Interactive Video Streaming via Serverless Cloud Computing

Mohsen Amini Salehi, Chavit Denninnart
High Performance Cloud Computing (HPCC) Laboratory,
School of Computing and Informatics,
University of Louisiana at Lafayette, LA 70503, USA
E-mail: {amini,cxd9974}@louisiana.edu

1 Extended Abstract

Thanks to the high speed Internet, basic video streaming has become an ordinary service nowadays. However, what is offered currently is far from the higher level services that enable stream viewers to *interact* with the video streams. Interactive video streaming enables processing of the video streams upon viewers requests for a particular video. For instance, a viewer may request to watch a video stream with a particular resolution [3]. Another example, is a viewer who requests to view a summary of a video stream.

Current interactive video streaming services are very limited and often require preprocessing of the video streams. However, given the diversity of services offered in an ideal interactive video streaming and the long tail access pattern to the video streams [6], offering interactive video streaming based on lazy (i.e., on-demand) processing of the video streams is required. Such computationally-intensive processing should be achieved in a real-time manner and guarantee the QoS demands of the viewers.

Serverless cloud services have provided an ideal platform for video streaming providers to satisfy the computational demands needed for interactive video streaming [3]. However, the problem in utilizing cloud services for interactive video streaming is: *how to provide a robust interactive video streaming service through guaranteeing QoS desires of the viewers, while spending the minimum cost for the cloud services*? Accordingly, the objective of this tutorial is to present challenges, structures, and methods required to enable interactive video streaming that guarantee QoS in a cost-efficient manner. In particular, we present a framework for interactive video streaming called *Interactive Video Streaming Engine* (IVSE) that deals with the challenges of interactive video streaming services and provides methods to address these challenges in a serverless cloud computing platform.

The reason that video streaming tasks need independent study is that they have unique characteristics. Video streaming tasks have individual deadlines that can be a hard deadline (in live streams [2]) or a soft deadline (in Video On Demand (VOD) [3]). Recent studies (*e.g.,* [4]) show that viewers often watch the beginning of video streams, as such, the quality of delivering the startup of video streams is of paramount importance. Accordingly, video streams have unique QoS demands that are defined as: minimizing missing tasks individual deadlines and minimizing the startup delay of the streams. Depending on the type of video stream content, their processing times (*i.e.,* execution time) vary on different types of processing services (*i.e.,* resource types) offered by cloud providers. Hence, to schedule video streaming tasks, we potentially deal with mapping tasks to heterogeneous cluster of machines (*e.g.,* VMs). In such a heterogeneous computing environment, predicting the execution time of of video streaming tasks is necessary to efficiently map tasks to machines. Execution time prediction is viable thorough historic execution information for VOD streams, however, this is not the case in live streams, where video streaming tasks are generated and processed for the first time. Processing performance of cloud machines may vary over time or even machine failure can occur. In this case, all video streams assigned to those machines cannot proceed with streaming. Hence, execution of video streams are required and failed tasks have to be rescheduled with a high priority to enable smooth video streaming. The access rate to video streams in a repository is not uniform. In fact, access patterns to video streams exhibits a long-tail pattern [6]. As such, caching methods are required to identify *hot* video streams and appropriately cache (store) them using different cloud storage services.

To satisfy the diverse demands of video streams, IVSE framework includes several components. It includes an Admission Control component whose job is to prioritize video streams according to the position of the task in the stream (*e.g.,* beginning portions of video streams have to be prioritized higher) or based on its urgency (*e.g.,* those tasks that are missing in the output window and are required urgently). Resource allocation are required to map video streaming tasks to potentially heterogeneous cloud machines. These methods should take the priority of streaming tasks into account. Resource Allocation component is required to schedule video streaming tasks on potentially heterogeneous set of machines. The Resource Allocation component operates

based on execution time prediction of the tasks. Hence, a component in the IVSE framework is dedicated to predict video streaming tasks execution times that operates based on the type of operation and the content type of the video stream. A Resource Provisioning component is required to monitor the performance of the system and reconfigure that based on the arrival pattern of video streaming tasks and the QoS violation rate of them. For the sake of cost-efficiency, the Resource Provisioning component supplies heterogeneous types of machines where each machine has a different affinity with various arriving video processing tasks. As a result, the component creates and manages a dynamic heterogeneous cluster where the configuration of the machines (*e.g.,* VMs) varies to conform to the arriving workload. Video Merger component in IVSE is in charge of fetching missing video stream tasks and request the missing ones to the Admission Control. Caching policy is in charge of identifying hot video streams and store them using cloud storage services.

In summary, this tutorial describes innovations in interactive video streaming particularly in the following areas:

- Robust, cost-efficient, and self-configurable service provisioning policy: We will explain novel methods to provision a dynamic cluster that conforms its heterogeneity according to the arriving requests.
- Heterogeneity- and QoS-aware scheduling method: It efficiently schedules streaming tasks on available heterogeneous machines with the goal of minimizing both missing tasks deadlines and their startup delays.
- Execution time prediction for video streaming tasks: We will elaborate on the influential factors of the video streaming tasks execution times. In addition, we will explain the way to model affinity exists between heterogeneous machines and services while considering their cost difference.
- A priority-aware admission control method: That prioritizes submission of streaming tasks to minimize the startup delay. The method can also consider the viewer subscription priority, and network speed at the viewers' end.
- Cost-efficient caching methods: We will elaborate on the trade-off between computation versus storage for video streams. We also provide a formal way to measure the hotness of video streams and provide methods that perform caching based on the hotness measure.

1.1 Importance to CCGrid 2019

The way people watch videos has dramatically changed over the past years; from traditional TV systems, to video streaming on desktops, laptops, and smart phones through the Internet. Applications of video streaming has also got extended to areas such as video surveillance, e-learning, and video conferencing, and movie industry. Based on the Global Internet Phenomena Report [5], video streaming currently constitutes approximately 64% of all U.S. Internet traffic. It is estimated that streaming traffic will increase up to 80% of the whole Internet traffic by 2019 [1]. Processing and storage of video streams are time-consuming and costly. Also, video streams are becoming more interactive and video stream providers offer new interactive services on the video streams. As such, it is critical for the CCGrid community of researchers to be aware of the characteristics of the video streaming applications and be able to tailor solutions for these applications.

This tutorial explains the first rigorous interactive video streaming engine, IVSE , that processes video streaming tasks on a dynamic self-configurable heterogeneous cluster of machines. We will present the IVSE architecture and the way its components function individually and interact with each other. Specifically, this tutorial presents state-of-the-art in the following areas: 1) video streaming, 2) serverless cloud computing, and 3) heterogeneous computing.

In addition, we discuss the following items:

- Theoretically sound solutions for heterogeneous resource provisioning and scheduling.
- Policies and methods for admission control, resource provisioning, and scheduling of streaming requests.
- Policies and methods for caching video streams.
- Empirical validation and integration of the IVSE framework for interactive video streaming.

1.2 Benefits to CCGrid 2019 Attendees

Our tutorial provides guidelines for other researchers and practitioners in Cluster and Cloud Computing areas who are interested into incorporating video streaming into cloud computing systems. Particularly, the tutorial will be of interest of those who research in heterogeneous computing, performance modeling, task scheduling, resource provisioning, and caching policies.

We believe that the tutorial can also benefit researchers who do not particularly work on the application of cloud for video streaming. The findings, approaches, and outcomes can be beneficial to researchers who work on other High Performance Computing applications. We present our empirically-validated tools and evaluation benchmarks to real-world practice for use by practitioners and researchers. The tutorial provides a technology-transfer and will break down the barriers between the interactive video streaming and heterogeneous high performance computing based on cloud.

2 Table of Contents

The table of contents of the tutorial will be as follows:

1) Introduction: Interactive Video Streaming
2) Motivations for Lazy Video Stream Processing
3) Background, Challenges, and Existing Solutions for video stream processing
   - Video Streaming Service Workflow
   - Cloud Computing
   - Content Delivery Network (CDN)
   - Storage, Caching, and Security of video streams
4) IVSE : Interactive Video Streaming Engine (IVSE)
   - Video Splitter
   - Admission Control
   - Video Processing Services
   - Service Execution Time Estimator

- Video (GOP) Service Scheduler
- Resource Provisioner
- Video Merger
- Video Caching

5) QoS-Aware Lazy Video Stream Processing Using Homogeneous Cloud Resources
6) Performance Analysis of Video Processing Tasks Using Cloud Services
   - Analysis of Different Video Streaming Task Types
   - Analysis of Different Machine Types
   - Analysis of Different Video Contents
7) Performance Modeling of Using Heterogeneous Cloud VMs for Video Stream Processing
   - Consistent versus Inconsistent Heterogeneity
   - GOP Suitability Matrix Model
8) Cost-Efficient and Robust On-Demand Video Stream Processing Using Heterogeneous Cloud Resources
   - QoS- and Heterogeneity- Aware Video Streaming Task Scheduler
   - Utility-based Video Task Prioritization
   - Self-Con figurable Heterogeneous Resource Provisioner
   - Identifying Suitability of Machine Types for Video Streaming Services
9) Aggregating Video Streaming Tasks for cost-efficiency
10) Live Streaming Using Heterogeneous Cloud Services
11) Task Pruning for Live Video Streaming
12) Video Stream Caching
    - Video Stream Processing versus Storing
    - Quantifying Video Stream Hotness
13) Making use of fog computing for low latency video streaming.
14) Demonstration of Interactive Video Streaming Engine (IVSE)

3 Potential Attendee Profile

The tutorial will attract students and people from academia mostly with the Computer Science and Engineering backgrounds. Specifically, those who have interests in video streaming, video processing, datacenter management, cloud computing, cost-efficiency, heterogeneous computing, and task scheduling.

Industries active in High Performance Computing, video streaming, datacenter resource management will be interested into the tutorial.

4 Biography of the instructor(s)

Dr. Mohsen Amini Salehi received his Ph.D. in Computing and Information Systems from Melbourne University, Australia, in 2012. He is currently an Assistant Professor and director of the High Performance Cloud Computing (HPCC) laboratory, School of Computing and Informatics, at University of Louisiana Lafayette, USA.

His research focus is on different aspects of Distributed and Cloud computing including heterogeneity, load balancing, virtualization, resource allocation, energy-efficiency, and security. Dr. Amini has been working on cloud-based video streaming since 2015. He has been advising three PhD students on this topic. He has also authored seven papers on this topic.

Chavit Denninnart is a Ph.D student at HPCC lab, University of Louisiana at Lafayette. His research is on reusing computation for video streaming via serverless cloud computing. He proposed methods to aggregate streaming tasks while they are in the scheduling queue. His other research direction is on pruning unlikely-to-succeed streaming tasks to reduce the cost and improve the overall quality of video streaming. He has been the author of four papers in the area of cloud-based video streaming.

5 Infrastructure Requirements to Deliver the Tutorial

For this tutorial, we need a projector and Internet access for the presenter and attendees.

6 Previous Tutorials

An early version of this tutorial was presented in Utility and Cloud Computing (UCC '17) conference, Austin, Texas. This version has evolved from the previous version in the following ways:

1) Dynamic video streaming task aggregating;
2) Video Streaming task pruning that includes task dropping and deferral;
3) Making use of fog computing for low latency video streaming;
4) A demonstration of IVSE platform on serverless clouds.

References

[1] Cisco Visual Networking Index. Forecast and methodology, 2014-2019. 2015.

[2] Xiangbo Li, Mohsen Amini Salehi, and Magdy Bayoumi. Vlsc: Video live streaming using cloud services. In *Proceedings of 5th IEEE International Conferences on Big Data and Cloud Computing*, pages 595–600, Oct. 2016.

[3] Xiangbo Li, Mohsen Amini Salehi, Magdy Bayoumi, and Rajkumar Buyya. CVSS: A Cost-Efficient and QoS-Aware Video Streaming Using Cloud Services. In *Proceedings of the 16th IEEE/ACM International Conference on Cluster Cloud and Grid Computing*, CCGrid '16, May 2016.

[4] Lucas CO Miranda, Rodrygo LT Santos, and Alberto HF Laender. Characterizing video access patterns in mainstream media portals. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 1085–1092, Apr. 2013.

[5] Global Internet Phenomena Report. https://www.sandvine.com/trends/global-internet-phenomena/. accessed Oct. 1, 2015.

[6] Navin Sharma, Dilip Kumar Krishnappa, David Irwin, Michael Zink, and Prashant Shenoy. Greencache: Augmenting off-the-grid cellular towers with multimedia caches. In *Proceedings of the 4th ACM Multimedia Systems Conference*, pages 271–280, June 2013.