. Furthermore, we limit the number of acceptors per donor to $C_{\max}$. In the *under* scenario discussed in this paper (when resources are limited), amplifiers often reach their capacity.

3.2 Test Description

In order to test the accuracy of our derivations and relevance of our assumptions, we ran several tests on our system. In each test we used N_{V} viewers and $N_{\text{A}}(\equiv N - N_{\text{V}})$ amplifiers from a specific country. Each test was run for *circa* 40 minutes with a media-program with bit-rate R. All nodes are instructed to enter the grid together, creating a 'flash crowd' scenario as common in real broadcasting, and creating a significant dynamic strain on the system. Calculating the *actual* P2P (Equation 1) is simple, one only needs to know the number of connected viewers (N_{V}) and to measure the total server upload (U_{s}). For calculating the *optimal* P2P under the available resources (Equation 5), the average upload capacity of the nodes in the grid needs to be measured. As mentioned above, the capacity of a node is defined as the maximum upload the node can handle without experiencing losses. Therefore, we can get a good evaluation of the true capacity only when a donor has indeed reached its upload limit.

3.3 Results

The first three tests were run on USA nodes, with $R = 1200$ kbps and $C_{\max} = 800$ kbps, with $N/N_{\text{V}} = 1.$, 1.25 & 1.5. Figure 2 presents the results of these three tests. All measurements were taken in 90-second intervals. The upper panel of each test shows P2P vs. time, with the actual P2P (Equation 1) shown as a black line, and the optimal P2P (Equation 5) as a gray area. The range in optimal P2P is due to donors whose capacity is unknown, since they have not experienced loss in their upload. We expect their mean capacity, C_{unknown}, to be larger than C_{known} (the mean capacity of peers that have experienced loss). Therefore, the gray area depicts the area which satisfies $C_{\text{known}} < C_{\text{unknown}} < C_{\max}$. Note that if our assumption that $C_{\text{unknown}} > C_{\text{known}}$ is incorrect, our actual P2P gets even closer to the optimal P2P.

As can be seen, after the initial 700 seconds, the static analysis is quite adequate. In fact, the actual P2P is only 5-8% below the optimal P2P. Between the 3 panels, P2P indeed increases with additional amplifiers which enlarge the total available upload. As mentioned in §2.3, Equation 5 provides the maximum possible P2P using *any* topology. Therefore, Figure 2 demonstrates that our system operates at within 8% of the optimum for the given amount of upload resources.

The lower panel of each test presents the number of viewers and amplifiers in the grid as a function of time. Notice that nodes tend to leave the experiment before it ends, where viewers tend to leave more than amplifiers. This departure causes an increase in N/N_{V}, which in turn causes a rise in P2P. However, we see that the cost of the dynamics is low, and P2P remains close to the optimum.

In Figure 3 R was changed to 1600 kbps, and N/N_{V} was altered between 1., 1.5 & 2. As before, after the grid stabilizes, the static P2P Equation is a good measurement of the behavior of the grid. Again, our grid is operating near the theoretical optimum, though somewhat farther from it at high P2P (right panel). The optimality of the grid is probably somewhat reduced when β (§2.2) is close to 0.

In Figure 4, the results of a test on South Korean nodes is shown. Their strong FTTH (fiber to the home) infrastructure allows peers to reach much higher upload. For this test we changed $C_{\max}$ to 1600 kbps. As can be seen, even without amplifiers the P2P ratio on a 1600 kbps stream reaches 0.7.

Table 1 summarizes the 7 tests we performed. The variance in C_{known} of the 6 USA tests is small - the standard deviation of C_{known} is only 5% of the mean value. This solidifies the notion that the mean capacity is a well-defined characteristic of the nodes, and can be used for evaluating the performance of a grid. Also, we see that the mean measured capacity of the South-Korean nodes is almost twice the USA capacity. This is an expected result knowing the nature of the different infrastructures and only partially an outcome of the change in $C_{\max}$.

3.3.1 Dynamics

From Figures 2, 3 and 4, one can see that during the initial highly dynamic part of the test, P2P is much lower than predicted by the static analysis and that after a certain time (denoted as τ_{FC}) the line follows the static predictions. Also, the rise in P2P during build-up of the grid seems to be linear with time. For example in the left panel of Figure 2, τ_{FC} seems to be around 700 seconds. In order to consistently evaluate this build-up duration, we fit a broken line (4 parameters: slope 1, slope 2, τ_{FC} and y-intercept) to the measured P2P data points, using a standard least-squares algorithm and a common error in all points.

As can be seen in table 1, τ_{FC} is ≈ 730 seconds in the USA tests, and ≈ 1000 seconds in the South Korean test. This insinuates that grid stabilizing time grows

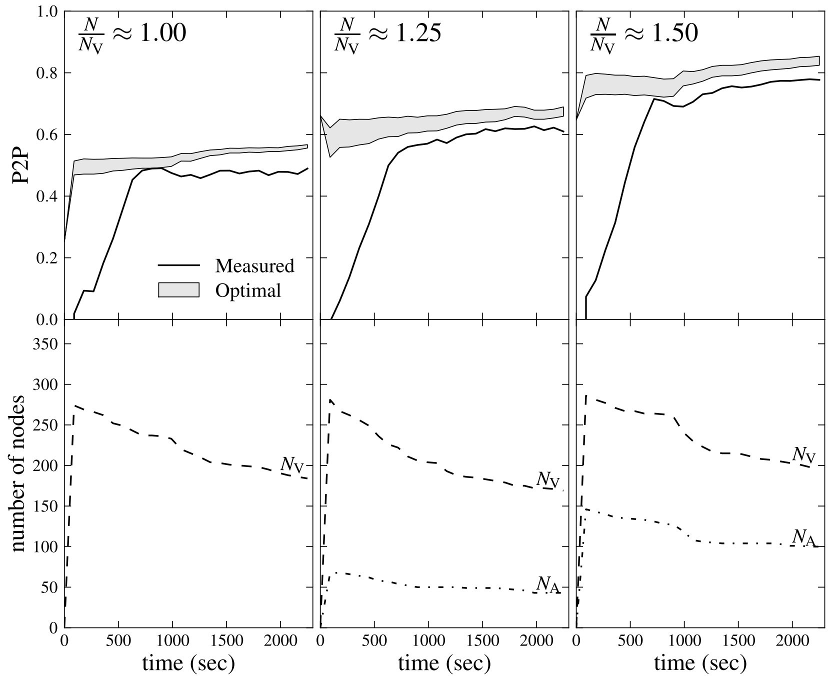

Figure 2: $R = 1200$ **kbps, USA**

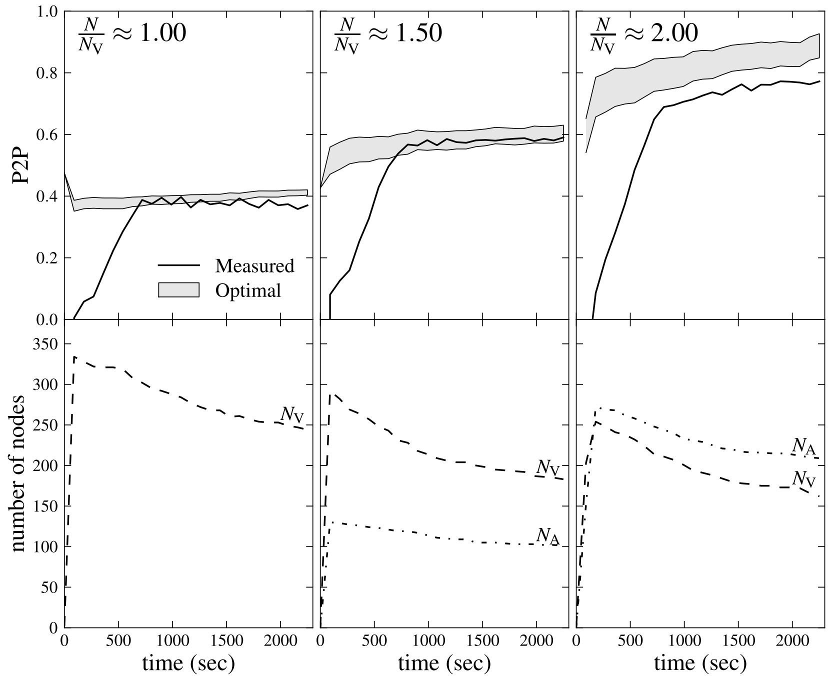

Figure 3: $R = 1600$ **kbps, USA**

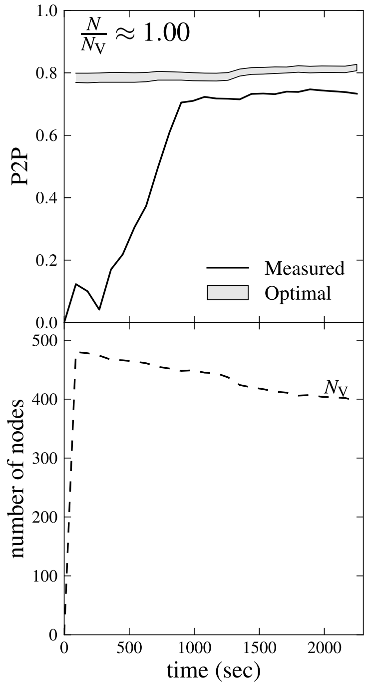

Figure 4: $R = 1600$ kbps, South Korea

Country	$C_{\max}$ (kbps)	R (kbps)	N/N_V	C_{known} (kbps)	τ_{FC} (s)	P2P	$\text{P2P}_{t>\tau_{\text{FC}}}$
USA	800	1200	1.00	582	719	0.39	0.48
USA	800	1200	1.25	525	747	0.48	0.60
USA	800	1200	1.50	592	695	0.61	0.74
USA	800	1600	1.00	573	731	0.30	0.38
USA	800	1600	1.50	542	750	0.47	0.58
USA	800	1600	2.00	531	775	0.59	0.74
Korea	1600	1600	1.00	1275	1017	0.54	0.73

Table 1: Summary of the performed tests

with the mean capacity, and not with the viewer to amplifier ratio or the bit-rate of the broadcasted stream. We plan to further investigate this result in future research.

Also, from the figures, one can measure the relative cost in server upload due to the dynamics, by comparing the average P2P of the entire test to the average P2P after the system stabilized. This comparison is a lower limit to the total dynamic costs, because some dynamics are also present during the stable period. In table 1 both measures are given. Note that the ratio between them depends on the length of the broadcasted program, which is 40 minutes in the above tests. A thorough analysis of grid dynamics and an analytical derivation of its cost will be the focus of future work.

4. CONCLUSIONS AND FUTURE WORK

In this paper we proved that for achieving the minimum possible server upload in a static network under limited resources, a two-level grid topology suffices. We matched the analytical research with a running commercial network using different viewing to non-viewing nodes ratios and different stream rates. These tests were performed on the highly different infrastructures of the US and South-Korea. Also, we were able to deduce a quantitative evaluation of the cost and duration of the common flash-crowd scenario.

In future work, we plan to analyze the case of $\beta < 0$ ('over'). Also, we intend to further analyze the dynamical costs of the system and their dependence on the grid's parameters.

5. REFERENCES

[1] Chou, P. A., Wu, Y., and Jain, K. Practical network coding. In *Microsoft Research* (2003).

[2] Chu, X., Zhao, K., Li, Z., and Mahanti, A. Auction based on-demand p2p min-cost media streaming with network coding. *IEEE transactions on Parallel and Distributed Systems 20* (2009), 1816–1829.

[3] Dan, G., Fodor, V., and Chatzidrossos, I. On the performance of multiple-tree-based peer-to-peer live streaming. In *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM)* (2007), pp. 2556 – 2560.

[4] Feng, C., and Li, B. On large-scale peer-to-peer streaming systems with network coding. In *Proceedings of the 16th ACM International Conference on Multimedia* (2008).

[5] Gkantsidis, C., and Rodriguez, P. Network coding for large scale content distribution. In *Proceedings of the 24th IEEE International Conference on Computer Communications (INFOCOM)* (2005), pp. 2235 – 2245.

[6] Huang, F., Ravindran, B., and Kumar, V. An approximation algorithm for minimum-delay peer-to-peer streaming. In *IEEE Ninth International Conference on Peer-to-Peer Computing* (2009), pp. 71 – 80.

[7] Kumar, R., Liu, Y., and Ross, K. Stochastic fluid theory for p2p streaming systems. In *Proceedings of the 26th IEEE International Conference on Computer Communications (infocom)* (May 2007), pp. 919 – 927.

[8] Li, J., Chou, P., and Zhang, C. Mumutualcast: an efficient mechanism for one-to-many content distribution. In *Proceedings of the ACM SIGCOMM* (2005).

[9] Li, Z., Li, B., Jiang, D., and Lau, L. On achieving optimal throughput with network coding. In *Proceedings of the 24th IEEE International Conference on Computer Communications (INFOCOM)* (2005), pp. 2184 – 2194.

[10] Liu, F., Li, B., Zhong, L., and Li, B. Understanding the flash crowd in p2p live video streaming systems. In *Proceedings of the 17th International Packet Video Workshop* (May 2009), pp. 1 – 10.

[11] Liu, H., and Wu, I. Metree: A contribution and locality-aware p2p live streaming architecture. In *IEEE International Conference on Advanced Information Networking and Applications (AINA)* (2010), pp. 1136 – 1143.

[12] Lun, D., Ratnakar, N., Koetter, R., Medard, M., Ahmed, E., and Lee, H. Achieving minimum-cost multicast: a decentralized approach based on network coding. In *Proceedings of the 24th IEEE International Conference on Computer Communications (INFOCOM)* (2005), pp. 1607 – 1617.

[13] Qiu, T., Nikolaidis, I., and Li, F. On the design of incentive-aware p2p streaming. *Journal of Internet Engineering 1* (2007), 61–71.

[14] Rabin, M. O. Probabilistic algorithms in finite fields.

[15] Sengupta, S., Liu, S., Chen, M., Chiang, M., Li, J., and Chou, P. P2p streaming capacity, TechReport number: Msr-tr-2009-68.