AZEEM: Efficient Video Playback Optimization Under Device Diversity and Drift

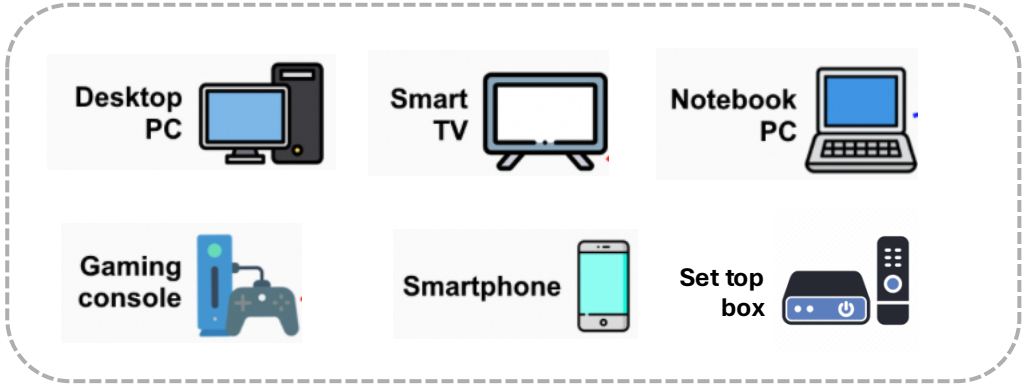
Harsha Sharma[1]*, Pouya Hamadanian[1]*, Arash Nasr-Esfahany[1], Zahaib Akhtar[2], Mohammad Alizadeh[1]

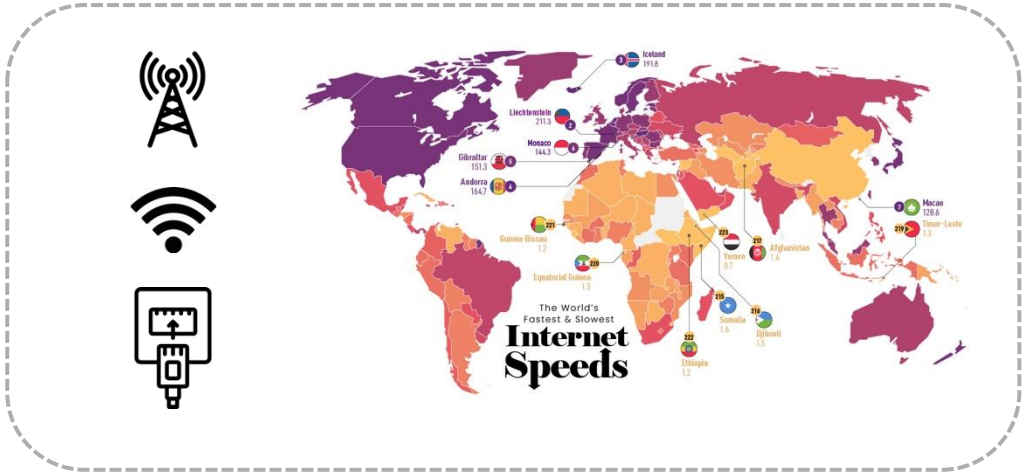
Equal Contribution*

Massachusetts Institute of Technology

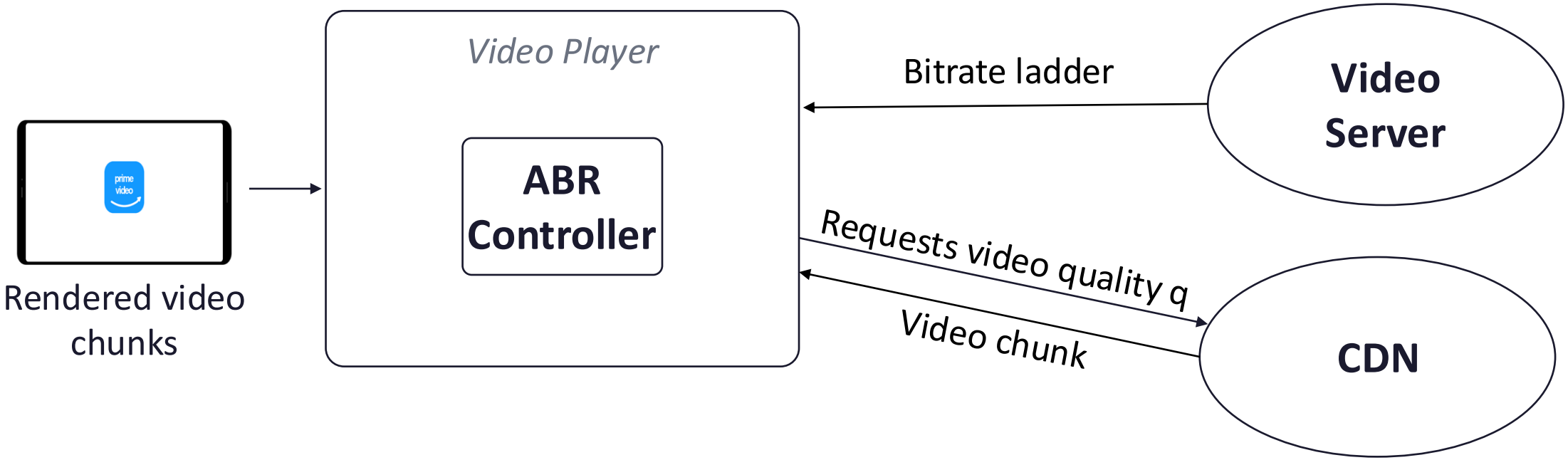
amazon prime video

Large-scale Video Service Providers

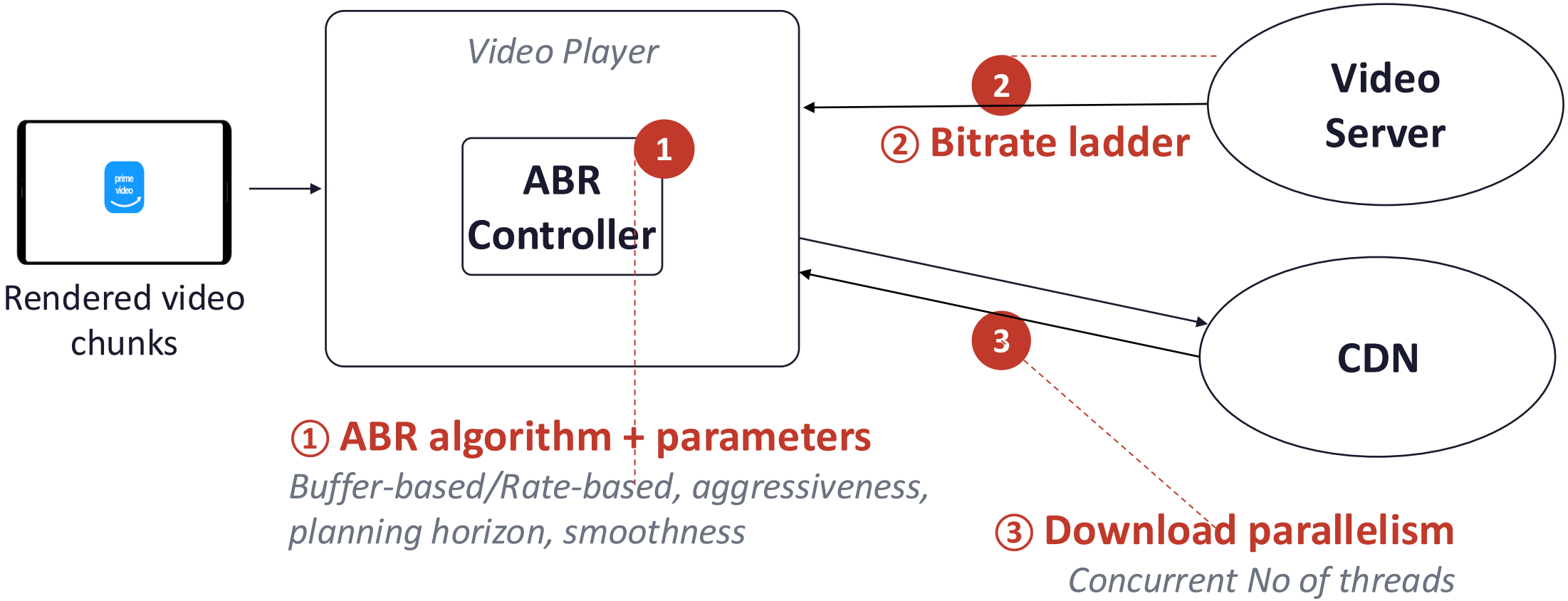




Video playback system



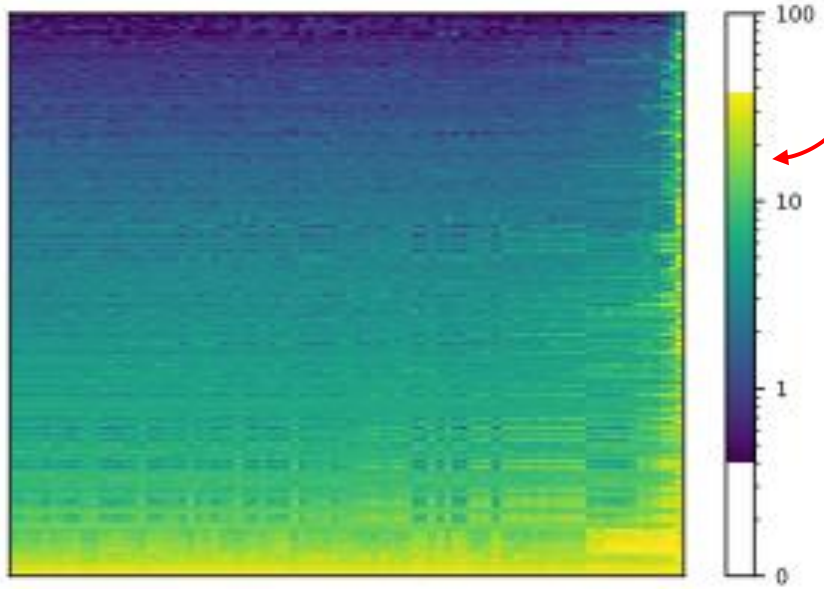
Video playback has many tunable knobs



Prime Video RCT Data

~750 million sessions

250 Cohorts:
<Device,
network cluster,
content quality>



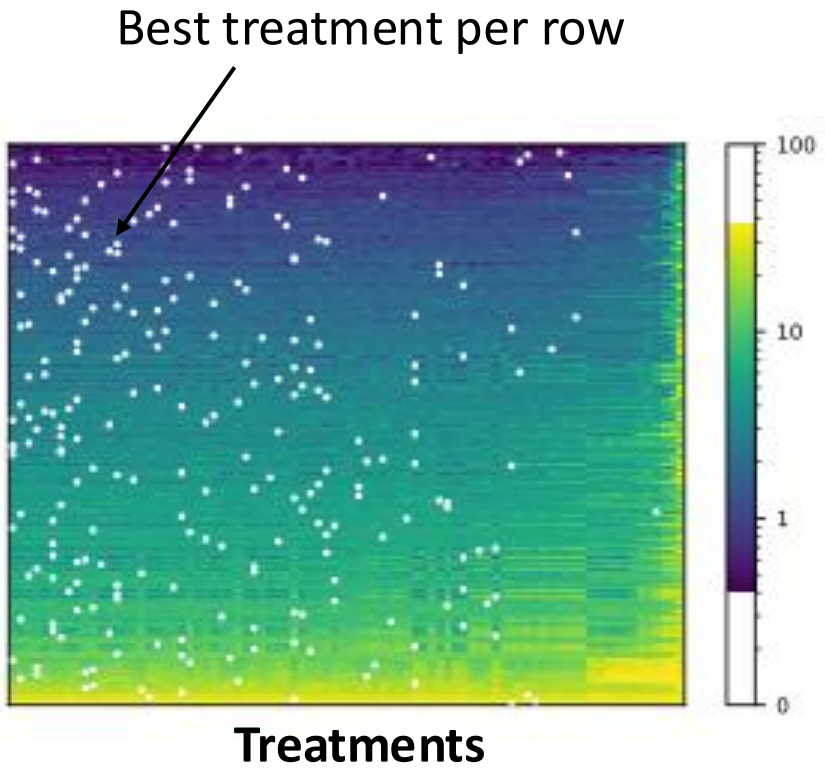
84 Treatments: Tunable knobs

QoE degradation:

- % of sessions with rebuffering
- % of non-HD fragments
- % of sessions with startup delay
- % of sessions with bitrate changes

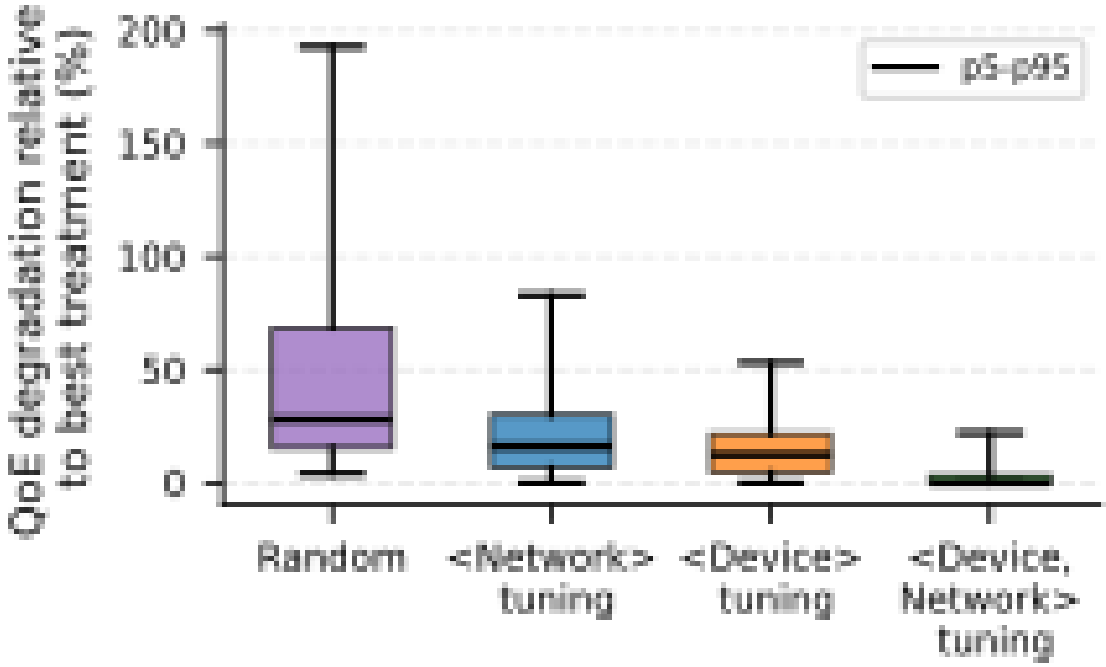
Playback Optimization: Configuration tuning

Cohorts:
<Device,
network cluster,
content quality>



Tuning Granularity: <Network> vs <Network, Device>

High performing, production treatments



Case for device-level tuning

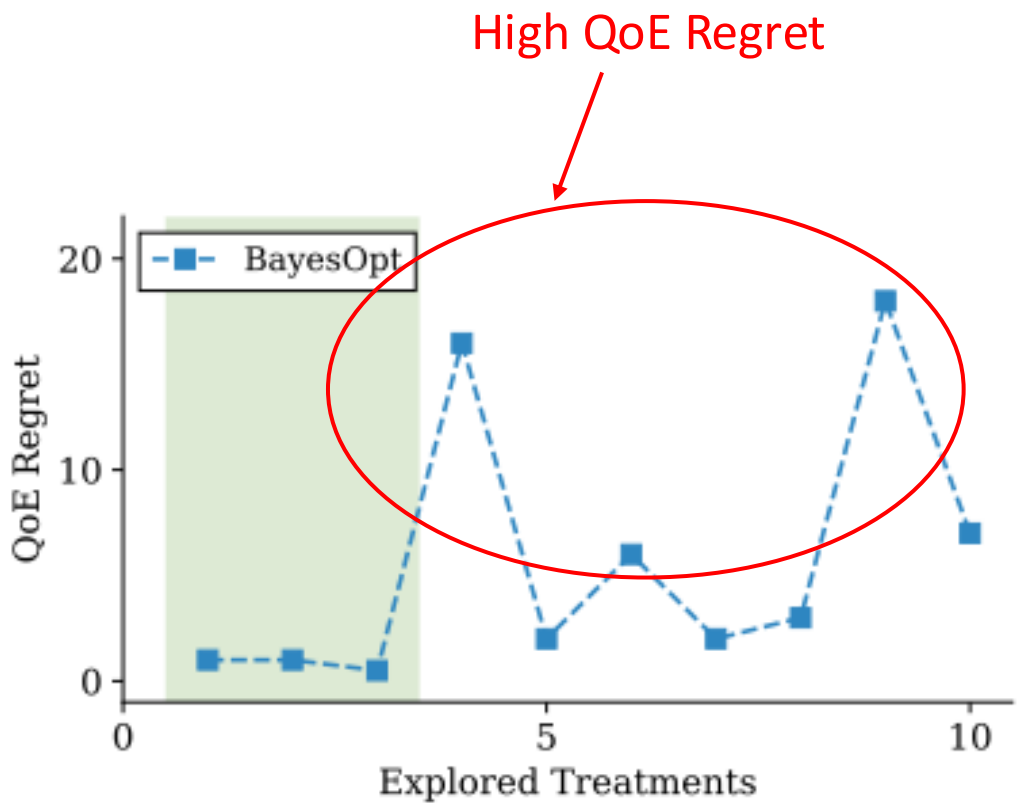
Streaming devices differ widely in

- hardware: CPU, memory, screen size
- software: OS, firmware, decoder, app stack

% Sessions with Rebuffering

# of Concurrent Downloads	1	2	3
Device A	3.34	2.64	2.57
Device B	6.19	5.74	5.68
Device C	1.88	2.55	3.29
Device D	7.76	11.43	13.85

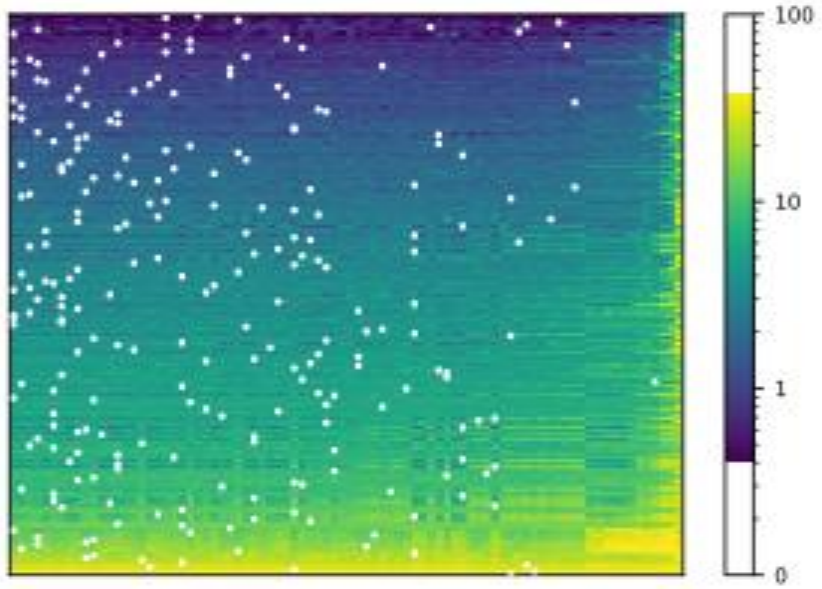
Per-cohort tuning is expensive



- Bayesian Optimization/Bandits -> exploration cost => High QoE regret
- Hundreds of device types in production
- Heavy-tailed QoE → 1000s of sessions to compare treatments
- **Per-cohort tuning -> high QoE regret + slow convergence**

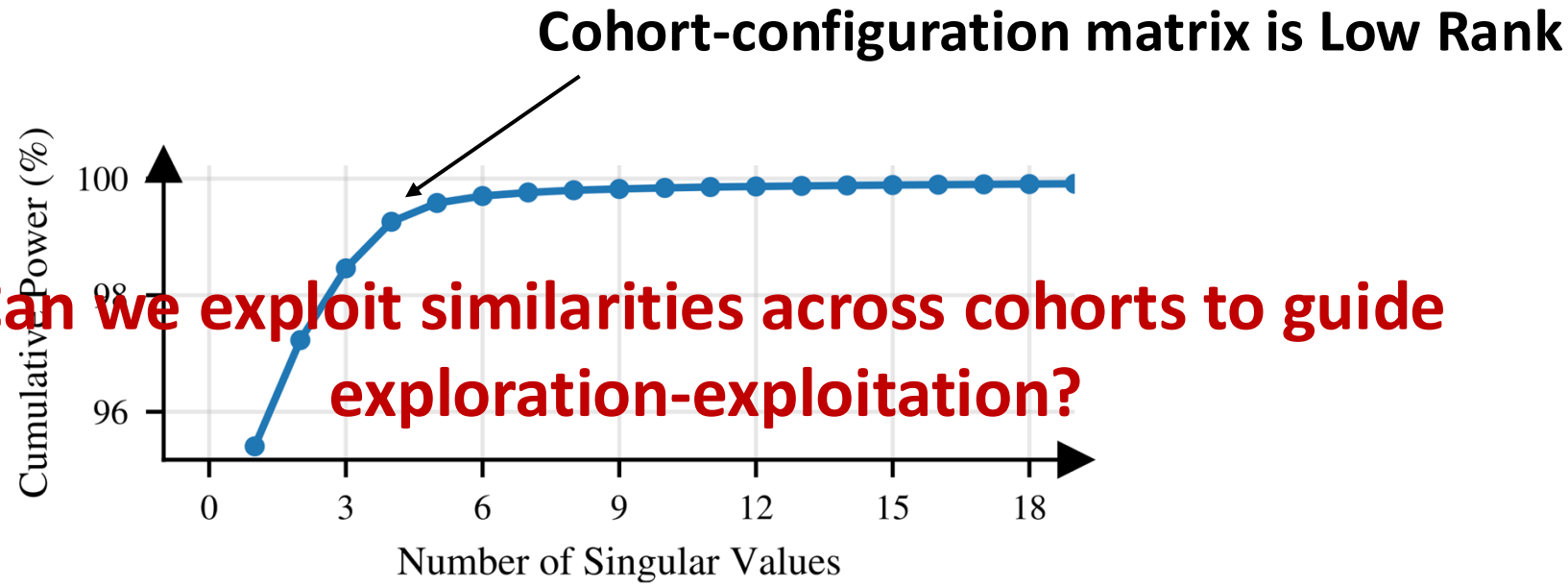
Similarities across Devices

Cohorts:
<Device,
network cluster,
content quality>



Treatments

Similarities across Devices



Can we exploit similarities across cohorts to guide exploration-exploitation?

AZEEM overview

New Cohort

C1	C2	C3	C4	C5	C6	C7	C8

QoE Unknown

Already Onboarded Cohorts

Cohort	C1	C2	C3	C4	C5	C6	C7	C8
1								
2								
3								
4								

Bayesian Optimization/Bandits ->
Explores entire configuration space

AZEEM: Use historical data->
few-shot prediction for new cohort

AZEEM overview

New Cohort

C1	C2	C3	C4	C5	C6	C7	C8

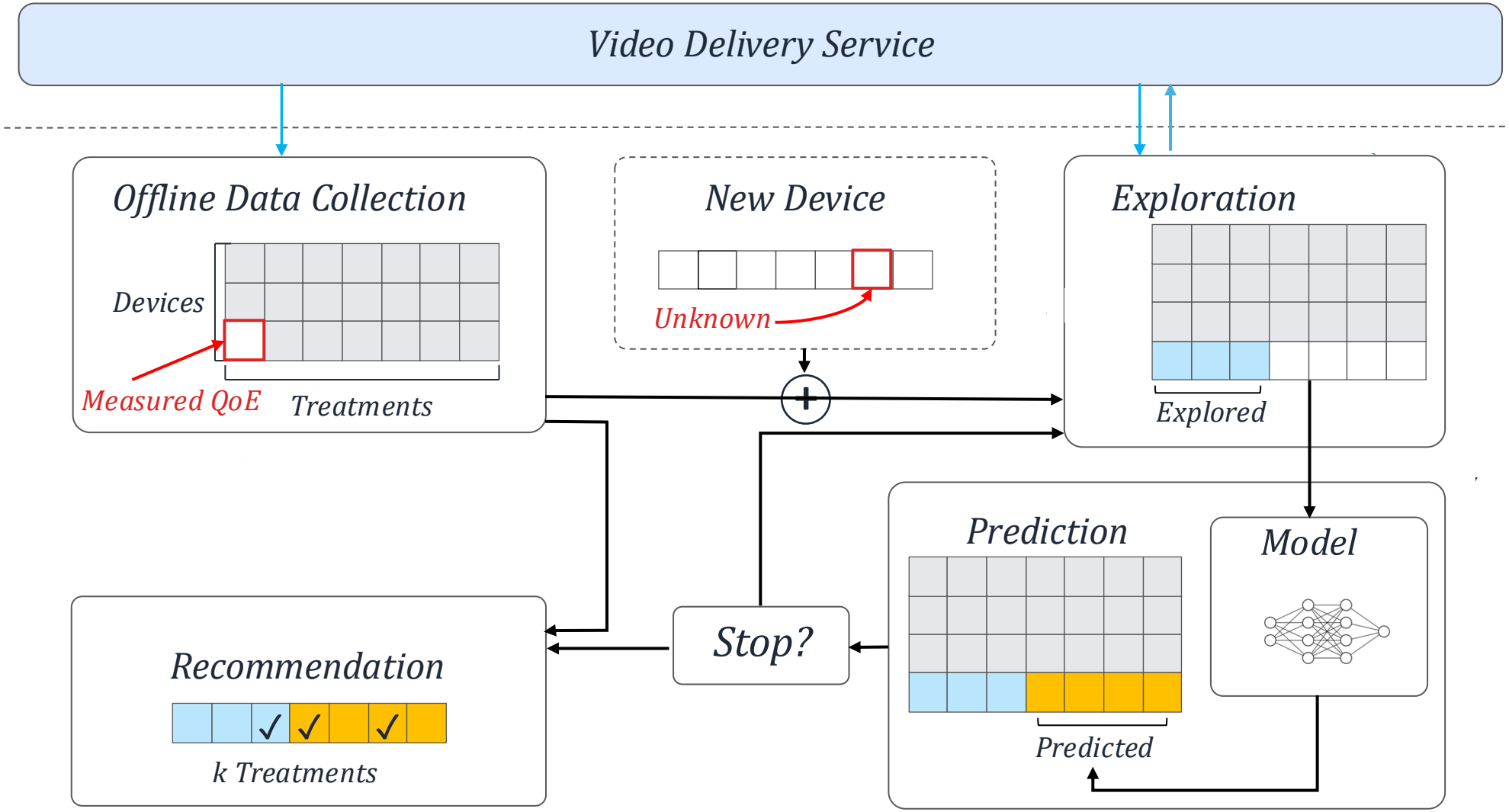
Already Onboarded Cohorts

Cohort	C1	C2	C3	C4	C5	C6	C7	C8
1								
2								
3								
4								

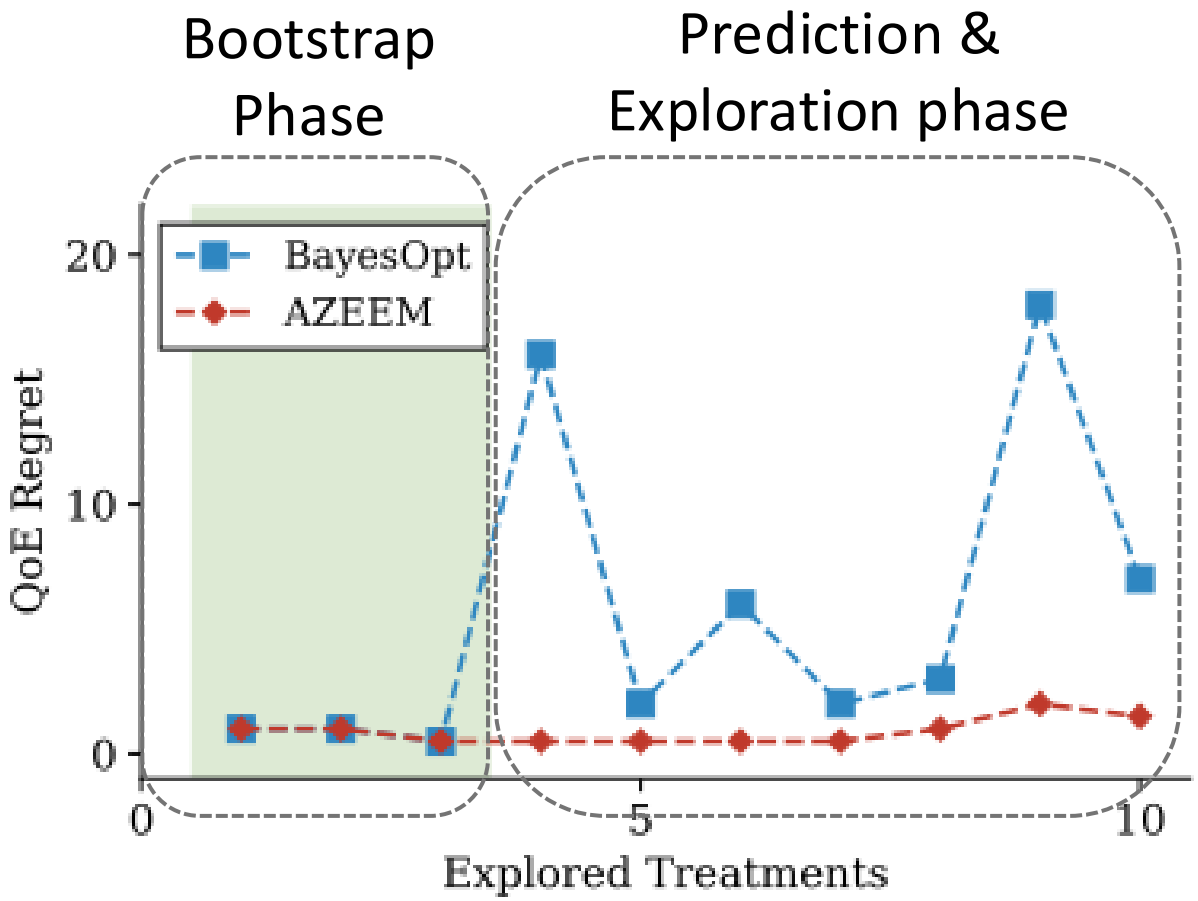
- **Efficient exploration**
- **Robust to temporal drift**

AZEEM: Use historical data-> few-shot prediction for new cohort

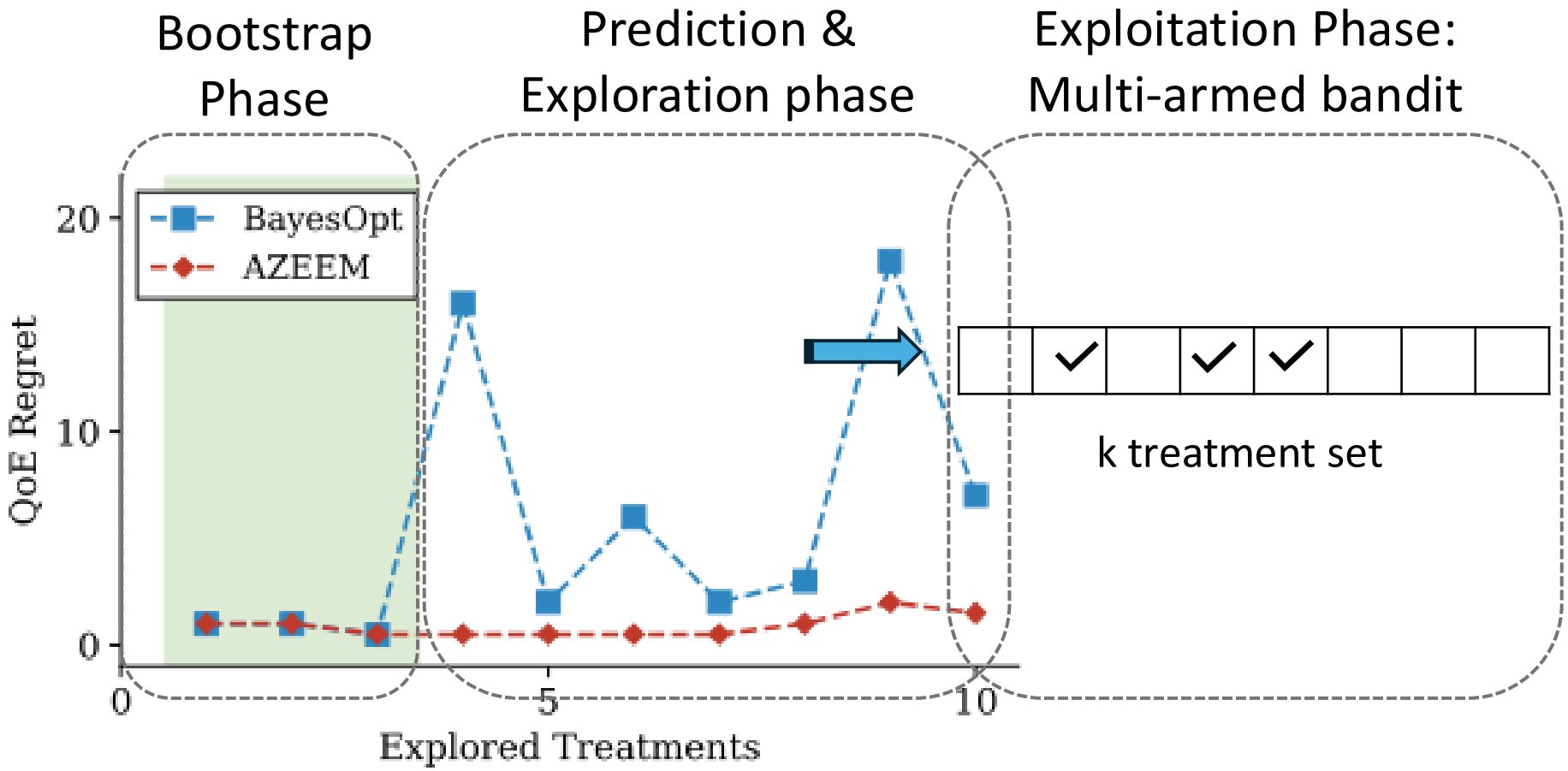
AZEEM design



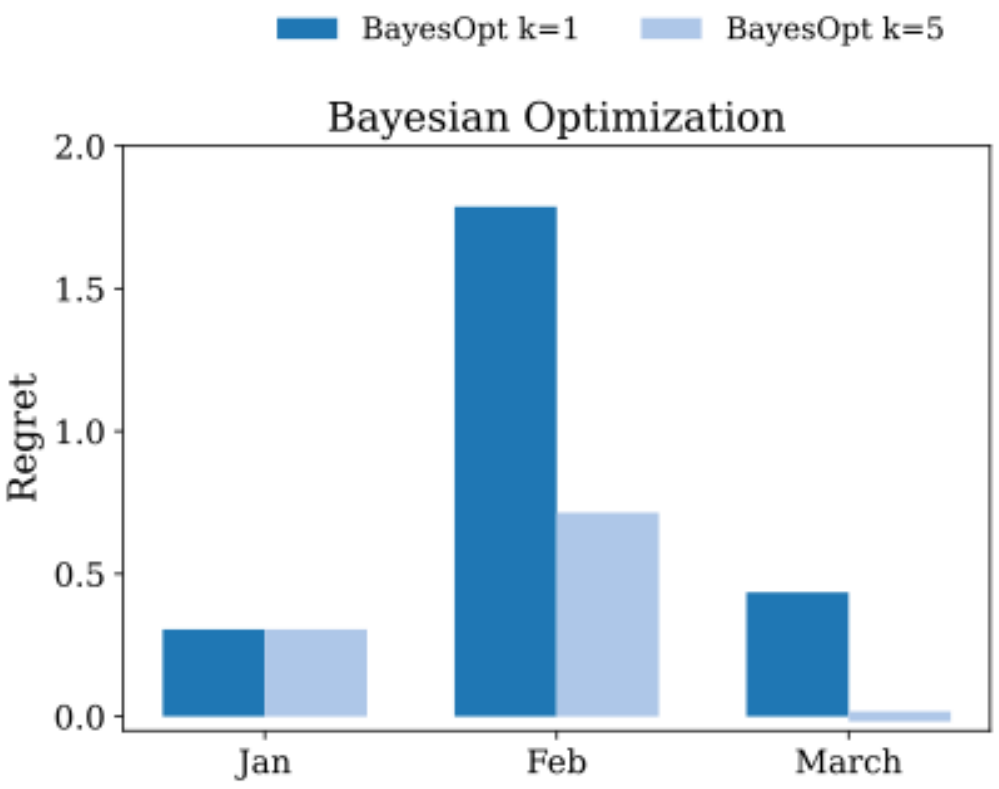
Guided exploration-exploitation with AZEEM



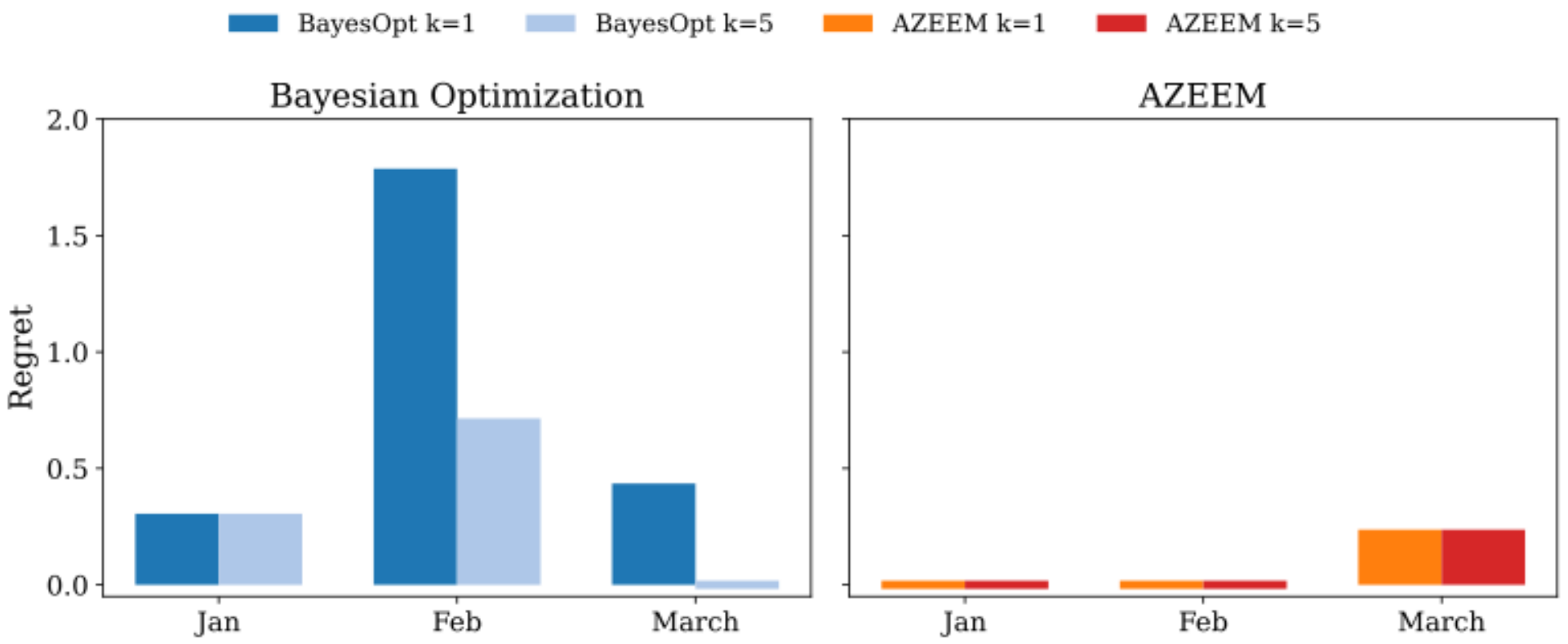
Guided exploration-exploitation with AZEEM



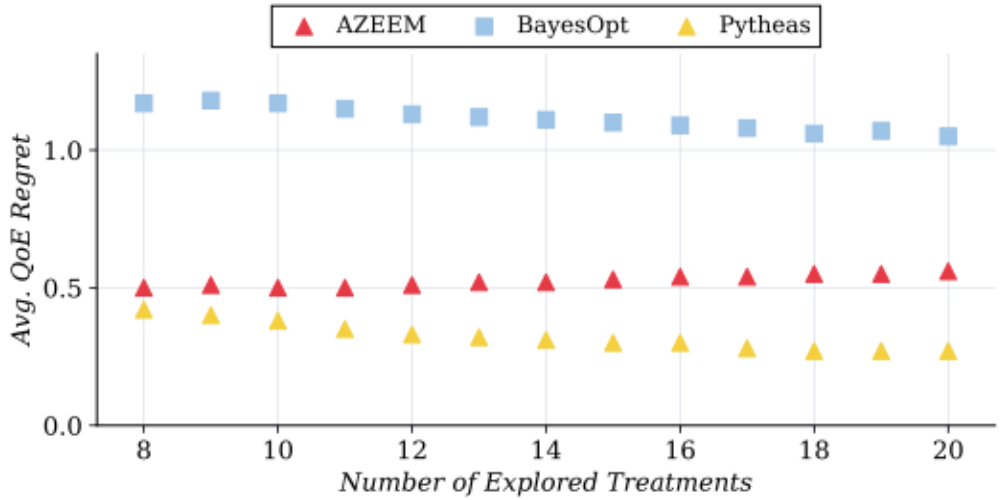
Robust to temporal drift



Robust to temporal drift

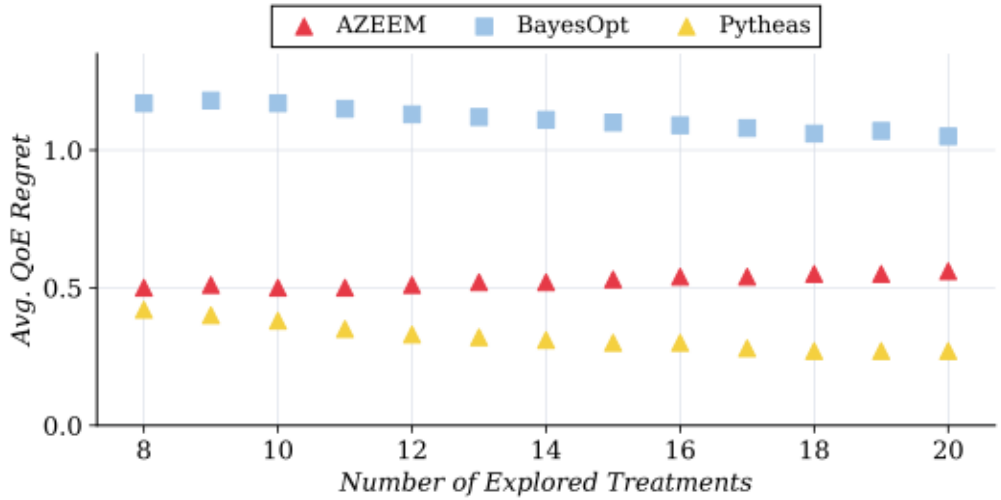


AZEEM results

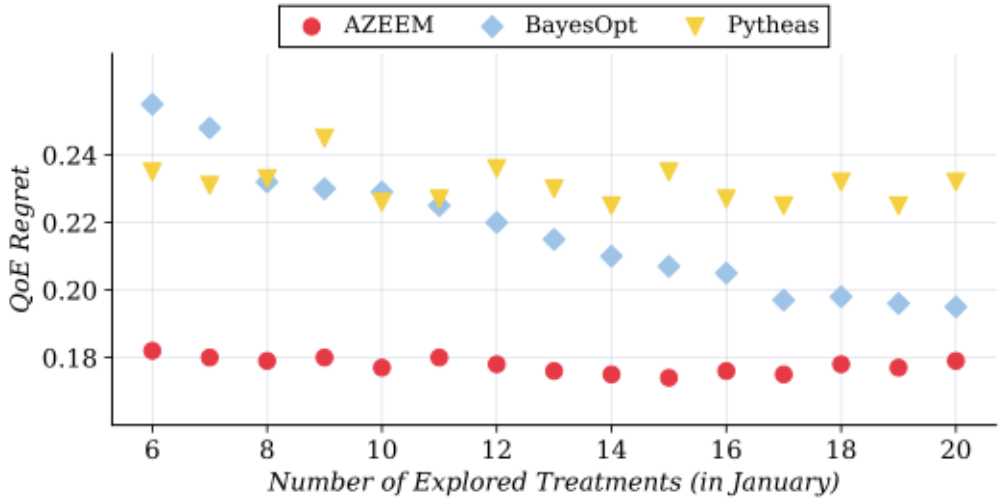


(a) Exploration Period (January 2025)

AZEEM results



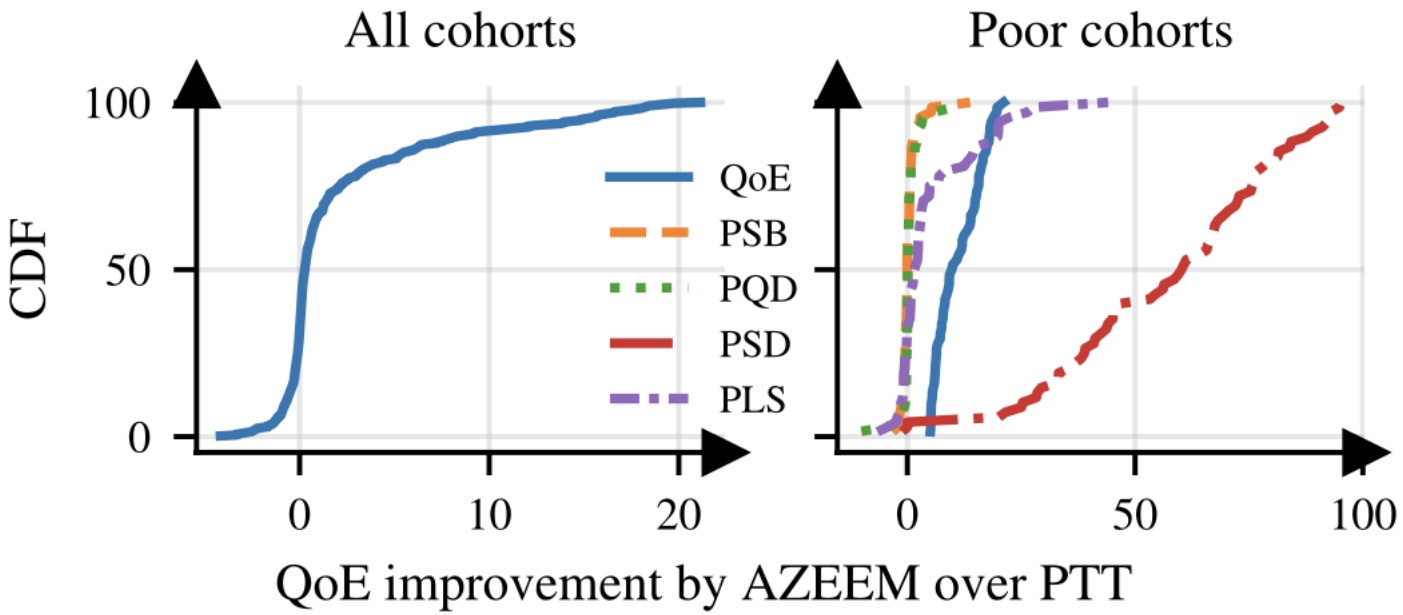
(a) Exploration Period (January 2025)



(b) Exploitation Period (February 2025), $k = 5$

AZEEM Live deployment case study

PTT: Production service that runs bandits over handpicked treatments



- **PSB**: % of sessions with rebuffering
- **PQD**: % of non-HD fragments
- **PSD**: % of sessions with startup delay
- **PLS**: % of sessions with bitrate changes

Conclusion

- **Device-level tuning is critical**
- **AZEEM:**
  - Efficient exploration via cross-cohort sharing
  - Robust to temporal drift
- **RCT evaluation:**
  - Outperforms BO + bandits
  - **3× faster convergence**
- **Live deployment (vs Production Tuning Service):**
  - **+2.2% avg QoE**
  - **>5% gains for 17% of worst cohorts**

Questions?!