Delivering Stable High-quality Video with DASH Assisting Network Elements

Jan Willem Kleinrouwele
Distributed and Interactive Systems
Centrum Wiskunde & Informatica
Dynamic Adaptive Streaming over HTTP
Dynamic Adaptive Streaming over HTTP

delivery of multimedia content over HTTP
Dynamic Adaptive Streaming over HTTP

delivery of multimedia content over HTTP

video on-demand & live streaming
Dynamic Adaptive Streaming over HTTP

delivery of multimedia content over HTTP
video on-demand & live streaming
adapt video quality to network conditions
Dynamic Adaptive Streaming over HTTP

delivery of multimedia content over HTTP
video on-demand & live streaming
adapt video quality to network conditions
efficient use of existing distribution infrastructures
Multiple (DASH enabled) devices share a network connection
Multiple (DASH enabled) devices share a network connection
Performance problems due to a competition for bandwidth
Multiple DASH enabled devices share a network connection
Multiple DASH enabled devices share a network connection
Multiple DASH enabled devices sharing a network connection with other services
Multiple DASH enabled devices sharing a network connection with other services
DASH Assisting Network Elements
DASH Assisting Network Elements

smart network elements that are “DASH aware”
DASH Assisting Network Elements

smart network elements that are “DASH aware”

traffic control
DASH Assisting Network Elements

smart network elements that are “DASH aware”
traffic control
bitrate assistance
in-band implementation as proxy server
add signaling via extra HTTP headers/
modify the manifest/segment requests

X-Dash-Target: 2000
out-of-band
websocket connection to DANE
signaling via this extra channel
Multiple DASH enabled devices sharing a network connection with other services
Multiple DASH enabled devices sharing a network connection with other services
Multiple DASH enabled devices sharing a network connection with other services
DASH performance with DANEs is predictable
DASH performance with DANEs is predictable programmable
Markov model with states: 

\((n_1, n_2, \ldots, n_K)\)

such that:

\[
\sum_{k=1}^{K} n_k \tilde{B}_k \leq C,
\]

for each state, for each group, select a bitrate according to a policy.

---

**Table 1**

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(C)</td>
<td>Capacity of the network connection</td>
</tr>
<tr>
<td>(\lambda_k)</td>
<td>Rate of the Poisson process at which group-(k) users start the video streams</td>
</tr>
<tr>
<td>(\beta_k)</td>
<td>Mean duration of the video streams for group (k)</td>
</tr>
<tr>
<td>(B_k)</td>
<td>Available bitrates for video streams for group (k)</td>
</tr>
<tr>
<td>(T_{\text{segment}_k})</td>
<td>Segment size used in the video streams for group (k)</td>
</tr>
<tr>
<td>(S)</td>
<td>State space of the Markov process</td>
</tr>
<tr>
<td>(\pi(x))</td>
<td>The probability that the Markov process is in state (x)</td>
</tr>
<tr>
<td>(n_k(x))</td>
<td>The number of group-(k) players in state (x)</td>
</tr>
<tr>
<td>(q_k(x))</td>
<td>The bitrate of group-(k) players in state (x)</td>
</tr>
<tr>
<td>(\gamma(x \rightarrow y))</td>
<td>The number of group-(k) players that change video bitrate when transitioning from state (x) to state (y)</td>
</tr>
<tr>
<td>(\mathbb{E}[N_k])</td>
<td>The expected number of players of group (k)</td>
</tr>
<tr>
<td>(\mathbb{E}[Q_k])</td>
<td>The expected number of bitrate switches for players of group (k)</td>
</tr>
<tr>
<td>(\mathbb{E}[B_k])</td>
<td>The expected bitrate for players of group (k)</td>
</tr>
</tbody>
</table>
Erlang multi-rate formula:

\[
\pi(n_1, n_2, \ldots, n_k) = \frac{1}{G} \prod_{k=1}^{K} \frac{\left(\lambda_k \beta_k\right)^{n_k}}{n_k!},
\]

\[
G = \sum_{x \in S} \prod_{k=1}^{K} \frac{\left(\lambda_k \beta_k\right)^{n_k(x)}}{n_k(x)!}.
\]

Uniformization:

\[
P_{x,y} = e^{-bT_{\text{segment}}} \sum_{m=0}^{\infty} \frac{(bT_{\text{segment}})^m}{m!} P_{x,y}^m \quad \text{for } x, y \in S.
\]

Nr. of players that switch video quality:

\[
\gamma_k(x \rightarrow y) = \begin{cases} 
0 & \text{if } q_k(x) = q_k(y), \\
\min(n_k(x), n_k(y)) & \text{if } q_k(x) \neq q_k(y)
\end{cases}
\]

Number of players:

\[
\mathbb{E}[N_k] = \sum_{x \in S} \pi(x)n_k(x),
\]

Expected video bitrate:

\[
\mathbb{E}[B_k] = \frac{1}{\mathbb{E}[N_k]} \sum_{x \in S} \pi(x)n_k(x)q_k(x).
\]

Expected nr. of switches:

\[
\mathbb{E}[Q_k] = \frac{1}{T_{\text{segment}} \mathbb{E}[N_k]} \sum_{x, y \in S} \pi(x)P_{x,y}\gamma_k(x \rightarrow y).
\]
360p video: 60s up to 1000 kbit/s
720p video: 120s up to 2000 kbit/s
1080p video: 180s up to 4000 kbit/s
Policy1 - equal share
Policy2 - device aware:

Table 2

<table>
<thead>
<tr>
<th>Quality level</th>
<th>360p</th>
<th>720p</th>
<th>1080p</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1000</td>
<td>2000</td>
<td>4000</td>
</tr>
<tr>
<td>2</td>
<td>600</td>
<td>1500</td>
<td>2000</td>
</tr>
<tr>
<td>3</td>
<td>400</td>
<td>1000</td>
<td>1500</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
<td>600</td>
<td>1000</td>
</tr>
<tr>
<td>5</td>
<td>400</td>
<td>400</td>
<td>600</td>
</tr>
<tr>
<td>6</td>
<td>400</td>
<td>400</td>
<td>400</td>
</tr>
</tbody>
</table>

Fig. 7. Model-based comparison of mean bitrates of a non device-aware and a device-aware sharing policy.

Fig. 8. Model-based comparison of quality switches for a non device-aware and a device-aware sharing policy.
Model versus Experiments

Policy2 - device aware:

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Device-aware video quality mapping (in kbit/s).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality level</td>
<td>360p</td>
</tr>
<tr>
<td>1</td>
<td>1000</td>
</tr>
<tr>
<td>2</td>
<td>600</td>
</tr>
<tr>
<td>3</td>
<td>400</td>
</tr>
<tr>
<td>4</td>
<td>400</td>
</tr>
<tr>
<td>5</td>
<td>400</td>
</tr>
<tr>
<td>6</td>
<td>400</td>
</tr>
</tbody>
</table>

![Fig. 9. Model-based mean bitrate versus mean bitrate achieved in experiments for a device-aware sharing policy.](image1)

![Fig. 10. Model-based instability rate versus instability rate achieved in experiments for a device-aware sharing policy.](image2)
DASH assisting network elements:

- improve streaming performance
- are programmable
- can be modelled → evaluated

Future efforts:

- Generalization to other experiences
- New applications of video (e.g. immersive experiences)
References:


Jan Willem Kleinrouweler
j.w.m.kleinrouweler@cwi.nl