Learning in situ: a randomized experiment in video streaming†

https://puffer.stanford.edu

Francis Y. Yan
Senior Researcher

Microsoft Research

†This work was completed at Stanford University with Hudson Ayers, Chenzhi Zhu, Sadjad Fouladi, James Hong, Keyi Zhang, Emily Marx, Philip Levis, and Keith Winstein.
Introduction: adaptive bitrate (ABR) video streaming

- Video streaming dominates Internet traffic

- Adaptive bitrate (ABR) is a key algorithm to optimize quality of experience (QoE)
  - primary goals: higher video quality, fewer stalls
  - prior work: BBA [SIGCOMM '14], MPC [SIGCOMM '15], CS2P [SIGCOMM '16], Pensieve [SIGCOMM '17], Oboe [SIGCOMM '18]
Introduction: adaptive bitrate (ABR) video streaming
Introduction: adaptive bitrate (ABR) video streaming

<table>
<thead>
<tr>
<th>1080p</th>
<th>1080p</th>
<th>1080p</th>
<th>1080p</th>
<th>1080p</th>
</tr>
</thead>
<tbody>
<tr>
<td>720p</td>
<td>720p</td>
<td>720p</td>
<td>720p</td>
<td>720p</td>
</tr>
<tr>
<td>480p</td>
<td>480p</td>
<td>480p</td>
<td>480p</td>
<td>480p</td>
</tr>
<tr>
<td>360p</td>
<td>360p</td>
<td>360p</td>
<td>360p</td>
<td>360p</td>
</tr>
</tbody>
</table>

...
Introduction: adaptive bitrate (ABR) video streaming

- ABR decides the quality level of each video chunk to optimize total QoE

```
<table>
<thead>
<tr>
<th>Time</th>
<th>1080p</th>
<th>1080p</th>
<th>1080p</th>
<th>1080p</th>
<th>1080p</th>
</tr>
</thead>
<tbody>
<tr>
<td>1080p</td>
<td>720p</td>
<td>720p</td>
<td>720p</td>
<td>720p</td>
<td>720p</td>
</tr>
<tr>
<td>720p</td>
<td>480p</td>
<td>480p</td>
<td>480p</td>
<td>480p</td>
<td>480p</td>
</tr>
<tr>
<td>480p</td>
<td>360p</td>
<td>360p</td>
<td>360p</td>
<td>360p</td>
<td>360p</td>
</tr>
</tbody>
</table>
```
Outline

1. **Puffer**: a live streaming platform for video streaming research

2. **Finding**: confidence intervals in video streaming are bigger than expected

3. **Fugu**: an ML-based ABR algorithm learned in situ
1. **Puffer**: a live streaming platform for video streaming research

2. **Finding**: confidence intervals in video streaming are bigger than expected

3. **Fugu**: an ML-based ABR algorithm learned *in situ*
Puffer: a live streaming platform running a randomized experiment

• Free live TV streaming website (puffer.stanford.edu)
• Opened to public December 2018
• User sessions are randomized to different algorithms
• Goal: realistic testbed and learning environment for video streaming research
Website: puffer.stanford.edu
Ads for “live tv” and “tv streaming”
Puffer architecture

TV Antenna

ATSC Demodulator

Decoder/Encoder 1

Decoder/Encoder 2

Decoder/Encoder 3

Video Server (ABR)

Video Client

User Database

Time Series Database

Monitoring System

Francis Y. Yan (MSR)
Puffer statistics

- 32,000 lines of code
  - 1,606 commits
  - 78,497++
  - 46,623--

- 130,000 real users

- 60 years of video streamed
Reproducible research and open platform
Outline

1. **Puffer**: a live streaming platform for video streaming research

2. **Finding**: confidence intervals in video streaming are bigger than expected

3. **Fugu**: an ML-based ABR algorithm learned *in situ*
Confidence intervals in video streaming are bigger than expected

• Existing ABR algorithms found benefits like 10%–20% based on experiments lasting hours between a few network nodes

• We found: 2 years of data per scheme are needed to measure 20% precision
Confidence intervals in video streaming are bigger than expected

- Results on the *day* of Jan. 26, 2019, with 17 days of video streamed to 600 users
Confidence intervals in video streaming are bigger than expected

- Results in the week starting from Jan. 26, 2019, streaming 42 days of video

![Graph showing average SSIM vs. time spent stalled for different video streaming methods: BBA, MPC-HM, Pensieve, Fugu, RobustMPC-HM. The graph indicates Better QoE with higher average SSIM and lower time spent stalled.](image-url)
Confidence intervals in video streaming are bigger than expected

• Results in the month starting from Jan. 26, 2019, streaming 169 days of video
Confidence intervals in video streaming are bigger than expected

- Results in an *eight-month* period after Jan. 26, 2019, streaming > 13 years of video
Confidence intervals in video streaming are bigger than expected

- Need 2 years of video per scheme to reliably measure a 20% difference

- Reason: Internet is way more noisy and heavy-tailed than we thought
  - only 4% of the 637,189 streams had *any* stalls
  - distributions of throughputs and watch times are highly skewed
1. **Puffer**: a live streaming platform for video streaming research

2. **Finding**: confidence intervals in video streaming are bigger than expected

3. **Fugu**: an ML-based ABR algorithm learned *in situ*
The only system uncertainty is *transmission time* of each chunk.
Fugu’s transmission time predictor (TTP)

- Neural network predicts “how long would each chunk take?”
Fugu’s transmission time predictor (TTP)

- Neural network predicts “how long would each chunk take?”

- Input:
  - sizes and transmission times of past chunks
Fugu’s transmission time predictor (TTP)

- Neural network predicts “how long would each chunk take?”

- Input:
  - sizes and transmission times of past chunks
  - size of a chunk to be transmitted \(\text{not}\ \text{a throughput predictor}\)

Fact: observed throughput varies with file size
Fugu’s transmission time predictor (TTP)

- Neural network predicts “how long would each chunk take?”

- Input:
  - sizes and transmission times of past chunks
  - size of a chunk to be transmitted (*not* a throughput predictor)
  - low-level TCP statistics (*min RTT, RTT, CWND, packets in flight, delivery rate*)
Fugu’s transmission time predictor (TTP)

- Neural network predicts “how long would each chunk take?”

- Input:
  - sizes and transmission times of past chunks
  - size of a chunk to be transmitted (not a throughput predictor)
  - low-level TCP statistics (min RTT, RTT, CWND, packets in flight, delivery rate)

- Output:
  - probability distribution over transmission time (not a point estimate)
Learning TTP *in situ* (in place)

- Training: supervised learning *in situ* on real data from deployment environment
  - chunk-by-chunk series of each individual video stream
  - chunk $i$: size, timestamp sent, timestamp acknowledged, TCP statistics right before sending
Learning TTP *in situ* (in place)

- Training: supervised learning *in situ* on real data from deployment environment
  - chunk-by-chunk series of each individual video stream
  - chunk $i$: size, timestamp sent, timestamp acknowledged, TCP statistics right before sending

- Learning *in situ* does **not** replay throughput traces or require network simulators
  - we don't know how to faithfully simulate the Internet
Fugu’s model-based controller

- Objective function: expected sum of QoE in the lookahead horizon
- QoE: +video quality, −quality variation, −rebuffering
Fugu’s model-based controller

- Given TTP, optimal plan can be computed in real time

with dynamic programming

\[
v_i^*(B_i, K_{i-1}) = \max_{K_i^s} \left\{ \sum_{t_i} \Pr[\hat{T}(K_{i}^s) = t_i] \cdot (QoE(K_i^s, K_{i-1}) + v_{i+1}^*(B_{i+1}, K_i^s)) \right\}
\]
Fugu’s model-based controller

- Replan at every step (model predictive control)
- Mitigate accumulation of errors

10 versions

5-step lookahead
Fugu’s model-based controller

- Replan at every step (model predictive control)
- Mitigate accumulation of errors

10 versions

5-step lookahead

10 versions
Fugu is a model-based reinforcement-learning algorithm
Evaluation: SSIM vs stalls

Average SSIM (dB) vs Time spent stalled (%)

- BBA
- MPC-HM
- Pensieve
- RobustMPC-HM

637,189 streams
13.1 stream-years

Better QoE
Evaluation: SSIM vs stalls

![Graph showing the comparison of Average SSIM (dB) vs Time spent stalled (%). The graph includes data points for BBA, MPC-HM, Pensieve, Fugu, and RobustMPC-HM. The graph indicates Better QoE with 637,189 streams and 13.1 stream-years.](image-url)
### Results of primary experiment (Jan. 26–Aug. 7 & Aug. 30–Oct. 16, 2019)

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Time stalled</th>
<th>Mean SSIM</th>
<th>SSIM variation</th>
<th>Mean duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fugu</td>
<td>0.13%</td>
<td>16.64 dB</td>
<td>0.74 dB</td>
<td>33.6 min</td>
</tr>
<tr>
<td>MPC-HM</td>
<td>0.22%</td>
<td>16.61 dB</td>
<td>0.79 dB</td>
<td>30.8 min</td>
</tr>
<tr>
<td>BBA</td>
<td>0.19%</td>
<td>16.56 dB</td>
<td>1.11 dB</td>
<td>32.1 min</td>
</tr>
<tr>
<td>Pensieve</td>
<td>0.17%</td>
<td>16.26 dB</td>
<td>1.05 dB</td>
<td>31.6 min</td>
</tr>
<tr>
<td>RobustMPC-HM</td>
<td>0.12%</td>
<td>16.01 dB</td>
<td>0.98 dB</td>
<td>31.0 min</td>
</tr>
</tbody>
</table>
Evaluation: cold-start performance

Average first-chunk SSIM (dB) vs. Startup delay (s)

- Fugu
- MPC-HM
- RobustMPC-HM
- Pensieve
- BBA

Better QoE

Francis Y. Yan (MSR)
Takeaways

1. **Puffer**: a video streaming platform opened to research community
   - 130,000+ real users, streamed 60+ years of video

2. **Finding**: confidence intervals in video streaming are bigger than expected
   - we need 2 years of data per scheme to measure 20% precision

3. **Fugu**: an ML-based ABR algorithm learned *in situ*
   - Transmission Time Predictor (TTP)

Francis Y. Yan <francisy@microsoft.com>