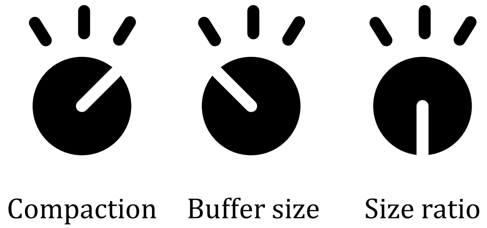
ENDURE: A Robust Tuning Paradigm for LSM Trees Under Workload Uncertainty

Andy Huynh, Harshal A. Chaudhari, Evimaria Terzi, Manos Athanassoulis

Age of Log-Structured Merge-Trees

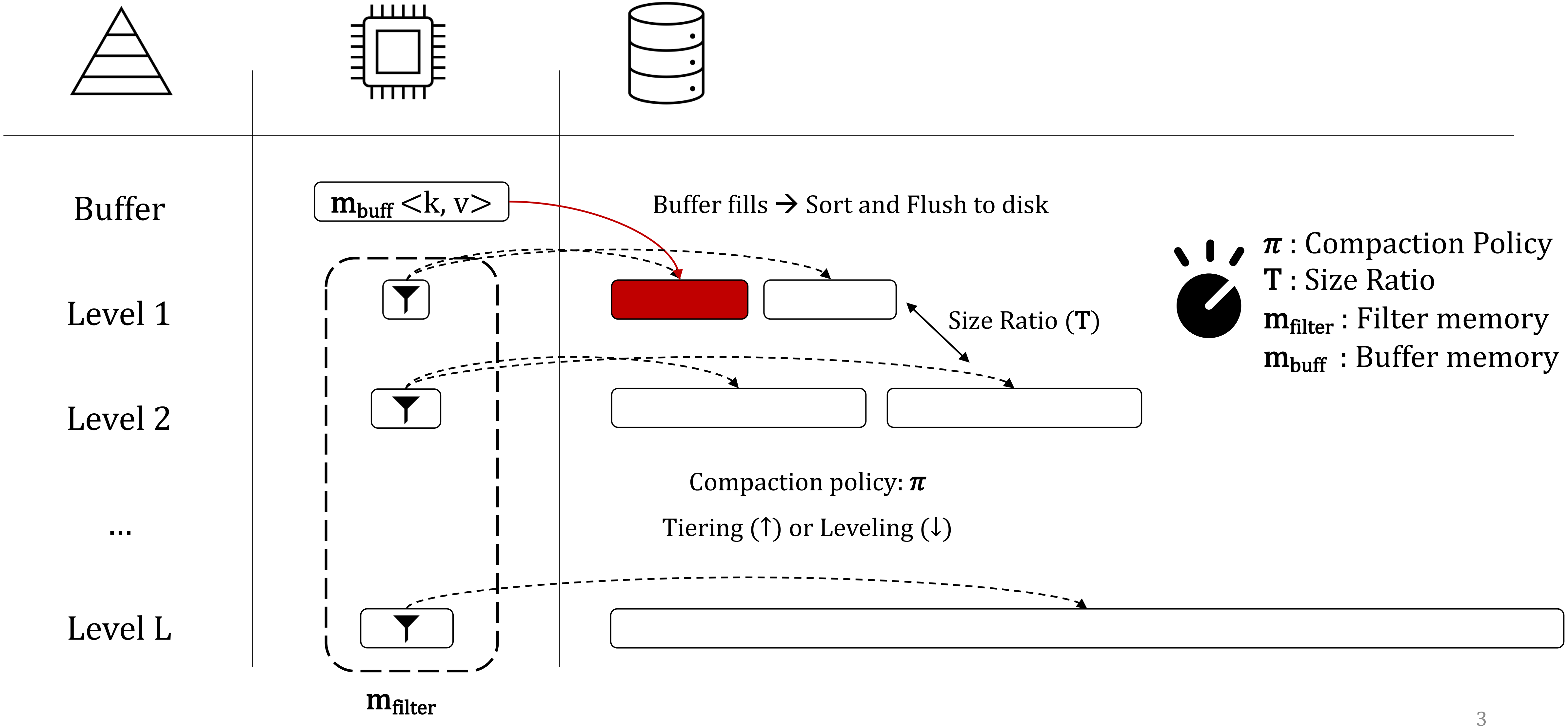
Flexibility for applications



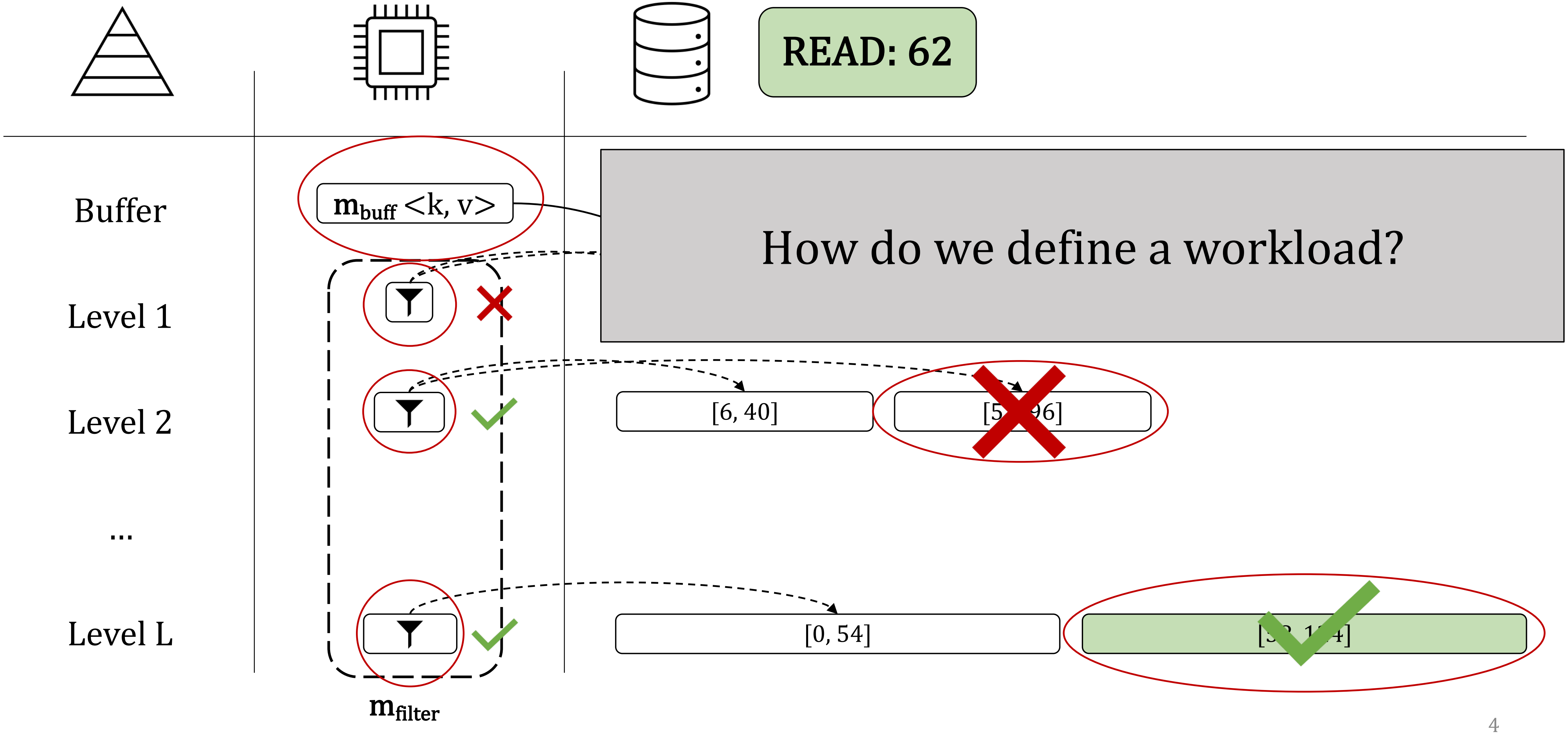
High impact tuning knobs

How do we go about tuning these knobs?

LSM Trees – Mechanics



LSM Trees – A Point Read

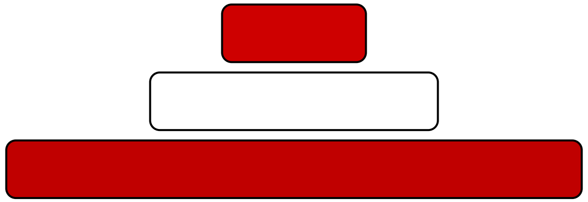


How do we define a workload?

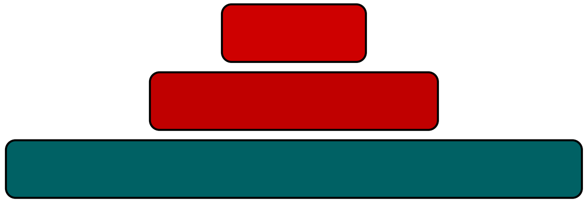
Query Types

Workload : (z_0, z_1, q, w)

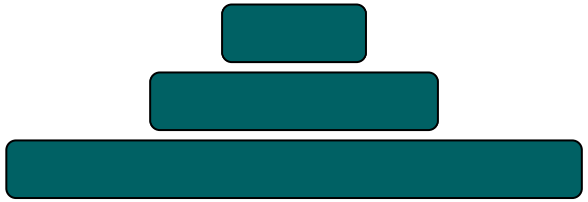
Empty Reads : z_0



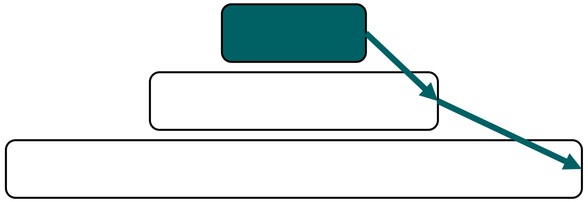
Non-Empty Reads : z_1



Range Reads: q



Writes : w



Cool! How do we go about tuning?

The LSM-Tuning Problem

$\boldsymbol{w}$: Workload   (z_0, z_1, q, w)

Φ : LSM Tree Design $(m_{buff}, m_{filter}, T, \pi)$

C : Cost

$$\Phi^* = argmin_{\Phi}\, C(\boldsymbol{w}, \Phi)$$

The LSM-Tuning Problem

$\boldsymbol{w}$: Workload (z_0, z_1, q, w)

Φ : LSM Tree Design $(m_{buff}, m_{filter}, T, \pi)$

C : Cost (I/O)

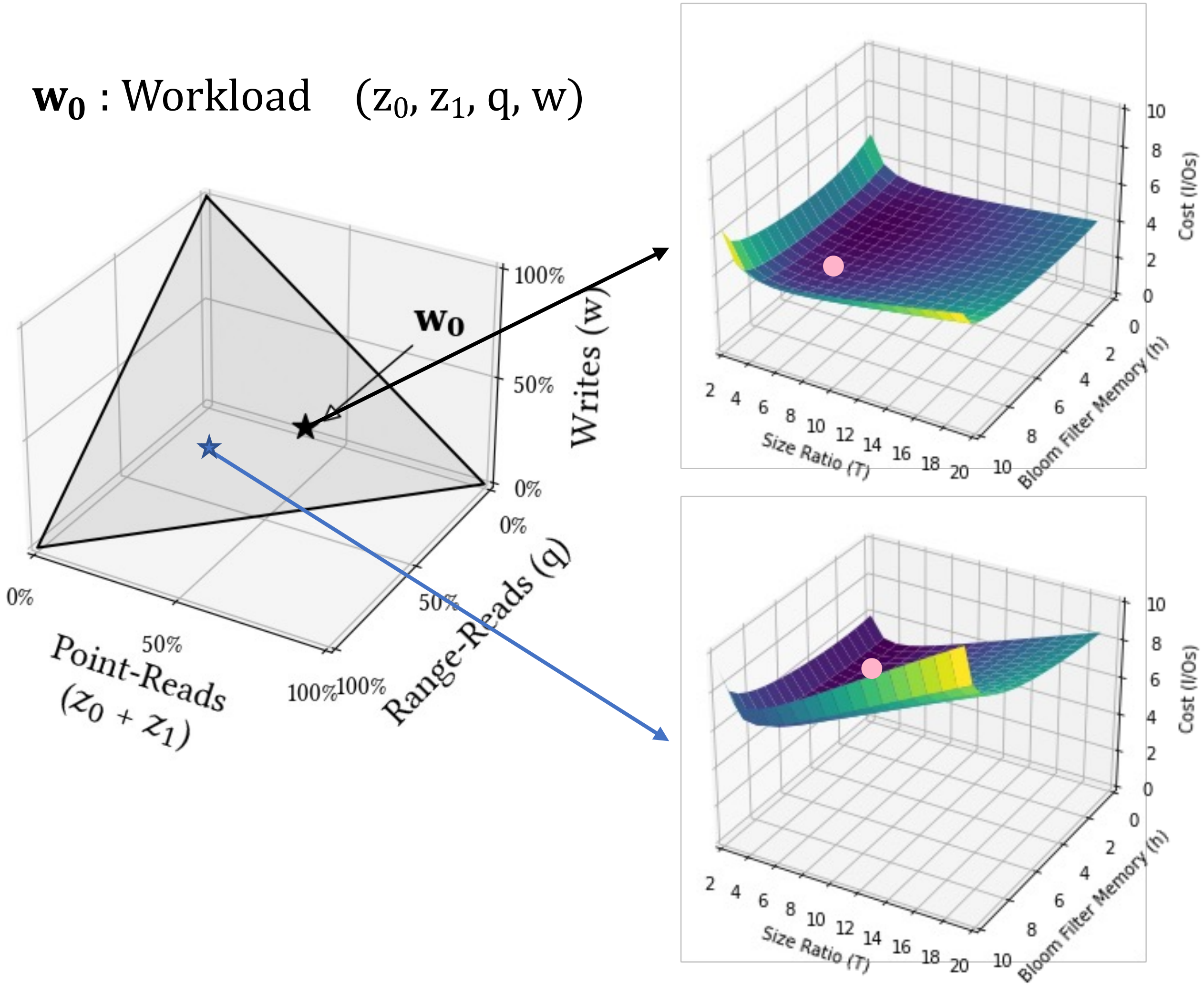
$$\Phi^* = argmin_{\Phi}\, C(\boldsymbol{w}, \Phi)$$

Define our cost function

$$C(\hat{\mathbf{w}}, \Phi) = \hat{\mathbf{w}}^\intercal \mathbf{c}(\Phi) = z_0 \cdot Z_0(\Phi) + z_1 \cdot Z_1(\Phi) + q \cdot Q(\Phi) + w \cdot W(\Phi)$$

Tuning Problems

$\mathbf{w_0}$: Workload (z_0, z_1, q, w)



Optimal configuration for the workload

Optimal tuning depends on workload

Workload uncertainty leads to **<u>sub-optimal</u>** tuning

ENDURE So Far

Introduction

LSM Trees Notation

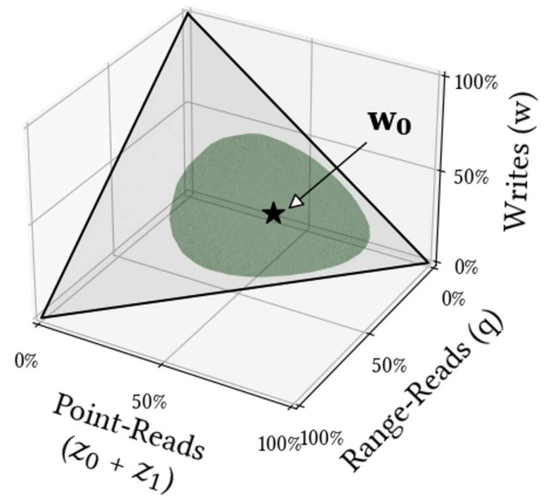
Nominally Tuning LSM Trees

ENDURE: Robustly Tuning LSM Trees

The ENDURE Pipeline

ENDURE Evaluation

The LSM-Tuning Problem



w : Workload (z_0, z_1, q, w)

Φ : LSM Tree Design $(m_{buff}, m_{filter}, T, \pi)$

C : Cost (I/O)

$$\Phi^* = \text{argmin}_{\Phi}\, C(\boldsymbol{w}, \Phi)$$

Nominal

U_{w}^{ρ}: Uncertainty Neighborhood of Workloads

ρ : Size of this neighborhood

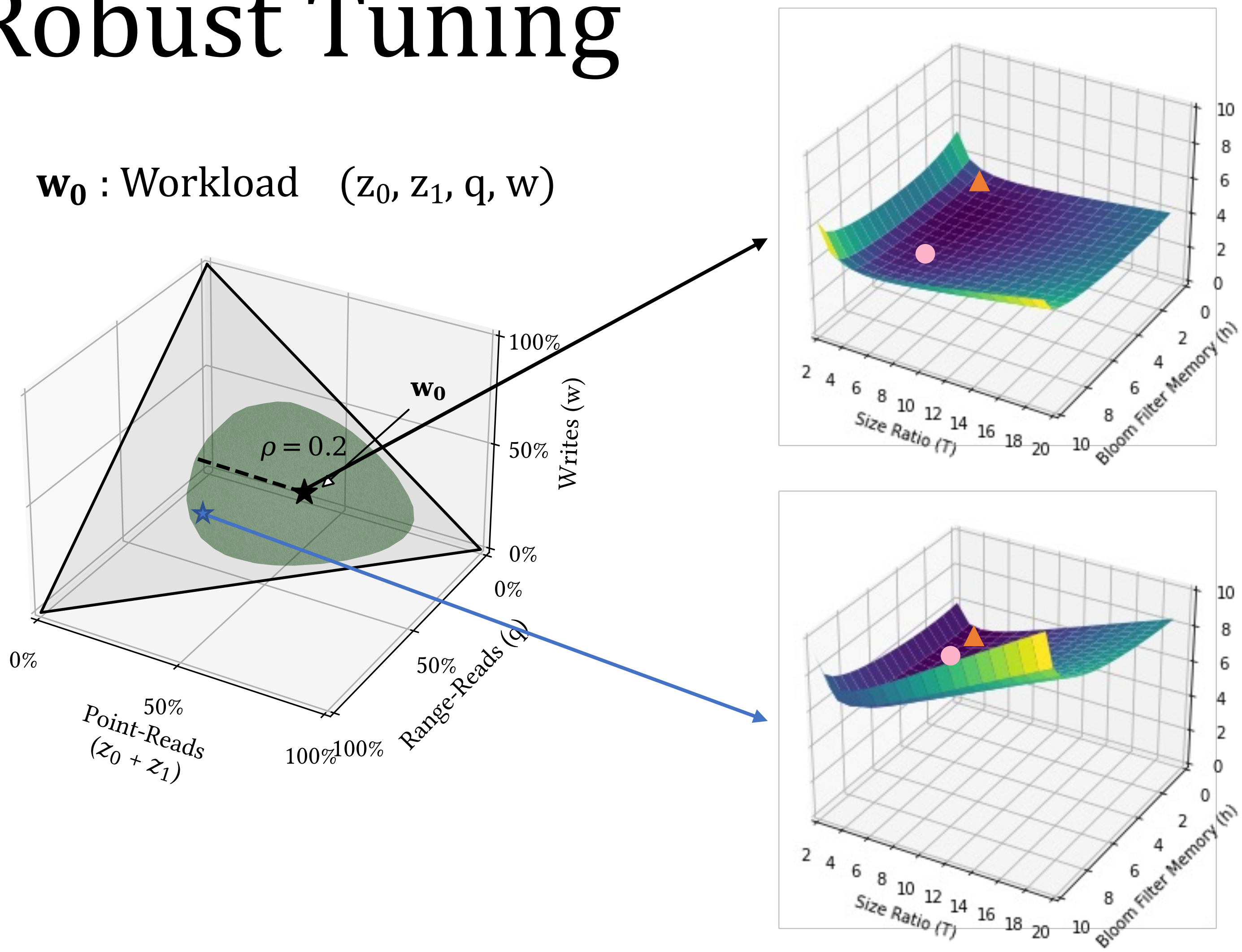
Robust

$$\Phi^* = \text{argmin}_{\Phi}\, C(\widehat{\boldsymbol{w}}, \Phi)$$

$$s.t., \quad \widehat{\boldsymbol{w}} \in U_{w}^{\rho}$$

Robust Tuning

$\mathbf{w_0}$: Workload (z_0, z_1, q, w)



$$\Phi^* = \text{argmin}_{\Phi}\, C(\hat{\boldsymbol{w}}, \Phi)$$

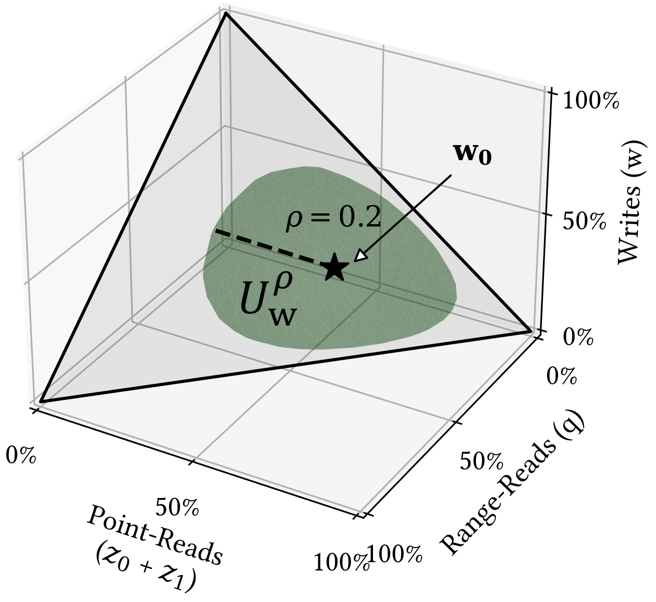
$$s.t., \qquad \hat{\boldsymbol{w}} \in U_w^{\rho}$$

Optimal configuration for the workload

Robust configuration for the workload neighborhood

Uncertainty Neighborhood

Workload Characteristic



Neighborhood of workloads (ρ) via the KL-divergence

$$I_{KL}(\widehat{w}, w) = \sum_{i=1}^{m} \widehat{w}_i \cdot \log\left(\frac{\widehat{w}_i}{w_i}\right)$$

Expected ρ?

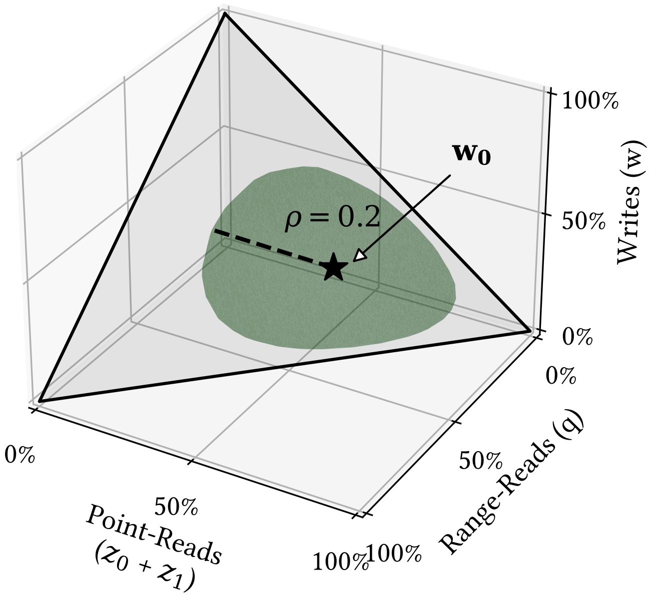
Historical workloads
maximum/average uncertainty among workload pairings

User provided workload uncertainty

U_{w}^{ρ}: Uncertainty Neighborhood of Workloads
ρ : Size of this neighborhood

ENDURE Pipeline

Workload Characteristic



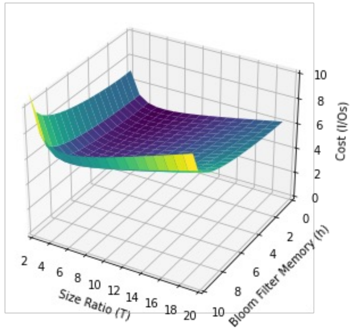
System Information

Page Size

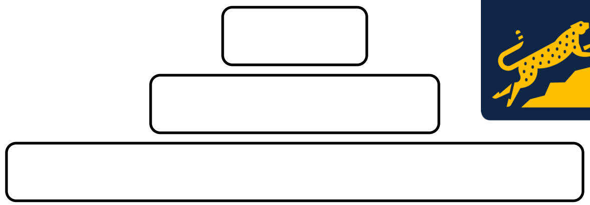
Memory Budget

ENDURE
Solves the
Robust Problem

Expected performance



RocksDB Configuration

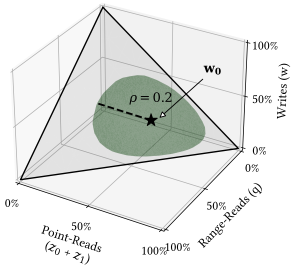


Testing Suite

ENDURE in Python, implemented in tandem with RocksDB

Uncertainty benchmark

- 15 expected workloads
- 10K randomly sampled workloads as a test-set



Index	(z_0, z_1, q, w)				Type
0	25%	25%	25%	25%	**Uniform**
1	97%	1%	1%	1%	**Unimodal**
2	1%	97%	1%	1%	
3	1%	1%	97%	1%	
4	1%	1%	1%	97%	
5	49%	49%	1%	1%	**Bimodal**
6	49%	1%	49%	1%	
7	49%	1%	1%	49%	
8	1%	49%	49%	1%	
9	1%	49%	1%	49%	
10	1%	1%	49%	49%	
11	33%	33%	33%	1%	**Trimodal**
12	33%	33%	1%	33%	
13	33%	1%	33%	33%	
14	1%	33%	33%	33%	

Normalized delta throughput

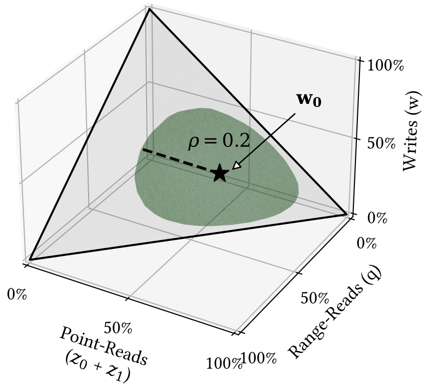
$$\Delta_{\mathbf{w}}(\Phi_1, \Phi_2) = \frac{1/C(\mathbf{w},\Phi_2) - 1/C(\mathbf{w},\Phi_1)}{1/C(\mathbf{w},\Phi_1)}$$

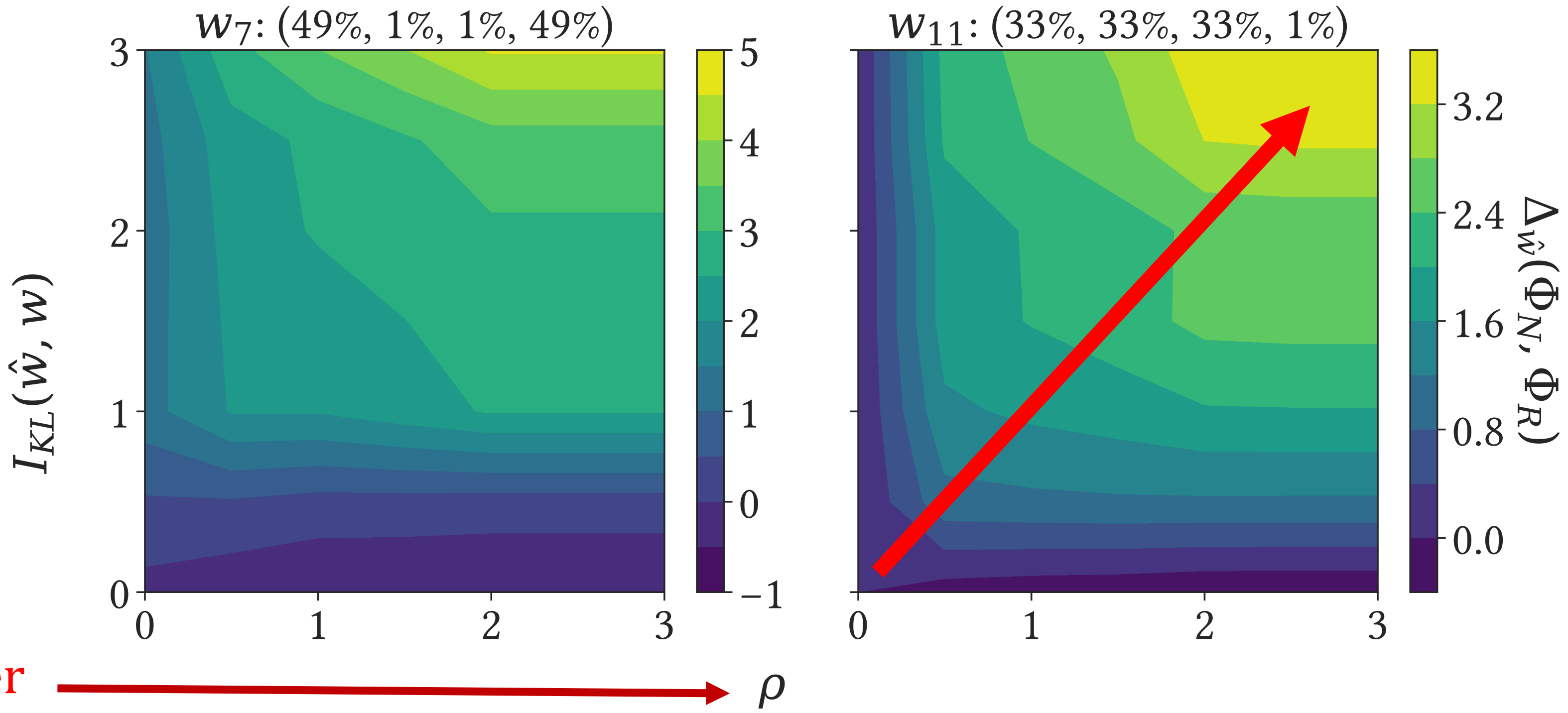
Nominal vs Robust: > 0 is better

1 means 2x speedup

Relationship of Expected and Observed ρ

Observed ρ: distance from executed workload to expected workload



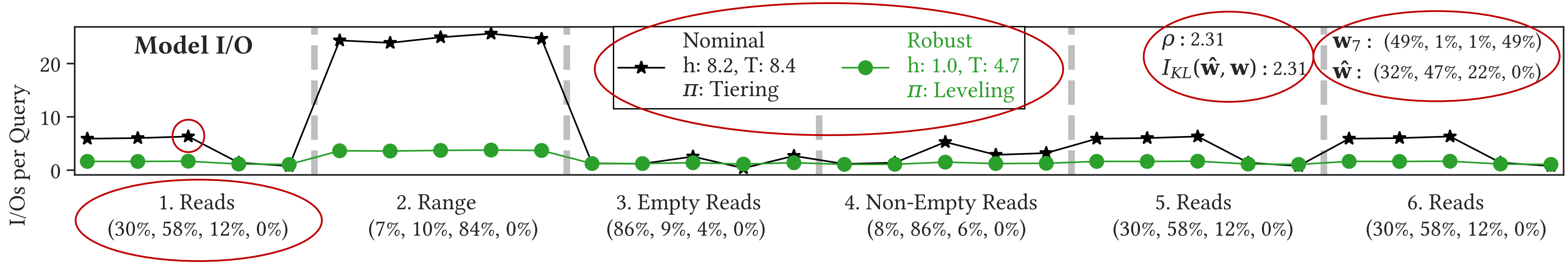


Expected ρ: workload given to tuner

Highest throughput when observed and expected ρ **match**

Lowest throughput when ρ is **mismatched**

Workload Sequence on RocksDB

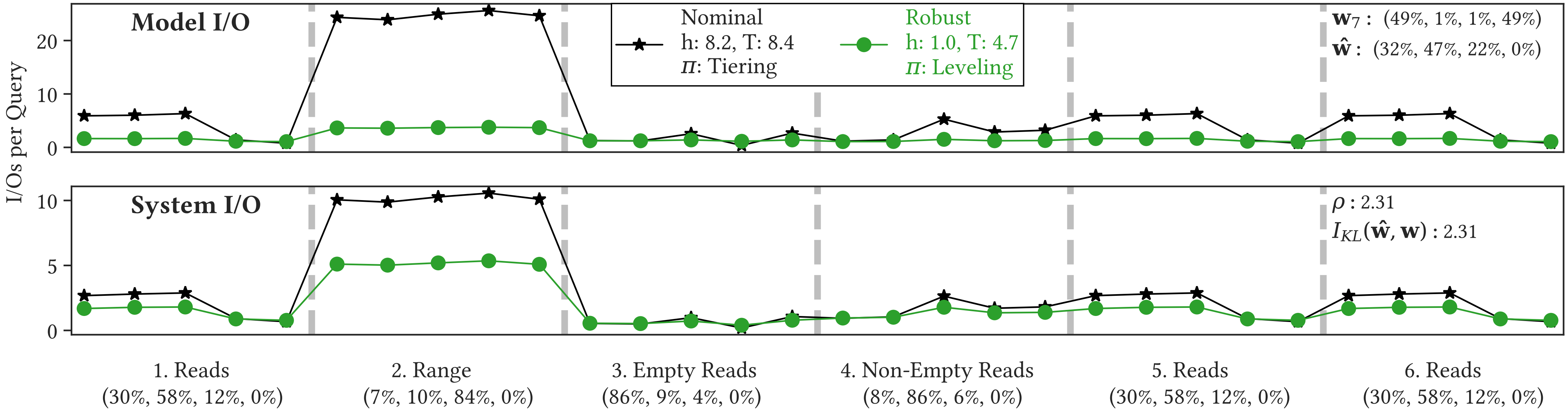


RocksDB instance setup with 10 million unique key-value pairs of size 1KB

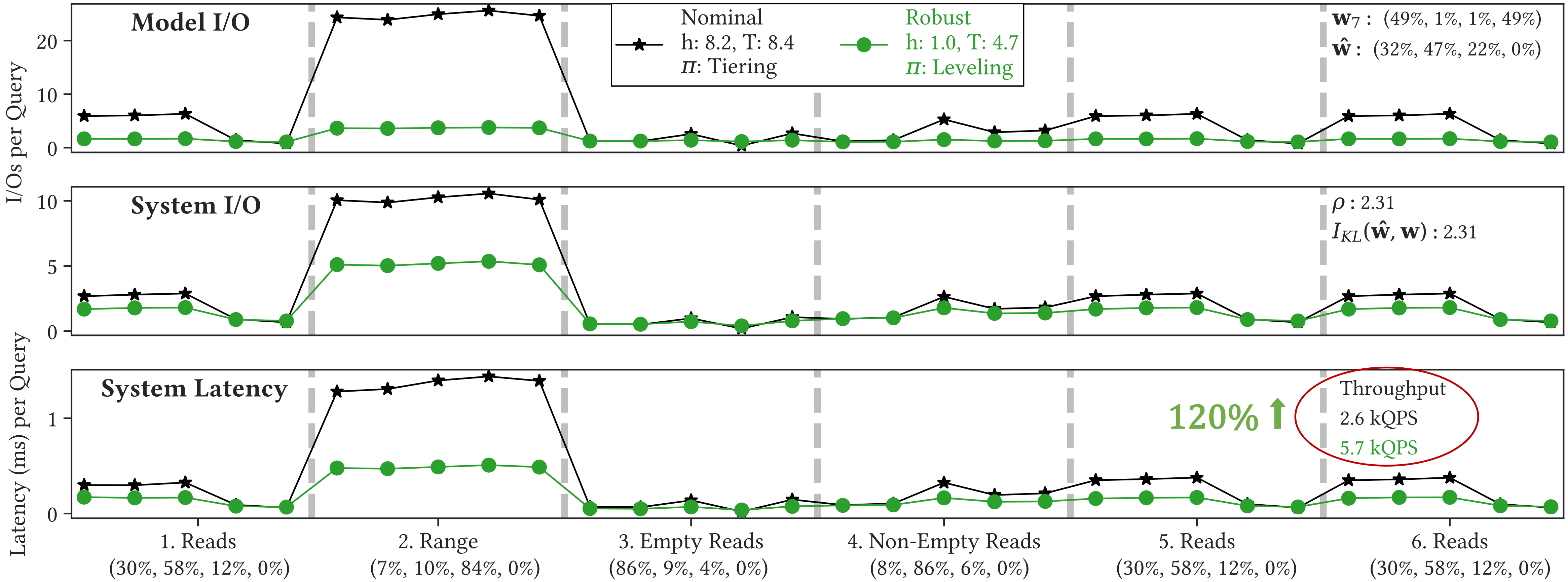
Each observation period is 200K queries, with 5 observations per session 6 million queries to the DB

Writes are unique, range queries average 1-2 pages per level

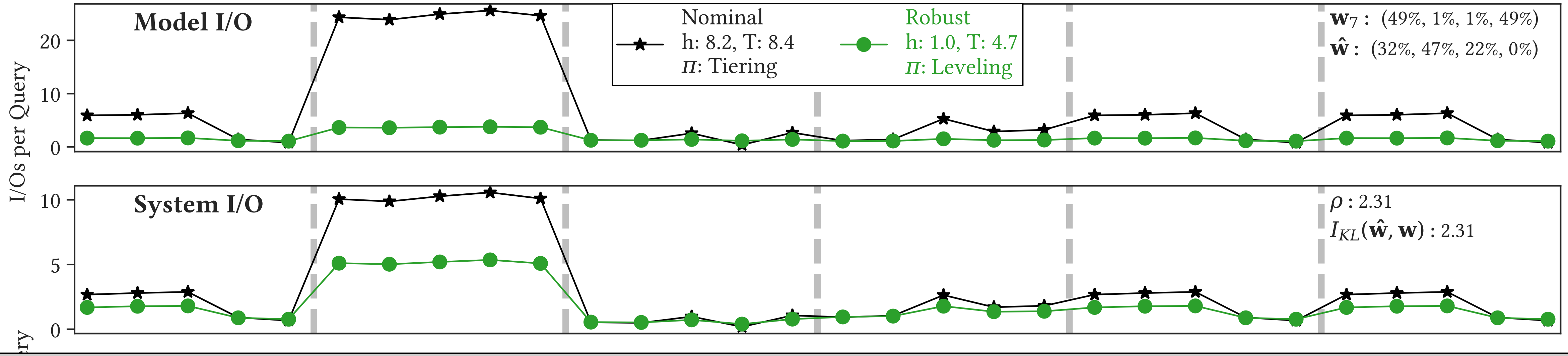
Workload Sequence



Workload Sequence



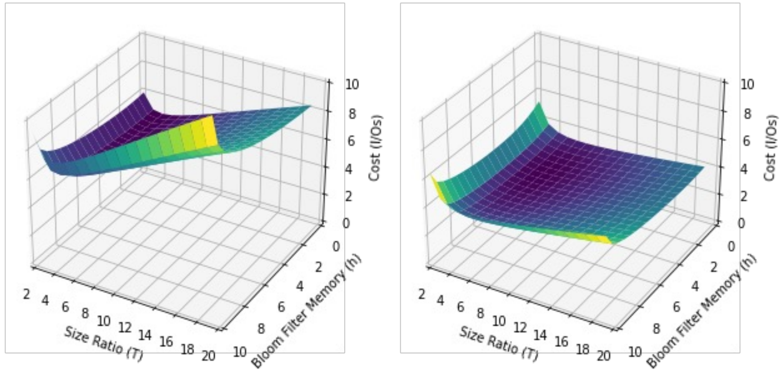
Workload Sequence



Small subset of results! Take a look at the paper for a more detailed analysis

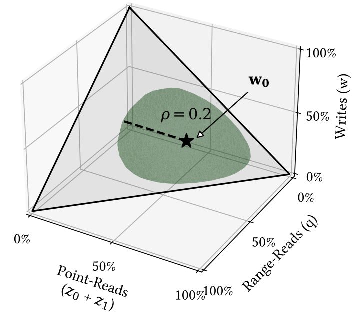
Thanks!

Workload uncertainty creates suboptimal tunings



ENDURE: robust tuning using neighborhood of workloads

Deployed ENDURE on RocksDB



Check out our poster tonight for more info!

disc.bu.edu/

www.ndhuynh.com/

@nd_huynh