CVS: A Cost-Efficient and QoS-Aware Cloud Video Streaming

Xiangbo Li, Mohsen Amini-S
Magdy Bayoumi, Rajkumar Buyya

University of Louisiana at Lafayette, Louisiana
The University of Melbourne, Australia
Video Streams

- FLV
- WMV
- Ultra HD
- MPEG
- AVI
- Full HD
- MOV
- mp4
- HD
- Mobile
- 720P
- 240

Streaming Providers

- YouTube
- HBO
- NETFLIX
- hulu
- vevo
- CNN
- YOUKU
- 优酷

Client Devices
• Video streaming constitutes approximately 64% of all the U.S. Internet traffic in 2014 [1].

• Cisco estimates that the streaming traffic will increase to 80% by 2019 [2].


- Storage solutions
- Hardware failover
- Networking infrastructure

- Video contents
- Customer experience

Netflix

Amazon Web Services™
Challenges

How to improve clients' QoS satisfaction?

- Minimize startup delay
- Reduce presentation deadline miss rate

How to minimize the cost for streaming service providers?

### Pricing Examples

1. A 10 minute source file in US West (Oregon) transcoded to an SD output will cost $0.15.

2. A 10 minute source file in US East (N. Virginia) transcoded to an HD output will cost $0.30.

3. A 10 minute source file in EU (Ireland) transcoded to one SD and one HD output will cost $0.51.

4. A 10 minute source file in US West (Oregon) transcoded to an audio-only output will cost $0.045.

5. A 10 minute source file in US West (N. California) transcoded to one audio output, three SD outputs and two HD outputs will cost $1.2422.
Cloud-based Video Streaming (CVS) Architecture

QoS-Aware Scheduling Method

Dynamic cost-efficient provisioning policy
QoS-Aware Scheduling Method

Step 1: Search for the shortest completion time VM.
Step 2: Insert GOP from startup queue in front of the GOP in the batch queue.
Step 3: Check if the GOP in the batch queue will miss deadline or not.
Dynamic Cost-Efficient Provisioning Policy

I. Back to the future

\[ n = \frac{(N_s - 1)}{10 \cdot \beta} \]

II. Look to the past

\[ \gamma = \frac{n \lambda}{N \lambda} \]

---

**Algorithm 2** Cluster Resource Resizing Policy

1: while true do
2:   Calculate current GOP’s deadline miss rate \( \gamma_i \)
3:   Calculate deadline missrate variation: \( \nu = \gamma_i - \gamma_{i-1} \)
4:   Update previous deadline miss rate: \( \gamma_{i-1} = \gamma_i \)
5:   if \( \nu \geq 0 \) \&\& \( \gamma_i \geq \beta \) then
6:     if \( \gamma_i < k \cdot \beta \) then
7:       Allocate one new VM
8:       break
9:     else
10:    Allocate two new VMs
11:    break
12:   end if
13:   else if \( \nu < 0 \) \&\& \( \gamma_i < \alpha \) then
14:     Find the minimum remaining time VM and set its flag to be destroyed
15:     break
16:   else
17:     No VM allocation or deallocation
18:     break
19:   end if
20: end while
Our dynamic system keeps the QoS violation constantly low and Stable in compare with static method.

Our method save the cost when the system is not oversubscribed.
Future Works

Machine Learning-based Scheduling on Heterogeneous Cluster

Live Video Streaming (e.g., video conference, live broadcasting)

Multiple Clouds, utilize nearest cloud to gain better response time
Questions?