J Reliable Intell Environ (2015) 1:147–158
DOI 10.1007/s40860-015-0012-4

ORIGINAL ARTICLE

Optimized congestion aware and rate-adaptive multimedia communication for application in fragile environment during disaster management and recovery

Amir Raza Khan[1] · Muddesar Iqbal[1] · Junaid Muzaffar[1] · Amir Nisar[1] · Amjad Ali[2] · Imran Raza[2] · Syed Asad Hussain[2]

Received: 19 August 2015 / Accepted: 24 October 2015 / Published online: 25 November 2015

Abstract Delivery of multimedia content over network has always been given vital importance since its inception. Different models for content delivery have been proposed with respect to multiple factors like bandwidth, quality of content, importance, and delivery method, etc. In this paper, we propose a Hyper Text Transfer Protocol-based rate adaptive transcoding and streaming technique that maintains its multimedia streaming quality, especially when network grows congested and fragile during Disaster Management and Recovery operations. The proposed solution provides users high-quality audio and video contents even when network resources are limited and fragile. We developed a real-time test bed that can deliver high-quality streaming videos by transcoding the original video in real time to a scalable codec, which allows streaming adaptation according to network dynamics. We also performed validation tests to ensure multimedia delivery in cloud computing, functionality on our hosted website for live communication between transmitter and receiver. From our experiments, we observed that the rate adaptive transcoding and streaming can efficiently deliver information services such as live multimedia in emergency and disasters under very low bandwidth environments.

Keywords Video transmission · Real time · Rate adaptive video streaming

1 Introduction

Since the beginning of human history, we are not safe from natural disasters like earthquakes, floods and dangerous typhoons. Infrastructures vanished in such conditions within minimum times and a lot of time is required to reinstate vanished services and facilities.

Previously, we have designed, developed and implemented a Wireless Mesh Network (WMN) called SwanMesh [1] which is a high performance multicast enabled dual-radio Wireless Mesh Network for application in Emergency and Disaster Recovery operations. The SwanMesh has been implemented using Wrap Boards and Linux Base System. We have developed a novel implementation of a Multicast extension to the AODV (MAODV) protocol in Linux Kernel 2.6 user space to support multicast operation of our SwanMesh testbed. A detailed architecture of the SwanMesh testbed is described in [1]. Furthermore, It also presents unicast communication throughput test results. We observed that SwanMesh can efficiently deliver data services such as broadband internet in emergency situations using its unicast functionality, although there were noticeable throughput drops after each hop. Therefore, the SwanMesh multicast operation may be suitable for delivery of real-time applica-

✉ Muddesar Iqbal
director.oric@gmail.com

Amir Raza Khan
mark@uog.edu.pk

Junaid Muzaffar
jmc@uog.edu.pk

Amjad Ali
amjad.ali@ciitlahore.edu.pk

Imran Raza
iraza@ciitlahore.edu.pk

Syed Asad Hussain
asadhussain@ciitlahore.edu.pk

[1] Pak-UK Institute of Innovative Technologies for Disaster Management, University of Gujrat, Gujrat, Pakistan

[2] Department of Computer Science, Communication and Networks Research Centre, COMSATS Institute of Information Technology, Lahore, Pakistan

tions in emergency communication, such as audio and video teleconferencing. We also proposed a Reliable Multimedia Multicast Communications scheme for SwanMesh [2,3]. SwanMesh can provide an alternative means of communication within minutes at the disaster site when traditional means of communication are destroyed. The plug and play and self organized, self-managed and self-healed capabilities of the developed WMN can provide an antifragile network for communication among stakeholders within minutes at the disaster site.

In a large-scale disaster, several incidents may require urgent response, such as when building collapses where trapped victims may require search, rescue and immediate medical treatment. During large-scale disasters, multiple incidents of this kind may occur within an area covered by a single mesh network. Operationally, these incidents require several wireless cameras to match the needs for required coverage of all incidents; hence, the network is required to facilitate several multimedia flows to support the rescue efforts and co-ordination among several rescue teams. If the medical consultant initiates a multimedia flow to provide medical consultancy to a remote location via video conferencing during an emergency or disaster recovery operation, this may be important to save lives; therefore, it is considered as critical, thus rejecting such flows when congestion grows. It could result in heavy loss of lives. These flows are to be accommodated even when the network is congested without rejecting any other multimedia streams as all the communication during disaster recovery is important to save precious lives.

In the present technology day era, a lot of research is being conducted in the field of how to reinstate the services if such disaster occurs. Antifragile and resilient nature was introduced by Professor Nassim Nicholas Talib, In some cases, some systems develop the ability to withstand with stress [4] Most of us are familiar with the Greek mythological creature Hydra and her many heads. When Hercules attempted to kill the monster by cutting off its head, two would grow in its place [5]. This may be an example of antifragility. Same methodology is being tried to implement in ICT infrastructures which design the solutions to dynamically adapt to difficult environments to support the ongoing operations.

As mentioned in the previous section, emergency and disaster recovery operations require the developed SwanMesh to accommodate all the critical priority flows even when the network is congested. Failing to do so, could result in loss of life if we fail to provide the critical informed needed at that particular time. Our developed WMN model must be capable of dynamically adapting the existing and new multimedia flows based on the network congestion. This can be achieved with our proposed Optimized Congestion Aware and Rate-Adaptive Multimedia Communication methodology for Disaster Management and Recovery. It adapts the network feeding rate of the flow based on network congestion. If the total bandwidth requirements of a new critical flow are greater than the network available resources, then the proposed scheme is equipped either lower the sender packet feeding rate of the new flow to match the available bandwidth resources at its edge. The proposed scheme also ensures that all the incidents a requiring response are matched against their priority level as critical or non-critical and are accommodated accordingly within the available resources of the network.

In this paper, we describe our proposed rate adaptive and congestion aware mechanism for the delivery of audio and video content during disaster management and recovery process. There are always lesser resources available in such conditions or they get choked due to a sudden increased demand. We assured the delivery of content by decreasing the bandwidth and processing requirements so the important multimedia enabled information could be made available where required for quick decision making by pulling the serving mechanism at edge level. To evaluate the performance of the proposed scheme, we have implemented a Cloud infrastructure using RedHat OpenStack cloud that will hold the servers in virtual forms integrated with messaging brokers ActiveMQ. In addition to this, different bandwidth scenarios are implemented that act as per situation.

2 Literature review

With the improved quality of mobile data services, improved performance terminal equipment and digital network have inclined the audio/video content delivery methodology with respect to low latency, jitter and efficiency. From 2003 to now, the US, South Korea, Japan and other countries are big stakeholders in developing TV technologies. According to the sources, TV broadcasting has increased from 34 million to 1.02 billion Euros [6]. As a distributed computing technology, cloud computing has been emerged as a savior. It is a structured-based allocation method which facilities all possible addressed issues of large service offering platforms. Multimedia broadcasting includes Content Providers, Network operators, Manufacturers and the End users.

Since the inception of multimedia in ICT, it has gained a lot of attention of researchers and professionals to provide hassle free and non-buffering video delivery to end users [7,8]. Different methodologies had been adopted with the passage of time. We will elaborate the different developments with respect to time and provide their advantages and issues. Services are getting more importance day by day in the form of its criticalness and provision in 24/7/365. Vendors and researchers are proposing different techniques and technologies to meet the futuristic needs to keep the services running all the time. Professor Nassem Nicholas has proposed the concept of antifragile in socioeconomic context. It has gain a lot of focus due to its resilient nature for provision of services in Cloud Computing [9]. There are three basic models that can be used in broadcasting the content to end user [10–12].

1. Client Server Streaming Model
2. Peer to Peer Streaming Model
3. HTTP Streaming over Cloud Model

Client Server Streaming Model: Due to the development and popularity of internet, concept of streaming videos over the internet has gained a lot of attention of researchers and professionals as well. Research in the field of offering new design and pattern of streaming protocols over internet got the boom along with providing Quality of Service (QoS) efficiently over the best-effort internet.

Transport Protocols for Video Streaming: In early 1990s, resource reservation protocols were replaced by new and simpler transport protocols which were not dependent on any way to reserve bandwidth in the best-effort method. To overcome packet lost issue and jitter while in its transmission in IP networks, Real-Time Protocol RTP was proposed. It has two basic principles, framing at application layer and processing of integrated layer [13]. It was based on UDP and Real-Time Control Protocol RTCP was also based on UDP. It is used to monitor the QoS statistics and synchronization patterns. RTP was adopted by many commercial and open source solution providers. It was also adopted by 3GPP. RTP/RTCP/RTSP to standardized this protocol suits specially for streaming over the internet with full functionality of VCR style playback, Stop, Forward, etc. It is also used to manage the sessions just like SIP (Session Initiation Protocol).

Rate Control and Rate Shaping: Rate Control was a technique planned to check the transmission rate of video stream based on estimated available bandwidth [14]. Rate shaping was to match the streaming rates to the target bandwidth [15].

Error Control: Packet loss can never be overlooked due to the best-effort nature of the internet. To overcome this issue, error control was proposed to ensure smooth streaming [16]. A lot of error control mechanisms were proposed in different time slots as follows:

(a) Forward Error Correction (FEC)
(b) Error Resilient Encoding
(c) Error Concealment
(d) Delay Constrained retransmission.

Proxy Caching: Caching techniques have also improved with the passage of time. There was a time when only server was enough to meet the requirements, but with the passage of time as the users increased, network got bigger and servers become inevitable to serve at multiple locations to meet the desired needs. With the development of heterogeneous network condition, caching patterns are also changed from one caching server to Sliding-Interval, Segment Caching, and rate-split caching [17].

IP Multicast for Video Streaming: With the increased number of sessions with the passage of time, uni-cast sessions were not practical with respect to the need. IP multicast was designed to serve the larger base of clients in 1990 [18]. IP multicast retained the semantics of IP and allowed users to dynamically join or leave the multicast group.

Application layer Multicast for Video Streaming: This approach was conceived due to the blocking of multicast at IP level from internet service providers. It has various security concerns [19]. The idea of using application layer protocol for multicast called application layer multicast ALM was proposed in 2000. Both of technologies require replication of packets and it was much easier to implement the multicast at application layer. It was shown that ALM performed well as compared with delay and bandwidth of IP multicast. To improve fairness in application layer multicast scheme, video streams can be split into multiple segments and distributed across network [20]. Yet the robustness problem remains unsolved.

Peer to Peer Video Streaming: Reciprocal to the client server streaming model where consumption and supply of resources are always decoupled, peer to peer streaming evolves in which peers are both suppliers and consumers of resources. It does not require support from underlying network hence is cost effective and easy to deploy. Secondly, in peer to peer design, host does not only download the files but also uploads the contents. It has a potential to grow larger in size. ALM is mainly push based, so the tree topology is usually established before starting the process. When we discuss the peer to peer design philosophy in torrents, it works on ALM and Pull-based technique on random mesh topology [21,22]. Normally, neighbor nodes exchange their information with each other and explicitly request the missing video buffer from their cache. One main concern of designing pull-based protocols is timing issue.

Traditional Pull-Based P2P Video Streaming: CoolStreaming [23] was the first pull-based streaming. Peers maintain a partial view of other peers and schedule the transmission of video segments by outgoing requests. The video is divided into segments of equal length. Active segments are presented in Buffer Map and these are exchanged with neighbor peers. One of the main advantages of Pull-Based streaming is its resilience to failures. A peer can leave gracefully or accidently due to unexpected failure.

HTTP Video Streaming Over the Cloud and Social Media: Although P2P streaming is highly efficient, it needs a standalone application installed on the client and needs some TCP/UDP port opened all the time. When the WebRTC was developed, these issues were resolved, but a solution to stream videos over HTTP had a support from the industry [24].

HTTP Streaming From Content Distribution Networks: Its major support comes from content distribution networks.

Multiple servers are deployed in several geographical locations distributed over multiple ISP. Users are allowed to stream the videos from the servers located near to his/her location. As a result, HTTP streaming and CDNs are effectively used for high-quality TV content [25].

Dynamic Adaptive Streaming Over HTTP: DASH framework is used by Netflix, which provides 29.7 % down streams in the US and is the largest DASH-Based streaming provider in the world.

Issues in DASH: Due to the best-effort nature of streaming, DASH was proposed to adapt the rates from the web servers. It was developed in 2010 and became new standard to enable high-quality streaming of media [26]. It works by breaking content into small segments and clients may adjust its requests according to the available bandwidth.

Segmentation: Video content is encoded and divided into multiple segments.

Media Presentation Segmentation: MPD, media presentation description defines how the segments will form a video.

Codec Implementation: DASH is codec agnostic and uses MP4 and MPEG-TS format. Seamless adoption of H265 is also supported in it.

Rate Adaptation Components: It only defined the segmentation and files' description and leaves the rate adaption either for client or server to implement.

Rate Adaptation Strategies: It defines how different versions of segmentations are received by the user to achieve the streaming stability, fairness and quality.

User Quality Experience: It has been observed with the survey analysis of users that they need gradual change between best and worst quality level instead of abrupt switching.

Video Streaming from Cloud: HTTP streaming is increasingly employed by major content providers. User-generated contents UGC has fundamentally changed the video streaming landscape. Highly centralized design of Data Centers has gained the popularity of Cloud Computing. As a result, design of network topologies in Data Centers has quickly become an active area of research which had introduced the concept of multimedia cloud computing. Cloud computing offers reliable, elastic and cost-effective resource provisioning. However, there are some issues with cloud computing, for example, Server's diverse capabilities and pricing, virtual machines and their corresponding charges are not so simple.

Antifragile and Resilient Cloud: Antifragility and resiliency was conceptualized on socioeconomic issues in its beginning, but researchers have proposed these features for cloud computing as well. A platform serving at such a large-scale should be capable enough to self-managed, self-maintained and self-created in some specific form. If any disaster occurs, infrastructure should be self-sustainable and manageable. Messaging servers are the main communicating entities between partially located clouds, if the services being offered from one site are turned down due to some reason, messaging servers will communicate accordingly and the services will be provided from different location. Meanwhile, messaging server will connect the damaged site to bring up the down serving resource itself within minimum possible time.

3 Problem statement

Cloud computing has emerged with lots of challenges and benefits. Work at large scale is being carried out in the field of Software defined networking, Virtualization and content delivery. Messaging brokers are also playing an important role in case when clouds are located in different geographical locations. We are proposing access of content with the messaging broker that will be connected with different clouds and will fetch the content from a server as per available bandwidth to the user. It will help to choose an adaptive bandwidth along with load balancing as well. It has been observed that many attempts have been done so far to broadcast the quality video over the lower bandwidth, but it is not ideal in case when a disaster occurs and network resources get congested. Normally, congestion occurs at mobile users who try to access the important information. Disaster area normally chocks with sudden, unexpected hits from other areas. It is mainly due to the downtime of pre-available resources.

4 Test bed description

We proposed rate adaptive and congestion aware content delivery mechanism that will best suit in case of disasters and would provide all the necessary information in case of congestion as well. Delivery of multimedia content will be made available as antifragile (self-aware, self-healed and self-maintained). Tests have been conducted using dynamic HTTP streaming with the adaptive bitrates model.

Table 1 describes the hardware components, specifications and vendor information while Table 2 describes the software details. 4x Dell 710 Servers were installed with Linux CentOS 7 to become the part of a Cloud at one site. Virtual servers for streaming and transcoding were created in the Cloud. Table 2 illustrates the applications and soft-

Table 1 Hardware used in test bed

Item	Description
Server	Dell 710
CPU	2.6 GHz Dual, Quad Core
RAM	16 GB
HDD	2 × 1 TB
NIC	Gbic Dual

Table 2 Software used in test bed

Item	Description
Operating System	Linux Debian, CentOS and Ubuntu
Hosting Application	Apache
Transcoding	Ffmpeg
Streaming	Apache
Hand Held Devices	BlackBerry, Apple iPhones, Android and Windows handsets
Cloud Application	OpenStack Juno
Messaging Server	ActiveMQ
Player	VLC, MX and FlowPlayer
AB.mov (480 × 360)	Video clip of 480 × 360 resolution size that has to be streamed across the network
CD.mov (640 × 360)	Video clip of 640 × 360 resolution size that has to be streamed across the network

ware installed and used for rate adaptive streaming. We tested these streams at different technological handsets like BlackBerry, Windows-based handsets, Android-based handsets, Apple and windows machines. OpenStack juno version was installed to work as a cloud.

Transcoding Server: We created virtual machine with Linux Debian 7, installed ffmpeg tool for transcoding.

Streaming Server: Virtual machine with Linux Debian 7, Apache 2.2 for hosting, H264 module integrated with Apache to serve videos requests.

Controller Node: Controller node maintains and controls the other physical machines in the cloud-like network node and compute node.

Network Node: Network node is used to maintain the virtual networking in Cloud and legacy network to connect with external network.

Compute Node: Compute node becomes the part of controller node to create virtual machines in it. 2x physical machines were given the role of compute node.

Figure 1 shows the physical infrastructure of servers. There is one Controller node, one Network node and 2 compute Nodes. Figure also shows a router to connect with legacy network, a storage server to store the content and layer-3 switch to connect with other servers if there are any. Figure 2 depicts the illustration of virtual infrastructure.

Figure 3 gives the brief of access mechanism of the user request. When a user will request some video content, it will cross check the available serving resource via messaging broker at two levels

1. Site-A is available, Streaming Server-A is available, video will be sent to user
2. Site-A is available but Streaming Server-A is down, messaging broker server will serve the request from Server-B in site-A and vice versa
3. Site-A is down, request will be routed to site-B
4. Site-A is down, Site-B is up with primary server-A, video will be sent to user
5. Site-A is down, Site-B is up with primary server-A is also down, messaging broker will connect the request with Server-B in site-B and request will be accomplished for the user.

4.1 Cloud infrastructure

Figure 4 presents the whole scenario of the content delivery to the user which may access the source of wired connectivity or from the mobile telecom operator. Transcoding servers are placed in secure and private zone, while streaming servers are placed in Demilitarized zone. Messaging broker servers are placed within the site as well to promote 2x level antifragility and resiliency level. It can shift the user's request in an interactive manner to other server if one server becomes problematic. For the security perspective, only assigned ports are opened for streaming server to receive the stream from the transcoding server. Mobile user requests from its telecom facility and this cycle gets completed from Telecom interfaces. We have made it possible to play the content at all possible available devices in the market. Mobile users of any type like BlackBerry, Android, and Windows are compatible to receive the video streams. 3G, 4G technologies are in market now with high provision of bandwidth. When

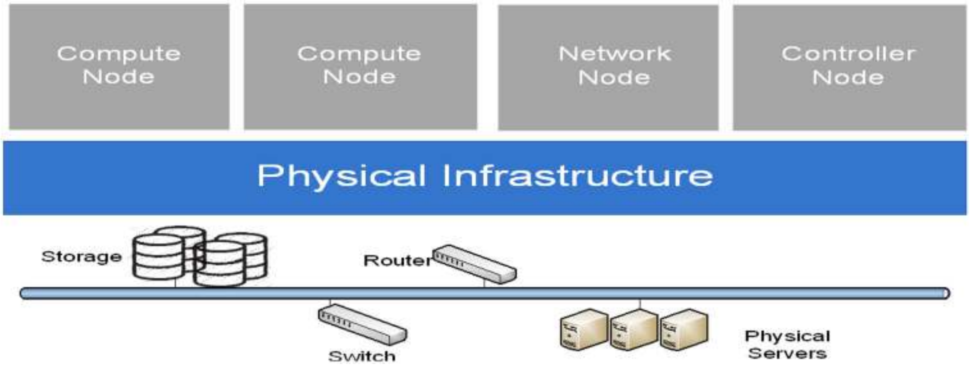

Fig. 1 Physical infrastructure of servers

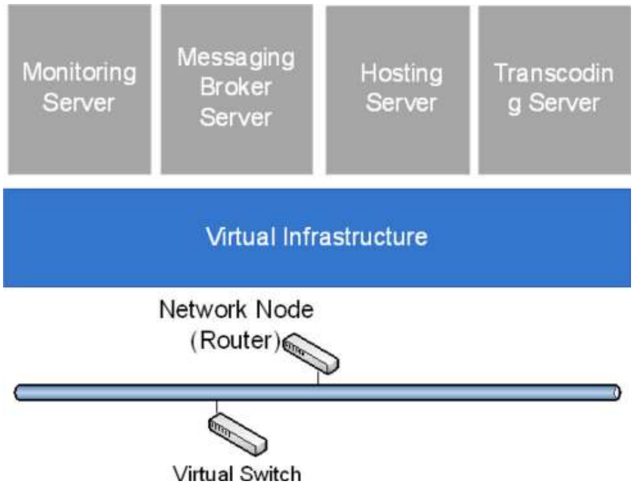

Fig. 2 Illustration of virtual infrastructure

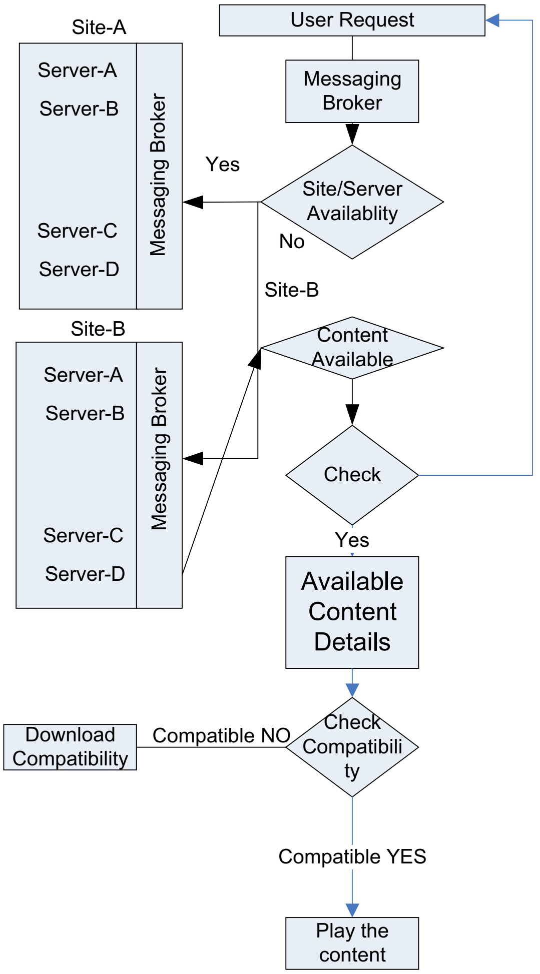

Fig. 3 Access mechanism of Multimedia Content for User

disaster hits some area, mobile networks get congested and people seeking the urgent information wait for a quite long time. We proposed very limited resource taking streaming method which would be able to perform well in such conditions as well.

Table 3 depicts the tested values and variable that may directly affect the quality and bandwidth of the deliverable content. Values of these variables are tested with command line tool ffmpeg in Linux environment.

5 Performance results

To begin with, original parameters of crf 18, qmin 10 and qmax 30 were used to perform the PSNR (Peak Signal Noise to Ratio) and SSIM (Structural Similarity Index Measure) calculations for two different reference files (AB and CD video clips). CRF (Constant rate factor) used to set the quality of H264 encoder, default value is 23, and we started from 18. Quantization factors “qmin” and “qmax” also support the quality of stream, using lower values for quantization factors will use more bandwidth. We started with 10 and 30. Experiments were performed on the file AB.mov and CD.mov shown below with sizes 480 × 360 and 640 × 360, respectively. PSNR and SSIM are calculated for these two videos with different sizes (Fig. 5).

B frames are bidirectional frames in compression that saves space between the most current frame and the following frame. CBR represents constant bitrates and VBR represents variable bitrates. *R* is number of frames per second. Final conclusion is based on the PSNR and SSIM. PSRN is peak signal to noise ratio and MSE (Mean Signal Error) is two error metrics; we used PSNR. SSIM (Structural Similarity Index) is a measure of image quality based on the initial uncompressed or distortion-free image as a reference.

Tests 1 and 5 are conducted with 12 number of frames per second, 85 kbps and bframes disabled and enabled. File sizes are mentioned in Table 4. Tests are repeated by increasing frames rate and decreasing bandwidth along with variable bandwidth which produces different files in sizes. Table 4 presents the values of PSNR plot. Table 5 presents the resulted values for SSIM and results are presented in table (Figs. 6, 7).

Table 6 represents the tests with crf 25, qmin 10 and qmax 30. The resulting file is smaller than the original file, but quality was compromised drastically. Tested values are with enabled and disabled B frames. The Graph shows PSRN plot (Figs. 8, 9, 10, 11, 12).

Table 7 shows the SSIM tested values with enabled and disabled B frames. Quality dropped drastically in this SSIM plot as well, although the file size was reduced.

The problem with these parameters was that the resulting files were larger in size than the reference file, in case of VBR encoding. That was due to the low crf value being used (crf 18 which is visually lossless). Choosing a higher

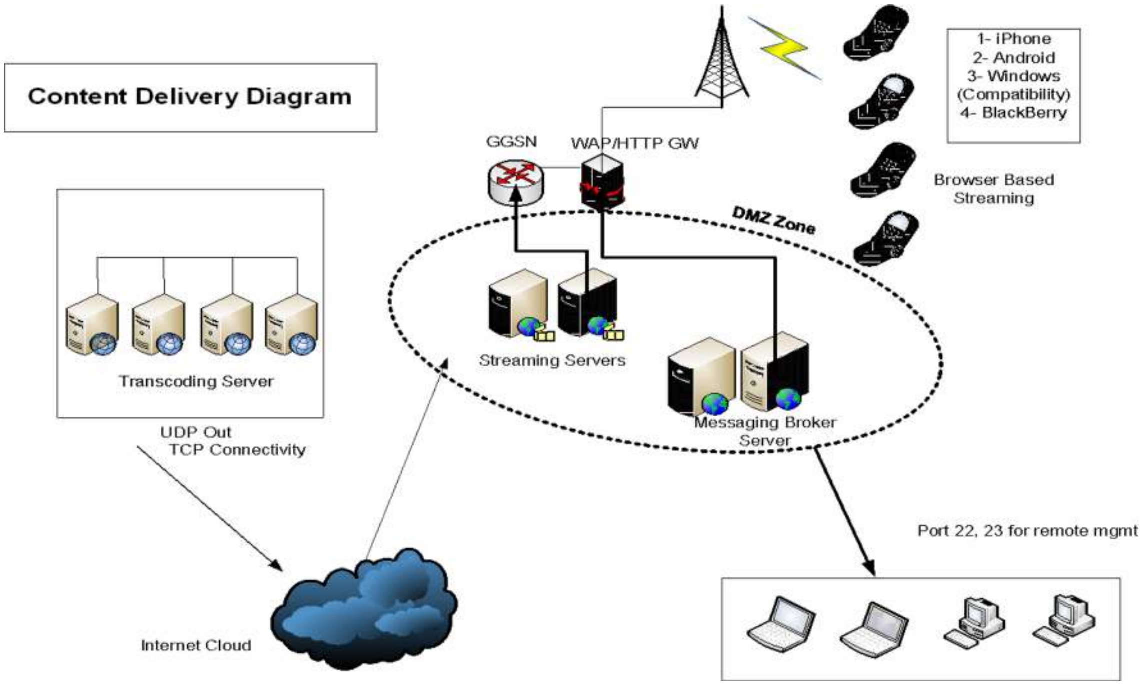

Fig. 4 Content Delivery Infrastructure

Table 3 Tested values and variables

Setting	Description	Adopted/reason
Key int	Key frame interval, GOP length, distance between I-frames	150, to contained the bandwidth
Profile	Provides compatibility for devices	V 3.0, works with almost every device
Trellis	Performs quantization to increase efficiency	1, for better compromise between speed and efficiency
Resolution	Screen Resolution	320 × 240 auto adjustable for different devices
Flags	Put global data in extra data instead of every key frame	Global_headers, best suited
Segment_list_flags	Outputs the streams into number of separate files of same duration	Live and HLS, To segment the content streams into chunks of TS
Bframes	For better efficiency in terms of compression ratio	Used in normal conditions with higher bandwidth and not used in congested conditions
Weightp	Improve compression in P-frames	0, not used
Cabac		Not used in baseline profile
Ref	Controls the size of DPB (decoded picture buffer)	1, keep the image quality sharp
Analyse	To control the split of H264 video	Default, to keep all values
No-fast-pskip	Disables early skip detection on P-frames	Default,
Video codec	Codec for video compression	Libx264
Audio Codec	Codec for Audio compression	AAC LATM mono
Segment_time	Time length of smaller segmented files	8 for recorded content and 4 for live both way communication
Segment_list_size	List variable	Any value to match the playlist
Hls_time	Time length for live stream	4
Hls_wrap	Total number of files to be created	6, in our scenario of live streams 6 files are created of 4s length

AB.mov - VLC media player
Media Playback Audio Video Tools View Help
AB.mov
AB.mov
1.00x
00:04/00:52

Fig. 5 AB.mov (file size 480 × 360)

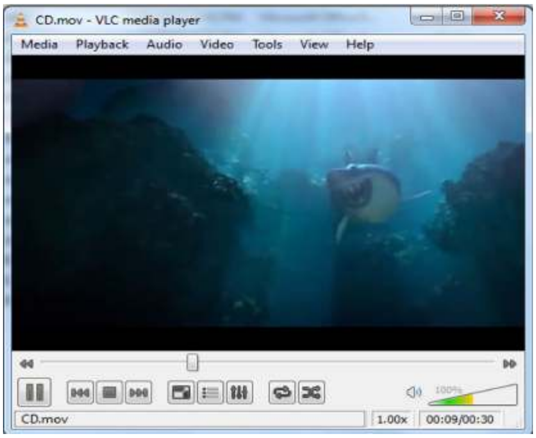
CD.mov - VLC media player
Media Playback Audio Video Tools View Help
CD.mov
1.00x
00:09/00:30

Fig. 6 CD.mov (file size 640 × 360)

CRF value results in reduced file sizes, whereas lower values would result in better quality (at the expense of higher file sizes). The default for x264 is 23. Hence, we tried different crf values, but the resulting file sizes did not reduce as significantly as expected, as the crf was changed from 18–25 (Table 9).

Finally, we changed the quantization parameters (qmin 10 × 0 and qmax 30 × 51) as well. The resulting files had considerably smaller sizes with improved objective quality (what a PSNR calculation sees); however, the subjective quality (what the human eye sees) was compromised. Lower qmin and qmax values generally mean ‘better quality’. By setting the bounds from 0 to 51, the encoder has free choice of how to assign the quality but this requires a bit of tweaking. Further refining of qmax and qmin parameters led to improved subjective quality and reduced file sizes. The results are given below.

To achieve the rate adaptive feature in case of disaster OR congestion, its antifragile nature will become active and streams bandwidth rate will be decreased automatically to the available bandwidth to the user. It is achieved by doing the segmentation of the recorded multimedia content. Four to five different types of segmentation have been put in server to best match with the available bandwidth ranging from 240, 180, 120, 100 and 72 kbps. This is achieved by adapting the TS segmentation and m3u8 playlist.

M3u8 is a Unicode file format which contains the file list to be played and TS is a container format which will contain the playable audios and videos for the user.

In our solution, we have divided the video clips into the duration of 4 s and segmented it with respect to our different offered bandwidth solutions depending on the user experience and available bandwidth to the user. If video is of 1 min, we segmented it into 15 files of 4 s, each totaling 75 for 5 offered bandwidth rates. Streaming server will provide the video to the user as per the network available resources. Stream could go as lower as till 72 kbps that works pretty good in pre-available G standard in telecom sector.

Table 4 PSNR test cases for AB.mov

bframes disabled	bframes enabled
1. Test 1: CBR r 12 b 85 k—file size 1.98 MB	5. Test 5: CBR r 12 b 85 k—file size 1.96 MB
2. Test 2: CBR r 24 b 55 k—file size 1.73 MB	6. Test 6: CBR r 24 b 55 k—file size 1.72 MB
3. Test 3: VBR r 12 b 85 k—file size 3.70 MB	7. Test 7: VBR r 12 b 85 k—file size 3.62 MB
4. Test 4: VBR r 24 b 55 k—file size 4.95 MB	8. Test 8: VBR r 24 b 55 k—file size 4.67 MB

Table 5 SSIM test cases for AB.mov

bframes disabled	bframes enabled
1. Test 1: CBR r 12 b 85 k—file size 1.98 MB	5. Test 5: CBR r 12 b 85 k—file size 1.96 MB
2. Test 2: CBR r 24 b 55 k—file size 1.73 MB	6. Test 6: CBR r 24 b 55 k—file size 1.72 MB
3. Test 3: VBR r 12 b 85 k—file size 3.70 MB	7. Test 7: VBR r 12 b 85 k—file size 3.62 MB
4. Test 4: VBR r 24 b 55 k—file size 4.95 MB	8. Test 8: VBR r 24 b 55 k—file size 4.67 MB

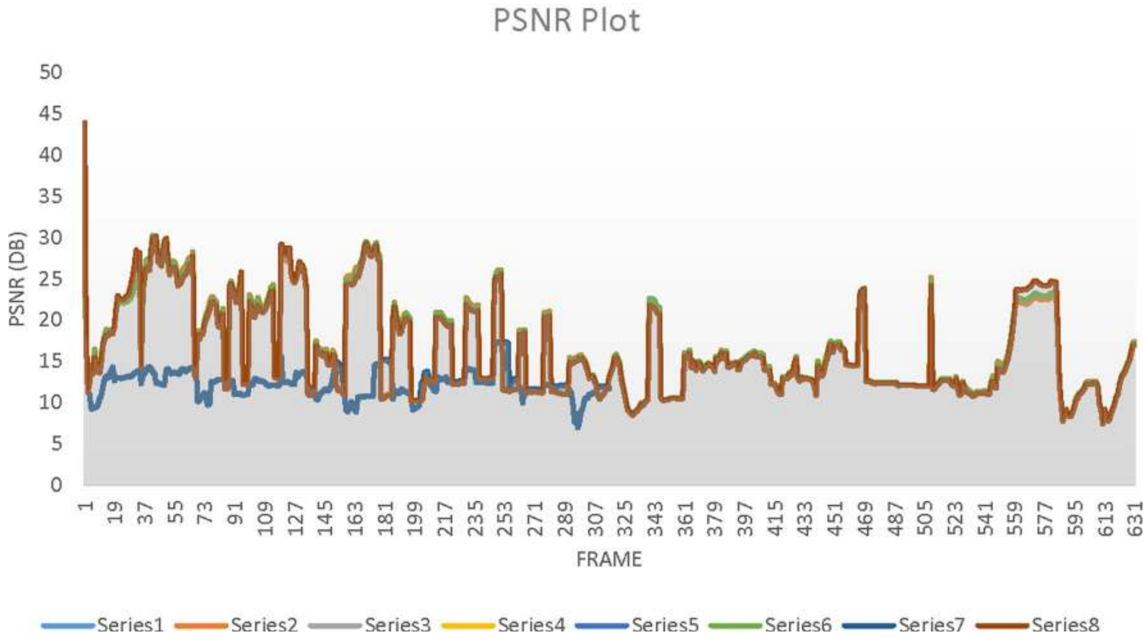

Fig. 7 PSNR of AB.mov for different test cases

Table 6 PSNR test cases for CD.mov

bframes disabled	bframes enabled
1. Test 1: CBR r 12 b 85 k—file size 699 KB	5. Test 5: CBR r 12 b 85 k—file size 684 KB
2. Test 2: CBR r 24 b 55 k—file size 595 KB	6. Test 6: CBR r 24 b 55 k—file size 586 KB
3. Test 3: VBR r 12 b 85 k—file size 3.04 MB	7. Test 7: VBR r 12 b 85 k—file size 3.01 MB
4. Test 4: VBR r 24 b 55 k—file size 3.32 MB	8. Test 8: VBR r 24 b 55 k—file size 3.32 MB

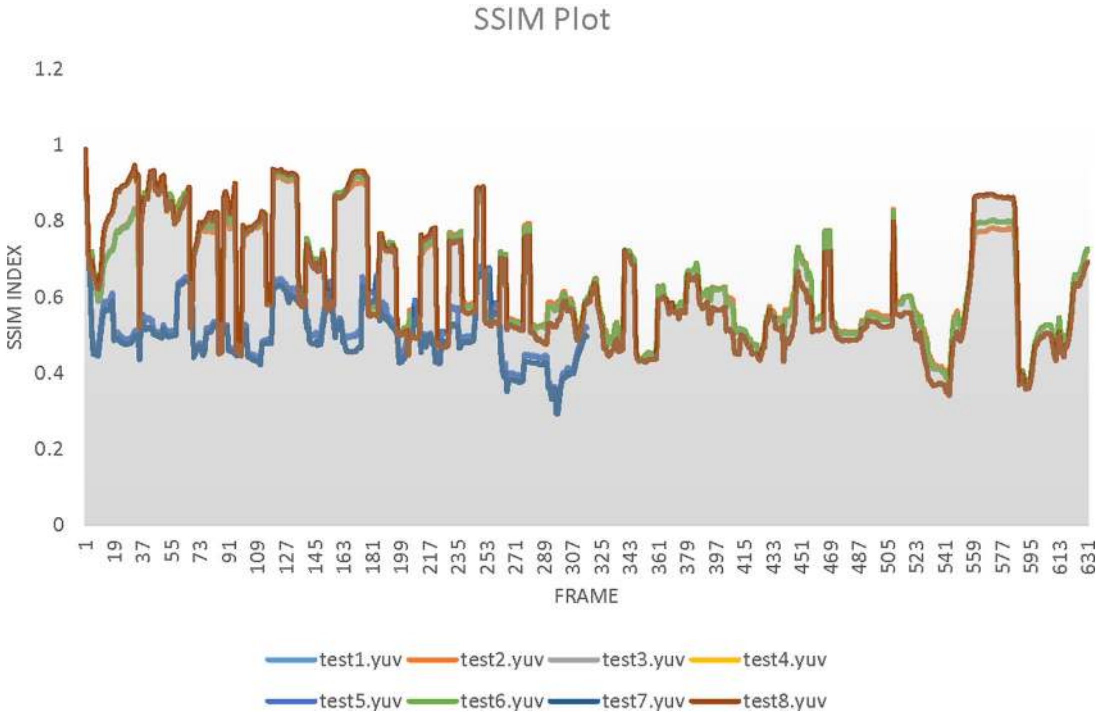

Fig. 8 SSIM of AB.mov for different test cases

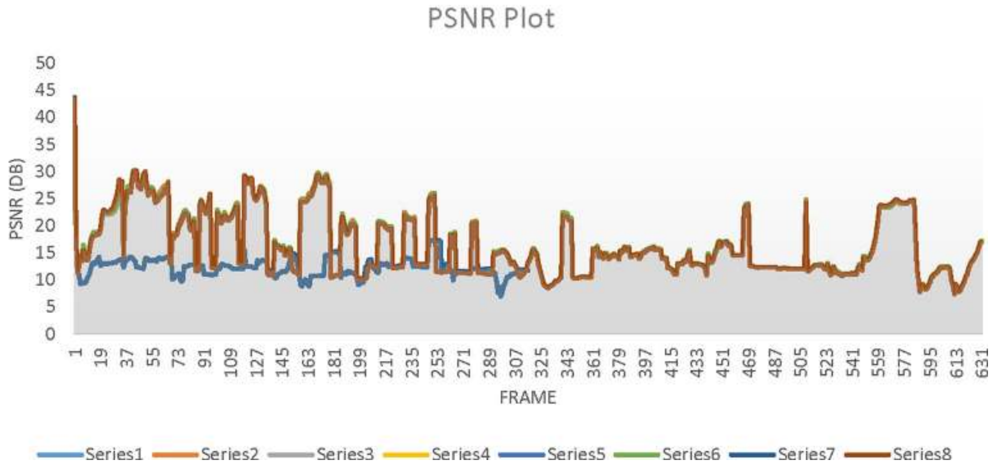

Fig. 9 PSNR of CD.mov for different test cases

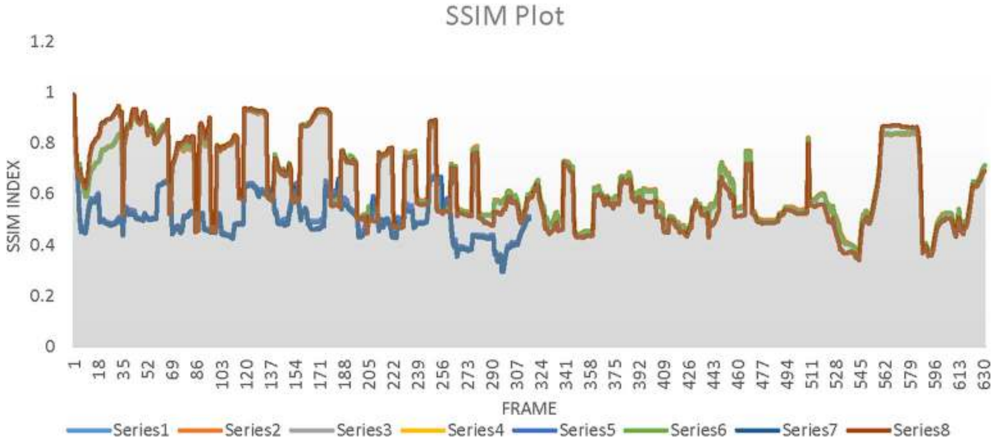

Fig. 10 SSIM of CD.mov for different test cases

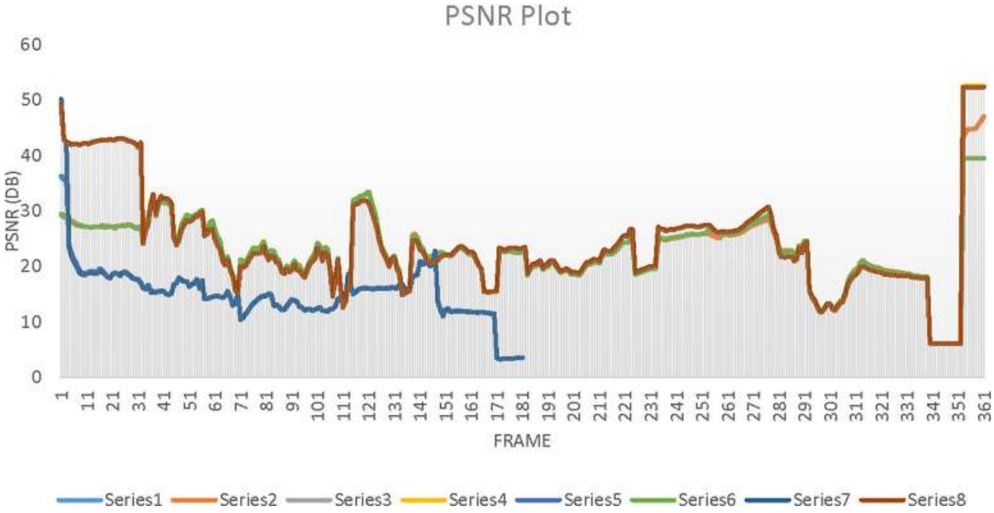

Fig. 11 PSNR for AB.mov for different test cases (reference Table 8)

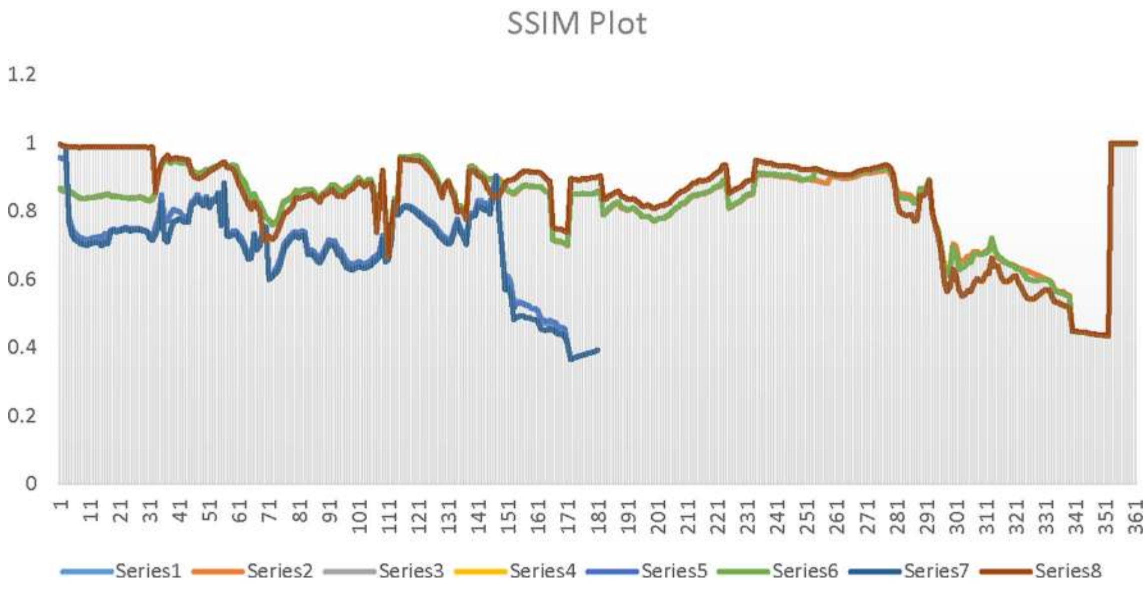

Fig. 12 SSIM for AB.mov for different test cases (reference Table 8)

Table 7 SSIM test cases for CD.mov

bframes disabled	bframes enabled
1. Test 1: CBR r 12 b 85 k—file size 699 KB	5. Test 5: CBR r 12 b 85 k—file size 684 KB
2. Test 2: CBR r 24 b 55 k—file size 595 KB	6. Test 6: CBR r 24 b 55 k—file size 586 KB
3. Test 3: VBR r 12 b 85 k—file size 3.04 MB	7. Test 7: VBR r 12 b 85 k—file size 3.01 MB
4. Test 4: VBR r 24 b 55 k—file size 3.32 MB	8. Test 8: VBR r 24 b 55 k—file size 3.32 MB

Table 8 PSNR test cases for AB.mov (file size 480 × 360)

bframes disabled	bframes enabled
1. Test 1: CBR r 12 b 85 k—file size 1.98 MB	5. Test 5: CBR r 12 b 85 k—file size 1.96 MB
2. Test 2: CBR r 24 b 55 k—file size 1.73 MB	6. Test 6: CBR r 24 b 55 k—file size 1.72 MB
3. Test 3: VBR r 12 b 85 k—file size 3.02 MB	7. Test 7: VBR r 12 b 85 k—file size 2.96 MB
4. Test 4: VBR r 24 b 55 k—file size 3.44 MB	8. Test 8: VBR r 24 b 55 k—file size 3.33 MB

Table 9 SSIM test AB.mov (file size 480 × 360)

bframes disabled	bframes enabled
1. Test 1: CBR r 12 b 85 k—file size 1.98 MB	5. Test 5: CBR r 12 b 85 k—file size 1.96 MB
2. Test 2: CBR r 24 b 55 k—file size 1.73 MB	6. Test 6: CBR r 24 b 55 k—file size 1.72 MB
3. Test 3: VBR r 12 b 85 k—file size 3.02 MB	7. Test 7: VBR r 12 b 85 k—file size 2.96 MB
4. Test 4: VBR r 24 b 55 k—file size 3.44 MB	8. Test 8: VBR r 24 b 55 k—file size 3.33 MB

6 Conclusion

In this paper, we tested and optimized video streams in cloud environment with HTTP-Dash streaming mechanism, segmentation, rate-adaption mechanism and rate-adaption strategy. Streams are optimized at different bit rates. In normal conditions, bit rates are optimized at 150–240 kbps with higher quality content delivery but as the congestion grows during disaster management and recovery operation, it will be lowered at 72 kbps to adapt to the available limited bandwidth resources, thus provide the vital information to the customers even when the circumstances are vulnerable. If service from one server gets down, resilient and antifragile nature of the stream will make sure the delivery of content with the help of messaging brokers. Different antifragile and resilient techniques were used which prove the better performance with the help of message brokers

References

1. Iqbal M, Wang X, Wertheim D, Zhou X (2009) SwanMesh: a multicast enabled dual-radio wireless mesh network for emergency and disaster recovery services. J Commun 4(5):298–306
2. Iqbal M, Wang X, Li S, Ellis T (2010) QoS scheme for multimedia multicast communications over wireless mesh networks. IET Commun 4(11):1312–1324
3. Iqbal M, Wang X, Wertheim D (2010) Reliable multimedia multicast communications over wireless mesh networks. IET Commun 4(11):1288–1299
4. Arney C (2013) Antifragile: things that gain from disorder. Math Comput Educ 47(3):238
5. Linebaugh P, Rediker M (2013) The Many-Headed Hydra: Sailors, Slaves, Commoners, and the Hidden History of the Revolutionary Atlantic. Beacon Press. ISBN 978-0807033173
6. Li L, Li X, Youxia S, Wen L (2010) Research on mobile multimedia broadcasting service integration based on cloud computing. In: 2010 International conference on multimedia technology (ICMT). IEEE, pp 1–4
7. Li B, Wang Z, Liu J, Zhu W (2013) Two decades of internet video streaming: a retrospective view. In: ACM transactions on multimedia computing, communications, and applications (TOMM), vol 9(1s), 33
8. Kalva H, Adzic V, Furht B (2012) Comparing MPEG AVC and SVC for adaptive HTTP streaming. In: 2012 IEEE international conference on consumer electronics (ICCE). IEEE, pp 158–159
9. Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I (2009) Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility. Future Gener Comput Syst 25(6):599–616
10. Yu FR, Zhang X, Leung VC (eds) (2012) Green communications and networking. CRC Press, Boca Raton
11. Zhang S, Zhang S, Chen X, Huo X (2010) Cloud computing research and development trend. In: Second international conference on future networks, 2010. ICFN'10. IEEE, pp 93–97
12. Yanhong Z (2009) Research of cloud computing multi-integration services in the mobile environment. Telecom Eng Tech Stand 11:005
13. Schulzrinne H, Rao A, Lanphier R. Real Time Streaming Protocol 2.0 (RTSP). https://tools.ietf.org/html/draft-ietf-mmusic-rfc2326bis-40. Retrieved 1 Nov 2015
14. Nahrstedt K, Yang Z, Wu W, Arefin A, Rivas R (2011) Next generation session management for 3D teleimmersive interactive environments. Multimed Tools Appl 51(2):593–623
15. Jacobs S, Eleftheriadis A (1998) Streaming video using dynamic rate shaping and TCP congestion control. J Vis Commun Image Represent 9(3):211–222
16. Wang Y, Zhu QF (1998) Error control and concealment for video communication: a review. Proc IEEE 86(5):974–997
17. Liu J, Xu J (2004) Proxy caching for media streaming over the Internet. Commun Mag IEEE 42(8):88–94
18. Deering SE (1990) Multicast routing in internetworks and extended LANs. J ACM Trans Comput Syst 8(2):85–110
19. Diot C, Levine BN, Lyles B, Kassem H, Balensiefen D (2000) Deployment issues for the IP multicast service and architecture. Netw IEEE 14(1):78–88
20. Castro M, Druschel P, Kermarrec AM, Nandi A, Rowstron A, Singh A (2003). SplitStream: high-bandwidth multicast in cooperative environments. In: ACM SIGOPS operating systems review, vol 37, no 5. ACM, pp 298–313
21. Pinto D, Barán B (2005) Solving multiobjective multicast routing problem with a new ant colony optimization approach. In: Proceedings of the 3rd international IFIP/ACM Latin American conference on networking. ACM, pp 11–19
22. Zhang M, Luo JG, Zhao L, Yang SQ (2005) A peer-to-peer network for live media streaming using a push-pull approach. In: Proceedings of the 13th annual ACM international conference on multimedia. ACM, pp 287–290
23. Zhang X, Liu J, Li BC, CoolStreaming YYP (2005) DONet: a data-driven overlay network for peer-to-peer live media streaming [A]. INFOCOM 2005. In: 24th Annual joint conference of the IEEE computer and communications societies. Proceedings IEEE, vol 3, pp 2102–2111
24. Bergkvist A, Burnett D, Jennings C, Narayanan A (2012) WebRTC 1.0: real-time communication between browsers. World Wide Web Consortium WD WD-webrtc-20120821
25. Cahill AJ, Sreenan CJ (2004) An efficient CDN placement algorithm for the delivery of high-quality tv content. In: Proceedings of the 12th annual ACM international conference on multimedia. ACM, pp 975–976
26. Stockhammer T (2011) Dynamic adaptive streaming over HTTP–: standards and design principles. In: Proceedings of the second annual ACM conference on multimedia systems. ACM, pp 133–144