PMPM: **P**rediction by Combining **M**ultiple **P**artial **M**atches

Hongliang Gao Huiyang Zhou

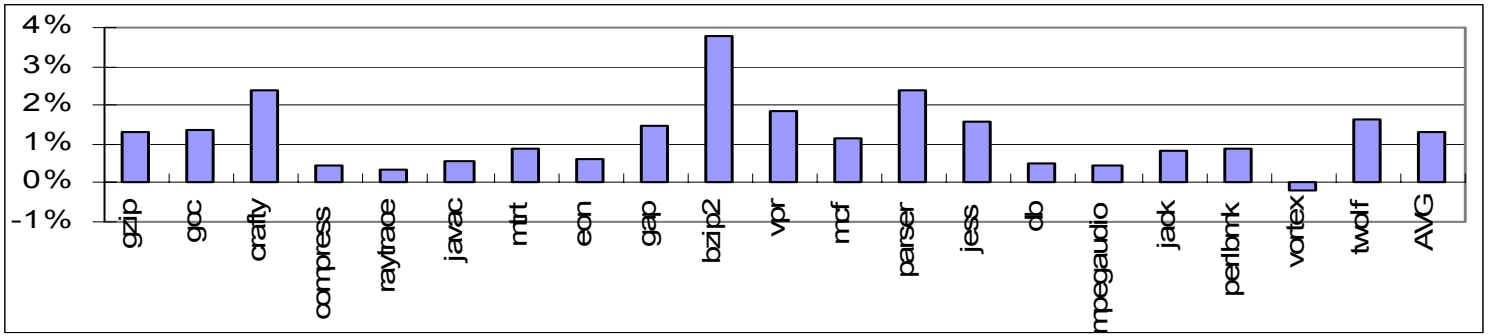
Computer Science Department
University of Central Florida

Prediction by Partial Matching (PPM)

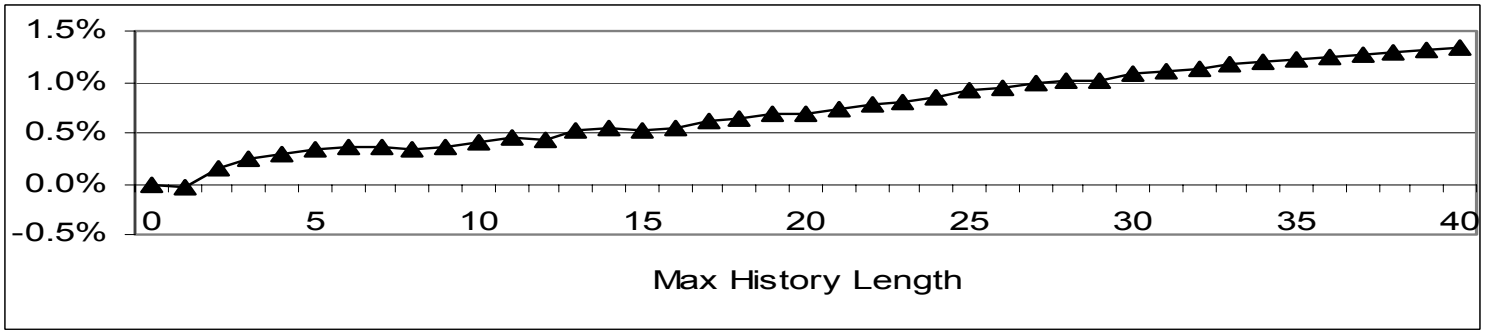
- Originally proposed for data compression by Cleary and Witten. Introduced to branch prediction by Chen et. al.
- For branch prediction:
  - Each static branch has a set of Markov predictors from order 0 to order *m*.
  - The "longest match" policy: use the *m* immediately preceding history bits to search a pattern in the highest order Markov predictor.
- Assumptions of the PPM algorithm
  - Longer history provides a more accurate context (true).
  - A prediction counter associated with a more accurate context will provide higher prediction accuracy (false).

The "longest match" policy is not optimal

- The confidence-based PPM
  - Use the longest *confident* *(ctr <> 0)* match.
  - Misprediction rate (MPKI) reductions vs. PPM.
  - Max H = 40:



  - Max H = 0 to 40:

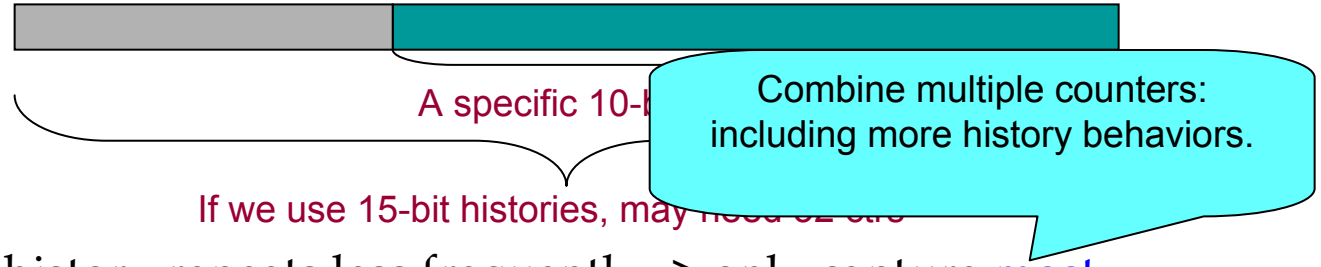


Introduction

- Key observation on PPM
  - The "longest match" policy is not optimal for branch prediction.
- ***Our contributions***
  - ***A novel algorithm: Prediction by combining Multiple Partial Matches (PMPM).***
  - ***A PMPM-based idealistic branch predictor.***
  - ***A PMPM-based realistic branch predictor.***

Prediction by combining Multiple Partial Matches

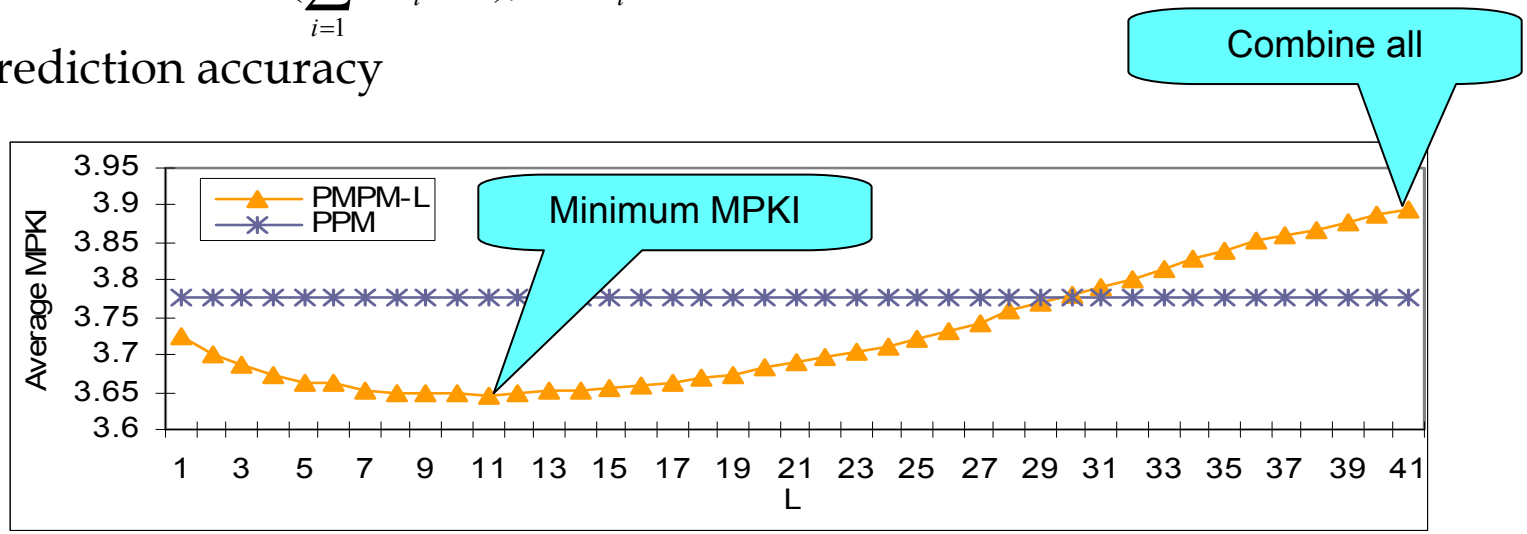
- Different branches favor different history lengths
  - Using a longer history than necessary:
    - Uncorrelated history information -> noise -> distribute useful information into more prediction counters.



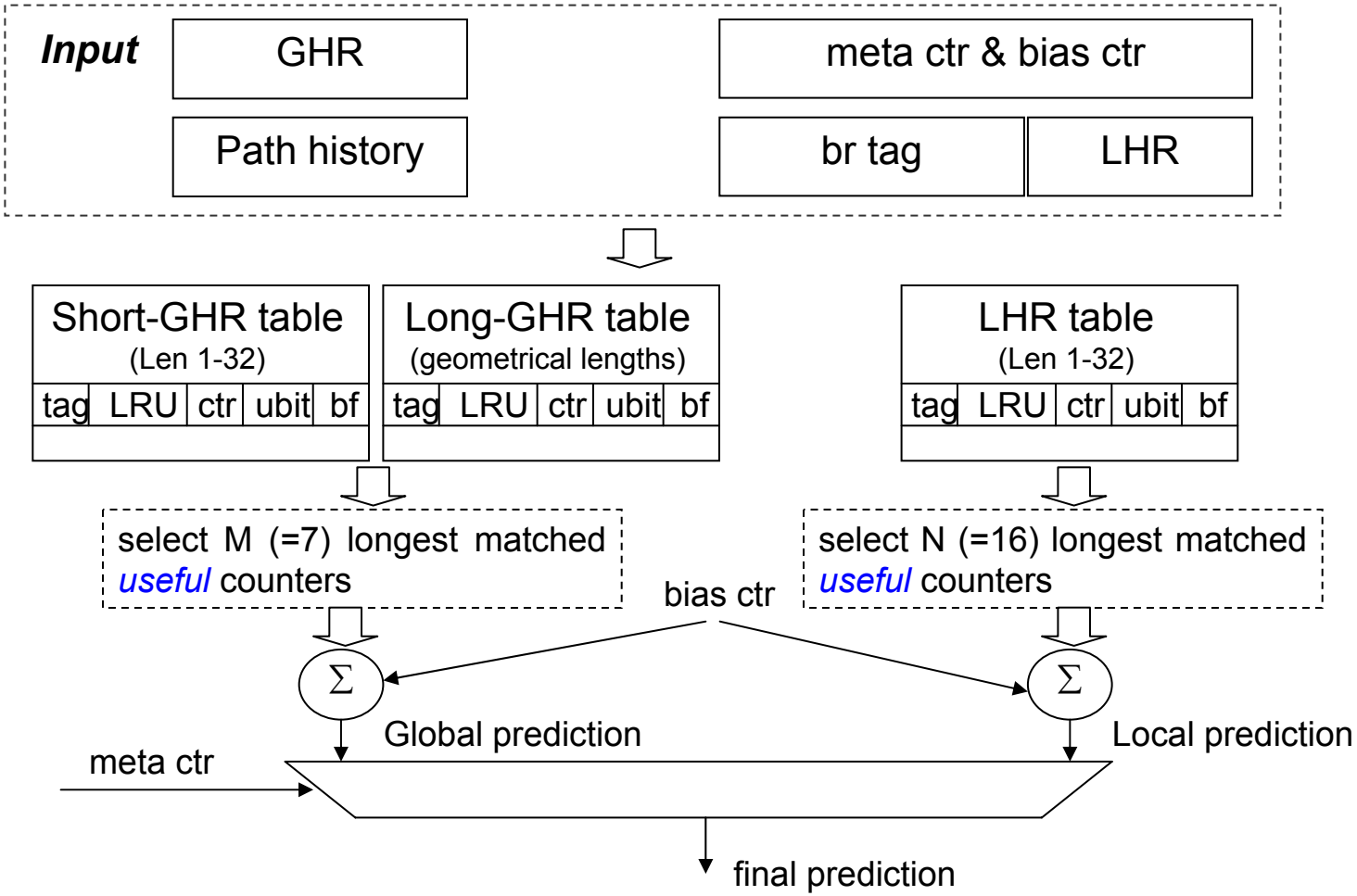
    - Long history repeats less frequently -> only capture most recently behaviors
      - Especially harmful for "not-correlated / random-like" branches.
- Solution
  - Combining multiple matches
    - Why?
    - How: summation -> integrates both direction AND confidence.
    - Which: several longest confident matches with non-zero prediction counters.

Prediction accuracy of PMPM

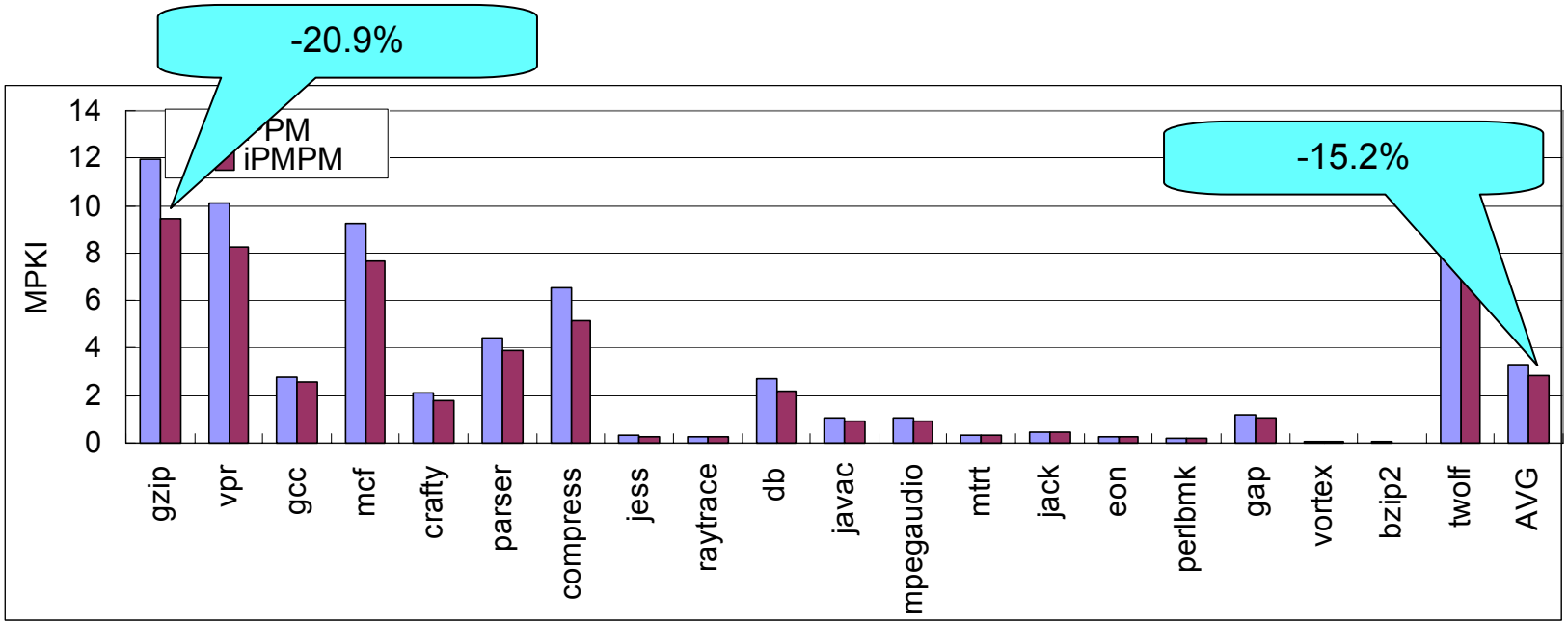
- Configuration
  - Combine the L longest confident matches.
  - Maximum global history length: 40.
  - Prediction = $(\sum_{i=1}^{L} Ctr_i \geq 0), \quad Ctr_i \neq 0$
- Prediction accuracy



The idealistic PMPM predictor

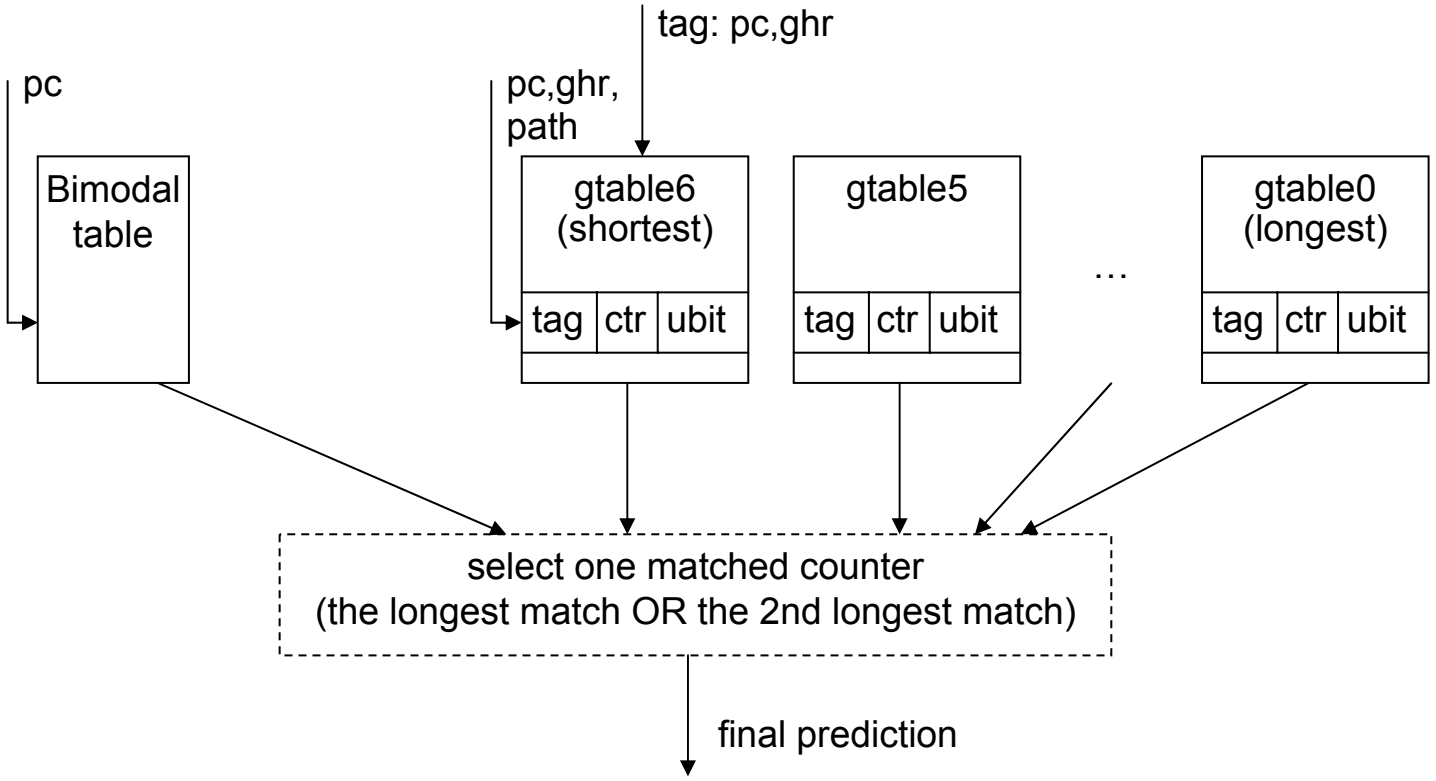


Prediction accuracy of the idealistic PMPM predictor

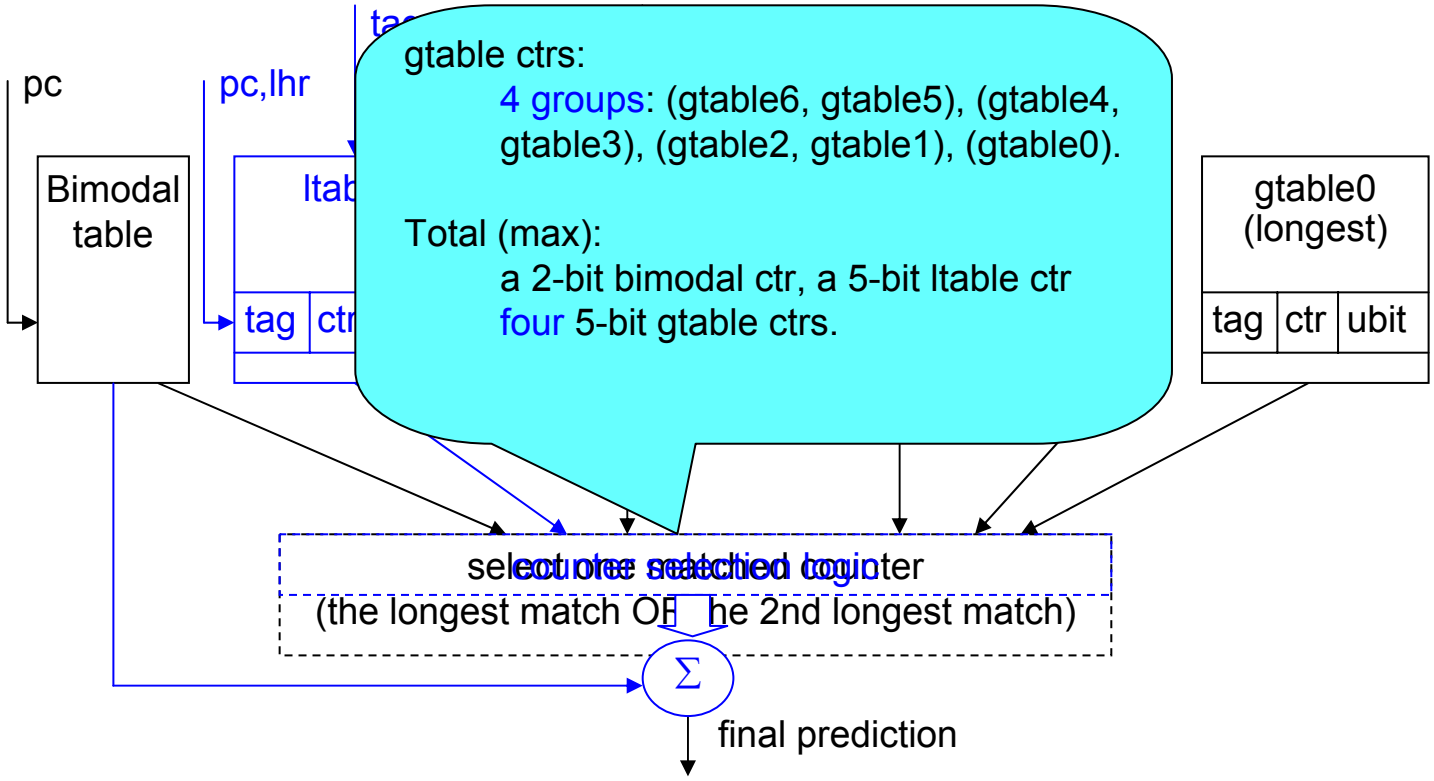


- PPM: Same predictor structure, but using the "longest match" prediction policy.
- Average MPKI
  - PPM: 3.330
  - PMPM: 2.824

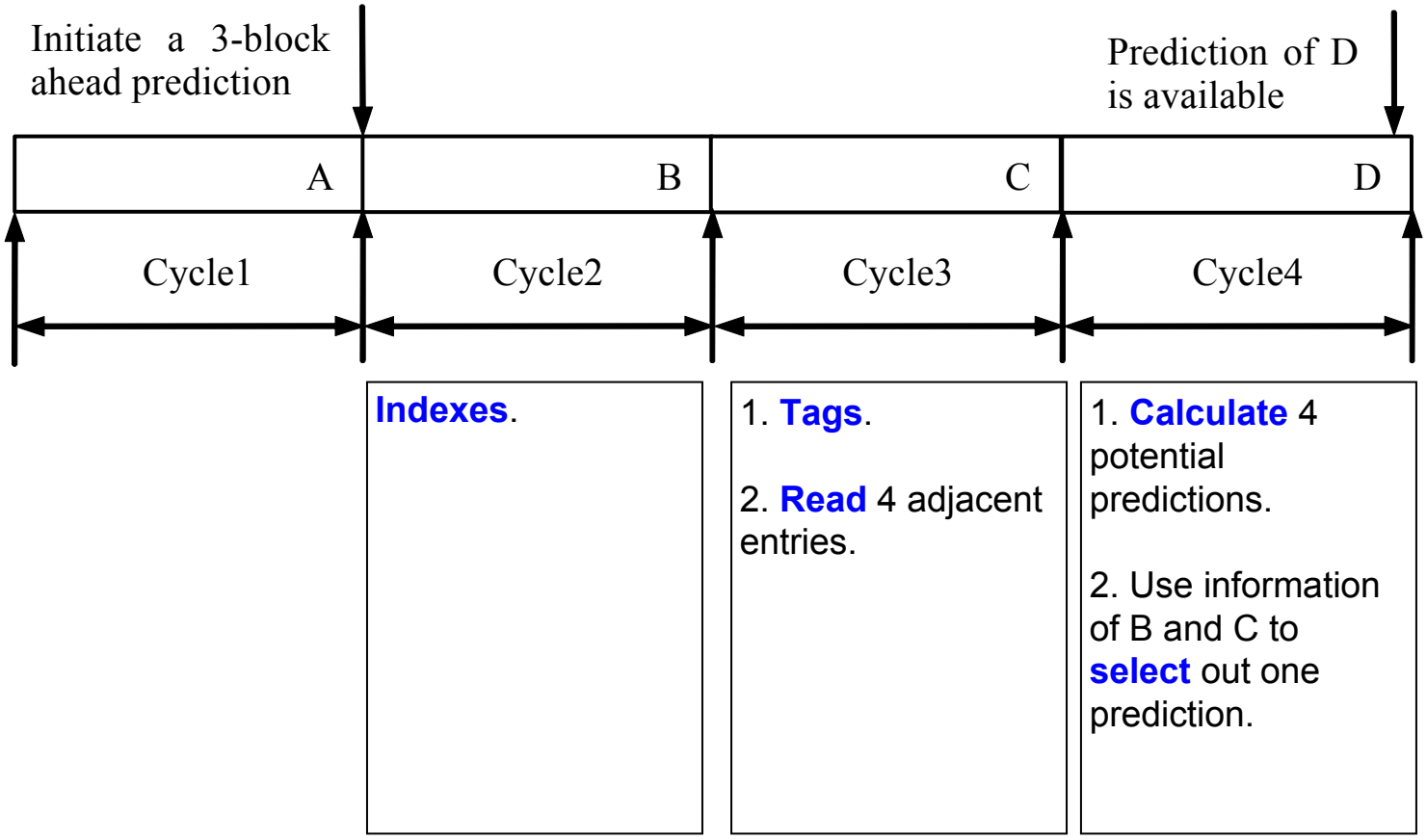
The TAGE predictor



The realistic PMPM predictor



Ahead pipelining



Compared to the TAGE predictors

- Configuration
  - 32kB, same global history series (5 - 131), similar structures.
  - Compared to the TAGE predictor:
    - PMPM-G (GH only): 2-bit larger ctrs, 2-bit smaller tags.
  - Compared to the PMPM-G predictor:
    - PMPM-GL(GH and LH): one ltable, smaller bimodal table, smaller tags for 3 gtables.
- Average MPKI:
  - TAGE: 3.666
  - PMPM-G: 3.597 (higher aliasing, gcc +7.3%)
  - PMPM-GL: 3.441

The realistic PMPM predictor for CBP2

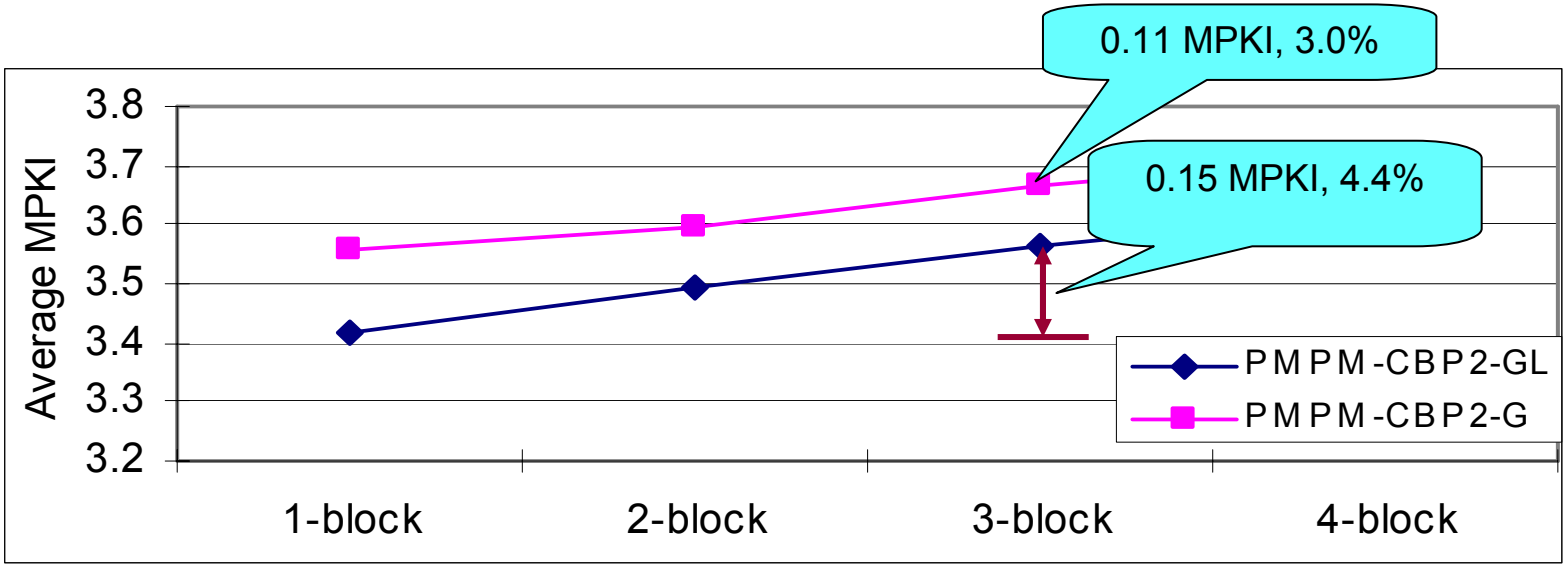
- Submitted configuration
  - Save some storage for miscellaneous registers, counters etc.
  - Empirically tuned inputs, tag widths etc.
- Several optimizations
  - Shared hysteresis bits in the bimodal table (proposed in the EV8 predictor).
  - Detect traces with high branch footprints and reset *ubits* periodically (borrowed from the TAGE predictor).
  - Limited *ubit* updates if all predictions from gtables are same.

The realistic PMPM predictor for CBP2 - accuracy

- Observations:
  - High accuracy: 3.416 MPKI
  - The local history is still important for some benchmarks (e.g., *raytrace, mtrt and vortex*) although we already use a very long (203) global history.

Trace	CBP2-GL	CBP2-G	Trace	CBP2-GL	CBP2-G
gzip	9.712	10.346	vpr	8.945	9.063
gcc	3.690	3.637	mcf	10.092	10.033
crafty	2.581	2.565	parser	5.215	5.244
compress	5.537	5.819	jess	0.393	0.433
raytrace	0.542	0.963	db	2.319	2.380
javac	1.107	1.159	mpegaudio	1.102	1.159
mtrt	0.657	1.009	jack	0.688	0.763
eon	0.276	0.359	perlbmk	0.314	0.484
gap	1.431	1.745	vortex	0.137	0.331
bzip2	0.037	0.042	twolf	13.551	13.616
Average	PMPM-CBP2-GL: 3.416		PMPM-CBP2-G: 3.557		

The realistic PMPM predictor for CBP2 – ahead pipelining



Summary

- Key observation on PPM
  - The "longest match" policy is not optimal for branch prediction.
- Solution
  - Prediction by combining Multiple Partial Matches (PMPM)
- PMPM-based predictor designs
  - Idealistic predictor: 2.824 MPKI.
  - Realistic predictor: 3.416 MPKI.

Thank you and Questions?

Computer Science Department
University of Central Florida