Ultra High Resolution Jitter Measurement Method for Ethernet Based Networks

Karel Hynek
DDD & Department 706
CTU FIT & CESNET a.l.e.
Prague, Czech republic
hynekkar@fit.cvut.cz

Tomáš Beneš
DDD
CTU FIT
Prague, Czech republic
benesto3@fit.cvut.cz

Matěj Bartík
DDD & Department 706
CTU FIT & CESNET a.l.e.
Prague, Czech republic
matej.bartik@fit.cvut.cz

Pavel Kubalík
DDD
CTU FIT
Prague, Czech republic
pavel.kubalik@fit.cvut.cz

***Abstract*—This document presents a new approach to network jitter measurement and analysis in asynchronous data networks such as Ethernet. The developed monitoring device is capable to analyze an incoming stream speed of 1 Gb/s with the resolution up to 8 ns. The system architecture supports speeds up to 100 Gb/s networks. The presented architecture can provide several statistical functions such as measuring a network jitter by Interarrival Histograms method providing the mean value and peak-to-peak value as well. The architecture was implemented and tested on Xilinx Kintex UltraScale FPGA chip using Avnet AES-KU040-DB-G development board.**

***Index Terms*—network, jitter, measurement, high, resolution, ethernet, FPGA, Xilinx, interarrival, histograms, statistical**

I. Introduction

Customers are demanding more and more multimedia data from online services such as Netflix or YouTube which are generating more than 50% of Internet traffic in North America region [1]. These services are working on unidirectional (downstream) principle where various kinds of buffers [2] can be used for masking negative properties of communication network such as jitter or latency. A large buffer could mitigate the network jitter trading the video signal delay (latency) against an original video source. The buffer size is growing rapidly with order of video stream resolution (4K and future 8K) despite a video compression scheme (such as H.264/H.265) is being used. The overall latency could be in order of seconds and it will be still acceptable. A problem arises in bidirectional streaming use cases such video conferencing systems or online games streaming platforms where it is essential to communicate in real-time. Keeping jitter and latency as low as possible is the crucial for successful data transmissions across/between continents. The knowledge of network jitter in particular cases could help us to diagnose and optimize the network settings or to tweak the transmission systems (reducing buffer size).

II. Motivation of Professional Systems for Audio/Video Streaming

In this section, we would like to describe some use cases where a network jitter could case a failure of a real-time multimedia transmission used for safety-critical purposes such as surgery telepresence [18] or remote controlled planes/drones.

978-1-7281-0554-3/19/$31.00©2019 IEEE

The knowledge of precise jitter value is essential parameter for the minimal possible latency on the network. This is achieved by adjusting the compensation buffers in the network using the jitter value.

Network devices, which are used for low latency transmissions, are using the SDI (Serial Digital Interface) infrastructure in the industry/professional sphere. There are several SMPTE standards defining a process of embedding SDI data into a different protocol/interface such as an 10G/1G Ethernet [14]. An another SMPTE standard also defines an inter-packet gap should remain constant during a transmission [11].

The particular example of such professional device is MVTP platform [17], which has an additional latency of 1 ms (from SDI to Ethernet and back). Devices based on the MVTP platform are widely used for remote collaboration in real-time, for example musical performances where musicians can be located virtually anywhere or for a surgery telepresence [18].

III. State of the Art

Jitter and transmission delay measurement in an Ethernet network with milliseconds accuracy is a relatively easy problem solved many times before. The difficulty changes rapidly when we want achieve micro or even nanosecond precision.

There were several attempts of high-resolution jitter measurement in publicly available network cards presented as Interarrival Histograms method [3]. This method is based on a continuous stream of equally distanced UDP packets of given rate and size, where receiving network cards are time-stamping incoming packets. A network jitter (delay difference) can be derived from Inter-packet gap calculations done afterward by a software [3].

Interarrival Histograms method were successfully tested on 802.11 network cards with a resolution of 1 μs. For application in WLANs is this kind of precision satisfactory, because the inter-frame intervals are in 802.11 networks defined in microseconds.

Another method is based on encapsulating time-stamps into packets [6], which is usually realized in software. This technique uses the loop-back hierarchy across an optical network to measure the delay of it. The receiver was configured as low latency 10G Ethernet mirror and all evaluations were made on the transmitter. This technique is capable to achieve high accuracy of 10 μs.

Encapsulated time-stamps can be also used for one-way jitter measurement. In this case, evaluation has to be made on a receiver. This brings a new problem of time synchronization between the sender and the receiver. The *Precision Time Protocol* (PTP) [9] can achieve up to 100 ns precision in Ethernet based networks [8]. On the other hand, PTP requires the IEEE 1588-2008 standard support of all network elements along the way between sender and receiver to satisfy time correction requirements [7].

Older, but more often used protocol NTP (Network Time Protocol) is capable of reaching a precision of 1 ms [8]. Network delay evaluation techniques, which are based on time synchronization protocols, cannot be more accurate then these protocols themselves.

The mentioned (software based) approaches are less accurate than our hardware oriented architecture. Some good results can be achieved using PTP [9] in combination with Synchronous Ethernet [10]. The White Rabbit project reached sub-nanosecond precision in time synchronization [12]. However, the Synchronous Ethernet is not commonly used and its application is limited for special cases only.

A hardware-based architecture is the only way to achieve a better precision in common Ethernet networks. An FPGA based wireless card has been developed for testing other 802.11 network cards [5]. These Wi-Fi cards can record time with resolution up to 50 ns. This implementation is targeted only on testing 802.11 network cards and it's unsuitable for general measurements. To our best knowledge, there is no solution capable of analyzing jitter with precision below 10 ns.

IV. Theoretical background

Interarrival Histograms method allow precise jitter estimation only on the receiver side. This method requires a data stream of equally distanced UDP packets with fixed rate and size. Any data streams that meet this condition can be used [3].

A. Measurement by Interarrival Histograms method

Network jitter is calculated as the difference between the received inter-packet gap and source's transmit inter-packet gap. In case we don't know the precise inter-packet gap of the source, we have to make an estimate. Let us briefly describe the math behind the estimation.

Let Ts_i represent an i^{th} inter-packet gap of the source, Tr_i represents i^{th} gap of the receiver and D_i represents an i^{th} network delay variation, then we can use this simple equation.

$$Tr_i = Ts_i + D_i \tag{1}$$

According to the Interarrival Histograms method [3] Ts_i is constant and has to be estimated by receiver. We assume the inter-packet gap is a random variable with Normal distribution which has expected value ($\mathbb{E}$) equal to source's transmit inter-packet gap. With the appropriate statistical sample, we can estimate the transmit gap Ts on the receiver side by the sample mean of measured gaps Tr_i [3].

Let $D_i \sim \mathbb{N}(0, \sigma)$ then

$$\frac{1}{n}\sum_{k=0}^{n}(Tr_i) = \frac{1}{n}\sum_{i=1}^{n}(Ts_i + D_i) = Ts + \frac{1}{n}\sum_{i=1}^{n} D_i \sim T_s \tag{2}$$

We can calculate jitter D_i with these acquired values as

$$D_i = Ts - Tr_i \tag{3}$$

B. Method accuracy

The precision of the method depends on the accurate knowledge of the inter-packet gap. The gap is either known or is estimated using the Sample mean. The precision of the estimation and magnitude of jitter itself are the main limiting factors for the jitter accuracy. The precision of the estimation depends on the number of processed samples for the estimation.

Given that jitter is a Random variable with Normal distribution $\mathbb{N}(0, \sigma)$ [3], we can use confidence interval as our error function. Following estimation can be used to achieve a confidence level of 99.7 % with a large number of samples.:

$$\epsilon = 3 \cdot \frac{\overline{S}}{\sqrt{N-1}} \tag{4}$$

Symbol ϵ represents the error estimate, $\overline{S}$ standard deviation of the samples and N number of samples [3].

The sample standard deviation is closely related to maximum expected jitter in a network. For the estimation of the deviation, we can follow the three σ rule [13].

The rule states that 99.7 % of $|\mu - X_i|$ are below the value of 3σ (three times Standard deviation) [13]. Where X_i is a value of a Normally distributed statistical sample $\mathbb{N}(\mu, \sigma)$.

The standard deviation $\overline{S}$ can be substituted (Eq. 4) and limited by using the three σ rule resulting in:

$$\epsilon \leq \frac{1}{2} \cdot \frac{J}{\sqrt{N-1}}. \tag{5}$$

Where J represents maximum possible jitter in the network, so it can be obtained by either low precision measurement or qualified estimation.

C. Oscillator accuracy

Consideration of the influence of the oscillator frequency deviation on the measurement accuracy is part of any high precision time measurement.

The frequency stability deviation is accumulated and added to measurement error during estimation of the inter-packet gap. The oscillator short term frequency stability has Normal distribution with Expected value equal to zero [4]. This error can be determined together with our measurement error, because the characteristics of frequency stability deviation and network jitter are the same.

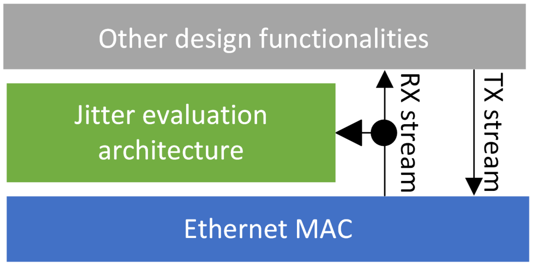

Fig. 1. Location of our jitter evaluation architecture in general design.

V. Our approach

In the following section, we present an FPGA based architecture for high precision jitter measurement. The proposed solution is based on Interarrival Histograms method. The target application is evaluation of video streams compliant with SMPTE ST 2110-21:2017, but any stream, that can guarantee equally distanced UDP packets with fixed rate and size, can be used. The standard guarantees that both of the requirements given by Interarrival Histograms method are satisfied [11]. It is commonly supported by professional media over managed IP networks.

A. Architecture design

Our architecture is compatible with network IP cores using AXI4 stream. It can be easily placed between the MAC core and rest of a design without any added latency (see Fig. 1). Our architecture has an AXI4-Lite interface for configuration and gathering results.

The function of our architecture can be separated into two phases. The first is an estimation phase, where the interpacket gap is estimated. The duration of this phase is given by the configuration variable. The accuracy of the measurement (In section IV-B) is determined by the duration of the estimation phase. The second is an evaluation phase, where the jitter is measured using the inter-packet gap period obtained in the previous phase. The estimation phase can be skipped with knowledge of exact inter-packet gap for the jitter measurement.

The architecture of the unit is divided into four blocks (see Fig. 2). The RX stream filter block chooses measurement packets, Inter-packet gap estimate block which is used in the estimation phase, Jitter evaluation block which then uses the estimate in the evaluation phase and result memory.

1) RX stream filter block: This block provides compatibility for both IPV4 and IPV6 protocols and different Ethernet speeds (10/100/1000 Mb/s). It also applies filters for a specific port and IP address to separate measurement packets from the rest of the communication. When the packet meet the criteria of the filter, then the control signal *Packet match* is generated. The control signal is always generated with constant latency from first received byte to prevent any result distortion.

2) Inter-packet gap estimate block: The source's Interpacket gap can be specified either manually or estimated from incoming packets. In the case of setting the estimate manually the block only propagates this estimate value to the Jitter evaluation block. Otherwise the architecture estimates the value according to the equation (Eq. 2), following the Interarrival Histograms method [3]. This estimation is implemented by the clock accumulator, the accumulator of arrived packets and the division unit. After reaching the desired number of processed packets in the arrival accumulator, the block stops the calculation to keep results in the evaluation phase consistent.

3) Jitter evaluation block: Jitter is computed as a difference between the value from the estimate block and received interpacket gap measured by the clock counter. The calculation is controlled by the *Packet match* signal, afterward the result is processed by averaging, peak to peak measurement and stored in the Result memory block. The average and peak to peak jitter values can be accessed via the AXI4-lite interface.

4) Result memory block: The memory is implemented as a circular memory buffer for a certain number of latest jitter values. These values can be processed later by more advanced software analysis. The AXI4 lite interface and the Jitter evaluation block are accessing the result memory concurrently. This could lead to data being updated during the read cycle and loss of their time continuity. The write operation of Jitter evaluation block can be disabled during the read cycle to prevent this problem.

The presented hardware architecture focuses on implementing real-time measurements of jitter. It features software access to precise values of the jitter for advanced software calculation. This reduces the number of calculations done by the hardware and the overall usage of FPGA resources to a minimum.

B. Accuracy of our architecture

The precision of our measurement depends on the unit's clock frequency and accurate knowledge of the inter-packet gap. The gap can be estimated by the unit or specified manually.

The unit's clock frequency is the main limiting factor in case of manual setting (the known inter-packet gap). Our architecture is limited by the maximal frequency of an Ethernet MAC core. The MAC core transfers packet data directly from the physical interface (an Ethernet port) in the form of wordstream. We are able to detect jitter with accuracy up to the resolution of the word-stream frequency despite this limitation.

In case the inter-packet gap is unknown, the precision of estimate and magnitude of jitter itself are the main limiting factors. The estimation depends on the number of samples processed in the estimation phase as mentioned in section IV-B.

The overall error of the measurement is determined by the number of used samples and by the architecture of the measurement. The table II presents the expected measurement accuracy of a common 1G Ethernet MAC with a 125 MHz frequency.

TABLE I
Resources utilization of jitter evaluation block on Kintex UltraScale (XCKU040-FBVA676).

Name	Used [1]	Utilization [%]
CLB LUTs	6,272	2.59
CLB registers	15,094	3.11
BRAM	7.5	1.25

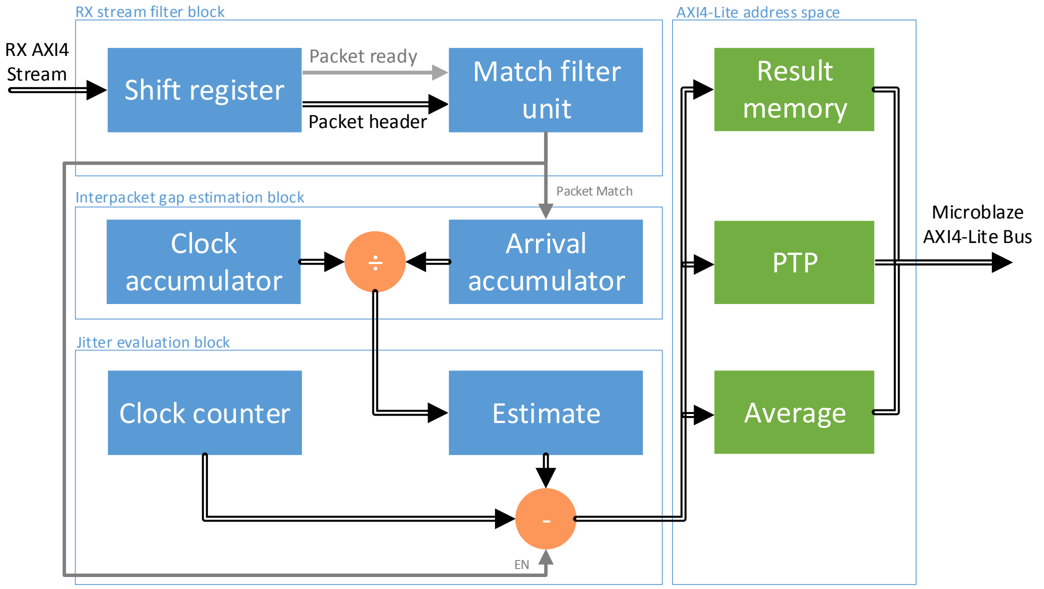

Fig. 2. Block scheme of the proposed architecture.

TABLE II
SAMPLE SIZE VALUES DEPENDING ON EXPECTED JITTER AND MAXIMUM ESTIMATED ERROR

Expected jitter	100 ns	1 μs	10 μs	100 μs	100 μs	1 ms	1 ms	1 ms
Maximum estimated error	8 ns	8 ns	8 ns	32 ns	64 ns	128 ns	256 ns	512 ns
Number of samples	50	3,907	390,625	2,441,410	610,000	15,258,789	3,814,697	953,674

VI. Validation and Experimental Results

In this section, we present results of our measurement architecture. The proposed design was implemented on Xilinx Kintex UltraScale FPGA *(XCKU040-FBVA676)* chip, using Avnet AES-KU040-DB-G board with 1G Ethernet. The utilization of FPGA is depicted in table I. The utilization of the FPGA resources is under 4 %, which shows that the unit is compact and can be added to almost any existing design.

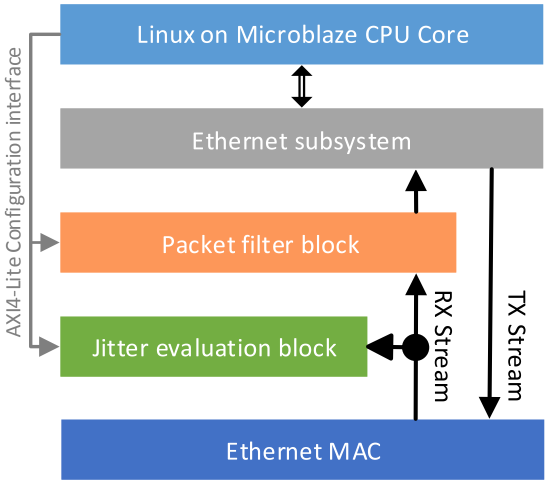

Fig. 3. Hardware platform designed for testing and measurement.

For testing and measurement, we designed a hardware platform using the MicroBlaze soft processor interconnected via an AXI4-stream and AXI4-lite interface to the Xilinx's Tri-Mode Ethernet MAC IP core (Fig. 3).

Our jitter evaluation unit is placed behind the Ethernet MAC core, with a result memory of 32 KB size and a clock source with frequency of 125 MHz from the Ethernet MAC, which allows us to reach precision of 8 ns. To prevent OS/CPU overloading, we put a packet filter block between the tested unit and the CPU. The packet filter block drops the measurement packets.

Estimated measurement errors with expected jitter for our presented architecture on 1G Ethernet are in the table II. The estimated precision is 8 ns, when the expected jitter is up to 10 μs with maximum number of samples of 390,625.

The table II presents the expected measurement accuracy of a common 1G Ethernet MAC with a 125 MHz frequency.

To make the testing process simple, we decided to create a custom Petalinux for MicroBlaze CPU. We have also implemented user-space Linux driver for setting jitter-measurement unit and reading the results.

We performed the experimental evaluation of the precision of the measurement in laboratory conditions using programmable high precision packet generator. The implemented packet generator design runs on the same Avnet's board to eliminate timing uncertainty in case a different physical layer is being used. The packet generator was connected directly to our test hardware platform. This environment provides zero jitter connection between boards, which we used to test out jitter measurement with a known value generated by our second board.

At first, we measured packet stream without added jitter to verify the connection between both boards. The packet generator was sending 1,000 UDP packets per second with 1 ms inter-packet gap size. The receiver was configured to estimate inter-packet gap with 50 samples. We verified the designed architecture estimated successfully the inter-packet-gap size and zero average jitter with a peak to peak of 8 ns (1 clock cycle), because we set up these parameters manually on the packet generator.

The packet generator was configured to send packets with a certain jitter. The network delay values were randomly generated from the Normal distribution $\mathbb{N}(0, 100)$ which are stored in FPGA memory. Parameters of jitter sent by the generator are depicted in the table III.

TABLE III
GENERATOR CONTROLLED JITTER PARAMETERS

Name	Value
Average jitter	653.584 ns
Standard deviation	493.314 ns
Peak to peak	5,216 ns

Duration of the estimation phase on the receiver was set to 4,000 samples. The measurements results are presented in table IV. The difference column show percentage difference between generator jitter and measured jitter. All differences are under 1%, which shows, that the unit can measure jitter with a high precision even on a network with a significant jitter.

TABLE IV
MEASURED VALUES WITH ADDED JITTER

Name	Value	Clocks	Generator	Difference
Est. inter-packet gap	1 ms	–	ms	–
Average jitter	648 ns	81	653.584 ns	0.854 %
Peak to peak	5,224 ns	653	5,216 ns	0.152 %
Last 8K samples mean	654 ns	–	653.584 ns	0.063 %
Last 8K samples Std. dev.	493 ns	–	493.314 ns	0.063 %

In case of the inter-packet gap of the source is known, the precision would be the same or even better. The architecture resolution is substantially higher than reference clock period in case the condition of minimum number of samples is satisfied.

VII. Conclusion

We presented an architecture for high precision jitter measurement in asynchronous networks using Interarrival Histograms method. We analyzed other techniques used for measuring a jitter and their precision.

Our proposed architecture reached the precision up to 8 ns, which is superior to competitors [5] by the order of magnitude. The precision of measurement was experimentally verified on our testing hardware platform with the Xilinx Kintex UltraScale FPGA with our high precision packet generator on a 1G Ethernet network.

This precision is suitable for a deep analysis of a network which can result into optimizing sizes of various buffers. This is very important in surgery application and other real-time remote collaboration systems. The FPGA based device is great for analyzing video streams compliant with SMPTE ST 2110-21:2017 standard but it can be used for any application where a data stream has equally distanced UDP packets with a fixed rate and size.

Our architecture is portable to any modern FPGA chip including low cost Xilinx 7-Series (for example an Artix-7 FPGA). To our best knowledge, there is no solution capable of analyzing jitter with better precision then our architecture.

Acknowledgment

This research has been partially supported by the CTU project SGS17/017/OHK3/1T/18 ”Dependable and attack-resistant architectures for programmable devices” and by the project ”E-infrastructure CESNET – modernization” no. CZ.02.1.01/0.0/0.0/16 013/0001797.

References

[1] Half of all internet traffic goes to netflix and youTube, 9 July 2018, [online] https://testinternetspeed.org/blog/half-of-all-internet-traffic-goes-to-netflix-and-youtube/ [Accessed 9 November 2018]

[2] R. Ashwin, A. Legout, Y. Lim, D. Towsley, C. Barakat, W. Dabbous, Network characteristics of video streaming traffic, Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies, pp.1-12, December 06-09, 2011, Tokyo, Japan. doi: 10.1145/2079296.2079321

[3] G. Berger-Sabbatel, Y. Grunenberger, M. Heusse, F. Rousseau, A. Duda, Interarrival Histograms: A method for measuring transmission delays in 802.11 WLANs, (2007).

[4] D. Elad and A. Madjar, ”Short term oscillator frequency stability as a function of its phase noise distribution,” 21st IEEE Convention of the Electrical and Electronic Engineers in Israel. Proceedings (Cat. No.00EX377), Tel-Aviv, Israel, 2000, pp. 129-133. doi: 10.1109/EEEI.2000.924342

[5] A. Di Stefano, G. Terrazzino, C. Giaconia, FPGA implementation of a reconfigurable 802.11 medium access control, International Conference on Wireless Reconfigurable Terminals and Platforms (WIRTEP), Rome, April 2006.

[6] M. Freiberger, T.J. Xia, D.L. Peterson, M. Carroll, and G. Wellbrock, Demonstration of a novel method for real-time network latency measurements in the optical transport network using G.709 overhead, 2009

[7] IEEE standard for a precision clock synchronization protocol for networked measurement and control systems,” in IEEE Std 1588-2008 (revision of IEEE Std 1588-2002), pp.1-300, 24 July 2008

[8] S.T. Watt, S. Achanta, H. Abubakari and E. Sagen, Understanding and applying precision time protocol, 2015

[9] IEEE standard for a precision clock synchronization protocol for networked measurement and control systems, IEEE Std. 1588-2008, 2008

[10] Timing characteristics of a synchronous Ethernet equipment slave clock (EEC), ITU-T Std. G.8262, 2007.

[11] SMPTE ST 2110-21:2017. Professional media over managed IP networks: traffic shaping and delivery timing for video.

[12] M. Lipinski, T. Wlostowski, J. Serrano and P. Alvarez, ”White rabbit: a PTP application for robust sub-nanosecond synchronization,” 2011 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication, Munich, 2011, pp. 25-30.

[13] N.V. Smirnov, I.V. Dunin-Barkovskii, ”Mathematische statistik in der technik”, Deutsch. Verlag Wissenschaft. (1969) (Translated from Russian)

[14] ST 2022-6:2012 - SMPTE Standard - Transport of High Bit Rate Media Signals over IP Networks (HBRMT),” in ST 2022-6:2012 , pp.1-16, 9 Oct. 2012 doi: 10.5594/SMPTE.ST2022-6.2012

[15] M. Miyazaki, T. Sakiyama, T. Yamashita and M. Van Dorpe, ”8K-TICO codec for miniaturized and simplified UHDTV production systems,” SMPTE 2017 Annual Technical Conference and Exhibition, Hollywood & Highland, Los Angeles, California, 2017, pp. 1-10. doi: 10.5594/M001785

[16] D. Stolitzka, W. Husak and S. Foessel, ”Introducing new profiles for broadcast applications in the JPEG 2000 international recommendation — standard”, SMPTE Annual Tech Conference & Expo, 2010, Hollywood, CA, USA, 2010, pp. 1-16. doi: 10.5594/M001390

[17] A. Friedl, J. Halak, M. Krsek, S. Ubik and P. Zejdl, ”Low-latency transmissions for remote collaboration in post-production,” The 2012 Annual Technical Conference & Exhibition, Hollywood, CA, USA, 2012, pp. 1-10. doi: 10.5594/M001465

[18] J. Navratil and S. Ubik, ”Surgery telepresence for universities and symposia,” 2012 IEEE 14th International Conference on e-Health Networking, Applications and Services (Healthcom), Beijing, 2012, pp. 218-222. doi: 10.1109/HealthCom.2012.6379411