Practical, Real-time Centralized Control for CDN-based Live Video Delivery

Matt Mukerjee, David Naylor, Junchen Jiang, Dongsu Han, Srini Seshan, Hui Zhang
Live Video is Becoming Wildly Popular

- Commercial sports streams
- User-generated streams
Live Video is Becoming Wildly Popular

- Commercial sports streams
  - **Single** World Cup stream = 40% global Internet traffic
- User-generated streams (e.g., Twitch)
  - Users watch **150b min of live video per month**
  - Amazon buys Twitch for ~$1 Billion
Our Contributions

- We design a video delivery network (VDN) to efficiently manage quality and cost, with high responsiveness.
Outline

Problems with Live Video Today

Centralized Control

Hybrid Control

Putting it all Together

Distributed Control
CDN Live Video Delivery Background

Video Sources

Reflector Clusters

Edge Clusters

Data: Control

HTTP RESPONSE

HTTP GET

Legend

Requests:
- Dotted blue: Video 1
- Red: Video 2

Responses:
- Blue: Video 1
- Red: Video 2

Video Requests
CDN Live Video Delivery Background

Video Sources

Reflector Clusters

Edge Clusters

Clients

Data ▶︎ Control

Link Cost

Link Capacity

Clients: H, I, J

Video Sources: A, B

Reflector Clusters: C, D

Edge Clusters: E, F, G

DNS

Link Cost:
- 0: 500
- 1: 300, 750
- 2: 700

Link Capacity:
- 120: 3K
- 100: 1K
- 100: 2K
- 20: 25
- 15: 10
- 15: 1

Diagram shows the flow of data and control from video sources to clients through edge clusters and reflector clusters.
CDN Live Video Delivery Background

Objective:
Maximize service quality
&
Minimize delivery cost
Problems with CDNs Today

Service Quality

Simulation using Conviva traces, modeling user-generated content

Delivery Cost

(per request)

Simulation using Conviva traces, modeling large sports events
Problems with CDNs Today

Service Quality

Delivery Cost

<table>
<thead>
<tr>
<th>CDN</th>
<th>OPTIMAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0x</td>
<td>1.0x</td>
</tr>
</tbody>
</table>

Not Fine-Grained

- Videos aggregated into large groups
- Can’t push updates
- DNS entries get cached

Slow DNS Updates

- Can’t push updates
- DNS entries get cached
Goals

Service Quality

- Delivery Cost
  - CDN: 2.0x
  - Optimal: 1.0x

Fine-Grained Control

- Per-video Control
- Real-time Response
  - Sub-second response to failures and joins

Room for improvement, but Internet latency / loss
Goals

Service Quality
- Fine-Grained Control
- Real-time Response
- Per-video Control
- Sub-second response to failures and joins

Centralization!
[Liu, Xi et. al. A Case for a Coordinated Video Control Plane. SIGCOMM 2012]

Room for improvement, but Internet latency / loss
Outline

Problems with Live Video Today

Centralized Control

Distributed Control

Hybrid Control

Putting it all Together
Motivating Centralized Optimization
Motivating Centralized Optimization

- Video Sources
- Reflector Clusters
- Edge Clusters
- Clients

Link Capacity

DNS

Congestion!
Motivating Centralized Optimization

Video Sources

Reflector Clusters

Edge Clusters

Clients

Link Capacity

DNS
Motivating Centralized Optimization

Needs global view to coordinate videos and network resources
Motivating Centralized Optimization

- Video Sources
- Reflector Clusters
- Edge Clusters
- Clients

Link Capacity

DNS

Clients

Motivating Centralized Optimization

- Video Sources
- Reflector Clusters
- Edge Clusters
- Clients

Link Capacity

DNS
Motivating Centralized Optimization

- **Video Sources**
- **Reflector Clusters**
- **Edge Clusters**
- **Clients**

Central Controller

Link Capacity

**Data & Control**

- 2K
- 300
- 200
- 300
- 750
- 700
- 500
- 300
- 200
- 300
- 200
- 300
- 300
- 300
- 500
- 300
- 750
- 700
- 700
- 700
- 300
- 500
- 300
- 200
Solving Centralized Optimization

MAXIMIZE  SERVICE QUALITY
MINIMIZE  DELIVERY COST
SUBJECT TO  DON’T EXCEED LINK CAPACITY
             SENDER MUST HAVE RECEIVED VIDEO
Solving Centralized Optimization

\[
\begin{align*}
\text{max} & \quad w_s \cdot \sum_{l \in L_{AS}, o \in O} \text{Priority}_o \cdot \text{Request}_{l, o} \cdot \text{Serves}_{l, o} \\
& \quad - w_c \cdot \sum_{l \in L, o \in O} \text{Cost}(l) \cdot \text{Bitrate}(o) \cdot \text{Serves}_{l, o} \\
\text{subject to:} & \\
\forall l \in L, o \in O & : \text{Serves}_{l, o} \in \{0, 1\} \\
\forall l \in L & : \sum_o \text{Bitrate}(o) \cdot \text{Serves}_{l, o} \leq \text{Capacity}(l) \\
\forall l \in L, o \in O & : \sum_{l' \in \text{InLinks}(l)} \text{Serves}_{l', o} \geq \text{Serves}_{l, o}
\end{align*}
\]
Flexibility of Centralized Optimization

- **Video Sources**
  - A
  - B

- **Reflector Clusters**
  - C

- **Edge Clusters**
  - D
  - E
  - F

- **Clients**
  - H
  - I
  - J

- **Central Controller**

- **Link Cost**
- **Link Capacity**
Flexibility of Centralized Optimization

Video Sources

Reflector Clusters

Edge Clusters

Clients

Link Cost

Link Capacity

Video Priority
Centralized Optimization

Service Quality

Delivery Cost (per request)

Simulation using Conviva traces, modeling large sports events

Simulation using Conviva traces, modeling user-generated content
Centralized Optimization

Service Quality

Delivery Cost

(per request)

Simulation using Conviva traces, modeling user-generated content

Simulation using Conviva traces, modeling large sports events

CDN 2.0x

VDN 1.0x
Unfortunately… No Free Lunch

Experiments on EC2 nodes with a centralized controller at CMU across the Internet
Problems with Centralization

Legend

Data Requests:
Video 1

Control Traffic:
Video 1

The Internet

HIGH LATENCY

Central Controller

Video Sources

Reflector Clusters

Edge Clusters

Clients
Outline

Problems with Live Video Today

Centralized Control

Slow join times

Distributed Control

Hybrid Control

Putting it all Together
Alternate Approach: Distributed

Legend
- Data Requests:
  - Video 1
- Responses:
  - Video 1
- Link Capacity

Video Sources

Reflector Clusters

Edge Clusters

Clients

Central Controller

Data: Control

Clients

800

Video 1

Data Requests:

Link Capacity
Alternate Approach: Distributed

Build “distance-to-video” tables at each cluster, for each video

Legend

Data Requests:
- Video 1

Responses:
- Video 1

Video Sources

Reflector Clusters

Edge Clusters

Clients

Central Controller

Legend

Data Requests:
- Video 1

Responses:
- Video 1

Build “distance-to-video” tables at each cluster, for each video
Alternate Approach: Distributed

Legend

Data Requests:
- Video 1

Responses:
- Video 1

Link Capacity

Video Sources

Reflector Clusters

Edge Clusters

Central Controller

Clients

DISTANCE AT CLUSTER F

VIDEO 1:
Alternate Approach: Distributed

![Diagram of a distributed video streaming system with video sources, reflector clusters, edge clusters, clients, and a central controller. The diagram includes link capacity, data requests, and responses.]
Alternate Approach: Distributed

Legend

Data Requests:
- Video 1: Video 1

Responses:
- Video 1

Link Capacity

Video Sources

Reflector Clusters

Edge Clusters

Clients

Central Controller

DISTANCE AT CLUSTER F

VIDEO 1: VIA C: 2; (B, 1K)

VIA D: 1; (D, 800)
Alternate Approach: Distributed

Legend
- Data Requests:
  - Video 1
  - Responses:
    - Video 1
- Link Capacity

Video Sources
Reflector Clusters
Edge Clusters
Clients

Central Controller

DISTANCE AT CLUSTER F

VIDEO 1:
- VIA C: 2; (B, 1K)
- VIA D: 1; (D, 800)

PICK SHORTEST PATH WITH ENOUGH CAPACITY
Alternate Approach: Distributed

Legend

Data Requests:
- Video 1

Responses:
- Video 1

Link Capacity

Video Sources

Reflector Clusters

Edge Clusters

Clients

DISTANCE AT CLUSTER F

VIDEO 1: VIA C: 2; (B, 1K)

VIA D: 1; (D, 800)

PICK SHORTEST PATH WITH ENOUGH CAPACITY

PICK SHORTEST PATH WITH ENOUGH CAPACITY
Alternate Approach: Distributed

Distributed decisions fast (ms) but sub-optimal
Alternate Approach: Distributed

Combine approaches?  
“Hybrid Control”

Legend

Data Requests:
- Video 1
- Link Capacity

Responses:
- Video 1

DISTANCE AT CLUSTER F
VIDEO 1:
- VIA C: 2; (B, 1K)
- VIA D: 1; (D, 800)

PICK SHORTEST PATH WITH ENOUGH CAPACITY
Outline

Problems with Live Video Today

Centralized Control

Slow join times

Low bitrate

Distributed Control

Hybrid Control

Putting it all Together
Hybrid Control

Central Optimization

Quality and cost management (minutes)

Distributed Control

Responsiveness to joins and failures (milliseconds)

Hybrid Control

Responsiveness to joins and failures (milliseconds)

Quality and cost management (minutes)
Challenges of Hybrid Control

- Forwarding loops
  - Always forward requests upwards

- State transitions
  - Versioning and “shadow FIBS”

- Avoid bad control loop interactions
Combining Approaches: **Hybrid**

![Diagram showing hybrid approach in networking](image)

- **Video Sources**
- **Reflector Clusters**
- **Edge Clusters**
- **Clients**

**Legend**
- **Data Requests:** Video 1
- **Responses:** Video 1

**High Latency**
- Central Controller

**The Internet**
Combining Approaches: **Hybrid**

- **Video Sources**
- **Reflector Clusters**
- **Edge Clusters**
- **Clients**

**Legend**
- Data Requests:
  - Video 1: 2K, 3K
- Responses:
  - Video 1: 800

**Central Controller**

**The Internet**
Combining Approaches: Hybrid

Legend:
- Video Sources
- Reflector Clusters
- Edge Clusters
- Clients

Data Requests:
- 2K
- 3K

Responses:
- 800

Control Traffic:
- Video 1

High Latency:
- The Internet
- Central Controller

Clients: H, I, J

Video Sources: A, B, C, D

Legend:
- Video 1
- Data Requests: 2K, 3K
- Responses: 800
- Control Traffic: Video 1

Central Controller

The Internet
Challenges of Hybrid Control

- Forwarding loops
  - Always forward requests upwards

- State transitions
  - Versioning and “shadow FIBS”

- Avoid bad control loop interactions
Challenges of Hybrid Control

• Avoid bad control loop interactions

1. Centralized decision has priority
2. Distributed uses residual after centralized
3. Distributed has no impact on current/future centralized decisions
4. Distributed’s changes don’t propagate
Hybrid Control and Responsiveness

Experiments on EC2 nodes with a centralized controller at CMU across the Internet
Hybrid Control and Responsiveness

Experiments on EC2 nodes with a centralized controller at CMU across the Internet

Slow join times!

Not stable

# of videos

Join Time (Seconds)

Light Load      Med. Load      Hvy. Load

Hybrid Control

- Fully Centralized
- Fully Distributed

0  50  100  150  200

0  5  10  15  20  25

Experiments on EC2 nodes with a centralized controller at CMU across the Internet

Slow join times!

Not stable
Experiments on EC2 nodes with a centralized controller at CMU across the Internet

Slow join times!

Not stable

Great join times and more stable
Outline

Centralized Control

Distributed Control

Problems with Live Video Today

Slow join times

Low bitrate

Hybrid Control

“Better than both”

Putting it all Together
Putting it all Together

- Video Sources
- Reflector Clusters
- Edge Clusters
- Clients

"Local Agent" per cluster

Logically centralized controller

Data : Control
Putting it all Together

TOPOLOGY AND VIDEO INFO

DISTRIBUTION TREES

HYBRID CONTROL

CENTRALIZED

DISTRIBUTED

DATA PLANE

LOCAL AGENT

HTTP Server

CENTRAL CONTROLLER

DISCOVERY

CONTROL

DISCOVERY

CONTROL
Key Results

- Trace-driven eval - **centralized optimization**
  - High quality & low delivery cost? 1.7x / 2x
  - Scalable / fine grain? 10K videos; 2K clusters

- End-to-end eval - **hybrid control**
  - Responsive? 200ms

- More results in paper
  - Operator Control? Failures? Partitions?
Conclusion

• VDN presents a new approach for CDN-based live video delivery
Practical, Real-time Centralized Control for CDN-based Live Video Delivery

Matt Mukerjee, David Naylor, Junchen Jiang, Dongsu Han, Srini Seshan, Hui Zhang
Backup slides...
Problems with Traffic Engineering

Video Sources

Reflector Clusters

Edge Clusters

Clients

Link Capacity

Even Split (1K)
Problems with Traffic Engineering

- **Video Sources**
  - A
  - B
  - E
  - Video Sources

- **Reflector Clusters**
  - C
  - Reflector Clusters

- **Edge Clusters**
  - E
  - F
  - G
  - Edge Clusters

- **Clients**
  - H
  - I
  - J
  - Clients

- **Uneven Split**
  - (1.5K / 500)

- **Link Capacity**
  - 200
  - 1.5K
  - 200
  - 1.5K
  - 300
  - 2K
  - 750
  - 700
  - 300
  - 2K
  - 2K
  - 300
  - 200
  - Uneven Split
  - Link Capacity
Distributed: Example of Sub-optimal
Distributed: Example of Sub-optimal

Video Sources

Reflector Clusters

Edge Clusters

Clients

Video 1

Legend

Data Requests:
- Video 1

Responses:
- Video 1

Coordination difficult without centralization

Link Capacity

Data: Control

800

300

750

700

800

3K

2K

2K

800

2K

2K

800

Legend

Data Requests:
- Video 1

Responses:
- Video 1

Coordination difficult without centralization

Link Capacity

Data: Control

800
Trace-Driven Eval

- **3 Traces**
  - Avg Day: raw trace of music video provider
  - Large Event: synthesized basketball game
  - Heavy Tail: synthesized twitch/ustream like workload

- **4 Systems**
  - Everything Everywhere: all vids to all servers
  - Overlay Multicast: globally optimal; no coordination
  - CDN: greedy distribution scheme w/ DNS
  - VDN: our system
# Trace-Driven Eval

<table>
<thead>
<tr>
<th></th>
<th>EE</th>
<th>CDN</th>
<th>VDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. Bitrate (kbps)</td>
<td>588</td>
<td>2,725</td>
<td>2,716</td>
</tr>
<tr>
<td>Cost / Sat. Req. (norm.)</td>
<td>103</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>Clients at Req. BR (%)</td>
<td>18.73%</td>
<td>100%</td>
<td>99.83%</td>
</tr>
</tbody>
</table>

**Table 1: Results for Average Day trace.**

<table>
<thead>
<tr>
<th></th>
<th>EE</th>
<th>CDN</th>
<th>VDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. Bitrate (kbps)</td>
<td>0.08</td>
<td>2,725</td>
<td>2,725</td>
</tr>
<tr>
<td>Cost / Sat. Req. (norm.)</td>
<td>178K</td>
<td>2.2</td>
<td>1</td>
</tr>
<tr>
<td>Clients at Req. BR (%)</td>
<td>0%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

**Table 2: Results for Large Event trace.**

<table>
<thead>
<tr>
<th></th>
<th>EE</th>
<th>CDN</th>
<th>VDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. Bitrate (kbps)</td>
<td>685</td>
<td>1748</td>
<td>3366</td>
</tr>
<tr>
<td>Cost / Sat. Req. (norm.)</td>
<td>8.9</td>
<td>1.21</td>
<td>1</td>
</tr>
<tr>
<td>Clients at Req. BR (%)</td>
<td>22%</td>
<td>49%</td>
<td>77%</td>
</tr>
</tbody>
</table>

**Table 3: Results for Heavy-Tail trace.**
Existing Solutions

• Traffic Engineering (SWAN, B4, …)
  • Works on aggregates at coarse timescales
• Overlay Multicast (Overcast, Bullet, …)
  • Not designed for coordinating across streams
• Modern CDNs
  • Previous work shows a centralized system could greatly improve user experience but would be difficult to design over Internet