In-Node Scalability for Adaptive Bitrate HTTP Streaming in Content Centric Network

Saeed Ullah, Sungwon Lee and Choong Seon Hong*

Department of Computer Engineering, College of Electronics and Information,
Kyung Hee University, South Korea
{saeed, drsungwon, cshong}@khu.ac.kr

Abstract

In Adaptive bitrate HTTP streaming multiple copies with different playback rates are stored for a single video in order to provide scalability to users. In Content Centric Network (CCN) a situation can arise that a user request a video with specific playback rate, CCN node may have the requested video in its cache but with different playback bitrate in such situation CCN node forward the request to get a copy of the video according to the requested playback rate. In this paper we provide in-node scalability solution according to which the video is delivered to the user more quickly and the network bandwidth is utilized more efficiently.

1. Introduction

Many advanced computing devices like, smart phones, tablets, netbooks, ultrabooks and laptops have been introduced in the markets which have the capability of displaying high quality video contents. Also these devices have the capability of supporting different video streaming services and interactive video applications. According to Cisco [1] mobile traffic will grow by factor of 25 until 2017 and video will be the dominant part in this traffic.

Content Centric Networking (CCN) [2-4] is the named data network in which content is got by their name instead of their location. Whenever a user need some content it generates interest packet for it. When a CCN node receives an interest packet it searches the requested contents in its cache, if found it is delivered to the user immediately otherwise forwards the interest packet to other CCN nodes. The intermediate nodes whenever forwarding a video it also store a copy in its cache to provide locally if it receives request for the same video in the near future.

HTTP adaptive bitrate (ABR) streaming [5-7] is gaining popularity and emerging to be the premier video streaming solution. It provides a stable playback experience by delivering high quality video according to network condition and user's device capability and also increase the network utilization. In HTTP ABR when a video is captured it is transcoded into a number of video files each having different playback rate. According to network condition and users' device resolution an appropriate file is chosen for delivery [8]. In CCN a node may get a request for a specific video it may have the same video in its cache but with different playback rate than the requested. In such situation the CCN nodes forwards the request to other CCN nodes to get video of the desired playback rate. In this paper we provide efficient solution in which the node will generate a copy of the video with the desired playback rate locally from the copy in its cache whenever possible. Thus the precious bandwidth will not be wasted in ordering a redundant copy and also the user will get back the video quickly.

2. System Architecture

Figure 1 shows network model of CCN for Adaptive HTTP streaming. There are three parts of the network. A media generating part that consists of video capturing device/camera, transcoding server and HTTP webserver. Transcoding server encodes the video and stores a number of copies with different play back rate in the HTTP webserver. Transcoding and HTTP webserver may be a single machine or separate dedicated machines. Second part of the model is the CCN network which consists of CCN routers. These routers or CCN nodes having cache in which it stores the contents that passes through it.

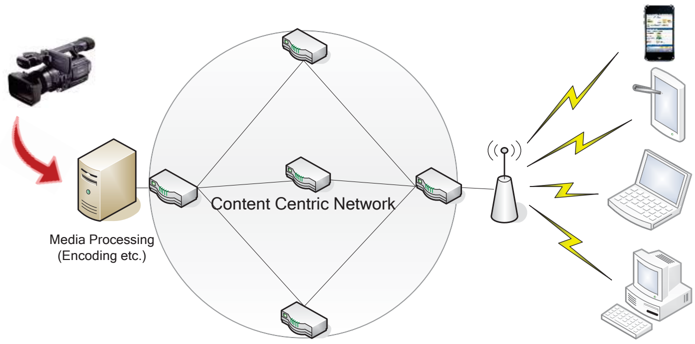

Figure 1: CCN scenario for video delivery system

When the cache fills up then the new contents are replace with the new one with different schemes like least recently used etc. The third part of the network consists of a variety of user devices that having the capability of running HTTP streaming client application.

3. Proposed Scheme

In CCN when a node receives request for a video with playback rate R_j then there can be two scenarios 1) the requested video is not present in its cache 2) the requested video is present in the cache with playback rate R_i. The first case is very simple, the node will forward the request to the node that either have the content or is on the shortest path to the origin of the content. In the second case, if R_i and R_j are equal then it is delivered to the user. If R_i and R_j are not equal then the node will see if R_i is lower than R_j then it is not possible to generate R_j from R_i because Video encoding is a lossy process i.e., the information that is lost during the encoding cannot be recovered therefore we cannot produce copy of a video with higher playback rate from a lower playback rate video. If R_i is higher than R_j then the node will calculate the cost of downloading the video from other nodes against the cost required to encode the video with rate R_j. If cost of encoding the video is higher than downloading it from other/sever node then it will forward the request to other node. And if the encoding computational cost is less than the downloading then it encode the video with the requested playback rate and deliver it to the user.

In video streaming it is very important that the user receive the video in time so that there is no break in video playback. Therefore in our proposed scheme time is the parameter for calculating encoding and downloading costs. Pseudo-code description of our proposed scheme is presented in Algorithm 1.

Algorithm 1: In-node scalability

```
Require: Ri, Rj
  if Rj is not present in cache or Ri < Rj
      forward the request to other nodes
  else
      if Ri = Rj
         deliver the video
      else
         calculate cost of encode(Ri → Rj) & download(Rj)
         if encode(Ri → Rj) > download(Rj)
             forward the request to other nodes
         else
             Rk = encode(Ri → Rj)
             deliver Rk
         end if
      end if
   end if
```

4. Analysis

Our proposed scheme does not require any specialized hardware or any change to the network protocol. There is no complexity or burden caused by our proposed scheme in the case of unavailability of requested contents in the cache. In case the requested content/video is present in cache and needs to be encoded into the requested playback rate then CCN node will have to bear the encoding computational complexity.

Total computational complexity caused by encoding is given as follow

$$C_N = \sum_{x=1}^{N} C(R_i \rightarrow R_j)_x$$

Where C_N is the total complexity cause by converting N videos from rate R_i to R_j. There will be no noticeable effect on the performance of CCN node when N is small because CCN nodes are computationally very strong. However when N is very large then it will effect performance of CCN node. In such situation the CCN node will go for to download content from other nodes and thus will continue performing normally.

5. Conclusion and Future Work

In this paper we presented in-node scalability that generate copy of video of the requested playback rate from the copy that is already stored in cache of a CCN node. Our proposed scheme do not require any

change to the protocol or hardware. Our proposed scheme uses the available computational capability of the CCN node to reduce traffic in the network and provide users' requested contents quickly.
In future we are aiming to perform experiments both numerically and by simulation to evaluate our proposed scheme.

6. Acknowledgement

"This research was funded by the MSIP(Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2013". *Dr. CS Hong is the corresponding author.

7. References

[1] http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.html
[2] Van Jacobson's talk at PARC Forum: "The Good, Bad, and Ugly of Digital Distribution: A Content-centric Networking Perspective on Evolving Network Architecture", February 2011.
[3] Van Jacobson's tutorial at Future Internet Summer School (FISS09) Short course, 2009.
[4] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael F. Plass, Nicholas H. Briggs, Rebecca L. Braynard, Networking Named Content, CoNext, Rome, Italy, 2009.
[5] Hofmann, N. Farber, and H. Fuchs, "A study of network performance with application to adaptive HTTP streaming," in IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), 2011.
[6] A. C. Begen, T. Akgul, and M. Baugher, "Watching video over the web," Internet Computing, vol. 15, pp. 54–63, 2011.
[7] C. Liu, I. Bouazizi, and M. Gabbouj, "Rate adaptation for adaptive HTTP streaming," in Proceedings of the second annual ACM conference on Multimedia systems, pp. 169–174, 2011.
[8] R. K. P. Mok, E. W. W. Chan, and R. K. C. Chang, "Measuring the quality of experience of HTTP video