PROCEEDINGS

55. IWK
Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

13 - 17 September 2010

Crossing Borders within the ABC

Automation,

Biomedical Engineering and

Computer Science

Faculty of
Computer Science and Automation

www.tu-ilmenau.de

TECHNISCHE UNIVERSITÄT
ILMENAU

Impressum
Published by

Publisher: Rector of the Ilmenau University of Technology
Univ.-Prof. Dr. rer. nat. habil. Dr. h. c. Prof. h. c. Peter Scharff

Editor: Marketing Department (Phone: +49 3677 69-2520)
Andrea Schneider (conferences@tu-ilmenau.de)

Faculty of Computer Science and Automation
(Phone: +49 3677 69-2860)
Univ.-Prof. Dr.-Ing. habil. Jens Haueisen

Editorial Deadline: 20. August 2010

Implementation: Ilmenau University of Technology
Felix Böckelmann
Philipp Schmidt

USB-Flash-Version.

Publishing House: Verlag ISLE, Betriebsstätte des ISLE e.V.
Werner-von-Siemens-Str. 16
98693 Ilmenau

Production: CDA Datenträger Albrechts GmbH, 98529 Suhl/Albrechts

Order trough: Marketing Department (+49 3677 69-2520)
Andrea Schneider (conferences@tu-ilmenau.de)

ISBN: 978-3-938843-53-6 (USB-Flash Version)

Online-Version:

Publisher: Universitätsbibliothek Ilmenau
ilmedia
Postfach 10 05 65
98684 Ilmenau

Home / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=16739

IPTV - THE CASE FOR PEER-TO-PEER LIVE-STREAMING

Mathias Fischer

Department of Telematics and Computer Networks, TU Ilmenau

ABSTRACT

Internet Protocol Television (IPTV) is becoming more and more popular and will burden future networks heavily with traffic, especially if deployed with a classic client-server-like approach. Besides, it does not scale. Application Layer Multicast (ALM) and its Peer-to-Peer (P2P)-like content distribution is a promising technology to overcome the bottleneck of client-sever and to relieve core networks from redundant transmissions.

Nevertheless, since relying on unreliable end-users, ALM systems are more susceptible to node churn and DoS attacks, so that appropriate countermeasures and a careful system design are required. We describe the architecture of an IPTV system based on ALM, present a formal IPTV model and summarize several research challenges in such a scenario. Our system is intended to optimally adapt its resources to a changing user behavior and to build topologies that are efficient and robust at the same time by minimizing reconstruction costs caused by changes in the user compound.

Index Terms— Application Layer Multicast, IPTV, Peer-to-Peer, Live-Streaming, Resilience, DoS

1. INTRODUCTION

In current days, we observe a convergence of classical media distribution with the Internet. People more and more shift their media consumption from conventional broadcast media like TV or radio to the Internet. They listen to radio streams, watch videos on YouTube and enjoy sport live-streams. Hence, for a broad audience IPTV is only a small step ahead. This challenges network operators as well as service providers. Network operators have to cope with steadily increasing data traffic in their networks and service providers have to deploy more and more servers to meet the increasing demand.

ALM can be a solution to both problems at once, since it can shift traffic from core networks and avoid the classical client-server bottleneck by providing an inherent scalable content distribution. The stream is distributed from a source to a large number of peers by again utilizing their resources for re-distributing the stream to other peers. So, the number of participants can grow independently of the sources' upload bandwidth. Moreover, a careful ALM topology design can shift traffic closer to the edges of the network and relieves network operators from traffic.

Problems arising from designing an IPTV system on the basis of ALM, concern especially robustness and Quality of Service (QoS). In order to become accepted by end-users, the QoS should not be significantly worse than with current client-server solutions. Furthermore, since relying on unreliable end-hosts, such systems are more susceptible to node churn and intended DoS attacks. Thus, the failure of a single node affects all of its successors and packet loss propagates in the distribution trees. This problem will even deteriorate due to an increasing mobile use of such systems in the future that induces much higher node churn.

In this article, we describe an IPTV system based on ALM that is intended for Live-Streaming and Near-Video-On-Demand-Streaming (a live-stream that is repeated in specific time intervals). The system distributes a multitude of streams to large and highly heterogeneous (e.g. regarding bandwidth) user groups and is intended to optimally adapt its resources to a permanently changing user behavior. Furthermore, the system maintains service (at least partially) during overlay and underlay failures, e.g. due to massive DDoS attacks. In the process, we present a formal model describing our IPTV system and summarize arising research challenges.

Sec. 2 specifies the requirements to an IPTV system and Sec. 3 summarizes the related work in combining IPTV with P2P. Sec. 4 gives an overview of our system architecture and introduces a formal model of our system. Sec. 5 outlines research questions and Sec. 6 concludes the article.

2. REQUIREMENTS TO AN IPTV SYSTEM

The following requirements to IPTV can be distinguished:

- **Scalability** An IPTV system remains scalable at any number of participants. Besides, by avoiding central components as potential Single-Point-of-Failure (SPoF)s, it should be possible to extend the system by any number of additional infrastructure components (e.g. sources).
- **Robustness/Stability** The system has to remain

This work was supported by the Deutsche Forschungsgemeinschaft under grant number SCHA 309/1-2.

functional during node churn and Denial of Service (DoS) attacks on participants, sources or other system components. In worst case scenarios, graceful degradation functionality has to be ensured. Besides pure overlay stability, an IPTV system has to be robust in the presence of single underlay link and node failures as well.

- **Transmission Efficiency** has to be achieved on user- as well as on network-side. On user side QoS requirements have to be kept, like minimizing packet loss, delay and jitter. On network side an IPTV system has to minimize redundant transmissions across core networks to relieve network operators from traffic.

- **Efficient Resource Usage** The IPTV system has to be economical regarding the resources on servers. So, at one hand the required server bandwidth has to be minimized, but on the other hand still strict QoS requirements have to be met.

- **Minimal Signaling Overhead** Arising overhead in terms of signaling has to be kept as minimal as possible.

- **Self-Configuration** An IPTV system should be functional without manual interaction. It adapts to changing conditions in the source or participant compound and adjusts to varying network conditions. Furthermore, the system is easily extensible by additional components (e.g. sources).

- **Agility** User agility/mobility imposes a challenge to IPTV systems and has to addressed, since probably a large fraction of participants in such systems will be mobile in the future. Hence, no mobility handling will further worsen the problem of node churn leading to service degradation until disruptions of the stream.

- **User Heterogeneity** Participants in a IPTV system are highly heterogeneous regarding their user behavior, memory, CPU, bandwidth or even their energy supply. This has to be considered at building an IPTV system.

3. RELATED WORK ON IPTV

Related work regarding P2P-IPTV covers a broad range of topics, like P2P streaming approaches or server load-balancing. Currently there is also standardization work of European Telecommunications Standards Institute (ETSI) within the Telecoms & Internet converged Services & Protocols for Advanced Network (TISPAN) project that attempts to define a new standard for IPTV. TISPAN proposes two IPTV solutions, an IPTV subsystem that focuses to integrate existing solutions into an Next Generation Network (NGN) and a solution based on 3G's IP Multimedia Subsystem (IMS) that allows to use IPTV together with other telecommunication services like different data services for instance. A P2P solution is discussed as well, but without any specific agreement so far.

ALM systems are usually distinguished in push-based systems that explicitly construct topologies and pull-based systems, in which topologies are constructed implicitly on the basis of loosely coupled nodes. Moreover, hybrid systems that combine fast push-based delivery with the improved resilience of a pull-based approach exist. Most ALM systems real deployment and with broader audiences are pull-based systems like CoolStreaming [1] or PPLive [2]. However, their startup-delay (in PPLive up to two minutes) renders them useless for live-streaming purposes, since the efficiency requirement is violated. For this reason, only push-based ALM like the own system proposed in [3, 4] and hybrid systems like mTreebone [5] are expected to be suitable for an efficient large-scale distribution of live-content.

Peer-assisted streaming summarizes a subgroup of ALM systems, in which the P2P content distribution is massively supported by the system itself, so that end-users are partly relieved from re-distribution compared to a pure P2P approach. For example peers are delivered with some stripes by sources directly and obtain the rest of them via ALM. In MediaGrid [6] content is disseminated from a source to distribution hubs close to the edges of a Fiber To The Node (FTTN) network. Distribution hubs disseminate the content via ALM in their local area of responsibility. Data exchange between peers across different areas is prohibited, since the authors assume that the traffic bottleneck appears most likely in local distribution hubs.

Furthermore, the allocation of resources to streams has to be considered, to minimize the bandwidth requirements on side of the sources by still providing reasonable stream qualities. The authors of [7] establish performance bounds for minimum server load, maximum supported stream rate and minimum tree depth in peer-assisted streaming approaches. These bounds were established for three different cases: First, peers have unlimited bandwidth, second peers have at most only one successor per stripe and third, peers can have several successors per stripe. The authors of [8] employ a proactive approach to assign resources to streams. Server bandwidth is allocated to a stream in dependence on its expected future popularity to guarantee a desired level of streaming quality. The prospective popularity of a stream is predicted based on the past evolution of its member compound.

4. P2P - IPTV SYSTEM

In the following, we give an architectural overview of an own IPTV system that utilizes the resources of its participants for distributing a multitude of streams. Furthermore, we derive a formal model describing this system and outline major research directions in P2P-IPTV.

4.1. Architecture Overview

In our P2P-IPTV system, a multitude of streams is distributed outgoing from a set of sources onto a large participant set, which re-distribute the stream on their behalf as well. This way, the inherent bottleneck of a classical client-server solution is avoided by incorporating the participants' resources to the content distribution.

The system attempts to construct several ALM overlays at once that reflect the current user behavior and supplies users with requested content under strict QoS-constraints. Our IPTV system constructs a push-based ALM streaming overlay that supplies users with requested content under strict QoS constraints by adapting to their behavior. Users choose freely from the full range of streams and their corresponding stripes. Therewith, on the basis of Multiple Description Coding (MDC) and scalable video coding [9] it is possible to obtain extra content in addition to a basic stream, e.g. an additional stripe containing the sign language interpreter or supplementary stripes to obtain a video stream in higher resolution.

The system comprises a multitude of sources for providing a basic distribution service. Participants receiving a stream, re-distribute it on their behalf as well. Furthermore, participants with spare resources that are not utilized in their respective stream are forced or encouraged to distribute other streams they are not actually interested in as well. In addition, we introduce the novel concept of ALM-Switches that are placed within networks and can relocate their positions within fixed sites in the network.

Sources cooperate to distribute the system's streams and are arranged in a flat hierarchy in a P2P network, so that there is no need for central coordination. Every source preferably delivers one or several streams to clients in its local network neighborhood, whereby the first source distributing a distinct stream takes over responsibility for its coordination and the basic distribution service. The coordinating source spends as many of its bandwidth as necessary in order to limit the height of distribution trees and to keep QoS parameters in tolerable range. If the coordinating source's resources are finally depleted, it requests additional supporting sources with spare bandwidth for stream re-distribution. If the number of users and therewith the height of distribution trees decreases again, coordinating sources take over remaining members served by supporting sources again, depending on the QoS tolerance range.

Besides the large fraction of participants that obtain a stream and spend most of their resources for its re-distribution, some of the participants may still have huge *spare bandwidth* that is currently not required in their respective stream. Alternatively, if the system is deployed on set-top boxes their spare bandwidth, in times the user is not interested in any stream, can be used as well. Therewith, by including these otherwise wasted resources to the distribution process, the performance of the system can be improved. In the process, a *careful supporter selection* for distinct streams is important in order to maintain the locality and robustness of stream distribution at the same time.

ALM-Switches can either spend spend their bandwidth completely for a whole stream, for a single stripe, or alternatively their resources are divided more fine-grained onto the set of streams/stripes (e.g.to distribute the most popular mix of stripes in their local network neighborhood). Moreover, their positioning has huge influence on the system as well. Positioning them in the middle of the network will decrease delays, whereas a placement close to the edges of the network will decrease redundant transmissions in core networks and thus dis-burden network operators from traffic.

Therewith, ALM-Switches are a first measure to adapt IPTV systems to the underlay. Nevertheless, the problem of several overlay links spanning the same underlay link remains. Hence, an underlay failure could affect several overlay links at once, inducing huge damage in IPTV overlays. For this reason, we attempt to construct *underlay-aware overlay connections* by trying to establish preferably node- and link-disjoint underlay paths. Therewith, it is may possible to keep the system (at least partially) functional during huge DDoS or BGP attacks, for example.

4.2. Formal IPTV Model

The communication network as basis of the ALM overlay can be described by a connected underlay-graph $\mathcal{G}_U = (V_U \cup H, E_U)$, with $V \geq 1$ and $V \cap H = \emptyset$. H constitutes a set of hosts, that support virtualization and are able to host virtual ALM-Switches. V comprises all ordinary routers and all possible positions of participants. Every underlay edge $e_U \in E_U$ is characterized by an edge capacity $c(e_U) \geq 0$ ($c : E_U \rightarrow \mathbb{N}$) and a delay $d(e_U) \geq 0$ ($d : E_U \rightarrow \mathbb{R}^+$). For communication between underlay nodes, a global static single-path routing is assumed

$$route : (V_U \cup H) \times (V_U \cup H) \rightarrow Paths(\mathcal{G}_U).$$

and assigns every pair of nodes (u, v) in $\mathcal{G}_U$ a path beginning in u and ending in v. The total delay of a path is given by $d(u, v) = \sum_{e \in route(u,v)} d(e)$. The set $\mathcal{M}$ comprises all overlay members, all sources and all ALM-Switches in $\mathcal{R}$. On top of $\mathcal{G}_U$ a streaming overlay $G_O = (V_O, E_O)$ is located, whereby $V_O = \mathcal{M}$ and $\forall e_O = (u, v) \in E_O : u, v \in \mathcal{M}$. The streaming system distributes r different streams $\mathcal{S}^0, ..., \mathcal{S}^{r-1}$. Furthermore, every stream $\mathcal{S}^i$ is divided into k^i distinct stripes $\mathcal{S}^i = \{\mathcal{S}^i_0, ..., \mathcal{S}^i_{k-1}\}$. All streams and their corresponding stripes in $\mathcal{S}^i$ form the overall stripe set $\mathbb{S} = \bigcup_{j=0}^{r-1} \mathcal{S}^j$. Stream sources announces the set of stripes they offer via a publish-function $pub : M \to 2^{\mathbb{S}}$, with the additional constraint:

$$\forall s \in \mathcal{S}^i : \exists m \in \mathcal{M} : s \in pub(m).$$

Accordingly, a subscribe function $sub : \mathcal{M} \to 2^{\mathcal{S}}$ assigns a subset of the overall stripe set $\mathcal{S}$ to every participant. So, for a member $m \in \mathcal{M}$ that receives the whole stream i, $sub(m) = \mathcal{S}^i$ applies. Beyond, a forwarding function $forw : M \to 2^{\mathcal{S}}$ indicates the willingness of a member m to contribute to a stream. It is assumed, that at least every member that publishes a stream and all members that subscribe to a stream provide resources for its distribution. In addition, members with spare resources are forced to assist in the distribution of additional stripes, so that

$$\forall m \in \mathcal{M} : sub(m) \cup pub(m) \subseteq forw(m).$$

The set of ALM-Switches $\mathcal{R} \subset M$ is assigned to underlay positions via a positioning function $pos : \mathcal{R} \to H$. ALM-Switches do not serve as source nor consume a stream. Instead their task lies in supporting the distribution of arbitrary stripes, so that the following holds:

$$\forall r \in \mathcal{R} : forw(r) = \mathbb{S}, pub(r) = sub(r) = \emptyset.$$

A member $m \in M$ can take over one or several of the following types:

- *Sources* $Src = \{m \in M | pub(m) \neq \emptyset\}$
- *Clients* $C = \{m \in M | sub(m) \neq \emptyset\}$
- *Forwarders* $F = \{m \in M | forw(m) \neq \emptyset\}$

All participants $m \in \mathcal{M}$ establish a streaming topology $\mathcal{T}$, which consists of a set of $\sum_{s \in Src} |pub(s)|$ trees. Every tree $T^i_{\mathbb{S}} \in \mathcal{T}$ constitutes a distribution topology for stripe i, starting by source node s. All trees that distribute stripe i are concentrated in $\mathcal{T}^i = \{\mathcal{T}^i_s \in T | s \in \mathcal{S}^i\}$. Trees of stream $\mathcal{S}^j$ form the set of topologies characterized by

$$\mathcal{T}^{\mathcal{S}^j} = \bigcup_{i \in \mathcal{S}^j} \mathcal{T}^i.$$

In order to model successor relationships, the function $succ_i(u)$ returns the set of nodes that are direct successors of node u in stripe i and $succ(u)$ returns the set of nodes that are direct successors across all stripes. In addition, $succ\_all_i(u)$ returns the overall set of successors of u in stripe i and $succ\_all(u)$ returns the corresponding set across all stripes.

Accordingly, function $pred_i(u)$ returns the predecessors of u in stripe i, $pred(u)$ returns the set of all predecessors across all stripes, $pred\_all_i(u)$ returns the set of all nodes dependent on node u in stripe i and function $pred\_all(u)$ returns the associated set across all stripes.

Members m in $\mathcal{M}$ have an overall capacity of $c(m)$. Under the assumption of equally sized stripes, the capacity determines how often they can forward a stripe and thus their maximum possible number of direct successors $\forall m \in M : c(m) \geq succ(m)$.

For every member $m \in M$ an embedding exists $\mathcal{E} : M \to V \cup H$, that determines m a distinct underlay position in $\mathcal{G}_U$. Since ALM-Switches can only be hosted at special places (at certain routers) in the network, the following constraint applies to them:

$$\forall r \in \mathcal{R} : \mathcal{E}(r) \in H, \forall p \in (\mathcal{M} \setminus \mathcal{R}) : \mathcal{E}(p) \in V.$$

Function $assign(\mathbb{S}, Src, \mathcal{R}, F)$ assigns subsets of $\mathbb{S}$ to sources, ALM-Switches and forwarders for further distribution.

In order to model a timely distinct system, the system has to be regarded in a sequence of timely successive, static states. Therefore, $\mathcal{M}$, $\mathcal{E}$, pub, sub, $forw$, c and d are additionally parameterized in time $t \in \mathbb{N}$. So, based on a sequence of discrete points in time $t = (t_0, ...t_z)$ we can construct a set of timely ordered, static topology snapshots $\mathcal{T} = (\mathcal{T}_0, ..., \mathcal{T}_z)$.

Function $cost(T_u, T_v)$ quantifies the minimum necessary adaption costs for the transition of one optimal resource allocation for topology T_u to another preferably optimal topology T_v. The costs involve the adaption of stripe-sets at sources and ALM-Switches, the re-positioning of the latter and optimization operations in stripe trees.

5. RESEARCH FOCUS

In a P2P-IPTV system like that described in Sec. 4 several research challenges regarding the requirements to an IPTV system arise. Especially the efficiency and robustness requirements are hard to achieve, since both are even in partially contraposition to each other. Embedding the overlay into the underlay results in higher efficiency (e.g. shorter delays) but usually leads to less robust overlay topologies (e.g. unbalanced trees) and the other way around.

Furthermore, the division of the set of stripes to the resources of the system (sources, ALM-Switches, forwarders) has large influence on both requirements as well. To make it worse the need for adapting the system to the behavior of its users additionally hardens the problem.

5.1. Degree of Overlay-Embedding

Mapping an overlay onto the underlay is a severe design decision and depending on the applied metric robustness and/or efficiency of the system are influenced. From the overlay perspective, the only possibility to adapt to an underlay lies in controlling the establishment of overlay links. Basically, three different approaches are thinkable:

First, overlay links are established without considering the underlay, which can lead to inefficient routing and decreased underlay robustness due to underlay links that are traversed by several overlay links at once. Nevertheless, as shown in [3] it is possible to build optimally stable overlay topologies on that basis.

In the second method, overlay links are established underlay-aware, so that links minimize delays between overlay nodes as given in Equation 1. Nevertheless, the resulting topologies are may lack overlay stability and remain fragile regarding underlay attacks as well.

$$\begin{aligned} d(\mathcal{T}) &= \sum_{u \in Src} \sum_{v \in succ(u)} d(route(u,v)) \\ \mathcal{T}_{efficient} &= min\{\mathcal{T}_{sol} \in 2^M |\, d(\mathcal{T}_{sol})\} \end{aligned} \tag{1}$$

The third possibility comprises the establishment of preferably link- and node-disjoint overlay links (see Equation 3), which increase the underlay robustness, but may cause a degradation in efficiency again. An overview of methods for making overlays more robust and for establishing preferably node- and link-disjoint underlay links was summarized by us in former work [10]. In the process, we introduce the new metric of *Embedding Efficiency (EE)* as shown in Equation 2. It counts all overlay links that traverse the same underlay link. A maximum underlay disjoint topology has to minimize *EE* according Equation 3.

$$EE(\mathcal{T}) = \sum_{\forall (u,v) \in \mathcal{T}} \Big|\Big(\bigcup_{\forall (w,x) \in (\mathcal{T} \setminus (u,v))} route(w,x)\Big) \cap route(u,v)\Big| \tag{2}$$

$$\mathcal{T}_{disjoint} = min\{\mathcal{T}_{sol} \in 2^M |\, EE(\mathcal{T}_{sol})\} \tag{3}$$

Besides selecting overlay partners for embedding overlay networks to the underlay, careful positioning ALM-Switches can be an additional method to make IPTV more underlay-aware or even underlay-disjoint. Their positioning poses a trade-off in between short delays, a reduction in redundant transmissions and increased robustness.

However, the main challenge in embedding an overlay lies in combining all methods, to construct efficient and robust (regarding overlay and underlay failure) overlays at the same time.

5.2. Division of Stripe Set on Resources

Based on the chosen form of overlay embedding, the *division of the stripe set* on the resources of the system that comprises sources, ALM-Switches and forwarding capacity on clients also largely impacts efficiency and robustness of the resulting streaming overlays. Hence, to keep the requirement of efficient resource usage an allocation $\mathbb{S} \rightarrow 2^M$ is required that optimally utilizes the resources of the system and provides a preferably efficient and robust content distribution.

This requires to carefully select forwarders among the set of participants. In former work [11] we showed that one criteria to build optimally stable topologies (see [3]) is the *one-stripe-only* requirement: Nodes are allowed to ascend in one stripe only and remain a leaf in the others to limit their relevance in the overall system. In a multi-stream scenario we propose to soften this requirement to *one-stream-only*, so that nodes are allowed to ascend in one stripe per stream. Hence, the impact of their possible failure on the respective stream can be limited by still utilizing their spare resources for forwarding stripes besides the stream they subscribed.

Furthermore, for choosing forwarders from the member set it is essential to consider their bandwidth capacity and their past behavior (e.g. mobility) to keep the requirements of agility and user heterogeneity. Mobile members should take over topology positions of low relevance only (e.g. leafs or positions close to leafs in distribution trees). In contrast, members with huge bandwidth should take over positions in which their bandwidth is utilized completely.

5.3. Adaption to User Behavior

ALM systems usually suffer from node churn, i.e. the simultaneous join and leave of huge member populations. In future IPTV systems this gets even worse due to a possibly increasing fraction of mobile nodes and an user behavior known as zapping from classical TV. Hence, an *adaption* of an IPTV system to a dynamic user compound is necessary in order to provide the best possible service on the basis of the employed resources. This additionally hardens the optimization problem at hand. For an adaption, either a proactive or reactive adaption can be applied. Proactive measures allow for a faster adaption, but induce more overhead in contrast to slower but in exchange more economical reactive measures.

Main emphasis by designing a solution for readjusting the system to a changing user behavior has to be put on minimizing reconstruction costs that can be either packet loss, communication overhead or degradations in robustness and efficiency (transmission as well as resource efficiency). Including this considerations to the topology construction process may lead to topologies that are not optimally for specific situation, but that are optimal if observed over a longer time period since they minimize reconstruction costs as given in Equation 4.

$$\mathcal{T}_{min} = min\{\mathcal{T}_i \in 2^M |\, \sum_{\forall \mathcal{T}_j \in 2^M} cost(\mathcal{T}_i, \mathcal{T}_j)\} \tag{4}$$

5.4. Combination of Mechanisms

Combining all of the afore-mentioned like underlay embedding, the division of the stripe set and adaption to the user behavior, a *multi-criteria optimization problem* arises. Since the underlying problems are even

NP-hard, heuristics and approximation algorithms are required to enable a distributed construction of topologies that satisfy the requirements to IPTV given in Sec. 2.

In future work it is required to evaluate the stability of the resulting multi-stream topologies and to check if there is an optimally stable class of them. In a first step it is planned to construct static topologies based on global knowledge and to evaluate them according to instant and greedy attackers as well as the metrics defined in [12]. Afterwards, we plan for a distributed construction of such topologies and to evaluate the construction mechanism by a detailed packet-level simulation. Possible metrics could be the vulnerability of the resulting topologies to internal attackers and as it was done in former work in [4] for a single stream.

6. CONCLUSION

In this article, we summarize requirements to an IPTV system intended for Live-Streaming and Near-Video-On-Demand-Streaming. Based on an analysis of the current state-of-the-art, we propose an own P2P-IPTV architecture and discuss research challenges that needs to be solved before a real deployment.

Especially, maintaining efficient and robust topologies in the presence of overlay and underlay failures poses the main challenge. In the process, we propose to include future reconstruction costs in the topology construction process, so that the resulting topologies may not be the most stable or efficient ones at a specific moments in time, but instead they are most stable or efficient if observed across larger time scales.

Furthermore, we show that the division of the stripe set on the resources of the system and the handling of a constantly changing user compound are essential building blocks of every IPTV system and may largely impact the resulting topologies.

7. REFERENCES

[1] Xinyan Zhang Yum, Jiangchuan Liu, Bo Li, and Tak shing Peter, “Coolstreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming,” in *Proceedings 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, 2005, pp. 2102–2111.

[2] X Hei, C Liang, J Liang, Y Liu, and K W Ross, “Insights into PPLive: A Measurement Study of a Large-Scale P2P IPTV System,” in *In Proc. of IPTV Workshop, International World Wide Web Conference*, 2006.

[3] Michael Brinkmeier, Guenter Schaefer, and Thorsten Strufe, “Optimally DOS Resistant P2P Topologies for Live Multimedia Streaming,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 6, pp. 831–844, 2009.

[4] Michael Brinkmeier, Mathias Fischer, Sascha Grau, and Guenter Schaefer, “Towards the design of unexploitable construction mechanisms for multiple-tree based P2P streaming systems,” in *Kommunikation in Verteilten Systemen (KiVS)*, 2009, pp. 193–204.

[5] Feng Wang, Yongqiang Xiong, and Jiangchuan Liu, “mTreebone: A Hybrid Tree/Mesh Overlay for Application-Layer Live Video Multicast,” *ICDCS ’07. 27th International Conference on Distributed Computing Systems*, p. 49.

[6] Yennun Huang, Yih-farn Chen, Rittwik Jana, Hongbo Jiang, Michael Rabinovich, Amy Reibman, Bin Wei, and Zhen Xiao, “Capacity analysis of MediaGrid: a P2P IPTV platform for fiber to the node (FTTN) networks,” *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 1, pp. 131–139, Jan. 2007.

[7] Shao Liu, Rui Zhang-Shen, Wenjie Jiang, Jennifer Rexford, and Mung Chiang, “Performance bounds for peer-assisted live streaming,” in *ACM SIGMETRICS Performance Evaluation Review*, Annapolis, MD, USA, June 2008, vol. 36, pp. 313–324, ACM.

[8] C. Wu, B. Li, and S. Zhao, “Multi-Channel Live P2P Streaming: Refocusing on Servers,” *IEEE INFOCOM - The 27th Conference on Computer Communications*, pp. 1355–1363, 2008.

[9] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the Scalable Video Coding Extension of the H.264/AVC Standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1103–1120, Sept. 2007.

[10] Michael Brinkmeier, Mathias Fischer, Sascha Grau, and Thorsten Strufe, “Methods for Improving Resilience in Communication Networks and P2P Overlays,” in *Praxis der Informationsverarbeitung und Kommunikation, PIK*. 2009, vol. 32, pp. 64–78, K. G. Saur Publishing.

[11] Andreas Brieg, Michael Brinkmeier, Sascha Grau, Mathias Fischer, and Guenter Schaefer, “Attacker Independent Stability Guarantees for Peer-2-Peer-Live-Streaming Topologies,” in *2009 Second International Conference on Communication Theory, Reliability, and Quality of Service*. July 2009, pp. 20–25, IEEE.

[12] Sascha Grau, Mathias Fischer, Michael Brinkmeier, and Gunter Schaefer, “On Complexity and Approximability of Optimal DoS Attacks on Multiple-Tree P2P Streaming Topologies,” *IEEE Transactions on Dependable and Secure Computing*, pp. 1–12, 2010.