An optimized 2bcgskew branch predictor

André Seznec
IRISA/INRIA
Campus de Beaulieu
35042 Rennes Cedex, France
seznec@irisa.fr

September 2003

In this research note, we briefly describe a few optimizations on the 2bcgskew branch predictor. When using these optimizations, the 2bcgskew predictor can handle very long global history (for instance 72 bits for a 1 Mbit predictor) without suffering from high aliasing.

These optimizations are included in the implementations of the simulator of 2bcgskew we are distributing.

The hybrid skewed predictor *2bcgskew* (Figure 1) was proposed in [4]. The Alpha EV8 branch predictor [3] was derived from this proposal. The original hybrid predictor *2bcgskew* combines an *e-gskew* predictor [2] and a bimodal predictor. *2bcgskew* consists of four 2-bit counters banks. Bank **BIM** is the bimodal predictor, but is also part of the *e-gskew* predictor. Banks **G0** and **G1** are two gshare-like predictors included in the *e-gskew* predictor. Bank **Meta** is the meta-predictor. Depending on **Meta**, the prediction is either the prediction coming out from **BIM** or the majority vote on the predictions coming out from **G0**, **G1** and **BIM**.

Optimizations were proposed for the design of the EV8 branch predictor [3]:

(a) different sizes for the distinct tables in the predictor,

(b) different prediction and hysteresis table sizes: prediction tables and hysteresis tables are accessed at different pipeline stages, and hence can be implemented as physically distinct tables and with distinct sizes,

(c) variable history lengths: the four logical tables in the *2bcgskew* predictor are accessed using four different history lengths, i.e., the BIM table can be indexed using a short history, G0 and G1 the two gshare-like tables are indexed with respectively a “long” and a “very long” history.

Further studies on the choice of history lengthes for *2bcgskew* lead us to propose a few other tradeoffs.

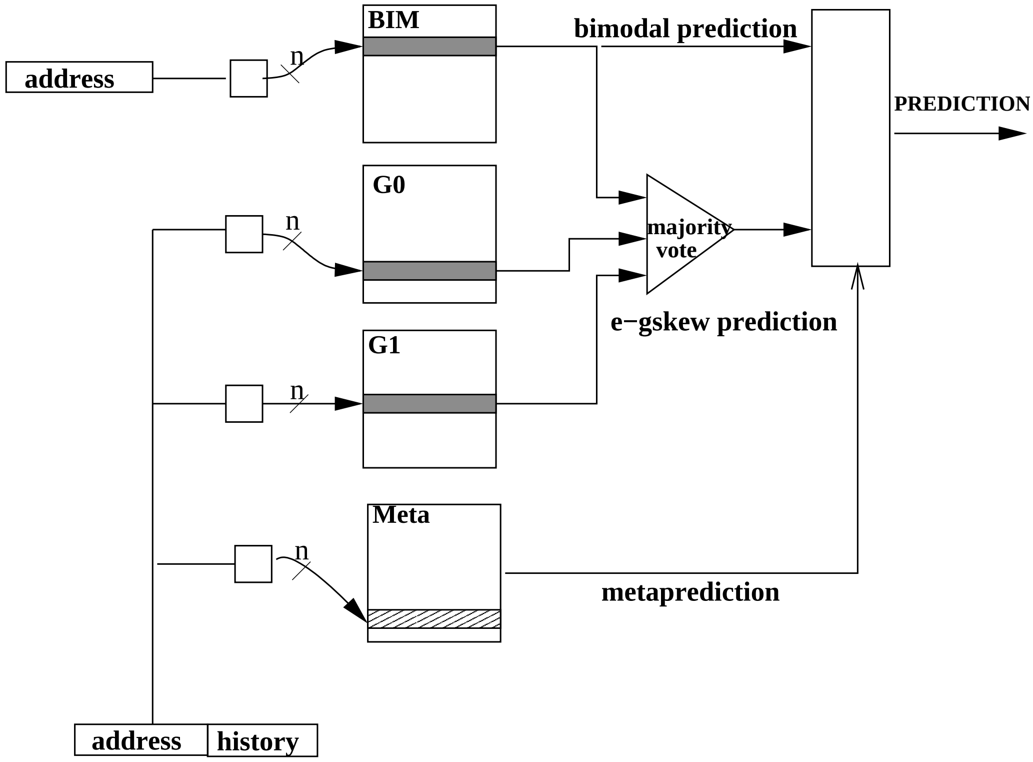

Figure 1. The *2bcgskew* **branch predictor**

(d) sharing physical tables between logical tables in the predictor. Normally a physical table is associated with a logical table in the predictor, but one can share the physical tables as follows: depending on the parity of branch address (or on a bit in history), table P0 (resp. P1) will be address with “long history” (resp. “very long history”) or “very long history” (resp. “long history”). 4 physical tables may even be shared among the 4 logical tables.

(e) Among the many experiments we run, we observed several applications which had the following: different initializations of the predictor were leading to completely different misprediction rates. For a non-demanding application, we observed a variation of a factor 7 in misprediction rate (from 0.25 % to 1.80 %). This was obviously associated with some ping-pong phenomenom in the predictor.

In order to break the ping-pong phenomenom, we introduce a degree of randomness in the update of the predictor as follows:
randomly, in one out of 32 mispredictions, if the majority vote and the bimodal component are agreeing then the three predictions from G0, G1 and BIM are randomly forced to weakly good prediction, if they are disageeing the meta predictor is forced to the correct component. With this introduction of this kind of randomness, the pathological behaviors associated with different initializations disappear.

The configurations that we propose in the distributed simulator were selected using set of traces from SPEC95, SPEC2000 and IBS [5].

To enable simple comparisons of branch predictor accuracy at equivalent storage budget, we have tried to define near optimal configurations for *2bcgskew* predictors with storage budgets proportional to 2^N bits.

The goal was not to find the "ultimate" best configuration for a particular set of traces, but rather to set an easy to implement (and simulate) set of rules leading to high accuracy on a very wide spectrum of predictor sizes ranging from 32Kbits to 8Mbits and a wide spectrum of applications.

The configurations we finally selected have the following characteristics for a total of 2^N bits of storage:

- 2^{N-1} bits are shared by GO and G1 prediction tables.
- 2^{N-2} bits are shared by BIM and Meta prediction tables.
- 2^{N-2} bits are shared by the four hysteresis tables.
- History lengths are respectively set to (N-11) for BIM and Meta, 4*(N-11) for the "long history" length and 8*(N-11) for the "very long history length.

One will note that the "very long history" is much longer than in most of the previously proposed global history branch predictors apart the perceptron predictor [1], 40 bits for a medium 64 Kbits predictor, 72 bits for a large 1 Mbits predictor.

Acknowledgement

This work was partially supported by an Intel research grant and an Intel research donation.

References

[1] D. Jimenez and C. Lin. Dynamic branch prediction with perceptrons. In *Proceedings of the Seventh International Symposium on High Perform ance Computer Architecture*, 2001.

[2] P. Michaud, A. Seznec, and R. Uhlig. Trading conflict and capacity aliasing in conditional branch predictors. In *Proceedings of the 24th Annual International Symposium on Computer Architecture (ISCA-97)*, June 1997.

[3] A. Seznec, S. Felix, V. Krishnan, and Y. Sazeidès. Design tradeoffs for the ev8 branch predictor. In *Proceedings of the 29th Annual International Symposium on Computer Architecture*, 2002.

[4] A. Seznec and P. Michaud. De-aliased hybrid branch predictors. Technical Report RR-3618, Inria, Feb 1999.

[5] R. Uhlig, D. Nagle, T. Mudge, S. Sechrest, and J. Emer. Instruction fetching: Coping with code bloat. In *Proc. of the 22nd Annual International Symposium on Computer Architecture (22nd ISCA'95) ACM SIGARCH Computer Architecture News*, June 1995.