# A Joint Online Transcoding and Delivery Approach for Dynamic Adaptive Streaming

Alan Zhuang
qzhuang@ust.hk

April 22, 2015

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Outline

# Outline

# Challenges in the Multi-screen Era

- multiple platforms

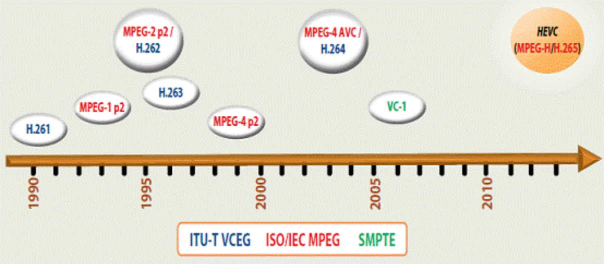

- multiple screen sizes



香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Challenges in the Multi-screen Era

- different supports for containers
  mp4, mkv, avi, flv, wmv, rmvb, webm, mpeg-ts...
- different supports for coding standards
  H.264(AVC), H.265(HEVC), VC-1, AVS, VP8/9, RealVideo...



# Challenges in the Multi-screen Era

- multiple decoding capabilities
  - ⟵2008: MPEG4
  - ⟶2009: H.264(AVC)
  - latest: H.265(HEVC)
    Apple iPhone6, Huawei Honor, Samsung Galaxy S4, Google/LG Nexus 5, XiaoMi 4, ...
- a hardware (chip series) example

| MediaTek | MT6572 | MT6582 | MT6588 | MT6592 |
|---|---|---|---|---|
| Display | 960×540P | 1280×720P | 1920×1280P | 1920×1280P |
| H.264 Decode | 720P@30fps | 1080P@30fps | 1080P@30fps | 1080P@30fps |
| HEVC Decode | N/A | N/A | 720P@30fps | 720P@30fps |

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Challenges in the Multi-screen Era

- competitions between giants

# Luckily,

almost all support:

- coding standard
  H.264/AVC
  (ISO/IEC 14496-10; ITU-T H.264; MPEG-4 Part 10)
- media container
  MP4
  (ISO/IEC 14496-14; MPEG-4 Part 14)

# Transcoding

- from origin to multiple bitrates

$$Source \rightarrow \{\ MP4, [H.264, AAC]\ \} \begin{cases} version_1 & (x_1\ \ kbps) \\ version_2 & (x_2\ \ kbps) \\ \vdots & \vdots \\ version_n & (x_n\ \ kbps) \end{cases}$$

- real-world examples





香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Problems in traditional approaches for adaptive streaming

- only a small set of candidate bitrates to manually choose from cannot effectively adapt to the changing network conditions
- huge computing resource consumption
  - coding to H.264: 1/3 to 2/3 of playback time
  - coding to H.265: 30+ times of playback time
  - one CPU core: only 1-2 concurrent coding tasks
- oblivious of users' *preferences* of different *peering servers*

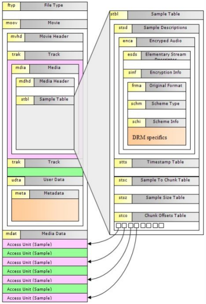
香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

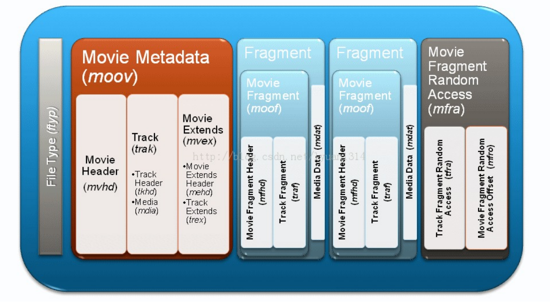
# Outline

# DASH

Problems in HTTP Progressive Downloading.

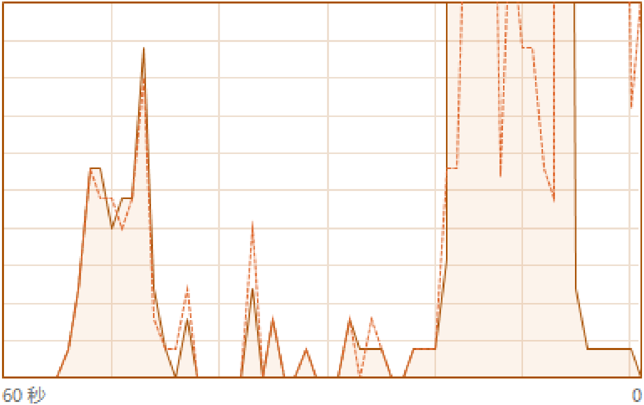
- big media head $\longrightarrow$ long startup/VCR delay




香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

- 2nd time buffering due to varying download speed



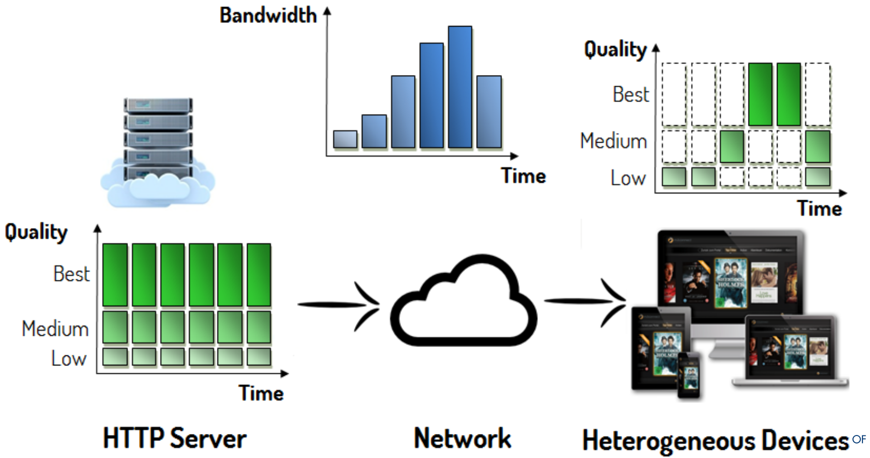
# DASH

DASH: Dynamic Adaptive Streaming over HTTP.
Several industrial & academic DASH standards:

- Apple HLS (HTTP Live Streaming) 2009
- Microsoft HSS (HTTP Smooth Streaming) 2010
- Adobe HDS (HTTP Dynamic Streaming) 2010
- MPEG-DASH (ISO/IEC 23009-1) 2012

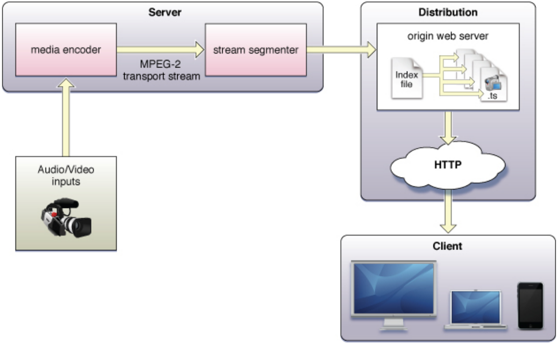
香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Working fashion of DASH: in a nutshell



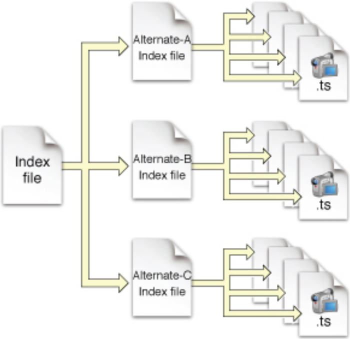
# Apple HLS

- Architecture



# Apple HLS

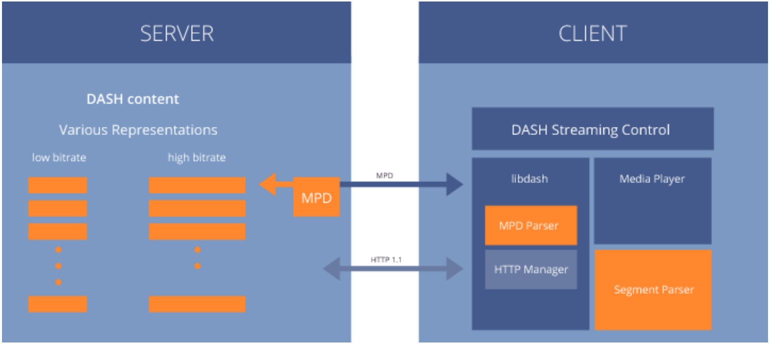
- Segment Indexing




香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

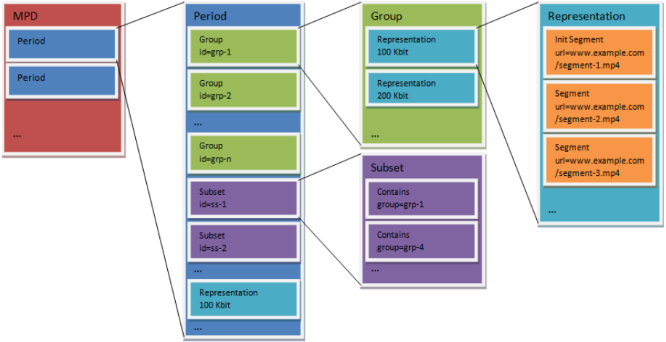
# MPEG-DASH

- Architecture



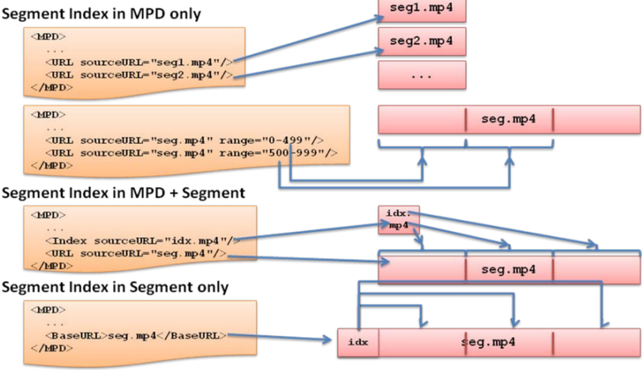
# MPEG-DASH

- Data Model



# MPEG-DASH

- Segment Indexing



# QoE in Streaming

- bitrate, packet loss, delay

  J. Klaue, B. Rathke, and A. Wolisz, "Evalvid c a framework for video transmission and quality evaluation," in *Computer Performance Evaluation. Modelling Techniques and Tools* (P. Kemper and W. Sanders, eds.), vol. 2794 of *Lecture Notes in Computer Science*, pp. 255–272, Springer Berlin Heidelberg, 2003

  Z. Wang, A. C. Bovik, and L. Lu, "Why is image quality assessment so difficult?," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, vol. 4, pp. IV–3313, IEEE, 2002

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# QoE in Streaming

- user activities also affect QoE in DASH
  R. K. P. Mok, X. Luo, E. W. W. Chan, and R. K. C. Chang, "Qdash: A qoe-aware dash system," in *Proceedings of the 3rd Multimedia Systems Conference*, MMSys '12, (New York, NY, USA), pp. 11–22, ACM, 2012
- no-reference metrics
  No-reference Video Quality: capture motion smoothness, motion artifacts, and spatial quality.

# Streaming over CDNs

- traditional studies more focused on the network aspect improving the connectivity between streaming servers and users
  V. Adhikari, Y. Guo, F. Hao, M. Varvello, V. Hilt, M. Steiner, and Z.-L. Zhang, "Unreeling netflix: Understanding and improving multi-cdn movie delivery," in *INFOCOM, 2012 Proceedings IEEE*, pp. 1620–1628, March 2012

# Video Transcoding Schemes

- dedicated transcoders
  V. Adhikari, Y. Guo, F. Hao, M. Varvello, V. Hilt, M. Steiner, and Z.-L. Zhang, "Unreeling netflix: Understanding and improving multi-cdn movie delivery," in *INFOCOM, 2012 Proceedings IEEE*, pp. 1620–1628, March 2012
- SVC based
  Z. Huang, C. Mei, L. Li, and T. Woo, "Cloudstream: Delivering high-quality streaming videos through a cloud-based svc proxy," in *INFOCOM, 2011 Proceedings IEEE*, pp. 201–205, April 2011

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

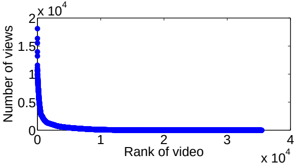
# Video Transcoding Schemes

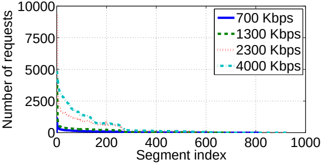
- MapReduce-based
  Alan Zhuang. Tencent TranscX. Cloud Transcoding System Reusing Idle Computational Resources on Storage Servers. CN201210490708.0; PCT/CN2013/085388.
  F. Lao, X. Zhang, and Z. Guo, "Parallelizing video transcoding using map-reduce-based cloud computing," in *Circuits and Systems (ISCAS), 2012 IEEE International Symposium on*, pp. 2905–2908, May 2012
- previous transcoding paradigms
  doing media transcoding and media delivery separately

# Outline

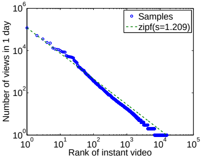
# Video Viewing Patterns

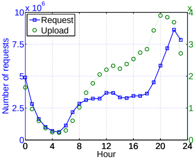
- in BesTV (Professional Content)

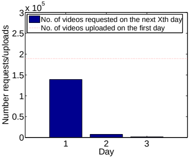


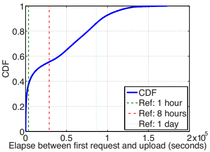




- in WeiShi (UGC)



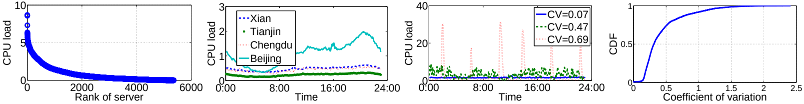




香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

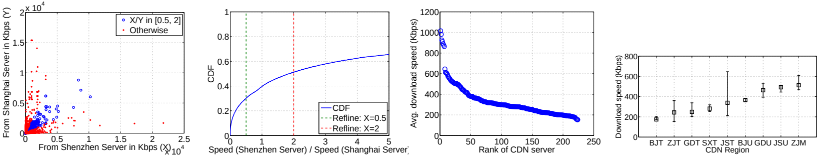
# CDN Patterns

- CPU load patterns



[1]

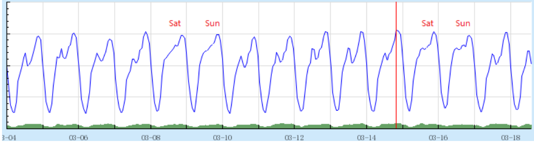
- bandwidth patterns



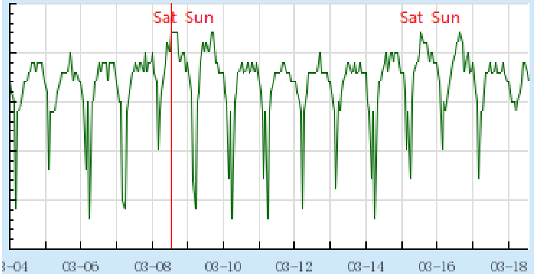

[1] $CV = 1/24\sum_{h=0}^{23}\sqrt{E[(X_h-\bar{X}_h)^2]}/\bar{X}_h$, $X_h$: CPU load in $h$

# CDN Patterns: a long-term view

- total bandwidth of a CDN region of Tencent CDN




- CPU utilization of a server in this region



# Insights

- pre-transcoding all to all versions is unnecessary
  pre-transcoding every segment of all videos to an increasing number of versions is a huge waste of computing resource
- most backend servers in CDNs are predictably(stably) idle thus can be scheduled for transcoding
- users have preferences towards CDN regions
- regions have preferences towards media versions

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Outline

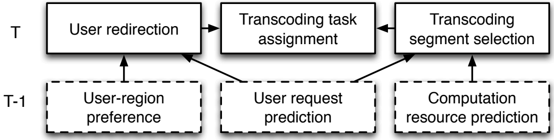


香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Joint Online Transcoding and GeoISP-Distributed Delivery

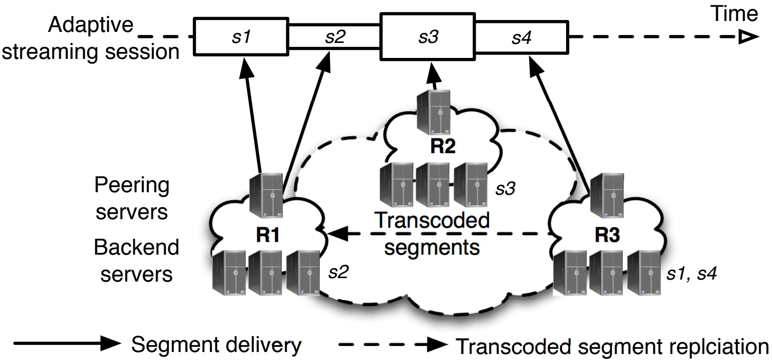
Framework: (in terms of time-slots)

- rank users' preferences towards different CDN regions
  available bandwidth estimation e.g. *abget*, for rank of download speeds
- predict number of requests for a particular segment
  assume mostly in a consecutive way, issuing few VCR operations
- predict idle computing resource
  just use the status of the last slot, or LR, ARIMA, ANN, ...



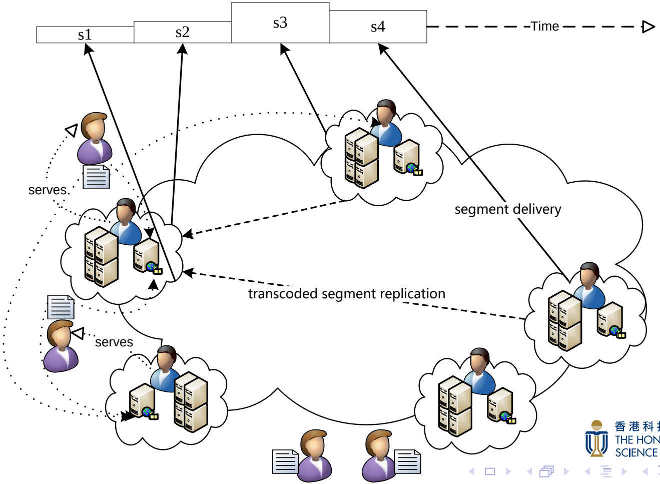
# Joint Online Transcoding and GeoISP-Distributed Delivery

Architecture:



# Joint Online Transcoding and GeoISP-Distributed Delivery

## Architecture: Girls Chasing Boys?



# Joint Online Transcoding and GeoISP-Distributed Delivery

Mechanism:

- user request redirection
  to her ideal CDN region, and then do a load-balancing e.g. round-robin
- transcoding segment selection
  using backend servers with idle computing resource, in terms of closed GoPs that can be transcoded independently;
  if cannot be transcoded timely, send a closest lower alternative bitrate version
- transcoding task assignment
  transcoding is performed in selected regions;
  transcoded segments will be cached by the backend servers and replicated to other regions;
  replication cost should be minimized

# QoE-Driven Redirection: Notations

| Symbol | Definition |
|---|---|
| $\mathbf{U}^{(T)}$ | Set of users requesting segments in time slot $T$ |
| $\mathbf{R}$ | Set of CDN regions |
| $D_{u,r}^{(T)}$ | Binary variable indicating whether user $u$ will download from region $r$ in time slot $T$ |
| $H_{u,r}$ | Preference level for user $u$ to receive video stream from region $r$ |
| $W_r$ | Bandwidth capacity of region $r$ |
| $B_v$ | Bitrate of a particular version $v$ |
| $L_{u,r}$ | Highest version that $u$ can receive when she downloads from region $r$ |

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# QoE-Driven Redirection: Problem Formulation

$$\max_{D^{(T)}} \sum_{u \in \mathbf{U}^{(T)}, r \in \mathbf{R}} H_{u,r} D_{u,r}^{(T)}, \qquad (1)$$

subject to:

$$\sum_{r \in \mathbf{R}} D_{u,r}^{(T)} \leq 1, \forall u \in \mathbf{U}^{(T)},$$

$$\sum_{u \in \mathbf{U}^{(T)}} D_{u,r}^{(T)} B_{L_{u,r}} \leq W_r, \forall r \in \mathbf{R},$$

$$D_{u,r}^{(T)} \in \{0, 1\}, \forall u \in \mathbf{U}^{(T)}, r \in \mathbf{R},$$

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# QoE-Driven Redirection: NP-hardness Proof

## Theorem

*Redirecting users to CDN regions such that their preferences can be maximally satisfied, as formulated in (1), is NP-hard.*

# QoE-Driven Redirection: NP-hardness Proof

## Theorem

*Redirecting users to CDN regions such that their preferences can be maximally satisfied, as formulated in (1), is NP-hard.*

## Proof.

Reduce a conventional 0/1 knapsack to this problem:

$$\max \sum_{i=1}^{n} v_i x_i,$$

subject to

$$\sum_{i=1}^{n} \alpha_i x_i \leq \beta, x_i \in \{0, 1\},$$

# QoE-Driven Redirection: NP-hardness Proof

## Proof.

1. Let $\mathbf{U}^{(T)} = \{1, 2, \ldots, n\}$, $\mathbf{R} = \{1\}$;
2. Let $H_{i,1} = v_i, i = 1, 2, \ldots, n$;
3. Let $B_{L_{i,1}} = \alpha_i, i = 1, 2, \ldots, n$;
4. Let $W_1 = \beta$.

The reduction operations take linear time, and the final results for the 0/1 knapsack problem are $x_i = D^{(T)}_{i,1}, i = 1, 2, \ldots, n$. Therefore, our problem is NP-hard. □

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# QoE-Driven Redirection: Our Heuristic Algorithm

1. Bootstrap
   When a user requests to watch a video, assign her a list of candidate peering servers from regions with the lowest load.
2. Users rank servers
   in descending order of the estimated download speeds, and send connection requests to these servers.
3. Servers rank users
   only accept a portion of users according to its available bandwidth $W_r$. The request from user $u$ is prioritized to be accepted if she has a larger $H_{u,r}/B_{L_{u,r}}$ with the CDN region $r$.
4. Users finally decide
   A user selects the best peering server from the ones accepting her request according to the ranked list.

香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# QoE-Driven Redirection: Essentially...

- Like a Stable Matching
  but simplified.
- More Boston-like
  rather than a DA(Deferred Acceptance) fashion.
- Easy to be implemented
  and executed.
- Works efficiently
  in real-world in distributed manner!

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Prioritizing Sgement Transcoding Tasks: Notations

| Symbol | Definition |
|---|---|
| $\mathbf{K}^{(T)}$ | The set of segments being requested in time slot $T$ |
| $P_{(s,v)}^{(T)}$ | Indicator: if segment $(s, v)$ will be transcoded |
| $e_{(s,v)}^{(T)}$ | Importance level of a particular segment $(s, v)$ in $T$ |
| $Q_{(s,v)}^{(T)}$ | Number of requests of segment $(s, v)$ from all regions in $T$ |
| $Y_{(s,v)}^{(T)}$ | Quality gain if segment $(s, v)$ is transcoded in $T$ |
| $B_v$ | Bitrate of a particular version $v$ |
| $L_{u,r}$ | Highest version $u$ can receive when downloads from region $r$ |
| $\mathbf{G}_s^{(T)}$ | The set of transcoded versions of segment $s$ |
| $C_{(s,v)}$ | Computing resource required to perform the transcoding task to generate a segment $s$ of version $v$ |
| $I_r^{(T)}$ | Available computing resource that can be allocated for video transcoding from region $r$ in $T$ |

# Prioritizing Sgement Transcoding Tasks: Formulation

$$e_{(s,v)}^{(T)} = Q_{(s,v)}^{(T)} Y_{(s,v)}^{(T)},$$

$$Y_{s,v}^{(T)} = \begin{cases} \min_w (B_v - B_w)/B_v, & \exists w \in \mathbf{G}_s^{(T)}, w < v \\ 1, & \text{otherwise} \end{cases}.$$

Problem:

$$\max_{P^{(T)}} \sum_{(s,v)\in\mathbf{K}^{(T)}} P_{(s,v)}^{(T)} e_{(s,v)}^{(T)}, \quad (2)$$

subject to:

$$\sum_{(s,v)\in\mathbf{K}^{(T)}} P_{(s,v)}^{(T)} C_{(s,v)} \le \sum_r I_r^{(T)},$$

$$P_{(s,v)}^{(T)} \in \{0,1\}, \forall (s,v) \in \mathbf{K}^{(T)}.$$

# Prioritizing Sgement Transcoding Tasks: Solution

It is a 0-1 knapsack problem.

1. Predicting
   Collect the information for prediction in a centralized manner, *e.g.*, users (*resp.* backend servers) report which segments they are downloading (*resp.* the CPU load information) to a centralized server.
2. Ranking
   Based on the prediction, rank the requested segments in descending order of $e^{(T)}_{(s,v)}/C_{(s,v)}$.
3. Selecting
   Iteratively select segments from the ranked list to transcode, and update computation resource consumption, until the available idle computing resource is used up.

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Scheduling Transcoding Tasks across Regions: Notations

| Symbol | Definition |
|---|---|
| $\mathbf{R}$ | Set of CDN regions |
| $\mathbf{E}^{(T)}$ | Set of segments to be transcoded in time slot $T$ |
| $C_{(s,v)}$ | Computation resource required to perform the transcoding task to generate a segment $s$ of version $v$ |
| $I_r^{(T)}$ | Available computation resource that can be allocated for video transcoding from region $r$ in $T$ |
| $F_{(s,v),r}$ | Overall replication cost when segment $(s,v)$ is transcoded in region $r$ |
| $A_{(s,v),r}^{(T)}$ | Indicator: whether $(s,v)$ is transcoded in region $r$ in $T$ |
| $J_{(s,v),r'}^{(T)}$ | Number of requests of $(s,v)$ to be served by a region $r'$ |
| $Z_{r,r'}(s,v)$ | Replication cost when $(s,v)$ is replicated from $r$ to $r'$ |

# Scheduling Transcoding Tasks across Regions: Formulation

$$F_{(s,v),r} = \sum_{r' \neq r, J^{(T)}_{(s,v),r'} > \beta} Z_{r,r'}(s,v),$$

Problem:

$$\min_{A^{(T)}} \sum_{(s,v)\in \mathbf{E}^{(T)}} \sum_{r \in \mathbf{R}} A^{(T)}_{(s,v),r} F_{(s,v),r}, \tag{3}$$

subject to:

$$A^{(T)}_{(s,v),r} \in \{0,1\}, \forall (s,v) \in \mathbf{E}^{(T)}, r \in \mathbf{R}$$

$$\sum_{r \in \mathbf{R}} A^{(T)}_{(s,v),r} = 1, \forall (s,v) \in \mathbf{E}^{(T)}$$

$$\sum_{(s,v)\in \mathbf{E}^{(T)}} A^{(T)}_{(s,v),r} C_{(s,v)} \leq I^{(T)}_r, \forall r \in \mathbf{R}.$$

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Scheduling Transcoding Tasks across Regions: Solution

**Algorithm** Transcoding task schedule.

Let $M_r = I_r^{(T)}, r \in \mathbf{R}$

Let $A_{(s,v),r}^{(T)} = 0, \forall (s,v) \in \mathbf{E}^{(T)}, r \in \mathbf{R}$

Rank CDN region and segment pairs $(r - (s,v))$ in ascending order of $F_{(s,v),r}$

**for** $\forall r - (s,v)$ in the ranked list **do**

  **if** $C_{(s,v)} \leq M_r$ **then**

    Let $M_r = M_r - C_{(s,v)}$

    Let $A_{(s,v),r}^{(T)} = 1$

    Remove pairs with $(s,v)$ from the ranked list

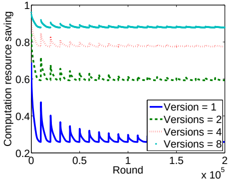
  **end if**

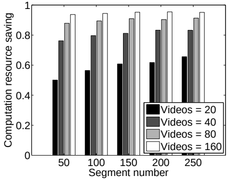
**end for**

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

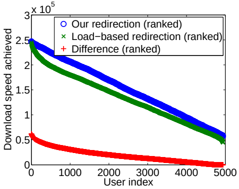
# Effects

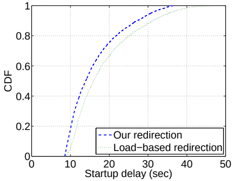
- Computing resource saved

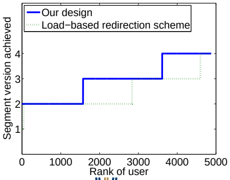



- Improvement of QoE

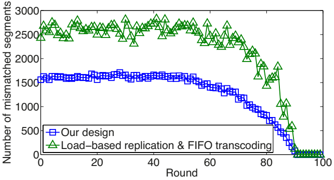


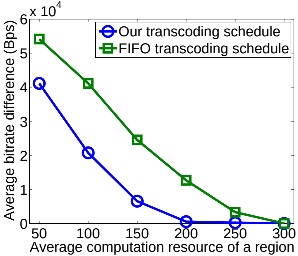



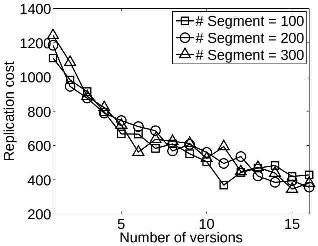
# Effects

- Mismatch



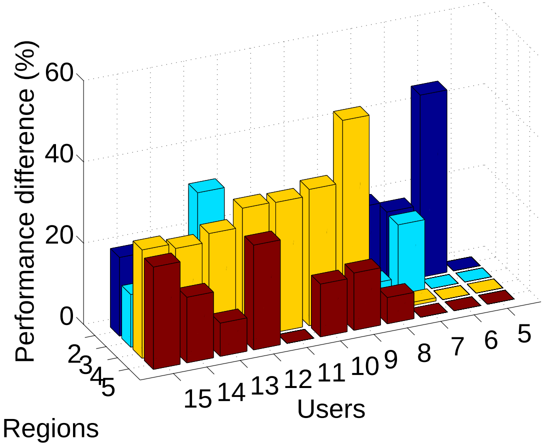

- Replication Cost



# Effects

- Gap to the Optimal Solution



# Effects: Highlights

- 44.8% users enjoy higher bitrate versions than the load-balanced redirection scheme
- 4.5x users enjoy the highest possible bitrate
- Mismatch rate reduced by over 42.2% who receive a segment of a mismatched version
- $\sim$ 80% computing resource saved when the number of versions is 4. The higher the number is, the more resource our approach saves.

香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Outline



香港科技大學 THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Conclusion & Q/A

Transcoding and Delivery can, and should be considered jointly.

Q/A?