Fluid Theory for Hierarchical P2P Streaming Networks

Bartosz Polaczyk, Piotr Cholda*

AGH University of Science and Technology, Department of Telecommunications, Krakow, Poland

*piotr.cholda@agh.edu.pl

Abstract-The paper extends the well-known fluid model derived by Kumar et al. for Peer-to-Peer (P2P) streaming systems mainly by taking into account such practical aspects like: heterogeneity of resources, limitation of server capacity, hierarchical structure of an underlying network and mesh topology of an overlay. The mathematical analysis on the maximum streaming rate that can be achieved is presented. The algorithmic steps to achieve a universal streaming under various system parameters are proposed. In addition, a proof that the universal video distribution streaming rate, which is achievable under such circumstances, is given. Specifically, by using the fluid model, it is shown how the server allocates its bandwidth to each peer in the system, and how each peer divides its limited upload bandwidth to its local neighbors and global peers.

Keywords- Fluid Theory; Peer-to-Peer (P2P); Performance Evaluation; Overlays; Reliability

I. INTRODUCTION

This paper analyzes the maximum achievable rate of a multimedia stream, e.g., video content, in a Peer-to-Peer (P2P) network. The used model takes into consideration several practical aspects of an overlay network:

- heterogeneous peers resources,
- limited server upload capacity,
- hierarchical structure of a network (e.g., autonomous systems and their sub-networks, where the peers are located),
- chunk-enabled communication (that is, it is suitable for a popular mesh topology),
- uniform video rate for all end-users.

The model is based on derivation of the well-known work by Kumar et al. [1]. The derived model provides a simple relation between the number of peers and their resources with the maximum achievable rate of streaming the content that can be distributed between them. The results provided by the model can be used by Internet Service Providers (ISPs), content providers, protocol designers and automated end-user applications. In general, it gives a view on an overlay network condition and locates its bottlenecks. Finding a bottleneck location gives a great opportunity for optimization and efficient network management.

We predict that our model can be used for optimization of P2P video streaming applications that use the underlay awareness (i.e., they know the IP network transferring P2P flows). A classification of such approaches can be found [2].

Currently, the majority of existing implementations and methods are related to the second type [5] [6] [7] [8] [9] [10]. This stems from their flexibility and high fault tolerance. Thus, we focus on this type of the P2P overlays in this paper. An important challenge related to P2P networks is the optimal topology, which satisfies each of the three entities involved in the overlay content delivery: end-users, ISPs, and content providers. Each of them has different requirements and prominent objectives. A user requires high streaming rate of content, short start-up times and lack of pre-fetching.

In particular, if live video transmission is addressed, a small delay between a content source and an end-user is vital. While the inter-domain traffic is essential to ISPs, a reduction of that load should not affect the quality experienced by the customers. Finally, content provider goals are in-between the end-users and ISPs’ needs. It appreciates a high customer satisfaction (enhanced video quality) and favorable network resource utilization as it enables a successful cooperation with the operators.

Recently, a lot of effort was put into improving the operation of P2P systems, by taking into account the network constraints. Sometimes, even a strong cooperation between an overlay operator and a network carrier is assumed [11]. The topic has been recognized as an important issue for the Internet community by establishing an IETF working group known as...
Application-Layer Traffic Optimization (ALTO) [12]. The research was focused on file-sharing mainly, with introduction of methods like biased neighbor selection [13][14][15] or biased unchokeing [16] in the BitTorrent context. They rely on centrally managed devices, which impedes dynamic and quick decisions in case of nodes departure or network partitions. Therefore, their application prospects in video multicast networks are limited, especially when failures are considered. Currently, the optimization methods are also introduced to video streaming, although it has not been as popular field so far. Generally, the operators’ viewpoint is adopted; however, sometimes the optimization proposals aim at user performance improvement as well. The optimization methods are frequently based on delay measurements. Interestingly to mention, the underlay-awareness is used in some of popular overlay networks operating over the Internet, but not necessarily being a subject of the published research. The measurement studies [10][17] show that several streaming P2P systems (e.g., PPStream, TVAnts and PPLive) apparently exploit localization methods. However, the mechanism of this operation is not clear, since the software code and protocols are not open.

This paper focuses on some theoretical bounds related to the optimization of P2P systems using the underlay awareness. The structure of the work is very simple: in the next section, basics on the so-called fluid model of P2P flows are given, and then in Section III the problem is formulated, our basic theorem is stated and then proved. The last section gives some conclusions, mainly related to the limitations of the presented model.

II. FLUID MODEL

An approach of a fluid model is well studied for file-sharing overlays. Some publications concern the maximum streaming rate in for P2P applications as well. A notable example of such a work is the paper by Kumar et al. [1]. They give a simple relation between the maximum achievable streaming rate and upload limits of peers/the server. However, they assume that all peers are connected with each other using an ideal backbone (no loss and no delay) and the only constraint is an upload limit of an individual peer. The extension we present here introduces, as happens in the Internet, a hierarchical structure, where peers are clustered within sub-networks. A comparison between hierarchical and non-hierarchical topologies is presented in Figure 1 and shows the core of our extension. In the fluid model, it is assumed that the server sends some chunks to randomly selected peers and they distribute these contents to others (that lack the respective chunks). The upload rate of both the server and peers is limited as in the real networks.

![Comparison between topology used in [1] and the one analyzed in the paper](image)

Thus, we assume the presence of the two layers: the infrastructure (first layer) and the sub-networks (the sub-layer). The bandwidth of both layers is infinite, and the only bottlenecks are peers’ upload bandwidth and the sub-networks access bandwidth. A summary of the used notation is given in Table 1.
Table I: Notation Used in This Work

<table>
<thead>
<tr>
<th>Not.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_i$</td>
<td>an access link upload limit of sub-network $i$</td>
</tr>
<tr>
<td>$u_s$</td>
<td>upload limit of server</td>
</tr>
<tr>
<td>$(j, i)$</td>
<td>peer $j$ in sub-network $i$</td>
</tr>
<tr>
<td>$u_{j,i}$</td>
<td>a local link upload limit of peer $(j, i)$</td>
</tr>
<tr>
<td>$U(P_i)$</td>
<td>sum of all $u_{j,i}$’s inside sub-network $i$</td>
</tr>
<tr>
<td>$n_i$</td>
<td>a number of peers in sub-network $i$</td>
</tr>
<tr>
<td>$N$</td>
<td>a number of sub-networks</td>
</tr>
<tr>
<td>$r_{max}$</td>
<td>the internal rate of sub-network $i$</td>
</tr>
<tr>
<td>$T$</td>
<td>a set of sub-networks</td>
</tr>
<tr>
<td>$H$</td>
<td>a set of sub-networks for which $\forall_{j \in H} r_{max, j} \leq r_{max}$ and $\forall_{j \in H} r_{max, j} &gt; r_{max}$</td>
</tr>
<tr>
<td>$h$</td>
<td>cardinality of set $H$</td>
</tr>
<tr>
<td>$U(T)$</td>
<td>a sum of all $U(P_i)$ that belong to $T$</td>
</tr>
<tr>
<td>$u(T)$</td>
<td>a sum of $u_i$ that belong to $T$</td>
</tr>
<tr>
<td>$n(T)$</td>
<td>a sum of $(n_i - 1)$ that belong to $T$</td>
</tr>
<tr>
<td>$l_{ij}$</td>
<td>amount of extra data for node $(j, i)$ received from $H$</td>
</tr>
<tr>
<td>$l_i$</td>
<td>a sum of $l_{ij}$ inside sub-network $i$</td>
</tr>
<tr>
<td>$s_{ij}$</td>
<td>amount of extra data for node $(j, i)$ received from the server</td>
</tr>
<tr>
<td>$s_i$</td>
<td>a sum of $s_{ij}$ inside sub-network $i$</td>
</tr>
<tr>
<td>$e_{ij}$</td>
<td>amount of extra data from $(j, i)$, $i \in H$ to all nodes in $H$</td>
</tr>
<tr>
<td>$e_i$</td>
<td>a sum of $e_{ij}$ inside sub-network $i$</td>
</tr>
<tr>
<td>$e_H$</td>
<td>sum of all $e_i$ in $H$; the total extra data received from $H$</td>
</tr>
<tr>
<td>$E$</td>
<td>a sum of all $s_i$ in $H$; the total extra data received from the server</td>
</tr>
<tr>
<td>$E_T$</td>
<td>total extra data received from the server and peers</td>
</tr>
</tbody>
</table>

III. PROBLEM FORMULATION

Due to the link asymmetry, we limit only upload bandwidth of links taking into account that the download is significantly larger than the upload. A similar assumption is taken in [1] [18] [19]. A sub-network is a part of a network that is connected to a backbone with a single link called the access link. Each sub-network $P_i$, $i \geq 1$, contains $n_i$ end-nodes, where $n_i > 1$. We index a single peer with couple $(j, i)$, which means that it is a $j$th node in sub-network $i$. Each peer has limited upload connection bandwidth $u_{j,i}$ over its link called a local link. Sum of all local links in a sub-network $i$ is denoted as $U(P_i)$:

$$U(P_i) = \sum_{j \in P_i} u_{j,i}. \hspace{1cm} (1)$$

Sub-network $i$ has a limited upload bandwidth to the backbone, denoted as $u_i$. For the remainder of this paper we assume that the access link is a bottleneck for the nodes attached to the sub-network:

$$U(P_i) > u_i. \hspace{1cm} (2)$$

If the access link of a sub-network does not introduce any bottlenecks and all peers from the sub-network can be directly connected to a backbone, this is a trivial case and we do not deal with it.

The streaming server is connected directly to a backbone with upload limit $u_s$. The set of all sub-networks (not taking into account the one with the streaming server node) is defined as $\mathbb{N}$, and $N = \mathbb{N} \setminus \{H\}$ corresponds to a number of sub-networks. Each set of sub-networks $T \subset \mathbb{N}$ has its complementary set $\overline{T}$ defined as $\overline{T} = \mathbb{N} \setminus T$. The functions on subset of $\mathbb{N}$, i.e., $U(\cdot)$, $u(\cdot)$, $n(\cdot)$ are defined as follows:

$$U(T) = \sum_{P_i \in T} U(P_i), \hspace{1cm} (3)$$

$$u(T) = \sum_{i \in T} u_i, \hspace{1cm} (4)$$

$$n(T) = \sum_{i \in T} (n_i - 1), \hspace{1cm} (5)$$

and inform about an aggregated upload limit of all (a) local links, (b) access links, and (c) a number of all nodes in a set $T$. 

- 204 -
diminished by its size $|T|$, respectively.

A streaming for which each node receives a full copy of the stream is depicted as a universal streaming with rate $r$. Otherwise, if at least one node does not receive a full stream, such a streaming is called to be in a degraded service mode. In this model, we do not assume any churn, all peers are active and do not leave the swarm.

**Definition 1**: The internal rate of sub-network $P_i$, $r_{\text{max}_i}$ is defined as a maximum rate that can be distributed for the sake of sub-network $Pi$, when it receives only one full copy of a stream with rate $\max_i r$ and fully utilizes its access link. Additionally $r_{\text{max}_u} = 0$ and $r_{\text{max}_s,u} = \infty$.

**Lemma 1**: The internal rate of a sub-network $P_i$, $i \in \{1, \ldots, N\}$ is equal to:

$$r_{\text{max}_i} = \frac{U(P_i) - u_i}{n_i - 1}. \quad (6)$$

**Proof**: We can transform a situation presented in Definition 1 into a case, where we have one virtual streaming server with a flexible upload limit $r_{\text{max}_i}$ (it sends in total only one copy of the stream) and there is a sink that has to be filled with sub-streams with the aggregated rate of $u_i$. Theorem 1 [1] states that the maximum achievable streaming rate $r_{\text{max}}$ with the streaming server rate $u_i$ is given by:

$$r_{\text{max}} = \min \left\{ u_i, \frac{u_i + U(P)}{n_i} \right\}. \quad (7)$$

By (a) replacing $u_i$ with $r_{\text{max}_i}$, (b) omitting minimum function, and (c) reducing an aggregated upload by $u_i$ in the numerator, we have:

$$r_{\text{max}_i} = \frac{U(P_i) + r_{\text{max}_i} - u_i}{n_i}, \quad (8)$$

$$r_{\text{max}_i} (n_i - 1) = U(P) - u_i, \quad (9)$$

$$r_{\text{max}_i} = \frac{U(P_i) - u_i}{n_i - 1}. \quad (10)$$

To accomplish the maximal streaming rate, the nodes can follow the algorithm described below: the virtual streaming server with the stream rate $r = r_{\text{max}_i}$ divides a stream into $n_i$ sub-streams with rates $s^{(i)}_{ji}$:

$$s^{(i)}_{ji} = \left( \frac{u_{ij} - u_{ij} U(P)}{U(P_i) - u_i} \right) r, \quad (11)$$

and sends each sub-stream to a corresponding node $(j, i)$. It can do that, because the total upload that it uses is as follows:

$$\sum_{j=1}^{n_i} \left( \frac{u_{ij} - u_{ij} U(P)}{U(P_i) - u_i} \right) r = r \frac{U(P) - u_i}{U(P_i) - u_i} - r. \quad (12)$$

Each node that receives a sub-stream from a virtual stream resends this sub-stream to all its $n_i - 1$ neighbors. Furthermore, it transmits data to the access link (sink) with rate $s^{(2)}_{ji} = \frac{u_{ij} U(P)}{U(P)}$. It can do that, because its total upload is equal to:

$$(n_i - 1) \left( \frac{u_{ij} - u_{ij} U(P)}{U(P_i) - u_i} \right) r + u_{ij} U(P_i) - u_{ij} U(P) + u_{ij} U(P) = u_{ij}. \quad (13)$$

Furthermore, an aggregated rate of the all streams that are sent over the access link is equal to:
A sum of all sub-streams that node \((j, i)\) receives is:

\[
\sum_{j=1}^{n_j} \frac{U_{j,i}}{U(P)} = u_i.
\]  

(14)

QED.

In the reminder of the paper we assume that all sub-networks are sorted in a non-decreasing order of \(r_{\max, i}\). We will also use a non-strict term sub-network capability to denote the maximum possible internal rate of a sub-network. When the sub-network capability is high, it means that a particular sub-network has a high ability to distribute a full copy of the stream standalone.

As a set \(H \subset N\), we denote a set of sub-networks for which \(r_{\max, i} \leq r_{\max}\), where \(r_{\max}\) is the maximum streaming rate that allows the universal streaming. Therefore,

\[
\forall_{i \in H} r_{\max, i} \leq r_{\max, i}.
\]  

(16)

\[
\forall_{i \in H} r_{\max, i} > r_{\max}.
\]  

(17)

Now, we can give a core theorem of our study:

**Theorem 1:** The maximum achievable streaming rate, \(r_{\max}\), that enables the universal streaming is given by:

\[
r_{\max} = \min \left\{ u_s, \min_{r \in [0, \ldots, 1]} \frac{U(V) + u(V) + u_i}{\sum n_i - 1 + N} \right\}.
\]  

(18)

**Proof:** Proof of the Theorem 1 consists of two sections. Section I has to define the maximum rate that allows the universal streaming and Section II gives an algorithm that distributes a copy of a stream with the given rate. The algorithm presented in Section II proves that the upper bound given in Eq. (18) is attainable.

**Section I:** Let us consider two conditions, when all sub-networks have a relatively (a) big and (b) small sub-network capability in comparison to the server. In the former case, when each sub-network has upload rate \(U(P)\) sufficient to perform the universal streaming, the total upload \(U(P)\) does not introduce a bottleneck in a maximum-rate transmission. Then, each sub-network receives only one instance of the stream (which is always necessary) and redistributes the stream on its own. We can model such a case with a network, where we have:

- a streaming server with upload \(u_s\),
- \(N\) sub-networks treated as single nodes with upload limit \(u_i\).

From the formula presented in [1] we can state, that the maximum rate is given by:

\[
r_{\max} = \min \left\{ u_s, \frac{u_i + u(N)}{N} \right\}.
\]  

(19)

Such a case converges with a final maximum streaming rate, when this rate is bounded by limits of the server and access links. In the latter case, when we have a low sub-network capability in the overlay network, the access links do not form a bottleneck. Therefore, we can model the network as a structure with:

- a streaming server with upload rate \(u_s\),
- nodes that are connected directly to the backbone with upload rate \(u_{i,j}\).

From the formula presented in [1] we can state, that the maximum rate is given by:

\[
r_{\max} = \min \left\{ u_s, \frac{u_i + U(N)}{\sum n_i} \right\}.
\]  

(20)
The two upper bounds presented above in Eqs. (19)-(20) correspond to a situation when: (a) all sub-networks have a very high aggregate upload $U(P)$ corresponding to a low number of members $n_i$ and a low access limit $u_i$, such that each sub-network can distribute content on its own sake; and (b) when aggregated upload $U(P)$ is relatively small and with the access link of rate $r_{max}$, the limits $u_i$ do not form a bottleneck.

For interim cases, when only some of sub-networks can distribute the stream within themselves and others cannot, we need to define a set of sub-networks, $H$, where each sub-network has the maximum sub-network rate $r_{max}$ smaller than the streaming rate $r$. $H = |H|$ denotes a size of $H$, which is equal to the highest index of sub-network $i$ that belongs to $H$.

It is clear that each sub-network $i \in N$ has to receive at least one instance of the stream over an access link. As $l_i$ we denote an additional, redundant datium that is received by a sub-network $i$ that allows for the universal streaming in a less capable sub-network. From definition, sub-networks in $H$ cannot distribute a full copy of the stream on their own sake, therefore they receive extra data from $H$ and/or the streaming server. We call this process helping and the total extra datium that flows to $H$ is defined as $E$. $E$ consists of $E_H$ and $E_s$, which are sums of all extra data provided by a sub-network $H$ and the server, respectively:

$$E = E_H + E_s = \sum_{i \in H} e_i = \sum_{j \in H} l_j + E_s,$$

(21)

where $e_i$ is extra data received by sub-network $i$. Similarly, $e_{ji}$ is extra data received by node $(j,i), i \in H$. From definition of $e_i$ :

$$e_i = \sum_{j \in H} e_{ji},$$

(22)

$E_s$ is a volume of a server uploads, thus:

$$E_s = u_s - r = (u_s - r)f + (u_s - r)(1 - f),$$

(23)

where $f$ corresponds to a fraction of the server help that is dedicated to $H$. A fraction of the server help dedicated for $\overline{H}$ is $(1-f)$. However, to obtain the maximum rate universal streaming we have to set $f=1$, because there is no reason to send extra data to a well capable sub-networks $i \in \overline{H}$.

$s_{ji}$ is a rate of extra data that is provided to node $(j,i)$ by the server. $s_i$ is a sum of all $s_{ji}$ in a sub-network $i$:

$$s_i = \sum_{j \in H} s_{ji},$$

(24)

Observation 1: To obtain the maximum streaming rate $r_s$, which allows for universal streaming, each sub-network $i$ that has its internal rate $r_{max}$ smaller than $r$, has to receive extra data (help) from other sub-networks and cannot provide extra data for other sub-networks.

Proof: The fact, that each sub-network $i$ has $r_{max} < r$ is obvious, since otherwise it would not be able to perform the universal streaming (it goes directly from definition of the maximum sub-network rate). Furthermore, a sub-network that needs help and simultaneously provides extra data for others just diminishes the maximum possible streaming rate $r$. QED.

Observation 2: The maximum rate that can be distributed standalone by a single sub-network $i$, when it fully utilizes $u_i$ and receives/sends extra data, is limited by $r$, such that:

$$r = \begin{cases} \frac{U(P) - u_i + l_i + s_i}{n_i - 1} & \text{if } i \text{ belongs to } H \\ \frac{U(P) - u_i}{n_i - 1} & \text{otherwise, if } i \text{ belongs to } \overline{H} \end{cases}$$

(25)

Proof: The aggregated upload dedicated for all peers in sub-network $i \in H$ is a sum of:

- all uploads of the nodes,
- rate of the original stream,
• help provided to sub-network \(i\) \((l_i)\) from subset \(H\),
• help provided to sub-network \(i\) \((s_i)\) from the server,
and is diminished by:
• its upload rate \((u_i)\).

Therefore, using Lemma 1 we can state that the possible stream for the universal streaming \(r\) is bounded by:

\[
r < \frac{U(P) + l_i + s_i - u_i}{n_i - 1}
\]  
(26)

The aggregated upload dedicated for all peers in sub-network \(i \in H\) is a sum of:
• all uploads of the nodes,
• the rate of the original stream, and is diminished by
• its upload rate \((u_i)\).

Therefore, a possible stream for the universal streaming \(r\) is bounded by:

\[
r < \frac{U(P) - u_i}{n_i - 1}
\]  
(27)

QED.

The maximum rate with the universal streaming formula given in Eq. (25) for all sub-networks in \(H\) is equal to the constraints determined by Eqs. (28a)-(28c) and the same as the rate of the original stream (the constraint in Eq. (28d)):

\[
r = \frac{U(P) + l_i + s_i - u_i}{n_i - 1}
\]  
(28a)

\[
r = \frac{U(P) + l_2 + s_2 - u_2}{n_2 - 1}
\]  
(28b)

\[
r = \frac{U(P) + l_{H} + s_{H} - u_{H}}{n_{H} - 1}
\]  
(28c)

\[
r = \frac{u(H) + u(H) - E_{u} + u_i - E_s}{N}
\]  
(28d)

From Eqs. (28a)-(28c) we have:

\[l_i + s_i = r(n_i - 1) - U(P) + u_i.\]  
(29)

\[E_u + E_s = r n(H) - U(H) + u(H).\]  
(30)

Combining the above equations with Eq. (28d) we have:

\[
r = \frac{u(H) + u(H) + u_i - r n(H) + U(H) - u(H)}{N},
\]  
(31)

\[
r = \frac{U(H) + u(H) + u_i - r n(H)}{N},
\]  
(32)

\[
r = \frac{U(H) + u(H) + u_i}{n(H) + N}.
\]  
(33)

The formula given in Eq. (33) is an equivalent to a situation, when all nodes located in the sub-networks from \(H\) are connected directly to a backbone, and sub-networks \(H\) are replaced by a single node with a maximum upload \(u_i\) (see Figure 2). The main hurdle is a fact, that we do not know value of \(H\), i.e., the number of sub-networks that define the set \(H\).
Now, let us observe the right hand side of the expression given in Eq. (33). With a small notation abuse, we will define set of sub-networks \( T \) as a number \( q \), which means that \( T = \bigcup_{i=1}^{q} i \). Furthermore as \( r_i(q) \) we will consider a value that allots \( r \) from Eq. (33), when a cardinality of set \( H \) is equal to \( q \):

\[
    r_i(q) = \frac{U(q) + u_i(nq) + u_i}{n(q) + N}. \tag{34}
\]

Assume, that the correct value of \( H = \{ H | \) fulfills constraints (16) and (17), is equal to \( h \), yet unknown.

**Lemma 2:** \( r_i(i) \) has the smallest value when \( i \) is equal to \( h \).

**Proof:** First, let us consider \( i = h + x, x \geq 0 \Rightarrow i \geq h \). \( r_i(h + x) \) can be written as:

\[
    r_i(h + x) = \frac{U(h) + U(P_{x=1}) + \ldots + U(P_{x=q}) + u_i - u_{x=1} - \ldots - u_{x=q} + u_i}{n(h) + \sum_{k=1}^{x}(n_k - 1) + N}. \tag{35}
\]

Now, by replacing \( U(h) + u_i(nh) \) in (35) as \( r_i(h)(nh + N) \) from the definition in Eq. (34) we have:

\[
    r_i(h + x) = \frac{r_i(h)(nh + N) + \sum_{k=1}^{x}(U(P_k) - u_i)}{n(h) + \sum_{k=1}^{x}(n_k - 1) + N} = r_i(h) + \frac{-r_i(h)\sum_{k=1}^{x}(n_k - 1) + \sum_{k=1}^{x}(U(P_k) - u_i)}{n(h) + \sum_{k=1}^{x}(n_k - 1) + N}. \tag{36}
\]

From definition of \( H \), Eq. (17), and Eq. (25) we know, that for \( i > h \), \( \frac{U(P_i) - u_i}{n_j - 1} \geq r_i(h) \) thus \( U(P_i) - u_i \geq r_i(h)(n_j - 1) \). Applying it to (36) we obtain:

\[
    r_i(h + x) \geq r_i(h) + \frac{-r_i(h)\sum_{k=1}^{x}(n_k - 1) + r_i(h)\sum_{k=1}^{x}(n_k - 1)}{n(h) + \sum_{k=1}^{x}(n_k - 1) + N} = r_i(h). \tag{37}
\]

Now, let us consider that \( i = h - x, x \leq 0 \Rightarrow i \leq h \). \( r_i(h - x) \) can be written as:

\[
    r_i(h - x) = \frac{U(h) - U(P_{x=1}) - \ldots - U(P_{x=q}) + u_i + u_{x=1} + \ldots + u_{x=q} + u_i}{n(h) - \sum_{k=1}^{x}(n_k - 1) + N}. \tag{38}
\]
Similarly as in the previous case, we will replace \( U(h) + n(h) + N \) as \( r_c(h)(n(h) + N) \) from Eq. (34) and we obtain:

\[
r_c(h + x) = \frac{r_c(h)(n(h) + N) - \sum_{i=1}^{\pm \infty} n_i + u_i + \ldots + n_j + u_j}{n(h) - \sum_{i=1}^{\infty} (n_i - 1) + N} = r_c(h) + \frac{r_c(h) \sum_{i=1}^{\infty} (n_i - 1) - \sum_{i=1}^{\infty} (U(P_i) - u_i)}{n(h) - \sum_{i=1}^{\infty} (n_i - 1) + N}.
\]

(39)

On the basis of the fact, that \( \forall i < h \) we know that \( \frac{U(P_i) - u_i}{n_i - 1} \leq r_c(h) \), thus \( U(P_i) - u_i \leq r_c(h)(n_i - 1) \). We can bound Eq. (39) as:

\[
r_c(h - x) \geq r_c(h) + \frac{r_c(h) \sum_{i=1}^{\infty} (n_i - 1) - r_c(h) \sum_{i=1}^{\infty} (n_i - 1)}{n(h) + \sum_{i=1}^{\infty} (n_i - 1) + N} = r_c(h).
\]

(40)

From Eqs. (37) and (40) we see that for all \( i \neq h \), \( r_c(i) \) has a higher value than \( r_c(h) \). From this fact, we can state that the minimum value of \( r_c(i) \) corresponds to \( r_c(h) \). QED.

Furthermore, it can be proved that \( r_c \) is a monotonic function in intervals \((-\infty; h) \) and \((h; \infty) \), but for brevity this lemma is omitted. An example of the \( r_c \) function plot is presented in Figure 3.

![Fig. 3 Example graph of \( r_c(i) \) when \( h = 4 \)](image)

**Lemma 3:** For all values \( q \geq h \) and \( q \leq N \), \( r_c(q) \leq r_{\text{max},c} \).

**Proof:** From definition, \( r_{\text{max},c} \) is defined as:

\[
r_{\text{max},c} = \frac{U(P_{\text{q}+1}) - u_{\text{q}+1}}{n_{\text{q}+1} - 1}.
\]

(41)

By rewriting Eq. (34) and replacing \( U(P_{\text{q}+1}) - u_{\text{q}+1} \) from Eq. (41) we have:

\[
r_c(q) = \frac{U(q+1) - U(P_{\text{q}+1}) + u(q+1) + u_{\text{q}+1} + u_i}{n(q) + N} = \frac{U(q+1) + u(q+1) + u_i - r_{\text{max},c}(n_{\text{q}+1} - 1)}{n(q) + N}.
\]

(42)

\( U(q+1) + u(q+1) + u_i \) is equal to \( r_c(q + 1)(n(q + 1) + N) \) from the definition of \( r_c(q + 1) \), so:

\[
r_c(q) = \frac{r_c(q + 1)(n(q + 1) + N) - r_{\text{max},c}(n_{\text{q}+1} - 1)}{n(q) + N} = r_c(q + 1) + \frac{r_c(q + 1)(n_{\text{q}+1} - 1) - r_{\text{max},c}(n_{\text{q}+1} - 1)}{n(q) + N}.
\]

(43)

By Lemma 2 from a case when \( q \geq h \) we can state that \( r_c(q) \leq r_c(q + 1) \). Therefore, the second element in Eq. (43) has to be negative. Thus,
From a fact that \( r_s(q) \leq r_s(q+1) \) and Eq. (44) we see that:

\[
\begin{align*}
r_s(q+1) - r_{\text{max},i} &\leq 0 \\
r_s(q+1) &\leq r_{\text{max},i}.
\end{align*}
\]  

(45)

QED.

Lemma 4: For all values \( q \leq h \) and \( q \geq 1 \): \( r_s(q) \geq r_{\text{max},i} \).

Proof: From definition, \( r_s(q-1) \) and \( r_{\text{max},i} \) are equal respectively as follows:

\[
\begin{align*}
r_s(q-1) &= \frac{U(q-1) + u(q-1) + u_i}{n(q-1) + N} \\
r_{\text{max},i} &= \frac{U(P_s) - u_i}{n_i - 1}.
\end{align*}
\]  

(46a) (46b)

We can rewrite Eq. (46a) and replace \( U(P_s) - u_i \) from Eq. (46b) as:

\[
\begin{align*}
r_s(q-1) &= \frac{U(q) - U(P_s)}{n(q-1) + N}.
\end{align*}
\]  

(47)

From (34) we have that \( U(q) + u(q) + u_i = r_s(q)(n_q - 1) \), so:

\[
\begin{align*}
r_s(q-1) &= \frac{r_s(q)(n_q(q) + N) - r_{\text{max},i}(n_q - 1)}{n(q-1) + N} = r_s(q) + \frac{r_s(q)(n_q - 1) - r_{\text{max},i}(n_q - 1)}{n(q-1) + N}.
\end{align*}
\]  

(48)

From Lemma 2, when \( q \leq h \) we know that \( r_s(q-1) \geq r_s(q) \), thus the second component in Eq. (48) is positive. Therefore,

\[
r_s(q)(n_q - 1) - r_{\text{max},i}(n_q - 1) \geq 0 \Rightarrow r_s(q) \geq r_{\text{max},i}.
\]  

(49)

QED.

As a result of Lemmas 2-4 we can conclude, that finding the minimum value of \( r_s(i) \) gives a case when \( i = h \). Moreover, we have showed that only such the case determines the maximum rate of the universal streaming, because only when \( i = h \) we have \( r_s(i) \geq r_{\text{max},i} \) and \( r_s(i) \leq r_{\text{max},i} \) which are necessary to fulfill Eq. (16) and Eq. (17).

Note that the minimum value of \( r_s(i) \) satisfies a bound given in Eq. (19) and Eq. (20), because \( r_s(0) \) corresponds to Eq. (19) and \( r_s(N) \) corresponds to Eq. (20).

Section II: In this section we have to show, that the universal streaming with the maximum rate given by Theorem 1 is possible to obtain. A specific algorithm is sketched.

Case 1: \( r > \frac{u(N)}{N-1} \).

Let us introduce \( \xi > 0 \) such that:

\[
\xi = r - \frac{u(N)}{N-1}.
\]  

(50)

At the first step, the streaming server sends disjoint sub-streams to each node \((j,i)\) with streaming rate \( s_{i,j}^{(1)} \):

\[
s_{i,j}^{(1)} = \frac{u(P_s)}{(N-1)U(P_s)}.
\]  

(51)

The total amount of uploaded data from the server is smaller than the total streaming rate:
Next, each node uploads its sub-stream \( s_{ji}^{(1)} \) to other sub-networks in such a way that each member in another sub-network \( k \) \((k \neq i)\) has a unique part of sub-stream \( s_{ji}^{(1)} \). Therefore, the upload rate from node \((j, i)\) to node \((l, k)\) noted as \( s_{j, i}^{(2)} \) is equal to:

\[
s_{j, i}^{(2)} = s_{ji}^{(1)} U_{jk} = \frac{u_{ji} U_{ji}}{(N-1)U(P_i)} U_{jk}.
\]

The total amount of upload that passes the access link of sub-network \( i \) does not exceed \( u_i \) because the sum can be expressed as in:

\[
\sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{N} \frac{u_{ji} U_{ji}}{(N-1)U(P_i)} U_{jk} = \sum_{i=1}^{N} \frac{u_{ji} U_{ji}}{(N-1)U(P_i)} \sum_{k=1}^{N} U_{jk} = \sum_{i=1}^{N} \frac{u_{ji} U_{ji}}{U(P_i)} = u_i.
\]

The streaming server, which sent sub-streams with an aggregated rate \( r - \xi \), for each sub-network \( i \) divides the remaining data into sub-streams with rate \( s^{(3)}_{ji} = \frac{u_{ji}}{U(P_i)} \) and sends it to a corresponding node \( i \). The streaming server can do that, because from a fact that \( r < \frac{u(N)}{N} + u_i \), it is known that its aggregated upload does not exceed \( u_i \):

\[
r - \xi + N\xi = r + r(N-1) - u(N) =
\]

\[
rN - u(N) < u(N) + u_i - u(N) = u_i.
\]

**Observation 3:** At that point, each node in a network received a unique part of a stream across the sub-network, and the aggregated rate of the received sub-streams is equal to \( r \frac{u_{ji}}{U(P_i)} \).

**Proof:** Since all sub-streams \( s_{ji}^{(1)}, s_{j, k \rightarrow j, i}^{(2)}, \) and \( s_{ji}^{(3)} \) are disjoint across a given sub-network that node \((j, i)\) belongs to, it has a unique part of the stream within sub-network \( i \). Furthermore, the total rate of the received sub-streams is equal to the following value:

\[
\frac{u_{ji}}{(N-1)U(P_i)} \sum_{i=1}^{N} \sum_{j=1}^{N} \frac{u_{ji} U_{ji}}{(N-1)U(P_i)} U_{jk} + \frac{\xi u_{ji}}{U(P_i)} = \frac{u_{ji} U_{ji}}{(N-1)U(P_i)} \sum_{i=1}^{N} \frac{u_i}{(N-1)U(P_i)} U_{ji} + \frac{\xi u_{ji}}{U(P_i)} =
\]

\[
\frac{u_{ji} U_{ji}}{(N-1)U(P_i)} \left[ \frac{u_i}{(N-1)} - \frac{u_i}{(N-1)} \right] + \frac{\xi u_{ji}}{U(P_i)}
\]

\[
= \frac{u_{ji} U_{ji}}{(N-1)U(P_i)} \frac{u_i}{(N-1)} + \frac{\xi u_{ji}}{U(P_i)} = \frac{u_{ji} u_i}{(N-1)U(P_i)} + \frac{\xi u_{ji}}{U(P_i)}
\]

\[
r \frac{u_{ji}}{U(P_i)}
\]

QED.

From the fact stated in Observation 3 that each node has a unique part of a stream across a sub-network, each node \((j, i)\) uploads data to all its neighbors (located in the same sub-network) with streaming rate \( s_{ji}^{(4)} \),

\[
s_{j, i}^{(4)} = \begin{cases} \frac{r u_{ji}}{U(P)} & \text{if } i \text{ belongs to } H \\ \frac{r u_{ji}}{U(P)} \alpha_i & \text{if } i \text{ belongs to } \overline{H} \end{cases}
\]

where \( \alpha_i \) is a constant defined for each sub-network \( i \):

\[
\alpha_i = \frac{U(P_i) - u_i}{r(n-1)}.
\]
\[ H \]

If \( i \in H \) satisfies Eq. (16), we know that \( \frac{U(P) - u_i}{n_i - 1} < r \), what implies that \( \alpha_i < 1 \). Nodes in \( H \) can upload all streams \( s^{(2)}_{j; i, l, k} \) and \( s^{(4)}_{j; i, l, k} \) since their upload for the local sub-network (noted as a \textit{local upload}) and other sub-networks (noted as a \textit{global upload}) are bounded respectively:

- \textit{local upload} is equal to \( (n_i - 1)r \frac{u_{ji}}{U(P)} - \frac{u_i}{U(P)} \),

- \textit{global upload} is equal to \( (N - 1) \alpha_i = (N - 1) \frac{u_{ji}}{U(P)} \).

The sum of all \textit{global uploads} from nodes in \( i \) does not exceed \( u_i \),

\[
\sum_{j \neq i} u_{ij} \frac{U(P)}{U(P)} < u_i. \tag{59}
\]

On the other hand, a sum of a local and global upload of a single node \((j, i)\) is equal to:

\[
u_{ji} \frac{u_{ji}}{U(P)} + (n_i - 1)r \frac{u_{ji}}{U(P)} - \frac{u_i}{U(P)} \frac{U(P) - u}{U(P)} + \frac{u_{ji}}{U(P)} = u_{ji}. \tag{60}
\]

thus, it never exceeds a local limit \( u_{ji} \). On the other hand, nodes in \( H \) upload the corresponding amount of \textit{local} and \textit{global upload}:

- \textit{local upload} for nodes in \( H \) is equal to \( (n_i - 1) \frac{u_{ji}}{U(P)} \),

- \textit{global upload} for nodes in \( H \) is equal to \( (N - 1) \frac{u_{ji}}{U(P)} \).

The sum of all \textit{global uploads} is the same as in Eq. (59), therefore it never exceeds \( u_i \). The maximum amount of utilized upload on a \textit{local link} does not exceed limit \( u_{ji} \):

\[
(N - 1) \frac{u_{ji}}{U(P)} + (n_i - 1) r \frac{u_{ji}}{U(P)} - \frac{u_i}{U(P)} - u_{ji} [u_i + r(n_i - 1)] \tag{61}
\]

\( i \in H \), what implies that Eq. (17) is satisfied, and as \( u_i + r(n_i - 1) < U(P) \), the utilized upload on a \textit{local link} is smaller than \( u_{ji} \).

\textbf{Observation 4:} At this moment, each node in \( H \) obtained a full copy of stream \( r \), and nodes from \( H \) received sub-streams with rate \( r \left[ \frac{u_{ji}}{U(P)} (1 - \alpha_i) + \alpha_i \right] \).

\textbf{Proof:} From Observation 3 we know that each node in \( H \) received the sub-streams from other sub-networks with rate \( r \frac{u_{ji}}{U(P)} \). Furthermore, nodes in \( H \) received from its neighbors \( \sum_{i \neq j} r \frac{u_{ji}}{U(P)} \), so their total rate is equal to \( r \). On the other hand, nodes in \( H \) received from its neighbors the sub-streams with rate \( \sum_{i \neq j} \frac{r u_{ji}}{U(P)} \alpha_i \), so their aggregated content rate is equal to:

\[
r \frac{u_{ji}}{U(P)} + \sum_{i \neq j} r \frac{u_{ji}}{U(P)} \alpha_i = r \frac{u_{ji}}{U(P)} \left[ \sum_{i \neq j} r \frac{u_{ji}}{U(P)} \alpha_i - r \frac{u_{ji}}{U(P)} \alpha_i \right] =
\]

\[
r \frac{u_{ji}}{U(P)} (1 - \alpha_i) + r \alpha_i = r \left[ \frac{u_{ji}}{U(P)} (1 - \alpha_i) + \alpha_i \right]. \tag{62}
\]
QED.

According to Observation 4, the nodes that belong to \( H \) still wait for data to recover the entire content with a streaming rate. Moreover, we know that the server and the nodes in \( H \) have the entire stream, so they can provide help for other nodes suffering missing parts of the stream. We consider this help as an *offered upload*, where a destination node (being in need of more streaming data) requests a specific part of a missing sub-stream. Thus, we assume that the aggregated amount of the offered upload has to be equal to the aggregated amount of the missing data in \( H \). From Observation 4 we can derive a sum of the entire missing data in \( H \) as in:

\[
\sum_{i=1}^{H} \sum_{i' = 1}^{n}(r - rU(P)) \left( 1 - \alpha_{i} \right) = \sum_{i=1}^{H} (m_{i} - rU(P) - r\alpha_{i}) = \sum_{i=1}^{H} r(n_{i} - 1)(1 - \alpha_{i}) = \sum_{i=1}^{H} r(n_{i} - 1) \left( \frac{r(n_{i} - 1) - U(P) + u_{i}}{r(n_{i} - 1)} \right) = \sum_{i=1}^{H} (r(n_{i} - 1) - U(P) + u_{i}) = m(H) - U(H) + u(H) \tag{63}
\]

From (18) we have that \( m(H) \leq U(H) + u(H) + u_{r} - rN \), so we can assess upper bound of (63) as:

\[
U(H) + u(H) + u_{r} - rN - U(H) + u(H) = u(N) + u_{r} - rN \tag{64}
\]

The amount of the available *offered upload* from the streaming server is equal to:

\[
u_{r} - (r - \xi) - \xi \cdot N = u_{r} - r + \xi - \xi \cdot N = u_{r} - r - \xi(N - 1) = \]

\[
u_{r} - (r - \frac{u(N)}{N - 1})(N - 1) = u_{r} - r - r(N - 1) + u(N) = \]

\[
u_{r} - rN + u(N). \tag{65}
\]

It means that the server can upload the missing data of the stream to all the nodes that have not received the entire stream yet.

**Case 2:** \( r \geq \frac{u(H)}{N - 1} \) and \( r \leq \frac{u(N)}{N - 1} \).

**Definition 2:** Set \( W \) (with size \( W = |W| \)) is the set of sub-networks that fulfill the following conditions:

\[
\sum_{i=1}^{W} u_{i} \leq r \land \sum_{i=1}^{W+1} u_{i} > r. \tag{66}
\]

Interpretation of \( W \) can be seen as a maximum set of indexed sub-networks whose aggregated access *upload links* do not exceed \( r(N - 1) \). Note that Set \( W \) can be identical with \( N \) when sub-networks *upload links* will be relatively small with a given streaming rate \( r \). The second assumption from Eq. (66) can be rewritten as:

\[
\frac{u(W) + u_{r}}{N - 1} > r. \tag{67}
\]

Let us assume that \( W < N \), otherwise follow Case 1. The server divides the stream into \( n(W) + W + [n(W + 1) + 1] \) sub-streams with streaming rate \( s_{i,j}^{\text{th}} \) of corresponding node \((j, i)\), as:

\[
s_{i,j}^{\text{th}} = \begin{cases} 
\frac{u_{i,j}}{(N - 1)U(P)} & \text{if } i \text{ belongs to } W \\
\frac{r - u(W)}{N - 1}u_{i,j} \quad \frac{U(P)}{U(P)} & \text{if } i \text{ belongs to sub-network } W + 1
\end{cases} \tag{68}
\]

The server sends sub-streams \( s_{i,j}^{\text{th}} \) to the corresponding nodes. The sum of all sub-streams \( s_{i,j}^{\text{th}} \) is equal to the streaming rate \( r \):
Observation 5: At this moment, each node \((j,i)\) has a unique part of the stream and the rate of this sub-stream is equal to
\[
r = \frac{u_{ji}}{U(P_i)}.
\]

Proof: Similarly to Observation 3 in Case 2 all sub-streams \(s^{(1)}_{j,j \rightarrow k}, s^{(2)}_{j,k \rightarrow j,j},\) and \(s^{(3)}_{j,j} \) are disjoint across a given sub-network that node \((j,i)\) belongs to, so it has a unique part of a stream within sub-network \(i\). Let us count the aggregated rate of streams in nodes from \(W, i = W+1, \) and \(i > W+1\). The nodes in \(W\) receive sub-streams from the streaming server and the nodes from other sub-networks with an aggregated rate as in:

\[
\sum_{i=1}^{W+1} \sum_{j=1}^{n_i} s^{(1)}_{j,j} + \sum_{k=1}^{W+1} \sum_{j=1}^{n_k} s^{(2)}_{j,k \rightarrow j,j} =
\]

\[
n_{j,j} \sum_{i=1}^{W+1} u_{ji} - \sum_{k=1}^{W+1} \sum_{i=1}^{n_i} u_{ji} \frac{u_{ki}}{(N-1)U(P)} \frac{u_{ij}}{U(P)} + \sum_{i=1}^{W+1} \sum_{j=1}^{n_i} \frac{[r - \frac{u(W)}{N-1}] u_{ji}}{U(P)} \frac{u_{ki}}{U(P)}.
\]

\[
\frac{u_{ji}}{(N-1)U(P)} + \frac{u_{ji}}{U(P)} - \frac{u(W) - u_{ji}}{U(P)} + \frac{u(W)}{U(P)} = \frac{u_{ji}}{U(P)}
\]

The nodes in sub-network \(i = W+1\) receive sub-streams from the server and the nodes in \(W\) with a total rate given by:

\[
\sum_{i=1}^{W} \sum_{j=1}^{n_i} s^{(1)}_{j,j} + \sum_{k=1}^{W+1} \sum_{j=1}^{n_k} s^{(2)}_{j,k \rightarrow j,j} =
\]

\[
n_{j,j} \sum_{i=1}^{W} u_{ji} - \sum_{k=1}^{W+1} \sum_{i=1}^{n_i} u_{ji} \frac{u_{ki}}{(N-1)U(P)} \frac{u_{ij}}{U(P)} + \sum_{i=1}^{W} \sum_{j=1}^{n_i} \frac{[r - \frac{u(W)}{N-1}] u_{ji}}{U(P)} \frac{u_{ki}}{U(P)} + \sum_{i=1}^{W} \sum_{j=1}^{n_i} \frac{u_{ji}}{(N-1)U(P)} \frac{u_{ki}}{U(P)}.
\]

\[
\frac{r - \frac{u(W)}{N-1}}{U(P)} = \frac{u(W)}{U(P)} + \frac{u_{ji}}{U(P)} = \frac{u_{ji}}{U(P)}
\]

The nodes in sub-networks indexed by \(i > W+1\) receive the data from all nodes in \(W\) and \(W+1\) with the rate expressed in:
\(\sum_{j=1}^{n_{il}} \sum_{i=1}^{n_{ll}} s_{i,i,i,j}^{(4)} = \sum_{j=1}^{n_{il}} \sum_{i=1}^{n_{ll}} u_{i,j} u_{j,i} + \sum_{i=1}^{n_{il}} \left( \frac{r - u(W)}{N-1} \right) \frac{u_{i,i,i-1}}{U(P)} \)  

(74)

\[
\frac{u_{i,j}}{U(P)} \left( \frac{u(W)}{N-1} + \frac{\sum_{j=1}^{n_{il}} u_{i,j}^{(N-1)} U(P)}{U(P)} \right) = r \cdot \frac{u_{i,i,i-1}}{U(P)}
\]

QED.

In the next step, the nodes that belong to \(N\) send a sub-stream to all neighbors (the nodes that belong to the same sub-stream) with streaming rate \(s_{i,i,i,j}^{(4)}\) such that:

\[
s_{i,i,i,j}^{(4)} = \begin{cases} 
    r \cdot \frac{u_{i,j}}{U(P)} & \text{if } i \text{ belongs to } H \\
    r \cdot \frac{u_{i,j}}{U(P)} \alpha_i & \text{if } i \text{ belongs to } \overline{H} 
\end{cases}
\]

(75)

where \(\alpha_i\) is a constant defined for each sub-network \(i\) in Eq. (58).

Since \(s_{i,i,i,j}^{(4)}\) is expressed by a function identical like \(s_{i,i,i,j}^{(4)}\) in Case 1 and \(s_{i,i,i,j}^{(2)}\) is not greater than \(s_{i,i,i,j}^{(2)}\) in Case 1 (see Eqs. (53), (70), and (71)), we can assume that the all nodes in \(H\) and \(\overline{H}\) can handle such uploads.

An amount of the available offered upload is bounded by:

- \(u_{i,i} - r_{i,i}\) from the streaming server,
- \(u_{i,i} - r_{i,i} + u(W)\), from each sub-network \(i\) such that \(i \in \overline{H}\) and \(i = W + 1\).
- \(u_{i,i}\) from each sub-network \(i\) such \(i \in \overline{H}\) and \(i > W + 1\).

As this case assumes \(r \geq \frac{u(H)}{N-1}\) and from a definition of set \(W\) we know that \(\sum_{i=1}^{n_{il}} u_{i,i} > r\), we can conclude that:

\[
\sum_{i=1}^{n_{il}} u_{i,i} = u(W) + u_{i,i,i-1} > r \geq \frac{u(H)}{N-1} = |W + 1| > |H|,
\]

(76)

so set \(\overline{H}\) contains all sub-networks \(i \geq W + 1\).

The amount of available offered upload of sub-network \(i = W + 1\) comes from the fact that access links on those sub-networks have been unused yet and the local link uses only total stream \((n_i - 1)r)\ \frac{u_{i,i,i-1}}{U(P)}\). From a definition of Set \(H\), we know that \(r(n_i - 1) < U(P) - u_{i,i}\), therefore:

\[
(n_i - 1)r \frac{u_{i,i,i-1}}{U(P)} < (U(P) - u_{i,i}) \frac{u_{i,i,i-1}}{U(P)} = u_{i,i,i-1} - \frac{u_{i,i,i-1}}{U(P)}.
\]

(77)

It means that sum of all unused upload in a sub-network is at least equal to \(u_{i,i}\).

The amount of a free upload on all local upload links in \(i = W + 1\) can be bounded by (a maximum limit decreased by an amount of global and local upload):

\[
U(P) - r(N - 1) + u(W) - \sum_{j=1}^{n_{il}} \frac{r(n_i - 1)}{U(P)} \frac{u_{i,i,j}}{U(P)} =
\]

(78)

\[
U(P) - r(N - 1) + u(W) - r(n_i - 1) \geq u_{i,i} - r(N - 1) + u(W),
\]

since \(r(n_i - 1) \leq U(P) - u_{i,i}\), as given in Eq. (17). The amount of the available offered upload of sub-networks \(i > W + 1\) can be derived as the amount of a free global upload left after sending \(s_{i,i,i,j}^{(2)}\), and is expressed as:
Therefore, we can sum the lower bounds of offered uploads as:

\[ u_j - r + \sum_{i \in w_{j,l}} u_i + u_{w_{l,i}} - r(N-1) + u(W) = u_j - rN + u(N), \]

which is equal to the upper bound of the missing data in \( H \) as in Case 1, see Eq. (64).

**Case 3:** \( r < \frac{u(H)}{N-1} \) and \( r < u_j \).

When \( \hat{r} < u_j \) then \( r = \frac{U(H) + u(H)}{n(H) + N} \). We define \( F > 0 \) such that:

\[ F = \frac{u(H)}{N-1} - r, \]

At the first step, the server splits the stream into \( n(H) + H \) disjoint streams and sends them into corresponding nodes in the sub-networks from \( H \) with the following rate:

\[ s_{j,l}^{(1)} = \frac{ru_i u_j}{u(H) U(P)}, \]

The sum of all sub-streams \( s_{j,l}^{(1)} \) is equal to \( r \), because:

\[ \sum_{i \in n_{j,l}} \sum_{j \in j_{l}} s_{j,l}^{(1)} = \frac{r}{u(H)} \sum_{i \in n_{j,l}} U(P_j) \sum_{j \in j_{l}} u_j = \frac{r}{u(H)} \sum_{i \in n_{j,l}} u_i = r. \]

Then, each node \((j,i), i \in H\), sends a fraction of received \( s_{j,l}^{(1)} \) to all nodes \((l,k), k \in N, k \neq i\) (all the nodes except for its neighbors) with the following stream rate:

\[ s_{j,l}^{(1)} = \frac{u_{i,k}}{u(H) U(P_j)}, \]

in such a way, that stream \( s_{j,l}^{(1)} \) is split into disjoint \( n_k \) sub-streams that are sent to nodes in sub-network \( k \). The total upload used on a local link of node \((j,i), i \in H\), is given as:

\[ \sum_{k \in n} \sum_{i \in n_k} s_{j,l}^{(1)} U(P_j) = \sum_{k \in n} s_{j,l}^{(1)} = (N-1)s_{j,l}^{(1)} = (N-1) \frac{ru_i u_j}{u(H) U(P_j)} = \]

\[ \frac{(N-1)}{u(H) - N-1} u_{i,j} = \frac{u_i u_j}{u(H) U(P_j)} + F \frac{u_i u_j}{u(H) U(P_j) u(H)} < \frac{u_i u_j}{u(H) U(P_j)} < u_j. \]

At this moment each node in \( H \) received sub-streams with a total rate defined as \( s_{j,l}^{(2)} \) and derived as:

\[ s_{j,l}^{(2)} = s_{j,l}^{(1)} + \sum_{k \in n} \sum_{i \in n_k} \frac{ru_i u_j}{u(H) U(P_j)} = s_{j,l}^{(1)} + \frac{u_i u_j}{U(P_j) u(H)} \sum_{k \in n} \sum_{i \in n_k} \frac{ru_i u_j}{u(H) U(P_j)} = \]

\[ s_{j,l}^{(2)} + \frac{ru_i u_j}{U(P_j) u(H)} \sum_{k \in n} u_k = \frac{ru_i u_j}{U(P_j) u(H)} (u(H) - u_i) + \frac{ru_i u_j}{U(P_j) u(H)} (u(H) - u_i) = \frac{ru_i u_j}{U(P_j)} \]

Note that the sub-streams received by the nodes in \((l,k), k \in H\), have a total rate equal to \( s_{j,l}^{(2)} \) as well. Moreover, those sub-
streams are unique within a single sub-network \( k \). In the next step, all nodes \((l,k), k \in \overline{H}\), send its \( s_{l,j}^{(2)} \) to all neighbors in sub-network \( k \). They can do that, because the total upload on a local link is smaller than \( u_{l,j} \):

\[
\frac{r n_{l,k} - (n_l - 1) - (U(P_l) - u_{l,k})}{U(P_l)} < u_{l,j}.
\]

(86a)

Since all sub-streams \( s_{l,k}^{(2)} \) in a single sub-network \( k \) are unique, after this procedure all nodes in \( \overline{H} \) received a full copy of a stream. Parameter \( y_{j,i} \) is defined for each node \((j,i), i \in H\), in the following way:

\[
y_{j,i} = \frac{(U(P_l) - u_{l,k})\{r(n_l - 1)a(H) - (U(P_l) - u(H) + u L r(N - 1)]}{(n_l - 1)U(P_l)a(H)\{r(n(H) + N - 1) - U(H)\].
\]

(86b)

Now, a sum of parameters \( y_{j,i} \) of nodes in a single sub-network \( i \) is given by:

\[
y_i = \sum_{j=1}^{n_i} y_{j,i} = \frac{r(n_l - 1)a(H) - (U(P_l) - u(H) + u L r(N - 1)]}{(n_l - 1)U(P_l)a(H)\{r(n(H) + N - 1) - U(H)\]\n
(87)

Sum of all \( y_i \) is equal to the following:

\[
\sum_{i=1}^{n} y_i = \frac{r n_l \{r(n(H) + N - 1) - U(H)\]}{r(n_l - 1)a(H) - (U(P_l) - u(H) + u L r(N - 1)]} = 1
\]

(88)

From Eq. (87) we know that each node \((l,k), k \in \overline{H}\), has at least \( \frac{u_{l,k}}{U(P_l)} \) free upload. It divides it between all nodes in \( H \) according to parameter \( y_{j,i} \). Since \( l \) has a full copy of the stream, it can send those parts of the stream that are unique for a destination node. Using Eq. (88) we know that the total upload on a local link from node \((l,k)\) equals:

\[
u_{l,j} = \sum_{j=1}^{n_j} y_{j,i} = \frac{u_{l,k} U(P_l)}{U(P_l)}.
\]

(89)

The aggregated upload on local links in a single sub-network \( k \in \overline{H} \) is equal to:

\[
\sum_{j=1}^{n_j} u_{l,j} = u_{l,k}.
\]

(90)

Therefore, those sub-streams are not limited on an access link of sub-network \( k \). Similarly, the server sends its free upload \( (u_s - r) \) to nodes \((j,i), i \in H\), according to their parameter \( y_{j,i} \).

From Eq. (85) we can see that each \((j,i), i \in H\) has a free upload equal to:

\[
u_{j,i} = \frac{u_{j,i} U(P_l)}{U(P_l)} = \frac{F(N - 1)a(H)}{U(P_l)u(H)}.
\]

(91)

It sends data to its \( n_i - 1 \) neighbors sub-streams with the same rate. Due to the fact that all \( s_{l,j}^{(1)} \) are disjoint within one sub-network, the data sent to its neighbors must be new for them.

The accumulated rate of all sub-streams received by node \((j,i), i \in H\), is given by:
Therefore, we know that all nodes in a network receive a full copy of an original stream with rate 

\[ r = \frac{U(H) + u(H) + u_s}{n(H) + N}. \]

**Case 4: \( r = u_s. \)**

By considering a case when \( r = u_s \) we know that \( r \leq \frac{U(H) + u(H) + u_s}{n(H) + N} \). For the universal streaming we perform the same steps like in a previous case with the following minor changes:

- the server does not send the extra data to nodes in \( H \),
- instead of \( y_{ij} \) defined in Eq. (86), we introduce \( w_{ij} \) as:

\[ w_{ij} = \frac{(U(P) - u_s)(n - 1)u(H) + u(H) - F(N - 1)u_s}{(n - 1)u(H)u(H)U(P)} \]  

(93)

and use it instead of \( y_{ij} \).

From Eq. (87) we know that each node \((l, k) \in H\) has at least \( \frac{u_{H,l,k}}{U(P)} \) free upload. It divides it between all nodes in \( H \) according to Parameter \( w_{ij} \). Since \((l, k)\) has a full copy of the stream, it can send those parts of a stream that are unique for a destination node. Now, we have to prove that \( \sum_{j=1}^{n} \sum_{j=1}^{n} w_{ij} \leq 1. \) \( w_i \) is a sum of Parameters \( w_{ij} \), in a single sub-network \( i \) as given by:

\[ w_i = \sum_{j=1}^{n} w_{ij} = \sum_{j=1}^{n} \frac{(U(P) - u_s)(n - 1)u(H) - U(P)u(H) + u(H) - F(N - 1)u_s}{(n - 1)u(H)u(H)U(P)} \]  

(94)

The sum of all parameters \( w_{ij} \) in \( H \) is expressed as:

\[ \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} = \sum_{i=1}^{n} w_i = \frac{u(H)u(H) - U(H)u(H) + u(H) - F(N - 1)u_s}{U(H)u(H)} \]  

(95)

By replacing \( F(N - 1) \) with \( u(H) - (N - 1)u_s \), we have from Eq. (81) that:
\[
\sum_{i=1}^{n} \sum_{j=1}^{N} w_{ij} = \frac{u_{i} n(H) - U(H) + u(H) - u(H) + (N-1)u_{i}}{u(H)} = \frac{u_{i} [n(H) + N-1] - U(H)}{u(H)}.
\]  

(96)

From the fact that \(u_{i} \leq r_{i}(h)\) we have:

\[
u_{i} \leq \frac{U(H) + u(H)}{n(H) - N} \Rightarrow u_{i} \leq \frac{U(H) + u(H)}{n(H) - N - 1} \Rightarrow u_{i} [n(H) + N-1] \leq U(H) + u(H).
\]

(97)

Using this fact with Eq. (96), we can bound the sum of all Parameters \(w_{ij}\) as:

\[
\sum_{i=1}^{n} \sum_{j=1}^{N} w_{ij} \leq \frac{U(H) + u(H)}{u(H)} - 1.
\]

(98)

The sum of all sub-stream rates received by Node \((j,i), i \in H\) is equal to the data given in

\[
\frac{u_{i} u_{ij}}{U(P)} + \frac{1}{n_i-1} \sum_{j=1}^{N} u_{ij} \frac{u_{i} u_{ij}}{U(P)} - \frac{u_{i} u_{ij}}{U(P)} u(H) - \frac{U(P) - u(H) + u(H) + u(H) - u(H)}{U(P)} u_{i} = \frac{u_{i} u_{ij}}{U(P)} - \frac{u_{i} u_{ij}}{U(P)} u(H) - \frac{U(P) - u(H) + u(H) + u(H) - u(H)}{U(P)} u_{i} = \frac{u_{i} u_{ij}}{U(P)} - \frac{u_{i} u_{ij}}{U(P)} u(H) - \frac{U(P) - u(H) + u(H) + u(H) - u(H)}{U(P)} u_{i}.
\]

(99)

That means, that all nodes in the network receive a full copy of the original stream with rate \(r = u_{i}\). That ends the proof of Theorem 1. QED.

IV. CONCLUSIONS

The presented theorem gives a simple, yet powerful tool for modeling a streaming P2P network. Firstly, it can be applied to design of streaming server capabilities. It reveals a weak link of an overlay network, which introduces service quality degradation. It also shows that placing additional servers inside resources lacking networks can easily increase the achievable streaming rate. We can observe that the sub-networks forming set \(H\) need additional care and consideration. To address low-resources problem of customers, a content provider can increase the server upload limit or provide additional servers inside sub-networks lacking transferring resources. Furthermore, this model can be used to estimate a maximum gain of a P2P streaming structure, comparing to a standard client-server topology.

As a future work, the churn phenomena can be considered according to the idea presented by Kumar et al. [1]. Results of a downloading rate in a file-sharing overlay may converge with those presented by this fluid streaming-based model. Such investigations could be also a vital research.

Unfortunately, this model cannot be applied in a tree-based topology. It has been proven by Wu et al. [19] that finding the maximum streaming rate with a limited neighborhood is an NP-hard problem even in a non-hierarchical fluid model. Therefore, the tree topology which limits number of neighbors to \(1 + i\), where \(i\) is a number of children cannot be considered by this model. On the other hand, tree-based streaming P2P networks are not used in practice broadly, thus this fact is not necessarily harmful from our theorem application viewpoint.

ACKNOWLEDGMENT

This work was supported by the Polish National Science Centre under Grant N N517 555239.
REFERENCES


[18] R. Cholda obtained a doctorate in telecommunication in 2006 from AGH University of Science and Technology (Krakow, Poland). Then, he joined the Department of Telecommunications there, and now he works as an Assistant Professor. He specializes in design of computer and communications networks. Although his background comes from technology, as he started his research in the reliability modelling and optimization of resilient multi-layer long-haul networks, later he moved towards softer problems, like the effective provisioning of services to the network clients. Thus, he extended his interests to problems embracing the so-called Quality of Recovery (QoR), resilience-based differentiation in communications/computer networks and QoS/QoR for peer-to-peer networking. Recently, he has focused on risk-based communications networking, as he perceives this area as a very promising research field, where the interface between business needs and threats from one side and the resilience technology from the other can be used to obtain a practical equilibrium where the balance between client needs and technological possibilities can be found. Dr. Cholda is the co-author of eleven refereed journal papers and two conference tutorials. He is the recipient of the Communications QoS, Reliability and Performance Modeling Symposium Best Paper Award from ICC ’06. He worked on network recovery problems in some of the 6. and 7. FP European Projects (IP Nobel/Nobel II, NoE EuroNGI/EuroFGI/EuroNF, SmoothIT). He was a Technical Program Committee (TPC) Co-Chair of NGI 2008 and NGI 2012, a TPC Co-Chair of Communications QoS, Reliability and Modeling Symposium at ICC 2011, and a TPC Co-Chair of DRCN 2011. Now, he serves as a TPC Co-Chair of NOMS 2014. He is a member of the editorial board of IEEE Communications Surveys and Tutorials. He also leads the Book Review Column in IEEE Communications Magazine.

Bartosz Polaczyk was a Ph.D. student at AGH University of Science and Technology during working on this paper.

Piotr Cholda obtained a doctorate in telecommunications in 2006 from AGH University of Science and Technology (Krakow, Poland). Then, he joined the Department of Telecommunications there, and now he works as an Assistant Professor. He specializes in design of computer and communications networks. Although his background comes from technology, as he started his research in the reliability modelling and optimization of resilient multi-layer long-haul networks, later he moved towards softer problems, like the effective provisioning of services to the network clients. Thus, he extended his interests to problems embracing the so-called Quality of Recovery (QoR), resilience-based differentiation in communications/computer networks and QoS/QoR for peer-to-peer networking. Recently, he has focused on risk-based communications networking, as he perceives this area as a very promising research field, where the interface between business needs and threats from one side and the resilience technology from the other can be used to obtain a practical equilibrium where the balance between client needs and technological possibilities can be found. Dr. Cholda is the co-author of eleven refereed journal papers and two conference tutorials. He is the recipient of the Communications QoS, Reliability and Performance Modeling Symposium Best Paper Award from ICC ’06. He worked on network recovery problems in some of the 6. and 7. FP European Projects (IP Nobel/Nobel II, NoE EuroNGI/EuroFGI/EuroNF, SmoothIT). He was a Technical Program Committee (TPC) Co-Chair of NGI 2008 and NGI 2012, a TPC Co-Chair of Communications QoS, Reliability and Modeling Symposium at ICC 2011, and a TPC Co-Chair of DRCN 2011. Now, he serves as a TPC Co-Chair of NOMS 2014. He is a member of the editorial board of IEEE Communications Surveys and Tutorials. He also leads the Book Review Column in IEEE Communications Magazine.

Communications in Information Science and Management Engineering May, 2013, Vol. 3 Iss. 5, PP. 202-221