34th IEEE International Conference on Computer Design

ICCD 2016

Dynamic Prefetcher Reconfiguration for Diverse Memory Architectures

Junghoon Lee, **Taehoon Kim,** and Jaehyuk Huh*

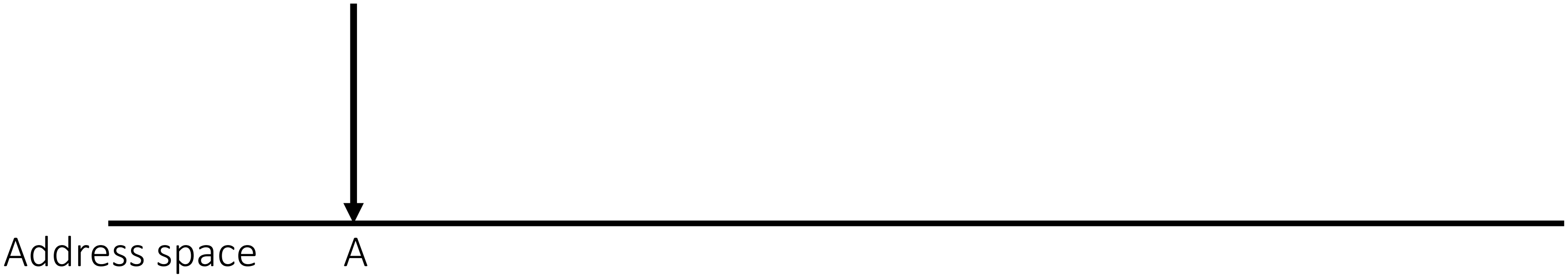
SAMSUNG ADVANCED INSTITUTE OF TECHNOLOGY

KAIST School of Computing

Prefetching

- Stream prefetcher
  - Stream: a sequence of consecutive memory blocks
  - If any demand request accesses a block in a *stream*(from A to A+P), generate prefetch request A+P, A+P+1, … , A+P+N

- Parameters
  - Distance (P): how far future the prefetcher predicts
  - Degree (N): how many prefetch requests are generated



Prefetching

- Stream prefetcher
  - Stream: a sequence of consecutive memory blocks
  - If any demand request accesses a block in a *stream*(from A to A+P), generate prefetch request A+P, A+P+1, … , A+P+N

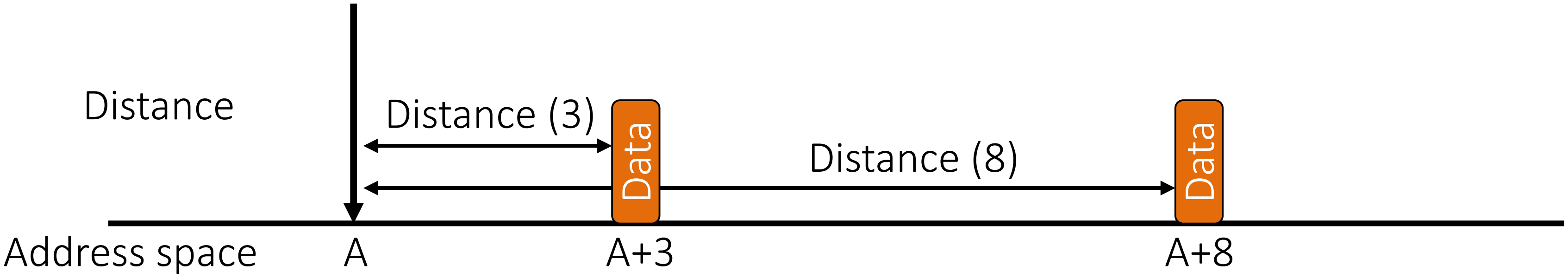
- Parameters
  - Distance (P): how far future the prefetcher predicts
  - Degree (N): how many prefetch requests are generated

Distance
Address space
A

Prefetching

- Stream prefetcher
  - Stream: a sequence of consecutive memory blocks
  - If any demand request accesses a block in a *stream*(from A to A+P), generate prefetch request A+P, A+P+1, … , A+P+N

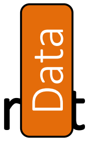
- Parameters
  - Distance (P): how far future the prefetcher predicts
  - Degree (N): how many prefetch requests are generated

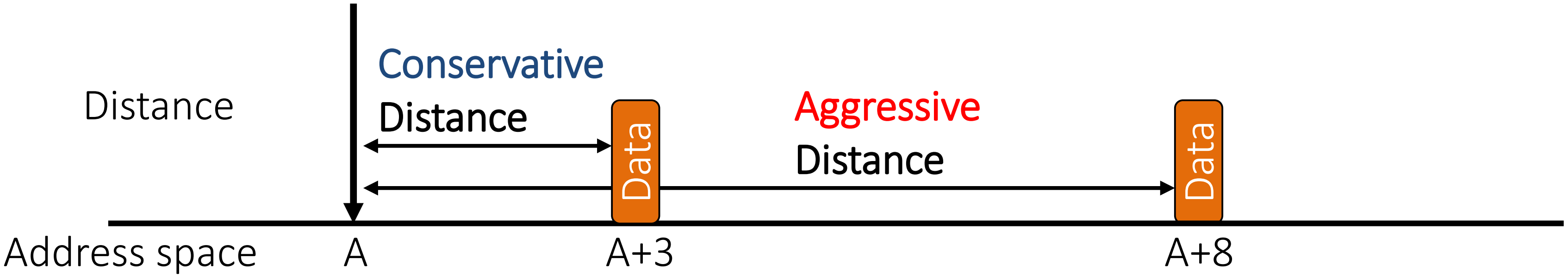


Prefetching

- Stream prefetcher
  - Stream: a sequence of consecutive memory blocks
  - If any demand request accesses a block in a *stream*(from A to A+P), generate prefetch request A+P, A+P+1, … , A+P+N

- Parameters
  - Distance (P): how far future the prefetcher predicts
  - Degree (N): how many prefetch requests are generated

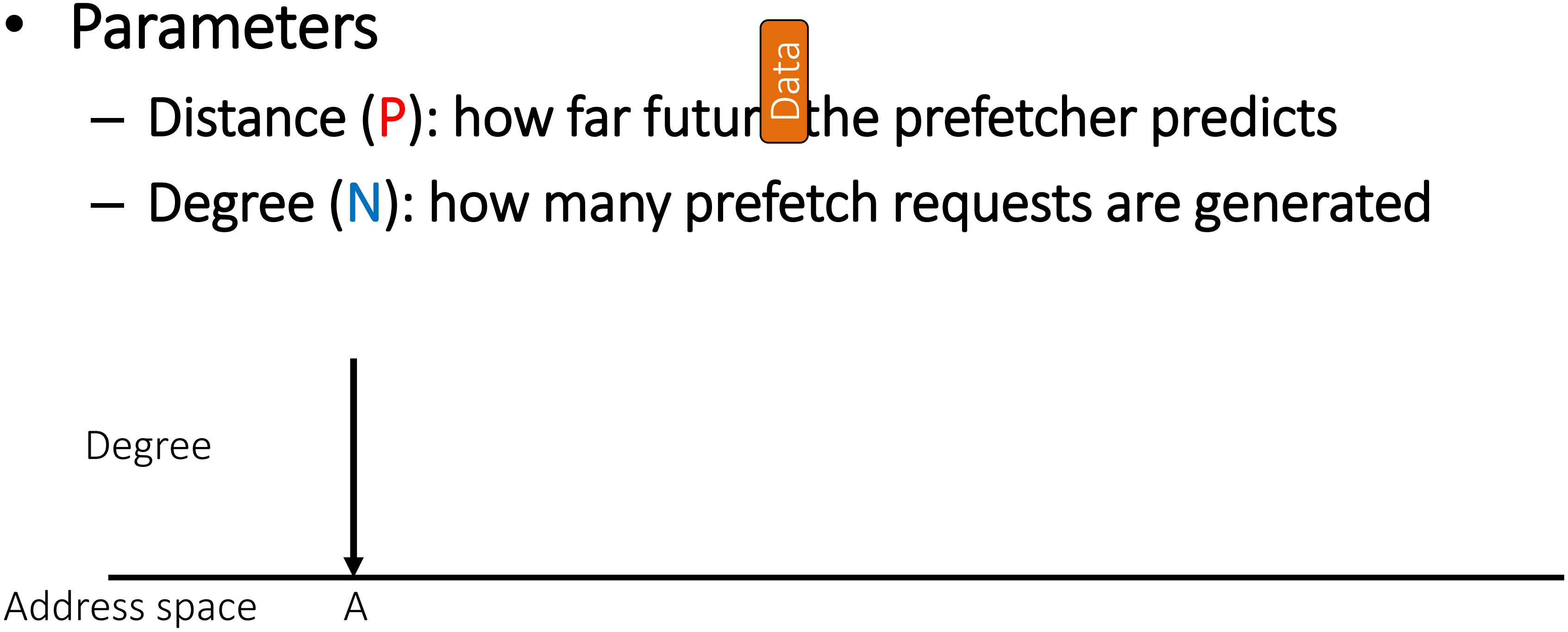




Prefetching

- Stream prefetcher
  - Stream: a sequence of consecutive memory blocks
  - If any demand request accesses a block in a *stream*(from A to A+P), generate prefetch request A+P, A+P+1, … , A+P+N

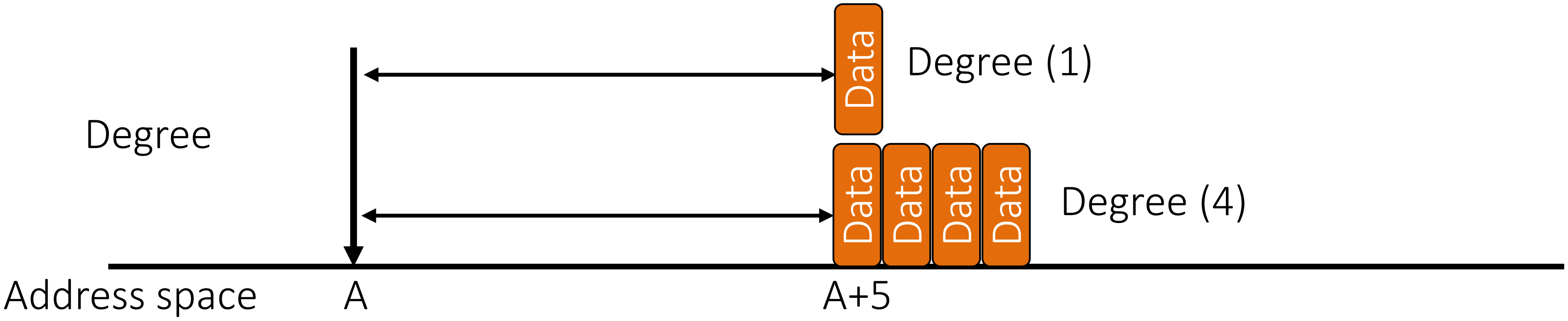
- Parameters
  - Distance (P): how far futur the prefetcher predicts
  - Degree (N): how many prefetch requests are generated



Prefetching

- Stream prefetcher
  - Stream: a sequence of consecutive memory blocks
  - If any demand request accesses a block in a *stream*(from A to A+P), generate prefetch request A+P, A+P+1, … , A+P+N

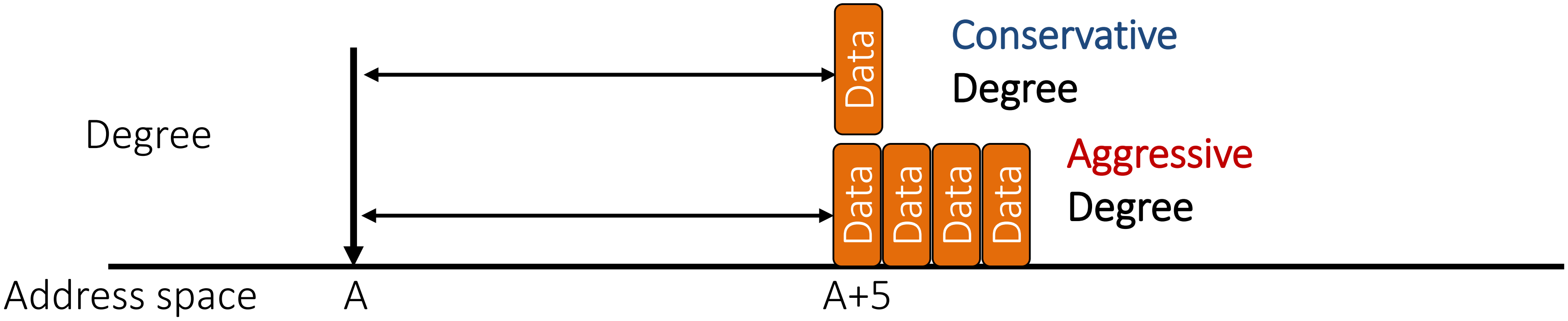
- Parameters
  - Distance (P): how far futur Data the prefetcher predicts
  - Degree (N): how many prefetch requests are generated



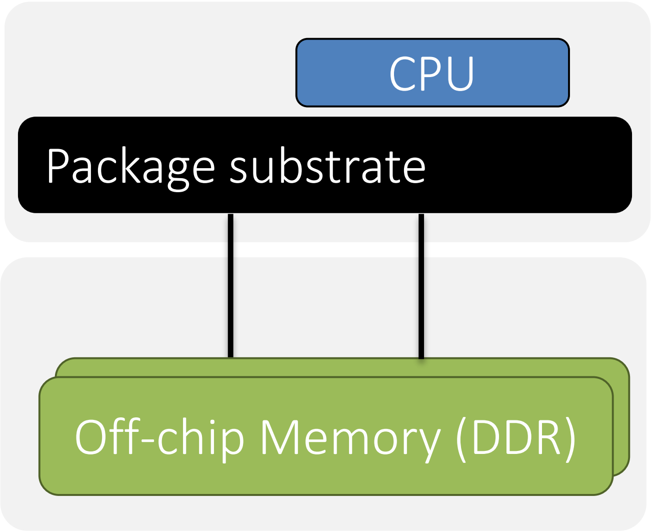
Prefetching

- Stream prefetcher
  - Stream: a sequence of consecutive memory blocks
  - If any demand request accesses a block in a *stream*(from A to A+P), generate prefetch request A+P, A+P+1, … , A+P+N

- Parameters
  - Distance (P): how far futur the prefetcher predicts
  - Degree (N): how many prefetch requests are generated



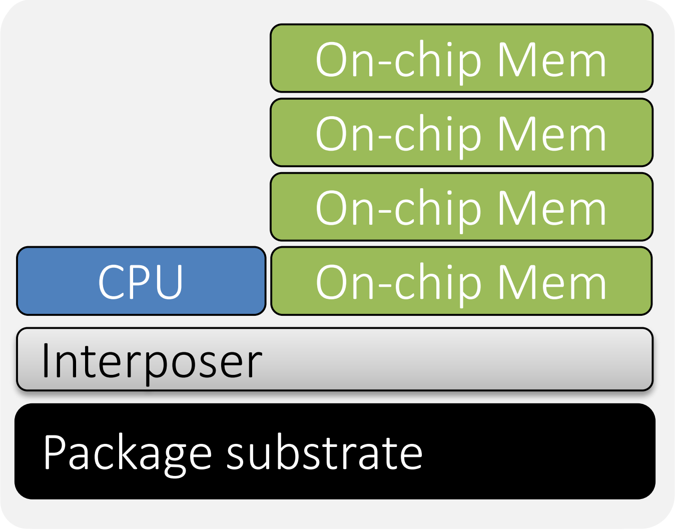
Diverse Memory Architecture



- Traditional memory architecture
  - DDR: one dominant memory type
  - Relatively predictable bandwidth

[1] Loh et al. ISCA 2008
[2] Qureshi et al. ISCA 2009
[3] Chou et al. MICRO 2014

Diverse Memory Architecture



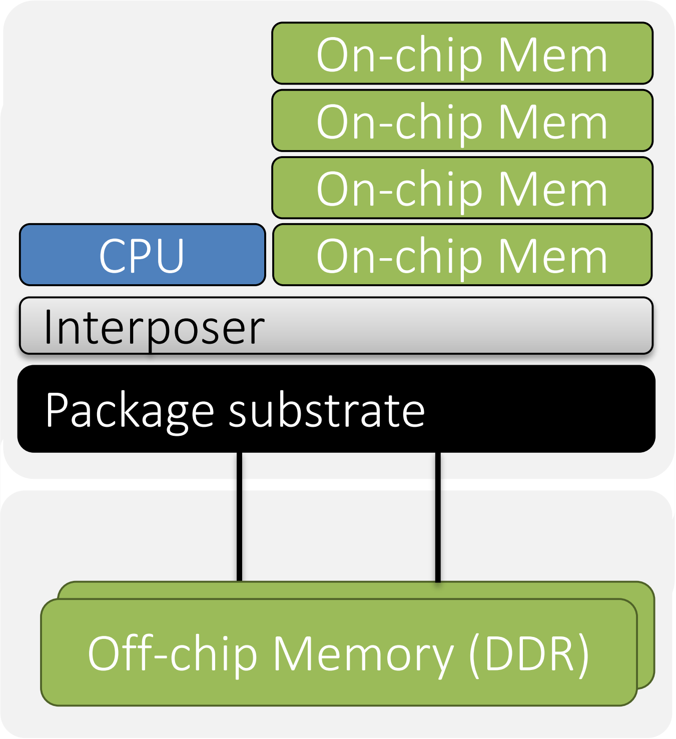
- Traditional memory architecture
  - DDR: one dominant memory type
  - Relatively predictable bandwidth
- Memory heterogeneity
  - DDR, HBM[1], non-volatile memory[2], hybrid memory[3]
  - Wide range of bandwidth/latency

[1] Loh et al. ISCA 2008

[2] Qureshi et al. ISCA 2009

[3] Chou et al. MICRO 2014

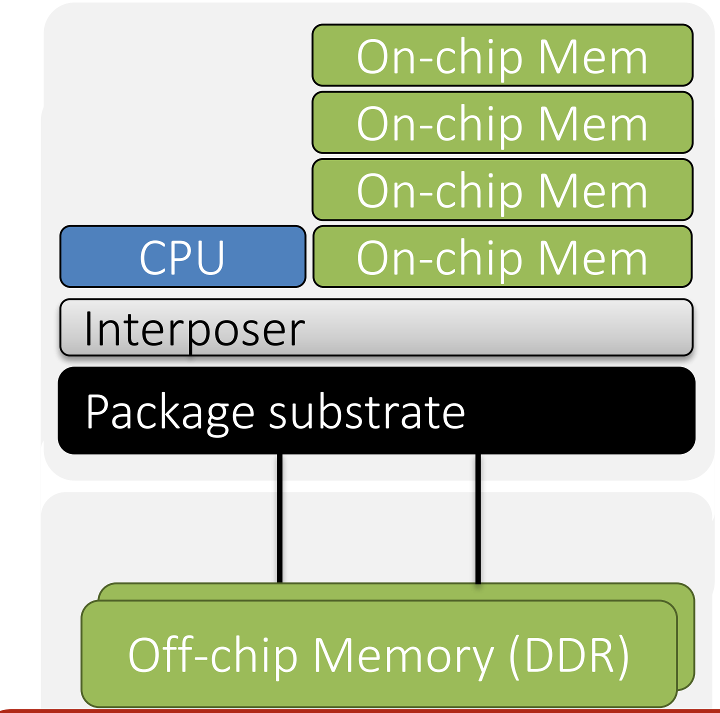
Diverse Memory Architecture



- Traditional memory architecture
  - DDR: one dominant memory type
  - Relatively predictable bandwidth
- Memory heterogeneity
  - DDR, HBM[1], non-volatile memory[2], hybrid memory[3]
  - Wide range of bandwidth/latency

[1] Loh et al. ISCA 2008
[2] Qureshi et al. ISCA 2009
[3] Chou et al. MICRO 2014

Diverse Memory Architecture



- Traditional memory architecture
  - DDR: one dominant memory type
  - Relatively predictable bandwidth
- Memory heterogeneity
  - DDR, HBM[1], non-volatile memory[2], hybrid memory[3]
  - Wide range of bandwidth/latency

Prefetcher should consider various memory characteristics

[1] Loh et al. ISCA 2008

[2] Qureshi et al. ISCA 2009

[3] Chou et al. MICRO 2014

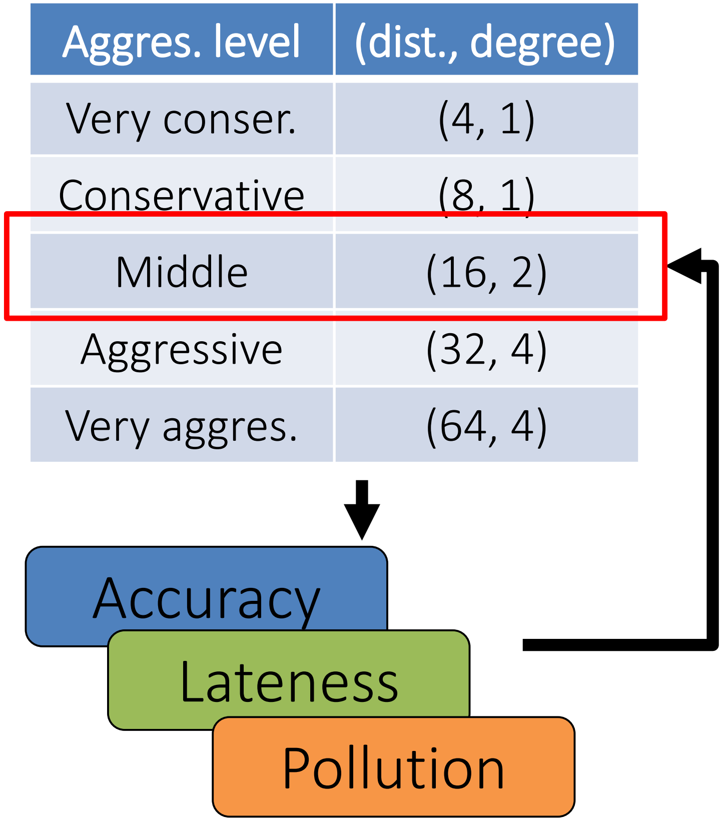
Prior Work

Aggres. level	(dist., degree)
Very conser.	(4, 1)
Conservative	(8, 1)
Middle	(16, 2)
Aggressive	(32, 4)
Very aggres.	(64, 4)

- Feedback-directed prefetching [4]
  - Use stream prefetcher: distance & degree
  - Choose one of five aggressive levels
  - Consider application's memory bandwidth requirement
- Limitation
  - Five levels of pre-selected prefetch configurations
  - Consider DDR memory only

[4] Srinath et al. HPCA 2007

Prior Work



Aggres. level	(dist., degree)
Very conser.	(4, 1)
Conservative	(8, 1)
Middle	(16, 2)
Aggressive	(32, 4)
Very aggres.	(64, 4)

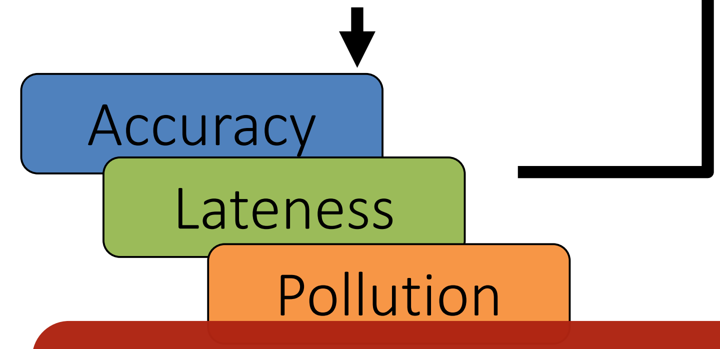
- Feedback-directed prefetching [4]
  - Use stream prefetcher: distance & degree
  - Choose one of five aggressive levels
  - Consider application’s memory bandwidth requirement

- Limitation
  - Five levels of pre-selected prefetch configurations
  - Consider DDR memory only

[4] Srinath et al. HPCA 2007

Prior Work

Aggres. level	(dist., degree)
Very conser.	(4, 1)
Conservative	(8, 1)
Middle	(16, 2)
Aggressive	(32, 4)
Very aggres.	(64, 4)

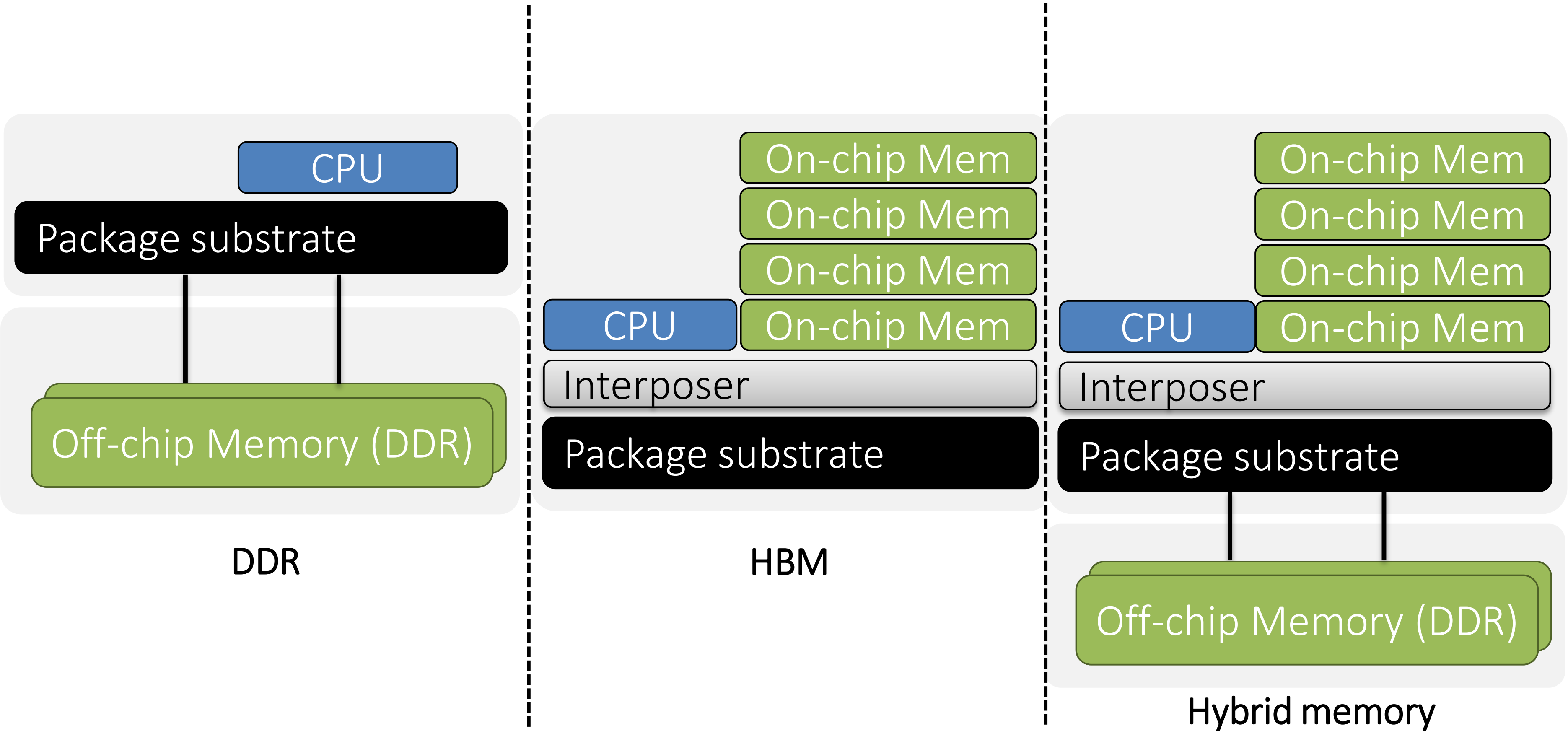


- Feedback-directed prefetching [4]
  - Use stream prefetcher: distance & degree
  - Choose one of five aggressive levels
  - Consider application's memory bandwidth requirement
- Limitation
  - Five levels of pre-selected prefetch configurations
  - Consider DDR memory only

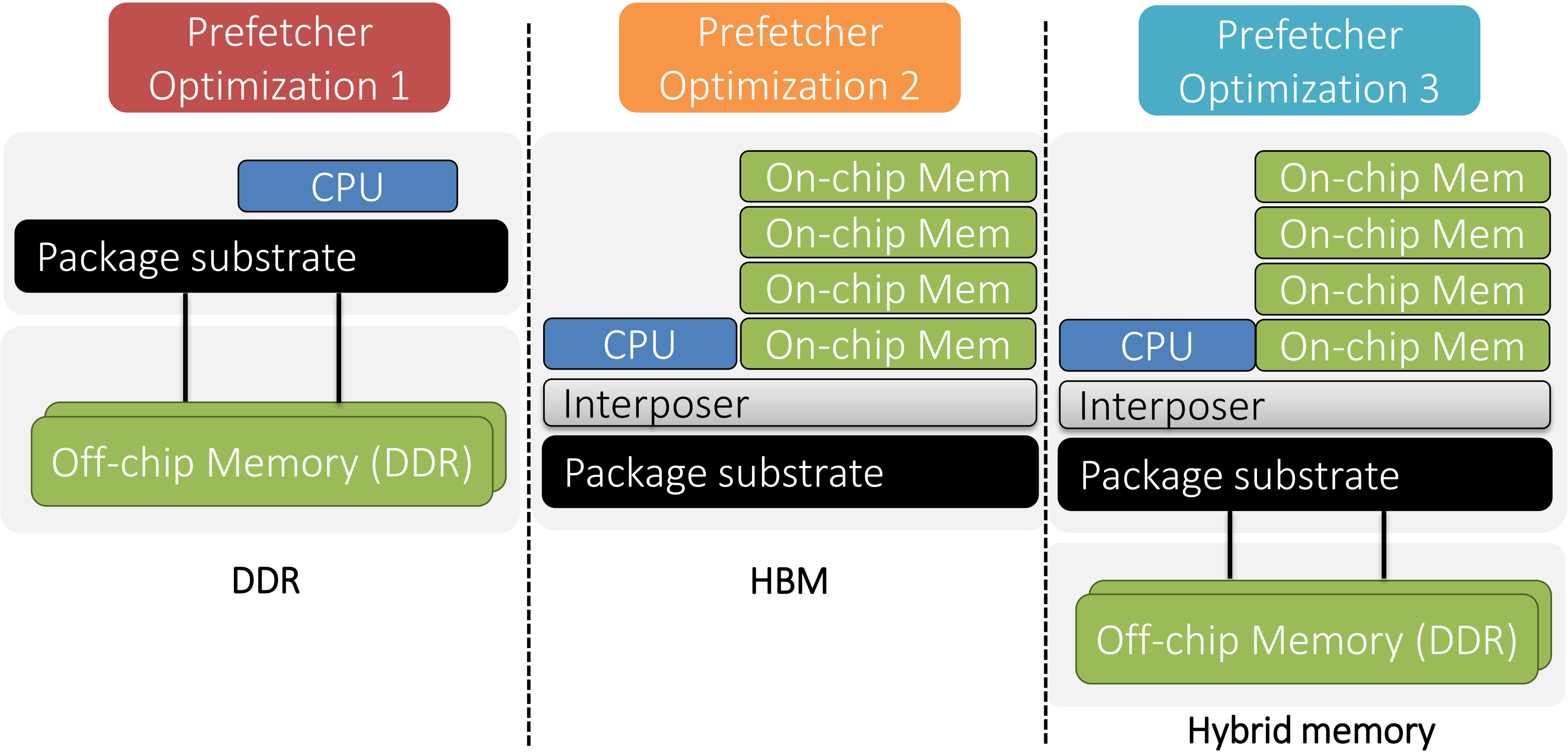
Only a small number of pre-selected configurations are not enough to cover the diversity of memory architectures

[4] Srinath et al. HPCA 2007

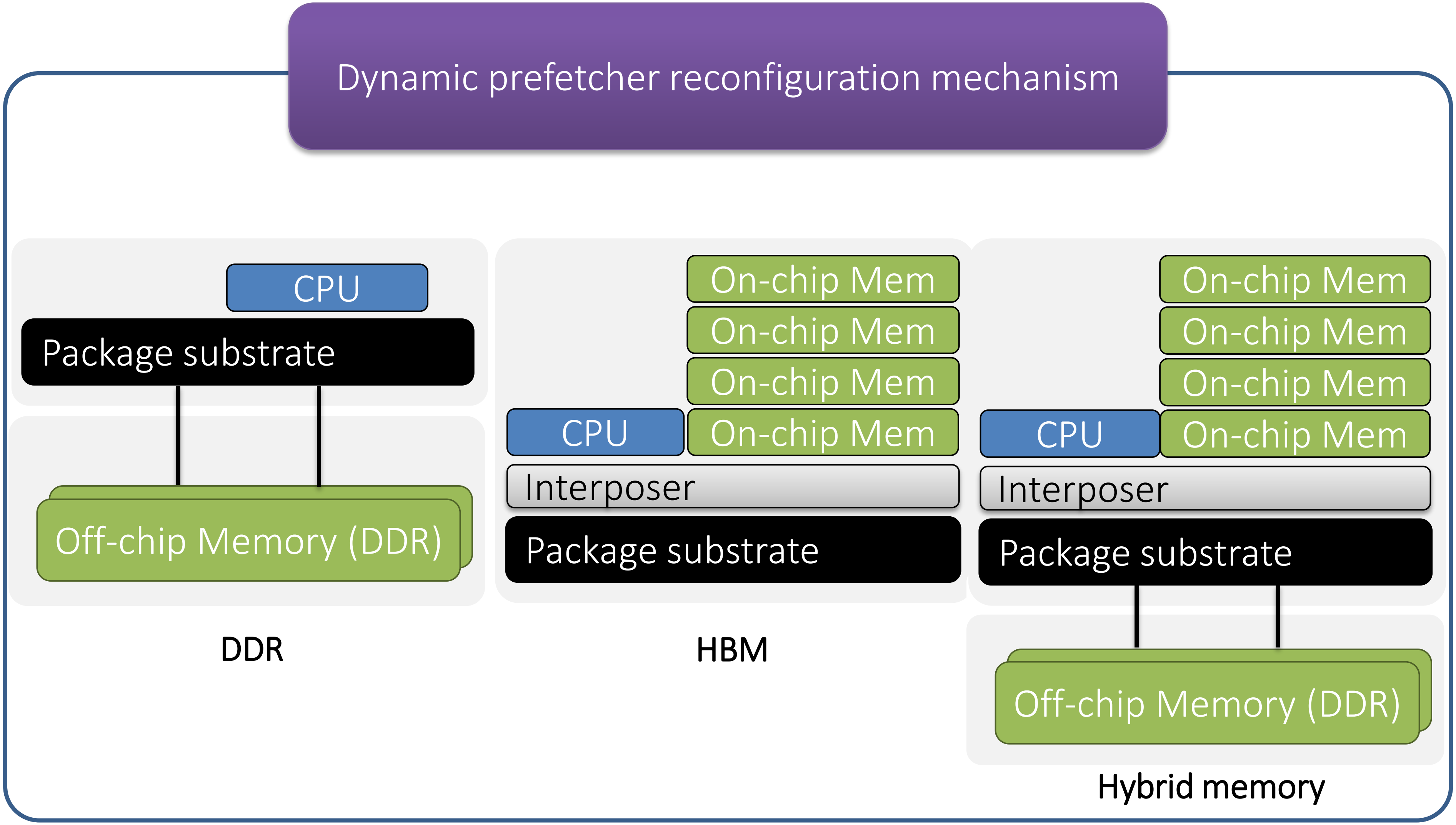
Dynamic Prefetcher



Dynamic Prefetcher



Dynamic Prefetcher

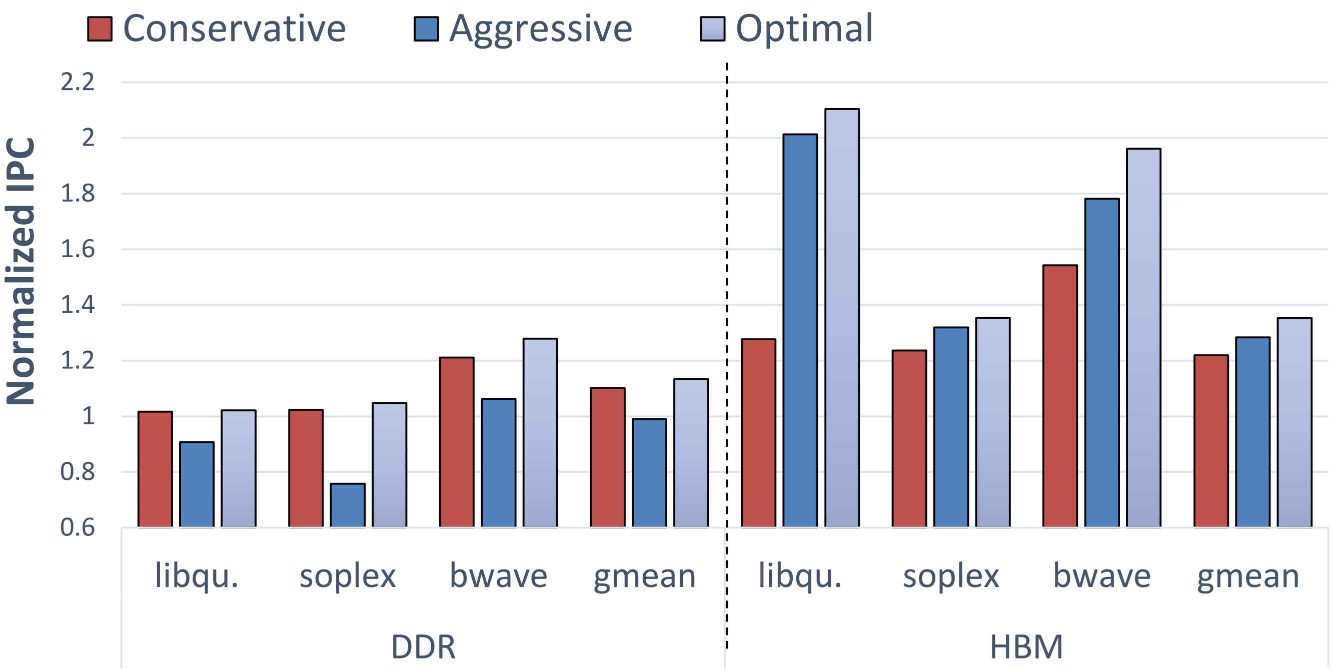

Dynamic prefetcher reconfiguration mechanism

Outline

- Motivation : the effect of available memory bandwidth on prefetcher designs
  - Effect on the aggressiveness of prefetcher
  - Dominant factor: distance vs degree
  - Cache pollution by prefetcher

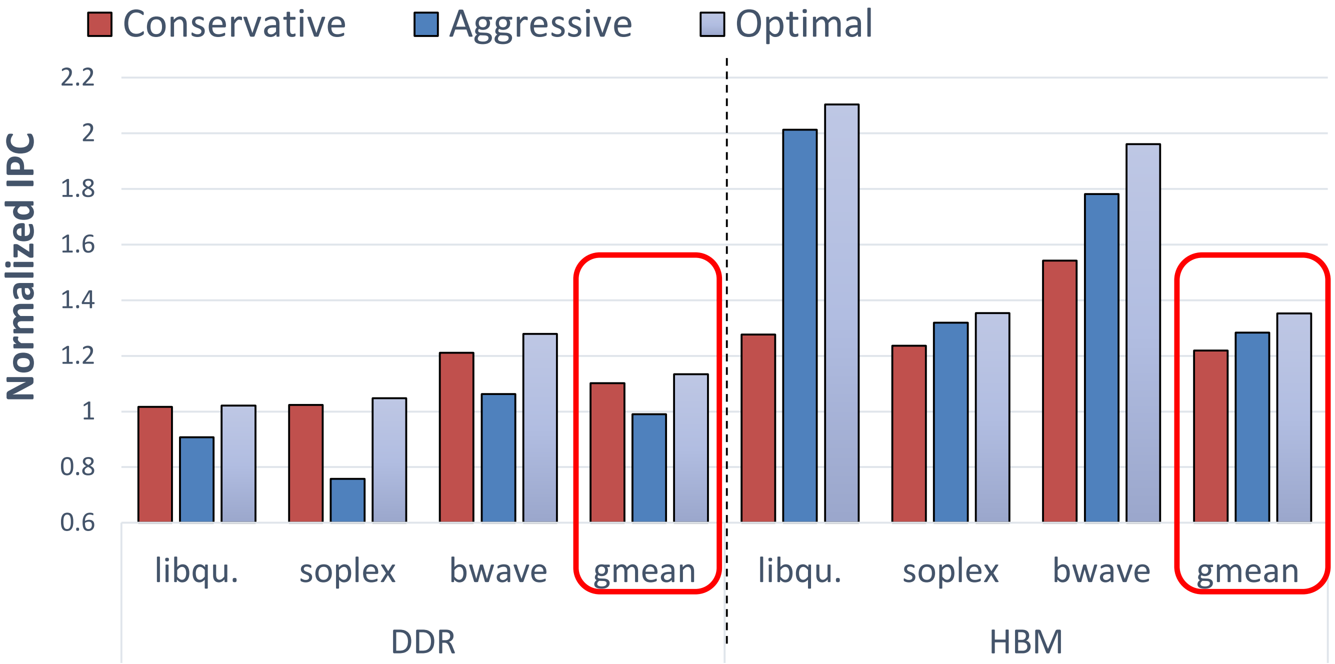
- Contributions
  - Propose a prefetcher reconfiguration mechanism
  - Propose a pollution mitigation mechanism

The Effect of Bandwidth on Prefetcher(1/2)



* conservative (8,1) and aggressive (8,64)

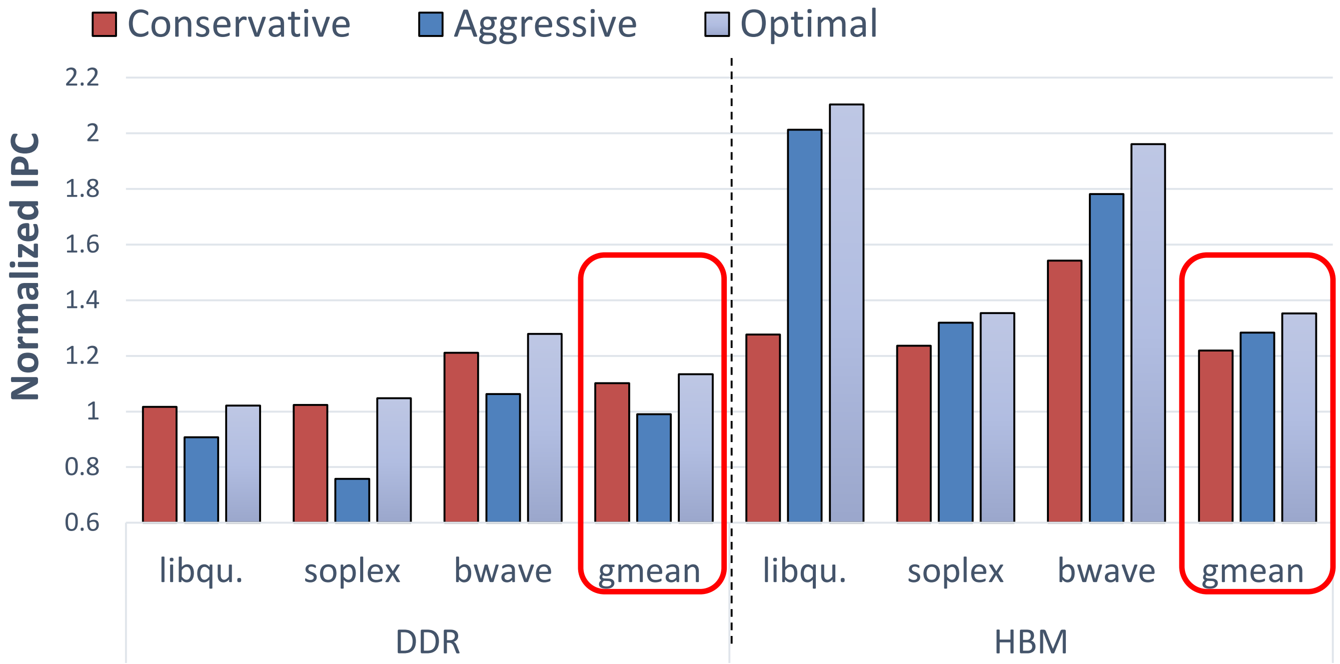
The Effect of Bandwidth on Prefetcher(1/2)



* conservative (8,1) and aggressive (8,64)

	Conservative	Aggressive
DDR	10%	-1%
HBM	20%	28%

The Effect of Bandwidth on Prefetcher(1/2)

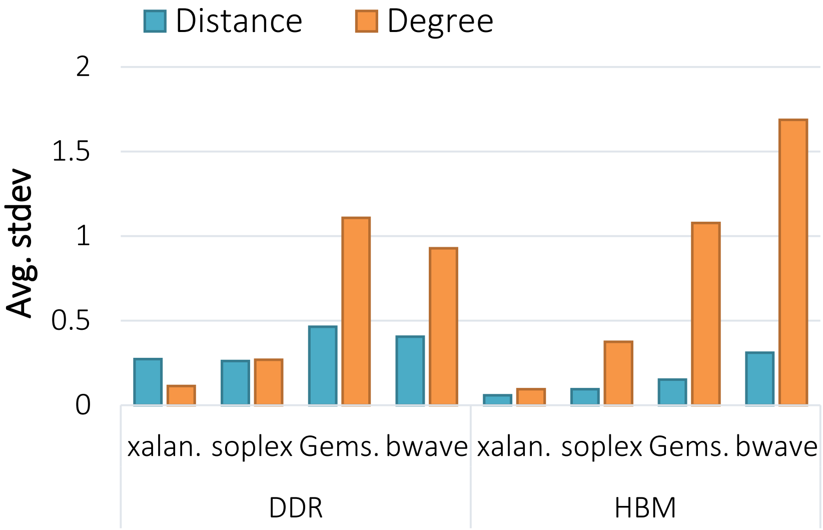


* conservative (8,1) and aggressive (8,64)

	Conservative	Aggressive
DDR	10%	-1%
HBM	20%	28%

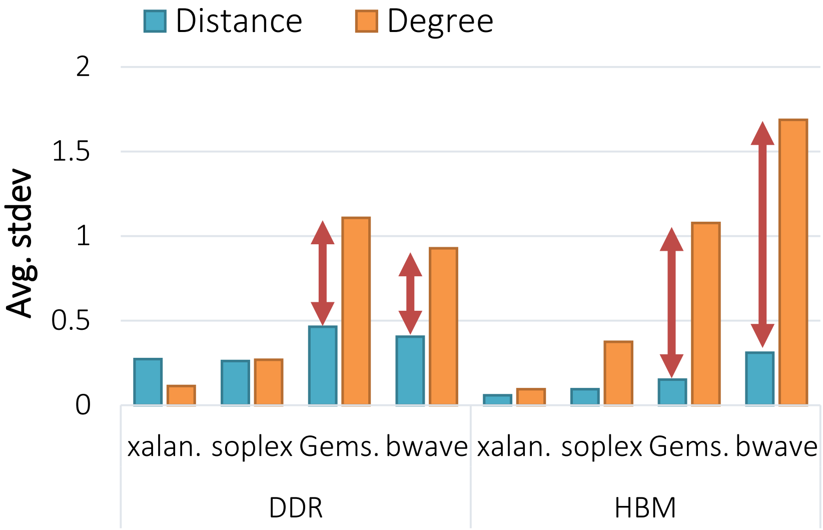
Observation 1:
The best prefetcher aggressiveness
differs for each memory type

The Effect of Bandwidth on Prefetcher(2/2)



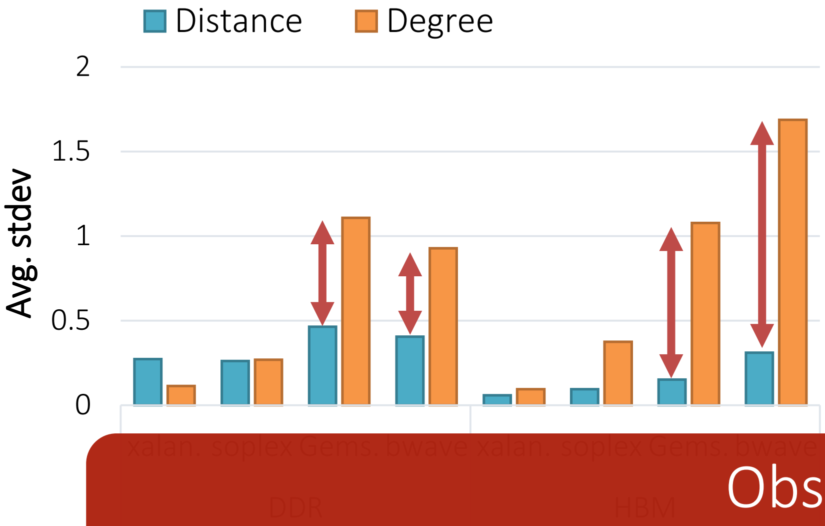
- Distance vs. degree
  - Performance variation is higher on degree

The Effect of Bandwidth on Prefetcher(2/2)



- Distance vs. degree
  - Performance variation is higher on degree

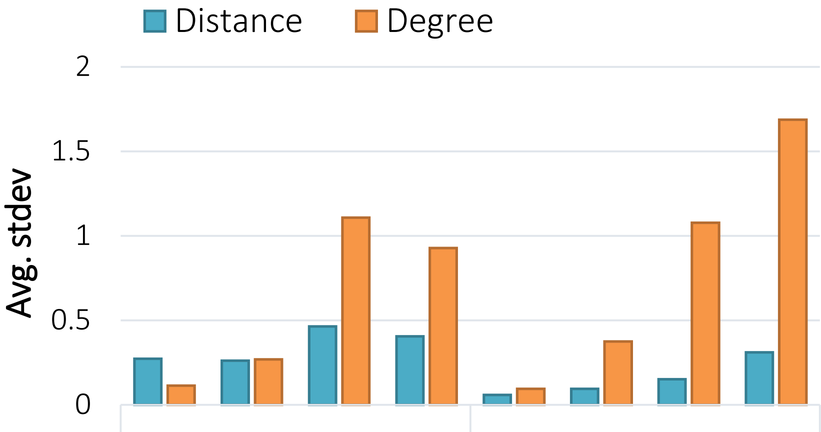
The Effect of Bandwidth on Prefetcher(2/2)



- Distance vs. degree
  - Performance variation is higher on degree

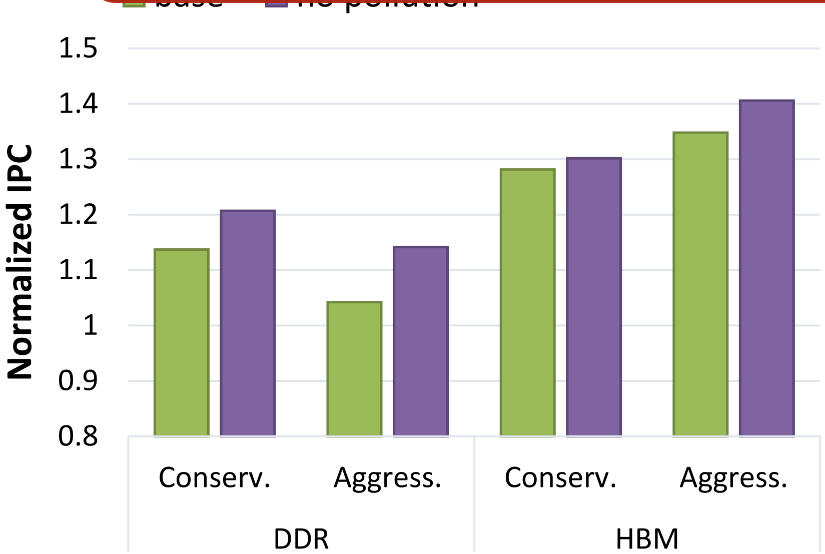
Observation 2:
Degree has a much higher impact on performance

The Effect of Bandwidth on Prefetcher(2/2)



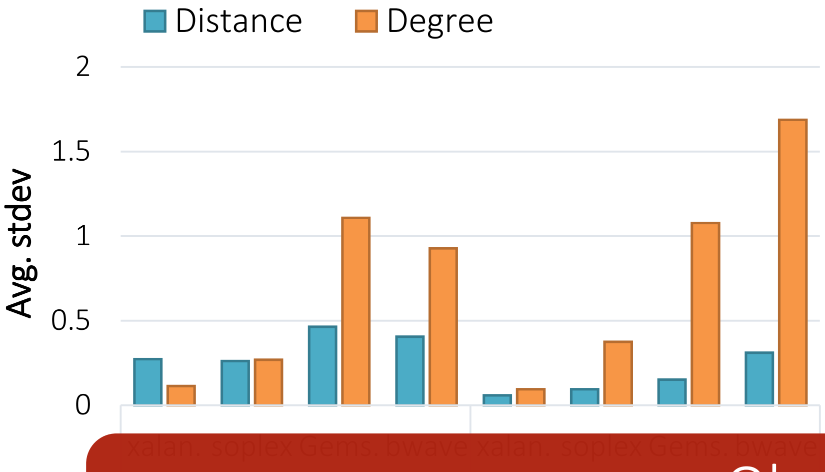
- Distance vs. degree
  - Performance variation is higher on degree

Observation 2:
Degree has a much higher impact on performance



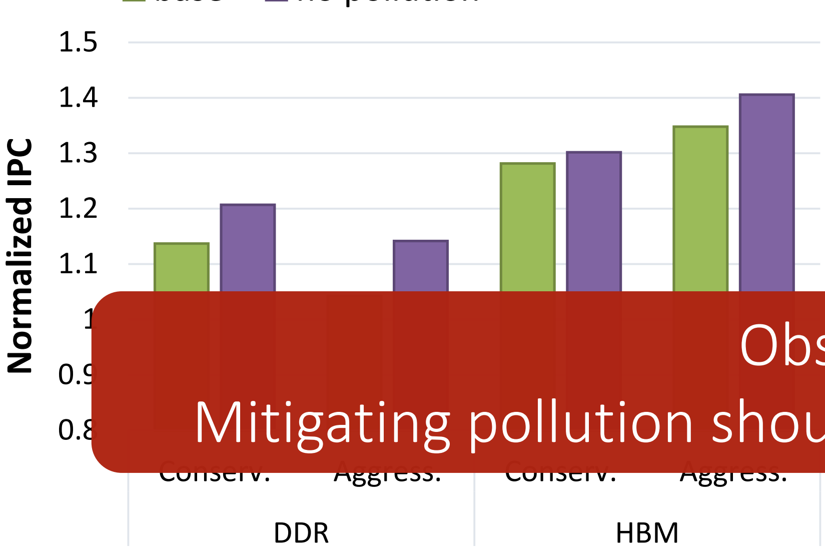
- Effect on cache pollution
  - Modest performance benefits compared with DDR

The Effect of Bandwidth on Prefetcher(2/2)



- Distance vs. degree
  - Performance variation is higher on degree

Observation 2:
Degree has a much higher impact on performance



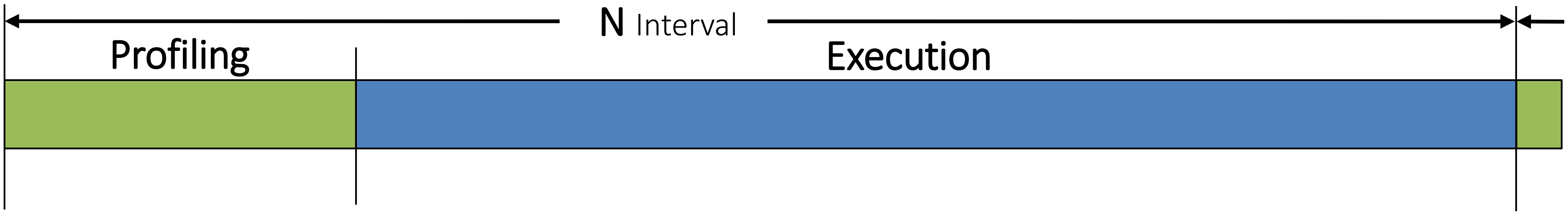
- Effect on cache pollution
  - Modest performance benefits compared with DDR

Observation 3:
Mitigating pollution should still be needed and be simple

Dynamic Prefetcher Reconfiguration

- Search by Random Profiling(RP)
  - Execute trial runs with randomly selected parameters
  - Adopt hill climbing algorithm
  - Direct performance metric (IPC: Instruction Per Cycles)
  - Profiling phase : Execution phase = 1 : 4
- Optimizations
  - Two-step profiling (decision order: distance → degree)
  - Start profiling phase with previously used best parameters

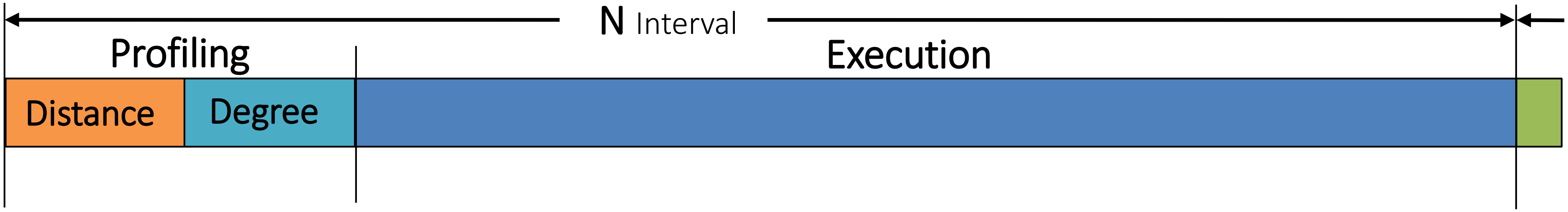
Dynamic Prefetcher Reconfiguration



- Search by Random Profiling(RP)
  - Execute trial runs with randomly selected parameters
  - Adopt hill climbing algorithm
  - Direct performance metric (IPC: Instruction Per Cycles)
  - Profiling phase : Execution phase = 1 : 4

- Optimizations
  - Two-step profiling (decision order: distance → degree)
  - Start profiling phase with previously used best parameters

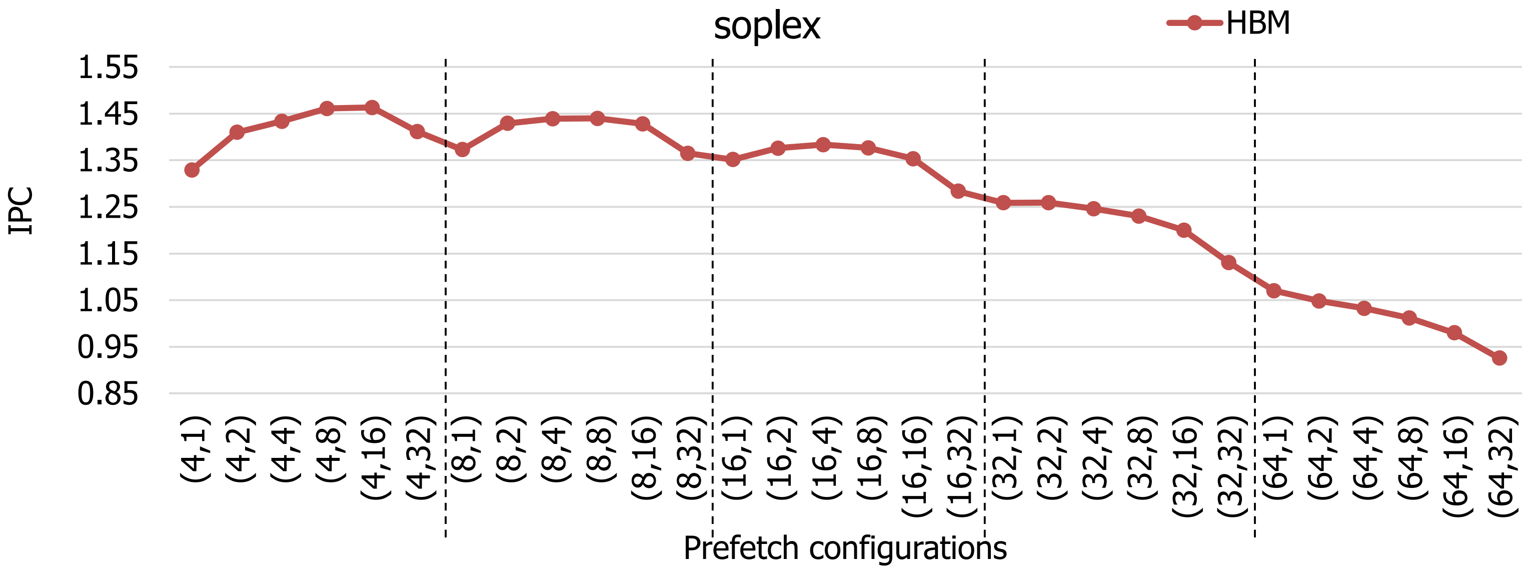
Dynamic Prefetcher Reconfiguration



- Search by Random Profiling(RP)
  - Execute trial runs with randomly selected parameters
  - Adopt hill climbing algorithm
  - Direct performance metric (IPC: Instruction Per Cycles)
  - Profiling phase : Execution phase = 1 : 4

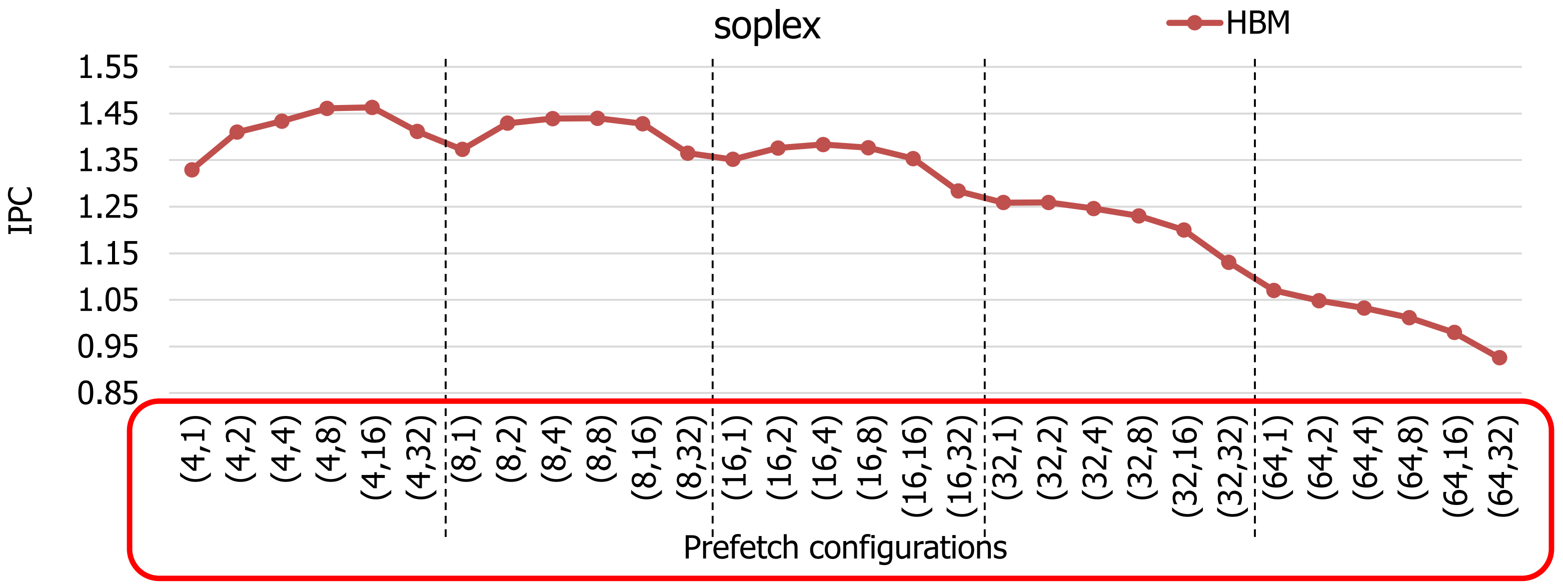
- Optimizations
  - Two-step profiling (decision order: distance → degree)
  - Start profiling phase with previously used best parameters

Hill Climbing Algorithm



- The performance curve has common form
- The curve rarely exhibits multiple local maximums
- Average trial runs is 3.77

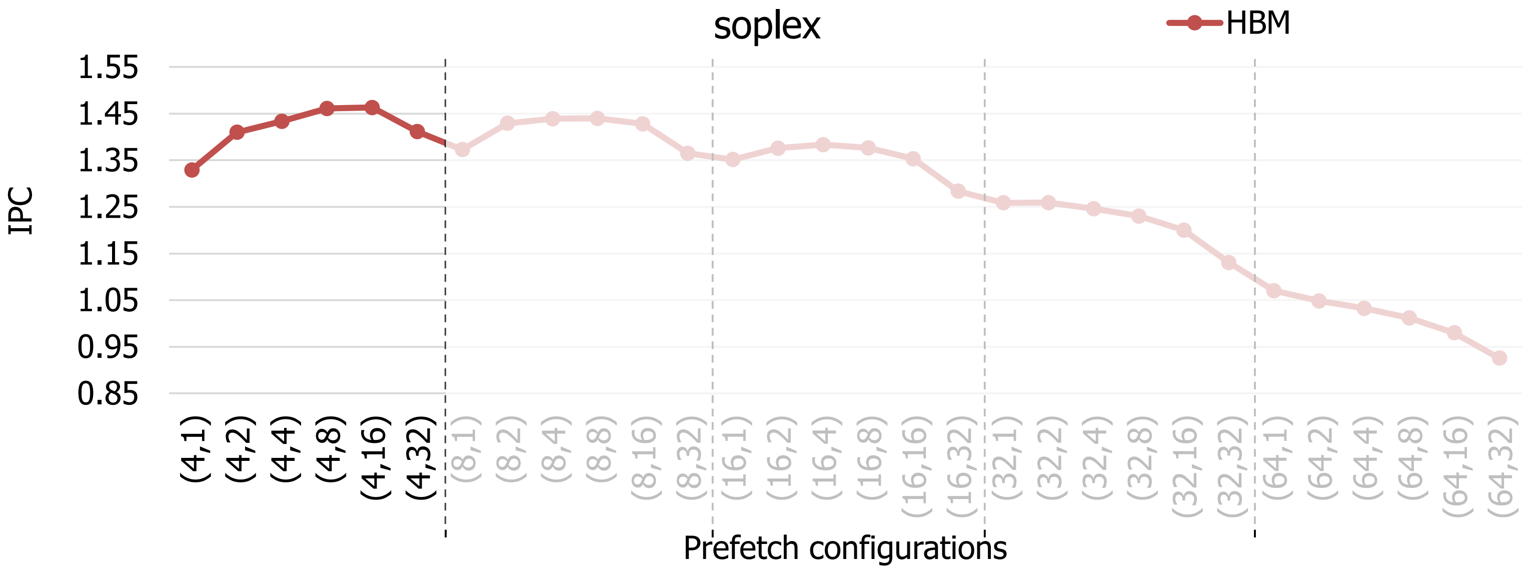
Hill Climbing Algorithm



(distance, degree)

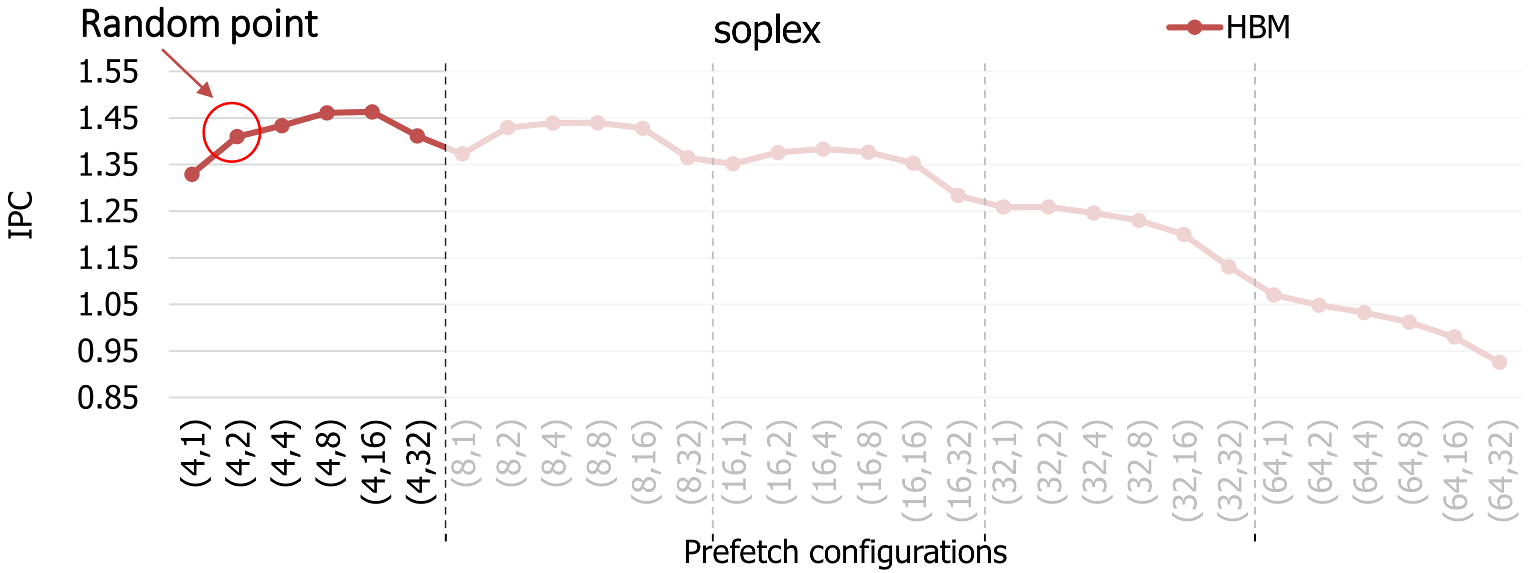
- The performance curve has common form
- The curve rarely exhibits multiple local maximums
- Average trial runs is 3.77

Hill Climbing Algorithm



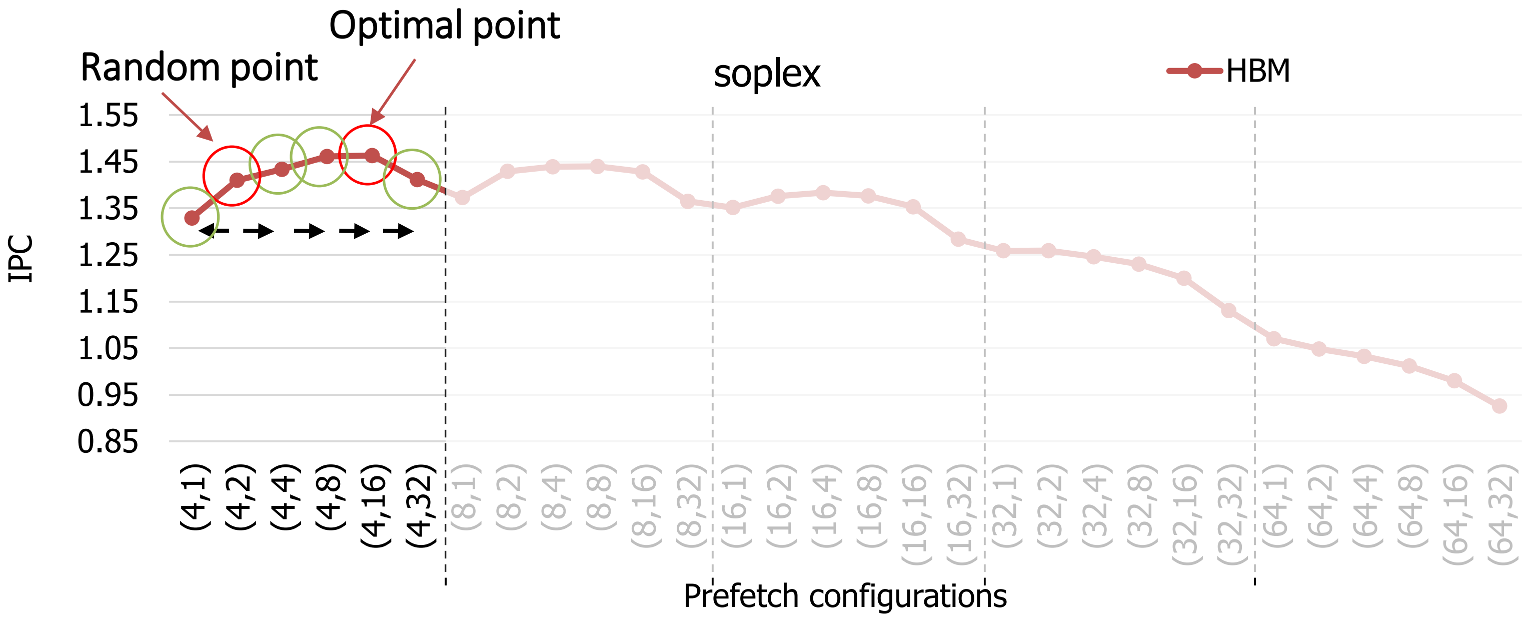
- The performance curve has common form
- The curve rarely exhibits multiple local maximums
- Average trial runs is 3.77

Hill Climbing Algorithm



- The performance curve has common form
- The curve rarely exhibits multiple local maximums
- Average trial runs is 3.77

Hill Climbing Algorithm

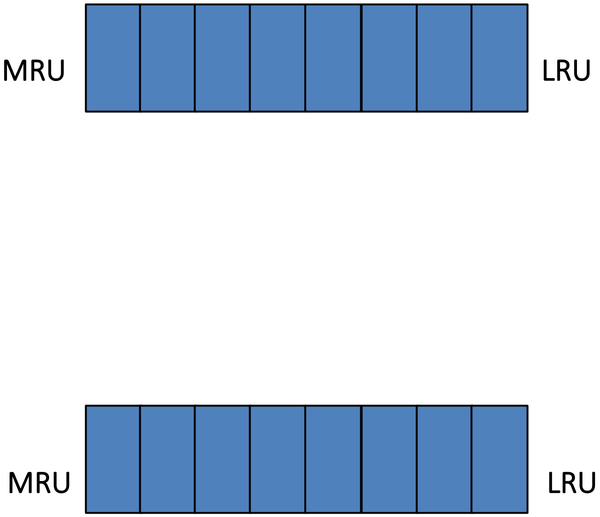


- The performance curve has common form
- The curve rarely exhibits multiple local maximums
- Average trial runs is 3.77

Mitigation of Cache Pollutions

🟩 Demand 🟪 Prefetch

- Insertion-only (Prior work [4])
  - Adjust insertion location of prefetch data
  - Promote to MRU directly

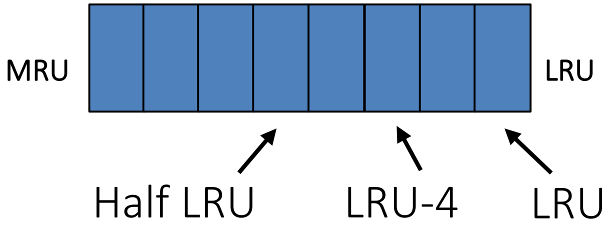


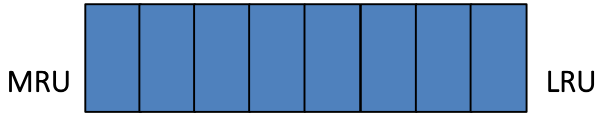
[4] Srinath et al. HPCA 2007
[5] Xie et al. ISCA 2009

Mitigation of Cache Pollutions

🟩 Demand 🟪 Prefetch

- Insertion-only (Prior work [4])
  - Adjust insertion location of prefetch data
  - Promote to MRU directly

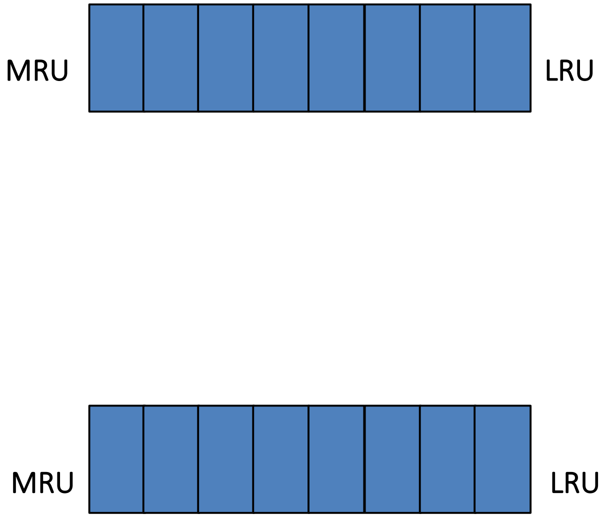




[4] Srinath et al. HPCA 2007
[5] Xie et al. ISCA 2009

Mitigation of Cache Pollutions

🟩 Demand 🟪 Prefetch



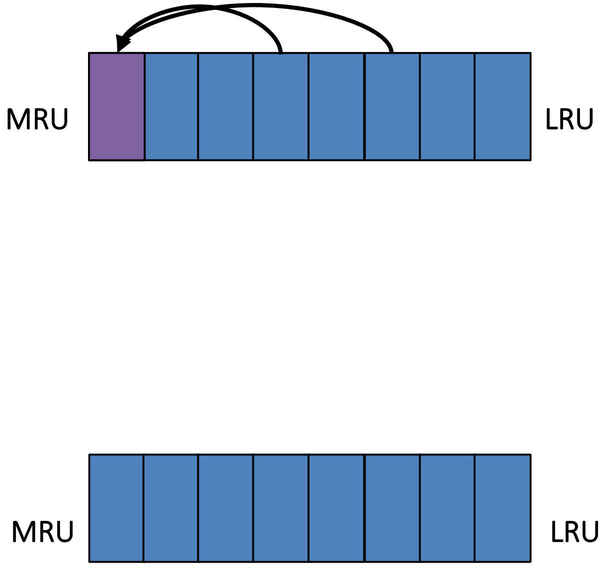
- Insertion-only (Prior work [4])
  - Adjust insertion location of prefetch data
  - Promote to MRU directly

[4] Srinath et al. HPCA 2007
[5] Xie et al. ISCA 2009

Mitigation of Cache Pollutions

🟩 Demand 🟪 Prefetch

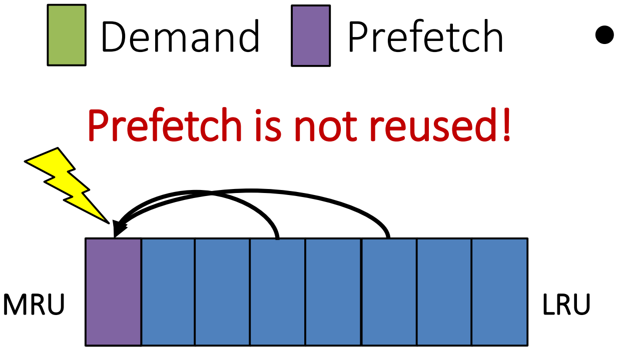
- Insertion-only (Prior work [4])
  - Adjust insertion location of prefetch data
  - Promote to MRU directly



[4] Srinath et al. HPCA 2007

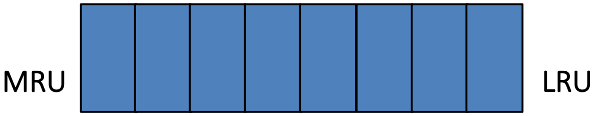
[5] Xie et al. ISCA 2009

Mitigation of Cache Pollutions



Prefetch is not reused!

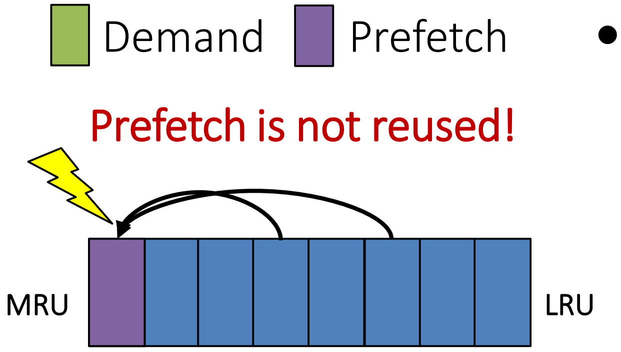
- Insertion-only (Prior work [4])
  - Adjust insertion location of prefetch data
  - Promote to MRU directly



- Prefetch Partition(PP)
  - Insight: prefetch data are often not reused after the initial demand hit
  - Soft-partition: adopt simple pseudo-partitioning from PIPP [5]
  - Optimization: using top two policies
    - (MRU:LRU-4), (MRU: LRU)
    - Can reap out most of the benefits

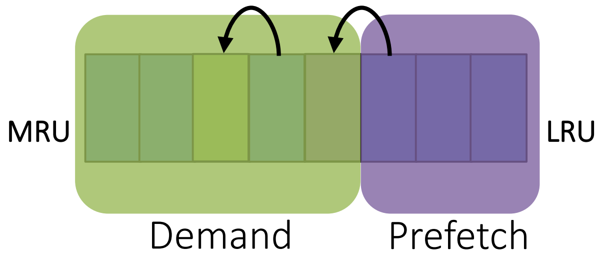
[4] Srinath et al. HPCA 2007
[5] Xie et al. ISCA 2009

Mitigation of Cache Pollutions



Prefetch is not reused!

- Insertion-only (Prior work [4])
  - Adjust insertion location of prefetch data
  - Promote to MRU directly



- Prefetch Partition(PP)
  - Insight: prefetch data are often not reused after the initial demand hit
  - Soft-partition: adopt simple pseudo-partitioning from PIPP [5]
  - Optimization: using top two policies
    - (MRU:LRU-4), (MRU: LRU)
    - Can reap out most of the benefits

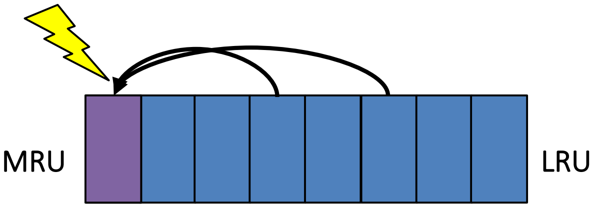
[4] Srinath et al. HPCA 2007

[5] Xie et al. ISCA 2009

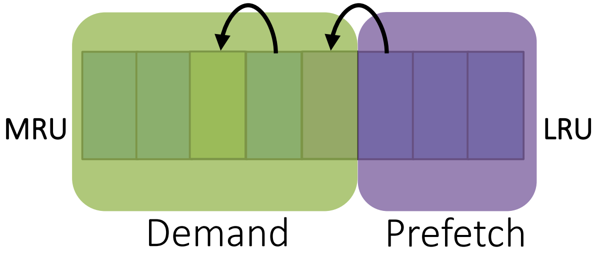
Mitigation of Cache Pollutions

🟩 Demand 🟪 Prefetch

Prefetch is not reused!



- Insertion-only (Prior work [4])
  - Adjust insertion location of prefetch data
  - Promote to MRU directly



- Prefetch Partition(PP)
  - Insight: prefetch data are often not reused after the initial demand hit
  - Soft-partition: adopt simple pseudo-partitioning from PIPP [5]
  - Optimization: using top two policies
    - (MRU:LRU-4), (MRU: LRU)
    - Can reap out most of the benefits

[4] Srinath et al. HPCA 2007
[5] Xie et al. ISCA 2009

Evaluation

- Full system OoO simulation on *McSim + Gems + DRAMSim2*; *8 cores*, 128 ROB, 4 ways
- *Three Level Cache hierarchy*
- *Memory Configuration*

Parameter	Values
DDR	2channels, DDR3-1600(800Mhz)
HBM	16channels, HBM-1600(800Mhz)
fast HBM	HBM with x2 frequency : 1600MHz
half HBM	HBM with /2 channels : 8 channels

Parameter	Values
hybrid 1	HBM + DDR
hybrid 2	fast HBM + DDR
hybrid 3	half HBM + DDR

- *Stream prefetchers* with 8 streams
- *Benchmarks: SPECCPU, Mixed* workloads

Evaluation

- Full system OoO simulation on *McSim + Gems + DRAMSim2*; *8 cores*, 128 ROB, 4 ways
- *Three Level Cache hierarchy*
- *Memory Configuration*

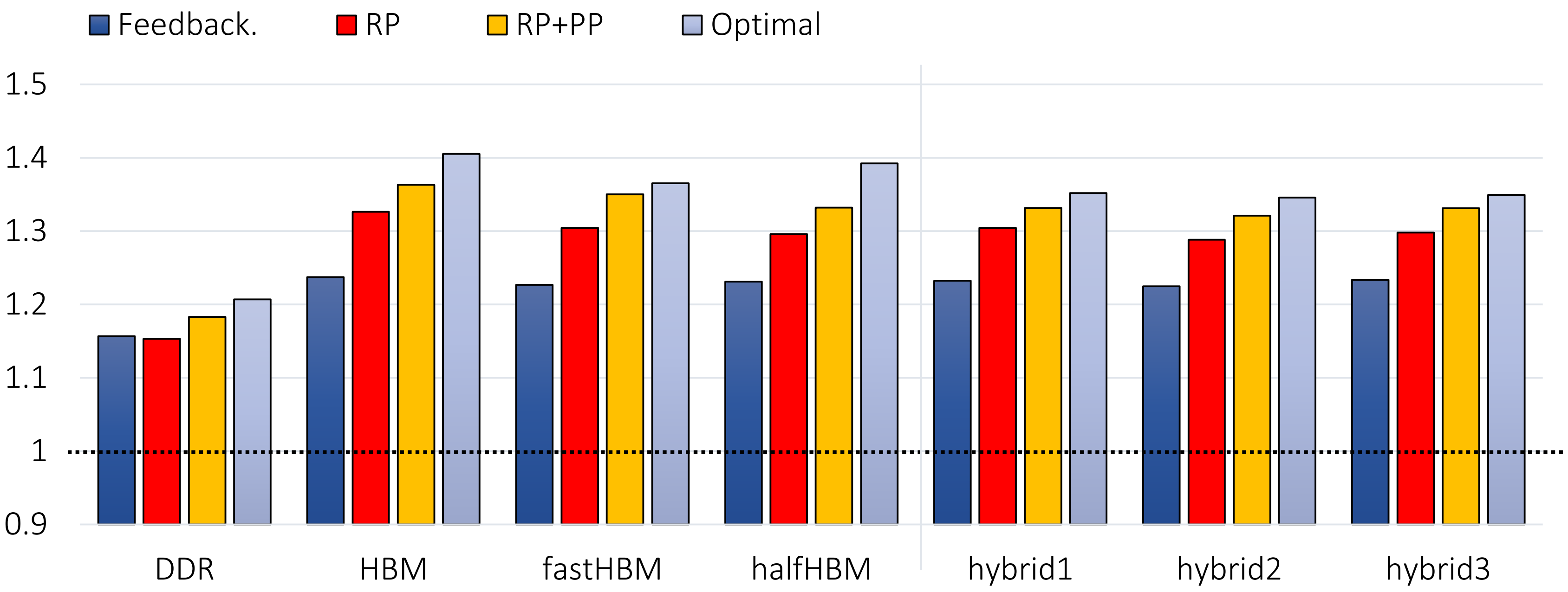
Parameter	Values
DDR	2channels, DDR3-1600(800Mhz)
HBM	16channels, HBM-1600(800Mhz)
fast HBM	HBM with x2 frequency : 1600MHz
half HBM	HBM with /2 channels : 8 channels

Parameter	Values
hybrid 1	HBM + DDR
hybrid 2	fast HBM + DDR
hybrid 3	half HBM + DDR

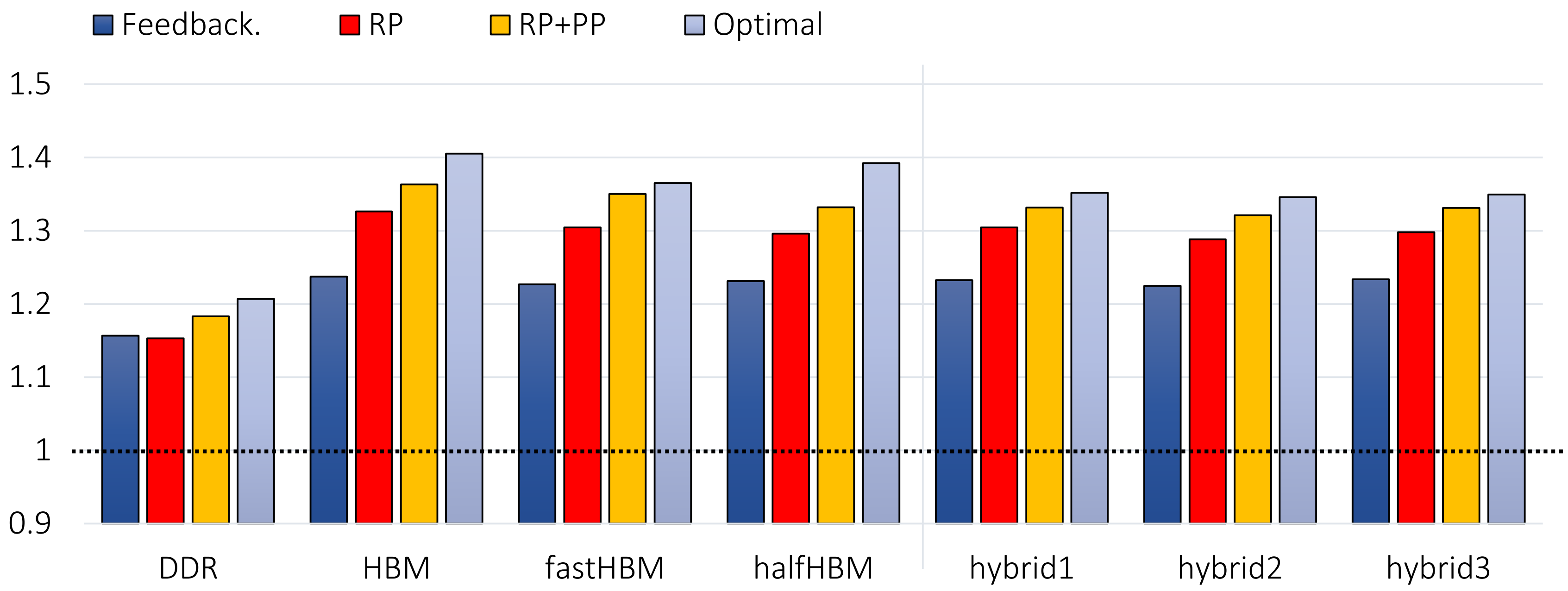
- *Stream prefetchers* with 8 streams
- *Benchmarks: SPECCPU, Mixed* workloads

Performance on diverse memory types

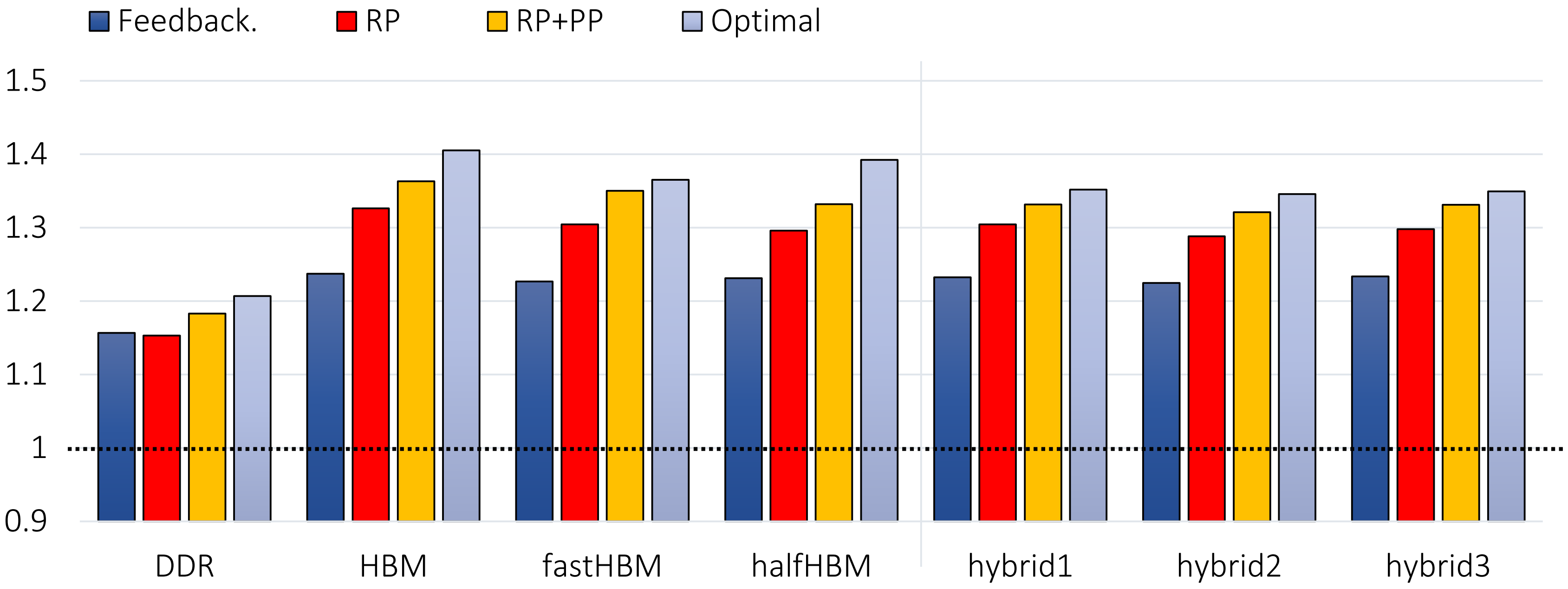
Performance on diverse memory types



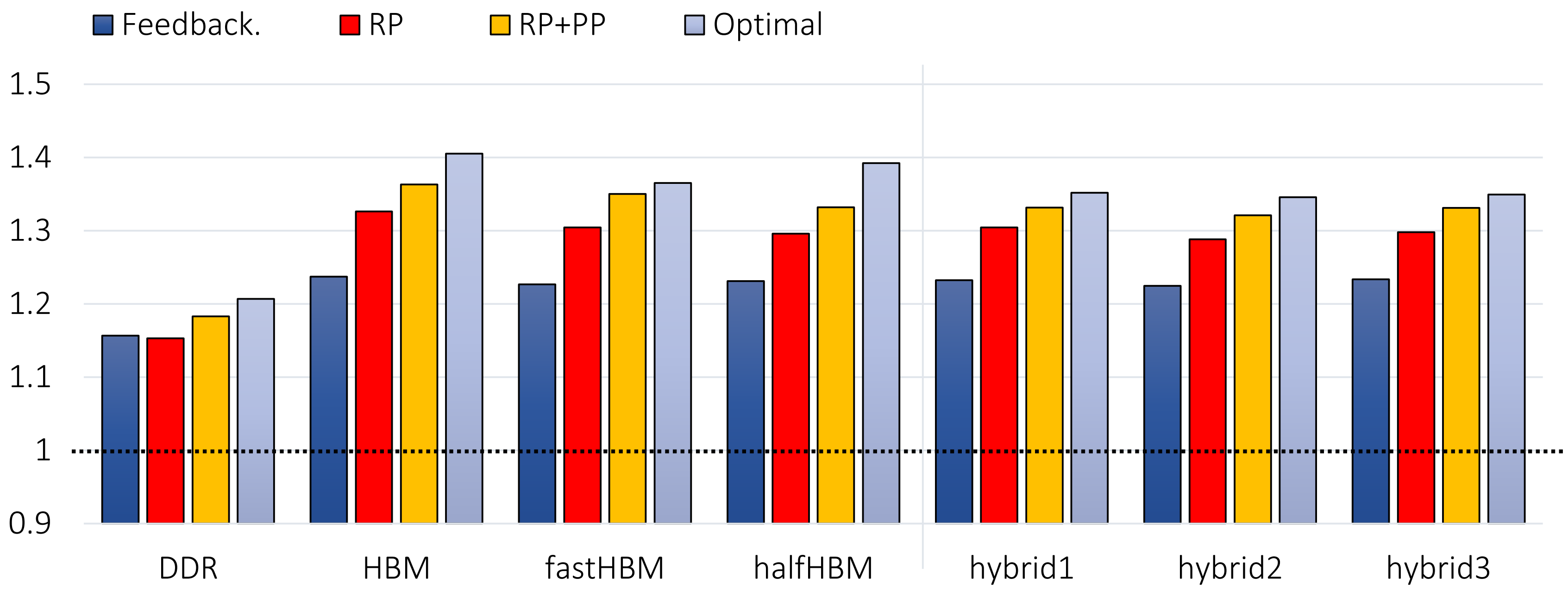
Performance on diverse memory types



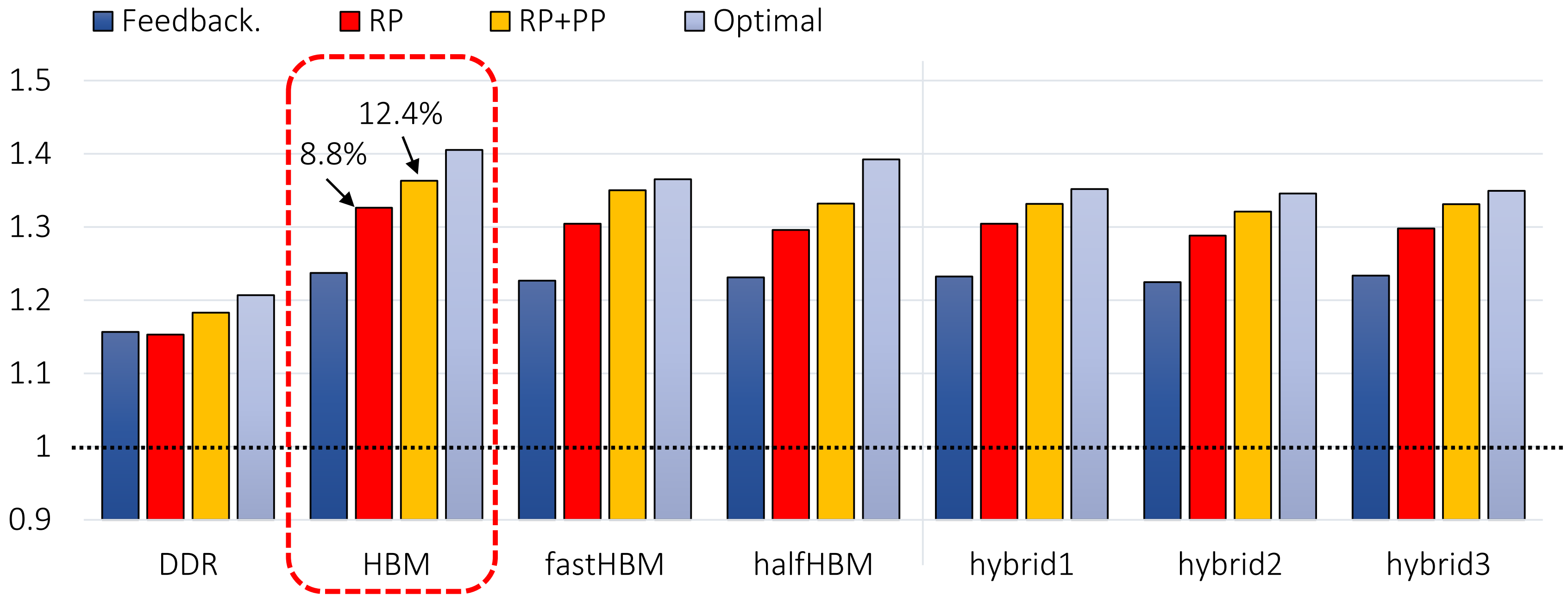
Performance on diverse memory types



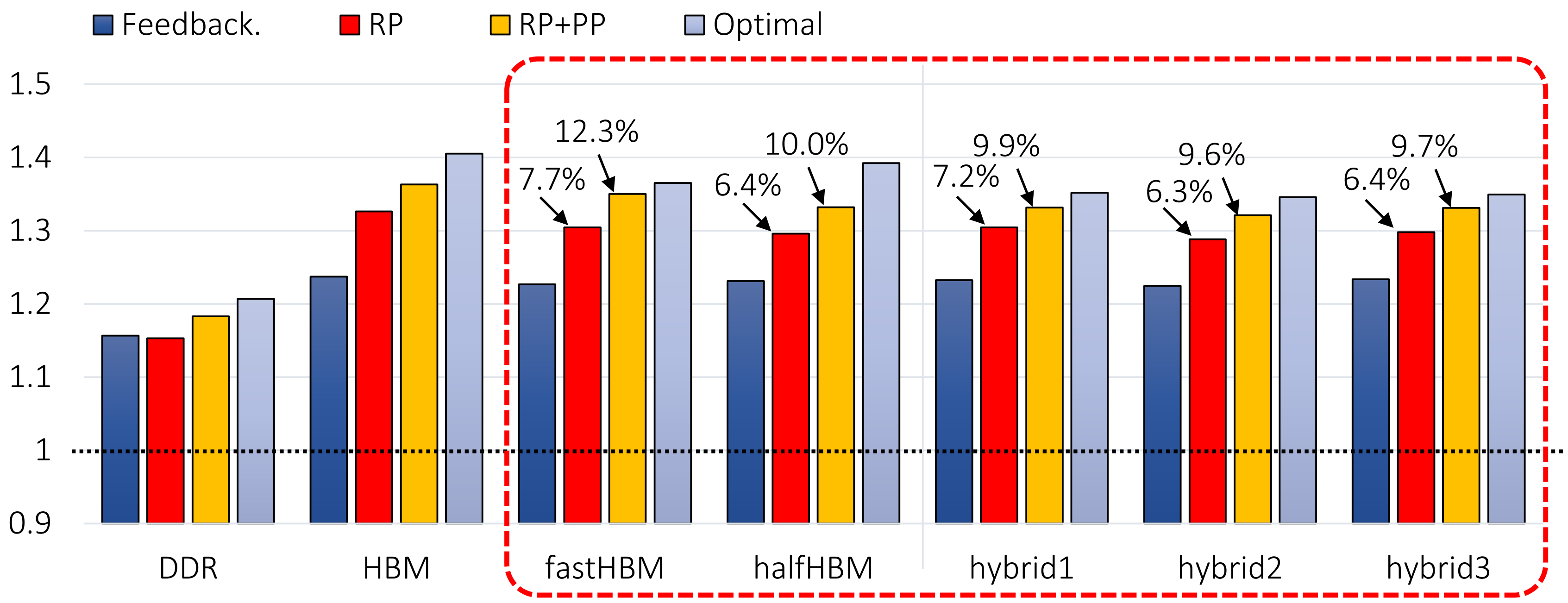
Performance on diverse memory types



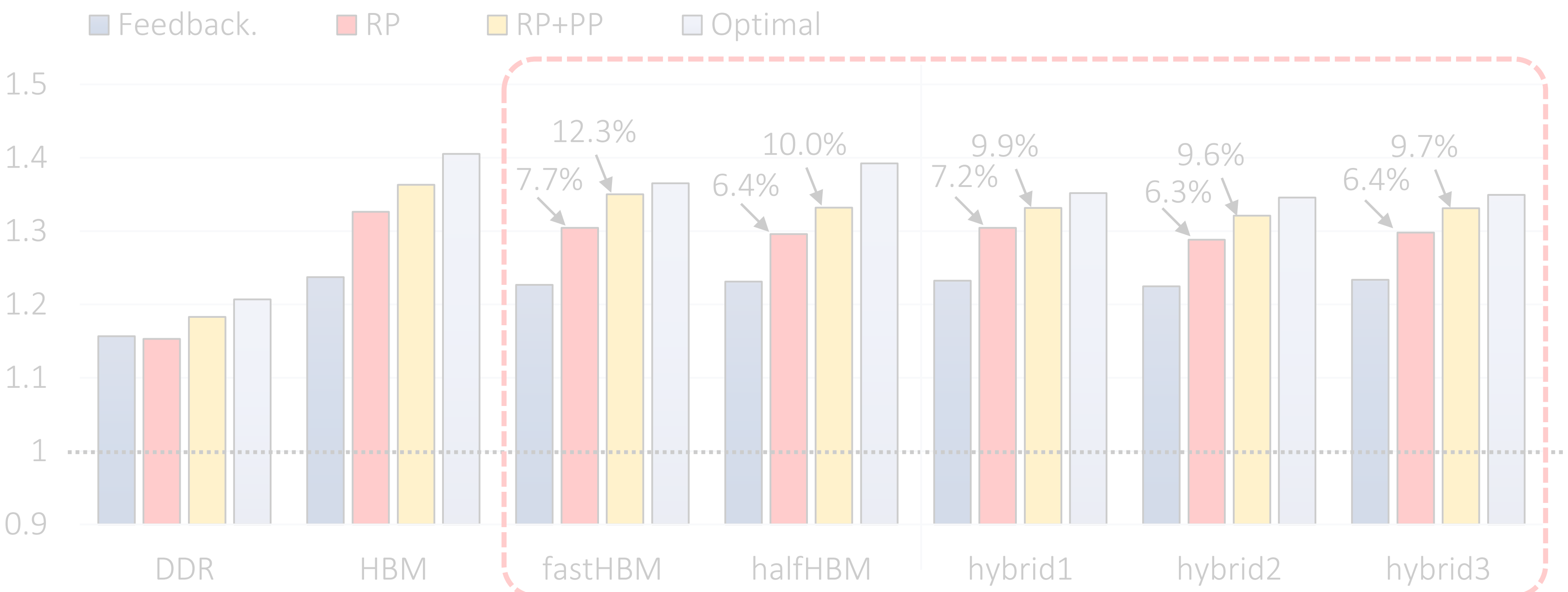
Performance on diverse memory types



Performance on diverse memory types



Performance on diverse memory types



Average 10.0% performance improvement (avg. 12.4% on HBM) on diverse memory architectures compared to the prior approach

Summary

- Investigate how the differences in memory architecture affect the optimal prefetching scheme
- Study how the prefetching parameters can be dynamically and effectively adjusted

Summary

- Investigate how the differences in memory architecture affect the optimal prefetching scheme
- Study how the prefetching parameters can be dynamically and effectively adjusted
- Dynamic Prefetcher Reconfiguration
  - Effective search by random profiling Prefetcher design on hybrid memory
  - Simple soft-partition mechanism to mitigate pollution
  - Average 10.0% performance improvement (avg. 12.4% on HBM) compared to the prior approach

34th IEEE International Conference
on Computer Design
ICCD 2016

Dynamic Prefetcher Reconfiguration for Diverse Memory Architectures

*Junghoon Lee**, ***Taehoon Kim***, *and Jaehyuk Huh*

SAMSUNG ADVANCED INSTITUTE OF TECHNOLOGY

KAIST School of Computing