Server-based and Network-assisted Solutions for Adaptive Video Streaming

Koffka Khan
Department of Computing and Information Technology The University of the West Indies, Trinidad and Tobago, W.I
Email: koffka.khan@gmail.com
Wayne Goodridge
Department of Computing and Information Technology The University of the West Indies, Trinidad and Tobago, W.I
Email: wayne.goodridge@sta.uwi.edu.com

ABSTRACT

Server-based adaptive video streaming is gaining popularity in recent years. This is because clients (client-based) and in-network devices (network or proxy-based) are not powerful enough to run state of the art adaptation algorithms, for example, traffic shaping and machine learning. When decision making is placed at the server new and exciting possibilities are obtained for next best segment selection. This work highlights server-based solutions to adaptive video streaming. It provides a taxonomy of current state of the art solutions. It then illustrates various approaches used for server-based adaptive video streaming. Advantages and disadvantages are discussed. Network-assisted or in-network DASH solutions have certain advantages over traditional client-based approaches. It is proposed that the sharing of information would result in better network and client bandwidth estimations. This measure would ensure better next segment selections. In this paper a novel network-assisted DASH taxonomy is proposed. It consists of cache-based, optimization, rate-quality model, and co-operative elements. Recent approaches using the elements of the taxonomy are illustrated. These approaches show the advantages of using network-assisted entities in DASH-based systems.

Keywords - **Server-based; adaptive video streaming; traffic shaping; machine learning; taxonomy; network-assisted; in-network; bandwidth; segment; cache; optimization; rate-quality; co-operative; DASH.**

Date of Submission: Nov 01, 2017 Date of Acceptance: Nov 24, 2017

I. INTRODUCTION

This work provides a body of work that builds on an AVS taxonomy presented in Section II. It provides a detailed review on current state of the art solutions in server- and network-assisted AVS. After the taxonomy is given a discussion on server-based AVS techniques is portrayed in Section III. This is followed by classical approaches in general computing to server-based AVS techniques in Section IV. A taxonomy of server-based AVS techniques is then illustrated in Section V. Section VI gives state of the art approaches to server based AVS. Section VII starts the network-assisted solutions by giving a discussion of in-network AVS techniques. A taxonomy of in-network-based approaches is illustrated in Section VIII. Then state of the art in-network-based approaches is given in Section IX. Finally, the conclusion is given in Section X.

II. AVS TAXONOMY

This work categorizes AVS into five types: (1) Client-based, (2) Server-based, (3) Network-assisted, (4) SAND-based, and (5) Cloud-based, cf. Figure 1. However, only Server-based and Network-assisted solutions are discussed. Another paper talks about SAND- and Cloud-based solutions.

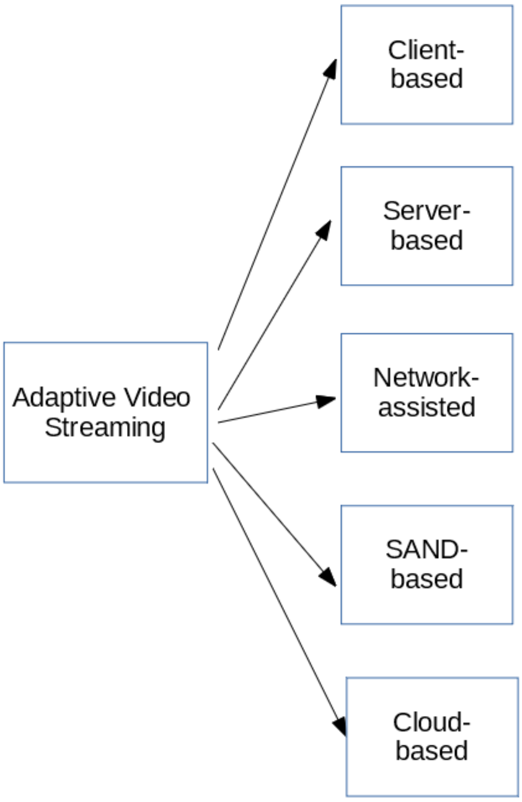

Fig. 1.AVS taxonomy

III. DISCUSSION OF SERVER-BASED AVS TECHNIQUES

Adaptive video streaming is a relatively new and rapidly adopted method for streaming video files over the Internet. The problem with older methods streaming on-demand video is that video files are too large so this keeps the user forever downloading all the video segments to watch a short movie clip. Adaptive video streaming chops up a massive video file into numerous pieces of different sizes and stores them on a video server. A user might want, for example, to stream an FLV file to his/her player in the H.264 codec using HTTP. Streaming is time-based and multi-process centric.

With streaming there is a core video format, a container used for storing the data as well as a time-based dimension of moving the data in a continuous stream. Thus, video streaming contains three layers of data management: (1) one the encoded bits like the H.264 for video (2) the container that holds the encoded bits together such as FLV or mp4 and (3) the transport that is used to move the stream from the video/media server to the player such as HTTP. A special file called an m3u8 tells the player the order in which to play the stream.

The focus of adaptive video streaming is at the time-based video segment selection and download periods. The segments for a particular timestamp contain different sizes, quality levels and resolutions. Segments for a download are of a certain length, for example, 2, 4, … , 20 seconds. Adaptation occurs where a desired segment is selected for the next segment download. This is followed by the download of the selected segment and a decision for the next segment.

The irregular ON-OFF traffic patterns caused by multiple adaptive video players sharing a common bottleneck link causes degraded QoE for users. Stalling and freezing caused by buffer underruns are very harmful for the user's viewing experience. Many users suffer from viewing poor video quality. This results from inadequate or unfair bandwidth allocation, that is, either over- or underestimation.

The many solutions proposed for adaptive video streaming offers tradeoffs among goals. This occurs as competing objectives conflict with each other as different parameters are adjusted. For example, increasing the quality level of the next selected segment may drain the buffer and cause buffer underruns at a later stage in streaming.

The streaming may be client- , network- or server-based. This classification depends upon where the decision for the next segment selection is taking place. In client-based solutions the decision is made at the client or video player. For network-based solutions the decision is made by intermediate network devices. With server-based solutions the decision is made at the server.

This work highlights server-based solutions to adaptive video streaming (Section IV). It provides a taxonomy of current state of the art solutions (Section V). It then illustrates various places where these solutions may be advantageous to use over network- or client-based approaches (Section VI).

IV. CLASSICAL APPROACHES IN GENERAL COMPUTING TO SERVER-BASED AVS TECHNIQUES

A. *Traffic Shaping*

Traffic shaping is a bandwidth management technique. It usually involves the manipulation and/or prioritization of network traffic (cf. Figure 2). These measures reduce the impact of heavy traffic flows from affecting other users or groups. Traffic shaping is used to optimize or guarantee performance, improve latency, or increase usable bandwidth for some kinds of packets by delaying other kinds [5]. If there is a high level of contention either upstream or downstream the link may become saturated. This causes the latency to increase substantially. In this scenario traffic shaping can be used as a preventative method, which keeps the latency in check. Traffic shaping provides a means to control the volume of traffic being sent into a network or across a link/path. It can be done within a specified period, in which case it is called bandwidth throttling. Traffic shaping is always achieved by delaying packets [19]. Traffic shaping is commonly applied at the network edges to control traffic entering the network. However, it can also be applied by the traffic source, for instance, an information server sending data to clients across a network.

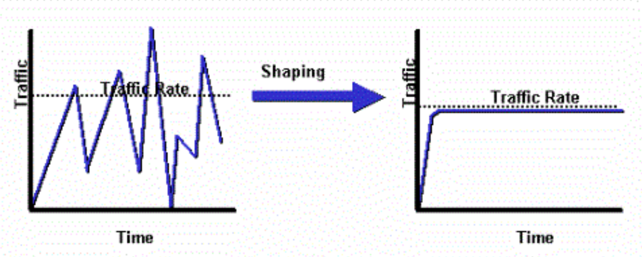

Fig. 2. Traffic Shaping.

B. *Video/Segment Pacing*

Pacing is a technique used for reducing traffic over a network [6]. When pacing is used traffic segment delivery to a device or client is slowed down to a point. "Just in time" delivery takes place. Pacing uses methods that avoids traffic bursts and massages the data flow to get a more even flow. However, when an object is delivered in its entirety, pacing provides no benefit.

C. *Rate Limiting*

Rate limiting is used to control the rate of traffic sent or received by a network interface controller, such as, a server in a computer network [11]. Web servers typically use a central in-memory key-value database, like Redis [24] or Aerospike [25], for session management. A rate limiting algorithm is used to check if the user session (or IP-address) must be limited based on the information in the session cache.

D. *Multi-path routing*

Multipath Routing is the spreading of traffic from a source node to a destination node over multiple paths through the network [42]. Multi-path routing offers a source application multiple paths on which to send data to a specified destination, cf. Figure 3. There are various types of multi-path varieties for many of the major routing protocols (e.g. MDSR [56], MAODV [64] or MAOMDV[32], SMR [33]) where some send data on only one or multiple of the available paths. Multi-path routing has advantages in congested networks and where destination data application requirements are large.

Multi-path routing offers three main concepts:

- A Multipath Calculation algorithm to compute multiple paths.
- A Multipath Forwarding algorithm to ensure that packets travel on their specified paths.
- An End-Host Protocol that effectively uses the determined multiple paths.

Path algorithms generate paths based on a desired characteristic of the path, e.g., maximized throughput or minimized delay. The algorithms can generate multi-option paths and/or multi-service paths. Path requirements depend on the end-user application, e.g., Telnet [50] vs. FTP [4].

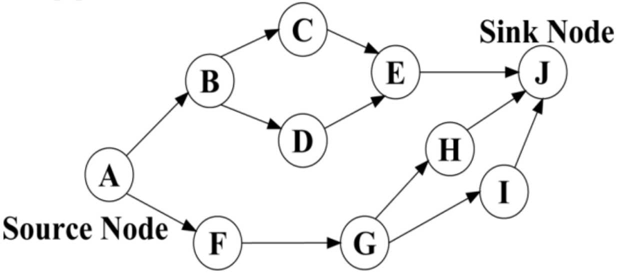

Fig. 3. Multiple paths from Source Node A to Destination Node J.

E. *TCP Variants*

The most common transport protocol used in internet for data transmission is Transmission Control Protocol [51]. There are many variants of TCP each variant being used for a specific purpose. TCP has many variants namely Tahoe [62], Reno [47], NewReno [48], SACK [38], Vegas [8], BIC [21], CUBIC [22], Peach [2] and many more. New transport protocols are always evolving with an objective to increase throughput and decrease the chance of getting into congestion. All these variants basically differ in the way they deal with congestion, control the data rate, and react with the lack of arrival of acknowledgments.

F. *Machine Learning*

Machine learning is a supervised or unsupervised learning technique which allows an actor to establish relationships between objects, actions and goals using a set of gathered data in an environment that is unfamiliar to it. It has gained a lot of popularity in the past thirty years and as computers have become more powerful has taken its place as one of the major areas of research in academia and industry. It uses a lot of statistical techniques, such as, linear regression, Markov Decision Process, and Bayesian analysis. There are many machine learning toolboxes available at present which provide algorithms for support vector machines (SVMs), boosted and bagged decision trees, k-nearest neighbor, k-means, k-medoids, hierarchical clustering, Neural Networks (cf. Figure 4), Gaussian mixture models, and hidden Markov models.

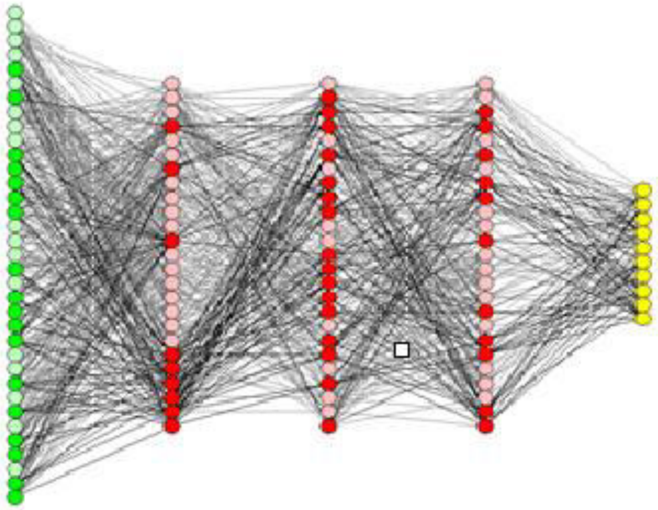

Fig. 4. Neural Network Maching Learning.

V. TAXONOMY OF SERVER-BASED AVS TECHNIQUES

Based on existing literature server-based adaptive streaming techniques can be broken up into two categories: (1) Traffic Management and (2) Protocol and Parameter-centric. Each category has three different approaches, cf. Figure 5. Traffic Shaping, Segment Pacing, and Rate Limiting are approaches which fall under the Traffic Management category. Multi-path, TCP Variants, and Machine Learning are approaches which fall under the Protocol and Parameter-centric category. Of the approaches undertaken so far by the research community some are relatively new and are now being realized. To this point the attempts at implementing Machine Learning approaches were first published in 2017. Some attempts treat the cloud as a server-based approach, but I place this approach in the network-based AVS technique.

VI. STATE OF THE ART APPROACHES TO SERVER BASED AVS

We first describe some traffic management approaches to server-based adaptive video streaming.

A. *Traffic Shaping approach*

A server-based traffic shaping method that greatly reduce bandwidth oscillations with minor loss in bandwidth utilization is proposed in [1]. Only when oscillations are detected is the shaper activated.

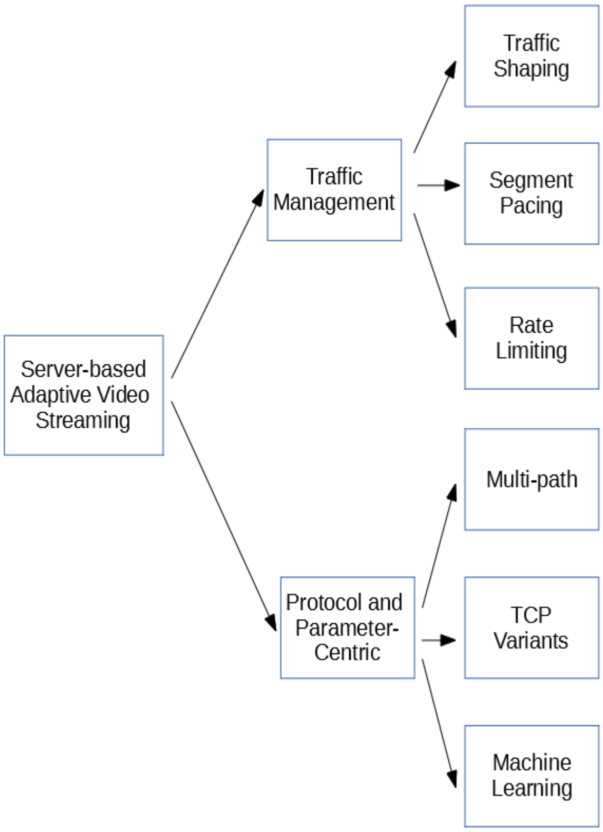

Fig. 5. Server-based AVS approaches.

It dynamically adjusts the shaping rate so that the player should preferably receive the highest available video profile. A shaping module that limits the throughput for each chunk to the encoding rate of the chunk, so that the download duration will roughly equal to the chunk duration, reducing/eliminating the OFF period [29] (as long as the available bandwidth is higher than the shaping rate). The proposed method is evaluated experimentally in terms of instability and utilization and compared with the unshaped case, under several scenarios.

Feedback control theory is used in a Quality Adaptation Controller (QAC) for live adaptive video streaming in [13]. It is compared with Akamai adaptive video streaming. The following results were obtained:

1) The available bandwidth is matched to the video quality by QAC throttling. This occurs with a transient of less than 30s. However, continuous video reproduction is ensured;
2) In the cases of a concurrent TCP greedy connection or a concurrent video streaming flow QAC fairly shares the available bandwidth;
3) The conservativeness of its heuristic algorithm causes Akamai underutilizes the available bandwidth; Consequently, the video reproduction is affected by interruptions, whenever sudden available bandwidth reductions occur.

B. *Segment Pacing*

Block sending is widely used by YouTube servers. Authors in [3] reveal and describe the basic properties of YouTube application flow control. They examine how the block sending algorithm impacts the flow control provided by TCP. It was shown that the block sending approach was responsible for over forty percent of packet loss events in YouTube flows. These tests used a residential DSL dataset. Also, the retransmission of over one percent of all YouTube data was sent after the application flow control began. It is noted that having less bursty YouTube block sending improves the performance and reduces the bandwidth usage of YouTube video streams.

Authors in [57] propose an adaptive video pacing method that enables both reducing unnecessary video traffic and maintaining user-perceived quality. Predicting the stochastic diffusion of TCP throughput and controlling the playout buffer size by considering the future throughput are two unique video-pacing functionalities. The proposed method can decrease the average playout buffer size in stable environments by up to 42.62% compared with the conventional methods and is shown in experimental results from simulating mobile networks. In unstable environments the frequency of the playback discontinuity of the proposed method is shown to be up to 34.1% less when compared to the conventional methods.

C. *Rate Limiting*

Authors in [20] evaluated Trickle on YouTube production data centers in Europe and India. They analyzed its impact on losses, bandwidth, RTT, and video buffer under-run events. Trickle reduces the average TCP loss rate by up to 43%. Trickle reduces the average RTT by up to 28%. These results were obtained with Trickle, while maintaining the streaming rate requested by the application. In addition, results show that Trickle has little impact on video buffer under-run events experienced by the users. The effectiveness of Trickle based on user bandwidth was investigated. It was demonstrated that Trickle has more benefits for high bandwidth users than low bandwidth users.

We now describe some protocol and parameter-centric approaches to server-based adaptive video streaming.

D. *Multi-path*

Authors in [18] presents a set of extensions to traditional TCP to support multipath operation. They provide the components necessary to establish and use multiple TCP flows across potentially disjoint paths. The TCP protocol offers the same type of service to applications as TCP (i.e., reliable byte stream).

Authors in [23] propose MP-DASH, a multipath framework for video streaming with the awareness of network interface preferences from users. The basic concept behind MP-DASH is to strategically schedule video segment delivery. This done with the aim of satisfying user preferences. A wide range of off-the-shelf video rate adaptation algorithms can incorporate MP-DASH with very minor modifications. An extensive field studies at 33 locations in three U.S. states were conducted. The studies suggest that MP-DASH is very effective: (a) it can reduce cellular usage by up to 99%, (b) it can reduce radio energy consumption by up to 85%, (c) The reductions in (a) and (b) can occur with negligible degradation of QoE. Comparisons are made with off-the-shelf MPTCP [58].

E. TCP Variants

Authors in [39] present TCP Hollywood. The TCP Hollywood protocol is wire-compatible [45] with TCP. It offers an unordered, partially reliable message-oriented transport service. This is well suited for multimedia applications. TCP Hollywood reduces latency and increases utility. In this way the analytical results obtained show that TCP Hollywood extends the feasibility of using TCP for real-time multimedia applications. Preliminary it was show that TCP Hollywood is applicable on the public Internet, with safe failure modes. In addition, measurements across all major UK fixed-line and cellular networks. This validates the possibility of local deployment. The TCP Hollywood implementation uses an intermediate logic layer between the application layer and the kernel. The TCP Hollywood stack is modified to support out-of-order delivery. This can be enabled or disabled using socket options. The concept of inconsistent retransmissions is explored: if the RTT estimator indicates that a packet will arrive too late to useful, or if the packet depends on the previously unsuccessful transmitted packet, then TCP Hollywood will exploit the retransmission slots to send new packets instead of retransmitting useless data. TCP Hollywood preserves the sequence numbers to determine if retransmission is required.

Authors in [16] provides an experimental update to TCP that allows a TCP sender to restart quickly following a rate-limited interval. This method is expected to assist applications that send rate-limited traffic using TCP. It also provides an appropriate response if congestion is experienced.

TCP-new CWV is a recent proposed update to TCP congestion control [44]. It has targeted congestion control for rate-limited applications. These methods are explored in the context of rate-adaptive applications, such as DASH. The new method enables a client to exploit the persistence of a DASH connection. It enables the DASH server to rapidly resume transmission of a series of video segments using a single TCP connection. 'Pacing' is another technique which smoothes DASH bursts. This occurs when there is no TCP ACK clock. It is shown to significantly reduce burst loss. The application performance is increased by the combination of these two methods. Authors investigate the effect of implementing these techniques on a DASH flow. Different congestion scenarios are considered. The measured outcomes were whether the methods can promote better capacity sharing while minimizing the latency experienced by other flows sharing a common network bottleneck. Experimental evidence shows that newCWV with Pacing provides a benefit as a platform for DASH transport.

F. Machine Learning

Authors in [36] propose Pensieve, a system that generates ABR algorithms using reinforcement learning (RL). Pensieve trains a neural network model that selects bitrates for future video segments. Pensieve does not rely on pre-programmed models or assumptions about the environment. Instead, Pensieve learns to make ABR decisions solely through observations. These observations are based on the resulting performance of past decisions. Thus, Pensieve automatically learns and instructs ABR algorithms that adapt to a wide range of environments and QoE metrics. Pensieve is compared to state-of-the-art ABR algorithms. Both trace-driven and real-world experiments spanning a wide variety of network conditions, QoE metrics, and video properties are used. In all considered scenarios, Pensieve outperforms the best state-of-the-art scheme. Average QoE improvements of 12%–25% were observed. It was shown that Pensieve generalizes well. It outperforms existing schemes. This happens even on networks for which it was not explicitly trained.

Authors in [12] presents Machine Learning-based Adaptive Streaming over HTTP (MLASH). They provide an elastic framework that exploits a wide range of useful network-related features. This is done to train a rate classification model. The MLASH machine learning-based framework can be incorporated with any existing adaptation algorithm. MLASH utilizes big data characteristics to improve prediction accuracy. The MLASH machine learning-based adaptation can achieve a better performance than traditional adaptation algorithms in terms of their target quality of experience (QoE) metrics. This was shown via trace-based simulations.

An adaptive ML-based framework to optimize the LTE scheduler operation is shown in [65]. Multiple objective scheduling strategies are targeted in the proposed technique. The weights of the different objectives are adjusted to optimize the resources allocation. This is done on a per transmission basis or on the user's demand pattern. The traditional scheduling methods have tradeoffs which this method overcomes. The technique can be used as a generic framework with any scheduling strategy. A Genetic Algorithm-based (GA-based) multi- objective scheduler is considered. It illustrates the efficiency of the proposed adaptive scheduling solution. The combination of clustering and classification algorithms along with the GA optimizes the GA scheduler functionality and makes use of the ML process to form a closed loop scheduling mechanism. Results are validated experimentally.

Authors in [46] present a methodology for the estimation of end users' QoE when watching YouTube videos which is based only on statistical properties of encrypted network traffic. The system is called YouQ. It includes tools for monitoring and analysis of application-layer KPIs (Key Performance Indicator) and corresponding traffic traces. This data is used for the development of machine learning models for QoE estimation based on traffic features. To test this approach, a collected dataset of 1060 different YouTube video traces using 39 different bandwidth scenarios are used. All video traces are annotated with application-layer KPIs. The traces are classified into one of three QoE classes, (“low”, “medium” or “high”). The dataset was used to test various machine learning algorithms. Results showed that up to 84% QoE classification accuracy could be achieved using only features extracted from encrypted traffic.

VII. DISCUSSION OF IN-NETWORK AVS TECHNIQUES

Adaptive bitrate streaming is a technique used in streaming multimedia over computer networks. While in the past most video streaming technologies utilized streaming protocols such as RTP, today's adaptive streaming technologies are almost exclusively based on HTTP and designed to work efficiently over large distributed HTTP networks such as the Internet. It works by detecting a user's bandwidth and/or buffer capacity in real time and adjusting the quality of a video stream accordingly [28], [17]. It requires the use of an encoder which can encode a single source video at multiple bit rates. The player client switches between streaming the different encodings depending on available resources. The result of very little buffering are fast start time and a good experience for both high-end and low-end connections.

The implementations in use today streams video over HTTP where the source content is encoded at multiple bit rates [60]. Then each of the different bit rates streams are segmented into small multi second parts. The streaming client player is made aware of the available streams of differing bit rates and segments of the streams by a manifest file. When starting the client requests a segment from the lowest bit rate stream is selected [30]. If the client finds the download speed is greater than the bit rate of the segments downloaded, then it will request the next higher bitrate segments later. If the client finds the download speed for a segment is lower than the bit rate for the segment and therefore the network throughput has deteriorated, then it will request a lower bitrate segment. The segment size can vary depending on the particular implementation, but they are typically between 2 and 10 seconds.

Post production houses content delivery networks and Studios use adaptive bitrate technology to provide consumers with higher quality video using less manpower and fewer resources. The creation of multiple video outputs particularly for adaptive bitrate streaming adds great value to consumers [10]. If the technology is working properly the end user or consumers content should playback without interruption and potentially go unnoticed. Media companies have been actively using adaptive bitrate technology for many years now and it has essentially become standard practice for high-end streaming providers permitting little buffering [61]. When streaming, high resolution feeds begin with low resolution and climbs.

Traditional server driven adaptive bitrate streaming proved friendly to consumers of streaming media with the best possible experience since the media server automatically adapts to any changes in each user's network and playback conditions. The media and entertainment industry also benefit from adaptive bitrate streaming. As the video space grows content delivery networks and video providers can provide customers with a superior viewing experience. Adaptive bitrate technology requires additional encoding but simplifies the overall workflow and creates better results. HTTP based adaptive bitrate streaming technologies yield additional benefits over traditional server driven adaptive bitrate streaming in the following ways (1) since the streaming technology is built on top of HTTP contrary to RTP based adaptive streaming the packet has no difficulties traversing firewall and net devices, (2) since HTTP streaming is purely client-driven all adaptation logic resides at the client [53]. This reduces the requirement of persistent connections between server and client application, (3) the server is not required to maintain session state information on each client increasing scalability, and (4) existing delivery infrastructure such as HTTP caches and servers can be seamlessly adopted.

However, in adaptive streaming environments, a purely client-driven approach [31], [37] has several significant disadvantages. In client-driven approaches the lack of coordination among clients leads to competing behavior, resulting in incorrect throughput estimations. This causes excessive quality oscillations and suboptimal decisions [35], [59], [40], which negatively impacts QoE. Also, user subscription constraints and guarantees on the delivered quality management policies, cannot be easily enforced [43]. These challenges should be tackled in order to facilitate adoption of HAS for the delivery of multimedia services in a managed environment.

A natural method to overcome the challenges posed by client-based approaches is network-assisted or in-network solutions. These solutions use intermediate network devices to do processing or decision making for the next segment bitrate selection for clients [9]. These devices can have many different functionalities including sensing network conditions and monitoring client state. The result is a global or partial (for example, based on the number of clients an in-network device is monitoring) view of the network which is proposed to be advantageous in adaptive streaming environments.

The remainder of this article is structured as follows. Section VIII gives a novel taxonomy of current in-network based adaptive video streaming solutions. The approaches that are present in each solution is described in Section IX.

VIII. TAXONOMY OF IN-NETWORK-BASED APPROACHES

The network-assisted AVS solutions are categorized into (1) Cache-based, (2) Optimization, (3) Rate-Quality Model, and (4) Co-operative. These categories are shown on Figure 6.

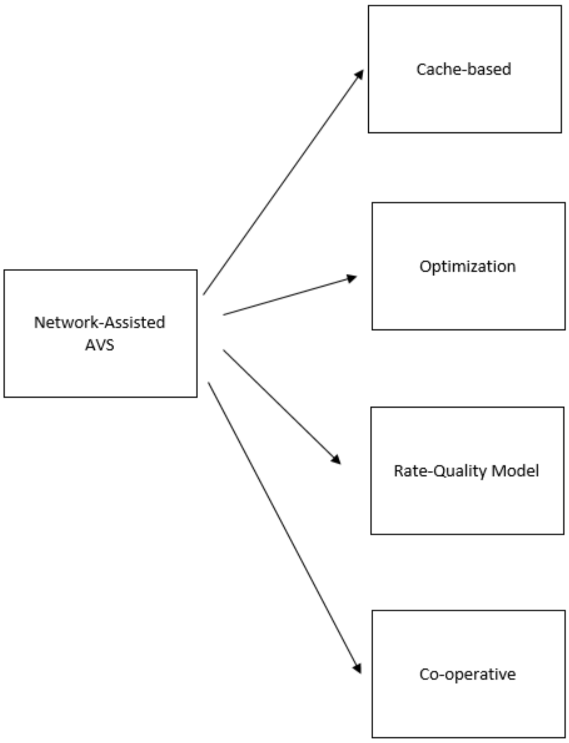

Fig. 6. Network-Assisted AVS taxonomy

IX. STATE OF THE ART IN-NETWORK-BASED APPROACHES

A. *Cache-based*

The authors in [14] argue that QoS may not accurately reflect the user-perceived video quality. This is especially the case in time-varying wireless networks. The concept of quality of experience (QoE) is adopted. A caching-based adaptation scheme to improve the overall QoE is proposed. It can comprehensively gauge the subjective user satisfaction of video quality. The proposed method outperforms existing adaptation schemes in terms of QoE in various network scenarios. This is shown by experiment. Properly leveraging channel bandwidth prediction and proxy-based content prefetching are the main aspects in this scheme.

The design and implementation of Mocha is presented in [54]. Mocha illustrates a quality adaptive multimedia proxy cache for layered encoded streams. Mocha can adjust quality of cached streams based (1) on their popularity and (2) on the available bandwidth between proxy and interested clients. Mocha can improve caching efficiency without compromising delivered quality. Mocha implements fine-grained replacement and fine-grained prefetching mechanisms to perform quality adaptive caching. Various design challenges such as managing partially cached streams are discussed.

B. *Optimization*

Authors in [7] proposes an in-network solution to streaming. It uses in-network quality optimization agents, which monitor the available throughput. This is done using sampling-based measurement techniques. Consequently, the quality of each client is optimized, based on a HAS Quality of Experience (QoE) metric. Centralized as well as distributed algorithms are used to solve a linear optimization problem. The proposed hybrid QoE-driven approach achieves two outcomes. Firstly, it allows the client to consider the in-network decisions during the rate adaptation process. Secondly, it keeps the ability to react to sudden bandwidth fluctuations in the local network. Quality selection heuristics are improved by at least 30%. The purely bitrate-driven optimization approach also outperforms an in-network approach using by up to 19%.

A Markov Decision Process (MDP) based network-assisted adaptation framework is proposed in [31]. The parameters investigated are: (1) cost of buffering, (2) significant playback variation, (3) bandwidth management and (4) income of playback. Its promising service provisioning and maximal profit for a mobile network is exhibited experimentally.

A wireless DASH (WiDASH) proxy to enhance the Quality-of-Experience (QoE) of wireless DASH is proposed in [52]. The WiDASH proxy is placed at the edge between the Internet and wireless core networks. The WiDASH proxy is in charge of video adaptation. This is different from server and client-based conventional DASH approach, where rate adaptation logics are either implemented in the DASH server or locally in the user equipment. The proxy makes it feasible to perform global optimization over multiple concurrent DASH flows. The WiDASH proxy uses three novel technologies, which improves DASH QoE: (1) architecture level innovation - both split TCP and parallel TCP feature sets are explored concurrently. A wireless user TCP connection from the DASH server is split into one wired TCP and multiple wireless TCPs. Without sacrificing inter user fairness this approach can increase video streams' average throughput. (2) Video quality aware dynamic prioritization is used by the WiDASH proxy. WiDASH gives low bit rate video streams high priority to guarantee minimum QoE for users with poor wireless channel quality. For inter flow fairness between video streams and background best effort flows to be maintained a video stream's priority is gradually dropped to normal state with the growth of its bit rate. (3) a multiple-input multiple-output adaptive optimal controller is designed, based on adaptive control theory. This rate controller minimizes a quadratic cost function. This function is defined as the weighted sum of multiple concurrent DASH videos' (a) distortion, (b) bit rate variation, and (c) playback jitter. The solution of the optimization problem implicitly coordinates multiple DASH streams to improve overall streaming QoE. Experiments used wireless HTTP video streaming simulations. Results verify that the WiDASH proxy can improve QoE by providing smoother video streams with higher average visual quality.

C. *Rate-Quality Models*

A proactive QoE-based approach for rewriting the client HTTP requests at a proxy in the mobile network proposed in [15]. The approach is applicability for over-the-top (OTT) streaming. This is because it requires no adaptation of the media content. The proposed scheme is compared to reactive QoE-optimized and standard-DASH HTTP streaming. The authors show that: 1) standard OTT DASH leads to unsatisfactory performance. This is because the content agnostic resource allocation by the LTE scheduler is far from optimal. Also, a clear QoE improvement is achieved when considering the content characteristics. 2) proactively rewriting the client requests gives control of the video content adaptation to the network operator. The network operator has better information than the client on the load and radio conditions in the cell. Thus, additional gains in user perceived video quality is obtained. 3) standard unmodified DASH client remains unaware of the proposed rewriting of the HTTP requests. Thus, it can decode and play the redirected media segments.

A proxy-based solution for adapting the scalable video streams at the edge of a wireless network is proposed in [27]. It can respond quickly to highly dynamic wireless links. The design adopts the recently standardized scalable video coding (SVC) technique for lightweight rate adaptation at the edge. Previously developed rate and quality models of scalable video with both temporal and amplitude scalability is leveraged. The rate-quality model that relates the maximum quality under a given rate by choosing the optimal frame rate and quantization stepsize value are derived. Different video streams are used to maximize a weighted sum of video qualities associated with different streams. The proxy iteratively allocates rates based on the periodically observed link throughputs and the sending buffer status. The temporal and amplitude layers included in each video are determined to optimize the quality. The rate assignment is satisfied at the same time. The approach consistently outperforms TFRC in terms of (1) agility to track link qualities and (2) overall subjective quality of all streams. The proposed scheme supports differential services for different streams. It also competes fairly with TCP flows.

D. Co-operative

A mechanism based on traffic chapping that allow bandwidth arbitration to be implemented in the home gateway is given in [26]. This is done by determining desirable target bit-rates to be reached by each stream. In addition, clients are constrained to stay within their limits. This enables the maximum number of users to reach the delivery goal of obtaining the optimal quality of experience. Results are shown through a set of objective measurement criteria which is validated through experimentation.

QoE-aware DASH system (or QDASH) is proposed in [41] to improve the user-perceived quality of video watching. Available bandwidth measurement is integrated into the video data probes. The system uses a measurement proxy architecture. The available bandwidth measurement method facilitates the selection of video quality levels. QoE is assessed using quality transitions by carrying out subjective experiments. The results show that users prefer a gradual quality change between the best and worst quality levels. Abrupt switching is detrimental to users. Thus, the proposed QoE-aware quality adaptation algorithm for DASH is based on these findings. Network measurement and the QoE-aware quality adaptation is integrated to produce a comprehensive DASH system.

A novel rate adaptation algorithm called FINEAS (Fair In-Network Enhanced Adaptive Streaming) is proposed in [49]. FINEAS is capable of increasing clients' Quality of Experience (QoE) and achieving fairness in a multiclient setting. FINEAS uses an in-network system of coordination proxies. These oversee facilitating fair resource sharing among clients. There are three main elements of this approach: (1) Fairness is achieved without explicit communication among clients. Therefore, no significant overhead is introduced into the network, (2) the clients do not need to be aware of the presence of the system of coordination proxies, and (3) the HAS principle is maintained. This is because the in-network components only provide the clients with new information and suggestions, while the rate adaptation decision remains the sole responsibility of the clients themselves. FINEAS is evaluated through simulations. Conditions, such as, variable bandwidth conditions and in several multiclient scenarios are tested. FINEAS can improve fairness up to 80% compared to state-of-the-art HAS heuristics. This occurs in a scenario with three networks. Each network contains 30 clients streaming video at the same time.

Dynamic Adaptive Streaming over Content centric networking (DASC) is presented in [34]. DASC implements MPEG Dynamic Adaptive Streaming over HTTP (DASH). It utilizes a Content Centric Networking (CCN) naming scheme to identify content segments in a CCN network. Video segments formatted according to MPEG-DASH are available in different quality levels. However, instead of HTTP, CCN is used for referencing and delivery. The DASC client issues interests for segments achieving the best throughput based on the conditions of the network. Subsequent requests for the same content can be served quicker due to segment caching within the network. The quality of the video a user receives progressively improves as a result. This effectively overcomes bottlenecks in the network. Two sets of experiments are used to evaluate the performance of DASC. They show that throughput improves. As a disadvantage, the generated overhead is relatively large.

X. Conclusion

Server-based adaptive video streaming is gaining popularity in recent years. This is because clients (client-based) and in-network devices (network or proxy-based) are not powerful enough to run state of the art adaptation algorithms, for example, traffic shaping and machine learning. When decision making is placed at the server new and exciting possibilities are obtained for next best segment selection. Current approaches can be categorized into (1) Traffic Management or (2) Protocol and Parameter-centric. Traffic Management approaches include (a) Traffic Shaping, (b) Segment Pacing, and (c) Rate Limiting. Protocol and Parameter-centric approaches include (a) Multi-path, (b) TCP Variants, and (c) Machine Learning.

Network-assisted or in-network DASH solutions have certain advantages over traditional client-based approaches. It is proposed that the sharing of information with network devices enhances the streaming process. A novel network-assisted DASH taxonomy is proposed and presented. It details approaches of cache-based, optimization, rate-quality model, and co-operative AVS elements of the taxonomy. These approaches show the advantages of using network-assisted entities in DASH-based systems.

REFERENCES

[1] Akhshabi, Saamer, Lakshmi Anantakrishnan, Constantine Dovrolis, and Ali C. Begen. "Server-based traffic shaping for stabilizing oscillating adaptive streaming players." In Proceeding of the 23rd ACM workshop on network and operating systems support for digital audio and video, pp. 19-24. ACM, 2013.

[2] Akyildiz, Ian F., Giacomo Morabito, and Sergio Palazzo. "TCP-Peach: a new congestion control scheme for satellite IP networks." IEEE/ACM Transactions on Networking (ToN) 9, no. 3 (2001): 307-321.

[3] Alcock, Shane, and Richard Nelson. "Application flow control in YouTube video streams." ACM SIGCOMM Computer Communication Review 41, no. 2 (2011): 24-30.

[4] Allcock, William, Ian Foster, Steven Tuecke, Ann Chervenak, and Carl Kesselman. "Protocols and services for distributed data-intensive science." In AIP Conference Proceedings, vol. 583, no. 1, pp. 161-163. AIP, 2001.

[5] Awduche, Daniel, Angela Chiu, Anwar Elwalid, Indra Widjaja, and XiPeng Xiao. Overview and principles of Internet traffic engineering. No. RFC 3272. 2002.

[6] Blackard, Joe Wayne, Richard Adams Gillaspy, William John Henthorn, Lynn Erich Petersen, Lance W. Russell, and Gary Roy Shippy. "Data processing system and method for pacing information transfers in a communications network." U.S. Patent 5,918,020, issued June 29, 1999.

[7] Bouten, Niels, Ricardo de O. Schmidt, Jeroen Famaey, Steven Latré, Aiko Pras, and Filip De Turck. "QoE-driven in-network optimization for Adaptive Video Streaming based on packet sampling measurements." Computer networks 81 (2015): 96-115.

[8] Brakmo, Lawrence S., and Larry L. Peterson. "TCP Vegas: End to end congestion avoidance on a global Internet." IEEE Journal on selected Areas in communications 13, no. 8 (1995): 1465-1480.

[9] Bronzino, Francesco, Dragoslav Stojadinovic, Cedric Westphal, and Dipankar Raychaudhuri. "Exploiting network awareness to enhance DASH over wireless." In Consumer Communications & Networking Conference (CCNC), 2016 13th IEEE Annual, pp. 1092-1100. IEEE, 2016.

[10] Bruneau-Queyreix, Joachim, Mathias Lacaud, Daniel Negru, Jordi Mongay Batalla, and Eugen Borcoci. "MS-Stream: A multiple-source adaptive streaming solution enhancing consumer's perceived quality." In IEEE Consumer Communications and Networking Conference-CCNC. 2017.

[11] Callon, Ross W., and Frank Kastenholz. "Rate limiting data traffic in a network." U.S. Patent 7,389,537, issued June 17, 2008.

[12] Chien, Yu-Lin, Kate Ching-Ju Lin, and Ming-Syan Chen. "Machine learning based rate adaptation with elastic feature selection for HTTP-based streaming." In Multimedia and Expo (ICME), 2015 IEEE International Conference on, pp. 1-6. IEEE, 2015.

[13] De Cicco, Luca, Saverio Mascolo, and Vittorio Palmisano. "Feedback control for adaptive live video streaming." In Proceedings of the second annual ACM conference on Multimedia systems, pp. 145-156. ACM, 2011.

[14] Dong, Kai, Jun He, and Wei Song. "Qoe-aware adaptive bitrate video streaming over mobile networks with caching proxy." In Computing, Networking and Communications (ICNC), 2015 International Conference on, pp. 737-741. IEEE, 2015.

[15] El Essaili, Ali, Damien Schroeder, Dirk Staehle, Mohammed Shehada, Wolfgang Kellerer, and Eckehard Steinbach. "Quality-of-experience driven adaptive HTTP media delivery." In Communications (ICC), 2013 IEEE International Conference on, pp. 2480-2485. IEEE, 2013.

[16] Fairhurst, Gorry, Raffaello Secchi, and Arjuna Sathiaseelan. "Updating TCP to support rate-limited traffic." (2015).

[17] Fan, Qilin, Hao Yin, Geyong Min, Po Yang, Yan Luo, Yongqiang Lyu, Haojun Huang, and Libo Jiao. "Video delivery networks: Challenges, solutions and future directions." Computers & Electrical Engineering (2017).

[18] Ford, Alan, Costin Raiciu, Mark Handley, and Olivier Bonaventure. TCP extensions for multipath operation with multiple addresses. No. RFC 6824. 2013.

[19] Georgiadis, Leonidas, Roch Guérin, Vinod Peris, and Kumar N. Sivarajan. "Efficient network QoS provisioning based on per node traffic shaping." IEEE/ACM Transactions on Networking (TON) 4, no. 4 (1996): 482-501.

[20] Ghobadi, Monia, Yuchung Cheng, Ankur Jain, and Matt Mathis. "Trickle: Rate Limiting YouTube Video Streaming." In Usenix Annual Technical Conference, pp. 191-196. 2012.

[21] Gupta, Diwaker, Kenneth Yocum, Marvin McNett, Alex C. Snoeren, Amin Vahdat, and Geoffrey M. Voelker. "To infinity and beyond: time warped network emulation." In Proceedings of the twentieth ACM symposium on Operating systems principles, pp. 1-2. ACM, 2005.

[22] Ha, Sangtae, Injong Rhee, and Lisong Xu. "CUBIC: a new TCP-friendly high-speed TCP variant." ACM SIGOPS Operating Systems Review 42, no. 5 (2008): 64-74.

[23] Han, Bo, Feng Qian, Lusheng Ji, Vijay Gopalakrishnan, and N. J. Bedminster. "MP-DASH: Adaptive Video Streaming Over Preference-Aware Multipath." In CoNEXT, pp. 129-143. 2016.

[24] Han, Jing, E. Haihong, Guan Le, and Jian Du. "Survey on NoSQL database." In Pervasive computing and applications (ICPCA), 2011 6th international conference on, pp. 363-366. IEEE, 2011.

[25] Harrison, Guy. "Database survey." In Next Generation Databases, pp. 217-228. Apress, 2015.

[26] Houdaille, Rémi, and Stéphane Gouache. "Shaping HTTP adaptive streams for a better user experience." In Proceedings of the 3rd Multimedia Systems Conference, pp. 1-9. ACM, 2012.

[27] Hu, Hao, Xiaoqing Zhu, Yao Wang, Rong Pan, Jiang Zhu, and Flavio Bonomi. "Proxy-based multi-stream scalable video adaptation over wireless networks using subjective quality and rate models." IEEE Transactions on Multimedia 15, no. 7 (2013): 1638-1652.

[28] Jana, Rittwik, Jeffrey Erman, Vijay Gopalakrishnan, Emir Halepovic, Rakesh Sinha, and Xuan Kelvin Zou. "Method and system for managing service quality according to network status predictions." U.S. Patent 9,756,112, issued September 5, 2017.

[29] Jiang, Wenyu, and Henning Schulzrinne. "Analysis of on-off patterns in VoIP and their effect on voice traffic aggregation." In Computer Communications and Networks, 2000. Proceedings. Ninth International Conference on, pp. 82-87. IEEE, 2000.

[30] Kaufman, Matthew. "Providing integration of multi-bit-rate media streams." U.S. Patent 9,680,892, issued June 13, 2017.

[31] Kesavan, Selvaraj, and J. Jayakumar. "Effective client-driven three-level rate adaptation (TLRA) approach for adaptive HTTP streaming." Multimedia Tools and Applications (2017): 1-34.

[32] Khan, Koffka, and Wayne Goodridge. "Fault Tolerant Multi-Criteria Multi-Path Routing in Wireless Sensor Networks." International Journal of Intelligent Systems and Applications 7, no. 6 (2015): 55.

[33] Lee, S-J., and Mario Gerla. "Split multipath routing with maximally disjoint paths in ad hoc networks." In Communications, 2001. ICC 2001. IEEE International Conference on, vol. 10, pp. 3201-3205. IEEE, 2001.

[34] Liu, Yaning, Joost Geurts, Jean-Charles Point, Stefan Lederer, Benjamin Rainer, Christopher Muller, Christian Timmerer, and Hermann Hellwagner. "Dynamic adaptive streaming over CCN: a caching and overhead analysis." In Communications (ICC), 2013 IEEE International Conference on, pp. 3629-3633. IEEE, 2013.

[35] Mangla, Tarun, Nawanol Theera-Ampornpunt, Mostafa Ammar, Ellen Zegura, and Saurabh Bagchi. "Video through a crystal ball: effect of bandwidth prediction quality on adaptive streaming in mobile environments." In Proceedings of the 8th International Workshop on Mobile Video, p. 1. ACM, 2016.

[36] Mao, Hongzi, Ravi Netravali, and Mohammad Alizadeh. "Neural Adaptive Video Streaming with Pensieve." In Proceedings of the Conference of the ACM Special Interest Group on Data Communication, pp. 197-210. ACM, 2017.

[37] Martín, Virginia, Julián Cabrera, and Narciso García. "Design, optimization and evaluation of a Q-Learning HTTP Adaptive Streaming Client." IEEE Transactions on Consumer Electronics 62, no. 4 (2016): 380-388.

[38] Mathis, Matt, Jamshid Mahdavi, Sally Floyd, and Allyn Romanow. TCP selective acknowledgment options. No. RFC 2018. 1996.

[39] McQuistin, Stephen, Colin Perkins, and Marwan Fayed. "TCP goes to Hollywood." In Proceedings of the 26th International Workshop on Network and Operating Systems Support for Digital Audio and Video, p. 5. ACM, 2016.

[40] Mok, Ricky KP, Weichao Li, and Rocky KC Chang. "IRate: Initial video bitrate selection system for HTTP streaming." IEEE Journal on Selected Areas in Communications 34, no. 6 (2016): 1914-1928.

[41] Mok, Ricky KP, Xiapu Luo, Edmond WW Chan, and Rocky KC Chang. "QDASH: a QoE-aware DASH system." In Proceedings of the 3rd Multimedia Systems Conference, pp. 11-22. ACM, 2012.

[42] Mueller, Stephen, Rose Tsang, and Dipak Ghosal. "Multipath routing in mobile ad hoc networks: Issues and challenges." Performance tools and applications to networked systems (2004): 209-234.

[43] Nam, Hyunwoo. Measuring and Improving the Quality of Experience of Adaptive Rate Video. Columbia University, 2016.

[44] Nazir, Sajid, Ziaul Hossain, Raffaello Secchi, Matthew Broadbent, Andreas Petlund, and Gorry Fairhurst. "Performance evaluation of congestion window validation for DASH transport." In Proceedings of Network and Operating System Support on Digital Audio and Video Workshop, p. 67. ACM, 2014.

[45] Nowlan, Michael F., Nabin Tiwari, Janardhan Iyengar, Syed Obaid Aminy, and Bryan Fordy. "Fitting square pegs through round pipes: Unordered delivery wire-compatible with TCP and TLS." In Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation, pp. 28-28. USENIX Association, 2012.

[46] Orsolic, Irena, Dario Pevec, Mirko Suznjevic, and Lea Skorin-Kapov. "A machine learning approach to classifying YouTube QoE based on encrypted network traffic." Multimedia Tools and Applications (2017): 1-35.

[47] Padhye, Jitendra, Victor Firoiu, Donald F. Towsley, and James F. Kurose. "Modeling TCP Reno performance: a simple model and its empirical

validation." IEEE/ACM Transactions on Networking (ToN) 8, no. 2 (2000): 133-145.

[48] Parvez, Nadim, Anirban Mahanti, and Carey Williamson. "An analytic throughput model for TCP NewReno." IEEE/ACM Transactions on Networking (ToN) 18, no. 2 (2010): 448-461.

[49] Petrangeli, Stefano, Jeroen Famaey, Maxim Claeys, Steven Latré, and Filip De Turck. "QoE-driven rate adaptation heuristic for fair adaptive video streaming." ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 12, no. 2 (2016): 28.

[50] Postel, Jon. "Telnet protocol specification." (1980).

[51] Postel, Jon. "Transmission control protocol." (1981).

[52] Pu, Wei, Zixuan Zou, and Chang Wen Chen. "Video adaptation proxy for wireless dynamic adaptive streaming over HTTP." In Packet Video Workshop (PV), 2012 19th International, pp. 65-70. IEEE, 2012.

[53] Rahman, Waqas Ur, Dooyeol Yun, and Kwangsue Chung. "A client side buffer management algorithm to improve QoE." IEEE Transactions on Consumer Electronics 62, no. 4 (2016): 371-379.

[54] Rejaie, Reza, and Jussi Kangasharju. "Mocha: A quality adaptive multimedia proxy cache for internet streaming." In Proceedings of the 11th international workshop on Network and operating systems support for digital audio and video, pp. 3-10. ACM, 2001.

[55] Sánchez, Antonio. Digital Services in the 21st Century: A Strategic and Business Perspective. John Wiley & Sons, 2017.

[56] Sarkar, Nurul I., and Wilford G. Lol. "A study of manet routing protocols: Joint node density, packet length and mobility." In Computers and Communications (ISCC), 2010 IEEE Symposium on, pp. 515-520. IEEE, 2010.

[57] Satoda, Kozo, Hiroshi Yoshida, Hironori Ito, and Kazunori Ozawa. "Adaptive video pacing method based on the prediction of stochastic TCP throughput." In Global Communications Conference (GLOBECOM), 2012 IEEE, pp. 1944-1950. IEEE, 2012.

[58] Scharf, Michael, and Alan Ford. Multipath TCP (MPTCP) application interface considerations. No. RFC 6897. 2013.

[59] Susanto, Hengky, ByungGuk Kim, and Benyuan Liu. "User Experience Awareness Network Optimization for Video Streaming Based Applications." Advances in Computer Communications and Networks: From Green, Mobile, Pervasive Networking to Big Data Computing (2016): 395.

[60] Wagenaar, Arjen, Dirk Griffioen, and Rufael Mekuria. "Unified Remix: a Server Side Solution for Adaptive Bit-Rate Streaming with Inserted and Edited Media Content." In Proceedings of the 8th ACM on Multimedia Systems Conference, pp. 221-224. ACM, 2017.

[61] Wang, Chen-Chi, Zih-Ning Lin, Shun-Ren Yang, and Phone Lin. "Mobile edge computing-enabled channel-aware video streaming for 4G LTE." In Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th International, pp. 564-569. IEEE, 2017.

[62] Wang, Zheng, and Jon Crowcroft. "Eliminating periodic packet losses in the 4.3-Tahoe BSD TCP congestion control algorithm." ACM SIGCOMM Computer Communication Review 22, no. 2 (1992): 9-16.

[63] Yan, Zhisheng, Cedric Westphal, Xin Wang, and Chang Wen Chen. "Service provisioning and profit maximization in network-assisted adaptive HTTP streaming." In Image Processing (ICIP), 2015 IEEE International Conference on, pp. 2786-2790. IEEE, 2015.

[64] Zhu, Yufang, and Thomas Kunz. "MAODV implementation for NS-2.26." Systems and Computing Engineering, Carleton University, Technical Report SCE-04-01 (2004).

[65] Zwierzykowski, Piotr. "Design, Dimensioning, and Optimization of 4G/5G Wireless Communication Networks."

Biographies and Photographs

Koffka Khan received the M.Sc., and M.Phil. degrees from the University of the West Indies. He is currently a PhD student and has up-to-date, published numerous papers in journals & proceedings of international repute. His research areas are computational intelligence, routing protocols, wireless communications, information security and adaptive streaming controllers.

Wayne Goodridge is a Lecturer in the Department of Computing and Information Technology, The University of the West Indies, St. Augustine. He did is PhD at Dalhousie University and his research interest includes computer communications and security.