# Using In-Network Adaptation to Tackle Inefficiencies Caused by DASH in Information-Centric Networks

Daniel Posch, Christian Kreuzberger, Benjamin Rainer and Hermann Hellwagner Institute of Information Technology (ITEC) Alpen-Adria-Universität (AAU) Klagenfurt, Austria firstname.lastname@itec.aau.at

# ABSTRACT

The consumption of audio-visual content is the most dominant traffic source in today's Internet. Novel architectural approaches, such as Information-Centric Networking (ICN), are developed to support efficient multimedia dissemination. As ICN and MPEG-DASH have several concepts in common, recent proposals consider a fusion of both technologies. However, MPEG-DASH relies on pure client-driven adaptation. This often rather selfish adaptation strategy inhibits benefits gained from ICN's inherent caching and multi-path transmission capabilities. In order to overcome this challenge, the contribution of this work is the integration of innetwork adaptation (INA) in ICN. We illustrate that INA can be realized despite ICN's content-based security model. Our proposal rests on scalable content, which enables INA without management and transmission overhead.

# **Categories and Subject Descriptors**

C.2.1 [Network Architecture and Design]: Network communications

# **General Terms**

Design, Theory

## Keywords

Information-Centric Networking; In-Network Adaptation; Adaptive Streaming; Multimedia Dissemination

# 1. INTRODUCTION

Information-Centric Networking (ICN) [2] is a novel and nascent approach to deal with the current limitations of the Internet. For this paper, our understanding of ICN is coincident with [11]. While today's host-based communication model has been designed for legacy requirements, ICN is able to address the latest trends. According to Sandvine [20] these trends can be found in the context of real-time

VideoNext'14, December 2, 2014, Sydney, Australia.

Copyright 2014 ACM 978-1-4503-3281-1/14/12 ...\$15.00. http://dx.doi.org/10.1145/2676652.2676653.

entertainment, which mainly encompasses video, audio and image sharing on social or entertainment platforms. This draws the attention of ICN research towards technologies that enable effective dissemination of audio-visual content. Today, prevalent design properties for multimedia distribution frameworks include:

1. Consumer Demands: Consumers expect today's networks to offer specific and individualized services. Demands may vary strongly, e.g., heterogeneous end-devices and different pricing of diverse service quality may drive consumers' expectations.

**2. Transport Efficiency:** Content distribution services depend on efficient delivery. This enables service providers to reduce costs and increases consumer satisfaction due to better service quality.

**3.** Flexibility: Computer networks are shared mediums and provide best effort service only. This requires dissemination strategies to be reactive to deal with resource shortages. Furthermore, consumer mobility, which often causes bandwidth fluctuations, needs to be considered.

One well known and studied state-of-the-art approach for multimedia streaming is MPEG-DASH (ISO/IEC 23009-1). MPEG-DASH is able to ensure properties (1) and (3) using an elegant approach. Content is split into independent segments, which are encoded in various quality levels. The consumer software is responsible for selecting the appropriate segments, which leads to a well balanced trade-off between consumer demands and available network resources.

Due to its effectiveness in IP-based networks, the integration of MPEG-DASH in ICN was investigated [14] [15]. However, reusing MPEG-DASH in ICN makes the realization of (2) rather difficult. ICN-inherent characteristics, such as in-network caching and multipath delivery, were introduced to enhance transport efficiency as compared to classical IP-based networks. Notwithstanding, it is hard for MPEG-DASH to benefit from those properties as its adaptation process is purely client-driven, considering only local parameters, e.g., throughput and/or video buffer fill state.

Recent research [18] has shown that the aforementioned behaviour leads to a problem called *client starvation*: competing MPEG-DASH streams requesting content with diverse popularity gain diverse service quality. In the worst cases this leads to an unacceptable Quality of Experience (QoE) for consumers of unpopular content, while others still are able to consume high quality content. This paper proposes in-network adaptation (INA) to provide all clients with acceptable video quality, regardless of the requested video's popularity. Readily available data (content distribu-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

tion state, network load, etc.) in ICN nodes can be used to guide INA and to tackle *client starvation*.

The remainder of this paper is organized as follows. Section 2 briefly illustrates the concept of INA and presents related work. Section 3 discusses emerging challenges when INA is blended with ICN and outlines potential approaches. Section 4 provides a first implementation of INA in Named Data Networking (NDN) [23], which is a concrete ICN approach. Simulation results of the prototype are provided. Section 5 concludes our findings.

## 2. RELATED WORK

There has been a lot of work dedicated to multimedia services in ICN, which is surveyed in [17]. Either the proposed concepts do not consider adaptive streaming or they consider only pure client-driven adaptive streaming. None of those approaches considered INA as potential technology for ICN. Nevertheless, in traditional IP-based networks several approaches to enable adaptive audio-visual content delivery exist and can be classified into two categories:

**Transcoding** describes a technique of transforming multimedia streams on the fly. This transformation could be a simple conversion of the encoding format, e.g., recoding a video from MPEG-2 to H.264/MPEG-4 AVC, though more complex transformations like spatial or quality scaling are possible. Transcoding offers an unlimited set of adaptation opportunities, but it may be very resource demanding. This limits the range of real-time transformations. Overviews of various video transcoding techniques are given in [3] [16].

Scalable content encodings are another possibility to enable INA by providing multimedia in a layered fashion. The content is split either into a base layer and several interdependent enhancement layers [21] or into completely independent layers [8]. Each additional layer available during playback increases the quality of the given content in some particular dimension(s). It can be assumed that adaptation of scalable content is less resource demanding than transcoding, since adaptation is as simple as discarding enhancement layers. The drawback of this approach is the limited set of adaptation possibilities, which are predetermined at the time the content is encoded. [13] presents an implementation of real-time adaptation of scalable video content on ordinary home routers. [19] illustrates the use case of Scalable Video Coding (SVC) in peer-to-peer streaming.

Recently, Jin et. al [12] proposed PAINT, a PArtial In-Network Transcoding of video streams in ICN. However, their approach does not consider ICN specific security characteristics, which prohibits the manipulation of content at intermediate nodes. Our work differs from [12] as we focus on scalable content encodings and take account of ICN's security model.

## 3. IN-NETWORK ADAPTATION IN ICN

This section deals with the challenges and realization approaches of INA in ICN. We will outline three potential approaches and discuss their properties.

#### 3.1 Challenges

At a first glance the integration of INA in ICN seems as trivial as reusing the working approaches from traditional IP-based networks, which have been presented in Section 2. Unfortunately, it is not, since ICN's communication paradigm is very diverse from the IP-based host-centric view of today's Internet. This requires ICN to use a security model different from today's favoured connection-based one. In ICN data is split into so called chunks, which are small named pieces of a file. Names make chunks self-identifying, and additionally, chunks are authenticated with a digital signature issued by its creator with the help of Public Key Cryptography (PKC). In more detail, the linkage between a chunk's data and its name is authenticated, as shown in Figure 1. The linkage is usually computed by a cryptographic hash function, e.g., SHA-256. The authenticated linkage gives content consumers the possibility to verify the data's origin and protects chunks from manipulation by malicious third parties. If the linkage or the authentication is broken, consumers should reject the received chunk. The challenge is to enable INA without breaking this protection, which will be discussed in detail in the following two subsections.

### **3.2** Adaptation through Transcoding

INA through transcoding is inhibited, since an intermediate network entity can not modify a data packet without invalidating the linkage. Only the publisher is able to modify a packet's content and renew the linkage. Consumers trust content only if its expected origin and its indicated origin (by the signature) match. This fact limits the opportunities for content adaptation through transcoding to two basic approaches:

1. Transfer of Trust: If the content publisher knows trustworthy entities in the network, e.g., operators of Content Delivery Networks (CDN), the publisher could enable them to modify data packets under certain circumstances. For instance, the capability to manipulate packets published under a specified name prefix is given to selected CDN operators. To enable this, two steps are required. First, the adapting entity and the content publisher need to establish trust among each other and agree on which data may be altered, e.g., by signing a contract. Second, the content publisher has to provide information that data signed by the adapting entities is as trustworthy as if it was issued by itself. Then consumers can assume that the modified content is legitimate.

2. Remote Signing: Instead of transferring trust, the second approach tries to re-establish it via a method we call *remote signing*. The content publisher still has to select trustworthy entities that are enabled to manipulate data packets. Once packets are altered, the authenticated linkage is of course invalidated. The trustworthy entity may compute the linkage, but it is not able to authenticate it as the content publisher, since it misses the publisher's private



Figure 1: The linkage between content and name is authenticated, protecting from manipulation.

key. To authenticate the altered data, some form of communication (offline or online) between the trustworthy entity and the publisher is required. E.g., the computed linkage is transmitted back to the publisher, who may sign it and reply with the signature bits.

Both concepts enable data manipulation on a per packet basis, which makes them compatible to classical IP-based transcoding services. We estimate that the signalling overhead for both approaches can be kept rather small. For approach (1) a list of trustworthy entities between content producer and consumer has to be exchanged only once. The overhead for approach (2) would be slightly higher if an online approach is envisaged. In this case for each manipulated packet the corresponding linkage and its signature have to be exchanged among publisher and adapting entity. To reduce this overhead one may aggregate the linkages by the usage of Merkle-Hash-Trees as proposed in the ICN literature [11]. Using an offline approach, e.g., the publisher provides a hardware black box which maintains the private key, would reduce the overhead to zero.

Our main concern with the aforementioned approaches is the resulting management overhead. Both, (1) and (2), require established trust between the publisher and the adapting entity. Considering real world content providers, such as NetFlix or YouTube, they would have to establish trust with dozens of different entities (CDNs, ISPs, etc.), but even worse, the adaptation would not work out of the box. This would make it very hard for small companies, start-ups or even home users to benefit from INA if they want to offer or share content. Our vision is that INA should follow the same principles as in-network caching in ICN does. Any node should be able to perform INA if required, e.g., to ensure network stability, without having to establish trust. This vision led us to the solution presented in the next subsection, which provides INA without transcoding techniques.

#### **3.3** Adaptation through Scalable Content

The idea to use scalable content as enabler for INA in ICN looks promising. As Figure 2 shows, scalable content encodings split raw content into different layers during encoding. These layers may be interdependent or independent depending on the concrete approach used. A consumer needs at least one layer (in case of interdependent layers this layer is distinct and is called the base layer) to successfully decode and display the content. The more enhancement layers a consumer has available, the better the quality of the decoded content will be. This layering approach of scalable content is the key to simple and efficient INA in ICN. Adaptation of multimedia data can be as simple as discarding the enhancement layers. This does not require the manipulation of data, which makes the approach compatible to



Figure 2: Content is encoded in a layered fashion and translated into adaptable segment streams.

ICN's content-based security model. Even if the content is additionally encrypted, adaptation still can be done in the same way by simply discarding packets. This principle has already worked in IP-based networks as shown in [4, 10].

Scalable content enables the design of an inherent ICN adaptation mechanism similar to network-inherent caching. No explicit trust between content publisher and adaptation entity nor trust between consumer and adaptation entity has to be established. The adaptation process is extremely resource saving compared to transcoding approaches. Information which is anyway maintained by ICN nodes can be used to guide the adaptation process and if needed additional information can be gathered by the nodes. Currently the only disadvantage we see is that scalable content formats are less efficient than non-scalable formats in terms of compression efficiency. Compared to the drawbacks that the previously presented approaches based on transcoding have, we consider this disadvantage as minor.

#### 4. USING INA IN ICN

This section deals with our simulation setup and the implementation of INA to tackle the *client starvation* problem identified in [18]. We adopted the described scenario, where 100 MPEG-DASH clients stream ten different videos over a bottlenecked ICN link according to the popularity distribution provided in Table 1, as depicted in Figure 3. The bottleneck node is equipped with an LRU cache, which can buffer 10% of the total content catalogue. For implementation and simulation the NDN approach implemented in ns-3/ndnSIM [1] has been used.

## 4.1 Video Dataset and Client Behaviour

The ten videos requested by the clients are actually only one. The clients and the network consider them as different due to diverse naming. As content we took the open-source movie Big Buck Bunny [5]. The raw 720p YUV source was encoded according to the SVC extension of H.264/MPEG-4AVC [21]. We used the JSVM reference software to encode the content into six different layers using SNR-scalability resulting in the following average bitrates: 1290, 1768, 2220, 2608, 2933 and 3197 kbit/s. The open source tool SVC Demux & Mux [9] has been used to extract the individual layers and to partition them into 300 segments per layer. Each segment represents two seconds of video and is compatible to MPEG-DASH. The reason why we have chosen SVC-based encoding even for the comparative DASH scenario is that ICN caches can be used more efficiently compared to nonscalable content encodings. The client software uses a buffer based adaptation logic as described in [22].

As there is currently no standardized congestion avoidance algorithm defined in the ICN context, clients adopt a TCP-like strategy to additively increase and multiplicatively decrease (AIMD) their data request rate. We justify this step by the latest developments in the Future Internet

Content consumed	by X Clients
Video 0 / Video 1	1 / 1
Video 2 / Video 3 / Video 4	2 / 2 / 2
Video 5 / Video 6 / Video 7	4 / 8 / 16
Video 8 / Video 9	32 / 33

Table 1: Popularity distribution of the videos.

community [6], which argues for a dual-stack deployment of ICN and TCP/IP, rather than a clean-slate approach. This ensures fair bandwidth sharing among the two concepts.

## 4.2 Model for Implementation

For our prototype we have focused on a hybrid approach, which combines client-driven with in-network adaptation in ICN. The basic design approach of this network-guided adaptation is illustrated in Figure 3, which sketches the three different entities of the proposed system:

The **content provider** offers pre-encoded scalable content. It is the provider's responsibility to choose appropriate encoding settings, which allows consumers with heterogeneous requirements and resources to consume the content without running into shortages. Requests received by the provider are simply answered with the corresponding data.

Consumers are enhanced MPEG-DASH clients that request content according to their specified demands by choosing a subset of the available layers. This is possible, since like ordinary DASH-clients, consumers first retrieve the Media Presentation Description (MPD), a manifest file that includes information regarding the content and its layering. Requests are tagged with a *layer information* which allows to identify the specific quality layer the request corresponds to. Requests can be discarded on purpose by the adapting entity if congestion emerges. In this case a client gets feedback by receiving a special not acknowledge (NACK) message. Clients receiving a NACK become aware of the emerging congestion and consider this for their future requests, e.g., by switching to a lower video quality. In our implementation the client reacts with the simplest possible action on a NACK. It stops downloading the current requested segment and revises its decisions according to the previously mentioned BIEB algorithm [22].

The **adapting entity** is an ordinary ICN router enhanced by the capability to perform INA. The adaptation itself is performed by simple packet parsing and packet discarding. For instance, consider Figure 3, where one hundred consumers request diverse content from a single provider. For each of the contents a base layer and two additional enhancement layers are available at the content provider. There is sufficient bandwidth for all consumers to stream the base layer and the first enhancement layer of the demanded content. However, Consumer 100 is greedy and additionally requests the second enhancement layer (indicated by the solid green lines). Since all requests pass through an adapting entity which monitors the requests, that entity notices the



Figure 3: Basic setup and entities for INA in ICN showing requests from consumers, where request(s) are dropped due to congestion.

misbehaviour of Consumer 100, drops the request(s) for the content of the second enhancement layer, and signals this via a NACK message to Consumer 100. This behaviour can be easily implemented, e.g., by classifying requests by layer (base layer, enhancement layer 1, ...) and dropping requests based on a given adaptation policy. Section 4.3 provides an abstract definition of such a policy. For the evaluations performed in this paper, adapting entities use the PerLayerCountingPacketPolicy. This is a concrete approach for a simple packet-based adaptation policy, which is also discussed in Section 4.3.

We are aware that the aforementioned simple classification of requests by layers may lead to problems, e.g., with videos that are encoded with a different number of layers and quality settings. However, this classification is sufficient for the purposes of this paper and does not limit our evaluations in any way. For large-scale deployments a more sophisticated QoE-aware classification for layered video is proposed in [7].

# 4.3 Adaptation Policy

There are many possibilities to design adaptation policies to execute INA on adapting entities. A policy P(m, C)consists of a metric  $m(\cdot) \in [0,1]$  and a classification C = $\{c_0,\ldots,c_n\}$  of request. We define that m=0 signals that the current rate of requests cause no congestion, while any m > 0 indicates that congestion is likely to emerge and the policy should become active. C is required to be a nonintersecting classification of requests, which must be ordered by importance. We assume that  $c_0$  is more important than  $c_1$ , and so on. Based on the importance of each class  $c_i$  the dropping probabilities  $x_i \in [0, 1]$  are defined.  $x_i$  is the probability for a request of class  $c_i$  to be selected for dropping by the adaptation policy P when metric m indicates link congestion. The adaptation operations are defined by the vector  $x = \{x_0, \ldots, x_n\}$ . The goal of the adaptation policy P is to keep m as close to 0 as possible, while dropping a minimum number of requests. Any adaptation policy P that suffices the aforementioned definitions can be used for INA in the described system.

#### PerLayerCountingPacketPolicy

Let maxp be the maximum number of packets a link can handle over a predefined time frame t, and cnt the number of packets requested over the same time frame. We define a metric  $m_{packets}$  as follows:

$$m_{packets} := \max\left\{0, \frac{cnt - maxp}{cnt}\right\} \in [0, 1]$$

Further our policy uses the previously mentioned classification of requests. The base layer is represented by classs  $c_0$ , while the enhancement layers are represented by classes  $c_1, c_2, \ldots, c_n$  in increasing order, where  $c_n$  represents the highest enhancement layer.  $P(m_{packets}, C)$  is a valid adaptation policy  $[m_{packets} = 0 \Leftrightarrow cnt < maxp]$ .

To avoid congestion and therefore random packet discarding in queues, the condition  $m_{packets} = 0$  has to be ensured. To reach this the following inequality must be satisfied by choosing  $x_i \in [0, 1]$  properly:

$$\sum_{i=0}^{n} \underbrace{\left(\frac{cnt_i}{cnt}\right)}_{=p(i)} \cdot x_i \ge \frac{cnt - maxp}{cnt} = m_{packets}.$$
 (1)



Figure 4: Results for client-driven and network-guided adaptation, visualizing the differences for the received average layer and playout stalling time, per bottleneck link speed (median, 50%, 95% quartiles and outliers).

Let p(i) denote the probability for a request to be in class  $c_i$ . This probability can be calculated by the adapting entity by simply counting incoming interests. To avoid any short term spikes a weighted moving average is applied to p(i). Algorithm 1 determines vector x which satisfies Equation 1 in O(n) steps.

<b>Algorithm 1</b> Calculate vector $x$	
1:	Initialize $x: x_i = 0, \forall i \in \{0, \dots, n\}$
2:	$sum \leftarrow 0$
3:	$i \leftarrow n$
4:	repeat
5:	$sum \leftarrow sum + p(i)$
6:	$x_i \leftarrow 1$
7:	$i \leftarrow i - 1$
8:	until $i < 0 \lor sum \ge m_{packets}$
9:	$i \leftarrow i + 1$
10:	if $p(i) \neq 0$ then
11:	$x_i \leftarrow \frac{m_{packets} - sum}{p(i)} + 1$
12:	end if
13:	return x

In lines 1-2 x and sum are initialized. sum represents the left-hand side of Equation 1. Then, starting with class  $n, x_i$  is set to 1 and sum is increased by p(i) until sum is greater or equal to the metric  $m_{packets}$ . Finally, in line 11 the last  $x_i \neq 0$  is determined to strictly satisfy the inequality in Equation 1.

## 4.4 Results

Figure 4 visualizes the results of 30 simulation runs for pure client-driven adaptation (left) and network-guided adaptation (right). The X-axis represents the number of concurrent clients that requested the same video, while the Y-axis shows the playout stalling time and the average received layer. Note that the X-axis represents grouped data. For instance, *Video 2, Video 3* and *Video 4* are requested by the same number of clients and are therefore presented by the same value on the X-axis. The box plots provide the median, 50% quartile (box), 95% quartile (whiskers) and outliers (circles). The figure shows that the more popular a video is, the better the received quality and the lower the number of stalls on average for the streaming clients is.

Exceptions are the videos requested by only one client in the case of 20 Mbit/s with pure client-driven adaptation. They have a higher average quality than videos requested by two clients. Due to the high playout stalling time, those clients receive more bandwidth towards the end, therefore resulting in a higher average quality.

The results of the presented INA approach, as depicted on the right side of Figure 4, visualize that our hybrid approach of using INA and MPEG-DASH together could significantly reduce the stalling time for clients that request less popular videos. For instance, considering the simulation runs with a bottleneck link of 20 Mbit/s, we managed to reduce the median stalling time significantly. For videos requested by only one client the median stalling time was reduced from 209 to 49 seconds, and for videos requested by two clients from 99 to 28 seconds. Of course, this has the side effect that clients requesting popular videos receive a lower average quality. Our results indicate that INA can be used beneficially to enhance multimedia streaming in ICN, considering a TCP-compliant congestion control protocol. With INA, the available bandwidth can be shared more fairly in terms of providing acceptable video quality to all clients. The problem of *client* starvation emerging from pure client-driven adaptation can be relaxed by the usage of INA. There still is room for improvement, on both sides, at the client and in the network. For example, choosing a different classification for layered content may provide a better foundation for INA decisions. A more sophisticated adaptation logic could learn from the network's NACK messages to further improve the results. The open question is how far to go in reducing quality for many consumers, to provide acceptable video quality for all consumers. The answer definitely depends on the business model content providers and ISP's are envisaging for their services.

# 5. CONCLUSION

This paper proposed the integration of INA in ICN. Since INA has not been envisaged in ICN's content-based security model, we presented an approach based on scalable content encodings that enables adaptation of multimedia content within ICN nodes. This approach can be inherently integrated into the ICN architecture, which enables multimedia adaptation as a tool for tackling network resource shortages in a content-aware way. Inefficiencies, such as *client star*vation, emerging from pure client-driven adaptation as envisaged by MPEG-DASH may lead to unacceptable quality experience for some clients. This work proposed the usage of INA to tackle such challenges. We implemented a prototype application that uses hybrid streaming approach based on MPEG-DASH and INA. We could show that blending client-driven and network-guided adaptation relaxes problems such as *client starvation*, providing acceptable video quality to all consumers.

# 6. ACKNOWLEDGMENTS

This work was partially funded by the Austrian Science Fund (FWF) under the CHIST-ERA project CONCERT (A Context-Adaptive Content Ecosystem Under Uncertainty), project number *I1402*.

## 7. REFERENCES

- A. Afanasyev, I. Moiseenko, and L. Zhang. ndnSIM: NDN Simulator for NS-3. Tech. Rep. NDN-0005, 2012.
- [2] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman. A Survey of Information-Centric Networking. *IEEE Communications Magazine*, 50(7):26–36, 2012.
- [3] I. Ahmad, X. Wei, Y. Sun, and Y.-Q. Zhang. Video Transcoding: An Overview of Various Techniques and Research Issues. *IEEE Transactions on Multimedia*, 7(5):793–804, Oct 2005.
- H. K. Arachchi, X. Perramon, S. Dogan, and A. Kondoz. Adaptation-aware Encryption of Scalable H.264/AVC Video for Content Security. *Signal Processing: Image Comm.*, 24(6):468 – 483, 2009.
- [5] Blender Foundation. Big Buck Bunny. http://bigbuckbunny.org/.
- [6] S. Braun, M. Monti, M. Sifalakis, and C. Tschudin. CCN & TCP Co-Existence in the Future Internet: Should CCN be Compatible to TCP? In Proceedings of the 5th IFIP/IEEE International Symposium on Integrated Network Management, 2013.
- [7] B. Fu, D. Staehle, G. Kunzmann, E. Steinbach, and W. Kellerer. QoE-aware Priority Marking and Traffic Management for H.264/SVC-based Mobile Video Delivery. In Proceedings of the 8th ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks, 2013.
- [8] V. Goyal. Multiple Description Coding: Compression Meets the Network. *IEEE Signal Processing Magazine*, 18(5):74–93, Sep 2001.
- [9] M. Grafl. SVC Demux & Mux, 2013. tinyurl.com/ITEC-SVCMux.
- [10] H. Hellwagner, R. Kuschnig, T. Stütz, and A. Uhl. Efficient In-Network Adaptation of Encrypted

H.264/SVC Content. Signal Processing: Image Communication, 24(9):740 – 758, 2009.

- [11] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard. Networking Named Content. In *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '09, New York, USA, 2009. ACM.
- [12] Y. Jin and Y. Wen. PAINT: Partial In-Network Transcoding for Adaptive Streaming in Information Centric Network. In *Proceedings of IEEE/ACM International Symposium of Quality of Service*, 2014.
- [13] I. Kofler, R. Kuschnig, and H. Hellwagner. In-Network Real-Time Adaptation of Scalable Video Content on a WiFi Router. In Proceedings of IEEE Consumer Communications and Networking Conference, 2009.
- [14] S. Lederer, C. Müller, B. Rainer, C. Timmerer, and H. Hellwagner. Adaptive Streaming over Content Centric Networks in Mobile Networks using Multiple Links. In *Proceedings of the IEEE Communications* Workshops (ICC), 2013.
- [15] S. Lederer, C. Müller, B. Rainer, C. Timmerer, and H. Hellwagner. An Experimental Analysis of Dynamic Adaptive Streaming over HTTP in Content Centric Networks. In *Proceedings of the IEEE International Conference Multimedia and Expo (ICME)*, 2013.
- [16] S. Moiron, M. Ghanbari, P. Assunção, and S. Faria. Video Transcoding Techniques. In *Recent Advances in Multimedia Signal Processing and Communications*, pages 245–270. Springer Berlin Heidelberg, 2009.
- [17] G. Piro, L. A. Grieco, G. Boggia, and P. Chatzimisios. Information-Centric Networking and Multimedia Services: Present and Future Challenges. *Transactions* on Emerging Telecommunications Technologies, 2013.
- [18] D. Posch, C. Kreuzberger, B. Rainer, and H. Hellwagner. Client Starvation: A Shortcoming of Client-driven Adaptive Streaming in Named Data Networking. In *Proceedings of 1st ACM Conference on Information-Centric Networking*, Paris, France, 2014.
- [19] N. Ramzan, E. Quacchio, T. Zgaljic, S. Asioli, L. Celetto, E. Izquierdo, and F. Rovati. Peer-to-Peer Streaming of Scalable Video in Future Internet Applications. *IEEE Communications Magazine*, 49(3):128–135, 2011.
- [20] Sandvine. The Global Internet Phenomena Report, 1H 2014. Online: http://sandvine.com/.
- [21] 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*, 17(9):1103–1120, 2007.
- [22] C. Sieber, T. Hoßfeld, T. Zinner, P. Tran-Gia, and C. Timmerer. Implementation and User-centric Comparison of a Novel Adaptation Logic for DASH with SVC. In Proceedings of IFIP/IEEE International Workshop on Quality of Experience Centric Management, 2013.
- [23] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, K. Claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang. Named Data Networking. Tech. Rep. NDN-0019, 2014.