INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING

ICPP/2021/CHICAGO/USA

acm In-Cooperation

sighpc

AUGUST 9-12, 2021

Matryoshka: A Coalesced Delta Sequence Prefetcher

Shizhi Jiang

Yiwei Ci

Qiusong Yang

Mingshu Li

中国科学院大学
University of Chinese Academy of Sciences

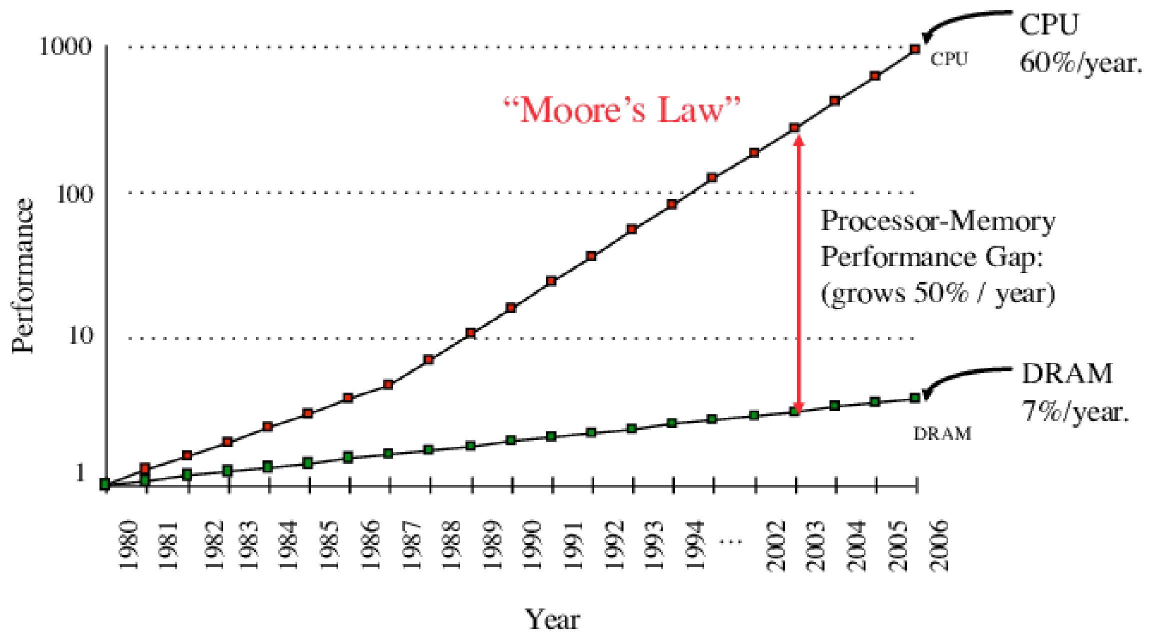
ISCAS

Presentation Structure

- Background
- Motivation
- Design & Implementation
- Evaluation
- Conclusion

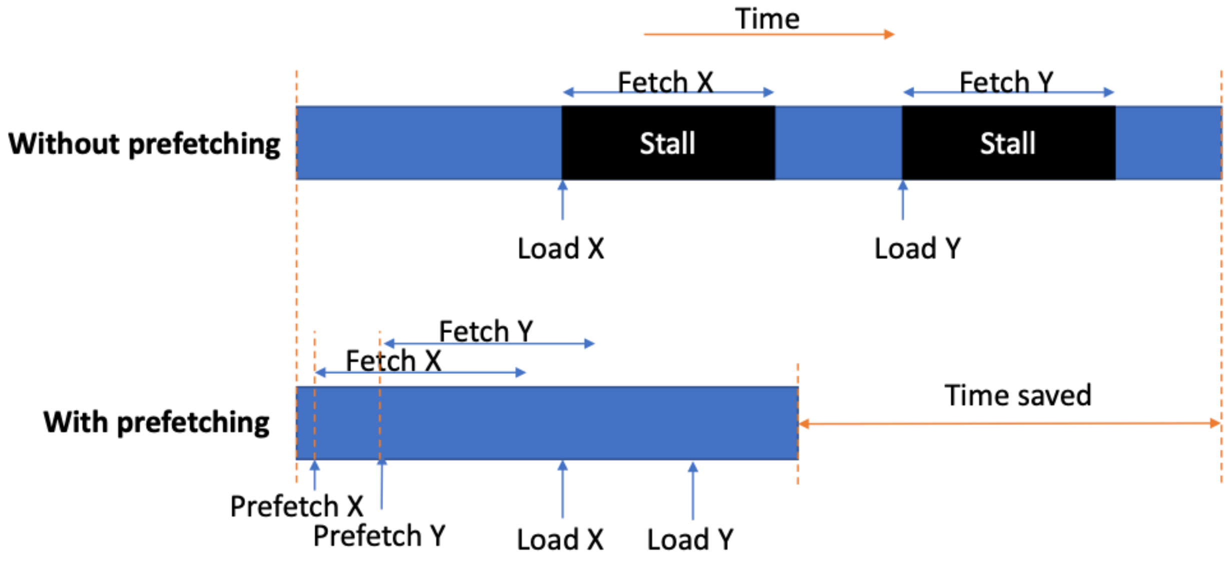
Background

The “Memory Wall”



Prefetcher

- learning access history
- predict future accesses
- low-overhead
- high-performance



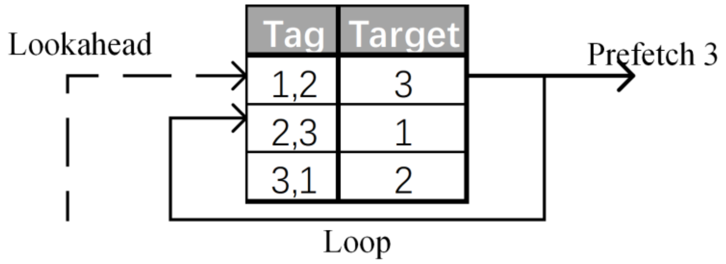
Background

Delta Sequence

Delta = Current Addr - Last Addr

e.g. delta seq : 10, 5, 15

Recursive Lookahead



Advantages

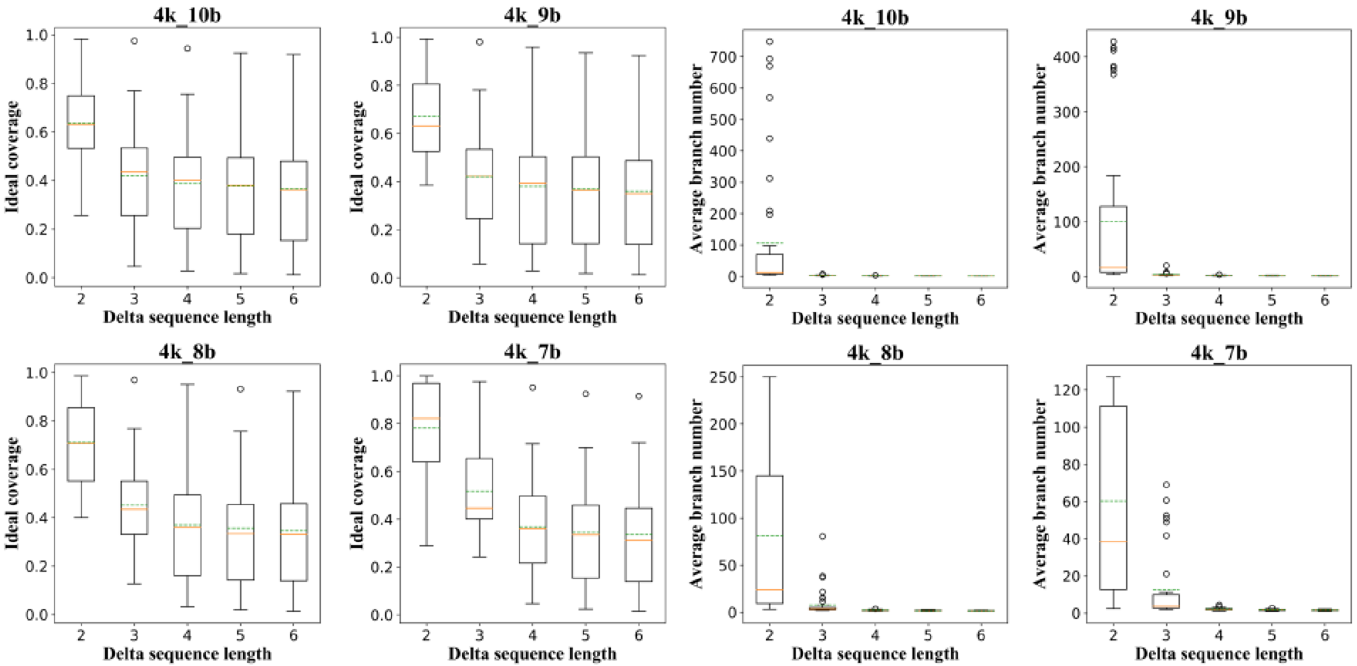
a) Shared Among Different Regions
b) Record Access Order

Motivation

Why we need multi-matching ?

- Ideal Coverage
  - Always scattered in workloads
- Average Branch Number (Ideal Accuracy)
  - Good in 3-delta or longer sequences

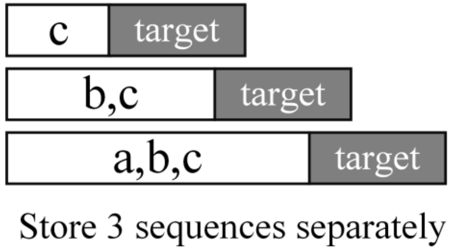
We can hardly decide the best delta sequence length!



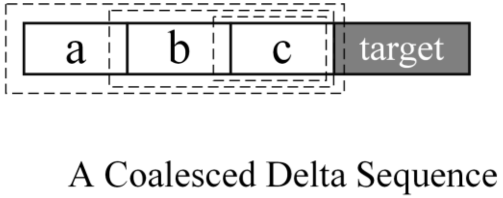
(a) (b)

Motivation

Better Form Stoing Patterns



VLDP’s Design

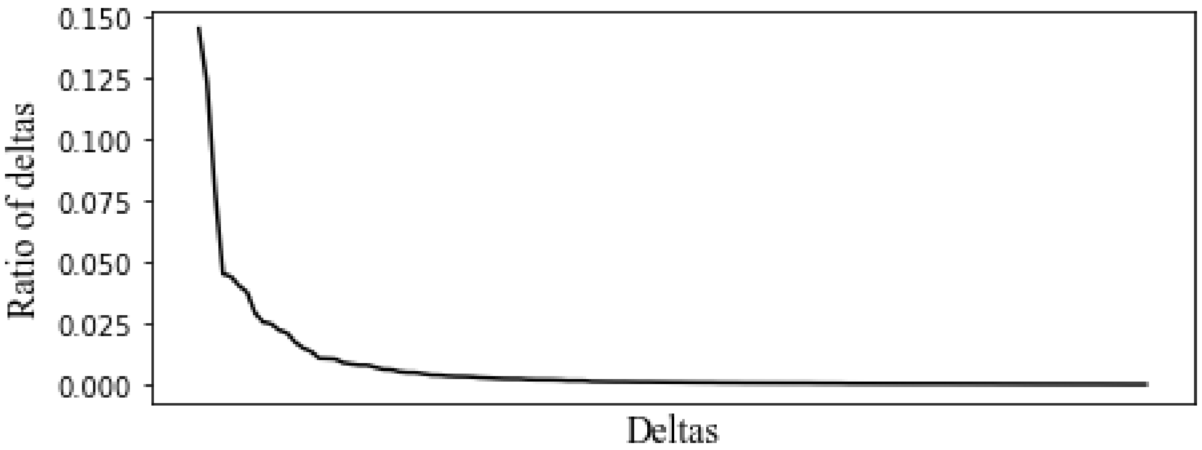


Our Design

It’s obvious more efficient if we can find a method to coalesce variable-length sequences

Motivation

Storing Hot Patterns



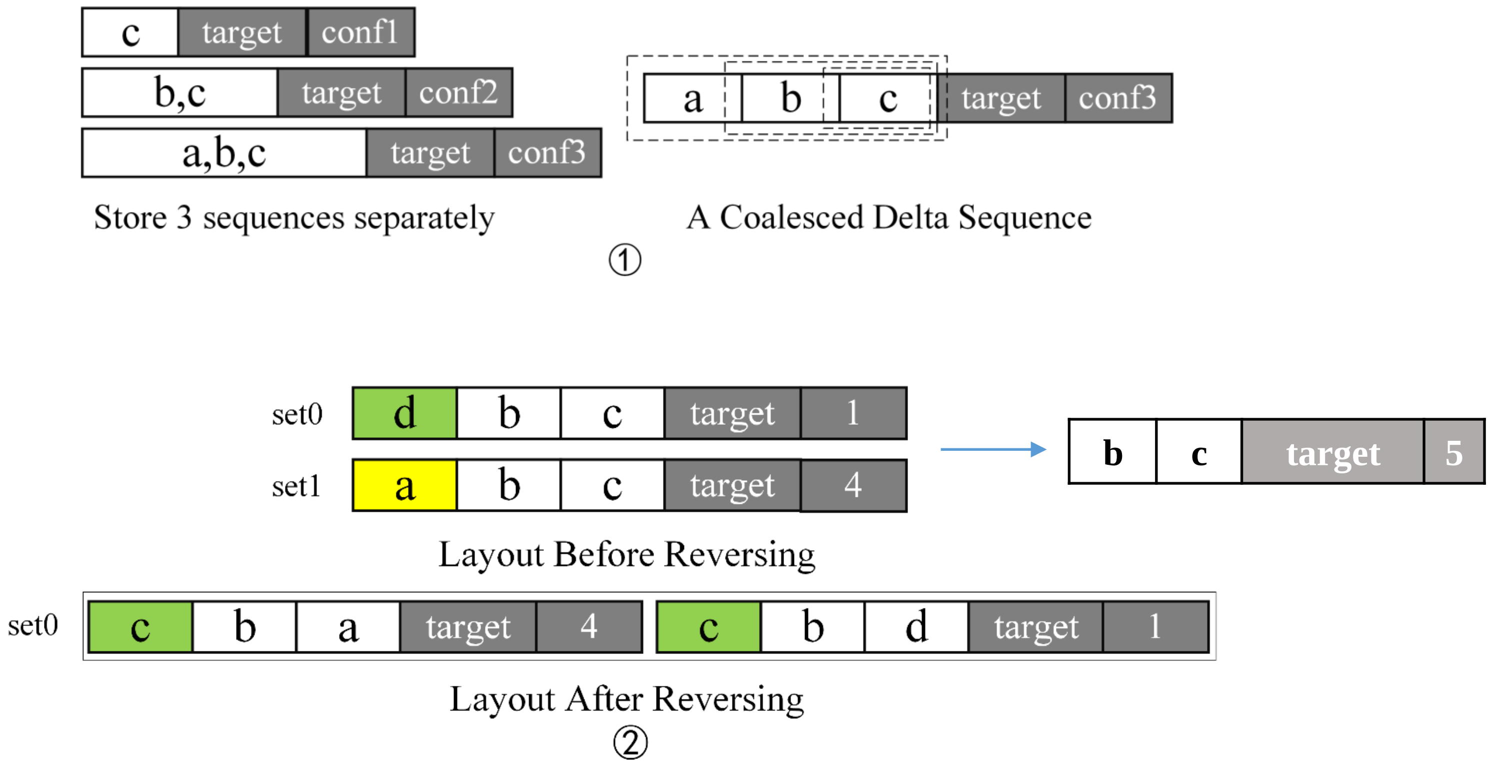
Only a small part of deltas reappear frequently

Design

Reversed Coalesced Delta Sequences

The confidence (repeating times) of a sequence is vital for accurate prefetch! How to deal with them during coalescing?

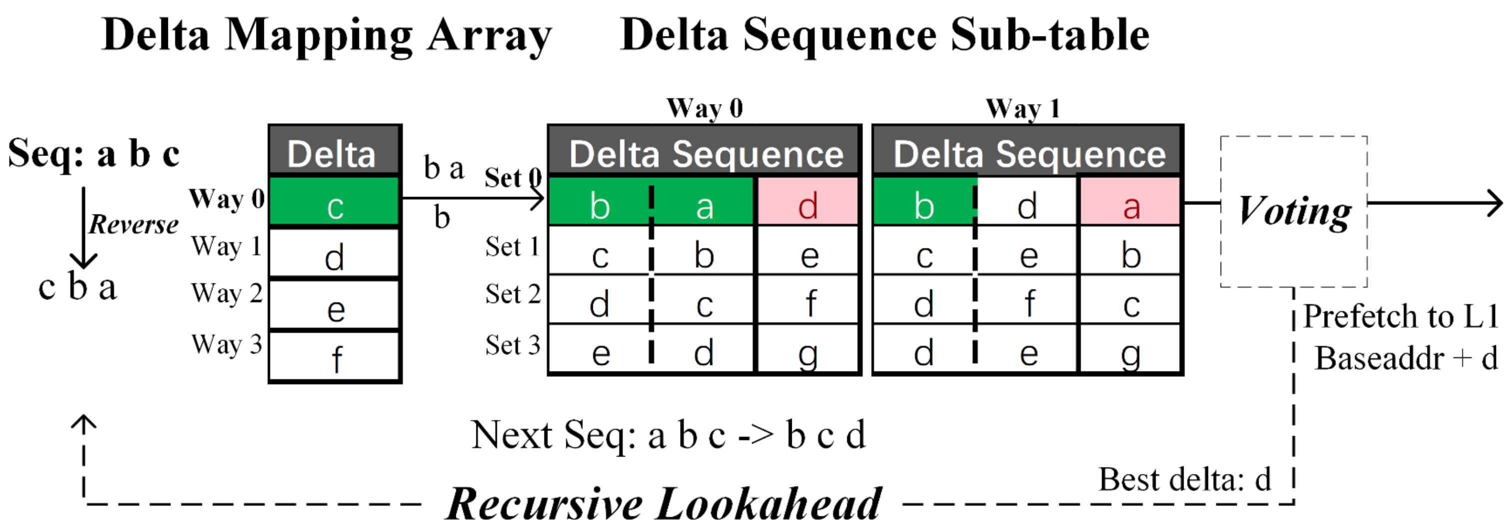
- With same confidences
  - Discard confidences of short sequences
- With different confidences



Coalesce + Reverse

Design

Dynamic Indexing Strategy & Overview



Dynamically record the mapping relations between deltas and table entries

Design

Consider both long and short matches

Adaptive Voting Strategy

$$S\,core_d = \sum_{i \in L} W_i \sum_{f \in M_i} Conf_i$$

$$\frac{Score_d}{Score_{total}} > T_p$$

In our configuration,

$T_p = 0.5$ for prefetching into L1

$W_2 = 3$ *and* $W_3 = 4$ for 2-delta and 3-delta matched prefixes.

W_i is the weight for each matched prefix of length i;

L is the set of the lengths;

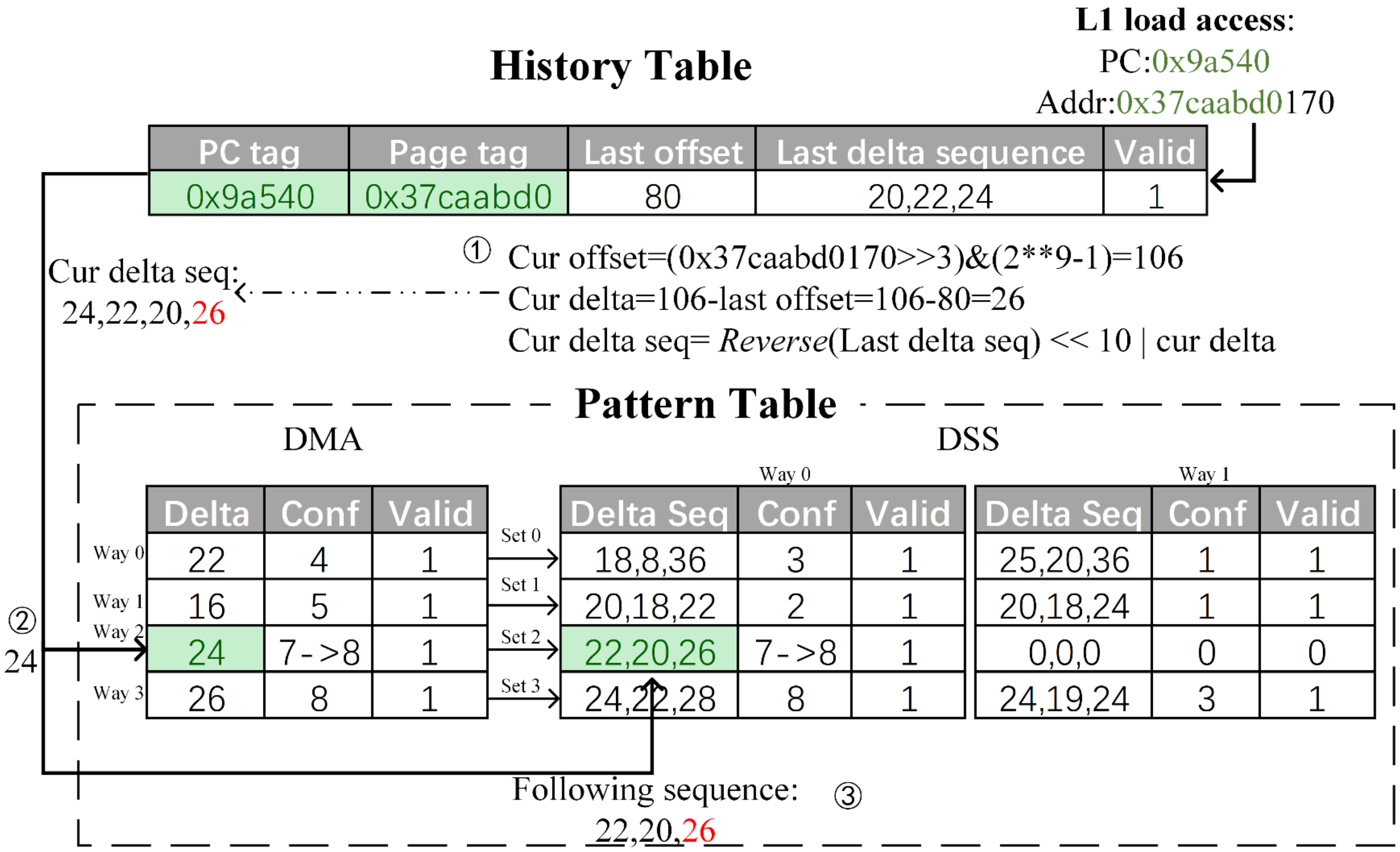
M_i indicates the entries, containing the matched prefixes of length i, that generate d;

$Conf_j$ is the confidence of the corresponding entry j;

D is the set of candidate deltas.

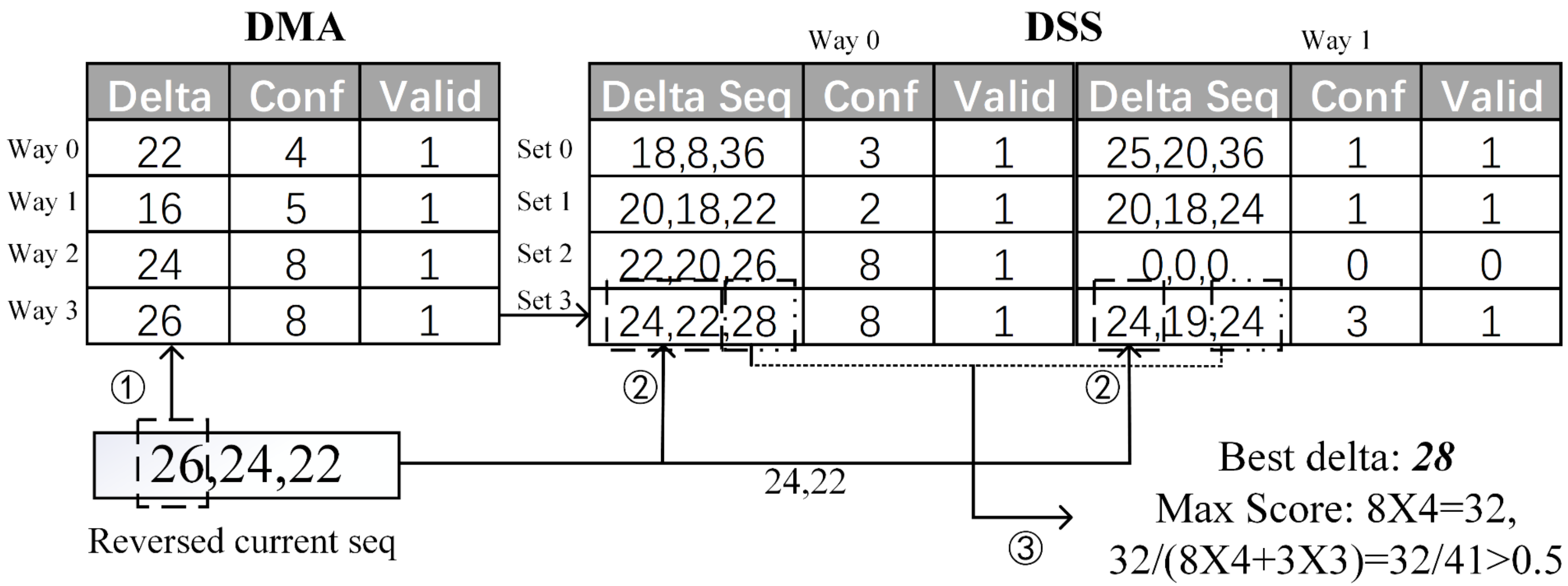
Implementation

Training



Implementation

Predicting



Evaluation: Methodology

Simulalor: ChampSim

Benchmark: 45 traces from SPEC 2017, CloudSuite

Table 1: Detailed Storage Overhead of Matryoshka

Structure	Entry	Field	Storage
History Table	128×1	PC tag (12 bits) Page tag (8 bits) Last offset (9 bits) Last delta sequence (30 bits) Valid (1 bit)	7680 bits
Delta Mapping Array	1×16	Delta (10 bits) Confidence (6 bits) Valid (1 bit)	272 bits
Delta Sequence Sub-table	16×8	Delta sequence (30 bits) Confidence (9 bits) Valid (1 bit)	5120 bits
Candidate Array	128×1	Score (10 bits)	1280 bits
Candidate Offset Array	32×1	Score (10 bits)	320 bits
$Total : 14,672 bits \approx 1.79KB$			

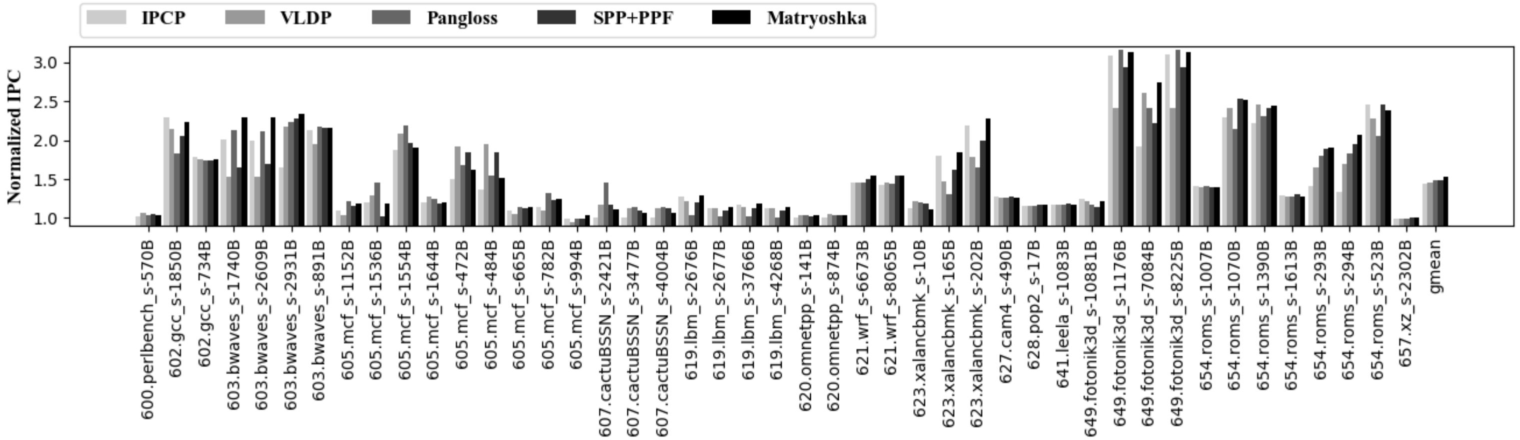
Table 3: Prefetcher Overhead

VLDP	SPP+PPF	Pangloss	IPCP	Matryoshka
48.34 KB	48.39KB	45.25KB	740 B	1.79KB

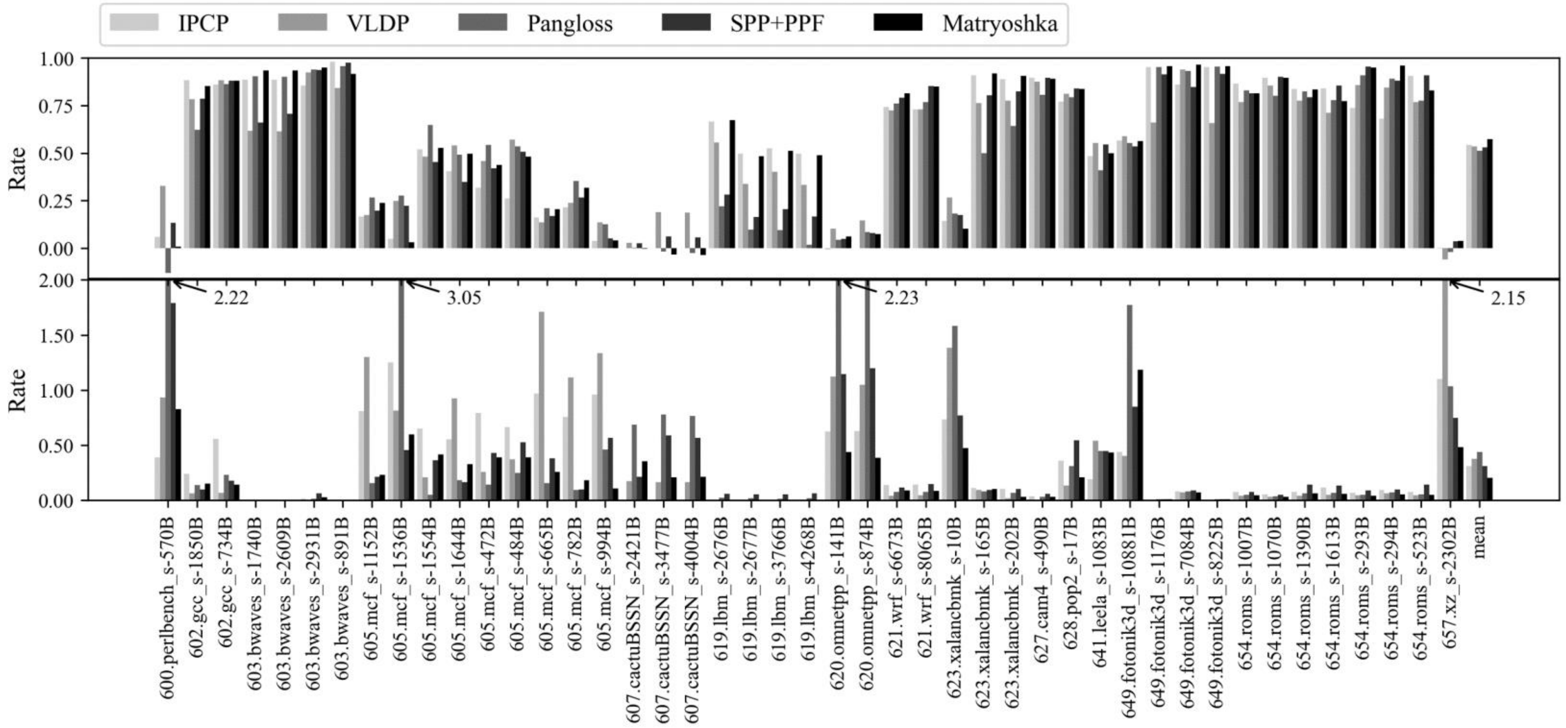
we expand VLDP’s storage capacity to 48 KB

Evaluation: Single-core

Matryoshka yields the best geometric mean speedup of 53.1% over the non-prefetching system, an improvement of 6.5% over IPCP.

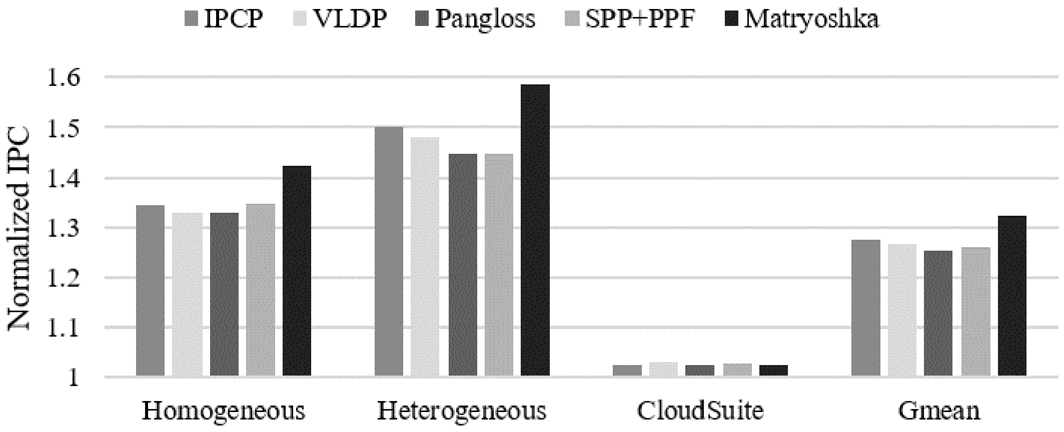


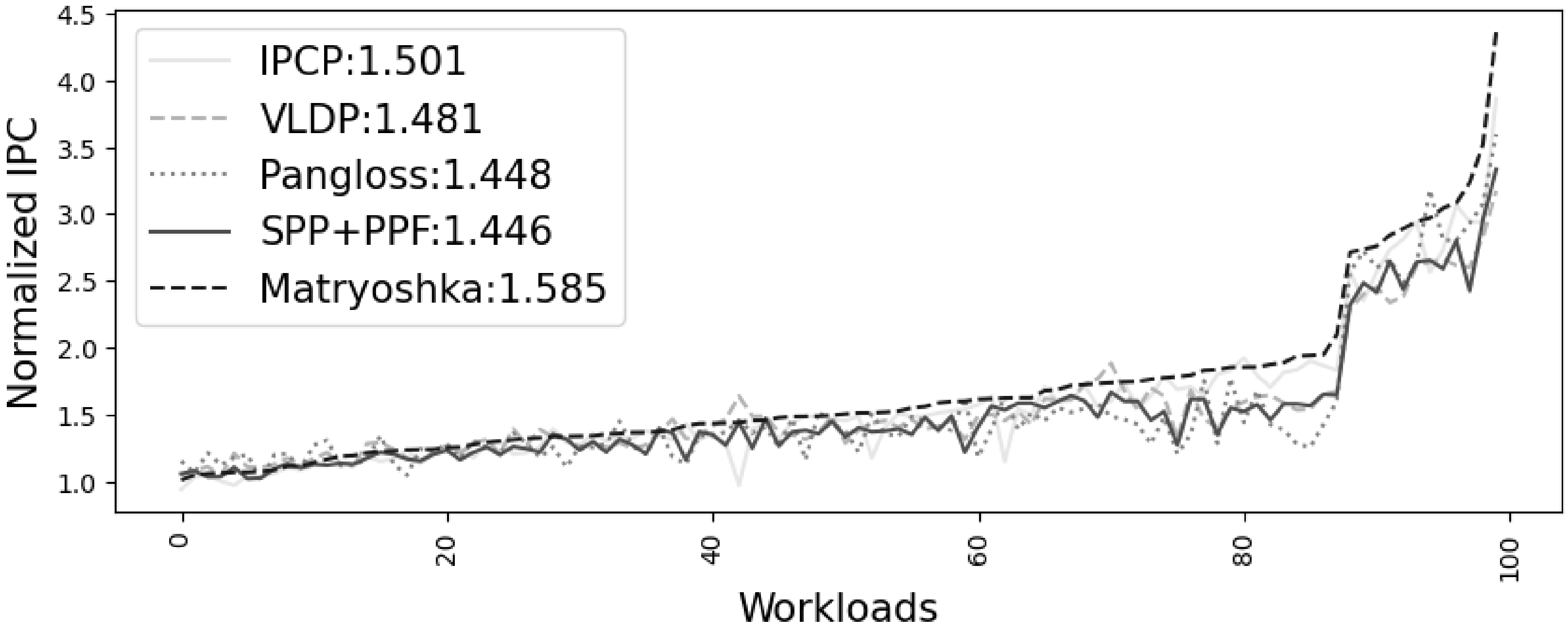
- ➢ Compared with the other prefetchers, Matryoshka achieves the highest average coverage (57.4%) at L1, which exceeds the second best IPCP by 6.0%.
- ➢ The average overprediction rate for Matryoshka is 20.6%, which is the lowest.



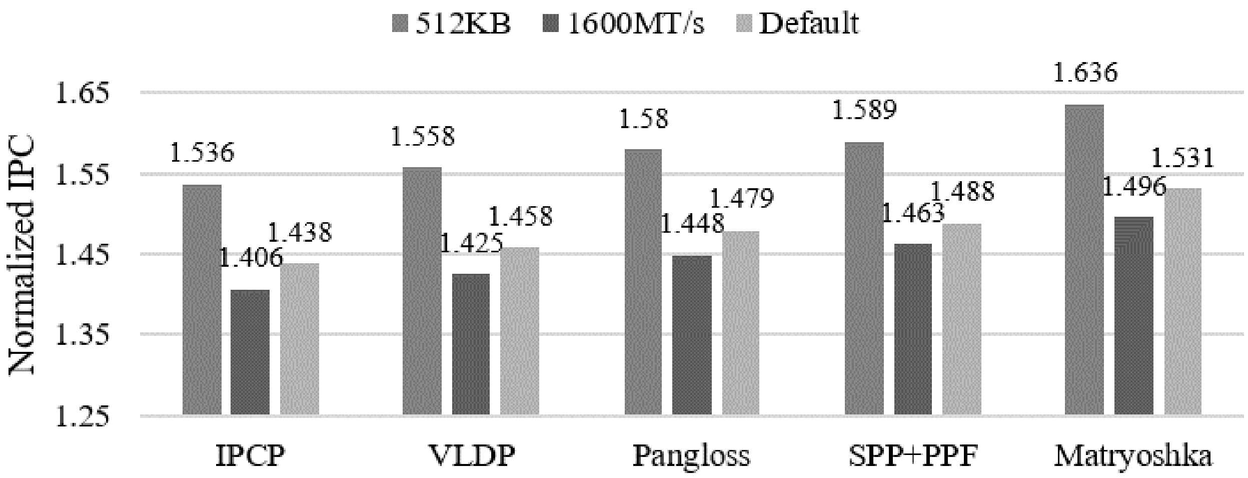
Evaluation: Multicore

In the 4-core system, Matryoshka offers a 32.2% speedup over the baseline





Evaluation: Sensitivity Study



Compared with the other four prefetchers, Matryoshka still obtains better performance in different configurations of the cache system.

Conclusion

- Coalesced Delta Sequence
    - Simpler Structure for Multiple Matching
- Adaptive Voting Strategy
    - Ensure both coverage and accuracy
- Dynamic Indexing Strategy
    - Keep key information