Exploring branch predictability limits with the MTAGE+SC predictor*

André Seznec
Inria/Irisa, Rennes

1 Outline

In the previous championship CBP-4, the winner of the unlimited storage track [5], poTAGE-SC was combining several TAGE based predictors using different forms of histories (local, global, and frequency), a COLT inspired [3] prediction combiner and a statistical corrector (SC) predictor [8, 10] fed with various forms of branch histories.

With MTAGE-SC, we improve this predictor in two ways. First through incorporating new forms of branch histories, adding a new TAGE component and incorporating other forms of histories in the statistical corrector predictor. Second in conveying more information from the TAGE predictors stage to the statistical corrector and to the final prediction computation stage.

On the CBP-4 traces, the proposed MTAGE-SC predictor achieves 1.600 mispredictions per thousand instructions (MPKI), while the winner of CBP-4 was achieving 1.691 MPKI. On CBP-5 train traces, the MTAGE-SC predictor achieves **2.596** MPKI, 4.7% lower than the winner of CBP-4 (2.717)[1].

2 MTAGE-SC predictor outline

The prediction computation on MTAGE-SC is illustrated on Figure 1.

On MTAGE-SC, the prediction is computed in 4 steps. First, the TAGE predictors stage provides predictions and associated confidences (counter values).

Second this information is fed in the TAGE prediction combiner that provides two predictions and some confidence estimation (2 bits). Third a statistical correlator or SC [8, 9] uses the second stage output, and many forms of branch history (global, local, path, etc) to compute a prediction and a confidence estimation. Finally the predictions and confidences flowing from the TAGE prediction combiner and from SC are combined in a final prediction.

3 The TAGE predictors

TAGE has been presented as a global history predictor [12]. However prediction by partial matching can be adapted to other forms of input information as long as one can extract a set of strictly increasing information vectors. Therefore as in [4, 5], MTAGE+SC features a first stage composed of several TAGE predictors, 6 in the proposed design, all with different form of histories. As in [4, 5], we use a conventional global branch/path history, 3 local histories (respectively per address and per set), one frequency-based history [4, 5] TAGE predictor. The 6th TAGE predictor uses a global backward branch history, i.e. only backward branches are included in the history.

3.1 Predictor update

For limiting the very long cold start impact that is encountered on unlimited storage predictors, we adopt an aggressive update policy as in [4, 5]. During the warming phase, the following policy is used:

- Instead of updating only the longest hitting counter, all the hitting predictor entries providing the same prediction as the longest matching entry are updated whether or not the branch was correctly predicted.
- We systematically allocate entries for all the path lengths greater than the longest hitting length, whether or not the branch was correctly predicted. We stop doing aggressive allocation for path lengths longer than 200 branches if all the counters of the hitting TAGE entries are saturated.

When the warming phase is over, we switch to the careful update policy implemented in the ISL-TAGE predictor [8].

*This work was partially supported by an Intel research grant

[1] To limit computation time, simulations of LONG-MOBILE-10 and SHORT-MOBILE-22 were stopped after 100 million branches

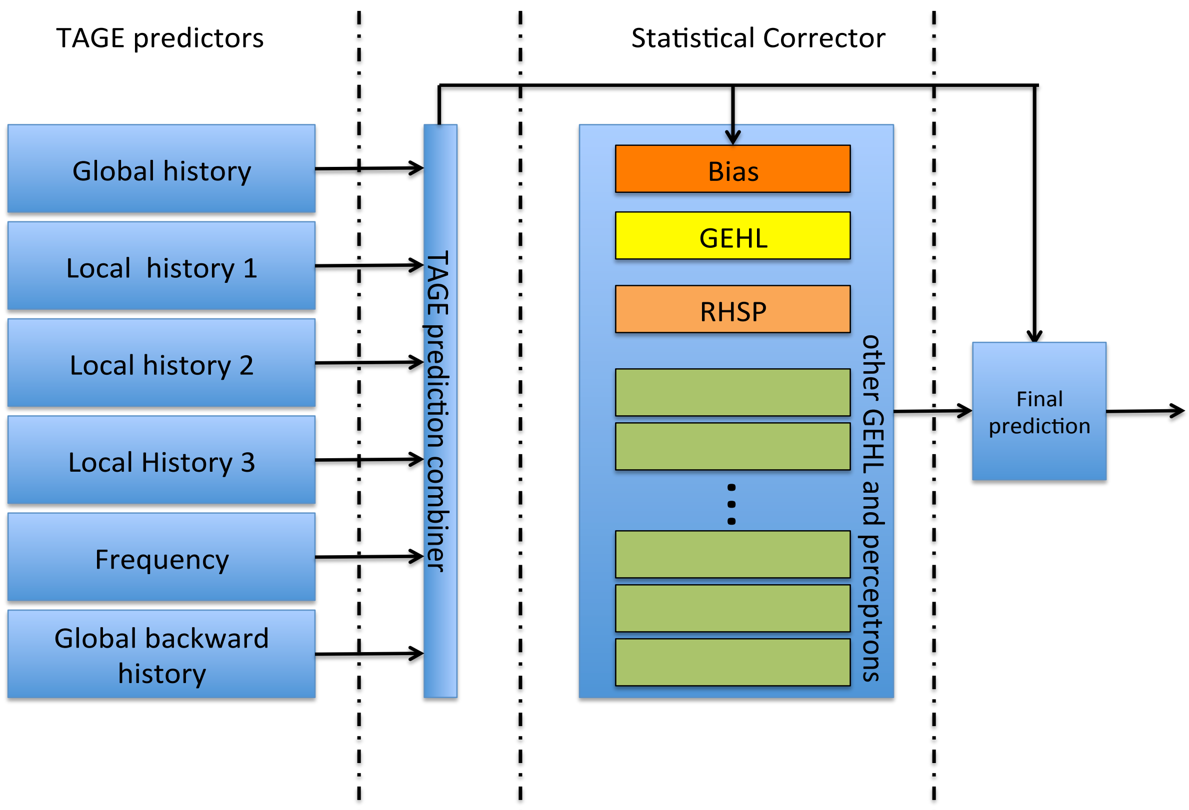

Figure 1: Prediction computation flow on the MTAGE-SC predictor

For the submitted predictor, we experimentally determine that an interval of 100,000 mispredictions works fine on the set of training traces.

3.2 The TAGE prediction combiner

On the TAGE-SC-L predictor [10], the statistical corrector uses the prediction flowing out the TAGE predictor as part of the index of some tables. In [4], the prediction information is directly used as an index to get in a table and read the final prediction P_{colt} as suggested for the COLT predictor [3].

For [5], this COLT prediction is used in the index of some tables for the SC predictor. In the submission for the limited budget track [11], we use extra information provided by the TAGE predictor (confidence, number of providing banks) to index the SC predictoror. However experiments to incorporate use all this information flowing out from the 6 TAGE predictors to index the SC predictor were not satisfactory: too much information in practice.

Instead we combine the predictions, the confidences (in practice the prediction counter) of the 6 TAGE predictors through a neural-like predictor, thus providing a prediction P_{neural} and a 4-level confidence C_{neural} (i.e. a total of 3 bits). We also retain the P_{colt} prediction.

The 4 levels of confidence are determined as follows: $|sum < threshold/4|$, $|threshold/4 \geq sum < threshold/2|$, $|threshold/2 \geq sum < threshold|$, $|threshold \geq sum$

4 The Statistical Corrector

The statistical corrector used in our MTAGE-SC is very similar to the one previously used in [10] and [5]. It is a perceptron-inspired [1] Statistical Corrector [8, 9], that combines multiple components:

The components that have introduced over [5] underlined in *italics*

- Two Bias tables: indexed through the PC, the 4-bit output of TAGE predictions combiner and the TAGE predictions.
- 4 LGEHL components, 2 using a 16-entry history table, one using 1K-entry history table, and the last one 32K-entry history table. Each one features 15 tables.
- 4 perceptron-derived local history components using similar history tables. In these perceptron-derived components, we use the MAC representation of the counters[6]; a counter is associated with 6 consecutive bits of history. Each of these components features 10 tables.
- 2 perceptron-derived components using respectively global branch history and global path history: 10 tables each.
- a global history GEHL component: 209 tables
- a global history component inspired from the MAC-RHSP predictor [6]; a counter is associated with 6 consecutive bits of history and part of the global branch history (1/3) is hashed with the PC: 80 tables.
- Four path skeleton history GEHL components. The first path skeleton are the taken branches whose targets are not too close to the branch source. By too close, we mean 16 bytes for backward branches and 128 bytes for forward branches. *A second one record only the path of taken branches at condition that the target is more than 64 bytes away. A third one records the history of branches which target are not more than 64 bytes away.* The four path skeleton history registers the branch in the path only if it was not among the last 8 encountered branches. These components feature 15 tables each.
- Two path skeleton history perceptron-derived components: 10 tables each.
- *A few IMLI [13] inspired GEHL-components: IMLI counter + local history, IMLI counter + global history, history at constant IMLI, IMLI outer history (the IMLI-OH component in [13]).*
- *A LGEHL component, but not indexed through the PC*

All the tables hold 8 bit counters. The prediction is computed as the sign of the sum of the (centered) predictions read on all the Statistical Corrector tables: a total of more than 500 counters are summed.

The prediction is the sign of the sum. The Statistical Corrector predictor tables are updated using a dynamic threshold policy as suggested for the GEHL predictor [7]. As suggested in [2], we use a PC-indexed table of dynamic threshold, which yields marginal benefit.

Except for the Bias component, any of the components of the statistical has only a limited accuracy impact, but if one removes all the components exploiting local history or exploiting global branch/path history the impact on accuracy is more significant.

5 Final branch prediction computation

In practice, the prediction flowing out from the SC predictor is more accurate than the predictions flowing out from the TAGE predictions combiner. However, in the case where the output of the SC predictor is not high confidence, P_{neural} and/or P_{colt} are sometimes more accurate. Therefore the final branch prediction computation in the MTAGE-SC predictor uses the TAGE prediction combiner output (4 bits) and the SC prediction and confidence to index a table.

6 Conclusion

On the CBP-5 train traces, the submitted MTAGE-SC predictor achieves about 4.7 % fewer mispredictions than the poTAGE-SC predictor [5] that won the previous championship. Moreover it also outperforms it significantly on the CBP-4 traces.

The benefit comes from all the stages in the predictor; however the most significant benefit comes from the incorporation of the confidence level of intermediate predictions for feeding the next stage in the predictors.

References

[1] D.A. Jimenéz and C. Lin. Neural methods for dynamic branch prediction. *ACM Transactions on Computer Systems*, 20(4), November 2002.

[2] Daniel A. Jiménez. Oh-snap: Optimized hybrid scaled neural analog predictor. In *Proceedings of the 3rd Championship on Branch Prediction, http://www.jilp.org/jwac-2/*, 2011.

[3] G.H. Loh and D.S. Henry. Predicting conditional branches with fusion-based hybrid predictors. In *Proceedings of the 11th Conference on Parallel Architectures and Compilation Techniques*, 2002.

[4] Pierre Michaud. Five poTAGEs and a COLT for an unrealistic predictor. In *Proceedings of the 4th Championship Branch Prediction*, 2014.

[5] Pierre Michaud and André Seznec. Pushing the branch predictability limits with the multi-potage+sc predictor. In *Proceedings of the 4th Championship on Branch Prediction, http://www.jilp.org/cbp2014/*, 2014.

[6] A. Seznec. Revisiting the perceptron predictor. PI-1620, IRISA, May 2004.

[7] A. Seznec. Analysis of the O-GEHL branch predictor. In *Proceedings of the 32nd Annual International Symposium on Computer Architecture*, june 2005.

[8] André Seznec. A 64 kbytes ISL-TAGE branch predictor. In *Proceedings of the 3rd Championship Branch Prediction*, June 2011.

[9] André Seznec. A new case for the tage branch predictor. In *Proceedings of the MICR0 44*, 2011.

[10] André Seznec. Tage-sc-l branch predictors. In *Proceedings of the 4th Championship on Branch Prediction, http://www.jilp.org/cbp2014/*, 2014.

[11] André Seznec. Tage-sc-l branch predictors again. In *submitted to the 5th Championship on Branch Prediction*, 2016.

[12] André Seznec and Pierre Michaud. A case for (partially)-tagged geometric history length predictors. *Journal of Instruction Level Parallelism (http://www.jilp.org/vol8)*, April 2006.

[13] André Seznec, Joshua San Miguel, and Jorge Albericio. The inner most loop iteration counter: a new dimension in branch history. In *Proceedings of the 48th International Symposium on Microarchitecture, MICRO 2015, Waikiki, HI, USA, December 5-9, 2015*, pages 347–357, 2015.