International Journal of Scientific Engineering and Research (IJSER)
www.ijser.in
Issn (Online): 2347-3878
Volume 2 Issue 11, November 2014

# Efficient Video Streaming and Social Video Sharing

**Shashikantha Raddi[1], Suresh Reddy[2]**

[1]Asst. Professor, CSE Dept, SMSMPITR, Akluj, Solapur, Maharashtra, India
[2]Asst. Professor, CSE Dept, SMSMPITR, Akluj, Solapur, Maharashtra, India

**Abstract:** ***While demands on video traffic over mobile networks have been souring, the wireless link capacity cannot keep up with the traffic demand. The gap between the traffic demand and the link capacity, along with time-varying link conditions, results in poor service quality of video streaming over mobile networks such as long buffering time and intermittent disruptions. Leveraging the cloud computing technology, we propose a new mobile video streaming framework, dubbed AMES-Cloud, which has two main parts: adaptive mobile video streaming (AMoV) and efficient social video sharing (ESoV). AMoV and ESoV construct a private agent to provide video streaming services efficiently for each mobile user. For a given user, AMoV lets her private agent adaptively adjust her streaming flow with a scalable video coding technique based on the feedback of link quality. Likewise, ESoV monitors the social network interactions among mobile users, and their private agents try to prefetch video content in advance. We implement a prototype of the AMES-Cloud framework to demonstrate its performance. It is shown that the private agents in the clouds can effectively provide the adaptive streaming, and perform video sharing (i.e., prefetching) based on the social network analysis.***

**Keywords:** Video Streaming, Social Viseo Sharing

## 1. Introduction

Over the past decade, increasingly more traffic is accounted by video streaming and downloading. In particular, video streaming services over mobile networks have become prevalent over the past few years [1]. While the video streaming is not so challenging in wired networks, mobile networks have been suffering from video traffic transmissions over scarce bandwidth of wireless links. Despite network operators' desperate efforts to enhance the wireless link bandwidth (e.g., 3G and LTE), soaring video traffic demands from mobile users are rapidly overwhelming the wireless link capacity. While receiving video streaming traffic via 3G/4G mobile networks, mobile users often suffer from long buffering time and intermittent disruptions due to the limited bandwidth and link condition fluctuation caused by multi-path fading and user mobility [2] [3] [4]. Thus, it is crucial to improve the service quality of mobile video streaming.

While using the networking and computing resources efficiently [5] [6] [7] [8]. Recently there have been many studies on how to improve the service quality of mobile video streaming on two Aspects:_ Scalability: Mobile video streaming services should support a wide spectrum of mobile devices; they have different video resolutions, different computing powers, different wireless links (like 3G and LTE) and so on. Also, the available link capacity of a mobile device may vary over time and space depending on its signal strength, other user's traffic in the same cell, and link condition variation. Storing multiple versions (with different bit rates) of the same video content may incur high overhead in terms of storage and communication.

To address this issue, the Scalable Video Coding (SVC) technique (Annex G extension) of the H.264 AVCvideo compression standard [9] [10] [11] defines a base layer (BL) with multiple enhance layers (ELs). These sub streams can be encoded by exploiting three scalability features: (i) spatial scalability by layering image resolution (screen pixels), (ii) temporal scalability by layering the frame rate, and (iii) quality scalability by layering the image compression. By the SVC, a video can be decoded/played at the lowest quality if only the BL is delivered. However, the more ELs can be delivered, the better quality of the video stream is achieved._ Adaptability: Traditional video streaming techniques designed by considering relatively stable traffic links between servers and users perform poorly in mobile environments [2]. Thus the fluctuating wireless link status should be properly dealt with to provide 'tolerable" video streaming services. To address this issue, we have to adjust the video bit rate adapting to the currently time-varying available link bandwidth of each mobile user. Such adaptive streaming techniques can effectively reduce packet losses and bandwidth waste. Scalable video coding and streaming techniques can be jointly combined to accomplish effectively the best possible quality of video streaming services. That is, we can dynamically adjust the number of SVC layers.

Depending on the current link status [9] [12].However most of the proposals seeking to jointly utilize the video scalability and adaptability rely on the active control on the server side. That is, every mobile user needs to individually report the transmission status (e.g. Packet loss, delay and signal quality) periodically to the server, which predicts the available bandwidth for each user. Thus the problem is that the server should take over the substantial processing overhead, as the number of users increases. Cloud computing techniques are poised to flexibly provide scalable resources to content/service providers, and process offloading to mobile users [13] [14] [15] [16] [17] [18] [19]. Thus, cloud data centers can easily provision for large-scale real-time video services as investigated in [9] [20]. Several studies on mobile cloud computing technologies have proposed to generate personalized intelligent agents for servicing mobile users, e.g., Cloudlet [21] and Stratus [22]. This is because, in the cloud, multiple agent instances (or threads) can be maintained dynamically and efficiently depending on the time-varying user demands. Recently social network services (SNSs) have been increasingly popular. There have been proposals to improve the quality of content delivery using SNSs [23] [24]. In SNSs, users may share, comment or re-post videos among friends and members in the same group, which implies a user, may watch a video that her friends have recommended (e.g. [24]). Users in SNSs can also follow famous and popular users based on their interests (e.g., an official Facebook or twitter account that shares the newest pop music videos), which is likely to be watched by its followers. In this regard, we are further motivated to exploit the relationship among mobile users from their SNS activities in order to prefetch in advance the beginning part of the video or even the whole video to the members of a group who have not seen the video yet. It can be done by a background job supported by the agent (of a member) in the cloud; once the user clicks to watch the video, it can instantly start playing. In this paper, we design a adaptive video streaming and prefetching framework for mobile users with the above objectives in mind, dubbed AMES-Cloud. AMES-Cloud constructs a private agent for each mobile user in cloudcomputing environments, which is used by its two main parts: (i) AMoV (adaptive mobile video streaming), andESoV (efficient social video sharing). The contributions of this paper can be summarized as follows:

- AMoV offers the best possible streaming experiences by adaptively controlling the streaming bit rate depending on the fluctuation of the link quality. AMoV adjusts the bit rate for each user leveraging the scalable video coding. The private agent of a user keeps track of the feedback information on the link status. Private agents of users are dynamically initiated and optimized in the cloud computing platform. Also the real-time SVC
- Coding is done on the cloud computing side efficiently.
- AMES-Cloud supports distributing video streams efficiently by facilitating a 2-tier structure: the first tier is a content delivery network, and the second tier is a data center. With this structure, video sharing can be optimized within the cloud. Unnecessary redundant downloads of popular videos can be prevented [25] [26].
- Based on the analysis of the SNS activities of mobile users, ESoV seeks to provide a user with instant playing of video clips by prefetching the video clips in advance from her private agent to the local storage of her device.

The strength of the social links between users and the history of various social activities can probabilistically Determine how much and which video will be prefetched. The rest of the paper is organized as follows. We first introduce related work in SectionII, and explain the AMES-Cloud framework in Section

## 2. Related Framework

### A. Adaptive Video Streaming Techniques

In the adaptive streaming, the video traffic rate is adjusted on the fly so that a user can experience the maximum possible video quality based on his or her link's time-varying bandwidth capacity [2]. There are mainly two types of adaptive streaming techniques, depending on whether the adaptively is controlled by the client or the server. The Microsoft's Smooth Streaming [27] is a live adaptive streaming service which can switch among different bit rate segments encoded with configurable bit rates and video resolutions at servers, while clients dynamically request videos based on local monitoring of link quality. Adobe and Apple also developed client-side HTTP adaptive live streaming solutions operating in the similar manner. There are also some similar adaptive streaming services where servers control the adaptive transmission of video segments, for example, the Quavlive Adaptive Streaming. However, most of these solutions maintain multiple copies of the video content with different bit rates, which brings huge burden of storage on the server. Regarding rate adaptation controlling techniques, TCP-friendly rate control methods for streaming services over mobile networks are proposed [28] [29], where TCP throughput of a flow is predicted as a function of packet loss rate, round trip time, and packet size. Considering the estimated throughput, the bit rate of the streaming traffic can be adjusted. A rate adaptation algorithm for conversational 3G video streaming is introduced by [30]. Then, a few cross-layer adaptation techniques are discussed [31] [32], which can acquire more accurate information of link quality so that the rate adaptation can be more accurately made. However, the servers have to always control and thus suffer from large workload.

Recently the H.264 Scalable Video Coding (SVC) technique has gained a momentum [10]. An adaptive video streaming system based on SVC is deployed in [9], which studies the real-time SVC decoding and encoding at PC servers. The work in [12] proposes a quality-oriented scalable video delivery using SVC, but it is only tested in a simulated LTE Network. Regarding the encoding performance of SVC, Cloud Stream mainly proposes to deliver High-quality streaming videos through a cloud-based SVC proxy [20], which discovered that the cloud computing can significantly improve the performance of SVC coding. The above studies motivate us to use SVC for video streaming on top of cloud computing.

### B. Mobile Cloud Computing Techniques

The cloud computing has been well positioned to provide video streaming services, especially in the wired Internet because of its scalability and capability [13]. For example, the quality-assured bandwidth auto-scaling for VoD streaming based on the cloud computing is proposed [14], and the CALMS framework [33] is a cloud assisted live media streaming service for globally distributed users. However, extending the cloud computing-based services to mobile environments requires more factors to consider: wireless link dynamics, user mobility, and the limited capability of mobile devices [34] [35]. More recently, new designs for users on top of mobile cloud computing environments are proposed, which virtualize private agents that are in charge of satisfying the requirements (e.g.QoS) of individual users such as Cloudlets [21] and Stratus [22]. Thus, we are motivated to design the AMES-Cloud framework by using virtual a gents in the cloud to provide adaptive video streaming services.

## 3. AMES-Cloud Framework

In this section we explain the AMES-Cloud framework includes the Adaptive Mobile Video streaming (AMoV) and the Efficient Social Video sharing (ESoV).the whole video storing and streaming system in the cloud is called the Video Cloud (VC).
In the VC, there is a large-scale video base (VB), which

International Journal of Scientific Engineering and Research (IJSER)
www.ijser.in
Issn (Online): 2347-3878
Volume 2 Issue 11, November 2014

stores the most of the popular video clips for the video service providers (VSPs). A temporal video base (tempVB) is used to cache new candidates for the popular videos, while tempVB counts the access frequency of each video. The VC keeps running a collector to seek videos which are already popular in VSPs, and will re-encode the collected videos into SVC format and store into tempVB first.By this 2-tier storage, the AMES-Cloud can keep serving most of popular videos eternally. Note that management work will be handled by the controller in the VC.

Specialized for each mobile user, a sub-video cloud (subVC) is created dynamically if there is any video streaming demand from the user. The sub-VC has a sub video base (subVB), which stores the recently fetched video segments.

Note that the video deliveries among the subVCs and the VC in most cases are actually not "copy", but just "link" operations on the same file eternally within the cloud data center [36]. There is also encoding function in subVC (actually a smaller-scale encoder instance of the encoder in VC), and if the mobile user demands a new video, which is not in the subVB or the VB in VC, the subVC will fetch, encode and transfer the video. During video Streaming, mobile users will always report link conditions to their corresponding subVCs, and then the subVCs offer adaptive video streams. Note that each mobile device also has a temporary caching storage, which is called local video base (localVB), and is used for buffering and prefetching. Note that as the cloud service may across different places, or even continents, so in the case of a video delivery and prefetching between different data centers, an transmission will be carried out, which can be then called "copy". And because of the optimal deployment of data centers, as well as the capable links among the data centers, the "copy" of a large video file takes tiny delay

## 4. AMOV: Adaptive Mobile Video Streaming

A. **SVC**

Traditional video streams with fixed bit rates cannot adapt to the fluctuation of the link quality. For a particular bit rate, if the sustainable link bandwidth varies much, the video streaming can be frequently terminated due to the packet loss.

In SVC, a combination of the three lowest scalability is called the Base Layer (BL) while the enhanced combinations are called Enhancement Layers (ELs). To this regard, if BL is guaranteed to be delivered, while more ELs can be also obtained when the link can afford, a better video quality can be expected.

By using SVC encoding techniques, the server doesn't need to concern the client side or the link quality. Even some packets are lost, the client still can decode the video and display. But this is still not bandwidth-efficient due to the unnecessary packet loss. So it is necessary to control the SVC-based video streaming at the server side with the rate adaptation method to efficiently utilize the bandwidth.

**B. Adaptability with Monitoring on Link Quality**

We design the mobile client and the subVC with the structure. The link quality monitor at mobile client keeps tracking on metrics including signal strength, packet round-trip-time (RTT), jitter and packet loss with a certain duty cycle. And the client will periodically report to the subVC. Hereby we define the cycle period for the reporting as the "time window", denoted by Twin, Note that the video is also split by temporal segmentation by interval Twin. Once the subVC gets the information of the link quality, it will perform a calculation and predict the potential bandwidth in the next time window. Note that we will use "predicted bandwidth" and "predicted good put" Interchangeably in following parts.

Suppose sequence number of current time window is i, the predicted bandwidth can be estimated by:

BW estimate
i+1 = BWpractical
i
_ [_ _ f (pi; pi□1)
+_ _ g (RTTi; RTTi□1) + _ h (SINRi; SINRi□1)]

Where, _ + _ + = 1 indicating the importance of each factor, p is for packet loss rate, RTT is for RTT, SINR

Is for the signal to interference and noise ratio, and f(), g(), h() are three functions reflecting the value change of each factor compared with that of last time window.

Actually in this paper we deploy a measurement-based prediction that is we directly use BWpractical i of last time window as the BW estimate i+1 of next time window, which is proved with already high accuracy.

**C. Matching between Bandwidth Prediction and SVC Segments**

After obtaining the predicted bandwidth, or say good put, of next time window, subVC will match and decide how many video segments of BL and ELs can be transmitted approximately. We hereby define the term "resolution" to indicate the level of temporal segmentation and the number of ELs. If Twin is small and there are more ELs, we say the SVC-based video source is with a higher resolution. We illustrate two cases of low resolution and a relatively high resolution for matching between the SVC segments and the predicted good put

## 5. ESOV: Efficient Social Video Sharing

**A. Social Content Sharing**

In SNSs, users subscribe to known friends, famous people, and particular interested content publishers as well; also there are various types of social activities among users in SNSs, such as direct message and public posting. For spreading videos in SNSs, one can post a video in the public, and his/her subscribers can quickly see it; one can also directly recommend a video to specified friend(s); furthermore one can periodically get noticed by subscribed content publisher for new or popular videos. Similar to studies in [23] [24], we define different strength levels for those social activities to indicate the probability that the video shared by one user may be watched by the receivers of the one's sharing activities, which is called a "hitting probability", so that subVCs can carry out effective background prefetching at subVB and even localVB. Because after a video sharing activity, there may be a certain delay that the recipient gets to know the sharing, and initiates to watch [38]. Therefore the prefetching in prior will not impact the users at most cases. Instead, a user can click to see without any buffering delay as the beginning part or even the whole video is already prefetched at the localVB. The amount of prefetched segments is mainly determined by the strength of the social activities. And the prefetching from VC to subVC only refers to the "linking" action, so there is only file locating and linking operations with tiny delays; the prefetching from subVC to localVB also depends on the strength of the social activities, but will also consider the wireless link status.

### B. Prefetching Levels

Different strengths of the social activities indicate different levels of probability that a video will be soon watched by the recipient. Correspondingly we also define three prefetching levels regarding the social activities of mobile users:

- "Parts": Because the videos that published by subscriptions may be watched by the subscribers with a not high probability, we propose to only push a part of BL and ELs segments, for example, the first 10% segments.
- "All": The video shared by the direct recommendations will be watched with a high probability, so we propose to prefetch the BL and all ELs, in order to let the recipient(s) directly watch the video with a good quality, without any buffering.

"Little": The public sharing has a weak connectivity among users, so the probability that a user's friends (followers) watch the video that the user has watched or shared is low. We propose to only prefetch the BLsegment of the first time window in the beginning to those who have seen his/her activity in the stream.

The prefetching happens among subVBs and the VB, also more importantly, will be performed from the subVB to localVB of the mobile device depending on the link quality. If a mobile user is covered by Wi-Fi access, due to Wi-Fi's capable link and low price (or mostly for free), subVC can push as much as possible in most cases.

However if it is with a 3G/4G connection, which charges a lot and suffers limited bandwidth, we propose to downgrade the prefetching level to save energy and cost as listed in Table. 1, but users can still benefit from the prefetching effectively. Note that some energy prediction methods can be deployed in order to actively decide whether current battery status is suitable for "parts" or "little" [39]. If a user, A, gets the direct recommendation of a video from another user, B, A's subVC will immediately prefetch the video either from B's subVB, or from the VB (or tempVB) at the level of "all", if A is with Wi-Fi access. However if user A is connected to 3G/4G link, we will selectively prefetch a part of the video segment to A's local storage at the level of "parts". Note that the subscribed videos will be not prefetched when user A is at 3G/4G connection, as it is downgraded from "little" to none.

A better extension of the prefetching strategy by social activities can be designed by an self-updating mechanism from the user's hitting history in an evolutionary manner. This learning-based prefetching is out of the scope of this paper, and will be explored as our future work.

## 6. Video Storage and Streaming Flow by AMOV and EMOS

The two parts, AMoV and EMoS, in AMES-Cloud framework have tight connections and will together service the video streaming and sharing: they both rely on the cloud computing platform and are carried out by the private agencies of users; while prefetching in EMoS, the AMoV will still monitor and improve the transmission considering the link status; with a certain amount of prefetched segments by EMoS, AMoV can offer better video quality.

With the efforts of AMoV and EMoS, we illustrate the flow chart of how a video will be streamed. Note that in order to exchange the videos among the localVBs, subVBs, tempVB and the VB, a video map (VMap) is used to indicate the required segments.

Once a mobile user starts to watch a video by a link, the localVB will first be checked whether there are any prefetched segments of the video so that it can directly start. If there is none or just some parts, the client will report a corresponding VMap to its subVC. if the subVC has prefetched parts in subVB, the subVC will initiate the segment transmission. But if there is also none in the subVB, the tempVB and VB in the center VC will be checked. For a non-existing video in AMES-Cloud, the collector in VC will immediately fetch it from external video providers via the link; after re-encoding the video into SVC format, taking a bit longer delay, the subVC will transfer to the mobile user. Also in AMES-Cloud, if a video is shared among the subVCs at a certain frequency threshold (e.g., 10 times per day), it will be uploaded to the tempVB of the VC; and if it is further shared at a much higher frequency (e.g., 100 times per day), it will be stored with a longer lifetime in the VB. In such a manner, which is quite similar to the leveled CPU cache, the subVB and VB can always store fresh and popular videos in order to increase the Probability of re-usage.

## 7. Conclusion

In this paper, we discussed our proposal of an adaptive mobile video streaming and sharing framework, called AMES-Cloud, which efficiently stores videos in the clouds (VC), and utilizes cloud computing to construct private agent (subVC) for each mobile user to try to offer "non-terminating" video streaming adapting to the fluctuation of link quality based on the Scalable Video Coding technique. Also AMES-Cloud can further seek to provide "nonbuffering" experience of video streaming by background pushing functions among the VB, subVBs and localVB of mobile users. We evaluated the AMES-Cloud by prototype implementation and shows that the cloud computing technique brings significant improvement on the adaptively of the mobile streaming.

The focus of this paper is to verify how cloud computing can improve the transmission adaptability and prefetching for mobile users. We ignored the cost of encoding workload in the cloud while implementing the prototype. As one important future work, we will carry out large-scale implementation and with serious consideration on energy and price cost. In the future, we will also try to improve the SNS-based pre-fetching, and security issues in the AMES-Cloud.

## References

[1] J. Brandt, V. Kahmann, and L. Wolf. A flexible Reflector for media streams. In KIVS, volume PI-61 of Lecture Notes in Informatics, pages 41–48, Mar. 2005
[2] J. Brandt and L. Wolf. Multidimensional Transcoding for Adaptive Video Streaming. In Proceedings of the 17th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'07), Urbana, Illinois, June 2007
[3] World Wide Web Consortium (W3C). Composite Capabili ties/PreferencProfiles (CC/PP).http://www.w3.org/Mobile/CCPP
[4] CISCO, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2011-2016," Tech. Rep., 2012
[5] K. Zhang, J. Kong, M. Qiu, and G.L Song, "Multimedia Layout Adaptation Through Grammatical Specifications," in ACM/Springer Multimedia Systems, vol. 10, no. 3, pp.245–260, 2005
[6] N. Davies, "The Case for VM-Based Cloudlets in Mobile Computing," in IEEE Pervasive Computing, vol. 8, no. 4, pp. 14–23, 2009
[7] B. Aggarwal, N. Spring, and A. Schulman, "Stratus: Energy- Efficient Mobile Communication using Cloud Support," in ACM SIGCOMMDEMO, 2010