DOI: 10.19729/j.cnki.1673-5188.2018.03.003
http://kns.cnki.net/kcms/detail/34.1294.TN.20180813.1727.002.html, published online August 13, 2018

Adaptive Mobile Video Delivery Based on Fountain Codes and DASH: A Survey

WU Kesong[1], CAO Xianbin[1], CHEN Zhifeng[2], and WU Dapeng[3]
(1. Beihang University, Beijing 100191, China;
2. Fuzhou University, Fuzhou 350116, China;
3. University of Florida, Gainesville 32611, USA)

Abstract

Recent years have witnessed an explosive growth in mobile video-based services and efficient and reliable video delivery draws more and more attention. As a type of rateless codes, fountain codes can automatically adapt to wireless channel conditions without any knowledge of channels. This paper provides an overview of several typical Forward Error Correction (FEC) codes, such as Reed-Solomon (RS) code, Tornado code, Luby-Transform (LT) code, and Raptor code. We focus on a novel delay-aware fountain coding (DAF) technique that maximizes the code word length under the constraint of a given delay. Based on DAF, this paper also presents Unequal Error Protection DAF (UEP-DAF) which improves the Peak Signal to Noise Ratio (PSNR) without additional coordination between the encoder and the decoder, as well as Model Predictive Control DAF (MPC-DAF) which reduces the computational complexity to an affordable level for real-time video communications. Moreover, we review video streaming technologies, then introduce Dynamic Adaptive Streaming over HTTP (DASH) and DASH over Multiple Content Distribution Servers (MCDS-DASH) in detail. Based on MCDS-DASH that adapts video bitrate at the block level to alleviate video fluctuation, we propose a novel approach to integrating fountain codes with MCDS-DASH, which is capable of achieving unprecedented high throughput.

Keywords

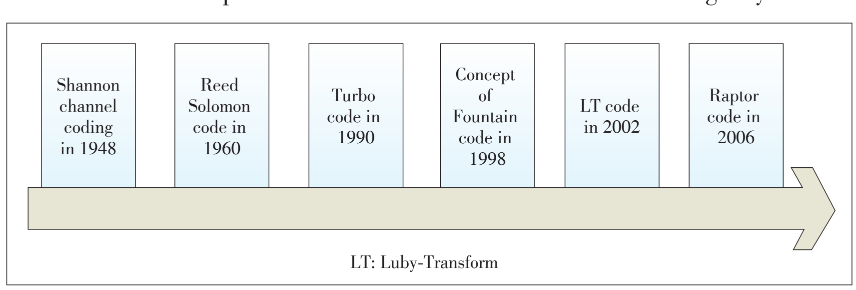
mobile video delivery; DAF; UEP; MPC; DASH

1 Introduction

During the past few years, mobile video-based services have witnessed an explosive growth, such as YouTube, FaceTime, and WeChat, with the popularity of smart mobile devices as well as the evolution of 3G, Long Term Evolution (LTE), Wi-Fi and other communication technologies. It is also expected that traffic from virtual reality (VR), augmented reality (AR), 3D games, surveillance video, vehicle networking and drone video will increase significantly in the near future. However, the problems including packet loss and insufficient bandwidth easily deteriorate in even higher demanding video applications due to the stochastic nature of wireless channels. As a result, how to deliver low latency and high quality video over wireless networks poses an unprecedented challenges for both academia and industry.

Erasure codes (**Fig. 1**), such as Reed-Solomon (RS) code proposed first by Reed and Solomon in 1960 and Tornado code that Luby et al. put forward in 1997, are comparatively powerful in error correction for data transmission over network. Nevertheless, they are not suitable for wireless transmission due to high co-decoding complexity. In addition, some fixed code rate must be chosen when the encoding begins, which could also lead to bandwidth waste if the erasure rate is overestimated, otherwise poor video quality.

Luby, Byers, et al. first presented the concept of digital fountain (DF) in 1998 for large-scale data distribution services and reliable broadcast/multicast services. In this concept, the number of encoded symbols that can be generated from the source data is potentially limitless and the code rate can automatically adapt to wireless environment without assuming any knowl-



▲**Figure 1. Development of typical erasure codes.**

edge of channels.

In 2002, Luby further proposed the first practical DF codes, Luby-Transform (LT) code [1]. Although the LT code exhibits excellent efficiency compared to traditional erasure codes, the shortcomings are obvious: decoding cannot be successful only until all the source packets are recovered; the decoding complexity of LT code is linear logarithmic order $O(k \log k)$, k is the number of source symbols.

In 2006, Shokrollahi put forward the most effective DF code, Raptor code, which was obtained by concatenating a high rate low-density parity-check (LDPC) code with an LT code of constant average degree distribution [2]. The Raptor code exhibits linear co-decoding time while still keeping low coding overhead. More importantly, it has been employed in the Digital Video Broadcasting—Handheld (DVB-H) for IP Datacast (IPDC) and the Third Generation Partnership Project (3GPP) standard for multimedia broadcast multicast services (MBMS) [3].

Internet video streaming services have also witnessed a tremendous growth with the evolution of Internet and the popularity of mobile intelligent devices. Traditional Real-Time Transport Protocol/Real Time Stream Protocol (RTP/RTSP) has increasingly become unable to answer the demand for the following reasons: high deployment cost, indispensable stream session management, exceedingly difficult traversing restrictive network address translators and firewalls, as well as not supported by prevalent content delivery networks (CDN).

Advanced streaming media technologies are urgently needed to solve the aforementioned problems. HTTP-based dynamic adaptive streaming media technology emerges as the times require, and has already developed rapidly.

The rest of this paper is organized as follows. Section 2 briefly describes block coding and sliding window coding, and introduces delay-aware fountain coding (DAF) in detail. Section 3 presents Unequal Error Protection DAF (UEP-DAF) and Model Predictive Control DAF (MPC-DAF) based on DAF. Section 4 reviews Dynamic Adaptive Streaming over HTTP (DASH) and proposes our novel scheme. Section 5 concludes the paper.

2 Delay-Aware Fountain Codes

Fountain codes are widely used in network transmission, such as reliable multicast, multi-source parallel downloading and distributed storage. However, if a video streaming file is delivered with conventional fountain codes, it cannot be displayed until the entire file is successfully decoded. Unfortunately, video streaming is of timeliness that means the time interval of video generation and display must not exceed a certain threshold. In addition, due to limited memory capabilities, receiver devices may also impose some restrictions on the receiving time of frames.

2.1 Block and Sliding Window Coding

The most direct solution to transmit video streaming with fountain codes is to partition video streams into blocks (**Fig. 2**), and then encode and transmit them sequentially. In [3], Ahmad et al. present a block coding design of fountain codes for video transmission. Considering the playback delay, the smaller the block size is, the shorter it is, however, from the fountain code point of view, the larger the block size, the lower the coding overhead and the higher the coding efficiency.

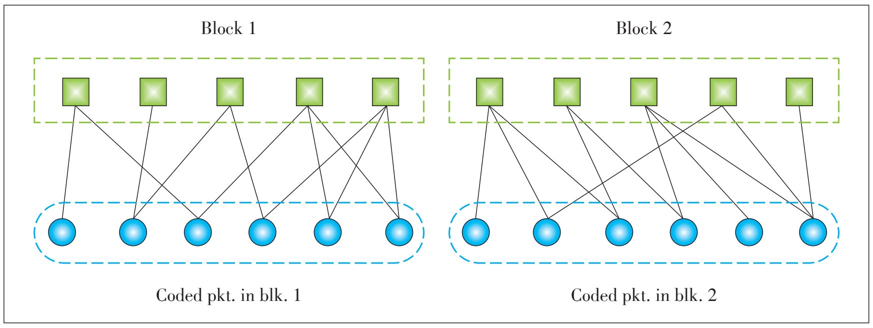
In [4], Bogino et al. proposed a sliding window approach to segment the source data, leaving some overlaps between successive steps, as shown in **Fig. 3**. Consequently, decoded packets in one window are valuable for decoding the coded packets of subsequent windows. The sliding window scheme virtually extends the size of source block with respect to the non-overlap block coding, thereby obtains a higher co-decoding efficiency with the overhead decrease. In addition, expanding window was presented in [5]. In the aforementioned three schemes, the block size is fixed in block coding, virtually extended in the sliding window, and kept growing in expanding window.

2.2 Delay-Aware Fountain Codes

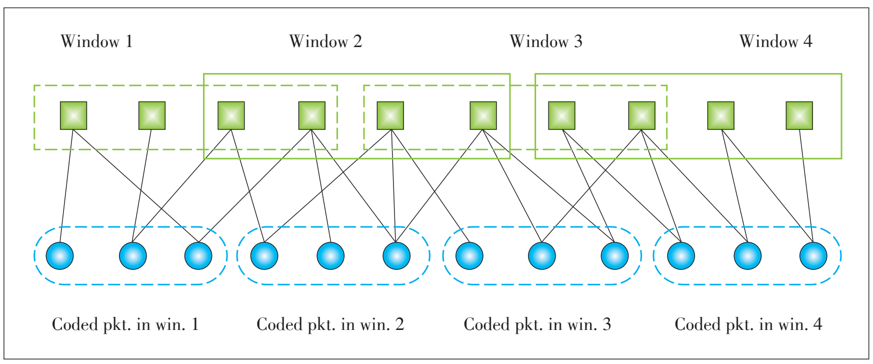
In 1977, Elience first proposed the joint source channel coding scheme to enhance the overall performance of the communication system. Although numerous studies on joint fountain codes and video coding have also been done, such as [3], [4], [5], and [6], they have a common shortcoming that is to divide the video streaming into blocks with the fixed number of packets without considering the video content characteristics. Accordingly, the same frame is probably divided into different blocks, which has a negative impact on the decoding of compressed video stream. Moreover, due to the fluctuation of video bit rate (**Fig. 4**), each packet-based block may contain different number of frames and the resulting video jitter increases as a result of the varying latency.

As is known to all that the concepts relevant to time in video sequence can be measured with number of frames. Consequently, different from the existing sliding window schemes, Sun, Wu et al. [7] innovatively proposed to establish the sliding window with the fixed number of frames to provide the much desired delay awareness in video streaming, namely DAF. In this way, the code word length can be maximized under the condition of limited delay, so as to achieve higher coding gain. In addition, a low-complexity online version DAF-L was also proposed, adopting only one parameter–slope factor to quantify the sampling distributions.

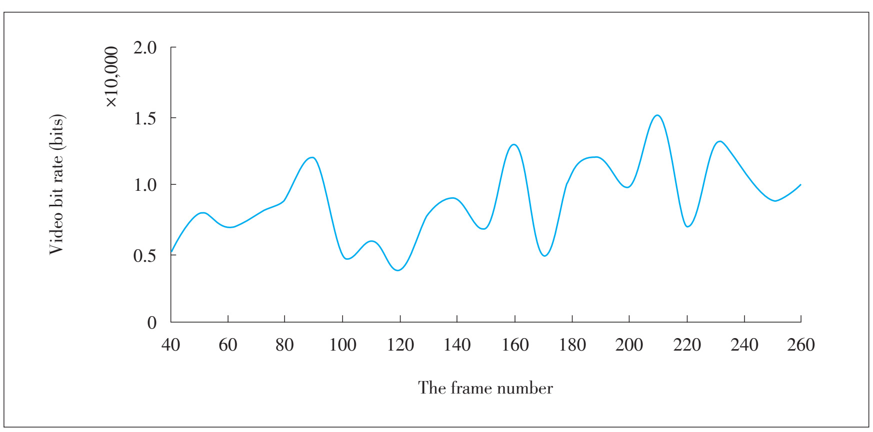
Design of degree distribution has a critical influence on the decoding performance of fountain codes, and optimization of degree distribution functions has been done in certain literatures, such as [1]. It is worth noting that in fountain codes the more uniform the sampling distribution of the source packets, the higher the coding efficiency [7]. As a result, we proposed the strategy of window-wise sampling which dynamically adjusts each window sampling mode according to the fluctuation



▲**Figure 2. Block coding.**



▲**Figure 3. Sliding window coding.**



▲**Figure 4. Common intermediate format (CIF) sequence foreman.**

of video bit rate.

$$ASP(t) = \sum_{\omega \in \text{all windows covert}} p_{\omega}^{pkt}(t), \quad (1)$$

where $p_{\omega}^{pkt}(t)$ denotes the average sampling probability of each packet in frame t within window ω. $ASP(t)$ denotes the total probability of every packet in frame t accumulated through all the sliding windows covering that frame. The objective function of the major optimization process in DAF is to minimize the variance of the accumulated sampling possibility $ASP(t)$ as in (1).

Fig. 5 shows the experiments, where the code rate is defined using the ratio of total number of native packets to total number of coded packets, and IDR denotes in-time decoding ratio, prove that this scheme has higher video quality than existing schemes with low delay and constant data rate.

3 UEP-DAF and MPC-DAF

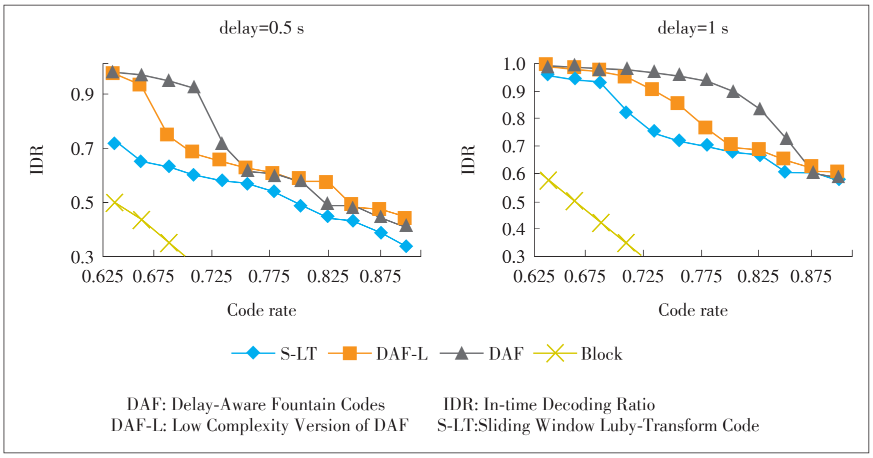
All the video data has been assumed of the same importance in DAF, in order to further enhance the practical applicability, we propose a method to integrate UEP into DAF.

3.1 UEP-DAF

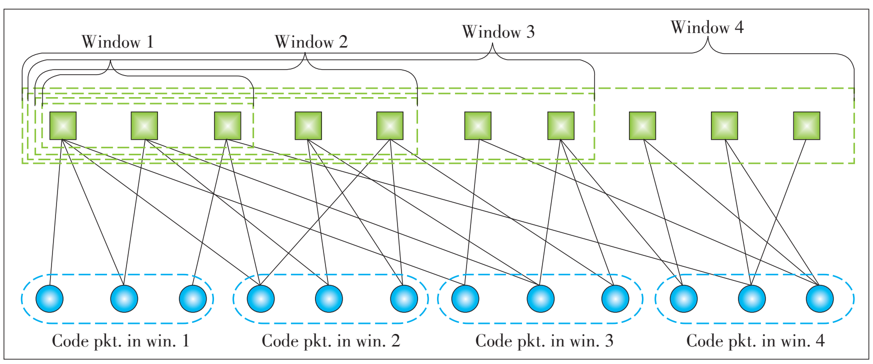
The aforementioned schemes are all based on the assumption that all the data has the same importance. However, the importance of different levels for video stream could be reflected in the following aspects, such as picture types, data types, position of the frame in a group of pictures (GOP), layers in scalable video coding (SVC), and picture content [8]. Fountain codes with equal error protection (EEP) are obviously inefficient, especially in the case of video delivery over wireless channel, the bandwidth of which is more often insufficient and decoding all packets with equal chance sometimes could induce suboptimal quality.

In recent years, fountain codes with unequal error protection (UEP) characteristics have attracted extensive attentions. In 2005, Rahnavard and Fekri suggested for the first time UEP-based fountain codes [9] which principle was to encode data by a method of adopting different degree distribution according to unequal importance. In [5] and [6], Sejdinovic and Vukobratovic proposed an approach of expanding window fountain codes, in which all windows had the same starting position and packets in each window must be a subset of the subsequent window (**Fig. 6**), so packets in the innermost window had the highest sampling probability. In order to achieve high efficient transmission of data in multiple source relay channels, Talari et al. presented a distributed unequal error protection fountain codes [10] to meet the requirements of different terminals. Ahmad, Hamzaoui et al. divided video source data block into several segments, and then duplicated them according to the protection factors to obtain a new block [11], as shown as in **Fig. 7**, where Mib denotes most important bits, and Lib denotes least important bits.

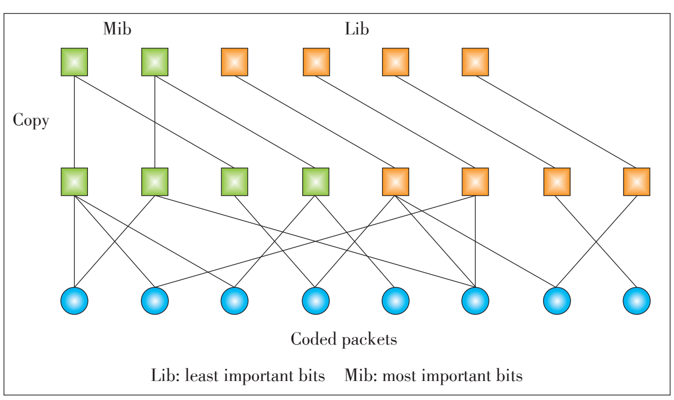
The aforementioned UEP-based schemes can improve remarkably the peak signal-to-noise ratio (PSNR) without the bit rate increasing. Nevertheless, they all have an unfavorable



▲**Figure 5. IDR vs. the code rate of four window schemes.**



▲**Figure 6. Expanding window coding.**



▲**Figure 7. UEP by duplicating Mib packets.**

problem which is the encoder needs to send importance description information to the decoder. Whether coordinating between encoder and decoder beforehand or explicit transmission in the packet headers, the resulting overhead will increase the possibility of packet loss.

In [8], Sun, Wu et al. proposed a novel scheme UEP-DAF to apply UEP to video communication applications based on delay-aware fountain codes. The proposed scheme does not need additional coordination between encoder and decoder; besides, the frame-level importance profile may be specified in advance. Simulation experiments show that compared to the result of DAF, UEP-DAF allocates higher sampling probability to the frames in the front of a GOP than that in the back (**Fig. 8**) where ASP denotes the accumulated sampling possibility. Consequently, the proposed system achieves higher decoding ratios and PSNR compared to EEP under the same network conditions.

3.2 MPC-DAF

DAF based on delay-aware sliding window and window-wise sampling, takes full advantage of channel-adaptive rateless feature, effective delay control and optimal sampling pattern, therefore it outperforms the other existing schemes. However, high computational complexity induced by the per-window optimization of sampling pattern and that the bit rate information of all frames needs to be obtained in advance, prevent its applications in live video streaming. The encoding computational complexity of DAF is $O\left(\left((T\cdot W)/\Delta t^{2}\right)^{3}\right)$, where T denotes video length, W denotes sliding window size, and Δt denotes step size. The detailed deduction can be found in [7]. Although the performance of low-complexity DAF-L proposed in [7] is higher than that of many other existing schemes, compared to DAF the gap is still significant, especially when packet loss is relatively serious.

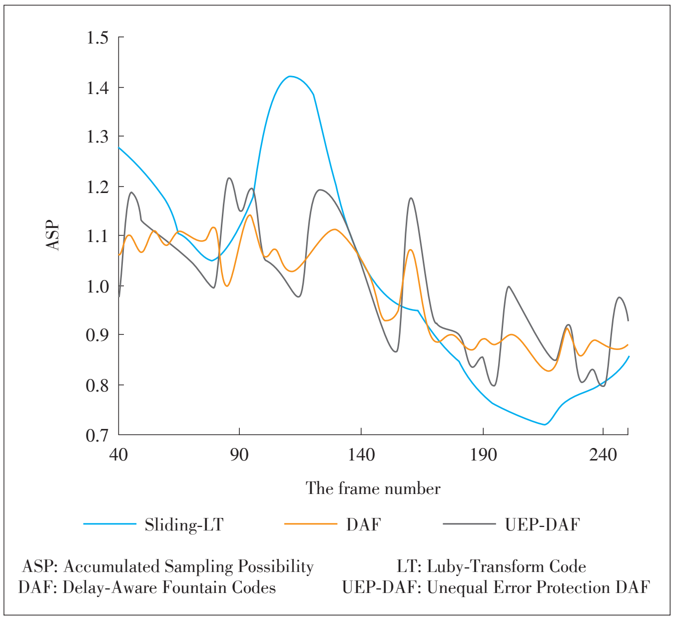
DAF-M based on the Model Predictive Control (MPC) was proposed in [12], which is an online optimization algorithm with a lower computational complexity with respect to DAF, but higher performance compared to DAF-L. More details of MPC-DAF code can be found in [12]. As a result, the computational complexity is lowered to an affordable level for real-time video communications. Moreover, the results of simulation experiments show that the decoding ratio of DAF-M is close to the global optimum in DAF codes [12].

4 MCDS-DASH Based on Fountain Codes

HTTP-based incremental download technology, overcoming the aforementioned problems, has become widespread, including Adobe's flash player, Microsoft Corp.'s Silverlight and windows media player, etc. However, it is inevitable that video playback interruption and video quality degradation occur as a result of the constantly changing network bandwidth, especially for the stochastic wireless networks.

4.1 MPEG-DASH

New generation streaming media technology, so-called the



▲**Figure 8. ASP of three sampling schemes.**

adaptive streaming media scheme, becomes more and more popular, because it can provide users with real-time, smooth and high quality streaming service [13]. Meanwhile, numerous HTTP adaptive streaming schemes were also proposed, including HTTP Live Streaming (HLS) from Apple Corp., Smooth Streaming (SS) from Microsoft Corp. and HTTP Dynamic Streaming (HDS) from Adobe Corp. However, the diversity of different streaming media service solutions brings about a large number of compatibility issues, which increases the difficulty of system maintenance.

DASH, also known as MPEG-DASH, is the first international standard of HTTP-based adaptive streaming media solution. Moving Picture Experts Group (MPEG) issued a call for proposal for an HTTP streaming standard in April 2009, then started the evaluation of the submitted fifteen full proposals since July 2009. The DASH international standard draft was completed in January 2011 and became the international standard in November 2011. In April 2012, the international standard of DASH with ISO/IEC 23009-1 was officially promulgated. Since then DASH has been widely adopted for providing uninterrupted video streaming service to users with dynamic network conditions and heterogeneous devices [14], [15].

Media presentation description (MPD) is a manifest file of encoding information defined by DASH server, including the descriptions of time-based periods, representations based on the bit rate, frame rates and resolutions, as well as video data segments. The DASH client is capable of choosing to download and display the most appropriate video segment according to the network conditions and the receiver buffer state.

Adaptive bit stream switching algorithm, as the core technology, has always been the most critical factor affecting the efficiency of DASH. An efficient rate adaptation algorithm should prevent frequent hopping between video bitrates, as well as frequent interruptions or non-optimal visual quality due to higher or lower than the available bandwidth.

In [16], Ojanper et al. proposed an adaptive network aided method for adaptive HTTP video stream based on cognitive network management architecture and distributed control. Müller et al. proposed an improved DASH implementation employing scalable video coding (SVC), which was suitable for mobile applications with large bandwidth variations [17]. Besides, a new model predictive control algorithm to optimize the throughput and buffer occupancy information was also proposed in [18].

4.2 DASH over Multiple Content Distribution Servers (MCDS-DASH)

Generally traditional DASH is based on single server, so no matter how excellent adaptation algorithm is adopted, all efforts seem helpless once the server is unreachable or even down. In order to improve the bandwidth and stability of the transmission, a parallel download technology based on MCDS was proposed in [19].

MCDS-DASH deploys DASH over multiple servers; thereby the same content can be available at multiple URLs concurrently and the DASH client can obtain video segments at the maximum bandwidth from the optimal server. Compared with the single server node, MCDS can obviously provide higher bandwidth, reliability and scalability.

However, rate adaption control becomes a more challenging problem in MCDS-DASH. Because if it adapts video bitrate still at the segment level as traditional methods do, frequent video bitrate switching would occur due to diverse bandwidths over multiple heterogeneous servers. In that case, viewing experience of users will decline dramatically [20]. Moreover, disorder downloading of video segments from different servers and the compulsory requirements of playback according to the correct order would cause additional delay.

In [19], Chao Zhou et al. presented to group multiple segments into a block and adapt video bitrate at the block level rather than at the segment level to alleviate the video fluctuation. Furthermore, downloading each segment according to the playback deadline was also proposed.

4.3 MCDS-DASH Based on Fountain Codes

Bitrate smoothness and bandwidth utilization are a couple of contradictions due to the inherent bandwidth variations in MCDS-DASH. Although the scheme proposed in [19] adapting video bitrate at the block level alleviates actually the video drastic fluctuation, this is at the expense of lower bandwidth utilization.

The order of video segments download completion may not be in accordance with the one you assume based on the current network status, especially in stochastic wireless networks, which would cause additional delay. In fact, the approaches mentioned in [19] are not completely parallel, because the servers having completed download will keep idle in one block unit, which actually is a waste of bandwidth resources.

Nevertheless, the block-based scheme prompts us an innovative inspiration to employ fountain codes in MCDS-DASH. In this scheme, we code video streaming files by a method of adopting fountain codes, then send the coded packets to the MCDS. In that sense DASH clients need not care about how many packets downloaded from each specific DASH server, and as long as the number of received packet reaches the threshold, the video streaming file can be decoded successfully. In that sense our novel scheme can achieve actually parallel download.

5 Conclusions

In this paper we presented an overview of several typical FEC codes, and focused on the state-of-the-art fountain codes DAF as well as its extended UEP-DAF and MPC-DAF. We also did research on respective advantages, disadvantages, suitable application scenarios etc. Furthermore, we reviewed the development of video streaming technologies, and paid a special attention to both DASH and MCDS-DASH in this work. Based on grouping multiple segments into a block and adapting video bitrate at the block level in MCDS-DASH, we put forward an innovative scheme which is able to integrate fountain codes with MCDS-DASH. Our scheme is capable of achieving unprecedented high throughput when multiple servers exert best efforts to transmit the same video file to clients.

References

[1] M. Luby, "LT codes," in *43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, Vancouver, Canada, 2002, pp. 271–280.

[2] A. Shokrollahi, "Raptor codes," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551–2567, 2006. doi: 10.1109/TIT.2006.874390.

[3] S. Ahmad, R. Hamzaoui, and M. M. Al-Akaidi, "Unequal error protection using fountain codes with applications to video communication," *IEEE Transactions on Multimedia*, vol. 13, no. 1, pp. 92– 101, 2011. doi: 10.1109/TMM.2010.2093511.

[4] M. C. Bogino, P. Cataldi, M. Grangetto, E. Magli, and G. Olmo, "Sliding-window digital fountain codes for streaming of multimedia contents," in *IEEE International Symposium on Circuits and Systems*, New Orleans, USA, May 2007, pp. 3467–3470. doi: 10.1109/ISCAS.2007.378373.

[5] D. Sejdinovic, D. Vukobratovic, A. Doufexi, V. Senk, and R. J. Piechocki, "Expanding window fountain codes for unequal error protection," *IEEE Transactions on Communications*, vol. 57, no. 9, pp. 2510–2516, 2009. doi: 10.1109/TCOMM.2009.09.070616.

[6] D. Vukobratovic, V. Stankovic, D. Sejdinovic, L. Stankovic, and Z. Xiong, "Scalable video multicast using expanding window fountain codes," *IEEE Transactions on Multimedia*, vol. 11, no. 6, pp. 1094– 1104, 2009. doi: 10.1109/TMM.2009.2026087.

[7] K. Sun, H. Zhang, and D. Wu. (2016). Delay-aware fountain codes for video streaming with optimal sampling strategy [Online]. Available: http://arxiv.org/abs/1605.03236

[8] K. Sun and D. Wu, "Unequal error protection for video streaming using delay-aware fountain codes," in *IEEE ICC 2017 Communications Software, Services, and Multimedia Applications Symposium*, Paris, France, May 2017. doi: 10.1109/ICC.2017.7996740.

[9] N. Rahnavard and F. Fekri, "Finite-length unequal error protection rateless codes: design and analysis," in *Global Telecommunications Conference*, St. Louis, USA, Dec. 2005. doi:10.1109/GLOCOM.2005.1577872.

[10] A. Talari and N. Rahnavard, "Distributed unequal error protection rateless codes over erasure channels: a two-source scenario," *IEEE Transactions on Communications*, vol. 60, no. 8, pp. 2084– 2090, 2012. doi: 10.1109/TCOMM.2012.051512.110109.

[11] N. Rahnavard, B. N. Vellambi, and F. Fekri, "Rateless codes with unequal error protection property," *IEEE Transactions on Information Theory*, vol. 53, no. 4, pp. 1521–1532, 2007. doi: 10.1109/TIT.2007.892814.

[12] K. Sun and D. Wu, "MPC-based delay-aware fountain codes for live video streaming," in *2016 IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1– 6. doi: 10.1109/ICC.2016.7510691.

[13] C. Müller and C. Timmerer, "A test-bed for the dynamic adaptive streaming over HTTP featuring session mobility," in *Second Annual ACM Conference on Multimedia Systems*, 2011, pp. 271–276. doi: 10.1145/1943552.1943588.

[14] A. Begen, T. Akgul, and M. Baugher, "Watching video over the web: part 1: streaming protocols," *IEEE Internet Computing*, vol. 15, no. 2, pp. 54–63, Mar. 2011. doi: 10.1109/MIC.2010.155.

[15] R. Kuschnig, I. Kofler, and H. Hellwagner, "Evaluation of HTTP-based request-response streams for Internet video streaming," in *Proc. ACM MMSys11*, Feb. 2011, pp. 245–256.

[16] T. Ojanperä, and H. Kokkoniemi-Tarkkanen, "Wireless bandwidth management for multiple video clients through network-assisted DASH," in *IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks*, Coimbra, Portugal, Jun. 2016. doi:10.1109/WoWMoM.2016.7523530.

[17] C. Müller, D. Renzi, S. Lederer, S. Battista, and C. Timmerer, "Using scalable video coding for dynamic adaptive streaming over HTTP in mobile environments," in *20th European Signal Processing Conference (EUSIPCO 2012)*, Bucharest, Romania, Aug., 2012, pp. 2208–2212.

[18] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over HTTP," in *2015 ACM Conference on Special Interest Group on Data Communication*, Larnaca, Cyprus, Jul. 2015, pp. 325–338. doi: 10.1109/ISCC.2015.7405532.

[19] C. Zhou, C.-W. Lin, X. G. Zhang, and Z. M. Guo, "A control-theoretic approach to rate adaption for DASH over multiple content distribution servers," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 4, Apr. 2014. doi: 10.1109/TCSVT.2013.2290580.

[20] E. C. R. Mok, X. Luo, and R. Chang, "QDASH: a QoE-aware DASH system," in *Proc. ACM Multimedia Syst.*, Chapel Hill, USA, Feb. 2012, pp. 11–22. doi: 10.1145/2155555.2155558.

Manuscript received: 2018-04-03

Biographies

WU Kesong (sdtwks@163.com) received his M.S. in communication & information system from Beihang University, China in 2014. He is now a Ph.D. candidate in Department of Electronic and Information Engineering, Beihang University. His research interests include multimedia communications, digital image and video processing, channel encoding, and machine learning.

CAO Xianbin (xbcao@buaa.edu.cn) is the Dean and a professor at the School of Electronic and Information Engineering, Beihang University, China. His current research interests include intelligent transportation systems, airspace transportation management, and intelligent computation. Currently, he serves as Associate Editor of *IEEE Transactions on Network Science and Engineering*, and *Associate Editor of Neurocomputing*.

CHEN Zhifeng (10799126@qq.com) received the Ph.D. degree in electrical and computer engineering from the University of Florida, USA in 2010. He is a professor with the College of Physics and Information Engineering, Fuzhou University, China. His research interests include video coding, video transmission, computer vision, machine learning, etc.

WU Dapeng (dpwu@ufl.edu) is a professor at the Department of Electrical and Computer Engineering, University of Florida, USA. His research interests are in the areas of networking, communications, signal processing, computer vision, machine learning, smart grid, and information and network security. He serves as Editor in Chief and Associate Editor of multiple IEEE journals. He was the founding Editor-in-Chief of *Journal of Advances in Multimedia*. He has served as a member of executive committees and/or technical program committees of over 80 conferences. He is an IEEE Fellow.