ACE-in-Action: A Smart DBMS Bufferpool for SSDs

Teona Bagashvili Tarikul Islam Papon Manos Athanassoulis

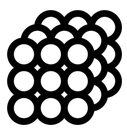
BOSTON UNIVERSITY

SSD Properties

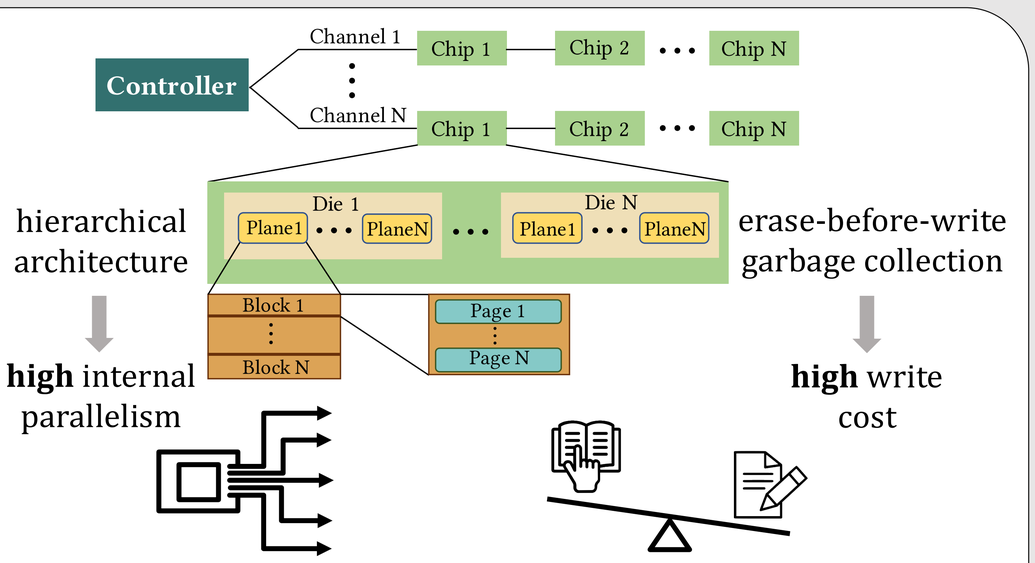
no mechanical movement

fast access

low energy



high chip density



hierarchical architecture → **high** internal parallelism

concurrency (*k*)

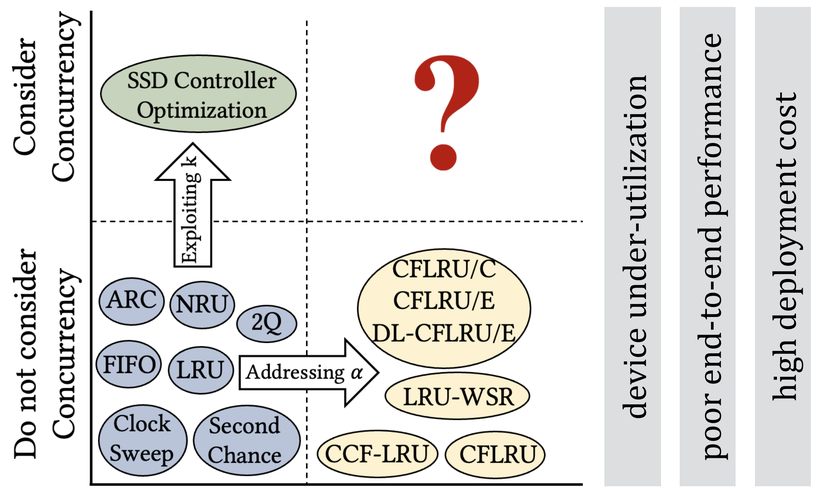
erase-before-write garbage collection → **high** write cost

read/write asymmetry (*α*)

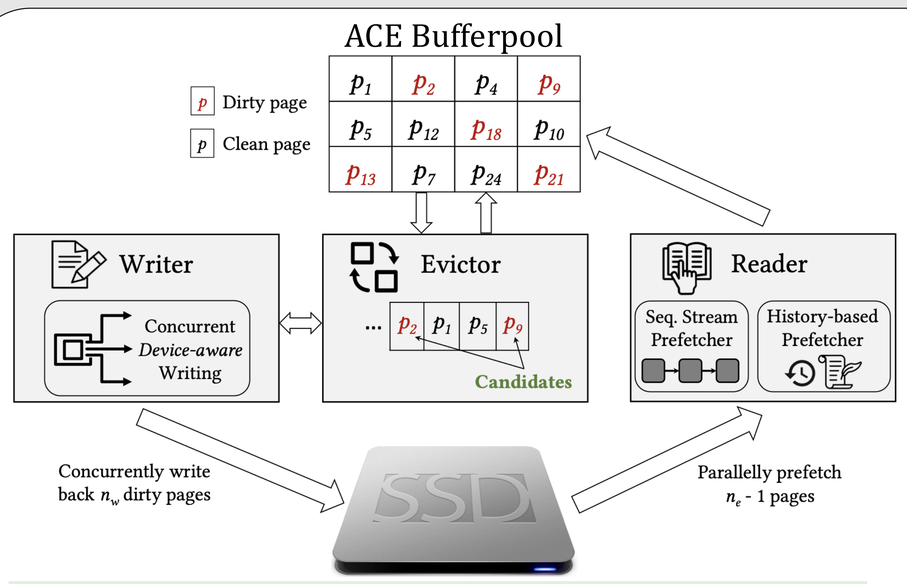
[Parametric I/O Model, **DAMON 2021**]

Bufferpool Challenges

DBMS bufferpool is tightly connected to the storage



Asymmetry/Concurrency-Aware (ACE) Bufferpool [**ICDE 2023**]

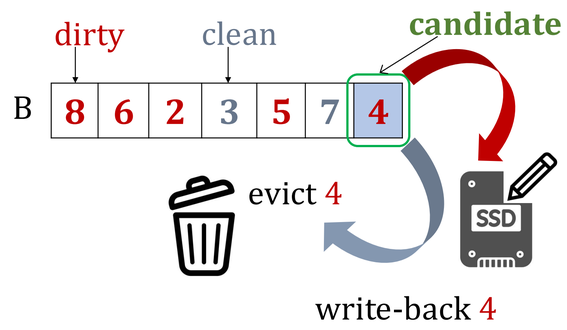


works with *any* replacement algorithm & prefetcher

always **flush multiple dirty pages** to *amortize the high write cost*

write page 1 (k=3)

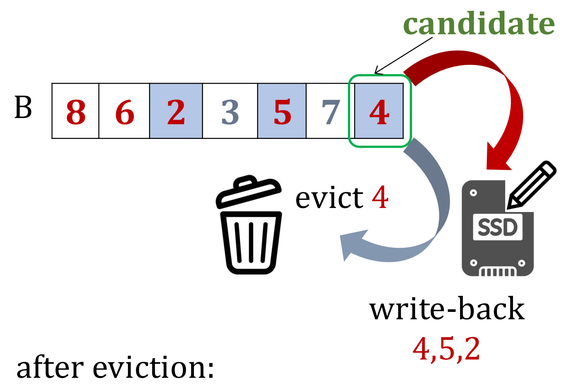
LRU



after eviction:

lots of dirty pages!

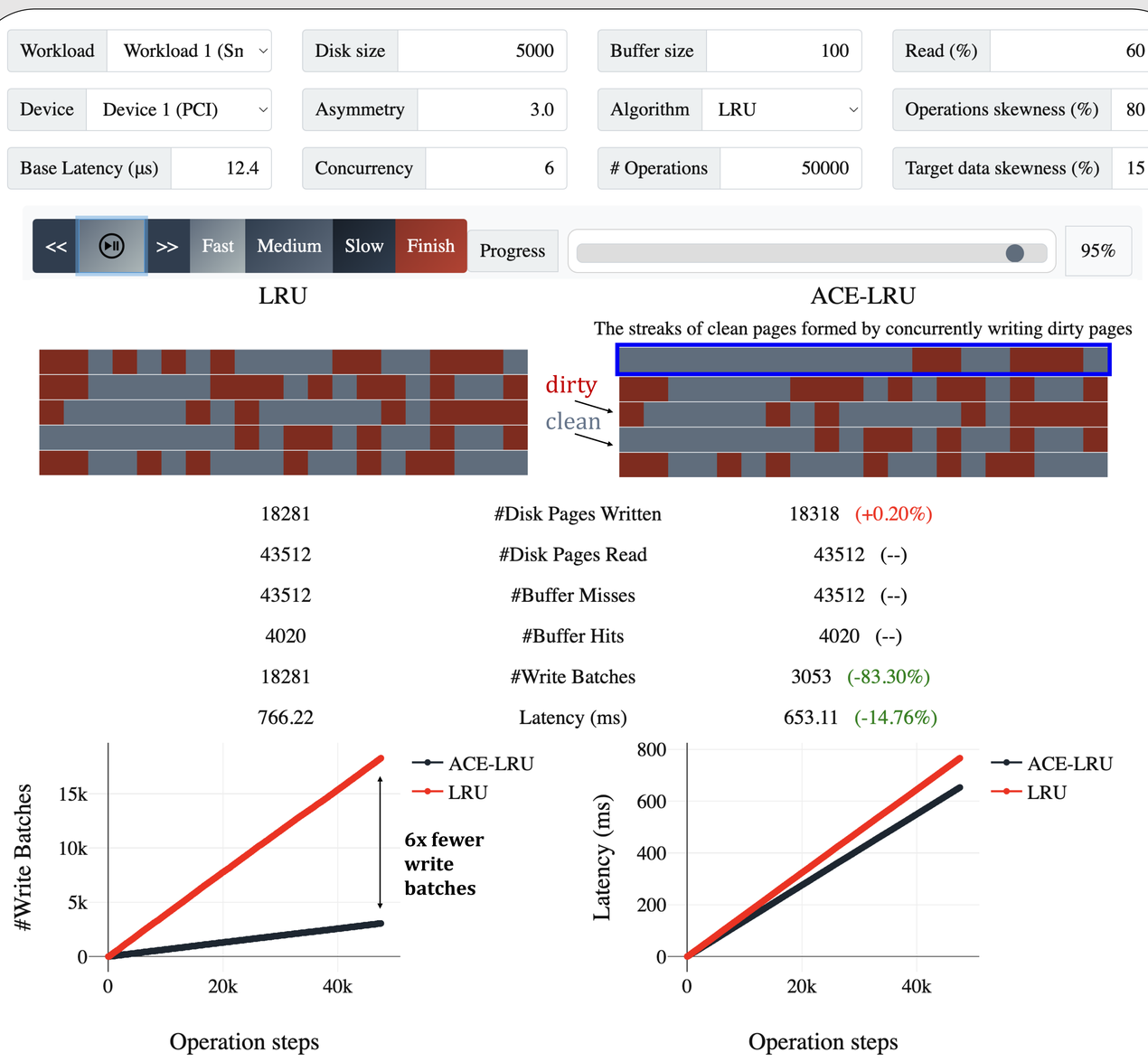
ACE-LRU



after eviction:

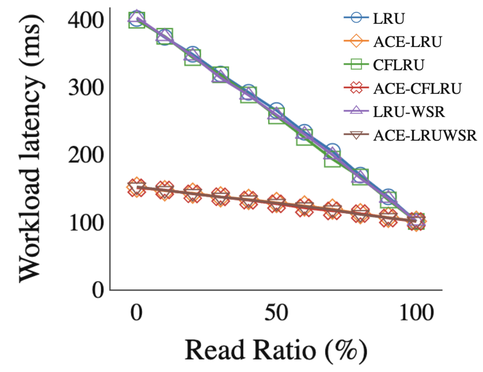
more clean pages!

ACE in Action

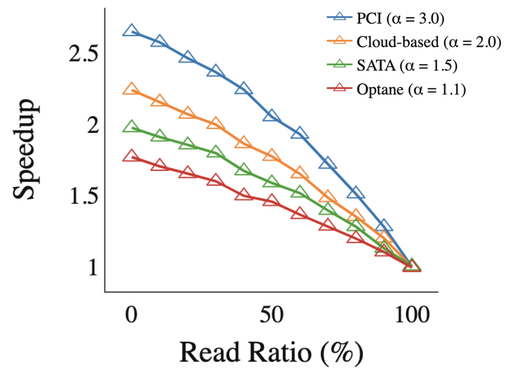


ACE **writes 6x fewer batches** using the ideal device concurrency, essentially **amortizing the write cost**

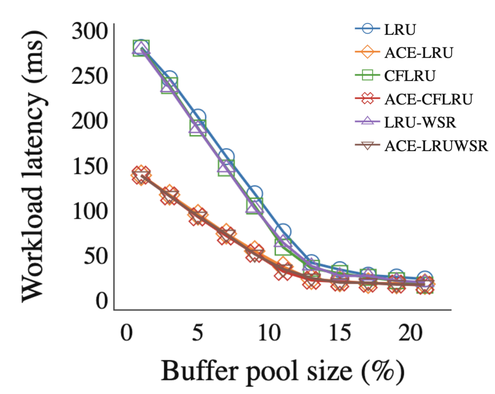
Fewer write batches reduce latency, especially in **write-heavy** workloads and on **high-asymmetry devices**



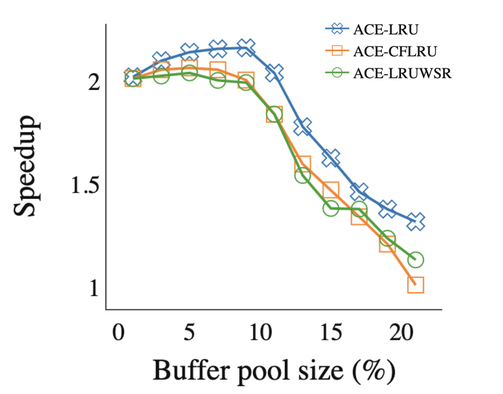
Write-heavy workloads benefit more from ACE (BP: 2%, locality: 90/10)



High asymmetry devices gain more from ACE (BP: 2%, locality: 90/10)



ACE is beneficial across a wide range of bufferpool sizes (R: 50%, locality: 90/10)



Higher benefits under memory pressure (R: 50%, locality: 90/10)

Key Insights

Understand the device's concurrency and asymmetry

Adapt the bufferpool's write-back and pre-fetching strategies

Improve the bufferpool's performance by leveraging the device's properties

BOSTON UNIVERSITY
College of Arts & Sciences
Department of Computer Science

NSF
DiSC lab