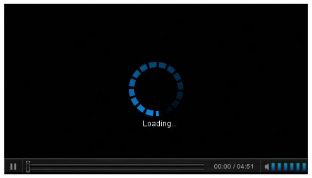
Yi Sun et al. “CS2P: Improving Video Bitrate Selection and Adaptation with Data-Driven Throughput Prediction”, in SIGCOMM 2016

Taeju Park, Yibo Pi

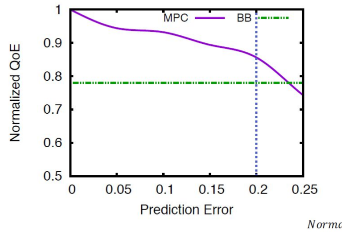
Introduction

- Bitrate selection and adaptation is critical to ensure good quality-of-experience (QoE) for Internet video.
  - Initial startup latency
  - The amount of rebuffering during the session
  - Average bitrate of the rendered video



Need better throughput prediction

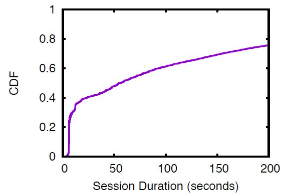
- Accurate throughput prediction helps two aspects.
  - Initial bitrate selection
    - Higher bitrate with no rebuffering or short startup time.
  - Midstream bitrate adaptation
    - When the error is <= 20%, N-QoE of MPC is close to optimal >85%.



$$\text{Normalized QoE} = \frac{\text{Actual QoE}}{\text{Theoretical optimal}}$$

Dataset

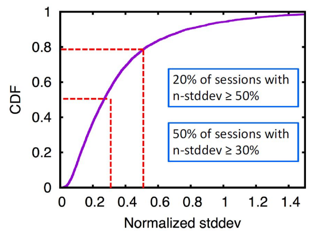
- Throughput variability across sessions and within a session
- Use proprietary dataset
  - iQIYI, leading commercial video provider in China (over 219 million users)
  - Over 20 million sessions covering 3 million unique client IPs and 18 server IPs over 8 days
  - The client spans 736 cities and 87 ISPs in China.
  - Within each session, they have recorded the average throughput for each 6 second “epoch”



(a) Duration

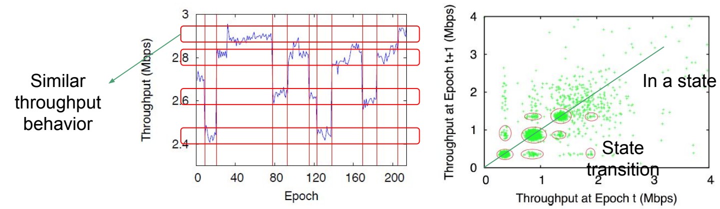
Observations from dataset

- (Observation 1) There is a significant amount of throughput variability within a video session



Observations from dataset

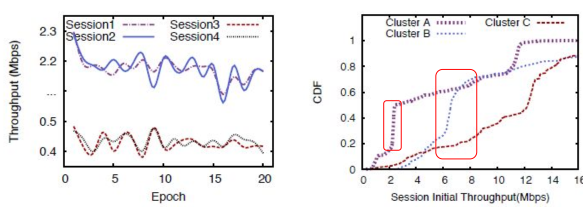
- (Observation 2) The evolution of the throughput within a session exhibits stateful/persistent characteristics.



(a) An example session (b) Throughput variation at two consecutive epochs

Observation from dataset

- (Observation 3) Sessions with similar features exhibit similar initial throughput and evolution pattern.



(a) Example of similar sessions (b) CDF of initial throughput at different clusters

Observations from dataset

- (Observation 4) The relationship between session features and throughput is quite complex.

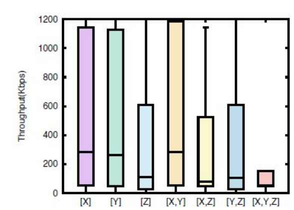


Figure 6: The throughput variation of sessions matching all and a subset of three features: X=*ISP*, Y=*City*, Z=*Server*.

CS2P Workflow

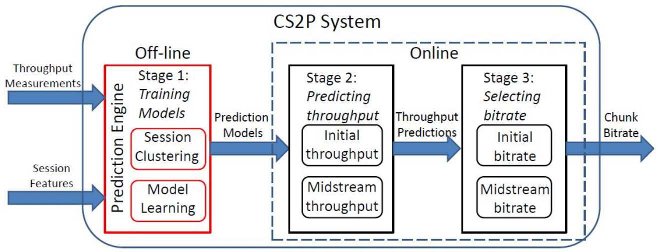
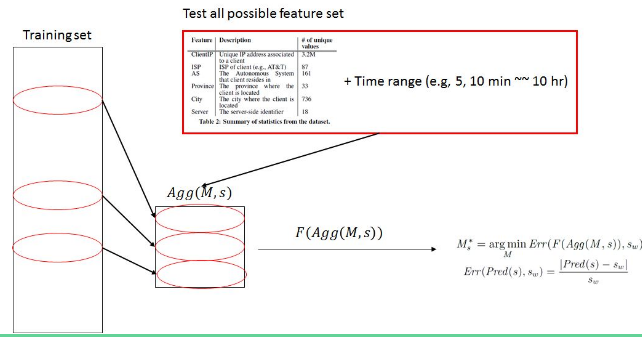


Figure 1: Overall workflow of CS2P.

Session Clustering

- How to cluster similar session?
  - Choice the key features and time range which minimize prediction error



Modeling behavior

- HMM-based predictor capturing the state-transition behavior in each cluster.
  - Throughput depends on the hidden state (e.g, the number of users at a bottleneck link)
  - Given the hidden state, assume pdf of throughput is Gaussian $W_t | X_t = x \sim N(\mu_x, \sigma_x^2)$
  - Learn HMM parameters (Initial probability, transition probability, emission probability) through expectation-maximization(EM) algorithm.

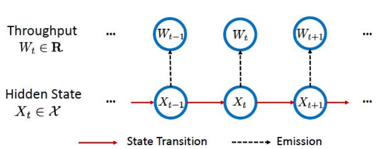


Figure 7: Overview of HMM model.

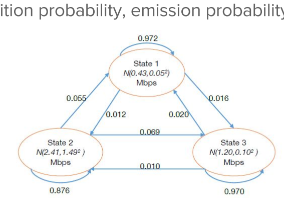


Figure 8: Example of hidden-markov model of session clusters.

Online prediction

- A new session is mapped to the most similar session in the training dataset
- Throughput prediction for initial epoch

Predicted throughput $$\hat{W}_1 = Median(Agg(M_s^*, s))$$

- Throughput prediction for midstream epoch

$$\hat{W}_t = \mu_x, \quad x = \arg\max_{x \in \mathcal{X}} \mathbb{P}(X_t = x | W_{1:t-1})$$

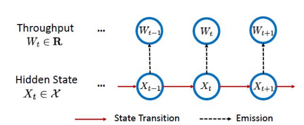


Figure 7: Overview of HMM model.

Bitrate Selection

- Midstream bitrate selection

1. Throughput prediction
2. Current bitrate
3. Buffer occupancy

→ Model Predictive Control (MPC) → Next bitrate

Key idea: maximize quality of experience (QoE)

- Initial bitrate selection
  - MPC cannot be used used due to lack of current bitrate
  - Highest bitrate below predicted initial throughput

Player Integration

- How to use CS2P?
  - Server-side solution
    - Centralized server computes bitrates for each video session
    - **Advantage**: simple, no modifications on the clients
    - **Disadvantage**: the server is a potential bottleneck
  - Client-side solution
    - Each client downloads their own HMM and initial throughput
    - **Advantage**: quickly detect performance change and respond
    - **Disadvantage**: clients need to maintain HMM

Experiment Implementation

Video player → 1. Send a POST request for throughput prediction → Server

Server → 2. Send back predicted throughput → Video player

Multiple bitrate algorithms:
1. MPC,
2. Rate-based (RB)
3. Buffer-based (RB) ------ BBA

Session clustering
HMM model building
Online throughput prediction

Evaluation:
Data-driven simulation and pilot deployment

Data-driven simulation

- Baseline solutions
  - History-based predictors
    - LS (Last Sample)
    - HM (Harmonic mean)
    - AR (auto regression)
  - Machine-learning predictors
    - SVR (Support vector regression)
    - GBR (Gradient Boosting Regression)

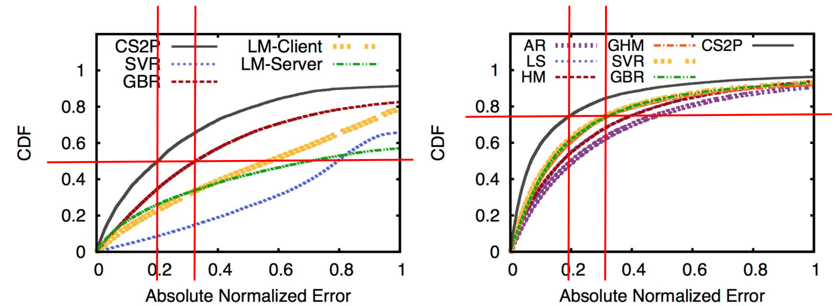
$$H = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \cdots + \frac{1}{x_n}} = \frac{n}{\sum\limits_{i=1}^{n} \frac{1}{x_i}} = \left(\frac{\sum\limits_{i=1}^{n} x_i^{-1}}{n} \right)^{-1}$$

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t$$

Data-driven simulation

- Model configuration
  - Cross-validation for design parameter selection
    - Divide sessions in a day into 4 subsets
    - 3 subsets train and 1 subset test
    - Resulting parameters
      - 6-state HMM
      - group size (# of sessions in a cluster) 100
  - Limitation
    - Throughput data from fixed bitrate video download
- Video parameters: video length (260s) and 5 bitrate levels

Improvement in Prediction Accuracy



(a) Initial epoch

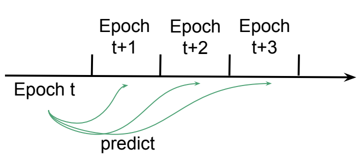
(b) Midstream epoch

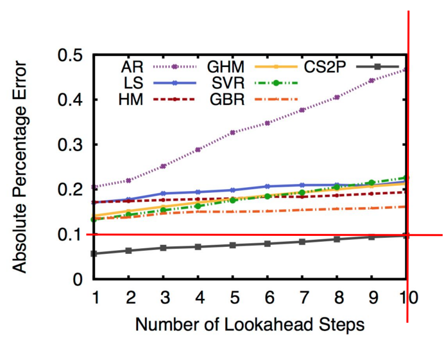
CS2P: median error < 20%
Others: median error > 35%

CS2P: 75th percentile < 20%
Others: 75th percentile > 30%

Improvement in prediction accuracy

Impact of look-ahead horizon





(c) 10-epoch lookahead

Improvement in QoE: Overall QoE

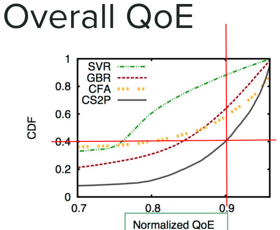
bitrate

$$QoE_1^K = \sum_{k=1}^{K} R_k - \lambda \sum_{k=1}^{K-1} |R_{k+1} - R_k| - \mu \sum_{k=1}^{K} \left(\frac{d_k(R_k)}{C_k} - B_k \right)_+ - \mu_s T_s$$

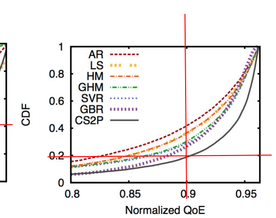
Size of chunk k

Buffer occupancy

Startup delay



(a) Initial chunk



(b) Midstream chunk

Normalized with offline optimal QoE
(future throughput is known)

Initial chunk: 61% sessions have >90% QoE
Midstream chunk: 81% sessions have > 90% QoE

Improvement in QoE: Overall QoE

	Initial AvgBitrate	Initial GoodRatio	Midstream AvgBitrate	Midstream GoodRatio
AR	NULL	NULL	3.31Mbps	96.6%
LS	NULL	NULL	4.08Mbps	93.2%
HM	NULL	NULL	3.80Mbps	97.2%
CFA	1.93Mbps	87.9%	NULL	NULL
SVR	1.52Mbps	81.4%	4.64Mbps	92.6%
GBR	2.09Mbps	93.8%	4.28Mbps	98.0%
CS2P	**4.27Mbps**	**98.5%**	**4.97Mbps**	**99.1%**

Table 3: Comparing AvgBitrate vs. GoodRatio among different predictors.

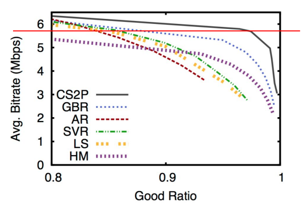
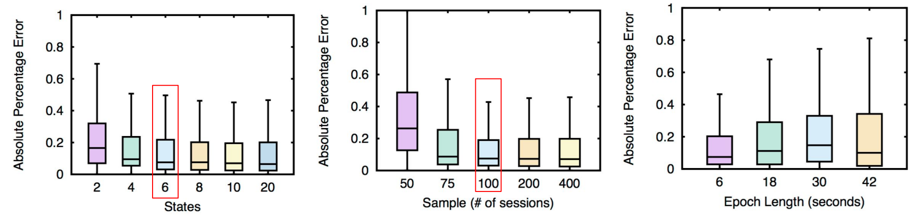


Figure 11: Tradeoff between AvgBitrate and GoodRatio.

AvgBitrate: average value of selected birates
GoodRatio: percentage of chunks with no re-buffering

Sensitivity Analysis



(a) Error vs. HMM states (b) Error vs. Group size (c) Error vs. Measurement frequency

Figure 12: Sensitivity analysis of CS2P parameters.

Generally independent of measurement granularity??

Pilot Deployment

- Evaluate in the wild
- Scale: 200+ client video players from 5 university campuses

Metrics	vs. HM+MPC	vs. BB
AvgBitrate	10.9%	9.3%
GoodRatio	2.5%	17.6%
Bitrate Variability	-2.3%	5.6%
Startup Delay	0.4%	-3.0%
Overall QoE	**3.2%**	**14.0%**

Table 4: QoE improvement by CS2P +MPC compared with HM+MPC and BB in a real-world experiment in 4 cities of China.

Pilot Deployment

- Deployment in video on demand (VoD) service
- Estimate total rebuffering time at the beginning of **fixed-bitrate** sessions

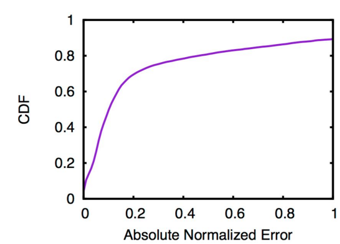


Figure 13: Prediction error on total rebuffering time.

Discussion

- Weakness
  - States from the training set cannot capture unexpected situations
    - Training set only contains limited situations
  - High complexity of feature selection
    - 6 static features + large amount of possible window sizes
- Extensions
  - Clustering clients based on other features (e.g., throughput)
    - Attributes (city, ISP) of clients may be wrong
  - Other methods for choosing initial throughput
    - Instead of median, how about other models, e.g., regression?