APPLICATION LAYER QOS SUPPORT FOR VIDEO SERVICES

Bruno Guedes

Instituto Superior Técnico – Taguspark
Av. Prof. Dr. Cavaco Silva, 2744-016 Porto Salvo, Portugal
E-mail: bruno.guedes@tagus.ist.utl.pt

Abstract – The ubiquity of IP technology is the starting-point to many current services, such as IPTV. Most of these services are supported in access network by xDSL networks (last mile), which are the service provider's bottleneck. The main goal of this thesis is to present a solution that adapts video contents to the xDSL networks.

In order to provide clients with different video qualities - according to their reception capabilities -, we propose a solution applied to *Media-aware Network Element* (MANE) consisting on a IP packets discarding technique at network level. However, this solution was not adequate, due to the huge visual impact experienced by clients even when a small number of video packets were dropped. To overcome this drawback, we developed a second technique, where a media server sends streams with different video qualities of the same video content and the quality level of the stream received by the client is selected by MANE, which delivers contents through the access network. Experimental results show that later technique provides a higher video transmission bitrate reduction without a huge visual impact experienced by clients.

The proposed adapting technique in which media server sends different instances of the same content with different video quality to a MANE selecting process is performed by constantly gathering client side information using Axess solution, based on TR-069 DSL Forum specification. Using Axess information, it is possible to select the best quality flow for each specific customer, with specific reception capability. This dynamic adaptation technique assures that clients always receive best video quality level according to their current characteristics that include both access network load conditions and receivers devices capacities.

Keywords: IPTV, xDSL, dynamically, adaptation, access network.

1. INTRODUCTION

Internet Protocol Television (IPTV) is one of the forms of IP convergence that is gaining interest worldwide [1]. In order to offer new services, traditional telephone operators, exploited *Digital Subscriber Line* (xDSL) technologies to allow their entrance in a new market using their existing copper networks. Moreover, IPTV changes the television paradigm, allowing more interactivity and contents control than traditional television service.

This type of services, which relies on *Internet Protocol* (IP) multicast protocol to deliver the same content to a large number of clients, has strict *Quality of Service* (QoS) requirements (very sensitive to losses and delays) and are fully controlled by the service provider. To avoid content's degradation the video is sent with its best available quality. However, clients may not have a good quality reception, so frames could be lost which could imply a bad visual experience by clients.

The IPTV service requires a meaningful bandwidth allocation, particularly if we talk about high definition service. For that reason, currently, providers are very interested in fiber; however, this technology is very expensive to deploy. Therefore, IPTV service is commercially viable in very populated areas. Hence, for the next years to come, the most widely used technology for IPTV service will be the xDSL. On the other hand, this technology looses efficiency with the increasing distance between *Digital Subscriber Line Access Multiplexer* (DSLAM) and customer's house. In rural areas, DSLAM devices are not located as nearer to customers as in urban areas. Thus, according to the distance, the video quality reception could vary for each customer. Therefore, to solve this problem, a solution was developed to adapt the received quality video to the client conditions. This solution was schemed with three requirements: adaptation should be dynamic; should adapt to the customer's access network; should be scalable, supporting a high number of IPTV subscribers.

In this paper, we evaluate a solution to adapt streaming video to the access network, which is the bottleneck of the service provider infrastructure. In section 2, it is shown some of the state-of-the-art, introducing some video coding standards, specially related to coding techniques that allow content adaptation to client conditions. It is also shown, a typical IPTV architecture and described different solutions for each part of the network. In the section 3, we describe a technique to define different quality flows, which will be used in our proposed solution. In section 4, we explain a method to get dynamic information from the client device to allow not only an initial adaptation but also a periodic adaptation. Section 5 focuses on four aspects: proposed solution; test architecture; tests and results; how could this solution model be implemented in a real environment. Finally, conclusions and future work are described in section 6.

2. STATE OF THE ART

This section will focus on encoding scalability solutions and video adaptation techniques. Encoding scalability describes two *Moving Picture Experts Group* (MPEG) standards that allow video scalability. In adaptation techniques subsection is described different solutions (divided for each IPTV architecture part), that allow video content adaptation to the clients.

2.1. Encoding scalability

A video can be scaled in three dimensions: temporal by changing its frame-rate; quality by altering its signal-to-noise ratio; and spatial by using different resolutions [2]. With the aim to get this scalability it was considered two standards: H.264/SVC[1] and MPEG-J[2].

H.264/SVC is an extension of H.264/AVC, which is the tenth part of the MPEG-4 [3]. *Joint Video Team* (JVT) developed this standard. H.264/SVC allows the encoding of video streams, which can be scalable into one or more of the following dimensions: temporal, spatial and quality. The video is divided into several sub-streams, known as layers. The base layer provides the lowest quality/resolution/frame rate, which can be improved by adding other layers. The decoding of the full quality video requires the availability of all layers. If the client cannot receive the best quality, the higher layers can be dropped, creating a new video flow with lower quality. The communication between the server and the client is mediated by a media gateway, known as *Media-Aware Network Element* (MANE), which adapts the original stream to the client capacity by dropping specific layers. The problem is that, currently, there are non-commercial streamers able to support this standard.

MPEG-J [4] is part of MPEG-4 System and defines a set of *Java Application Programming Interface* (JAPI) to access and control contents in the client's device. Thus, it is possible to adapt dynamically the contents based on network and client device conditions. However, this standard is not yet implement by industry.

2.2. Adaptation techniques

An IPTV network is compose by three fundamental parts [1]:

- Headend, where video contents arrive from other sources (typically satellite). Here, video contents are codified and encapsulated in IP;
- Distribution network, which may be further divided into: core network, which carries the content to several distribution areas and provides the logical connection with other services, such as *Voice over IP* (VoIP) or Internet; and access network, which may be xDSL, cable TV, wireless broadband or *Power Line Communications* (PLC);
- Home network is the client's environment, with their STBs, *Personal Digital Assistants* (PDAs), etc.

Figure 1 illustrates some codification techniques, in the different parts of an IPTV network, which allow QoS management through content's adaptation. It was not consider any solution in the access network, because in this network part, information is transmitted by an analogue signal.

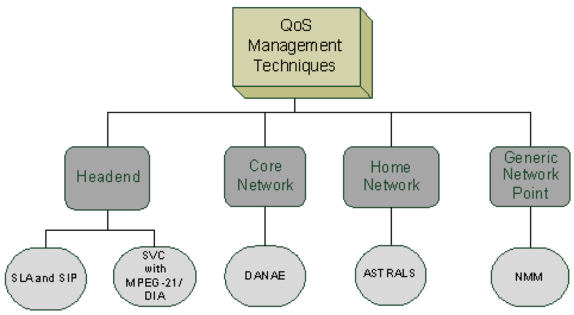

Figure 1 - Techniques to manage the QoS.

In the headend, *Service Level Agreement* (SLA) and *Session Initiation Protocol* (SIP) solution, negotiate the QoS between IPTV clients and servers, using SIP [5]. A client starts a session by sending an INVITE message to the server, describing the desired SLA. The server evaluates the request and if it has sufficient resources, it will send an OK message. Otherwise, the request is refused, forcing the client to try another request with a lower QoS. The negotiation might be complicated if the

[1] Defined by *H.264 and ISO/IEC in 14496-10 Amd.3 Scalable video coding*.
[2] Defined in ISO/IEC 14496-21.

server permanently rejects the requested quality. This can be a very slow process if the customer does not know what quality level can be supported by the network.

MPEG-21 *Digital Item Adaptation* (DIA) can be used to adapt a SVC stream [6]. The client uses the *Usage Environment Description* (UED) tool to describe the characteristics of the terminal and network connection, by transmitting this information to the server which contains the *Adaptation Decision taking Engine* (ADTE). This element selects the optimal value for each of H.264/SVC stream scaling dimensions (temporal, spatial and quality), instructing the *Dynamic Extractor* (DE). The DE will adapt the video contents (cutting the layers) based on ADTE's information. The ADTE also receives information about the network from the client, using *Real-Time Transport Control Protocol* (RTCP). The scalability constraints on this solution are the media server capacity, as it also runs the DE, and the use of RTCP on large multicast groups.

In the core network, the *Dynamic and distributed Adaptation of scalable multimedia coNtent in a context-Aware Environment* (DANAE) project, suggests the use of distributed *Adaptation Nodes* (AN) [7]. Here, clients send a request to the server for a video supply, which will be forwarded to the AN. The AN fits the content according to the client location. Then the client will inform the AN, using MPEG-21 *Digital Item Adaptation* (DIA), about its capacities. The AN will continuously probe the transmission channel, estimating the available bandwidth. This information allows the AN to adapt the video, that it will be received for each client served. The server will supply the AN with a video stream with enough quality to satisfy its clients. The adaptation is a distribution process, which takes place at the ANs. The drawback of this solution is the amount of time it takes. First, the AN needs to receive the client information, then compare with other current served clients , and finally send to the server to encode the contents with a specific quality.

In the home network, the *Audio-visual STReaming plAtaform for domestic Leisure and Security* (ASTRALS) project is based on residential gateway to optimize the content [8]. Using the client profile, the residential gateway adapts the content. This content can be H.264/AVC or H.264/SVC. In addition, it adapts the contents based on *Media Access Control* (MAC) (for wireless environments) and network layer. This solution has poor scalability and needs one residential gateway per client. Furthermore, if the content arrives to the media gateway degraded, the residential gateway cannot do anything, to improve its quality.

The *Network-Integrated Multimedia Middleware* (NMM) is a solution that allows video adaptation in a generic network point, where all software and hardware pieces are named as nodes [9]. In each node, it should be possible to transcode the contents and adapt them in any network point to the client, such as: PDA or STB. This approach requires that all relevant elements in the network for the streaming contents implement this middleware. Besides, this solution does not support the most recent codification techniques, like H.264/AVC or H.264/SVC. The adaptation scalability is a heavy burden because it is necessary to decode and code again.

3. THE ADAPTATION TECHNIQUE SELECTION

The first step to reach the solution was the attempt to differentiate video quality by decreasing the bandwidth. The technique to reduce bandwidth emerged from two fundamental requirements:

- The content degradation should have the minimal visual impact with maximum bandwidth reduction.
- Should support the most recent coding techniques, like H.264/AVC and H.264/SVC.

Thus, we devised a technique that created different video qualities of the same content in the MANE. This was achieved by using *iptables* (user level) to analyze and discard the packets. However, as it is shown in [10], this solution had a huge visual impact in the client even with a small packet loss. Another way to send different video qualities to the clients was designed, doing a server content degradation.

To measure the quality of the contents, we adopted a *Quality of Experience* (QoE) analysis, that it is a subjective metric based on mean

Table 1 – QoE values.

QoE value	Description
0	No image.
1	Video with a lot of freezes, the content is very degraded.
2	Video degraded and could be noted a few freezes.
3	Video with frequent block effects, rarely has freezes.
4	Vídeo with rare block effect.
5	Vídeo without losses.

opinion score. In the Table 1 it is illustrated the QoE scale.

Figure 2 illustrates the best metric found, using the content degradation by MANE (blue), compared to the content degradation by server (red). There is a significant difference between the red and the blue line, because for the same visual impact the server faces a 50% packet loss while MANE faces only 5%.

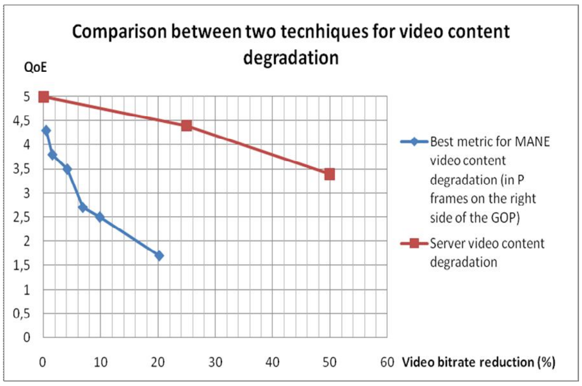

Figure 2 - Comparison between "server degradation" and "best metric in MANE degradation".

4. DINAMIC ADAPTATION

To get information dynamically it was adapted the Axess solution [11], based on TR-069 [12]. That allowed the server to ask for specific information (bytes received, IP, etc.) through *Auto Configuration Server* (ACS).

To calculate the bandwidth consumed by the client's *Customer Premises Equipment* (CPE), it was done two requests with a specific time interval, to get information about bytes received and uptime. With this information, it was possible to determine the value, which was called as "bitrate with minor time interval". The time interval of 10 seconds was considered a reasonable value. This particular time interval is a tradeoff between fast adaptations and the accuracy of the adaptation process. For short periods, if a peak occurs within the period it definitively can influence the adaptation. For long intervals, typically it takes more time to adapt the contents to the client state.

To limit the "peak's influence" in the bitrate (avoiding wrong adaptations), it was defined a second parameter, which was called as "bitrate with major time interval". This parameter calculate the bitrate, using the values from the second request in the last "bitrate with minor time interval" and the first request from the current "bitrate with minor time interval" as illustrated in Figure 3.

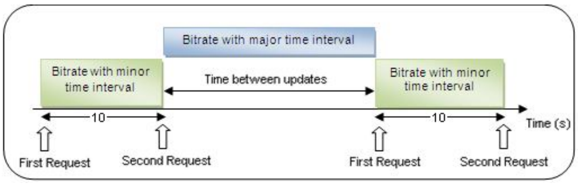

Figure 3 – "Bitrate with major time interval" and "Bitrate with minor time interval".

When both parameters have opposite adaptation decision results, the "Bitrate with major time interval" prevails. Therefore, if "Bitrate with minor time interval" indicates to decrease the video quality while "Bitrate with major time interval" does not, it is considered just a period with less bitrate. If the "Bitrate with minor time interval" indicates a video quality increase while "Bitrate with major time interval" does not, it is considered as a part of the movie with more motion than the majority parts of this video and this peak occurred during the "Bitrate with minor time interval" calculation. In both cases, the video quality is not changed, even if the parameter "Bitrate with minor time interval" indicates to change it.

5. PROPOSED SOLUTION

The proposed solution is based on a server that sends to clients a set of video streams, each having different qualities. In the network a MANE is responsible to select the best flow for each client, based on client information provided by the Axess solution. In this section, we describe the environment test, as well as the results we achieved. Moreover, it is also discussed how this proof of concept could be implemented in a real environment.

5.1. Architecture

Figure 4 illustrates our proposed architecture. The Axess's solution was used, where ACS is an element that allows the MANE to get information from the CPE's. There black arrows represent video flows, while the server (streamer) sends three flows, only one arrives to the CPE, which is selected by MANE. The red and violet arrows represent communication between MANE and ACS, which enables dynamic adaptation. The difference between the two arrows is that the red arrow only communicates with ACS (e.g. get IP from CPE), whereas the violet sends a request to the CPE (e.g. get bandwidth from CPE). The grey arrows represent the different interactions between the blocks that constitute the MANE.

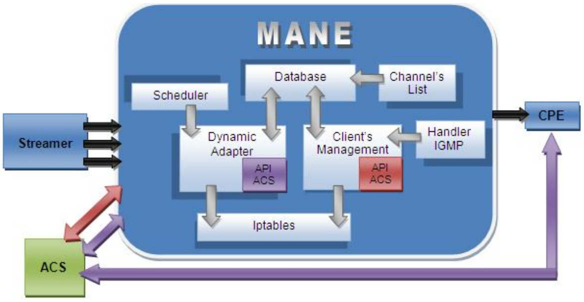

Figure 4 – Solution's architecture.

The "IGMP Handler" module gets the client multicast requests. This module issues join and leave requests containing the client and the channel IP addresses, in which the client wants to be associated/disassociated. This information will be sent to the "Client's Management" module.

The "Client's Management" module receives information from the client requests, and updates the "Database" module and the "Iptables" module. The "Client's Management" communicates with ACS to get the cpeid associated to the CPE IP, this is mandatory for future communications, as the cpeid never changes while the CPE IP might change.

The "Schedule" module is responsible to defuse periodically (using cron tool from Linux) the "Dynamic Adapter" module.

The "Dynamic Adapter" module, when activated, updates the "Database" module with the CPE's IP and the amount of consumed bandwidth. Whenever the CPE's IP is changed, it will be necessary to change the "Iptables" module as well. To get the amount of consumed bandwidth, two requests are send to the CPE, with ten seconds interval, to get your Ethernet Bytes Received and Uptime in the WAN port. With this information and the uptime in each request, it is then possible to calculate the amount of consumed bandwidth.

The "Iptables" module uses the "mangle table" to mark different received flows from the server. The video qualities are distinct by port and the client always will be associated to the best quality video port. In this way, it also uses the *Network Address Translator* (NAT) table when MANE selects a video quality (which will be watched by the client) lower than the best one. Thus, it is necessary to do a port forward from the port which the flow was sent to the port that the client was associated (best quality flow). Therefore, the client does not need to perform a channel change whenever quality changes. Additionally, a filter table allows to assign a specific port to a client and blocking other quality flows.

The "Channel's List" module is a web service that receives a channel list (including information about qualities). This information is added to the "Database" module, allowing to know when the client sends a request, which are the quality ports that should be blocked and those that should be passed to clients.

The "Database" module is implemented using *mysql*. It was separated in four tables:

1. Client table has the information related with the clients, such as: received port, *cpeid* and IP client.
2. Channel table identifies the channels for the univocal combination between IP and port. This table has information, such as: IP, port and bitrate.
3. Historic table has the old information related with the clients.
4. Statistics table has the aggregate information from historic table, determining the average bitrate to a specific client. This allows to know when the client requests a new channel and what should be the best flow for him.

5.2. Test environment

Figure 5 illustrates the test environment used to carry out a set of experiences that validates this adaptation technique proposal, which has the following components:

- A server (streamer) with 2 GB RAM and a 2,8 GHz CPU. It was used *VideoLan Client* (VLC) to transcode and send the contents.
- A ACS with 512 MB RAM and a 1,5 GHz CPU.
- *Switch* is a SURECOM EP-816DX-FW, with 16 ports.
- A PC (client) with 1GB RAM and a 1,73 GHz CPU.
- A Thomson 585 v6 modem (modem1), and a Thompson 546 v6 modem (modem2).
- A MANE with 1GB RAM and a 1,73 GHz CPU.

In this environment, we considered three flows (three lines with three different grey levels), that represent different qualities of the same channel.

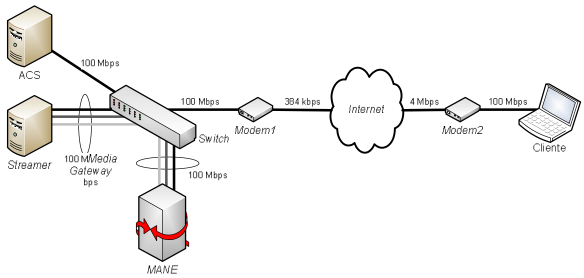

Figure 5 – Solution test environment.

In the current situation, it was necessary to consider streams crossing the Internet, because it was crucial to check the bytes received and uptime, in *Wide Area Network* (WAN) port. The video used in the tests was codified H.264/AVC with 320x240. However, due to a network element shortage (missing a DSLAM), it turned out impossible to carry out tests in a real environment. Nevertheless, our goal was to develop a model (not a prototype) and test it. The "internet" element was deemed enough to test random condition changes.

Tests were carried out using five viewers, who classified the content in a QoE basis. Two test types were specified:

- Run-time adaptation test: to measure the amount of time needed by the MANE to adapt the contents (after have been called), when the client condition changes. It also measures the client impact time when the MANE changes the video quality, in other words, the amount of time between the MANE do a adaptation (finish the script) and the client sees this new quality.

- Quality adaptation test: to measure the visual impact when the video quality changes. It was checked if this new quality was better than the previous one.

5.3. Test Results

Run-time adaptation test results are shown in Table 2. The first column at the left side is the time interval between two CPE requests (predefined) and was used to calculate the "Bitrate with minor time interval" parameter (see Figure 3). The second column represents the amount of time that the adaptation script in the MANE takes to be executed. The third column, is a standard deviation for the time script adaptation. The fourth column is the time between the MANE finishes the adaptation and the visual impact occurs in the client side.

Table 2 - Results related with run-time adaptation.

Time interval (s)	Update time with change quality (s)	Adaptation time standard deviation	Client impact time (s)
10	30,35	0,14	< 1

The results indicate that it takes per client about half a minute to adapt. This amount of time was not considered excessive, since this model was built using copper transmission media, which does not have sudden quality changes.

In the client impact time, it was only possible to assure that it was less than 1s (almost instantaneous). This was consistent, because this time is composed by three factors: time in the VLC buffers has typically 300 ms as maximum delay; time to decode the contents, it depends from client device capacity; time to transmit the contents, this factor has the most impact in the client (depending on network conditions). Therefore, it is possible to deduce that we had a good quality of transmission.

In the quality adaptation test, it was considered: block effect; blur; image freeze and color distortion. When quality changes, no freeze images and very few color distortion was perceived. However, it was observed blur and block effect. These artifacts only fully disappeared when a frame I came out. When changes occur without a frame I, the quality improvement was only perceived in the video parts with movements. This happened because P frames code only video differences, so only these pixels were coded again.

The QoE results for quality adaptation test are present in Table 3. The first column at the left side is the QoE average before adaptation. The second and the fourth column are the standard deviation values before and after adaptation, respectively. The third column is the QoE average after adaptation.

Table 3 - QoE analysis before and after adaptation.

QoE before adaptation	Standard deviation before adaptation	QoE after adaptation	Standard deviation after adaptation
1,8	0,27	3,9	0,41

The results confirm the idea that the received video quality after adaptation is much better than the previous one.

To show the difference before and after adaptation we took snapshots of video content in these situations - Figure 6. On the top is the video coded with best quality (300 kbps). This quality was received before adaptation, but since the client did not have, capacity to receive the video content, this resulted in a bad experience to client. Thus, MANE did the adaptation and changed to a second quality (225 kbps), which is the image on the bottom in Figure 6. With this quality, the client had a better experience, because he could receive all content without losses.

Figure 6 - Quality difference between after and before update the client.

5.4. Implementation in a real environment

The implementation of the proposed model in a real environment has some constraints, such as:

1. A web server should feed the “Channel’s List” module, which would be responsible to provide multicast channel IP to the client.
2. To get the real volume of consumed bandwidth from the client side, it should be assign a *Permanent Virtual Circuit* (PVC) between DSLAM and client. This prevents the bandwidth’s calculation with additional traffic that in other way would be associated to the video flow.
3. The “Iptables” module should disappear, because in a multicast environment (used in IPTV), clients served by the same MANE, could be associated to the same multicast address. Therefore, if a MANE decides that one client should receive a lower video quality, all clients in the same DSLAM will be affected as well, by receiving that same video quality. This happens because the “Iptables” module will do a port forward, from the lower quality port to the best quality port. This inconvenience is likely to occur, whenever the number of clients attached to the same DSLAM increases. Because it has a higher probability, that more than one client is watching the same video channel. To solve this problem two solutions were proposed: in the DSLAM and using the TR-135 [13].

5.4.1. DSLAM solution

At the current time DSLAMs has functions at the network layer and could act as an *Internet Group Management Protocol* (IGMP) proxy and it has always done an association between physic port and an IP. However, if a client requests a channel, all qualities will arrive to the client but he/she will see only the best one.

The logic port in the routing table should be controlled by MANE. If the video stream quality is not the best actually available, the DSLAM should rewrite the port to the destination, because clients always receive a specific channel in the same port. Therefore, the entity responsible to manage multicast message requests should inform (using web services) the MANE (“IGMP Handler” module). With this information, MANE should adapt (using web services) the DSLAM routing table (logic port).

5.4.2. TR-135 solution

The TR-135 defines a data model to manage the STBs via TR-069. This standard specifies a way to get information directly from the STB, instead of getting it from CPE, enabling to perform the adaptation process based on additional parameters and not simply on the actual bitrate value. Additionally, it allows to change a client channel in a mandatory way, which means that it senses different video qualities from the same channel by multicast addresses, instead of logic ports. Thus, if a MANE wants to change the channel quality for a specific client, it just sends a request via ACS to change the STB *Uniform Resource Identifier* (URI). The network extra load on transmitting different video qualities do not exist anymore, because the server only sends streams with the video qualities that have been requested. Thus, MANE does not execute video adaptation anymore; it only takes the task of deciding when the adaptation occurs.

Despite the major advantages of using TR-135, this has not been implemented in the STBs yet. However, a solution using this standard will be very intrusive, because a change in a video quality equals a channel change, which is an operation that takes some latency. Actually, this latency problem is minimized by using some mechanism, such as the *Instant Channel Change* (ICC) [14]. The ICC consists in servers located near the client, who send a unicast flow while the multicast flow does not arrive to the client.

6. CONCLUSION

The IPTV service is common use to deliverer media contents to clients in DSL networks. This type of network has a bottleneck in the access network. The goal of this work is to design a solution to adapt media contents to the client access network, using video adaptation techniques.

To achieve this adaptation, we proven that discard selective packets at IP level was not a good technique, due to its high visual impact experienced by clients even when few packets were dropped. Thus, we proposed a technique were video adaptation is carried out by the content server, which sent different qualities. MANE just selects the best quality flow that currently matches client reception capacity and transmits it, afterwards. With this solution, it is possible to obtain higher bandwidth reduction getting a minor visual impact. This allows some flexibility to the service provider on selecting the number of flows (qualities) and the quality of each one to be delivered.

To adapt dynamically the contents, it was used the Axess solution to get CPE information. With that information, MANE selects a flow with the best video quality in each time adaptation interval.

Not being able to deploy this solution in a real environment, this thesis contributes to a view of how this could be implemented. It was suggested to use TR-135 as the best of two choices. However, changing video quality can be very intrusive and thus it must be previously tested. Even if TR-135 is not used to change the video stream quality, it could be used to get important information about a STB instead of a CPE.

This solution is independent of the codification type and it is identical to the new H.264/SVC scalability codification scheme. As it was described in [15], H.264/SVC could be implemented by sending layers independently as streams. A drawback using this solution is related to the increased server processing activity to code different video streams of the same content. However, this problem is smoothed attending to the high transmission rates available in the core network.

6.1. Future Work

The future work consists on devising and testing the dynamic video quality adaptation solution to be deployed in a real environment.

Once TR-135 is implemented in STBs, it will be interesting using this standard not only to test the video quality change mechanism but also to get more information about the contents reception, in order to a better adaptation.

The proposed model is suitable on *Video on Demand* (VoD) context. In this scenario, MANE should be located in the server, because each flow is sent to a specific client (unicast). In the MANE architecture, it is necessary to change the "IGMP Handler" module, which is a specific component of multicast environment.

Another interesting issue consists on exploring the bitrate percentages defined for each different video qualities and grouping them by content type. For example, having content with less picture motion it might be possible to reduce more bandwidth than content with more picture motion, without worsening the client visual impact.

In an audacious context, having available information about the number and type of client devices, it will be interesting to adapt video content based on the spatial resolution from receiver device, allowing to fit other device types, such as PDAs.

7. REFERENCES

[1] L. Harte, "IPTV *Basics*". Fuquay-Varina USA: Althos Publishing, 2006.

[2] Ye-Kui Wang et al., "*System and Transport Interface of SVC*", IEEE, vol. 17, pp. 1149-1163, September 2007.

[3] H. Schwarz, D. Marpe, T. Wiegand, "*Overview of the Scalable Video Coding Extension of the H.264/AVC Standard*", *IEEE,*

vol. 17, pp. 1103-1120, September 2007.

[4] V. Swaminathan, M. Bourges-Sévenier, "*MPEG-J White Paper*", July 2005. Disponível em *http://www.chiariglione.org/mpeg/technologies/mp04-j/index.htm*.

[5] H. J. Park et al., "*QoS negotiation for IPTV service using SIP*", vol. 2, February 2007.

[6] H. Choi et al., "*Dynamic and Interoperable adaptation of SVC for QoS-Enabled Streaming*", IEEE*,* vol. 53, pp. 384-389, May 2007.

[7] A. Hutter et al., "Automatic adaptation of streaming multimedia content in a dynamic and distributed environment", IEEE, vol. 3, September 2005.

[8] L. Celetto, K. Grüneberg, T. Zahariadis, "*SVC Payload Format and Signalling*", *ASTRALS Consortium*, December 2006.

[9] M. Lohse, "*The Network-Integrated Multimedia Middleware (NMM): Basic Introduction*", July 2005. Disponível em *http://graphics.cs.uni-sb.de/NMM/current/Docs/intro/*

[10] B. Guedes, R. Pereira, T. Vazão, A. Varela, "*Simple Media-Aware Packet Discard Algorithms*" in "International Conference on Information Networking 2009", Chiang Mai Thailand, January 2009.

[11] http://axiros.com/

[12] J. Bernstein, T. Spets, "CPE WAN Management Protocol", May 2004. Disponível em http://www.broadband-forum.org/technical/download/TR-069.pdf.

[13] M. Quacchia, W. Lupton, G. Straub, "TR_135 – Data Model for a TR-069 Enabled STB", Dezembro de 2007. Disponível em http://www.broadband-forum.org/technical/download/TR-135.pdf.

[14] D. Banodkar, K.K. Ramakrishnan, S. Kalyanaraman, A. Gerber, O. Spatscheck, "*Multicast Instant Channel Change in IPTV Systems*", Bangalore, IEEE, pp. 370-379, January 2008.

[15] S. Wenger, Y. Wang, e T. Schierl, "*Transport and Signaling of SVC in IP Networks*", IEEE*,* vol. 17, pp. 1164-1173, September 2007.