Branch prediction: Jim, Yale, André, Daniel and the others

André Seznec
Daniel A. Jiménez
Title genuinely inspired by:

4 stars, but many other actors

Yeh, Pan, Evers, Young, McFarling, Michaud, Stark, Loh, Sprangle, Mudge, Kaeli, Skadron and many others
Prehistory

• As soon as one considers pipelining,
  ➢ branches are a performance issue

• I was told that IBM considered the problem as early as the late 50’s.
Jim

"Let us predict the branches"
History begins

• **Jim** Smith (1981) :
  - A study of branch prediction strategies

• Introduced:
  - Dynamic branch prediction
  - PC based prediction
  - 2-bits counter prediction

2bc prediction performs quite well
Yale

“let us use branch history”
By 1990, (very) efficient branch prediction became urgent

- Deep pipeline: 10 cycles
- Superscalar execution: 4 inst/cycle
- Out-of-Order execution
  - 50-100 instructions inflight considered possible

- Nowadays: much more!!
Two level history

- Tsu Yeh and Yale Patt 91:
  - Not just the 2-bit counters indexed by PC
  - But also the past:
    - Of this branch: local history
    - Of all branches: global history
      - global control flow path
global branch history
Yeh and Patt 91, Pan, So, Rameh 92

B1: if cond1
B2: if cond2
B3: if cond1 and cond2

B1 and B2 outputs determine B3 output
Look at the 3 last occurrences:

If all loop backs then loop exit
otherwise: loop back

for (i=0; i<100; i++)
for (j=0; j<4; j++)
loop body

• A local history per branch
• Table of counters indexed with PC + local history

Loop count is a particular form of local history
Nowadays most predictors exploit:

Global path/branch history

Some form of local history
Branch prediction: Hot research topic in the late 90’s

- McFarling 1993:
  - Gshare (hashing PC and history) + Hybrid predictors
- « Dealiased » predictors: reducing table conflicts impact
  - Bimode, e-gskew, Agree 1997

Essentially relied on 2-bit counters
Two level history predictors

• Generalized usage by the end of the 90’s

• Hybrid predictors (e.g. Alpha EV6).
A few other highly mentionable folks

- Marius Evers (from Yale’s group) showed
  - Power of hybrid predictors to fight aliasing, improve accuracy
  - Most branches predictable with just a few selected ghist bits
  - Potential of long global histories to improve accuracy

- Jared Stark (also Yale’s)
  - Variable length path BP: long histories, pipelined design
  - Implements these crazy things for Intel, laughs heartily when I ask him how it works

- Trevor Mudge could have his own section
  - Many contributions to mitigating aliasing
  - More good analysis of branch correlation
  - Cool analysis of branch prediction through compression
Daniel

"let us apply machine learning"
A UFO: The perceptron predictor
Jiménez and Lin 2001

signed 8-bit Integer weights

branch history as (-1,+1)

Update on mispredictions or if |SUM| < θ

Sign=prediction
(Initial) perceptron predictor

- Competitive accuracy
- High hardware complexity and latency
- Often better than classical predictors
- Intellectually challenging
Rapidly evolved to

Can combine predictions:
- global path/branch history
- local history
- multiple history lengths
- ...

4 out of 5 CBP-1 (2004) finalists based on perceptron, including the winner (Gao and Zhou)

Oracle, AMD, Samsung use perceptron (Zen 2 added TAGE)
Path-Based Perceptron (2003, 2005)

Path-based predictor reduces latency and improves accuracy. Turns out (2005) it also eliminates linear separability problem.
Scaled Neural Analog Predictor (2008)

Mixed-signal implementation allows weight scaling, power savings, very low latency
Multiperspective Perceptron Predictor (2016)

Traditional perceptron. Few perspectives: global and local history.

New idea: multiple perspectives: global/local plus many new features e.g. recency position, blurry path, André’s IMLI, modulo path, etc.etc.

Greatly improved accuracy. Can combine with TAGE. Work continues…
André

"let us use very long histories"
In the old world
EV8 predictor: *(derived from)* 2bc-gskew
Seznec et al, ISCA 2002 *(1999)*

Learnt that:
- Very long path correlation exists
- They can be captured
In the new world
An answer

• The geometric length predictors:
  
  ▪ GEHL and TAGE
The basis: A Multiple length global history predictor

With a limited number of tables
Underlying idea

- H and H’ two history vectors equal on N bits, but differ on bit N+1
  - e.g. L(1) ≤ N < L(2)
- Branches (A, H) and (A, H') biased in opposite directions

Table T2 should allow to discriminate between (A, H) and (A, H')
GEometric History Length predictor

The set of history lengths forms a geometric series

\[ L(0) = 0 \]
\[ L(i) = \alpha^{i-1} L(1) \]
\[ \{0, 2, 4, 8, 16, 32, 64, 128\} \]

What is important: \( L(i)-L(i-1) \) is drastically increasing

Spends most of the storage for short history !!
GEHL (2004)
prediction through an adder tree

Using the perceptron idea with geometric histories
TAGE (2006)
prediction through partial match

Tagless base predictor

prediction
The Geometric History Length Predictors

- Tree adder:
  - O-GEHL: Optimized GEometric History Length predictor
    - CBP-1, 2004, best practice award
- Partial match:
  - TAGE: TAgged GEometric history length predictor
    - Inspired from PPM-like, Michaud 2004
      + geometric length
      + optimized update policy
    - Basis of the CBP-2,-3,-4,-5 winners
GEHL (CBP-1, 2004)

- Perceptron-inspired
  - Eliminate the multiply-add
  - Geometric history length: 4 to 12 tables
  - *Dynamic threshold fitting*
    - Jiménez consider this the most important contribution to perceptron learning
  - 6-bit counters appears as a good trade-off
Doing better : TAGE

- Partial tag match
  - almost ..

- Geometric history length

- Very effective update policy
Altpred

Hit

Miss

Pred

Hit
TAGE update policy

Minimize the footprint of the prediction.

- Just update the longest history matching component
- Allocate at most one otherwise useless entry on a misprediction
TAGE vs OGEHL

Rule of thumb:

At equivalent storage budget
10% less misprediction on TAGE
Hybrid is nice
From CBP 2011,
« the Statistical Corrector targets »

- Branches with poor correlation with history:
  - Sometimes better predicted by a single wide PC indexed counter than by TAGE

- More generally, track cases such that:
  - « For this (PC, history, prediction, confidence), TAGE is likely (>50 %) to mispredict »

statistically
TAGE-GSC (CBP 2011)
(was named a posteriori in Micro 2015)

≈3-5% MPKI red.

Just a global hist neural predictor:
+ tables indexed with PC, TAGE pred. and confidence
TAGE-SC

- Micro 2011, CBP4, CBP5

Use any (relevant) source of information at the entry of the statistical correlator.

- Global history
- Local history
- IMLI counter (Micro 2015)

TAGE-SC = Multiperspective perceptron + TAGE
A BP research summary (CBP1 traces)

- 2bit counters 1981: 8.55 misp/KI
  - No real work before 1991: win 37 %

- Gshare 1993: 5.30 misp/KI
  - Hot topic, heroic efforts: win 28 %

  - The perceptron era, a few actors: win 25 %

- CBP-1 2004: 2.82 misp/KI
  - TAGE introduction: win 10 %

- TAGE 2006: 2.58 misp/KI
  - A hobby for AS and DJ: win 10 %

- TAGE-SC 2016: 2.36 misp/KI
Future of Branch Prediction research?

- See the limit study at CBP-5:
  - about 30% misp. gap
    512K ↔ unlimited
- New workloads are challenging
  - Server
  - Mobile
  - Web
  - These were in CBP-5, expected in CBP-6
- Need other new ideas to go further
  - Information source?
  - Some better way to extract correlation?
  - Deep learning?