Branch Predictors For Sat Solvers

- Team Architects

ASHUTOSH SINGH 210050022

ARYAN KAMAT 210050020

YASWANTH MOTUPALLI 210050099

SAI LOKESH KOMINNENI 210050087

Branch Predictors

- Premise: Branch behavior repeats, which means branch behavior can be learned and predicted.
- Assumption: Outcome of a branch is a function of two inputs:
    1. the address of the branch, which distinguishes it from the other branches
    2. branch history, a sequence of prior branch outcomes.

Branch Prediction Tradeoff & Hybrid Branch Predictors:

On the one hand, longer histories enable accurate predictions for some harder-to-predict branches. On the other hand, with a longer history, the predictor must track more branch scenarios and thus spend more time warming up, reducing accuracy for easier-to-predict branches.

This fundamental branch prediction tradeoff was the inspiration behind **hybrid branch predictors** which use multiple branch histories. Roughly speaking, for each branch, hybrid predictors track prediction accuracy for that branch given different history lengths. The history length that results in highest accuracy is the one used to generate the predictions for that branch.

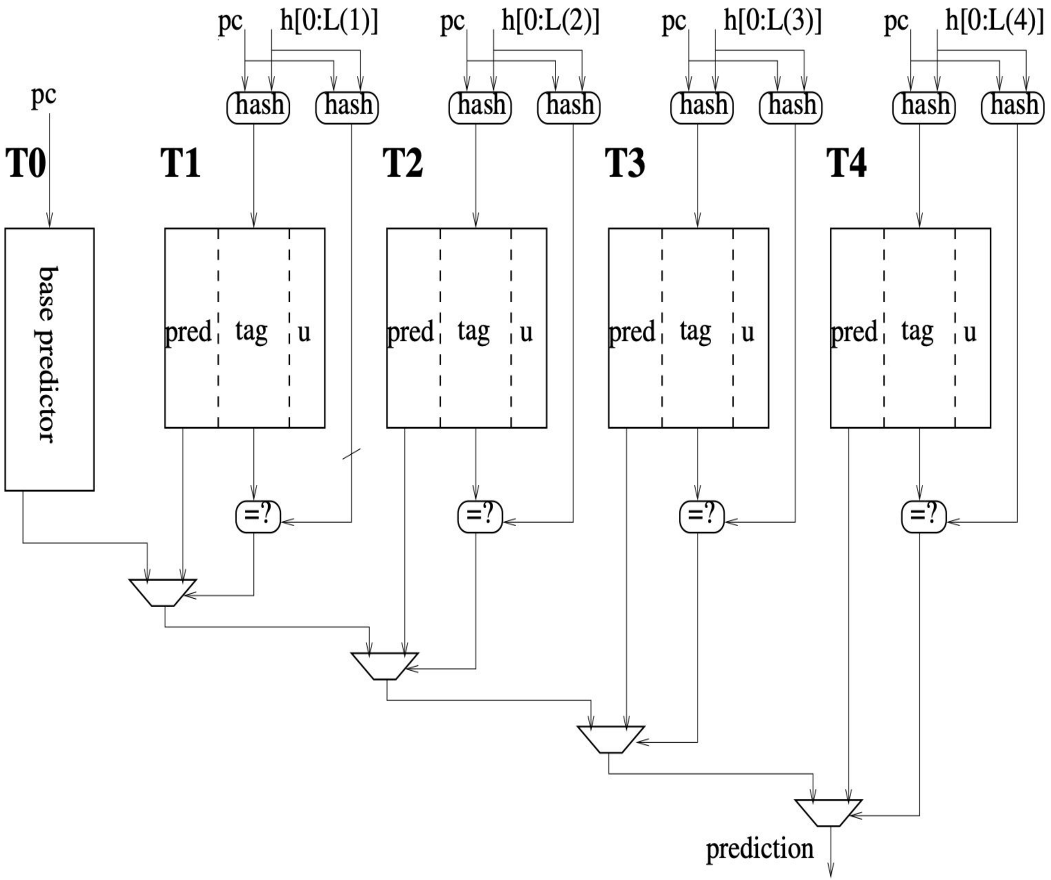
TAGE Predictor

TAgged GEometric length predictor

- TAGE is one such hybrid branch predictor with three major improvements that set it apart:
  - Entry Tagging
  - Entry Selection
  - Longer Maximum History
- Tagless bimodal base predictor + partially tagged components indexed using history lengths in GP
- Prediction by either base predictor or tag match on a tagged component
- The three improvements given above are the reason why TAGE outperforms all other branch predictors.

TAGE

A base predictor + several tagged predictor components indexed with increasing history lengths



Geometric History Length Prediction

- M distinct predictor tables
- Indexed with hash functions of branch address and global branch history
- Distinct history lengths used for computing index of distinct tables
- Base table T_0 indexed using branch address only
- For table T_i (i = 1 to M-1), history length:

$$L(i) = (int)(\alpha^{i-1} * L(1) + 0.5)$$

Update Policy

- Useful Counters:
    - Each tagged component entry has a useful counter u
    - The useful counter u of the provider component is updated when the alternate prediction is different from the final prediction i.e $altpred \neq pred$
    - The useful u counter is also used as an age counter and it is reset periodically. The period used in the presented predictor for this alter- nate resetting is 512K branches.
- Prediction Counters:
    - The prediction counter of the provider component is updated.
    - When the useful counter of the provider component is null, the alternate prediction is also updated.

Update Policy

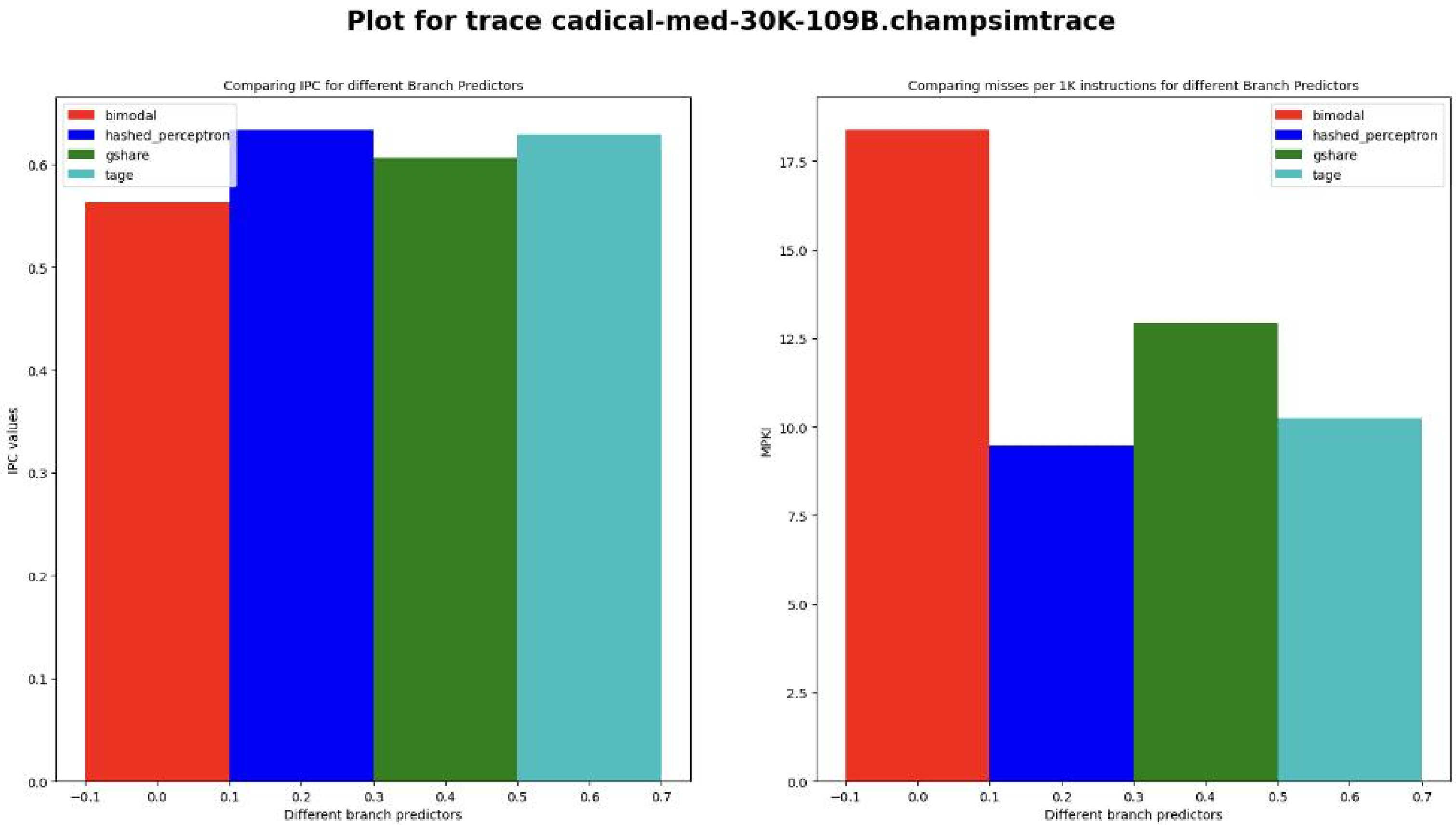
- Allocating tagged entries on mispredictions :
  - On mispredictions at most one entry is allocated.
  - If the provider component Ti is not the component using the longest history (i.e. i <= M), we try to allocate an entry on a predictor component Tk with i < k <= M using the following process:
    - The M-i counters are read from predictor components Tj, i < j <= M . Then we apply the following rules:
      - Avoiding ping-pong phenomenon: in the presented predictor, the search for a free entry begins on table Tb, with b=i+1 with probability 1/2, b=i+2, with probability 1/4 and b=i+3 with probability 1/4. The pseudo-random generator used in the presented predictor is a simple 2-bit counter.
      - Initializing the allocated entry: An allocated entry is initialized with the prediction counter set to weak cor- rect. Counter u is initialized to 0 (i.e., strong not useful).

Why this update policy?

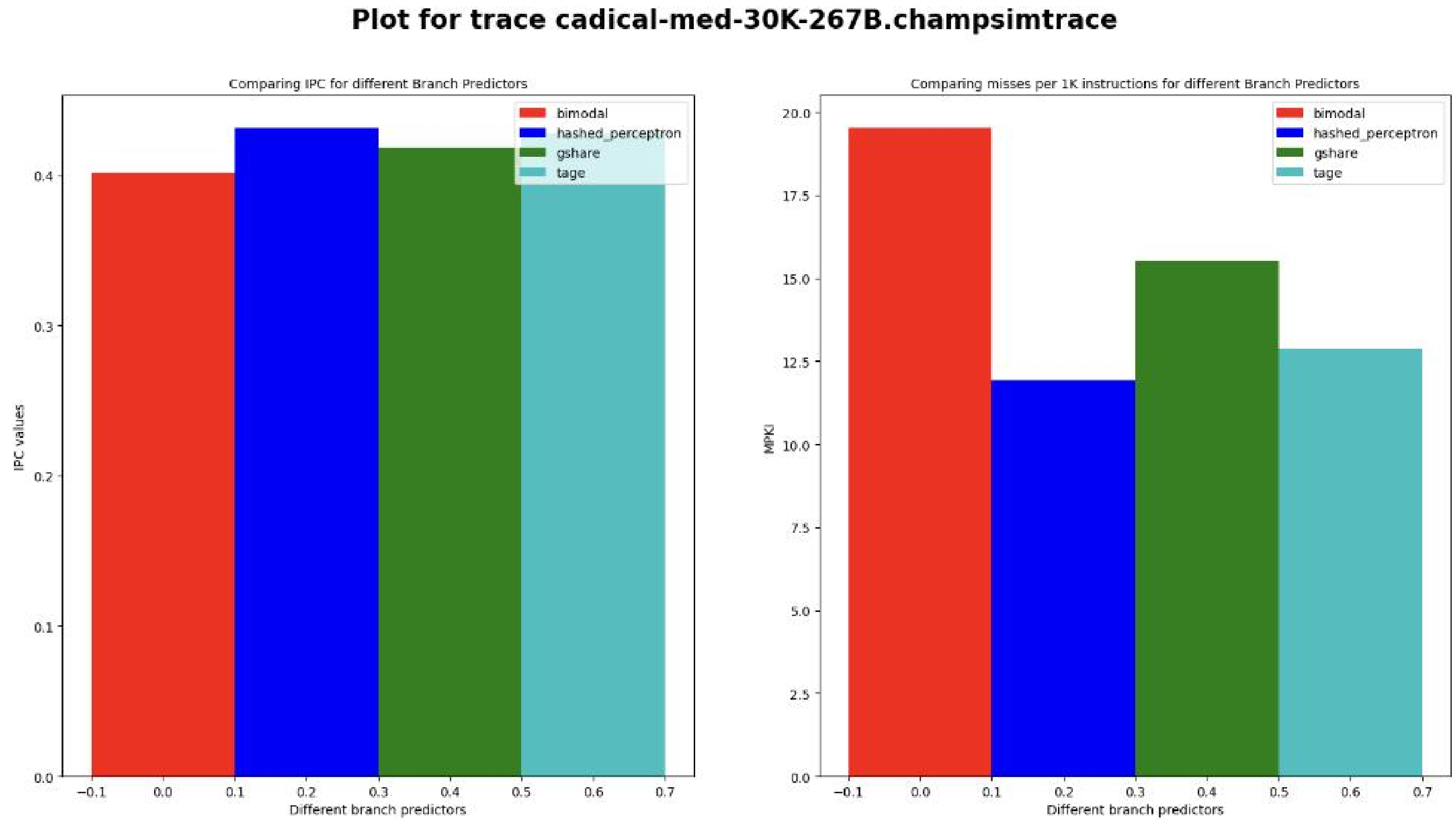
- Minimizes perturbation caused by single occurrence of a branch
- At most one tagged entry is allocated on a misprediction
- The useful counter u helps mimic a pseudo LRU policy

What did we observe?

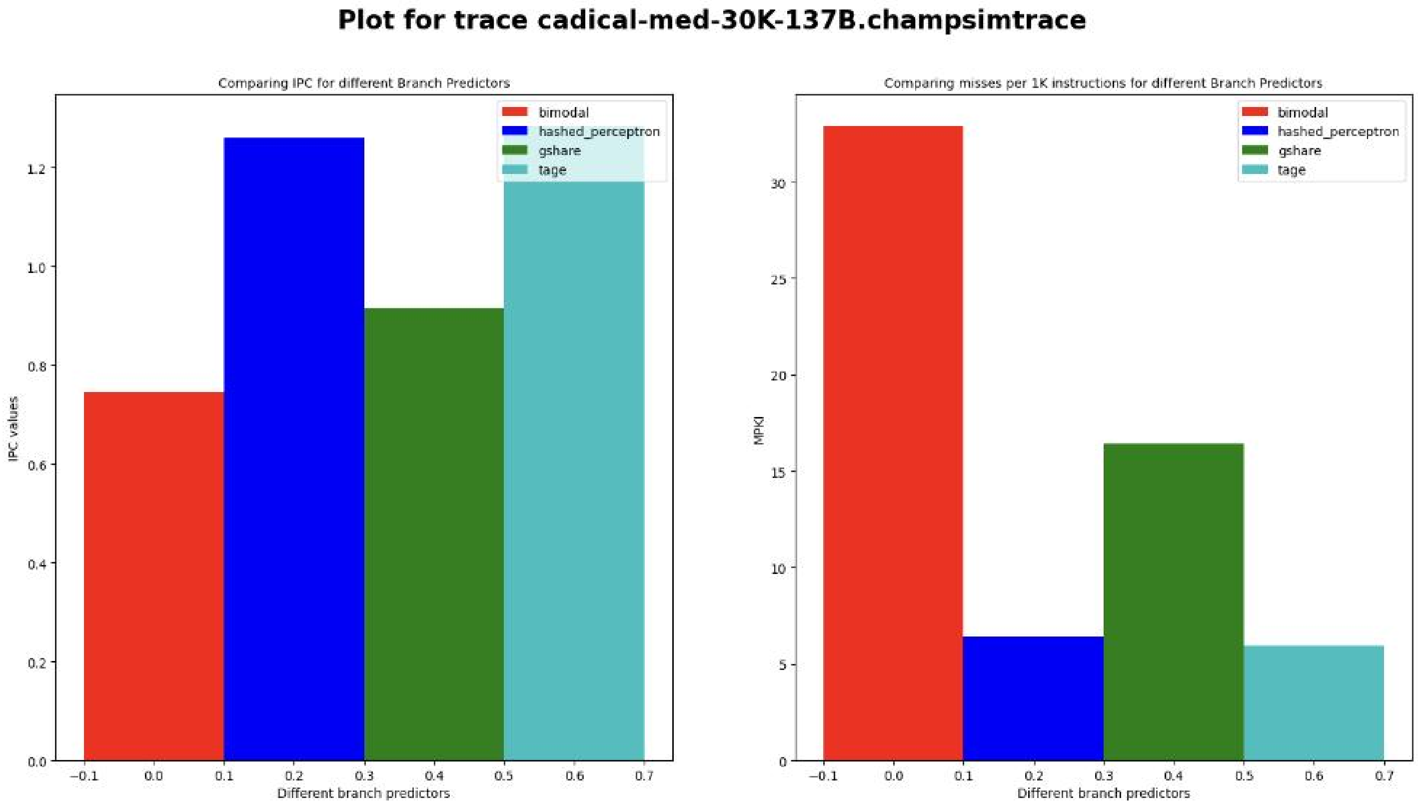
Comparison of IPC and MPKI values of different branch predictors for cadial-med-30K-109B.champsimtrace



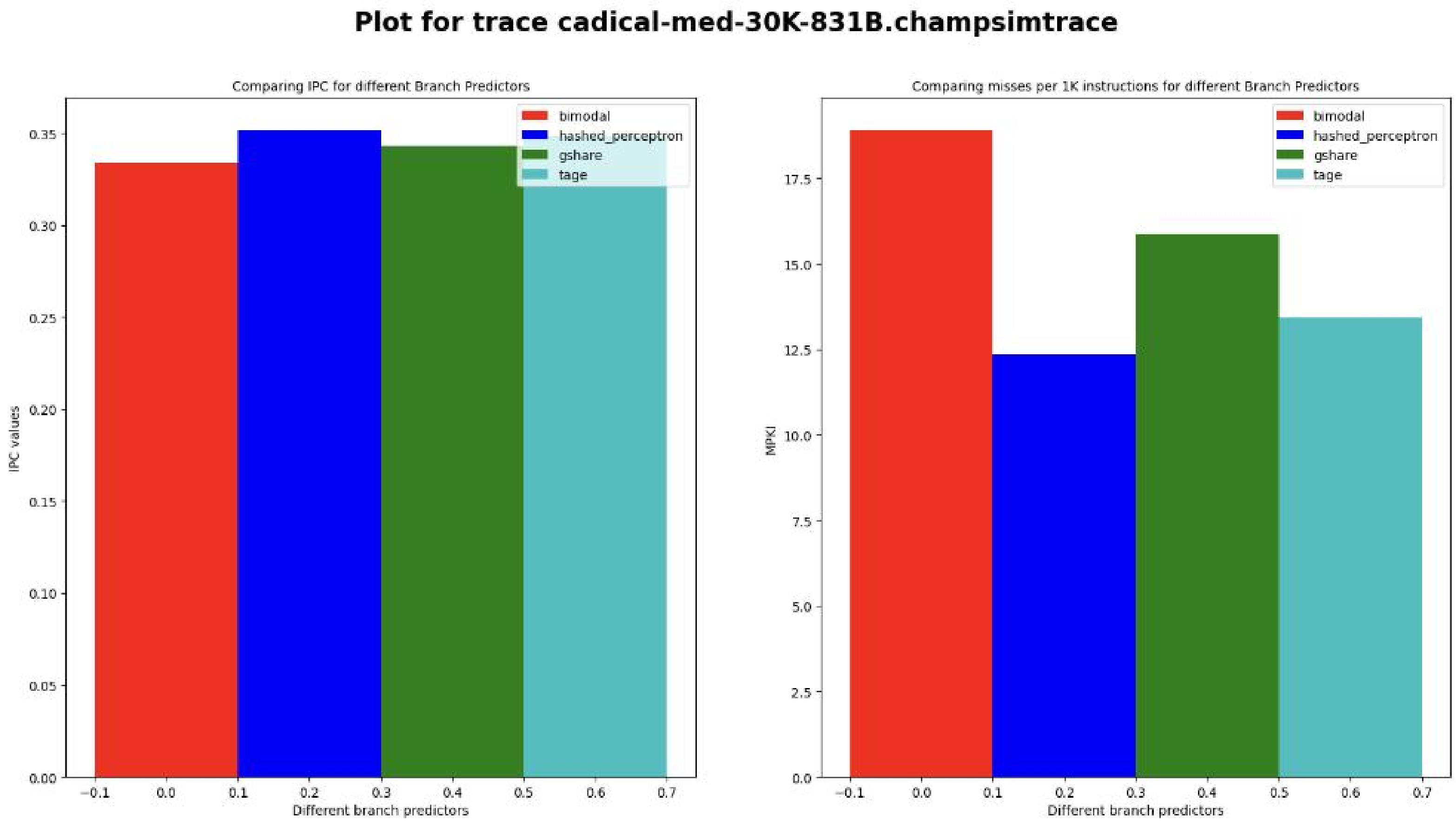
Comparison of IPC and MPKI values of different branch predictors for cadial-med-30K-267B.champsimtrace



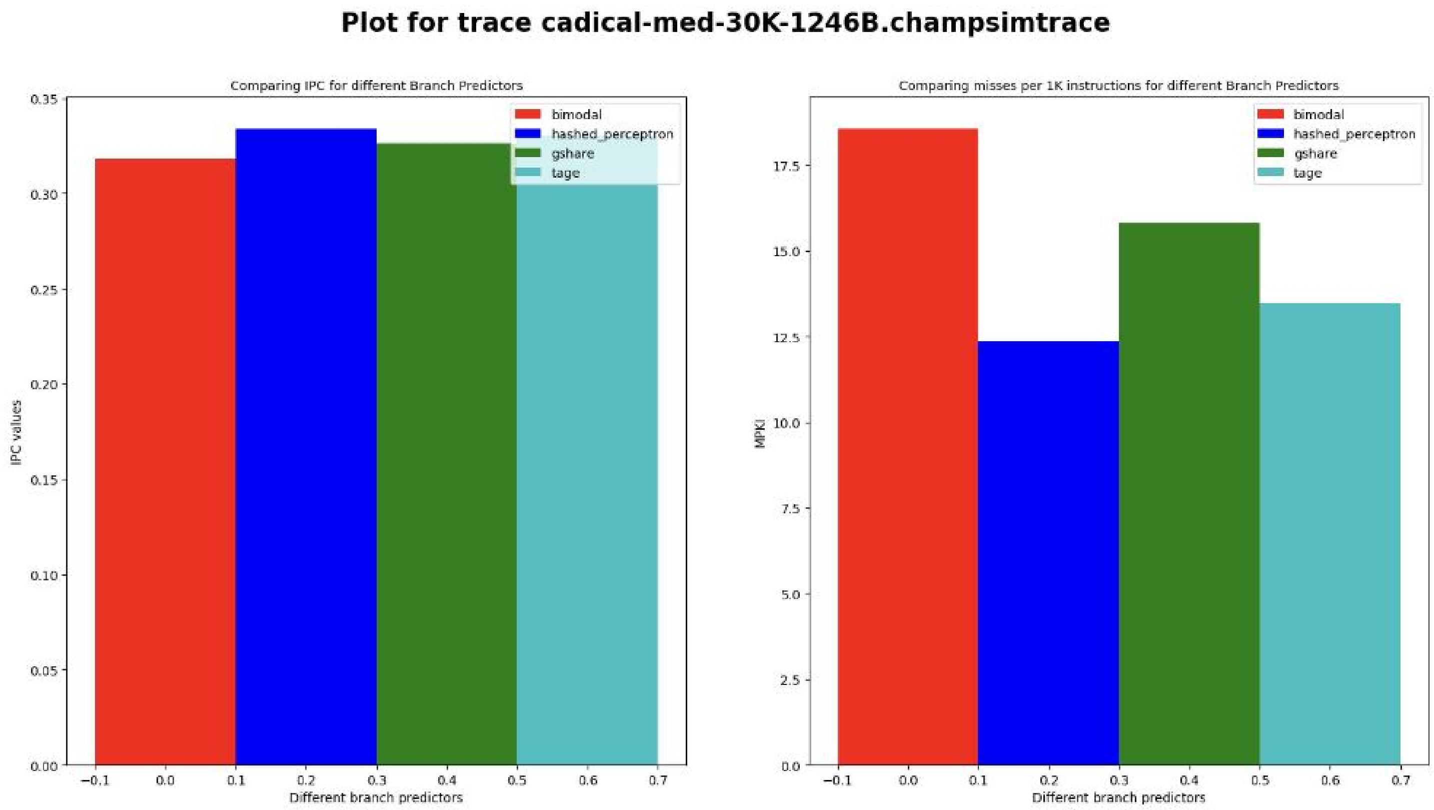
Comparison of IPC and MPKI values of different branch predictors for cadillac-med-30K-137B.champ simtrace



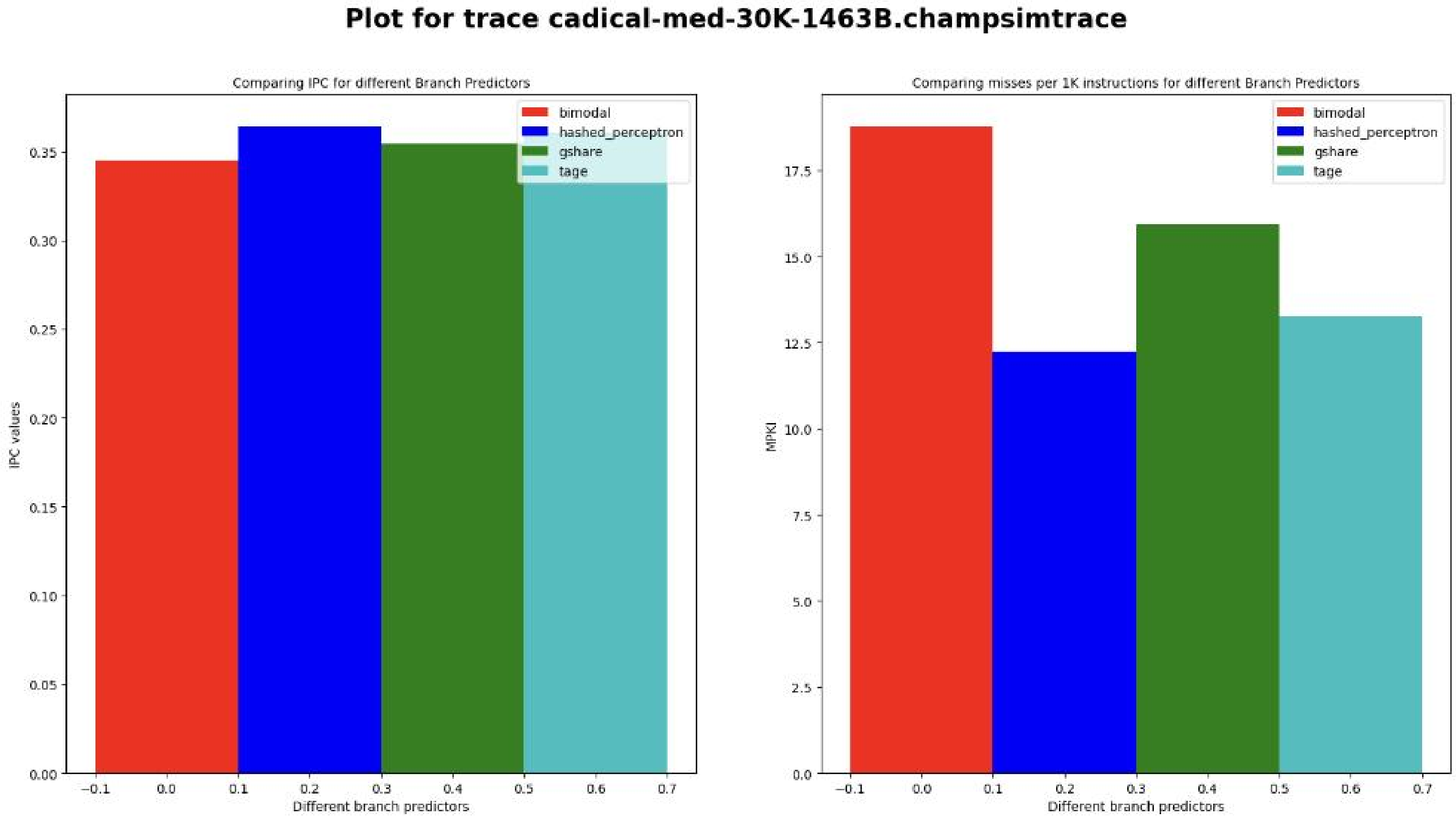
Comparison of IPC and MPKI values of different branch predictors for cadillac-med-30K-831 B.champ simtrace



Comparison of IPC and MPKI values of different branch predictors for cadillac-med-30K-1246B.champ simtrace



Comparison of IPC and MPKI values of different branch predictors for cadillac-med-30K-1463B.champ simtrace



WHAT WE LEARNT?

CONCLUSION

1.Initially the effectiveness of geometric length-based branch predictors, such as O-GEHL, was significantly enhanced by us with the introduction of TAGE.

2.WE DEVELOPED A RATIONALE TO SUPPORT THE SELECTION OF HISTORY LENGTHS FOR THE GLOBAL PREDICTOR, UPDATE RULE, AND LOOP PREDICTOR COMPONENTS IN L-TAGE, BASED ON OUR UNDERSTANDING AND INSIGHTS.

CONCLUSION

3.Implementing these predictors efficiently is essential to minimize hardware cost and energy consumption.

- 4.THE ANALYSIS AND IMPLEMENTATION OF TAGE AND L-TAGE HAVE DEEPENED OUR UNDERSTANDING OF BRANCH PREDICTORS AND PROVIDED VALUABLE INSIGHTS INTO THEIR DESIGN AND OPTIMIZATION. WHICH FINALLY LEAD US TO PRESENT OUR CODE AND SHOW EXTENSIVE RESULTS BY STUDYING EACH MAJOR COMPONENT OF L-TAGE AND TAGE IN DETAIL.

References

Branch predictors for SAT solvers
Paper: https://www.irisa.fr/caps/people/seznec/L-TAGE.pdf
Blog: https://comparch.net/2013/06/30/why-tage-is-the-best/
Simulator: ChampSim

Traces:
https://www.dropbox.com/sh/xs2t9y4cuqlgrlp/AACpzGOj6BcSB-BUolGaBjbta?dl=0