

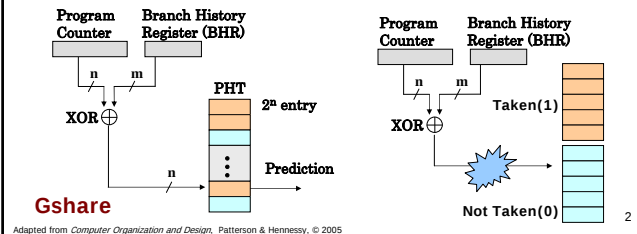
計算機アーキテクチャ特論 (Advanced Computer Architectures)

8. 分岐予測(2)

1

Gshare の改良

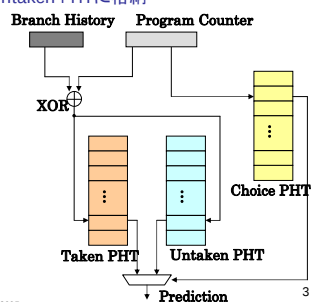
- PHTの競合が発生して性能が低下
- PCとBHRによって特定される予測(成立, 不成立)には偏りが存在するので、これらを別のテーブルに格納することで競合の悪影響を緩和
 - 分岐成立に偏っているもの
 - 分岐不成立に偏っているもの



2

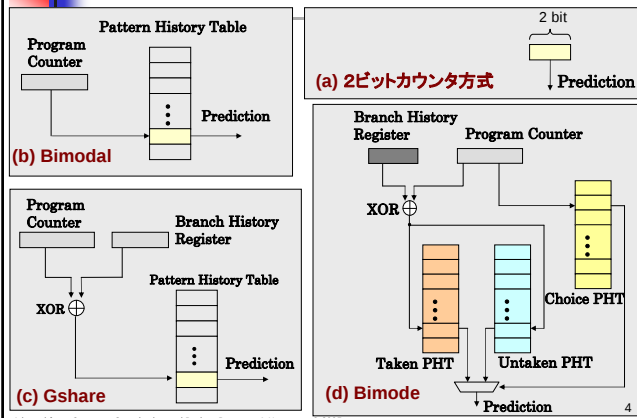
Bimode (MICRO 1997)

- 偏りを利用して競合の悪影響を緩和
 - 分岐成立に偏っているものをTaken PHTに格納
 - 分岐不成立に偏っているものをUntaken PHTに格納
- Choice PHT の内容で、どちらのテーブルを利用するか選択
- インデックスを工夫
 - Choice PHT は命令アドレス
 - Taken PHT, Untaken PHT は命令アドレスと分岐履歴



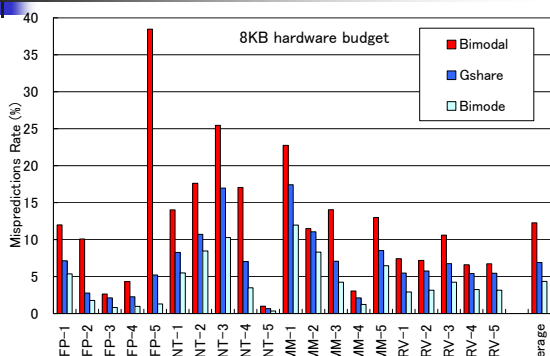
3

ここまでの分岐予測器



4

予測精度(予測ミス率)

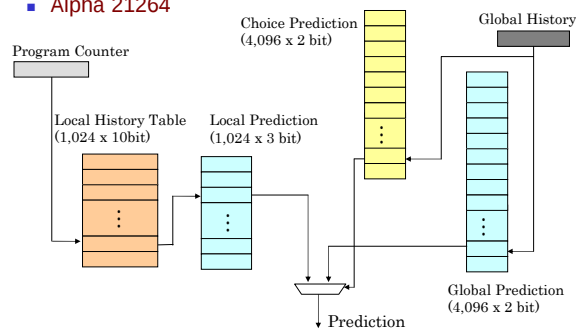


Benchmark for CBP(2004) by Intel MRL and IEEE TC uARCH.
Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

5

ハイブリッド方式 Alpha 21264

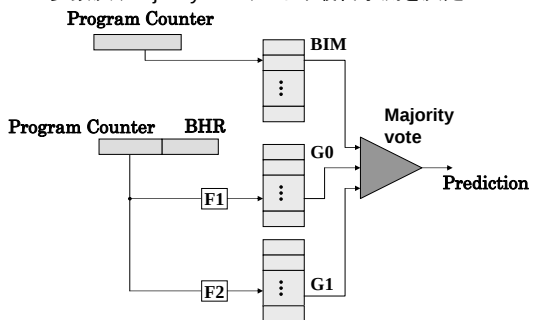
- 複数の分岐予測の利点
- Alpha 21264



6

ハイブリッド方式 E-gskew (ISCA 1997)

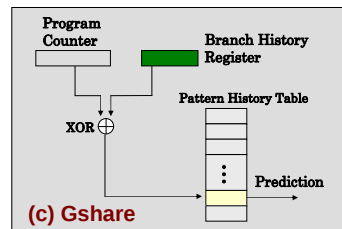
- 多数決 (Majority vote) により最終予測を決定



Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

ハードウェア量と分岐履歴の長さ

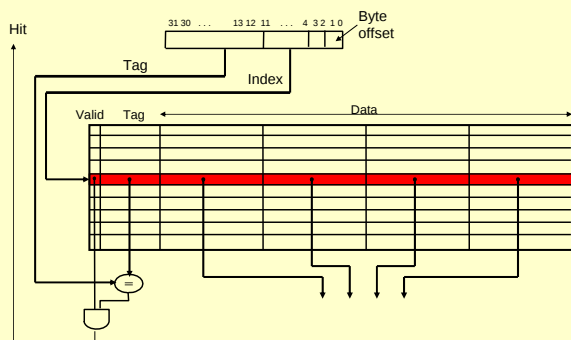
- パターン履歴表のハードウェアが支配的
- 2ビットカウンタのNエントリのテーブル
 - $N \times 2 \text{ bit}$ (例えば, 32Kエントリで8KB)
- 分岐履歴レジスタのとうる最大長を $n \text{ bit}$ とすると
 - $2^n \times 2 \text{ bit}$ (例えば, $n = 15$ で8KB)



Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

命令キャッシュの実装

- ラインサイズ 4ワード (16 Byte)



Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

YAGS (Yet Another Global Scheme)

Figure 3.
Bi-Mode

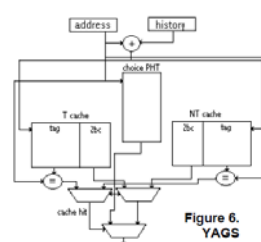
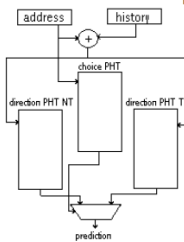


Figure 6
YAGS

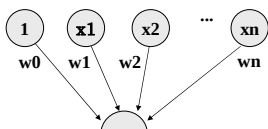
From YAGS paper

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

パーセプトロン

- ## ■ パーセプロロンモデル

- x_1 から x_n までの n ビットの分岐履歴を入力とする。
- y を計算する。
 - w は符号付きの整数で表現
- y の値がある閾値より高い場合に成立と予測する。



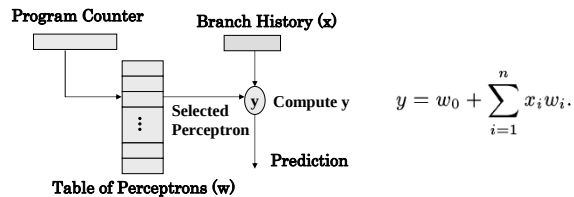
Perceptron Model

$$y = w_0 + \sum_{i=1}^n x_i w_i.$$

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

Perceptron (HPCA 2001)

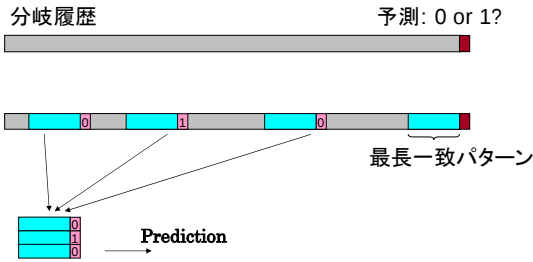
- 分岐アドレスによりパーセプトロンを選択
- グローバル分岐履歴を用いて予測
 - 8KBのハードウェア量で、34ビットの分岐履歴
- 分岐ミス、 $|y| < \theta$ の時にトレーニング



Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

12

パターンマッチング



- 過去の履歴との最長一致のパターンを見つける。
- それぞれの最長一致パターンの次に出現する0と1の回数を数え、多い方を予測値とする。

13

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

パターンマッチングの可能性

マッチングの対象となる分岐履歴の長さ N



$N = 1024 \times 1024$ (one million) に設定

$M = 16, 32, 64, 128, 256, 512, 1024$ に設定

M	16	32	64	128	256	512	1024
SpecFP	5.32	2.34	1.59	0.98	0.54	0.46	0.43
SpecINT	9.92	4.13	2.75	2.39	2.30	2.26	2.16
MultiMedia	8.90	6.51	5.43	5.09	4.99	4.94	4.93
Server	9.20	3.91	2.25	1.91	1.78	1.75	1.73
Average	8.34	4.22	3.00	2.59	2.40	2.35	2.31

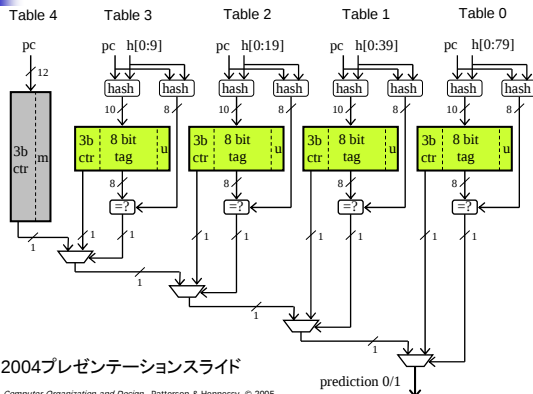
- テーブルのエントリ数を 224 と非常に大きく設定した Gshare の予測ミス率は 4.71

吉瀬, 岩田: 分岐予測の精度と履歴情報との関係について, 信学会ソサイエティ大会, 2005

14

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

Partial Pattern Matching (CBP 2004)

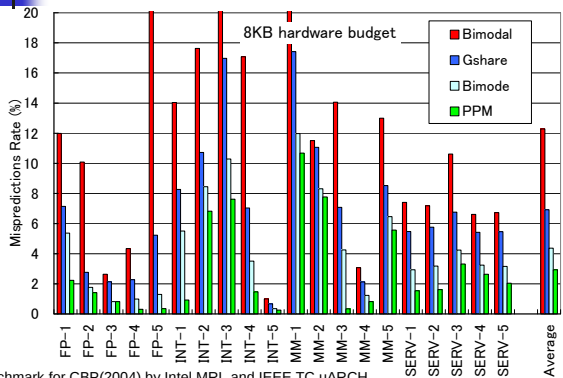


CBP2004プレゼンテーションスライド

15

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

予測精度 (予測ミス率)

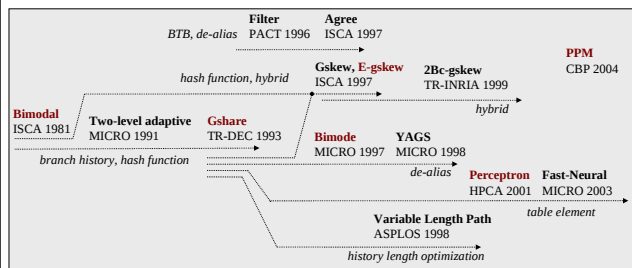


16

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

分岐予測のまとめ

- 重要な分岐予測の根底にあるアイデアとその構成を解説



ISCA (International Symposium on Computer Architecture)
MICRO (International Symposium on Microarchitecture)
PACT (International Conference on Parallel Architectures and Compilation Techniques)
ASPLOS (International Conference on Architectural Support for Programming Languages and Operating Systems)

17

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

分岐予測のまとめ

- Bimodal
 - 2ビットカウンタの配列, プログラムカウンタ
- Gshare
 - グローバル分岐履歴の利用
- Bimode
 - 偏りを用いて表を分離, メタ予測器
- ハイブリッド方式
 - 複数の予測方式の利用, 多数決
- パーセプトロン
 - パーセプトロンモデル, 長い分岐履歴の利用
- パターンマッチング (PPM)
 - 最長一致パターンの利用

18

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

分岐予測に関する今後の展望

- 新しい予測方式の開発
 - 予測精度の向上
- 分岐予測の精度の計算機性能
 - 投機的な更新
- パーセプトロンの実現
 - 高い予測精度, 複雑なハードウェア
- 分岐予測の精度と計算機アーキテクチャ
- 他の予測への適用

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

19

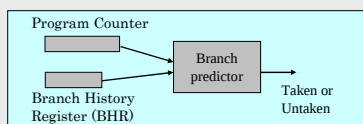
Prophet-Critic Hybrid Branch Prediction

Ayose Falcon, UPC, Jared Stark, Intel, Alex Ramirez, UPC, Konrad Lai, Intel, Mateo Valero, **UPC**
ISCA-31 pp. 250-261 (2004)

20

研究の目的と概要

- 今までにない, 斬新なアイデアの分岐予測器を提案
- 分岐予測器の予測精度の向上, プロセッサの性能向上を目指す.
- 予測ミス率を大幅に低減する.
 - Gccの場合には, 予測ミス率を 3.11% から 1.23%に削減
- Intel Pentium 4 プロセッサをベースとして, uPC (Uops Per Cycle) で5.2%の性能向上 (Perceptron分岐予測の場合).



従来の分岐予測器の枠組み

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

21

Prophet/critic hybrid の構造

- **Prophet:** 予言者
- **Critic:** 批判家, 鑑定家

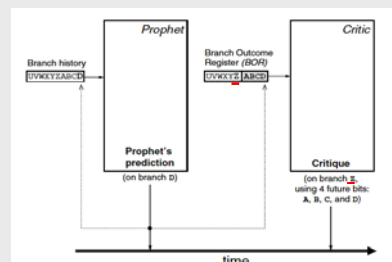
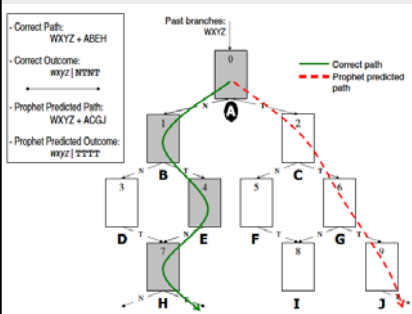


Figure 1. Structure of a prophet/critic hybrid.

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

22

Prophet/critic hybrid の予測の例

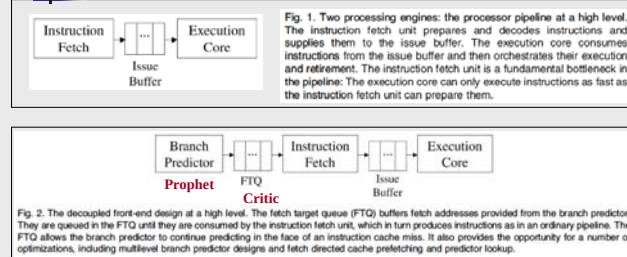


- 基本ブロック 0, 1, 4, 7 が正しい制御の流れとする.
- 分岐 A の予測を考える.
- Branch History: W, X, Y, Z
- Branch Future: A, C, G and J
- 過去の履歴において, W, X, Y, Z | A=T, C=T, G=T, J=T の場合には, 分岐予測ミスが多いことを記録し, これを用いて鑑定する.

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

23

Decoupled front-end architecture, ISCA-1999

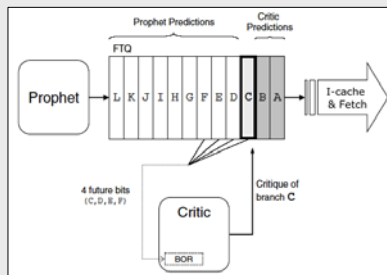


Optimizations Enabled by a Decoupled Front-End Architecture

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005

24

Decoupled front-end architecture を利用した Prophet/critic hybrid の実装



- Prophet, Critic それぞれの予測機構には、従来の分岐予測機構の任意の組み合わせを利用できる。

25

評価環境

- 現実的な、サイクル当たり6命令を処理する高性能マイクロプロセッサ。

Processor Frequency	3.8 GHz
Fetch/Issue/Retire Width	6 uops
Branch Mispredict Penalty	30 cycles
BTB	4096 entries, 4-way
FTQ Size	32 entries
Instruction Window Size	2048 uops
Scheduling Window Size	256 int, 128 mem, 384 fp (sizes in uops)
Load/Store Buffer Sizes	768 load, 512 store (sizes in uops)
Functional Units	6 int, 4 mem, 2 fp
Hardware Data Prefetcher	Stream-based (16 streams)
Instruction Cache	64 KB, 8-way, 64-byte line
L1 Data Cache	32 KB, 16-way, 64-byte line, 3 cycle hit
L2 Unified Cache	2 MB, 16-way, 64-byte line, 16 cycle hit
Memory Latency	100 ns

Table 2. Simulation parameters.

26

Future bit の重要性、何回先の予測結果まで利用すればよいか？

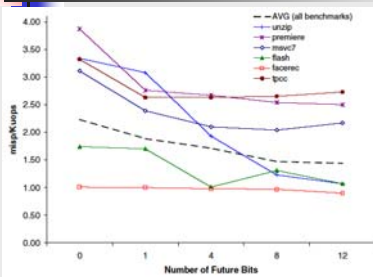
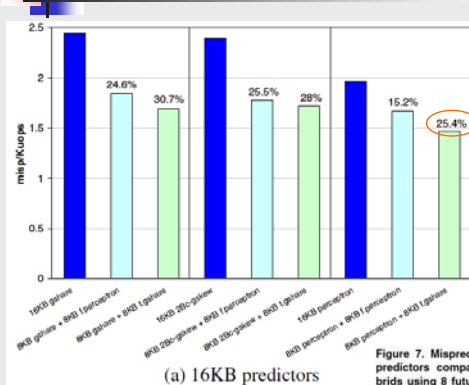


Figure 5. Effect of varying the number of future bits used by the critic on prediction accuracy for selected benchmarks. (prophet: 8KB perceptron; critic: 8KB tagged gshare)

- Future bits = 0 の構成は従来のハイブリッド分岐予測器
- Future bits = 1 でも、予測精度が向上する。
- 平均では、Future bits の数を増やすことで予測精度が向上する。
- ただし、大きくしすぎること精度が低下するベンチマークが存在する。
- 8ビット前後で、最適なものを用いる。

27

予測精度の比較 (16KBのハードウェア資源)



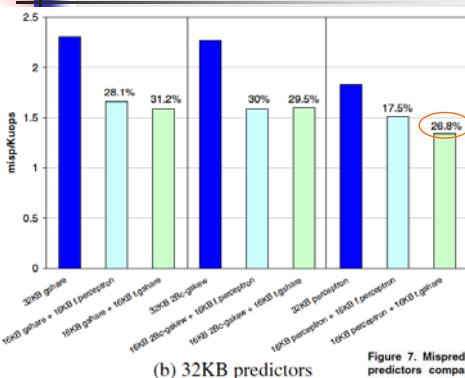
- 単体では perceptron がベスト。
- 更に、提案手法により 25.4% の予測ミスを削減

(a) 16KB predictors

Figure 7. Mispredict rates of conventional predictors compared to prophet/critic hybrids using 8 future bits. Numbers indicate percent reduction in mispredict rate.

28

予測精度の比較 (32KBのハードウェア資源)

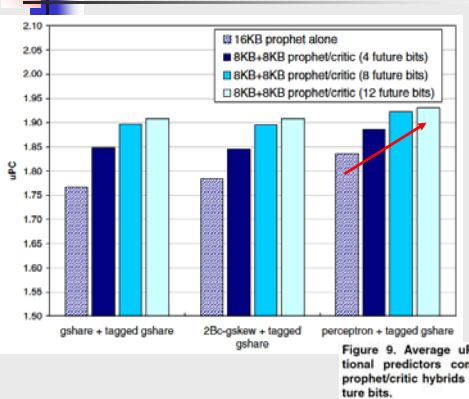


(b) 32KB predictors

Figure 7. Mispredict rates of conventional predictors compared to prophet/critic hybrids using 8 future bits. Numbers indicate percent reduction in mispredict rate.

29

プロセッサの性能向上の比較



- Perceptron がベスト。
- Perceptron, 12 future bit の場合には、従来手法と比較して、5.2% の性能向上

Figure 9. Average uPC of 16KB conventional predictors compared to 8KB+8KB prophet/critic hybrids using 4, 8, and 12 future bits.

30



Prophet-Critic Branch Prediction まとめとコメント

- 今までにない, 斬新なアイデアの分岐予測の提案と評価.
- 予測ミス率を大幅に低減する.
 - Gccの場合には, 予測ミス率を 3.11% から 1.23%に削減することができる.
- Intel Pentium 4 プロセッサをベースとして, uPC (Uops Per Cycle) で 5.2% の性能向上 (Perceptron分岐予測の場合).
- コメント
 - アイデアとして面白い.
 - ハードウェアの規模と複雑さも問題にならないレベル.
 - アイデアは分岐予測以外の分野でも利用できるのではないか.

31

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005



アナウンス

- 講義スライド, 講義スケジュール
 - www.arch.cs.titech.ac.jp

32

Adapted from *Computer Organization and Design*, Patterson & Hennessy, © 2005