Configanator: A Data-driven Approach to Tackle Network Diversity with Heterogeneous Configurations

Usama Naseer, Theophilus A. Benson

BROWN

Web Performance Matters!

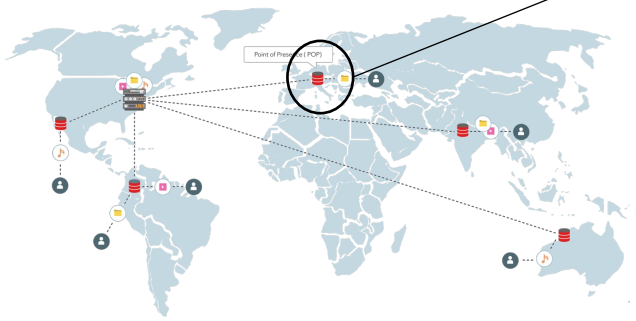
Performance of digital services have a direct impact on businesses & society.

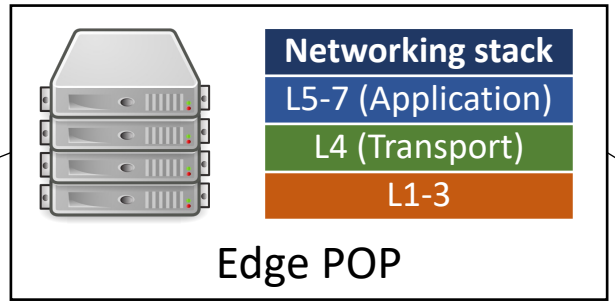
Increases user engagement

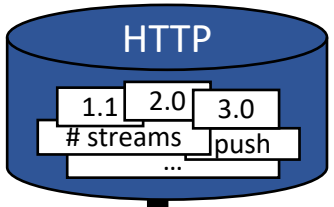
Growth in revenue

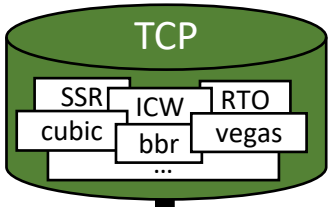
Improves productivity

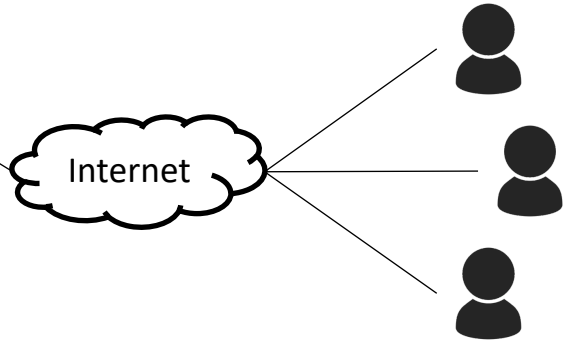
CDNs and Protocol Configurations at Edge









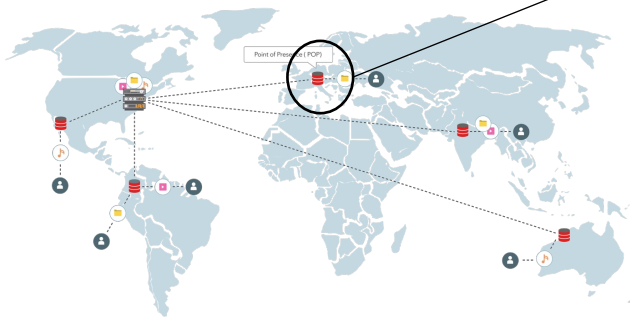


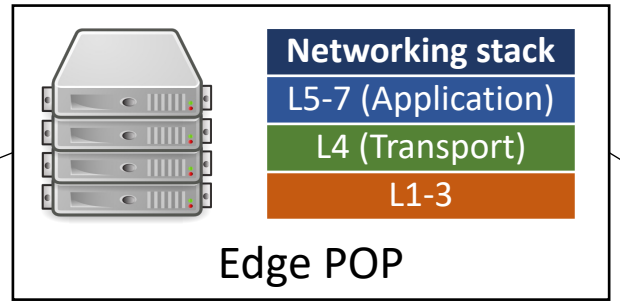
Goal: maximize delivery performance

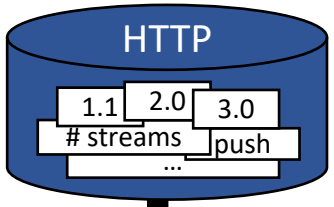
HTTP=2.0, #_streams=100, …
TCP=BBR, ICW=10, RTO=1s, …

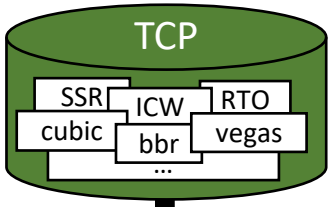
One set of configurations

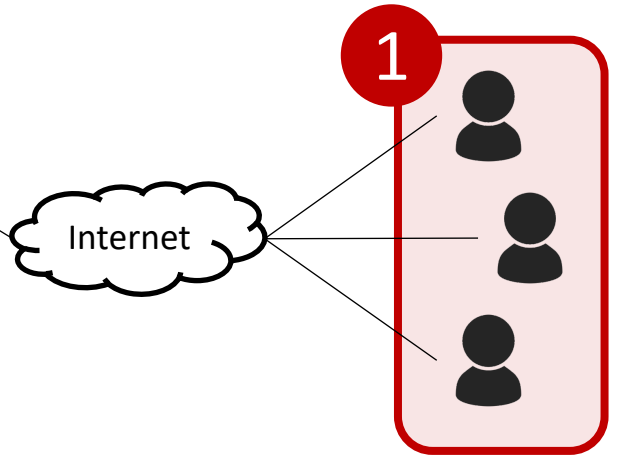
CDNs and Protocol Configurations at Edge



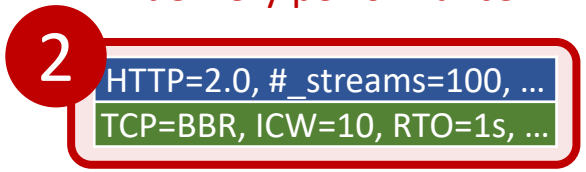








Goal: maximize delivery performance

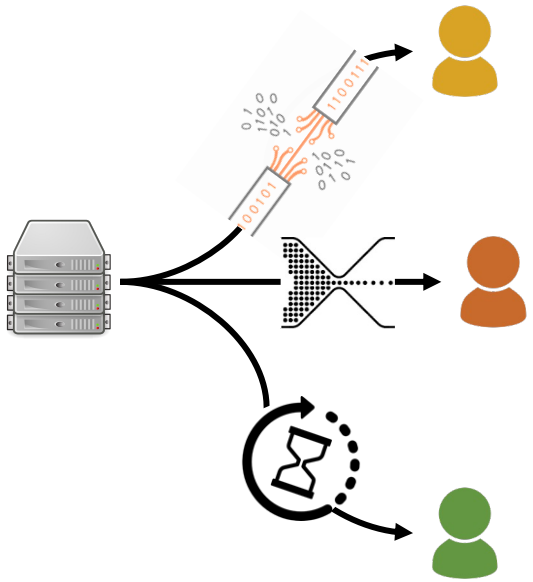


One set of configurations

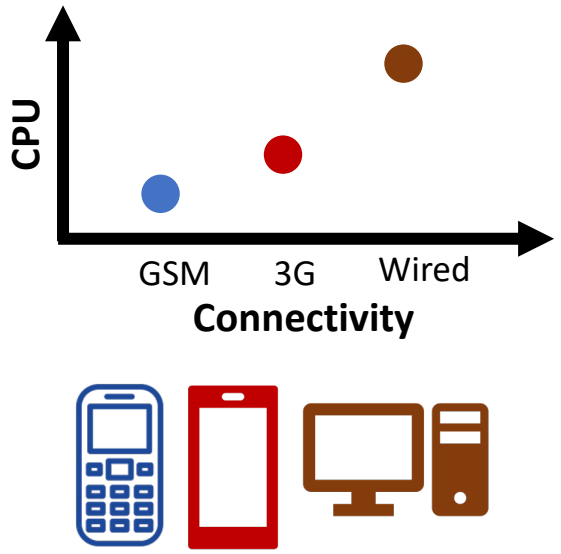
Two problems:

1. Are users homogeneous?
2. Is “one-size-fits-all” optimal?

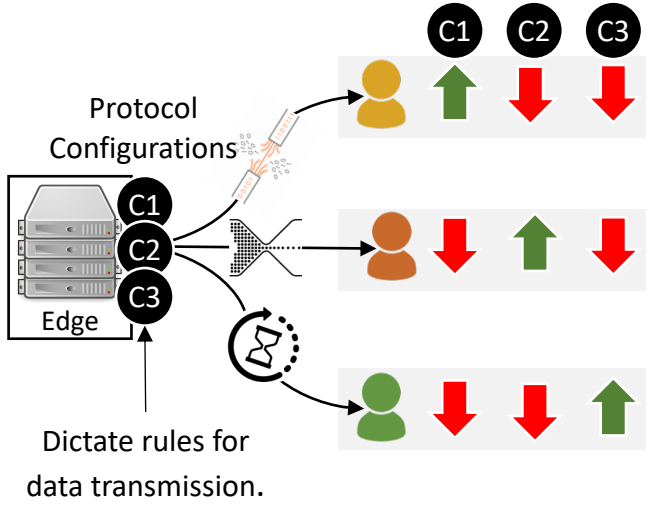
User Heterogeneity & Performance Sensitivity.



Different congestion control models (delay/loss/bottleneck-bw) [Yan et al. ATC'18]



Impact of device capabilities [Ahmad et al. IMC'16]



Performance Sensitivity of protocols for diverse networks

If ”one-size-fits-all” approach is sub-optimal…

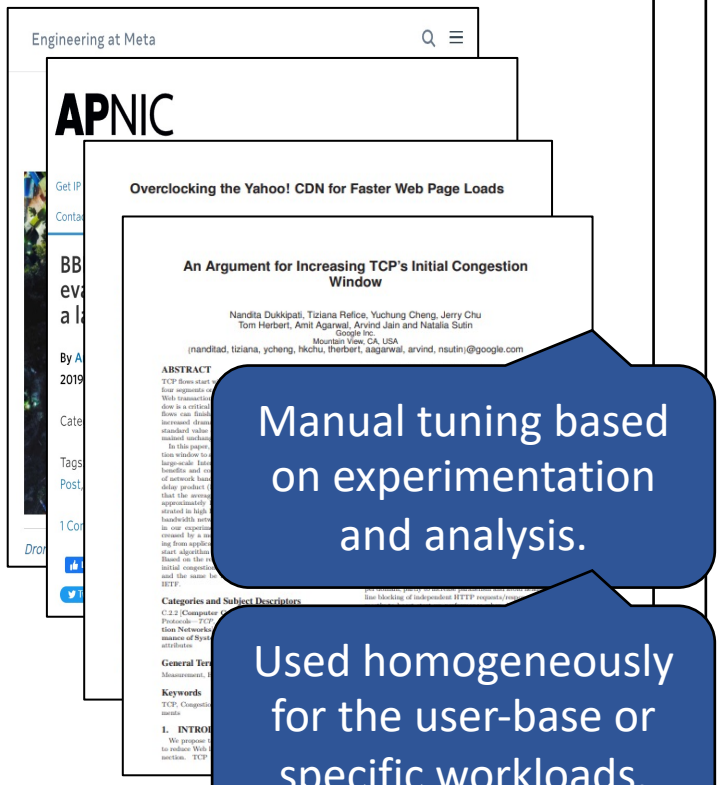
How to dynamically tune the networking configurations to maximize performance for the diverse connections?

Traditional Approaches for Selecting Configurations

Default

```
net.ipv4.tcp_congestion_control = cubic
net.ipv4.tcp_low_latency = 0
net.ipv4.tcp_autocorking = 1
default via IP dev eth0 initcwnd 10
Protocols http/1.1
```

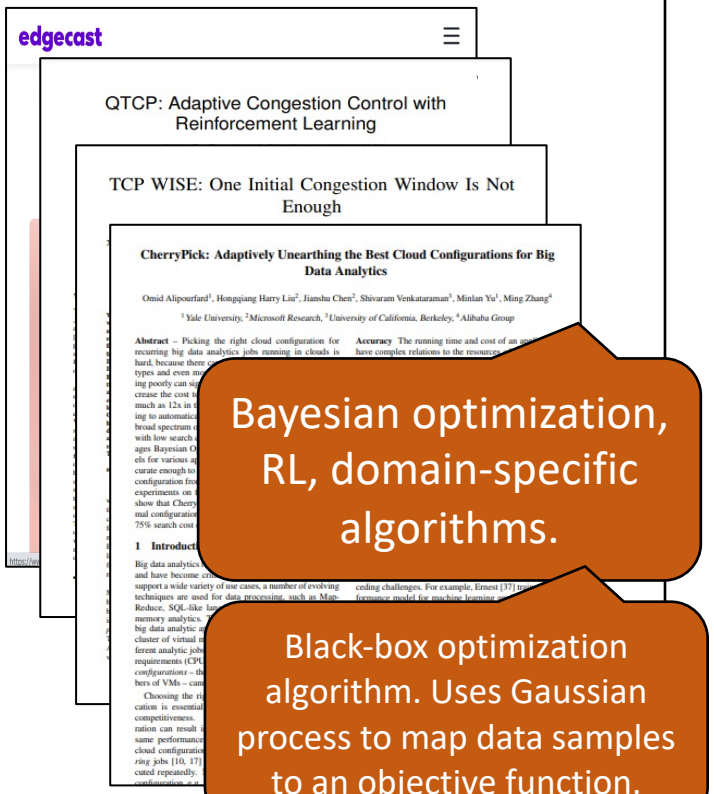
HandPicked



Manual tuning based on experimentation and analysis.

Used homogeneously for the user-base or specific workloads.

Dynamic



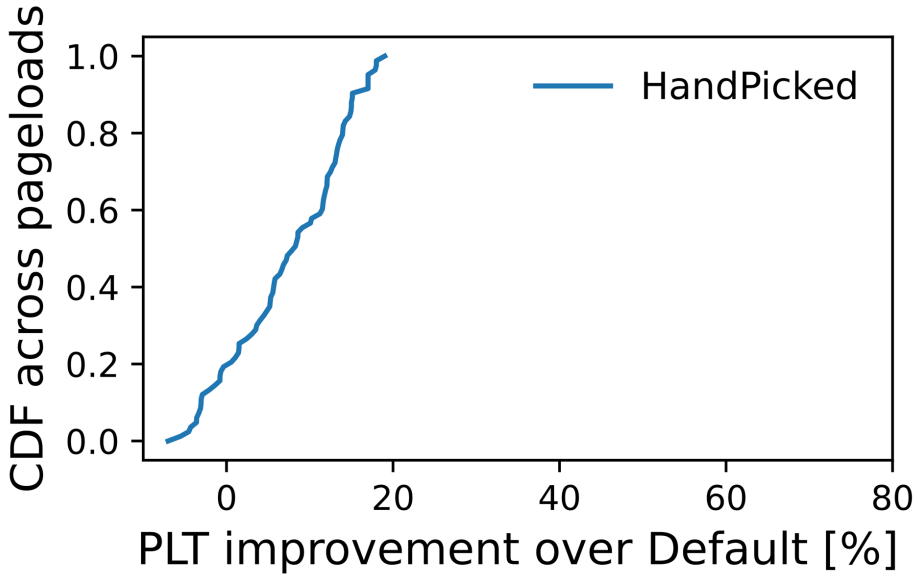
Bayesian optimization, RL, domain-specific algorithms.

Black-box optimization algorithm. Uses Gaussian process to map data samples to an objective function.

Are Traditional Approaches Optimal?

- Emulated diverse traces from a CDN in local testbed.
- Brute-force exploration of TCP and HTTP configuration space.
- *Oracle:* Selects optimal configurations that minimizes page load time (PLT).

Pre-computed configs. statically applied to diverse connections.

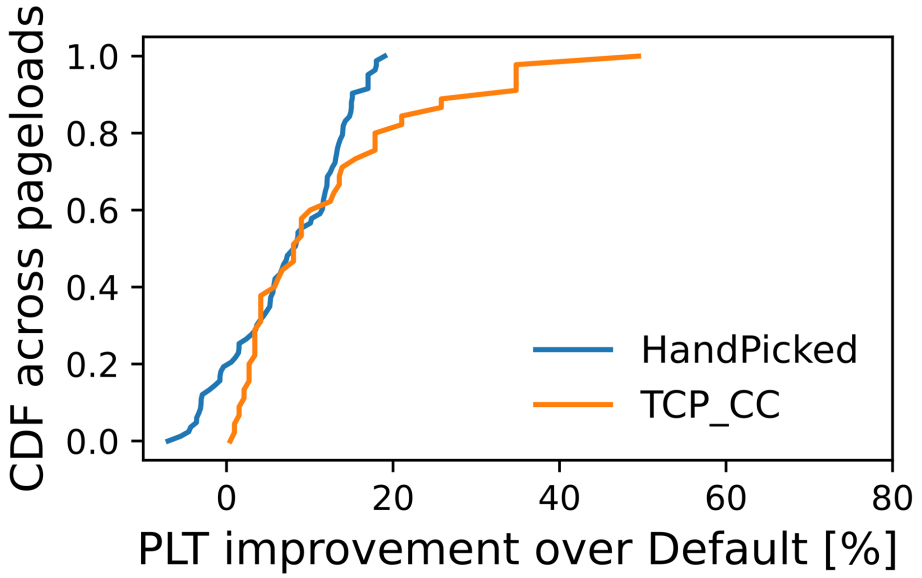


Are Traditional Approaches Optimal?

- Emulated diverse traces from a CDN in local testbed.
- Brute-force exploration of TCP and HTTP configuration space.
- *Oracle:* Selects optimal configurations that minimizes page load time (PLT).

Single configuration dynamically tuned for the connections.

Pre-computed configs. statically applied to diverse connections.

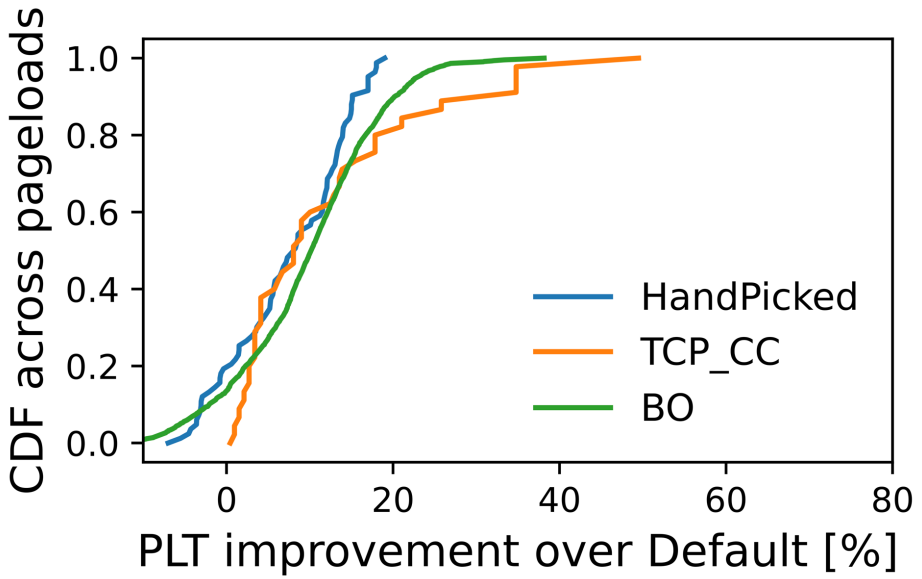


Are Traditional Approaches Optimal?

- Emulated diverse traces from a CDN in local testbed.
- Brute-force exploration of TCP and HTTP configuration space.
- *Oracle:* Selects optimal configurations that minimizes page load time (PLT).

Single configuration dynamically tuned for the connections.

Pre-computed configs. statically applied to diverse connections.



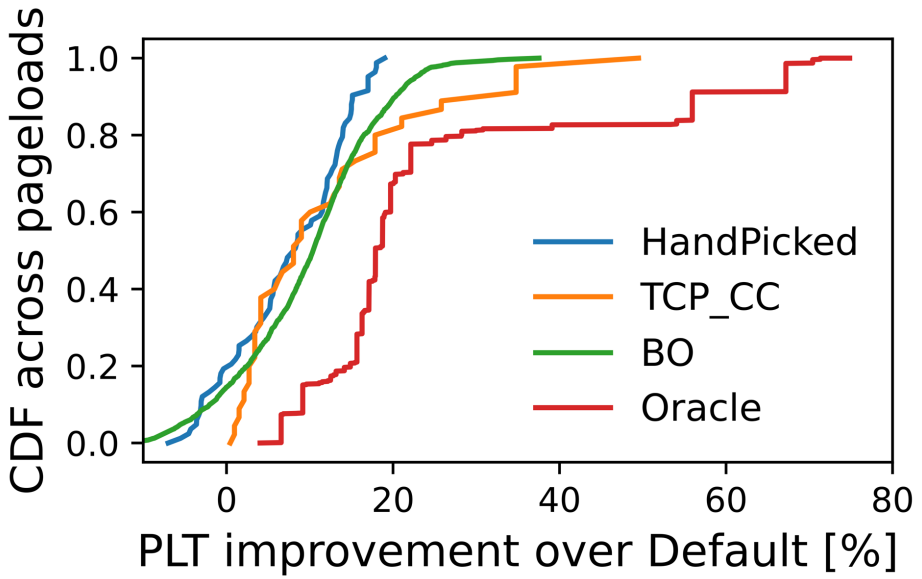
BO -> sub-optimal convergence, no cross-connection learning.

Are Traditional Approaches Optimal?

- Emulated diverse traces from a CDN in local testbed.
- Brute-force exploration of TCP and HTTP configuration space.
- *Oracle:* Selects optimal configurations that minimizes page load time (PLT).

Single configuration dynamically tuned for the connections.

Pre-computed configs. statically applied to diverse connections.



BO -> sub-optimal convergence, no cross-connection learning.

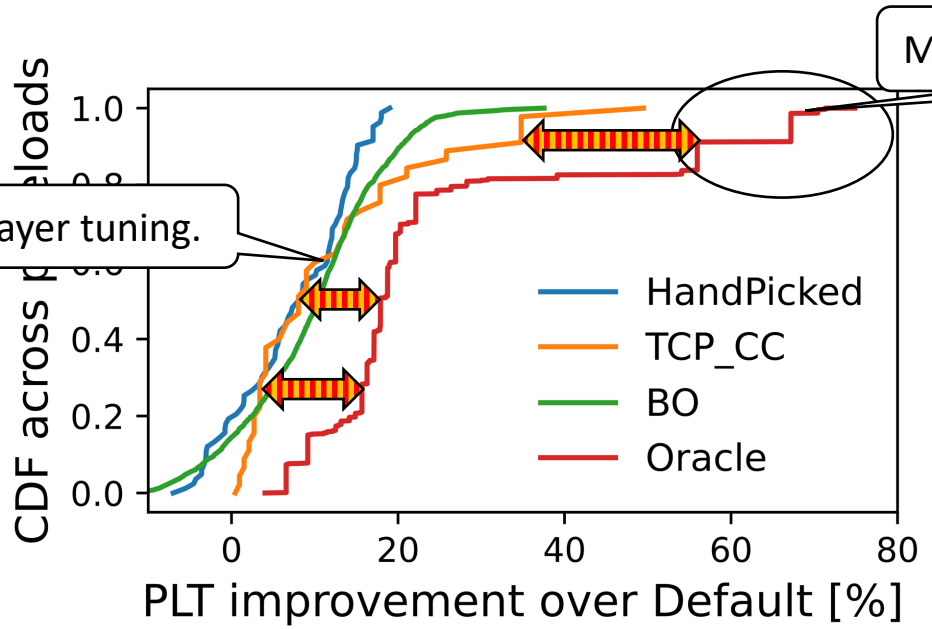
Per-connection optimal configs. -> significant performance gains.

Are Traditional Approaches Optimal?

- Emulated diverse traces from a CDN in local testbed.
- Brute-force exploration of TCP and HTTP configuration space.
- *Oracle:* Selects optimal configurations that minimizes page load time (PLT).

Single configuration dynamically tuned for the conne

Pre-computed configs. statically applied to diverse connections.



BO -> sub-optimal convergence, no cross-connection learning.

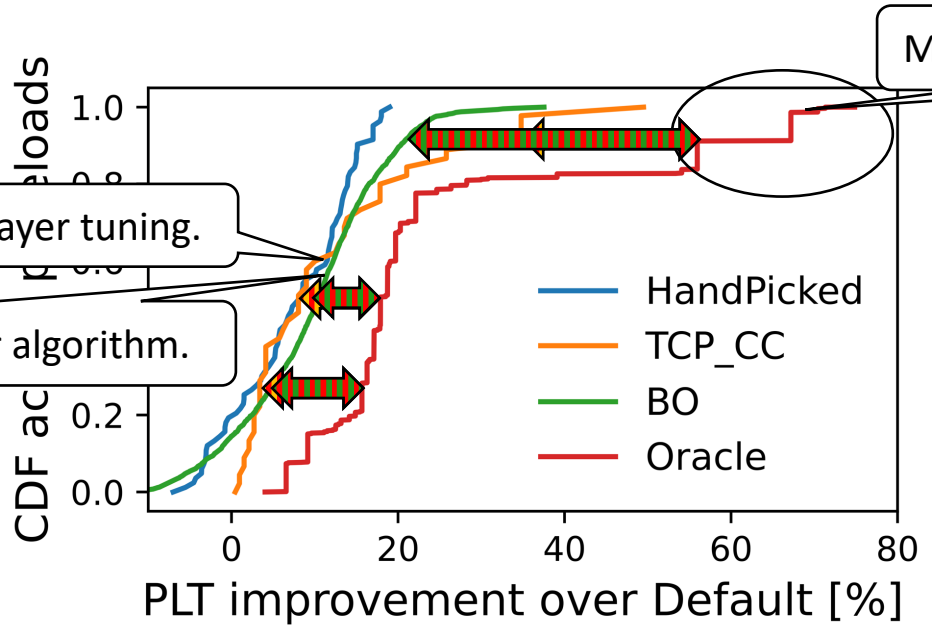
Per-connection optimal configs. -> significant performance gains.

Are Traditional Approaches Optimal?

- Emulated diverse traces from a CDN in local testbed.
- Brute-force exploration of TCP and HTTP configuration space.
- *Oracle:* Selects optimal configurations that minimizes page load time (PLT).

Single configuration dynamically tuned for the conne

Pre-computed configs. statically applied to diverse connections.

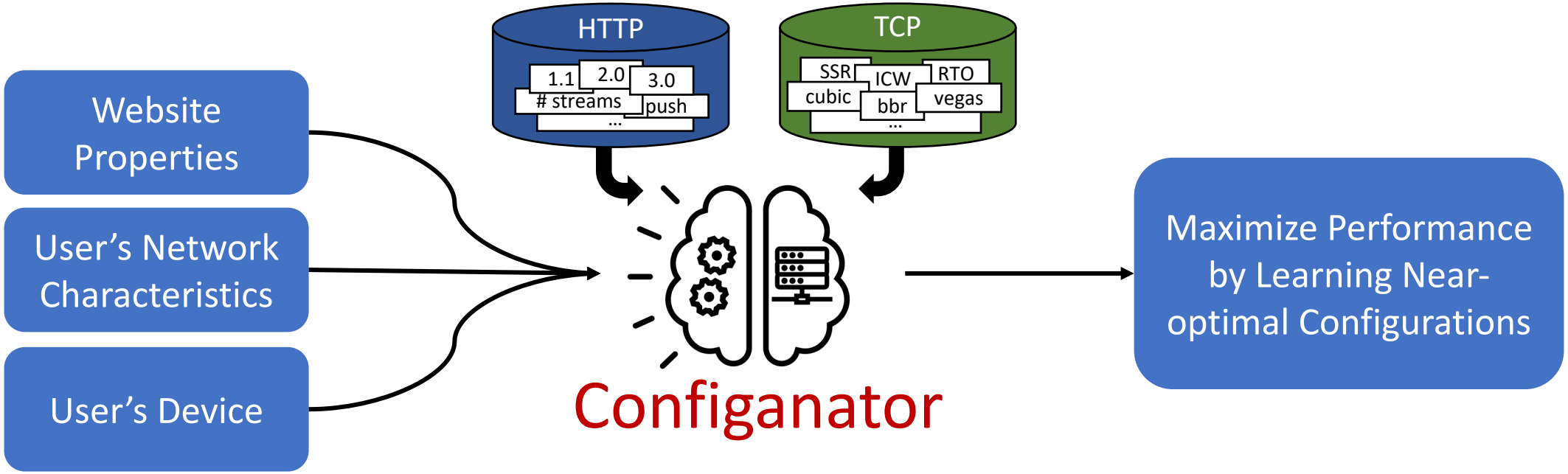


BO -> sub-optimal convergence, no cross-connection learning.

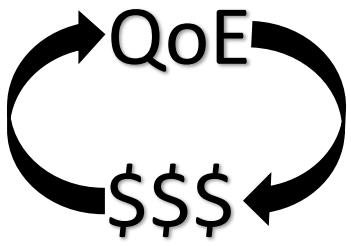
Per-connection optimal configs. -> significant performance gains.

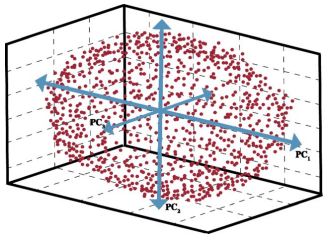
Configanator

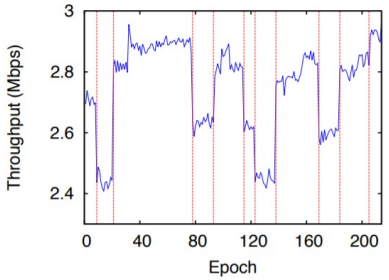
Optimizes web performance by systematically reconfiguring network stack in a principled manner.

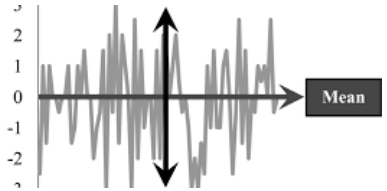


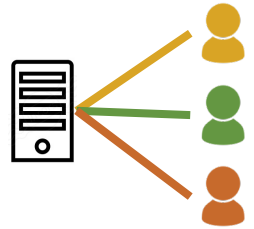
Challenges with Configuration Tuning











<u>QoE Cost</u>
Bad configurations hurts revenue.

<u>High dimensionality</u>
Connection types, devices & websites.

<u>Network dynamics</u>
Network & performance changes over time.

<u>Noise & variability</u>
Inherent noise in performance signals.

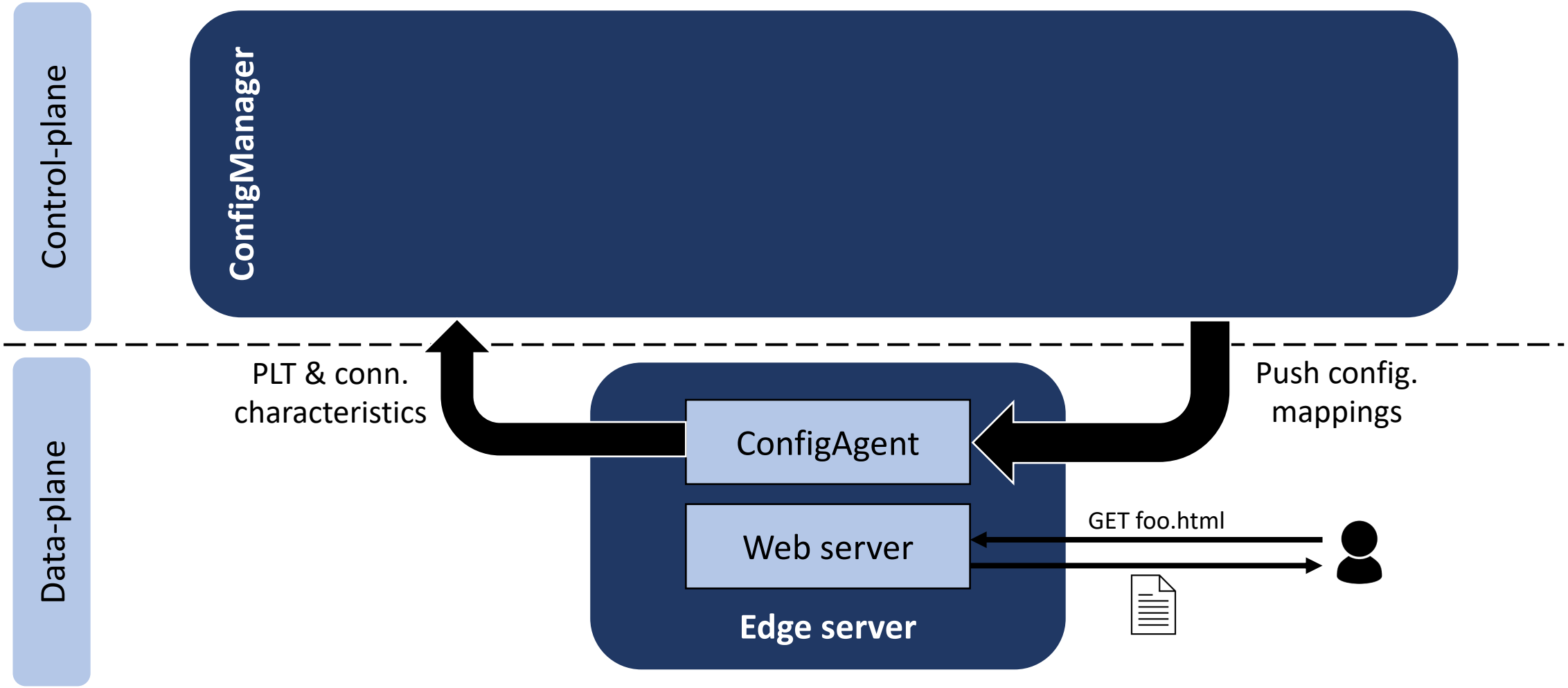
<u>System limitations</u>
Low-overhead, fine- grained tuning.

Algorithm design

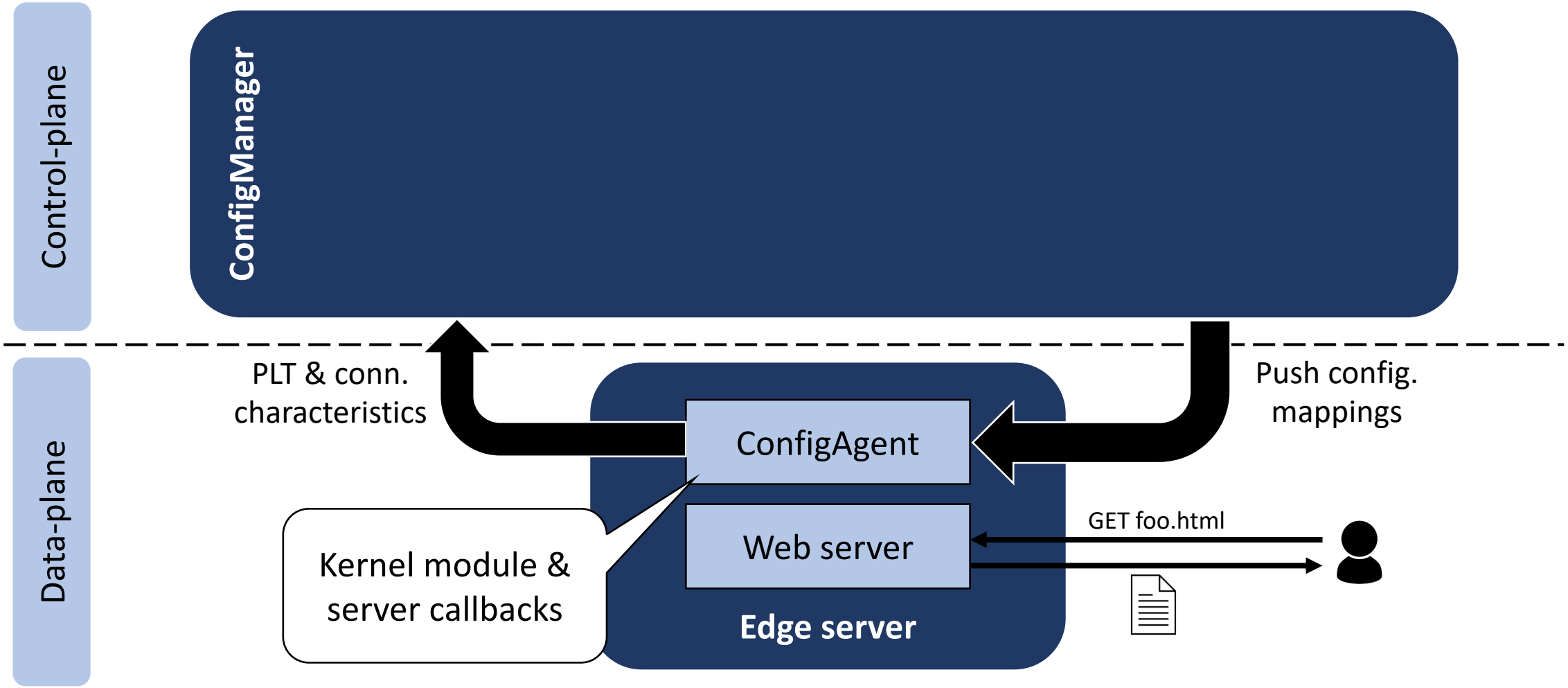
System design

Network dynamics figure credits: Zahaib Akhtar

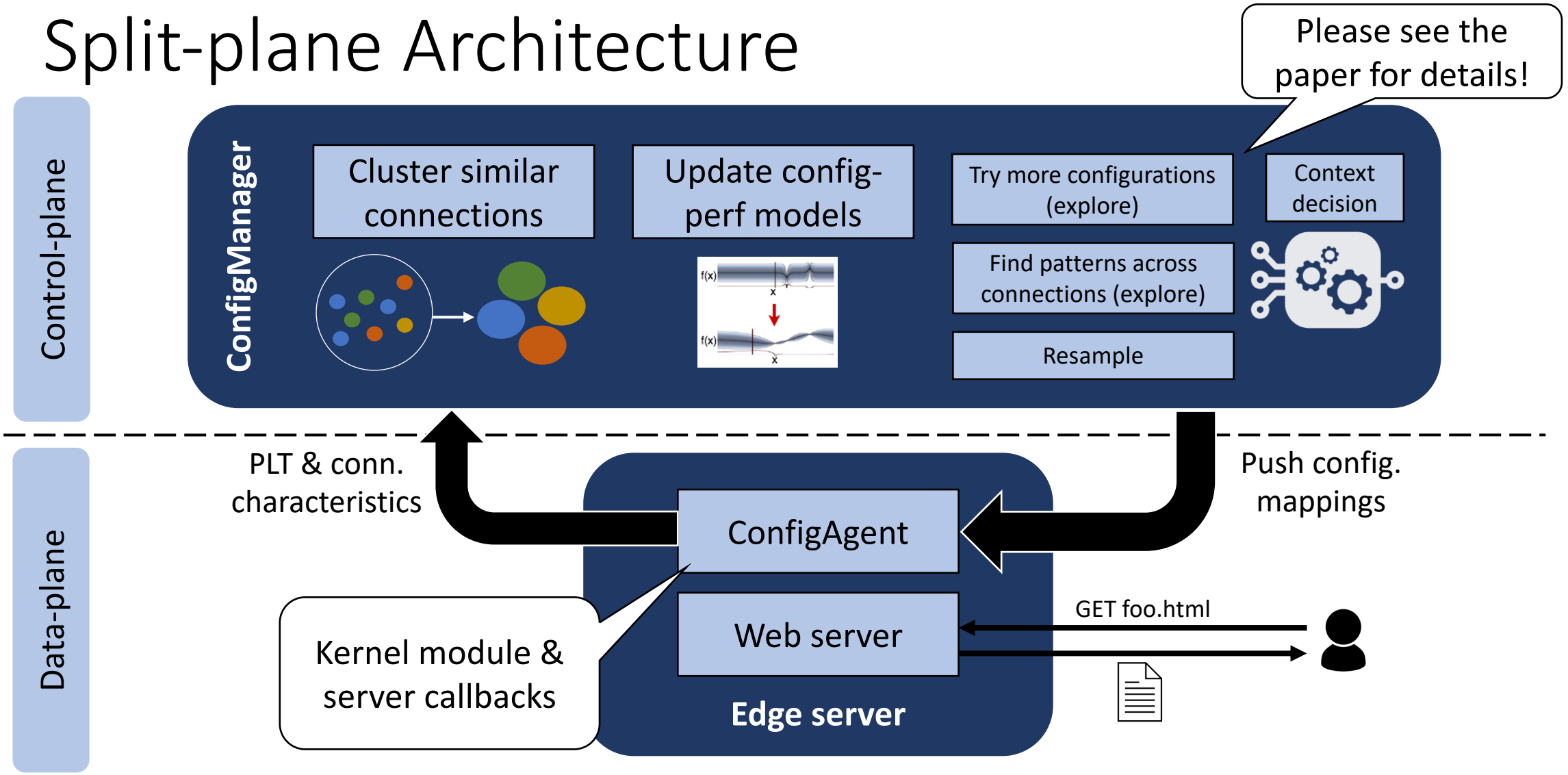
Split-plane Architecture



Split-plane Architecture



Split-plane Architecture

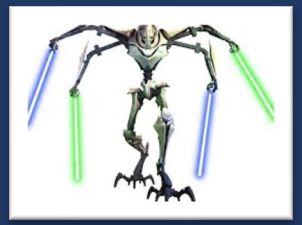


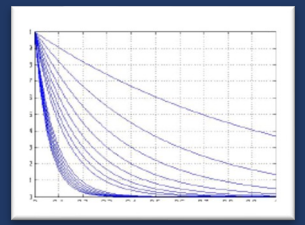
Evaluation

nsdi '22 ATTEND PROGRAM PARTICIPATE SPONSORS ABOUT

Configanator: A Data-driven Approach to Improving CDN Performance.

Authors:
Usama Naseer and Theophilus A. Benson, *Brown University*



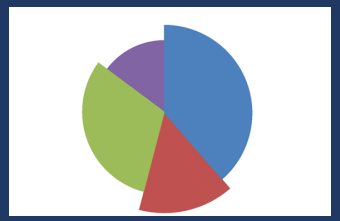


Algorithm design (benefits of arms, convergence, NCs)

Performance improvements.

Fairness implications.

System overheads.

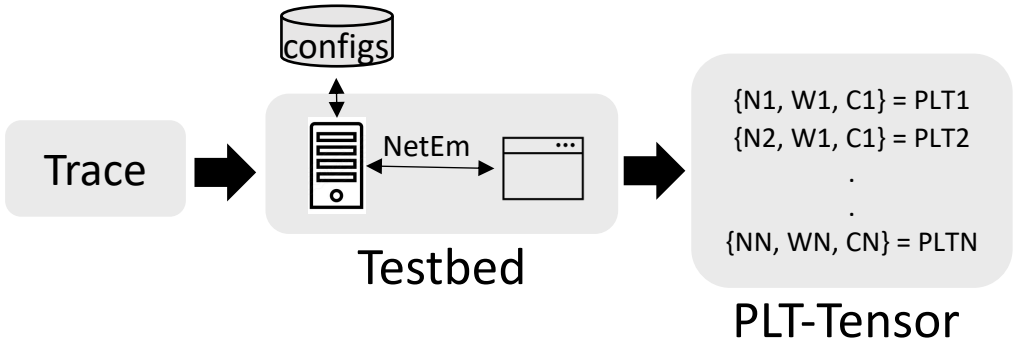


Dissection of improvements, deployment considerations.

Evaluation Setup

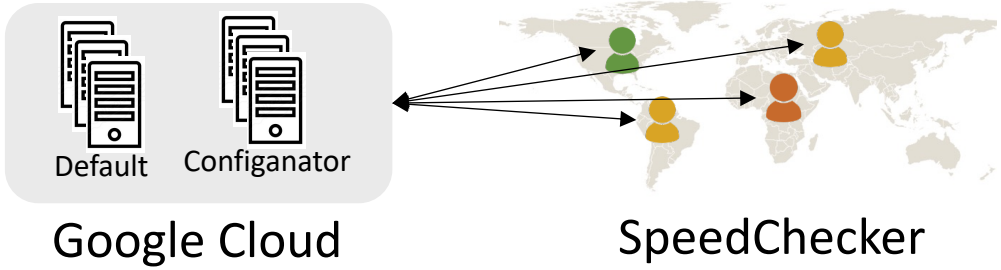
Trace-driven simulation

Traces from multiple regions.
- US (CAIDA, FCC)
- Japan (MAWI)
- Global (CDN trace, Pantheon)

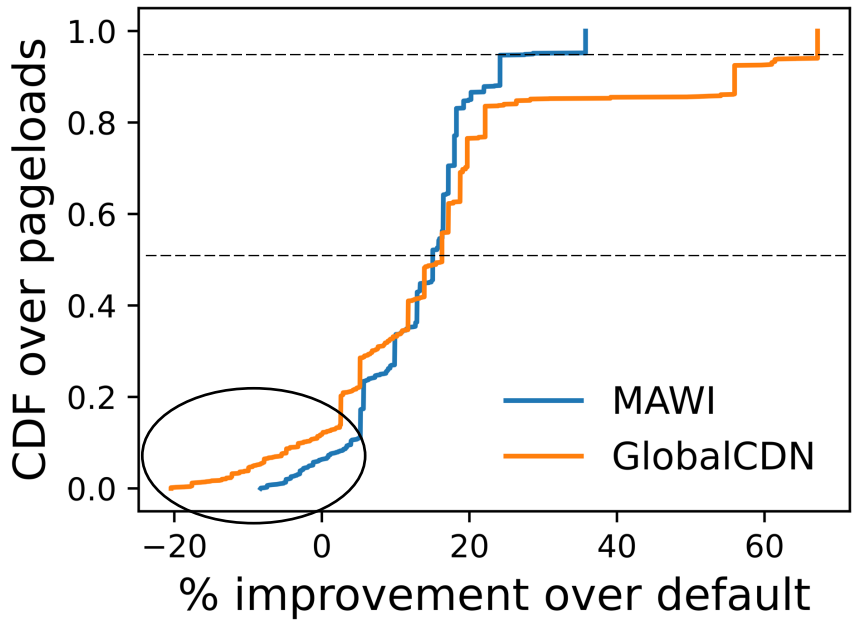


Live deployment

Experiments in-the-wild.
- Servers (Google cloud in US)
- Users (spread across globe)



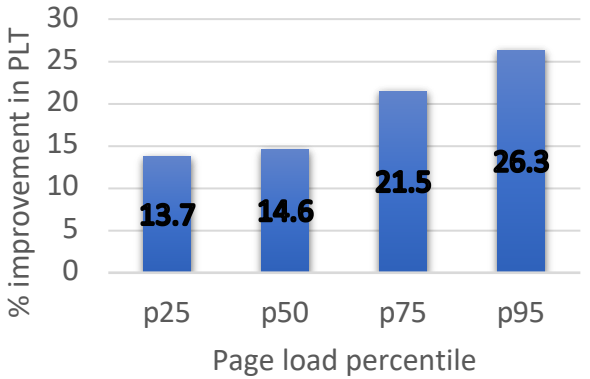
Performance Improvements



36-67% improvement.
Low BW, low-high RTT/loss networks, content-rich sites.

15-17% improvement

Live deployment



Conclusion + Future Work

Customizable or generalizable protocols

System/algorithm to tackle diversity.

Contextual MAB

Cross-layer protocols

Broader configurations and flexibility.

QUIC

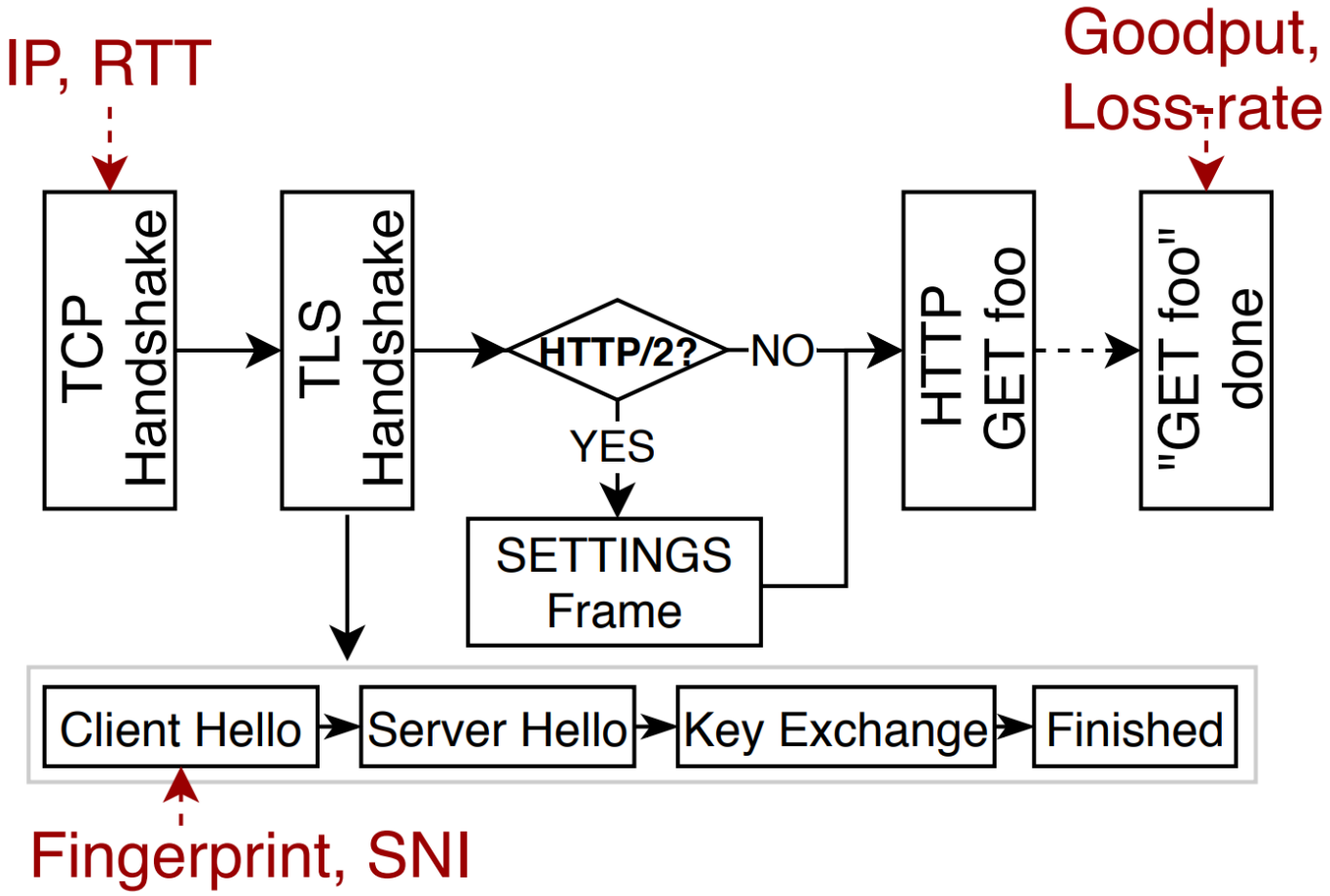
Transport loss

Web priorities

Infrastructure support

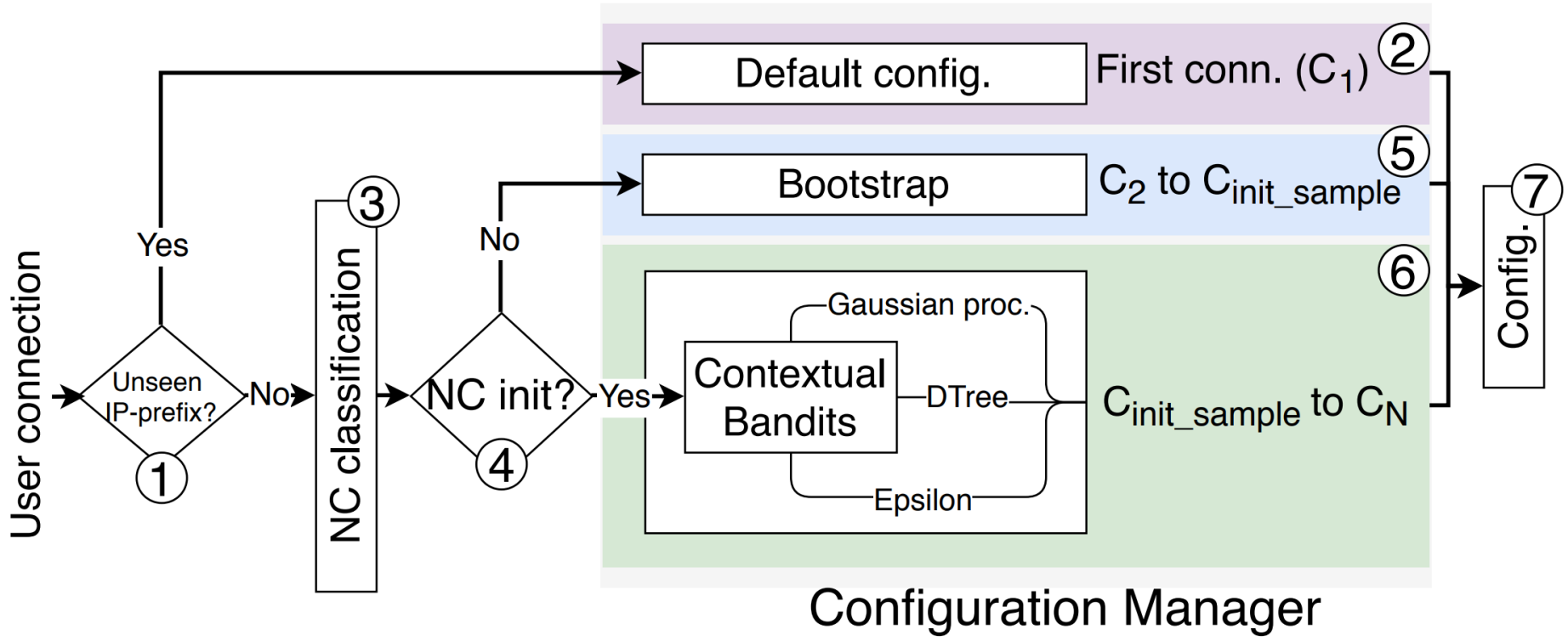
END

Connection Features



Context Decision

- Bayesian optimization's Expected Improvement
  - Switch to exploitation arm if IE < threshold

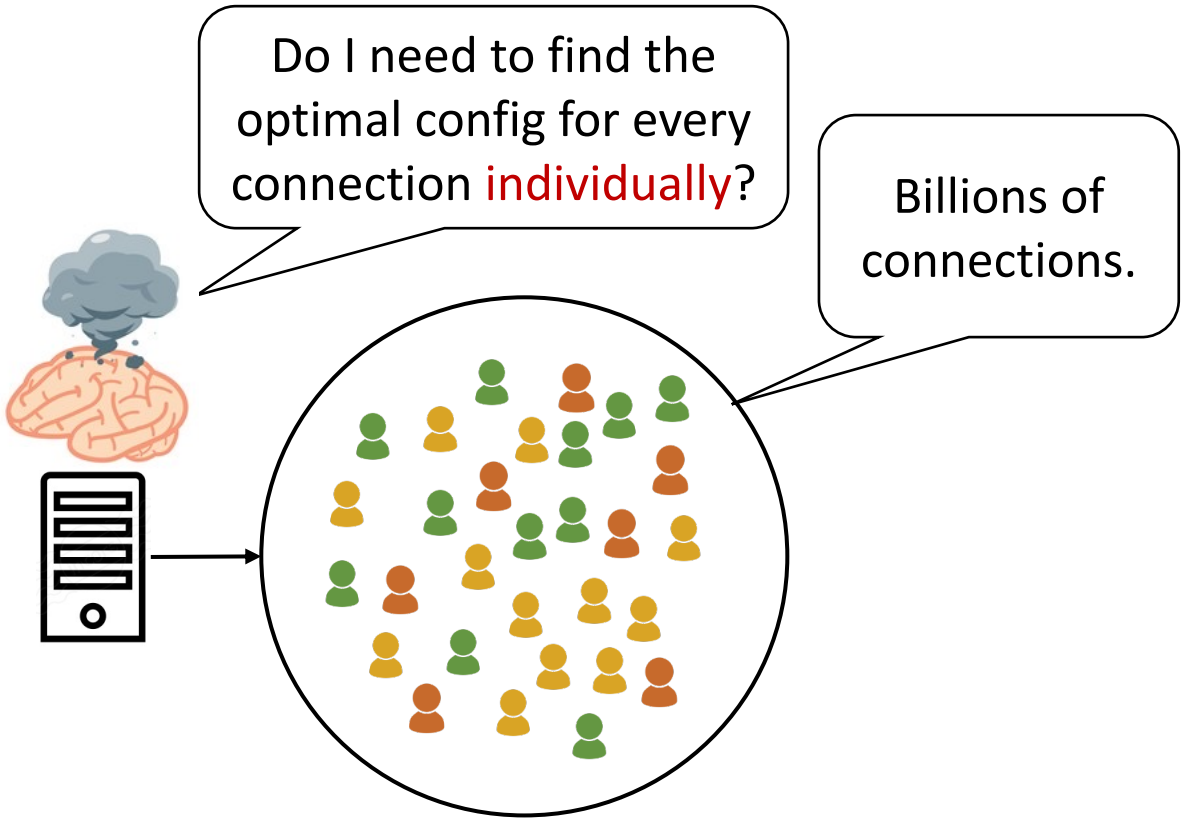


Algorithmic Design

Tuning and exploration granularity?

Coarse-granularity -> ASN, POP, prefixes?

- ✓ Amortizes QoE cost.
- ⊘ Fine-grained diversity.
- ⊘ High dimensionality.



Algorithmic Design

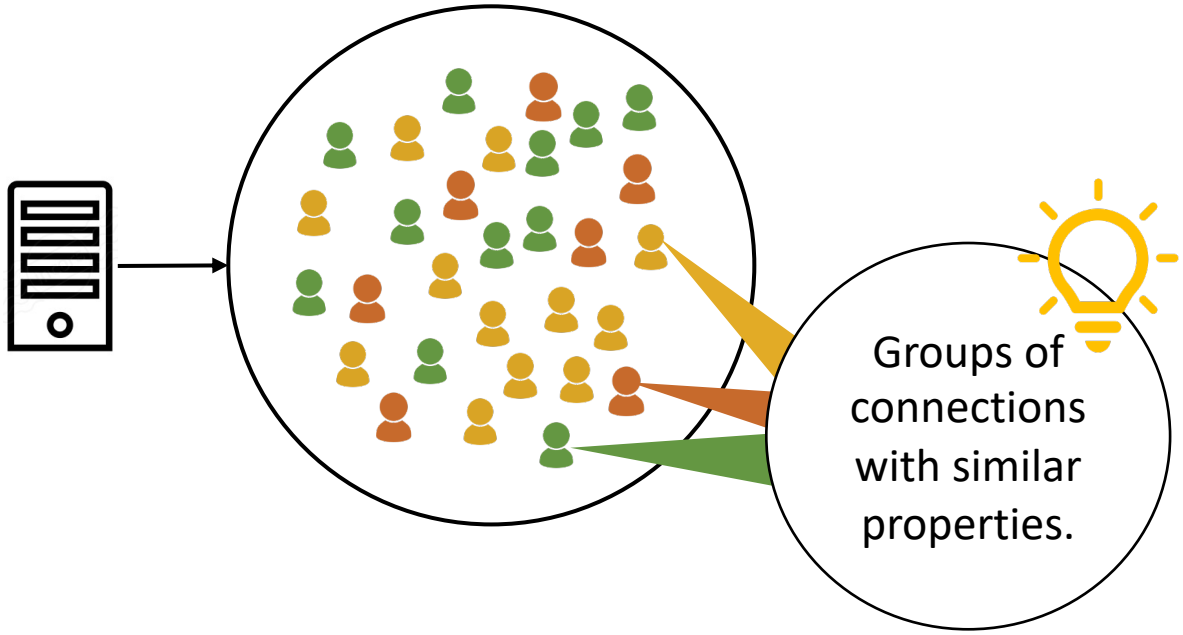
Tuning and exploration granularity?

Coarse-granularity -> ASN, POP, prefixes?

- ✓ Amortizes QoE cost.
- ⊘ Fine-grained diversity.
- ⊘ High dimensionality.

Network Classes (NC) -> connections with similar properties.

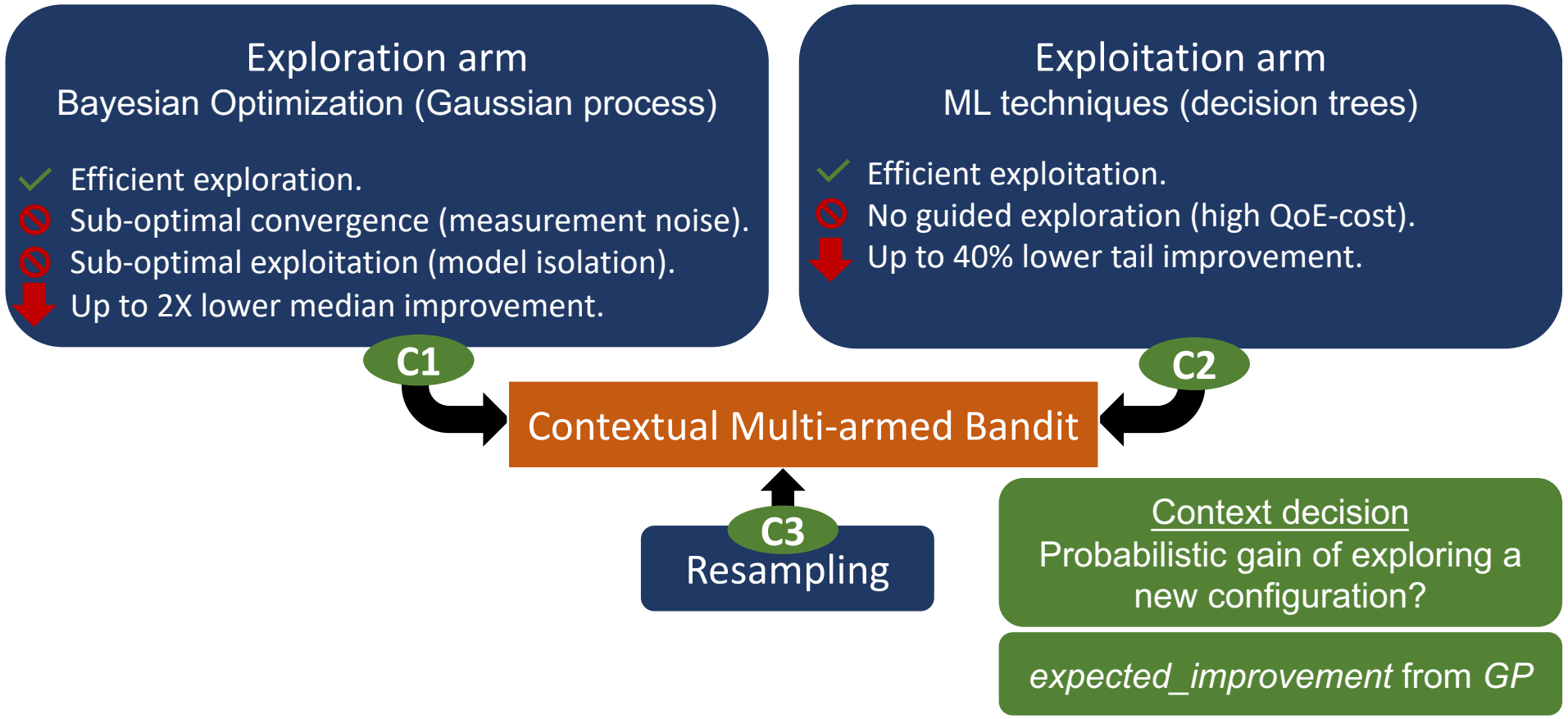
- ✓ Amortizes QoE cost.
- ✓ Fine-grained diversity.
- ✓ High dimensionality.



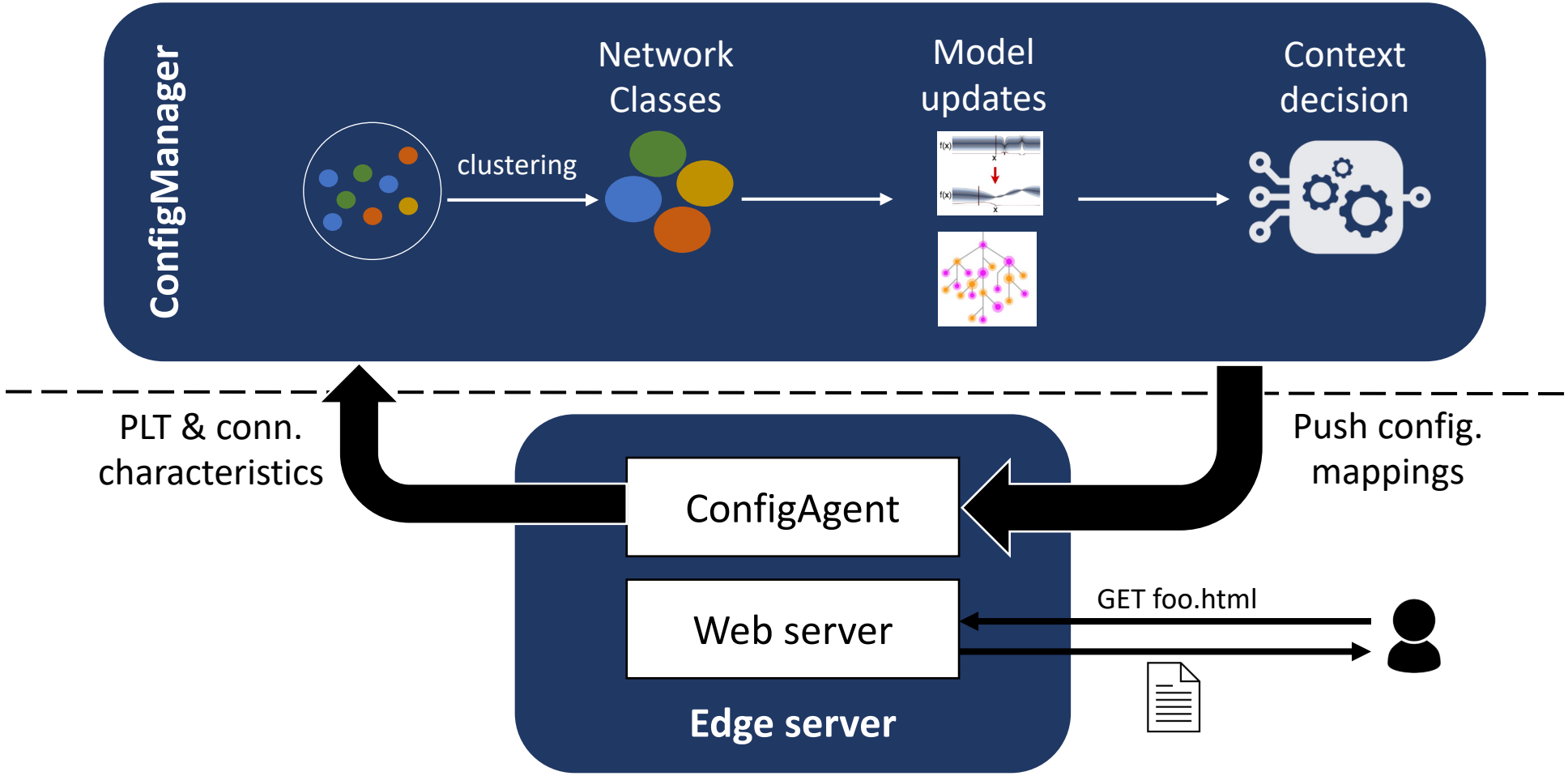
Algorithmic Design

Offline approach not representative of high dimensional Internet.

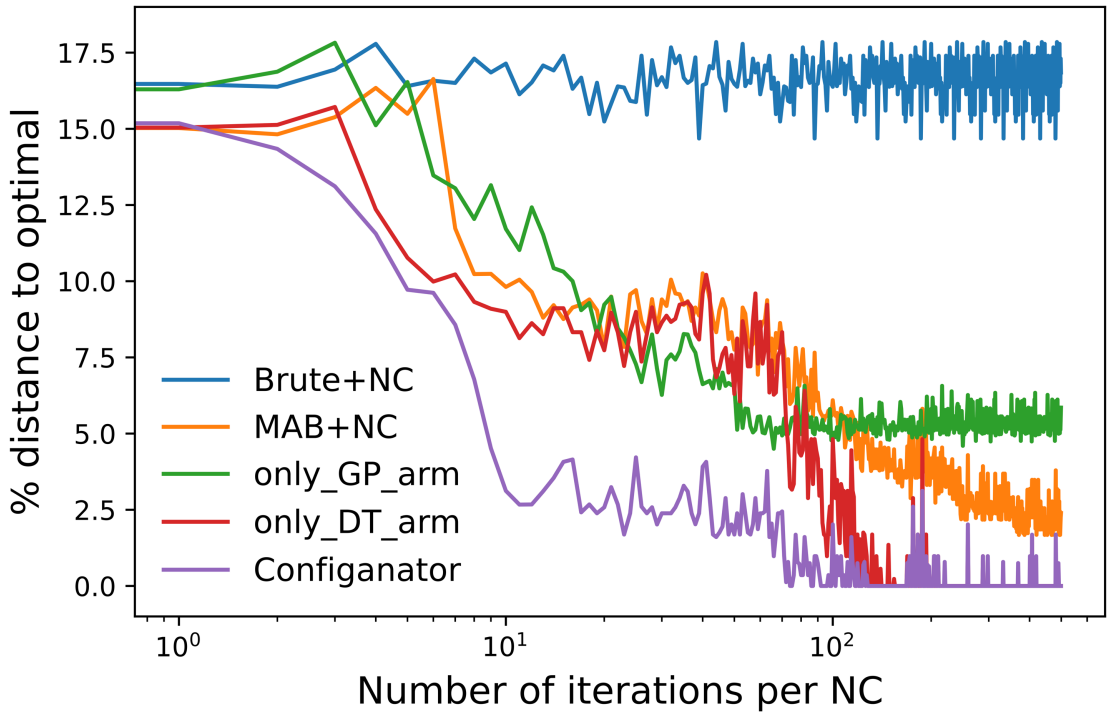
Build performance models for the high-dimensional web in a low-cost, online manner.

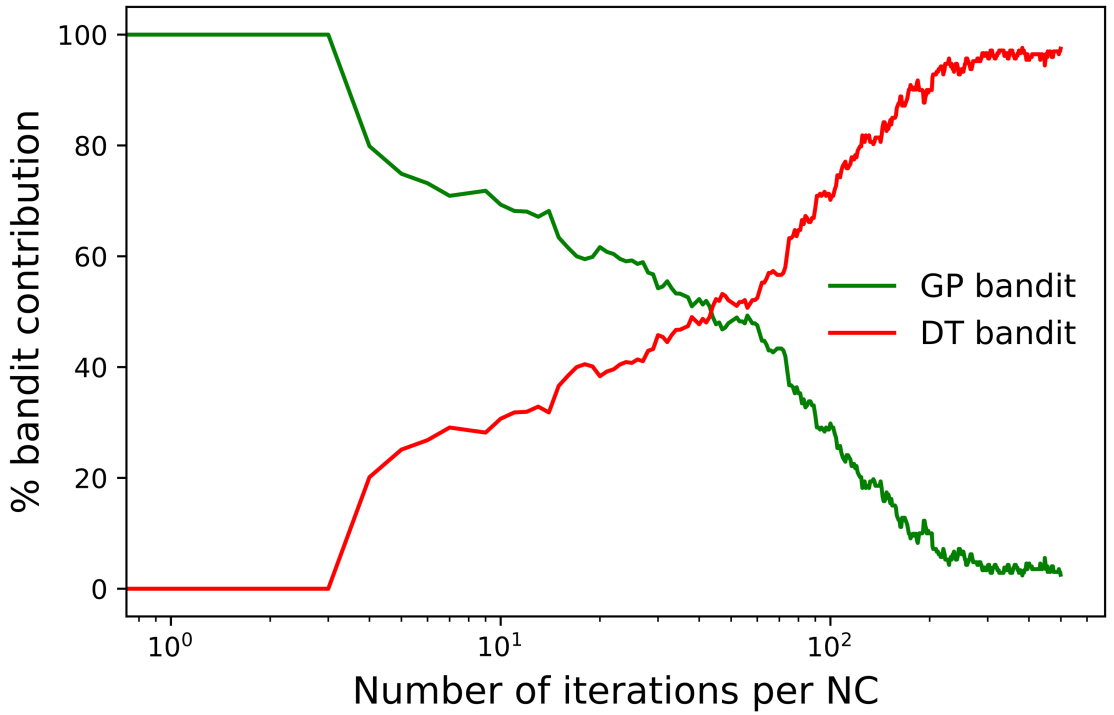


Split-plane Architecture



Convergence





MAB+NC: Multi-armed bandit with epsilon greedy reward.

Ablation analysis

Only_GP: Guided exploration, no cross-NC exploitation.

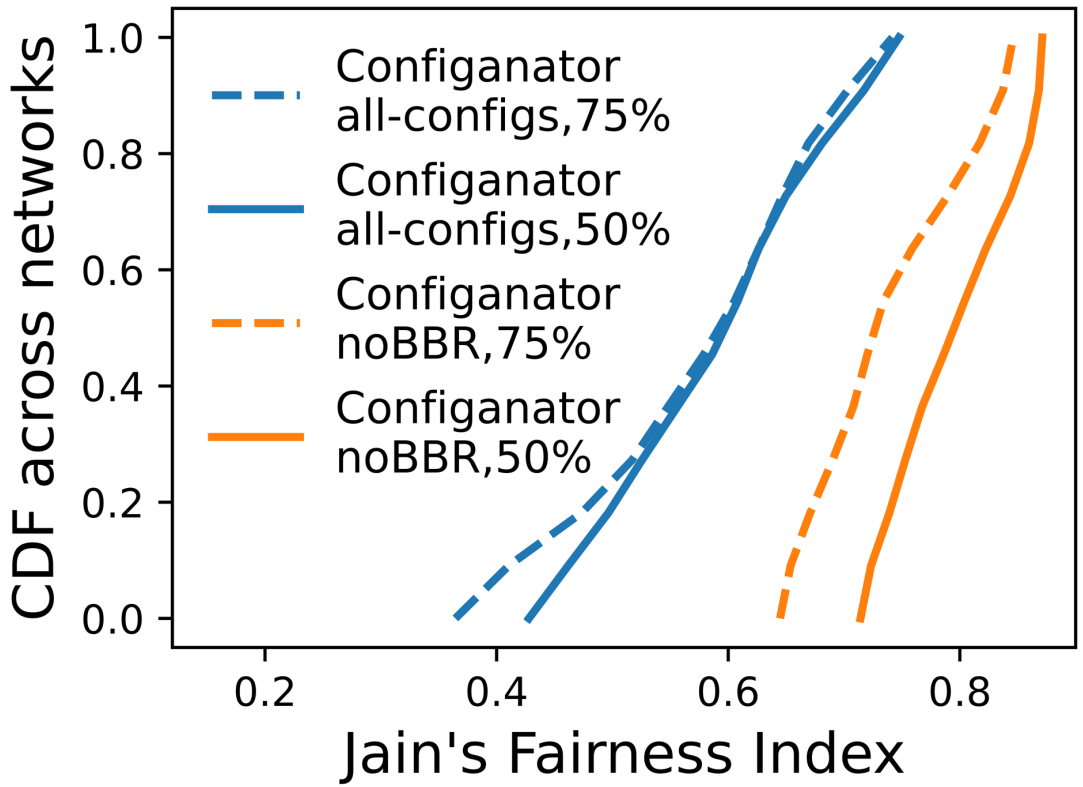
Only_DT: Random exploration, DT-based exploitation.

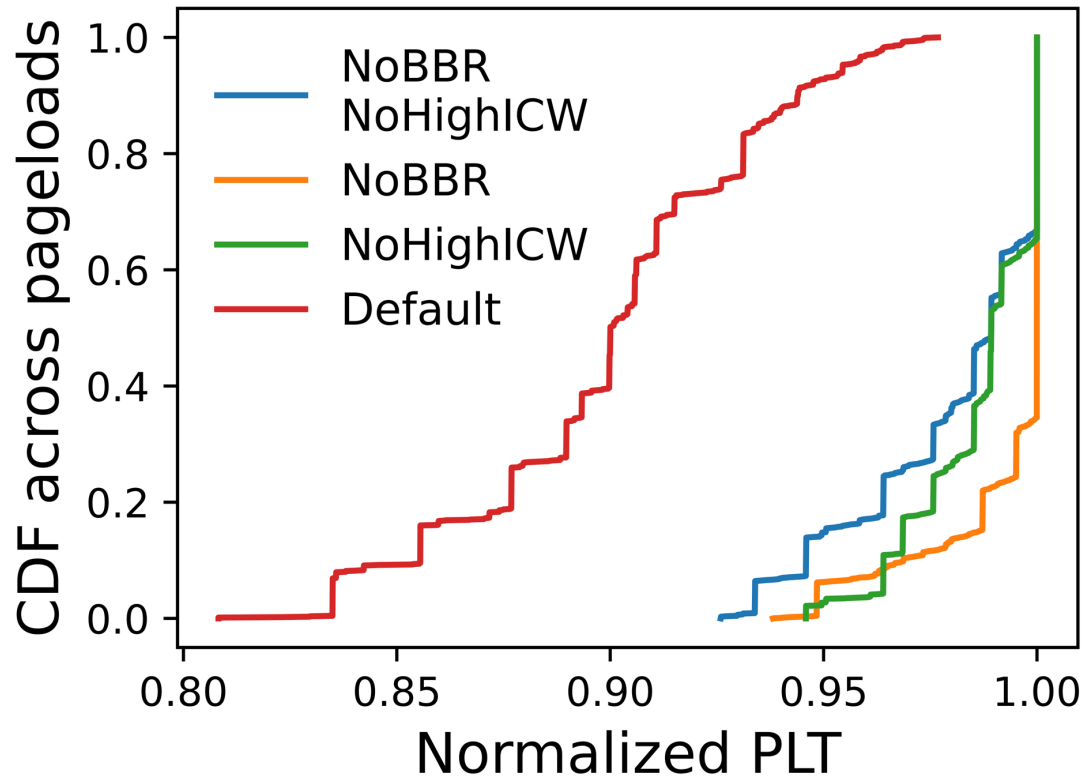
System Design

- Goals:
  - Flexible -> Can arbitrarily tune configuration per connection.
  - Low-overhead -> Minimal resource (CPU, RAM, Latency etc.) overhead.
  - Non-invasive -> Requires minimal changes to applications.

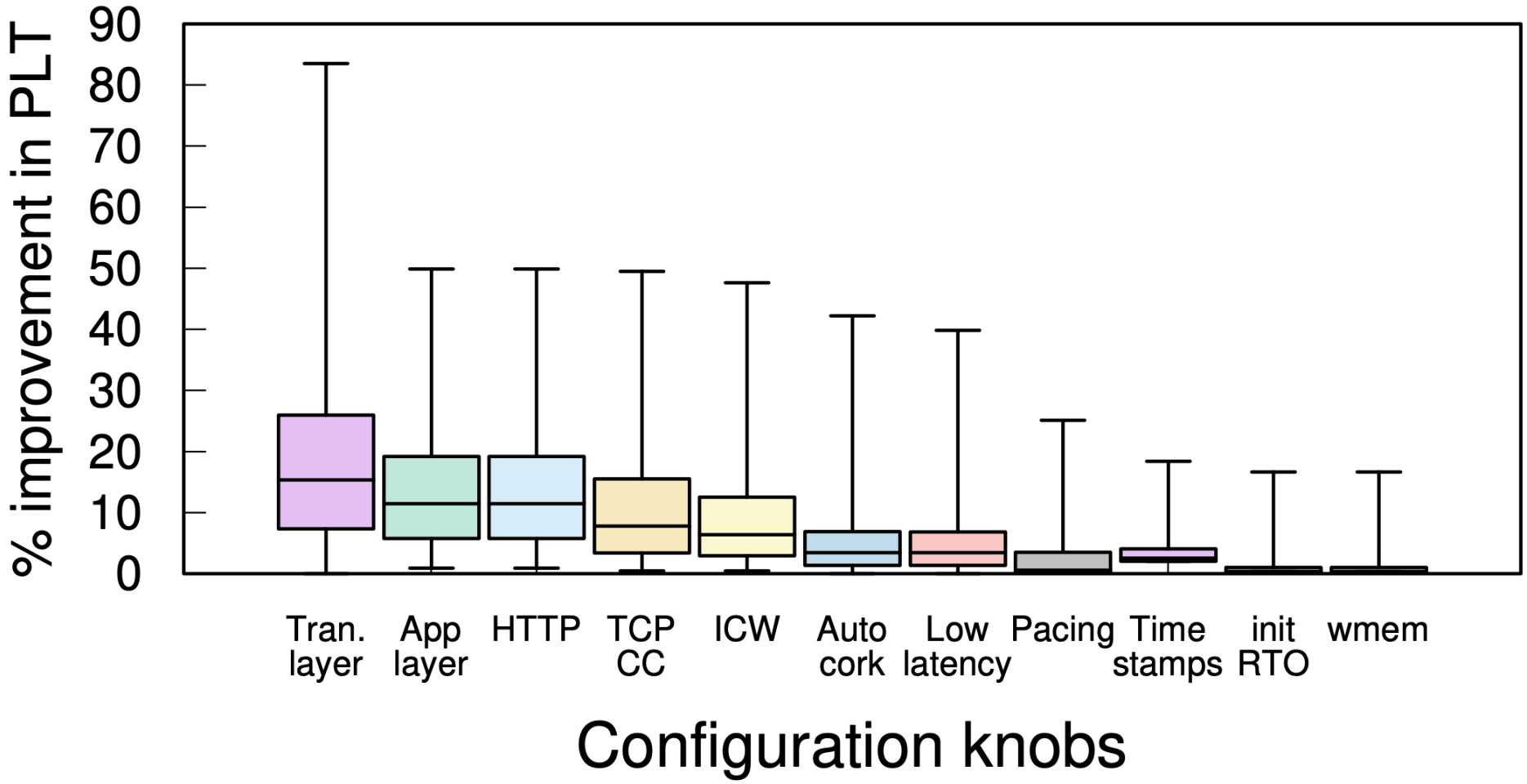
		Flexible	Low-overhead	Non-invasive
TCP	Sysctl / ip route	X	✓	✓
	VM / containers	✓	X	✓
	SetSockopt	✓	✓	X
	SetSockopt + LD_Preload	✓	X	✓
	eBPF	✓	✓	✓
	Kernel module	tcp_congestion_ops		
HTTP	Multiple instances	✓	X	✓
	Modify application code	OpenSSL ALPN callback, H2 settings callbacks		

Fairness Implications

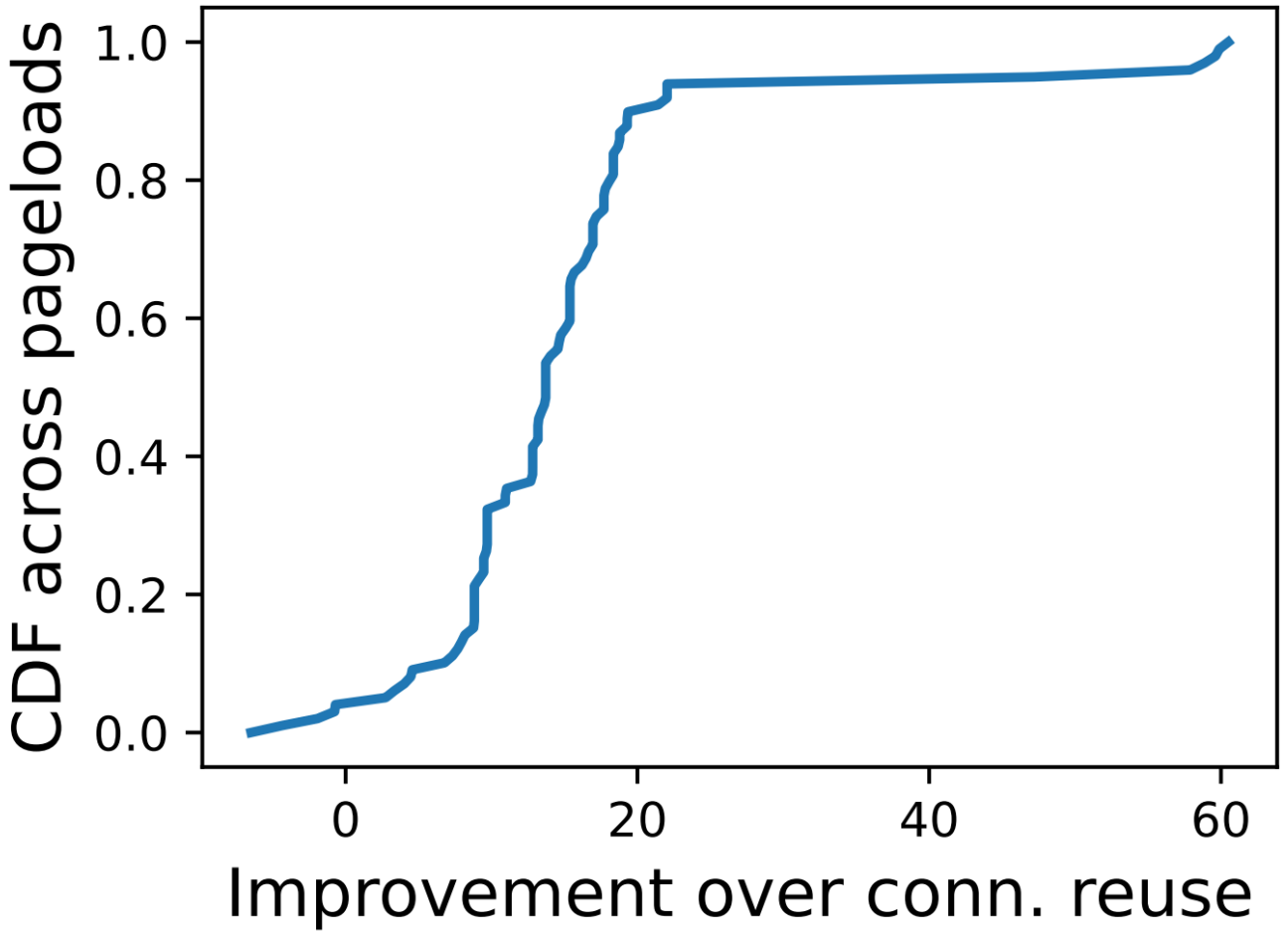




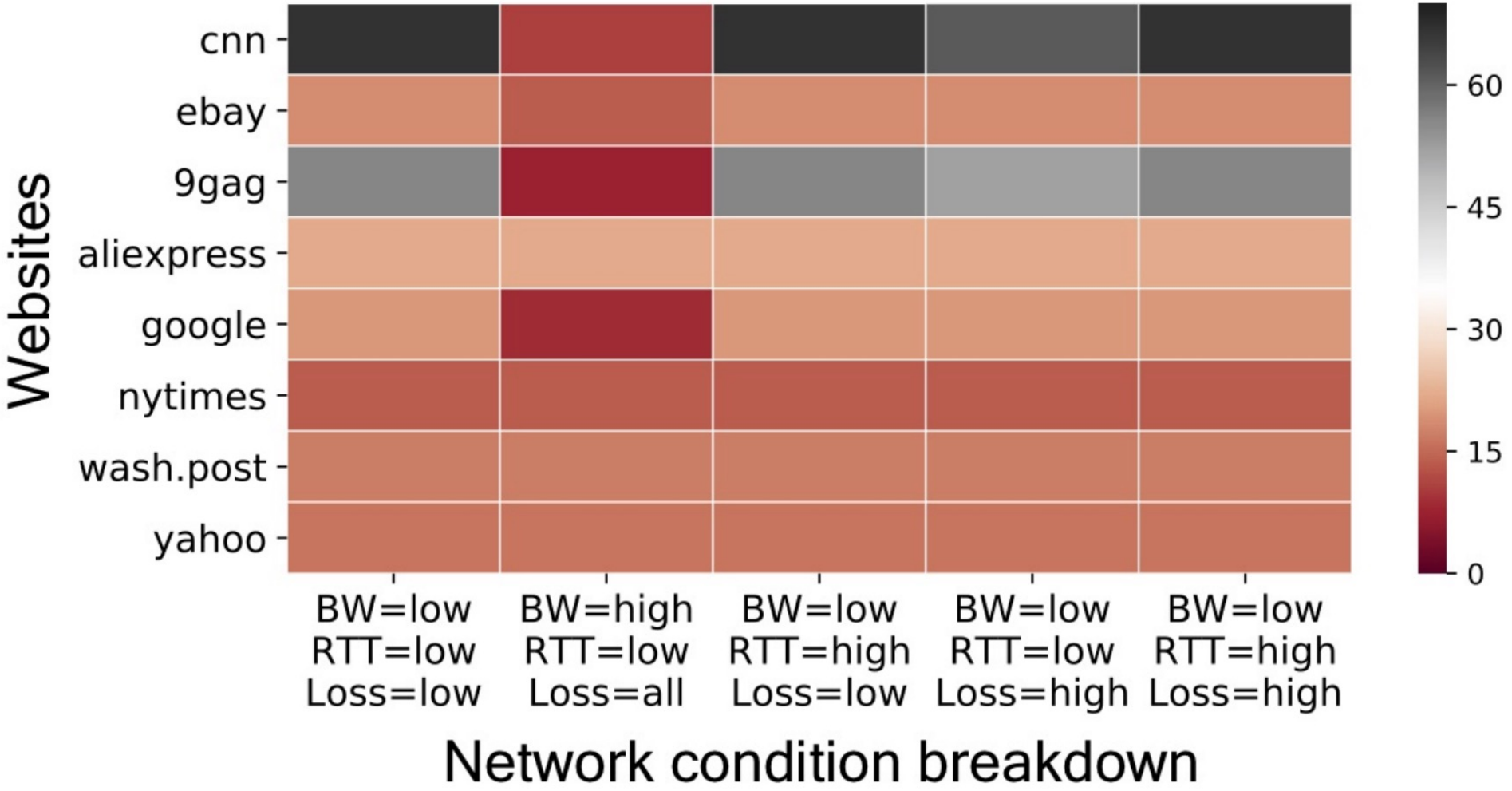
Impact of Different Knobs



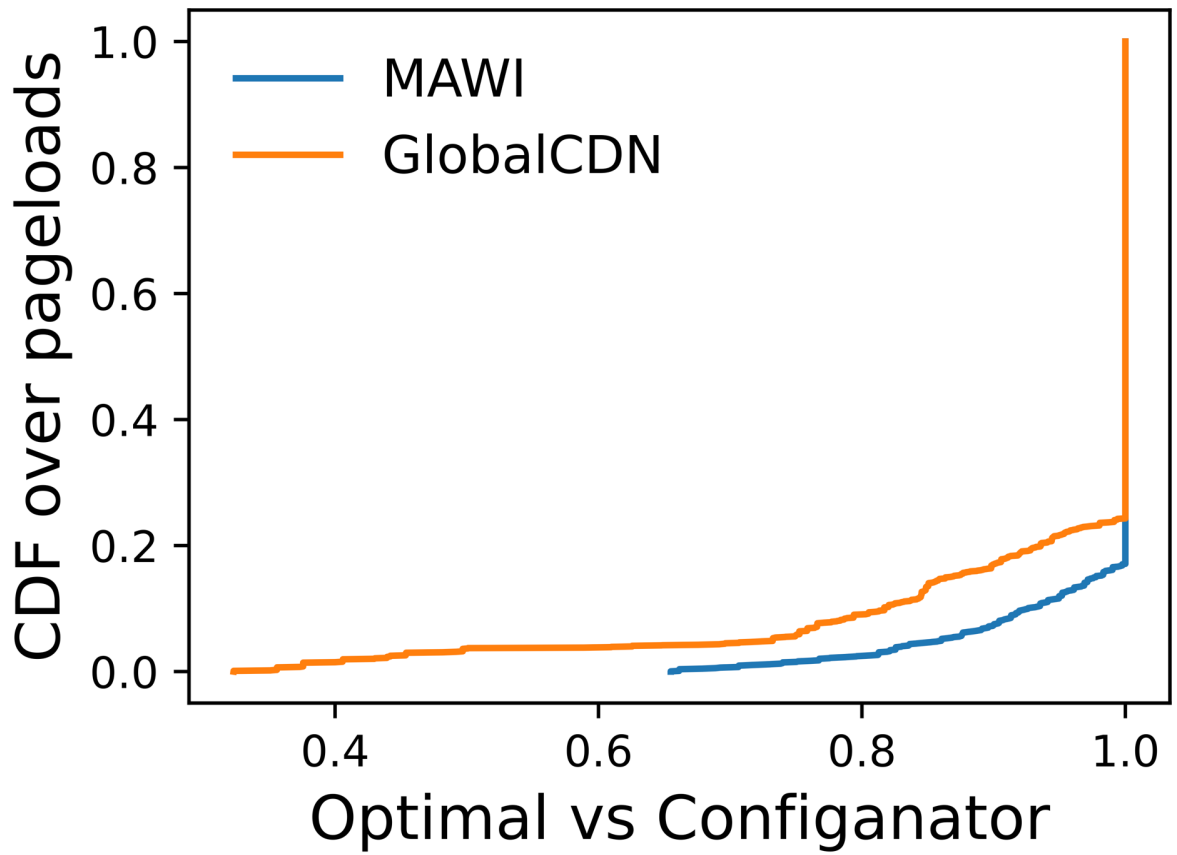
TCP Connection Reuse



Improvements Breakdown

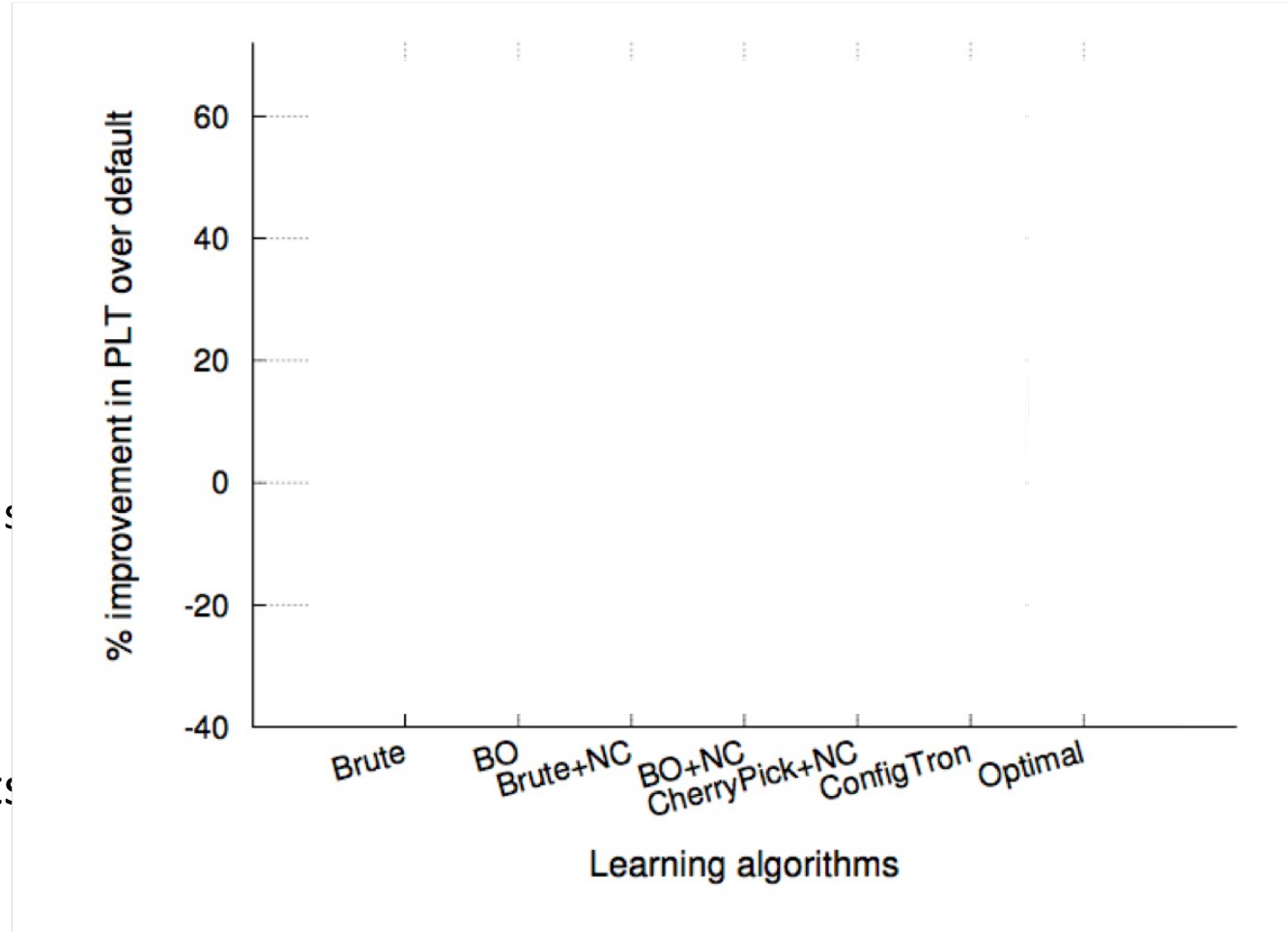


PLT Improvements Compared to Optimal



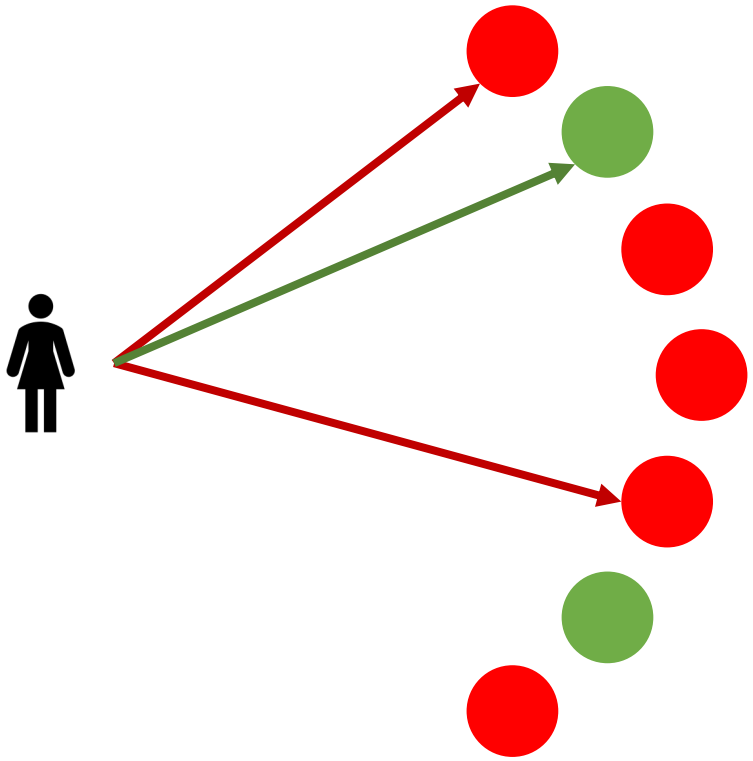
Evaluation

- Trace-driven simulation
  - Packet traces from CAIDA, MAWI. Network measurements from FCC, Pantheon.
  - Baseline algorithms.
    - Brute-force (Brute).
    - Bayesian Optimization (BO).
    - Brute-force with Network Classes (Brute+NC).
    - Bayesian Optimization with Network Clas (BO+NC).
    - CherryPick with Network Classes (CherryPick+NC).
- Real-world deployment
  - Using servers hosted in AWS and clients distributed across the globe



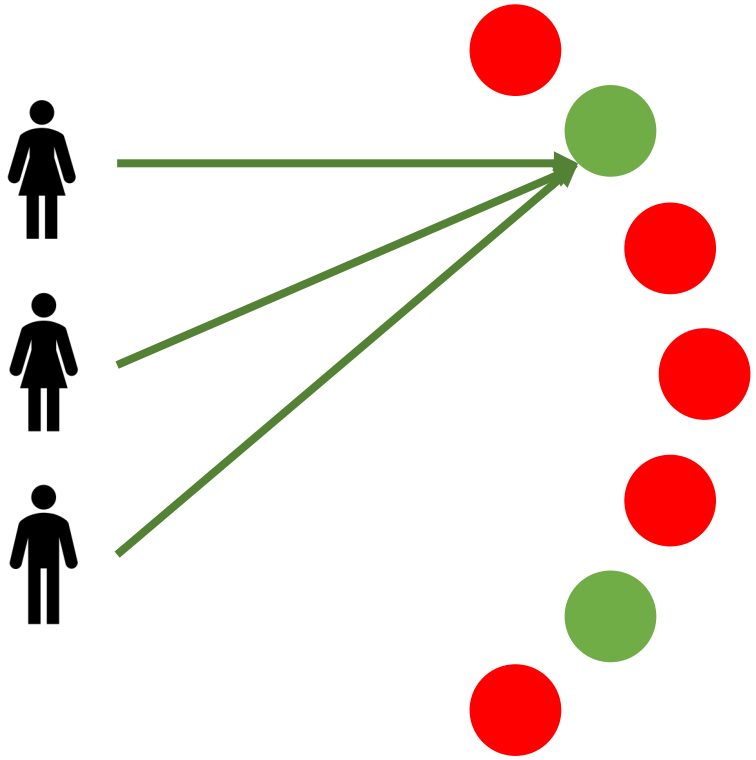
Exploration

- Bayesian optimization.
  - Guided exploration.
  - Expected improvement.



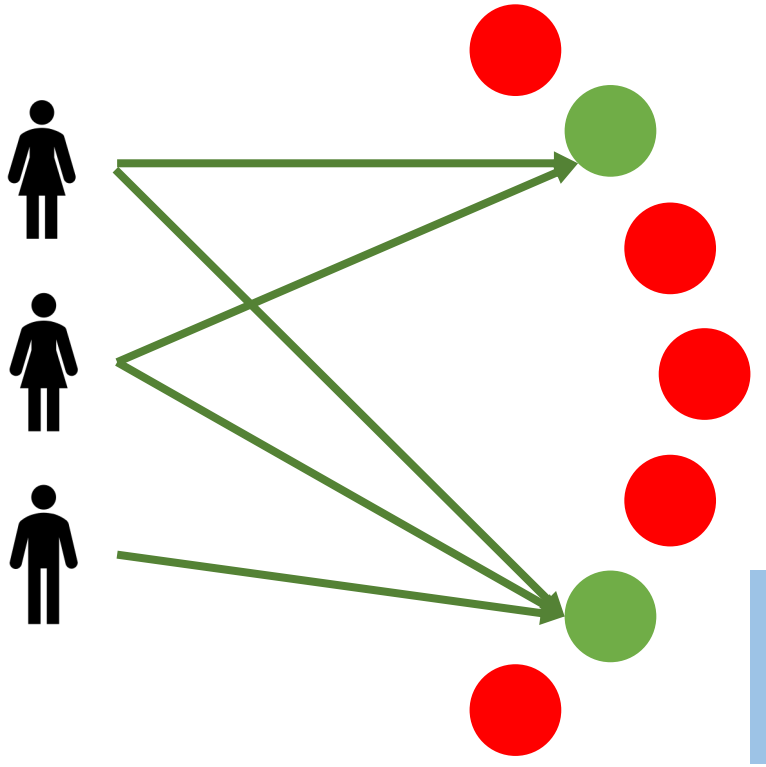
Exploitation

- Decision Tree.
  - Exploits a good direction.
  - Uses what's learnt from other users for a new user.
  - Aids in bootstrapping.

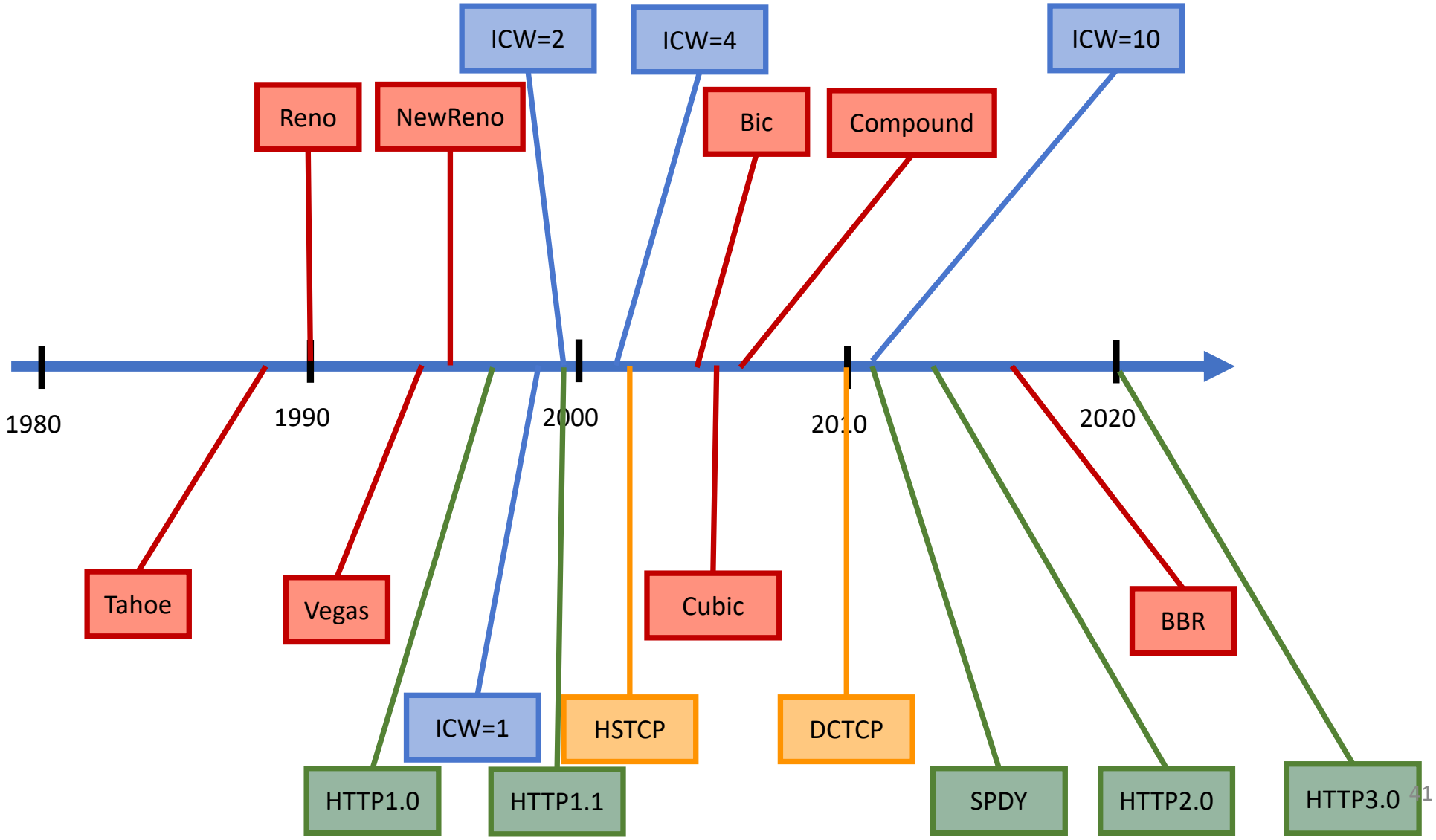


Re-sample

- Random (A/B testing).
  - Tests random directions.
  - Update past models.

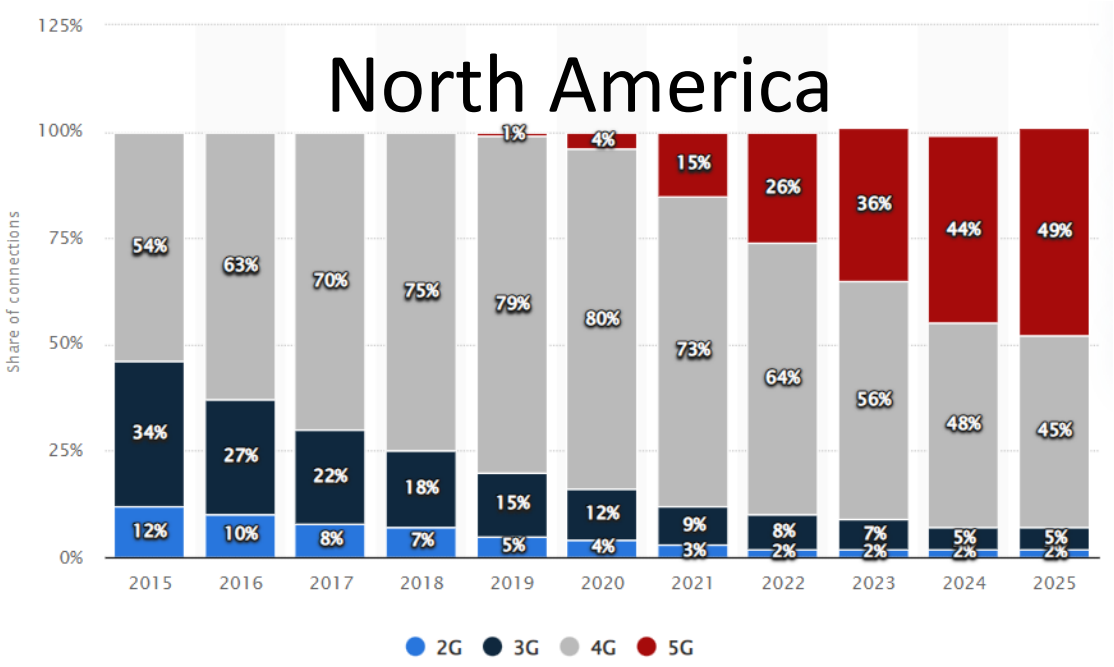


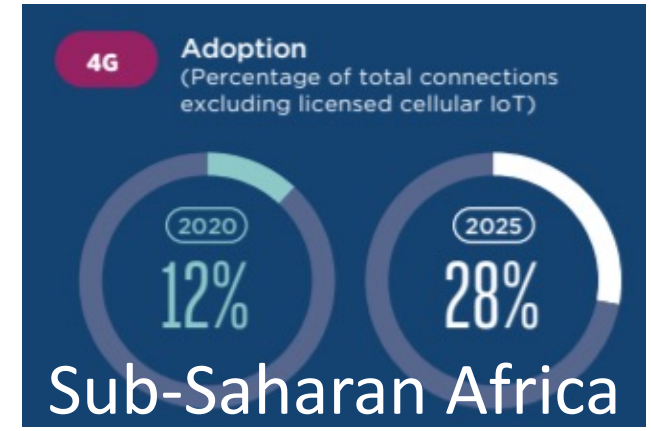
Networking stack evolution

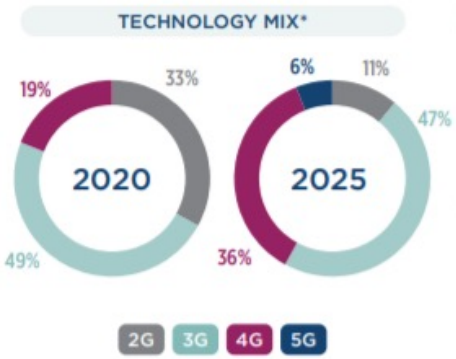


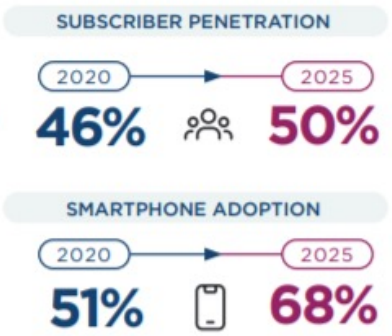
Improving performance for <u>everyone</u> is challenging.

- Diversity of networks across regions.









Conclusion + Future work

- One-size-fits-all approach to tuning networking stack in sub-ooptimal.
- Tuning network configurations intelligently can improve PLT by as much as 20% on median.
- Calls for a re-design of networking stack.
- Time of the day aspect of learning.
  - Traffic patterns change across different times of the day.
  - Separate model for different time intervals? Time as a feature?
- Understanding why a config is better than others for a network.