NTU Journal of Engineering and Technology (2023) 2 (1) : 23-38
DOI: https://doi.org/10.56286/ntujet.v2i1

IRAQI Academic Scientific Journals

P-ISSN: 2788-9971 E-ISSN: 2788-998X

NTU Journal of Engineering and Technology

Available online at: https://journals.ntu.edu.iq/index.php/NTU-JET/index

A Survey of Streaming Protocols for Video Transmission

1st Huthaifa L. Mohamed [1], 2nd Ahlam Fadhil Mahmood [2]
1. Department of Computer Engineering, University of Mosul, Mosul, Iraq, huthaifa.enp165@student.uomosul.edu.iq
2. Department of Computer Engineering, University of Mosul, Mosul, Iraq, ahlam.mahmood@uomosul.edu.iq

Article Informations

Received: 20-10- 2022,
Accepted: 16-02-2023,
Published online: 15-03-2023

Corresponding author:
Name: Huthaifa L. Mohamed
Affiliation : University of Mosul
Email:
huthaifa.enp165@student.uomosul.

Key Words:
RTP protocol,
DASH,
MMT protocol,
SCTP protocol,

ABSTRACT

Human life has always relied heavily on communication. Communication through the Internet has advanced significantly in recent years, and activities such as live video streaming are now commonplace. Streaming protocols, which are always growing and maturing, are utilized to guarantee video streaming is quick and smooth. Many streaming protocols are already available, and many people find it difficult to choose one that best meets their needs. The main purpose of this review is to study the protocols, methods and latest standards that have been proposed in the literature to improve the functionality and quality of video content in multipath and multi-network overlay networks. Although multipath is a more general term, for the purposes of this study, we describe it as improving a network routing strategy by using multiple paths that do not always completely disjoint. The many benefits of multipath include load balancing, link durability, reliability, and increased perceived throughput. In contrast, multihoming allows multiple network interfaces to be used when connecting to the Internet to improve performance, stability, and fault tolerance. This paper presents a review of studies of several available streaming protocols and supporting the multipath video streaming, as well as their main characteristics along with their performance, based on several criteria to demonstrate the validity of choosing one of them.

Introduction

Video streaming has dominated Internet traffic in the last decade. The worldwide video streaming industry is expected to reach to 321.5$ billion by 2030 [1]. In 2021, 71% of Internet traffic will be video [2]. Researchers and scientists struggle to meet customer needs for Quality of Service and Experience despite breakthroughs in video compression, computing power, and bandwidth for quality of service (QoS) and quality of experience (QoE). Clients anticipate a satisfactory QoE (customer-focused performance assessment) while watching videos. QoE is about the end-user's pleasure while viewing video broadcasts. Mok et al. [3] estimate QoE of HTTP-based Video Streaming grounded on three levels of QoS, (from user, application and network perspectives). QoS includes packet loss, bitrate, throughput, availability, transmission delay, jitter, goodput, latency, error rates, and uptime/downtime probability.

Many video streaming protocols have been created in recent years, still, most have obstacles or constraints for Internet deployment. Real Time Protocol/User Datagram Protocol (RTP/UDP) is a transport layer application layer protocol that provides cutting-edge multimedia capabilities like multicast and employs the UDP connectionless protocol at the application layer. However, because RTP/UDP is seen to be less secure than connection-oriented TCP, most firewalls reject its packets. TCP delivers dependable connections, but it's not built to handle multimedia QoS. Most video streaming traffic is based on maximum Internet access, which does not allow QoS routing due to distributed system management and commercial relationships with network providers. Most network layer routing methods deliver content as efficiently as possible without guarantees against data loss or quality degradation. The features of real-time video streaming applications at the time were not taken into account by the inventors of the routing and transport layer protocols [4].

Recently, improvements to the existing standards of dynamic adaptive streaming over HTTP (DASH), also known as MPEG-DASH, were made in response to the pressing requirement to deliver satisfactory QoS and QoE over the Internet [5]. DASH works on HTTP web servers, including Apple's HLS [7][6]. MPEG-DASH video streams are encoded at various bitrates. Depending on the client's reception settings, DASH chooses the best quality to be transmitted.

DASH can thereby minimize re-buffering and adjust video quality in accordance with network conditions. emissions that are harmful to the environment, and the use of solar energy removes that pollution [3].

Due to a number of benefits, multihomed overlay networks have been more popular over the past ten years. Researchers have attempted to construct multihomed overlay networks at the network, transport, and application levels of the protocol stack. An overlay network is a virtual network built on top of another network without additional hardware, typically using software to create network abstraction layers that can run virtualized network layers to deliver or support new services. The purpose of an overlay network is to add missing features or improve performance and security without completely redesigning the network. Overlay networks include virtual private networks (VPNs), content delivery networks (CDNs), and voice over IP (VoIP). Video footage may be obtained over numerous Internet routes from different places. TCP's inability to handle many interfaces led to the development of Multipath TCP (MPTCP) [4] and Stream Control Transmission Protocol (SCTP). Multipath TCP (MPTCP) uses numerous pathways to connect TCP connections, whereas SCTP uses multihoming as a redundant service when a network interface is inaccessible [5].

Figure 1 categorizes multipath wireless video streaming by protocols. The diagram shows which network components (any combination of client, server, and network) need to be changed in order for multipath transmission to work. Since they are compatible with network architecture, the most adaptable solutions require client-side modification. No server or network modifications are needed. Other methods need server-side or client-side changes. Most problems need network infrastructure changes. The papers will be classified in the research taking into account two main aspects. Real Time Protocol (RTP), Dynamic Adaptive Streaming over HTTP (DASH), and MPEG Media Transport (MMT) are the methods to the application layer's protocols that are discussed in the first part. The second aspect of the research concerns transport layer approaches in addition to scanning protocols related to video streaming: User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Multipath TCP (MPTCP), and Stream Control Transmission Protocol (SCTP). The last part is the conclusion.

Application Layer Approaches

Approaches to streaming video that concentrates on the application layer have the benefit of having access to information about the player's buffer and pertinent video material, such as frame priority and coding dependencies. Multipath method uses application-specific data to develop video streaming scheduling techniques. No lower layer protocol changes are required. These methods sometimes involve video software adjustments. In application layer implementations, a sequence number is utilized for missing detection, increasing protocol overhead. To make informed packet scheduling choices, the application must estimate network route performance using TCP congestion control data or application-specific probes [8]. RTP, DASH, and MMT-based works. The Real-Time Transport Protocol (RTP) [23], first released by the IETF in 1992, was the first widely used video streaming protocol.

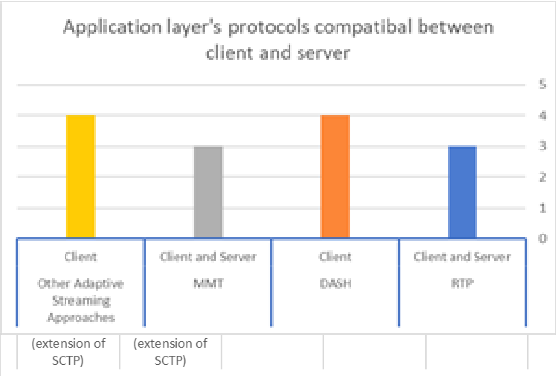

Figure 1: Surveyed Works' Classification Using Protocol Layers [5]

It is a unidirectional real-time video streaming protocol based on UDP. The MPEG-2 Transport System (MPEG-2 TS) is a well-known and widely used video streaming protocol [25]. Since 1957, It is frequently utilized in Internet streaming, mobile broadcasting, and digital television. The protocol has also been incorporated into a number of standards including Internet Protocol Television (IPTV), Terrestrial Digital Multimedia Broadcasting (T-DMB), Portable Digital Video Broadcasting (DVBH), and Advanced Television Systems Committee (ATSC) [26]. The next protocol is the Real-Time Messaging Protocol (RTMP) [28]. It is a proprietary protocol owned by Adobe that was first developed by Macromedia and standardized in 2002. For two-way video streaming, the RTMP protocol built on TCP is employed. Multiplexing is a feature of this protocol, although it necessitates the Flash Player plugin. To be played with Flash Player, all video and audio files must be provided in Small Web Format (SWF) [29].

The next highlighted event in the timeline is not a protocol, but rather a video streaming technique developed in 2006 and widely incorporated into later streaming protocols. Move Networks has created Adaptive Bit Rate (ABR) streaming that uses HTTP [31] and includes several modifications, described in succession in [40, [35]. Video sequences are divided into short fragments and stored on the server with different resolutions (bit rates). The client-side streaming logic is responsible for selecting the appropriate segment based on various factors (eg, bandwidth availability and media playback situations [35]). In the absence of such adaptable services, if just one bitrate video is accessible, a smooth video will be produced using less network capacity. On the other side, the delay will occur if the video data rate is larger than the available network capacity. The simplicity of deployment in the existing Internet infrastructure is one benefit solutions for HTTP-based video streaming.

MPEG-DASH (MPEG Dynamic Adaptive Streaming over HTTP) [43] created in 2011 and became a unified, codec-independent standard because all of these protocols were exclusive and unworkable. Because of its adaptable distribution and codec independence, MPEG-DASH is a popular protocol used by content producers [44]. For instance, MPEG-DASH and HTML are now the main streaming technologies utilized by Netflix and YouTube [44]. The fact that DASH allows both multiplexed and non-multiplexed encoded material provides an additional benefit. The protocol does, however, have significant shortcomings. Because the server needs to wait for movie fragments or the full film to finish downloading before transmitting, DASH, for instance, is unable to enable low latency delivery [24]. Additionally, incorrect bandwidth estimate causes numerous switches, hangs, and poor quality of service (QoE) for DASH, particularly on mobile networks [39]. MPEG Media Transport (MMT) is the following protocol [46]. Due to the requirement for Internet technologies like IP and HTML for solutions for Internet video streaming, and in consideration of the most current developments in media distribution, it was standardized by MPEG in 2014 [24]. Additionally, this protocol supports the HEVC video codec and UHD quality.

The MMT upgrade for mobile was released in 2015 and describes multipath capability, which has already been incorporated into the protocol and standardized [47]. In order to do this, multipath has only been investigated in a few of the aforementioned video streaming protocols, not in all of them, especially since proprietary protocols have not been investigated because of their closed and incompatible architecture.

Real-Time protocol (RTP)

The Real-time Transport Protocol (RTP) was introduced in 1992 [23], modified in RFC 1889 [33][24], and eventually superseded by RFC 3550 [34]. The most recent protocol spec is RFC 8108 [35], which was released in 2017. RTP is a transport protocol for the application layer that offers end-to-end network operations to allow interactive, live, and on-demand multimedia applications. Next, highlight more protocol characteristics before exploring how multipath based on RTP works.

RTP may send data through TCP or SCTP even though it is designed to work over UDP. RTP's capacity to be used with RTP Control Protocol (RTCP) for regularly provide monitored data and QoS settings is another advantage. Another protocol that RTP may be used with is the Real-time Streaming Protocol (RTSP) [36], which regulates the playback of multimedia. Lack of congestion control and unfair treatment of other flows are two major issues with RTP operating over UDP. Additionally, there is no assurance of dependable delivery, and a means for safeguarding high-priority frames is required (I-frames). Additionally, Since RTP builds at the multimedia session level and the receiver reports per multimedia (video or audio) flow, it is challenging to modify RTP to accommodate multipath streaming [2].

RTP has been enhanced to provide multi-channel video streaming using Real-time Multi-Stream Transport Protocol (MRTP), Multi-Path RTP (MPRTP), and Application Layer Relay-based Real-Time Multi-Path Transport Protocol (MPRTP-AR) methods [1], [2], and [50]. Since RTP lacks congestion control, it is necessary to use a sizable receiver buffer while playing a Constant Bit Rate CBR video to accommodate for the varied route latencies of the RTP streams [7]. Similar to RTCP reporting in RTP, MRTP and MPRTP both employ QoS reporting (such as transmitter report and receiving report) to transmit periodical per flow and session information. The application in MRTP determines the amount of time between reports, and the receiver can adjust it depending on the state of the network.

The Multi-flow Real-time Transport Protocol (MRTP) addresses the failure and congestion in mobile wireless ad hoc networks [1]. The method, according to the authors, is also relevant to the Internet. The Multi-flow Real time Transport Control Protocol is combined with MRTP (MRTCP). As a natural extension of RTP/RTCP, MRTP/MRTCP supports media delivery across different wireless networks. Comparatively to RTP, MRTP is a session-oriented protocol. In order to share information, MRTCP initiates the session in a three-handshake fashion (e.g., available paths). Depending on the QoS reports, it is possible to add or delete pathways during data transfer through UDP, TCP, or SCTP. In particular, media divides into flows, each of which is for a single route (in MRTP, the term "flow" refers to a sequence of video packets delivered over a single way). The ADD/DELETE acknowledgments (ACKs) are used by MRTCP to regulate flows. To ensure dependable delivery, Reports on Quality of Service are sent over the optimal path or many pathways. The sender can adjust to transmission faults using these reports. By allocating data to more appropriate channels and increasing redundancy, for instance, error resistance can be improved. At the receiver side, there is a reassembly buffer that uses the session ID, flow ID, and flow sequence number to reorder, reassemble, and adjust for jitter.

In order to deal with unstable UDP/IP, MRTP utilizes a retransmission mechanism to retransmit packets. RTT determines the retransmission timeout value, and the application determines the maximum number of retransmissions. MRTP might be integrated with a variety of error control techniques, such as Forward Error Correction (FEC), Multiple Description Coding (MDC), or Automatic Repeat reQuest (ARQ). Finally, the research's findings demonstrate that MRTP performs better than single path RTP in terms of received video quality. The data distribution mechanism may be selected in MRTP. Simple Round Robin, striping (across several servers), layered coding, multiple description coding, or object-oriented coding are a few examples (video or audio objects encode individually).

A RTP modification that permits multipath transmission of real-time media is called Multipath RTP (MPRTP) [2]. MPRTP aims to decrease latency. Each path's data distribution is recalculated by the scheduler. after acquiring information on its characteristics, initially allocating an equal traffic rate to each way. The greatest priority is given to I-frames, which are sent along the route with the most available bandwidth, the smallest amount of latency, and the fewest packet losses. Retransmitting packets on the route with the most bandwidth is done by the sender, the shortest latency, and the least degree of packet loss after receiving the NACK command to do so. The method seeks to maintain load balancing by exploiting network characteristics but does not include congestion control. With an adjustable playback buffer, A de-jitter was created by the authors. method for the receiver side to handle RTT fluctuation and packet reordering. An MPRTP sender sends a sub-flow ID and sub-flow-specific sequence numbers to each path in order to compute at the receiver side, packet jitter, packet loss, and packet discards that are connected to a sub-flow. A sub-flow in MPRTP refers to a group of video packets that are broadcast over a single route. The strategy is less unjust than RTP. since it aims to distribute data along paths and balance the system.

The MPRTP-AR, or based on application-level relay, the Multipath Real-Time Transport Protocol, was developed by the Internet Engineering Task Force (IETF). The two sub-layers of the proposed MPRTP- AR protocol stack are the Sub-layer for multipath transport control (MPTC) and the RTP sub-layer, as shown in Figure 2. This protocol benefits from the RTP sub-support layer's for current RTP applications. Consequently, it is unnecessary to alter the application programming interface (API). Functions including flow partitioning, sub-flow packing, and recombination, as well as sub-flow reporting, are handled by the MPTC sub-layer [9].

Data from the application layer is prepared at the sender side and transferred to the MPTC sub-layer in the form of RTP packets. Then converted by MPTC into data packets MPRTP-AR. When MPRTP-AR data packets are received, MPTC removes their fixed header and delivers them to the RTP sublayer. RTCP packets may also be generated by the RTP sub-layer in order to generate media transport statistics. MPRTP-AR data packets may contain RTCP data. which the MPTC sub-layer would disseminate across a number of different pathways.

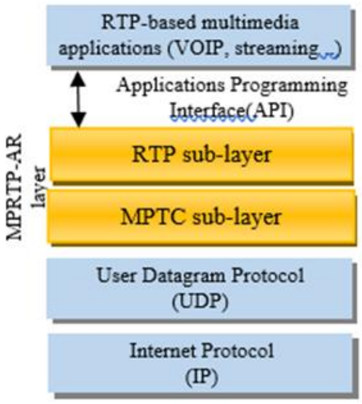

Figure 2: MPRTP-AR protocol stack [9]

In order to offer MPRTP-AR reports and keep-alive packets, MPRTP-AR control packets are specified in addition to MPRTP-AR data packets. Packet loss rate, jitter, and other transport properties of active pathways are included in MPRTP-AR reports, Flow Recombination Report (FRR) and sub-flow Receiver Report (SRR), as well as their effects on scheduling and flow partitioning. Flow partitioning approaches are divided into two categories: coding-aware methods and coding-unaware methods. On the RTP sub-layer, coding-aware approaches are utilized for object-oriented coding, multiple description coding, and layering coding. This allows each coding flow to have its own sub-flow, or numerous coding flows can be multiplexed into a single sub-flow. On the MPTC sub-layer, coding-aware approaches are used, and the effectiveness of the related active routes affects how evenly the RTP/RTCP from the upper layer is distributed. The entire recombined flows are also optionally provided for flow reporting.

HTTP Dynamic Adaptive Streaming (DASH)

In 2011, MPEG standardized Dynamic Adaptive Streaming over HTTP (MPEG-DASH) [9]. Both VoD and live video transmission are supported by DASH. The DASH system and its primary performance constraints are first described, then discuss rate adaption techniques. After that, works are based on this protocol that is discussed in detail last. The DASH system illustrated in Figure 3, uses the same foundational technology as HTTP adaptive streaming. At the server side of the DASH system, representations are broken up into tiny pieces. Text, video, audio, and other DASH component attributes are specified in an XML file called (MPD). The DASH client is typically in charge of selecting the following media segments and requesting the relevant HTTP URL. As a result, a rate adaptation technique, shown by an adaptation engine in Figure 3, is required to select the suitable segments' bit rate while taking into consideration the network circumstances, the media playout environment (such as the playout buffer level), and the segment availability provided by the MPD [35].

Key problems that affect QoE, such as delay, stall, and video quality shifts, are caused by the rate adaption mechanism. Pre-buffering, also known as startup delay, is the period of time between a client request and the beginning of the playback of a video. Because one or more portions typically need to be downloaded in full before the video can begin to play, there is a delay. Although studies show that it frequently leads to people ceasing from viewing the video, this delay does help to prevent stalling under bad network conditions [37]. It's crucial to remember that live and interactive video streaming only has a few hundred milliseconds of pre-buffering capacity for video, whereas VoD streaming programs can pre-buffer a few seconds of video [2].

When the playback buffer is empty and the system needs to wait to re-buffer the video, the system experiences pause in the video playback that are referred to as "stalls" or "interruptions" [38]. According to studies, stalling occurs in approximately 40 to 70 percent of all sessions [37]. This problem typically arises as a result of limited bandwidth.

After the coding procedure is complete, each segment in DASH is ready to send. Dead time is the period of time between receiving the last packet of one segment and requesting the subsequent segment. For instance, it takes at least 3 seconds (when segments last 1 second) to complete this method plus the time it takes for TCP to transfer data [39]. Using a dynamic way to determine an appropriate segment size is one strategy for addressing the stall issue (segment duration). According to studies in [41] and [40], segment size significantly affects live latency. While latency is greatly reduced with lower size segmentation, the amount of HTTP requests rises and, as a result, the bandwidth is constrained [14].

Applying sub-segmentation transmission is another strategy to reduce latency and, as a result, resolve the stall problem. This method breaks each segment into numerous smaller segments, and the receiver retrieves the smaller segments before the entire segment coding process is complete [41]. This method is enhanced by delivering subsegments via many links concurrently, which entails adding multipath transmission capacity, in order to speed up the fetching segment speed. In a few papers [16], [14], [3], the multipath transmission technique is employed to reach this goal. Although it also increases HTTP request cost, the sub-segmentation transmission mechanism [14]. Sub-segmentation transmission in particular is to blame for the overhead issue as it takes an average RTT of RTT after each request for the client to obtain a response from the server. This cost results in a significant delay when dealing with This method is enhanced by delivering subsegments via many links concurrently, which entails adding multipath transmission capacity, in order to speed up the fetching segment speed. The huge file has a few little segments or subsegments. A method to reduce the quantity and duration of each idle time is the HTTP pipelining [42]. In this method, the client submits the request for the following subsegment before the current sub-segment has finished downloading. In contrast, pipelining requires that server replies to requests be returned to clients in the same sequence in which they were sent to the server. The other requests would thus be stopped if one request takes a long time to process. Pipelining is not extensively used because of this. The new version of HTTP (HTTP/2) may be utilized to address the overhead issue since it allows material to be pre-pushed, which lowers live latency and network traffic [41]. Another issue that affects user-side video quality and irritates viewers is switching between multiple video-quality representations. When the network bandwidth or buffer occupancy status changes, video switching occurs. In order to provide the best possible user experience, it is crucial to create an appropriate rate adaption mechanism that could quickly identify network resources and congestion [48].

DASH and Multipath support

Although multipath is being touted as the future of DASH, the current version does not support it. Table I lists several attempts to combine MPTCP and MP-DASH (e.g., [12], [48], MP-DASH [46], [45], and [15]) with Multipath Separated Rate TCP (e.g., [14], [3], [16]). The table demonstrates how interest in fusing MPTCP and DASH has grown recently. This is because MPTCP provides mobility and bandwidth aggregation. Additionally, middlebox friendly, the Linux kernel supports MPTCP. Additionally, MPTCP has attracted a lot of industry attention [8], [51]. "Whether MPTCP will always enhance mobile video streaming?" was a topic that James et al., [37] investigated. The performance of various DASH over MPTCP situations was examined in this study. The findings demonstrate that even with a limited bandwidth capacity on the secondary way, having two paths with consistent band-widths is advantageous. The secondary link also performs well in this area when the primary channel has severe bandwidth changes. There are, however, some bad ones as well. Adding an unstable secondary way might harm the stable primary path, for example, if the second path's bandwidth is insufficient to support increased video bit rates. As a result, MPTCP is quite sensitive to fluctuations in bandwidth. More multipath waste raises the cost of the transition to quality and increases energy consumption, according to the research.

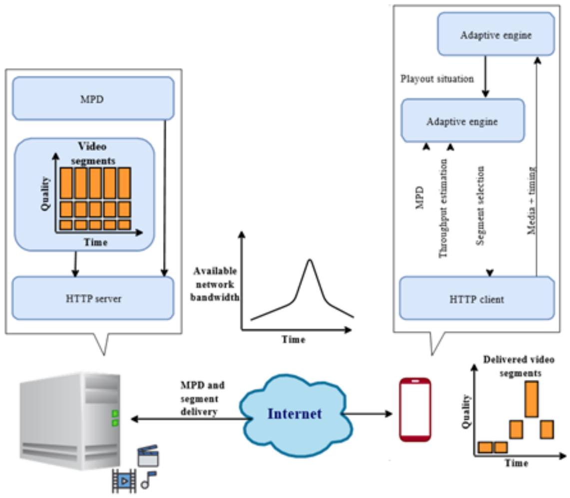

Figure 3: DASH system [10]

When offering multipath delivery for DASH, it should be noted that the client or server chooses how to schedule the packets. Because the DASH logic is client-side, the client is responsible for choosing the appropriate path and getting the pertinent segments/subsegments along the way in all the work evaluated that distributes data over separate TCP connections in table I. However, because the MPTCP scheduler is server-side and the DASH logic is client-side, integrating DASH with MPTCP is difficult. Additionally, the application layer cannot see MPTCP. As a result, in the works of Table I that were investigated, when MPTCP is utilized as the transport protocol, the rate adaption logic is kept on the client side. However, choosing which packages to use might be difficult and how to distribute them along the pathways is made at the server side or at both the client and server sides.

A Markov decision process (MDP) was used by Xing et al. [52] to identify the video streaming process as a reinforcement learning problem. This description is based on current research for non-scalable and scalable video coding (SVC) [14][3]. Reduce beginning latency while boosting video quality and smoothness are the project's goals. According to the predicted available bandwidth and the current queue length in each of these works, the rate adaption technique is used to choose the subsequent segment. The Markov channel model is used to calculate an accurate available bandwidth estimate. In this manner, adaption logic ascertains the most effective course of action based on the transit probability of each connection in real-time (e.g., using both links, using only WiFi links, client wait, or smoothing). Additionally, a mechanism has been built to reward each activity that considers the requirements for video quality of service. However, the main drawback of MDP is its high computational cost for tackling complicated optimization problems, particularly for online and high mobile speed users [52], it is not a content-aware solution.

DASH protocol was changed by Chowrikoppalu et al., [15] to take use of multipath capabilities. In this study, the suggested total path stability and buffer level parameters as well as a proposed bandwidth estimate technique are input into the adaptation logic. The technique for estimating bandwidth is based on intercepting interface-level packets. To display the variance in bandwidth on each path and on the MPTCP connection, respectively, both total path stability and path stability are specified. The biggest issue with this strategy is that it cannot access the information about the video content.

Houze' et al. created a video player with multipath capabilities over several TCP connections in [16]. This strategy aims to achieve low-latency video transmission using DASH (under 26 ms). The server in this approach encodes each representation's frames and puts them into the appropriate segment once per x milliseconds (x is dependent on frame rate; for instance, x for 25 fps is 40 ms). Before the frame's deadline (play time) and in x milliseconds before a new frame becomes available for fetching, the client must fetch each entire frame. The authors used video distribution via various pathways to decrease latency for this target. Each frame is divided into byte ranges to be sent via various pathways, and the method incorporates a system to determine which byte range size will result in the shortest inter-arrival time.

The Head of Line (HoL) problem is lessened as a result of the bigger byte ranges being sent via quicker pathways, which reduces the fluctuation in transfer latency. Additionally, a different modified process is suggested to choose the appropriate representation. The greatest frame of each representation is taken into account when a segment starts in this technique. Typically, the first frame of each depiction is the largest frame (I-frame). Therefore, a representation that has a high likelihood of arriving at the destination on time would be chosen for the largest frame. The approach's failure to take into account the video content information is the problem, though. Additionally, whereas the work predicts the speed of each path using RTT, the scheduling approach must be improved to manage the pathways.

MPEG Media Transport (MMT)

The Motion Picture Experts Group (MPEG) standardized MPEG Media Transport (MMT) [46] in 2014. The MPEG-H (High Efficiency Coding and MMT is part of the Media Delivery in Heterogeneous Environments) standard [24]. VoD and real-time streaming of video are supported by this application layer transport protocol. MMT has been extensively used for significant global developments in televisual technology, including Multi-View Video (MVV), Virtual Reality (VR), and Augmented Reality (AR) technologies, as well as three-dimensional (3-D) scene communication [53], [54]. Here, the authors firstly list additional protocol characteristics. Then, they go through the functions of MMT as well as the specifics of data transfer. The surveyed works that use the MMT protocol are finally discussed.

MMT is capable of handling any media distribution that is bidirectional, unidirectional, unicast, multisource, multicast, and even multi-path. Additionally, MMT allows internet and streaming video broadcasts [55], [56]. Additionally, All-Internet Protocol (All-IP) networks are available and conventional IPTV broadcasting services. One of the most crucial characteristics of MMT is its capacity to offer hybrid media. The mix of supplied media components through several types of networks is referred to as hybrid media delivery [57]. For instance, it may consist of one broadcast channel and one broadband channel or two broadband channels. According to ISO/IEC 23008-13 [48], there are three categories of hybrid service scenarios for MMT: live and non-live, presentation and decoding, and same/different transport methods. The phrase "live and non-live" refers to combining live streaming components or live with previously stored components. The combination of the stream components for synchronized presentation or synchronized decoding makes up the second category, presentation and decoding. The combination of just MMT components or MMT components plus other format components is supported by the third category, which includes the same transport schemes and various transport schemes (e.g., MPEG-2 TS). Additionally, ISO/IEC 23008-13 [48] presents an example of a hybrid paradigm that uses DASH (as a broadband channel) and MMT (as a broadcast channel) over heterogeneous networks.

MMT and Multipath support

A technique to establish multipath delivery through MMT was defined by Kolan et al., [17]. This approach uses the MMT protocol to initiate and manage multipath sessions between the transmitter and receiver using signaling protocols like RTSP or HTTP (transport connection could be either TCP or UDP). By sending each other OPTIONS requests, the client and server in RTSP, for instance, might be made aware of the multipath capabilities. The header of the OPTIONS request might be used to implement this new option tag, termed multipath. The client provides the "Multipthid" header to inform the server of its multipath capabilities while HTTP is used to establish multipath connections. During the connection, a network path can also be added or removed. MMT occasionally delivers feedback to the sender during media delivery to let them know about the route quality information (e.g., loss, delay, and jitter). As a result, the sender might observe the conditions on several pathways and choose higher performing paths on the fly for packets.

In order to broadcast real-time video over heterogeneous wireless networks, Afzal et al. [18] proposed a special path-and-content-aware scheduling approach for MMT. The authors assert that their work is the first to attempt to enhance the MMT standard through the use of multipath scheduling techniques. Some of the techniques used to enhance the perceived video quality include adaptive video traffic split schemes, Markovian-based methods, a discard, and a content-aware scheduling strategy that is implemented at the server side. The adaptive video traffic split system distributes an appropriate bit rate for each transmission line while taking into consideration the heterogeneous network environment in order to conduct load balancing, relieve congestion, and effectively utilize each way's capacity. The route conditions and transition probabilities are calculated using the Markovian-based approach. By avoiding delivering packets that would likely be lost, the discard method lessens congestion. By duplicating or allocating them to the optimal path, content aware strategy preserves packets of high priority (I frame and the nearest N (P frames), referred to as near-I (NI) frames in this study). The client calculates the path metrics and continually evaluates the condition of the path; These measurements are then transmitted to the server over the most efficient path as feedback information packets. The MMT standard's feedback signaling techniques are used for this objective. Finally, by significantly reducing packet loss rates (62%), the suggested path-and-content-aware scheduling technique resulted in QoE gains of about 12 dB for PSNR and 0.15 for SSIM. Because the scheduler may be integrated into client/server applications, the technique does not necessitate any modifications to the protocol itself.

Table 1. Dash's support for multipath

Ref.	year	Separate TCP	MPTCP
[16]	2016	Y	N
[25]	2012	N	Y
[3]	2014	Y	N
[27]	2016	Y	N
[21]	2004	N	Y
[22]	2014	N	Y
[23]	2012	N	Y
[24]	2014	N	Y

An implementation of a synchronization system for hierarchical video streams over heterogeneous networks was proposed by Sohn et al. [19]. This method combines HTTP video streaming over broadband with MMT for broadcasting video. Each layer is divided into time segments (measured in seconds), and the duration value can change depending on how the user defines it. In the experiment, a 3-layer SHVC-encoded stream is used: a base layer (High-Definition HD), a first improved layer (Full HD (FHD): 2K), and a second enhanced layer (Ultra HD UHD: 4K). Video is sent across broadcast networks for its base layer and first enhanced layer (MMT allows multiplexing at the packet level), and over broadband networks for its second enhanced layer. Even if the receiver can connect to the networks, if the receiver's display has HD resolution, it will drop the first enhanced layer's data among the data delivered through the broadcasting channel and not require a connection to the server for the second enhanced layer. Presentation Information PI may be transmitted through broadcast routes and provides crucial data such the content resolution, where the material is located, and the MMT eXtension Document (MXD). MXD, which imitates the MPD of DASH-SVC, is put into PI. MXD organizes content synchronization metadata and synchronizes the contents across diverse networks. The receiver side is where the synchronization strategy is put into action. The portions that can deliver on time are requested by the receiver. Based on bandwidth calculations and information from MXD about segment sizes, the anticipated download time for each segment for this destination is calculated. There is no scheduling method used to control the pathways in this approach, which is unaware of video content.

Transport Layer Approaches

The network data is directly accessible via video streaming techniques concentrating on transport layer protocols. As a result, they are able to predict relevant End-to-End properties of each link in multipath scenarios, such capacity and congestion [58]. The major issue with these solutions is that they typically need for alterations to the established multipath transport protocols, which may even call for alterations to the operating system kernel. Although there are many works that take use of multipath transmission at the transport layer, MPTCP and SCTP are the two most often used transport protocols with multihomed capability. In the following paragraphs a survey of researches that use SCTP/CMT, MPTCP, and UDP in this subsection.

User Datagram Protocol (UDP)

The IETF standardized the User Datagram Protocol (UDP), which is now extensively used for multicast, unicast, broadcast, and anycast communications. Then, briefly reviewing the UDP's fundamentals and talk about pertinent multipath initiatives. UDP was created to transmit data using just one route. It is a connectionless protocol that does not ensure reliable and in-order delivery and does not transmit data using sequence numbers [51]. Additionally, UDP lacks congestion management for bandwidth adaptability. These characteristics make UDP a quick transfer protocol [47] that makes it simple to develop video streaming solutions. UDP transmits data at the application's bit rate is the same, albeit, because to the lack of bandwidth adaption. As a result, when the network is busy, packets are ignored unless the application delays, which results in video distortion and decreased QoE [59]. Furthermore, UDP may use a significant portion of the available bandwidth if congestion management is not implemented, which would be unjust to other network flows that avoid congestion [22].

UDP and Multipath support

Several initiatives exist to include multiple transmission and bandwidth aggregation into UDP for streaming video [13], [20], [21]. Be aware that rate balancing techniques were implemented in the strategies suggested in BEMA [13] and [20] in order to prevent network congestion.

The Bandwidth Efficient Multipath Streaming (BEMA) protocol was developed by Wu et al. [13] for real-time High-definition video streaming over heterogeneous wireless networks. It was the first research, according to them, to employ Raptor coding and prioritized scheduling. This content-aware model transmits I-frame packets across all accessible pathways and sends packets with a greater priority on the better-qualified paths. In addition, Forward Error Correction (FEC) is used in the strategy to safeguard transmission data. TCP-Friendly Rate Control (TFRC) is another feature that BEMA provides to ensure fairness for TCP traffic. The term "TFRC" refers to an equation-based congestion control approach for unicast multimedia traffic [59]. After receiving information from the TFRC on the rate of loss events at the receiver, the sender modifies the rate based on the congestion estimate and an equation that depicts the behavior of TCP congestion management. TFRC responds to the congestion over longer times and with less variation than typical TCP congestion management [59]. During wireless losses, TFRC may, however, result in an unnecessarily reduced transmission rate. Then, in order to differentiate between wireless losses and congestion losses, BEMA adds a Zig-Zag scheme [59]. TFRC will only count a packet loss as a lost packet if Zig-Zag assesses it as a congestion loss [59]. Since the feedback data is crucial for correct scheduling and has a significant impact on performance, it is routinely transferred from the client to the server through a dependable TCP connection. A distortion-aware scalable video streaming to several multihomed clients was proposed by Freris et al., [20]. The authors assert that their work is the first to concurrently take into account end-to-end rate control and scalable flow adaptation for multipath in heterogeneous access networks. The server-side division of the requested video stream into sub-streams occurs in this method. In order to reduce visual distortion, The authors developed an algorithm that takes into consideration network information (such available bandwidth and RTT) and visual characteristics to calculate the pace of each sub-stream and the packets that should be included in each sub-stream. In addition, several cost functions are suggested to offer service distinction and user fairness. Heuristic methods for deterministic packet scheduling were also created by the authors. Each packet is only communicated if all other relevant packets in lower tiers have already been sent once it is a scalable streaming strategy. On the client, the sub-streams are combined into one scalable video stream.

Transmission Control Protocol (TCP)

The Internet Engineering Task Force (IETF) defined the transport protocol known as Transmission Control Protocol (TCP) [33] in 1981. This protocol is widely used in real-time communications (RTC) [59] and in HTTP-based applications for video streaming. Here we take a closer look at TCP and talk about one study based on it.

TCP is intended to transmit data via a single path. Sequence numbers are used by TCP during the data transmission process to identify losses, ensure that packets are sent in the correct order, and reassemble the received data [51]. For packets that were successfully received, the receiver sends ACKs. The usage of these ACKs enables dependable communication. Two situations result in retransmission: first, if the receiver doesn't provide an ACK, which is determined by utilizing a Retransmission Time-Out retransmission timer (RTO), second, loss happened if the sender receives three duplicate ACKs. Retransmission uses bandwidth inefficiently and causes noticeable delays. Numerous protocol upgrades have been suggested. For example, selective acknowledgments (SACK) [60], in which the receiver informs the sender of all successfully arriving packets so the sender only retransmits those segments that were actually lost, and cumulative ACK, which informs the sender of the most recent successfully received packet. Additionally, Explicit Congestion Notification (ECN) has been suggested as an extra feature to gather data on congestion gradually and inform the sender of the level of congestion [61]. TCP allows for network congestion adaptation and minimizes packet loss by using congestion management by monitoring packet losses and/or delay fluctuations [51]. When there is insufficient network capacity, TCP broadcasts video data at a lower bit rate than what is necessary. Video playing may stall as a result because video transmission takes longer than video playback. Although delay severely degrades the apparent video quality, it is often preferable over video distortions when it comes to VoD distribution [59]. Additionally, TCP can pass through NATs and firewalls, which is a problem with UDP and makes TCP a more advantageous transport protocol for video services [12].

Multipath TCP (MPTCP)

Since 2009, the IETF has been developing Multipath TCP (MPTCP), a well-known multipath transmission protocol [60]. Both the Linux kernel [59] and the FreeBSD-10.x kernel [58] include experimental kernel patches that support MPTCP. On smartphones, the industry has also deployed MPTCP [70]. Since 2013, two significant releases have been the speech recognition (SIRI) application [46] and for any IOS11 application [57]. Another significant MPTCP implementation in high-end Android handsets depends on network-operated SOCKS proxies, with KT Corporation attaining bandwidth of 1 Gbps in Korea 2015 [57].

Multiple pathways were intended to be used by MPTCP for transmitting data. MPTCP establishes several sub-flows for a single MPTCP session in particular. A sub-flow resembles a typical TCP connection since it is a TCP flow over a specific route. To further indicate that the connection is MPTCP rather than TCP, there is the MP CAPABLE option. The MPTCP session also has a token attached to it. To add sub-flows to this specific session, use this token. Figure 4 illustrates how the application layer in MPTCP views MPTCP connections as distinct. As a result, without the application layer being aware of it, the transport layer of the sender packetizes data into TCP packets, and the transport layer of the receiver rearranges and recreates the byte stream. As a result, the application layer is left alone, and a common socket API is used.

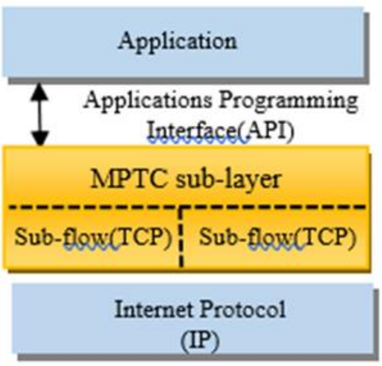

Figure 4: MPTCP protocol stack

Each packet in the data transmission process has two sequence numbers: the SPTC (SSN) for loss detection and a second Contents Sequence Number (DSN) for the recipient to use to reassemble the original data. ACKs are also used by MPTCP at the sub-flow and connection level. At the sub-flow level, SACK/Cumulative ACKs are utilized, while DSN-ACKs are used at the connection level [51]. MPTCP employs the same retransmission mechanism as standard TCP for data transmission security. Additionally, retransmission across a different sub-flow may be used in the event of packet loss over a sub-flow.

Default to prevent an unfair TCP connection, MPTCP utilizes linked congestion control (Congestion control is unique to each MPTCP connection). Because MPTCP over ordinary TCP connections may act unjustly, our approach offers better balance of congestion than merely applying Control of TCP congestion across each sub-flow (uncoupled) [56], [55].

At the receiver side, packets from several pathways are received and reordered using a common MPTCP receiving buffer [12]. In other words, at the receiver side, a single window is shared by all sub-flows.

There have been several packet scheduling algorithms created for MPTCP since they play a significant part in multipath approaches. The authors in [38] analyzes the performance comparison of scheduling approaches for multipath transmission, and [55] implements and assesses several MPTCP schedulers. First-In-First-Out (FIFO) order selection and RTT-based policy-based route mapping is the default MPTCP packet scheduling strategies.

Middleboxes are supported by MPTCP, which is also compatible with the present network architecture [51]. This is because each sub-flow packet in the SSN has a successive sequence number. It can therefore go via middleboxes [54]. However, MPTCP manages middleboxes by reverting to the standard TCP in the event of a dispute [53]. Additionally, MPTCP offers load balancing, mobility, and resilience [47].

According to research in [45] and [38], MPTCP demonstrates performance issues most noticeably in the case of different paths. Here is a breakdown of what causes MPTCP performance constraints:

- Out-of-order packets: MPTCP has difficulty with out-of-order packets. Comparing accept (SPTCP) and MPTCP in [45] demonstrates that SPTCP outperforms MPTCP when pathways are severely throughput-balanced. Because so many packets are transmitted out of order, MPTCP performs badly in this situation. When using the 5G network alongside other wireless networks, such throughput imbalances may occur often. The method suggested in [45] showed a flexible MPTCP route control to address the out-of-order problem in our reviewed works.
- ARQ mechanism causing HOL blocking: Even more so than a single TCP connection, using the ARQ mechanism via MPTCP commonly results in HOL blocking issues [12]. HOL performs poorly and has a significant End-to-End delay. The suggested solutions in AD-MIT [23], [24], [25], and

[26] made an effort to address the retransmission issue in order to reduce End-to-End latency in this work.

- Traffic scheduling without regard to content: The MPTCP application is unaware of the availability of multipath connections. As a result, MPTCP is not aware of application metadata or functionality for video content. Cross-layer solutions for accessing the video material and deadlines were introduced via the methods suggested in [12] and [46], respectively.
- Completely dependable and arranged service: Video streaming is not needed by MPTCP, which is an extension of the TCP protocol with fully legacy, reliable and organized services. Some attempts [24], [25], and PR-MPTCP+ [26] have been made to apply the idea of partial dependability in MPTCP for real-time video distribution in the works we evaluated. This idea prevents retransmission at allowable loss rates and offers partly reliable video data delivery to the top layers [24], [25], [26].
- Improved network performance characteristics, such as bandwidth and latency, result from partial dependability, which ultimately improves the quality of experience [24].

SCTP and CMT protocols (extension of SCTP)

The initial SCTP standard was released in 2000 and later modified in RFC 3309 [51] and RFC 4415 [33]. RFC 2205 is no longer supported. The most recent version of the protocol specification, RFC 4205 [34], was standardized by the IETF in 2007. Different operating systems and platforms support SCTP, and it offers multihoming and multi-streaming (e.g., FreeBSD, Linux and Android). First off, the paper has a general review of SCTP, including information on the data transmission mechanism and SCTP features. after that, listed any performance restrictions. Finally, the paper talk about the works based on this methodology that were surveyed.

Like UDP, SCTP is a message-oriented protocol that promotes dependability utilizing TCP-like congestion control and retransmission [34]. By default, SCTP transfers data packets across one path as the primary way, with the usage of additional paths for redundancy (retransmission and backup packets). Using several pathways instead of only one increases resilience and reliability of data flow. Particularly, SCTP establishes a relationship with several IP addresses for every end host [35]. SCTP's association term describes the link between SCTP end hosts. Multi-streaming features offered by SCTP help to solve the HoL stalling issue. Each stream in SCTP is a subsection of the total data flow, and multi-streaming is the concurrent transmission of many separate streams of data inside an SCTP association. The way SCTP multi-streaming operates is by assigning stream sequence numbers to the individual stream pieces. While unordered delivery is possible between streams, sequence numbering ensures that packets are sent inside streams in the order they are received. As a result, even if other streams are blocked as a result of losses, the application layer can still receive the data that has come into a stream.

The default SCTP protocol also uses a separate sequence space, called a transmission sequence number, for each fragment, which is the basic unit of data in an SCTP packet (TSN) [34]. In order to identify missing data and recreate the receiver's source data, TSN is available worldwide for all streams. Additionally, SACK and cumulative TSN ACK are used as acknowledgment techniques. A field of the SACK called cumulative TSN ACK is used to tell the sender that the final successfully received DATA chunk has been received. SCTP utilizes a retransmission mechanism upon two different sorts of events for data transfer safety. Initially, when an RTO expires. Second, the identical data chunk is missing after four SACK pieces have reported gaps. Additionally, SCTP has uncoupled congestion control, and all routes on the receiver side share a buffer.

Enhanced Scheduling Techniques

There are numerous ways to make SCTP better in order to address the aforementioned issues and offer video streaming using this transport protocol. Extended primary path exchange in SCTP was mentioned as a concern before. Kelly et al. [27] suggested a delay-oriented technique for establishing the main path based on the least end-to-end delay and RTT. Utilizing this adaptive primary path selection in a lossy wireless environment improves quality, but SCTP performance is slowed by the frequent path changes. This strategy only completely utilizes the backup secondary channels while using the primary link for data delivery.

In [28], a more reliable SCTP-based method is demonstrated. With suggesting a selective bi-casting strategy, the authors were able to overcome packet loss. Therefore, the selective bi-casting approach repeats just crucial packets rather than transmitting the identical data across two distinct pathways (bi-casting), which would cause severe congestion and lower throughput. These significant packets have been transmitted again. However, this method does not classify private information, such as I-frames, as significant packets.

A Selective-Redundancy Multipath Transfer (SRMT) system was suggested by Da Silva et al., [29]. In this method, duplicate packets with higher priority and more effective delay limiting are sent over secondary pathways while the primary path is used to convey data. These backups prevent a drop in quality of experience. For packet selection via secondary pathways, there are two important considerations. The total number of redundant packets that must be transmitted is the first, It is calculated using the application's maximum tolerated delay and the smooth Round Trip Time (sRTT) of the major path. The second is the determination of which packets should be delivered again depending on their significance for recreating the video (a content-aware approach). I-frames, for instance, have the highest priority, and the packets that are first ordered within an I-frame have a higher priority than other packets. B-frames have the lowest importance, followed by P-frames. On the recipient side, duplicate packets would be rejected. To prevent the HOL issue, SRMT employs the standard SCTP handover method. To make Reliable SCTP flexible for video streaming, an extension to Partially Reliable SCTP (PR-SCTP) was first developed in [36], and later additional rules were provided in [37]. Some reliability level selection criteria were provided in PR-SCTP, much as how partial reliability was specified for MPTCP in the preceding Section. With PR-SCTP, you may select the retransmission policy by specifying a time limit or a maximum number of retransmissions, after which the packet will no longer be retransmitted. For time-sensitive applications requiring streaming video and audio, PR-SCTP is advantageous [38]. The suggested method in [30] used the partial dependability services of PR- SCTP for real-time H.264/AVC video streaming in their studied works. The Network Adaptation Layer (NAL) component of H.264/AVC is an abstraction layer placed over the actual encoded data. Decoding parameters and their relevance for decoding are contained in the NAL header. To determine the number of retransmissions for each I, P, and B-frame, PR-SCTP uses this information. The optimal values for the maximum number of retransmissions for different types of frames are determined using a probabilistic model, offering a trade-off between reliability and latency. The secondary pathways are used for retransmissions. The outcome demonstrates that the suggested approach outperforms TCP and UDP.

Concurrent Multipath Transfer (CMT) is another SCTP extension solution [49]. To increase performance and network resilience, the majority of CMT systems transport data simultaneously along all feasible pathways. There are several CMT-based systems, including CMT-DA [31], CMT-CA [32], and CMT-QA [11]. Among these studies, CMT employs Round Robin for data distribution instead of any path selection approach. Inefficient retransmissions, out-of-order delivery, and HOL blockage at the receiver all rise with round robin for CMT. It also increases SACK overhead. In CMT-QA [11], CMT-DA [31], and CMT-CA [32], CMT evolved to conduct improved network situation estimates and pick qualified pathways for data transmission. In addition to the network condition, CMT-CA [32] is also provided with video content attributes. The architecture of the congestion control and retransmission mechanisms in these works also differs.

The route and quality-aware adaptive parallel multipath transfer (CMT-QA) method for packet scheduling over network channels was developed by Xu et al. [11]. This plan aims to lessen the out-of-order issue by cutting back on quick retransmissions and reordering delays. An optimal retransmission policy (ORP), a route quality estimation model (PQEM), and a data distribution scheduler (DDS) are presented in order to meet this goal. PQEM determines the quality of each link by calculating the dispersed data rate, which depends on the sending transmission latency and buffer size. The common sender buffer in PQEM is broken up into smaller sub-buffers. There is a dynamic allocation of buffer space., and each path has its own sub-buffer that is managed separately. Packet loss distinction is handled by ORP, which retransmits dropped packets through quicker routes. The quantity of data to be transported is depending on the congestion control parameters including: cwnd, sender buffer size, and rwnd. Data Distribution Scheduler (DDS) forecasts the arrival time of data spread across each lane. As a result, DDS distributes data according to each path so that they reach the recipient in the correct sequence. The acknowledgment technique is SACK, however, it ignores TCP's fairness to other traffic flows and is inefficient for video due to the lack of video content options.

According to Wu et al. [31], this was the first research to take into account video distortion in SCTP. They developed a distortion-aware parallel multipath transmission (CMT-DA) system to enhance HD video quality in diverse wireless situations. By reducing the effective loss rate for variable bit rate video streaming, this method seeks to reduce video distortion. There are three suggested major strategies to accomplish this goal: Data retransmission control, flow rate allocation, and route status estimate and congestion management. By analyzing ACK feedbacks, CMT-DA calculates route circumstances (such as RTT and available bandwidth) and then employs a distortion-aware model at the flow level to plan the packets. After each packet delivery, aggregated feedback packets are transmitted. To prevent losing or dropping during network transmission, the employed SACK/Cumulative ACK feedback packets return to the sender over the most secure routes. Additionally, RTT, cwnd, and RTO have established parameters for congestion control, which is developed per path. ECN adjusts the congestion window size when it senses route congestion. It is suggested that the rate controller dynamically chooses a selection of routes and establishes data transmission rates. In accordance with its definition, data retransmission control entails retransmitting packets that are anticipated to reach their destination before the deadline. For video streaming, it is insufficient to merely take flow level distortion into account without also looking at frame priority and decoding reliance of frames.

In another evaluated study, Wu et al. [32] developed a content-aware CMT (CMT-CA) strategy to enhance HD video quality in various wireless environments. They said that this approach was the first SCTP to do so. To achieve the best quality, CMT-CA strives to accurately predict video content attributes and schedule video frames accordingly. For this purpose, three crucial tactics are proposed: data distribution, congestion control, and quality evaluation-based decision making. Network circumstance and frame level distortion are estimated by quality evaluation-based decision-making. Furthermore, packet scheduling uses these bits of data. Similar to how it was mentioned for CMT-DA, SACK/Cumulative ACK feedbacks are delivered following the delivery of each packet over the most reliable pathways and are utilized to estimate the path condition. For CMT-CA, the congestion control is per path, Markov model-based (MDP), and TCP-Friendly. Additionally, RTT, cwnd, and RTO have established parameters for congestion control, which is developed per path. When a route is congested, the Zig-Zag scheme [59] recognizes it and MDP adjusts the congestion window size. Packet scheduling is made possible by data distribution, and I and P frames are sent in separate ways. As a result, high priority frames may be delivered first, reducing video jerkiness. Additionally, if a bandwidth limitation prevents the delivery of the parent frame, the suggested technique discards the video frame.

Table 2. Scheduling techniques used with multipath video streaming

Ref	Strategy	Description
34	SCTP protocol	More reliable SCTP-based method is demonstrated. By suggesting a selective bi-casting strategy that repeats just crucial packets rather than transmitting the identical data across two distinct pathways
35	SRMT	In this method, duplicate packets with higher priority and more effective delay limiting are sent over secondary pathways while the main path is used for data transfer.
60	CMT	To increase performance and network resilience, the majority of CMT systems transport data simultaneously along all feasible pathways.
16	CMT-QA	By lowering quick retransmissions and reordering delays, this method of packet scheduling decreases the out-of-order issue.
37	CMT-DA	Claiming that this was the first research to add video distortion in SCTP, in order to enhance HD video quality in diverse wireless settings. This method seeks to lessen video distortion by lowering the effective loss rate for variable bit rate video streaming.
38	CMT-CA	This method aims to enhance HD video quality in various wifi settings. They claimed that this strategy was the first SCTP to do this. CMT-CA works to correctly forecast the characteristics of video material and arrange video frames in order to produce the highest quality. To achieve this goal, three crucial strategies quality-based decision making, congestion control, and data distribution are recommended.

For instance, SACK [60], which gives the sender a list of correctly and wrongly received packets, and This approach thereby conserves network resources. In addition to the suggested techniques, CMT-CA makes use of related data retransmission techniques created in CMT-DA. cumulative ACK, which notifies the sender of the most recent successful packet reception.

Conclusion

In order to improve QoE for video streaming using multipath and multihomed overlay networks, a wide range of scholarly literature was surveyed. In an effort to develop the techniques and fully utilize the potential of many network interfaces with intelligent network routing, many research papers on multipath and multihoming are combined. The study found that there is now a lot of activity and dominance in both research and business at the transport and application levels. Due to the prevalence of devices with numerous network interfaces, such as smartphones and laptops, multihoming research is still expanding (like Wi-Fi, Ethernet, and 4G). Multihomed solution implementation still faces difficulties, sadly. The majority of multihoming operating systems deployments are either absent or difficult to implement. Additionally, due to energy restrictions and higher fees for 4G and 5G networks, mobile and wireless devices cannot fully benefit from multihoming. The survey concluded that Multi-Path TCP's (MPTCP) influence was reflected in the vast number of research articles on the subject. In order to increase network throughput and improve fail-over by switching between paths in the event of a path failure, Multi-Path TCP (MPTCP) uses several paths. Unfortunately, network middle boxes hurt Multi-Path TCP (MPTCP), causing MPTCP connections to revert to conventional TCP. Dynamic adaptive video streaming over HTTP like MPEG-DASH appears to be the most extensively used at the application layer. Numerous studies on DASH and multihomed multipath solutions were found, according to the survey. The fact that neither the server nor the client side needs to be modified is a significant benefit of client-based approaches like DASH players. Since, the multipath algorithm will continue to advance, and that fog computing will enable more video streaming to occur at the network edge. And the most important problem facing multipath technology is the process of dividing the data in the server and sending it on the two paths depending on the network conditions for each path, and then reassembling it at the user in the least time so that it does not affect the video stream. This could be a wide field for future research because there is no practical application, whether in mobile devices or portable devices that support multipath technology.

References

[1] M. Xing, S. Xiang, and L. Cai, “A real-time adaptive algorithm for video streaming over multiple wireless access networks,” IEEE Journal on Selected Areas in communications, vol. 32, no. 4, pp. 724–715, 2014.

[2] S. K. Singh, T. Das, and A. Jukan, “A survey on internet multipath routing and provisioning,” IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2157–271, 2015.

[3] M. Li, A. Lukyanenko, Z. Ou, A. Yla-Jaaski, S. Tarkoma, M. Coudron, and S. Secci, “Multipath transmission for the internet: A survey,” IEEE Communications Surveys Tutorials, vol. PP, no. 56, pp. 1–41, 2016.

[4] Y. Sani, A. Mauthe, and C. Edwards, “Adaptive bitrate selection: A survey,” IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2566–3014, 2017.

[5] Y. Lim, S. Aoki, I. Bouazizi, and J. Song, “New MPEG transport standard for next generation hybrid broadcasting system with IP,” IEEE Transactions on Broadcasting, vol. 15, no. 2, pp. 57–76, 2014.

[6] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hobfeld, and P. Tran- Gia, “A survey on quality of experience of HTTP adaptive streaming,” IEEE Communications Surveys & Tutorials, vol. 17, no. 1, pp. 469– 421, 2015.

[7] C. James, E. Halepovic, M. Wang, R. Jana, and N. Shankaranarayanan, “Is Multipath TCP (MPTCP) Beneficial for Video Streaming over DASH?” 24th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MAS- COTS), pp. 331–336, 2016.

[8] M. Xiao, V. Swaminathan, S. Wei, and S. Chen, “Evaluating and improving push-based video streaming with HTTP/2,” Proceedings of the 26th International Workshop on Network and Operating Systems Support for Digital Audio and Video, p. 3, 2016.

[9] Y. Sohn, M. Cho, M. Seo, and J. Paik, “A Synchronization Scheme for Hierarchical Video Streams over Heterogeneous Networks,” KSII Transactions on Internet & Information Systems, vol. 9, no. 8, 2015.

[10] J. Wu, C. Yuen, B. Cheng, M. Wang, and J. Chen, “Streaming high- quality mobile video with multipath TCP in heterogeneous wireless networks,” IEEE Transactions on Mobile Computing, vol. 15, no. 9, pp. 2345–2361, 2016.

[11] C. Xu, H. Huang, H. Zhang, C. Xiong, and L. Zhu, “Multipath Transmission Control Protocol (MPTCP) Partial Reliability Extension”, 2016.

[12] Y. Cao, Q. Liu, G. Luo, Y. Yi, and M. Huang, “PR-MPTCP+: Context- aware QoE-oriented multipath TCP partial reliability extension for real-time multimedia applications,” Visual Communications and Image Processing (VCIP), 2016, pp. 1–4, 2016.

[13] Mahmood, Alaa & Alsaif, Omar & Saleh, Ibrahim. (2022). Routing flying Ad Hoc network using salp swarm algorithm. Indonesian Journal of Electrical Engineering and Computer Science. 28. 946. 10.11591/ijeecs.v28.i2.pp946-953.

[14] C. A. G. da Silva, E. P. Ribeiro, and C. M. Pedroso, “Preventing quality degradation of video streaming using selective redundancy,” Computer Communications, vol. 61, pp. 39–51, 2016.

[15] J. Wu, B. Cheng, C. Yuen, Y. Shang, and J. Chen, “Distortion-aware concurrent multipath transfer for mobile video streaming in heterogeneous wireless networks,” IEEE Transactions on Mobile Computing, vol. 14, no. 4, pp. 664–701, 2015.

[16] J. Wu, C. Yuen, M. Wang, and J. Chen, “Content-aware concurrent mul- tipath transfer for high-definition video streaming over heterogeneous wireless networks,” IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 3, pp. 710–723, 2016.

[17] J. Lennox, M. Westerlund, Q. Wu, and C. Perkins, “Sending Multiple RTP Streams in a Single RTP Session,” Internet Requests for Comments, March 2017.

[18] S. Wei and V. Swaminathan, “Low latency live video streaming over HTTP 2.0,” Proceedings of Network and Operating System Support on Digital Audio and Video Workshop, p. 37, 2014.

[19] J. Le Feuvre, C. Concolato, N. Bouzakaria, and V.-T.-T. Nguyen, “MPEG-DASH for low latency and hybrid streaming services,” Pro- ceedings of the 23rd ACM international conference on Multimedia, pp. 751–752, 2015.

[20] Le, H. T., Nguyen, H. N., Pham Ngoc, N., Pham, A. T., & Thang, T. C. (2016). A novel adaptation method for HTTP streaming of VBR videos over mobile networks. Mobile Information Systems, 2016.

[21] Samba, Alassane, Yann Busnel, Alberto Blanc, Philippe Dooze, and Gwendal Simon. "Instantaneous throughput prediction in cellular networks: Which information is needed?." In 2017 IFIP/IEEE Symposium on Integrated

Network and Service Management (IM), pp. 624-627. IEEE, 2017..

[22] Saleh, Ibrahim, Alsaif, Omar and Yahya, Maan. (2020). Optimal Distributed Decision in Wireless Sensor Network Using Gray Wolf Optimization. 10.11591/ijai.v8.i1.ppxx-xx.

[23] Y. Shuai, G. Petrovic, and T. Herfet, "Server-driven rate control for adaptive video streaming using virtual client buffers," in Consumer Electronics–Berlin (ICCE-Berlin), 2014 IEEE Fourth International Conference on. IEEE, 2014, pp. 45–49.

[24] S. Wilk, D. Stohr, and W. Effelsberg, "VAS: a video adaptation service to support mobile video," in Proceedings of the 25th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video. ACM, 2015, pp. 37–42.

[25] H. Mao, R. Netravali, and M. Alizadeh, "Neural adaptive video streaming with pensieve," in Proceedings of the Conference of the ACM Special Interest Group on Data Communication. ACM, 2017, pp. 60–210.

[26] A. Samba, Y. Busnel, A. Blanc, P. Dooze, and G. Simon, "Instantaneous Throughput Prediction in Cellular Networks: Which Information Is Needed?" IFIP/IEEE International Symposium on Integrated Network Management (IM), 2017.

[27] J. Wu, C. Yuen, B. Cheng, Y. Yang, M. Wang, and J. Chen, "Bandwidth-efficient multipath transport protocol for quality- guaranteed real-time video over heterogeneous wireless networks", IEEE Transactions on Communications, vol. 64, no. 6, pp. 2477–2422, 2016.

[28] Ojanperä, Tiia, and Heli Kokkoniemi-Tarkkanen. "Wireless bandwidth management for multiple video clients through network-assisted DASH." In 2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 1-3. IEEE, 2016.

[29] J. W. Kleinrouweler, S. Cabrero, and P. Cesar, "Delivering stable high- quality video: An SDN architecture with DASH assisting network elements," in Proceedings of the 7th International Conference on Multimedia Systems. ACM, 2016, p. 4.

[30] Cofano, Giuseppe, Luca De Cicco, Thomas Zinner, Anh Nguyen-Ngoc, Phuoc Tran-Gia, and Saverio Mascolo. "Design and performance evaluation of network-assisted control strategies for HTTP adaptive streaming." ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 13, no. 3s (2017): 1-24.

[31] Cofano, Giuseppe, Luca De Cicco, Thomas Zinner, Anh Nguyen-Ngoc, Phuoc Tran-Gia, and Saverio Mascolo. "Design and performance evaluation of network-assisted control strategies for HTTP adaptive streaming." ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 13, no. 3s (2017): 1-24.

[32] "Information technology – Dynamic adaptive streaming over HTTP (DASH) – part 5: Server and network assisted DASH (SAND)," ISO/IEC CD 23009-5, 2014.

[33] E. Thomas, M. van Deventer, T. Stockhammer, A. C. Begen, M.-L. Champel, and O. Oyman, "Applications and deployments of server and network assisted DASH (SAND) ", 2016.

[34] M. Luoto, T. Rautio, T. Ojanpera¨, and J. Ma¨kela¨, "Distributed decision engineAn information management architecture for autonomie wire- less networking," IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 713–719, 2015.

[35] S. Habib, J. Qadir, A. Ali, D. Habib, M. Li, and A. Sathiaseelan, "The past, present, and future of transport-layer multipath," Journal of Network and Computer Applications, vol. 75, pp. 236–258, 2016.

[36] A. Bokani, M. Hassan, and S. Kanhere, "Http-based adaptive streaming for mobile clients using markov decision process," IEEE 20th International Packet Video Workshop (PV), pp. 1–8, 2013.

[37] R. I.-R. BT.2074-1, "Service configuration, media transport protocol, and signalling information for MMT-based broadcasting systems", 2017.

[38] I. Bouazizi, "MPEG Media Transport Protocol (MMTP)," Working Draft, IETF Secretariat, Internet-Draft draft-bouazizi-mmtp-01, September 2014. [Online]. Available: http://www.ietf.org/internet- drafts/draft-bouazizi-mmtp-01.txt

[39] T.-J. Jung, H.-r. Lee, and K.-d. Seo, "Overview on MPEG MMT Technology and Its Application to Hybrid Media Delivery over Hetero- geneous Networks," Pacific Rim Conference on Multimedia, pp. 660– 669, 2015.

[40] "Information technology High efficiency coding and media delivery in heterogeneous environments Part 13: 3rd Edition MPEG Media Transport Implementation Guidelines", ISO/IEC 23008-13, 2013.

[41] B. Li, C. Wang, Y. Xu, and Z. Ma, "An MMT based heterogeneous multimedia system using QUIC," IEEE 2nd International Conference on Cloud Computing and Internet of Things (CCIOT), pp. 129–133, 2016.

[42] X. Corbillon, R. Aparicio-Pardo, N. Kuhn, G. Texier, and G. Simon, "Cross-layer scheduler for video streaming over MPTCP", Proceedings of the 7th International Conference on Multimedia Systems, p. 7, 2016.

[43] Hoßfeld, Tobias, Raimund Schatz, and Udo R. Krieger. "QoE of YouTube video streaming for current Internet transport protocols." In International Conference on Measurement, Modelling, and Evaluation of Computing Systems and Dependability and Fault Tolerance, pp. 136-150. Springer, Cham, 2014.

[44] S. Cen, P. C. Cosman, and G. M. Voelker, "End-to-end differentia- tion of congestion and wireless losses," IEEE/ACM Transactions on networking, vol. 11, no. 5, pp. 703–717, 2003.

[45] Floyd, Sally, Jamshid Mahdavi, Matt Mathis, and Matthew Podolsky. An extension to the selective acknowledgement (SACK) option for TCP. No. rfc2658. 2000.

[46] Black, D. Relaxing restrictions on explicit congestion notification (ECN) experimentation. No. rfc6811. 2018.

[47] M. Li, A. Lukyanenko, S. Tarkoma, Y. Cui, and A. Yla¨-Ja¨a¨ski, "Tolerating path heterogeneity in multipath TCP with bounded receive buffers," Computer Networks, vol. 64, pp. 1–14, 2014.

[48] O. Bonaventure and S. Seo, "Multipath TCP deployments," IETF Journal, vol. 12, no. 2, pp. 24–27, 2016.

[49] S. C. et al., "Advances in networking, part 1," https://developer.apple. com/videos/play/wwdc2017/707/, accessed:05-may-2019.

[50] Wischik, Damon, Costin Raiciu, Adam Greenhalgh, and Mark Handley. "Design, implementation and evaluation of congestion control for multipath {TCP}." In 8th USENIX Symposium on Networked Systems Design and Implementation (NSDI 11). 2011.

[51] Raiciu, Costin, Mark Handley, and Damon Wischik. Coupled congestion control for multipath transport protocols. No. rfc6356. 2011.

[52] Paasch, Christoph, Simone Ferlin, Ozgu Alay, and Olivier Bonaventure. "Experimental evaluation of multipath TCP schedulers." In Proceedings of the 2014 ACM SIGCOMM workshop on Capacity sharing workshop, pp. 27-32. 2014.

[53] Barré, Sébastien, Christoph Paasch, and Olivier Bonaventure. "Multipath TCP: from theory to practice." In International conference on research in networking, pp. 444-457. Springer, Berlin, Heidelberg, 2011.

[54] Hesmans, Benjamin, Fabien Duchene, Christoph Paasch, Gregory Detal, and Olivier Bonaventure. "Are TCP extensions middlebox-proof?." In Proceedings of the 2013 workshop on Hot topics in middleboxes and network function virtualization, pp. 37-42. 2013.

[55] S. Deng, R. Netravali, A. Sivaraman, and H. Balakrishnan, "Wifi, LTE, or both?: Measuring multi-homed wireless internet performance", Proceedings of the Conference on Internet Measurement Conference, pp. 68–61, 2014.

[56] Wu, J., Yuen, C., Cheng, B., Yang, Y., Wang, M., Chen, J.: Bandwidth-efficient multipath transport protocol for quality-guaranteed real-time video over heterogeneous wireless networks. IEEE Trans. Commun. 64(6), pp. 2477–2422, (2016).

[57] Y.-C. Chen, D. Towsley and R. Khalili, "MSPlayer: Multi-source and multi-path video streaming", IEEE J. Sel. Areas Commun., vol. 34, no. 8, pp. 259-2206, Aug. 2016.

[58] S. Habib, J. Qadir, A. Ali, D. Habib, M. Li, and A. Sathiaseelan, "The past, present, and future of transport-layer multipath", Journal of Network and Computer Applications, vol. 75, pp. 236–258, 2016.

[59] B. Han, F. Qian, L. Ji, V. Gopalakrishnan, and N. Bedminster, "MP- DASH: Adaptive Video Streaming Over Preference-Aware Multipath", Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies, pp. 48–62, 2016.

[60] C. A. G. da Silva, E. P. Ribeiro, and C. M. Pedroso, "Preventing quality degradation of video streaming using selective redundancy", Computer Communications, vol. 20, pp. 39–51, 2016.

[61] S. Petrangeli, T. Wu, T. Wauters, R. Huysegems, T. Bostoen and F. De Turck, "A machine learning-based framework for preventing video freezes in HTTP adaptive streaming", J. Netw. Comput. Appl., vol. 23, pp. 17-21, Sep. 2017.