Performance Evaluations of a QoE-Based Multipath Video Streaming Mechanism over Video Distribution Network (VDN)

Majd Ghareeb and Cesar Viho

IRISA, University of Rennes I, Campus Beaulieu
35042 Rennes, France
{majd.ghareeb,cesar.viho}@irisa.fr

Abstract. Multipath video streaming over Video Distribution Network comes as a promising solution to overcome the limitations of the classical single path and IP-level video streaming approaches. For this purpose, we have proposed a dynamic multipath video streaming mechanism, based on quality of experience evaluations and available bandwidth estimations. In this paper, we present the results of performance evaluation when applying our video streaming method. Results show the ability of our mechanism to adapt automatically and in real-time to the load variation on different paths in order to maximize the overall quality perceived by end-users.

Keywords: Video streaming, Multipath, VDN, QoE, PSQA.

1 Introduction

With the aim at keeping a high perceived video quality, video streaming over Internet still faces important challenges. Recently, several researches as [3] [4] attend to cope with these challenges by deploying a new promising idea, which is to stream the video over logical or physical multiple paths instead of the classical single path streaming [1] [2]. Multipath Implementation over IP layer is a possible mechanism, but it faces a deployment issue as it needs to modify the existing Internet infrastructure. However, an application layer multipath streaming mechanism is far easier to implement and to deploy. For this reason, a recent trend was to exploit the path diversity provided by the Video Distribution Network (VDN) in streaming the video. We have proposed a multipath video streaming dynamic mechanism, in order to improve the video quality as perceived by end-users. In our mechanism, video is streamed over multiple VDN paths. Path selection is dynamically done based on available bandwidth (AbW) estimations, as presented in [1] with the advantage of using multiple paths to stream the video. Furthermore, the evaluation of path selection is done in real time by observing the Quality of Experience (QoE) at the recipient side. To measure QoE we used a new measurement tool that copes with the limitations of the existing subjective and objective evaluation tools. This tool is called Pseudo-Subjective Quality Assessment (PSQA). In this paper, we present the experiment results of

A. Mauthe et al. (Eds.): FMN 2009, LNCS 5630, pp. 236–241, 2009.
© Springer-Verlag Berlin Heidelberg 2009

applying our mechanism over different video streaming schemes (which paths to use for multipath video streaming) and strategies (how the video sub-flows are distributed among the selected paths). Results show the ability of our mechanism to adapt automatically and in real-time to the load variation over the different paths, in a way that it maximizes the overall quality perceived by end-users. The rest of this paper is organized as follows. In Section 2 we briefly explain our multipath video streaming mechanism. Section 3 highlights the results of the performance evaluation when applying our method. Conclusion and future work are summarized in Section 4.

2 Our Proposition for Multipath Video Streaming in VDN Based on AbW and QoE Estimations

The objective of our proposed mechanism is to keep the quality of the perceived video by end-users as high as possible, during the entire video streaming process. This is not an easy mission, especially with the changeable load over the different paths from the server to the clients. Our method dynamically adapts to the load variation over the multiple overlay paths. Consequently, it needs to consider several important issues ,like how to split the video flow in an efficient way before streaming it over multipath, how to select these paths, and how to adapt this selection, in order to maintain the requested QoE.

Our method is designed to stream stored MPEG video flows. It depends on frame-type in order to split the video into several sub-flows. In our experiments, we use two different strategies: (i) *($I \| P \| B$ strategy)* that splits the video into three separated sub-flows, the first contains I-frames, the second contains P-frames, and the last contains B-frames , and then it sends I,P,B frames respectively over the three best paths; And (ii) *($IP \| B$ strategy)* that puts I and P frames in the same sub-flow and B-frames in second one. Here, IP-frames sub-flow is the one that will be sent over the best path, and B-frames will be sent over the second best path. Deciding the best set of paths among the existing paths from the server to the client is done based on the variation range of AbW estimations as it presented in [1]. On the other hand, evaluating the QoE at the client side is done using Pseudo-Subjective Quality Assessment (PSQA) tool. PSQA [6] comes as a hybrid mechanism between subjective and objective ones. It allows quantifying the quality of a video stream at the receiving end, in a manner that is very close to the human observations. To achieve this evaluation,

Table 1. Mean Opinion Score - MOS

MOS	Quality	Impairment
5	Excellent	Imperceptible
4	Good	Perceptible but not annoying
3	Fair	Slightky annoying
2	Poor	annoying
1	Bad	Very annoying

Fig. 1. Video streaming process over VDN

PSQA bases on training a random neural network (RNN) to learn the mapping between QoE score and technical parameters so that the trained-RNN will be used as a function to give QoE Mean Opinion Score (MOS) in real time as it is indicated in Table 1. Fig. 1 presents the general idea of our mechanism. When a client requests a video, the server selects an initial strategy (video splitting method) and an initial scheme (paths used to stream the sub-flows) to start streaming the video to the client. This is done according to the loads over the available paths from the server to the client. At the client side, PSQA evaluates the QoE of the streamed video, and then it sends back the MOS values of this evaluation to the server. According to the quality observed by end-user, video server will decide to keep or to change its strategy and/or its scheme. This leads to a better quality than the quality achieved when streaming the video using the same strategy and scheme during the entire video streaming period.

3 Performance Evaluation

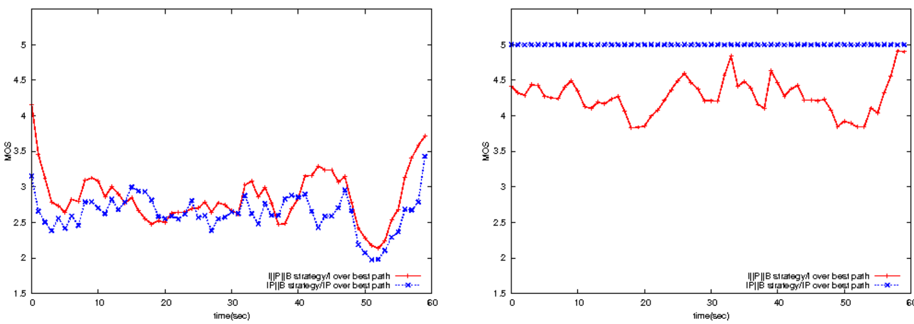
In this section we present the results of the different simulations that we have done using NS-2 simulator. We present first the environment conditions that we chose for our experiments. We show then how our algorithm can adapt with the different load variations over overlay paths. This adaptation can be implemented by changing the streaming strategy and/or the streaming scheme.

3.1 Experiment Setup

We use an MPEG-2 video flow of 60 seconds duration in CIF format (352x288 pixels/frames 30 frames/sec). This flow have been encoded in H-264, with an encoding bit rate of 384 kbps. The first step was to train and validate PSQA tool with this video and to apply it over the clients. Our experiments have been done via the network simulator NS-2 version 2.29, with adding a video packet transmission module to simulate the video streaming application. We implement three separated paths from the server to the client: P1, P2, P3, and then we overload these paths with different CBR (Constant Bit Rate) traffics. These traffics have been chosen in a way that shows the adaptation of our algorithm with load variation as it can happen in real context. For simplifying the results explanation, we will not present the whole range of load values that we used in our experiments. We select two levels of loads that are sufficient to explain the different behaviors of our algorithm, and that lead to the following AbW values over the three paths respectively: Level1 (5%, 10%, 30%) and Level2 (1%, 2%, 3%). Our results are represented in term of MOS values of the streamed video. In our experiments, we supposed that the video quality requested by the client can be achieved by a MOS threshold of 4 that corresponds to a good quality.

3.2 Results

Scenario 1. As we have mentioned in Section 2, choosing the initial strategy is done according to path loads. The objective of our first scenario is to compare $I\|P\|B$ and $IP\|B$ strategies, by applying different load-levels and different schemes. Our experiments on large scale of loads led to the results in Fig. 2. If the loads over all the paths from the client to the server are considerably high, $I\|P\|B$ *strategy* gives an overall quality that is slightly better than the quality given by $IP\|B$ *strategy* (see Fig. 2(a)). In contrast, Fig. 2(b) shows that $IP\|B$ *strategy* is the best strategy to stream the video as long as at least one path to the client has an AbW more than 10%.

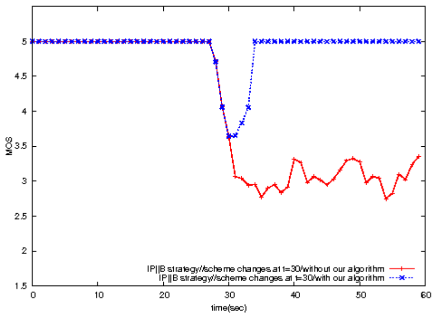


(a) (b)

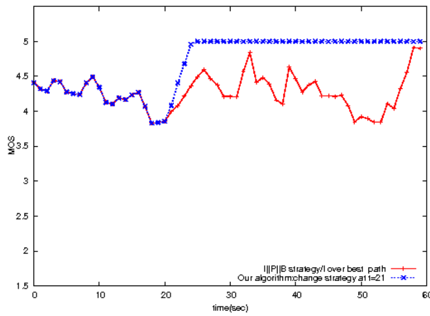
Fig. 2. Comparing different strategies in different load-levels

Scenario 2. The objective of these set of experiments is to demonstrate the adaptability of our algorithm with the variant changeable loads over the different paths, and to show the advantage of using this dynamic method rather than the classical multipath streaming methods.

Scheme Adaptation. Our aim in this experiment is to show how the adaptation of our algorithm to scheme changing gives better video quality. We use the level1 of loads. Thus, the algorithm selects $IP\|B$ as an initial strategy. At *t=30* we change the loads over the paths in a way that the best path changes. In Fig. 3(Scheme adaptation), we can see that our proposed solution maintains good subjective quality despite the change of the best path at *t=30*. Few seconds after perceived quality decreasing, our mechanism recalculates the new AbW of the



(Scheme adaptation)



(Strategy adaptation)

Fig. 3. Strategy or scheme adaptationg using our mechanism

existing paths, chooses the best path again, and then redirects the IP-frames to be streamed over this new best path, and B-frames over the second best path. In contrast, streaming the video without scheme adaptation keeps a good perceived quality only for the first 30 seconds.

Strategy Adaptation. The objective here is to show the benefit of adapting with the load variation by changing the streaming strategy. We start our simulation by the level2 of loads. Consequently, our method selects $I\|P\|B$ as an initial strategy. At *t=5* we change the traffics over the three paths to the level1 of loads. Even with the new traffics, the best path remains the best. We notice that the server keeps the same streaming strategy as long as the MOS value observed by end-user equals or exceeds the MOS threshold. As shown in Fig. 3(Strategy adaptation), the quality decreases to less than 4 starting from *t=19*. Without applying our algorithm the server keeps the same initial strategy until the end of the streaming. Using our method the decreasing of the observed MOS value will lead the server to recalculate the AbW over the paths. As the best path is the same, the server changes the strategy according to the current load-level, to $IP\|B$ strategy. Thus, starting from *t=21* till the end of the streaming, our algorithm enhances the obtained subjective quality.

Hybrid Adaptation. This experiment evaluates the adaptability of our mechanism when both the scheme and the strategy need to be changed. We first apply the level2 of loads till *t=5*. Then we change path loads to level1 keeping the best path as a best path. At *t=40* we change the loads over the three paths, in a way that changes the best path. as Figure. 4 shows that without our mechanism, server keeps the $I\|P\|B$ strategy till the end of the video streaming. However, using our mechanism, at *t=21* the MOS decreasing is taken into account and the streaming strategy is changed to $IP\|B$ strategy while the best path is the same. Moreover, at *t=42* and since the best path has changed leading to new quality decreasing, our method redirects the IP-frames to the new best path, and B-frames to the second best path. We notice that after the best path changing, the quality will decrease to less than 4 (but still ≥ 3.5) for a very short period (the time needed for our algorithm to apply its

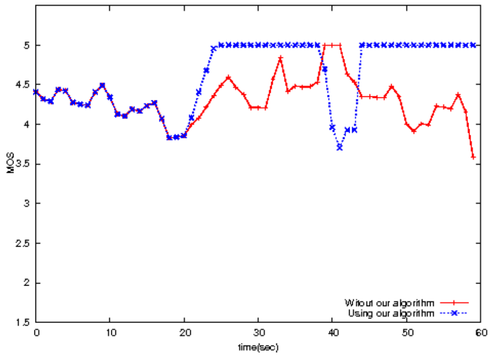


Fig. 4. Hybrid adaptation of our algorithm

adaptation). Even with this decreasing, the overall quality is much better when using our proposed method.

4 Conclusion and Future Work

In this paper, we presented the results of different experiments that show the benefit of our QoE and AbW based mechanism. Results show that our algorithm adapts automatically to the load variation over different paths. It dynamically changes the streaming strategy and/or the streaming scheme, in a way that it guarantees a best perceived quality by end-users. An interesting future work is to apply our approach on a longer video in real platform. another important work is to extend our mechanism to stream the video to several end-users.

References

1. Manish, J., Constantine, D.: Path selection using available bandwidth estimation in overlay-based video streaming. In: Akyildiz, I.F., Sivakumar, R., Ekici, E., de Oliveira, J.C., McNair, J. (eds.) NETWORKING 2007. LNCS, vol. 4479, pp. 628–639. Springer, Heidelberg (2007)
2. Yuan, J., Sun, Q.: An efficient mechanism for video streaming over wide-area networks. In: ISM 2006, Washington,USA, pp. 465–474. IEEE, Los Alamitos (2006)
3. Chow, A.L.H., Golubchik, L., Lui, J.C.S.: Toward picture perfect streaming on the Internet. In: QEST 2005, Washington, DC, USA. IEEE Computer Society, Los Alamitos (2005)
4. Apostolopoulos, J., Wong, T., Tan, W.-T., Wee, S.: On multiple description streaming with content delivery networks. In: IEEE INFOCOM, pp. 1736–1745 (2002)
5. Begen, A., Altunbasak, Y., Ergun, O.: Multi-path selection for multiple description encoded video streaming. IEEE on Communications, 1583–1589 (2003)
6. Rubino, G., Varela, M., Bonnin, J.-M.: Controlling multimedia QoS in the future home network using the PSQA metric. Comput. J. 49(2), 137–155 (2006)
7. Andersen, D., Balakrishnan, H., Kaashoek, F., Morris, R.: Resilient overlay networks. SOSP 35, 131–145 (2001)
8. Manish, J., Constantine, D.: End-to-end available bandwidth: measurement methodology, dynamics, and relation with tcp throughput, pp. 537–549. IEEE/ACM (2003)