Prefetching in Hybrid Main Memory Systems

Subisha V[†], Varun Gohil[†], Nisarg Ujjainkar[†], Manu Awasthi[*]

[†]IIT Gandhinagar [*]Ashoka University

HotStorage 2020

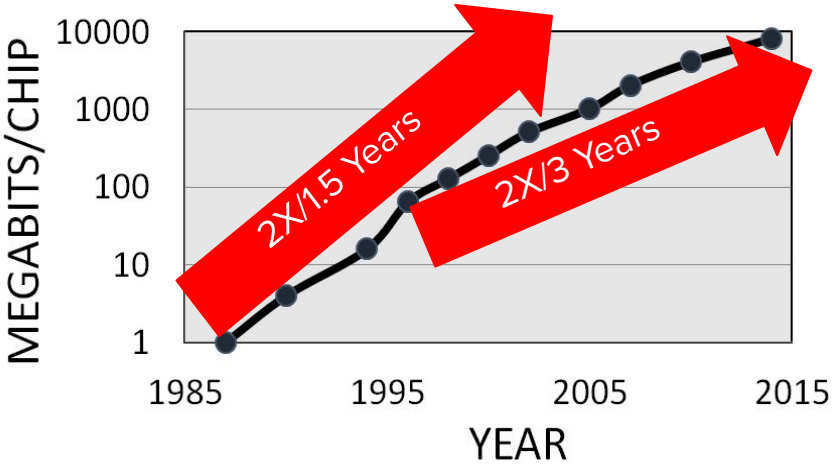
Outline of the Presentation

- Background
- Insights
- Prefetcher Design
- Evaluation
- Future Work

Outline of the Presentation

- Background
- Insights
- Prefetcher Design
- Evaluation
- Future Work

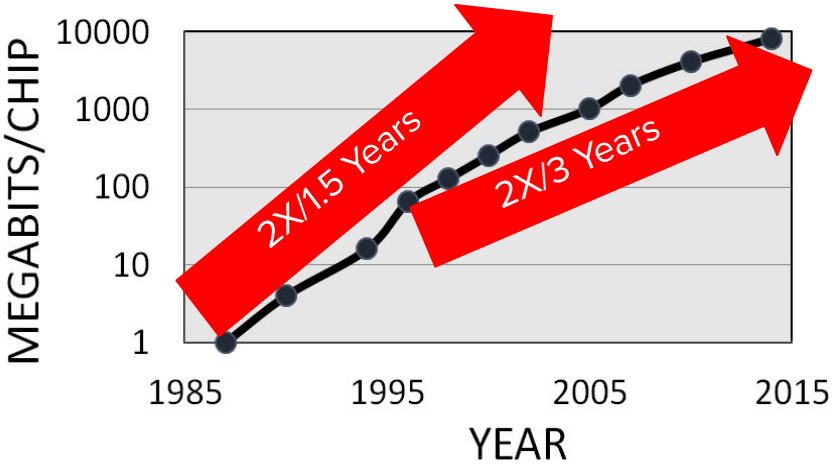
DRAM Scaling Challenge



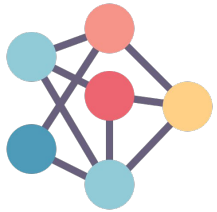
DRAM Density Scaling slowing down

Solving the DRAM Scaling Challenge, Samira Khan, ARM Research Summit 2018

DRAM Scaling Challenge



Genomics



Neural Nets

amazon DynamoDB

In-Memory Frameworks

Virtual Reality

DRAM Density Scaling slowing down

Workloads require higher memory capacity

Emerging Memory Technologies

Architecting Phase Change Memory as a Scalable DRAM Alternative

Benjamin C. Lee† Engin Ipek† Onur Mutlu‡ Doug Burger†

Design for ReRAM-based Main-Memory Architectures

Meenatchi Jagasivamani Mehdi Asnaashari Donald Yeung
Candace Walden Sylvain Dubois Bruce Jacob

Architecture Design with STT-RAM: Opportunities and Challenges

Ping Chi†, Shuangchen Li†, Yuanqing Cheng†, Yu Lu‡, Seung H. Kang‡, Yuan Xie†

and many more …

Emerging Memory Technologies

+ Better density

+ Energy efficient

Emerging Memory Technologies

+ Better density

+ Energy efficient

❌ Longer access latencies

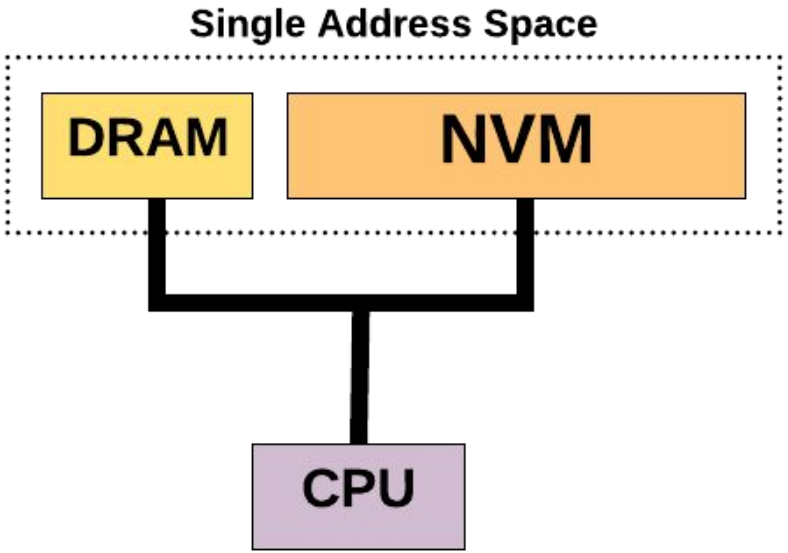
❌ Finite write endurance

Hybrid Main Memory

Use DRAM and NVM synergistically

Hybrid Main Memory

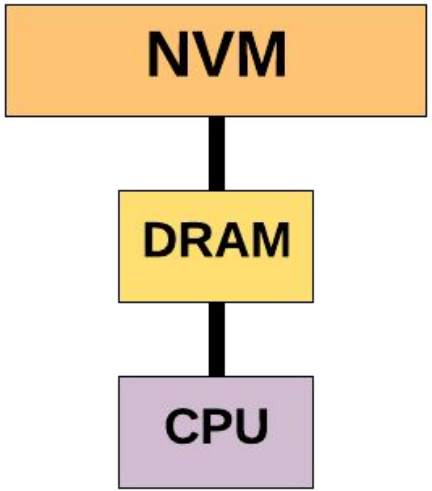
Use DRAM and NVM synergistically



Single Address Space Variant

Hybrid Main Memory

Use DRAM and NVM synergistically



DRAM as a Cache Variant

Alloy Cache

- State of the art DRAM Cache design

Alloy Cache

- State of the art DRAM Cache design
- Acts as a direct mapped cache to NVM
- Fetches data at cacheline granularity

Alloy Cache

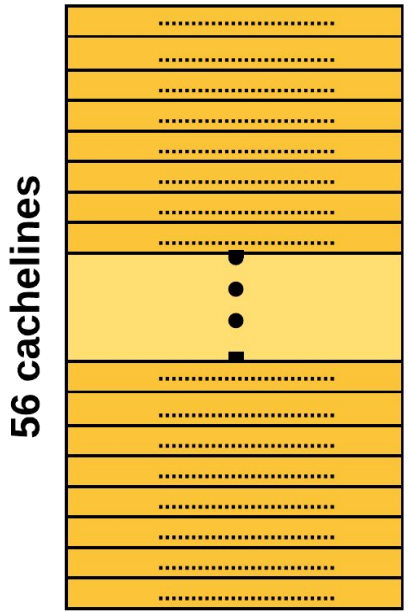
- State of the art DRAM Cache design
- Acts as a direct mapped cache to NVM
- Fetches data at cacheline granularity
- Cacheline size is 72B

Alloy Cache Page

- 4KB contiguous memory chunk

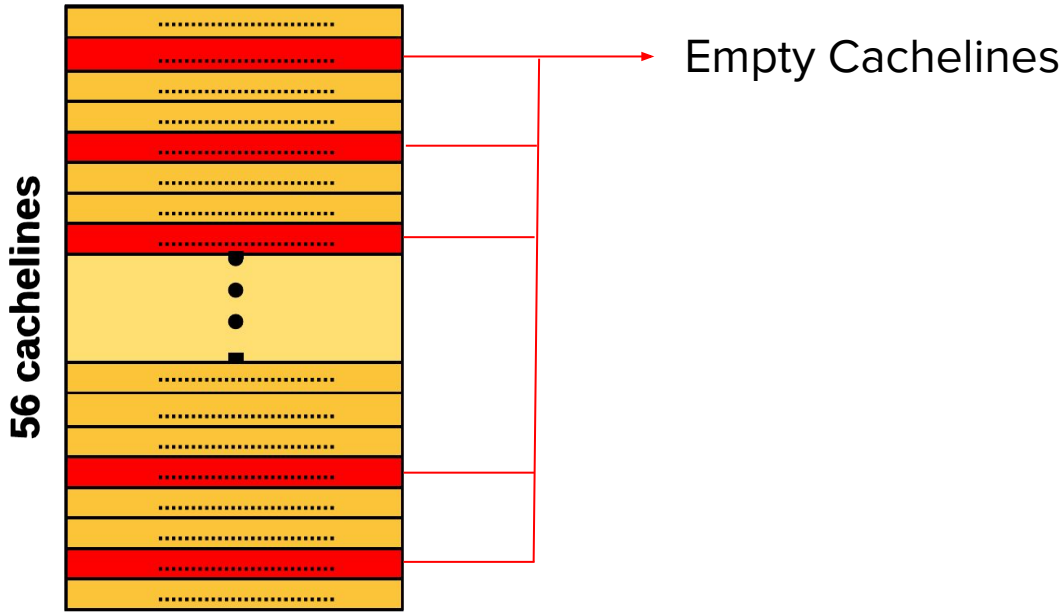
Alloy Cache Page

- 4KB contiguous memory chunk



Alloy Cache Page

- 4KB contiguous memory chunk



Outline of the Presentation

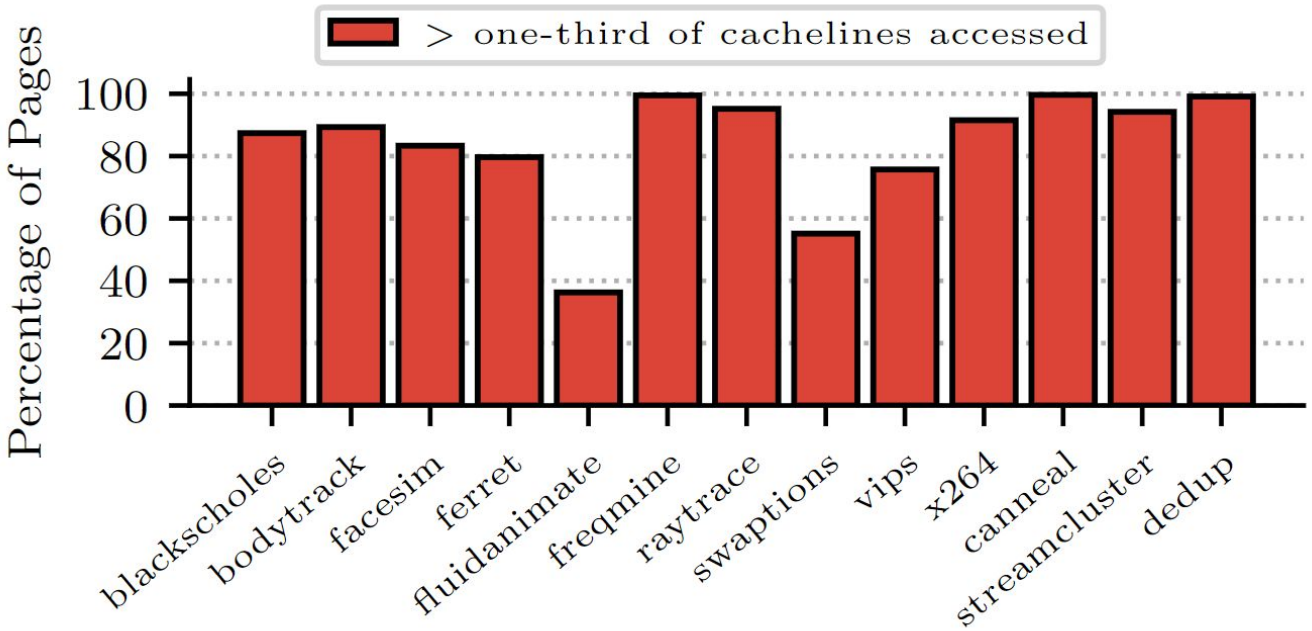
- Background
- Insights
- Prefetcher Design
- Evaluation
- Future Work

Insights

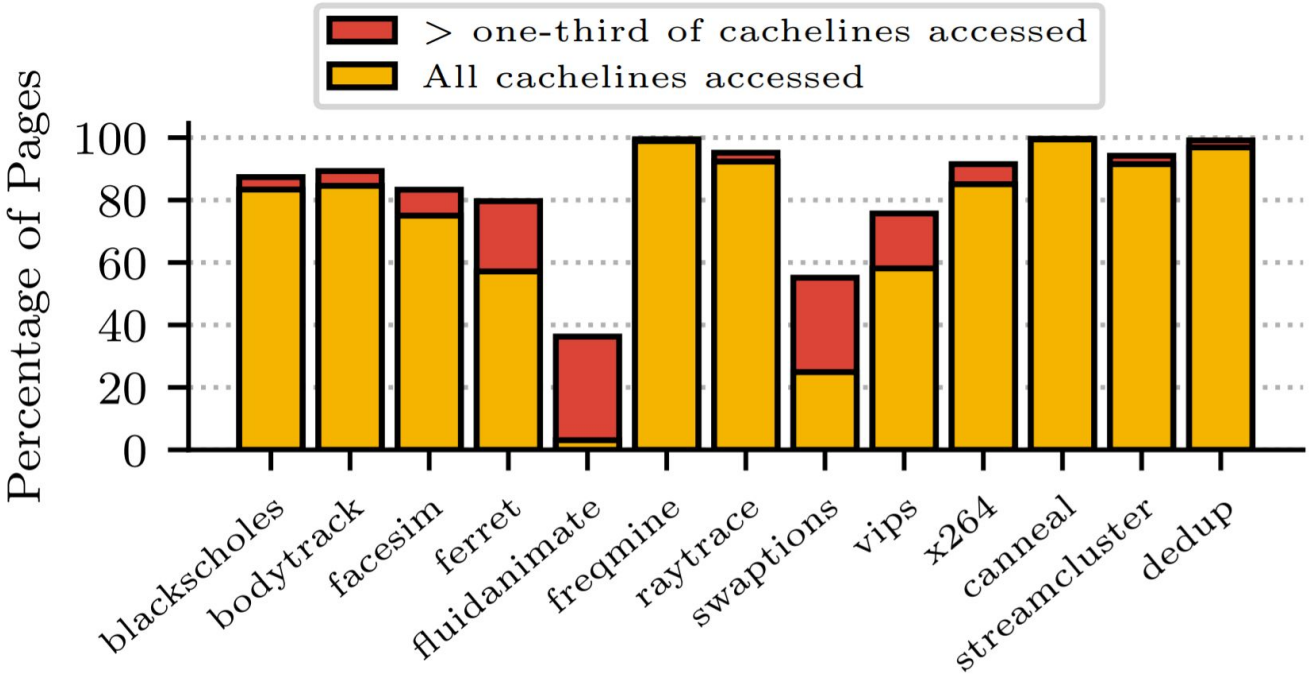
1 GB Alloy Cache, 64 GB PCM

PARSEC

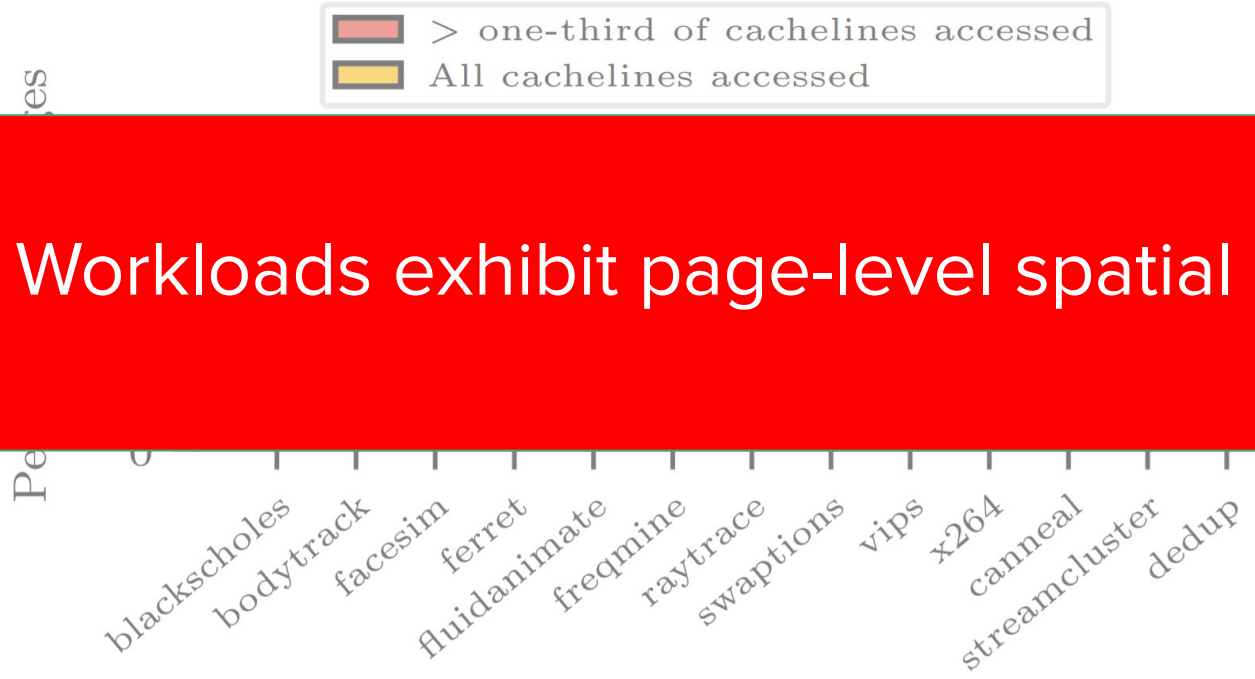
Insights


> one-third of cachelines accessed
Percentage of Pages
100
80
60
40
20
0
blackscholes
bodytrack
facesim
ferret
fluidanimate
freqmine
raytrace
swaptions
vips
x264
canneal
streamcluster
dedup

Insights

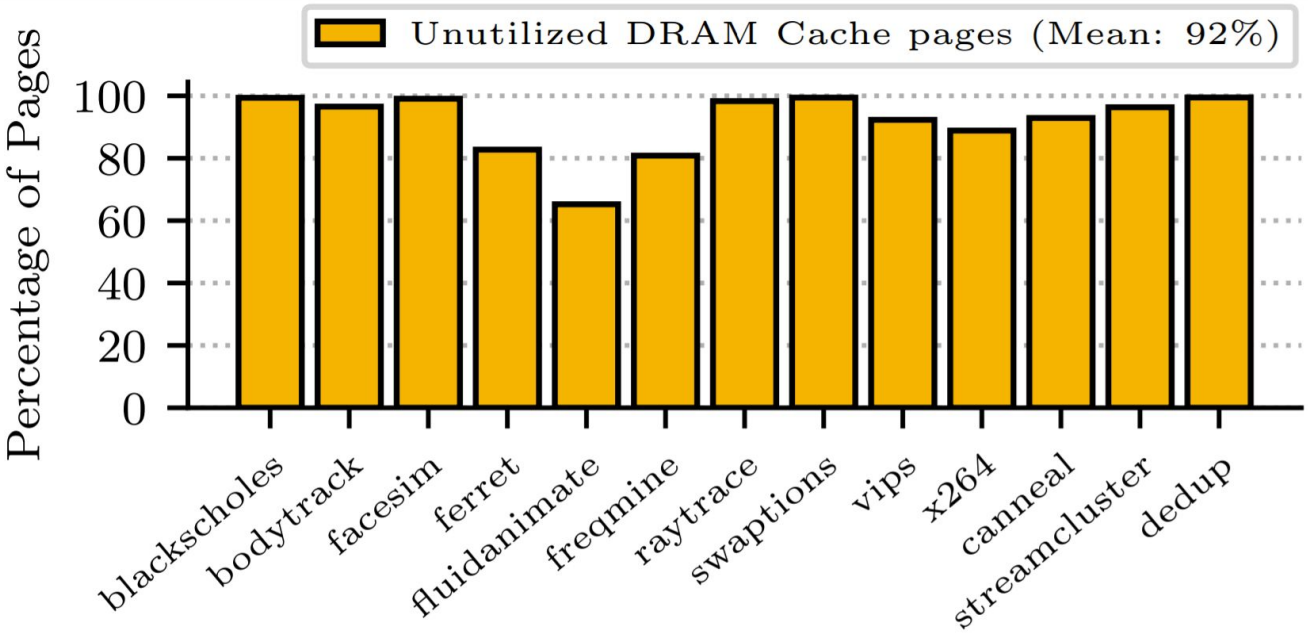

> one-third of cachelines accessed
All cachelines accessed
Percentage of Pages
100
80
60
40
20
0
blackscholes
bodytrack
facesim
ferret
fluidanimate
freqmine
raytrace
swaptions
vips
x264
canneal
streamcluster
dedup

Insights



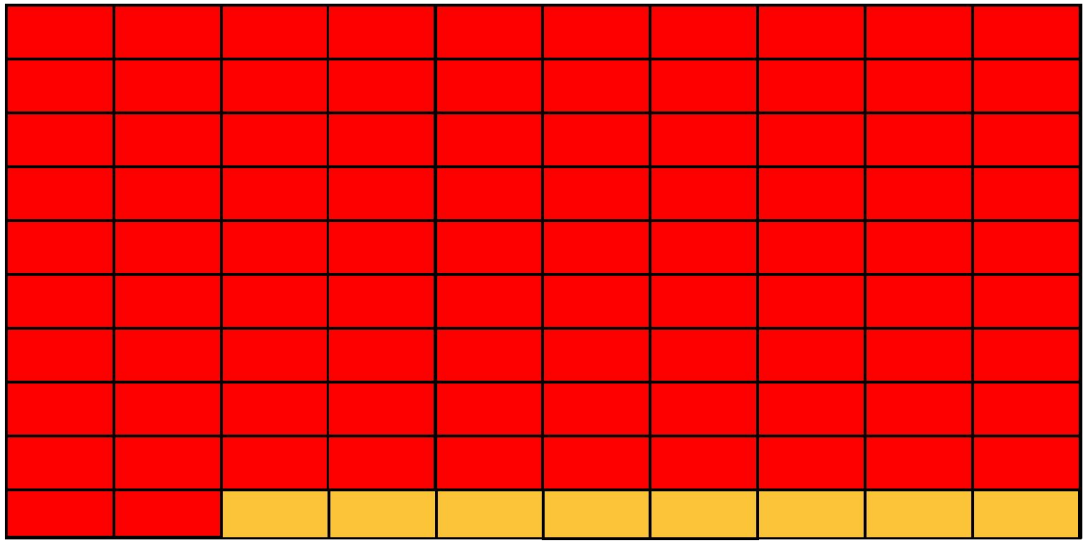
Workloads exhibit page-level spatial locality in NVM

Insights


Unutilized DRAM Cache pages (Mean: 92%)
Percentage of Pages
100
80
60
40
20
0
blackscholes
bodytrack
facesim
ferret
fluidanimate
freqmine
raytrace
swaptions
vips
x264
canneal
streamcluster
dedup

Insights

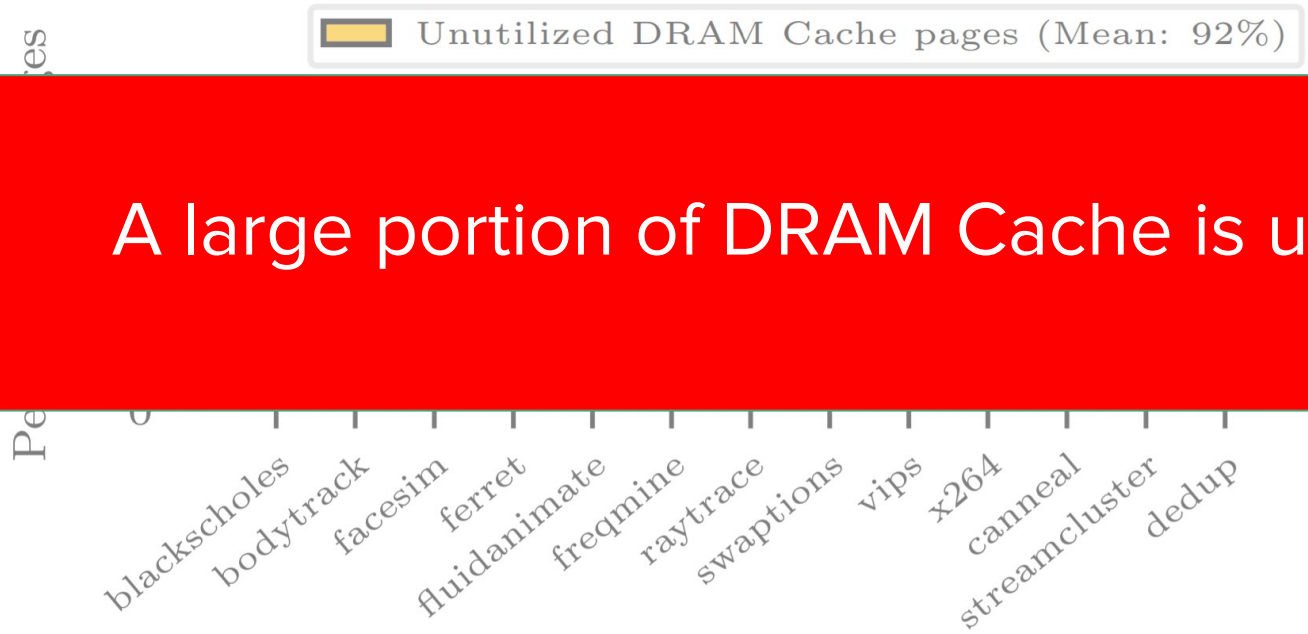
92% of DRAM Cache pages are completely empty !



Unallocated/Empty Page

Allocated Page

Insights



A large portion of DRAM Cache is unallocated

Outline of the Presentation

- Background
- Insights
- Prefetcher Design
- Evaluation
- Future Work

Prefetcher Design

- Page-Level Spatial Locality in NVM

  ⇒ Prefetch at page granularity

Prefetcher Design

- Page-Level Spatial Locality in NVM

  ⇒ Prefetch at page granularity

- DRAM Cache is largely unallocated

  ⇒ Place prefetched pages in DRAM Cache

Prefetcher Design

Prefetcher Design

- When to prefetch?

Prefetcher Design

- When to prefetch?

- Where to place prefetched data in DRAM Cache?

Prefetcher Design

- When to prefetch?
- Where to place prefetched data in DRAM Cache?
- How to identify type of data at DRAM Cache location?

Prefetcher Design

- When to prefetch?
- Where to place prefetched data in DRAM Cache?
- How to identify type of data at DRAM Cache location?
- How to check if data is in a prefetched page?

When to Prefetch?

When to Prefetch?

Prefetch a page if

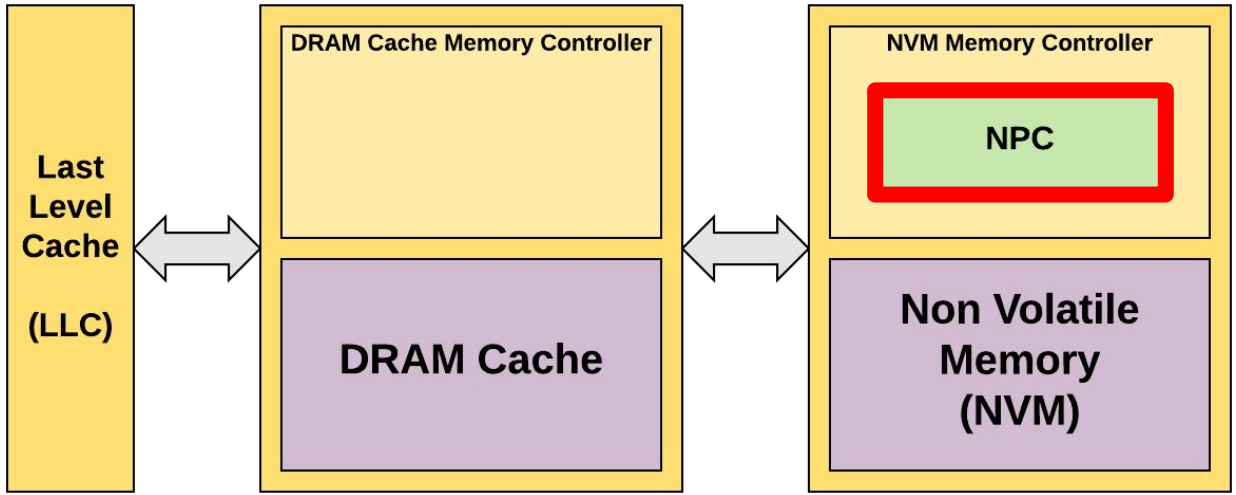
⇒ #cacheline access ≥ Access Threshold (AT)

⇒ #unique cacheline access ≥ Unique Access Threshold (UAT)

When to Prefetch?

NVM Page Classifier (NPC)

⇒ Stores cacheline access history of recently used pages



NVM Page Classifier Entry

Page Number: log_2N	Access Counter: log_2AT	Cacheline Access Vector: 64	Unique Access Counter: log_2AT

N : Max number of pages that can be present in NVM

AT: Access Threshold

NVM Page Classifier Entry

N : Max number of pages that can be present in NVM

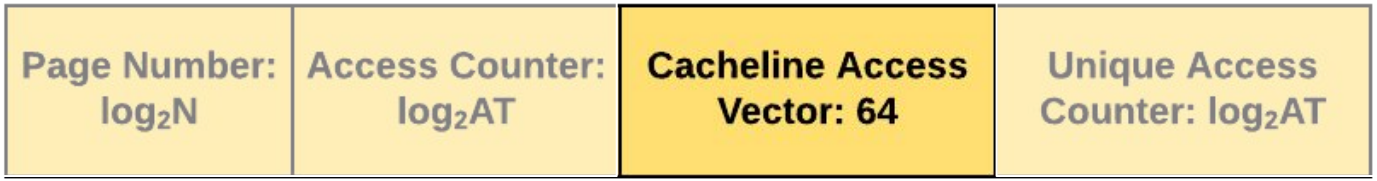
AT: Access Threshold

NVM Page Classifier Entry

N : Max number of pages that can be present in NVM

AT: Access Threshold

NVM Page Classifier Entry



N : Max number of pages that can be present in NVM

AT: Access Threshold

NVM Page Classifier Entry

N : Max number of pages that can be present in NVM

AT: Access Threshold

NVM Page Classifier Entry

Page Number: log_2N	Access Counter: log_2AT	Cacheline Access Vector: 64	Unique Access Counter: log_2AT

N : Max number of pages that can be present in NVM

AT: Access Threshold

Where to place Prefetched Page?

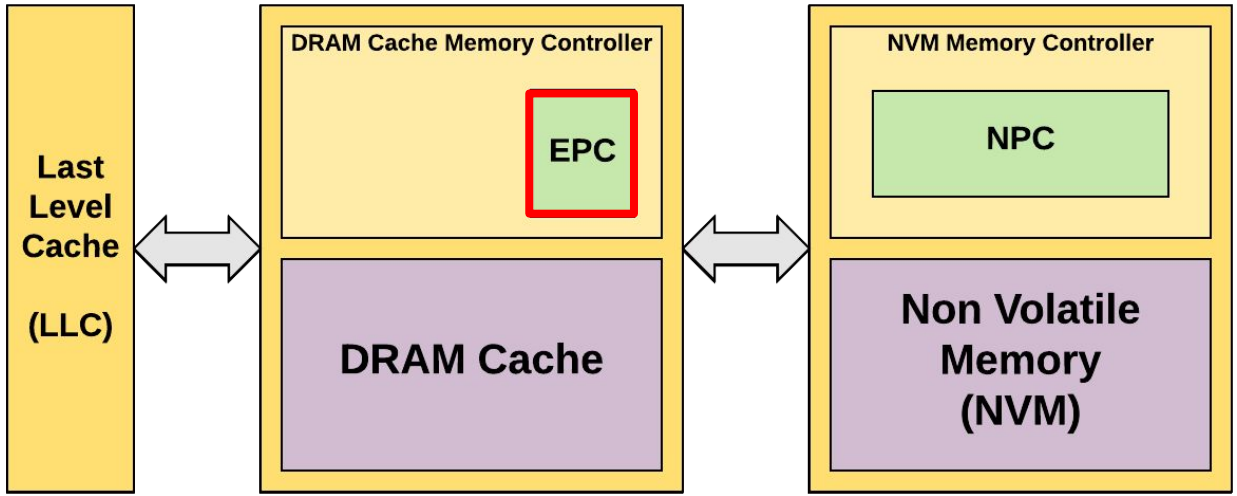
Where to place Prefetched Page?

Last Unallocated DRAM Cache page

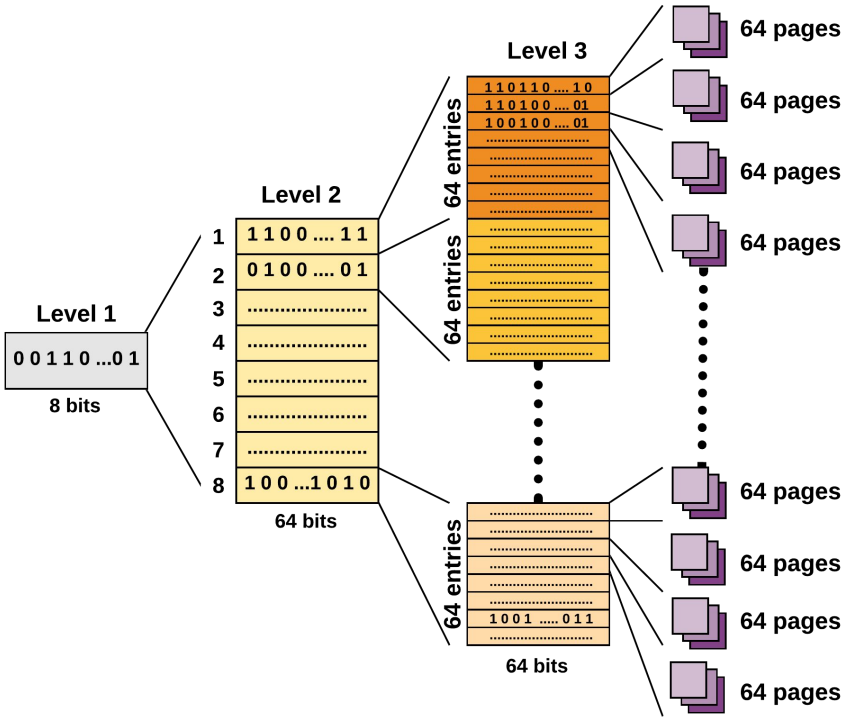
Where to place Prefetched Page?

Empty Page Classifier (EPC)

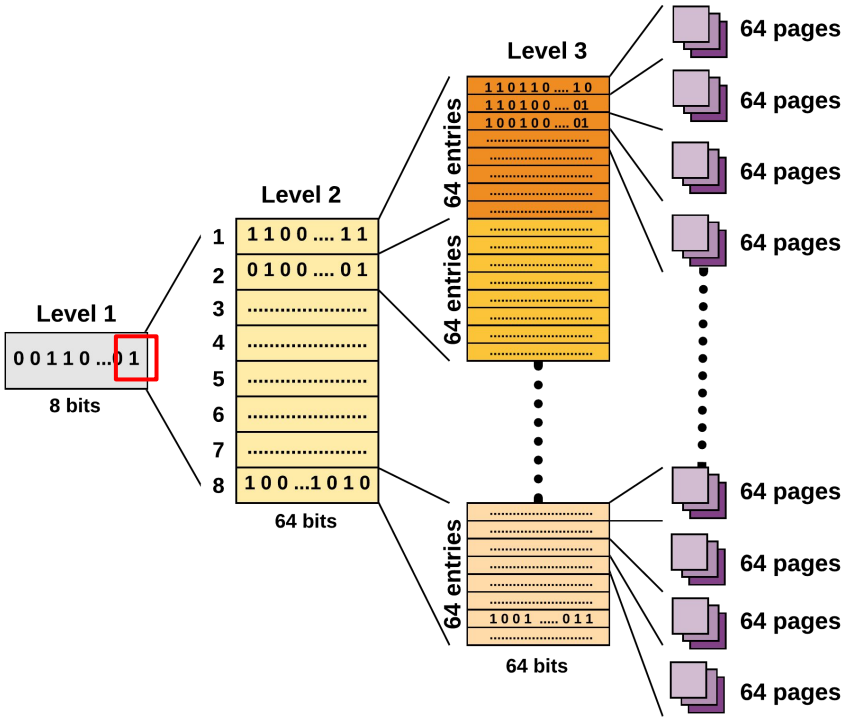
⇒ Stores the location of unallocated DRAM Cache pages



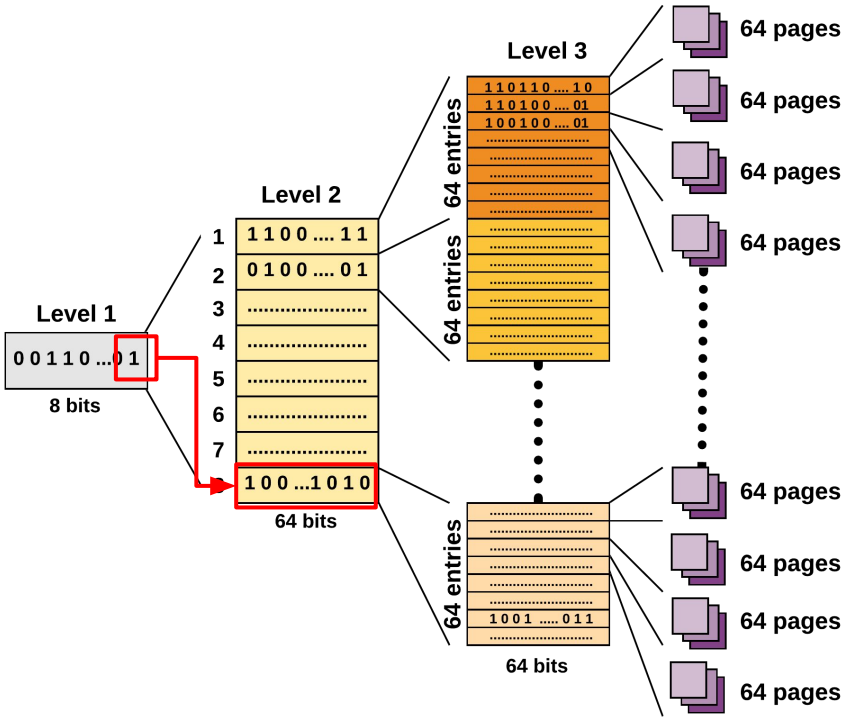
Empty Page Classifier (EPC)



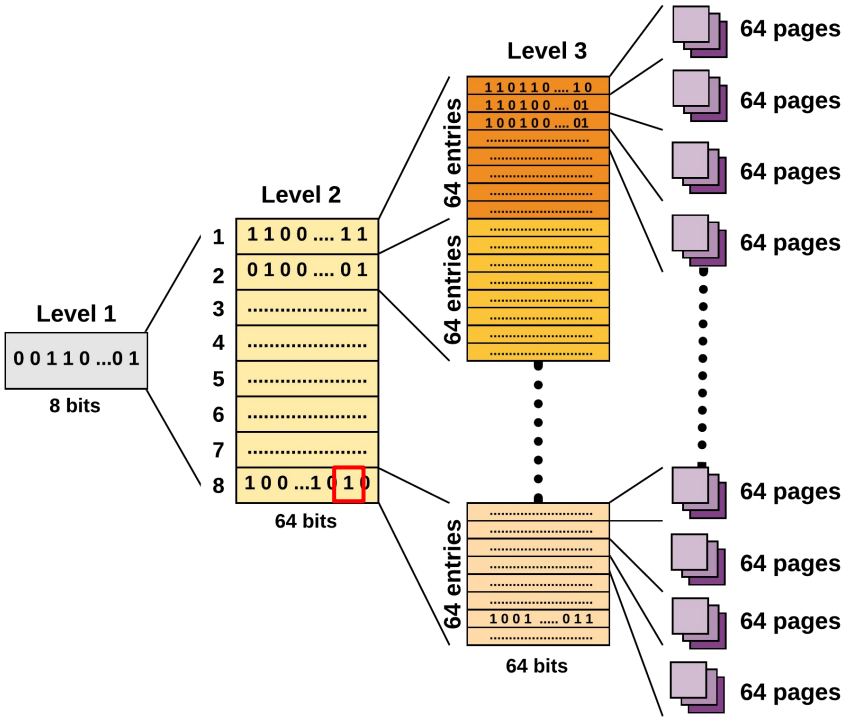
Empty Page Classifier (EPC)



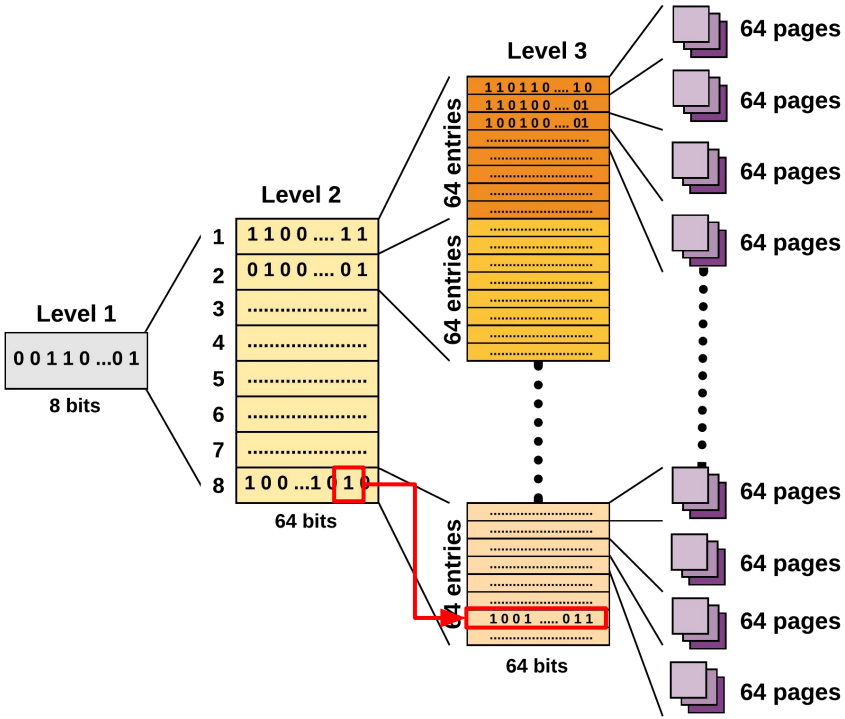
Empty Page Classifier (EPC)



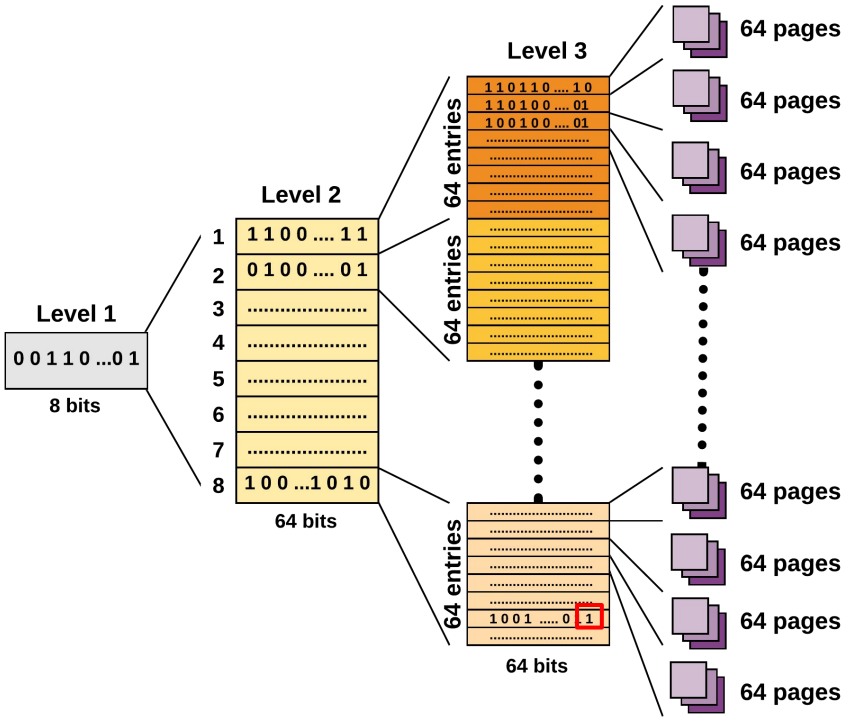
Empty Page Classifier (EPC)



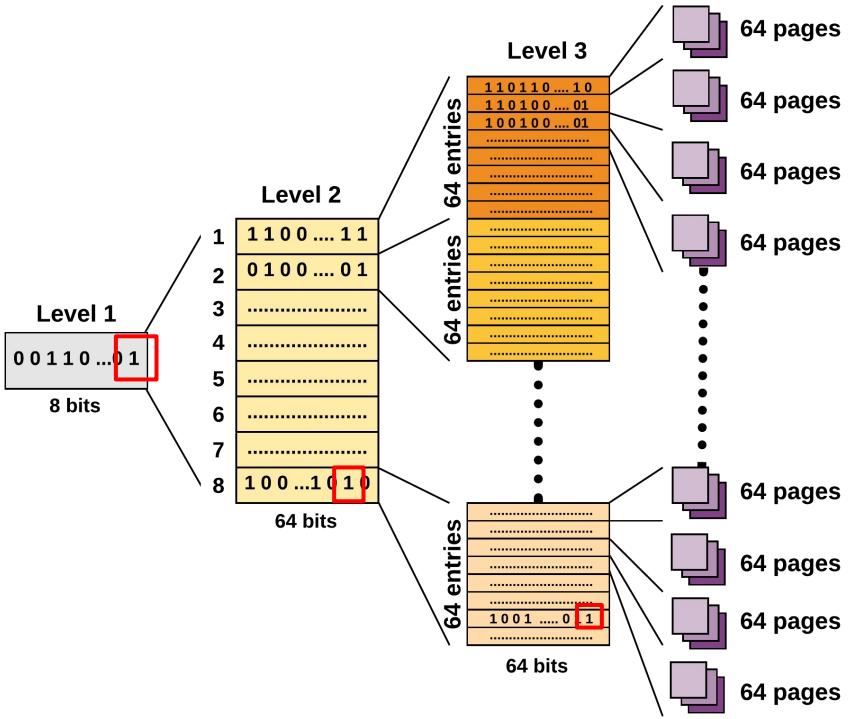
Empty Page Classifier (EPC)



Empty Page Classifier (EPC)



Empty Page Classifier (EPC)



Page Number =

(4096 ✕ Level 1 index) +

(64 ✕ Level 2 index) +

Level 3 index

Identifying type of data in DRAM Cache

Identifying type of data in DRAM Cache

A DRAM Cache location might be

⇒ Prefetched page

Identifying type of data in DRAM Cache

A DRAM Cache location might be

⇒ Prefetched page

⇒ Alloy Cache Page

Identifying type of data in DRAM Cache

A DRAM Cache location might be

⇒ Prefetched page

⇒ Alloy Cache Page

⇒ Empty

Identifying type of data in DRAM Cache

A DRAM Cache location might be

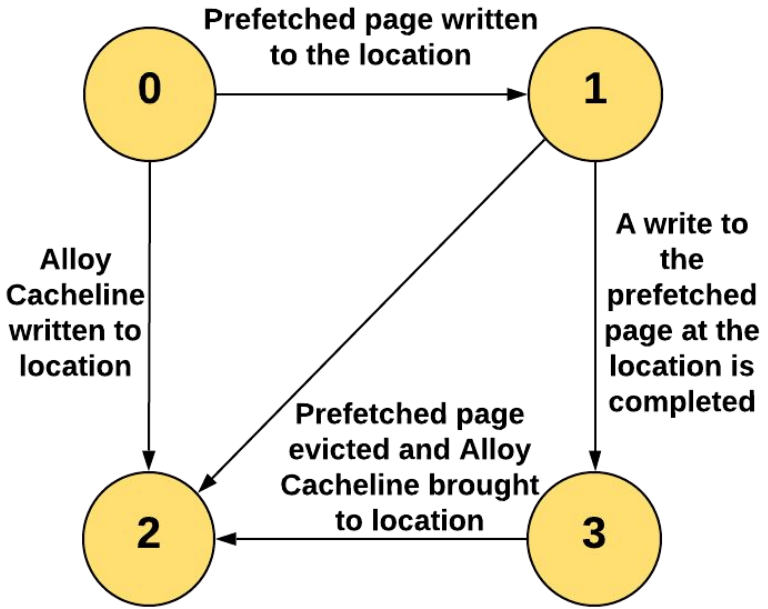
⇒ Prefetched page

⇒ Alloy Cache Page

⇒ Empty

Need to distinguish them to ensure correctness

Identifying type of data in DRAM Cache



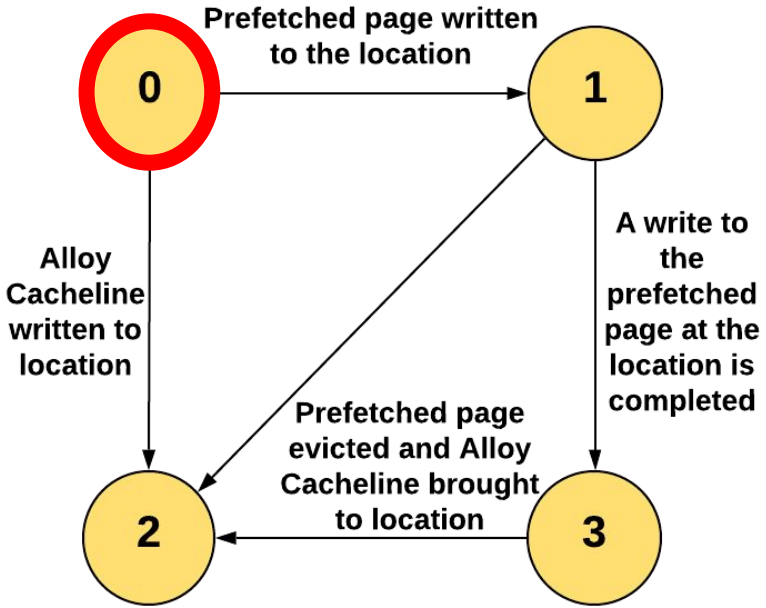
State 0: Empty Location

State 1: Clean Prefetched Page

State 2: Alloy Cache Page

State 3: Dirty Prefetched Page

Identifying type of data in DRAM Cache



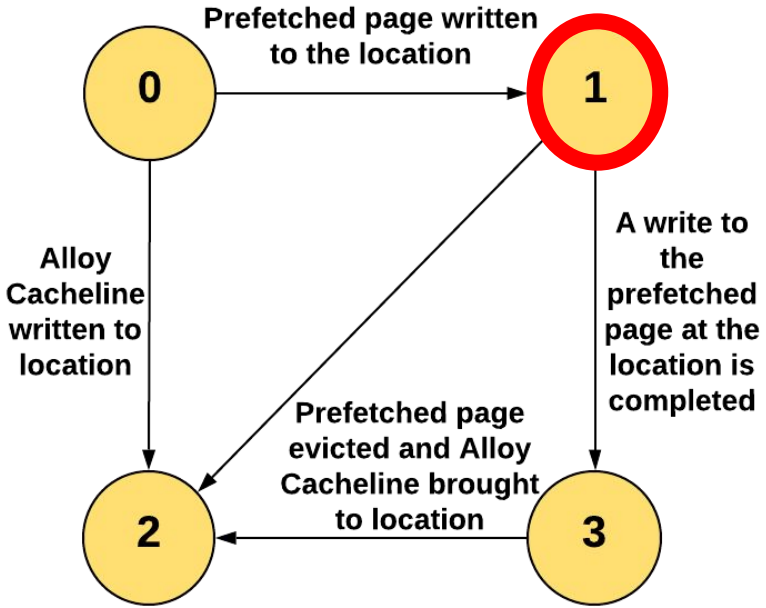
State 0: Empty Location

State 1: Clean Prefetched Page

State 2: Alloy Cache Page

State 3: Dirty Prefetched Page

Identifying type of data in DRAM Cache



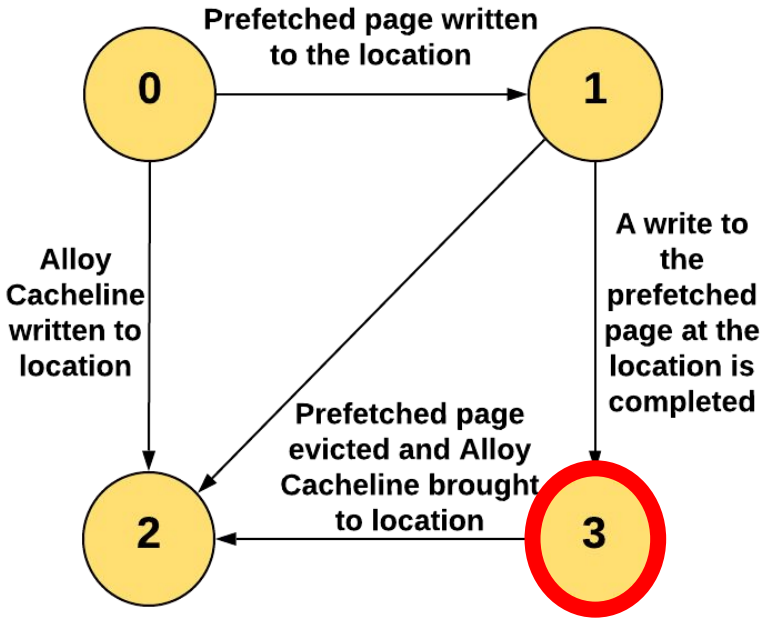
State 0: Empty Location

State 1: Clean Prefetched Page

State 2: Alloy Cache Page

State 3: Dirty Prefetched Page

Identifying type of data in DRAM Cache



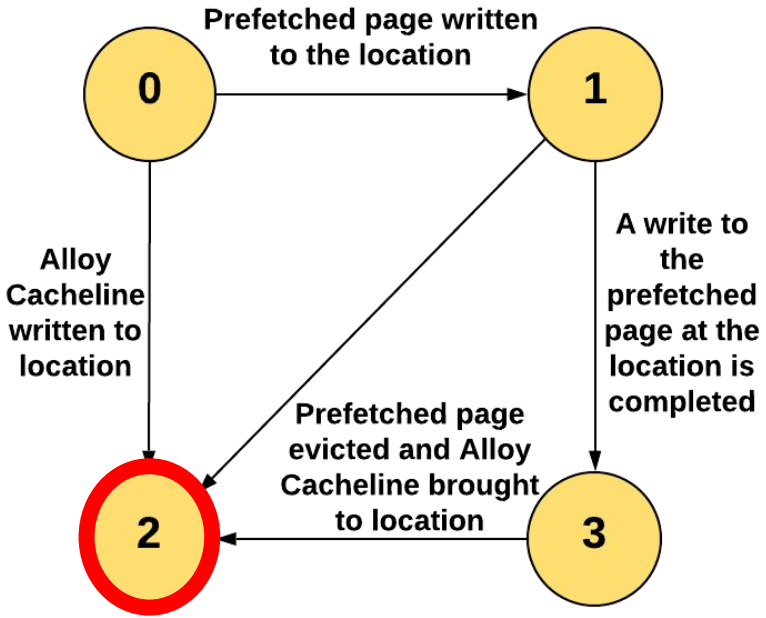
State 0: Empty Location

State 1: Clean Prefetched Page

State 2: Alloy Cache Page

State 3: Dirty Prefetched Page

Identifying type of data in DRAM Cache



State 0: Empty Location

State 1: Clean Prefetched Page

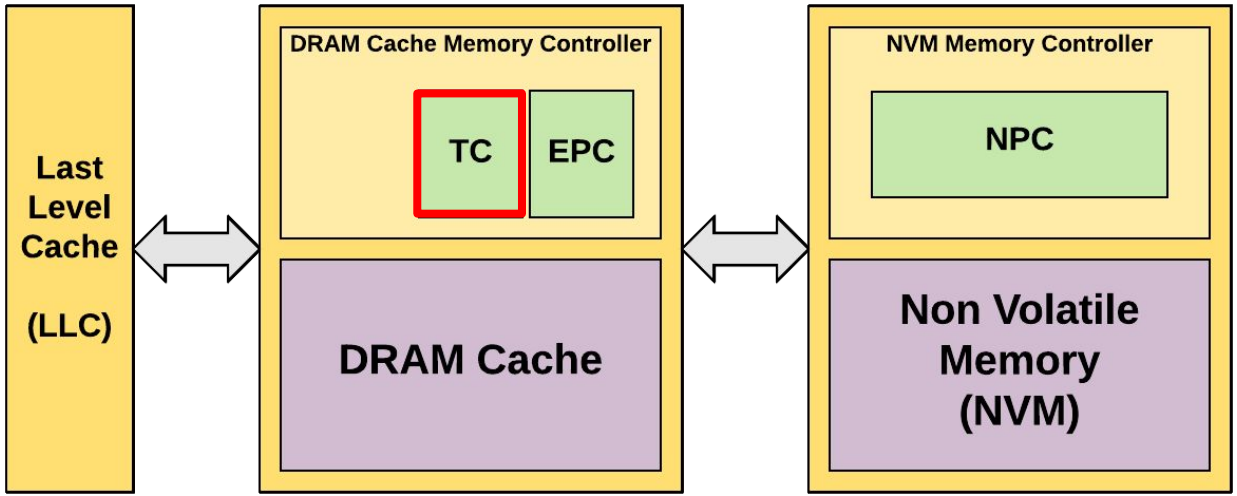
State 2: Alloy Cache Page

State 3: Dirty Prefetched Page

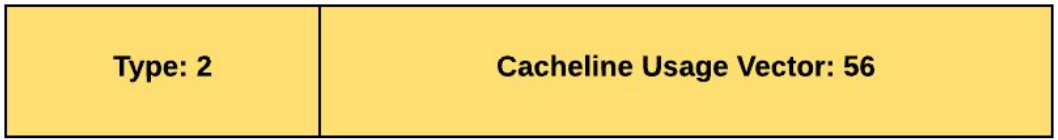
Identifying type of data in DRAM Cache

Type Classifier (TC)

⇒ Stores the state of the DRAM Cache location

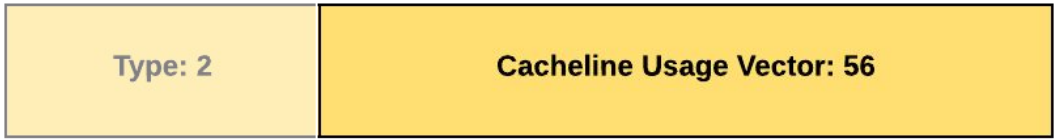


Type Classifier Entry



Type Classifier Entry

Type Classifier Entry

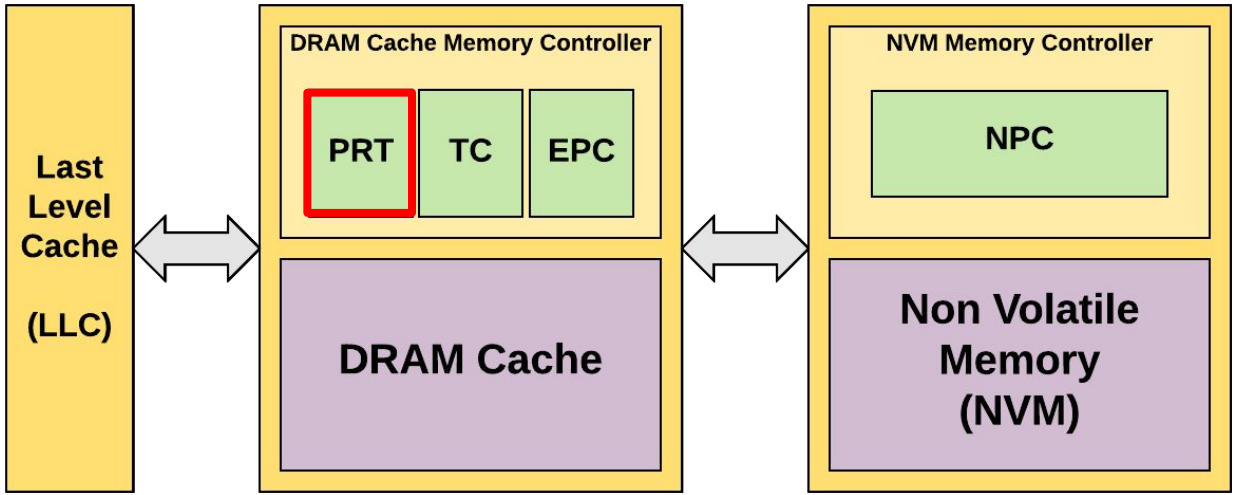


Checking if data is in a prefetched page

Checking if data is in a prefetched page

Page Redirection Table (PRT)

⇒ Hash Table storing tags of prefetched data



Page Redirection Table Entry

D : Max number of pages that can be present

in DRAM Cache

Page Redirection Table Entry

D : Max number of pages that can be present in DRAM Cache

Page Redirection Table Entry

D : Max number of pages that can be present

in DRAM Cache

Page Redirection Table Entry

D : Max number of pages that can be present

in DRAM Cache

Outline of the Presentation

- Background
- Insights
- Prefetcher Design
- Evaluation
- Future Work

Evaluation

ZSim + NVMain

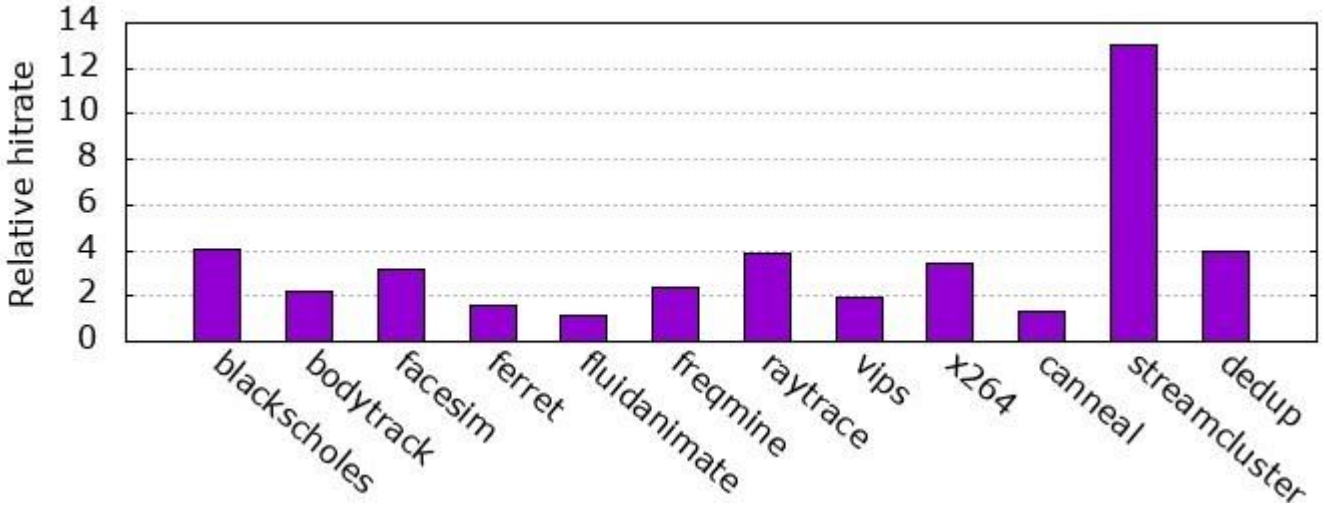
⇒ 1 GB Alloy Cache, 64 GB Phase Change Memory

⇒ 8 core, 2.6 GHz processor

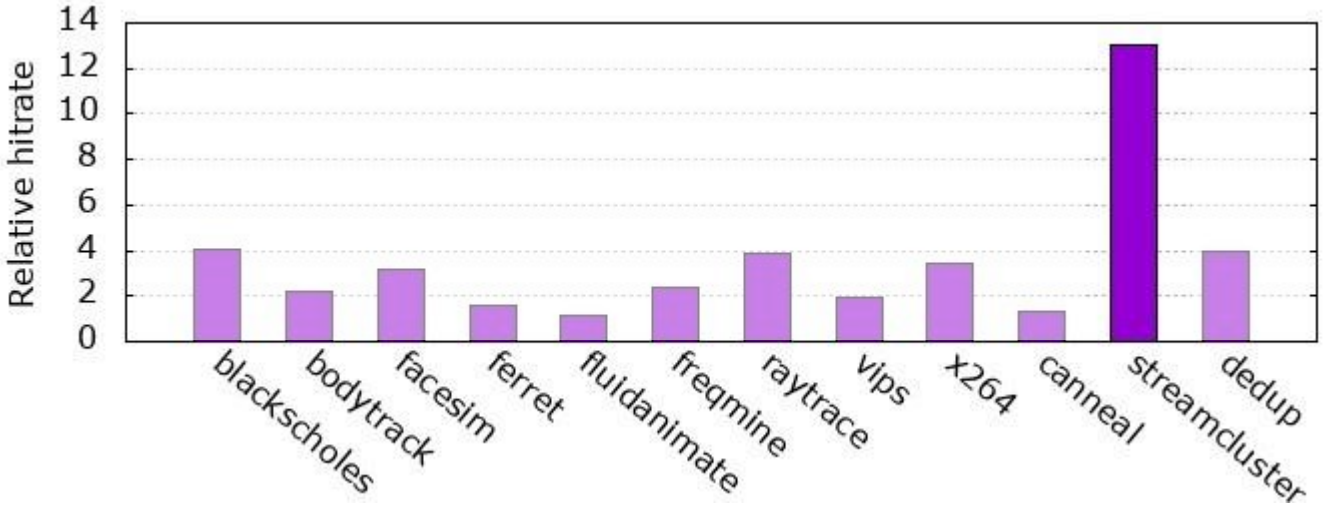
⇒ Use CACTI for access latency of structures

⇒ PARSEC benchmark

Evaluation

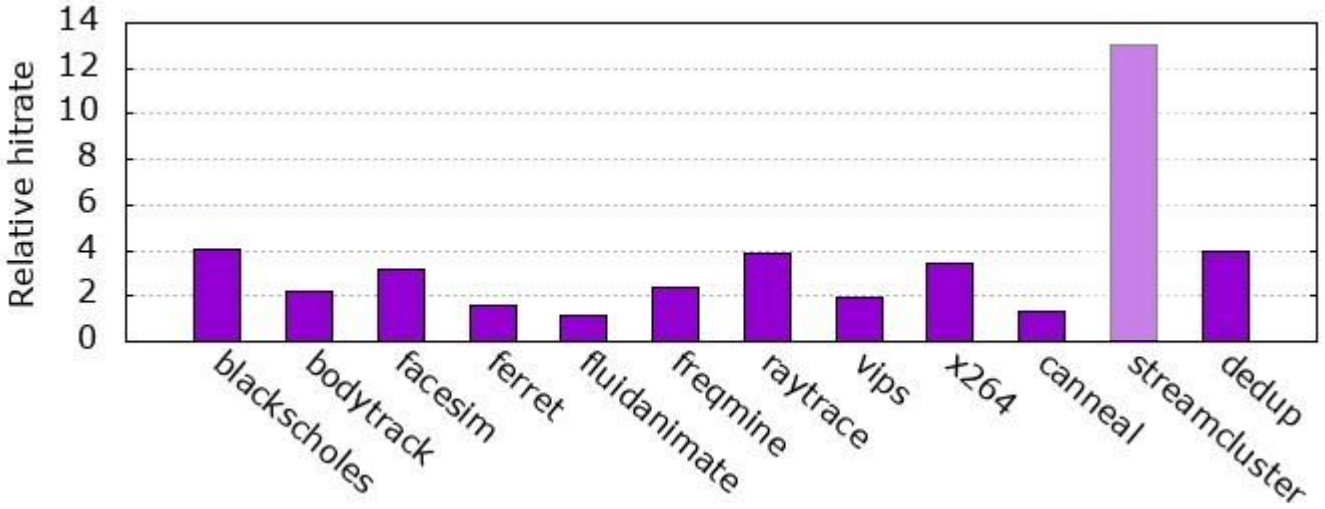

Relative hitrate
14
12
10
8
6
4
2
0
blackscholes
bodytrack
facesim
ferret
fluidanimate
freqmine
raytrace
vips
x264
canneal
streamcluster
dedup

Evaluation



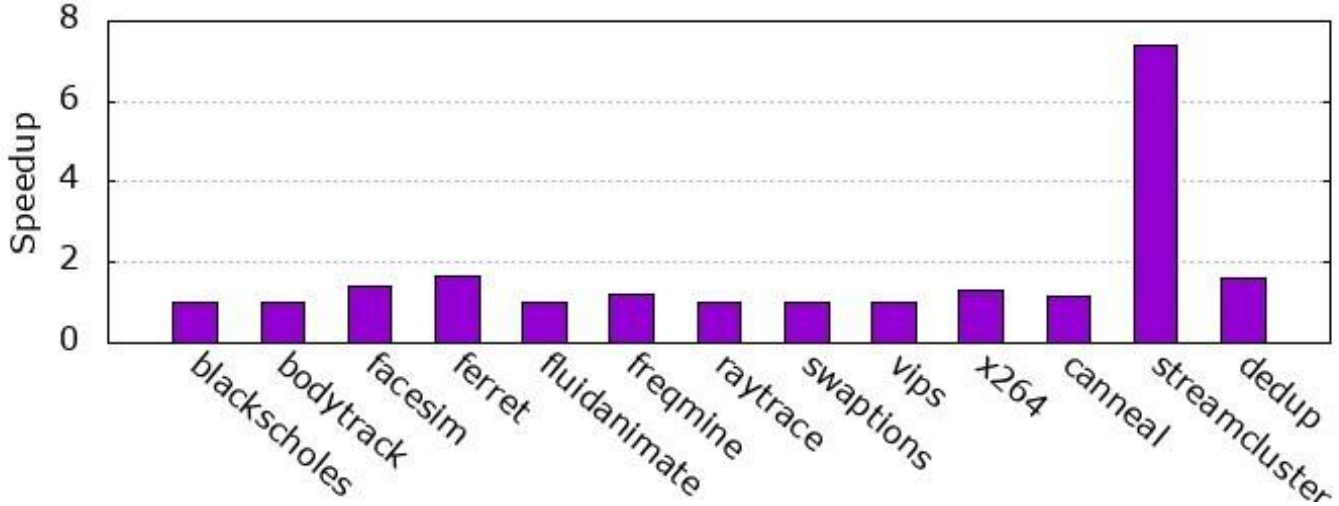
Sequential access behavior

Evaluation

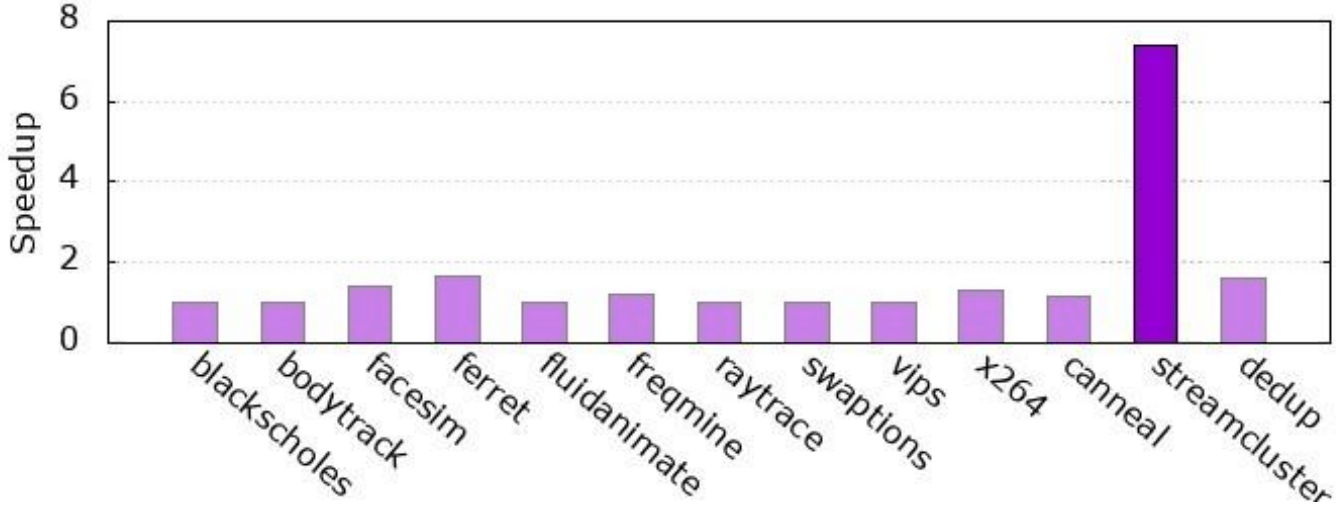


1.5×-4× improvement

Evaluation

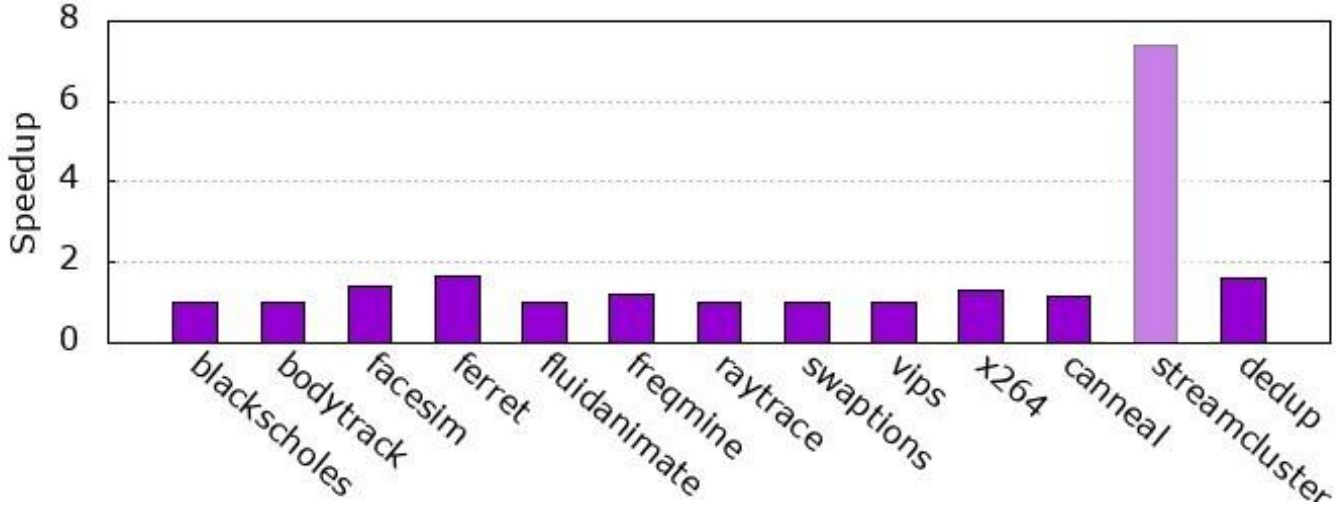

Speedup
8
6
4
2
0
blackscholes
bodytrack
facesim
ferret
fluidanimate
freqmine
raytrace
swaptions
vips
x264
canneal
streamcluster
dedup

Evaluation



7× speedup

Evaluation



16-40% higher IPC

Outline of the Presentation

- Background
- Insights
- Prefetcher Design
- Evaluation
- Future Work

Future Work

Evaluate our prefetcher on

⇒ Memory-intensive SPEC workloads

⇒ Graph workloads having irregular memory access patterns

⇒ Compare with similar recent works

Key Takeaways

- Prefetch at page granularity to exploit page-level spatial locality.
- Place prefetched page in DRAM Cache to improve its utilization
- We observe 16-40% increase in IPC on PARSEC.

Link to Paper:

Contact Us:
gohil.varun@iitgn.ac.in
manu.awasthi@ashoka.edu.in