

Survey of Transportation of Adaptive Multimedia Streaming Service in Internet

Miran Taha^{1,2}, Jaime Lloret¹, Alejandro Canovas¹, Laura Garcia¹

¹Integrated Management Coastal Research Institute, Polytechnic University of Valencia.

C/ Paranimf nº 1, Grao de Gandía – Gandia, Valencia, Spain

²Department of Computer Science, University of Sulaimani,

Tasluja Street, n 1, 46001, Kurdistan region, Iraq

E-mail: miab2@doctor.upv.es, jlloret@dcom.upv.es, alcasol@posgrado.upv.es,
laugarg2@teleco.upv.es

Received: April 10, 2017

Accepted: June 15, 2017

Published: June 30, 2017

DOI: 10.5296/npa.v9i1-2.12412

URL: <https://doi.org/10.5296/npa.v9i1-2.12412>

Abstract

Streaming media systems such as YouTube, Netflix, and Apple Music are reining the multimedia world with frequent popularity among users. A key concern of quality perceived for video streaming applications over Internet is the Quality of Experience (QoE) that users experience. Due to changing network conditions, bit rate and initial delay, the multimedia file freezes or provides poor video quality to the end users. Researchers across industry and academia have explored HTTP Adaptive Streaming (HAS), which splits the video content into multiple segments and offers the clients varying qualities. The video player at the client side plays a vital role in buffer management and choosing the appropriate bit rate for each segment of video to be transmitted. A higher bit rate transmitted video pauses in between whereas, a lower bit rate video lacks in quality, requiring a tradeoff between them. The need of the hour was to adaptively vary bit rate and video quality to match the transmission media conditions. Furthermore, the main aim of this paper is to give an overview on the state of the art of HAS techniques across multimedia and networking domains. A detailed survey was conducted to analyze challenges and solutions in adaptive streaming algorithms, QoE, network protocols, buffering, etc. It also focuses on various challenges on QoE influence

factors in a fluctuating network condition, which are often ignored in present HAS methodologies. Moreover, this survey will enable network and multimedia researchers a fair amount of understanding about the latest happenings of adaptive streaming and the necessary improvements that can be incorporated in future developments.

Keywords: HTTP adaptive video streaming, QoE, QoS, Heterogeneous Networks, Network Transportation, Multimedia content.

1. Introduction

Multimedia streaming is the most demanding and bandwidth hungry application in today's world of Internet. The amount of video traffic over the Internet is expected to consume 85% of the overall web data few years down the line [1]. Unreliable network bandwidth and playback issues have an adverse impact on delivered video quality, which is creating trouble for service and network providers towards providing high quality video streaming services. Routing video segments over HTTP caters seamless interaction through a series of firewalls and reliable delivery. Moreover, the best-effort nature of the Internet makes this HTTP-based approach vulnerable to bandwidth fluctuations and network congestion. This leaves a negative impact on the Quality of Experience (QoE) experienced by the consumer market. Recent research reveals that most of the video service providers like YouTube, Hulu, and CNN are migrating from server based streaming to HTTP adaptive streaming (HAS) [2]. Nowadays, many business technology giants like Microsoft, Apple and Adobe have commercial HAS implementations. MPEG, in collaboration with other standard groups such as 3GPP, standardized the HAS interfaces and protocol data, in Dynamic Adaptive Streaming over HTTP (DASH), in 2011 [3]. In this way, a generic standard was formed among the vast amount of available implementations. In a typical HAS technique, the video at the source location is encoded with multiple bit rates, which is represented by different quality layers and each quality is divided into many small chunks. Each chunk contains a short duration of video playback content such as movie or live events. The client application system selects the chunks in an appropriate bit rate depending on the current network scenario and playback buffer status. Once the streaming is started, the client system requests the chunks at an initial bitrate. After elapsing an initial short duration time known as startup delay, the playout buffer gets filled, and the client starts displaying (rendering) the video. While this video is played back, further chunks are requested by the client to maintain the desired buffer level. In this procedure, if the downlink throughput drops off, the client buffer gets empty causing an interruption on ongoing video playback. In order to avoid stalling or rebuffering, the client demands chunks of lower bitrate from the server. Consequently, the user may perceive a quality switch from a higher to a lower quality. On the other hand, when the throughput increases, the client requests higher bit rate chunks, which may again result in a visual change in video quality. This helps to avoid any potential disruption on playback or stalling video, and bandwidth is effectively utilized. The trend of the video service provides at present is to use HAS instead of conventional streaming due to its many inherent benefits. Firstly,

encoding video at variable bit rates aids service providers to provision adaptive solutions as demanded by the end user. For example, the scenario of broadband home users requesting HD video is different from mobile Internet users accessing video over a cellular network. Secondly, it gives an opportunity to the operators to introduce different service classes and pricing. For example, users will have the flexibility to select both high and low-resolution video service class and they can be charged based on the usage. In conventional video QoE related research, the deterioration of video quality temporally varying over larger sequences have not achieved a significant milestone. As a result, most of the standardized quality assessment methodologies for subjective testing were deficient in handling these gaps [4, 5]. Some existing surveys disclosed HAS visual quality [6, 7] or various streaming methodologies [8, 9,10,11]. However, no in-depth analysis on essential application and transport protocols, challenges of streaming with heterogeneous network, and problems and solution of adaptive streaming towards providing better quality multimedia content delivery has been performed so far. Hence, to throw light towards those gray areas of HAS, in this paper we present a survey of transportation of adaptive video streaming in the Internet. All challenges facing the transportation of the adaptive multimedia streaming service in massive networks will be discussed. The structure of the paper is as follows; Section 2 depicts the evolution in Multimedia technology such as video, sound, ads and closed caption. The protocols employed in the delivery of multimedia for both on-demand and live streaming are outlined in Section 3. Thereafter, various transmission techniques such as broadcast, multicast and unicast in multimedia video streaming are discussed in Section 4 and their problems with their respective solutions are explained as well. Then, the issues and solutions of multimedia streaming over the Internet in diversified networks like wired, wireless including cellular networks are illustrated in Section 5. Working methodologies of various adaptive algorithms for streaming and how they aim to improve the problems on QoE and Quality of Service (QoS) for video users, are described in Section 6. In Section 7, the difference between cache servers and CDN servers in context of adaptive bit rate streaming is elaborated. The varieties of codecs and containers for streaming videos are highlighted over section 8. Section 9 narrates the next generation of adaptive video streaming technology used in the Internet. Finally, Section 10 concludes the work giving a holistic summary.

2. Evolution in multimedia technology

Multimedia broadly stands for "more than one medium." In real world scenario, television programs, YouTube songs, movies, even illustrated books are all classic examples of multimedia – as they mostly use a mixture of images, sounds, text and movement. One of the earliest usages of multimedia was found in the form of video game Pong, which was developed in 1972 by Nolan Bushnell (the founder of a then-new company called Atari). The game was composed of two elementary paddles that banged a square "ball" backward and forward across the screen, just like tennis. Another revolution occurred in 1976, when Steve Jobs and Steve Wozniak founded a corporation named Apple Computer [12]. Within a year, they came up with the Apple II, which was the world's first computer to launch color graphics. The computer innovation moved rapidly: 1981 witnessed IBM's first Personal Computer (IBM PC), three years down the line, in 1984, Apple introduced the Macintosh, the

first mass market personal computer system to use an integral graphical user interface (GUI) with a mouse. This remarkably changed the way of performing human-machine interaction functionalities. Another year later, Microsoft released the first version of its Windows operating system. In the same year, 1985, Commodore floated the Amiga [13], a computer based on Motorola 68000 families of microprocessor. It was considered to be the first multimedia computer due to its state-of-the-art graphics processing power, wide range of multimedia gaming, and creative user interface. The Amiga could not sustain over the years and lost its major pie of market share because of competing platforms like Windows, Macintosh, fourth generation game consoles and later, IBM PC [14]. Innovations pioneered by Windows and the Macintosh operating systems laid the foundation stone for the rapid research and developments in multimedia community that were to come. Since both of the operating systems, Windows and Mac, were dealing with graphics and sound – something that was earlier managed by specific applications software – developers were able to build programs that adopt multimedia to have a more influential effect. One American company that has played a vital role in graphics and multimedia domain from the very beginning is Macromedia (earlier known as Macromind-Paracom). In 1988, Macromedia launched its innovative multimedia-authoring tool Director [15], which allowed daily computer users to create stunning multimedia presentations, animations and, CD-ROMs. Later in 2005, Macromedia was acquired by Adobe Systems and Adobe Flash [16] lead most of the animation and multimedia graphics over the Internet. Director is still used to build high-end interactive productions. Figure 1 elaborates the evolution of multimedia applications that has taken place over the past couple of decades.



Fig.1.Multimedia Evolution.

The innovative advancement over the years is absorbed into technological evolution, providing the multimedia content with a superior, swift, and more delightful experience. Modern age multimedia technology is gaining huge popularity in the field of education.

Interactive multimedia courseware, which can be embedded on a CD/thumb drive, have provided an attractive aspect to both teaching and learning. The key components developing the multimedia contents like text, graphics, audio, video, caption are elaborated as follows.

A. Text in Multimedia

Text is the most widely used medium of presenting a message. In multimedia, various colored texts are frequently used for headlines, titles, content, and menus [17]. Several factors that impact the textual communication are font and style, special characters, typeface, kerning, animation, antialiasing, special effects, and hypertext.

Fonts: A font is a set of characters that have a single size. Style and size are the prime attributes of a font [18]. The most generic font styles found are bold and italic. Font sizes are expressed in points. A point is approximately 1/72 of an inch. Text was made more attractive and enjoyable to the eye by including the combination of colors for the font and background. **Text Animation:** Gradually people started presenting text combined with graphics, making it more interesting. A wide range of schemes was devised to animate the text. Some of the methods are: scrolling (vertical and horizontal), fade-in and fade-out, zoom-in and zoom-out, dissolve, etc. Text was also given an impressive look by making it 3D [19]. **Authoring platforms** like Macromedia's Director have an in-built provision to animate text. **Kerning:** It specifies the adjustment of the space between two characters. Kerning helps a few combinations of letters, such as WA, MW, TA, and VA, look prominent. Only the most mature word processors and desktop publishing systems started using kerning. It can be activated or deactivated for individual fonts. **Anti-Aliasing:** Aliasing is the notable event on computer screens and all pixel devices where edges of letters get distorted while preparing a text presentation. Anti-aliasing is the method designed to make smooth edges. Post anti-aliasing, the text is referred as "grey-scale" text. Further adaptations of anti-aliasing have raised both the precision and aesthetics of the on-screen type [20]. **Hypertext:** Hypertext was architected to build links and create an index of words. The index serves to find and group words on the basis of user's search operation. Hypertext systems proved to be fruitful in multimedia interactive courseware for the education sector. Moreover, such systems facilitate both unidirectional and bi-directional navigation.

B. Audio in Multimedia

Audio is another revolution in the multimedia community. The Sound wave in the form of an analog signal is recorded on an audiotape via a microphone or other sources. The analog information was converted to a digital format for storage in a computer. This method is called sampling. Sampling rate is defined as the number of times the analog sound is sampled during each period and converted into digital information. The most frequently used sampling rates in multimedia applications are 44.1KHZ, 22.05JHZ and 11.025KHZ. It was found to achieve higher quality of sound by using higher sampling rates [21]. As the higher sampling rate requires more disk space, an idea was derived to change a higher sampling rate to a lower one (Down Sampling) if needed. Sampling rate and sound bit depth are the audio equivalent of resolution and color depth of a graphic image. Multimedia sound started with 8-bit, 16-bit, 32-bit and 64-bit formats. A single bit rate and a single sampling rate are suggested

throughout the work. Digital sound started getting recorded over microphone, synthesizer, keyboard or DAT (Digital Audio Tape). Progressively, it was revealed that recording employing a microphone connected to a sound card creates difficulty to control the audio consistency and also, the noise was getting amplified in the process. A better practice emerged by initially recording on a tape recorder after making all the changes required and then, capturing it through the sound card. Sound Editors: Sound editors proved to be handy in generating sound, transforming file formats, and augmenting the quality of sound by eliminating the noise. Sound editors like Sound Forge, Cool Edit and Sound Edit 16 became popular for multimedia applications. Furthermore, tools for editing speech signal such as “SoundCutter” and “Symbolic Sound Editor” came into picture around one and half decade back [22]. Sound Forge for PC is one of the most remarkable software for audio recording and editing. A lower cost software called Cool Edit is user friendly, producing good quality of audio. Sound Edit 16 is capable of recording, editing and transforming digital audio easily and quickly. The most commonly used sound file formats are: Window wave format (WAV), compressed file format using MPEG1 Layer three compression techniques (MP3), Digital audio quick time movies (QT), Wave format developed by SUN Microsystems (AU) and Shock Wave audio with compression ratio of 176:1(SWA).

C. Video in Multimedia

Video is a widely accepted communication tool for multimedia presentations. It demonstrates perception besides capturing real world events. Video files consume large memory space and it can be stored such data by two ways:

- 1) Short video clips (length of a minute or two)
- 2) Compressed video files (MPEG, AVI files)

Digital video provides impressive means of communicating real world images and sounds. It is displayed on the screen as the number of frames per second. A rate of 15 frames per second (fps) is proposed across the computing domains and commonly used for most computers, although it could not meet the high quality of 30 fps. The universally accepted video formats are: AVI, MPEG [23], MPG, MOV, and MOOV. AVI file format was promoted by Microsoft for windows. It is also known as video for windows (VFW). MOV, MOOV, QT files are applied for Apple Quick Time Movie. Any Unix system and IBM PC can play Quick Time video clips on their media players. Likewise, MPEG video files can also be viewed with IBM PCs and Unix workstations. Video Compression: As digital video files take up extensive bandwidth and high space in comparison with audio and graphics file formats, the need to reduce the file size was important. In order to cater this, various CODEC techniques came up in the market. As an example, to achieve a compression ratio around 200:1, the MPEG approach that uses inter-frame compression was employed [24]. Although, the quality of the video is compromised if it achieves this level of compression. An inter-frame compression omits the visual information that is not perceivable to the human eye. Video editing software like Adobe Premiere 6.0, Apple's Movie 2.0.1, Pinnacle Studio, and CoolEdit entered into the business place.

D. Graphics in Multimedia

Graphic presentations contributed the richness of today's multimedia. There seemed to be two ways in designing graphics which are: a) Raster graphics; and b) Vector graphics [25]. Raster graphics, broadly known as bitmap images, depend on a grid of pixels; whereas vector graphics are based on mathematical formulas. Bitmap images are related with 'paint' or 'photo' and vector graphics are correlated with 'drawing' or 'illustration'. Vector graphics involve lower memory and there is not much loss of resolution when the image size is altered. A few of the regularly used graphic formats are: Graphics Interchange Format (GIF), Joint Photographic Experts Group (JPEG), and Portable Network Graphics (PNG). GIF images load faster due to their small size in comparison to other formats. GIF files are smaller in size and do not blur any part of the image. But GIF is capable of supporting only up to 256 colors, whereas its counterpart JPEG can exhibit up to 16 million colors. The main drawback of JPEG lies in the loss of quality. PNG was designed to be an alternate solution of the GIF file format. PNG formats have basically two variants: PNG-8 format, containing 8 bits of color information (same as GIF), and PNG-24 format, holding 24 bits of color (like JPEG). PNG 24 is a lossless format. Scanning: The basic process of scanning converts an analog image to a digital one. An image is generally scanned at resolutions between 50 to 1200 Dots per Inch (DPI). Resolution of an image specifies the number of Pixels per Square Inch. This is known as dots per inch or dpi. Higher resolution is provided improving the quality of the image. Image Editing: After digitization, images can be fine-tuned through any image editing software such as Adobe Photoshop or JASC's Paint Shop programmed. This software was able to improve the image quality, and do certain manipulations like rotate, crop, flip, duplicate, and fill the image. Deletion and addition of one image to another image is also an impressive feature.

E. Animation in Multimedia

Animation is described as a simulated representation of a series of moving pictures or frames. It adds the dimension of time to graphics. Computer animation is significant to multimedia as all presentations are made on the computer. The major components of animation are: key frames and tweening. Key Frames: The Key frames are comprised of major frames of animation and they are created first. Many changes are captured within key frames. Such frames are designed to show the behavior of moving objects with time. Many techniques are researched for automatically detecting the key-frames in a captured motion data [26]. Tweening: It is the mechanism of creating intermediate frames between two images to give a feel that the first image expands smoothly into the second image. Tweening is a vital technique in computer animation. Software Tools: The software used for animation decides the quality of the multimedia animation composition. A few trendy animation software packages for windows are Animator Studio, 3D Studio Max, Adobe Premiere, Flash, Softimage etc. Software packages for Mac incorporate Elastic Reality, Adobe Premiere, Strata Studio pro, etc. Animation File format: The output file formats for animation depend on the kind of software employed. Accordingly, various files formats with extensions like .fla (for flash), .dir (for Director), .dcr (for shockwave animation file), .max (for 3d studio max) are available.

3. Transportation of Multimedia streaming

Transmitting multimedia signals with reliability and good quality of experience from one place to another is a challenging task. A communication system joins these source-destination pairs applying specific protocols. Firstly, multimedia signals are compressed and encoded by application-layer specific protocols. Thereafter, the encoded bit stream is delivered among end-to-end systems (for example, client-server) using a transport layer protocol. Transport protocols are responsible for the packetization and delivery of the encoded stream, whereas the original media stream is reconstructed from the received delivery packets by efficiently decoding the signal. Post decoding, it is played back by media players. Figure 2 depicts the correlation among all the layers in the standard system for processing multimedia information. Application layer supports end-user processes such as audio/video, encoding/decoding and packetization. Communication partners define the quality of service for the multimedia transmission, user authentication and privacy [27]. This layer provides specific application services for HTTP streaming as a whole. The transport layer takes the responsibility of end-to-end data transfers between a set of hosts and it also administers error recovery with a flow control mechanism. The network layer contains critical protocols like IP, ICMP, IGMP and other services, which cater to switching and routing, creation of logical paths, or virtual circuits for transporting information from one node to another..

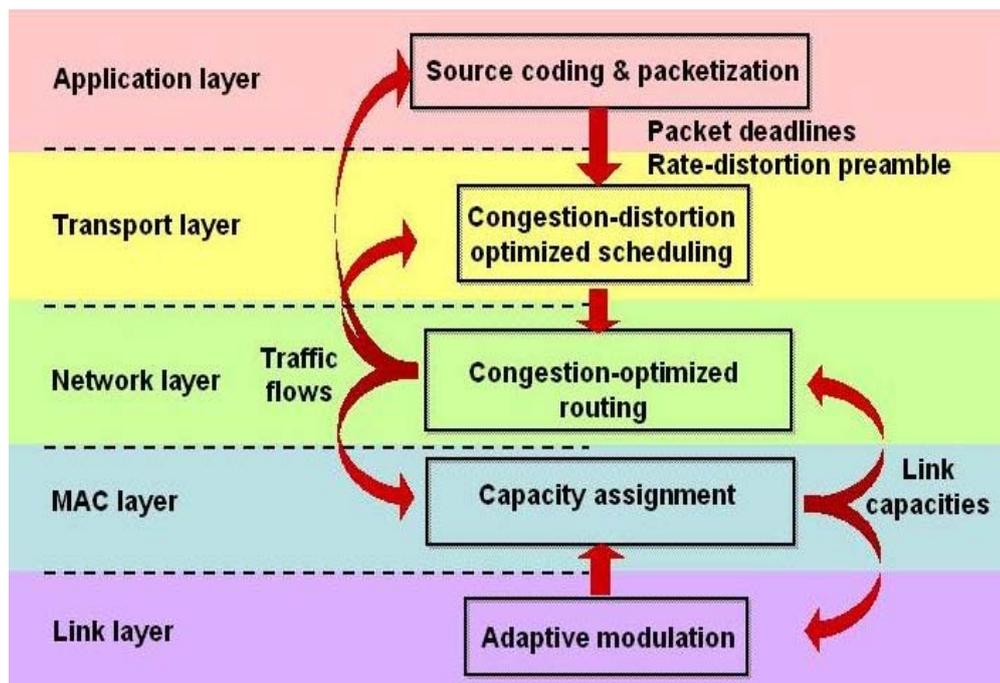


Fig.2 Layer wise media processing.

This layer performs the vital tasks of addressing, routing, data forwarding, internetworking, packets sequencing and congestion control. At layer 2, data packets are represented into bits while encoding and decoding. It is equipped with functionalities like transmission protocol management, error handling of physical layer, frame synchronization and flow control. The data link layer is further branched into two sub layers known as Media Access Control (MAC)

and the Logical Link Control (LLC) layer. The MAC layer is mostly concerned with controlling the media access techniques for a computer to transmit over the network [28]. The LLC layer supervises the frame synchronization, flow control and error checking procedure. The physical layer converts the bit stream into an electrical, light or radio signal and dispatches it towards the external network at the electrical and mechanical levels. It involves a variety of hardware and interfaces for sending and receiving data over a carrier, such as coaxial or twisted pair cables and other media, which uses EthernetRJ45, RS232, RS485, USB, CAN and ATM protocols in general [29]. Over the past few years, media analysts have come up with many protocols for multimedia content streaming as described follows:

A. RTP

Real-time transport protocol (RTP) is basically a protocol that aids in transmitting a video and audio data stream over an IP-based network. RTP is widely deployed in real time streaming media, like video conferencing, television services and web-based push-to-talk features. RTP is also the key building block of the voice over IP network and was developed by the Audio-Video Transport Working Group of the Internet Engineering Task Force (IETF). It was first published in 1996 as RFC 1889, superseded by RFC 3550 in 2003. When multimedia content is transferred via IP network, often the data arrives in an out of sequence fashion. Services provided by RTP are out-of-sequence packet detection upon arrival, time reconstruction, jitter compensation, and security. Primarily, RTP is mostly used for real time data transfer using IP multicast but it can also be utilized in unicast transmission. Video-on-demand services that require one-way data transport, employ RTP [31]. RTP has got different profiles and payload formats for each class of application type i.e. audio, video, etc. Each profile needs separate codecs to encode the actual payload information. The codecs for audio include G.711, G.723, G.729, and QCELP and, for video, it is H.263, H.264, and MPEG-4. Therefore, RTP works in association with RTCP to get transmission-quality related feedback and relevant user details.

B. RTCP

Real-Time Control Protocol (RTCP) is another control protocol that takes part along with RTP for delivery and packaging of multimedia data. The purpose of RTCP is to monitor the necessary Quality of Service and delay compensation imparted by RTP. RTCP is deployed for applications like Voice over IP (VoIP), media streaming, Internet protocol Television (IPTV), and video conferencing [30]. RTCP also enables session control in a RTP session and acts as an end point identifier to all the participants in a session. All members in a session send periodic transmission and reception statistics reports or RTCP reports. With the increasing number of participants using a session, these report messages are going to consume high bandwidth, creating probability of congestion. Thus, RTCP dynamically controls the frequency of report transmission and ensures that only 5% of the total session bandwidth is engaged. In order to adopt RTCP for large-scale applications like IPTV, there might be long delay between RTCP reports. This certainly poses a challenge towards controlling congestion and network bandwidth usage [31]. Moreover, in such case, the acceptable frequency of sending reports are generally less than one per minute, making it prone to inaccurate

reporting of necessary statistics by the receiver system. Schemes like RTCP filtering or RTCP biasing have been designed to mitigate such issues.

C. RTSP

Real-Time Streaming Protocol (RTSP) is an application layer protocol, which is used in communication systems for controlling and setting up real-time sessions between end users, especially between a streaming server and its client. In a typical video on demand scenario, the client issue commands like play, pause, and record to control the streaming from server to client. RTSP protocol is used along with RTCP for delivering multimedia streams. The streaming technique divides the data into many variable size packets depending on the network bandwidth condition. After the client receives sufficient packets, the client's player can start playing the first packet, decompressing the second one and downloading the third at the same time. This facilitates the users to enjoy multimedia contents almost in real time without downloading the whole media file from server. This principle is applied for both live and on demand streaming methodologies [32]. RTSP deals with request and response packets to handshake with the server by using TCP or UDP protocols. A company named RealNetworks along with Netscape and Columbia University jointly developed RTSP first in 1996. Later, RealNetworks customized RTSP to Real Data Transport (RDP) protocol [33] for streaming server.

D. RSVP

Resource Reservation Protocol is a transport layer protocol that allows reserving resources, such as bandwidth, across the communication network. A network host or router can request a specialized end-to-end quality of service for application data flows. RSVP is considered the foundation stone of future Integrated Services Internet that caters both best-effort and real-time services [34]. Many real-time applications also use RSVP to retain the required assets at intermediate routers along the transmission paths to make the demanded bandwidth available during an ongoing transmission. The receiver contains a special software or browser embedded program that sends a RSVP request before the actual transmission starts. This assists in allocating sufficient bandwidth and relevant priority for scheduling packets. Such a request will reach to the nearest Internet gateway with a RSVP server. Then it will figure out, whether the receiver node is eligible to get such a reservation arrangement and, if so, then the satisfactory bandwidth continues to be reserved without disturbing previous reservations. In such process, if the reservation request is granted, the gateway node then relays reservation information to the next active gateway towards the destination. Thus, reservation is assured throughout the path to the destination. In the case when a reservation request gets rejected, all reservations are dismissed. Over the past couple of years many extensions of RSVP have been proposed to address protocol scalability issues [35].

E. TCP

Transmission Control Protocol (TCP) is a connection-oriented method, which is reliable and supports flow control with byte-stream in a full duplex fashion. In most of the multimedia applications, the suitable delivery mode is multicast transmission, which serves a group of

specific clients simultaneously. So, in this case, the connection-oriented property of TCP is not appropriate, as it supports only application level point-to-point connection between a server and one client. Hence, the multicast technique to transmit data to more than a single client is not feasible by the TCP protocol. In earlier days' research, it was thought that a reliable connection would be beneficial for multimedia. Reliable ensures that there will not be any packet loss. In other way, the client needs to acknowledge every single received packet and will wait until the missing packets are received by a retransmission. To obtain such high quality of service [36], it will incur in wait cycles introducing significant delays. Delay is the enemy of real-time data transmissions. In reality, it does not matter much if a single media packet is lost. For example, while transmitting a video, a lost packet can be dropped, and the previous packet can be displayed a second time. For human vision, this little deviation is not perceivable and does not have any quality effect. So, it is deduced that a reliable transport layer protocol is not the best fit for multimedia transmissions. TCP's ability of flow control makes sure that audio or video data frames are received in the right order, as IP is not responsible for delivering the packets from the network in a sequence. The full duplex mode byte-streaming ability of TCP is desirable for real-time application. But, an in depth observation reveals that, video broadcasting for multiple clients does not require this full duplex mode. Therefore, TCP is not the most applicable approach to transmit multimedia real-time data over a network [37].

F. UDP

User Datagram Protocol (UDP) is a connectionless protocol that employs best effort delivery and does not present any flow-control technique. UDP is unreliable, causing a boon for real-time multimedia data transmissions. As the connection orientation property for multimedia is not so important, UDP is a good choice. UDP also supports multicast methods [38, 39]. In the manner of broadcasting a video to more than one client, this is exactly what is needed. UDP seamlessly supports multicast and broadcast methods for transporting the data on a network segment to more than one client. Without using a UDP-like protocol, it would be necessary to set up sessions with every client and to overload the data on the application layer. Then only it could be able to create multiple streams on a single network segment. UDP is comparatively much faster than TCP [37], because it does not employ flow control, error checking, error correction, or acknowledgment procedures. So, through the Internet, if the transmitted data gets affected by collisions, UDP will provide erroneous data. Streaming media such as video and audio mostly use UDP because of its acceptable high speed and lesser delay.

G. HTTP

The Hypertext Transfer Protocol (HTTP) is an application layer protocol for transferring information like text, audio, video, graphics and images over the World Wide Web. It also uses TCP or UDP as the underlying protocol for end to end content delivery. The progressive development of HTTP was regulated by the Internet Engineering Task Force (IETF) and the World Wide Web Consortium (W3C). HTTP operates as a request-response protocol in a typical client-server architecture. In general, a server hosts any website or web application

that runs on a computing platform and the client uses a simple browser running on the other end of the internet across routers and firewalls. The client sends a HTTP request message directed to the server and upon receiving request, replies with a response packet with most of the requested information. HTTP is widely used for multimedia content transportation such as HTTP Live Streaming (HLS) by Apple Inc, as part of their QuickTime, OSX software. Furthermore, HTTP takes a major role in progressive media download where an end consumer begins the playback of the digital media file while the download is yet to be completed from the server. In such case, the end users sense the presented video as streaming media. However, in most occurrences of progressive download, the video is being played from a HTTP web server in comparison to a streaming server [40]. Moreover, HTTP first stores the progressive video in the hard drive and then plays it, whereas the streaming video is not cached on local drive for playback.

H. SDP

The Session Description Protocol (SDP) is designed to depict a multimedia session with the goals of session announcement, session invitation, and parameter negotiation. Although SDP does not hand over the media files itself, it negotiates media type, format and other session related parameters between end points. In order to illustrate a multicast session, SDP is utilized in partnership with RTP, RTSP and SIP protocols [41]. The primary session parameters are session name, session time and purpose, required bandwidth, addresses or ports of all end points.

Of late, progressive video streaming technology was used for delivering video content over HTTP which is quite similar to the way images gets loaded in a web browser, from top to bottom. However, it has got some major pitfalls as users can't watch a video without fully loading it. The video file, which is downloaded, can't be changed on the fly and quality of service varies based on video file format. In addition to that, such videos are a little unsafe as anybody can copy and save them. To overcome these roadblocks, many streaming providers used Adobe's RTMP protocol and their Flash Media Server (FMS), which adds much cost and complexity. The emerging market of mobile consumers and their incompatibility for Adobe's Flash, made HTTP streaming to surge again. Few different HTTP streaming approaches have been launched to resolve the shortcomings found in progressive HTTP streaming, enabling more flexibility compared to RTMP. Such HTTP streaming services deploy adaptive-bitrate techniques that support changing video quality on the fly as the source multimedia files are divided into smaller chunks and encoded into various bitrates. Based on the bandwidth availability, a superior or lower quality stream can be demanded by an end user for smooth playback. In addition to that, HTTP streaming utilizes CDNs to cache all video content very close to end-viewers [42]. Thereby, such content availability closer to viewers improves the performance by decreasing the number of hops needed for fetching the requested video file. TV is also slowly moving towards the mobile web and this opens the platform for adaptive HTTP streaming to be the backbone of many streaming mechanism that want to provide superior viewing experiences on many devices. As all HTTP streaming formats add security and DRM capabilities, a variety of multimedia users kept shifting from old media to IP. HTTP streaming protocols have been developed in recent time to enable an

easy and efficient way of video content delivery. Some of these techniques are highlighted and their comparison is also performed as follows.

1) ***HDS***

HTTP Dynamic Streaming (HDS) was introduced by Adobe Systems as a substitute to their RTMP protocol. HDS permits adaptive streaming using HTTP to most of the Adobe Flash compatible devices. The streaming technique with HDS notably reduces the cost of operating the stream as compared to RTMP. Adobe has also come up with a module for Apache, the widely used Open Source HTTP server, which makes it work as an origin streaming server. As the Flash player is already widely used in the consumer market, HDS has got an added advantage to be used for streaming to desktop computers. Although, the compatibility issues of Flash with platforms like Android and iOS restrict the practical use for broadcasting to many mobile devices like smartphones and tablets [41].

2) ***HLS***

HLS or HTTP Live Streaming is a standard protocol introduced by Apple Inc. for their proprietary iOS devices and QuickTime players during an initial period. Later, their support was extended for Android 3.0 or Honeycomb making it an ideal choice for streaming to plenty of mobile devices. Most of the mobile devices are equipped with a large number of client video players with the default HTML5 player for existing mobile browsers. HLS is widely deployable utilizing many HTTP servers (such as Apache) or a couple of commercially available streaming servers like Adobe FMS and Wowza. Apart from that, there are many streaming services and CDNs, which are able to stream for HLS-supported players through trans-muxing, or on the fly re-bundling the existing video streams into their HLS enabled packets [42]. Because of all these inherent benefits, HLS took over as the key mobile streaming protocol for the majority of the available video content websites.

3) ***HSS***

The HSS or HTTP Smooth streaming is Microsoft's initiative into the field of adaptive HTTP streaming that executes upon their IIS web server and Silverlight player. Silverlight player intelligently detects local bandwidth and CPU conditions. Seamless streaming services are provided by adaptively switching bitrates. Multiple audio and video codecs are supported by HSS and they can also be easily customized [43]. It is very used for large-scale streams, for example, NBC's online broadcasting of the Olympic Games, and its form of a functional block of Netflix's streaming protocol stack. Using H.264 video codec streaming for Apple, IIS server is capable of transmuxing HSS fragments into a format, which is compatible with iOS 3.0 and later devices. Therefore, Table1 discloses a comparative study among the proprietary adaptive streaming technologies [44].

Table 1: Comparison of HAS commercial platforms

Features	HDS	HLS	MSS
Multiple audio channels	-	Yes	-
Closed captions	Yes	Yes	Yes
Effective ad insertion	-	Yes	-
Fast channel switching	Yes	-	Yes
Content encryption	Yes	Yes	Yes
Audio and video multiplexing	Yes	Yes	-

4. Multimedia Transportation Methods

Any multimedia information from source to destination can be transported by either unicast, broadcast or multicast methods. In unicast transmission, the desired content is sent to only one user at a time whereas in broadcast, the content is transferred to all the users present in the current transmission area. Sometimes users are divided into specific groups as per their needs. The multicast technique is used for a desired group of users instead of sending the message to all of them.

A. Unicast

Unicasting mode simply sends packets from a single source to single destination . This can be as simple as transmitting information from a web server to a user looking at a particular page on a web browser. Moreover, in many streaming content operators' services, unicast-based media servers open and impart a stream for every unique user. Data (or video) can be delegated via UDP or TCP, with RTP and RTSP, and can be handed over using unicast. That is why unicast is considered to be a method, not a protocol. While unicasting, an IP packet passes through many switches and routers over the Internet. Initially, the switch or router does not have the knowledge of the location of the destination MAC address. In such case, the packet will be broadcasted to all the ports/interfaces on the switch/router. This shortcoming of Unicast to locate a single device is known as Unicast flooding. Unicast messaging mode is implemented for some network processes in which a private or unique resource is demanded. A few network applications where traffic needs to be mass distributed, proves to be pretty costly if conducted with unicast transmission. Because each network connection here consumes computing resources towards sending to the host side and demands its own isolated network bandwidth to transmit. Such applications incorporate

streaming media of several forms. Internet radio stations that use unicast connections might incur in high bandwidth expenses. The limitation caused by unicast flooding and bandwidth wastage, as observed in multimedia unicast distribution, is shown in Figure 3. A video server is streaming a 1.5Mb file to each of the 3 users, consuming a total of 4.5 Mb and also, each intermediate router/switch is flooding packets due to the unknown unicast target device. In a situation where the group size is large, the same copy of information will be transported multiple times, posing as a serious restriction of network scaling.

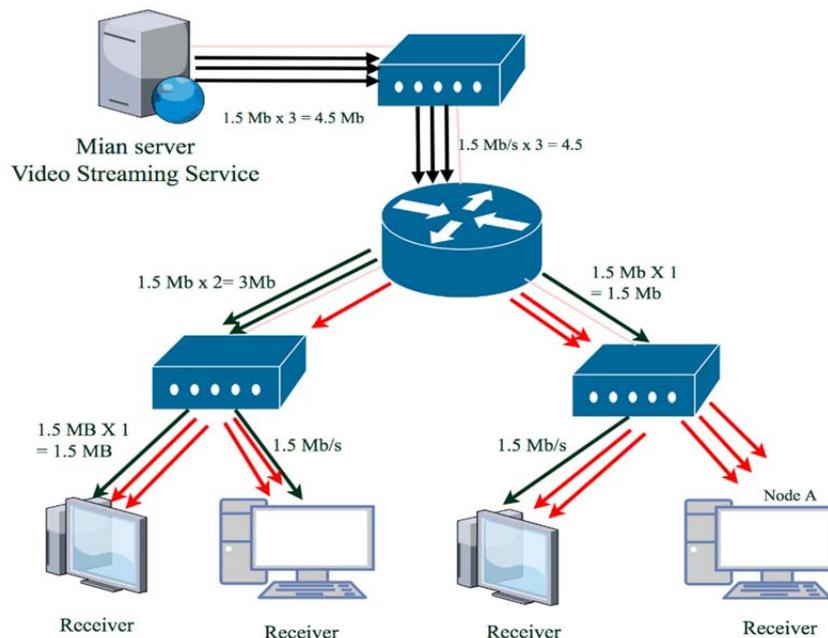


Fig.3. Multimedia Traffic using Unicast.

B. Broadcast

In Broadcasts, media traffic is sent to all reachable devices in the network [45]. All the switches and routers upon receiving broadcast-type traffic on one input port/interface, distribute to all other output ports. That is why a large amount of broadcast traffic can bring down the network performance creating a situation called “broadcast storm”. Figure 4 is depicting the phenomena where a remote video server is broadcasting multimedia contents to all hosts, but a few recipients don’t want to receive it. Still, users’ system resources like CPU will be consumed to process the broadcast packets. Thus, this approach is inefficient if the intended recipients are limited to only a small subsection of a larger network. However, the volume of broadcast traffic can be reduced through using the Broadcast Limit parameter. This limit is defined as the percentage of the theoretically feasible maximum of attainable network bandwidth, which can be utilized for broadcast traffic. Any broadcast traffic will be dropped upon exceeding that limit. The authors [45] exposed the novel methodologies in multimedia broadcasting by economizing the network capacity. They used forward error correction (FEC) and interleaving for multimedia content that is to be broadcasted and also designed a receiver

technique for decoding, forwarding error correction and de-interleaving the received multimedia data for playback.

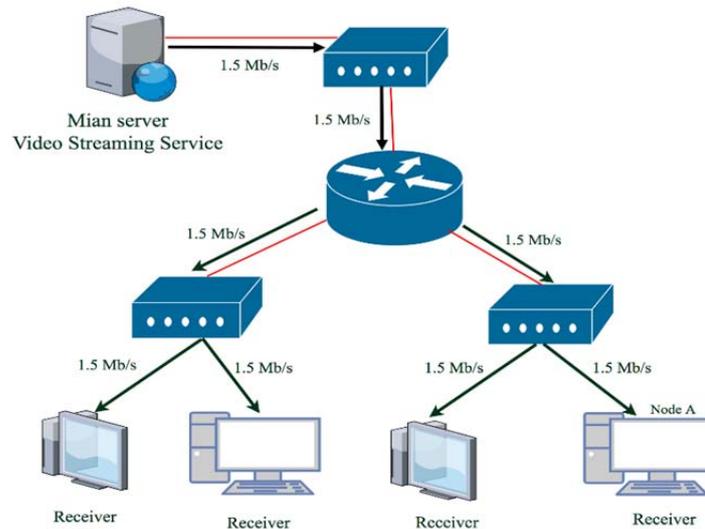


Fig.4. Multimedia Traffic using Broadcast.

C. Multicast

Multicasting is transferring data from a source to a group of destination nodes. Each node expresses its interest in receiving the data, by participating as a member of the group [46, 47]. On the other hand, in unicast and broadcast scenarios everybody obtains the packets whether they want it or not. Multicast traffic is addressed towards a group of network devices which requires one-way connection, hence, using TCP is ruled out. Usually, if a switch receives multicast traffic on one incoming port, it spreads the traffic through all the other outgoing ports. Whichever devices have availed multicast membership by joining a particular multicast address group, accept the multicast traffic. This acceptance filtering is done in the network layer of the end devices. Multicast-based multimedia servers can serve a large scale of the network audience by delivering content simultaneously to many users. An example of multicast mode transmission is exhibited in Figure 5 [48], where multicast traffic is accepted by only selecting a number of hosts and not all the devices. In [49], the authors used MPEG4 FGS video bi-stream to multicast to a set of heterogeneous receivers over the 802.11 wireless device through a UDP socket. They formulated a problem scenario, where each user experienced quality degradation due to targeting multiple receivers for multicast. They derived an optimal solution that effectively clubbed FEC coding with progressive source coding by expanding the existing concepts of MDFEC to the multicast case. Reference [50] proposed a multicast architecture that was scalable in nature and designed to support large overlay networks. Their approach addressed sub-optimality of the adaptive hybrid error correction (AHEC) process in multicast scenarios. They constructed a hierarchical multicast tree topology to upgrade the performance of AHEC and ensure QoS for the multicast clients. They showed significant reduction of redundancy information into the network by introducing multistage multicast architecture.

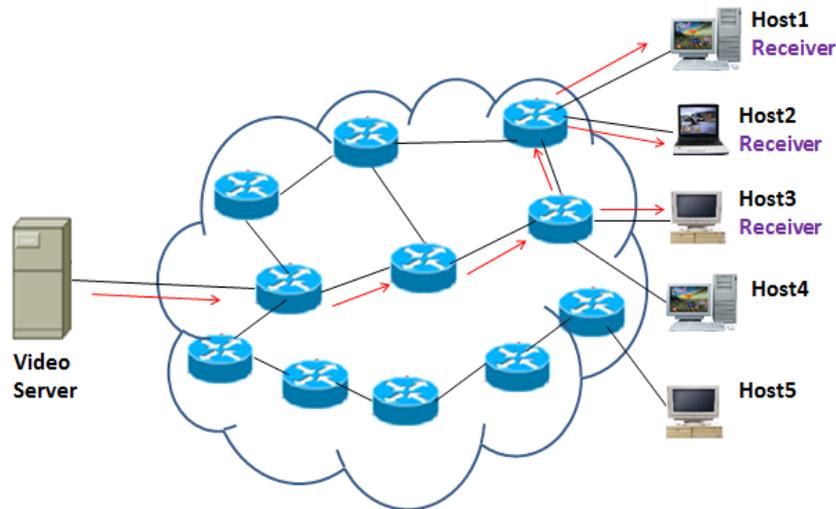


Fig. 5. Multicast Transmission

5. Multimedia streaming over heterogeneous networks

Real time Multimedia data imposes serious timing constraints. Due to this nature, audio and video traffic generally has delay, bandwidth and loss requirements. Moreover, multimedia packets may traverse a variety of networks, e.g., wired networks, wireless local area networks (WLAN) and cellular networks. Each of these networks has different tendencies of delay, reliability, bandwidth, jitter and medium access control (MAC) procedures. Among such diversities, delivering multimedia content over a wireless channel commands more challenges in comparison with traditional wired Networks [51]. This is because of varying channel conditions, signal interferences and wireless devices' impediment of battery capacity, memory size, processing power, screen size, limited support of all video formats, etc. Thereupon, there are the influential factors that affect the media transmission over wired and wireless network as investigated by the research community and the industry.

Packet Loss: Packet losses in wired networks appear primarily as a result of congestion in between the sender and the receiver, whereas in wireless networks, packets are lost mainly due to data corruption [52]. As a consequence of the low Signal to Noise Ratio (SNR), multi-path propagation of signal with fading and the interference from the transmissions of neighboring wireless entities surges.

Mobility: Wired networks are mostly static and are not affected by node movements. But in wireless scenario, mobility is a roadblock for multimedia data transmission [52]. This ushers a new design ultimatum to the networking world as it creates uncertainty towards achieving the guaranteed high bit rates.

Bandwidth: The network bandwidth fluctuates with time, and wireless networks face high loss due to lower bandwidth compared to its wired counterparts.

Latency: It is defined as the time it takes a multimedia transmission to traverse from its source to its destination. Wired network delay is significantly lesser than its wireless counterpart. This is because the multipath fading nature of wireless channels.

Jitter: Jitter is caused by a variation of latency or delay. Non-uniformity in delays demands effective buffering at the endpoints and application level sequencing. For similar reason as with delay, wireless networks are prone to jitter that affect voice and video applications more than to the wired infrastructure.

Security: For multimedia data, the application should have the provision of user authentication and encryption enabling an end-to-end security for real time data. Wireless networks are more susceptible to security hacking as the broadcast nature of wireless channel provides easy access to intruders.

User Heterogeneity: Wireless users have smartphones, tablets/laptops and smart watches as end devices to play the multimedia whereas, wired users might use LCD/LED TVs of 2" to 42" screens, desktop PCs and many more items [53]. This diversity introduces a challenge for the audio/video quality and smooth playback.

Interference: Interference is mainly spotted in wireless networks, caused by neighboring station transmission that disrupt the ongoing data transfer.

Therefore, authors of [54] evaluated the performance of a video streaming and VoIP application over IPV4 and IPV6 networks for wired and wireless communication. The output of the measures of the video streaming and VoIP is summarized in Table 2. It shows that wireless networks incur in 2-3% more packet loss than wired IPv4/IPv6. They observed IPv6 to be a much reliable network protocol providing more throughputs and lesser packet loss.

Table 2. Wired versus wireless communication performance

Scenarios	Video streaming	VoIP
IP4 Wired Network	0.52%	0.20%
IP4 Wireless Network	0.55%	0.22%
IP6 Wired Network	0.38%	0
IP6 Wireless Network	0.40%	0.03%
IP4 Wired Network	191ms	182ms
IP4 Wireless Network	205ms	175ms
IP6 Wired Network	185ms	181ms
IP6 Wireless Network	194ms	185ms

Another recent work [55] surveyed various challenges involving video streaming over Heterogeneous Wireless Networks. It addressed a few benchmarking techniques to overcome those challenges. They proposed a Fuzzy logic based MADM algorithm to deal with wireless network uncertainty and decision-making. A cross layer design was also suggested for

monitoring the link state at the MAC layer and for video rate controlling on the application layer. Loss Tolerant Bandwidth Aggregation (LTBA) technique was analyzed based on Gilbert loss and Markovian time chain. The LTBA can handle video frames over a lossy channel with a loss probability of 40% and guarantees timely delivery of 97% of the frames. A customized source channel coding technique was introduced to incorporate redundancy under high demands when video transmission is going on, adapt to challenging delay issues and reduce the video distortion during delivery.

Buffer and Rate Optimization for Streaming (BROS) algorithm [56] was developed to enhance streaming media application performance by minimizing the buffer underflow, frame loss, and buffer delay for wireless environment. BROS applied an application layer bandwidth estimation tool to pass on information about the effective capacity, bandwidth availability, and bandwidth variation for the wireless bottleneck scenario. It optimized the streaming rate selection and the playout buffer size by deploying a Markov chain model.

However, some amount of frame loss and buffer underflow was still there from the experimental analysis and wireless domain phenomena as if strong interference and mobility was not fully considered in the design. BROS was developed into the Emulated Streaming client-server system over Linux and tested on IEEE 802.11 wireless test bed by varying wireless channel parameters like low signal strength, idle channel, contention based channel, and rate adaptation.

A novel cache placement mechanism was introduced for wireless two-tier content delivery networks [57], which engaged separate channels for content distribution and content service. Furthermore, the authors considered a delay cost because of contention, which hampers the performance in shared wireless channels, as a vital design parameter for cache placement strategy. They also came up with a cross-layer heuristic approach that yielded better performance than existing methods in terms of packet delivery ratio, delay, and the amount of delivered packets within the scheduled timeline.

Security is still a matter of concern for multimedia traffic over networks. Interestingly, wireless networks are more vulnerable to denial of service attacks than conventional wired networks. Security issues can be addressed by embedding robust logic in physical and application layers. Physical layer configuration is dynamic and dependent on the condition of wired/wireless networks, and the application layer mainly deals with multimedia delivery aspects. Considering these factors, a joint framework was proposed [58] incorporating both the physical and application layer security parameters. The signal processing mechanism and security capacity was tweaked for physical layer, whereas watermarking strategies were enforced at the application level. Moreover, this combined plan of action was capable of using quite low communication overhead, making it flexible for large-scale wireless multimedia systems.

With the advancement of adaptive streaming using HTTP over internet, it becomes essential to evaluate its performance over different access networks. Few critical aspects in this regard like segment duration, transmission bitrate, and buffer specifications are still an open research question. Authors in [59] experimented HTTP streaming sessions over classic networks such as DSL, WLAN and UMTS to study the effect of these characteristics on the overall execution of a streaming. Simulation analysis showed an occurrence of late segments

between 7.88% for wireless cellular networks and 17.72% for the wireless LAN case with around 50% of the segments still turning up in less than 50% of the corresponding segment duration. The results concluded on the immense instability of these access technologies and hence, indicted the prompt longing for adaptively in the arena of HTTP streaming.

Authors in [60] assessed the performance of streaming MPEG-4 video in an admixture of wired and wireless architecture. The streaming video server was placed on the wired network and it was streaming to wireless clients. On the other hand, another video server situated in the wireless network was streaming to video clients in a wireless network. These two scenarios were compared for a number of simultaneous streams of videos with alternating frame sizes and frame rates. The investigation determined that the server located at wired side is susceptible to significant loss rates when there are multiple concurrent video streams whereas, it's wireless counterpart (wireless server) achieved a lower loss rate but higher delays due to accessing the wireless channel among other competing stations and getting its chance to transmit as per the MAC protocol.

An enterprise wireless network possesses its intrinsic challenges for multimedia transmission. Enterprise solutions like B2BSIP (Back-to-Back Session Initiation Protocol) were proposed [61] to cater application layer multimedia QoS adaptation and link layer dynamic rate control. A back-to-back user agent in enterprise network is principally dependent on wired network due to its ample bandwidth availability and so, improvising wireless part of network is neglected. In this context, a B2BSIP server architecture was designed to persuade high quality of service by administering a contemporary rate adaptation algorithm that diminishes I frame instances of the video while still satisfying the delay prerequisite of the video, which was saving bandwidth. This method determines that the application layer QoS needs feedback from the data link layer and employs optimal QoS configuration for the B2BUA server which quickly adapts to the new parameters. The server is generally located in the proximity of the wireless base station in an enterprise network. This concept was verified by ns-2 simulation framework on a simulated access link of mobile WIMAX system.

Research in [62] compared wireless access technologies regularly used in Poland. The first group of technologies chosen was the IEEE 802.11 wireless LAN family, which is extensively used as the "last mile" connectivity in rural areas for indoor environments. The second group included cellular packet switching technologies starting from initial GPRS up to advanced HSPA+. The work disclosed feasible data transfers in theory and correlated them against results collected by testing existing real networks. The average jitter and overall delay parameter was monitored while running streaming for wireless connections. Evaluation on real networks revealed that wireless networks executed streaming with not much degradation in quality. However, a large disparity was observed between the popular 3G standard - UMTS R99 and its extensions - HSDPA and HSUPA. However, the latest cellular technologies like 4G LTE and LTE-Advanced were not considered for evaluation in this research.

6. QoE in HAS

The surveys on the earlier section suggest that adaptation to the network is the key

condition towards achieving good quality of multimedia experience. Also, it is essential to understand the root influence factors, which impacts the perception of the viewers. In this regard research [63] inspected few critical aspects as following:

A. Initial Delay

A specific amount of video packets must be transported to the client before the actual decoding and play back starts. This creates a certain amount of initial delay in multimedia streaming. The realistic value of such initial delay is dependent on transmission bandwidth and encoding techniques. While the video play back is delayed, the play out buffer is filled out with a longer playtime at the receiver end. There has to be a tradeoff between the span of the actual delay (as longer buffer playtime = higher initial delay) and the uncertainty of buffer deficiency, i.e., stalling (longer buffer playtime = higher resilience to withstand throughput variations). They revealed that the effect of initial delays varies from application to application. Because of this, applications like Web page loading, UMTS connection setups and IPTV connection set ups can be easily transferred to adaptive streaming. However, many earlier works presumed a logarithmic relationship between initial waiting period and mean opinion score (MOS), which indicates a subjectively measurement of perceived video quality (QoE).

B. Stalling

This is known as the discontinuation in playback of video data when the play out buffer depletes. In some instances, when the throughput of a streaming phenomenon is lesser than the video bit rate, the buffer will run out of content to play back. This stops the continuation of ongoing video due to insufficient buffer data. The playback can resume only after the buffer collects a threshold quantity of video data. Once again, the period of rebuffered playtime should be traded off between the length of the discontinuation (longer buffered playtime = larger stalling duration) and the uncertainty of a quick persisting stalling event (extended buffered playtime = stretched playback until next possible stalling event). The fact that an increased period of stalling reduces the quality is exhibited [63]. Moreover, it was observed that one long stalling action is favored compared to recurrent short ones. Although they concluded that the position of stalling might not have much significant impact on the quality, as debated in [63], which emphasizes the position. There is no doubt that stalling, even a little amount, downgrades the anticipated video quality. Thus, all video-streaming services should avoid stalling whenever possible, as little stalling severely degrades the perceived quality. Conventional HTTP video streaming is unable to handle oscillating network conditions and can only adjust to the stalling period and playout buffer size. Whereas, HTTP adaptive streaming is quite flexible to regulate the delivery of streaming information to the present network anomalies, and cater the stalling restriction.

C. Adaptation technique

The adaptive principle for streaming takes place both in the client and on server side. Initially, all clients send HTTP request packets to the server for retrieving metadata of the

various audio and video profiles available. A dynamic switching among such profiles can happen at a periodic or variable time gap while the playback is on progress. Also, there is a decision engine running at the client device, which determines about what profile or representations to demand from the server at the present scenario. On the other hand, the streaming contents are prepared at the server location, which takes care of choosing suitable encoding, compression method, length of the segment, etc, to make it palatable for the client.

D. Segment Length

Another vital parameter for HAS is segment length that indicates the minimum video period, elapsing which quality adjustment can occur in terms of bitrate and other settings [64, 65, 66]. As an example, for a live streaming, the video content is available at the location of server and simultaneously user is viewing at client side, which will ensure a low overall delay by the system. This requires smaller segment lengths and the segments should be downloaded as they appear on the server. On the contrary, for video on demand systems, longer segment length along with a large receive buffer is needed to avoid interruption due to periodic or dynamic quality profile changes. However, determining the value of the optimal segment length is difficult, and varies on the network (fixed access vs. mobile users) and the content (High resolution vs. Low resolution, premium vs. non-premium). As an example, short segment length is better to adapt rapid bandwidth variation and avoid stalls, but longer segments yield for an improved quality and encoding efficiency.

For the network/internet point of view, there exist other influencing factors to be taken care. Longer segment size may create stalling when using Internet by wireless communication with much bandwidth fluctuation, but shorter segment lengths may cause inadequate performance due to large overhead generated through requests messages and the network delay. In order to analyze the effect of segment size in adaptive streaming content, the author of [64] created an emulation platform showing many aspects like standard Internet connections and HTTP server configuration. This was achieved by employing a standard HTTP based Web server with both persistent HTTP 1.1 compliant connections and non-persistent HTTP 1.0 compliant connections can be enabled. Additionally, they also emulated the network functionalities of an end-to-end (e.g., ADSL) Internet connection, with an additional network delay of 150 ms for their evaluation research. The performance output for different segment lengths of 1, 2, 4, 6, 10 and 15 seconds were recorded and put into a graph disclosing the behavior of the segment sizes with effective media throughput. On the other hand, using HTTP 1.0, the obtained segment length is somewhere between 5 and 8 seconds. Moreover, the accomplished media throughput using the optimal segment lengths for both configurations varies only by 50 kbps. This shows that the overall media throughput is not improved much with the increased segment length as the configured network bandwidth is varying over time. In case of applying longer segment size, the end user is unable to adjust in a fast and flexible way as compared to shorter segments and hence the effective throughput gets degraded for longer segments. On the contrary, the effect of the network round trip delay becomes greater when using shorter segment lengths. This created trouble for the non-persistent/HTTP1.0 connection results as one round-trip-time (RTT) is required for setting up the TCP connection

with the server after every segment. Additionally, the results for persistent connection/HTTP1.1 also get affected from the longer delay while using too short segments, which is visible in this connection type with a segment length of just 1 second in Fig.8. In this scenario the 50% RTT needed for requesting the segment, is getting larger and the average throughput becomes poor. On the basis of the aforementioned results of such evaluations along with a vast research analysis from various implementations, it is suggested to use DASH or HLS chunk lengths of approximately 2 to 4 seconds, which can be a good trade-off between effective encoding and adaptive streaming for varying bandwidths. Furthermore, it is seen that a segment time of 10 seconds for many media files seems to strike a reasonable balance for most broadcast content.

Throughout the literature, many areas are highlighted regarding adaptive ways of streaming and video quality. Maintaining a good video QoE is a great task for operators due to diversified video play back end devices like televisions, smartphone, desktop, and tablets. Hence, this urge to facilitate smooth video experience as per the device and network characteristics has driven the discovery of adaptive bitrate streaming (ABR). The function of adapting bitrate invokes a trading off between minimizing the chances of video halts and improving the video quality displayed to the consumer. A high bitrate causes recurrent video freezing (i.e., buffer underrun), while a low bitrate leads to degraded video quality. Bitrate adaptation was considered as a utility maximization issue and algorithm BOLA (Buffer Occupancy based Lyapunov Algorithm) [67] was devised to optimize rebuffering enabling acceptable video quality. Two key performance parameters that influence the over quality of users scrutinized: (a) average quality of playback over a stipulated time that depends on the supplied bitrates of video chunks and (b) the amount of time to wait before rebuffering. To conceptualize these metrics, the system model unfolding the BOLA algorithm is presented in a time-slotted fashion. The formulated model here increased its utility with an increase in the average bitrate whereas the event of rebuffering reduced it. The utility was projected to be defined randomly based on the user equipment and multimedia content. This methodology of BOLA disagrees with bitrate adaptation algorithms employed in general, which provides limited flexibility. The work of [67] evaluated adaptive and generic streaming with mobile vehicular environments and exposed that when bandwidth reduces, the quality adaptation technique can adequately decrease the stalling by 80% and enable a superior usage of the accessible bandwidth at the time when bandwidth enhances. Moreover, they declared HAS to be effective even in static environments, as it removes the majority of the stalling by switching the quality. With the rapid advancement of mobile video traffic usage, improving video quality of experience (QoE) is a daunting affair. HTTP Adaptive Streaming (HAS) solves this by catering different video data rates in a variable bandwidth scenario. In this direction, a HAS engine was designed for LTE network [68] to evaluate QoE.

The QoE of HAS was predicted with evolved-PSNR (ePSNR) and predicted-MOS (pMOS) for LTE networks. The proposed design encoded source video under six different data rates using a HAS module.

The QoE evaluation model of HAS engine is described in Figure 7. It consisted of three primary components: the server, the EPC and the client. Towards the server, the elementary video Sintel was encoded into diverse video rates through HAS engine. Post encoding, the video was saved in a m3u8 file in the web server. The video was also released over the website and viewers could input the URL to watch it. The Evolved Packet Core (EPC) is comprised of the LTE platform and base station eNodeB that modeled the LTE system-level test cases, and the output was collected in the form of trace files. The trial was conducted on a 10MHz channel LTE system with a velocity of 0.3km/h. All the LTE clients in the form of UEs, were watching the HAS video on their smart phone and rated the video quality from a 1 to 5 scale score. The results concluded an improved performance in this evaluation scheme with good accuracy and scalability. Authors in [69], optimized the video segments length depending on the content, so that I-frames and various quality representation borders are directed to optimum place, e.g., video cuts. Their result showed up to 30% reduction of required bit rate without any quality deprivation for H.264 adaptive streaming. This also suggests that the higher image quality encoding for a video can be performed with the same bit rate. Many of the earlier works suggested video segments of fixed length with an average duration of 2s to 10s. This long segment period enables the users to send fewer request and comparatively lower overhead. In spite of these pros, a client reacts slowly to the network changes, if encountered with longer segment period as adaptation strategies are executed towards segment edges. However, there is a work found in the HTTP/2 standard [64] that contributes a new element, a server push that balances segment length and the number of requests providing improved quality. In this approach, a server device pushes assets to the client even if the client is not asking for it. This empowers the client to use short segment period without sending additional request packets to the server and asks the server to reply a single request with many successive segments of the same type. Innovative adaptation method based on server push over HTTP/2 was introduced in [68]. It could adjust the number of segments pushed depending on the network and buffer conditions. A cost function was derived out of request cost and buffer cost to figure out the optimal number of demanded segments of a request. The recommended strategy was able to bring a balance between the number of requests and buffer stability. The type of encoder immensely influences QoE and is applicable for scalable video codecs (SVC) that inspect streaming using H.264/SVC with wavelet-based coding.

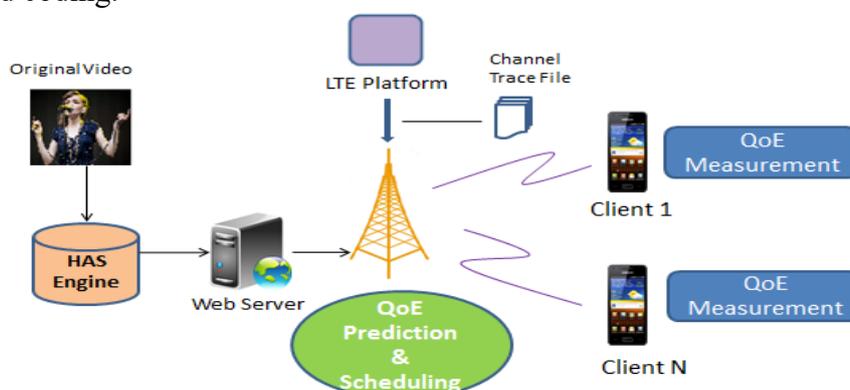


Fig. 7. Mobile QoE evaluation system

Authors of [70, 71, 72, 73, 74] introduced a new approach to encode and segment video information for adaptively streaming H.264 video. They also achieved up to 30% reduction in encoding bitrate while maintaining the same or higher quality of video. With the growth of software defined networking (SDN) technology, a dynamic SDN-based traffic framing method (DASH-SDN) was launched which utilized the benefit of long OFF time of adaptive media players and increased throughput for ongoing streaming as shown in Figure 8. DASH-SDN considered administering the bandwidth, divergence of wireless transmission characteristics in terms of quality of channel, round trip travel time, and remaining battery power. In this strategy the video players were allowed to transcend the traffic shaping restraint and initiate the streaming at higher bitrates. This was accomplished by exploiting the lengthy OFF periods of the commercial media players such that vacant bandwidth of the passive players could be temporarily assigned to potential active players over the wireless network. This opens up the avenue for players to fill out their buffers and overcome any deterioration of the video quality while performing the playback due to fluctuating wireless channel and encoding bit rates. Figure 8 illustrates the preliminary building blocks of DASH-SDN consisting on the wireless infrastructure and mobile devices. The wireless infrastructure module is responsible for carrying out several important jobs including monitoring the network to estimate the bitrate availability, inspection of flow to identify the bitrate profiles for every stream, and managing the bandwidth to distribute it among the contending media devices. Another module named mobile device, is comprised of a critical sub module such as: flow manager, which is an OpenFlow switch software that presents controller with measurement information of parameters, e.g. round trip time, in every received chunk. Apart from this, the OpenFlow manager also takes part in executing the traffic shaping function by customizing the TCP/IP packets upon exchanging the value from the mobile controller.

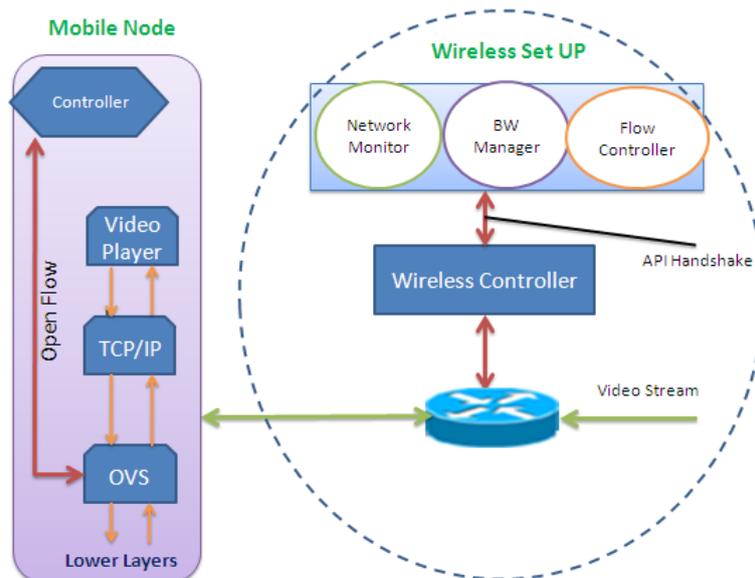


Fig. 8. Mobile QoE evaluation system

7. Content distribution in Internet

Content delivery networks (CDN) [41] are the obvious backbone of today’s Internet network in delivering a plethora of contents from one place to another. In, day to day life we interact with CDNs while reading online news, e-shopping, enjoying YouTube videos or social networking feeds. CDNs were designed to solve the underlying latency, which is defined as the time gap from the instant a user requests a web page to the moment it actually turns up on screen. This delay interval is caused by many factors like geographical distance between end client and web hosting server or specific content on the web page. A CDN store a cached version of its content in multiple geo-locations or points of presence (PoPs), which is elaborated in Figure 9. Each PoP consists of a number of caching servers present for content distribution to nearby visitors [41]. The content replication at many locations improves coverage. Over a decade ago Akamai, for the first time, conceptualized the concept of Content Delivery Networks (CDNs) [90] to assist business enterprises overcome the technical roadblocks. Then onwards, both the Web and the Akamai framework have emerged staggeringly. In the marketplace there is bewilderment in respect to the operating differences between a Cache server and a CDN server. Many researchers assumed that CDNs need a caching mechanism to distribute and deliver contents, which is a wrong notion. That is why many times, the term ‘cache’ and CDN delivery server are interchanged mistakenly. Hence, the following part illustrates this inequality to provide more clarity on it

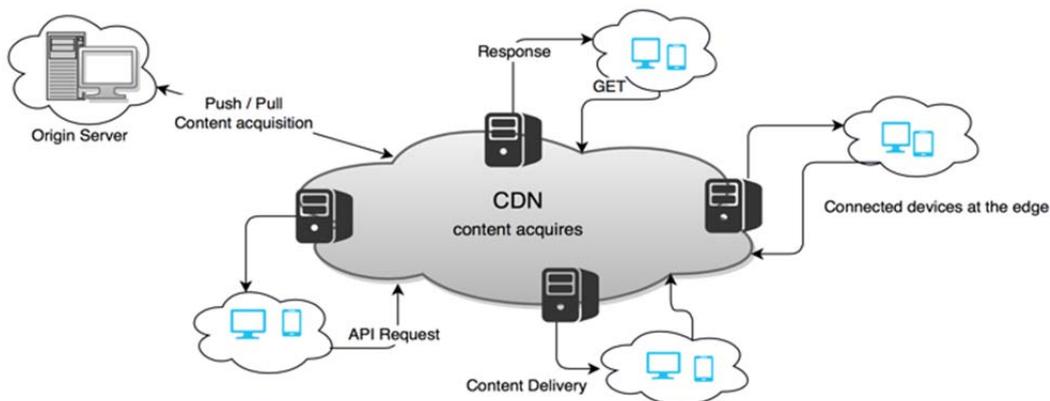


Fig. 9. CDN architecture

A. Cache Server

A Web Cache is a software [41], which temporarily stores static web objects submitted through HTTP. Caching is a narrative part of HTTP that characterizes functionalities like freshness and authorization. Caching is not a narrated portion of RTSP, RTMP streaming protocols. Any web cache in Internet service providers’ networks are normally implanted between the path of original source/server and the end users/clients. This is known as transparent caching mechanism, which caches any HTTP object over web transported through the network. Web caches in CDNs are not arranged in the intermediate path between

the original HTTP source and destination. Here, the CDN redirects the user towards the cache server. In both the cases, end users actually request information from the nearest cache server instead of original HTTP source.

B. CDN Server

A CDN server is a type of delivery node within a CDN that is responsible to hand over objects for end users [41]. It is a combination of hardware and software components. This might be a commercial server executing web services like IIS or Apache, or streaming services from Adobe, Microsoft, and Apple, which append many utilities in comparison to basic web caches features which are essential for content publishers. The CDN server can have caching features, but it's not mandatory, rather than that, it can adopt other approaches along with caching. Streaming protocols in general are not designed to be cached. In spite of the arrival of ABR and HTTP streaming, there exists still a demand for live video watching, as live video is not cacheable like the on demand one. Moreover, the bulk amount of video content delivered over Internet is on-demand, which is estimated to be around 95% of the overall online video traffic [75]. Hence, live video streams should be redirected to delivery CDN nodes by unicast or multicast methods, instead of caching them locally. A CDN server that sustains HTTP based delivery for video on demand and live streaming cannot bank only on caches for drawing in objects. The CDN server should support vital distribution technologies like pull, push, and relaying mechanism for various streaming techniques. A good example of a non-caching CDN server available today is Stream Zilla, which is running without any single cache and built upon commercial servers executing both streaming and web services. Stream Zilla does not provide website acceleration, web caching and gaming application acceleration services, it is the exclusive CDN Server in Europe that is entirely dedicated to video delivery. More than 500 content publishers over the world like Apple HLS, 3GPP mobile streaming, Microsoft Smooth Streaming, etc, are using Stream Zilla today due to its improved resource control and content distribution strategy. All the inherent benefits of CDN indicated that a cache server is not the best solution for HTTP Streaming compared to its counterpart CDN server. Most of the CDNs existing today apply two types of techniques for load balancing the huge user generated requests. The first one is DNS [41] based and the other one is known as anycast routing.

C. DNS for CDN

DNS protocol provides services in achieving “hostname to IP resolution” (and vice versa). In a DNS based method, whenever a user sends a request to the content server, the hostname gets resolved to the corresponding IP address. This user IP address is checked by the CDN and based on the implemented policies on location, uptime, and load, the IP of best-suited server will be returned. After any DNS request for a domain name (e.g.mydomain.yolasite.com) arrives at the CDN, the server dealing with DNS requests for the domain name keeps an eye to the incoming request to finalize the best set of servers. To attain this, a DNS server performs a geographic search based on the DNS resolver's IP address and retorts an IP address of the physically closest edge server.

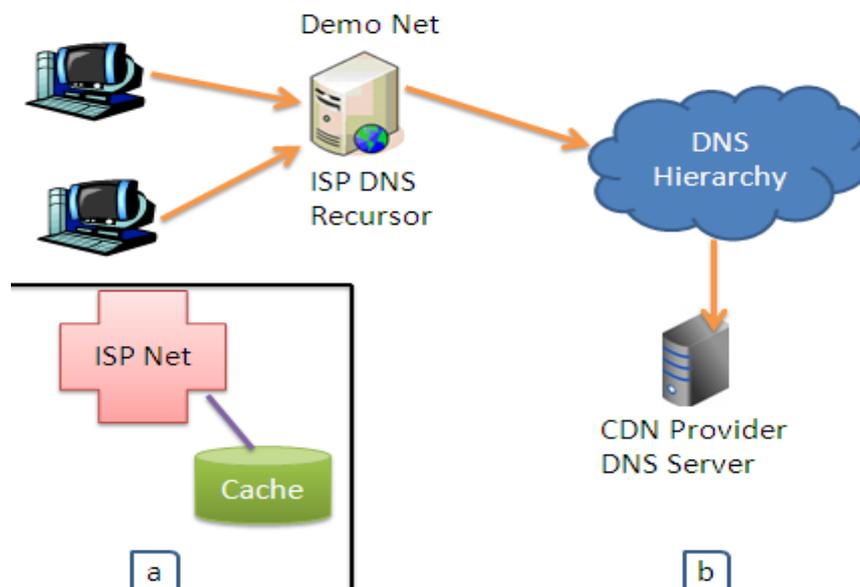


Fig.10. Flow of DNS Information

CDN features the “edge delivery” concept where all users reach to its nearest CDN entity and get served from there instead of from a faraway node. Figure 10 shows one ISP net containing a Cache or edge server deployed very near to it. Considering this as the base concept, the figure elaborates how DNS works for a CDN. A user within the “Demo” ISP is considered requesting for the domain “cdn.information.com” and the IP address.

These requests will travel to the DNS recursor of the ISP, which in turn will seek the help of authoritative DNS servers of CDN providers through the DNS network hierarchy. The arrows represent the flow of DNS information. At the end, depending on the requested DNS recursor’s IP address, authoritative DNS server will return back the IP address of the cache entity, which is closest to the network “Demo”. Many large CDN providers such as Akamai, Amazon Cloud front are based on the DNS method.

D. Anycast Routing

Anycast DNS is a traffic routing algorithm used for the speedy delivery of website content that advertises individual IP addresses on multiple nodes [76]. User requests are directed to specific nodes based on factors such as the capacity and health of your server, as well as the distance between it and the website visitor. While some of new entrants like Cloudflare rely very much on anycast routing. Figure 11 depicts the anycast routing mechanism where client1 and client2 on ISP 1 are connected to ISP 1 router., client 3 and client 4 on ISP 2 are connected to ISP 2 router and client 5 and client 6 on ISP 3 connected on ISP 3 router [77].

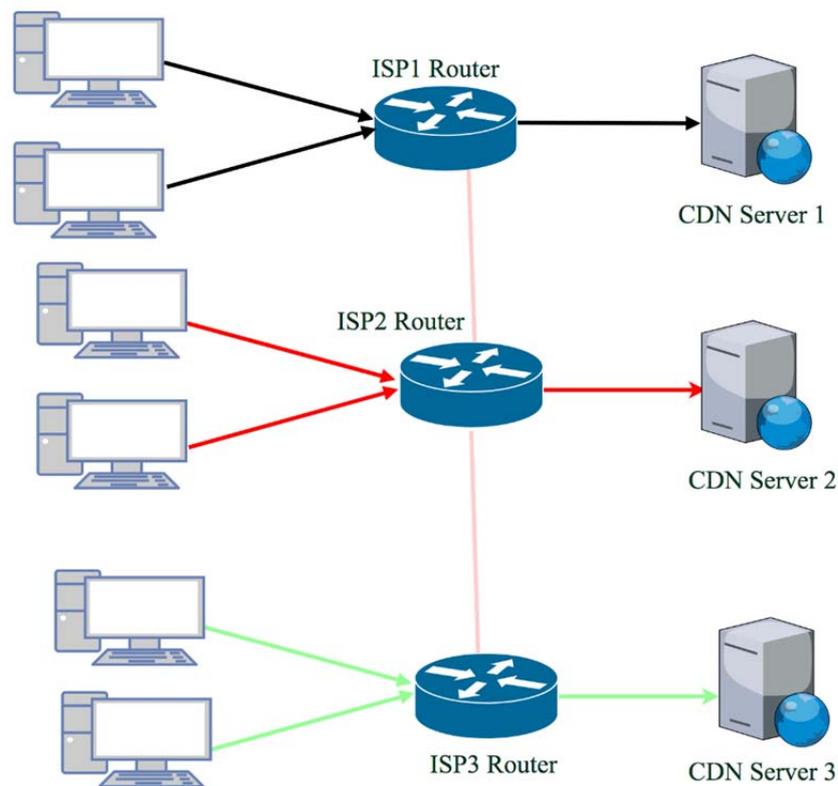


Fig.11. Anycast Routing in CDN.

8. Multimedia codecs and characteristics

A codec is a hardware or software component in multimedia technology that compresses and decompresses digital media information, such as a video or music. Basically the “codec” consists of two functionalities: encode and decode. The encoder carries out the compression (encoding) [78] job and the decoder executes the decoding or decompression. Hereafter are the popularly used codecs for video:

XviD/DivX: DivX is a commercially available codec, whereas XviD is open source software (under the GNU General Public License) that functions as a substitute to its commercial counterpart. Both of the codecs are constructed on the principle of MPEG-4 and are capable of decoding the output of each other. DivX can compress prolonged video segments into small sizes and still maintain comparatively high visual quality because it uses lossy MPEG-4 Visual compression techniques, where quality is poised against file size. Xvid can be utilized for most of the platforms and operating systems unlike DivX and its source code can be easily compiled.

MPEG-4: MPEG-4 [78] is a popular available streaming codec and it has many parts, where only MPEG-4 Part II is deployed for video coding. MPEG-4 Part II relies on video encoders such as DivX or XviD for encoding the video, while audio is typically performed in MP3 format. A generic 120-minute movie (the file without encoding) can be compressed to about

300MB using MPEG-4.

H.264: Most of the modern camcorders and digital cameras apply H.264 codec and it captures file-based devices like hard drives, SD cards, etc. It is also recognized as MPEG-4 Part 10, or MPEG-4 AVC (Advanced Video Coding). Typically, H.264 is the best suitable codec for high definition video streams. It is capable of performing both lossy and lossless compression based on user defined encoding settings, which are quality, frame rate, and target file size. H.264 depends on DivX/ XviD, X264 to encode the video and AAC or MP3 for encoding audio. H.264 is considered to be 1.5 to 2 times efficient than basic MPEG-4 compression that enables smaller compressed file sizes with seamless playback on multiple platforms.

SVC: Scalable Video Coding (MPEG-4 SVC) was designed as a modified extension over MPEG4-AVC. It provides effective scalable manifestation of video through adjustable multi-dimensional resolution adaptation. This scalable representation simplifies the relation between compression techniques and transmission/storage. The technique is compatible to different networks and end terminals providing significant improved error resilience by easy stream truncation. Many modalities like spatial and temporal scalability, SNR/quality/fidelity scalability and a hybrid one combining all of them are incorporated as salient features.

Hype of VP9: Today, the majority of people love to watch high-quality videos across multiple platforms. This creates the need of enhanced video formats that can cater improved resolution without using more bandwidth. Because of this, YouTube videos started getting encoded in VP9, the open-source codec that enables High Definition and also 2160p quality video using only 50% bandwidth as compared to other standard codecs. VP9 has gained much popularity as a highly effective compression video codec widely used across multimedia domains. In the previous year, the consumer market using YouTube had spent more than 25 billion hours watching VP9 video. Without the bandwidth saving advantage that VP9 provides, many of such YouTube videos would not have been experienced in HD. The encoder emphasizes the feature of sharpest images, and the codec utilizes asymmetrical transforms to enable the most difficult scenes to be vibrant and block-free. The very purpose of the new VP9 format to boost the aim of high quality, prompt and buffer free video. The beneficial impact of this can be observed when a user's internet connection was playing up to 480p without any buffer on YouTube and after applying VP9 the rate increased to smooth 720p. So, this new technique aids the end user with limited internet bandwidth or a costly plan. As the bitrate is reduced to almost a half, the probability of users experiencing 360p quality video without re-buffering increases.

So, all these study and analysis greatly reveals the fact that, H.264 is the standard choice of codec for high definition resolution video streaming and iOS devices as well. Besides providing high DVD quality video output by using H.264 codec, it utilizes advanced video coding (AVC) technology and Apple's advanced audio coding (AAC) to enhance old audio container formats. For Android based devices (e.g. Samsung Galaxy S4, HTC One), Google

recommends H.264 as the best option for video, having a bit rate of 500kbps and AAC-LC at 128kbps for the audio. Furthermore, MPEG4 and H.264 can also be a good option for uploading video online to sites like YouTube.

A container exclusively bundles all the video, audio, and codec files into single coordinated package [79]. Furthermore, the container often contains section info for DVD, metadata, Blu-ray movies, subtitles, and extra audio like various languages. It is mostly useless bits of information compared to video. For example, requesting an MP4 file from someone doesn't provide any necessary information without realizing how the actual video and audio were encoded. So, a container is just a place to stock up the required audio, video and the codecs for future playback. Hereafter are the most commonly used containers or file formats for video:

Flash Video (.flv, .swf): Macromedia long back introduced Flash before they were eventually taken over by Adobe in year 2005. Flash is becoming an aging container and being obsolete of late due to many restrictions in the technology, such as erroneous file handling. This resulted in global removal of flash from iOS devices from Adobe. After HTML5 standard emerged, very less Flash videos are found online, and the usage of this container is quite minimum today.

MKV: MKV is a speedily arising container that supports most of the audio or video format which makes it flexible, effective, and one of the best ways to stock music and video files. Moreover, it also supports many video, audio and subtitle files which are encoded in various other formats. MKV is widely used container today because of its error handling mechanism and many features which enable to play back even corrupted files on end user devices.

MP4: The task of uploading video to webs like YouTube, Vimeo and so on can be easily accomplished by the MP4 format. The MP4 container employs MPEG-4 or H.264 encoding [41], in addition to AAC or AC3 for audio. MP4 can also be utilized as a reference for all other multimedia files like 3GP, Motion JPEG 2000, etc.

QuickTime: QuickTime is Apple's own container format. QuickTime sometimes gets criticized because codec support (both audio and video) is limited to whatever Apple supports. This is true, but QuickTime supports a large array of codecs for audio and video. Apple is a strong proponent of H.264, so QuickTime files can contain H.264-encoded video.

The majority of the research manifests H.264 along with mp4 or MKV as worthy containers. MP4 might get the edge, as it is widely compatible with consumer devices, and the de facto standard for many streaming video sites. This is because MP4 supports multiple bits of data with the capability to serve in a narrow bandwidth situation. The suitable containers for MPEG-4 are found to be MOV, AVI, and raw TS, along with MP4 and MKV; those can be selected on the basis of end playback devices.

9. Towards adaptive streaming and challenges

Adaptive bitrate streaming is much-hyped across industry for switching the video quality in midstream [41, 79, 80, 81, 82, 83, 84, 85], without any potential re-buffering and application restarting. YouTube has gone through a paradigm shift for switching progressive downloads towards adaptive streaming [86] in its desktop player in recent years. Companies like Netflix and Hulu are also using this technique quite often to revamp their video streaming experience. But YouTube faced a few exclusive challenges to fix when it launched out its own application [87, 88]. As an example, Netflix usually commences with a lower-bitrate stream and then gradually increases, which claims approximately a minute before original HD quality appears. This method doesn't prove good for YouTube video, as they exist for a minute or two. YouTube is inclined to be more determined in streaming out higher-quality video, and then, scale down the video if required. The site also considers the phenomena that users tend to watch more than one YouTube video in a row, and keeps upgrading the bit rate throughout an entire session. All these efforts resulted in encouraging consequences for YouTube. It has achieved 20% reduced buffering since it floated adaptive streaming for its desktop player. So, they are venturing to TVs and mobile devices with this technology. Although Netflix and Hulu have been practicing this for quite long time, they focus in long-form content. Netflix initiates with a lower bitrate and keeps rising [88], and the entire operation consumes a minute or so. In YouTube, handling mostly short-form, that introductory poor quality can make people shutting down the video instantly, according to the study of Conviva, it showed that people don't wait for 2 seconds in general. Mobile users running YouTube videos will face challenges as they move from one cell tower to another tower during their watching duration. This becomes more challenging when they roam to different countries throughout the world having various average speeds. The next-generation televisions sets introduced recently are based on adaptive streaming.

It can be seen that due to the inherent benefits of HTTP-Streaming based technology, the generic standard DASH emerged, which also saves bandwidth in a large extent. In recent years, YouTube and other regular services, such as Netflix, have enforced DASH as the favored streaming technology instead of FLV (Flash Video) streaming [83, 84, 86, 87, 88]. Of late, YouTube embraced HTML5 as video playback technique, which can completely abolish Flash. The streaming giant is also applying MPEG-DASH in browsers, just like its counterpart Netflix, who have preferred this adaptive bitrate method of streaming. In [95, 96] the authors did a comparative study on the performance of three streaming providers Hulu, Netflix and YouTube on the basis of streaming characteristics, traffic redundancy and bandwidth usage. To experiment on traffic redundancy, a test was conducted considering an apple iPad device that was running the YouTube application. It was observed that YouTube was able to play back an unspecified 750 seconds of video clip when the device was exchanging information over Wi-Fi to a bottleneck link. The bottleneck link was designed to use bandwidth from a predefined list of varying capacity, which is [5; 0.5; 1; 2; 4; 5]. In this test, initially, the bandwidth was fixed to the first value from the list (i.e. 5 Mbps) and the succeeding values are selected every 2-minutes interval from the list. As a consequence of this investigation, the authors found that whenever the bandwidth is raised, YouTube sharply changes over to the next higher bitrate and drops all buffered data having inferior qualities.

For the aforementioned experiment, the authors computed a surplus traffic of 16% for YouTube, 21% for Hulu 21 % and 22.5% for Netflix revealing the fact that YouTube had least redundant traffic among the three.

Netflix concentrated on QoE metrics to further improve streaming experience for end users. One such metric is known as re-buffer rate, which defines how frequently the playback is shortly disrupted while additional data is getting downloaded from the server side, which is a measure of how often playback is temporarily interrupted while more data is downloaded from the server to refill the play out buffer at the client device. Another such parameter, bitrate, signifies the quality of the image that is viewed – too low bitrate results in a fuzzy scene on screen. A concerning relationship exist between the re-buffer rate and bitrate. In a limited capacity network, adopting too high bitrate enhances the risk of touching the bottleneck limit causing local buffer to run out and then, playback is halted to replenish the buffer. Here is the right tradeoff that needs to be taken.

The supply chain overviews in Figure 12 are the ones that Netflix follow on regular basis for maintaining good QoE. There are also other metrics through which QoE can be characterized but the potential impact of each metric on user behavior, and their possible tradeoffs need to be realized well. In order achieve so, Netflix came up with a mapping function which is able to quantify and forecast how user behavior gets affected which the changes in QoE metrics [89]. In this fashion, the interpretation of the QoE impact on user behavior enables Netflix to fine tune their logical algorithms that decides QoE and upgrade aspects that have a great impact on consumer market’s viewing and enjoyment.

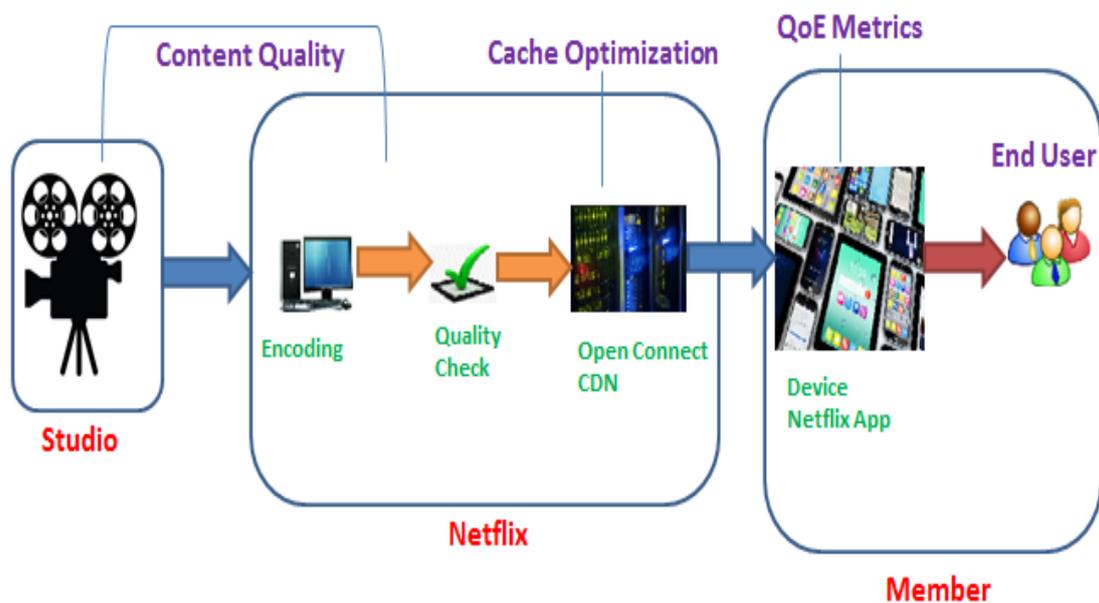


Fig. 12. The Netflix Streaming Supply Chain.

10. Conclusion and future work

In the current paper, a wide range of approaches for adaptive HTTP based streaming has been presented. We studied how multimedia technology evolved over the time and generally how streaming took birth with the advancement of Internet technology using HTTP. The high demand of adaptive technology is observed by most researchers and industrial organizations like Microsoft, Adobe, Apple, YouTube and many more, and each of them introduced their own innovative standard like MPEG-DASH, HLS, etc. Moreover, network impairments while streaming over wired and wireless networks have been our one area of focus, which has definitely provided a deep insight of all time multimedia operators. However, the upcoming technology on adaptive streaming for high-speed wireless networks like LTE-4G and LTE-A will be a major concern for operators due to high speed mobile data demand. This can be taken up as a future area of research.

References

- [1] Cisco, “Visual Networking Index”, available at: <http://bit.ly/KXDUaX>.
- [2] J. Roettgers, “Don’t touch that dial: How YouTube is bringing adaptive streaming to mobile, TVs,” 2013. [Online]. Available: <http://gigaom.com/2013/03/13/youtube-adaptive-streaming-mobile-tv>
- [3] T. Stockhammer, “Dynamic adaptive streaming over HTTP: standards and design principles,” 2011 ACM Conference on Multimedia Systems. DOI: <https://doi.org/10.1145/1943552.1943572>
- [4] ITU-R, “Methodology for the Subjective Assessment of the Quality of Television Pictures,” ITU-R Recommendation BT. 500, Jan 2012.
- [5] ITU-T, “Subjective Video Quality Assessment Methods for Multimedia Applications,” ITU-T Recommendation P.910, Apr 2008.
- [6] W. Lin and C.-C. J. Kuo, “Perceptual visual quality metrics: A survey,” *J. Vis. Commun. Image Represent.*, vol. 22, no. 4, pp. 297–312, May 2011. DOI: <https://doi.org/10.1016/j.jvcir.2011.01.005>
- [7] M. Garcia, A. Canovas, M. Edo, J. Lloret, A QoE management system for ubiquitous IPTV devices. The Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies. UBICOMM 2009. October 11-16, 2009 - Sliema, Malta. DOI: <https://doi.org/10.1109/UBICOMM.2009.31>
- [8] K. D. Singh, Y. Hadjadj-Aoul, and G. Rubino, “Quality of experience estimation for adaptive HTTP/TCP video streaming using H.264/AVC,” in *Proc. IEEE CCNC*, Las Vegas, NV, USA, 2012, pp. 127–131. DOI: <https://doi.org/10.1109/CCNC.2012.6181070>
- [9] N. Zong, Survey and gap analysis for HTTP streaming standards and implementations, Internet Engineering Task Force Network Working Group, Fremont, CA, USA, Tech. Rep. [Online]. Available: <http://tools.ietf.org/html/draft-zong-httpstreaming-gap-analysis-01>
- [10] Garcia, Miguel, Jaime Lloret, Miguel Edo, and Raquel Lacuesta. "IPTV distribution network access system using WiMAX and WLAN technologies." 4th edition of the

- UPGRADE-CN workshop on Use of P2P, GRID and agents for the development of content networks, June 09, 2009, Garching, Germany. DOI: <https://doi.org/10.1145/1552486.1552513>
- [11] Alejandro Cánovas, Miran Taha, Jaime Lloret, and Jesús Tomás. "Smart resource allocation for improving QoE in IP Multimedia Subsystems." *Journal of Network and Computer Applications*, Volume 104, 15 February 2018, Pages 107–116. DOI: <https://doi.org/10.1016/j.jnca.2017.12.020>
- [12] F. Guterl, "Design case history: Apple's Macintosh: A small team of little-known designers, challenged to produce a low-cost, exceptionally easy-to-use personal computer, turns out a technical milestone," *IEEE Spectrum*, vol. 21, no. 12, pp. 34-43, Dec. 1984. DOI: <https://doi.org/10.1109/MSPEC.1984.6370374>
- [13] A. Reilly, "Commodore Makes a Splash with Amiga," *IEEE Computer Graphics and Applications*, vol. 6, no. 3, pp. 10-14, March 1986. DOI: <https://doi.org/10.1109/MCG.1986.276626>
- [14] G. Singh, "The IBM PC: The Silicon Story," *Computer*, vol. 44, no. 8, pp. 40-45, Aug. 2011. DOI: [10.1109/MMSP.1998.738931](https://doi.org/10.1109/MMSP.1998.738931)
- [15] R. Barneva and G. M. Cortelazzo, "An extension of Macromedia DIRECTOR for multimedia virtual visits," *IEEE Second Workshop on Multimedia Signal Processing*, 1998, Redondo Beach, CA, 1998, pp. 175-180. DOI: <https://doi.org/10.1109/MMSP.1998.738931>
- [16] M. Ketterl, R. Mertens and O. Vornberger, "Vector Graphics for Web Lectures: Comparing Adobe Flash 9 and SVG," *Ninth IEEE International Symposium on Multimedia Workshops (ISMW '07)*, Beijing, 2007, pp. 389-395. DOI: <https://doi.org/10.1109/ISM.Workshops.2007.71>
- [17] R. Vetter, C. Ward and S. Shapiro, "Using color and text in multimedia projections," *IEEE MultiMedia*, vol. 2, no. 4, pp. 46-54, Winter 1995. DOI: <https://doi.org/10.1109/93.482295>
- [18] Sonam and M. Kumar, "Implementation of MD algorithm for text extraction from video," *2013 Nirma University International Conference on Engineering (NUiCONE)*, Ahmedabad, 2013, pp. 1-5. DOI: <https://doi.org/10.1109/NUiCONE.2013.6780069>
- [19] H. Shim, B. Kang and K. Kwag, "Web2Animation - Automatic Generation of 3D Animation from the Web Text," *IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies (WI-IAT '09)*, Milan, Italy, 2009, pp. 596-601. DOI: <https://doi.org/10.1109/WI-IAT.2009.101>
- [20] A. Boev, R. Bregovic, D. Damyanov and A. Gotchev, "Anti-aliasing filtering of 2D images for multi-view auto-stereoscopic displays," *2009 International Workshop on Local and Non-Local Approximation in Image Processing*, Tuusula, Finland, 2009, pp. 87-97. DOI: <https://doi.org/10.1109/LNLA.2009.5278399>
- [21] RP 2072:2014 - SMPTE Recommended Practice - Emphasis and Preferred Sampling Rate for AES/EBU Digital Audio in Television," *SMPTE RP 2072:2014*, vol., no., pp.1-5, Jan. 14 2014 DOI: <https://doi.org/10.5594/SMPTE.RP2072.2014>
- [22] T. Sagisaka and T. Munakata, "Two types of sound tool for editing speech signal: sound cutter and symbolic sound editor," *IEEE International Conference on Multimedia Computing and Systems 1999*, Florence, Italy, June 7-11, 1999. pp. 1020-1021 vol.2. DOI: <https://doi.org/10.1109/MMCS.1999.778642>

- [23] K. T. Wang and O. T. C. Chen, "A High-performance Video Format Conversion System For MPEG-4," 1997 International Symposium on VLSI Technology, Systems, and Applications, Taipei, Taiwan, 1997, pp. 93-97. DOI: <https://doi.org/10.1109/VTSA.1997.614736>
- [24] Y. Miura and M. Katsumoto, "An overview of a real-time contents edition system for MPEG-4," 2003 IEEE Pacific Rim Conference on Communications, Computers and signal Processing, PACRIM 2003, Victoria, BC, Canada, 2003, pp. 81-85 vol.1. DOI: <https://doi.org/10.1109/PACRIM.2003.1235724>
- [25] G. D. Flinchbaugh, P. L. Martinez and D. S. Rouse, "Network capabilities in support of multimedia applications," IEEE Global Telecommunications Conference, 1990, and Exhibition. 'Communications: Connecting the Future', GLOBECOM '90, San Diego, CA, 1990, pp. 322-326 vol.1. DOI: <https://doi.org/10.1109/GLOCOM.1990.116530>
- [26] Shiyu Li, M. Okuda and S. Takahashi, "Embedded Key-Frame Extraction for CGAnimation by Frame Decimation," 2005 IEEE International Conference on Multimedia and Expo, Amsterdam, 2005, pp. 1404-1407. DOI: <https://doi.org/10.1109/ICME.2005.1521693>
- [27] S. S. Ceballos and J. A. Lopez Leyva, "An optimized collaborative platform for educational cloud computing in application and presentation layers of OSI model," 2015 International Conference on Computing Systems and Telematics (ICCSAT 2015), 28-30 October 2015, Xalapa, Mexico, pp. 1-5. DOI: <https://doi.org/10.1109/ICCSAT.2015.7362922>
- [28] S. Khan, J. Saastamoinen, M. Majanen, J. Huusko and J. Nurmi, "Analyzing transport and MAC layer in system-level performance simulation," 2011 International Symposium on System on Chip (SoC), 31 October - 2 November 2011, Tampere, Finland, pp. 1-8. DOI: <https://doi.org/10.1109/ISSOC.2011.6089223>
- [29] IEEE Standard for Ethernet," in IEEE Std 802.3-2015 (Revision of IEEE Std 802.3-2012), vol., no., pp.1-4017, March 4 2016. DOI: <https://doi.org/10.1109/IEEESTD.2016.7428776>
- [30] D. Chu, C. h. Jiang, Z. b. Hao and W. Jiang, "The Design and Implementation of Video Surveillance System Based on H.264, SIP, RTP/RTCP and RTSP," Sixth International Symposium on Computational Intelligence and Design (ISCID 2013), Hangzhou, China, 2013, pp. 39-43. DOI: <https://doi.org/10.1109/ISCID.2013.124>
- [31] M. Sivabalakrishnan and D. Manjula, "Analysis of decision feedback using RTCP for multimedia streaming over 3G," International Conference on Computer and Communication Engineering (ICCCE 2008), Kuala Lumpur, Malaysia. 13-15 May 2008, pp. 1023-1026. DOI: <https://doi.org/10.1109/ICCCE.2008.4580763>
- [32] V. Q. Do, N. Binh, S. T. Chung and S. Cho, "Design and implementation of an embedded multimedia live streaming decoder system," 2014 International Conference on Advanced Technologies for Communications (ATC 2014), 15-17 October 2014, Hanoi, Vietnam, pp. 377-382. DOI: <https://doi.org/10.1109/ATC.2014.7043415>.
- [33] Mauricio Arregoces; Maurizio Portolani (2003). Data Center Fundamentals. Cisco Press. p. 460.
- [34] Z. Lin, C. Yingmei, L. Zhen and H. Zhuzhen, "An Improved Video Monitoring System Based on RSVP Protocol," International Conference on Intelligent Transportation, Big Data

and Smart City (ICITBS 2015), 19-20 December 2015, Halong Bay, Vietnam, pp. 94-97. DOI: <https://doi.org/10.1109/ICITBS.2015.29>

[35] F. Pana and F. Put, "RSVP Bundle Extension in OPNET Modeler: Implementation and Evaluation," *2013 European Modelling Symposium (EMS)*, 20-22 November 2013, Manchester, United Kingdom., pp. 623-628. DOI: <https://doi.org/10.1109/EMS.2013.104>

[36] R. Ando, T. Murase and M. Oguchi, "Characteristics of QoS-guaranteed TCP on real mobile terminal in wireless LAN," *2011 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*, Naples, FL, 2011, pp. 1-6. DOI: <https://doi.org/10.1109/CQR.2011.5996094>

[37] R. Rajaboina, P. C. Reddy and R. A. Kumar, "Performance comparison of TCP, UDP and TFRC in static wireless environment," *Electronics and Communication Systems (ICECS), 2015 2nd International Conference on*, Coimbatore, India, 26-27 Feb. 2015, pp. 206-212. DOI: <https://doi.org/10.1109/ECS.2015.7124893>

[38] Jong Wook Nam, Kam Yong Kim, Kee Sung Cho, Hwa Suk Kim and Seong Gon Choi, "Design of IP sharing device for multimedia streaming using UDP datagram switching mechanism," 2010 The 12th International Conference on, Phoenix Park Advanced Communication Technology (ICACT), Phoenix Park, South Korea, 7-10 Feb. 2010, pp. 81-86.

[39] D. Gregorczyk, "WS-Eventing SOAP-over-UDP Multicast Extension," *Web Services (ICWS), IEEE International Conference on*, Washington, DC, USA, DC, 4-9 July 2011, pp. 660-665. DOI: <https://doi.org/10.1109/ICWS.2011.92>

[40] K. J. Ma, M. Li, A. Huang and R. Bartos, "Video Rate Adaptation in Mobile Devices via HTTP Progressive Download of Stitched Media Files," in *IEEE Communications Letters*, vol. 15, no. 3, pp. 320-322, March 2011. DOI: <https://doi.org/10.1109/LCOMM.2011.012511.102044>

[41] Miran Taha. "A Novel CDN Testbed for Fast Deploying HTTP Adaptive Video Streaming." *Proceedings of the 9th EAI International Conference on Mobile Multimedia Communications*, ACM, Xi'an, China, pp. 65-71, June 2016.

[42] T. C. Thang, H. T. Le, A. T. Pham and Y. M. Ro, "An Evaluation of Bitrate Adaptation Methods for HTTP Live Streaming," in *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 4, pp. 693-705, April 2014. <https://doi.org/10.1109/JSAC.2014.140403>

[43] "HDS, HLS, HSS — Adaptive HTTP Streaming Demystified" July 11, 2013. [Online], Available on:

<https://www.verizondigitalmedia.com/blog/2013/07/hds-hls-hss-adaptive-http-streaming>

[44] Christopher Mueller, "MPEG-DASH vs. Apple HLS vs. Microsoft Smooth Streaming vs Adobe HDS", 29 March 2015, [Online], Available on: <https://bitmovin.com/mpeg-dash-vs-apple-hls-vs-microsoft-smooth-streaming-vs-adobe-hds/>

[45] Alejandro Canovas, Fernando Boronat, Carlos Turro, and Jaime Lloret. "Multicast TV over WLAN in a University campus network." In *Networking and Services, 2009. ICNS'09. Fifth International Conference on*, Valencia, Spain, pp. 57-62, 20-25 April 2009. <https://doi.org/10.1109/ICNS.2009.105>

[46] Jaime Lloret, Alejandro Canovas, Joel JPC Rodrigues, and Kai Lin. "A network algorithm for 3D/2D IPTV distribution using WiMAX and WLAN technologies." *Multimedia*

- tools and applications, Vol. 67, no. 1, 2013, pp. 7-30.
<https://doi.org/10.1007/s11042-011-0929-4>
- [47] M. Luby, "Best practices for mobile broadcast delivery and playback of multimedia content," IEEE international Symposium on Broadband Multimedia Systems and Broadcasting, Seoul, 27-29 June 2012, pp. 1-7. <https://doi.org/10.1109/BMSB.2012.6264233>
- [48] <http://unboltingbinary.in/networking-and-i-s/basics-of-networkingcontinuation/>
- [49] Abhik Majumdar, Daniel Grobe Sachs, Igor V. Kozintsev, "Multicast and Unicast Real-Time Video Streaming Over Wireless LANs" IEEE transactions on circuits and systems for video technology, vol. 12, no. 6, June 2002. <https://doi.org/10.1109/TCSVT.2002.800315>
- [50] T. Polishchuk, M. Karl, T. Herfet and A. Gurtov, "Scalable architecture for multimedia multicast Internet applications," World of Wireless, Mobile and Multimedia Networks (WoWMoM), IEEE International Symposium on a, San Francisco, CA, 25-28 June 2012, pp. 1-6. <https://doi.org/10.1109/WoWMoM.2012.6263720>
- [51] Samana Zehra, Adeel Akram, "Collaborative Systems for Video Streaming in Heterogeneous Wireless Networks with Mobile Peers" Journal of Basic and Applied Scientific Research, Vol. 4, no. 6, pp. 200-214, 2014
- [52] Santhosha Rao, Kumara Shama, "Cross Layer Protocols for Multimedia Transmission in wireless Networks", International Journal of Computer Science & Engineering Survey (IJCSES) Vol.3, No.3, June 2012. [Online], Available on: goo.gl/1T9CSy
- [53] Raj Jain, "Video Streaming over Mobile Networks: Issues, Challenges, and Opportunities", Keynote at National Workshop on Wireless Communications and Mobile Networks, Saudi Arabia, March 26, 2011. [Online], Available on: goo.gl/pQieSi
- [54] Khanista Namee, "Performance Evaluation of Multimedia Application QoS over wireless and wired IPV6 networks", International conference on Communication Software and Networks, Macau, China, 27-28 Feb. 2009. <https://doi.org/10.1109/ICCSN.2009.158>
- [55] D.Poornima, S.Vijayashaarathi, "A Review on Challenging Issues of Video Streaming Over Heterogeneous Wireless Networks", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 4, Issue 3, March 2015. DOI: <https://doi.org/10.15662/ijareeie.2015.0403048>
- [56] Li, M., Claypool, M., and Kinicki, R. 2009. Payout buffer and rate optimization for streaming over IEEE 802.11 wireless networks. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM). Vol. 5, No. 3, Article 26, August 2009, 25 pages. DOI: <http://doi.acm.org/10.1145/1556134.1556143>
- [57] J. Sung; M. Kim; K. Lim; J. K. K. Rhee, "Efficient Cache Placement Strategy in Two-Tier Wireless Content Delivery Network," in IEEE Transactions on Multimedia, vol. PP, no.99, pp.1-1. DOI: <https://doi.org/10.1109/TMM.2016.2543658>.
- [58] L. Zhou, D. Wu, B. Zheng and M. Guizani, "Joint physical-application layer security for wireless multimedia delivery," in IEEE Communications Magazine, vol. 52, no. 3, pp. 66-72, March 2014. DOI: <https://doi.org/10.1109/MCOM.2014.6766087>
- [59] I. Hofmann, N. Färber and H. Fuchs, "A study of network performance with application to Adaptive HTTP Streaming," Broadband Multimedia Systems and Broadcasting (BMSB), 2011 IEEE International Symposium on, Nuremberg, Germany, 8-10 June 2011, pp. 1-6. DOI: <https://doi.org/10.1109/BMSB.2011.5954886>.

- [60] T. Debnath, N. Cranley and M. Davis, "Experimental Comparison of Wired versus Wireless Video Streaming over IEEE 802.11b WLANs," Irish Signals and Systems Conference, 2006. IET, Dublin, Ireland, 28-30 June 2006, pp. 235-240.
- [61] S. Amir, Kyungtae Kim and Young-June Choi, "Enterprise solution of bandwidth optimization for multimedia transmission, " 2012 Fourth International Conference on Ubiquitous and Future Networks (ICUFN), Phuket, 2012, pp. 423-427. DOI: <https://doi.org/10.1109/ICUFN.2012.6261742>
- [62] D. Bulira and K. Walkowiak, "Voice and video streaming in wireless computer networks - evaluation of network delays," Future Internet Communications (BCFIC), IEEE, 2nd Baltic Congress on, Vilnius, 25-27 April 2012, pp. 156-161. DOI: <https://doi.org/10.1109/BCFIC.2012.6217996>
- [63] Michael Seufert, Sebastian Egger, Martin Slanina, "A Survey on Quality of Experience of HTTP Adaptive Streaming", IEEE COMMUNICATION SURVEYS & TUTORIALS, VOL. 17, No. 1, FIRST QUARTER 2015, pp. 469-492. DOI: <https://doi.org/10.1109/COMST.2014.2360940>
- [64] Laura García, Jaime Lloret, Carlos Turro, Miran Taha. "QoE assesment of MPEG-DASH in polimedia eLearning System." 5th International Conference on Computing, Communications and Informatics (ICACCI 2016), Jaipur, India, September 21-24, 2016. DOI: <https://doi.org/10.1109/ICACCI.2016.7732194>
- [65] Le Feuvre, Jean, J. M. Thiesse, Matthieu Parmentier, Mickaël Raulet, and Christophe Daguët. "Ultra high definition HEVC DASH data set." In Proceedings of the 5th ACM Multimedia Systems Conference, Singapore, pp. 7-12., March 19 - 19, 2014. DOI: <https://doi.org/10.1145/2910017.2910625>
- [66] 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., Budapest, Hungary pp. 3629-3633. IEEE, 9-13 June 2013. DOI: <https://doi.org/10.1109/ICC.2013.6655116>
- [67] Kevin Spiteri, Rahul Uргаonkar , Ramesh K. Sitaraman, "BOLA: Near-Optimal Bitrate Adaptation for Online Videos", IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications, IEEE, 2016, San Francisco, CA, USA, DOI: <https://doi.org/10.1109/INFOCOM.2016.7524428>
- [68] Fei Wang, Liang Chen, Théogène Habineza, "Poster: Quality of Experience Prediction of HTTP Adaptive Streaming Traffic in LTE System", 2014 9th International Conference on Communications and Networking in China (CHINACOM), Maoming, China, 14-16 Aug. 2014, DOI: <https://doi.org/10.1109/CHINACOM.2014.7054391>
- [69] V. Adzic, H. Kalva, and B. Furht, "Optimizing video encoding for adaptive streaming over HTTP," IEEE Trans. Consum. Electron., vol. 58, no. 2, pp. 397–403, May 2012. DOI: <https://doi.org/10.1109/TCE.2012.6227439>
- [70] Miran Taha, Laura Garcia, Jose M. Jimenez, and Jaime Lloret. "SDN-based throughput allocation in wireless networks for heterogeneous adaptive video streaming applications." IEEE International Wireless Communications and Mobile Computing Conference (IWCMC), 2017 13th, Valencia, Spain, pp. 963-968, May 2017. DOI:

<https://doi.org/10.1109/IWCMC.2017.7986416>

[71] Yao J, Kanhere SS, Hossain I, Hassan M. Empirical evaluation of HTTP adaptive streaming under vehicular mobility. In International Conference on Research in Networking, Springer, Berlin, Heidelberg, 2011 May 9, pp. 92-105.

[72] Cetinkaya, Cihat, Erdem Karayer, Muge Sayit, and Cornelius Hellge. "SDN for segment based flow routing of DASH." In Consumer Electronics–Berlin (ICCE-Berlin), 2014 IEEE Fourth International Conference on, pp. 74-77. IEEE, 2014.

DOI: <https://doi.org/10.1109/ICCE-Berlin.2014.7034284>

[73] Kleinrouweler, Jan Willem, Sergio Cabrero, and Pablo Cesar. "An SDN Architecture for Privacy-Friendly Network-Assisted DASH." ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), Vol. 13, no. 3s, 2017, pp. 44. DOI: <https://doi.org/10.1145/3092838>

[74] X. Zhu, Z. Li, R. Pan, J. Gahm, and H. Hu, "Fixing multi-client oscillations in HTTP-based adaptive streaming: A control theoretic approach," in Proc. IEEE 15th Int. Workshop MMSP, Pula, Italy, 2013, pp. 230–235.

DOI: <https://doi.org/10.1109/MMSP.2013.6659293>

[75] [http://www.streamingmedia.com/Articles/Editorial/What-Is-.../What-is-a-Content-Delivery-Network-\(CDN\)-74458.aspx](http://www.streamingmedia.com/Articles/Editorial/What-Is-.../What-is-a-Content-Delivery-Network-(CDN)-74458.aspx)

[76] Li, Xiang-Yang, Symeon Papavassiliou, and Stefan Ruehrup, eds. Ad-hoc, Mobile, and Wireless Networks: 11th International Conference, ADHOC-NOW 2012, Belgrade, Serbia, July 9-11, 2012. Proceedings. Vol. 7363. Springer, 2012.

[77] Fei Hu. "Network Innovation through OpenFlow and SDN: Principles and Design". CRC Press; 2014 Feb 18.

[78] Aree Ali Mohammed and Miran Taha Abdullah. "Compressed medical image transfer in frequency domain." International Journal of Image Processing (IJIP), Vol. 5, no. 4, 2011, 371. DOI: <https://doi.org/10.1.1.224.9608>

[79] Peter Ackeret,. "Container for video and sound recording media." U.S. Patent 4,747,484 A, issued May 31, 1988.

[80] Tian, Guibin, and Yong Liu. "Towards agile and smooth video adaptation in dynamic HTTP streaming." In Proceedings of the 8th international conference on Emerging networking experiments and technologies, ACM. Nice, France, pp. 109-120., 2012.

DOI: <https://doi.org/10.1145/2413176.2413190>

[81] Zliobaite, Indre, Albert Bifet, Mohamed Gaber, Bogdan Gabrys, Joao Gama, Leandro Minku, and Katarzyna Musial. "Next challenges for adaptive learning systems." ACM SIGKDD Explorations Newsletter, Vol. 14, no. 1, 2012, pp. 48-55.

[82] Agarwal, Vikash, and Reza Rejaie. "Adaptive multisource streaming in heterogeneous peer-to-peer networks." In Proc. SPIE, vol. 5680, pp. 17 January 2005. DOI: <https://doi.org/10.1117/12.587465>

[83] Javier Añorga, Saioa Arrizabalaga, Beatriz Sedano, Maykel Alonso-Arce, and Jaizki Mendizabal, "YouTube's DASH implementation analysis", Recent Advances in Communications, IEEE 38th Conference on, Sydney, NSW, Australia, 21-24 Oct. 2013.

DOI: <https://doi.org/10.1109/LCN.2013.6761273>

- [84] A. Mansy, M. Ammar, J. Chandrashekar, and A. Sheth, "Characterizing Client Behavior of Commercial Mobile Video Streaming Services," in Proceedings of Workshop on Mobile Video Delivery, MoViD'14, Singapore, 2014. DOI: <https://doi.org/10.1145/2579465.2579469>
- [85] Nirmal Govind, "Optimizing the Netflix Streaming Experience with Data Science," [Online]: Available on, <http://techblog.netflix.com/2014/06/optimizing-netflix-streaming-experience.html>
- [86] Shane Alcock and Richard Nelson, "Application flow control in YouTube video streams," SIGCOMM Comput. Commun. Rev., vol. 41, no. 2, Pp. 24-30. Apr. 2011. DOI: <https://doi.org/10.1145/1971162.1971166>
- [87] P. Ameigeiras, J. J. Ramos-Munoz, J. Navarro-Ortiz and J. M. Lopez-Soler, "Analysis and modelling of YouTube traffic," Transactions on Emerging Telecommunications Technologies, vol. 23, pp. 360-377, 26 June 2012. DOI: <https://doi.org/10.1002/ett.2546>
- [88] V. K. Adhikari et al., "Unreeling netflix: Understanding and improving multi-CDN movie delivery," INFOCOM, 2012 Proceedings IEEE, Orlando, FL, 2012, pp.1620-1628. DOI: <https://doi.org/10.1109/INFCOM.2012.6195531>
- [89] J. Martin, Yunhui Fu, N. Wourms and T. Shaw, "Characterizing Netflix bandwidth consumption," 2013 IEEE 10th Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, 2013, pp. 230-235. DOI: <https://doi.org/10.1109/CCNC.2013.6488451>

Copyright Disclaimer

Copyright reserved by the author(s).

This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).