

PR-SCTP for Real Time H.264/AVC Video Streaming

Horacio Sanson

Alvaro Neira

Luis Loyola

Mitsuji Matsumoto

Abstract—In this paper, we evaluate via real experiments the performance of the Stream Control Transmission Protocol (SCTP) and the Partial Reliability extension (PR-SCTP) for transmitting real-time H.264 video streaming over the Internet. This work looks into ways to apply the partial reliability services of PR-SCTP to exploit the implicit temporal scalability of P and B pictures in H.264/AVC video streams. We develop a simple probabilistic model that allow us to assign a limited number of retransmissions, using PR-SCTP, to each H.264/AVC frame type (e.g. I, P or B) before transmission. With our model we achieve a high probability of frame arrival at the receiver while at the same time keeping a low delay and bandwidth.

I. INTRODUCTION

In real-time applications like video conferencing, where delay in packet arrivals degrades the interaction among participants, the use of retransmissions to recover from lost packets has been for long considered very harmful. Nevertheless, in recent years we have seen the emergence of real-time applications like *Video on Demand* and *Live Streaming* that have more relaxed delay constraints and thus the once considered harmful retransmissions are now considered a viable and effective option to compensate for network impairments, as long as they are timely performed.

In fact most streaming solutions today utilize some type of retransmission mechanisms to allow recovery of lost packets. Some servers (e.g. Adobe Flash Media Server) use TCP thus enforcing the retransmission of all lost packets. This feature allows all packets to arrive at the cost of larger delays and inefficient use of network resources especially for video applications that can tolerate some degree of packet losses. Other streaming applications add retransmission mechanisms on the top of UDP (e.g. Darwin Reliable RTP and Microsoft UDP resend) thus contributing with an additional level of complexity to the already complex RTP/UDP scheme. Furthermore, they are not standard solutions and for that reason they raise incompatibility problems between servers and clients. Finally, these methods do not solve any of UDP limitations like the lack of congestion control.

Our research looks into the relatively new transport protocol Stream Control Transmission Protocol (SCTP) with its Partial Reliability extension (PR-SCTP) as a promising alternative to enable retransmission-based reliability to H.264/AVC video streaming applications. PR-SCTP allows developers to configure different retransmission policies at the application layer to each individual packet before transmission. This kind of granular control over the reliability of the transport protocol has been pointed out as a necessary feature to enable better

support for time sensitive applications such as real time video and audio streaming [4].

Previous research on video streaming over PR-SCTP shows that by limiting the number of retransmissions of video packets it is possible to achieve reliability comparable to TCP while reducing the incurred delay. In this paper we evaluate through experiments the suitability of PR-SCTP for H.264/AVC video streaming in different network settings. We find that simply assigning number of retransmissions based on the video frame types (I, P, B) without taking into account the characteristics of the channel, as most previous research did, is not enough to improve the streaming quality. This is because practical issues like the network paths packet loss rate, round trip time delay and video frame size play an important role when selecting retransmission limits.

We have also developed a simple probabilistic model that allows us to find optimum values for the maximum number of retransmissions that offer the best trade off between reliability and delay for H.264/AVC video streaming. The analytical model has helped us to find optimum retransmission numbers for I, P and B frames, through which PR-SCTP achieves more efficient use of resources than TCP and UDP, especially in presence of large delay and high packet error ratio.

The rest of the paper is organized as follows: In section II we present some background on SCTP and H.264/AVC, section III presents related work that also evaluate retransmission mechanisms to improve video streaming quality. Sections IV and V explain in detail the experimental setup and some preliminary results. Finally section VI concludes the paper and points out some future research topics.

II. BACKGROUND

A. The Stream Control Transmission Protocol

The Stream Control Transmission Protocol [9] was designed by the IETF SIGTRAN workgroup to transport call control signaling (i.e. SS7) over packet switched IP networks like the Internet. To accommodate the stringent requirements to transport such signals SCTP inherited the best features of TCP (reliability, congestion control) and UDP (message based), and added new features like *multi streaming* and *multi homing*.

In order to accommodate future rich media applications SCTP was designed in a flexible and extensible manner. This gave rise to several extensions including the Partial Reliability [10], which enables control of different reliability levels on individual packets transported over an SCTP association. Reliability in PR-SCTP is adjusted by setting a limit for the retransmissions of packets using either a maximum number of

retransmissions or a timeout after which the packet will not be retransmitted any more.

B. The H.264/AVC Video Coding Standard

The H.264/AVC video coding standard [7] is the most recent of the standardized video compression technologies. As it's predecessors H.264/AVC can achieve high compression rates thanks to a compression technique called motion compensation. When using motion compensation, similar sections between pictures are encoded and stored once in an Intra Frame (I frame), and only the motion differences (e.g. motion vectors) that indicates the direction and distance these similar sections have translated between the I frame and subsequent pictures are encoded and stored as Predicted (P frame) or Bi-predicted (B frame) pictures.

The difference between P and B frames is that P frames only contain motion vectors referencing previous I or P frames, while B frames contain motion vectors referencing previous and posterior I or P frames. At the decoder, the Intra frame is self contained, which means that all information needed to decode it is contained within itself and subsequent P and B frames are decoded using the previously decoded I and P frames and the motion vectors.

Due to this motion compensation technique, I frames are usually an order of magnitude larger in size than P and B frames and have a more important role at the decoder. For example, the quality degradation caused by a lost I frame propagates to all subsequent P and B frames that depend on it for motion compensation and similarly the degradation caused by a lost P frame propagates to all subsequent P and previous B frames that depend on the lost P frame for motion compensation.

Another feature of H.264/AVC, of greater interest to us, is the introduction of a Network Adaptation Layer (NAL) that offers a layer of abstraction over the actual encoded data [12]

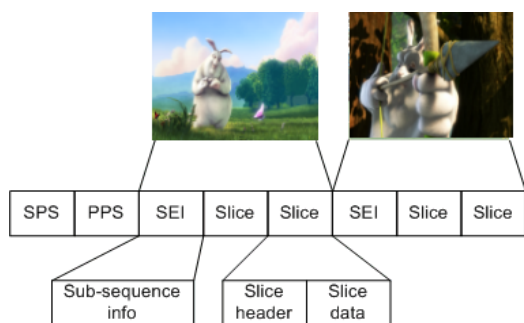


Fig. 1. H.264/AVC NAL Layer Structure

Each video frame (e.g. I, P or B) is encapsulated in self-contained NAL units (NALUs) that contain a header that can be easily inspected and mapped directly over any transport protocol such as RTP/UDP [11] or storage media [6]. The header of each NALU contains decoding parameters and flags indicating its level of importance at decoding time. With this information we are able to send each NAL unit as a PR-SCTP

message with different retransmission limits based on both their type and level of importance without need to decode the whole frame first.

III. RELATED WORK

Retransmission-based error correction for time-sensitive applications has been studied under theoretical frameworks by Podolsky [8] and Bhattacharya [2] but the simplified models they use are not applicable to current streaming applications. In a more practical approach Feamster [5] implemented an RTP variant called *Selective Retransmit RTP* (SR-RTP) that allows retransmission of lost RTP packets and found a significant improvement in perceived quality and achieved frame rate of MPEG-4 coded video by allowing limited retransmission of Intra frames and no retransmissions for P and B frames.

Several researchers [1], [3] have evaluated PR-SCTP for real-time streaming of MPEG-4 and H.264/AVC video sequences. Their work shows that by allowing limited number of retransmission of only I frames or I and P frames allows TCP friendliness and reliability while providing lower delay than TCP.

All these studies limit the retransmissions of video frames by using empirical values like 2000ms timeout or maximum one retransmission for I frames but none tries to find optimal number of retransmissions for the different frame types over different network conditions.

In our research we try to develop a simple model that considers characteristics of the H.264/AVC video to be streamed and the streaming network channel to obtain optimal values for the maximum number of retransmissions for each NAL type and level that results in the best video quality and low delay at the receiver. We further test our model performing several experiments over real networks using a real streaming server/client platform.

IV. EXPERIMENTAL SETUP

A. Streaming Application

We implemented a simplified streaming server and client in pure ANSI C directly on top of a patched version of the PR-SCTP ¹ stack of FreeBSD 7.1. The client was a simple sink that constantly read messages from an SCTP socket and dropped them. The server was implemented to read NAL units of a raw H.264/AVC coded video, assign them partial reliability values based on the NAL type/level and our probabilistic model, and write them as messages to an SCTP socket.

The video sequence used in all experiments was a two minute clip of the open-source movie Big Buck Bonnie encoded with H.264/AVC main profile, 24 fps, 1280x720 resolution and 768kbps bit rate. These encoding parameters are similar to those used by SkillupJapan Corporation to stream multimedia contents to clients. The parameters associated with the sample video are summarized in Table I. The table shows

¹Thanks to Randall Stewart for providing several patches to fix the PR-SCTP stack of FreeBSD.

for each NAL type importance level, H.264 slice type, number of NALs in the sample video clip, and maximum NAL size in the sample video clip.

The video sequence has only four different NAL types: SPS (Sequence Parameter Set) and PPS (Picture Parameter Set) NAL units contain high priority decoder configuration parameters that are necessary to decode subsequent sequence of frames. These are usually transmitted out of band using a reliable transport (i.e. RSTP) to guarantee full delivery. IDR (Instantaneous Decoder Refresh) NAL units mark synchronization points where all previous NAL units can be discarded without affecting the decoding process. In our case all IDR NAL units correspond to Intra coded slices used as reference for subsequent frames. Non-IDR NAL units may contain either P (Predicted) or B (Bi predicted) slices. Since they are assigned different importance level it is possible to determine whether the NAL unit contains a P or a B slice by simply inspecting the NAL header.

B. Network Environment

The experiments were performed in a controlled emulated environment using the Netem Linux emulator. The parameter used in Netem were obtained from actual network measurements of three different paths shown in Table II. The first Chile Link represents a real path over the Internet between an SCTP server in Tokyo and an SCTP client machine in Santiago, Chile. The Tokyo Link represents a communication link set up between central Tokyo and Higashi Murayama, northern Tokyo, a common example of real video streaming applications found at SkillUpJapan Corporation servers. The FSO (Free Space Optics) link is an experimental high speed wireless optical link (2.5 Gbps) deployed at Waseda University in Tokyo between two antennas located at approximately 1 km from each other.

C. Packet Loss Error Ratio Model

To understand the reasoning behind our PER model we first need to understand the mechanics behind SCTP when dealing with varying packet sizes such as video NALUs in our case. SCTP has the ability to bundle or fragment data messages to fit them in SCTP packets no larger than the network channel MTU size (1500 bytes in our case). If one of those packets never arrives to the receiver, the whole SCTP message is lost (a NAL unit in our setup).

If n is the amount of MTU sized packets that fit into a single NAL unit (from either I, P or B video frame) and p

TABLE II
NAL PRIMITIVES OF OUR SAMPLE VIDEO CLIP

RTT (ms)	PER (%)	MTU (bytes)	Name
286	1	1500	Chile Link
9.2	0.138	1500	Tokyo Link
15	6.7	1500	FSO Link

is the characteristic packet error ratio of the end-to-end path, then the probability that at least one packet is lost during the first transmission is

$$1 - (1 - p)^n \quad (1)$$

After the first transmission we expect np packets to be lost and retransmitted, thus the probability that at least one of these retransmitted packets get lost is

$$1 - (1 - p)^{np} \quad (2)$$

after the second retransmission we expect np^2 packets to be lost and retransmitted again, thus the probability that at least one of these are lost is

$$1 - (1 - p)^{np^2} \quad (3)$$

Thus for m retransmissions we get

$$1 - (1 - p)^{np^m} \quad (4)$$

Since packet losses are independent across different retransmission stages, we can approximate the probability of having at least one packet retransmitted $m + 1$ times as

$$\Phi = \prod_{i=0}^m (1 - (1 - p)^{np^i}) \quad (5)$$

Subsequently, the probability of not having even one packet retransmitted $m + 1$ times is $1 - \Phi$. This probability is equivalent to the probability of having all n packets from the NAL unit successfully received after m retransmissions. So the probability of successful NAL transmission can be approximated by equation 6.

$$P_m = 1 - \prod_{i=0}^m (1 - (1 - p)^{np^i}) \quad (6)$$

Using Netem we performed several streaming experiments using an emulated channel with packet loss ratios from 0% to 10% and a RTT of 15ms using one, two and three retransmission maximum for all three frame types I, P and B. The resulting frame loss probabilities of these experiments were averaged and plotted along our model. As we can see from figures 2, 3 and 4, the model fits well the real behavior.

TABLE I
NAL PRIMITIVES OF OUR SAMPLE VIDEO CLIP

NAL Type	Level	Slice Type	Quantity in the file	Max Size
IDR Slice	3	I	24	125000
SPS	3	none	24	16
PPS	3	none	24	16
Non IDR	2	P	439	51000
Non IDR	0	B	2417	24000

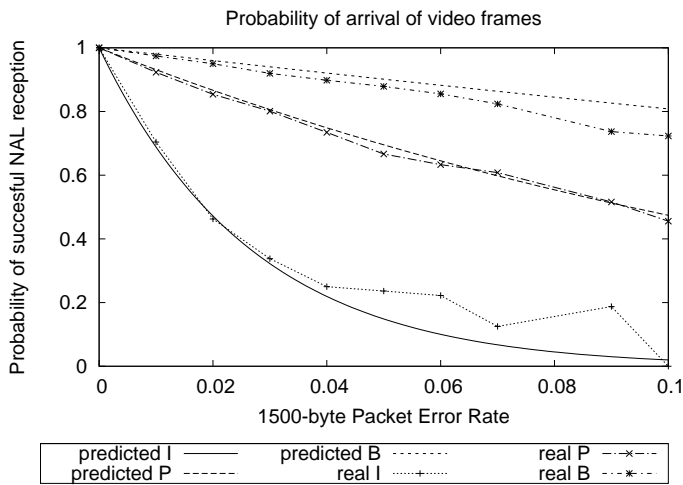


Fig. 2. I, P and B frames successful transmission ratio with one retransmission

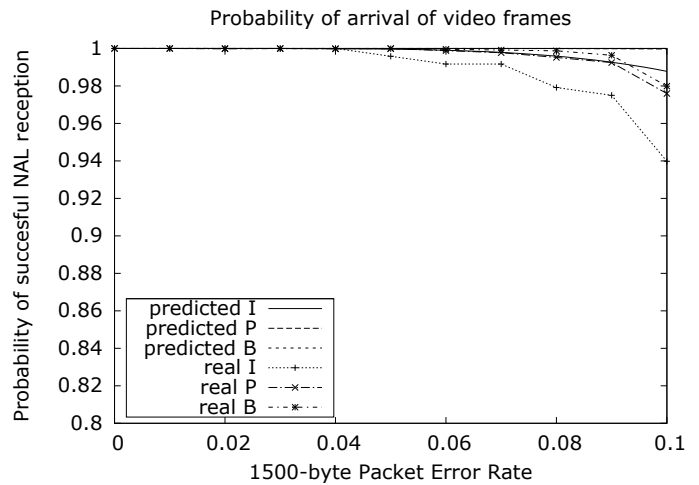


Fig. 4. I, P and B frames successful transmission ratio three retransmissions

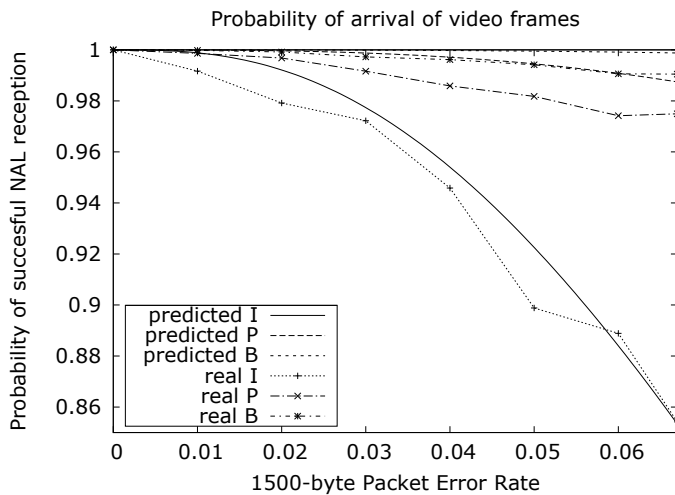


Fig. 3. I, P and B frames successful transmission ratio with two retransmissions

V. RESULTS

Based on the network settings shown in table II we carried out an extensive set of experiments using four retransmission policies: (1) Reliable SCTP that allows infinite number of retransmission like TCP; (2) Unreliable SCTP that allows no retransmissions like UDP; and two partial SCTP policies (3) SCTP-3:2:1 and (4) SCTP-1:1:1 where 3:2:1 and 1:1:1 denote the number of retransmissions for I, P and B frames, respectively. For example the policy SCTP-3:2:1 means three retransmissions for I slices, two for P slices and one for B slices.

In all experiments SPS and PPS NAL units are transmitted with unlimited retransmissions to mimic the behavior of current streaming servers that send these NAL units out of band using a reliable transport such as TCP.

For each experiment we obtain a late index value as the difference between the arrival time and the projected playback time of each complete NAL unit. The projected playback

time is calculated from the NAL unit presentation order count (POC) and the video frame rate minus a five second delay to emulate an initial 5-second buffering period at the receiver.

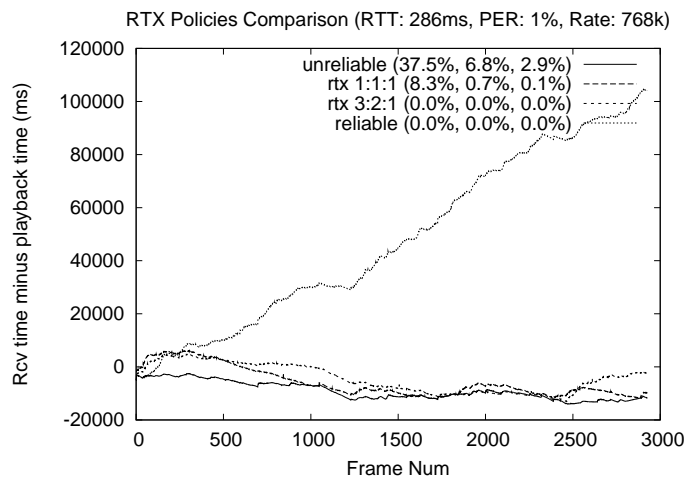


Fig. 5. Chile Link experiment.

Figure 5 shows the frame late index as a function of the frame POC for the Chile Link from table II. The chart shows how in the reliable SCTP policy case very similar to TCP almost all NAL units arrive after their corresponding playback time, resulting in frequent and long buffering periods. In the other extreme, the unreliable SCTP policy similar in nature to UDP has a very good late index but an extremely bad performance from the video frame loss perspective. The unreliable SCTPs video frame error ratio attains 37%, 6% and 2% for I, P and B frames, respectively, resulting in an unplayable video.

In contrast to both reliable and unreliable SCTP policies, figure 5 shows the excellent result achieved using our optimum PR policy SCTP-3:2:1, which obtains no frame losses while still maintaining a relatively low late index value.

Figure 6 shows the experimental results for FSO link from table II. It can be seen that in all cases an initial buffering

time of less than five seconds is required. However, we also see that in the unreliable case the video frame loss ratio gets very high values: 87%, 39% and 15% for I, P and B frames, respectively, resulting into an unplayable movie. The reliable SCTP and the SCTP-3:2:1 policies have comparable frame loss ratios with SCTP-3:2:1 presenting a slightly larger 1% loss for B frames. With both policies the late index is below zero which means that all frames arrive on time for playback but for the reliable case we require an initial buffering period of at least five seconds over the emulated 5-seconds buffer i.e. 10 seconds in total - while SCTP-3:2:1 needs only three seconds i.e. 8 seconds in total.

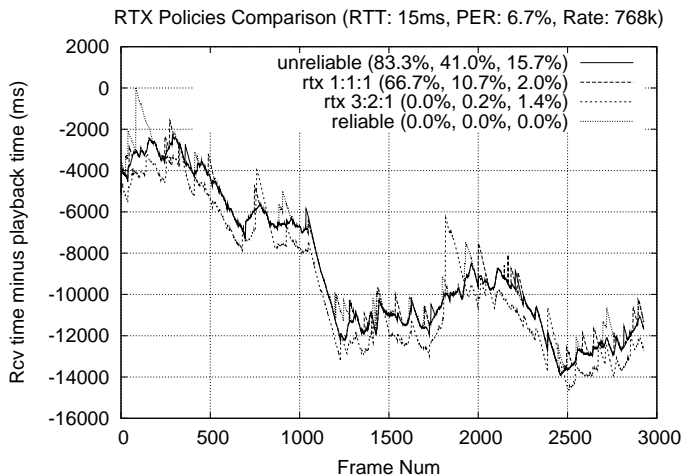


Fig. 6. FSO experiment.

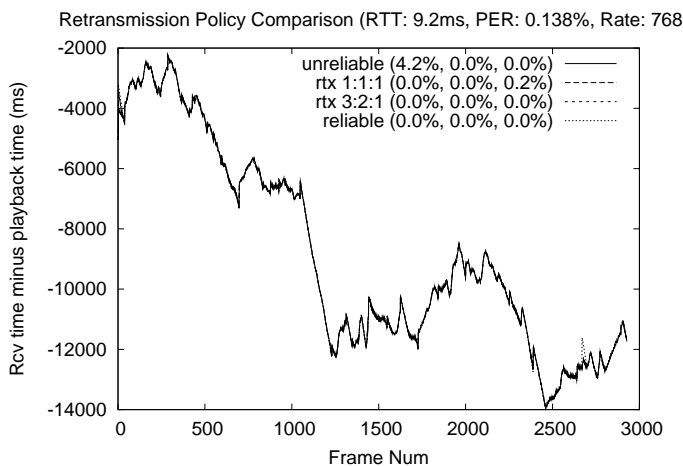


Fig. 7. Tokyo link.

Figure 7 shows the results obtained in the Tokyo Link. As it can be seen in the chart all reliability mechanisms basically perform equally in terms of video frame loss ratio and late index with only the unreliable one featuring a 4% loss ratio of I frames. From these results we can conclude that for very low RTT links with very low packet error ratio all transport protocols perform very similarly. However, as the packet error ratio and RTT increase PR-SCTP show better results in terms

of late index and video frame losses than both reliable and unreliable SCTP.

VI. CONCLUSIONS AND FUTURE WORK

From our analytical model and experimental results we can conclude that by using PR-SCTP to limit the maximum number of retransmissions for each NAL unit of a H.264/AVC video we can achieve reliable delivery similar to TCP while maintaining equal or lower delay in all cases. We found that the SCTP-3:2:1 policy is optimal for our sample video clip and tested network settings but this tailored number of retransmissions logically depends on both NAL unit size statistics in the video and network path characteristics. Based on this observation it is clear that adaptive mechanisms that can dynamically adjust the retransmission policy in response to changes in network conditions (e.g. congestion) or changes in the media itself (e.g. bitrate adaptation) are necessary to cover all possible situations.

During this research we only evaluated partial reliability by limiting the maximum number of retransmissions. Now we are working on limiting the number of retransmissions using timeouts (timed partial reliability). The audio stream, which is also important for the quality of the whole video experience, will be addressed in the future. We are also exploring other SCTP features like multi-stream and multi-homing to improve current video streaming applications.

REFERENCES

- [1] A. Argyriou and V. Madiseti. Streaming h.264/avc video over the internet. pages 169–174, Jan. 2004.
- [2] P.P. Bhattacharya and A. Ephremides. Optimal scheduling with strict deadlines. *Automatic Control, IEEE Transactions on*, 34(7):721–728, Jul 1989.
- [3] Ashfiqua T. Connie, Panos Nasiopoulos, Yaser P. Fallah, and Victor C.M. Leung. Sctp-based transmission of data-partitioned h.264 video. In *WMuNep '08: Proceedings of the 4th ACM workshop on Wireless multimedia networking and performance modeling*, pages 32–36, New York, NY, USA, 2008. ACM.
- [4] R. Droms et al. Report from the joint siggraph/sigcomm workshop on graphics and networking ' 1991.
- [5] Nick Feamster and Hari Balakrishnan. Packet loss recovery for streaming video. In *12th International Packet Video Workshop*, 2002.
- [6] ISO. *ISO/IEC 14496-14:2003 - Coding of audio-visual objects Part 14: MP4 file format*. ISO recommendation. International Organization for Standardization, Geneva, Switzerland, 2003.
- [7] ISO. *ISO/IEC 14496-10:2008 - Coding of audio-visual objects Part 10: Advanced Video Coding*. ISO recommendation. International Organization for Standardization, Geneva, Switzerland, 2008.
- [8] M. Podolsky, M. Vetterli, and S. McCanne. Limited retransmission of real-time layered multimedia. pages 591–596, Dec 1998.
- [9] R. Stewart. Stream Control Transmission Protocol. RFC 4960 (Proposed Standard), September 2007.
- [10] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, and P. Conrad. Stream Control Transmission Protocol (SCTP) Partial Reliability Extension. RFC 3758 (Proposed Standard), May 2004.
- [11] S. Wenger, M.M. Hannuksela, T. Stockhammer, M. Westerlund, and D. Singer. RTP Payload Format for H.264 Video. RFC 3984 (Proposed Standard), February 2005.
- [12] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 13(7):560–576, July 2003.