Instruction-based Reuse-Distance Prediction for Effective Cache Management

Pavlos Petoumenos, Georgios Keramidas and Stefanos Kaxiras

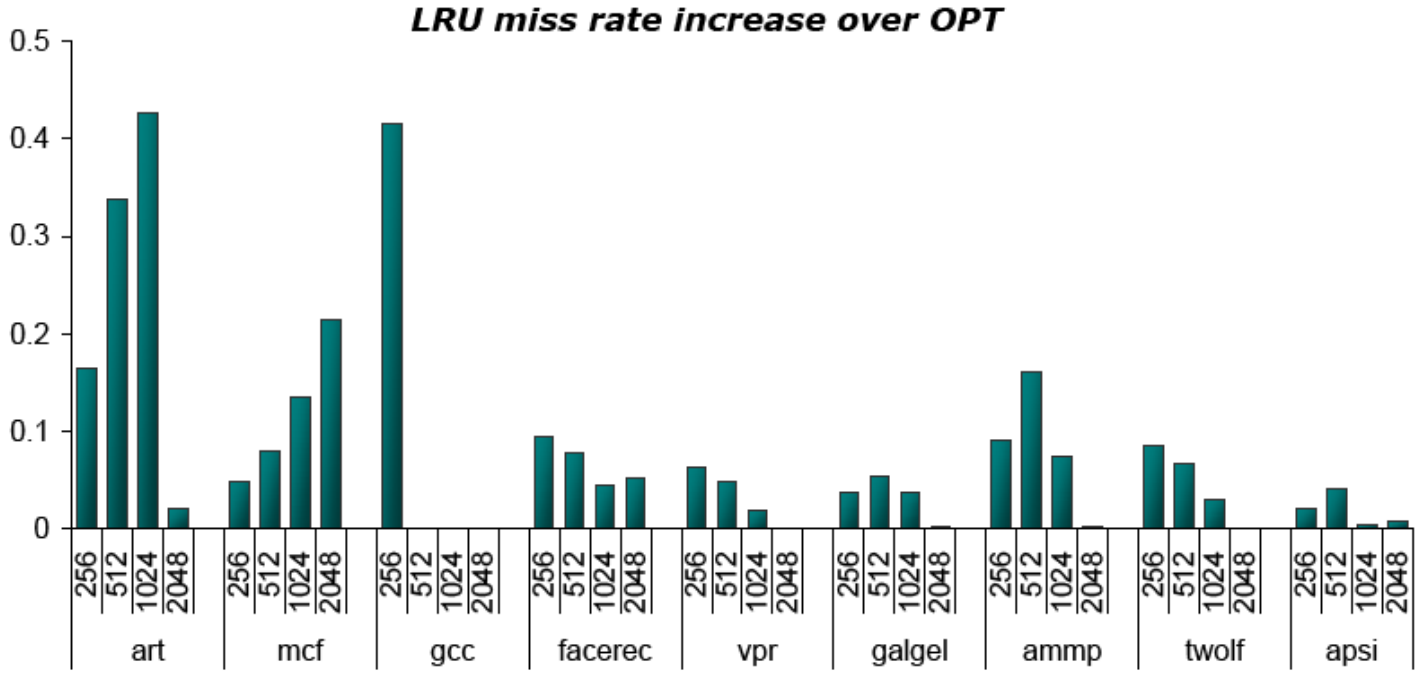
University of Patras, Greece

Introduction

- The memory wall rises more and more
  - L2 Cache: Last line of defense

- Contributions:
  - Improved and less resource demanding Instruction based Reuse-Distance Prediction
  - Efficient reuse-distance sampling
  - Case study: a replacement algorithm for second level caches

LRU vs Belady's Optimal Replacement

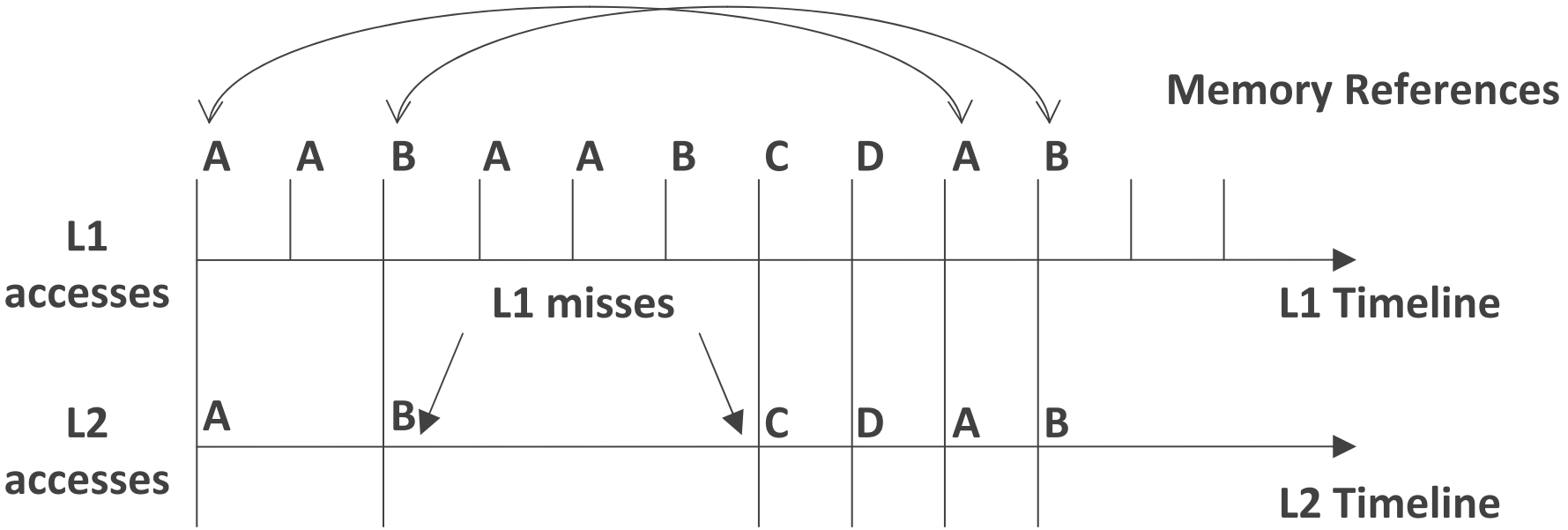


- LRU: Inefficient for L2 caches
- Reasons:
  - L1 filtering
  - Highly associative caches

Can we predict the future?

- Yes we can!
- Memory behavior → repeating patterns
- Our Motivation:
  - Strong correlation between instructions (PC) and moment of next access (reuse distance)

Reuse Distance



Reuse Distance (measured in mem accesses)

A: 7, B: 6

Which one is better?

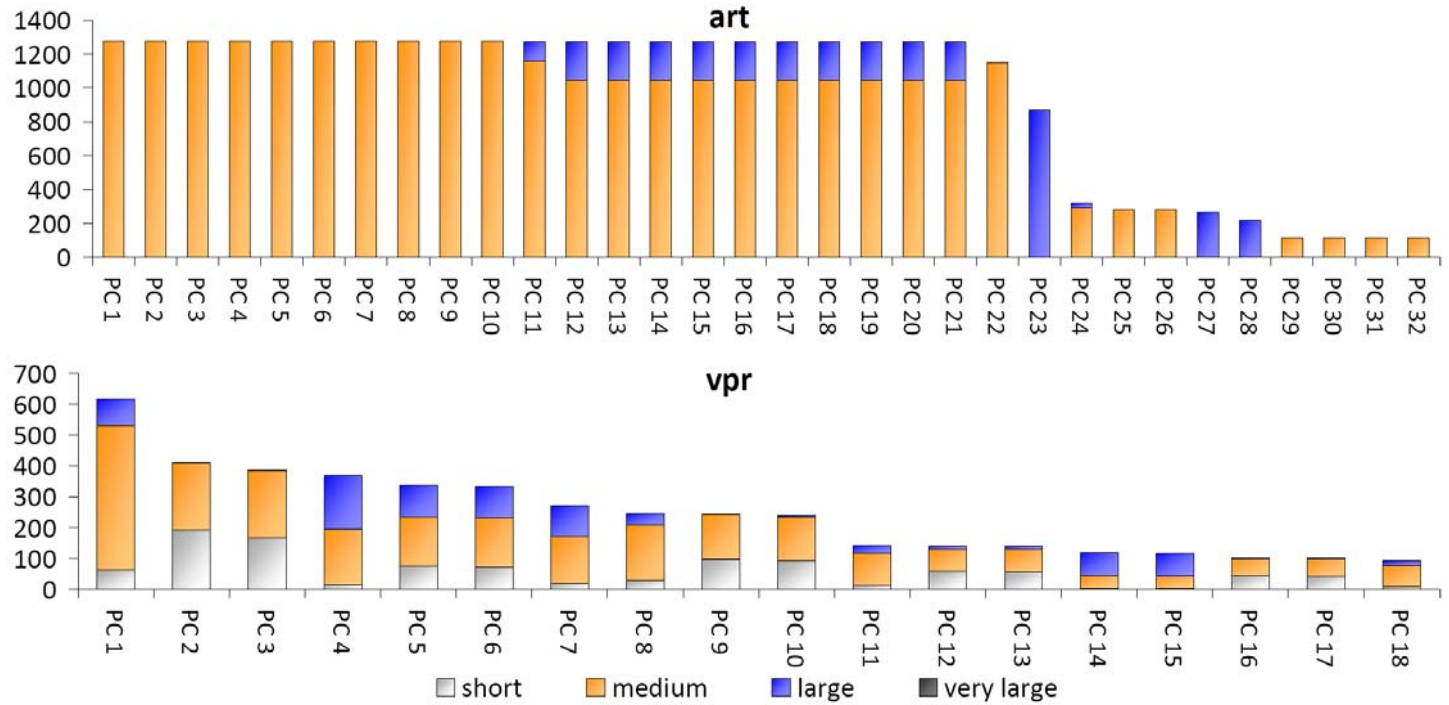
Reuse Distance (measured in L2 accesses)

A: 3, B: 3

Discount Irrelevant Accesses

- L1 writebacks:
  - Appear as write accesses in the L2
  - Not related to any instruction

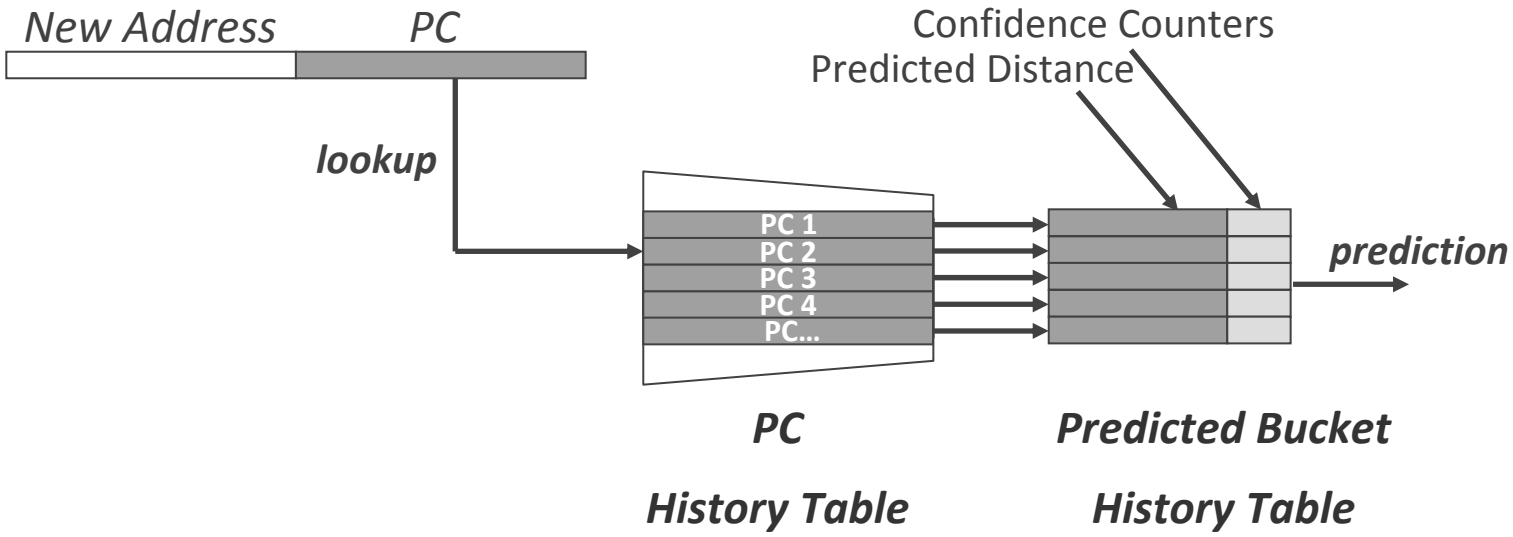
Reuse distances are predictable



- Significant PCs:
  - Are few → Easily stored
  - Have predictable reuse distances

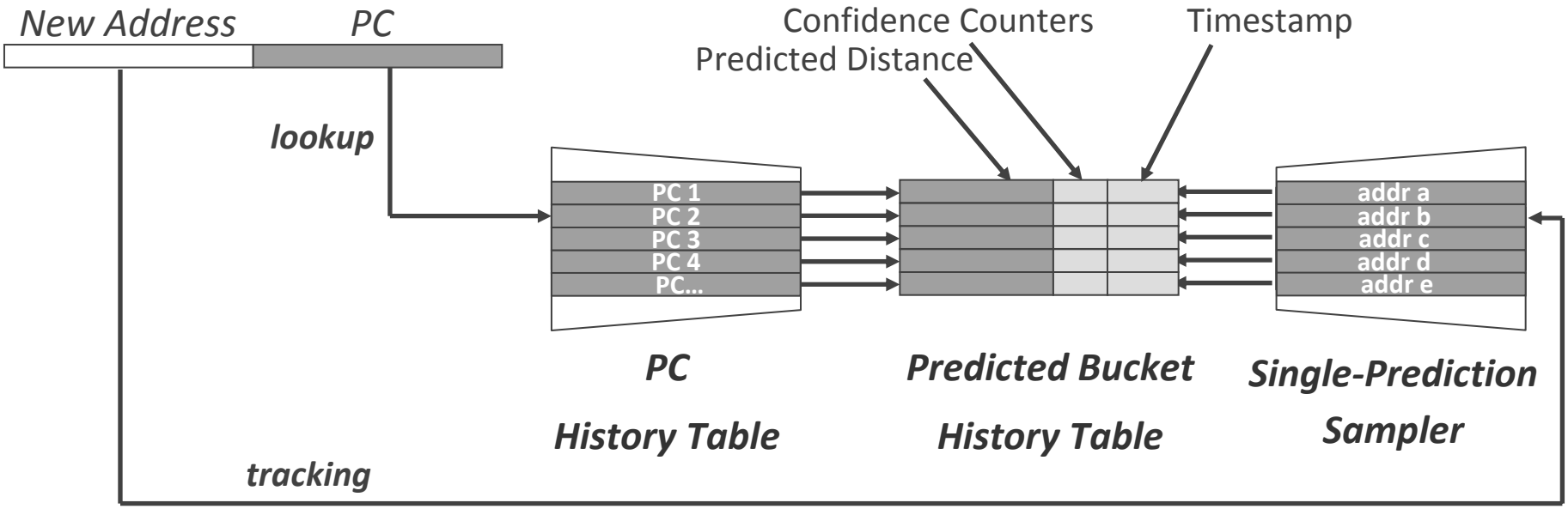
Instruction based Predictor

- Predicts the reuse distance of an access based on the PC which initiated the access
- 256 entries are enough!

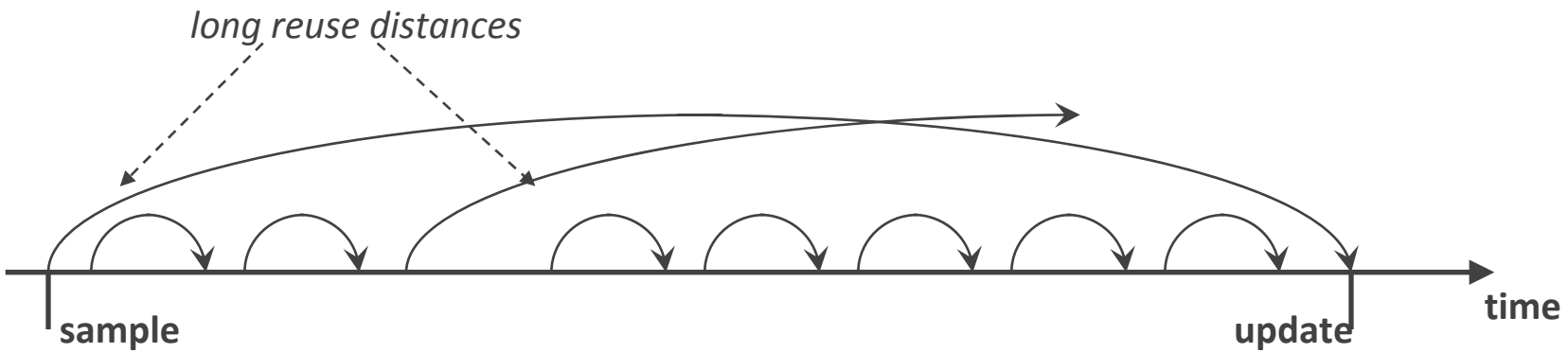


Reuse Distance Sampling

- Perfect reuse-distance information is not possible
- Single-Prediction Sampler (SPS)
  - One sample per predictor entry
  - Almost perfect reuse-distance information
  - Minimal overhead



Reuse Distance Sampling

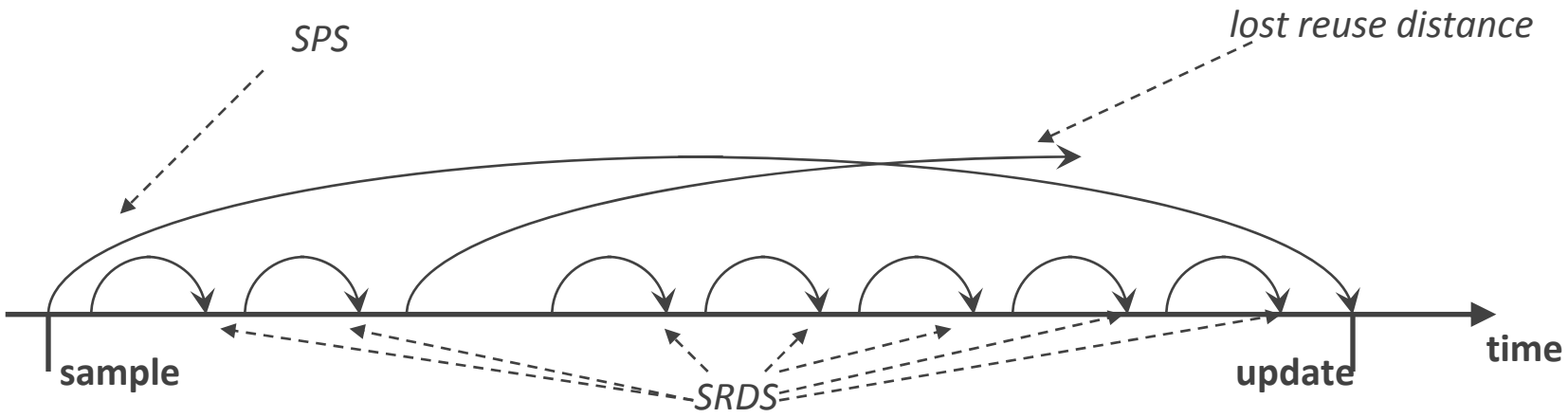


- Long reuse-distances “block” multiple short reuse-distances
  - Long rds are overrepresented

Reuse Distance Sampling

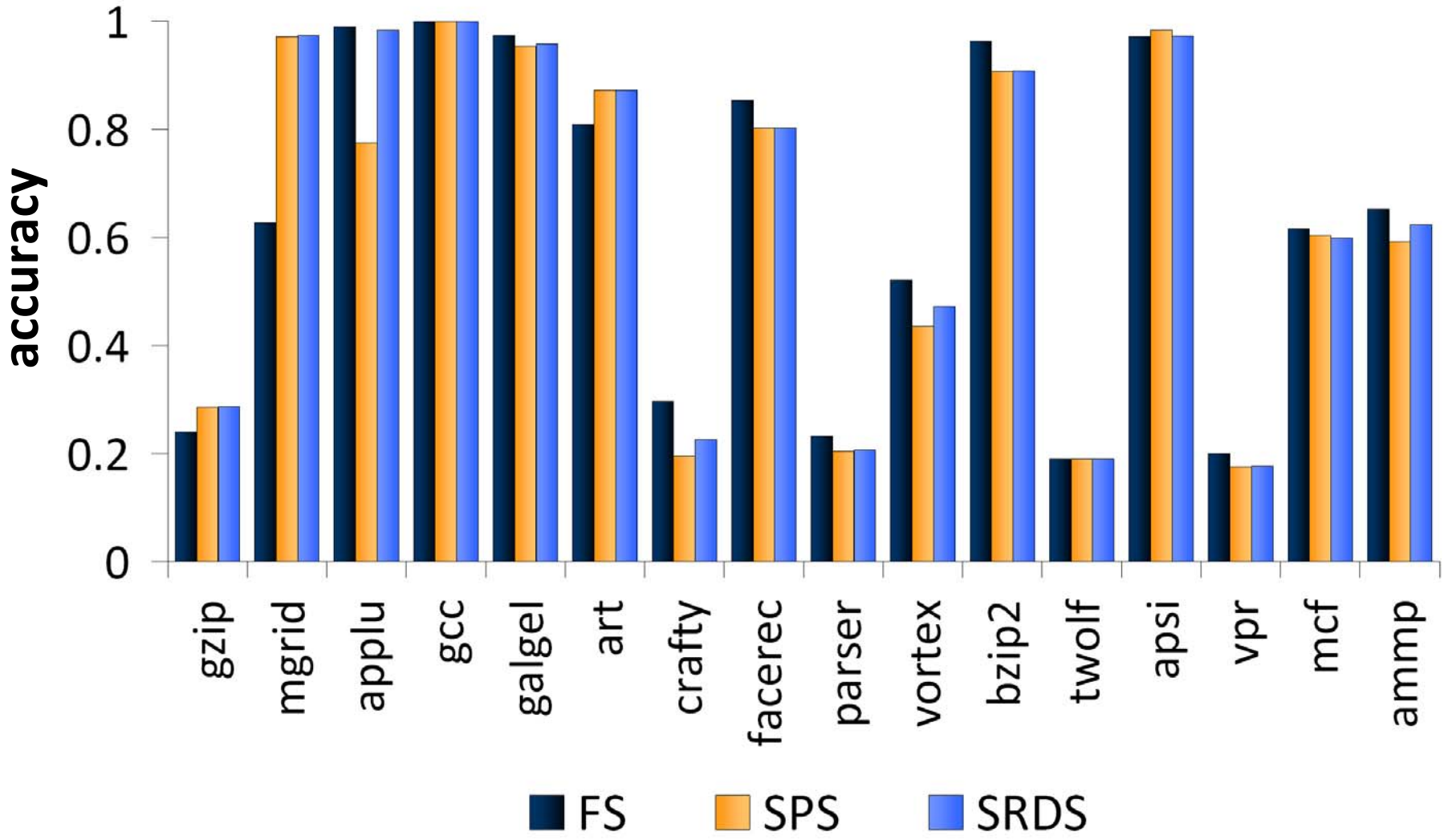
- Short Reuse-Distance Sampler (SRDS)
  - Small FIFO queue
  - Samples the L2 access stream
  - Captures only small reuse-distances
  - Can capture overlapping reuse-distances

Reuse Distance Sampler



- Overlapping long reuse distances are lost
  - Bias in favor of short reuse distances
- For each rd which doesn't fit in SRDS inhibit one which does

Sampling mechanisms – Comparison



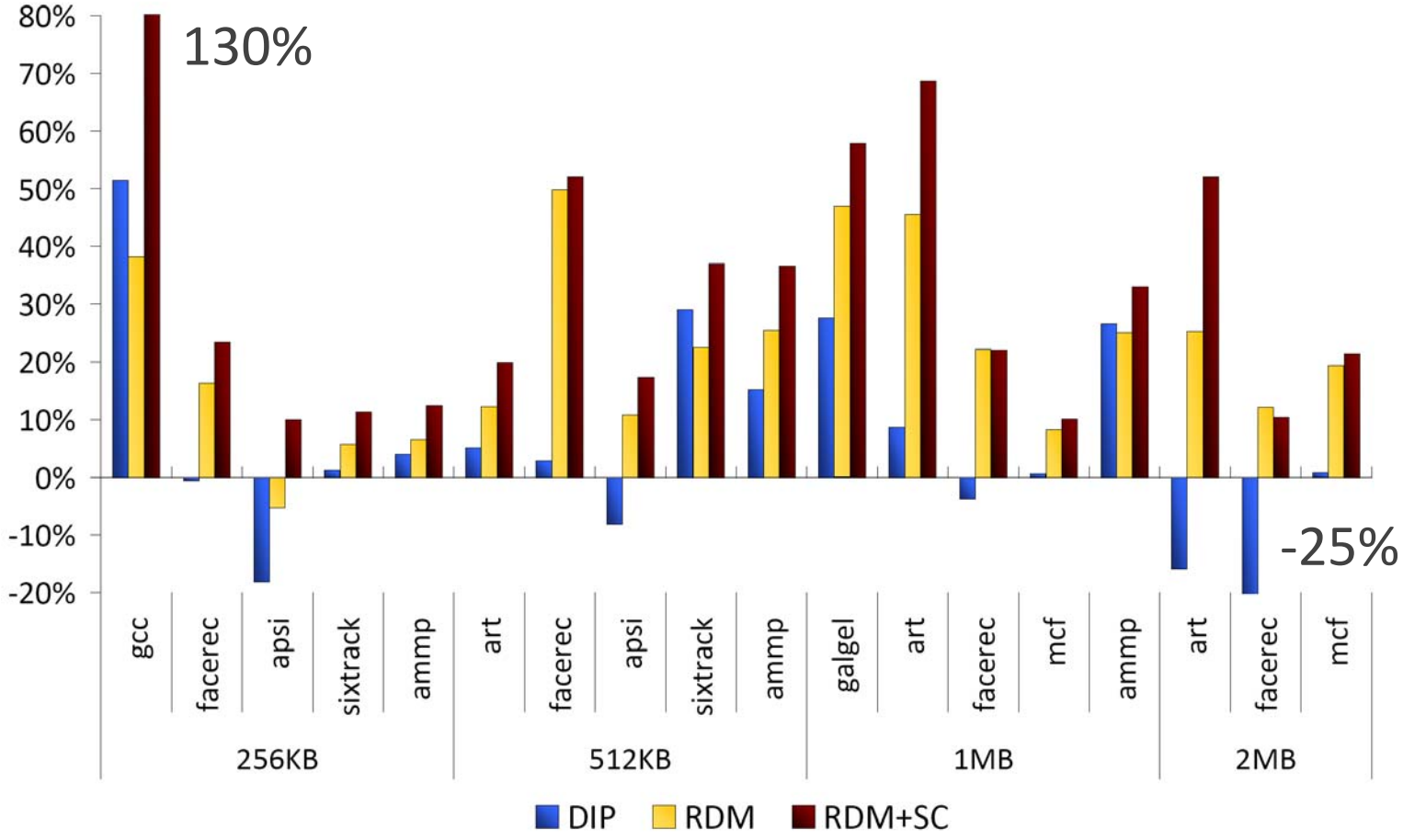
A case study: replacement policy

- Belady's OPT is within our grasp
  - Our predictor fulfills the need for quantitative info about the future
- Reuse-Distance Management (RDM):
  - Replace the line used farthest in the future (OPT) when you have enough info
  - Replace the line used farthest in the past (LRU) if you don't
- Selective Caching (RDM-SC):
  - RDM can choose the currently fetched block

Evaluation

- Basic Setup:
  - Memory intensive SPEC2000 benchmarks
  - 256KB-2MB unified L2 caches
- Comparison against Dynamic Insertion Policy
  - ISCA 2007, Qureshi, Jaleel, Patt, Steely and Emer.
  - Best performing algorithm at the moment
- All results are relative to LRU
- Only results with improvement or degradation relative to LRU are presented

IPC improvement



RDM+SC never reduces IPC

Conclusions

- Instruction-based reuse-distance prediction
  - Feasible
  - Offers high predictability of reuse patterns
- Cache Replacement using reuse-distance information outperforms state-of-the-art algorithms
- Future Work
  - More cache management schemes
  - Integrate MLP-prediction in the predictor

Thank You!

Any questions?

Misses Reduction