An analysis of Video Transcoding in multi-core Cloud Environment

Sampa Sahoo
NIT, Rourkela
Odisha, India
sampaa2004@gmail.com

Bibhudatta sahoo
NIT, Rourkela
Odisha, India
bibhudatta.sahoo@gmail.com

Ashok Kumar Turuk
NIT, Rourkela
Odisha, India
akturuk@gmail.com

ABSTRACT

In recent years, watching video online has become a popular form of infotainment. Online Video on Demand (VoD) services allows users to view video content such as user generated videos, movies, TV shows, music videos and live streams anytime and anywhere. The presence of heterogeneous networks, devices, and user preferences, demand different versions (concerning resolution, frame rate, format, etc.) of a source video. Transcoding the process of converting video file from one format to another format, is a time-consuming and QoS-sensitive project. Cloud computing offers a flexible and scalable framework for online video transcoding. In this paper, we introduce a cloud-based multi-core transcoding system to improve the throughput of the system. Simulation evaluation shows how multi-core is worthwhile as compared to single core environment for a particular set of video chunks.

Keywords

Cloud, Delay, Throughput, Transcoding

1. INTRODUCTION

Now the video contents on the Internet is increasing rapidly. Video content providers like YouTube, Netflix, Hulu as well as TV channels are available on the web. There is diversity in the user's demand concerning resolution, bit rates, frame rate or a combination of these. The user device may also differ in processing power, display size, formats, etc. In addition to this, the number of video compression standards is also increasing day by day. The communication channel carrying the video may also have different capacity. It is not possible to store videos with all possible formats, resolutions, frame rates as it requires massive storage and computational resources. This process will also increase the financial cost of the video content providers. One of the solutions is to store the video in some suitable formats and then transcode it on demand. This scheme will help to utilize the storage and computing power in a better way. A video transcoder converts the input video from one format to another. The video transcoder can perform transcoding through bit rate reduction, spatial and temporal resolution reduction [9, 10]. Transcoding process is computation-intensive and needs more resources. It is a challenge to handle flash crowds and maintain small transcoding delay. Cloud computing due to its flexibility and scalability of resource allocation can handle a large number of transcoding jobs. Computing in the Cloud is quick, convenient and straightforward. So, the video content provider can take the benefit of the cloud for video transcoding [17]. The transcoding speed negatively impacts the user's subjective perception of the video. The slow transcoding speed results in long video access time and frequent video freezes. So, a scalable and real-time transcoding environment, like the cloud needed for its operation. Transcoding in the Cloud also deals with the following issues: heterogeneity in the computation, default FCFS scheme, speed should not be achieved at the cost of quality of video [7]. Cloud provide all the necessary resources as a service to create a dynamically scalable transcoding server [3]. A media content provider may over/under-provision resources to meet the QoS requirements of streaming traffic. A long-term investment in infrastructure is also required to handle flash crowds. But, most of the time a huge amount of servers remain idle, which makes the large expenditure wasteful and inefficient. Cloud computing offers an opportunity for media content providers to convert capital expenses to operational expenses. It also eliminates the need of over/under-provisioned servers and private data centers. Use of cloud reduces the expenditure of the media content providers [1]. Every day a large number of videos are uploaded to YouTube, Facebook. A user always expects a high quality video with shortest possible download time. For efficient use of storage and communication media, videos are stored and transmitted in compressed formats. Video transcoding can be done in real-time or in batch processing. For an on-demand video streaming service, if the required video is not available in the desired format, transcoding is performed on the fly. Various ways to perform video transcoding in the cloud are as follows: (i) entire video stream on a dedicated VM. (ii) Split the video stream into smaller segments and then transcode them independently. The first approach requires a significant number of VMs to transcode several simultaneous streams, whereas the second method can transcode a large number of video segments belonging to different video streams in a single VM [8]. In this paper, we are using the second approach i.e. small video chunks of a video in different VMs.

1.1 Motivation

As reported in [17, 14] the global internet video traffic will occupy approximately 69% of total traffic in 2017 up from 57% in 2012. Further mentioned in [5] that Netflix and YouTube make 50.32% of the downstream traffic during the peak period. The video distribution and consumption is proliferating at an exponential rate. The network and device heterogeneity generate the need for different versions of a single video file. It is not possible to store all forms as it is time-consuming, complicated and increase CAPEX & OPEX. Cloud computing offers unlimited scalability to handle peak demand in real-time. As transcoding needs are accelerating not declining and provide options to choose, we can take advantage of the cloud to perform conversion operation.

This paper displays a cloud transcoding approach with multi-core virtual machines (VMs). The rest of the paper organized as follows: related work described in section 2. System model and problem formulation are shown in section 3 and section 4 respectively. Simulation results displayed in section 5. Finally, paper is concluded in section 6.

2. RELATED WORK

Fareed *et.al.* [10] proposed a scalable distributed MPI-based transcoder based on bit rate reduction transcoding technique. Here video segments of equal size with the unequal number of intra-frames and uneven size with an equal number of intra-frames are used to measure the performance of distributed video transcoder built on a multi-core system. Zixia *et.al.* [7] developed a cloud-based video proxy that transcodes the original video into a scalable codec, adaptable to network dynamics for high-quality streaming video delivery. Hallsh-based and lateness-first mapping is used to improve transcoding speed and reduce transcoding jitter. Weiwen *et.al.* [17] investigated scheduling algorithm to route transcoding job in the multimedia cloud, to reduce cloud service engine's energy consumption. A dispatching algorithm REQUEST is proposed to minimize power consumption with queue stability. Yu *et.al.* [16] introduced a queuing network model for viewing behaviors of users in a cloud-based VoD application. Two optimization problems are formulated for VM provisioning and storage rental respectively, for client-server and P2P streaming models. Adnan *et.al.* [3] proposed SBACS a stream based admission control and scheduling algorithm for cloud-based video transcoding process. Amr *et.al.* [1] studied resource allocation problem for cloud-based media streaming application. An algorithm based on the prediction of future demand for streaming capacity proposed to minimize the financial cost of the media content provider. Jiyan *et.al.* [15] proposed sub-frame level (SFL) scheduling method for HD video streaming in a heterogeneous wireless network. An optimization problem is formulated for video streaming allocation to reduce end-to-end delay and find its solution by water filling algorithm. Zhenhua *et.al.* [12] implemented a prediction based cloud transcoder and used intra-cloud ISP aware data upload technique to accelerate data transfer of transcoded video to the user. Xiaofei *et.al.* [13] proposed a framework for mobile video streaming named AMES-cloud that pre-fetch video content in advance based on social network behavior of users and adapt itself according to link dynamics. Kwei *et.al.* [4] presented transcoder dispatching problem (TDP) in a cloud platform, designed an auto-scaling mechanism and attempt to minimize the cost of virtual machines. Guanyu *et.al.* [5] reported a partial transcoding scheme to minimize the overall cost. The content management in media cloud is presented as a stochastic optimization problem and solved by the Lyapunov optimization framework with Lagrangian relaxation. Nawaf *et.al.* [2] proposed a service selection method for video transcoding in a cloud that satisfies the QoS level and user requirements. The selection is made based on best-fit transcoder whose output is as tightly as desired QoS. Chin *et.al.* [11] presented a network and device aware QoS approach for cloud-based mobile streaming that can adjust transmission frequency according to system condition obtained from a prediction based module transcoding on the fly to reduce waste of bandwidth and terminal power. Wu *et.al.* [6] proposed a method to map QoS to QoE and an adjustment model to translate network QoS parameters into user's QoE for a cloud-based multimedia environment. Zuqing *et.al.* [18] layout a network infrastructure based on MaxFlow and Auxiliary graph, which use network status to design a multipath scalable video coding (SVC) video streaming strategy for heterogeneous clients in a cloud environment. Fareed *et.al.* [8] presented prediction based cost efficient dynamic resource allocation algorithm to transcode videos in a cloud environment. Lei *et.al.* [14] discussed a queuing model of a cloud-based transcoding system and derived system delay, targeted video chunk size based on queuing theory. They performed profiling for the cloud cluster to assist resource prediction and scheduling.

3. CLOUD-BASED TRANSCODING MODEL

A cloud-based video transcoder presented in Fig.1 inspired from [8, 3] consists of following modules:

- *Admission Control*: This module is responsible for scheduling of original video streams retrieved from the streaming server. It consists of two submodules: splitter and scheduler. Splitter will break the original video into different video segments. The video segments can be of equal size, a same number of frames, equal number of group of pictures (GOPs), even size with an uneven number of intraframes or different size with equal number of intraframes [9, 10]. Scheduler use various scheduling policy like round-robin, FCFS, weighted fair queue, etc. to send video segments to transcoding module.

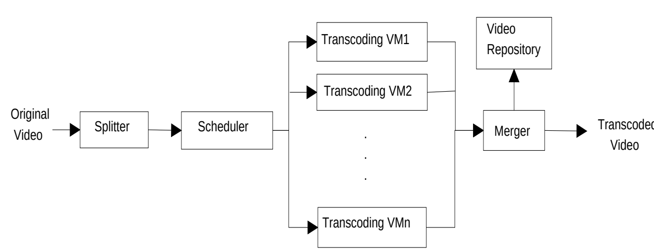

Figure 1: Cloud-based Transcoding Model

- *Transcoding*: Transcoding module will convert the original video stream into transcoded video streams having different bit rates, frame rate, frame size or combination of these or a new compression standard than the original one. This module has merger and video repository as sub-modules. Here we assume that all the transcoding VM has the same capability. Video segments of a single video transcoded at different VM simultaneously. Merger component is used to aggregate the transcoded video streams of a video. If the video is requested for the first time, store the transcoded video in the video repository, so that further demand of the same video can be served directly without performing the computation intensive transcoding process.

4. PROBLEM SPECIFICATION

Let there are v video chunks with transcoding time t_i and deadline d_i, $i = 1, 2, ..., v$. There are N parallel VMs and each VM has c cores. We assume that each core is a VM. So, total number of cores in the system is c.N. Let the number of video chunks transcoded at k^{th} core of j^{th} VM is co_{kj}. The transcoding rate of j^{th} VM is

$$tr_j = \sum_{k=1}^{c} co_{kj}$$

$j = 1, 2, ...N$ and $t_i < d_i$ for all chunks at j^{th} VM. We can say that the throughput of j^{th} VM is tr_j. The overall transcoding rate or Throughput (TH_m) of the system is

$$TH_m = \sum_{j=1}^{N} tr_j \quad (1)$$

For a single core VM throughput(TH_s) is

$$Th_s = \sum_{j=1}^{N} tt_j \quad (2)$$

Where tt_j is the transcoding rate of the j^{th} VM and $t_i < d_i$ for all chunks at j^{th} VM. We know that for any system "rate in=rate out", so we can say that throughput is 100% if $k = TH_m$ for multi-core and $k = TH_s$ for single-core machines.

We would like to compare the throughput of multi-core and single-core machines in the system and find the minimum number of cores to transcode k video chunks.

5. SIMULATION RESULTS

The proposed model presented in above section is validated through MATLAB2014a. The arrival of video chunks is generated by Poisson distribution with slow($\lambda = 0.5$), moderate($\lambda = 0.8$) and high($\lambda = 1$) arrival rate. We consider the speed of the machines in the range [2000-4000] MIPS. For multi-core scenario we consider each machine consists of two cores. The chunks are assigned to the first available core that satisfy the deadline constraint otherwise rejected. First, we present the number of task miss deadline in multi and single core scenario in Figure 2-4. From the figure, we can observe that in a multi-core environment the number of jobs misses the time limit is less, as transcoding operations are performed in parallel on different cores of a single VM. The throughput of the system is dependent on the number of task miss deadline. If less number of tasks miss the deadline, then the throughput will be more. The comparison of Single-core and multi-core machines throughput for slow, moderate and high arrival rate is shown in Figure 5-7. The simulation results show that multi-core environment will increase the throughput as less number of chunks miss the deadline. We can also observe that v (here v=10000) video chunks can be accommodated in 8-10 cores.

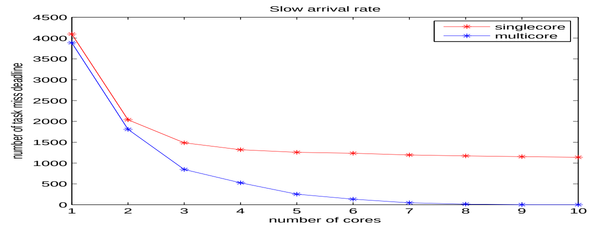

Figure 2: Slow Arrival Rate

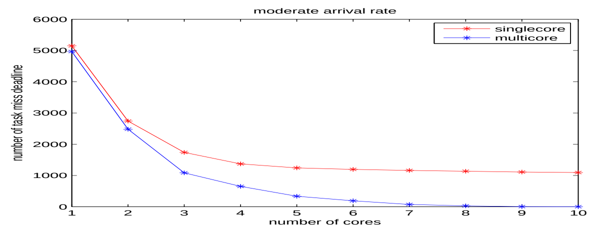

Figure 3: Moderate Arrival Rate

6. CONCLUSIONS

Online transcoding is a time consuming and delay-sensitive task. The system must be able to produce more output in less time. Multi-core technology allows users to execute transcoding tasks in parallel. So, cloud-based video transcoding can take advantage of multi-core technology to perform operations faster. In this paper, we have proposed a cloud-based transcoding system. The rigorous simulation conducted to validate proposed cloud-based transcoder with multi-core VMs with slow, moderate and high arrival rate. The performance of the proposed model is evaluated in terms of throughput and number of task miss deadline. It can be observed that multi-core technology can help to complete more transcoding tasks simultaneously than single core VMs. In future, our aim is to validate the proposed model with the help of real-world workloads.

7. REFERENCES

[1] A. Alasaad, K. Shafiee, H. M. Behairy, and V. C. Leung. Innovative schemes for resource allocation in the cloud for media streaming applications. *IEEE Transactions on Parallel and Distributed Systems*, 26(4):1021–1033, 2015.

[2] N. O. Alsrehin and S. W. Clyde. Qos-aware video transcoding service selection process. *Journal of Media & Mass Communication Vol*, 1(2), 2015.

[3] A. Ashraf, F. Jokhio, T. Deneke, S. Lafond, I. Porres, and J. Lilius. Stream-based admission control and scheduling for video transcoding in cloud computing. In *Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium on*, pages 482–489. IEEE, 2013.

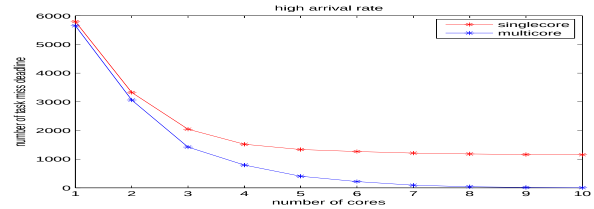

Figure 4: High arrival Rate

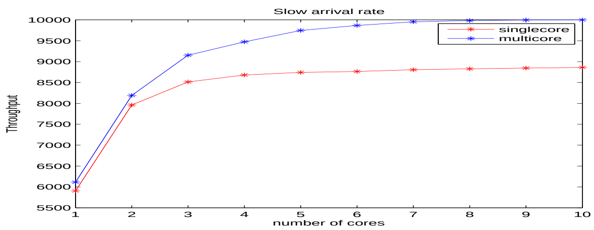

Figure 5: Slow Arrival Rate

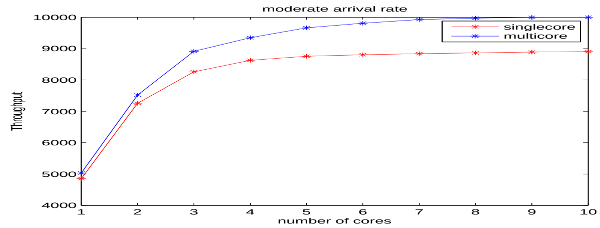

Figure 6: Moderate Arrival Rate

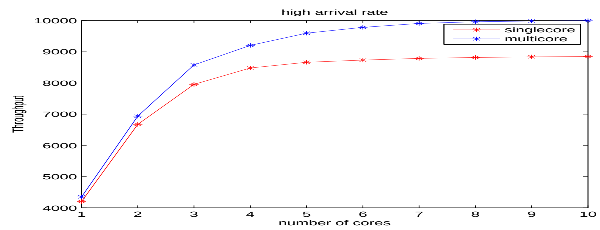

Figure 7: High arrival Rate

[4] K.-B. Chen and H.-Y. Chang. Complexity of cloud-based transcoding platform for scalable and effective video streaming services. *Multimedia Tools and Applications*, pages 1–18, 2016.

[5] G. Gao, W. Zhang, Y. Wen, Z. Wang, and W. Zhu. Towards cost-efficient video transcoding in media cloud: Insights learned from user viewing patterns. *IEEE Transactions on Multimedia*, 17(8):1286–1296, 2015.

[6] W.-H. Hsu and C.-H. Lo. Qos/qoe mapping and adjustment model in the cloud-based multimedia infrastructure. *IEEE Systems Journal*, 8(1):247–255, 2014.

[7] Z. Huang, C. Mei, L. E. Li, and T. Woo. Cloudstream: Delivering high-quality streaming videos through a cloud-based svc proxy. In *INFOCOM, 2011 Proceedings IEEE*, pages 201–205. IEEE, 2011.

[8] F. Jokhio, A. Ashraf, S. Lafond, I. Porres, and J. Lilius. Prediction-based dynamic resource allocation for video transcoding in cloud computing. In *2013 21st Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, pages 254–261. IEEE, 2013.

[9] F. Jokhio, T. Deneke, S. Lafond, and J. Lilius. Analysis of video segmentation for spatial resolution reduction video transcoding. In *Intelligent Signal Processing and Communications Systems (ISPACS), 2011 International Symposium on*, pages 1–6. IEEE, 2011.

[10] F. Jokhio, T. Deneke, S. Lafond, and J. Lilius. Bit rate reduction video transcoding with distributed computing. In *2012 20th Euromicro International Conference on Parallel, Distributed and Network-based Processing*, pages 206–212. IEEE, 2012.

[11] C.-F. Lai, H. Wang, H.-C. Chao, and G. Nan. A network and device aware qos approach for cloud-based mobile streaming. *IEEE transactions on multimedia*, 15(4):747–757, 2013.

[12] Z. Li, Y. Huang, G. Liu, F. Wang, Z.-L. Zhang, and Y. Dai. Cloud transcoder: Bridging the format and resolution gap between internet videos and mobile devices. In *Proceedings of the 22nd international workshop on Network and Operating System Support for Digital Audio and Video*, pages 33–38. ACM, 2012.

[13] X. Wang, M. Chen, T. T. Kwon, L. Yang, and V. C. Leung. Ames-cloud: a framework of adaptive mobile video streaming and efficient social video sharing in the clouds. *IEEE Transactions on Multimedia*, 15(4):811–820, 2013.

[14] L. Wei, J. Cai, C. Foh, and B. He. Qos-aware resource allocation for video transcoding in clouds. 2016.

[15] J. Wu, X. Qiao, Y. Xia, C. Yuen, and J. Chen. A low-latency scheduling approach for high-definition video streaming in a heterogeneous wireless network with multihomed clients. *Multimedia Systems*, 21(4):411–425, 2015.

[16] Y. Wu, C. Wu, B. Li, X. Qiu, and F. C. Lau. Cloudmedia: When cloud on demand meets video on demand. In *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, pages 268–277. IEEE, 2011.

[17] W. Zhang, Y. Wen, J. Cai, and D. O. Wu. Toward transcoding as a service in a multimedia cloud: energy-efficient job-dispatching algorithm. *IEEE Transactions on Vehicular Technology*, 63(5):2002–2012, 2014.

[18] Z. Zhu, S. Li, and X. Chen. Design qos-aware multi-path provisioning strategies for efficient cloud-assisted svc video streaming to heterogeneous clients. *IEEE transactions on multimedia*, 15(4):758–768, 2013.