

SIP and SDP based content adaptation during real-time video streaming in Future Internets

M. Ramakrishna¹  · A. K. Karunakar²

Received: 8 January 2016 / Revised: 17 September 2016 / Accepted: 27 September 2016 /
Published online: 22 October 2016
© Springer Science+Business Media New York 2016

Abstract Future Internet technology supports content adaptation to improve the QoS / QoE in a heterogeneous environment. The adaptation process removes partial data to meet either receiver capability or network capability. The process of adaptation is efficient when it is performed in the network rather than in the sender side or receiver end. In-network adaptation is performed by intelligent intermediate devices, which implement Content Aware Network and Content Centric Network, to process the contents. The adaptation decision taking module requires prior knowledge about end devices, network capacity and content's meta-data to decide the extraction points. This paper proposes a model to signal the terminal, network and media capabilities with the intermediate devices. The proposed model reduces the number of messages exchanged between end devices, intermediate devices and media server during session creation. Additionally, this method proposes a way to signal network dynamics such as change in bandwidth and buffer space with the intermediate devices to improve the video quality with the available resources. The proposed session signaling scheme is developed by extending traditional Session Initiation Protocol and Session Description Protocol and tested for suitability using virtual networking environment.

Keywords Video streaming · Scalable video coding · Video adaptation · Future internets · Media aware network elements · Session establishment · Software defined networks

✉ A. K. Karunakar
karunakar.ak@manipal.edu

¹ Department of Information and Communication Technology, Manipal Institute of Technology, Manipal University, Manipal, 576104, India

² Department of Computer Applications, Manipal Institute of Technology, Manipal University, Manipal, 576104, India

1 Introduction

Nowadays, video based communications are involved in almost all human activities such as meeting, chatting, entertainment, medical treatments, and many more. These activities mainly use the Internet as a media for transmitting the user data. Most commonly used multimedia applications over the Internet are video conferencing, video-on-demand, IPTV, telemedicine, etc. In such applications, user produces and transmits the media contents over the network to another user, who is acting as a consumer. The comfort in video communication increases with, quality of the video, reduced streaming delay and seamless transmission. However in these applications, the devices used are heterogeneous in display, processing and streaming capacities. In the same way, network used also has bandwidth limitation and packet processing issues. In addition to these constraints, video formats used by the devices also vary from device to device. This heterogeneity in device capabilities leads to quality degradation and hence poor user satisfaction. These factors influence the overall QoS and QoE of video communication over the Internet. Majority of the challenges involved in streaming video over heterogeneous environment are well handled by the layered video coding, where partial removal of contents from valid video bit-stream is done. Adapting video to a required level, serves the heterogeneous device requirements [39]. Scalable Video Coding (SVC) is one such layered coding technique and an extension to H.264/AVC [33].

Current Internet supports, receiver and sender driven adaptation methods at end devices and server side respectively. In receiver-driven approach, the content is adapted by the receiving device just before displaying it. In sender-driven method, user signals the device capabilities while creating the session, accordingly sender adapts the content and streams the adapted content over the Internet. Both approaches are suitable for one-to-one communication than the multicast communication, where multiple receiving devices and quality requirements make the adaptation and streaming complex. This can be simplified by performing the adaptation within the network i.e. at intermediate devices to reduce the bandwidth consumption. Also, in-network adaptation [14, 23] reduce processing overhead of both, sender and receiver and it helps in improving quality of the video contents with network dynamics [18]. This can be realized only when intermediate devices are intelligent to adapt the contents based on terminal and network capabilities, resource availability and network conditions. Such intelligent intermediate devices are proposed in the Future Internet Architecture (FIA) [22], where devices are capable of processing and routing the packets based on the contents flow through it.

Future Internet Architecture (FIA) implements Content Aware Networking (CAN) and Content Centric Networking (CCN) methods [4, 13]. These enable intermediate devices with intelligent features such as adaptation and content based routing. The Media Aware Network Element (MANE) [1, 15] is a proposed intelligent intermediate device in FIA for multimedia communication. MANE decides the extraction points and removes the unwanted layers to meet the requirements of heterogeneous user devices and networks. Terminal properties such as display resolution, frame rate support, processing capacity, power availability and region of interest and network resource details such as bandwidth, maximum and minimum bitrate, helps in deciding the number of layers that need to be delivered to an end device. Similarly, content properties like number of scalable layers and layer dependency describe the video content. Hence, prior knowledge of terminal capacities, media

properties and network capabilities need to be made available at decision taking module to decide the number of layers to be extracted.

Currently available methods [2, 32, 38] for video adaptation and streaming are using either embedded or static semantics for describing the meta-data, which are not suitable for a real-time application having dynamic change in the resource requirements. Communication in the current system starts with session initiation and is followed by meta-data sharing and video streaming. During session initiation phase, the communicating parties exchange packets of Session Initiation Protocol (SIP) [31] and Session Description Protocol (SDP) [16], then create the sessions for each peer in the case of multicast scenario. Later, meta-data of the contents are shared with the intermediate devices, where adaptation takes place through the established connection. Hence, meta-data remains static for a session. If at all meta-data needs update, then the session has to be re-created, which leads to quality degradation and makes available methods not suitable for real-time communications. In addition, resource management becomes problematic, as multiple sessions need to be maintained by the media server, where multiple connections are maintained for the same content streaming. This motivated us to come up with a new signaling technique to signal the static and dynamic meta-data for a multicast session.

This paper proposes a model to exchange terminal and network capabilities and video meta-data with the intermediate devices. This model extends available features of SIP and SDP. The media description attribute of SDP is used for customization, where user can describe about media, receiving capabilities and network resource requirements for a session. SIP supports negotiation of resources required in the network and end devices before committing a session.

The paper is organized as follows: brief background is provided in Sections 2 and 3 discusses the literature, Section 4 describes proposed model, Section 5 overviews of our implementation. The experimental setup and results are detailed in Section 6. Finally, Section 7 presents conclusion and briefly mentions our future plans.

2 Background

Brief overview on Layered video coding, Future Internet and Session creation using SIP and SDP are presented in this section. These concepts are the base for our proposed model. The readers familiar with these may prefer to continue reading Section 3.

2.1 Layered video coding - scalable video coding (SVC)

The video can be coded using multiple description coding and layered coding. In Multiple Description Coding (MDC) [6, 41], each description guarantees a basic level of reconstruction quality of the source media content and every additional description can further improve the quality. Layered video coding, encodes the video in the form of one base layer and multiple enhancement layers. The base layer provides basic level of video quality and further refinement can be done using enhancement layers.

SVC is one of the layered video coding techniques that provides adaptation ready video data, which addresses the challenges in having common video formats for heterogeneous receiving devices. It reduces the bandwidth consumption unlike streaming multiple video

format having same video contents over the network. Also, network dynamics will have less impact on video quality due to in-network on the fly video adaptation [24].

SVC generates a base and many enhancement layers to meet multiple frame rates, resolutions and quality (SNR) levels of heterogeneous devices and network capabilities. It makes use of temporal, spatial and quality scalability modes to generate multiple layers. The base layer is coded independently and each enhancement layer is coded in reference to previous layers. Therefore, only a single bit-stream is generated but parts of it can be extracted in such a way that the resulting sub-stream forms another valid bit-stream for a given decoder, as shown in Fig. 1.

The extractor is an intermediate device to perform adaptation [36]. Adaptation Decision Module (ADM) and Extraction Module are main components of an extractor. ADM decides the extraction points based on the end device and network capability information available at the extractor. Accordingly, extractor removes the scalable layer to form adapted video bit-stream. The extractor is an intelligent networking device, it acquires dynamic network conditions and availability of the network resources while taking the decision. Finally, adapted video bit-stream is transmitted over the network. To ease the network transmission and adaptation, SVC uses packetization through Network Abstraction Layer (NAL) [42], which generates NAL units having header and video data. The header shares the scalable video details such as temporal, spatial, and quality level, dependency information, profile and level metrics. These are used by the ADM and extractor to identify and extract the unwanted scalable video layers.

The latter said features of SVC make it a natural choice for multicasting the video data in a heterogeneous environment. Additionally, one video format for all types of receiving devices solve the problems of having multiple video formats such as storage space and multiple encoders at media server.

2.2 Future internet architecture

In general, the Internet is becoming CCN and CAN [29], which enable new services, scalable and trusted multimedia content delivery, enriching the QoS in the network and terminals. In CAN, the devices are intelligent to differentiate the communication based

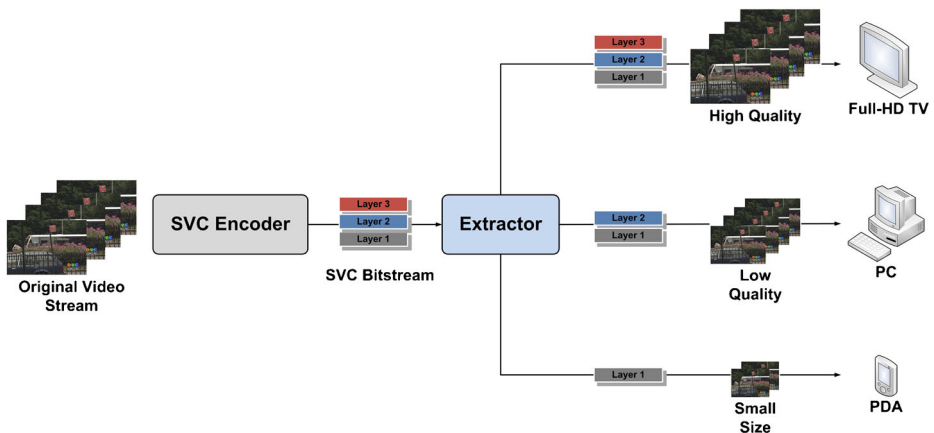


Fig. 1 Scalable video adaptation for heterogeneous environment [27]

on the content [17]. When content is audio visual, then these intermediate devices should assign high priority based on the type of multimedia communication such as video streaming, video conferencing, live streaming and telemedicine. The CCN has evolved to make the data independent of both logical address (Ex. IP address) and location. Here, routing devices route the packets based on the content. FIA is a new paradigm to implement CCN and CAN, and improve efficiency and effectiveness of the communication in the network [4].

Layered model of FIA is demonstrated in Fig. 2. The infrastructure layer is the physical network consisting of interconnecting devices with limited intelligence and functionality. The users, who produce and consume (prosumer) contents are connected to infrastructure layer. Distributed Content/ Service Aware overlay includes Content Aware nodes, which are intelligent to process the contents that flow through them. These devices can monitor quality of the contents and have the capacity to improve the quality by processing the contents. The information related to content, consumer and producer can be stored and used locally for improving the QoS/QoE. The Information/Service overlay is used for content indexing, content adaptation, caching and optimal delivery. This layer is created dynamically based upon the application and services. The highest application overlay implements network applications such as multimedia communication, telemedicine and web services.

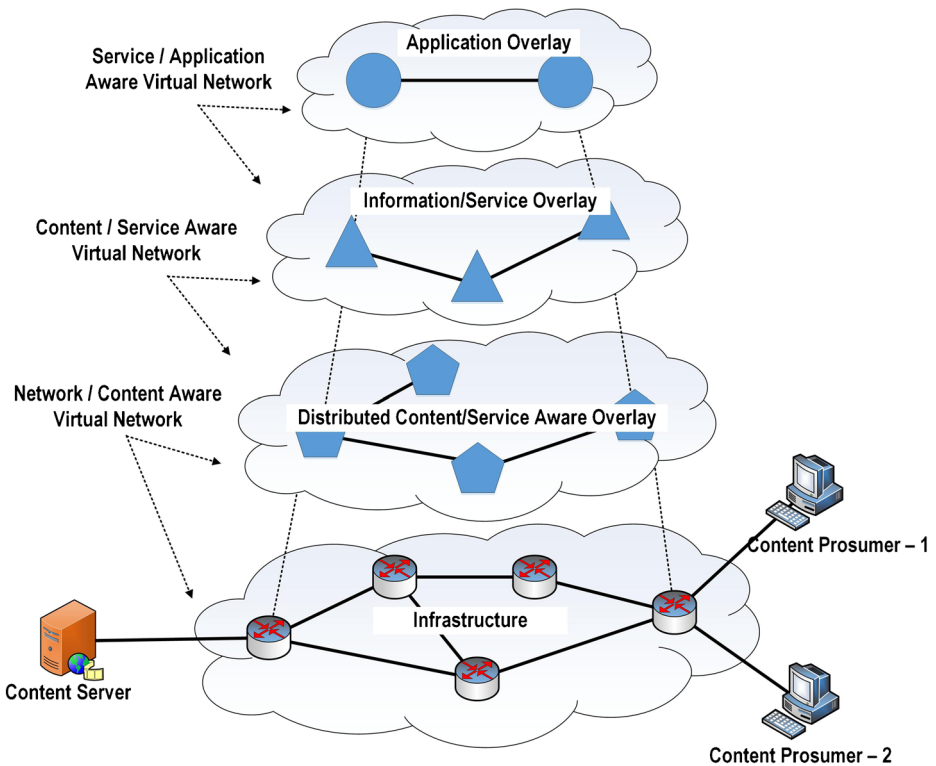


Fig. 2 Future Content Centric Internet Architecture [29]

MANEs [1] are CAN-enabled routers and associated managers, offering content-aware and context-aware Quality of Service/Experience, security and monitoring features, in cooperation with other elements of FIA. In general, content-aware MANEs can offer multimedia storage, dynamic content adaptation and dynamically combining multiple multimedia contents from various sources. Moreover, information of the underlying network conditions/characteristics can be utilized by cross-layer control modules to adapt the multimedia streams in the delivery path.

The video data coded using SVC can be transmitted effectively over FIA, where devices such as MANE adapt the contents on the fly. As CCN and CAN are enabled in the intermediate devices, they can access the SVC NAL units for fetching the content details and deliver the content according to the context.

2.3 Session creation using SIP and SDP

The Session Initiation Protocol (SIP), is an application-layer control (signaling) protocol for creating, modifying and terminating the session with one or more participants. SIP invitations are used to create sessions and carry session descriptions that allow participants to agree on a set of compatible media types. SIP makes use of intermediate device called proxy server to route the session requests to the user's current location, authenticate and authorize users for different services, implement provider call-routing policies and provide features to users such as registration, redirect services and access control. The SIP messages can be encapsulated in UDP, TCP, SCTP, IPv4 and IPv6 to transmit over the Internet.

The SIP eases signaling and data transmission by separating them into control and data plane respectively. SIP signaling and data packets are transmitted as shown in the Fig. 3. Signaling packets traverse through control-plane and locate the participating peers. Later, connection is established through data-plane. Control-plane consists of SIP proxy servers, which forward the session initiation through the paths and maintains all the devices registered for SIP communication. The Data plane is an underlying IP network, hence IP routing takes place for forwarding the data from source to destination.

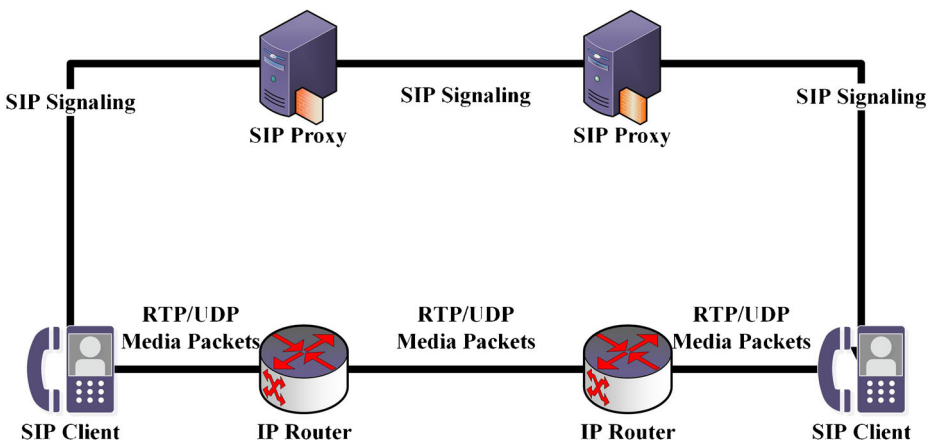


Fig. 3 SIP Signaling and Media Data Transmission [31]

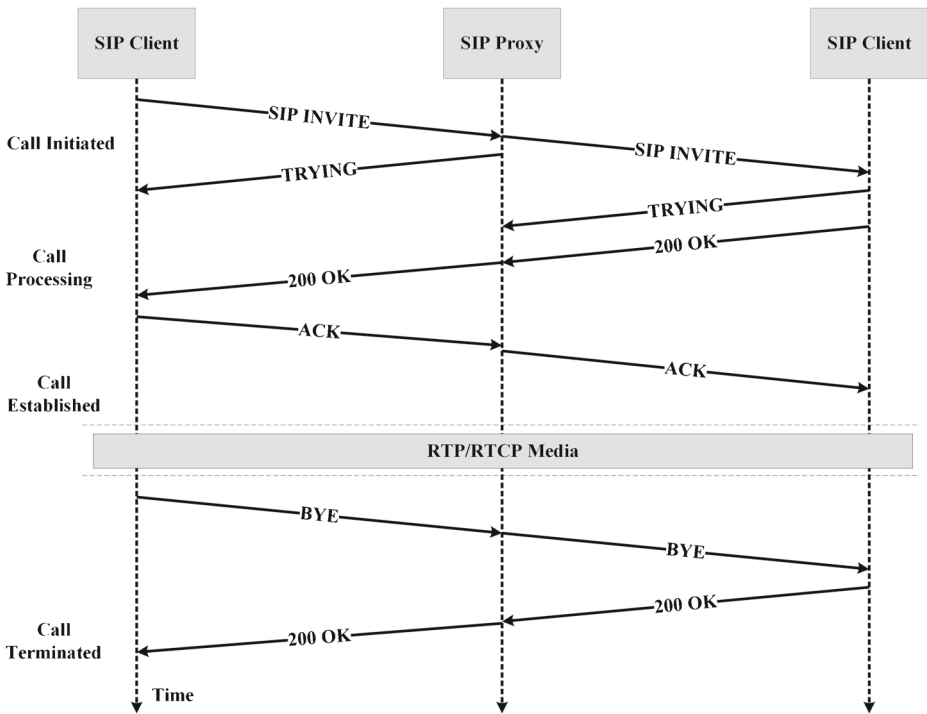


Fig. 4 SIP Call Flow [31]

Figure 4 illustrates call flow used by the SIP to signal creation and termination of a session. Initially, the session creator generates SIP INVITE message, where the session and content requested is described using the Session Description Protocol (SDP). SIP Proxy server is an intermediate device to route the packet towards receiver. Proxy server locates the receiving devices, calculates the end-to-end path and forwards the messages along the path. Receiver acknowledges the invitation with 200 OK message to indicate that message has been delivered at the destination successfully if available, else signals ‘busy’ back to the session creator. After creating the session, actual media transmission takes place between SIP parties. For updating the session details, SIP UPDATE message is used. Finally, termination of the session can be initiated by any participant, where BYE message signals terminate the procedure.

SIP conveys available multimedia session and description of the media session through SDP to all devices in the network. The purpose of SDP is to detail and negotiate the parameters while announcing and update the session. The description is divided into three sections - session, timing and media description. The structure of the SDP message is as shown in Table 1. Server describes the media content and resources required for joining a session and participating peers use these descriptions to take decision on joining the session. The section session, describes user information such as session name, ID and originators contact details and address. Time section shares starting time, stopping time and duration of the

Table 1 SDP message format [31]

 Session description:

v = (protocol version number, currently only 0)
 o = (originator, id, version number, network address)
 s = (session name)
 i =* (session title or short information)
 u =* (URI of description)
 e =* (zero or more email address)
 p =* (zero or more phone number)
 c =* (connection information)

One or more Time descriptions ("t=" and "r=" lines; see below)

z =* (time zone adjustments)
 k =* (encryption key)
 a =* (zero or more session attribute lines)

Zero or more Media descriptions (each one starting by an "m=" line)

Time description (mandatory):

t = (time the session is active)
 r =* (zero or more repeat times)

Media description:

m = (media name and transport address)
 i =* (media title or information field)
 c =* (connection information)
 k =* (encryption key)
 a =* (zero or more media attribute lines)

session with the participating peers. The media section details the properties of media data such as IP address, port number, protocol and codec used.

Attribute lines in session and media section allows user to define additional details related to media contents, session resource and network requirements. This helps in extending the core protocol to meet different multimedia applications. The attribute information improve the resource allocation at participating peers by describing the requirements in detail.

3 Literature Survey

There are few notable methods, which are proposed in literature on signaling network properties. A special packet named Interest Packet is proposed in [21] to describe the content. The packet uses semantic technique to reduce the packet size and signal the meta-data. This method limits the application to signal only content descriptions. The federated network management is the new paradigm for managing the networks in Future Internet [9, 19, 34]. In these papers, the need of inter-domain networking, network infrastructure coordination, service monitoring and dynamic agreements to improve the end-to-end communications

are explained. The articles [10, 25] propose methods to negotiate the network capabilities for a media communication. Once the resources are allocated, the media streamed and the resource allocated cannot be altered dynamically. A cache based dynamic signaling method is developed in [11], which affects the quality of video while updating the caches in intermediate devices. Similarly, a high level program abstraction based agent is used for sharing the network information in [12], where agents are used for exchanging the meta-data.

A considerable amount of work has been carried in the area of Dynamic Adaptive Streaming over HTTP (DASH) [35]. The in-network adaptation in CCN [26, 30] is developed on the basis of DASH. Here, XML-based Media Presentation description is used to signal the Uniform Resource Identifier (URI) of media to the clients. The URI locates media having different bitrate, resolution and codec characteristics. Clients stream the media based on their capabilities. The MPEG-21 Digital Item Adaptation (DIA) [5, 38, 40] is a framework proposed to share the content meta-data through out-of band signaling technique. This method is extended to signal video meta-data and source specific details in [3, 7, 8, 27, 37]. The main problem with MPEG-21 DIA is delay in processing the XML or BSD based files.

The SDP based meta-data sharing methods are proposed in [2, 20, 28, 32]. In [32], a method is proposed to signal decoding dependency of different media descriptions with the same media through SDP. The model discussed is suitable for both MDC and layered coding media bit-stream. Here, media description field of SDP is used to describe the decoding dependency information. Network and media capability negotiation using SDP is proposed in [2]. The model is backward compatible with SDP, it specifies how to provide attributes and transport protocols as capabilities and framework for negotiating those parameters. Similarly, [20, 28] describe a model to signal source specific information and media data details to ease the transmission and processing of media contents respectively. These related works provide details of customizing the SDP for a specific purpose.

4 Proposed Signaling method

This section presents design of a signaling and session creation method for layered video communication over Future Internet. Figure 5 demonstrates a typical multicast streaming topology. The end devices heterogeneity is depicted using smart phone, laptop, personal computer and LCD monitor. The capabilities of these devices vary by display resolution, frame-rate, processing capacity and power availability. The media server maintains and manages RAW video sequences. It is capable of generating layered video and streaming the same through the network. MANEs are the FIA proposed intermediate devices that perform adaptation processes. The architecture of MANE is as shown in the Fig. 5. Scalable video bit-stream is received by the MANE and generates adapted video bit-stream from it. Here, MANE-1 is expected to generate two adapted video streams, where one stream is forwarded to MANE-2 and the other is delivered to MANE-3. MANE-3 is having HD TV-1 as one of the receivers, which expects highest available quality, hence all the layers of scalable video stream needs to be delivered to MANE-3 and adapted stream delivered to MANE-2. To carry out this operation, MANE-1 need to know capability details of receiving devices and network connected to it. Communicating these information during session creation is the aim of this proposed work.

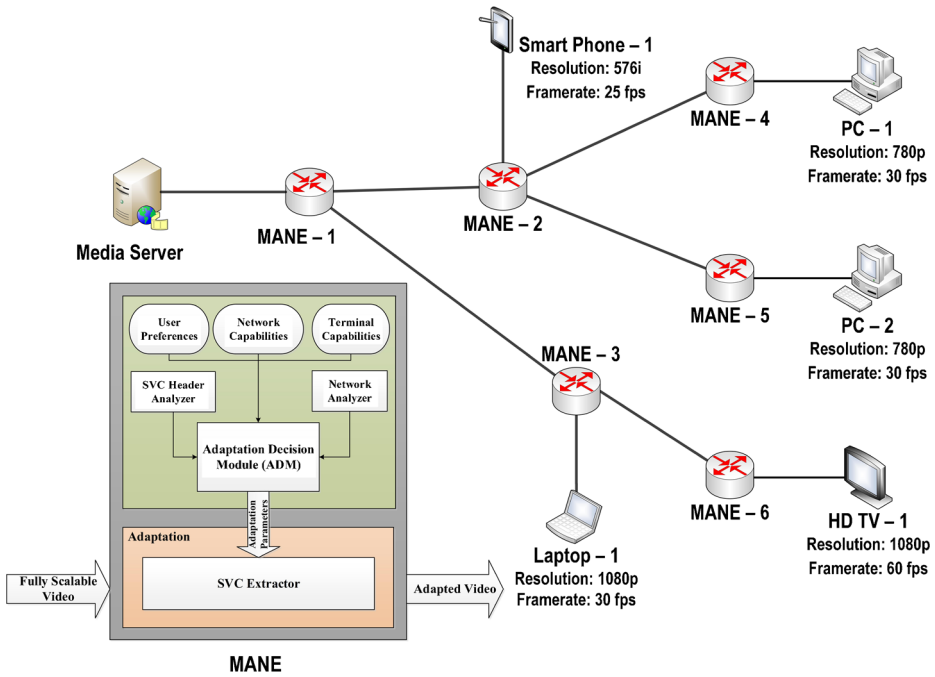


Fig. 5 Multicast Streaming network Topology

4.1 Signaling capabilities through SDP

The proposed signaling scheme considers SDP protocol for describing the terminal, network and media properties. There are many RFCs such as RFC 4568, 4574, 5939 and 6871, which extend the features of SDP to describe application dependent parameters. The method proposed uses the flexibility of SDP in defining the parameters and exchange parameters required for adaptation and streaming. These parameters are used by MANE devices, which are capable of reading these parameters from SDP and store the same throughout the session. The changes in the network and terminal capabilities can also be communicated through SDP without altering the session.

The session, timing and media information of the proposed SDP message is as shown in Table 2. The proposed work is concentrated on the attribute field as it supports new parameters to customize the sessions. Here, attribute field of the session information is used to define the network capabilities such as Average Bitrate, Maximum Bitrate and bandwidth. This field is also used for communicating the terminal capabilities such as Frame rate, resolution, processing capacity and power level. Similarly, attribute fields of media description is used to communicate Supplemental Enhancement Information (SEI) [33] between the layered video encoder of the server and terminal’s video decoder to enhance the efficiency of decoding. The attribute names for above parameters can be defined under “Specification Required” policy of RFC 2434.

Table 2 SDP message used in proposed model

Session description:

v = (protocol version number, currently only 0)
o = (originator, id, version number, network address)
s = (session name)
a =* (zero or more session attribute lines)
a = dispresolution: height = (Display Height); width = (Display Width)
a = framerate: min-fps = (Minimum FPS); max-fps = (Maximum FPS)
a = processcapacity: clock-speed = (CPU clock cycle in Hz)
a = battery: charge-level = (Value in percentage %)
a = networkcapacity: bandwidth = (Link BW); avg-bitrate = (Average Bitrate); max-bitrate = (Maximum Bitrate)
Zero or more Media descriptions (each one starting by an "m=" line)
Time description (mandatory):
t = (time the session is active)
Media description (mandatory for SVC video):
m = (media name and transport address)
a =* (zero or more media attribute lines)
a = framerate:(Media Framerate fps)
a = depend:" dependent-fmt SP dependency-tag (RFC 5583)

The terminal and network capabilities are signaled through SDP. The capability parameters communicated in the message are considered to be infrequent compared to media related parameters. Here, few keywords are used in attribute field to classify the parameters easily at MANE, client and server. MANE stores and uses these parameters for deciding the extraction points. Some of the parameters considered in the proposed work are listed below:

- *dispresolution*: Defines the terminal display resolution using *height* and *width*
- *framerate*: The framerate supported by the end device. Here minimum and maximum frame rate supported are signaled using *min-fps* and *max-fps*
- *processcapacity*: This field represents the processing capacity of the end device in terms of *clock-speed*
- *battery*: The availability of power supply is initialized in the format of percentage to *charge-level* field
- *networkcapacity*: Network capacity is measured by Average Bitrate, Maximum Bitrate and link bandwidth. These are defined through *bandwidth*, *avg-bitrate*, and *max-bitrate*.

Media segment of SDP is used to share the media specific parameters of layered video. RFC 5583, RFC 6236 and RFC 5939 describes different possibilities of signaling media meta-data. The proposed method uses those media attributes to describe SVC video content.

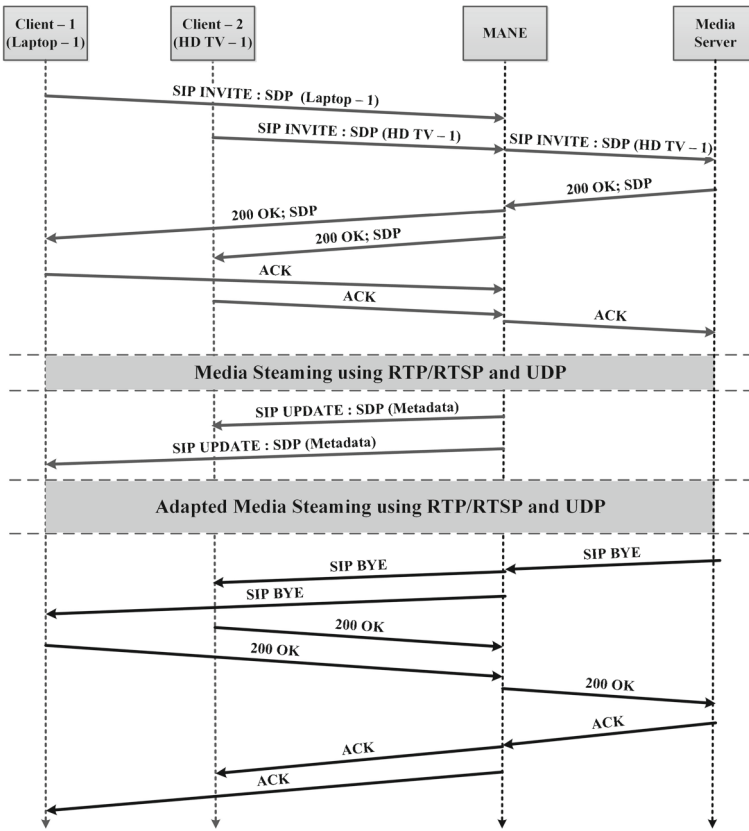


Fig. 6 SIP Call Flow in Future Internet

4.2 Session Establishment using SIP

SIP protocol establish the session by exchanging SIP messages as discussed in the Section 2.3. The proposed SIP call flow is as shown in the Fig. 6. Here, client devices (laptop and HD TV) are connected to media server through MANE. Client devices initiate or request for the same video contents with a media server. The media server streams the video using layered video coding and allows MANE to adapt the video on the fly.

The session creation starts with SIP INVITE message. Client devices create the SDP packets to describe the terminal and network capabilities as explained in the Section 4.1. Generated SDP is encapsulated in the SIP INVITE and then transmitted to MANE. The received SIP INVITE are processed at MANE, where it maintains and uses these capability information for adaptation decisions. As MANE is an intelligent device, it generates a new consolidated SIP INVITE instead of forwarding all received SIP INVITE to media server. While describing terminal and network capabilities to media server, MANE compares all received SDPs and selects the one which has highest requirements. Hence, MANE forwards single SIP INVITE to create a session with media server. Once SIP INVITE is received at media server, it acknowledges the invitation by sending 200 OK message back to MANE, subsequently MANE acknowledges the client SIP INVITE messages.

Here, 200 OK acknowledgment is used to piggyback the SEI data with MANE extractor and client decoder. The main intention of SEI is to enhance the performance of an extractor and a decoder. Finally, session is confirmed by sending ACK message to MANE and media server. In this method, session is created in such a way that, client to MANE is one session and MANE to Media server is another session. The basic advantage of this model is adapted video can be streamed to each end devices independently as adapted video will have dynamic description.

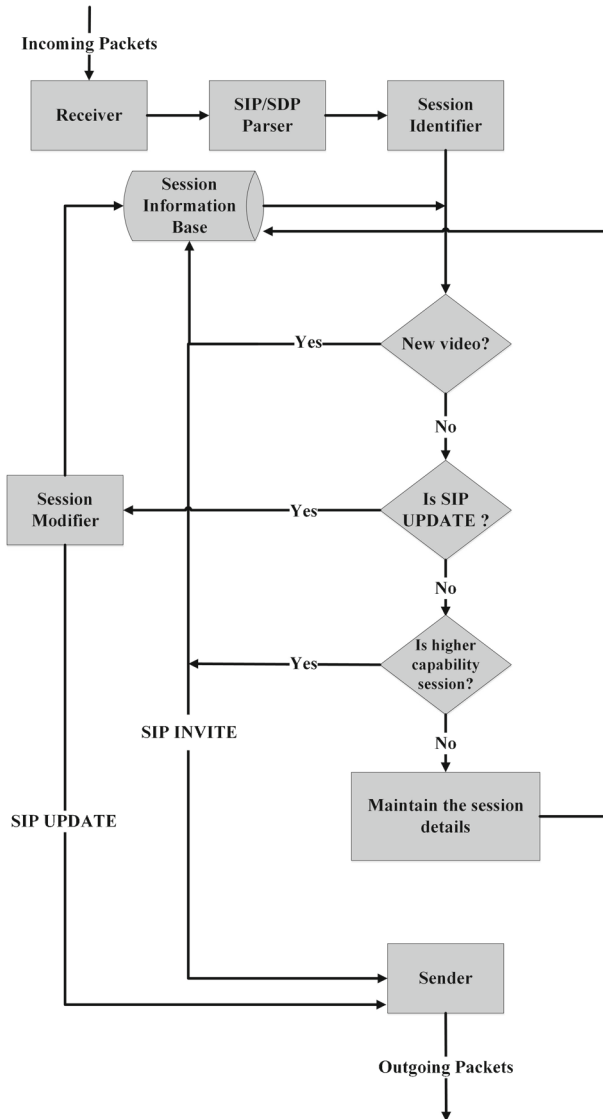


Fig. 7 SIP and SDP packet parser

Once session is created, media server starts streaming fully scalable video to MANE using RTP/UDP streaming protocols. Scalable video is adapted based on terminal and network capabilities available at MANE. Changes in the terminal or network capabilities are signaled using SIP UPDATE. The SIP UPDATE can also be used to signal media meta-data changes. This updates the adaptation decision taken by the MANE and modifies the adapted video stream with a new decision. In these cases, the session need not be recreated unlike traditional system.

The session can be terminated by either client or media server. Termination of the session is signaled through SIP BYE. Then, devices start releasing resources allocated for the session. The devices acknowledge the resource releasing with 200 OK message. Later, media server also releases the resources and sends ACK message to all the peers participated in the session.

5 Implementation details

The implementation of proposed session creation method is detailed in this section. Figure 7 illustrates the SIP and SDP packet parser implemented in the MANE. The rectangular boxes represent functional modules implemented. The cylinder depicts the information-base used to maintain the session descriptions belonging to each end device. Later, it is used by the adaptation decision taking module. The arrows represent the flow of execution.

The packets received at *Receiver* are parsed by the *Parser*. Here, interest is on SIP and SDP packets, hence details obtained by parsing the packets are analyzed for session details by the *Session Identifier*. This module checks if session is new by looking into the *Session Information Base*, where all sessions and participating peers' details are maintained. If so, the session detail is entered into *Session Information Base* as a new entry. In case multiple SIP INVITE messages are received by the MANE, it compares the capabilities of each session, that belong to the same video content. The highest capability details available in *Session Information Base* are considered for sending to the media server. Accordingly, a SIP INVITE message is generated by the MANE and sent to the media server by the *Sender*.

The *Receiver* is capable of receiving acknowledgments and other responses such as SIP BYE and 200 OK from the participating devices. In case of SIP UPDATE message, *Session Modifier* modifies the existing session and forwards the SIP UPDATE to corresponding devices through *Sender*. The *Session Information Base* keeps all details such as network, terminal and media properties through the session. These entries are removed once SIP BYE is received and parsed at MANE.

Pseudocode-1 shows MANE's main module used for establishing the session by processing SIP and SDP messages. The important component is identifying the SIP packets and also comparing the capability details with the available session details. When clients send SIP INVITE to MANE, the session details in SIP INVITE message are different from device to device, but the content they are interested is same. As MANE is a CAN enabled device, content details are considered as unique details to traverse through the session information base. For a matching content, corresponding terminal and network capabilities are read from the information base and then compared with recently received SIP INVITE message. The highest capability details are considered for signaling with media server. Accordingly MANE creates new SIP INVITE packet to communicate with media server. In this way, session is created and managed by the proposed SIP and SDP module.

Pseudocode 1 Session handler at MANE

```

1: sib = Session Information Base
2: Check for content details in sib
3: if not found then
4:   if INVITE then
5:     /* new session and new content */
6:     Read SDP details
7:     Add new entry in sib
8:     Send INVITE to Media Server
9:   end if
10: else
11:   if INVITE then
12:     /*new session and existing content */
13:     Compare capability details with existing session
14:     Select the higher capability one
15:     Send INVITE to Media Server
16:   else if UPDATE then
17:     /* session update */
18:     Identify the parameters needs to be updated
19:     Modify the session
20:     Send SIP UPDATE to participants
21:   else
22:     /* other packet such as 200 OK, BYE */
23:     Process the packets and acknowledge
24:   end if
25: end if

```

6 Experimental setup and results

This section gives the details of software used and network topology considered for the experimentation. The results obtained from the experiments and observation derived from it are discussed here. The proposed model is tested for availability of the meta-data at intermediate devices and media server in a multicast video streaming scenario. Figure 8 shows the network topologies considered for testing performance of the proposed SIP and SDP based session creation. Here, network topologies are created in a virtual platform using ESXi hypervisor. We created bus, ring, star, tree mesh and random topologies. Each topology consists of 9 nodes and 5 terminals, where Node-1 to Node-9 are the intelligent intermediate devices, Receiver-1 to 4 are receiving terminals, and sender-1 is the media server. The devices are connected via virtual networks. The network uses Dijkstra's algorithm to form shortest path tree. The tree is formed considering sender as a root and same is shared with all the nodes in the network. All the devices (Nodes, Receiver and Sender) are installed with Ubuntu 14.04 operating system. The heterogeneous environment is created by configuring Receiver-1 to 4 with different display resolution, frame-rate supported, RAM capacity, memory size and processor as shown in Table 3.

The SIP and SDP modules are developed on python platform, which are used by the terminals to create and the intermediate devices to process the SIP messages. A text-based SIP and SDP message is generated from the modules implemented and then transmitted using UDP/IP over the network. The Media server has video data which are of receivers' interest. Here, receivers initiate the session creation process by generating the SIP INVITE message. This message is then sent to the directly connected node. The nodes use SIP and SDP modules discussed in Section 4 to process the received SIP INVITE messages. If

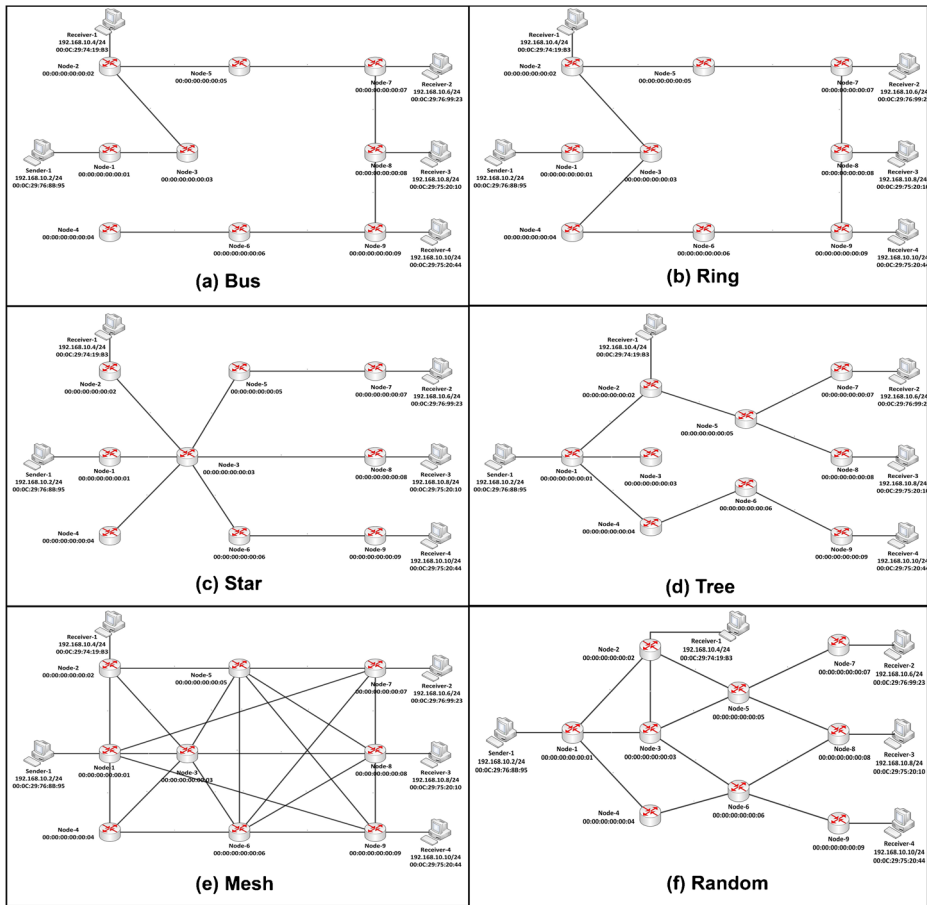


Fig. 8 Network topologies for the experimentation

modification required for the SIP message, nodes create the new message and forward to next hop as proposed in Pseudocode 1. Then, node forwards the message towards media server.

The proposed model is verified by creating a sample SDP message and then transmitting the same through a SIP message. Here, all four Receivers request for a session with

Table 3 Heterogeneity in Receiving Devices

Devices / Capabilities	Resolution WidthxHeight	Framerate (fps)	RAM (MB)	Memory Size (GB)	CPUs (Number of cores)
Receiver - 1	800x600	18	512	16	1
Receiver - 2	1280x768	25	1024	80	2
Receiver - 3	1680x1050	40	2048	256	4
Receiver - 4	1856x1392	60	4096	1024	8

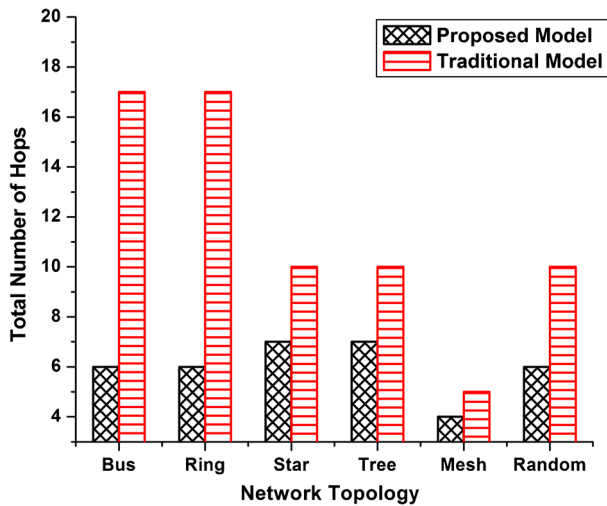


Fig. 9 Total Number of hops taken by the SIP messages

the Sender. The video content requested by all the Receivers is same, hence all 4 requests have the same video identification in the media section of SIP INVITE message. We have derived the results for analyzing the performance and verifying the correctness of the proposed model. It is observed that irrespective of any kind of topology used by the network, media server receives one SIP INVITE message having highest requirements as proposed model combines SIP messages at intermediate devices and generates a message. This helps the server in maintaining one sessions instead of 4.

In Fig. 9 total hops taken by the SIP INVITE message are shown. Here, total hops stands for the number of hops taken by all request messages. The results obtained show that the links are utilized effectively, when compared to traditional mode. In the traditional

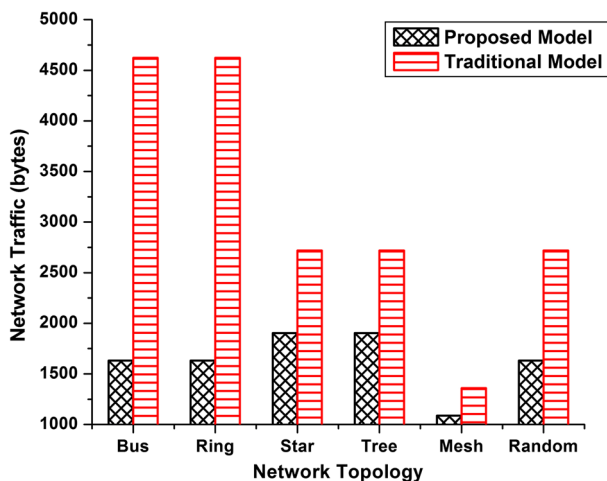


Fig. 10 Total network traffic due to SIP messages

model all 4 session request messages are considered as independent and routed individually, therefore more number of hops are visited by the messages. The proposed model combines individual requests and makes a request from it.

The SDP message is created with all mandatory details along with the attribute parameters discussed in the Section 4.1, then the same is encapsulated in a SIP INVITE message. Hence, SIP INVITE message generated by the proposed model is of 272 bytes, which is used for deriving the network traffic. Figure 10 demonstrates the total network traffic generated due to SIP messages. The results show that the proposed model creates less traffic and hence improve the bandwidth utilization of the network.

From these results, it is observed that the network traffic is reduced by 40–50 % in the proposed model of session establishment than the traditional model. Another important observation is that the number of sessions maintained by the Sender is always one irrespective of number of receiving devices and topology. This enhances the performance of the Sender and simplifies the process of video adaptation in the network.

7 Conclusion and future work

Multicast multimedia communication over the Future Internets will be the interest of both user and provider in coming years. Here, important challenge is providing better QoS and QoE by reducing the bandwidth consumption. In-network adaptation is the better alternate to improve QoS/QoE in multicast multimedia communication over FIA. To enable adaptation at intermediate devices such as MANE, prior knowledge about the participating peers and network are required for deciding the extraction points and then adapting the contents. This paper proposes a method of exchanging the terminal and network capability and media content descriptions with intermediate devices as a prior knowledge. The method proposed extends SIP and SDP standards for signaling SVC and adaptation related parameters. Here, an in-band signaling method is developed, which uses the bandwidth of the network effectively by making use of intermediate devices' intelligence. From the experiment, it is proved that the information received at MANEs are matching with the source generated information. Also, server receives one copy out of many requests generated by multiple participants. It is also proved that model reduces the network traffic by 40–50 % compared to traditional signaling method. The plan for future work is to develop an adaptation module, which uses these prior knowledge to decide extraction points and studying the performance of video steaming.

References

1. Adzic V, Kalva H, Furht B (2011) A survey of multimedia content adaptation for mobile devices. *Journal of Multimedia Tools and Applications* 51:379–396. doi:[10.1007/s11042-010-0669-x](https://doi.org/10.1007/s11042-010-0669-x)
2. Andreassen F (2010) Session description protocol (sdp) capability negotiation. RFC 5939
3. Arnaiz L, Mene andndez J, Bermejo D (2011) Efficient personalized scalable video adaptation decision-taking engine based on MPEG-21. In: 2011 IEEE International Conference on Consumer Electronics (ICCE), pp 381–382. doi:[10.1109/ICCE.2011.5722639](https://doi.org/10.1109/ICCE.2011.5722639)
4. Borcoci E, Negru D, Timmerer C (2010) A novel architecture for multimedia distribution based on content-aware networking. In: 2010 3rd International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ), pp 162–168. doi:[10.1109/CTRQ.2010.35](https://doi.org/10.1109/CTRQ.2010.35)
5. Burnett I, Pereira F, van de Walle R, Koenen R (2006) *The MPEG-21 Book*. Wiley

6. Chakareski J, Han S, Girod B (2005) Layered coding vs. multiple descriptions for video streaming over multiple paths. *Multimedia Systems* 10(4):275–285. doi:[10.1007/s00530-004-0162-3](https://doi.org/10.1007/s00530-004-0162-3)
7. De Schrijver D, Poppe C, Lerouge S, De Neve W, Van de Walle R (2006) MPEG-21 bitstream syntax descriptions for scalable video codecs. *Journal of Multimedia Systems SpringerLink* 11:403–421. doi:[10.1007/s00530-006-0021-5](https://doi.org/10.1007/s00530-006-0021-5)
8. De Schrijver D, De Neve W, De Wolf K, De Sutter R, Van de Walle R (2007) An optimized MPEG-21 BSDL framework for the adaptation of scalable bitstreams. *J Vis Comun Image Represent* 18:217–239. doi:[10.1016/j.jvcir.2007.02.003](https://doi.org/10.1016/j.jvcir.2007.02.003). <http://dl.acm.org/citation.cfm?id=1242858.1243301>
9. Famaey J, De Turck F (2012) Federated management of the future internet: status and challenges. *International Journal of Network Management* 22(6):508–528. doi:[10.1002/nem.1813](https://doi.org/10.1002/nem.1813). <http://dx.doi.org/10.1002/nem.1813>
10. Famaey J, De Turck F (2013) Federated and autonomic management of multimedia services. In: 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), pp 927–933
11. Famaey J, Itebeke F, Wauters T, Turck FD (2013) Towards a predictive cache replacement strategy for multimedia content. *J Netw Comput Appl* 36(1):219–227. doi:[10.1016/j.jnca.2012.08.014](https://doi.org/10.1016/j.jnca.2012.08.014). <http://www.sciencedirect.com/science/article/pii/S1084804512001919>
12. Fortino G, Russo W, Vaccaro M (2014) An agent-based approach for the design and analysis of content delivery networks. *J Netw Comput Appl* 37:127–145. doi:[10.1016/j.jnca.2012.11.005](https://doi.org/10.1016/j.jnca.2012.11.005). <http://www.sciencedirect.com/science/article/pii/S1084804512002469>
13. Gardikis G, Xilouris G, Kourtis A, Negru D, Chen Y, Anapliotis P, Pallis E (2011) Media ecosystem deployment in a content-aware future internet architecture. In: 2011 IEEE Symposium on Computers and Communications (ISCC), pp 544–549. doi:[10.1109/ISCC.2011.5983894](https://doi.org/10.1109/ISCC.2011.5983894)
14. Go andrkemli B, Tekalp A (2010) Adaptation strategies for streaming SVC video. In: 2010 17th IEEE International Conference on Image Processing (ICIP), pp 2913–2916. doi:[10.1109/ICIP.2010.5652838](https://doi.org/10.1109/ICIP.2010.5652838)
15. Graff M, Timmerer C, Hellwagner H, Xilouris G, Gardikis G, Renzi D, Battista S, Borcoci E, Negru D (2013) Scalable media coding enabling content-aware networking. *IEEE MultiMedia* 20(2):30–41. doi:[10.1109/MMUL.2012.57](https://doi.org/10.1109/MMUL.2012.57)
16. Handley M, Jacobson V (1998) SDP:Session description protocol. RFC 2327
17. Hartwig S, Luck M, Aaltonen J, Serafat R, Theimer W (2000) Mobile multimedia-challenges and opportunities. *IEEE Trans Consum Electron* 46(4):1167–1178. doi:[10.1109/30.920475](https://doi.org/10.1109/30.920475)
18. Hellwagner H, Kuschnig R, Stütz T, Uhl A (2009) Efficient in-network adaptation of encrypted H.264/SVC content. *Journal of Image Communication, Elsevier Science* 24:740–758. doi:[10.1016/j.image.2009.07.002](https://doi.org/10.1016/j.image.2009.07.002). <http://dl.acm.org/citation.cfm?id=1598093.1598590>
19. Jennings B, Brennan R, Donnelly W, Foley S, Lewis D, O'sullivan D, Strassner J, van der Meer S (2009) Challenges for federated, autonomic network management in the future internet. In: IFIP/IEEE International Symposium on Integrated Network Management-Workshops, 2009. IM '09, pp 87–92. doi:[10.1109/INMW.2009.5195942](https://doi.org/10.1109/INMW.2009.5195942)
20. Johansson I, Jung K (2011) Negotiation of generic image attributes in the session description protocol (sdp). RFC 6236
21. Kawasaki K, Ata S, Murata M (2015) Design of communication architecture to support stream data over content-centric networking. In: The 10th International Conference on Future Internet, CFI '15. ACM, New York, pp 57–62. doi:[10.1145/2775088.2775091](https://doi.org/10.1145/2775088.2775091)
22. Kim SS, Choi MJ, Ju HT, Ejiri M, Hong JK (2008) Towards management requirements of future internet. In: Ma Y, Choi D, Ata S (eds) *Challenges for Next Generation Network Operations and Service Management, Lecture Notes in Computer Science*, vol 5297. Springer, Berlin, pp 156–166. doi:[10.1007/978-3-540-88623-5_16](https://doi.org/10.1007/978-3-540-88623-5_16)
23. Kofler I (2010) In-network adaptation of scalable video content. *SIGMultimedia Rec* 2:7–8. doi:[10.1145/2039331.2039335](https://doi.org/10.1145/2039331.2039335)
24. Kuschnig R, Kofler I, Ransburg M, Hellwagner H (2008) Design options and comparison of in-network H.264/SVC adaptation. *J Vis Comun Image Represent* 19:529–542. doi:[10.1016/j.jvcir.2008.07.004](https://doi.org/10.1016/j.jvcir.2008.07.004). <http://dl.acm.org/citation.cfm?id=1465756.1466321>
25. Latr S, Famaey J, Turck FD (2014) A semantic context exchange process for the federated management of the future internet. *Int J Netw Manag* 24(1):1–27. doi:[10.1002/nem.1840](https://doi.org/10.1002/nem.1840)
26. Lederer S, Mueller C, Timmerer C, Hellwagner H (2014) Adaptive multimedia streaming in information-centric networks. *IEEE Netw* 28(6):91–96. doi:[10.1109/MNET.2014.6963810](https://doi.org/10.1109/MNET.2014.6963810)
27. Lee H, Kang JW, Kim JG (2007) A SVC adaptation decision engine based on MPEG-21 DIA for universal multimedia access, pp 1–5. doi:[10.1109/ISCE.2007.4382160](https://doi.org/10.1109/ISCE.2007.4382160)
28. Lennox J, Ott J, Schierl T (2009) Source-specific media attributes in the session description protocol (sdp). RFC 5576

29. Daras P, Williams CGD et al (2009) Why do we need a content centric Future Internet? European Commission Information Society and Media. Future Content Networks Group, Prague
30. Posch D, Kreuzberger C, Rainer B, Hellwagner H (2014) Using in-network adaptation to tackle inefficiencies caused by dash in information-centric networks. In: Proceedings of the 2014 Workshop on Design, Quality and Deployment of Adaptive Video Streaming, VideoNext '14. ACM, New York, pp 25–30. doi:[10.1145/2676652.2676653](https://doi.org/10.1145/2676652.2676653)
31. Rosenberg CG, Schulzrinne H, Schooler V (2002) SIP: Session initiation protocol. RFC 3261
32. Schierl T, Wenger S (2009) Signaling media decoding dependency in the session description protocol (sdp). RFC 5583
33. Schwarz H, Marpe D, Wiegand T (2007) Overview of the scalable video coding extension of the H.264/AVC standard. IEEE Transactions on Circuits and Systems for Video Technology 17(9):1103–1120. doi:[10.1109/TCSVT.2007.905532](https://doi.org/10.1109/TCSVT.2007.905532)
34. Serrano M, Davy S, Johnsson M, Donnelly W, Galis A (2011) Review and designs of federated management in future internet architectures. In: The Future Internet, Lecture Notes in Computer Science, vol 6656. Springer, Berlin, pp 51–66. doi:[10.1007/978-3-642-20898-0_4](https://doi.org/10.1007/978-3-642-20898-0_4)
35. Sodagar I (2011) The mpeg-dash standard for multimedia streaming over the internet. IEEE MultiMedia 18(4):62–67. doi:[10.1109/MMUL.2011.71](https://doi.org/10.1109/MMUL.2011.71)
36. Thang TC, Kim JG, Kang JW, Yoo JJ (2009) SVC adaptation: Standard tools and supporting methods. J Signal Process Image Commun 24(3):214–228. doi:[10.1016/j.image.2008.12.006](https://doi.org/10.1016/j.image.2008.12.006). <http://www.sciencedirect.com/science/article/pii/S0923596509000034>
37. Cong TT, Suk KY, Jae-Gon K (2006) SVC bitstream adaptation in MPEG-21 multimedia framework. In: Journal of Zhejiang University Science A, vol 7, pp 764–772
38. Timmerer C, Hellwagner H (2010) Mpeg-21 digital items in research and practice. In: Proceedings of the 1st International Digital Preservation Interoperability Framework Symposium, INTL-DPIF '10, vol 8. ACM, New York, pp 1–8. doi:[10.1145/2039263.2039271](https://doi.org/10.1145/2039263.2039271)
39. Valloppillil V, Ross KW (2010). D2.2 service/content adaptation definition and specification. www.ict-alicante.eu/userfiles/file/.../ALICANTE_D2_2_Final.pdf
40. Vetro A, Timmerer C (2005) Digital item adaptation: Overview of standardization and research activities. IEEE Transactions on Multimedia Special Issue on MPEG 21:418–426. doi:[10.1109/TMM.2005.846795](https://doi.org/10.1109/TMM.2005.846795)
41. Wang Y, Reibman AR, Lin S (2005) Multiple description coding for video delivery. In: Proceedings of the IEEE, vol 93, pp 57–70. doi:[10.1109/JPROC.2004.839618](https://doi.org/10.1109/JPROC.2004.839618)
42. Wenger S, Wang YK, Schierl T (2007) Transport and signaling of SVC in IP networks. IEEE Transactions on Circuits and Systems for Video Technology 17(9):1164–1173. doi:[10.1109/TCSVT.2007.905523](https://doi.org/10.1109/TCSVT.2007.905523)



M. Ramakrishna is working as Assistant Professor in the department of Information and Communication Technology, MIT, Manipal. He has finished his B.E. (Computer Science and Engineering) from Vivekananda College of Engineering and Technology, Puttur, Karnataka, INDIA (recognized by VTU, Belgaum) in 2008. He pursued M.Tech. (Network Engineering) from Manipal Institute of Technology, Manipal, INDIA in 2010. Currently, he is pursuing Ph.D. in the field of Multimedia Communication under the guidance of Dr. Karunakar A. K. at Manipal University, Manipal. He has also worked as Guest Researcher at Telecommunication Labs, Saarland University, Germany. His research interests are Scalable Video Coding, SVC adaptation, Media Aware Network Elements, Network Protocols, Routing Algorithms, and Software Defined Networks.



A. K. Karunakar is working as an Professor at Department of Computer Applications, Manipal Institute of Technology of Manipal University and Deputy Director (International Collaborations) of Manipal University, India. He finished his PhD in 2009 in the area of "Scalable Video Coding" and Postdoctoral fellowship (16 months) at AIT Ireland on "Scalable Video Streaming over Peer-to-peer Networks". His research interest includes image/video processing and communication, Scalable Video Coding, Media Aware Network Elements, Multi-view Video Coding, Scalable video over Peer-to-Peer Networks, Error resilient and concealment for scalable video, stereo vision, image & video forensics, etc.,