International Journal of Multimedia and Ubiquitous Engineering
Vol.9, No.2 (2014), pp.33-44
http://dx.doi.org/10.14257/ijmue.2014.9.2.04

QoE-aware Traffic Shaping for HTTP Adaptive Streaming

Xinying Liu[1] and Aidong Men[2]

[1,2]Beijing University of Posts and Telecommunications
No.10 Xitucheng Road, Haidian District, Beijing, China

[1] liuxinying2@gmail.com, [2] menad@bupt.edu.cn

Abstract

HTTP Adaptive Streaming (HAS) has become a prevailing technology for media delivery nowadays. It enables high quality streaming of media content over the internet delivered from conventional HTTP web servers. However, its on-off traffic pattern constructed by segmented transmission can lead to some performance problems, such as instability, inefficiency and unfairness, while multiple clients compete for limit bandwidth. And the reason caused these poor performance has been pointed out in the past studies. In this paper, we present a QoE-aware traffic shaping method that is based on two QoE maximization metrics. A key benefit of this approach is that it calculates optimal shaping rate to help clients to adjust its request for subsequent segment quality level. Additionally, this method can ensure optimal quality of experience toward end users. The simulation results show that this proposed approach improves the stability, utilization and provides the optimal quality of experience for large number of users.

Keywords: *HTTP Adaptive Streaming, quality of experience (QoE), traffic shaping, instability, utilization*

1. Introduction

Recently, HTTP adaptive streaming is the prevailing technique to stream live and video on demand (VoD) contents as it is used by the solutions from, for example, Apple, Microsoft, and Adobe. Using this new technique, high quality streaming of media contents are delivered over the Internet from conventional web servers. The primary function of a web server is to deal with requests of clients using the HTTP protocol. Therefore, HAS can traverse easily through the network address translation (NAT) routers and firewalls widely deployed in the Internet. Moreover, it uses a flexible rate-adaptation scheme that delivers the highest quality video possible despite changing network conditions dynamically without stops and stutters.

HTTP adaptive streaming partitions the video content, such as a movie, into many small file segments, each segment containing a short interval of playback time. Usually, this short interval is called segment duration. The content is made available at a variety of different bit rates. The clients automatically select from the alternatives the next segment to download and playback based on current network conditions [1]. Segmented transmission and application-layer adaption create a different traffic pattern than traditional progressive video download where the entire video is downloaded with a single request. This characteristic of HAS leads to many successive download-and-wait operations creating an on-off traffic pattern in steady-state [2].

HTTP adaptive streaming technique enables continuous delivery of live or video on demand (VoD) contents over the Internet. This relies on the client terminals select the

ISSN: 1975-0080 IJMUE

segment according to the measured available bandwidth. This works well when just one video service is used. But if two or more video players share the same bottleneck link, and depending on the temporal overlap of their ON-OFF periods, they may overestimate the available bandwidth. This overestimation makes clients oscillate between different quality levels and unfairness between players [7]. On the other hand, limit bandwidth causes more packet loss leading to poor video quality on the client and the bandwidth wastage [8]. Thus, a large scale of on-off traffic pattern HAS streams competing for available bandwidth in the same bottleneck will lead to three performance problems: player instability, unfairness between players, and resource underutilization. These problems affect not only the performance of video application, but also the performance of the Internet as a whole.

In this paper, we propose a method to improve performance: instability and utilization. This shaping method is based on two kinds of QoE guarantees. Therefore, the fairness is not considered in this paper. The shaper limits the throughput for each segment to the target shaping rate. This rate can lengthen the ON period and achieve the goal of QoE optimization. After reaching a steady state, the download duration will be roughly lengthened to the segment duration as long as the available bandwidth is higher than the shaping rate. On the other hand, this method enables the delivery of optimal quality of experience to users. In addition, we assume that a player receives most successive video from the same video server. Finally, our method is server-based, not requiring any cooperation from the client. This makes it particularly beneficial for HAS servers, thus providing motivation for service provider to deploy.

The remainder of this paper is organized as follows. In Section II we introduce an overview of relevant work in the area of improving the performance of HTTP Adaptive streaming. Section III presents the traffic shaping method. In Section IV the simulation framework and simulation results are provided. Finally, Section V summarizes the main findings and contributions of this article.

2. Related work

HTTP adaptive streaming has been a hot topic over last few years, most of previous works have focused on the observation and the improvement of a single connection [3-6]. However, recent studies have identified some performance problems arise in huge multi-user scenarios [9-11]. Multiple on-off traffics competing for available bandwidth in bottleneck lead to poor fairness, stability and utilization.

To the best of our knowledge, there have only been two prior studies focusing on improving aforementioned three metrics through traffic shaping method. Houdaille and Gouache [12] proposed to implement a dedicated “bandwidth manager” in the residential gateway reducing the instability and unfairness with competing adaptive streaming players. Akhshabi and Begen [13] proposed a server-based traffic shaping method to mitigate the instability problem. Above two literatures did not consider the perceptual video quality and user grade. In this paper, a traffic shaping method is proposed to improve the stability and resource utilization. More importantly, our method is QoE-aware. To our knowledge, the QoE-aware traffic shaping method in this paper is the first one to consider both the quality of experience and the server-side bottleneck.

3. Optimizing Methods

This section describes how to deal with the competition in a server-side bottleneck by implementing a “bandwidth manager” that can limit the bandwidth for individual clients by means of a traffic shaping method. The shaper has the ability to get knowledge of the server-side bottleneck and the segment quality level requested by clients concurrent. And it can take appropriate actions on them.

This paper concentrates on the effect of controlling the bandwidth from the bottleneck on the server-side. The shaper is able to limit the throughput for each segment to a target shaping rate. Firstly, the shaper limits the throughput for each segment to the initial shaping rate. Then the remaining bandwidth is allocated to appropriate clients according to maximization algorithms for different goals. There are two goals in this paper. The one is to maximize the global perceptual video quality on the client in the scenario where users have no grade. Another is to achieve revenue maximization in the scenario where users have grade. As soon as the target shaping rates are obtained, the shaper applies traffic shaping so that the clients get best sustainable quality of experience.

3.1. Initial shaping rate selection

To eliminate the effect of on-off pattern and the limit bandwidth on the global performance, the average encoding rate corresponding to the profile requested by client is chosen as initial shaping rate. However, there are two factors to consider. Firstly, client determines the profile in a conservative way. It is actually lower than the estimated available bandwidth. So the initial shaping rate is chosen a little higher than the average encoding rate. For the i-th client, it's average encoding rate corresponding to the current profile is denoted as $R_c(i)$. Then, the initial shaping rate is denoted as $R_I = R_c/\alpha$, where α is an adjustment parameter. We set α = 0.9 in following simulations. Secondly, when the sum of initial shaping rate is higher than the bandwidth on the server-side, bottleneck occurs on server-side. In this case, the shaper sends server a request with one lower profile and limits the initial shaping rate to a little higher than the average encoding rate of that lower profile until the sum of initial shaping rate is no more than the bottleneck bandwidth (when the requested profile is the lowest one, then just ignore it). At this moment, the initial shaping rate is denoted as $R_I = R_{c-1}/\alpha$, where $R_{c-1}(i)$ is the average encoding rate corresponding to the one lower profile. Although the above traffic shaping method makes the download duration of shaped chunks near to their segment duration, it will cause resource wastage on server-side. The following two parts allocate the remaining bandwidth to certain clients in the form of increasing initial shaping rate according to two different kinds of goals. After a few of segment transfer, the download duration of shaped segments approximately achieve to their segment duration again. The two goals are overall perceptual video quality maximization and revenue maximization.

3.2. Global perceptual video quality maximization algorithm

3.2.1. Formulation: This section assumes that all clients have no grade. Let $\Psi = \{s_1, s_2, \ldots, s_i, \ldots\}$ be current clients which request next segments from the server. The bandwidth on server-side is set to C. The remaining bandwidth after initial shaping rate selection is $C - \sum_i R_I(i)$, denoted as B. Our goal is to allocate the remaining bandwidth to certain clients with bandwidth allocation policy π, so as to maximize the global perceptual video quality, subject to the remaining bandwidth constraint $R(\pi) \leq B$. π is the policy vector, which consists of bandwidth allocation policy for each client. Let $\pi = \{\pi_l\}_{s_l \in \Psi}$, where π_l is a bandwidth allocation policy: π_l = 1 indicates that we will allocate bandwidth to the client s_l, while π_l = 0 indicates that we will not. $R(\pi)$ is the bit-rate function under policy π and can be expressed as follow:

$$R(\pi) = \sum_i \pi_i \Delta R_i \tag{1}$$

where,

$$\Delta R_i = \begin{cases} R_{c+1}(i) - R_c(i) & if \ \frac{1}{\alpha}\sum_i R_c(i) < C \\ R_c(i) - R_{c-1}(i) & if \ \frac{1}{\alpha}\sum_i R_c(i) > C \end{cases} \tag{2}$$

3.2.2. Solution: We now discuss the solution of the optimization problem

$$\max_{\pi} QoE(\pi) \quad \text{s.t.} \quad R(\pi) \leq B \tag{3}$$

Where $QoE(\pi)$ is the global perceptual video quality on the client under the policy π and $R(\pi)$ is defined in (1). Amongst the various quality metrics commonly used (PSNR, MSE, VQM, and SSIM), we have chosen the SSIM (structural similarity index measure) to measure the video quality on the client. It was developed by Wang et al. [14] and is considered to correlate relatively well with the quality perception of the human visual system (HVS). To achieve the global perceptual video quality maximization on the client through using remaining bandwidth allocation policy, we first derive the relationship between the SSIM and average encoding rate, and then we use that relationship to establish the remaining bandwidth allocation policy. For a range of predefined bit-rate, the SSIM quality indexes of a video sequence are measured at a various resolutions. Such results are represented in Figure 1. The average encoding rates vary in the interval of 250 kbps. It can be seen that the curve increases shapely when average encoding rates are below 1000kbps. Then the curve becomes smooth after 1000kbps. It proves that the SSIM increases more when average encoding rate increases to the next one when it is smaller than 1000kbps. Thus, the results imply that client will perceive an obvious improvement in subjective quality of experience when the segment's profile increases to the next from a smaller one. The subjective quality of experience will be improved a little when the current profile is high.

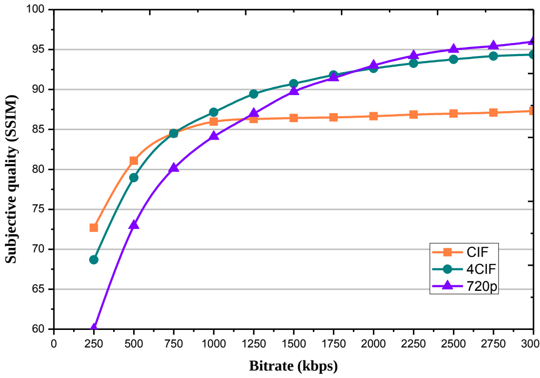

Figure 1. Variation of the SSIM as function of the average encoding rate

Finally, the remaining bandwidth allocation policy is established that it allocate the remaining bandwidth to the clients requesting lower profile. It will allocate $R_{c+1}(l) - R_c(l)$ or $R_c(l) - R_{c-1}(l)$ to the l-th client. The value is determined by the relationship between the bandwidth on server-side and the sum of the average encoding rate corresponding to the profile requested by clients.

3.3. Revenue maximization algorithm

3.3.1. Formulation: This section assumes that all clients have grade. Let $\Psi = \{s_1, s_2, \ldots\}$ be the set of the current clients which request next segment from the server. The grade of each client is described by the vector $U = \{u_1, u_2, \ldots\}$. There are m grades for clients. It means that $u_i \in (1, 2, \cdots, m)$ and $u_i \in U$. Profits produced by clients are given by vector $P = \{p_1, p_2, \ldots\}$. Let $R_c(i)$ is the average encoding rate corresponding to the i-th current client's requesting video profile. The bandwidth on server-side is set to C. The remaining bandwidth after initial shaping rate allocation is $C - \sum_i R_I(i)$, denoted as B. Our goal is to allocate the remaining bandwidth to current clients with bandwidth allocation policy π, so as to maximize the total expected reward, subject to the remaining bandwidth constraint $R(\pi) < B$. π is the policy vector, which consists of the remaining bandwidth allocation policy for each current client. Let $\pi = \{\pi_l\}_{s_l \in \Psi}$, where π_l is a remaining bandwidth allocation policy: $\pi_l = 1$ indicates that we will allocate bandwidth to the client s_l, while $\pi_l = 0$ indicates that we will not. $R(\pi)$ is the bit-rate function under policy π and can be expressed as (1). And the total expected reward $W(\pi)$ under the policy π is expressed as follow:

$$W(\pi) = \sum_i p_i \cdot \pi_i \tag{4}$$

3.3.2. Solution: We now discuss the solution of the optimization problem

$$\max_{\pi} W(\pi) \quad \text{s.t.} \quad R(\pi) \leq B \tag{5}$$

where $W(\pi)$ have been defined in (4) and $R(\pi)$ is the same with (1). Therefore, the optimization problem in (5) is a 0-1 Knapsack problem. Greedy algorithm, dynamic programming, and genetic algorithm are common algorithms to solve 0-1 Knapsack problem. In terms of computational complexity, the greedy algorithm is chosen to solve this problem in this paper which is described in Algorithm 1 (revenue–based bandwidth allocation algorithm). Therefore, the complexity of this greedy algorithm is $O(NlogN)$, where N is the number of current clients requesting for segments from server. In terms of memory, this algorithm only requires a one dimensional array to record the solution string.

Table 1. Revenue-based bandwidth allocation algorithm

```
Algorithm 1 Revenue-based bandwidth allocation algorithm
Initialization: $I(M) = 0$
for $t = 1 \cdots M$ do
    Calculate $r(t) = p(t)/\Delta R(t)$
end for
Sort clients in non-increasing order of $r(t)$
for $k = 1 \cdots N$ do
    $I(k) += r(k)$
    if $I(k) < B$ then
        Allocate $\Delta R(k)$ bit-rate to $k$-th client
    else
        Proceed to the next one
    end if
end for
```

3.4. Traffic shaping on the server

Our idea is leverage shaper to implement a "bandwidth manager" that can allocate the bandwidth for individual sessions by means of a simple traffic shaping mechanism. The shaper has the ability to get appropriate knowledge of the bandwidth capacity on the server and the profile requested by each client. It is able to determine a set of target bit-rate for sharing the bandwidth among the streaming session. According to different kinds of QoE guarantee, the target shaping rate for each client is established, denoted as $\mathrm{R_s(i)}$. It can be expressed as follow:

$$R_s(i) = R_I(i) + \pi_i \Delta R(i) \tag{5}$$

As soon as target bit-rate for each client is known, the shaper applies traffic shaping so that the available bandwidth perceived by the clients lead them to adopt the optimal bit-rate.

4. Evaluation

4.1. Simulation setup

In the following experiments, we use the ns-2 network simulator [15] to analyze the performance of HAS streams in a multi-user server scenario. The simulation setup is based on the observation that network bottlenecks are close to the server. Figure 2 shows the network used in our simulation. The links between clients and R2 router are provisioned for more than the highest media bit-rate in all media profile. The link between the router R1 and the router R2 is over-provisioned and can support the highest media bit-rate for all clients. The link between the HTTP server and R1 is simulated bottleneck link on the server-side with a capacity of 100Mbps (we have also tested with a 1Gbps bottleneck link, but the trends are the same, i.e., only the total number of clients are scaled up). The delays between the clients and the server are normally distributed with an average of $\mu = 55\text{ms}$, and a variation of $\sigma = 5\text{ms}$. Distributing the delays prevents phase effects in the simulation, and it also seems to be a reasonable assumption for an ADSL access network [11]. The server router uses a 'tail drop' packet drop policy and a buffer size equals to two times the bandwidth delay product (BDP). Setting the buffer size to 2 BDP ensures that the network throughput is not limited by the size of the buffer. In addition, we model client arrivals as a Poisson process. Finally, each HAS video is encoded at six different quality levels. They are presented in Table 2.

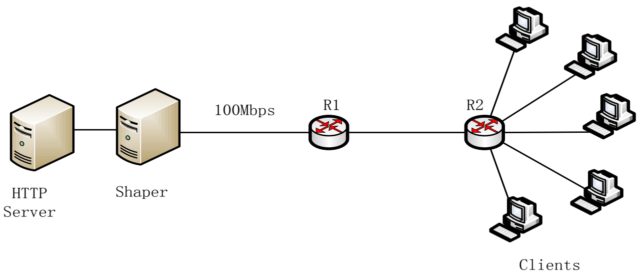

Figure 2. Simulation setup

4.2. Measurement metrics

In this section, to look at the impact of shaping methods on the adaptive HTTP streaming, we choose a classic adaption strategy [11] that is very similar to the strategy used by Adobe's HTTP Dynamic Streaming [16, 2]. The estimated available bandwidth is related with the last estimated bandwidth, the size of current segment and its download time. After the bandwidth is estimated, the strategy finds the size of the next segment using a HTTP HEAD request and chooses to download the segment in the highest bit rate (quality) possible (shown in Table 2).

Table 2. Video qualities

Quality level	Average encoding bit rate	Quality level	Average encoding bit rate	Quality level	Average encoding bit rate
0	250 kbps	2	750 kbps	4	1500 kbps
1	500 kbps	3	1000 kbps	5	3000 kbps

In addition, we use two metrics to measure the performance when multiple HAS traffic competing network resource at the server-side bottleneck. The first one is instability metric, which is defined as the aggregate number of quality profile fluctuation divided by the total number of chunk requests. The second one is the utilization metric, which is defined as aggregate throughput divided by the available bandwidth.

4.3. Scenario where clients with no grade

The first simulation is the scenario where all clients are with no grade. In order to alleviate the server-side bottleneck effect and achieve the goal of maximizing global perceptual video quality on the client, we use perceptual video quality maximization policy to determine the optimal shaping rate. From experiment results, we analyze the effect of this policy on the network performance and video quality on the client.

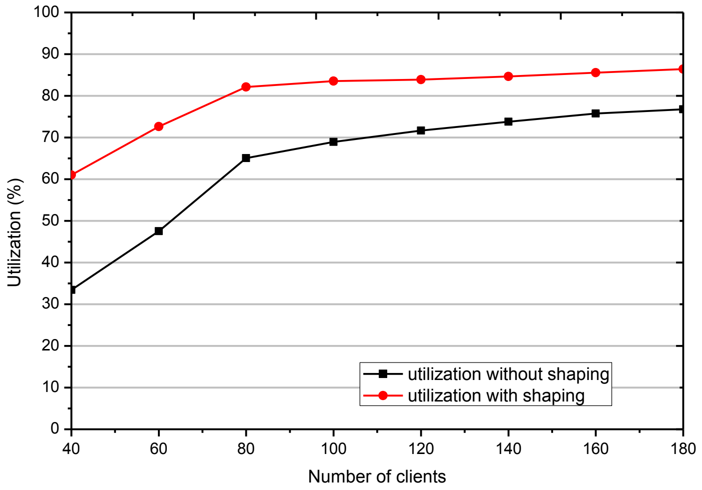

Figure 3. Utilization metric of the perceptual video quality maximization

Figure 3 shows the utilization metric as a function of the number of clients while the perceptual video quality maximization algorithm is used in the shaper. It shows both the shaped case and unshaped case. From the graph, we can see that utilization metrics increase as the number of clients is increased for the shaped case and unshaped case. This is related to the fact that the effect of off period on utilization metric during the transmission of HAS streams decreases as the number of competing clients increase. However, the bottleneck on the server-side occurs when a large number of clients are requesting segment simultaneously. Thus, after traffic shaping, the utilization metric gets an average of 16% improvement compared with the unshaped case. This phenomenon occurs because the effect of server-side bottleneck on performance is relieved in the shaped case. For the shaped case, the off period is greatly reduced in steady state and the dropped packets for the bottleneck become less.

Figure 4 shows the instability metric as a function of the number of clients while the perceptual video quality maximization algorithm is used in shaper. It includes shaped and unshaped case. The results in the graph show that the shaped case gets an average of 10% improvement in stability metric compared with the unshaped case. This improvement is obtained because clients with perceptual video quality maximization shaped policy achieve their steady states after several segment transmissions. After this steady state achieved, the client's optimal shaping rate approximately equals to the average encoding rate corresponding to the segment profile requested by client. So the client's profile will not change unless there are some clients requesting segments join in or leave the network.

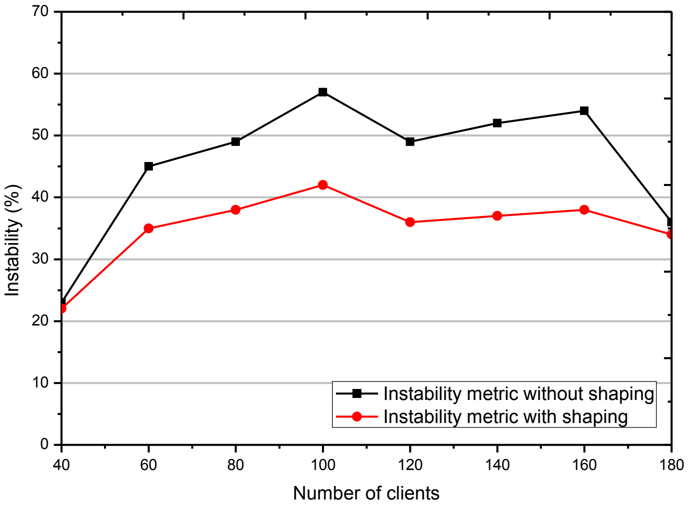

Figure 4. The instability metric utilization metric of the perceptual video quality maximization

Figure 5 shows the change of segment quality level with the increasing of the number of clients. From the graph, we can see the decrease in the relative number of lower quality and the increase in the relative number of next higher one. Although the relative number of highest quality segments is a little lower compared with unshaped case, it satisfy the goal of global perceptual video quality maximization on the client.

The 4.3 section showed a better system performance when limiting the client's throughput to a certain bit-rate according the perceptual video quality maximization algorithm. The performance improvement is obtained in utilization and stability. In addition, it satisfies the goal of maximizing the global perceptual video quality on the client.

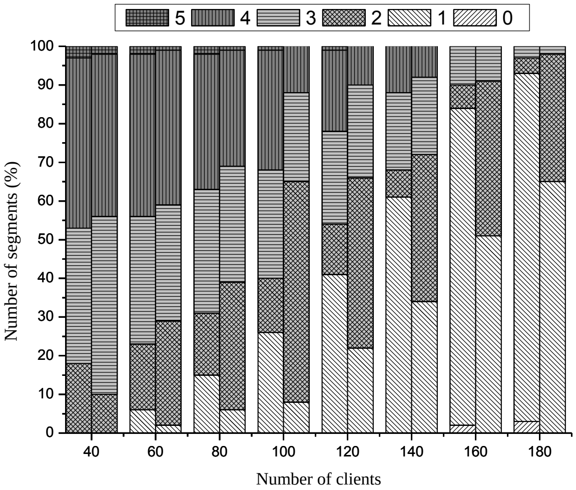

Figure 5. Segment quality utilization metric of the perceptual video quality maximization

4.4. Scenario where clients with grade

The second simulation is the scenario where clients are classified into three grades according to the profit it makes. They are VVIP, VIP and ordinary clients. The i-th client's grade is denoted as c_i where $c_i \in \{0,1,2\}$. Assuming the profit gotten from i-th client is $c_i \times R_s(i)$. In order to alleviate the server-side bottleneck effect and guarantee better service for higher-grade clients, we use the shaping mechanism according to the revenue maximization algorithm to determine the optimal shaping rate. There are 100 clients in this experiment, 10 clients as VVIP users, 15 clients as VIP users and the remaining ones are ordinary users. Then, we observe the effect of this policy on the network performance and video quality on the client by analyzing simulation results.

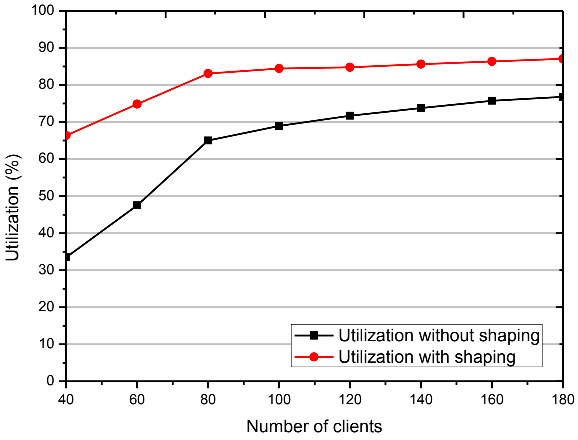

Figure 6. Utilization metric of the revenue maximization

Figure 6 shows the utilization metric as a function of the number of clients while the revenue maximization algorithm is used in shaper. It includes shaped and unshaped case. From the graph, we can see that, in average, utilization metrics in the shaped case is higher about 17% than the unshaped case. This phenomenon occurs because the effect of server-side bottleneck on performance is relieved and the policy based on revenue maximization algorithm decreases the packet loss during the transmission. In addition, the utilization increases more when the number of clients is low. The degree of improvement in utilization becomes is limited when the number increases.

Figure 7 shows the instability metric as a function of the number of clients while the revenue maximization algorithm is used in shaper. The results in the graph show that the shaped case gets an average of 14% improvement in stability metric compared with the unshaped case. This improvement is obtained because clients with shaped policy will achieve their steady states soon after several segment transmissions. After that, the client's optimal shaping rate will not change unless there are clients requesting segments join in or leave the network. So is the client profile.

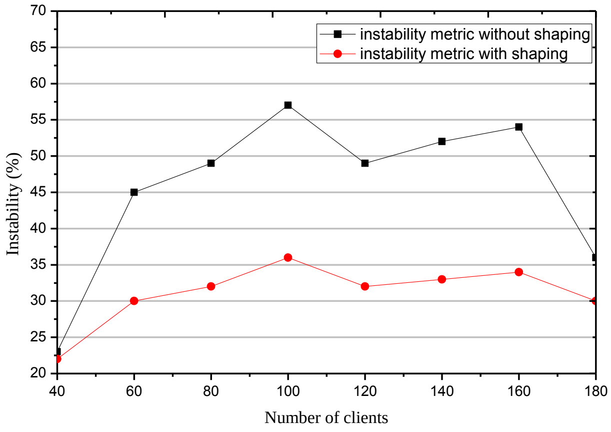

Figure 7. Instability metric of the revenue maximization

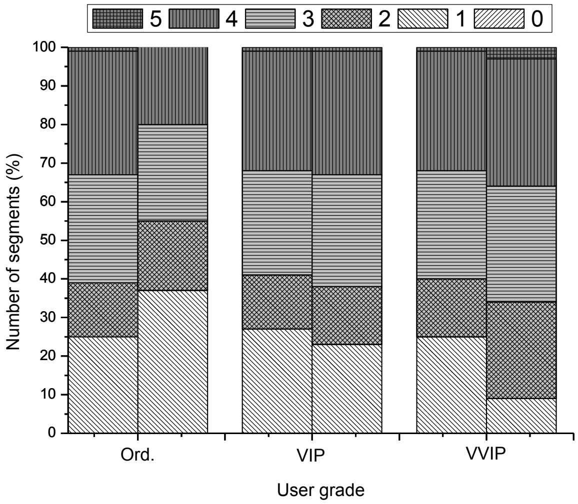

Figure 8. Segment quality of the revenue maximization

Figure 8 shows the change of segment quality level with the increasing of the number of clients. From the graph, we can see all three kinds of clients have a similar proportion before shaping. After using the revenue maximization algorithm, the proportion of relative higher profile increases for VVIP and VIP clients. It satisfies the goal of revenue maximization on the client.

The 4.4 section showed a better system performance when limiting the client's throughput to a certain shaping rate according to the revenue maximization algorithm. The performance improvement is obtained in utilization and stability. In addition, it satisfies the goal of the revenue maximization on the client.

5. Conclusion

In this paper, a QoE-aware traffic shaping method is proposed to improve the performance of HTTP adaptive streaming. The two algorithms are described to determining the optimal shaping rate. The method aims to increase the utilization and stability, and guarantee maximization of perceptual video quality on the client. We evaluated the effectiveness of this method in simulation experiments. Simulation results prove that our method improve the performance. Meanwhile, it guarantees the video quality on the client. However, we assume that a player receives most successive video from the same video server. In our future work, we will consider the broader gains of CDNs.

References

[1] S. Benno, J. O. Esteban and I. Rimac, "Adaptive streaming: The network HAS to help", Bell Labs Technical Journal, vol. 16, no. 2, (**2011**), pp. 101-114.

[2] S. Akhshabi, A. C. Begen and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http", in Proceedings of the second annual ACM conference on Multimedia systems, ACM, (**2011**), pp. 157-168.

[3] C. Liu, I. Bouazizi and M. Gabbouj, "Rate adaptation for adaptive HTTP streaming", Proceedings of the second annual ACM conference on Multimedia systems, ACM, (**2011**), pp. 169-174.

[4] K. Miller, E. Quacchio, and G. Gennari, *et al.*, "Adaptation algorithm for adaptive streaming over HTTP", Packet Video Workshop (PV), 2012 19th International. IEEE, (**2012**), pp. 173-178.

[5] Esteban, Jairo, S. A. Benno, A. Beck, Y. Guo, V. Hilt and I. Rimac, "Interactions between HTTP adaptive streaming and TCP", in Proceedings of the 22nd international workshop on Network and Operating System Support for Digital Audio and Video, (**2012**), pp. 21-26.

[6] S. Akhshabi, L. Anantakrishnan, A. C. Begen, *et al.*, "What happens when HTTP adaptive streaming players compete for bandwidth?", in Proceedings of the 22nd international workshop on Network and Operating System Support for Digital Audio and Video. ACM, (**2012**), pp. 9-14.

[7] T. Kupka, P. Halvorsen and C. Griwodz, "An evaluation of live adaptive HTTP segment streaming request strategies", 2011 IEEE 36th Conference on Local Computer Networks (LCN), IEEE, (**2011**), pp. 604-612.

[8] K. Evensen, D. Kaspar, C. Griwodz, *et al.*, "Improving the performance of quality-adaptive video streaming over multiple heterogeneous access networks", in Proceedings of the second annual ACM conference on Multimedia systems, (**2011**), pp. 57-68.

[9] J. Jiang, V. Sekar and H. Zhang, "Improving fairness, efficiency, and stability in http-based adaptive video streaming with festive," in Proceedings of the 8th international conference on Emerging networking experiments and technologies, ACM, (**2012**), pp. 97-108.

[10] T. Kupka, P. Halvorsen and C. Griwodz, "Performance of on-off traffic stemming from live adaptive segmented HTTP video streaming", 2012 IEEE 37th Conference on Local Computer Networks (LCN), IEEE, (**2012**), pp. 401-409.

[11] R. Houdaille and S. Gouache, "Shaping http adaptive streams for a better user experience," in Proceedings of the 3rd Multimedia Systems Conference, ACM, (**2012**), pp. 1-9.

[12] A. Saamer, L. Anantakrishnan, C. Dovrolis and A. C. Begen, "Server-based traffic shaping for stabilizing oscillating adaptive streaming players," in Proceeding of the 23rd ACM Workshop on Network and Operating Systems Support for Digital Audio and Video, ACM, (**2013**), pp. 19-24.

[13] Z. Wang, A. C. Bovik, H. R. Sheikh, *et al.*, "Image quality assessment: From error visibility to structural similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, (**2004**), pp. 600-612.
[14] Information Sciences Institute, University of Southern California, (**2006**), The Network Simulator - ns-2.
[15] "HTTP dynamic streaming on the Adobe Flash platform", (**2010**), http://www.adobe.com/products/httpdynamicstreaming/pdfs/httpdynamicstreaming_wp_ue.pdf.

Authors

Xinying Liu

She received her M.S. degree in Communication and Information system from Beijing University of Posts and Telecommunications in 2007. Now she is a Ph.D. candidate in the School of Information and Communication at informatics at Beijing University of Posts and Telecommunications. Her current research interests include network protocol and multimedia communication.

Aidong Men

He received his B.S. in Electronic Information Engineering (1987) and Ph.D. in Communication and Electronic System (1994) from Beijing University of Posts and Telecommunications. Now he is professor of the Multimedia Communication Center in Beijing University of Posts and Telecommunications. His current research interests include multimedia communication, digital television, and image and video signal processing.